THÈSE POUR OBTENIR LE GRADE DE DOCTEUR DE L'UNIVERSITÉ DE MONTPELLIER

En Biostatistiques

École doctorale: Information, Structures, Systèmes

Unité de recherche : Mathématiques, Informatique et Statistique pour l'Environnement et l'Agronomie

Titre de la thèse

UNSUPERVISED ACTIVE LEARNING

Présentée par Victor THUOT

Le [XX mois année]

Sous la direction de Nicolas Verzelen et Alexandra Carpentier

Devant le jury composé de

[Prénom NOM, titre, établissement]	[Statut jury]
[Prénom NOM, titre, établissement]	[Statut jury]
[Prénom NOM, titre, établissement]	[Statut jury]
[Prénom NOM, titre, établissement]	[Statut jury]
[Prénom NOM, titre, établissement]	[Statut jury]
[Prénom NOM, titre, établissement]	[Statut jury]
[Prénom NOM, titre, établissement]	[Statut jury]
[Prénom NOM, titre, établissement]	[Statut jury]

UNIVERSITÉ DE MONTPELLIER

Table of Contents

Chapter 1

Introduction (version française)

1.1 Contexte et motivations

Un objectif central en statistique moderne et en apprentissage automatique est de retrouver une structure cachée à partir de données. Alors que l'apprentissage supervisé entraîne des modèles à partir de données annotées, l'apprentissage non supervisé vise à inférer une structure latente sans accès aux annotations (Berry et al., 2020). Dans les formulations classiques, l'apprentissage est typiquement batch : l'apprenante (désignée au féminin dans toute la suite) reçoit un jeu de données collecté à l'avance, en un bloc, et infère la structure cible à partir d'observations passives.

Cependant, dans de nombreuses applications modernes, les données sont collectées de manière séquentielle et adaptative, par exemple dans les systèmes de recommandation (Zhou et al., 2012; Lu et al., 2015), l'expérimentation en ligne comme les essais cliniques adaptatifs (Chow and Chang, 2008), ou l'apprentissage sur des plateformes collaboratives *crowdsourcing* (Raykar et al., 2010; Ariu et al., 2024). Dans ces contextes, où l'apprenante peut adapter sa stratégie d'échantillonnage au fur et à mesure de la collecte, la ressource clé est le budget d'échantillonnage, c'est-à-dire le nombre d'observations nécessaires pour retrouver la structure cible de manière fiable. Cela motive naturellement une perspective de bandit en exploration pure (Lattimore and Szepesvári, 2020), dans laquelle l'apprenante alloue les échantillons de manière adaptative pour identifier une structure inconnue avec une garantie d'erreur contrôlée.

L'apprentissage actif peut améliorer fortement l'efficacité par rapport à l'échantillonnage passif, en concentrant les observations sur les parties informatives du problème. Mais l'adaptation a aussi un coût : avant d'exploiter des requêtes informatives, l'apprenante doit d'abord découvrir où se trouve cette information. Cette thèse étudie plusieurs problèmes d'exploration pure sous cet angle, en se concentrant sur le compromis entre le coût d'identification de requêtes informatives et le coût de reconstruction de la structure latente. Pour traiter les régimes de grande dimension et de grande échelle, de plus en plus courants en pratique (Giraud, 2021; Rigollet and Hütter, 2023), nous développons une analyse non asymptotique qui capture à la fois les bénéfices et les limites intrinsèques de l'échantillonnage adaptatif pour l'apprentissage non supervisé.

Plan de l'introduction. Cette introduction est organisée comme suit. La Section 1.2 présente un cadre général qui couvre l'ensemble des problèmes étudiés dans cette thèse. La Section 1.3 énonce nos questions directrices. Nous concluons par un résumé des contributions, chapitre par chapitre (Section 1.4), puis par une présentation de la structure de la thèse (Section 1.5). Remarquez que l'introduction en anglais (Chapitre 2) est une version plus détaillée de cette introduction : elle contient notamment une revue de littérature détaillée en Section 2.3 et un exemple motivant de détection de signal en Section 2.5.

1.2 Apprentissage actif non supervisé

1.2.1 Exploration pure dans des environnements de bandits matriciels

Modèle de structure latente. Nous étudions des problèmes dans lesquels une apprenante vise à retrouver une structure inconnue $\mathcal{C}^*_{\boldsymbol{M}}$ encodée par un modèle matriciel latent $\boldsymbol{M} \in \mathbb{R}^{n\times d}$.

Considérons n objets indexés par $[n] = \{1, \dots, n\}$ et d attributs indexés par $[d] = \{1, \dots, d\}$. Chaque objet i possède un vecteur latent d'attributs $M_{i,\cdot} \in \mathbb{R}^d$, formant les lignes d'une matrice inconnue $\boldsymbol{M} \in \mathbb{R}^{n\times d}$. La structure inconnue $\mathcal{C}^*_{\boldsymbol{M}}$ est définie comme une fonction de $\boldsymbol{M}$. L'objectif de l'apprenante est de retrouver exactement $\mathcal{C}^*_{\boldsymbol{M}}$, ou à une précision donnée.

Le problème classique d'identification du meilleur bras en bandits (Best Arm Identification, BAI) peut être vu comme un cas particulier, où $\mathcal{C}^*_{\boldsymbol{M}}$ est l'indice de l'entrée maximale de $\boldsymbol{M}$ (Audibert and Bubeck, 2010). Plus généralement, la structure latente peut prendre différentes formes selon le problème : $\mathcal{C}^*_{\boldsymbol{M}}$ est une partition en clustering (Kaufman and Rousseeuw, 2009), un graphe en détection de communautés (Jin et al., 2021), une permutation en *ranking* (Saad et al., 2023), ou un sous-ensemble d'entrées en détection de points de rupture (Niu et al., 2016). Dans l'identification du vainqueur de Condorcet, $\mathcal{C}^*_{\boldsymbol{M}}$ est l'indice de la ligne qui domine toutes les autres dans les comparaisons deux à deux (Bengs et al., 2021). Dans cette thèse, nous étudions plusieurs problèmes relevant de ce cadre. Les Chapitres 3 à 5 portent sur le clustering, où $\mathcal{C}^*_{\boldsymbol{M}}$ est une partition de $[n]$ en K clusters. Le Chapitre 6 traite de l'identification du vainqueur de Condorcet, où $\mathcal{C}^*_{\boldsymbol{M}}$ est l'indice de ce vainqueur, c'est-à-dire l'objet préféré comparativement à tous les autres. Le Chapitre 7 traite de l'identification de multiples points de rupture, où $\mathcal{C}^*_{\boldsymbol{M}}$ est l'ensemble des positions de rupture de la distribution sous-jacente.

Protocole d'apprentissage séquentiel et adaptatif. Dans l'apprentissage non supervisé classique que l'on nomme ici "batch", l'apprenante reçoit un ensemble fixe d'échantillons indépendants collectés a priori. Ici, au contraire, elle collecte des observations séquentiellement et adaptativement : à chaque étape, elle décide quelles entrées observer en fonction des observations passées. La matrice $\boldsymbol{M}$ est inconnue de l'apprenante, qui y accède en interagissant avec un environnement de type bandit (Lattimore and Szepesvári, 2020).

Nous considérons un environnement de bandit ν qui renvoie des observations bruitées des entrées de $\boldsymbol{M}$. À chaque tour, l'apprenante sélectionne des entrées de $\boldsymbol{M}$, reçoit des observations bruitées des entrées sélectionnées, puis, après $\mathcal{T}$ requêtes, produit une estimation $\hat{\mathcal{C}}$. Nous considérons le cadre fixed-confidence : étant donné un niveau de confiance prescrit $\delta \in (0, 1)$, l'objectif est d'assurer une reconstruction exacte, c'est-à-dire $\hat{\mathcal{C}} = \mathcal{C}^*_{\boldsymbol{M}}$, avec probabilité au moins $1 - \delta$ (ou une reconstruction exacte à précision fixée au Chapitre 7).

Au temps t, elle sélectionne un objet $I_t \in [n]$ et un sous-ensemble d'attributs $J_t \subset [d]$, puis reçoit $X_t = (X_{t,j})_{j\in J_t}$. Nous nous concentrons sur deux cas extrêmes : (i) observation de tous les attributs ($J_t = [d]$), où elle observe tous les attributs de l'objet sélectionné ; (ii) sélection des attributs ($|J_t| = 1$), où elle n'observe qu'un seul attribut. Des régimes intermédiaires ont également été étudiés (Ariu et al., 2024), nous ne les considérons pas ici. Conditionnellement au choix $I_t = i$ et $J_t = J$, X_t est tiré selon une distribution inconnue $\nu_{i,J}$ de moyenne $\mathbb{E}[X_t] = (M_{i,j})_{j\in J}$. Le

bruit $\varepsilon_t = (X_{t,j} - M_{I_t,j})_{j \in J_t}$ est indépendant des observations passées. Nous notons $\nu_i = \nu_{i,[d]}$ la loi d'une observation de ligne complète de l'objet i, et $\nu_{i,j} = \nu_{i,\{j\}}$ la loi d'une observation d'une unique entrée (i, j). Après $\mathcal{T} \in \mathbb{N}$ tours, l'apprenante produit $\hat{\mathcal{C}}$ pour estimer $\mathcal{C}^*_{\boldsymbol{M}}$. Le temps d'arrêt $\mathcal{T}$ est une variable aléatoire choisie de manière adaptative.

Étant donné $\delta \in (0, 1)$, l'apprenante vise à retrouver exactement $\mathcal{C}^*_{\boldsymbol{M}}$ avec probabilité au moins $1 - \delta$. Une stratégie π est δ-correcte si $\mathbb{P}_{\pi,\nu}(\hat{\mathcal{C}} = \mathcal{C}^*_{\boldsymbol{M}}) \geq 1 - \delta$. Son budget d'échantillonnage $\mathcal{T}_\pi(\nu, \delta)$ est le nombre (aléatoire) de données collectées pendant l'exploration, qui sert de mesure de performance.

Une stratégie d'exploration pure π comporte trois composantes (Kaufmann et al., 2016) : (i) une règle d'échantillonnage qui choisit (I_t, J_t) selon l'historique ; (ii) une règle d'arrêt qui décide quand s'arrêter ; (iii) une règle de recommandation qui produit $\hat{\mathcal{C}}$ à partir des échantillons. Ces trois composantes peuvent être aléatoires ; $\mathbb{P}_{\pi,\nu}$ désigne la loi induite.

Budget d'échantillonnage comme mesure de performance. Dans de nombreuses applications modernes, l'acquisition d'échantillons est coûteuse ; le budget d'échantillonnage est donc une ressource à économiser (Bubeck et al., 2009). L'apprentissage actif peut procurer des gains substantiels en concentrant l'effort sur les entrées les plus informatives. L'objectif est donc de minimiser $\mathcal{T}_\pi(\nu, \delta)$ tout en garantissant la δ-correction. Nous nous intéressons en particulier à la caractérisation du budget d'échantillonnage optimal, défini comme le budget minimal requis par n'importe quelle stratégie pour retrouver la structure inconnue avec probabilité au moins $1 - \delta$. L'obtention de bornes supérieures et inférieures fines de cette quantité est une question centrale de cette thèse.

Notre analyse est non asymptotique : aucun des paramètres $(n, d, \delta$, ainsi que les quantités spécifiques au problème telles que le nombre de clusters $K)$ n'est supposée dominer asymptotiquement les autres. Ce point de vue capture les bénéfices de l'échantillonnage séquentiel et adaptatif dans différents régimes et met en évidence les situations où l'adaptation devient coûteuse, en particulier en grande dimension (grand d) ou à grande échelle (grand n). Nous utilisons $\mathcal{T}_\pi(\nu, \delta)$ comme métrique principale de performance. En particulier, nous étudions le budget moyen $\mathbb{E}_{\pi,\nu}[\mathcal{T}_\pi(\nu, \delta)]$ et des garanties en quantiles de la forme $\mathbb{P}_{\pi,\nu}(\mathcal{T}_\pi(\nu, \delta) \leq T) \geq 1 - \delta$.

Le Tableau 1.1 donne une vue unifiée de tous les problèmes étudiés dans cette thèse. Nous décrivons maintenant chacun d'eux plus en détail.

1.2.2 Clustering actif (CBP)

Dans le problème de clustering actif (Clustering with Bandit Feedback, CBP) (Yang et al., 2024; Ariu et al., 2024; Yavas et al., 2025), la structure latente $\mathcal{C}^*_{\boldsymbol{M}}$ est une partition de $[n]$ en K clusters : $\mathcal{C}^*_{\boldsymbol{M}} = \{\mathcal{C}^*_1, \ldots, \mathcal{C}^*_K\}$, avec $\mathcal{C}^*_k \subset [n]$ et $\bigcup_k \mathcal{C}^*_k = [n]$. Deux objets $i, i' \in [n]$ appartiennent au même cluster si et seulement s'ils partagent les mêmes distributions d'attributs $\{\nu_{i,j}\}_{j \in [d]}$. L'objectif de l'apprenante est de retrouver exactement la partition $\mathcal{C}^*_{\boldsymbol{M}}$. Nous étudions trois variantes du CBP, qui diffèrent par le modèle d'observation et les hypothèses sur la distribution du bruit d'échantillonnage.

Chapitre 3 (CBP paramétrique). L'apprenante choisit adaptativement un objet et reçoit

une évaluation bruitée de tous ses attributs ($J_t = [d]$). Chaque ligne $M_{i,\cdot} \in \mathbb{R}^d$ est la moyenne de ν_i. Elle choisit $I_t \in [n]$ et observe $X_t \sim \nu_{I_t}$ avec $\mathbb{E}[X_t] = M_{I_t,\cdot}$. Le bruit est σ-sous-gaussien ($\sigma > 0$ connu), ce qui couvre les cadres gaussiens et bornés. Introduit dans Yang et al. (2024), ce cadre généralise le clustering classique dans un cadre séquentiel et adaptatif.

Chapitre 4 (CBP non paramétrique). Dans le chapitre 4, nous étudions une version non paramétrique du CBP, où l'objectif est un clustering distributionnel (Dhillon et al., 2004). Comme dans le cas paramétrique, l'apprenante choisit adaptativement un objet et reçoit une évaluation bruitée de tous ses attributs ($J_t = [d]$). Cependant, aucune hypothèse paramétrique n'est imposée sur les distributions ν_i. Nous supposons que les distributions sont supportées sur un espace topologique séparable $\mathcal{X}$ (potentiellement de dimension infinie), de sorte que le modèle de bruit est arbitraire. Nous abordons ce cadre général à l'aide de méthodes à noyau.

Chapitre 5 (CBP avec sélection d'attributs). Au chapitre 5, nous étudions une variante du CBP paramétrique avec sélection adaptative d'attributs, proche de Ariu et al. (2024). Dans ce protocole entièrement adaptatif, l'apprenante sélectionne à chaque tour à la fois un objet et un seul attribut ($|J_t| = 1$). Au temps t, elle choisit $I_t \in [n]$ et $J_t \in [d]$, puis observe $X_t \sim \nu_{I_t,J_t}$, une évaluation bruitée de M_{I_t,J_t} avec bruit σ-sous-gaussien. Nous nous concentrons sur le cas à deux clusters ($K = 2$).

1.2.3 Identification du Vainqueur de Condorcet (CWI)

Dans le Chapitre 6, nous étudions le problème d'Identification du Vainqueur de Condorcet (Condorcet Winner Identification, CWI) dans des *dueling bandits* (Bengs et al., 2021). Nous considérons K objets $[K]$, avec une matrice d'environnement $\boldsymbol{Q} \in [0,1]^{K \times K}$. Dans ce cadre, les rôles des objets et des attributs sont symétriques, donc $n = d = K$.

Pour une paire d'objets (i, j), la quantité $Q_{i,j}$ est la probabilité que l'objet i soit préféré à l'objet j. Au temps t, l'apprenante choisit $(I_t, J_t) \in [K]^2$ et observe $X_t \sim \text{Bernoulli}(Q_{I_t,J_t})$. Par construction, $\boldsymbol{Q}$ vérifie la relation $Q_{i,j} = 1 - Q_{j,i}$. Cela correspond à des situations où l'apprenante n'accède à l'environnement qu'au travers de comparaisons entre pairs d'objets. La matrice des écarts est définie par $\Delta_{i,j} := Q_{i,j} - \frac{1}{2}$, donc $\Delta_{i,j} = -\Delta_{j,i}$ (anti-symétrie). La structure inconnue à retrouver est le vainqueur de Condorcet (Condorcet Winner, CW) : un bras $i^* \in [K]$ tel que $\Delta_{i^*,j} > 0$ pour tout $j \neq i^*$, ce qui signifie qu'il gagne en espérance contre tous les autres bras. L'objectif de l'apprenante est d'identifier i^* avec grande probabilité. On peut voir cela comme une généralisation de l'identification du meilleur bras (BAI) à un cadre de *dueling bandit*.

1.2.4 Identification de Points de Rupture Multiples (MCP)

Dans le Chapitre 7, nous étudions le problème de détection de rupture (Multiple Change Point Identification, MCP), dans une formulation proche de Lazzaro and Pike-Burke (2025b). Nous considérons un espace continu d'objets $[0, 1]$. L'environnement est caractérisé par une fonction $f : [0, 1] \to \mathbb{R}$, constante par morceaux, avec m points de rupture en des positions inconnues $0 < x_1^* < \cdots < x_m^* < 1$, les points de rupture étant définis comme les positions où la fonction change de valeur. Ainsi, f est constante sur chaque intervalle $(x_{i-1}^*, x_i^*]$, avec $x_0^* = 0$ et $x_{m+1}^* = 1$,

et présente des discontinuités aux points de rupture. L'apprenante choisit séquentiellement des points $I_t \in [0,1]$ et observe $X_t = f(I_t) + \varepsilon_t$, où ε_t est un bruit σ-sous-gaussien. La structure spatiale de $[0,1]$ est une caractéristique essentielle qui distingue ce problème du CBP.

Dans ce modèle continu, la localisation exacte des points de rupture est impossible. À la place, l'apprenante cherche à localiser les points de rupture avec une précision donnée $\eta > 0$, et une probabilité d'erreur $\delta \in (0,1)$. L'objectif est de construire m estimateurs $\hat{x}_1, \ldots, \hat{x}_m$ tels que $\mathbb{P}\left(\max_{i\in[m]} |\hat{x}_i - x_i^*| \leq \eta\right) \geq 1 - \delta$. [1]

Table 1.1 – Résumé des différents problèmes et cadres étudiés dans cette thèse. Ici, CBP désigne le Clustering actif, CWI l'Identification du Vainqueur de Condorcet, et MCP l'Identification de Points de Rupture Multiples. Pour chaque problème, nous précisons l'ensemble d'objets, l'espace des attributs, la structure inconnue, le schéma d'observation, le modèle de bruit et l'objectif.

	Problème	**Objets**	**Attributs**	**Structure inconnue**	**Observation / tour**	**Bruit**	**Objectif**
3	CBP paramétrique	$[n]$	$\mathbb{R}^d$ $(d < \infty)$	K clusters	un objet, tous les attributs	σ-sous-gaussien	Confiance fixée
4	CBP non paramétrique	$[n]$	$\mathcal{X}$ $(d \leqslant \infty)$	K clusters	un objet, tous les attributs	arbitraire	Confiance fixée
5	CBP avec sélection d'attributs	$[n]$	$\mathbb{R}^d$ $(d < \infty)$	K clusters avec $K = 2$	un objet, un attribut	σ-sous-gaussien	Confiance fixée
6	CWI	$[K]$	/	Vainqueur de Condorcet	comparaisons par paires	Bernoulli	Confiance fixée & budget fixé
7	MCP	$[0,1]$ (continu)	$\mathbb{R}$ $(d = 1)$	positions des points de rupture	évaluation de f en un point	σ-sous-gaussien	Confiance fixée

1. En introduction, nous nous concentrons sur le problème de localisation de l'intégralité des points de rupture. Dans le chapitre 7, nous considérons un cadre plus général où l'apprenant doit ne produire qu'un sous-ensemble de $N \leq m$ points de rupture.

1.3 Questions directrices

Nous sommes motivés par les questions suivantes, qui guident les contributions de cette thèse.

Question 1. *Comment retrouver efficacement une structure inconnue dans des problèmes de bandits matriciels ?*

Une part centrale de cette thèse est consacrée à la conception d'algorithmes de reconstruction de structure dans des problèmes matriciels de bandits, ainsi qu'à l'analyse de leur complexité d'échantillonnage. Nos algorithmes partagent un principe commun : ils équilibrent une phase d'exploration/détection, où l'apprenante identifie les entrées informatives, et une phase de reconstruction/certification, où elle reconstruit la structure inconnue et certifie son estimation. Nous concevons des stratégies efficaces pour ces deux phases en exploitant la structure spécifique du problème, via des inégalités de concentration, du sous-échantillonnage, des schémas de doublement, de l'élimination séquentielle et l'agrégation de tests d'hypothèses. Nous établissons des garanties non asymptotiques à la fois en espérance et en grande probabilité.

Question 2. *Quels sont les coûts incompressibles de la reconstruction de structure ?*

Pour chaque problème, nous établissons des bornes inférieures informationnelles sur la complexité d'échantillonnage de tout algorithme, fournissant une référence d'optimalité algorithmique. Notre stratégie de preuve consiste à construire des environnements difficiles dans lesquels la reconstruction de structure se ramène à des problèmes classiques de test en grande dimension (tests d'adéquation, détection de signal), puis à appliquer des outils informationnels standards tels que l'inégalité de Fano ou des inégalités de *data-processing*. Nous établissons également des bornes inférieures sur les quantiles en grande probabilité du budget d'échantillonnage, ce qui met en évidence de nouvelles limites intrinsèques de l'exploration pure au-delà des bornes en espérance.

Question 3. *Dans quels régimes l'analyse non asymptotique apporte-t-elle de nouveaux éclairages?*

Nous développons une analyse non asymptotique qui caractérise la complexité d'échantillonnage optimale sur l'ensemble des régimes de paramètres, et pour tout niveau de confiance δ. Alors que la complexité optimale est souvent bien comprise asymptotiquement ($\delta \to 0$), notre analyse met en évidence la dépendance précise à tous les paramètres. Cela est particulièrement pertinent dans les régimes emblématiques de grande dimension (grand d) et de grande échelle (grand n), où nous révélons des effets jusqu'ici masqués.

Question 4. *Quels sont les bénéfices de l'apprentissage actif par rapport à l'apprentissage passif ?*

Nous comparons des stratégies adaptatives à des approches de référence "batch" passives. Premièrement, nous identifions les régimes dans lesquels l'adaptation conduit à une complexité d'échantillonnage strictement plus faible, et nous quantifions précisément ces gains. Deuxièmement, nous mettons en évidence les bénéfices computationnels des stratégies adaptatives. En particulier, pour le clustering, nous montrons que les observations séquentielles et adaptatives peuvent casser la barrière computationnelle (*information-computation gap*) présente en "batch", permettant à des algorithmes polynomiaux d'atteindre la complexité d'échantillonnage optimale.

1.4 Présentation des contributions

1.4.1 Clustering actif paramétrique (Chapitre 3)

Le Chapitre 3 est un travail conjoint avec Alexandra Carpentier [2], Christophe Giraud [3], et Nicolas Verzelen [4], publié dans les actes de la conférence *Algorithmic Learning Theory* (ALT 2025) (Thuot et al., 2025).

Cadre du problème. Le Chapitre 3 étudie le problème de clustering actif paramétrique (Clustering with Bandit Feedback Problem, CBP), tel que décrit en Section 1.2. Dans ce problème, chaque objet $a \in [n]$ possède un vecteur moyen inconnu $\mu_a \in \mathbb{R}^d$, et deux objets $a, b \in [n]$ appartiennent au même cluster si et seulement si $\mu_a = \mu_b$. La difficulté du problème dépend, au-delà de n, K, d et σ^2, de la séparation minimale entre clusters et de la proportion d'objets dans le plus petit cluster :

$$\Delta_* = \min_{k \neq \ell} \|\mu_k - \mu_\ell\|_2 \qquad \text{et} \qquad \theta_* = \min_{k \in [K]} \frac{|C_k|}{n} \in \left[\frac{1}{n}, \frac{1}{K}\right].$$

Ces deux quantités caractérisent la difficulté du problème : celui-ci devient plus difficile lorsque les clusters sont moins séparés (Δ_* petit) ou lorsque les groupes sont plus déséquilibrés (θ_* petit).

Borne inférieure. Nous établissons une borne inférieure non asymptotique sur le budget moyen minimal requis par tout algorithme δ-correct sur une classe d'environnements ayant une séparation minimale au moins égale à Δ_*, une proportion minimale au moins égale à θ_*, et un bruit σ^2-sous-gaussien. Dans le cas équilibré $\theta_* \approx 1/K$, à des constantes universelles près, Théorème 3.3.1 implique que

$$\mathbb{E}[\mathcal{T}] \gtrsim n + \frac{\sigma^2}{\Delta_*^2}\left[n \log\left(\frac{n}{\delta}\right) + \sqrt{dKn \log\left(\frac{n}{\delta}\right)}\right],$$

Le premier terme n est le coût incompressible lié à l'observation d'au moins un échantillon par bras. Le deuxième terme $n\frac{\sigma^2}{\Delta_*^2}\log(n/\delta)$ correspond au coût de certification de l'appartenance de chaque bras au bon cluster et apparaît même lorsque les centres de clusters sont connus. Ce terme domine en faible dimension, lorsque d est petit devant n, et il est optimal dans le régime asymptotique $\delta \to 0$ (Yang et al., 2024). Le troisième terme $\frac{\sigma^2}{\Delta_*^2}\sqrt{dKn\log(n/\delta)}$ capture le coût additionnel d'apprentissage des centres des clusters en grande dimension et domine dans ces régimes. Ce dernier terme, dépendant de d, est nouveau par rapport aux analyses asymptotiques telles que (Yang et al., 2024). Notre preuve requiert une réduction fine du clustering vers un problème de test d'adéquation en grande dimension entre gaussiennes de moyennes inconnues, ce qui constitue une contribution originale de ce chapitre.

Borne supérieure et optimalité. Nous concevons l'algorithme ACB (Algorithme 4), une procédure en temps polynomial qui est δ-correcte. Le Chapitre 3 montre qu'ACB atteint un budget moyen qui coïncide avec la borne inférieure informationnelle à des facteurs logarithmiques près pour tout $(n, K, d, \Delta_*, \theta_*, \delta)$, tout en restant computationnellement efficace — voir le

2. Institut für Mathematik, Universität Potsdam, Potsdam, Germany.
3. Université Paris-Saclay, Laboratoire de mathématiques d'Orsay, Orsay, France.
4. INRAE, Mistea, Institut Agro, Univ Montpellier, Montpellier, France.

Théorème 3.C.1. ACB suit une stratégie en deux étapes :

1. Une phase d'identification de représentants *Sequential Representative Identification* (Algorithme 2), qui identifie adaptativement un ensemble de K bras représentatifs contenant, avec grande probabilité, exactement un bras par cluster. Cette phase repose sur du sous-échantillonnage adaptatif et agrégation de tests d'adéquation en grande dimension.
2. Une phase de classification *Active Distance-based Classification* (Algorithme 3), qui concentre les échantillons sur les K représentants pour obtenir des estimations précises des centres de clusters, puis classe chaque bras restant en comparant sa moyenne empirique à ces centres.

Comparaison avec le clustering "batch". Le CBP est la version séquentielle et adaptative du clustering classique. Une comparaison naturelle est l'échantillonnage uniforme suivi d'un clustering "batch". Si l'apprenante dispose d'un budget fixé T, allouer T/n observations par bras réduit la variance à $n\sigma^2/T$, et les seuils de séparation de la Section 2.3.1 se traduisent directement en exigences de budget. Cette stratégie d'échantillonnage uniforme sert de base de comparaison pour évaluer les bénéfices de l'apprentissage séquentiel et adaptatif. Par exemple, lorsque $d \geqslant n$, que le bruit est isotrope et que les clusters sont équilibrés ($\theta_* \approx 1/K$), l'échantillonnage uniforme requiert un budget d'ordre $\frac{\sigma^2}{\Delta_*^2}\sqrt{dK^2n}$ pour réussir avec une probabilité d'erreur non triviale (Even et al., 2024), ce qui est sous-optimal d'un facteur $\sqrt{K}$ par rapport à la borne inférieure $\sqrt{dKn\log(n)}$.[5] Cet écart hérite du *gap computationnel* du clustering "batch", comme discuté en 2.3.1.

Le message principal du Chapitre 3 est qu'il fournit un exemple concret de problème où l'apprentissage séquentiel et adaptatif franchit une barrière computationnelle présente dans le problème "batch" correspondant.

Chapitre 3 **Résumé des contributions**

1. Nous formalisons le problème de clustering actif (CBP) sous bruit sous-gaussien, dans un cadre paramétrique, et identifions les quantités clés $(\Delta_*, \sigma^2, K, d, n, \theta_*)$ qui gouvernent sa complexité.
2. Nous établissons des bornes inférieures non asymptotiques sur le budget minimal requis par tout algorithme δ-correct, distinguant ainsi les régimes de faible et de grande dimension, et montrant que la complexité optimale suit
$$n + \frac{\sigma^2}{\Delta_*^2}\left[n\log\left(\frac{n}{\delta}\right) + \sqrt{dKn\log\left(\frac{n}{\delta}\right)}\right] \quad .$$
3. Nous introduisons l'algorithme ACB, une procédure en temps polynomial fondée sur la sélection de bras représentatifs, et prouvons qu'ACB est δ-correct et atteint la borne inférieure à des facteurs logarithmiques près dans les régimes équilibrés.
4. Dans le cadre séquentiel et adaptatif, nous prouvons l'absence de *gap computationnel*.

5. Pour faciliter la comparaison, nous omettons ici tous les termes en $\log(1/\delta)$; cette discussion est donc valable dans le régime de confiance modérée $\delta \simeq$ const.

1.4.2 Clustering actif non paramétrique (Chapitre 4)

Le Chapitre 4 est un travail conjoint avec Sebastian Vogt [6], Debarghya Ghoshdastidar [7], et Nicolas Verzelen [8], disponible en prépublication (Thuot et al., 2026).

Cadre non paramétrique. Le Chapitre 3 repose sur un cadre paramétrique structuré, avec des clusters linéairement séparables et une géométrie simple (par exemple gaussienne isotrope). Le Chapitre 4 étend le problème à un clustering non paramétrique avec des géométries plus complexes (voir Section 1.2), où les bras sont regroupés selon leurs distributions sous-jacentes plutôt que selon des vecteurs moyens de dimension finie. Chaque ν_i est supportée sur un espace topologique séparable $\mathcal{X}$ (potentiellement de dimension infinie), et nous n'imposons aucune hypothèse paramétrique au-delà de conditions faibles sur le noyau.

Noyau et méta-algorithme. Nous adoptons une approche à noyau (Gretton et al., 2012; Muandet et al., 2017; Wolfer and Alquier, 2025). Nous considérons le plongement de chaque distribution ν_i dans un espace de Hilbert à noyau reproduisant (RKHS) $\mathcal{H}$ via le *kernel mean embedding* (KME) $\mu_i = \mathbb{E}_{X\sim\nu_i}[g(X,\cdot)]$, où g est un noyau borné, invariant par translation et caractéristique sur $\mathcal{X}\times\mathcal{X}$. Sous l'hypothèse de noyau caractéristique, $\nu_i = \nu_j$ équivaut à $\mu_i = \mu_j$; le CBP non paramétrique se ramène donc à un clustering des bras selon leurs KME dans $\mathcal{H}$. Nous mesurons la séparation entre clusters par le carré de la *maximum mean discrepancy*, c'est-à-dire la distance carrée entre les KME correspondants dans $\mathcal{H}$: $\mathrm{MMD}^2(\nu_i,\nu_j) = \|\mu_i - \mu_j\|_{\mathcal{H}}^2$.

Borne supérieure. Pour capturer la difficulté du problème, nous introduisons un rapport signal-sur-bruit propre au chapitre, s_*^2, qui dépend conjointement de la séparation inter-clusters dans $\mathcal{H}$ (en MMD) et des variances en RKHS. Il est défini par

$$s_*^{-2}(\nu) := \max_{\substack{i\neq j\in[n]\\ \mu_i\neq\mu_j}} \left(\frac{\mathcal{V}_i^* \vee \mathcal{V}_j^*}{\|\mu_i-\mu_j\|_{\mathcal{H}}^2} \vee \frac{\sqrt{\bar{g}}}{2\|\mu_i-\mu_j\|_{\mathcal{H}}} \right) ,$$

où $\mathcal{V}_i^*$ est le proxy de variance RKHS associé au bras i (défini au Chapitre 4), et $\bar{g}$ est une borne supérieure du noyau g. Le Théorème 4.3.1 montre que notre algorithme KACB (Algorithme 9) atteint, avec probabilité $1-\delta$, un budget majoré par

$$\mathcal{T} \lesssim n\, s_*^{-2} \log\left(\frac{n}{\delta}\right) .$$

Cette borne comporte deux termes. Le premier, $n\,\frac{\mathcal{V}_i^*}{\|\mu_i-\mu_j\|_{\mathcal{H}}^2}\log(n/\delta)$, est l'analogue à noyau du deuxième terme du cadre paramétrique ($n\frac{\sigma^2}{\Delta^2}\log(1/\delta)$), et domine lorsque la variance est grande. Le second terme, $n\sqrt{\bar{g}}/\|\mu_i-\mu_j\|_{\mathcal{H}}$, constitue une limitation dépendant de la borne supérieure $\bar{g}$ du noyau ; ainsi, la prise en compte de la variance en RKHS n'améliore la borne que lorsque cette variance est suffisamment élevée.

L'idée algorithmique de KACB est volontairement simple. Nous échantillonnons uniformément

6. Contribution égale—Technical University of Munich, Munich, Germany.
7. Technical University of Munich, Munich, Germany.
8. INRAE, Mistea, Institut Agro, Univ Montpellier, Montpellier, France.

tous les bras, exécutons un test d'adéquation à noyau pour chaque paire (i, j), puis relions i et j dès que le test ne rejette pas l'hypothèse nulle $\nu_i = \nu_j$. Nous utilisons des inégalités de concentration tenant compte de la variance pour les KME empiriques (Tolstikhin et al., 2016; Wolfer and Alquier, 2025), afin de contrôler les déviations pour des MMD empiriques. Le clustering estimé est alors l'ensemble des composantes connexes du graphe obtenu sur $\{1, \dots, n\}$. Le budget par bras est augmenté de manière adaptative via un schéma de doublement jusqu'à ce que les tests par paires soient suffisamment précis pour séparer les clusters de façon fiable.

Limites. Contrairement à l'algorithme ACB paramétrique du Chapitre 3, cette procédure est plus naïve. Sa force principale est sa robustesse : elle est δ-correcte sous des hypothèses structurelles nettement plus faibles. En revanche, notre analyse est limitée par l'absence de bornes inférieures correspondantes, qui seraient nécessaires pour caractériser complètement la complexité d'échantillonnage optimale. Il s'agit d'une question ouverte importante pour des travaux futurs.

Chapitre 4 **Résumé des contributions**

1. Nous formalisons une version non paramétrique du problème de clustering actif (CBP), où les bras sont clusterisés selon leurs distributions sous-jacentes, et reformulons la tâche comme un clustering de *kernel mean embeddings* dans un RKHS.
2. Nous introduisons l'algorithme Kernel Active Clustering with Bandit (Algorithme 9), une méta-procédure simple reposant sur des tests d'adéquation à noyau. Nous montrons que KACB est δ-correct et atteint un budget total
$$\mathcal{T} \lesssim n\, s_*^{-2} \log\left(\frac{n}{\delta}\right),$$
à des facteurs logarithmiques près, où s_*^2 est un paramètre de complexité dépendant de l'instance, piloté par la séparation minimale en MMD et les variances à noyau.
3. Nous établissons que des garanties non asymptotiques pertinentes restent possibles au-delà du cadre paramétrique du Chapitre 3, étendant ainsi le cadre du clustering actif à une classe plus large de problèmes de clustering distributionnel.

1.4.3 Clustering actif avec sélection d'attributs (Chapitre 5)

Le Chapitre 5 est un travail conjoint avec Maximilian Graf [9], et Nicolas Verzelen [10], publié dans les actes de ICML 2025 (Graf et al., 2025).

Variante avec sélection d'attributs. Dans le Chapitre 5, nous étudions le problème de clustering actif (CBP) dans la variante avec sélection d'attributs. En plus de choisir un objet, l'apprenante choisit un attribut à chaque tour. Au temps t, elle sélectionne une paire objet-attribut

9. Contribution égale—Institut für Mathematik, Universität Potsdam, Potsdam, Germany
10. INRAE, Mistea, Institut Agro, Univ Montpellier, Montpellier, France

$(I_t, J_t) \in [n] \times [d]$ et reçoit une observation bruitée de M_{I_t,J_t} avec bruit sous-gaussien. Nous nous concentrons sur le cas à deux clusters ($K = 2$), où les deux groupes ont pour vecteurs moyens $\mu_0, \mu_1 \in \mathbb{R}^d$, et nous notons $\Delta = \mu_1 - \mu_0$ le vecteur d'écarts.

Borne supérieure. Nous introduisons un algorithme entièrement adaptatif, `BanditClustering` (Algorithme 13), construit à partir de deux sous-routines séquentielles. Les deux étapes reposent sur une adaptation de *Sequential Halving* (SH) avec sous-échantillonnage, utilisée pour équilibrer la détection (trouver des attributs informatifs) et la classification (classer tous les objets une fois un bon attribut trouvé).

1. `CandidateRow` (Algorithme 11) identifie un objet représentatif de chaque cluster avec grande probabilité. Cette étape est interprétée comme un cas particulier du problème de détection de signal (Problème (i), Section 2.5), et est réalisée via une combinaison de méthodes de sous-échantillonnage, couplées à la stratégie d'élimination SH de Karnin et al. (2013a).
2. `ClusterByCandidates` (Algorithme 12) identifie un attribut fortement discriminant et classe tous les objets à l'aide de cet attribut. Cela est réalisé via un schéma de doublement, qui augmente adaptativement le budget d'échantillonnage afin de détecter un attribut suffisamment informatif pour séparer les clusters avec grande probabilité.

La borne supérieure non asymptotique obtenue dépend de l'instance — voir le Théorème 5.3.1. À des facteurs logarithmiques près, avec probabilité au moins $1 - \delta$, le budget vérifie $\mathcal{T} \lesssim \log(1/\delta)\, H$, avec

$$H = \frac{d}{\theta}\left(\frac{1}{\|\Delta\|_2^2} + \frac{1}{s^*}\right) + \min_{s\in[d]}\left(\frac{d}{s} + n\right)\left(\frac{1}{\Delta_{(s)}^2} + 1\right) ,$$

où θ est le degré d'équilibrage de la partition, $\Delta_{(1)} \geq \cdots \geq \Delta_{(d)}$ sont les écarts absolus ordonnés, et $s^* \in \arg\max_{s\in[d]} s\Delta_{(s)}^2$ est un paramètre de parcimonie effective.

Nous exploitons la capacité des techniques d'élimination à bien fonctionner dans les situations où il existe de nombreux bons bras (ou de nombreux attributs informatifs) (Karnin et al., 2013a), ce qui est un aspect clé du problème. L'algorithme obtenu est computationnellement efficace et adaptatif à la structure inconnue de Δ. L'intuition centrale est que, dès qu'un bon attribut est identifié, le problème se ramène à classer n objets à partir de cet attribut, tâche nettement plus simple que le clustering dans l'espace original de dimension d.

Borne inférieure et optimalité. Nous établissons en outre une borne inférieure dépendant de l'instance, montrant que tout algorithme δ-correct doit dépenser au moins

$$\frac{d}{\theta\|\Delta\|_2^2}\log\frac{1}{\delta} \;\vee\; \frac{n}{\Delta_{(1)}^2}\log\frac{1}{\delta}$$

sur certaines instances permutées. Dans le Corollaire 5.3.2, nous montrons que les bornes supérieure et inférieure coïncident à des facteurs polylogarithmiques près dans les régimes à deux niveaux où $\Delta \in \{0, h\}^d$.

Discussion. Le terme de complexité comporte plusieurs composantes. Pour simplifier, nous discutons le cas $\Delta \in \{0, h\}^d$, étroitement lié aux exemples motivants de la Section 2.5. Dans ce

cas, les bornes supérieure et inférieure se réduisent à $\frac{d}{\theta s h^2}\log(1/\delta) + \frac{n}{h^2}\log(1/\delta)$. Le premier terme est un coût de détection, correspondant à l'identification de deux objets représentatifs, un par cluster. Le facteur $\frac{d}{s} \times \frac{1}{\theta}$ est le coût intrinsèque de sous-échantillonnage pour trouver une ligne du second groupe et un attribut discriminant entre les groupes. De plus, $\frac{1}{h^2}\log(1/\delta)$ est le coût de certification que l'écart correspondant est non nul avec un niveau de confiance $1-\delta$. Le second terme, $\frac{n}{h^2}\log(1/\delta)$, est le coût de classification de tous les objets à l'aide de cet attribut. Pour un Δ général, la stratégie optimale consiste à s'adapter à un certain niveau de parcimonie effective s^* qui réalise le minimum dans la borne supérieure.

Le message principal du Chapitre 5 est que la complexité d'échantillonnage dépend de la structure complète du vecteur d'écarts Δ, et qu'il est possible de s'adapter à la dimensionnalité effective du problème.

Chapitre 5 — **Résumé des contributions**

1. Nous formalisons un modèle de CBP avec sélection d'attributs, dans lequel l'apprenante doit identifier conjointement des coordonnées informatives et retrouver les clusters sous garantie de confiance fixée.
2. Nous proposons `BanditClustering`, un algorithme δ-correct, fondé sur une recherche d'attributs de type *Sequential Halving* (SH). Avec probabilité au moins $1-\delta$, le budget vérifie $\mathcal{T} \leqslant H\log(1/\delta)$, où H est le terme de complexité dépendant de l'instance défini par
$$H = \frac{d}{\theta}\left(\frac{1}{\|\Delta\|_2^2} + \frac{1}{s^*}\right) + \min_{s\in[d]}\left(\frac{d}{s}+n\right)\left(\frac{1}{\Delta_{(s)}^2}+1\right) .$$
3. Nous établissons une borne inférieure informationnelle dépendant de l'instance et montrons une quasi-coïncidence des bornes supérieure/inférieure à des facteurs polylogarithmiques près dans les régimes à deux niveaux.

1.4.4 Identification du Vainqueur de Condorcet (Chapitre 6)

Le Chapitre 6 est un travail conjoint avec El Mehdi Saad [11] et Nicolas Verzelen [12], disponible en prépublication sur arXiv (Saad et al., 2026).

Cadre du problème. Le Chapitre 6 étudie l'Identification du Vainqueur de Condorcet (CWI) dans des *dueling bandits* stochastiques (voir Section 1.2.3). Le vainqueur de Condorcet (CW) est un bras $i^* \in [K]$ préféré à tous les autres bras, c'est-à-dire tel que $\Delta_{i^*,j} > 0$ pour tout $j \neq i^*$. En supposant qu'un CW existe, l'objectif est d'identifier i^* avec probabilité au moins $1-\delta$ tout en minimisant la complexité d'échantillonnage $\mathcal{T}$. [13] Une caractéristique clé de notre analyse est

11. Contribution égale—UM6P College of Computing, Rabat, Morocco.
12. INRAE, Mistea, Institut Agro, Univ Montpellier, Montpellier, France.
13. Le cadre à budget fixé est également étudié dans ce chapitre.

qu'elle ne repose pas sur des hypothèses structurelles sur la matrice de préférences Δ, en particulier sur l'existence d'un ordre total des bras.

Méthodes de référence. Nous comparons nos algorithmes à deux références :

1. D'abord, nous comparons au budget $H_{\mathrm{cw}} = \sum_{i \neq i^*} \Delta_{i^*,i}^{-2} \log(1/\delta)$ obtenu dans la méthode de l'état de l'art de Maiti et al. (2024). La quantité H_{cw} correspond au coût d'élimination des bras sous-optimaux en les comparant directement au CW. Ce budget est optimal lorsque le CW est le bras le plus fort face à tous les autres, c'est-à-dire lorsque $\Delta_{i^*,j} = \max_{i \neq j} \Delta_{i,j}$ pour tout $j \neq i^*$. Toutefois, cette complexité peut être arbitrairement grande lorsque le CW est presque à égalité avec d'autres bras. Notre analyse améliore cette garantie en général en exploitant la matrice complète des écarts, tout en la retrouvant lorsque le CW est effectivement le bras le plus fort contre tous les autres.
2. Nous discutons aussi l'approche de Karnin (2016), avec le budget asymptotique $\sum_{i \neq i^*} \min_{j:\Delta_{i,j}<0} \Delta_{i,j}^{-2} \log(1/\delta)$, qui est optimal lorsque $\delta \to 0$. Cette borne est naturellement incompressible : décider que i^* est le CW revient à décider que chaque bras sous-optimal $i \neq i^*$ n'est pas un CW. Cela impose d'éliminer chaque bras de ce type en le comparant à un adversaire j tel que $\Delta_{i,j} < 0$, à coût minimal lorsque l'on utilise l'adversaire le plus fort, $\operatorname{argmax}_{j:\Delta_{i,j}<0} \Delta_{i,j}$. Ainsi, le budget asymptotique de Karnin (2016) correspond au coût minimal d'élimination de chaque bras sous-optimal via son adversaire le plus fort, et ce coût est incompressible, comme le montre la borne inférieure de Haddenhorst et al. (2021b). Néanmoins, cette borne est asymptotique et valide dans le régime $\delta \to 0$. Dans notre analyse non asymptotique, nous montrons que pour des niveaux de confiance modérés (par exemple en grande échelle avec grand K et δ pas trop petit), cette borne asymptotique masque un coût intrinsèque d'exploration que nous caractérisons entièrement.

Compromis entre exploration et certification. L'élimination d'un bras sous-optimal $i \neq i^*$ implique deux coûts : l'exploration (trouver un adversaire qui bat i) et la certification (prouver que l'écart correspondant est négatif avec grande confiance). Notons $K_{i;<0} := |\{j : \Delta_{i,j} < 0\}|$ et $\Delta_{i,(1)} \leq \cdots \leq \Delta_{i,(K_{i;<0})} < 0$ les écarts négatifs ordonnés contre i. Pour une certaine parcimonie effective $s_i \leq K_{i;<0}$, ces coûts se comportent comme $K/(s_i \Delta_{i,(s_i)}^2)$ et $\log(1/\delta)/\Delta_{i,(s_i)}^2$, comme le suggère l'exemple motivant de la Section 2.5. Ainsi, pour éliminer le bras i, il faut trouver un adversaire j tel que $\Delta_{i,j} = \Delta_{i,(s_i)}$ pour un certain s_i, puis certifier que $\Delta_{i,j} < 0$. Le choix optimal de s_i dépend à la fois de l'instance et de δ, et est sélectionné adaptativement par notre algorithme. Pour capturer les coûts d'exploration et de certification qui en résultent, nous définissons, pour tout $\boldsymbol{s} = (s_1, \ldots, s_K)$ avec $s_i \leq K_{i;<0}$,

$$H_{\mathrm{explore}}(\boldsymbol{s}, \delta) := \max_{i \neq i^*} \frac{K \log(1/\delta)}{s_i \Delta_{i,(s_i)}^2} + \sum_{i \neq i^*} \frac{K}{s_i \Delta_{i,(s_i)}^2}, \qquad H_{\mathrm{certify}}(\boldsymbol{s}, \delta) := \sum_{i \neq i^*} \frac{\log(1/\delta)}{\Delta_{i,(s_i)}^2}.$$

La complexité d'échantillonnage se décompose en un compromis entre ces termes.

Idées algorithmiques. FB-CWI (Algorithme 16) est un algorithme d'élimination fondé sur deux sous-routines : une recherche d'adversaire "fort" (adaptée de Sequential Halving Karnin et al. (2013a)) et une recherche d'adversaire "faible" basée sur une estimation adaptative de quantile

via RANGE-QUANTILE (Algorithme 15). Leurs sorties définissent un score qui élimine une fraction constante des bras actifs à chaque étape. Fait intéressant, RANGE-QUANTILE est une nouvelle procédure qui estime le s-ième quantile d'un vecteur et peut être d'intérêt indépendant.

FC-CWI (Algorithme 17) encapsule FB-CWI dans un schéma de doublement et ajoute deux tests d'arrêt : (i) une certification directe du CW, qui vérifie si le candidat CW possède une ligne positive, et (ii) une certification de frontière d'élimination, qui vérifie que notre élimination fondée sur le score est correcte. L'algorithme est δ-correct et, à des facteurs logarithmiques en K près et avec un terme additionnel en $\log\log(1/\delta)$ (voir le Théorème 6.3.1), satisfait avec probabilité au moins $1-\delta$:

$$\mathcal{T}_\delta \lesssim H_{\mathrm{cw}}(\delta) \wedge \min_{\boldsymbol{s}} \Big\{ \big(H_{\mathrm{certify}}(\boldsymbol{s},\delta) + H_{\mathrm{explore}}(\boldsymbol{s},\delta)\big) \Big\} .$$

Si le CW est le bras le plus fort contre tous les autres, on retrouve la garantie $H_{\mathrm{cw}}(\delta)$ de Maiti et al. (2024) [14] ; sinon, notre borne l'améliore. Cependant, notre borne est toujours moins bonne que la borne asymptotique de Karnin (2016). En effet, pour des niveaux de confiance modérés, le coût d'exploration domine le coût de certification, alors que la borne asymptotique ne capture que ce dernier. Par exemple, si l'on traite $\log(1/\delta)$ comme une constante, le membre de droite de notre borne supérieure est dominé par le terme $\min_{\boldsymbol{s}} \sum_{i\neq i^*} \frac{K}{s_i \Delta^2_{i,(s_i)}} \simeq \sum_{i\neq i^*} \frac{K}{\|\Delta_i^-\|_2^2}$, à un facteur $\log(d)$ près, [15] qui peut être plus grand que la borne de Karnin (2016) d'un facteur au plus K. Notre borne inférieure confirme que ce coût d'exploration est intrinsèque et incompressible.

Bornes inférieures et optimalité. La borne inférieure de Haddenhorst et al. (2021b) établit que le terme de complexité de Karnin (2016) est incompressible.

Dans le Théorème 6.4.2, nous établissons une borne inférieure en grande probabilité, apportant une compréhension nouvelle de la complexité du problème de CWI. La preuve repose sur une réduction à un problème de tests multiples actifs, ce qui est inédit dans ce cadre. Pour simplifier, nous donnons ici la version minimax (Corollaire 6.4.4), valable pour tout algorithme δ-correct et toute matrice d'écarts $\boldsymbol{\Delta}$. Pour tout algorithme δ-correct π,

$$\sup_{\tilde{\boldsymbol{\Delta}}\in\mathbb{D}(\boldsymbol{\Delta})} \inf \Big\{ \chi > 0 \text{ s.t.: } \mathbb{P}_{\tilde{\boldsymbol{\Delta}},\pi}(N_\delta \leqslant \chi) \leqslant \delta \Big\} \gtrsim \min_{\boldsymbol{s}} \{ H_{\mathrm{certify}}(\boldsymbol{s},\delta) + H_{\mathrm{explore}}(\boldsymbol{s},\delta) \} ,$$

où $\mathbb{D}(\boldsymbol{\Delta})$ est un ensemble de perturbations de $\boldsymbol{\Delta}$ qui préservent le CW, la structure de signe de $\boldsymbol{\Delta}$, et le niveau de parcimonie optimal $\boldsymbol{s}^*_{\boldsymbol{\Delta}}$. [16] Globalement, cette borne inférieure confirme que le compromis exploration-certification est intrinsèque et incompressible, et que notre méthode est d'ordre optimal.

Le message principal de ce travail est que nous améliorons les bornes existantes et obtenons une meilleure compréhension de la complexité du problème de CWI lorsque aucune hypothèse structurelle n'est imposée à la matrice de préférences Δ, à l'exception de l'existence d'un CW. De plus, notre analyse vaut pour tous les niveaux de confiance $\delta \in (0,1)$, ce qui permet une compréhension plus complète du problème, particulièrement pertinente en grande échelle lorsque K est grand.

14. Et, par corollaire, la borne de Karnin (2016), égale à $H_{\mathrm{cw}}(\delta)$ sous cette hypothèse.

15. D'après l'inégalité $\max_{k\in\{1,\dots,K\}} k x_k^2 \leq \sum_{i=1}^K x_i^2 \leq \log(4K) \max_{k\in\{1,\dots,K\}} k x_k^2$, valable pour toute suite décroissante $x_1 \geq \cdots \geq x_K$.

16. Nous renvoyons au Chapitre 6 pour la définition précise de cette classe locale $\mathbb{D}(\boldsymbol{\Delta})$.

Chapitre 6 **Résumé des contributions**

1. Nous introduisons FB-CWI et FC-CWI, deux nouveaux algorithmes d'élimination pour les cadres à budget fixé et à confiance fixée, construits à partir de Sequential Halving, d'une nouvelle sous-routine RANGE-QUANTILE, et d'une règle d'élimination fondée sur un score. FC-CWI est δ-correct et satisfait, avec probabilité au moins $1-\delta$,
$$\mathcal{T}_\delta \lesssim H_{\mathrm{cw}}(\delta) \wedge \min_{\boldsymbol{s}} \Big\{ \big(H_{\mathrm{certify}}(\boldsymbol{s},\delta) + H_{\mathrm{explore}}(\boldsymbol{s},\delta)\big) \Big\} ,$$
ce qui améliore, dans tous les régimes, la complexité fondée sur le seul vainqueur $H_{\mathrm{cw}}(\delta)$.
2. Nous établissons de nouvelles bornes inférieures pour CWI, obtenues via une réduction à un problème de tests multiples actifs. Cette borne fournit une borne inférieure locale-minimax montrant que le compromis exploration-certification est intrinsèque et incompressible.
3. Nous mettons en évidence des phénomènes invisibles aux analyses asymptotiques, ainsi que le rôle non trivial de la matrice complète des écarts plutôt que de la seule ligne du vainqueur. Ce compromis exploration-certification constitue une contribution nouvelle à la théorie du CWI.

1.4.5 Identification de points de rupture multiples (Chapitre 7)

Le Chapitre 7 est un travail conjoint avec Maximilian Graf[17], disponible en prépublication dans (Graf and Thuot, 2026).

Cadre du problème. Dans le Chapitre 7, nous étudions le problème de détection de ruptures multiples (Multiple Change Point Identification, MCP) dans un cadre actif. Nous considérons une fonction constante par morceaux $f : [0,1] \to \mathbb{R}$ avec m points de rupture en des positions inconnues $0 < x_1^* < \cdots < x_m^* < 1$, de sorte que f est constante sur chaque intervalle $(x_{k-1}^*, x_k^*]$ (avec la convention $x_0^* = 0$ et $x_{m+1}^* = 1$). À chaque tour, l'apprenante sélectionne un point $I_t \in [0,1]$ et observe $X_t = f(I_t) + \varepsilon_t$, où ε_t est un bruit σ-sous-gaussien indépendant des requêtes passées. Dans le cadre à confiance fixée, l'objectif est de retrouver les m points de rupture avec une précision η et une probabilité au moins égale à $1-\delta$, tout en minimisant le nombre total de requêtes $\mathcal{T}$.[18] Plus précisément, l'objectif est de construire m estimations $(\widehat{x}_1, \ldots, \widehat{x}_m)$ telles que $\max_{k\in[m]} |\widehat{x}_k - x_k^*| \leqslant \eta$ avec probabilité au moins $1-\delta$.

Comparaison avec la littérature. Ce problème peut être vu, après discrétisation, comme une

17. Contribution égale—Institut für Mathematik, Universität Potsdam, Potsdam, Germany
18. Nous considérons le cadre plus général de la localisation d'un sous-ensemble de $N \leqslant m$ points de rupture ; voir le chapitre 7 pour les détails.

variante structurée du problème de clustering actif (CBP), auquel cas la dimension de l'espace des attributs est $d = 1$. La différence clé avec le CBP est que la structure spatiale de $[0, 1]$ peut être exploitée : une requête en un point x est informative non seulement sur la valeur locale de f, mais aussi sur l'existence possible d'un point de rupture proche. Cela motive une approche algorithmique fondamentalement différente, fondée sur une recherche binaire récursive (dichotomique) et des tests d'hypothèses locaux.

Le problème a d'abord été étudié dans Lazzaro and Pike-Burke (2025b), où l'analyse se concentre sur le régime asymptotique $\delta \to 0$. Leur résultat est de la forme $\lim_{\delta \to 0} \frac{\mathbb{E}[\mathcal{T}]}{\log(1/\delta)} \lesssim H_{\text{localize}}$, avec $H_{\text{localize}} := \sum_{k=1}^{m} \Delta_k^{-2}$ et $\Delta_k := |f(x_k^{*+}) - f(x_k^{*-})|$ la hauteur du saut en x_k^*. Cette borne est optimale dans le régime asymptotique et correspond au coût de certification de chaque point de rupture estimé au niveau de confiance $1 - \delta$. Cependant, cette caractérisation asymptotique ne capture pas l'exploration, c'est-à-dire le coût de détection de la présence des points de rupture. En outre, la dépendance au paramètre de précision η est masquée par le régime asymptotique. Notre analyse est non asymptotique à la fois en δ et en η, et améliore la dépendance en η de linéaire en $1/\eta$ à logarithmique en $1/\eta$. Cela est crucial dans les régimes de grande échelle où une haute précision de localisation est requise.

Notre analyse quantifie aussi l'exploration. Inspirés par la littérature passive, nous introduisons un paramètre d'énergie $\mathcal{E}_k^2 := s_k \Delta_k^2$, où s_k capture les espacements locaux via $s_k := \vartheta_{k-1} \wedge \vartheta_k$, avec $\vartheta_k := x_{k+1}^* - x_k^*$ pour $k = 1, \ldots, m-1$, et par convention $\vartheta_0 = \vartheta_m = 1$. Cela induit un terme de complexité de détection d'ordre $H_{\text{detect}} := \max_k 1/\mathcal{E}_k^2$, qui est inévitable d'après nos résultats de bornes inférieures.

Conception algorithmique. Nous introduisons `LocalizeChangePoints` (`LCP`), un algorithme adaptatif qui encapsule une procédure à budget fixé dans un schéma de doublement et s'arrête dès qu'un ensemble estimé de points de rupture est certifié correct. La méthode comporte quatre étapes : (i) une étape de détection, qui identifie des points dans chaque plateau de la fonction constante par morceaux f via des tests multi-échelles ; (ii) une étape d'estimation, qui estime la hauteur de saut de chaque point de rupture ; (iii) une étape de localisation, qui raffine les estimations à la précision η via une recherche binaire ; et (iv) une étape de vérification, qui confirme les estimations. Les étapes d'estimation et de vérification contribuent à un coût d'ordre $H_{\text{localize}} \log(1/\delta)$, en accord avec Lazzaro and Pike-Burke (2025b).

Par ailleurs, l'étape de détection introduit un coût additionnel d'ordre $H_{\text{detect}} = \max_k 1/\mathcal{E}_k^2$. Il s'agit d'un effet véritablement non asymptotique, invisible lorsque $\delta \to 0$. De plus, le raffinement à la précision η induit un terme supplémentaire $H_{\text{localize}} \log(1/\eta)$, qui est la dépendance optimale à la précision η. Globalement, la méthode est δ-correcte et satisfait des garanties non asymptotiques à la fois en grande probabilité et en espérance. À des facteurs logarithmiques près,

$$\begin{aligned} \mathbb{E}[\mathcal{T}] &\lesssim H_{\text{detect}} + H_{\text{localize}} \log\left(\frac{1}{\delta\eta}\right) && \text{en espérance ,} \\ \mathcal{T} &\lesssim H_{\text{detect}} \log\left(\frac{1}{\delta}\right) + H_{\text{localize}} \log\left(\frac{1}{\delta\eta}\right) && \text{avec probabilité au moins } 1-\delta \end{aligned}$$

Pour la borne en espérance, nous retrouvons la même dépendance en δ, à savoir $H_{\text{localize}} \log(1/\delta)$,

que Lazzaro and Pike-Burke (2025b), tout en améliorant la dépendance en η. Nous identifions aussi une complexité non asymptotique additionnelle H_{detect}, qui peut dominer lorsque les points de rupture sont très proches. Ce coût de détection est intrinsèque et incompressible d'après nos bornes inférieures. La complexité H_{detect} est toujours plus grande que H_{localize}, et l'écart peut être arbitrairement grand lorsque les points de rupture sont extrêmement proches et difficiles à détecter.

Bornes inférieures et optimalité. Nous établissons des bornes inférieures informationnelles pour la localisation MCP. Ces bornes inférieures coïncident avec nos bornes supérieures à des facteurs logarithmiques près. À notre connaissance, il s'agit de la première caractérisation de la complexité de localisation MCP à ce niveau de précision. La preuve repose sur une réduction à un problème de détection de signal, proche de celui discuté en Section 2.5. Comparées à Lazzaro and Pike-Burke (2025b), dont la garantie est asymptotiquement optimale lorsque $\delta \to 0$ mais non optimale pour des valeurs générales de δ, nos bornes améliorent la dépendance en η de linéaire en $1/\eta$ à logarithmique en $1/\eta$, tout en conservant la même dépendance en δ. Cela est particulièrement pertinent dans les régimes de haute précision.

Chapitre 7 **Résumé des contributions**

1. Nous introduisons `LocalizeChangePoints` (`LCP`), un algorithme adaptatif qui combine détection multi-échelle, estimation des sauts, localisation par recherche dichotomique et vérification. La méthode est δ-correcte et satisfait des garanties non asymptotiques en grande probabilité comme en espérance. À des facteurs logarithmiques près,

$$\mathbb{E}[\mathcal{T}] \lesssim H_{\text{detect}} + H_{\text{localize}} \log\left(\frac{1}{\delta\eta}\right) \qquad \text{en espérance ,}$$

$$\mathcal{T} \lesssim H_{\text{detect}} \log\left(\frac{1}{\delta}\right) + H_{\text{localize}} \log\left(\frac{1}{\delta\eta}\right) \qquad \text{avec probabilité au moins } 1-\delta$$

   où $H_{\text{detect}} = \max_i 1/\mathcal{E}_i^2$ et $H_{\text{localize}} = \sum_{i=1}^m 1/\Delta_{(i)}^2$.
2. Nous prouvons de nouvelles bornes inférieures informationnelles, montrant que tout algorithme doit payer à la fois un coût de détection et un coût de localisation. Sur une classe locale-minimax d'instances, ces bornes inférieures coïncident avec les bornes supérieures à des facteurs logarithmiques près, pour les garanties de budget moyen comme de quantiles.
3. Nous mettons en évidence un phénomène véritablement non asymptotique : la complexité d'échantillonnage est gouvernée conjointement par les amplitudes de saut et les espacements locaux (via les énergies $\mathcal{E}_i^2 = s_i\Delta_i^2$), au travers d'un compromis détection-vérification. En particulier, cela affine notre compréhension du problème MCP dans les régimes de haute précision (petit η).

1.5 Structure de la thèse

La thèse est organisée comme suit. Dans les Chapitres 3–5, nous étudions plusieurs variantes du Clustering with Bandit Feedback (CBP). Le Chapitre 3 traite du cas paramétrique, le Chapitre 4 du cas non paramétrique, et le Chapitre 5 du CBP avec sélection d'attributs. Dans le Chapitre 6, nous étudions le problème d'Identification du Vainqueur de Condorcet (CWI) dans des dueling bandits. Dans le Chapitre 7, nous étudions la localisation de points de rupture multiples (MCP). Chaque chapitre est autonome et peut être lu indépendamment des autres. [19]

19. Par ailleurs, ces chapitres sont disponibles en ligne sous forme d'actes de conférence ou de prépublication.

Chapter 2

Introduction (extended English version)

2.1 Context and motivation

A central goal of modern statistics and machine learning is to recover hidden structure from data. While supervised learning trains models using labeled data, unsupervised learning aims to infer latent structure without access to labels (Berry et al., 2020). In classical formulations, learning is typically batch: the learner receives a dataset collected in advance, in a batch, and infers the target structure from passive observations.

However, in many modern applications, data are collected sequentially and adaptively, for example, in recommendation systems (Zhou et al., 2012; Lu et al., 2015), online experimentation such as adaptive clinical trials (Chow and Chang, 2008), and crowdsourcing (Raykar et al., 2010; Ariu et al., 2024). In such settings, where the learner can adapt the sampling strategy on the fly during data collection, the key resource is the sampling budget, namely the number of observations needed to recover the target structure reliably. This naturally motivates a pure-exploration bandit perspective (Lattimore and Szepesvári, 2020), in which the learner allocates samples adaptively to identify an unknown structure with high confidence.

Bandit feedback can substantially improve efficiency over passive sampling by concentrating observations on informative parts of the instance. Yet adaptation also incurs a cost: before exploiting informative queries, the learner must discover where this information lies. This thesis studies several pure-exploration problems through this lens, focusing on the trade-off between the cost of identifying informative queries and the cost of recovering latent structure. To address high-dimensional and large-scale regimes that are increasingly common in practice (Giraud, 2021; Rigollet and Hütter, 2023), we develop a non-asymptotic analysis that captures both the benefits and the intrinsic limits of adaptive sampling for unsupervised learning.

Outline of introduction. This introduction is organized as follows. Section 2.2 presents a general framework that covers all the problems studied in this thesis. Sections 2.3.1 and 2.3.2 review the relevant literature in unsupervised learning and pure-exploration bandits. Section 2.4 states our guiding questions, and Section 2.5 provides a motivating example that introduces the key concepts used throughout the thesis. We conclude with a chapter-by-chapter summary of contributions (Section 2.6) and an outline of the thesis structure (Section 2.7).

2.2 Unsupervised learning with bandit feedback

2.2.1 Pure exploration in matrix bandit environments

Latent structure model. We study problems in which a learner aims to recover an unknown structure $\mathcal{C}^*_{\boldsymbol{M}}$ encoded by a latent matrix $\boldsymbol{M} \in \mathbb{R}^{n\times d}$.

Consider n items indexed by $[n] = \{1, \dots, n\}$ and d features indexed by $[d] = \{1, \dots, d\}$. Each item i has a latent feature vector $M_{i,\cdot} \in \mathbb{R}^d$, forming the rows of an unknown matrix $\boldsymbol{M} \in \mathbb{R}^{n\times d}$. The unknown structure $\mathcal{C}^*_{\boldsymbol{M}}$ is defined as some function of $\boldsymbol{M}$. The learner's goal is to recover $\mathcal{C}^*_{\boldsymbol{M}}$ exactly, or up to some specified precision.

The classical best-arm identification problem in multi-armed bandits can be viewed as a special case, where $\mathcal{C}^*_{\boldsymbol{M}}$ is the index of the maximum entry of $\boldsymbol{M}$ (Audibert and Bubeck, 2010). More generally, the latent structure may take different forms depending on the problem at hand: $\mathcal{C}^*_{\boldsymbol{M}}$ is a partition in clustering (Kaufman and Rousseeuw, 2009), a graph in community detection (Jin et al., 2021), a permutation in ranking (Saad et al., 2023), or a subset of entries in change-point detection (Niu et al., 2016). In Condorcet winner identification, $\mathcal{C}^*_{\boldsymbol{M}}$ is the index of the row that dominates all others in pairwise comparisons (Bengs et al., 2021). In this thesis, we study several problems that fit this framework. Chapters 3 to 5 address clustering, where $\mathcal{C}^*_{\boldsymbol{M}}$ is a partition of $[n]$ into K clusters. Chapter 6 studies Condorcet winner identification, where $\mathcal{C}^*_{\boldsymbol{M}}$ is the index of the Condorcet winner, i.e., the item preferred to all others in pairwise comparisons. Chapter 7 studies Multiple Change Point Identification, where $\mathcal{C}^*_{\boldsymbol{M}}$ is the set of change points of the underlying distribution.

Sequential and adaptive learning protocol. In classical batch unsupervised learning, the learner (referred to as *she* throughout) receives a fixed set of i.i.d. samples collected a priori. Here, by contrast, she collects observations sequentially and adaptively: at each step, she decides which entries to query based on past observations. This allows the learner to focus on the most informative entries, potentially reducing the number of samples needed for accurate recovery. The matrix $\boldsymbol{M}$ is unknown to the learner, who accesses it by interacting with a bandit environment (Lattimore and Szepesvári, 2020).

We consider a bandit environment ν that returns noisy observations of the entries of $\boldsymbol{M}$. At each round, the learner selects entries of $\boldsymbol{M}$, receives noisy observations of the selected entries, and after $\mathcal{T}$ queries outputs an estimate $\hat{\mathcal{C}}$. We focus on the fixed confidence setting: given a prescribed confidence level $\delta \in (0, 1)$, the typical goal is to ensure exact recovery, i.e., $\hat{\mathcal{C}} = \mathcal{C}^*_{\boldsymbol{M}}$, with probability at least $1 - \delta$.[1]

At each time t, she selects an item $I_t \in [n]$ and a feature subset $J_t \subset [d]$. We focus on two extreme cases: (i) full-feature observation ($J_t = [d]$), where she observes all features of the selected item; (ii) feature selection ($|J_t| = 1$), where she observes a single feature. Intermediate regimes have also been studied (Ariu et al., 2024), but we do not consider them here. Conditionally on the choice $I_t = i$ and $J_t = J$, the learner receives $X_t = (X_{t,j})_{j\in J_t}$, which is sampled from an unknown

1. Or exact recovery up to some precision in Chapter 7.

distribution $\nu_{i,J}$ with mean $\mathbb{E}[X_t] = (M_{i,j})_{j\in J}$. The noise $\varepsilon_t = (X_{t,j} - M_{I_t,j})_{j\in J_t}$ is independent across time t and from past interactions. We write $\nu_i = \nu_{i,[d]}$ for the distribution of a full-row observation of item i, and $\nu_{i,j} = \nu_{i,\{j\}}$ for the distribution of a single-entry observation (i,j). After $\mathcal{T} \in \mathbb{N}$ rounds, the learner outputs $\hat{\mathcal{C}}$, an estimate of $\mathcal{C}^*_{\boldsymbol{M}}$. The stopping time $\mathcal{T}$ is a random variable chosen adaptively by the learner.

Given $\delta \in (0,1)$, the learner aims to recover $\mathcal{C}^*_{\boldsymbol{M}}$ with probability at least $1-\delta$. A strategy π is δ-correct if $\mathbb{P}_{\pi,\nu}(\hat{\mathcal{C}} = \mathcal{C}^*_{\boldsymbol{M}}) \geq 1-\delta$.[2] Its sampling budget $\mathcal{T}_\pi(\nu,\delta)$ is the (random) number of samples collected during the exploration, which serves as a measure of performance.

In the fixed confidence setting, a pure exploration strategy π consists of three components (Kaufmann et al., 2016): (i) a sampling rule selecting (I_t, J_t) based on history; (ii) a stopping rule deciding when to stop; (iii) a recommendation rule outputting $\hat{\mathcal{C}}$. All three components may use external randomness; $\mathbb{P}_{\pi,\nu}$ denotes the induced law.

Sampling budget as measure of performance. In many modern applications, collecting samples is costly, so the sampling budget is a scarce resource (Bubeck et al., 2009). The objective is therefore to minimize $\mathcal{T}_\pi(\nu,\delta)$ while guaranteeing δ-correctness. We are particularly interested in characterizing the optimal sampling budget, defined as the minimum budget required by any strategy to recover the unknown structure with probability at least $1-\delta$. Deriving tight upper and lower bounds on this quantity is a central question of this thesis.

Our analysis is non-asymptotic: none of the parameters $(n, d, \delta$, and problem-specific quantities such as the number of clusters K in clustering) is assumed to dominate asymptotically the others. This viewpoint captures the benefits of adaptive sampling across regimes, especially in high-dimensional settings (large d) or large-scale problems (large n). We use $\mathcal{T}_\pi(\nu,\delta)$ as the main performance metric. In particular, we study the expected budget $\mathbb{E}_{\pi,\nu}[\mathcal{T}_\pi(\nu,\delta)]$ and quantile guarantees of the form $\mathbb{P}_{\pi,\nu}(\mathcal{T}_\pi(\nu,\delta) \leq T) \geq 1-\delta$.

The next two Subsections, and Table 2.1 give a unified overview of all problems studied in this thesis.

2.2.2 Clustering with bandit feedback (CBP)

In the Clustering with Bandit Feedback Problem (CBP) (Yang et al., 2024; Ariu et al., 2024; Yavas et al., 2025), the latent structure $\mathcal{C}^*_{\boldsymbol{M}}$ is a partition of $[n]$ into K clusters: $\mathcal{C}^*_{\boldsymbol{M}} = \{\mathcal{C}^*_1, \dots, \mathcal{C}^*_K\}$, with $\mathcal{C}^*_k \subset [n]$ and $\bigcup_k \mathcal{C}^*_k = [n]$. Two items $i, i' \in [n]$ belong to the same cluster if and only if they share identical feature distributions $\{\nu_{i,j}\}_{j\in[d]}$. The learner's goal is to recover the partition $\mathcal{C}^*_{\boldsymbol{M}}$ exactly.[3] We study three variants of CBP, which differ in the observation scheme and noise model.

Chapter 3 (parametric CBP). The learner adaptively chooses an item and receives a noisy evaluation of all its features ($J_t = [d]$). Each row $M_{i,\cdot} \in \mathbb{R}^d$ is the mean of ν_i. She picks $I_t \in [n]$ and observes $X_t \sim \nu_{I_t}$ with $\mathbb{E}[X_t] = M_{I_t,\cdot}$. The noise is σ-sub-Gaussian ($\sigma > 0$ known), which covers Gaussian and bounded settings. Introduced in Yang et al. (2024), this generalizes classical clustering to a sequential and adaptive framework.

2. We use here δ-correct instead of the other common terminology δ-PAC.
3. As usual in clustering, one needs to recover $\mathcal{C}^*_1, \dots, \mathcal{C}^*_K$ up to a permutation of cluster labels.

Chapter 4 (non-parametric CBP). In Chapter 4, we study a non-parametric version of CBP, where the objective is distributional clustering (Dhillon et al., 2004). As in the parametric case, the learner adaptively chooses an item and receives noisy evaluations of its full feature distribution ν_i, but no parametric assumption is imposed on ν_i. Instead, we assume that the distributions are supported on a separable topological space $\mathcal{X}$ (possibly infinite-dimensional), so that the noise model is arbitrary. We tackle this general setting through kernel methods.

Chapter 5 (CBP with feature selection). In Chapter 5, we study a variant of CBP with adaptive feature selection, close to Ariu et al. (2024), in a parametric setting. In this fully adaptive protocol, the learner selects both an item and a single feature at each round ($|J_t| = 1$). At time t, she chooses $I_t \in [n]$ and $J_t \in [d]$, and observes $X_t \sim \nu_{I_t,J_t}$, a noisy evaluation of M_{I_t,J_t} with σ-sub-Gaussian noise. We focus on the two-cluster case ($K = 2$).

2.2.3 Condorcet Winner Identification in dueling bandits (CWI)

In Chapter 6, we study the Condorcet Winner Identification (CWI) problem in dueling bandits Bengs et al. (2021). We consider K items $[K]$, with environment matrix $\boldsymbol{Q} \in [0,1]^{K\times K}$. In this setting, the roles of items and features are symmetric, so $n = d = K$.

For a pair of items (i, j), the quantity $Q_{i,j}$ is the probability that item i is preferred to item j. At time t, the learner picks $(I_t, J_t) \in [K]^2$ and observes $X_t \sim \text{Bernoulli}(Q_{I_t,J_t})$.[4] This corresponds to settings where the learner can access the environment only through pairwise comparisons. The gap matrix is defined by $\Delta_{i,j} \coloneqq Q_{i,j} - \frac{1}{2}$, so $\Delta_{i,j} = -\Delta_{j,i}$ (anti-symmetric). The unknown structure to recover is the Condorcet winner (CW): an arm $i^* \in [K]$ such that $\Delta_{i^*,j} > 0$ for all $j \neq i^*$, meaning it wins in expectation against every other arm. The learner's goal is to identify i^* with probability at least $1 - \delta$. This can be viewed as a pairwise-comparison generalization of best-arm identification.

2.2.4 Multiple Change Point Identification (MCP)

In Chapter 7, we study Multiple Change Point Identification (MCP), in a formulation close to Lazzaro and Pike-Burke (2025b). We consider a continuous item space $[0, 1]$. The environment is characterized by a function $f : [0, 1] \to \mathbb{R}$ that is piecewise constant, with m discontinuities, denoted as Change Points (CP) at unknown locations $0 < x_1^* < \cdots < x_m^* < 1$. Hence, f is constant on each interval $(x_{i-1}^*, x_i^*]$, with $x_0^* = 0$ and $x_{m+1}^* = 1$. The learner sequentially queries points $I_t \in [0, 1]$ and observes $X_t = f(I_t) + \varepsilon_t$, where ε_t is σ-sub-Gaussian noise. The spatial structure of $[0, 1]$ is a key feature that distinguishes this problem from CBP.

As we work in a continuous action space, exact recovery of the change points is impossible. Instead, the learner aims to recover the change points with a localization precision $\eta > 0$ and error rate $\delta \in (0, 1)$. The goal is to output estimates $\{\hat{x}_1, \ldots, \hat{x}_m\}$ such that $\max_i |\hat{x}_i - x_i^*| \leq \eta$ with probability at least $1 - \delta$.[5]

4. By construction, $\boldsymbol{Q}$ satisfies the relation $Q_{i,j} = 1 - Q_{j,i}$.

5. We focus on the problem of localizing the full set of Change Points in this introduction. In Chapter 7, we consider the more general setting where the learner has to output only a subset of $N \leqslant m$ change points.

Table 2.1 – Summary of the different problems and settings studied in this thesis. Here, CBP stands for Clustering with Bandit Feedback, CWI for Condorcet Winner Identification, and MCP for Multiple Change Point Identification. For each problem, we specify the item set, the feature space, the unknown structure, the observation scheme, the noise model, and the objective.

Chap.	**Problem**	**Items**	**Features**	**Unknown Structure**	**Observation/ step**	**Noise**	**Objective**
3	Parametric CBP	$[n]$	$\mathbb{R}^d$ $(d < \infty)$	K clusters	one item, all features	σ-sub-Gauss.	Fixed confidence
4	Non-parametric CBP	$[n]$	$\mathcal{X}$ $(d \leqslant \infty)$	K clusters	one item, all features	any	Fixed confidence
5	CBP with feature selection	$[n]$	$\mathbb{R}^d$ $(d < \infty)$	K clusters with $K = 2$	one item, one feature	σ-sub-Gauss.	Fixed confidence
6	CWI	$[K]$	/	Condorcet winner	pairwise comparisons	Bernoulli	Fixed confidence & fixed budget
7	MCP	$[0, 1]$ (continuous)	$\mathbb{R}$ $(d = 1)$	Change Point locations	evaluation of f at one point	σ-sub-Gauss.	Fixed confidence

2.3 Related work

Our work lies at the intersection of unsupervised learning and pure-exploration bandits. First, we present a selective overview of unsupervised learning, with a focus on clustering, which serves as a comparison point for our bandit-based approach to clustering. Then, we review the literature on pure-exploration bandits, which provides the methodological foundation for our work.

2.3.1 Unsupervised learning

Classical batch clustering. Unsupervised learning aims at discovering patterns in data without access to labels. Typically, we consider a setting where one observes n independent samples $X_1, \dots, X_n$ on a d-dimensional space $\mathbb{R}^d$, and the goal is to recover some latent structure $\mathcal{C}^*$ over the data. The nature of the latent structure depends on the problem at hand: it can be a partition of the data (clustering), a graph (community detection), a permutation (ranking), or a subset of items (change point detection).

Clustering is arguably its most classical instance, where the goal is to partition the data into groups of similar items. Standard textbooks (Kaufman and Rousseeuw, 2009; Jain et al., 1999) and monographs (Giraud, 2021) present a wide range of clustering methods, from k-means to hierarchical procedures. The K-means paradigm, introduced in the foundation work of McQueen (1967), consists in finding a partition $\mathcal{C}_1, \dots, \mathcal{C}_K$ of $[n]$ that minimizes the within-cluster sum of squares

$$\sum_{k=1}^{K} \sum_{i \in \mathcal{C}_k} \|X_i - \bar{X}_{\mathcal{C}_k}\|^2,$$

where $\bar{X}_{\mathcal{C}_k}$ is the mean of the points in cluster $\mathcal{C}_k$. This simple objective has made K-means standard in applications (Jain et al., 1999) from image processing to market segmentation. However, the K-means problem is known to be NP-hard in general (Aloise et al., 2009), and the global minimizer of the K-means criterion is impossible to find in practice. Heuristic approximation algorithms such as Lloyd's method (Lloyd, 1982) are widely used in practice. It consists of iterating two steps: (i) assign each point to the nearest cluster center; (ii) update the cluster centers as the mean of the groups. The initialization of the cluster centers is crucial for the performance of Lloyd's algorithm, and it is common to run it multiple times with different initializations. Despite its simplicity, Lloyd's algorithm is only known to converge to a local minimum of the K-means objective; nevertheless, it often performs well in practice. Still, the underlying optimization problem is highly non-convex. Numerous variants and refinements have been proposed to improve its empirical behavior, for instance through smarter initialization strategies such as K-means++ (Arthur and Vassilvitskii, 2007; Celebi et al., 2013), or convex relaxation methods (Peng and Wei, 2007). Spectral clustering (Ng et al., 2001) is another popular approach, which leverages the spectrum of affinity matrices to capture complex similarity structures; it is commonly used to initialize k-means. Hierarchical clustering approaches (Ward Jr, 1963) are also widely used in practice.

Model-based clustering: methods and guarantees. From a theoretical perspective, we focus on model-based clustering, where data are assumed to be generated from an explicit proba-

bilistic model with latent structure, such as mixture models or stochastic block models. In latent-partition models, one assumes that each data point X_i is generated from a distribution F_k, where the distribution F_k is associated with its cluster label k, and the goal is to recover the partition of the data into clusters. In mixture models, the data are generated from a mixture distribution $F = \sum_{k=1}^{K} w_k F_k$, where w_k is the weight of cluster k. The center of the k-th group is then defined as the expectation of the associated distribution $\Lambda_k = \mathbb{E}_{X \sim F_k}[X]$. A common example is Gaussian mixture models (GMMs), where F_k is a Gaussian distribution with mean Λ_k and covariance Σ_k. In Stochastic Block Models (SBM), the data are represented as a graph, where edges are generated according to probabilities that depend on the cluster of the nodes (Abbe, 2018). Early works showed that mixtures of Gaussian variables can be learned under suitable separation conditions using spectral methods or method-of-moments techniques (Dasgupta, 1999; Vempala and Wang, 2002; Regev and Vijayaraghavan, 2017). More recent contributions obtained minimax characterizations of the estimation and clustering error in high-dimensional mixtures, clarifying how the separation between components and the ambient dimension jointly govern the difficulty of the problem (Azizyan et al., 2013; Ndaoud, 2022). On the algorithmic side, there is now a detailed understanding of when standard procedures like Lloyd's algorithm or Expectation-Maximization (EM) are statistically and computationally optimal: under appropriate initialization and separation regimes, these algorithms converge rapidly to minimax-optimal solutions (Lu and Zhou, 2016; Segol and Nadler, 2021). For convex relaxation methods, Peng and Wei (2007); Giraud and Verzelen (2019) obtain exponential decay of the misclassification error for general sub-Gaussian mixture models.

Computation-information gaps. A series of works has established evidence of a computation-information gap for clustering, that is, a gap between the information-theoretically optimal error rate and the error achievable by polynomial-time algorithms. Such phenomena have been first pointed out in sparse PCA (Cai et al., 2013).

For example, assume an isotropic Gaussian mixture model, where $X_i \sim \mathcal{N}(\Lambda_k, \sigma^2 I_d)$ if the label of item i is k. For isotropic Gaussian mixtures in high dimension ($d \geqslant n$, balanced clusters), exact recovery is achievable via exhaustive search over partitions, at information-theoretically optimal separation condition

$$\Delta_*^2 \gtrsim \sigma^2 \sqrt{\frac{dK \log n}{n}},$$

—which is intractable for large n. Polynomial-time algorithms such as Lloyd's (Lu and Zhou, 2016), spectral clustering, or SDP relaxations (Giraud and Verzelen, 2019) require the stronger condition

$$\Delta_*^2 \gtrsim \sigma^2 \sqrt{\frac{dK^2}{n}}$$

in regimes where d is larger than n (Giraud and Verzelen, 2019). Strong evidence of such a gap was proved in Even et al. (2024) via a low-degree polynomial barrier, confirming a conjecture of Lesieur et al. (2016). Chapter 3 shows that such gap does not appear when the sampling scheme is adaptive and sequential. Indeed, we exhibit a polynomial-time algorithm achieving the information-theoretically optimal budget, effectively breaking the gap in the bandit setting.

Kernel clustering and distributional clustering. Non-parametric clustering approaches relax strong distributional assumptions by embedding distributions into reproducing kernel Hilbert spaces (RKHSs) through kernel mean embeddings (KMEs), and using the maximum mean discrepancy (MMD) as a distance between distributions in the RKHS distances (Sriperumbudur et al., 2010; Muandet et al., 2017). Characteristic kernels ensure that distinct distributions map to distinct KMEs, enabling two-sample tests and clustering guarantees without parametric structure (Gretton et al., 2012). Early kernel methods include kernel k-means, which applies k-means to KMEs for non-linearly separable clusters (Dhillon et al., 2004), and spectral clustering, which leverages the spectrum of affinity matrices (Ng et al., 2001). Recent theoretical advances provide recovery rates for non-parametric mixture models under KME separation, as well as optimality of kernel choice for high-dimensional clustering (Vankadara and Ghoshdastidar, 2020; Vankadara et al., 2021).

Other unsupervised problems. Beyond clustering problems, many unsupervised learning tasks fall under the general umbrella of latent-structure recovery. There is a general interest in high-dimensional models, which is of general relevance in modern applications— see Giraud (2021); Rigollet and Hütter (2023). Among other classical unsupervised learning problems, we mention community detection, change-point detection, multiple testing, and ranking.

In change-point detection problems, one observes a sequence $X_1, \dots, X_n$ of points in $\mathbb{R}^d$ (where i indexes, for instance, time) and aims to detect points where the statistical properties of the sequence change (Aminikhanghahi and Cook, 2017; Niu et al., 2016). One classical assumption is that the sequence is piecewise constant, with m change points at unknown locations $1 \leqslant \tau_1 < \cdots < \tau_m \leqslant n$, so that $\mathbb{E}[X_i]$ is constant between change points, and the goal is to recover these change points. Typical techniques include CUSUM statistics (Hinkley, 1970), binary segmentation (Fryzlewicz, 2014), and penalized least-squares (Verzelen et al., 2023). In high-dimensional settings, the change points are often assumed to be sparse, meaning that only a small subset of features change at each change point (Wang and Samworth, 2018; Liu et al., 2021; Pilliat et al., 2024).

Unsupervised learning in matrix games. Assume now that we observe data organized in a matrix of size $n \times n$, $(X_{i,j})_{1 \leqslant i,j \leqslant n}$, where the observations are pairwise comparisons between items. In stochastic block models, the goal is to recover a partition of the items into clusters, where two items belong to the same cluster if and only if they share the same distribution of comparisons with other items (Abbe, 2018). In ranking problems, a particular case of SST (Strong Stochastic Transitivity), the goal is to recover a latent ordering over the items, such that after reordering, the latent matrix exhibits a monotonic structure (Shah and Wainwright, 2018; Pilliat et al., 2024), and in tournament settings, one seeks to recover some notion of "best" item, such as the Borda (Heckel et al., 2019) or Condorcet winners.

This broad family of latent-structure models motivates the matrix framework we adopt in this thesis, which encompasses structure recovery problems under noisy, sequential, and adaptive observations.

2.3.2 Pure exploration in bandit models

Multi-armed bandits. The multi-armed bandit model, introduced by Thompson (1933) and formalized by (Robbins, 1952), models decision-making under uncertainty, where a learner sequentially selects actions (arms) and receives feedback (rewards) from the environment. Typically, it involves an environment $\nu = (\nu_1, \dots, \nu_n)$, where ν_i denotes the distribution of the one-dimensional feedback received when the learner selects arm i. Sequentially, the learner chooses some $A_t \in [n]$ and observes $X_t \sim \nu_{A_t}$. The historical focus is on the regret setting, where the goal is to maximize cumulative expected reward over time $\sum_{t=1}^{T} \mathbb{E}[X_t]$ (or equivalently minimize regret) (see Lattimore and Szepesvári (2020); Slivkins (2019) for comprehensive reviews). A key feature of the bandit model is that, using past observations, the learner can adapt her learning strategy on the fly. In the regret framework, the challenge is balancing exploration, i.e., collecting information about the environment, and exploitation, i.e., using the collected information to maximize reward. Applications include recommendation systems (Ariu et al., 2020), health (Tewari and Murphy, 2017), and A/B testing (Kaufmann et al., 2014). This work focuses on the pure exploration setting, where the learner's goal is to identify some latent structure of the environment with high confidence, rather than maximize cumulative reward.

Best arm identification. Best arm identification (BAI) (Audibert and Bubeck, 2010; Bubeck et al., 2009) is the canonical pure exploration problem in multi-armed bandits, where the learner aims to identify the arm with the largest mean reward. This is motivated by the fact that, in some applications, the sampling budget is the scarce resource (Auer et al., 2002; Bubeck et al., 2009). Two main variants have received extensive attention over the past two decades: fixed budget and fixed confidence settings.

In the fixed budget setting, the learner is given a fixed number of samples T (also called budget), and aims at maximizing the probability of correctly identifying the best arm (Audibert and Bubeck, 2010; Wang et al., 2023). Successful approaches include elimination strategies (Audibert and Bubeck, 2010; Even-Dar et al., 2006; Karnin et al., 2013a) and UCB-based methods (Jamieson et al., 2014; Katz-Samuels and Jamieson, 2020a) (see the survey Jamieson and Nowak (2014)). Sequential Halving (Karnin et al., 2013a) stands out for its simplicity and efficiency, including for ε-best arm identification (Zhao et al., 2023). Sub-sampling (Jamieson et al., 2016) and bracketing (Zhao et al., 2023) techniques are crucial, especially in the presence of many good arms (De Heide et al., 2021; Carpentier and Valko, 2015). Tight lower bounds reveal an inherent cost of adaptation (Carpentier and Locatelli, 2016), and the optimality up to constant of these methods is well-understood.

In the fixed confidence (PAC) setting, given error probability δ, the objective is to minimize the sample complexity $\mathcal{T}$ while ensuring correct identification with probability $\geq 1 - \delta$ (Garivier and Kaufmann, 2016). Asymptotically ($\delta \to 0$), the optimal complexity (with exact constants) is reached via Track-and-Stop strategies, which solve an information-theoretic allocation problem (Kaufmann et al., 2016; Garivier and Kaufmann, 2016; Degenne and Koolen, 2019). Non-asymptotic refinements appear in (Degenne et al., 2019; Simchowitz et al., 2017).

Other pure exploration bandit problems. Direct variants of best arm identification include ε-good arm identification (Zhao et al., 2023) where the learner seeks to identify an arm whose mean is within ε of the best, or top-k identification (Kalyanakrishnan et al., 2012), where the goal is to identify the k-largest arms. More generally, pure exploration bandits encompass a wide range of problems, including thresholding bandits (Locatelli et al., 2016; Ariu et al., 2022), coarse ranking (Karpov and Zhang, 2020), and general combinatorial pure exploration (Chen et al., 2014). One may also identify other functionals of the environment (Baharav and Tse, 2019; Chaudhuri and Kalyanakrishnan, 2017). One line of work that is particularly important for this thesis is adaptive signal detection, where a learner tries to detect signal in a sparse vector through sequential and adaptive queries (Castro, 2014). Our work borrows techniques from this literature to derive optimal bounds for unsupervised active learning problems.

Clustering with bandit feedback. Clustering with bandit feedback was first introduced in (Yang et al., 2024). In this problem, the objective is to recover a partition of the set of arms into clusters, where two arms (or items) belong to the same cluster if and only if they share the same distribution. Interestingly, this setting is well-defined even when the distributions of the arms are multidimensional, bringing high-dimensional challenges to bandit models. In Yang et al. (2024), the authors propose the BOC algorithm based on the Track-and-Stop strategy, obtaining an asymptotically optimal sample complexity in the fixed-confidence setting, in the asymptotic regime where the probability of error goes to 0. Another strategy inspired by an information-theoretic approach is also considered in Ariu et al. (2024), which encompasses a more complex setting but does not provide theoretical guarantees. Other works tackled more general distribution-matching problems with Track-and-Stop approaches (Yavas et al., 2025; Chandran et al., 2025). Our work in Chapter 3 completes these works by providing a non-asymptotic analysis of the sample complexity for this problem, which is of practical relevance in high-dimensional settings. We extend this setting to non-parametric clustering in Chapter 4.

Pure exploration in matrix bandit environments. The first matrix bandit setting studied in the literature is the dueling bandit model with pairwise comparisons, where the learner only accesses the environment through comparisons between pairs of items (Bengs et al., 2021). In dueling bandits, the learner queries, at each time, a pair of items and observes the outcome of their comparison, which can be represented as a matrix $Q = (Q_{i,j})_{(i,j)}$. The quantity $Q_{i,j}$ is the probability that item i is preferred to item j. In this context, the notion of a "best arm" is ambiguous, and different winners have been considered, such as the Borda winner (Jamieson et al., 2015), the Copeland winner (Zoghi et al., 2015a), or the Condorcet winner—see Bengs et al. (2021) for a survey. Other works have considered related objectives, such as active ranking (Jamieson and Nowak, 2011; Saad et al., 2023), or the identification of a Nash equilibrium in a matrix game (Maiti, 2025).

In Condorcet Winner Identification, the objective is to identify the item that is preferred to all others, when it exists. Prior work has mainly focused on studying the budget in expectation, and in asymptotic regimes. The procedure of Karnin (2016) achieves, asymptotically,

$\lim_{\delta\to 0} \frac{\mathbb{E}[\mathcal{T}_\delta]}{\log(1/\delta)} \leq c \sum_{i\neq i^*} \min_{j:\,\Delta_{i,j}<0} \frac{1}{\Delta^2_{i,j}}$, corresponding to the cost of certifying one negative entry per suboptimal row when the best opponent is known. More recently, Maiti et al. (2024) obtained a high-probability upper bound of order $H_{\mathrm{cw}}(\delta) := \log(1/\delta) \sum_{i\neq i^*} \frac{1}{\Delta^2_{i^*,i}}$, which depends only on the gaps with the Condorcet winner. This follows a long line of work which focuses on the gaps with the Condorcet winner, and which is optimal when the best opponent of each suboptimal item is the Condorcet winner (Maiti, 2025; Bengs et al., 2021; Haddenhorst et al., 2021a). Our work in Chapter 6 shows that this is not always the case, and that the true sample complexity can be much smaller than $H_{\mathrm{cw}}(\delta)$, depending on the structure of the gap matrix.

2.4 Guiding questions

This stream of literature on unsupervised learning and pure exploration in bandit models motivates the following questions, which guide the contributions of this thesis.

Question 1. *How can we efficiently recover unknown structure in bandit matrix problems?*

A central part of this thesis is devoted to designing algorithms for structure recovery in matrix bandit problems and analyzing their sample complexity. Our algorithms share a common principle: they balance an exploration/detection phase, where the learner identifies informative entries, and a reconstruction/certification phase, where the learner recovers the unknown structure and certifies its estimate. We design efficient strategies for both phases by exploiting problem-specific structure, using concentration inequalities, sub-sampling, doubling schedules, sequential elimination, and hypothesis-test aggregation. We establish non-asymptotic guarantees both in expectation and in high probability.

Question 2. *What are the unavoidable costs of structure recovery?*

For each problem, we derive information-theoretic lower bounds on the sample complexity of any algorithm, providing a benchmark for algorithmic optimality. Our proof strategy is to construct hard environments in which structure recovery reduces to classical high-dimensional testing problems (two-sample tests, signal detection), from which we apply standard information-theoretic tools such as Fano's inequality or data-processing inequalities. We also establish lower bounds on high-probability quantiles of the sampling budget, revealing intrinsic new limits of pure exploration that go beyond expectation bounds.

Question 3. *What are the regimes where non-asymptotic analysis provides new insights?*

We develop a non-asymptotic analysis, characterizing the optimal sample complexity across all parameter regimes, and any confidence level δ. While optimal complexity is often well-understood asymptotically ($\delta \to 0$), our analysis uncovers the precise dependence on all parameters. This is particularly relevant for the emblematic high-dimensional (large d) and large-scale (large n) regimes, where we reveal previously hidden effects.

Question 4. *What are the benefits of bandit learning compared to batch learning?*

We compare adaptive bandit strategies with batch (passive) baselines. First, we identify regimes in which adaptation yields strictly smaller sample complexity, and we quantify these gains precisely. Second, we highlight computational benefits of bandit strategies. In particular, for clustering, we show that bandit feedback can break the computation-information gap present in the batch setting, allowing polynomial-time algorithms to reach information-theoretically optimal sample complexity.

2.5 Motivating example

We present an example that illustrates the questions raised above. This example is closely related to the problems studied in this thesis, both in its objective and in the methodology used to solve it, as well as in the intrinsic limits that can be derived. We frame it as a signal-detection problem, through the lens of balancing exploration (finding the signal) and certification (verifying it with high confidence). This example, which fundamentally differs from classical best-arm identification problems, serves as a guide for interpreting our contributions.

Before we detail the link between this example and the problems studied in this thesis, we present it as a problem of adaptive signal detection, inspired by the work of Castro (2014).

Adaptive signal detection, and support recovery. We consider a d-armed bandit associated with an environment ν_S with mean $\mu \in \mathbb{R}^d$. Consider a sparse setting where the vector μ is s-sparse, with a support $S = \{i : \mu_i \neq 0\}$ of size $s = |S|$. We assume that $s > 0$, so that the support is non-empty. We assume there is a fixed value $\Delta > 0$, so that all non-zero entries are equal to Δ. The learner has access to the distribution ν_S through sequential and adaptive sampling: at each time t, the learner chooses a feature $J_t \in [d]$ and observes $X_t \sim \mathcal{N}(\mu_{J_t}, \sigma^2)$, with Gaussian noise.

We discuss two problems in the fixed confidence setting. Given a prescribed confidence level $\delta \in (0, 1)$:

(i) signal detection: select $\hat{\imath} \in S$, such that $\mathbb{P}_{\nu_S}(\mu_{\hat{\imath}} \neq 0) \geq 1 - \delta$.
The goal is to select any entry of μ that is non-zero, with probability at least $1 - \delta$.

(ii) full support recovery: estimate $\hat{S} \subseteq [d]$, such that $\mathbb{P}_{\nu_S}(\hat{S} = S) \geq 1 - \delta$.
The goal is to recover the full support of μ, with probability at least $1 - \delta$.

Observe that in this formulation, the learner does not know the sparsity level s, the signal strength Δ, or the support S, but knows that the signal is structured as a sparse vector with non-zero entries of equal magnitude.

2.5.1 Problem (ii): full support recovery

We start by discussing the optimal sample complexity of support recovery (problem (ii)), which is well understood and closely related to classical pure-exploration problems.

Optimal sample complexity for support recovery. Problem (ii) is related to Top-s identification (Bubeck et al., 2013) and thresholding bandits (Chen et al., 2014), depending on which

parameters are known. If s is known, the task reduces to Top-s identification, where the goal is to identify the s largest arms. If s is unknown but Δ is known, the task can be cast as thresholding, where the task is to identify all arms above the threshold $\Delta/2$. In both formulations, the optimal sample complexity is of order $\frac{d\sigma^2}{\Delta^2}\log(1/\delta)$, up to logarithmic factors in d, s, and Δ^{-2}. See, for instance, Theorem 6 in Kalyanakrishnan et al. (2012), and also Castro (2014); Locatelli et al. (2016).

This complexity can be interpreted as the cost of testing each feature while controlling the global error rate at level δ. Interestingly, this mirrors the general Best Arm Identification (BAI) principle: one must verify each candidate before certifying the final answer.

Link with the problems studied in this thesis. Consider problem (*ii*) with an environment with n entries.

— Consider CBP (Chapter 3), where the goal is to recover a partition of n items into K clusters. When there are two clusters ($K = 2$), and the dimension of the feature space is $d = 1$, it reduces to support recovery, where the support corresponds to a cluster of items with mean Δ, and the complement corresponds to a cluster of items with mean 0. In this case, the optimal sample complexity is of order $\frac{n\sigma^2}{\Delta^2}\log(1/\delta)$.

— For CBP with feature selection (Chapter 3), imagine that the learner has discovered a feature $j \in [d]$ where the gap between the two clusters is $\Delta_j \neq 0$, and wants to cluster all items based on this feature. Then, the clustering task reduces to support recovery in the exact form of problem (*ii*), where μ is the j-th column of the feature matrix M.

— For Condorcet Winner Identification, imagine that the learner has i^* as a candidate for the Condorcet winner, and wants to verify that this is indeed the case by comparing i^* with all other items. This verification step is analogous to support recovery, where the goal is to confirm the positivity of all non-diagonal entries of the CW row.

Overall, problem (ii) is a recurring problem in this thesis, in situations where the learner must certify the correctness of a candidate answer by verifying multiple entries, while controlling the global error rate at level δ.

2.5.2 Problem (i): signal detection

We now discuss in detail Problem (i), for which we establish the optimal sample complexity. This problem is more subtle than support recovery, and it reveals new insights into the intrinsic limits of adaptive strategies for structured bandit problems. While the problem of signal detection in the sparse vector model consists in finding a best arm, it differs from classical BAI problems by the presence of multiple best arms, and by the fact that the mean vector is structured as a sparse vector with non-zero entries of equal magnitude. This structure allows for new strategies, and also new lower bounds, which are not captured by classical BAI analyses.

2.5.2.1 Link to this thesis.

In the next paragraph, we derive the optimal sample complexity for problem (i) using a methodology close to the one developed throughout this thesis. Beyond this methodological similarity, problem (i) appears directly in several chapters, either as a sub-problem or as a building block for our algorithms. For instance, consider the following situations:

— In CBP with feature selection (Chapter 5), suppose the learner has identified two items i and j from different clusters and now needs a feature that separates them. With these two representatives, the learner can sample the gap vector $\Delta = \mu_1 - \mu_2 \in \mathbb{R}^d$. Finding a discriminative feature then reduces to problem (i).

— In Chapters 3 and 5, the first stage of the algorithms is to identify cluster representatives; this stage can be framed as a signal-detection task analogous to problem (i).

— In Condorcet Winner Identification, a key step is to find, for each suboptimal item $i \neq i^*$, an opponent j such that $\Delta_{i,j} < 0$. This search step is analogous to problem (i).

— In change-point detection (Chapter 7), an initial step is to locate points in each plateau of the piecewise-constant function f. This is also closely connected to a signal-detection problem, where the signal is the gap between two plateaus.

Observe that, in all these applications, the signal vector μ is not necessarily structured as a two-valued sparse vector, and may be more general. Our analysis highlights that the simple sparse case already captures the core difficulty of these problems. Overall, although it takes problem-specific forms, the signal-detection task in problem (i) is a recurring sub-problem in this thesis. It appears in the exploration phase, where the learner must first detect informative entries of the environment before exploiting them to recover the underlying structure and certify correctness.

2.5.2.2 Intrinsic limits.

Let π be a δ-correct algorithm for Problem (i), meaning that for every environment ν_S with $\Delta > 0$ and $\emptyset \neq S \subseteq [d]$, $\mathbb{P}_{\pi,\nu_S}(\mu_{\hat{\imath}} \neq 0) \geq 1 - \delta$.

Proposition 2.5.1. *Assume $\delta \leqslant 1/6e$. Consider any δ-correct algorithm π solving Problem (i), and fix $\Delta \neq 0$ and $0 < s < d$. Then, the worst-case budget of π on environments ν_S with $|S| = s$ is lower bounded by*

$$\sup_{S:|S|=s} \mathbb{E}_{\pi,\nu_S}[\mathcal{T}] \;\geq\; \frac{1}{6e}\frac{d\sigma^2}{s\Delta^2} + \frac{\sigma^2}{\Delta^2}\log(1/4\delta)\;, \quad \textit{and} \quad \sup_{S:|S|=s} \mathbb{P}_{\pi,\nu_S}\left[\mathcal{T} \;\geq\; 2\frac{d\sigma^2}{s\Delta^2}\log(1/6\delta)\right] \geqslant \delta\;.$$

Sketch of the proof. We sketch the proof of both claims and defer full details to Appendix 2.A.

First, the term $\frac{\sigma^2}{\Delta^2}\log(1/\delta)$ in the expectation bound corresponds to the cost of certifying that a candidate feature is truly non-zero. For any support S, we compare ν_S and ν_{S^c}, which have disjoint supports. Since π is δ-correct, the event $\{\hat{\imath} \in S\}$ has probability at least $1-\delta$ under ν_S and at most δ under ν_{S^c}. A data-processing argument therefore gives $\mathrm{KL}(\mathbb{P}_{\pi,\nu_S}, \mathbb{P}_{\pi,\nu_{S^c}}) \gtrsim \log(1/\delta)$. By construction, this KL divergence is of order $\frac{\Delta^2}{\sigma^2}\,\mathbb{E}_{\pi,\nu_S}[\mathcal{T}]$, which implies $\mathbb{E}_{\pi,\nu_S}[\mathcal{T}] \gtrsim \frac{\sigma^2}{\Delta^2}\log(1/\delta)$. Intuitively, this is the classical one-dimensional testing cost between $\mathcal{N}(0,\sigma^2)$ and $\mathcal{N}(\Delta,\sigma^2)$.

Second, the quantile lower bound requires a different change-of-measure argument and captures the intrinsic cost of adaptation. Here, the key object is the stopping time $\mathcal{T}$ rather than the final decision $\hat{\imath}$. Define

$$\chi := \sup_{S:|S|=s} \inf\{t : \mathbb{P}_{\pi,\nu_S}(\mathcal{T} \le t) \ge 1-\delta\},$$

that is, the largest $(1-\delta)$-quantile of the budget over all supports of size s.
On the one hand, by definition of χ, the algorithm stops before χ with probability at least $1-\delta$ under ν_S, that is $\mathbb{P}_{\pi,\nu_S}(\mathcal{T} \le \chi) \ge 1-\delta$.
On the other hand, consider ν_0 as the environment with an empty support, in which the observation is distributed as $\mathcal{N}(0,\sigma^2)$. We prove that, with probability at least $1-2\delta$, the algorithm does not stop by χ, that is $\mathbb{P}_{\pi,\nu_0}(\mathcal{T} > \chi) \ge 1-2\delta$. Intuitively, this follows from the fact that under ν_0, there is no signal, so it should not certify any entry as containing signal in finite time.

Data processing then yields $\mathrm{KL}(\mathbb{P}_{\pi,\nu_0}, \mathbb{P}_{\pi,\nu_S}) \gtrsim \log(1/\delta)$, for every support S of size s. From there, a symmetrization argument [6] gives

$$\frac{1}{\binom{d}{s}} \sum_{S:|S|=s} \mathrm{KL}(\mathbb{P}_{\pi,\nu_0}, \mathbb{P}_{\pi,\nu_S}) = \frac{1}{\binom{d}{s}} \sum_{S:|S|=s} \sum_{i=1}^{d} \mathbb{E}_0[T_i] \mathbb{1}_{\{i\in S\}} \frac{\Delta^2}{2\sigma^2},$$

using that ν_0 and ν_S differ only on features in S, and that $\mathrm{KL}(\mathcal{N}(0,\sigma^2), \mathcal{N}(\Delta,\sigma^2)) = \frac{\Delta^2}{2\sigma^2}$. The main point is that $\mathbb{E}_0[T_i]$ does not depend on S, so we can swap the sums and obtain

$$\frac{1}{\binom{d}{s}} \sum_{S:|S|=s} \mathrm{KL}(\mathbb{P}_{\pi,\nu_0}, \mathbb{P}_{\pi,\nu_S}) = \frac{\Delta^2}{2\sigma^2} \sum_{i=1}^{d} \mathbb{E}_0[T_i] \frac{\binom{d-1}{s-1}}{\binom{d}{s}} \simeq \frac{\Delta^2}{\sigma^2} \mathbb{E}_0[T] \frac{s}{d}.$$

For interpretation, we can see that each sample on one of the s nonzero features contributes Δ^2/σ^2 to the global KL. Then, under ν_0 the algorithm must spread its effort almost uniformly (about $\mathcal{T}/d$ per feature). Finally, constraining the algorithm to verify $\mathbb{E}_0[T] \le \chi$ (which does not affect any of these arguments), we combine both inequalities and get $\chi \geqslant \frac{d\sigma^2}{s\Delta^2} \log(1/\delta)$, which is exactly the claimed quantile lower bound.

Coming back to the expected budget, this quantile bound at a constant confidence level (e.g., $\delta \simeq 1/2$) yields the expected-budget lower bound of order $\frac{d}{s\Delta^2}$, via a Chernoff-type argument. □

2.5.2.3 Optimal adaptive strategies.

Adaptive strategies can match these lower bounds up to constants and logarithmic factors, while adapting to unknown s and Δ. The key ingredients are: (i) Sequential Halving (SH) (Karnin et al., 2013a), (ii) sub-sampling through the variant of SH called Bracketing Sequential Halving (BSH) (Zhao et al., 2023), (iii) a doubling schedule, and (iv) a final verification test. A pseudocode for this procedure is given in Algorithm 1.

Proposition 2.5.2. *Let $\delta \in (0,1)$, then Algorithm 1 is δ-correct for Problem (i). Moreover, there exists polynomial factors l_1,l_2 in $\log(d\sigma^2\Delta^{-2})$, independent of δ, so that its sampling budget $\mathcal{T}$ satisfies,*

$$\mathbb{E}[\mathcal{T}] \leqslant l_1\left(\frac{d\sigma^2}{s\Delta^2} + \frac{\sigma^2}{\Delta^2}\log(1/\delta)\right) \qquad \textit{and,} \qquad \mathbb{P}\left[\mathcal{T} \leqslant l_2\left(\frac{d\sigma^2}{s\Delta^2}\log(1/\delta)\right)\right] \geqslant 1-\delta\ .$$

6. That is, averaging over all possible supports S of size s.

Algorithm 1: Adaptive signal detection (Informal)

$k \leftarrow 1$;
while *true* **do**
  Run BSH with a budget 2^k and output i_k;
  Sample i_k for 2^k additional pulls and compute $\widehat{\mu}_k$;
  if $\widehat{\mu}_k \geq \sqrt{2\sigma^2 \frac{\log(12k^2/\pi^2\delta)}{2^k}}$ **then**
    Output $\hat{\imath} \leftarrow i_k$ and stop;
  end
  $k \leftarrow k+1$;
end

Sketch of the analysis. The proof is postponed to Appendix 2.B, but we sketch the main arguments here.

We rely on Theorem 6 of Zhao et al. (2023), which provides guarantees for BSH. Up to logarithmic factors, if $T \gtrsim \frac{d\sigma^2}{s\Delta^2}\log(1/\delta)$, then BSH returns a best-arm with probability at least $1-\delta$. Intuitively, two effects are combined:

1. certifying a good arm requires about $\frac{\sigma^2}{\Delta^2}\log(1/\delta)$ samples for reliable testing;
2. sub-sampling introduces the multiplicative search cost $\frac{d}{s}$ necessary to hit an arm in S.

The sub-sampling mechanism of BSH, and sequential elimination, are crucial to achieve this search cost, which is optimal as shown by the lower bound. The verification threshold follows from Hoeffding's inequality. The factor $\log(k^2)$ enables a union bound over epochs, so if BSH outputs i_k with $\mu_{i_k} = \Delta$, verification succeeds as long as $2^k \gtrsim \frac{\sigma^2}{\Delta^2}\log(k^2/\delta)$, and if $\mu_{i_k} = 0$, verification discards the wrong candidate i_k with high probability, whatever the epoch.

For the quantile-budget bound, we exploit the high-probability guarantee of BSH. Define k^* as the first epoch such that $2^{k^*} \asymp \frac{d\sigma^2}{s\Delta^2}\log(1/\delta)$, up to logarithmic factors. At this scale, BSH and verification both succeed with high probability, so the algorithm stops by epoch k^*. Since $\sum_{k'=1}^{k^*} 2^{k'} \leq 2^{k^*+1}$, the budget is of order $\frac{d\sigma^2}{s\Delta^2}\log(1/\delta)$.

For the expected budget, for any epoch $k \geqslant k_0$ such that $2^{k_0} \simeq \frac{d\sigma^2}{s\Delta^2} + \frac{\sigma^2}{\Delta^2}\log(1/\delta)$, the stopping probability is bounded below by a positive constant. Hence, the number of additional epochs after k_0 is dominated by a geometric random variable, which implies $\mathbb{E}[\mathcal{T}] \lesssim \frac{d\sigma^2}{s\Delta^2} + \frac{\sigma^2}{\Delta^2}\log(1/\delta)$, matching the lower bound up to logarithmic factors. □

2.5.3 Observations.

We formulate observations on this example, inspired by our guiding questions (Section 2.4).

First, for signal detection (i), the expected complexity $\frac{d\sigma^2}{s\Delta^2} + \frac{\sigma^2}{\Delta^2}\log(1/\delta)$ splits into a detection cost $\frac{d\sigma^2}{s\Delta^2}$ (independent of δ) and a verification cost $\frac{\sigma^2}{\Delta^2}\log(1/\delta)$. The detection term dominates at moderate confidence, while the verification term dominates at high confidence. This detection/verification trade-off is a recurring motif in our problems, while asymptotic ($\delta \to 0$) analyses

emphasize the second term and can miss the first. This motivates our non-asymptotic analysis, which is particularly relevant in high-dimensional settings.

Second, expected and high-probability guarantees have intrinsically different scales. In high probability, detection contributes an extra $\log(1/\delta)$ factor, leading to a cost of order $\frac{d\sigma^2}{s\Delta^2}\log(1/\delta)$. This mirrors the fixed-confidence/fixed-budget gap already observed in best-arm identification (Locatelli et al., 2016). A similar phenomenon appears in most of the structure recovery problems that we study; this is a new contribution of this thesis.

Finally, the benefits of adaptive strategies are clear in this example. A non-adaptive strategy that allocates T/d samples per feature would require $T \gtrsim \frac{d\sigma^2}{\Delta^2}\log(1/\delta)$ to succeed, which is sub-optimal by a factor of s. Moreover, passive strategies are not able to choose the budget T adaptively. This illustrates the benefits of adaptation, and the importance of sub-sampling and sequential-elimination techniques to achieve optimal sample complexity. Moreover, the optimal adaptive strategy is computationally efficient. This is a key point for problems such as clustering, where the information-theoretic optimal sample complexity is often not achievable by polynomial-time algorithms in the batch setting, while it becomes achievable with bandit feedback.

2.6 Outline of contributions

2.6.1 Parametric Clustering with Bandit Feedback (Chapter 3)

Chapter 3 is a joint work with Alexandra Carpentier [7], Christophe Giraud [8], and Nicolas Verzelen [9], published in ALT 2025 as (Thuot et al., 2025).

Problem setting. Chapter 3 studies the parametric Clustering with Bandit Feedback (CBP), as described in Section 2.2. In this problem, each item $a \in [n]$ has an unknown mean vector $\mu_a \in \mathbb{R}^d$, and two items $a, b \in [n]$ are in the same cluster if and only if $\mu_a = \mu_b$. The hardness of parametric CBP is governed, beyond n, K, d, and σ^2, by the minimal cluster separation and the proportion of items in the smallest cluster:

$$\Delta_* = \min_{k\neq\ell} \|\mu_k - \mu_\ell\|_2 \qquad \text{and} \qquad \theta_* = \min_{k\in[K]} \frac{|C_k|}{n} \in \Big[\frac{1}{n}, \frac{1}{K}\Big].$$

These two quantities characterize the difficulty of the problem: it becomes harder when clusters are less separated (Δ_* small) or when groups are more unbalanced (θ_* small).

Lower bound. We derive a non-asymptotic lower bound on the minimal expected budget required by any δ–correct algorithm over a class of environments with minimal separation at least Δ_*, balancedness at least θ_*, and σ^2–sub-Gaussian noise. In the balanced case $\theta_* \approx 1/K$,

$$\mathbb{E}[\mathcal{T}] \gtrsim n + \frac{\sigma^2}{\Delta_*^2}\left[n\log\left(\frac{n}{\delta}\right) + \sqrt{dKn\log\left(\frac{n}{\delta}\right)}\right],$$

7. Institut für Mathematik, Universität Potsdam, Potsdam, Germany.
8. Université Paris-Saclay, Laboratoire de mathématiques d'Orsay, Orsay, France.
9. INRAE, Mistea, Institut Agro, Univ Montpellier, Montpellier, France.

up to universal constants— see Theorem 3.3.1.

The first term n is the unavoidable cost of observing each arm at least once. The second term $n\frac{\sigma^2}{\Delta_*^2}\log(n/\delta)$ corresponds to the cost of certifying that each arm belongs to the correct cluster, and appears even when the cluster centers are known. This term dominates in low-dimensional regimes, where d is small compared to n, and is optimal in the asymptotic regime $\delta \to 0$ (Yang et al., 2024). It is similar to the budget necessary for support recovery in our motivating example (Problem (ii) in Section 2.5). The third term $\frac{\sigma^2}{\Delta_*^2}\sqrt{dKn\log(n/\delta)}$ captures the additional cost of learning the cluster centers in high dimensions, and dominates in high-dimensional regimes. This last term, depending on d, is new compared to asymptotic analyses such as (Yang et al., 2024). Our proof requires a careful reduction from bandit clustering to high-dimensional two-sample testing, which is a novel contribution of this chapter.

Upper bound and optimality. We design the ACB algorithm (Algorithm 4), a polynomial-time procedure that is δ–correct. Chapter 3 proves that ACB achieves an expected budget matching the information-theoretic lower bound up to logarithmic factors for all $(n, K, d, \Delta_*, \theta_*, \delta)$ in the considered class, while remaining computationally efficient— see Theorem 3.C.1. ACB follows a two-step strategy:

1. Sequential Representative Identification (Algorithm 2), which adaptively identifies a set of K representative arms containing, with high probability, exactly one arm from each cluster. This phase relies on adaptive sub-sampling and a carefully chosen sequence of high-dimensional two-sample tests.
2. Active Distance-based Classification (Algorithm 3), which concentrates samples on the K representatives to obtain accurate estimates of the cluster centers, and then classifies every remaining arm by comparing its empirical mean to these estimated centers.

Comparison with batch clustering. The CBP is the sequential, adaptive counterpart of classical model-based clustering. A natural comparison is uniform sampling followed by classical batch clustering. If the learner has a fixed budget T, allocating T/n observations per arm reduces the variance to $n\sigma^2/T$, and the separation thresholds from Section 2.3.1 translate directly into budget requirements. This uniform sampling strategy serves as a baseline for evaluating the benefits of adaptive bandit feedback. For instance, when $d \geqslant n$, the noise is isotropic, and the clusters are balanced ($\theta_* \approx 1/K$), uniform sampling requires a budget of order $\frac{\sigma^2}{\Delta_*^2}\sqrt{dK^2n}$ to succeed with a non-trivial probability of error (Even et al., 2024), which is sub-optimal by a factor of $\sqrt{K}$ compared to the lower bound $\sqrt{dKn\log(n)}$. [10] This gap is inherited from the computation-information gap in batch clustering, as discussed in 2.3.2.

The main message of Chapter 3 is that adaptive sampling can turn a statistically feasible but computationally hard clustering problem into one that is both sample-efficient and polynomial-time solvable. More broadly, it provides a concrete example in which bandit feedback breaks a computational barrier present in the corresponding batch problem.

10. For the sake of comparison, we omit all $\log(1/\delta)$ terms here; this discussion is therefore valid in the moderate-confidence regime $\delta \simeq$ const.

Chapter 3 **Summary of contributions**

1. We formalize the Clustering with Bandit Feedback Problem (CBP) under sub-Gaussian noise, in a parametric setting, and identify the key quantities $(\Delta_*, \sigma^2, K, d, n, \theta_*)$ characterizing its complexity.
2. We establish non-asymptotic lower bounds on the minimal budget required by any δ–correct algorithm, thereby distinguishing low-dimensional and high-dimensional regimes and showing that the optimal complexity scales as
$$n + \frac{\sigma^2}{\Delta_*^2}\left[n \log\left(\frac{n}{\delta}\right) + \sqrt{dKn \log\left(\frac{n}{\delta}\right)}\right] \quad .$$
3. We introduce the ACB algorithm, a polynomial-time procedure based on the selection of representative arms, and prove that ACB is δ–correct and attains the lower bound up to logarithmic factors in the balanced regimes.
4. We show that, in the bandit setting, the computation–information gap disappears.

2.6.2 Non-Parametric Clustering with Bandit Feedback (Chapter 4)

Chapter 4 is a joint work with Sebastian Vogt [11], Debarghya Ghoshdastidar [12], and Nicolas Verzelen [13], available as a preprint (Thuot et al., 2026).

Non-parametric setting. Chapter 3 relies on a structured parametric setting with linear-separable clusters and simple geometry (e.g., isotropic Gaussian). Chapter 4 extends the problem to non-parametric clustering with more complex geometries (see Section 2.2), where arms are clustered by their underlying distributions rather than finite-dimensional mean vectors. Each ν_i is supported on a separable topological space $\mathcal{X}$ (possibly infinite-dimensional), and we impose no parametric assumption beyond mild kernel conditions.

Kernel and Meta-algorithm. We adopt a kernel-based approach (Gretton et al., 2012; Muandet et al., 2017; Wolfer and Alquier, 2025). We consider the embedding of each distribution ν_i into a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ through kernel mean embedding (KME) $\mu_i = \mathbb{E}_{X\sim\nu_i}[g(X, \cdot)]$, where g is a bounded, translation-invariant, characteristic kernel on $\mathcal{X} \times \mathcal{X}$. Under the characteristic assumption, $\nu_i = \nu_j$ is equivalent to $\mu_i = \mu_j$, so the non-parametric CBP reduces to clustering arms according to their KMEs in $\mathcal{H}$. We measure separation between clusters through the squared maximum mean discrepancy, the squared distance between the corresponding KMEs in $\mathcal{H}$, $\mathrm{MMD}^2(\nu_i, \nu_j) = \|\mu_i - \mu_j\|_{\mathcal{H}}^2$.

11. Equal contribution—Technical University of Munich, Munich, Germany.
12. Technical University of Munich, Munich, Germany.
13. INRAE, Mistea, Institut Agro, Univ Montpellier, Montpellier, France.

Upper bound. To capture instance difficulty, we introduce the chapter-specific signal-to-noise ratio s_*^2, which depends jointly on inter-cluster MMD separation and RKHS variances. It is defined by

$$s_*^{-2}(\nu) := \max_{\substack{i \neq j \in [n] \\ \mu_i \neq \mu_j}} \left(\frac{\mathcal{V}_i^* \vee \mathcal{V}_j^*}{\|\mu_i - \mu_j\|_{\mathcal{H}}^2} \vee \frac{\sqrt{\bar{g}}}{2\|\mu_i - \mu_j\|_{\mathcal{H}}} \right) ,$$

where $\mathcal{V}_i^*$ is the RKHS variance proxy associated with arm i (defined in Chapter 4), and $\bar{g}$ is an upper bound on the kernel g. Theorem 4.3.1 shows that our algorithm KACB (Algorithm 9) reaches a budget upper bounded with probability $1 - \delta$ by

$$\mathcal{T} \lesssim n\, s_*^{-2} \log\left(\frac{n}{\delta}\right) .$$

This bound contains two terms. The first term $n \frac{\mathcal{V}_i^*}{\|\mu_i - \mu_j\|_{\mathcal{H}}^2} \log(n/\delta)$ is a kernel equivalent of the second term in the parametric setting ($n \frac{\sigma^2}{\Delta^2} \log(1/\delta)$), and dominates when the variance is large. The second term $n \sqrt{\bar{g}} / \|\mu_i - \mu_j\|_{\mathcal{H}}$ is a limitation depending on the kernel upper bound $\bar{g}$, so that taking the kernel variance into account only improves the bound when the variance $\mathcal{V}_i^*$ is large. The algorithmic idea behind KACB is deliberately simple. We sample all arms uniformly, run a kernel two-sample test for each pair (i, j), and connect i and j whenever the test does not reject the null hypothesis $\nu_i = \nu_j$. We use variance-aware concentration inequalities for empirical KMEs (Tolstikhin et al., 2016; Wolfer and Alquier, 2025) to control deviations of empirical MMD statistics. The estimated clustering is then the set of connected components of the resulting graph on $\{1, \ldots, n\}$. The per-arm budget is increased adaptively through a doubling scheme until pairwise tests are accurate enough to separate clusters reliably.

Chapter 4 **Summary of contributions**

1. We formalize a non-parametric version of the CBP, where arms are clustered according to their distributions, and reframe the task as clustering KMEs in an RKHS.
2. We introduce the Kernel Active Clustering with Bandit algorithm (Algorithm 9), a simple meta-procedure that relies on variance-aware kernel two-sample tests and uniform sampling. KACB is shown to be δ-correct and to achieve a total budget

   $$\mathcal{T} \lesssim n\, s_*^{-2} \log\left(\frac{n}{\delta}\right),$$

   up to logarithmic factors, where s_*^2 is an instance-dependent complexity parameter driven by minimal MMD separation and RKHS variances.
3. We establish that meaningful non-asymptotic guarantees remain possible beyond the parametric setting of Chapter 3, thereby extending the bandit-clustering setting to a broader class of distributional clustering problems.

Limitations In contrast to the parametric ACB algorithm of Chapter 3, this procedure is more naïve. Its main strength is robustness: it is δ-correct under much weaker structural assumptions. Our analysis is limited by the absence of corresponding lower bounds, which would be needed to fully characterize the optimal sample complexity, and remains an important open question.

2.6.3 Feature Selection in Clustering with Bandit Feedback (Chapter 5)

Chapter 5 is a joint work with Maximilian Graf [14], and Nicolas Verzelen [15], published in ICML 2025 (Graf et al., 2025).

Feature selection variant. In Chapter 5, we study CBP in the feature-selection variant. In addition to choosing an item, the learner chooses one feature at each round. At time t, she selects an item-feature pair $(I_t, J_t) \in [n] \times [d]$ and receives a noisy observation of M_{I_t,J_t} with sub-Gaussian noise. We focus on the two-cluster case ($K = 2$), where the two groups have mean vectors $\mu_0, \mu_1 \in \mathbb{R}^d$, and we denote the gap vector by $\Delta = \mu_1 - \mu_0$.

Upper bound. We introduce a fully adaptive algorithm, `BanditClustering` (Algorithm 13), built from two sequential subroutines. Both steps rely on an adaptation of Sequential Halving with sub-sampling, used to balance detection (finding informative features) and classification (classifying all items once a good feature is found).

1. `CandidateRow` (Algorithm 11) identifies a representative item from each cluster with high probability. This step is seen as a specific instance of the signal detection problem (Problem (i), Section 2.5), and is performed through a combination of sub-sampling methods, together with the elimination strategy Sequential Halving from Karnin et al. (2013a).
2. `ClusterByCandidates` (Algorithm 12) identifies a strong discriminative feature and classifies all items using that feature. This is performed through a doubling schedule, which adaptively increases the sampling budget to detect a feature informative enough to separate clusters with high confidence.

The resulting non-asymptotic upper bound is instance-dependent— see Theorem 5.3.1. Up to logarithmic factors, with probability at least $1 - \delta$, the budget satisfies $\mathcal{T} \lesssim \log(1/\delta)\, H$, with

$$H = \frac{d}{\theta}\left(\frac{1}{\|\Delta\|_2^2} + \frac{1}{s^*}\right) + \min_{s\in[d]}\left(\frac{d}{s} + n\right)\left(\frac{1}{\Delta_{(s)}^2} + 1\right) ,$$

where θ is the balancedness of the partition, $\Delta_{(1)} \geq \cdots \geq \Delta_{(d)}$ are ordered absolute gaps, and $s^* \in \arg\max_{s\in[d]} s\Delta_{(s)}^2$ is an effective sparsity parameter.

We rely on the ability of elimination techniques to perform well in settings where there are many good arms (or features) (Karnin et al., 2013a), which is a key aspect of the problem. The resulting algorithm is computationally efficient and adaptive to the unknown structure of Δ. The key intuition is that, once a good feature is found, the problem reduces to classifying n items based on that feature, which is a much easier problem than clustering in the original d-dimensional space.

14. Equal contribution—Institut für Mathematik, Universität Potsdam, Potsdam, Germany
15. INRAE, Mistea, Institut Agro, Univ Montpellier, Montpellier, France

Lower bound and optimality. Additionally, we establish an instance-dependent lower bound showing that any δ-correct algorithm must spend at least

$$\frac{d}{\theta\|\Delta\|_2^2}\log\frac{1}{\delta} \;\vee\; \frac{n}{\Delta_{(1)}^2}\log\frac{1}{\delta}$$

on some permuted instance. In Corollary 5.3.2, we show that the upper and lower bounds match up to poly-logarithmic factors in the two-level regimes where $\Delta \in \{0, h\}^d$.

Discussion. The complexity term contains several components. For simplicity, we discuss the case $\Delta \in \{0, h\}^d$, which is closely related to the motivating examples in Section 2.5. In this case, the upper and lower bounds reduce to $\frac{d}{\theta s h^2}\log(1/\delta) + \frac{n}{h^2}\log(1/\delta)$. The first term is a detection cost, corresponding to finding two representative items, one from each cluster. The factor $\frac{d}{s} \times \frac{1}{\theta}$ is the intrinsic sub-sampling cost of finding a row in the second group and a feature that discriminates between the groups. Besides, $\frac{1}{h^2}\log(1/\delta)$ is the cost of certifying that the corresponding gap is non-zero at confidence level $1-\delta$. The second term, $\frac{n}{h^2}\log(1/\delta)$, is the cost of classifying all items using that feature. For general Δ, the optimal strategy involves adapting to some effective sparsity level s^* reaching the minimum in the general bound.

The main message of Chapter 5 is that the sample complexity depends on the full structure of the gap vector Δ, and one can adapt to the effective dimensionality of the problem.

Chapter 5 **Summary of contributions**

1. We formalize CBP with feature selection, where the learner must jointly identify informative features and recover the clusters under a fixed-confidence guarantee.
2. We propose `BanditClustering`, a δ-correct algorithm, based on Sequential-Halving-style feature search. With probability at least $1-\delta$, the budget satisfies $\mathcal{T} \leqslant H\log(1/\delta)$, with H the instance-dependent complexity term defined by
$$H = \frac{d}{\theta}\left(\frac{1}{\|\Delta\|_2^2} + \frac{1}{s^*}\right) + \min_{s\in[d]}\left(\frac{d}{s} + n\right)\left(\frac{1}{\Delta_{(s)}^2} + 1\right) .$$
3. We establish an information-theoretic lower bound, and show near-matching upper/lower bounds up to poly-logarithmic factors in two-level regimes.

2.6.4 Condorcet Winner Identification (Chapter 6)

Chapter 6 is a joint work with El Mehdi Saad [16], and Nicolas Verzelen [17], available as a preprint in (Saad et al., 2026).

Problem setting. Chapter 6 studies Condorcet Winner Identification (CWI) in stochastic dueling bandits (see Section 2.2.3). The Condorcet winner (CW) is an arm $i^* \in [K]$ that is preferred

16. Equal contribution—UM6P College of Computing Rabat, Morocco.
17. INRAE, Mistea, Institut Agro, Univ Montpellier, Montpellier, France.

to every other arm, i.e., $\Delta_{i^*,j} > 0$ for all $j \neq i^*$. Assuming a CW exists, the fixed-confidence objective is to output i^* with probability at least $1-\delta$ while minimizing the sample complexity $\mathcal{T}$.[18] A key feature of our analysis is that, except the existence of a CW, it does not rely on structural assumptions on the preference matrix Δ: no total order over the arms is assumed.

Baselines. We compare our algorithms to two baselines.

1. First, we compare to the budget $H_{\mathrm{cw}} = \sum_{i \neq i^*} \Delta_{i^*,i}^{-2} \log(1/\delta)$ obtained in the state-of-the-art method of Maiti et al. (2024). The quantity H_{cw} corresponds to the cost of eliminating suboptimal arms by comparing them directly to the CW. This is optimal when the CW is the strongest arm against all others, i.e., when $\Delta_{i^*,j} = \max_{i \neq j} \Delta_{i,j}$ for all $j \neq i^*$. However, this guarantee can be arbitrarily loose when the CW is nearly tied with other arms. Our analysis improves this guarantee in general by leveraging the full gap matrix, while recovering it when the CW is indeed the strongest arm against all others.

2. We also discuss the verification-based approach of Karnin (2016), with asymptotic budget $\sum_{i \neq i^*} \min_{j:\Delta_{i,j}<0} \Delta_{i,j}^{-2} \log(1/\delta)$, which is optimal as $\delta \to 0$. This bound is naturally unavoidable: deciding that i^* is the CW is equivalent to deciding that each suboptimal arm $i \neq i^*$ is not a CW. This requires eliminating each such arm by comparing it to some opponent j with $\Delta_{i,j} < 0$, at minimal cost when using the strongest opponent, $\mathrm{argmax}_{j:\Delta_{i,j}<0} \Delta_{i,j}$. Hence, the asymptotic budget of Karnin (2016) corresponds to the minimal cost of eliminating each suboptimal arm using its strongest opponent, and this cost is unavoidable, as shown by the lower bound in Haddenhorst et al. (2021b). Still, this bound is asymptotic in the regime $\delta \to 0$. In our non-asymptotic analysis, we show that for moderate confidence levels (e.g., large-scale settings with large K and not-too-small δ), this asymptotic bound hides an intrinsic exploration cost that we fully characterize.

Exploration and certification trade-off. Eliminating a suboptimal arm $i \neq i^*$ involves two costs: exploration (finding an opponent that beats i) and certification (proving that the corresponding gap is negative with high confidence). Let $K_{i;<0} := |\{j : \Delta_{i,j} < 0\}|$ and let $\Delta_{i,(1)} \leq \cdots \leq \Delta_{i,(K_{i;<0})} < 0$ be the ordered negative gaps against i. For some effective sparsity $s_i \leq K_{i;<0}$, these costs scale as $K/(s_i \Delta_{i,(s_i)}^2)$ and $\log(1/\delta)/\Delta_{i,(s_i)}^2$, as suggested by the motivating example in Section 2.5. Hence, to eliminate arm i, one should find an opponent j with $\Delta_{i,j} = \Delta_{i,(s_i)}$ for some s_i, and then certify that $\Delta_{i,j} < 0$. The optimal choice of s_i depends on both the instance and δ, and is selected adaptively by our algorithm. To capture the resulting exploration and certification costs, we define, for any $\boldsymbol{s} = (s_1, \ldots, s_K)$, with $s_i \leq K_{i;<0}$,

$$H_{\mathrm{explore}}(\boldsymbol{s}, \delta) := \max_{i \neq i^*} \frac{K \log(1/\delta)}{s_i \Delta_{i,(s_i)}^2} + \sum_{i \neq i^*} \frac{K}{s_i \Delta_{i,(s_i)}^2}, \qquad H_{\mathrm{certify}}(\boldsymbol{s}, \delta) := \sum_{i \neq i^*} \frac{\log(1/\delta)}{\Delta_{i,(s_i)}^2}.$$

The sample complexity decomposes as a trade-off between these terms.

18. The fixed-budget setting is also studied in this chapter.

Algorithmic ideas. FB-CWI (Algorithm 16) is an elimination algorithm based on two subroutines: a strong-opponent search (adapted from Sequential Halving Karnin et al. (2013a)) and a weak-opponent search based on adaptive quantile estimation via RANGE-QUANTILE (Algorithm 15). Their outputs define a score that eliminates a constant fraction of active arms at each stage. Interestingly, RANGE-QUANTILE is a new procedure that estimates the s-th quantile of a vector and may be of independent interest.

FC-CWI (Algorithm 17) wraps FB-CWI in a doubling schedule and adds two stopping tests: (i) a direct CW certification, which checks whether the candidate CW has a positive row, and (ii) an elimination-frontier certification, which verifies that our score-based elimination is correct. The algorithm is δ-correct and, up to logarithmic factors in K and an additional $\log\log(1/\delta)$ term (see Theorem 6.3.1), satisfies with probability at least $1-\delta$:

$$\mathcal{T}_\delta \lesssim H_{\mathrm{cw}}(\delta) \wedge \min_{\boldsymbol{s}} \Big\{ \big(H_{\mathrm{certify}}(\boldsymbol{s},\delta) + H_{\mathrm{explore}}(\boldsymbol{s},\delta)\big) \Big\} .$$

If the CW is the strongest arm against all others, this recovers the $H_{\mathrm{cw}}(\delta)$ guarantee of Maiti et al. (2024) [19]; otherwise, it improves upon it. However, our bound is always worse than the asymptotic bound of Karnin (2016). This is because, for moderate confidence levels, the exploration cost dominates the certification cost, while the asymptotic bound captures only the latter. For instance, if $\log(1/\delta)$ is treated as a constant, the right-hand side of our upper bound is dominated by the term $\min_{\boldsymbol{s}} \sum_{i\neq i^*} \frac{K}{s_i \Delta^2_{i,(s_i)}} \simeq \sum_{i\neq i^*} \frac{K}{\|\Delta_i^-\|_2^2}$, up to a $\log(d)$ factor, [20] which can be larger than the bound of Karnin (2016) by a factor of at most K. Our lower bound confirms that this exploration cost is intrinsic and unavoidable.

Lower bounds and optimality. The lower bound from Haddenhorst et al. (2021b) establishes that the complexity term from Karnin (2016) is unavoidable.

In Theorem 6.4.2, we establish a high-probability lower bound, yielding a new understanding of the instance-dependent complexity of CWI. The proof relies on a reduction to active multiple testing, which is novel in this setting. For simplicity, we state here the minimax version (Corollary 6.4.4), which holds for any δ-correct algorithm and any gap matrix $\boldsymbol{\Delta}$. For any δ-correct algorithm π,

$$\sup_{\tilde{\boldsymbol{\Delta}}\in\mathbb{D}(\boldsymbol{\Delta})} \inf \Big\{ \chi > 0 \text{ s.t.: } \mathbb{P}_{\tilde{\boldsymbol{\Delta}},\pi}(N_\delta \leqslant \chi) \leqslant \delta \Big\} \gtrsim \min_{\boldsymbol{s}} \left\{ H_{\mathrm{certify}}(\boldsymbol{s},\delta) + H_{\mathrm{explore}}(\boldsymbol{s},\delta) \right\} ,$$

where $\mathbb{D}(\boldsymbol{\Delta})$ is a set of perturbations of $\boldsymbol{\Delta}$ that preserve the CW, the sign structure of $\boldsymbol{\Delta}$, and the optimal sparsity level $\boldsymbol{s}^*_{\boldsymbol{\Delta}}$. [21] Overall, this lower bound confirms that the exploration-certification trade-off is intrinsic and unavoidable, and that our method is order-optimal.

The main message in this work is that we improve existing bounds and obtain a better understanding of the instance-dependent complexity of CWI when there are no structural assumptions on the preference matrix Δ except the existence of a CW. Besides, our analysis holds for all con-

19. And, by corollary, the bound of Karnin (2016), which is equal to $H_{\mathrm{cw}}(\delta)$ under this assumption.

20. From the inequality $\max_{k\in\{1,\dots,K\}} kx_k^2 \leq \sum_{i=1}^K x_i^2 \leq \log(4K)\max_{k\in\{1,\dots,K\}} kx_k^2$, valid for any decreasing sequence $x_1 \geq \cdots \geq x_K$.

21. We postpone to Chapter 6 for the precise definition of this local class $\mathbb{D}(\boldsymbol{\Delta})$.

fidence levels $\delta \in (0,1)$, which allows for a more comprehensive understanding of the problem, particularly in large-scale regimes where K is large.

Chapter 6 **Summary of contributions**

1. We introduce FB-CWI and FC-CWI, new elimination-based algorithms for the fixed-budget and fixed-confidence settings, built on Sequential Halving, a new RANGE-QUANTILE subroutine, and a score-based elimination rule. FC-CWI is δ-correct and satisfies, with probability at least $1-\delta$,
$$\mathcal{T}_\delta \lesssim H_{\mathrm{cw}}(\delta) \wedge \min_{\boldsymbol{s}} \Big\{ \big(H_{\mathrm{certify}}(\boldsymbol{s},\delta) + H_{\mathrm{explore}}(\boldsymbol{s},\delta)\big) \Big\},$$
improving over the winner-only guarantee $H_{\mathrm{cw}}(\delta)$ in all regimes.
2. We establish new instance-dependent lower bounds for CWI, obtained via a reduction to active multiple testing. This bound yields a local-minimax lower bound showing that the exploration-certification trade-off is intrinsic and unavoidable.
3. We uncover phenomena that are invisible to asymptotic analyses, and the non-trivial role of the full gap matrix rather than only the winner row. This exploration-certification trade-off is a new contribution to the theory of CWI.

2.6.5 Multiple Change Point Identification (Chapter 7)

Chapter 7 is a joint work with Maximilian Graf[22], available as a preprint in (Graf and Thuot, 2026).

Problem setting. In Chapter 7, we study Multiple Change Point Identification (MCP) under bandit feedback. We consider a piecewise-constant function $f : [0,1] \to \mathbb{R}$ with m change points at unknown locations $0 < x_1^* < \cdots < x_m^* < 1$, so that f is constant on each interval $(x_{k-1}^*, x_k^*]$ (with the convention $x_0^* = 0$ and $x_{m+1}^* = 1$). At each round, the learner selects a query point $I_t \in [0,1]$ and observes $X_t = f(I_t) + \varepsilon_t$, where ε_t is σ-sub-Gaussian noise independent of past queries. In the fixed-confidence setting, the goal is to recover the m change points with precision η and probability at least $1-\delta$, while minimizing the total number of queries $\mathcal{T}$.[23] More precisely, the objective is to output m estimates $(\widehat{x}_1, \ldots, \widehat{x}_m)$ such that $\max_{k\in[m]} |\widehat{x}_k - x_k^*| \leqslant \eta$ with probability at least $1-\delta$.

Comparison with the literature. This problem can be viewed as a structured variant of Clustering with Bandit Feedback, after discretization, in which case the features dimension is $d = 1$. The key difference from CBP is that the spatial structure of $[0,1]$ can be exploited: a query at a point x is informative not only about the local value of f, but also about the possible

22. Equal contribution—Institut für Mathematik, Universität Potsdam, Potsdam, Germany
23. We consider the more general setting of localizing a subset of $N \leqslant m$ change points; see Chapter 7 for details.

existence of a nearby boundary. This motivates a fundamentally different algorithmic approach based on recursive binary search and local hypothesis testing.

The problem was first studied in Lazzaro and Pike-Burke (2025b), where the analysis focuses on the asymptotic regime $\delta \to 0$. Their result has the form $\lim_{\delta\to 0} \frac{\mathbb{E}[\mathcal{T}]}{\log(1/\delta)} \lesssim H_{\text{localize}}$, with $H_{\text{localize}} := \sum_{k=1}^{m} \Delta_k^{-2}$ and $\Delta_k := |f(x_k^{*+}) - f(x_k^{*-})|$ is the jump magnitude at x_k^*. This bound is optimal in the asymptotic regime and corresponds to the cost of certifying each estimated change point at confidence level $1-\delta$. However, this asymptotic characterization does not capture exploration, namely the cost of detecting the presence of change points. In addition, the dependence on the precision parameter η is hidden by the asymptotic regime. Our analysis is non-asymptotic in both δ and η, and improves the dependence on η from linear in $1/\eta$ to logarithmic in $1/\eta$. This is crucial in large-scale regimes where high localization precision is required.

Our analysis also quantifies exploration. Inspired by the batch MCP literature, we introduce an energy parameter $\mathcal{E}_k^2 := s_k \Delta_k^2$, where s_k captures local spacing through $s_k := \vartheta_{k-1} \wedge \vartheta_k$, with $\vartheta_k := x_{k+1}^* - x_k^*$ for $k = 1, \dots, m-1$, and by convention $\vartheta_0 = \vartheta_m = 1$. This yields an additional detection complexity term of order $H_{\text{detect}} := \max_k 1/\mathcal{E}_k^2$, which our lower bounds show to be intrinsic.

Algorithmic design. The chapter introduces `LocalizeChangePoints` (`LCP`), an adaptive algorithm that wraps a fixed-budget procedure in a doubling schedule and stops once an estimated set of change points is certified to be correct. The method has four steps: (i) a detection step, which identifies points in each plateau of the piecewise-constant function f using multiscale tests; (ii) an estimation step, which estimates the jump of each change point; (iii) a localization step, which refines the estimates to precision η using binary search; and (iv) a verification step, which confirms the estimates. The estimation and verification stages contribute a cost of order $H_{\text{localize}} \log(1/\delta)$, matching Lazzaro and Pike-Burke (2025b).

Besides, the detection stage introduces an additional cost of order $H_{\text{detect}} = \max_k 1/\mathcal{E}_k^2$. This is a genuinely non-asymptotic effect, invisible when $\delta \to 0$. In addition, refinement to precision η induces an extra $H_{\text{localize}} \log(1/\eta)$ term, which is the optimal dependence in the precision η. Overall, the method is δ-correct and satisfies non-asymptotic guarantees in both high probability and expectation. Up to logarithmic factors,

$$\begin{aligned} \mathbb{E}[\mathcal{T}] &\lesssim H_{\text{detect}} + H_{\text{localize}} \log\left(\frac{1}{\delta\eta}\right) && \text{in expectation ,} \\ \mathcal{T} &\lesssim H_{\text{detect}} \log\left(\frac{1}{\delta}\right) + H_{\text{localize}} \log\left(\frac{1}{\delta\eta}\right) && \text{with probability at least } 1-\delta \end{aligned}$$

For the expectation bound, we recover the same dependence on δ, namely $H_{\text{localize}} \log(1/\delta)$, as Lazzaro and Pike-Burke (2025b), while improving the dependence on η. We also identify an additional non-asymptotic complexity H_{detect}, which can dominate when change points are very close. This detection cost is intrinsic and unavoidable by our lower bounds. The complexity H_{detect} is always larger than H_{localize}, and the gap can be arbitrarily large when change points are extremely close and hard to detect.

Lower bounds and optimality. We establish information-theoretic lower bounds for MCP localization. These lower bounds match our upper bounds up to logarithmic factors. To the best of our knowledge, this is the first characterization of MCP localization complexity at this level of precision. The proof relies on a reduction to a signal-detection problem, similar to the one discussed in Section 2.5. Compared with Lazzaro and Pike-Burke (2025b), whose guarantee is asymptotically optimal as $\delta \to 0$ but not non-asymptotically tight, our bounds improve the dependence on η from linear in $1/\eta$ to logarithmic in $1/\eta$, while maintaining the same dependence on δ. This is especially relevant in high-precision regimes.

Chapter 7 **Summary of contributions**

1. We introduce `LocalizeChangePoints` (`LCP`), an adaptive algorithm that combines multiscale detection, jump estimation, binary-search localization, and verification. The method is δ-correct and satisfies non-asymptotic guarantees in both high probability and expectation. Up to logarithmic factors,

$$\mathbb{E}[\mathcal{T}] \lesssim H_{\text{detect}} + H_{\text{localize}} \log\left(\frac{1}{\delta\eta}\right) \qquad \text{in expectation ,}$$

$$\mathcal{T} \lesssim H_{\text{detect}} \log\left(\frac{1}{\delta}\right) + H_{\text{localize}} \log\left(\frac{1}{\delta\eta}\right) \qquad \text{with probability at least } 1-\delta$$

   where $H_{\text{detect}} = \max_i 1/\mathcal{E}_i^2$ and $H_{\text{localize}} = \sum_{i=1}^m 1/\Delta_{(i)}^2$.
2. We prove new information-theoretic lower bounds, showing that any algorithm must pay both a detection cost and a localization cost. On a local-minimax class of instances, these lower bounds match the upper bounds up to logarithmic factors, for both expected-budget and quantile guarantees.
3. We reveal a genuinely non-asymptotic phenomenon: the sample complexity is jointly governed by jump magnitudes and local spacings (through energies $\mathcal{E}_i^2 = s_i\Delta_i^2$), via a detection-verification trade-off. In particular, this sharpens our understanding of MCP localization in high-precision regimes (small η).

2.7 Thesis structure

The thesis is organized as follows. In Chapters 3–5, we study several variants of the Clustering with Bandit Feedback (CBP) problem. Chapter 3 addresses the parametric case, Chapter 4 the non-parametric case, and Chapter 5 CBP with feature selection. In Chapter 6, we study the Condorcet Winner Identification (CWI) problem. In Chapter 7, we study Multiple Change Point Identification (MCP). Each chapter is self-contained and can be read independently of the others. [24]

24. Besides, these chapters are available online as conference proceedings or as an author's preprint.

Appendix of the Introduction

2.A Proof of Proposition 2.5.1

The proof of Proposition 2.5.1 relies on the same arguments as the proof of Theorem 5.4.1 and Theorem 7.4.1, but in a simplified setting. It is detailed below for the sake of completeness.

Proof of Proposition 2.5.1. First, we use a classical change-of-measure argument to obtain an instance-dependent lower bound for the expectation of the budget, which corresponds to the term $c\frac{\sigma^2}{\Delta^2}\log(1/\delta)$. Then, we establish the lower bound on the quantile of the budget of order $\frac{d\sigma^2}{s\Delta^2}\log(1/\delta)$, with a proof strategy that is inspired by the proof of Theorem 5.4.1. Finally, we show that the term $\frac{d\sigma^2}{s\Delta^2}$ is also a lower bound on the expectation of the budget, by using Markov's inequality.

Lower bound on the expectation of the budget. Fix an algorithm π, and consider a fixed vector μ with support S of size s, with non-zero entries equal to Δ. We denote as ν_S for the associated environment with Gaussian noise of variance σ^2. Assume that $s < d$, so that μ has at least one zero entry.

Let π be a δ-correct algorithm for problem (i). We also denote as $\hat{\imath}$ the random variable corresponding to the arm selected by π at the end of the learning process, and as T the random variable corresponding to the stopping time of π. We denote as $\mathbb{P}_S$ the probability distribution induced by π in the environment ν_S. By δ-correctness for the problem (i), we have $\mathbb{P}_S(\hat{\imath} \in S) \geq 1-\delta$.

Then, consider the alternative environment ν_{S^c} whose support is the complement of S, i.e., $S^c := [d] \setminus S$, and denote as $\mathbb{P}_{S^c}$ the distribution induced by π in this environment. Since the support is empty under $\mathbb{P}_{S^c}$, we have, by δ-correctness, $\mathbb{P}_{S^c}(\hat{\imath} \in S) \leqslant \delta$.

Applying the Bretagnolle–Huber inequality (see Lattimore and Szepesvári, 2020, Thm. 14.2), we obtain

$$\frac{1}{2}\exp\left(-\operatorname{KL}(\mathbb{P}_S, \mathbb{P}_{S^c})\right) \leqslant \mathbb{P}_S(\hat{\imath} \in S) + \mathbb{P}_{S^c}(\hat{\imath} \in S) \leqslant 2\delta \ ,$$

which implies

$$\log\frac{1}{4\delta} \leqslant \operatorname{KL}(\mathbb{P}_S, \mathbb{P}_{S^c}) \ . \tag{2.1}$$

Next, using the decomposition of KL divergence for bandit models (Lattimore and Szepesvári, 2020, Lemma. 15.1), and the Gaussian assumption, we have

$$\operatorname{KL}(\mathbb{P}_S, \mathbb{P}_{S^c}) = \sum_i \mathbb{E}_S[T_i]\operatorname{KL}(\mathcal{N}(\mu_i,\sigma^2), \mathcal{N}(|\mu_i-\Delta|,\sigma^2)) = \sum_i \mathbb{E}_S[T_i]\frac{\Delta^2}{2\sigma^2} = \mathbb{E}_S[T]\frac{\Delta^2}{2\sigma^2} \ , \tag{2.2}$$

where T_i is the number of times the learner samples the i-th arm, so that $\sum_i T_i = T$. We use the fact that under ν_S and ν_{S^c}, the distributions of the arms are Gaussian with means differing in absolute value by Δ.

Combining with (2.1), and rearranging, we obtain the first lower bound on the expectation of the budget $\mathbb{E}_S[T] \geq \frac{2\sigma^2}{\Delta^2} \log \frac{1}{4\delta}$.

Lower bound on the $(1-\delta)$-quantile of the budget. In this proof, we exploit events relative to the stopping time of π, instead of the recommendation $\hat{\imath}$. This allows us to obtain a lower bound on the quantiles of the budget. The result that we obtain is a local-minimax lower bound, on the class of environments with support of size s and non-zero entries equal to Δ.

Let $\mathcal{E}_{\mathrm{per}}(\mu)$ denote the set of Gaussian environments obtained by permuting μ by any permutation τ. Define the environment ν_τ as follows:

$$\nu_\tau = \Big(\mathcal{N}(\mu_{\tau(1)}, \sigma^2), \dots, \mathcal{N}(\mu_{\tau(d)}, \sigma^2)\Big) \ .$$

Under ν_τ, the unknown support is $\{s; \tau(s) \in S\} = \tau^{-1}(S)$. This permutation allows us to take into account in the lower bound that the learner does not know the support S.

Define χ as the smallest integer such that for all permutation τ of $1, \dots, d$, the following inequality holds:

$$\mathbb{P}_{\tau^{-1}(S)}(T > \chi) \leqslant \delta \ . \tag{2.3}$$

In other words, χ is the largest $(1-\delta)$-quantile of the budget of π over all environments in $\mathcal{E}_{\mathrm{per}}(\mu)$. Our goal is to derive a lower bound on χ.

Introduce $\mathbb{P}_0$ as the probability distribution induced by π, in an environment where the support is empty, and all observations are drawn from $\mathcal{N}(0, \sigma^2)$. We use this distribution as a reference distribution for the change-of-measure arguments. We justify that the event $T \leqslant \chi$ has small probability under $\mathbb{P}_0$. Since π is δ-correct, we have

$$\mathbb{P}_0(T \leqslant \chi) \leqslant 2\delta \ . \tag{2.4}$$

This comes from the fact that under $\mathbb{P}_0$, the support is empty, so the algorithm cannot stop in finite time, which would mean that the learner certifies a non-zero entry while there is none. This bound is obtained by decomposing the event $T \leqslant \chi$ into the union of the events $\{T \leqslant \chi, \hat{\imath} \in S\}$ and $\{T \leqslant \chi, \hat{\imath} \notin S\}$, and using the δ-correctness of π for both events, observing that ν_0 is as close as possible to environments with support S [resp. S^c] and non-zero entries equal to ϵ, with $\epsilon \to 0$. We refer to the proof of Lemma 7.C.4 for a rigorous argument.

Applying the Bretagnolle–Huber inequality (see Lattimore and Szepesvári, 2020, Thm. 14.2), and Equations (2.3) and (2.4), we obtain

$$\frac{1}{2} \exp\Big(-\mathrm{KL}(\mathbb{P}_0, \mathbb{P}_{\tau^{-1}(S)})\Big) \leqslant \mathbb{P}_0(T \leqslant \chi) + \mathbb{P}_{\tau^{-1}(S)}(T > \chi) \leqslant 3\delta \ ,$$

which implies

$$\log \frac{1}{6\delta} \leqslant \mathrm{KL}(\mathbb{P}_0, \mathbb{P}_{\tau^{-1}(S)}) \ . \tag{2.5}$$

Observe that the event $T \leqslant \chi$ is measurable with respect to the first χ samples, so that the above application of the Bretagnolle–Huber inequality is still valid for the truncated algorithm that stops

at time $T \wedge \chi$.

Next, using the decomposition of KL divergence, we have

$$\mathrm{KL}(\mathbb{P}_0, \mathbb{P}_{\tau^{-1}(S)}) = \sum_i \mathbb{E}_0[T_i]\,\mathrm{KL}(\mathcal{N}(0,1), \mathcal{N}(\mu_{\tau(i)}, 1)) = \sum_i \mathbb{E}_0[T_i]\mathbb{1}_{\tau(i)\in S}\frac{\Delta^2}{2} \ .$$

Averaging both sides over all permutations τ, and using Equation (2.5), we get

$$\log\frac{1}{6\delta} \leqslant \frac{1}{d!}\sum_\tau\sum_i \mathbb{E}_0[T_i]\mathbb{1}_{\tau(i)\in S}\frac{\Delta^2}{2} \ . \tag{2.6}$$

Now, observe that each element in $i \in \{1, \dots, d\}$ appears exactly $(d-1)!$ times in the multi-set $\{\tau(i)\}_\tau$, so that we can permute the sums to obtain

$$\begin{aligned}\frac{1}{d!}\sum_\tau\sum_i \mathbb{E}_0[T_i]\mathbb{1}_{\tau(i)\in S}\frac{\Delta^2}{2} &= \frac{(d-1)!}{d!}\sum_i\sum_j \mathbb{E}_0[T_i]\mathbb{1}_{j\in S}\frac{\Delta^2}{2} \\ &= \frac{1}{d}\frac{s\Delta^2}{2}\mathbb{E}_0[T] \ .\end{aligned}$$

Using the fact that we consider the truncated algorithm with stopping time $T \wedge \chi$, we can bound $\mathbb{E}_0[T] \leqslant \chi$. Finally, it follows that:

$$\chi \geqslant \frac{2d}{s\Delta^2}\log\frac{1}{6\delta} \ .$$

Since χ is the maximum over all permuted environments in $\mathcal{E}_{\mathrm{per}}(\mu)$ of the $(1-\delta)$-quantile of the budget, this inequality implies that there exists an environment in $\mathcal{E}_{\mathrm{per}}(\mu)$ for which $\mathbb{P}_{\pi,\tau^{-1}(S)}(\mathcal{T} \geqslant \frac{2d}{s\Delta^2}\log\frac{1}{6\delta}) \geq \delta$, which is exactly the claimed quantile lower bound.

Lower bound on the expectation of the budget.

Finally, we show that the term $\frac{d\sigma^2}{s\Delta^2}$ is also a lower bound on the expectation of the budget. Using Markov's inequality, and the quantile bound, we have, for any $\delta < 1/6$, there exists an environment ν_S with $|S| = s$ such that

$$\begin{aligned}\mathbb{E}_{\pi,\nu_S}[\mathcal{T}] &\geqslant \frac{2d\sigma^2}{s\Delta^2}\log\frac{1}{6\delta}\cdot\mathbb{P}_{\pi,\nu_S}\left[\mathcal{T} \geqslant 2\frac{d\sigma^2}{s\Delta^2}\log(1/6\delta)\right] \\ &\geqslant 2\frac{d\sigma^2}{s\Delta^2}\log(1/6\delta)\cdot\delta \geqslant \frac{1}{3e}\frac{d\sigma^2}{s\Delta^2} \ ,\end{aligned}$$

where the last inequality follows by choosing $\delta = 1/(6e)$. □

2.B Proof of Proposition 2.5.2

The meta-structure of the algorithm is close to the strategy used in *LocalizeChangePoints* (Algorithm 20).

Proof of Proposition 2.5.2. First, we derive two key ingredients: the guarantees of BSH and the verification test. Then, we combine these ingredients to prove the correctness guarantee and the bounds on the budget.

Key ingredients. First, we use Theorem 6 of Zhao et al. (2023), which provides guarantees on BSH (Algorithm 3 in Zhao et al. (2023)), applied with $\epsilon = \Delta/2$. From this theorem, there exists a poly-logarithmic factor l_1 in $d\sigma^2\Delta^{-2}$ such that if $T \geqslant l_1 \frac{d\sigma^2}{s\Delta^2}$, then, denoting by $\hat{\imath}$ the output of BSH with budget T, we have

$$\mathbb{P}(\mu_{\hat{\imath}} = 0) \leqslant \exp\left(-l_1 \cdot T \cdot \frac{s\Delta^2}{d\sigma^2}\right) . \tag{2.7}$$

We deduce two consequences: we can choose l_1, a polylogarithmic term in $d \cdot \sigma^2 \cdot \Delta^{-2}$, so that if $T \geqslant l_1 \frac{d\sigma^2}{s\Delta^2}$, the probability that BSH outputs a non-zero entry is at least $1/8$, and if $T \geqslant l_1 \frac{d\sigma^2}{s\Delta^2} \log(1/\delta)$, the probability that BSH outputs a non-zero entry is at least $1-\delta$.

Now, consider the verification test. Let $\tilde{\delta} < 1$, let $i \in [d]$, and let $\hat{\mu}_i$ be the empirical mean of i after T samples. From Hoeffding's inequality, for any $\tilde{\delta} \in (0,1)$,

$$\mathbb{P}\left(|\hat{\mu}_i - \mu_i| \geq \sqrt{2\sigma^2 \log(2/\tilde{\delta})/T}\right) \leq \tilde{\delta} \ .$$

So if $\mu_i = 0$, then $|\hat{\mu}_i| < \sqrt{2\sigma^2 \log(2/\tilde{\delta})/T}$ with probability at least $1-\tilde{\delta}$, proving Equation (2.8). On the other hand, as long as $\Delta > 2\sqrt{2\sigma^2 \log(2/\tilde{\delta})/T}$, this guarantees that $|\hat{\mu}_i| > \sqrt{2\sigma^2 \log(2/\tilde{\delta})/T}$ with probability at least $1-\tilde{\delta}$. This is equivalent to $T \geq 8 \cdot \frac{\sigma^2}{\Delta^2} \log\left(\frac{2}{\tilde{\delta}}\right)$, which concludes the proof of Equation (2.9). We then have the two following guarantees for the verification test:

$$\text{If } \mu_i = 0, \text{ then } \mathbb{P}\left(\hat{\mu}_i < \sqrt{2\frac{\sigma^2}{T} \cdot \log\left(\frac{2}{\tilde{\delta}}\right)}\right) \geq 1 - \tilde{\delta} \ . \tag{2.8}$$

$$\text{If } \mu_i = \Delta \text{ and } T \geq 8 \cdot \frac{\sigma^2 \log\left(\frac{2}{\tilde{\delta}}\right)}{\Delta^2}, \text{ then } \mathbb{P}\left(\hat{\mu}_i > \sqrt{2\frac{\sigma^2}{T} \cdot \log\left(\frac{2}{\tilde{\delta}}\right)}\right) \geq 1 - \tilde{\delta} \ . \tag{2.9}$$

Now, we have all the ingredients to prove correctness and the bounds on the budget.

Correction Consider a run of Algorithm 1 with input δ.

The correctness guarantee relies only on the verification step. Consider the k-th epoch, where BSH returns a candidate i_k. From Equations (2.8) applied with $T_k = 2^k$, and $\delta_k = \frac{6}{\pi^2}\frac{\delta}{k^2}$, we have that if $\mu_{i_k} = 0$, then with probability at least $1-\delta_k$, $\hat{\mu}_{i_k} < \sqrt{2\sigma^2 \log(12k^2/\pi^2\delta)/2^k}$. In this case,

our stopping condition is not met. By a union bound over epochs, the probability of outputting a zero entry is therefore bounded by $\sum_{k=1}^{+\infty} \frac{6\delta}{\pi^2 k^2} = \delta$, which proves the correctness guarantee.

Bound in high probability. Now, we prove the high-probability bound on the budget. Denote as k^* the smallest integer such that $2^{k^*} \geq l_1 \frac{d\sigma^2}{s\Delta^2} \log(2/\delta) \vee 8\frac{\sigma^2}{\Delta^2} \log(12k_*^2/\pi^2\delta)$, then, from Equation (2.7), BSH returns a non-zero entry with probability at least $1 - \delta/2$ at epoch k^*. Moreover, from Equation (2.9), the verification test succeeds with probability at least $1 - \delta/2$ at epoch k^*. By a union bound, the algorithm stops by epoch k^* with probability at least $1 - \delta$. Since $\sum_{k=1}^{k^*} 2^k \leq 2^{k^*+1}$, and epoch k uses a budget 2^{k+1}, the budget is bounded by 2^{k^*+2} with probability at least $1 - \delta$.

Now, from standard computation, we have that there exists a poly-logarithmic term l_2 such that, with probability at least $1 - \delta$, the budget $\mathcal{T}$ satisfies

$$\mathcal{T} \leqslant 2^{k^*+2} \leq l_2 \frac{d\sigma^2}{s\Delta^2} \log(1/\delta) \ ,$$

which concludes the proof of the high-probability bound on the budget.

Bound in expectation. Finally, we prove the bound on the expected budget. Let $k_0 \geqslant 3$ be the smallest integer such that

$$2^{k_0} \geq l_1 \frac{d\sigma^2}{s\Delta^2} \vee 8\frac{\sigma^2}{\Delta^2} \log\left(\frac{12k_0^2}{\pi^2\delta}\right) \ ,$$

where, if needed, we enlarge the constant hidden in l_1 so that Equation (2.7) implies that BSH returns a non-zero entry with probability at least $7/8$ whenever $T \geq l_1 \frac{d\sigma^2}{s\Delta^2}$.

Fix any epoch $k \geqslant k_0$, and condition on the event that the algorithm has not stopped before epoch k. At this epoch, BSH is run with budget $2^k \geq 2^{k_0}$, so it returns a non-zero entry with probability at least $7/8$. Moreover, since $k_0 \geqslant 3$, we have $\delta_k = \frac{6}{\pi^2}\frac{\delta}{k^2} \leqslant \frac{1}{8}$, and the second condition in the definition of k_0 together with Equation (2.9) ensures that the verification step accepts this non-zero entry with probability at least $1 - \delta_k \geqslant 7/8$. Therefore, at every epoch $k \geqslant k_0$, conditional on reaching epoch k, the stopping probability is at least $3/4$.

Denote by K the random epoch at which the algorithm stops. The previous argument shows that $(K - k_0) \vee 0$ is stochastically dominated by a geometric random variable with parameter $3/4$. Since each epoch k costs at most 2^{k+1} samples, the total budget satisfies

$$\mathcal{T} \leqslant \sum_{k'=1}^{K} 2^{k'+1} < 4 \cdot 2^K \leqslant 4 \cdot 2^{k_0} \cdot 2^{(K-k_0)\vee 0} \ .$$

Hence,

$$\mathbb{E}[\mathcal{T}] \leqslant 4 \cdot 2^{k_0}\, \mathbb{E}\big[2^{(K-k_0)\vee 0}\big] \leqslant 4 \cdot 2^{k_0} \sum_{\ell=0}^{\infty} 2^\ell \left(\frac{1}{4}\right)^\ell \frac{3}{4} \leqslant 6 \cdot 2^{k_0} \ .$$

By definition of k_0, this is of order $\frac{d\sigma^2}{s\Delta^2} + \frac{\sigma^2}{\Delta^2}\log(1/\delta)$, up to logarithmic factors, which concludes the proof of the expectation bound.

□

Chapter 3

Clustering with Bandit Feedback Breaking Down the Computation/Information Gap

Abstract. *We investigate the Clustering with Bandit feedback Problem (CBP). A learner interacts with an n-armed stochastic bandit with d-dimensional subGaussian feedback. There exists a hidden partition of the arms into K groups, such that arms within the same group, share the same mean vector. The learner's task is to uncover this hidden partition with the smallest budget — i.e., the least number of observation — and with a probability of error smaller than a prescribed constant δ. We provide two complementary results, (i) we derive a non-asymptotic lower bound for the budget, and (ii) we introduce the computationally efficient ACB algorithm, whose budget matches the lower bound in most regimes. We improve on the performance of a uniform sampling strategy. Importantly, contrary to the batch setting, we establish that there is no computation-information gap in the bandit setting.*

Related publication. This Chapter is a joint work with Alexandra Carpentier [1], Christophe Giraud [2], and Nicolas Verzelen [3], published in ALT 2025 as (Thuot et al., 2025).

3.1 Introduction

We consider a sequential and active clustering problem, the **Clustering with Bandit feedback Problem (CBP)** introduced, for instance in (Yang et al., 2024; Yavas et al., 2025). In this setting, there are n items, represented by a d-dimensional mean. At each time t, the learner chooses one of the items, and samples it — i.e., obtains a noisy evaluation of the d-dimensional mean that characterizes it — until termination of the sampling process at time $\mathcal{T}$. The number of samples collected $\mathcal{T}$, which we call the budget, is chosen by the learner. We assume that the items are clustered into K unknown groups — and two items are in the same group if and only if their (unknown) means are the same. For a prescribed confidence level δ, the aim of the learner is to recover perfectly this clustering, on an event of probability larger than $1-\delta$, and with a final budget $\mathcal{T}$ that is as small as possible. Clustering problems are ubiquitous in modern data analysis, and CBP arises e.g., in digital marketing, where accurate clustering of the customers is crucial for adapting recommendations to specific groups of customers, and where repeated feedback can

1. Institut für Mathematik, Universität Potsdam, Potsdam, Germany.
2. Université Paris-Saclay, Laboratoire de mathématiques d'Orsay, Orsay, France.
3. INRAE, Mistea, Institut Agro, Univ Montpellier, Montpellier, France.

be collected online. Since feedback collection is costly, the goal is to recover the clusters with a minimal number $\mathcal{T}$ of feedback requests. See (Yang et al., 2024) for further motivations.

In the low-dimensional setting, where K, d are small, Yang et al. (2024) proves that, when δ converges to 0, an asymptotic expected budget for perfectly recovering the groups is at most of the order

$$\frac{\sigma^2}{\Delta_*^2} n \log(1/\delta) \ , \tag{3.1}$$

where Δ_* is the minimal Euclidean distance between the means, and σ^2 is the variance of the observations.

High-dimensional setting. We consider the high-dimensional setting, where K, d can be large, possibly larger than $1/\delta$ or n (for d). In the classical clustering setting, where there is no repeated measurements on each item, clustering in high-dimension can be nearly impossible in practice. Indeed, in high-dimension, the best polynomial-time algorithms require a very large separation of the means for successful clustering with no repeated measurements. This requirement has two origins. First, it is difficult to localize the means in high-dimension, making the clustering problem harder when d becomes large compared to n/K. Second, a computation-information gap is conjectured (i) for clustering (Lesieur et al., 2016; Even et al., 2024) when d is very large, and (ii) for estimation (Diakonikolas et al., 2017, 2023) in some high-dimensional non-isotropic setting.

For instance, when there is no repeated measurement, that is for vanilla clustering problems where each item is only observed once, clustering a mixture of n isotropic Gaussians with covariance I_d and balanced size of the groups, in the high-dimensional setting where $d \geq n$ and $K \gg \log(n)$, low-degree polynomial algorithms requires a separation at least $\Delta_*^2 \gtrsim \sigma^2\sqrt{dK^2/n}$ (see Even et al., 2024, Thm. 1), while a separation $\Delta_*^2 \gtrsim \sigma^2\sqrt{dK\log(n)/n}$ is enough at the information level (see Even et al., 2024, Thm. 4). This is a strong evidence of a computation-information gap for the problem of clustering isotropic Gaussian mixture in high dimension.

When repeated measurements are possible, let us consider the simple scheme where we sample T times each item. This scheme corresponds to oracle-BOC sampling of Yang et al. (2024), when the groups have similar sizes, and the clusters are equidistant. Sampling T times each item is equivalent to shrinking the variance from σ^2 to σ^2/T. Then, applying standard polynomial time algorithms (Giraud and Verzelen, 2019) to the average values for each item, we can recover the clustering in polynomial time with confidence $\delta = 1/n$ when $T \gtrsim \frac{\sigma^2}{\Delta_*^2}\sqrt{dK^2/n}$. A clustering procedure is said to be a batch if it uses one single observation of each item to recover the partition, as it is the case in vanilla clustering problems. The total number of requests of this simple batch algorithm is then

$$\mathcal{T} = nT \gtrsim n + \frac{\sigma^2}{\Delta_*^2}\sqrt{dK^2 n}. \tag{3.2}$$

Question 5. *This set of results raises two fundamental questions:*

1. *Can we improve upon the number of requests of the simple batch algorithm, by implementing a more careful sequential design strategy?*
2. *What is the minimal budget for perfect recovery in high-dimension, and is there a fundamental computation-information gap for clustering with bandit feedback?*

Contributions. We provide an answer to these two fundamental questions.

1. First, we provide a polynomial-time algorithm that recovers exactly the clustering with probability higher than $1-\delta$. In the balanced case (all groups have a similar size), it has an expected budget of order
$$n + \frac{\sigma^2}{\Delta_*^2}\left[n\log(n/\delta) + \sqrt{dnK\log(n/\delta)}\right], \tag{3.3}$$
which outperforms the budget (3.2) required by the simple batch algorithm.
2. Second, we prove that the budget (3.3) is information-theoretical optimal, meaning that there is no computation-information gap for clustering with bandit feedback in high-dimension, contrary to the classical case with no repeated measurement.

Our results are non-asymptotic in n, K, d, and δ, in order to account for high-dimensional phenomenon, and possible computational barriers —see the discussion for more details. Compared to the asymptotic minimal budget (3.1) obtained in Yang et al. (2024) for $\delta \to 0$, an additional term pops up in the non-asymptotic minimal budget (3.3), which is dominant when $dK > n\log(n/\delta)$. Our algorithm is based on ideas related to sub-sampling, in order to localize in a more efficient way the mean of each group. The possibility of performing sub-sampling enables us to bypass combinatorial problems arising in clustering with no-repeated measurements. Our algorithm has a quasi-linear complexity, and is also order-optimal for all δ, n, K, and d, for a broader family of problems defined below. From a technical perspective, our information-theoretical results use novel techniques as those combine arguments from high-dimensional statistics and from bandit theory.

Related literature in clustering. The problem of clustering a mixture of subGaussian is a classical problem, which has lead to a large literature both in statistics and in machine learning (Dasgupta, 1999; Vempala and Wang, 2004; Lesieur et al., 2016; Lu and Zhou, 2016; Diakonikolas et al., 2018; Regev and Vijayaraghavan, 2017; Giraud and Verzelen, 2019; Fei and Chen, 2018; Chen and Yang, 2021; Kwon and Caramanis, 2020; Segol and Nadler, 2021; Romanov et al., 2022; Liu and Li, 2022; Diakonikolas et al., 2023). In low-dimension and for large values of n, state-of-the art polynomial-time procedures for recovering the groups have been introduced by (Liu and Li, 2022), and are based on generalization of higher moments methods —see also (Diakonikolas et al., 2018; Kothari and Steinhardt, 2017). In high-dimension, the best known conditions for exact reconstruction in polynomial-time are based on an SDP relaxation of K-means (Peng and Wei, 2007; Giraud and Verzelen, 2019). For $K = 2$, a simple Lloyd algorithm achieves perfect recovery at the information level (Ndaoud, 2022), thereby establishing the absence of computation-information gap for $K = 2$. For larger K, (Lesieur et al., 2016) conjectures a computation-information gap in high-dimension, and (Even et al., 2024) exhibits a low-degree computational barrier for the clustering of a mixture of isotropic Gaussians, when $d \geqslant n$. Some computation-information gaps have also been shown for Statistical-Query algorithms for learning mixture of non-isotropic Gaussian, with unknown covariance, in moderately high-dimension — see Diakonikolas et al. (2017, 2023). In the sequel, we refer to clustering with no repeated measurements as batch clustering.

Sequential literature related to CBP. When turning to the sequential learning literature, the CBP belongs to the family of pure exploration problems in the sequential active learning framework. An iconic such problem is the best-arm identification problem — see (Jamieson and Nowak, 2014) for a survey. In this stream of literature, the Thresholding Bandit Problem (TBP) is quite related — see (Chen and Li, 2015; Chen et al., 2014; Locatelli et al., 2016). This is a specific instance of our setting in dimension $d = 1$ and for two groups, i.e., $K = 2$. In this active binary classification problem, the learner aims at finding the arms that have a mean larger than a given threshold (here $d = 1$), and to divide them in $K = 2$ groups. Note that (Katariya et al., 2018) propose a generalization of these ideas to multiple groups, albeit still in dimension 1. The optimal asymptotic budget $\mathcal{T}$ for perfect recovery in the TBP is $\Delta_*^{-2} n \log(1/\delta)$ when δ goes to 0, and there are no computational gaps, see (Tirinzoni and Degenne, 2022) for state-of-the-art results on TBP.

The CBP, first introduced in (Yang et al., 2024), can be seen as a generalization of the TBP in dimension d. This generalization is highly non-trivial: subtle phenomenons make clustering problems with $d \geq 2$ very different from clustering in dimension 1. (Yang et al., 2024) provides an algorithm called BOC, which perfectly recovers the groups with probability higher than $1 - \delta$, and which has an expected budget at most of the order (3.1) in the asymptotic regime where δ goes to zero. Note that this rate is reminiscent of the TBP (where $d = 1$, $K = 2$). A closer look at the proofs in (Yang et al., 2024) exhibits an exponential dependence of second-order terms (in δ) on K, d. Then, Algorithm BOC — or at least its current analysis — is effective only in the asymptotic regime, when K, d are considered as being constants. In fact, since the oracle version of BOC samples equally all the arms when the clusters are balanced and equidistant, the BOC budget in this case is at least (3.2) in high-dimension (Even et al., 2024), which is suboptimal. Our non-asymptotic analysis allows to recover the shape of the optimal budget in the so-called high-dimensional regimes where d or K are not considered as constants. Quite recently,Yavas et al. (2025) have extended the analysis of Yang et al. (2024) to other distributions beyond subGaussian ones.

A somewhat related problem was studied in (Yun and Proutière, 2019), in the Stochastic Block Model within the fixed-budget setting. To extract hidden structure, the interaction between pairs of nodes can be sampled several times, in an active manner. The setting is however quite distinct from our work, and is also focusing on the asymptotic regime where δ goes to 0. In (Ariu et al., 2024), the related problem of clustering items based on binary feedback is studied — but therein, the feedback corresponds to a single coordinate of a chosen vector. In our work, we observe the full d-dimensional vector at each time, so that the settings differ. Finally, it is worth mentioning that our problem should not be confused with that of online clustering, for example studied in (Cohen-Addad et al., 2021).

Outline. We formally introduce the CBP in Section 3.2. An information-theoretical lower bound on the minimal budget for exact recovery is established in Section 3.3. We introduce and analyze our procedure ACB in Section 3.4. Numerical experiments are provided in Section 3.5. All the results are discussed in Section 3.6.

3.2 Setting and notation

The sequential and active setting. We consider a set of n arms, indexed by $[n]$. Each arm $a \in [n]$ is associated to an unknown probability distribution ν_a on $\mathbb{R}^d$. We refer to $\nu = (\nu_a)_{a\in[n]}$ as the environment. At each time t, the learner chooses an arm $A_t \in [n]$ based on the past observations. Conditionally on the chosen arm A_t, she receives from the environment a random observation $X_t \in \mathbb{R}^d$, distributed as ν_{A_t}.

For each arm $a \in [n]$, we write $\mu_a \in \mathbb{R}^d$ for the mean of the distribution ν_a. Both in the context of multi-armed bandits, and in the context of clustering, it is common to assume that the distributions are subGaussian.

Assumption 3.2.1 (σ-subGaussian arm observations)**.** *For any arm $a \in [n]$, we assume that there exists a symmetric $d \times d$ matrix Σ_a such that, (i) $\max_{a\in[n]} \|\Sigma_a\|_{op} \leq \sigma^2$, where $\|.\|_{op}$ is the operator norm; (ii) the coordinates (E_i) of $E = \Sigma_a^{-1/2}[X - \mu_a]$ are independent and fulfills $\mathbb{E}[\exp(tE_i)] \leq \exp(t^2/2)$ for all $t \in \mathbb{R}$.*

Remark 3.2.2. This assumption encompasses the emblematic settings where the data are Gaussian, and where the data are bounded. If the distributions (ν_a) are Gaussian, then Assumption 3.2.1 holds by e.g., choosing Σ_a's to be the covariance matrices, and associate σ. If the distributions $(\nu_a)_a$ are such that the coordinates are independent and lie in $[0, 1]$, the collection (ν_a) is 1/4-subGaussian.

Clustering with Bandit Feedback. As for the vanilla clustering problem, our objective is to partition the set of arms into groups of arms that share the same expectation μ_a. For this purpose, we make the following modeling assumption.

Assumption 3.2.3 (Hidden partition $\mathcal{C}^*$ of the arms into K groups)**.** *Consider $n \geq K \geq 1$. We assume that there exists a partition $\mathcal{C}^* = \{\mathcal{C}_1^*, \ldots, \mathcal{C}_K^*\}$ of $[n]$ into K groups such that any two arms a and b are in the same group if and only if they share the same expectation ($\mu_a = \mu_b$). For notation purpose, we introduce the vectors $\mu(1), \ldots, \mu(K) \in \mathbb{R}^p$ such $\mu(k)$ corresponds to the common expectation in $\mathcal{C}_k^*$. Henceforth, $\mu(k)$ is called the center of the group $\mathcal{C}_k^*$.*

In CBP, the goal of the learner is to uncover the true partition $\mathcal{C}^*$ of the arms, while using as few samples as possible. The learner samples arms sequentially and, when reaching some stopping time $\mathcal{T}$, she returns a partition $\hat{\mathcal{C}}$ of $[n]$ into K groups, which should ideally be equal to $\mathcal{C}^*$. More precisely, let π be an algorithm for the clustering problem with bandit feedback, also called the strategy of the learner. We write $(\mathcal{F}_t)_{t\geq 0}$ for the filtration $\mathcal{F}_t = \sigma(A_1, X_1, \ldots, A_t, X_t)$. A strategy π consists on three rules:

— A **selection rule** that chooses the next arm A_t to sample, based on the previously sampled arms and observations; A_t is $\mathcal{F}_t$-measurable.

— A **stopping rule** that controls when the learner stops sampling the arms, and which quantifies the budget of the strategy. This is modeled by a stopping time $\mathcal{T}$ with respect to the filtration $(\mathcal{F}_t)_{t\geq 0}$.

— A **recommendation rule** that outputs an estimated partition of the arms $\hat{\mathcal{C}}$, once the stopping time $\mathcal{T}$ is reached, the learner. This partition is $\mathcal{F}_{\mathcal{T}}$-measurable.

For an environment ν and an algorithm π, we write $\mathbb{P}_{\pi,\nu}$ for the probability induced by the interaction between the algorithm π and the environment.

In this Chapter, we aim at exactly recovering the partition $\mathcal{C}^*$ in the fixed confidence setting. While the partition $\mathcal{C}^*$ is identifiable, the groups $(\mathcal{C}_k^*)$ and the means $\mu(k)$ are identifiable only up to relabelling, i.e., up to a permutation of $[K]$. We denote by $C \sim C'$ two equivalent partitions of $[n]$, i.e., two partitions such that, for some permutation ρ of $[K]$, $C_k = C'_{\rho(k)}$ for all $k \in [K]$. For a fixed confidence level $\delta \in (0,1)$, and a given set of environments $\mathcal{E}$, a strategy $\pi = \pi(\delta)$ fulfilling

$$\mathbb{P}_{\pi,\nu}(\hat{\mathcal{C}} \sim \mathcal{C}^*) \geqslant 1 - \delta \ , \tag{3.4}$$

is said to be δ-correct on $\mathcal{E}$. We write $\Pi(\delta, \mathcal{E})$ for the family of such δ-correct strategies for the CBP on $\mathcal{E}$. Our aim is to design a δ-correct algorithm, whose budget $\mathcal{T}$ is as small as possible. For a family of environments $\mathcal{E}$, the optimal worst case (average) budget $T^*(\delta, \mathcal{E})$ is defined as

$$T^*(\delta, \mathcal{E}) = \inf_{\pi \in \Pi(\delta,\mathcal{E})} \sup_{\nu \in \mathcal{E}} \mathbb{E}_{\pi,\nu}[\mathcal{T}] \ . \tag{3.5}$$

In order to introduce relevant sets of environments $\mathcal{E}$, we introduce two quantities that characterize the difficulty of a clustering problem, let it be batch or active. First, we consider the minimal Euclidean distance between two distinct group centers

$$\Delta_* = \Delta_*(\nu) = \min_{k \neq k'} \|\mu(k) - \mu(k')\| > 0 \ . \tag{3.6}$$

Intuitively, the smaller Δ_*, the more difficult it is to distinguish the groups and to recover the partition $\mathcal{C}^*$. This quantity naturally appears in most clustering works in the batch setting (Dasgupta, 1999; Vempala and Wang, 2004; Giraud and Verzelen, 2019). Besides, we denote θ_* the balancedness of $\mathcal{C}^*$, that is the proportion of arms in the smallest cluster

$$\theta_* = \min_{k \in [K]} \frac{|\mathcal{C}_k^*|}{n} \ \in \left[\frac{1}{n}, \frac{1}{K}\right] . \tag{3.7}$$

When $\theta_* = 1/K$, all the groups $\mathcal{C}_k^*$ share the same size, and the partition is balanced.

Consider $\Delta > 0$, and $\theta > 0$, we define the set $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$ as the family of environments with n arms, divided into K groups as in Assumption 3.2.3, with a minimal gap Δ_* at least Δ, a balancedness θ_* at least θ, and with d-dimensional observations that are σ-subGaussian — see Assumption 3.2.1. Our main aim is to craft polynomial-time algorithms that attain the optimal worst case budget $T^*(\delta, \mathcal{E}(\Delta, \theta, \sigma, n, K, d))$, and to characterize this optimal worst-case budget.

3.3 Lower bound on the budget

We start by establishing a lower bound for the expected budget of any δ-correct algorithm over $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$.

Theorem 3.3.1. *There exists a numerical constant $c > 0$, such that we have for any $\sigma > 0$, any $\Delta > 0$, any $d \geq 1$, any $\theta > 0$, any $\delta \in (0, 1/12)$, and any $n \geqslant 2K \geq 4$ such that $\mathcal{E}(\Delta, \theta, \sigma, n, K, d) \neq \emptyset$*

$$T^*(\delta, \mathcal{E}(\Delta, \theta, \sigma, n, K, d)) \geqslant cn + c\frac{\sigma^2}{\Delta^2}\left[n \log\left(\frac{n}{\delta}\right) + \sqrt{dnK \log\left(\frac{n}{\delta}\right)}\right] \quad . \tag{3.8}$$

The lower bound in (3.8) involves three different terms. As in any pure exploration problem, the first term n is necessary because, when $\mathcal{T} \leqslant n/2$, then the label of at least one arm has to be guessed randomly inducing a constant probability of error for the exact clustering. This term is only relevant for very large Δ and is not discussed further. The second term is the largest in the low-dimensional regime where $d \leq n \log(n/\delta)/K$, whereas the third one is the largest in the high-dimensional regime where $d \geq n \log(n/\delta)/K$. This dichotomy between low-dimensional and high-dimensional clustering problems also occurs in the batch problem. Together with the results of the next section, we will establish that it is intrinsic here —see the discussion and the proof sketch for further details. Note that (3.8) does not depend on θ: we establish (3.8) for environments where θ_* is close to $1/K$, that is for balanced partitions. In fact, the total budget of our procedures ACB and ACB* — see below — does not depend on θ_* except for extremely unbalanced partitions (very small θ_*) so that the lower bound is tight even for mildly unbalanced partitions.

Sketch of proof of Theorem 3.3.1. The first two terms in the lower bound (3.8) — $\frac{\sigma^2}{\Delta^2} n \log\left(\frac{n}{\delta}\right)$ and $\frac{\sigma^2}{\Delta^2}\sqrt{dnK \log\left(\frac{n}{\delta}\right)}$ — are proved separately in Lemmas 3.B.1 and 3.B.2. Regarding the first term, we first observe that it depends neither on d, nor on K, nor on θ. For the sake of this sketch, we can therefore restrict ourselves to a one-dimensional ($d = 1$) multi-armed bandit setting where each arm has $a \in [n]$ has either mean $\mu_a = 0$ or $\mu_a = \Delta$, so that $K = 2$. For this simplified toy problem, recovering the partition $\mathcal{C}^*$ is equivalent to a Thresholding Bandit Problem (TBP), where the goal is to find the set of arms whose mean is higher or equal to Δ. By building upon some ideas introduced in (Cheshire et al., 2020), we establish the lower bound $\frac{\sigma^2}{\Delta^2} n \log\left(\frac{n}{\delta}\right)$. Note that one may easily interpret this quantity using the fact that, for a specific arm, deciphering whether the mean of a specific arm is 0 or Δ with probability $1 - \delta/n$, one needs to sample it at least $\frac{\sigma^2}{\Delta^2} \log\left(\frac{n}{\delta}\right)$ times.

The proof of the second term is both more challenging and more innovative. Again, for the purpose of this sketch, let us assume that $K = 2$ and $\theta_* = 1/2$. We use a Bayesian approach by putting a Gaussian prior distribution on $\mu(1)$ with variance $d^{-1/2}\Delta I_d$ and by fixing $\mu(2) = -\mu(1)$ so that, with high probability, $\|\mu(2) - \mu(1)\| \geq \Delta$. Introducing this prior distribution on $\mathbb{R}^d$ is instrumental to recover the dependency of the budget on the dimension d of the problem. First, we use the symmetry of the problem to show that the optimal budget is achieved by a strategy π which, in expectation, samples all the arms uniformly. Then, we use a series of reduction. First we prove that identifying the group of any node a is, in some sense, at least as difficult, as the supervised problem where we would know the group of all the arms, except that of a. In turn, we show that tackling this active supervised problem with a uniform strategy π is as difficult as tackling a batch supervised learning problem where each arm is sampled $\mathcal{T}/n$ times. Finally, we craft an impossibility result for the latter problem. We emphasize that there is no computational

restriction here, so that the lower bound for uniform sampling strategies is (3.8), and not the rate (3.2) which relates to polynomial-time algorithms (Even et al., 2024). □

3.4 ACB and Upper bound on the budget

To introduce the main ideas underlying our algorithm, we first assume in the next subsection that Δ, θ, σ, n, K and d are known quantities. Based on this oracle knowledge, we construct an algorithm, ACB, that is δ-correct for environments such that $\Delta_* \geqslant \Delta$ and $\theta_* \geqslant \theta$. We introduce our main algorithm, ACB*, adaptive to Δ_* and θ_* in Subsection 3.4.2.

3.4.1 Warm-up: optimal clustering with known Δ, θ

The main recipe of ACB is a two-step procedure. First it identifies a set $\hat{S}$ of K arms, which are used as representatives of each group. Then, it classifies all the arms based on a precise estimation of the means of the K arms in $\hat{S}$. The ACB algorithm built then on two subroutines:
1- SRI (Sequential Representatives identification), which constructs a set $\widehat{S}$ that contains, with high probability, exactly one arm for each group, called the representatives of each group. To construct $\widehat{S}$, we use a sequential elimination technique, combined with high-dimensional two-sample tests.
2- ADC (Active Distance-based classification), which computes precise estimates of the means of the arms in $\widehat{S}$, and classifies the remaining arms based on minimum estimated distance to the representatives.

Estimating distances. In order to detect whether two arms a and b are in the same group, a key ingredient for both SRI and ADC is to get a good estimation of the square distance $\|\mu_a - \mu_b\|^2$ between the means. Computing the empirical means $\hat{\mu}_a$ and $\hat{\mu}_b$ of collected samples of a and b, we can estimate $\|\mu_a - \mu_b\|^2$ by $\|\hat{\mu}_a - \hat{\mu}_b\|^2$. Yet, this simple estimator suffers from an unknown bias depending on the noise covariance matrix. This issue can be circumvented in active sampling, by:
(i) computing independent empirical means $\hat{\mu}_a$, $\hat{\mu}'_a$, and $\hat{\mu}_b$, $\hat{\mu}'_b$ for the arms a and b, based on repeated measurements,
(ii) estimating $\|\mu_a - \mu_b\|^2$ with the unbiased estimator

$$\hat{d}^2_{ab} = \langle \hat{\mu}_a - \hat{\mu}_b, \hat{\mu}'_a - \hat{\mu}'_b \rangle \ . \tag{3.9}$$

The construction of this estimator belongs to the statistical folklore for the problem of estimating the square norm of the mean of a random vector — see e.g., CARPENTIER (2015) for a previous occurrence. In the simpler case where the covariance structure would be known, one could instead use the simpler estimator from Collier and Dalalyan (2019).

SRI subroutine (Sequential Representative Identification). The core idea underlying the SRI subroutine is to start from a set $S = \{a_0\}$ made of a single arm, chosen uniformly at random. Then SRI successively samples new arms a, and adds them to S if they pass a sequence of tests ensuring that a is not represented in S with high probability. The sequence of tests checks if a is already represented in S, i.e., if $\min_{b \in S} \|\mu_a - \mu_b\|^2 = 0$, by repeatedly checking if

$\min_{b\in S} \hat{d}^2_{ab} \leq \Delta^2/2$, with a sequence of estimators $\hat{d}^2_{ab}$ based on increasing sample sizes, ensuring increasing confidence. It is based on the call of the RepresentedTest subroutine described below, where empirical_mean(a, l) refers to the action of sampling l times the a-th arm, and computing the empirical mean of the collected samples. This action is performed twice to compute $\hat{\mu}_a$ and $\hat{\mu}'_a$.

Function `RepresentedTest`$(a, (\bar{\mu}_b, \bar{\mu}'_b)_{b\in S}, \Delta, l)$: ▷ Test if a is rep. in S
  $\hat{\mu}_a, \hat{\mu}'_a \leftarrow$ empirical_mean(a, l);
  return Is.True$\left\{\min_{b\in S}\langle \hat{\mu}_a - \hat{\mu}_b, \hat{\mu}'_a - \hat{\mu}'_b\rangle \leq \frac{\Delta^2}{2}\right\}$

More precisely, let us define

$$U := \left\lceil 8\theta^{-1}\log\left(8K/\delta\right)\right\rceil \quad ; \qquad r := \lceil \log_2(\log(4U/\delta))\rceil \, ; \tag{3.10}$$

$$n_s := \left\lceil c_1\frac{\sigma^2}{\Delta^2}\left(2^s + \log(12K)\right) \vee c_2\frac{\sigma^2}{\Delta^2}\sqrt{d(2^s+\log(6))}\right\rceil \quad ; \tag{3.11}$$

$$s_0 := r \wedge \min\{s \geqslant 1; n_s \geqslant 2\} \; ; \qquad n_{\max} := n_r \vee \left\lceil c_3\frac{\sigma^2}{\Delta^2}\sqrt{d}\log(2K)\right\rceil \quad , \tag{3.12}$$

$$T_{\max} = 2K\left(n_{\max} + \sum_{s=s_0+1}^{r} n_s\right) + 2Un_{s_0} + 2U\sum_{s=s_0+1}^{r}\frac{n_s}{2^{s-4}} \quad , \tag{3.13}$$

with c_1, c_2, $c_3 > 0$ numerical constants, explicitly provided in the proof of Lemma 3.C.2. The SRI procedure (Algorithm 2) successively samples candidate arms a_u at random, and performs a sequence of RepresentedTest with (roughly) doubling sample size n_s for $s = s_0, s_0+1, \ldots$, until either a RepresentedTest returns True, in which case the arm a_u is rejected (Line 8); or all tests up to $s = r$ have answered False, in which case the arm a_u is added to S (Line 11). The procedure SRI stops when $|S| = K$, or when a maximal budget has been spent ($T_{\max}$ is defined in (3.13)) and it returns $\widehat{S} = S$. The minimal index s_0 ensures that the sample sizes n_s are not smaller than 2.

The sequence of tests is designed in order to use few samples to reject arms already represented in S, while wrongly rejecting an unrepresented arm with probability less than $1/2$. Indeed, the choice of the sample sizes n_s and $n_{\max}$ ensures that the probability to take a wrong decision at the s-th step is smaller than 2^{-s-1}. Hence, the probability that an arm already represented in S is rightly rejected before step s is at least $1-2^{-s}$, leading to a quick rejection with high-probability. In addition, the maximum sample size n_r is chosen large enough, to ensure a vanishing small probability of (wrongly) not rejecting such an arm. As for unrepresented arms, the probability to wrongly reject an arm a_u not already represented in S is smaller than $\sum_{s\geq 1} 2^{-s-1} = 1/2$. Then, with probability at least $1-\delta/4$, we need less than U candidate arms to identify one representative of each group.

We provide further guarantees on SRI subroutine in Section 3.C, Lemma 3.C.2. In particular, if S is the output of SRI applied with parameters Δ and θ, then with probability larger than $1-\delta$,

Algorithm 2: Procedure SRI (Sequential Representative Identification)

Procedure `SRI`(δ, Δ, θ)**:**
  Result: S a set of arms
  Compute $U, r, s_0, n_s, n_{\max}, T_{\max}$ according to (3.10) — (3.13) and sample $a_0 \in [n]$;
  Set $S = \{a_0\}$ and $\hat{\mu}_{a_0}, \hat{\mu}'_{a_0} \leftarrow$ empirical_mean$(a_0, n_{\max})$
  for $u = 1, \dots, U$ **do**
    Sample $a_u \in [n]$
    for $s = s_0, \dots, r$ **do**
      if *RepresentedTest*$\left(a_u, (\hat{\mu}_b, \hat{\mu}'_b)_{b\in S}, \Delta, n_s\right)$ **then**
        BREAK ; ▷ reject a_u
      end
      if $s = r$ **then** ▷ if a_u has passed all tests
        $S \leftarrow S \cup \{a_u\}$ ▷ Add a_u to S
        $\hat{\mu}_{a_u}, \hat{\mu}'_{a_u} \leftarrow$ empirical_mean$(a_u, n_{\max})$ ▷ Estimate μ_{a_u}
      end
    end
    if $|S| = K$ *or budget* $> T_{\max}$ **then**
      BREAK ▷ Terminate u loop
    end
  end
  return S ▷ Return a representative for each group

(a) S does not contain two arms from the same group. Moreover, if the true parameters Δ_* and θ_* are smaller than Δ, θ, then S contains exactly K arms, with one arm from each group. We also provide an upper bound on the budget used by SRI.

ADC subroutine (Active Distance-based Classification). Once a set $\widehat{S} = \{b_1, \dots, b_K\}$ of representatives of each group has been successfully obtained with SRI, the mean of each group can be precisely estimated, and remaining arms can be classified based on distance estimation $\hat{d}^2_{ab}$ to these means. This classification is performed by the ADC subroutine.

Let us define

$$J := \left\lceil c_4 \frac{\sigma^2}{\Delta^2} L \vee c_5 \frac{\sigma^2}{\Delta^2} \sqrt{\frac{dn}{K}} L \right\rceil \quad , \qquad I := \left\lceil c_4 \frac{\sigma^2}{\Delta^2} L \vee c_5 \frac{\sigma^2}{\Delta^2} \sqrt{\frac{dK}{n}} L \right\rceil \quad , \tag{3.14}$$

with $L = \log(6nK/\delta)$, and c_4, c_5 two universal constants defined in the proof of Lemma 3.C.3. Assume, without loss of generality, that $b_j \in \mathcal{C}^*{}_j$ for all $j \in [K]$. Then, ADC first computes two precise estimations $\hat{\mu}(j), \hat{\mu}'(j)$ of the mean of arms in $\mathcal{C}^*{}_j$ (Line 7), each based on J samples of arm b_j. As these mean estimations are the references for the classification, the sample size J is chosen large enough to ensure a small variance. Then, for each arm a, two mean estimations $\hat{\mu}_a, \hat{\mu}'_a$ are computed based on I samples, and the arm a is classified Line 13 according to the smallest estimated distance (3.15). The budget I for individual mean estimation is much smaller than J in high-dimension d, with $I = KJ/n$ for d large. This budget ensures yet that the probability of

misclassifying an arm is smaller than δ/n. Overall, we prove that as long as the set S obtained in the first step contains exactly one arm from each cluster, then subroutine ADC will provide with high probability a perfect clustering of the arms. We summarize our guarantees on ADC in Lemma 3.C.3.

Algorithm 3: Procedure ADC (Active Distance-based Classification)

Procedure ADC(δ, Δ, S)**:**
  if $|S| \neq K$ **then**
    return Null ; ▷ Return Null if size of S is not K
  end
  Enumerate $S = \{b_1, \dots, b_K\}$, and compute I, J according to (3.14) ;
  for $j \in [K]$ **do**
    $\hat{\mu}(j), \hat{\mu}'(j) \leftarrow$ empirical_mean(b_j, J) ; ▷ Estimate the centers
    $\hat{\mathcal{C}}_j \leftarrow \{b_j\}$;
  end
  for $a \in [n] \setminus S$ **do**
    $\hat{\mu}_a, \hat{\mu}'_a \leftarrow$ empirical_mean(a, I) ;
    Add a to the group $\hat{\mathcal{C}}_k$ such that ; ▷ Classify arm a

$$k \in \operatorname*{argmin}_{j=1,\dots,K} \left\langle \hat{\mu}_a - \hat{\mu}(j), \hat{\mu}'_a - \hat{\mu}'(j) \right\rangle \tag{3.15}$$

  end
  return $\{\hat{\mathcal{C}}_1, \dots, \hat{\mathcal{C}}_K\}$ ▷ Return a clustering

ACB algorithm. Combining the SRI and ADC subroutines, we get a simple clustering with bandit feedback algorithm ACB for the case where Δ_* and θ_* are known — see Algorithm 4.

Algorithm 4: ACB (θ_* and Δ_* known)

Input: δ, Δ, θ
$\hat{S} \leftarrow \text{SRI}(\delta/2, \Delta, \theta)$
return $\hat{\mathcal{C}} = \text{ADC}(\delta/2, \Delta, \hat{S})$

Algorithm 5: ACB* (θ_* and Δ_* unknown)

Input: δ
for $l = 0, 1, \dots$ **do**
  for $p = 0, \dots, l$ **do**
    Compute $S_{p,l} \leftarrow \text{SRI}(\delta_l, \Delta_p, \theta_{p,l} \vee \frac{1}{n})$
    if $|S_{p,l}| = K$ **then**
      for $a \in S_{p,l}$ **do**
        $\bar{\mu}_a, \bar{\mu}'_a \leftarrow$ empirical_mean(a, n'_p)
      end
      $\hat{\Delta}^2 \leftarrow \inf_{a,b \in S_{p,l}} \langle \bar{\mu}_a - \bar{\mu}_b, \bar{\mu}'_a - \bar{\mu}'_b \rangle$
      return $\hat{\mathcal{C}} = \text{ADC}(\delta/3, 2^{-1/2}\hat{\Delta}, S_{p,l})$
    end
  end
end

3.4.2 Main algorithm ACB*

When the parameters Δ_* and θ_* are unknown, we cannot rely on a single call to SRI and ADC as in the ACB algorithm. Multiscale calls to SRI are required, for different candidate levels Δ_p and $\theta_{p,l}$ for Δ_* and θ_*. These levels, related sample sizes n'_p, and confidence levels δ_l are defined by

$$\Delta_0^2 = \sigma^2[\log(K) + \sqrt{d} + \log\log(6n/\delta)], \qquad \delta_l = \frac{\delta}{6(l+1)^3} \tag{3.16}$$

$$\theta_{p,l} = \frac{1}{K2^{l-p}}, \quad \Delta_p = \Delta_0\sqrt{\frac{1}{2^p}} \quad n'_p = \left\lceil c_6 \frac{\sigma^2}{\Delta_p^2}\left(\log(3K^2/\delta) + \sqrt{d\log(3K^2/\delta)}\right)\right\rceil \tag{3.17}$$

where c_6 is a numerical constant, whose value is given in (3.56).

The main recipe in ACB*, is to scan decreasing candidate values Δ_p and $\theta_{p,l}$, until we find a scale where SRI returns a set $S_{p,l}$ of cardinality K, see Algorithm 5.

Below, we provide upper bounds on $T^*(\delta, \mathcal{E}(\Delta, \theta, \sigma, n, K, d))$ for both ACB and ACB*. We write $\mathcal{T}_{\mathrm{ACB}}$ and $\mathcal{T}_{\mathrm{ACB}^*}$ for the budget of the non-adaptive procedure $\mathrm{ACB}(\delta, \Delta, \theta)$, and of the adaptive one $\mathrm{ACB}^*(\delta)$. Define the quantities

$$\begin{aligned} A =& \frac{\sigma^2}{\Delta^2}\left[n\log(n/\delta) + \sqrt{dnK\log(n/\delta)} + \sqrt{d}\frac{\log(K)}{\theta}\right] \\ B =& \frac{1}{\theta}\log(K/\delta) + \frac{\sigma^2}{\Delta^2}\frac{1}{\theta}\log\left(\frac{K}{\delta}\right)\left[\sqrt{d} + \log\log(n/\delta)\right] \quad . \end{aligned}$$

Theorem 3.4.1. *Let $\delta > 0$. Let $\Delta > 0$, $\theta > 0$ be any two parameters such that $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$ is non-empty. Both the ACB (Algorithm 4) and its adaptive version ACB* (Algorithm 5) are δ-correct on $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$. There exist numerical constants c, c', c'', independent of all the parameters $\Delta, \theta, \sigma, n, K, d$ such that the following holds.*

For any environment ν in $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$, such that $\theta \geq \log(K)/n$, we have

$$\mathbb{E}_{\mathrm{ACB},\nu}[\mathcal{T}_{\mathrm{ACB}}] \leqslant cn + c'A \ ; \qquad \mathcal{T}_{\mathrm{ACB}} \leqslant cn + c'(A+B) \ \text{a.s.}$$
$$\mathbb{P}_{\mathrm{ACB}^*,\nu}\left[\mathcal{T}_{\mathrm{ACB}^*} \leqslant cn + c''L\log^2(L)(A+B)\right] \geq 1-\delta$$

where $L := \log_2\left(\frac{1}{\theta K}\left(\frac{\Delta_0^2}{\Delta^2} \vee 1\right)\right)$.

This theorem entails that the budget for both ACB and ACB* is optimal. We further comment on this in the discussion section.

3.5 Numerical experiments

In this section, we run experiments on synthetic data. In Figure 3.1, the dimension is fixed to $d = 1000$, and the number of clusters varies in $\{10, 15, 20, 25\}$. In Figure 3.2, the number of clusters is fixed to $K = 15$, and the dimension varies in $\{500, 1000, 1500, 2000\}$. We compare the budget of ACB to oracle-BOC, an oracle version of the BOC algorithm (Yang et al., 2024).

Competitor As the main competitor, we implement oracle-BOC, an oracle version of the BOC algorithm (Yang et al., 2024). As the setting is perfectly symmetric (balanced clusters, equidistant means), the Oracle-BOC policy is equivalent to the Uniform Sampling strategy, where Loyd algorithm initialized by `maximin`. We implement instead a `kmeans++` initialization, as it is known to outperform `maximin` (Celebi et al., 2013). Besides, the total budget of oracle-BOC is chosen in such a way that the procedure is empirically δ-correct. As a consequence, oracle-BOC both corresponds to a state-of-the art batch clustering procedure and to an oracle version of Yang et al. (2024) where both the stopping time and the sampling strategy are provided by an oracle.

We run the non-adaptive procedure ACB and a variation of ACB* called ACB†, which is adaptive to the unknown parameter Δ_*. Its structure is very similar to Algorithm 5 for ACB*, with the difference of assuming that θ is known, we provide more explanations in Section 3.A. In this experiment, we use the variant ACB† to allow for a more fair comparison to Oracle-BOC. Regarding ACB, we assume that Δ_* is known, and we implement the non-adaptive version of ACB with $\delta = 0.1$. In order to provide a tighter calibration of ACB, we slightly modify ACB algorithm in order to specialize it to the Gaussian distribution —see Appendix 3.A.

First experiment: varying the number of clusters. To illustrate our theory, we consider environments with standard Gaussian noise ($\sigma = 1$), with equidistant centers (with $\Delta_* = 1$), and balanced groups ($\theta_* \approx 1/K$). We choose a high-dimensional setting with $n = 200$, $d = 1000$, and $K \in \{10, 15, 20, 25\}$.

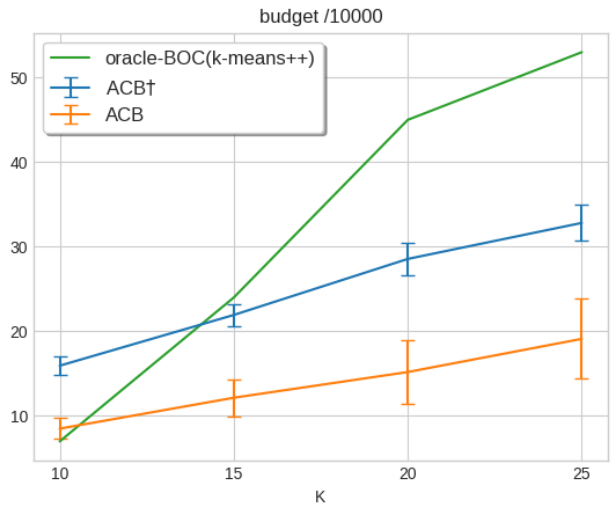


Figure 3.1 – Comparison of the necessary budget for ACB and oracle-BOC with varying number of clusters.
We represent in blue (resp. orange) the (empirical) budget of ACB† (resp. ACB) computed with 100 simulations, for $K = 10, 15, 20, 25$. The error bars are equal to twice the standard deviation. In green, we provide the smallest budget for which oracle-BOC (initialized with `kmeans++`) makes less than 10% of error out of 100 experiments. As this budget is a numerical constant, there are no error bars.

In Figure 3.1, we plot the estimated mean budget of ACB† and ACB as a function of K. We also plot the budget of oracle-BOC, where the budget has been chosen by an oracle so that the procedure is exactly δ-correct with $\delta = 0.1$. This figure confirms our theoretical findings that, in a high-dimensional setting ($d \gg n/K$), ACB improves over oracle-BOC — which is here equivalent to a state-of-the-art batch clustering algorithm — when the number K of groups increases. Also,

we have checked that ACB† and ACB are δ-correct. Fixing $\delta = 0.1$, we observe no more than 1 error out of 100 experiments. We detail further the experimental setup (including compute resources) in Appendix 3.A.

Second experiment: varying the dimension. We provide also an experiment for which the number of clusters is fixed, and the dimension varies. We consider artificial data, generated with standard Gaussian noise ($\sigma = 1$). We build environments with equidistant centers, and balanced groups. Precisely, we choose $\mu(k) = e_k/\sqrt{2}$, where $\{e_1, \dots, e_K\}$ are the K first vector of the canonical base of $\mathbb{R}^d$, so that the centers are equidistant, and $\Delta_* = 1$. We choose a number of arms $n = 200$. We fix a partition where each group has a size $\lfloor n/K \rfloor$ or $\lfloor n/K \rfloor + 1$, which makes the partition almost balanced, with $\theta_* = \frac{1}{n}\lfloor \frac{n}{K} \rfloor \sim \frac{1}{K}$.

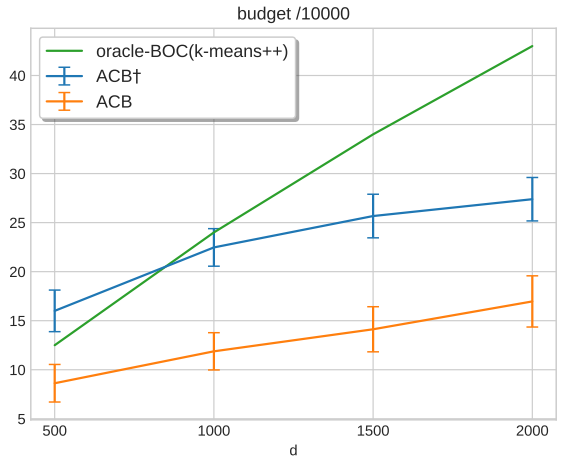


Figure 3.2 – Comparison of the necessary budget for ACB and oracle-BOC, with varying dimension.
In blue (resp. orange) the (empirical) budget of ACB† (resp. ACB) computed with 100 simulations, for $d = 500, 1000, 1500, 2000$. The error bars are equal to twice the standard deviation. In green, we provide the smallest budget for which oracle-BOC (initialized with `kmeans++`) makes less than 10% of error out of 100 experiments.

3.6 Discussion

Optimality of ACB. First, we discuss the budget of ACB, and we compare it to the information-theoretical lower bound of Theorem 3.3.1. To simplify the discussion, let us first consider the case where the partition $\mathcal{C}^*$ is almost balanced, that is when θ is of the order of $1/K$, and assume that $\frac{\Delta^2}{\sigma^2} \lesssim \log(n/\delta)$. According to Theorem 3.4.1, the δ-correct algorithm ACB has an expected budget upper bounded by (3.3), as long as $K \leq n/\log(n)$. In light of Theorem 3.3.1, we see that the expected budget is optimal with respect to all the quantities of the problem: the number of arms n, the minimum separation Δ, the number of groups K, the probability δ, and the subGaussian norm σ. The only restriction is that the number of groups K is smaller than $n/\log(n)$, but it is really mild as non-supervised learning problems are mostly relevant for dimension reduction, that is when K is much smaller compared to n. In fact, for larger $K \in [\frac{n}{\log(n)}, n/2]$, the expected budget $\mathbb{E}_{\mathrm{ACB},\nu}[\mathcal{T}_{\mathrm{ACB}}]$ is optimal, up to a possible $\sqrt{\log(n)}$ multiplicative term. Theorem 3.4.1 also states high probability controls of the budget $\mathcal{T}_{\mathrm{ACB}}$ and $\mathcal{T}_{\mathrm{ACB}^*}$ which again, are optimal (up to log terms for the latter), in most regimes.

When the true partition $\mathcal{C}^*$ is extremely unbalanced, that is for θ_* as small as $1/n$ but the dimension d is seen as a constant, our procedures turns out to still match the lower bound on the budget. When the true partition $\mathcal{C}^*$ is extremely unbalanced, so that $\theta_* \leq \frac{\log(K)}{\sqrt{\log(n/\delta)nK}}$ and the dimension d is really large, the bound A on the expected budget may be larger than the lower bound of Theorem 3.3.1. Note that this regime is extremely atypical for clustering problems. We conjecture that both the lower and the upper bounds could be improved in this extreme case, but we leave this for future work.

Further comparison with (Yang et al., 2024). When δ goes to zero while σ, Δ, K, and d are fixed, the average budget of ACB in (3.3) is at most of the order of $\frac{\sigma^2}{\Delta^2} n \log(1/\delta)$, and is consistent with the BOC algorithm of (Yang et al., 2024). Still, we mention that (Yang et al., 2024) manage to pinpoint the exact value of the asymptotic optimal budget, while our non-asymptotic bounds are only tight up to numerical constants. We however point out that this asymptotic expression hides dependencies on K, d, and n, which are not negligible unless δ is exponentially small with respect to d, K. In high dimension, such a high confidence regime is typically out of reach.

Comparison to batch clustering. We briefly come back to our fundamental questions on the comparison between the batch and clustering with bandit feedback problems. Contrary to the usual batch setting, we have established that the polynomial-time strategy ACB is information-theoretical optimal, thereby establishing the absence of computation-information gap. This is in contrast with the classical batch clustering problem, where strong evidence of a computation-information gap were proved in (Even et al., 2024) in high dimension, when there are many groups. We therefore illustrate here that clustering is an unsupervised learning problem, where repeated active sampling breaks a computational barrier, which is interesting and opens perspectives for other unsupervised clustering problems where computation-information gap are conjectured.

Conclusion and limitations. In this Chapter, we characterize the non-asymptotic minimal budget for recovering the groups in a collection of environment $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$. This corresponds to environments where the minimum distance between the groups is higher or equal to Δ, and all groups have a size larger than θ. We also crafted a strategy adaptive to both θ_* and Δ_*. Unlike in batch clustering, our results prove that there is no computation-information gap.

Our work still has limitations, and raises several open questions:

First, it remains to explore how sequential and active learning can be leveraged for adapting to heterogeneous distances between groups and heterogeneous group sizes. This has been investigated in (Yang et al., 2024) in the asymptotic regime, but not in the non-asymptotic regime. It would be interesting to have an algorithm which fully adapts to all inter-groups gaps — and not just to the minimal one, or to some target distance. This is a relevant and interesting direction for future works, but it goes beyond the present work whose main aim was to disprove the existence of a statistical-computational gap in sequential clustering. In our work, we also assume that the mean vector means within each group are exactly equal, as it allows us a simple comparison with batch clustering. An interesting way of relaxing this assumption would be to assume that the means within each group are not equal, but close within the groups. A recent work Chandran et al. (2025) explores this question in the asymptotic regime where δ goes to 0.

Second, when σ is unknown, building a sampling strategy that is adaptive to it, would require

to estimate the subGaussian norm of the noise, while at the same time estimating the distances between the means. We leave this question for a future work. Finally, as in most of the clustering literature, we assumed that the number K of groups was known to the learner. Investigating the problem of estimating or testing the number of groups in an active setting is also an interesting research direction.

Appendix of Chapter 3

3.A Details on the numerical experiments

Variant of the procedure and parametrization of ACB

We introduced and calibrated ACB to allow for subGaussian noise. In particular, the quantities n_s, $n_{\max}$, I and J, defined in eqs. (3.10) to (3.12) and (3.14) were calibrated by inverting concentration inequalities, at the cost of non-optimal numerical constants.

In order to study numerically our procedure, we implement a variant of the procedures whose tuning parameters is adjusted to the Gaussian setting. The test statistics and the classifier, so as the tuning parameters are adjusted to specifically work with Gaussian distribution.

The structure of SRI remains unchanged, however, we use the following calibration. In SRI, we avoid dual sampling when computing $\hat{d}_{ab}$ ((3.9)) estimate of $\|\mu_a - \mu_b\|^2$ in order to save a factor two in the budget. We modify indeed the test statistic in the function RepresentedTest used in SRI. We use Is.True $\left\{\min_{b\in S}\left[\|\hat{\mu}_a - \bar{\mu}_b\|^2 - d\sigma^2\left(\frac{1}{n_s}+\frac{1}{n_{\max}}\right)\right] \le \Delta^2/2\right\}$, so that there is no need to compute $\hat{\mu}'_a$ in RepresentedTest, and neither $\bar{\mu}'_b$ in SRI. Observe that $\|\hat{\mu}_a - \bar{\mu}_b\|^2$ is an estimator of $\|\mu_a - \mu_b\|^2$ which is biased. As in the experiment, the variance is known, we debias it, using the shift $d\sigma^2(\frac{1}{n_s}+\frac{1}{n_{\max}})$ in the statistics above.

In order to have a δ-correct algorithm, we choose $n_{\max} = 4\frac{\sigma^2}{\Delta^2}(x-d)$, where x is the $1-\delta/K$ quantile of a χ^2 distribution with d degrees of freedom. This quantile is obtained with the library scipy.stats. Now, we set $n_0, \dots, n_r$, by putting $n_s = \lceil 2^s n_0 \rceil$ for $s = 0, \dots, r$, for all s. We choose n_0 so that the budget spent on the rejected candidates should be close to the budget spent on the accepted representatives. We choose $n_0 = \lceil (K/U')n_{\max} \rceil$ where $U' = (1/\theta)\log(1/\delta)$. Finally, r is chosen such that $n_r = 2^r n_0$ is equal to $n_{\max}$, up to a factor 2.

To simplify the procedure, the stopping condition from Line 12 in SRI is modified, and we only stop when S contains K representatives, and not sooner.

In the Active Distance-based Classification routine (ADC), we also modify the sampling size I and J (Equation (3.14)), and the classifier from (13). In the classification, we label each arm with $\operatorname{argmin}_{j=1,\dots,K} \|\hat{\mu}_a - \hat{\mu}(j)\|$, which is a distance-based classifier as in Equation (3.15), but without dual sampling. By the analysis of the probability of error of this classifier, we choose

$$I = \left\lceil \frac{\sigma^2}{\Delta^2}\max(16\beta, 4\sqrt{2K/n\alpha}) \right\rceil \quad , \qquad J = \left\lceil \frac{\sigma^2}{\Delta^2}\max(16\beta, 4\sqrt{2n/K\alpha}) \right\rceil \quad ,$$

where β is the $1-\delta/(4K(n-K))$ quantile of a standard normal distribution (obtained with scipy.stats), and α is the $1-\delta/(4K(n-K))$ quantile of a product of independent standard $\mathcal{N}(0, I_d)$, that we had to compute empirically with Monte Carlo. With this choice of tuning parameters, one can prove that the corresponding variant of ACB is δ-correct for Gaussian data. The proof is analogous to the one in Section 3.C. Our numerical experiments confirm that, with these tuning parameters, the modified procedure is still δ-correct.

Description of the variant ACB$_\dagger$ and implementation

We now describe the variant ACB$_\dagger$ of ACB$_*$ used in the experiments. This version is calibrated to work well with balanced groups, or with a known balancedness θ. In ACB†, we assume no knowledge of Δ_*, and we perform SRI with growing values $(\Delta_k)_k$. We start with $\Delta_0 = \sqrt{4\sigma^2(x-d)}$, with x the $1-\delta/K$ quantile of a chi-square distribution with d degrees of freedom. We then use $\Delta_k = \Delta_0/2^k$ and $\delta_k = \delta/(6(k+1)^2)$.

First, for each call of SRI, we use the calibration of SRI described in the paragraph above, and we put a limit on its budget. We limit the budget of $\mathrm{SRI}(\delta_k, \Delta_k, \theta)$ by $T'_{\max} = \mathcal{T}_{\mathrm{ADC}}(\delta_k, \Delta_k) = (n-K)I + KJ$, which is the budget that we use to classify with $\mathrm{ADC}(\delta_k, \Delta_k)$, and where I, J are calibrated as in the paragraph above.

When we reach k such that $\mathrm{SRI}(\delta, \Delta_k, \theta)$ contains K arms, we estimate Δ_*, based on the data collected on this call of SRI. If S is the set of representatives identified with SRI, and $(\hat{\mu}_a)_{a\in S}$ are the estimates of the centers computed by SRI, we compute

$$\hat{\Delta}^2 = \operatorname*{argmin}_{a\neq b\in S} \left\{ \|\hat{\mu}_a - \hat{\mu}_b\|^2 - 2d\frac{\sigma^2}{n_{\max}} \right\} .$$

Then, ADC is applied with the parameter $\hat{\Delta}$.

Algorithm 6: ACB† (Δ_* unknown)

Input: δ, and θ
for $k = 0, 1, \ldots$ **do**
  Compute $S_k \leftarrow \mathrm{SRI}(\delta_k, \Delta_k, \theta \vee \frac{1}{n})$
  if $|S_k| = K$ **then**
    $\hat{\Delta}^2 \leftarrow \inf_{a,b\in S_k} \left\{ \|\hat{\mu}_a - \hat{\mu}_b\|^2 - 2d\frac{\sigma^2}{n_{\max}} \right\}$
    return $\hat{\mathcal{C}} = \mathrm{ADC}(\delta/3, \hat{\Delta}, S_k)$
  end
end

Experiments Compute Resource. We used for the experiment the version of Python Anaconda/3-5.1.0 and the scikit-learn/1.02 package. The experiment have been run in the cluster MESO@LR, working with CPUs of 4Gb and 8Gb. To give an idea on the computation cost, for $n, d, K = 200, 1000, 10$, each call for ACB takes approximately 6 minutes, while each call for ACB† took approximately 18 minutes. In total, the curve for ACB from Figure 3.1 took around 10 for each value of K and 30 hours for ACB†.

3.B Proof of the Lower Bound

Sketch of the proof

Throughout Section 3.B, we fix $\Delta > 0$, σ and d. In this section, we bound the worst case budget for any δ-correct algorithm on the collection of environments $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$ —see (3.5), and we prove Theorem 3.3.1.

We start in Section 3.B.1 by reducing the clustering with bandit feedback problem to a binary classification problem. For that purpose, we construct a family of environments, for which, the problem of clustering with bandit feedback essentially reduces to $\lfloor K/2 \rfloor$ independent and identical sub-problems of binary classification. The environments that we construct are symmetrical in some sense defined in the proof. We explain in Lemma 3.B.7 that we can find an optimal algorithm (as defined in Definition 3.B.6) that samples in expectation the same number of time each arm. This construction jointly deals with the low-dimensional (Lemma 3.B.1) and the high-dimensional (Lemma 3.B.2) regimes. For the construction, we will need to assume that $n \geqslant 2K$, that K is even, and that K divides n, and we explain in Lemma 3.B.4 how to reduce to this hypothesis.

We divide then the proof in two main lemmas, dealing with the low-dimensional and high-dimensional regimes. We recall that T^* — see (3.5) — is the optimal worst case budget.

Lemma 3.B.1. *If $n \geqslant 2K$, K is even, K divides n, and $\theta = 1/K$, then for any $\delta \in (0,1)$,*

$$T^*(\delta, \mathcal{E}(\Delta, \theta, \sigma, n, K, d)) \geqslant \frac{\sigma^2}{\Delta^2} n \,\mathrm{kl}\left(1-\delta, \frac{\delta}{n}\right) \quad ,$$

where kl *is the relative entropy defined as* $\mathrm{kl} : x, y \mapsto x \log(x/y) + (1-x)\log((1-x)/(1-\delta))$.

Lemma 3.B.2. *If $n \geqslant 2K$, K is even, K divides n, and $\theta = 1/K$, then for all $\delta \in (0, 1/6)$,*

$$T^*(\delta, \mathcal{E}(\Delta, \theta, \sigma, n, K, d)) \geqslant \frac{\sigma^2}{\Delta^2} \sqrt{\frac{dnK}{72} \,\mathrm{kl}\left(\frac{1}{3} - 2\delta, \frac{4\delta}{n}\right)} \quad .$$

In Section 3.B.2, we prove Lemma 3.B.1, the dimension-free lower bound. It is enough for this term to assume that the centers of the groups are known, and we use an information-theoretic method with the KL-divergence, which is somewhat related to previous works for the thresholding bandit problem derived by (Cheshire et al., 2020).

In Section 3.B.3, we prove Lemma 3.B.2 in the high-dimensional regime. For this purpose, we will consider a Bayesian setting and assume a Gaussian prior on the centers of the groups. The KL-divergence is hard to compute for the probability induced by the interaction between an algorithm and a Bayesian bandit environment. To overcome this technical problem, we formalize the following intuition. The clustering with bandit feedback Problem is "harder" than a supervised learning problem where the player knows the labels of every arm except one arm that has to be classified. It will reduce the problem into a two-sample (batch) testing problem (see Definition 3.B.13), and the conclusion will follow from some explicit computation and an impossibility result for this latter batch problem.

We postpone the proofs of some technical lemmas in Section 3.B.4

Remark 3.B.3. We explain quickly the term cn in the lower bound from Theorem 3.3.1. Assume that, for any $\nu \in \mathcal{E}(\Delta, \theta, \sigma, n, K, d)$, it holds that $\mathbb{E}_{\pi,\sigma}[\mathcal{T}] \leqslant cn$ with $c < 1/2$. Then, for any environment ν, there is a fixed probability that two arms from two different groups are not sampled at all during the procedure. The best to do for the learner is then to estimate randomly the groups of the arms, inducing a fixed probability of making at least one error in the clustering.

We do not discuss further this term cn in the lower bound in the remainder of the proof. Still, note that it is only relevant in an artificial regime where Δ is arbitrary large.

Now, we explain how the remark above, Lemmas 3.B.1 and 3.B.2 imply Theorem 3.3.1.

Proof of Theorem 3.3.1. Let n, K such that $n \geqslant 2K$. Let $\theta > 0$ such that $\mathcal{E}(\Delta, \theta, \sigma, n, K, d) \neq \emptyset$.

We first reduce the problem into a problem where K is even, n is a multiple of K, and the groups have the same size n/K. With this technical condition fulfilled, we will be able to Lemmas 3.B.1 and 3.B.2. We define n', K', and θ':

— if K is even, $K' := K$ and $n' := K\lfloor n/K \rfloor$;

— if K is odd, $K' := K - 1$ and $n' = K' \left\lfloor \frac{n - \lceil \theta n \rceil}{K'} \right\rfloor$;

— in both cases, $\theta' := 1/K'$.

We now use the following natural reduction result, whose proof is in Section 3.B.4.

Lemma 3.B.4. *The optimal worst case budget over $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$ is larger than the one over $\mathcal{E}(\Delta, \theta', \sigma, n', K', d)$,*

$$T^*(\delta, \mathcal{E}(\Delta, \theta, \sigma, n, K, d)) \geqslant T^*(\delta, \mathcal{E}(\Delta, \theta', \sigma, n', K', d)) \ .$$

It holds immediately that K' is even, and that K' divides n'. Moreover, $\lceil \theta n \rceil \leqslant n/K$ is a consequence of $\mathcal{E}(\Delta, \theta, \sigma, n, K, d) \neq \emptyset$. This inequality and the assumption $n \geqslant 2K$, implies that $n' \geqslant 2K'$. We can then use Lemma 3.B.1 and Lemma 3.B.2 with n' and K' in order to bound $T^*(\delta, \mathcal{E}(\Delta, \theta', n', K', d))$.

For any $\delta \in (0, 1/6)$,

$$T^*(\delta, \mathcal{E}(\Delta, \theta', \sigma, n', K', d)) \geqslant \frac{\sigma^2}{\Delta^2} n' \operatorname{kl}\left(1 - \delta, \frac{\delta}{n'}\right) \vee \frac{\sigma^2}{\Delta^2} \sqrt{\frac{dK'n'}{72} \operatorname{kl}\left(\frac{1}{3} - 2\delta, \frac{4\delta}{n'}\right)} \ .$$

We can also easily deduce from the expression of n' that $n' \geqslant n/6$.

Finally, we study $\delta \mapsto \operatorname{kl}(1 - \delta, 2\delta/n')$ to obtain the bound valid for all $\delta \in (0, 1)$ and for all $n' \geqslant 1$,

$$\operatorname{kl}\left(1 - \delta, \frac{2\delta}{n'}\right) \geqslant \log\left(\frac{1}{\delta}\right) + \log(n')(1 - \delta) - 1.5 \ .$$

In particular,we have the bound $\operatorname{kl}\left(1 - \delta, \frac{2\delta}{n'}\right) \geqslant \frac{1}{2}\log(n'/\delta)$ for $\delta \in (0, 1/4)$.

By studying the variation of $\delta \mapsto \operatorname{kl}(1/3 - 2\delta, 4\delta/n')$, we obtain the bound valid for all $\delta \in (0, 1/6)$ and for all n',

$$\operatorname{kl}\left(\frac{1}{3} - 2\delta, \frac{4\delta}{n'}\right) \geqslant \frac{1}{3}\left[\log\left(\frac{1}{4\delta}\right) + \log(n')(1 - 6\delta)\right] - 0.7 \ .$$

Combining all these inequalities and Remark 3.B.3, we obtain Theorem 3.3.1. □

3.B.1 From clustering with bandit feedback to binary classification

Construction of a family of environments. From now on, we assume that K is even, and n/K is an integer. In all the proof, we only consider perfectly balanced environments such that $\theta = 1/K$. We also assume that $n \geqslant 2K$. Define $L := \lfloor K/2 \rfloor$. In this subsection, we construct a family of environments defined with a prior on the centers of the groups.

We assume that the noises are Gaussian with covariance matrix $\sigma^2 I_d$. This fulfills the sub-Gaussian noise hypothesis from Assumption 3.2.1. In this Gaussian model, an environment is characterized by the hidden partition $\mathcal{C}^*$ and the (distinct) centers of the groups.

We use a Bayesian approach, and we define the $K = 2L$ centers of the groups, that we order as $\mu_{1,1}, \mu_{1,-1}, \ldots, \mu_{L,1}, \mu_{L,-1}$. For all $l \in [L]$, we construct the centers $\mu_{l,1}$ and $\mu_{l,-1}$ as symmetrical with respect to some offset. More specifically, for all $(l, g) \in [L] \times \{-1, 1\}$, we define

$$\mu_{l,g} := g\bar{\mu}(l) + C(l) \ , \tag{3.18}$$

where

— for all $l \in [L]$, $C(l) \in \mathbb{R}^d$ is a fixed offset defined as $C(l) = \beta(l\Delta, 0, \ldots, 0) \in \mathbb{R}^d$;

— $\beta > 1$ will be fixed later and is arbitrary large;

— $\bar{\mu} := \bar{\mu}(1), \ldots, \bar{\mu}(L)$ are i.i.d and $\bar{\mu}(l) \sim \gamma$. The prior distribution γ over $\mathbb{R}^d$ will be set differently if we consider the low or high-dimensional regime. We will specify later this prior.

Through the proof, we fix a partition $\mathcal{C}^*$ of $[n]$ into K groups. The partition $\mathcal{C}^*$ is composed of $K = 2L$ nonempty groups ${\mathcal{C}^*}_{1,1}, {\mathcal{C}^*}_{1,-1}, \ldots, {\mathcal{C}^*}_{L,1}, {\mathcal{C}^*}_{L,-1}$ associated to the means

$$\mu_{1,1}, \mu_{1,-1}, \ldots, \mu_{L,1}, \mu_{L,-1} \ .$$

For each arm $a \in [n]$, we denote as $(l_a^*, g_a^*) \in [L] \times \{-1, 1\}$ for the labels such that $a \in {\mathcal{C}^*}_{l_a^*, g_a^*}$ and $\mu_a = \mu_{l_a^*, g_a^*}$. Also, we will always restrict ourselves to balanced partitions $\mathcal{C}^*$ so that each group ${\mathcal{C}^*}_{l,g}$ has the same size n/K and thus $\theta_* = 1/K$.

In summary, we have

$$[n] = \bigsqcup_{(l,g) \in [L] \times \{-1,1\}} {\mathcal{C}^*}_{l,g} \ ,$$

where the groups $({\mathcal{C}^*}_{l,g})$ are nonempty and share the same size n/K.

We also define the so-called L "blocks". For $l \in [L]$, we define ${\mathcal{C}^*}_l := \{a \in [n] \ ; \ l_a^* = l\} = {\mathcal{C}^*}_{l,1} \sqcup {\mathcal{C}^*}_{l,-1}$. For each arm $a \in [n]$, l_a^* corresponds to the label of the pair of groups (block) ${\mathcal{C}^*}_l$ that contains a. If $l_a^* = l$, then the arm a belongs either to ${\mathcal{C}^*}_{l,1}$ or ${\mathcal{C}^*}_{l,-1}$ depending on the value of $g_a^* \in \{-1, 1\}$. We also denote as ${\mathcal{C}^*}_+ := \{a \in [L]; g_a^* = +1\}$.

We now construct a set of partitions obtained from $\mathcal{C}^*$ by switching two arms from the two different groups of the same block. Arbitrarily define a set $\{s(1), \ldots, s(L)\}$ of arms such that for all $l \in [L], s(l) \in {\mathcal{C}^*}_{l,-1}$. For any arm $a \in [n]$, we write $b_a := s(l_a^*)$. For an arm a in ${\mathcal{C}^*}_+ = \{a \in [L]; g_a^* = +1\}$, we define ${\mathcal{C}^*}_{(a)}$ as the partition equal to $\mathcal{C}^*$ except that the arm a is switched from $C_{l_a^*,1}$ to $C_{l_a^*,-1}$, and the arm b_a is switched from $C_{l_a^*,-1}$ to $C_{l_a^*,+1}$. This is a valid partition

with K nonempty and perfectly balanced groups. As we took $n \geqslant 2K$, it holds that, if any two distinct partition G and C' belong to $\{\mathcal{C}^*\} \cup \{\mathcal{C}^*{}_{(a)}\}_{a \in \mathcal{C}^*{}_+}$, we have $C \not\sim C'$. As a consequence, any δ-correct algorithm distinguishes, with probability higher than $1-\delta$, whether the environments are characterized by a partition $\mathcal{C}^*$ or by some $(\mathcal{C}^*{}_{(a)})_{a \in \mathcal{C}^*{}_+}$.

For any partition C' such that $[n] = \sqcup_{l,g} C'_{l,g}$, we denote as $\nu(C', \bar{\mu})$ for the environment constructed in this paragraph with the means $(\mu_{l,g})_{l,g} = (C(l) + g\bar{\mu}(l))$ and $\bar{\mu} \in \mathbb{R}^d$. We will use $\mathbb{P}_{\pi, C', \bar{\mu}}$ [resp. $\mathbb{E}_{\pi, C', \bar{\mu}}$] for the probability distribution [resp expectation] induced by the interaction between an algorithm π and the environment $\nu(C', \bar{\mu})$ for a fixed realization of $\bar{\mu}$. We also denote as $\mathbb{P}_{\pi, C'} = \int_{\bar{\mu}} \mathbb{P}_{\pi, C', \bar{\mu}} \, \mathrm{d}\gamma^{\otimes L}(\bar{\mu})$ [resp. $\mathbb{E}_{\pi, C'}$] as the integrated probability with respect to the prior $\gamma^{\otimes L}$ on $\bar{\mu}$ [resp expectation].

There is a technical detail that has to be handled with this Bayesian prior, if $\bar{\mu}_l$ is too small or too large, the environment $\nu(C', \bar{\mu})$ is not necessary in $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$. We define therefore $\mathcal{Y} := \bigcap_{l \in [L]} \{\Delta/2 \leqslant \|\bar{\mu}(l)\| \leqslant \Delta(\beta - 1)/2\}$. On $\mathcal{Y}$, the centers are distinct, the minimal gap is larger than Δ, and the set of possible values for $(\bar{\mu}(l))_l$ are disjoint.

We denote $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)$ as the Bayesian family of environments of the form $\nu(C', \bar{\mu})$, where $\bar{\mu} \sim \gamma^{\otimes L}$ and the partitions $C' \in \{\mathcal{C}^*\} \cup \bigcup_{a \in \mathcal{C}^*{}_+} \{\mathcal{C}^*{}_{(a)}\}$.

We explain a bit more the construction.

Remark 3.B.5.

1. The parameter β will be arbitrary large so that it is very easy to decide if two arms belong to different *blocks* or not. In this case, it is intuitively easy to first separate the arms into L blocks (that means to estimate $l_1^*, \dots, l_n^*$). Then the difficulty of the problem mostly lies in the L sub-problems of binary classification, where each block has to be partition into two groups.

2. In the low-dimensional regime, we will take $\bar{\mu}(l) = (\Delta/2, 0 \dots, 0)$ (γ is deterministic). It means that we will derive the lower bound from Lemma 3.B.1 for fixed centers of the groups $\mu(1), \dots, \mu(K)$ which basically amounts to the simpler setting where the learner knows the centers in advance.

3. In the high-dimensional regime, we will use a Gaussian prior on $(\bar{\mu}(l))_{l \in [L]}$. With this prior, we will be able to quantify to what extent we have to estimate the unknown means $(\bar{\mu}(l))_{l \in [L]}$ to be able to group the arms.

Symmetrization. Now, we exploit the different symmetries of the environments of the shape $\nu(C', \bar{\mu})$, and the symmetries of the distribution of the centers when $\bar{\mu} \sim \gamma$. Then, we restrict our study to algorithms that are δ-correct on $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)$ and that satisfies a symmetry property defined below.

Definition 3.B.6. Algorithm π is δ-correct on $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)$, if, conditionally on the event $\mathcal{Y}$, we have

$$\mathbb{P}_{\pi, C'}(\hat{\mathcal{C}} \sim C' | \mathcal{Y}) \geqslant 1 - \delta \ ,$$

for any $C' \in \{\mathcal{C}^*\} \cup \{\mathcal{C}^*{}_{(a)}\}_{a \in \mathcal{C}^*{}_+}$. An algorithm π is called symmetric on $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)$, if for any

$b \in [n]$ and $C' \in \{\mathcal{C}^*\} \cup \{\mathcal{C}^*{}_{(a)}\}_{a \in \mathcal{C}^*{}_+}$, then

$$\mathbb{E}_{\pi,C'}[N_b|\mathcal{Y}] = \frac{1}{n}\mathbb{E}_{\pi,C'}[\mathcal{T}|\mathcal{Y}] = \frac{1}{n}\mathbb{E}_{\pi,\mathcal{C}^*}[\mathcal{T}|\mathcal{Y}] \ ,$$

where $N_b = \sum_{t=1}^{\mathcal{T}} \mathbb{I}\{A_t = b\}$ is the number of times that the arm b is pulled during the procedure.

Denote as $\Pi_{Sym}(\delta, \mathcal{E}_{Sym}(\mathcal{C}^*, \gamma))$ the family of symmetric and δ-correct algorithms on the symmetric collection $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)$.

Finally, we define the optimal Bayesian budget for an algorithm in $\Pi_{Sym}(\delta, \mathcal{E}_{Sym}(\mathcal{C}^*, \gamma))$ as

$$T^*(\delta, \mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)) := \inf_{\pi \in \Pi_{Sym}} \mathbb{E}_{\pi,\mathcal{C}^*}[\mathcal{T}|\mathcal{Y}] \ ,$$

where the inf is taken over $\Pi_{Sym}(\delta, \mathcal{E}_{Sym}(\mathcal{C}^*, \gamma))$, recalling that $\mathbb{E}_{\pi,\mathcal{C}^*}$ is the integrated budget with respect to the prior γ.

The next lemma implies that we only need to lower bound the quantity $T^*_{Sym}(\delta, \mathcal{E}_{Sym}(\mathcal{C}^*, \gamma))$.

Lemma 3.B.7. *If K is even, K divides n, $\theta = 1/K$, and $n \geq 2K$, it holds that*

$$T^*(\delta, \mathcal{E}(\Delta, \theta, \sigma, n, K, d)) \geqslant T^*_{Sym}(\delta, \mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)) \ .$$

Remark 3.B.8. We highlight that this construction essentially reduces the problem into L sub-problems of active binary classification. On the family of environments $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)$, the offsets $C_1, \dots, C_L$ and the labels of the blocks $l^*_1, \dots, l^*_n$ are fixed and common to all the environments $\nu(\mathcal{C}^*, \bar{\mu})$ and $\nu(\mathcal{C}^*{}_{(a)}, \bar{\mu})$, it is equivalent to say that this is known by the learner. Then, the problem consists on estimating the partition into two groups $\mathcal{C}^*{}_l = \mathcal{C}^*{}_{l,1} \sqcup \mathcal{C}^*{}_{l,-1}$ (up to switching of the two groups) for any of the L blocks. If an algorithm is symmetric, it will have access in expectation to the same budget to solve each sub-problem.

The proof of this Lemma, technical but standard is provided in Section 3.B.4. In the proof, we explain how to use the knowledge of the blocks $\mathcal{C}^*_1, \dots, \mathcal{C}^*{}_L$ and the offsets $C(1), \dots, C(l)$ in order to transform any algorithm into a symmetric algorithm — see Definition 3.B.6. The rough idea is to permute the arms, and then to apply the algorithm to the permuted arms.

3.B.2 First Lower bound: proof of Lemma 3.B.1

In this section, we prove the Lower Bound from Lemma 3.B.1. We highlight that the lower bound from Lemma 3.B.1 does not depend on the dimension d. Thus, we will derive lower bound for fixed centers of the groups which basically amounts to the simpler setting where the learner knows them in advance.

We use the construction of Section 3.B.1, and we choose the prior distribution $\gamma_1 := \delta_\mu$ to be a Dirac, i.e, $\bar{\mu}(l) = \mu$ for all l and the centers are deterministic and fixed. We choose $\mu = (\Delta/2, 0, \dots, 0) \in \mathbb{R}^d$ and $\beta = 2$. The environment $\nu(\mathcal{C}^*, \bar{\mu})$ is in $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$, so the event $\mathcal{Y}$ from Definition 3.B.6 holds almost-surely.

Remark 3.B.9. The clustering with bandit feedback problem on $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)$ is highly connected to a specific instance of the Thresholding Bandit Problem (TBP), another pure exploration problem studied in (Cheshire et al., 2020). In this problem, a player interacts with a multi-armed bandit environment with one-dimensional rewards, and she has to recover the set of arms with a mean larger or equal to a certain threshold (for us, this threshold is Δ). The proof of Lemma 3.B.1 is inspired by the proof of Theorem 1 in (Cheshire et al., 2020). For the thresholding bandit problem, the authors derive a lower bound in the fixed budget setting. In this setting, the player has to minimize the simple regret (related to the probability of error), using a fixed budget. From their result, we could deduce a lower bound of the form $\frac{\sigma^2}{\Delta^2}(n-K)\log(\frac{n-K}{\delta})$. Here, we use a workaround to establish a slightly tighter lower bound of the form $\frac{\sigma^2}{\Delta^2} n \log(\frac{n}{\delta})$.

We consider $T^*_{Sym}(\delta, \mathcal{E}_{Sym}(\mathcal{C}^*, \gamma_1)) = \inf_{\pi \in \Pi_{Sym}} \mathbb{E}_{\pi, \mathcal{C}^*}[\mathcal{T}]$ where the inf is taken over all symmetric and δ-correct algorithm on $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma_1)$ —see Definition 3.B.6.

Lemma 3.B.10. *If K is even, K divides n and $n \geqslant K$, then,*

$$T^*_{Sym}(\delta, \mathcal{E}_{Sym}(\mathcal{C}^*, \gamma_1)) \geqslant n\frac{\sigma^2}{\Delta^2}\,\mathrm{kl}\left(1-\delta, \frac{2\delta}{n}\right).$$

Then, Lemma 3.B.1 simply follows from the reduction arguments of Lemma 3.B.7 and 3.B.10 that we prove now.

Proof of Lemma 3.B.10. Let π be a symmetric and δ-correct algorithm for the clustering with bandit feedback problem on $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma_1)$. It outputs a partition $\hat{\mathcal{C}}$ of $[n]$ such that for any $a \in {\mathcal{C}^*}_+$,

$$\mathbb{P}_{\pi, {\mathcal{C}^*}_{(a)}}(\hat{\mathcal{C}} \sim {\mathcal{C}^*}_{(a)}) \geqslant 1-\delta \ , \text{ and}$$

$$\mathbb{P}_{\pi, \mathcal{C}^*}(\hat{\mathcal{C}} \sim \mathcal{C}^*) \geqslant 1-\delta \ .$$

The main tool that we use is a data-processing inequality— see e.g.,(Gerchinovitz et al., 2020). We will use the KL-divergence which, in our setting, turns out to be explicitly computed. The difficulty of the proof is to recover the term $\log(n/\delta)$ in the lower bound of the budget. For that, we adapt the proof page 15 of (Cheshire et al., 2020) to the fixed confidence setting. The idea is that, instead of constructing one partition, different from $\mathcal{C}^*$, we constructed a collection of $\{{\mathcal{C}^*}_{(a)}\}_{a \in C^*_+}$, where any algorithm has to distinguish $\mathcal{C}^*$ from any of these environments (up to relabelling).

First, we use lemma 1 from (Kaufmann et al., 2016) which relies on the data-processing inequality and the decomposition of the KL-divergence in the multi-armed bandit model. It holds that, for any $a \in {\mathcal{C}^*}_+$,

$$\begin{aligned}\mathrm{kl}\left(\mathbb{P}_{\pi, {\mathcal{C}^*}_{(a)}}(\hat{\mathcal{C}} \sim {\mathcal{C}^*}_{(a)}), \mathbb{P}_{\pi, \mathcal{C}^*}(\hat{\mathcal{C}} \sim {\mathcal{C}^*}_{(a)})\right) &\leqslant \mathrm{KL}\left(\mathbb{P}_{\pi, {\mathcal{C}^*}_{(a)}}, \mathbb{P}_{\pi, \mathcal{C}^*}\right) \\ &= \mathbb{E}_{\pi, {\mathcal{C}^*}_{(a)}}[N_a + N_{b_{l^*(a)}}]\frac{\Delta^2}{2\sigma^2} \ ,\end{aligned} \tag{3.19}$$

the last equality follows from the fact that the environments $\nu(\mathcal{C}^*, \bar{\mu})$ and $\nu({\mathcal{C}^*}_{(a)}, \bar{\mu})$ only differ on arm a and b_a and $\mathrm{KL}(\mathcal{N}(-\Delta/2, \sigma^2), \mathcal{N}(\Delta/2, \sigma^2)) = \Delta^2/2\sigma^2$. We recall that for any $b \in [n]$, N_b is the number of times that the arm b is sampled.

Thanks to the joint convexity of the kl function (see Gerchinovitz et al., 2020, Corollary 3), we have

$$\begin{aligned} &\mathrm{kl}\left(\frac{1}{n/2}\sum_{a\in{\mathcal{C}^*}_+}\mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(\hat{\mathcal{C}}\sim{\mathcal{C}^*}_{(a)}), \frac{1}{n/2}\sum_{a\in{\mathcal{C}^*}_+}\mathbb{P}_{\pi,\mathcal{C}^*}(\hat{\mathcal{C}}\sim{\mathcal{C}^*}_{(a)})\right) \\ &\leqslant \frac{1}{n/2}\sum_{a\in{\mathcal{C}^*}_+}\mathrm{kl}\left(\mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(\hat{\mathcal{C}}\sim{\mathcal{C}^*}_{(a)}), \mathbb{P}_{\pi,\mathcal{C}^*}(\hat{\mathcal{C}}\sim{\mathcal{C}^*}_{(a)})\right) \ . \end{aligned} \tag{3.20}$$

By construction, the partition $\mathcal{C}^*$ and all the different partitions $({\mathcal{C}^*}_a)_{a\in{\mathcal{C}^*}_+}$ belong to different equivalence classes with respect to the relation $\sim$. As π is δ-correct — see Definition 3.B.6, we deduce that

$$\begin{aligned} &\forall a \in {\mathcal{C}^*}_+,\ \mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(\hat{\mathcal{C}}\sim{\mathcal{C}^*}_{(a)}) \geqslant 1-\delta \ ; \\ &\sum_{a\in{\mathcal{C}^*}_+}\mathbb{P}_{\pi,\mathcal{C}^*}(\hat{\mathcal{C}}\sim{\mathcal{C}^*}_{(a)}) = \mathbb{P}_{\pi,\mathcal{C}^*}(\sqcup_{a\in{\mathcal{C}^*}_+}\{\hat{\mathcal{C}}\sim{\mathcal{C}^*}_{(a)}\}) \leqslant \mathbb{P}_{\pi,\mathcal{C}^*}(\hat{\mathcal{C}}\not\sim\mathcal{C}^*) \leqslant \delta \ . \end{aligned}$$

With the monotony properties of the kl function, we obtain

$$\mathrm{kl}\left(1-\delta, \frac{\delta}{n/2}\right) \leqslant \mathrm{kl}\left(\frac{1}{n/2}\sum_{a\in{\mathcal{C}^*}_+}\mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(\hat{\mathcal{C}}\sim{\mathcal{C}^*}_{(a)}), \frac{1}{n/2}\sum_{a\in{\mathcal{C}^*}_+}\mathbb{P}_{\pi,\mathcal{C}^*}(\hat{\mathcal{C}}\sim{\mathcal{C}^*}_{(a)})\right) \ . \tag{3.21}$$

Gathering Equations (3.19), (3.20) and (3.21), we obtain

$$\mathrm{kl}\left(1-\delta, \frac{2\delta}{n}\right) \leqslant \frac{1}{n/2}\sum_{a\in{\mathcal{C}^*}_+}\mathbb{E}_{\pi,{\mathcal{C}^*}_{(a)}}[N_a + N_{b_a}]\frac{\Delta^2}{2\sigma^2} \tag{3.22}$$

We recall that π is symmetric. Hence, For any $a \in {\mathcal{C}^*}_+$, we have

$$\mathbb{E}_{\pi,{\mathcal{C}^*}_{(a)}}[N_a + N_{b_a}] = \frac{2}{n}\mathbb{E}_{\pi,{\mathcal{C}^*}_{(a)}}[\mathcal{T}] = \frac{2}{n}\mathbb{E}_{\pi,\mathcal{C}^*}[\mathcal{T}] \ .$$

Finally, with Equation (3.22), we conclude that

$$\mathbb{E}_{\pi,\mathcal{C}^*}[\mathcal{T}] \geqslant \frac{\sigma^2}{\Delta^2} n\,\mathrm{kl}\left(1-\delta, \frac{2\delta}{n}\right) \ . \tag{3.23}$$

We take now the inf over all algorithms π, which are δ-correct and symmetric, this proves Lemma 3.B.10. □

3.B.3 Second Lower Bound: proof of Lemma 3.B.2

In this section, we prove the lower bound from Lemma 3.B.2. If $d \leqslant (8/3)^2 \log(K/\delta)$, the lower bound from Lemma 3.B.2 is smaller than the dimension-free lower bound from Lemma 3.B.1, which is already proved. We may then assume that $d \geqslant (8/3)^2 \log(K/\delta)$. For the sake of the presentation, we postpone the proofs of some technical lemmas to the end of the next subsection.

Step 1: introduction of the Gaussian prior. In this regime, we choose the prior distribution γ to be Gaussian. Indeed, we introduce $\gamma_2 = \mathcal{N}(0, \rho^2 I_d)$ with $\rho^2 = \frac{\Delta^2}{d}$ and $\bar{\mu}(1), \dots, \bar{\mu}(L)$ are i.i.d of law $\mathcal{N}(0, \rho^2 I_d)$. Also, we choose $\beta = 4$. We consider the Bayesian family of environments constructed in Section 3.B.1 $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma_2)$.

Because of this Bayesian prior, we have some additional technical challenge in comparison to the low-dimensional case.

1. We can not use the decomposition of the KL-divergence for bandit in order to compute $\mathrm{KL}(\mathbb{P}_{\pi,\mathcal{C}^*{}_{(a)}}, \mathbb{P}_{\pi,\mathcal{C}^*})$ because the integral over the prior γ_1 is inside the KL-divergence. Most of the work consists on upper bounding this divergence with a divergence that can be computed.
2. We can not compare the maximum budget over $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$ (i.e.,$\sup_{\nu \in \mathcal{E}(\Delta)} \mathbb{E}_{\pi,\nu}[\mathcal{T}]$) to the Bayesian budget $\mathbb{E}_{\pi,\mathcal{C}^*}[\mathcal{T}]$ because the minimal gap of $\nu(\mathcal{C}^*, \bar{\mu})$ is not always larger than Δ. This is why we condition on the event $\mathcal{Y} = \bigcap_{l \in [L]} \{\Delta/2 \leqslant \|\bar{\mu}(l)\| \leqslant \Delta(\beta - 1)/2\} \subset \{\nu(\mathcal{C}^*, \bar{\mu}) \in \mathcal{E}(\Delta, \theta, \sigma, n, K, d)\}$.

We compute $\mathbb{P}_{\gamma^{\otimes L}}(\mathcal{Y}^c)$ for the Gaussian prior. This is the only time we will use the hypothesis $d \geqslant (8/3)^2 \log(K/\delta)$.

Lemma 3.B.11. *If we assume that $d \geqslant (8/3)^2 \log(K/\delta)$ and $\gamma_2 = \mathcal{N}(0, \rho^2)$, we have*

$$\mathbb{P}_{\gamma_2^{\otimes L}}(\mathcal{Y}) = \mathbb{P}_{\gamma_2^{\otimes L}}\left(\bigcap_{l \in [L]} \{\Delta/2 \leqslant \|\bar{\mu}(l)\| \leqslant 3\Delta/4\}\right) \geqslant 1 - \delta \ .$$

Step 2: From active binary classification to (batch) two-sample testing. Let $\pi \in \Pi_{Sym}$ be a δ-correct and symmetric algorithm for the clustering with bandit feedback problem on $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma_2)$ —see Definition 3.B.6. We define $t = 6\mathbb{E}_{\pi,\mathcal{C}^*}[\mathcal{T}|\mathcal{Y}]/n$ and $T = 6\mathbb{E}_{\pi,\mathcal{C}^*}[\mathcal{T}|\mathcal{Y}]/K$.

Recall that, for any $a \in \mathcal{C}^*{}_+ = \{a; g_a^* = 1\}$, $\mathcal{C}^*{}_{(a)}$ is obtained by switching one arm a with another arm $b_a \in \mathcal{C}^*{}_{l_a^*,-1}$. Also, $N_a := \sum_{t=1}^{\mathcal{T}} \mathbb{1}_{\{A_s = a\}}$ is the number of times the arm a is sampled. We also denote, $M_l = \sum_{b : l_b^* = l} N_b$ as the number of times the arms in the block $\mathcal{C}^*{}_l$ are sampled. As π is symmetric and as the blocks have the same size $2n/K$, there exists t and T two integers, such that for any $a \in \mathcal{C}^*{}_+$,

$$t = 3\mathbb{E}_{\pi,\mathcal{C}^*{}_{(a)}}[N_a + N_{b_a}|\mathcal{Y}] \ , \quad \text{and} \qquad T := 3\mathbb{E}_{\pi,\mathcal{C}^*{}_{(a)}}[M_{l_a^*}|\mathcal{Y}] \ .$$

Remark 3.B.12. We now give some heuristic in order to explain the rest of the proof. Imagine that, at time $\mathcal{T}$, the learner receives an oracle that gives the labels of all the arms except the

arm a, assume also that the learner knows that $a \in \mathcal{C}^*{}_l$. As in a supervised classification setting, the player has to find the label g_a of the unlabeled data sampled from a, using the labelled data available. It has access to N_a observations from a distributed as $\mathcal{N}(g_a\bar{\mu}(l), \sigma^2 I_d)$, and $M_{l_a^*} - N_a$ labelled data distributed as $\mathcal{N}(\bar{\mu}(l), \sigma^2 I_d)$. It also has access to data from the other blocks, but those data are not useful to find g_a. Moreover, N_a is of the order of $\mathbb{E}_{\pi,\mathcal{C}^*}[\mathcal{T}]/n$ and $M_{l_a^*} - N_a$ is of the order of $\mathbb{E}_{\pi,\mathcal{C}^*}[\mathcal{T}]/L$. As a consequence, with this amount of data, a learner should be able to correctly recover the labels in this simplified setting.

With this heuristic in mind, we introduce the following (batch) two-sample testing problem.

Definition 3.B.13. Let t, T be two integers, we consider data $Y_1, \dots, Y_t, Z_1, \dots, Z_T$ and two symmetric hypotheses $\mathcal{H}_1$ and $\mathcal{H}_{-1}$ such that, for $g \in \{-1, 1\}$, under $\mathcal{H}_g$, the data follows the law $\mathbb{P}_g$ defined as follows:

— $\mu \sim \gamma$ and conditionally on μ:

— $Y_1, \dots, Y_t, Z_1, \dots, Z_T$ are independent;

— $\forall r \in [t]$, $Y_r \sim \mathcal{N}(g\mu, \sigma^2 I_d)$

— $\forall s \in [T]$, $Z_r \sim \mathcal{N}(\mu, \sigma^2 I_d)$.

This problem is interesting because we can explicitly compute the KL-divergence.

Lemma 3.B.14. *Let $g \in \{-1, 1\}$ and $\mathbb{P}_g$ defined in Definition 3.B.13. It holds that*

$$\mathrm{KL}(\mathbb{P}_{-g}, \mathbb{P}_g) = \mathrm{KL}(\mathbb{P}_g, \mathbb{P}_{-g}) = \frac{2tT\rho^4 d}{\sigma^4 + \sigma^2\rho^2(t+T)} \leqslant \frac{2tT\rho^4 d}{\sigma^4} \wedge \frac{2\rho^2 d}{\sigma^2}\frac{tT}{t+T} \ .$$

Now, we explain properly the ideas introduced in the previous remark. We define the event $B_a = \{N_a + N_{b_a} \leqslant t\} \cap \{M_{l_a^*} \leqslant T\}$. Thanks to Markov inequality, the event B_a has a probability higher than a constant and conditionally on B_a, the algorithm π has access to strictly less information than in (batch) two-sample testing problem defined above with budget (t, T). We formalize this in the following coupling lemma.

Lemma 3.B.15. *Let $a \in \mathcal{C}^*{}_+$ be an arm and fix A_a an event. Consider the family of random variables $(Y_1, \dots, Y_t)$, $(Z_1, \dots, Z_T)$ that follows a distribution $\mathbb{P}_{-1}$ — see Definition 3.B.13. Consider also an independent sequence $(\epsilon_s, U_s)_{s\geqslant 1}$ of random variables such that for all $s \geqslant 1$, $\epsilon_s \sim \mathcal{N}(0, I_d)$ and $U_s \sim \mathcal{U}([0,1])$. Then, there exists a function f_a that is measurable according to the random variables Y, Z, ϵ, U and such that $A_a \cap B_a = f_a(Y, Z, \epsilon, U)$, where the equality holds with respect to the probability distribution $\mathbb{P}_{\pi,\mathcal{C}^*{}_{(a)}} = \int_{\bar{\mu}} \mathbb{P}_{\pi,\mathcal{C}^*{}_{(a)},\bar{\mu}} \, \mathrm{d}\gamma^{\otimes L}(\bar{\mu})$.*

Similarly, if $(Y, Z) \sim \mathbb{P}_1$, with the same function f_a, $A_a \cap B_a = f_a(Y, Z, \epsilon, U)$, under the probability distribution $\mathbb{P}_{\pi,\mathcal{C}^*}$.

In the previous lemma, we will consider $A_a := \{\hat{\mathcal{C}} \sim \mathcal{C}^*{}_{(a)}\}$ for $a \in \mathcal{C}^*{}_+$. By construction of $\mathcal{C}^*{}_{(a)}$ (because $n \geqslant 2K$), the events A_a are disjoint. By using the fact that π is δ-correct on $\mathcal{E}(\gamma, \Delta)$, we have the following property for A_a,

Lemma 3.B.16. *The family $(A_a \cap B_a)_{a\in\mathcal{C}^*{}_+}$ is such that*

1. $\sum_{a\in\mathcal{C}^*{}_+}\mathbb{P}_{\pi,\mathcal{C}^*}(A_a\cap B_a)\leqslant\delta+\mathbb{P}_{\gamma^{\otimes L}}(\mathcal{Y}^c)\leqslant 2\delta$;
2. $\mathbb{P}_{\pi,\mathcal{C}^*{}_{(a)}}(A_a\cap B_a)\geqslant 1/3-\delta-\mathbb{P}_{\gamma^{\otimes L}}(\mathcal{Y}^c)\geqslant 1/3-2\delta$.

We delay the technical proofs of Lemma 3.B.15 and Lemma 3.B.16. From there, we have all the tools that we need. We now use data-processing inequalities similar to the proof of Lemma 3.B.1 to conclude.

Step 3: Conclusion to the proof of Lemma 3.B.2. We assume that $\delta\in(0,1/6)$, so that $\mathrm{kl}\left(1/3-2\delta,\frac{2\delta}{n/2}\right)$ is defined.

We use the first point of Lemma 3.B.16. Notice that the events $(A_a)_a$ are disjoint by construction of $\mathcal{C}^*{}_{(a)}$ and because we took at least two arms by groups ($n\geqslant 2K$), it holds that

$$\sum_{a\in\mathcal{C}^*{}_+}\mathbb{P}_{\pi,\mathcal{C}^*}(A_a\cap B_a)=\mathbb{P}_{\pi,\mathcal{C}^*}(\sqcup_{a\in\mathcal{C}^*{}_+}A_a\cap B_a)\leqslant 2\delta\ .$$

With the second point of Lemma 3.B.16, for any $a\in\mathcal{C}^*{}_+$, we have

$$\mathbb{P}_{\mathcal{C}^*{}_{(a)}}(A_a\cap B_a)\geqslant 1/3-2\delta\geqslant 0\ .$$

We use the monotony properties of the kl function, it holds that

$$\mathrm{kl}\left(1/3-2\delta,\frac{2\delta}{n/2}\right)\leqslant\mathrm{kl}\left(\frac{1}{n/2}\sum_{a\in\mathcal{C}^*{}_+}\mathbb{P}_{\pi,\mathcal{C}^*{}_{(a)}}(A_a\cap B_a),\frac{1}{n/2}\sum_{a\in\mathcal{C}^*{}_+}\mathbb{P}_{\mathcal{C}^*}(A_a\cap B_a)\right)\ .$$

Thanks to the joint convexity of the kl function (see Gerchinovitz et al., 2020, corollary 3), we deduce that

$$\begin{aligned}&\mathrm{kl}\left(\frac{1}{n/2}\sum_{a\in\mathcal{C}^*{}_+}\mathbb{P}_{\pi,\mathcal{C}^*{}_{(a)}}(A_a\cap B_a),\frac{1}{n/2}\sum_{a\in\mathcal{C}^*{}_+}\mathbb{P}_{\pi,\mathcal{C}^*}(A_a\cap B_a)\right)\\ \leqslant&\frac{1}{n/2}\sum_{a\in\mathcal{C}^*{}_+}\mathrm{kl}\left(\mathbb{P}_{\pi,\mathcal{C}^*{}_{(a)}}(A_a\cap B_a),\mathbb{P}_{\pi,\mathcal{C}^*}(A_a\cap B_a)\right)\ .\end{aligned}$$

Now, we use the coupling Lemma 3.B.15,

$$\begin{aligned}\mathbb{P}_{\pi,\mathcal{C}^*{}_{(a)}}(A_a\cap B_a)&=\mathbb{P}_1\times\mathbb{P}_{\epsilon,U}(f_a(Y,Z,\epsilon,U))\\ \mathbb{P}_{\pi,\mathcal{C}^*}(A_a\cap B_a)&=\mathbb{P}_{-1}\times\mathbb{P}_{\epsilon,U}(f_a(Y,Z,\epsilon,U))\ .\end{aligned}$$

We use the data-processing inequality (see Gerchinovitz et al., 2020, corollary 2), for all $a\in\mathcal{C}^*{}_+$,

$$\begin{aligned}\mathrm{kl}\left(\mathbb{P}_{\pi,\mathcal{C}^*{}_{(a)}}(A_a\cap B_a),\mathbb{P}_{\pi,\mathcal{C}^*}(A_a\cap B_a)\right)&=\mathrm{kl}\left(\mathbb{P}_1\times\mathbb{P}_{\epsilon,U}(f_a(Y,Z,\epsilon,U)),\mathbb{P}_{-1}\times\mathbb{P}_{\epsilon,U}(f_a(Y,Z,\epsilon,U))\right)\\ &\leqslant\mathrm{KL}\left(\mathbb{P}_{-1}\otimes\mathbb{P}_{\epsilon,U},\mathbb{P}_1\otimes\mathbb{P}_{\epsilon,U}\right)\\ &=\mathrm{KL}\left(\mathbb{P}_{-1},\mathbb{P}_1\right)\ .\end{aligned}$$

Gathering the previous inequalities, we obtain

$$\mathrm{kl}\left(\frac{1}{3}-2\delta,\frac{4\delta}{n}\right)\leqslant\frac{1}{n/2}\sum_{a\in\mathcal{C}^*{}_{(a)}}\mathrm{KL}(\mathbb{P}_{-1},\mathbb{P}_1)\ .$$

We recall that $\rho^2=\Delta^2/d$. With the explicit computation from Lemma 3.B.14, we have

$$\frac{d\sigma^4}{2\Delta^4}\,\mathrm{kl}\left(\frac{1}{3}-2\delta,\frac{4\delta}{n}\right)\leqslant\frac{1}{n/2}\sum_{a\in\mathcal{C}^*{}_+}tT=tT\ .$$

Finally, we have, using the definition of t and T,

$$\mathbb{E}_{\pi,\mathcal{C}^*}[\mathcal{T}|\mathcal{Y}]^2\geqslant\frac{d\sigma^4nK}{72\Delta^4}\,\mathrm{kl}\left(\frac{1}{3}-2\delta,\frac{4\delta}{n}\right)\ .$$

As it is true for any $\pi\in\Pi_{\mathcal{O}}$, take the inf in the last inequality over $\pi\in\Pi_{\mathcal{O}}$ and use Lemma 3.B.7 to get

$$T^*(\delta,\mathcal{E}(\Delta,\theta,\sigma,n,K,d))\geqslant\frac{\sigma^2}{\Delta^2}\sqrt{\frac{dnK}{72}\,\mathrm{kl}\left(\frac{1}{3}-2\delta,\frac{4\delta}{n}\right)}\ ,$$

this is exactly the inequality of Lemma 3.B.2.

3.B.4 Proof of technical lemmas

Proof of Lemma 3.B.4. Let n,K such that $n\geqslant 2K$. Let $\theta>0$ such that the collection $\mathcal{E}(\Delta,\theta,\sigma,n,K,d)$ is non-empty. We prove Lemma 3.B.4 assuming that K is odd, the other case is simpler and can be proved with the same construction up to minor details.

Recall the expressions introduced before Lemma 3.B.4, $K'=K-1$, $n'=K'\left\lfloor\frac{n-\lceil\theta n\rceil}{K'}\right\rfloor$ and $\theta'=1/K'$.

Let π being δ-correct on $\mathcal{E}(\Delta,\theta,\sigma,n,K,d)$, we will use π to construct π', an algorithm which is δ-correct on $\mathcal{E}(\Delta,\theta',\sigma,n',K',d)$.

Let $\nu'\in\mathcal{E}(\Delta,\theta',\sigma,n',K',d)$ be an environment with $K-1$ perfectly balanced groups. We run the algorithm π where we create the data $X_1,\dots,X_{\mathcal{T}_\pi}$ with the following coupling.

— If $A_t^\pi\in[n']$, we sample X_t with the arm A_t^π from ν'.

— If $A_t^\pi\in[n'+1;N-\lceil\theta n\rceil]$, we sample X_t with a_1, the first arm from ν'.

— If $A_t^\pi\in[n-\lceil\theta n\rceil+1,n]$, we create $X_t=c$ where c is an arbitrary large constant.

Equivalently, we have created the environment ν where the n' first arms are the arms of ν; the $\lceil\theta n\rceil$ last arms are in an artificial group associated to a Dirac in c, and the remaining arms are in the same group as a_1. The environment ν has a hidden partition $\mathcal{C}_1^*,\dots,\mathcal{C}_K^*$ where $\mathcal{C}_1^*=C'_1\cup[n'+1;N-\lceil\theta n\rceil]$, $\mathcal{C}^*{}_2,\dots,\mathcal{C}^*{}_{K-1}=C'_2,\dots,C'_{K-1}$, and $\mathcal{C}_K^*=[n-\lceil\theta n\rceil+1,n]$. By construction, this environment is in $\mathcal{E}(\Delta,\theta,\sigma,n,K,d)$. In particular, the balancedness is larger than θ, and the minimal gap is larger than Δ if c is large enough.

When π reaches $\mathcal{T}_\pi$, it outputs a partition of $[n]$, $\hat{\mathcal{C}}_1^\pi, \dots, \hat{\mathcal{C}}_K^\pi$, and we output $\hat{\mathcal{C}}^{\pi'}$ as the partition defined by the restriction to $[n']$ of the partition $\hat{\mathcal{C}}^\pi$. This is what we call the algorithm π'.

As π is δ-correct on $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$, it holds that, with a probability $\mathbb{P}_{\pi,\nu}$ higher than $1-\delta$, $\hat{\mathcal{C}} \sim \mathcal{C}^*$, and this implies that $\hat{\mathcal{C}}^{\pi'} \sim C'$. Finally, we have $\mathbb{P}_{\pi',\nu'}(\hat{\mathcal{C}}^{\pi'} \sim C') \geqslant \mathbb{P}_{\pi,\nu}(\hat{\mathcal{C}}^{\pi} \sim \mathcal{C}^*) \geqslant 1-\delta$. This means that π' is indeed δ-correct on $\mathcal{E}(\Delta, \theta', \sigma, n', K', d)$.

In terms of budget, we have $\mathcal{T}_{\pi'} \leqslant \mathcal{T}_\pi$, because the data provided from the last group are artificially created by the algorithm. We deduce that

$$\mathbb{E}_{\pi',\nu'}[\mathcal{T}_{\pi'}] \leqslant \mathbb{E}_{\pi,\nu}[\mathcal{T}_\pi] \leqslant \sup_{\nu \in \mathcal{E}(\Delta,\theta,\sigma,n,K,d)} \mathbb{E}_{\pi,\nu}[\mathcal{T}] \ .$$

Then, we take the sup over $\nu' \in \mathcal{E}(\Delta, \theta', \sigma, n', K', d)$, and we have

$$T^*(\delta, \mathcal{E}(\Delta, \theta', \sigma, n', K', d)) \leqslant \sup_{\nu' \in \mathcal{E}(\Delta,\theta',\sigma,n',K',d)} \mathbb{E}_{\pi',\nu'}[\mathcal{T}] \leqslant \sup_{\nu \in \mathcal{E}(\Delta,\theta,\sigma,n,K,d)} \mathbb{E}_{\pi,\nu}[\mathcal{T}] \ .$$

Finally, we consider the inf over π δ-correct on $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$, which concludes the proof of Lemma 3.B.4.

Proof of Lemma 3.B.7. Let π' be a δ-correct algorithm on $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$.

We will use the algorithm π' to construct an algorithm π, which is symmetric and δ-correct on the class $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)$ — see Definition 3.B.6. We will use the symmetries in the structure of the environment $\nu(C', \bar{\mu})$ when $\bar{\mu}$ is distributed with the prior $\gamma^{\otimes L}$ as the main argument to prove that π will have the wanted properties. To avoid confusion, we index $(A_s^{\pi'})_s$, $\mathcal{T}_{\pi'}$ and $\hat{\mathcal{C}}^{\pi'}$ for the algorithm π' and without $'$ for the algorithm π. As explained in the previous remark, the algorithm π just need to perform well (i.e., being δ-correct) on the family $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)$, so we can use the offsets and the labels $l_1^*, \dots, l_n^*$ to construct the algorithm π.

Construction of π. In this paragraph, we describe how we symmetrize a strategy π' —see Algorithm 7. Let $C' \in \{\mathcal{C}^*\} \cup \{\mathcal{C}^*{}_{(a)}\}_{a \in \mathcal{C}^*{}_+}$ being a partition. In order to make the reading easier, we use the notation $l_a^* = l^*(a)$ for all $a \in [n]$. For any arm a, we denote as $g'(a) \in \{-1, 1\}$ as the label such that the mean of a is $\mu_a = g'(a)\bar{\mu}(l^*(a)) + C(l^*(a))$, in the environment $\nu(C', \bar{\mu})$, for any $\bar{\mu} \in \mathbb{R}^d$.

We need to define the behavior of π when facing the environment $\nu(C', \bar{\mu})$ for any $\bar{\mu}$.

Define $\mathcal{S}$ as the set of permutations of $[n]$ that switch the blocks in $\mathcal{C}^*$, that is to say if $\kappa \in \mathcal{S}$ then for all $l \in [L]$, $\exists l' \in [L]$, such that $\kappa(\mathcal{C}^*{}_l) = \mathcal{C}^*{}_{l'}$. For any $\kappa \in \mathcal{S}$, κ naturally induces a permutation of $[L]$ denoted as $\tilde{\kappa}$ such that for all $a \in [n]$, $l^*(\kappa(a)) = \tilde{\kappa}(l^*(a))$.

First, the strategy π uniformly samples a permutation κ in $\mathcal{S}$ and a vector $\chi \in \{-1, 1\}^L$. From a rough perspective, the strategy π will then apply the strategy π' by permuting the blocks using κ and reversing the means of each block using χ.

Algorithm 7: Symmetrization of π'.

Input: $\nu(C', \bar{\mu})$ an environment in $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)$

Result: $\hat{\mathcal{C}}^{\pi}$, partition of $[n]$

$t = 1$

Take $\kappa \sim \mathcal{U}(\mathcal{S})$

Take $\chi \sim \mathcal{U}(\{-1,1\}^L)$

while $t \leqslant \mathcal{T}_{\pi'}(A_1^{\pi'}, X_1^{\pi'}, \dots, A_{t-1}^{\pi'}, X_{t-1}^{\pi'})$ **do**

  Choose an arm with π' and get $A_t^{\pi'}(A_1^{\pi'}, X_1^{\pi'}, \dots, A_{t-1}^{\pi'}, X_{t-1}^{\pi'}) \in [n]$.

  Sample X_t^{π} from $A_t^{\pi} := \kappa(A_t^{\pi'})$

  Create the data $X_t^{\pi'} := \chi(\tilde{\kappa}(l^*(A_t^{\pi'})))[X_t^{\pi} - C(\tilde{\kappa}(l^*(A_t^{\pi'})))] + C(l^*(A_t^{\pi'}))$

  t=t+1

end

Compute $\hat{\mathcal{C}}^{\pi'}(A_1^{\pi'}, X_1^{\pi'}, \dots, A_{\mathcal{T}}^{\pi'}, X_{\mathcal{T}}^{\pi'}) := \hat{\mathcal{C}}_1^{\pi'}, \dots, \hat{\mathcal{C}}_K^{\pi'}$

return $\hat{\mathcal{C}}_1^{\pi}, \dots, \hat{\mathcal{C}}_K^{\pi} := \kappa(\hat{\mathcal{C}}_1^{\pi'}), \dots, \kappa(\hat{\mathcal{C}}_K^{\pi'})$

Within the procedure π, we run algorithm π' with modified data $X_1^{\pi'}, \dots, X_{\mathcal{T}}^{\pi'}$. At time t, the algorithm π' chooses to sample the arm $A_t^{\pi'}$, where the decision is based on the data $(X_s^{\pi'}, A_s^{\pi'})_{s \leqslant t-1}$. Instead of sampling the arm chosen by π', the algorithm π samples X_t^{π} from the arm $A_t^{\pi} := \kappa(A_t^{\pi'})$. Then, π' sends the data $X_t^{\pi'}$ to π', according to the formula

$$X_t^{\pi'} = \chi(\tilde{\kappa}(l^*(A_t^{\pi'})))[X_t^{\pi} - C(\tilde{\kappa}(l^*(A_t^{\pi'})))] + C(l^*(A_t^{\pi'})) \ ,$$

where we recall that $C(l)$ is the offset associated to block l. When π' decides to stop, π also stops; i.e., $\mathcal{T}_{\pi} = \mathcal{T}_{\pi'}$. Then, π' outputs a partition $\hat{\mathcal{C}}^{\pi'} = \hat{\mathcal{C}}_1^{\pi'}, \dots, \hat{\mathcal{C}}_K^{\pi'}$ based on the modified data, and π outputs $\hat{\mathcal{C}}_1^{\pi}, \dots, \hat{\mathcal{C}}_K^{\pi} := \kappa(\hat{\mathcal{C}}_1^{\pi'}), \dots, \kappa(\hat{\mathcal{C}}_K^{\pi'})$.

Lemma 3.B.17. *Take $\kappa \in \mathcal{S}$, and $\chi \in \{-1,1\}^L$. For all $l \in [L]$, define $\bar{\mu}_{\kappa}(l) := \bar{\mu}(\tilde{\kappa}(l))$. As $\bar{\mu}$ is sampled according to $\gamma^{\otimes L}$, then $\bar{\mu}_{\kappa}$ follows the same prior $\gamma^{\otimes L}$. Define $C'(\kappa, \chi)$ as a partition of $[n]$ into $2L$ groups such that for all $(l, g) \in [L] \times \{-1; 1\}$, then*

$$C'(\kappa,\chi)_{l,g} = \{a \in [n]; l^*(a) = l, \text{ and } g'(\kappa(a))\chi(\tilde{\kappa}(l^*(a))) = g\} = \kappa^{-1}\left(C'_{\tilde{\kappa}(l), g\chi(\tilde{\kappa}(l))}\right) \ .$$

Conditionally on $\kappa, \chi, \bar{\mu}$, the modified data $X_s^{\mathcal{T}_{\pi'}}$ are distributed according to the probability induced by the interaction between π' and the environment $\nu(C'(\kappa,\chi), \bar{\mu}_{\kappa})$, after integration on the prior γ, we have

$$\mathbb{P}_{\pi, C'}(\cdot|\mathcal{Y}, \kappa, \chi) = \mathbb{P}_{\pi', C'(\kappa,\chi)}(\cdot|\mathcal{Y}) \ .$$

Remark 3.B.18. It is very important to note that, as C' is a partition with $K = 2L$ groups of the same size, the partition $C'(\kappa, \chi)$ is also balanced.

Proof of Lemma 3.B.17. Let $\bar{\mu} \in (\mathbb{R}^d)^L$ be a realization of the prior $\gamma^{\otimes L}$.

When π' tries to sample the arm $a = A_t^{\pi'}$, we sample in fact $\kappa(a)$. Using the Gaussian assumption on the data, and the expression of the centers of the environment $\nu(C', \bar{\mu})$, it holds

that

$$X_t^{\pi} = g'(\kappa(a))\bar{\mu}(l^*(\kappa(a))) + C(l^*(\kappa(a))) + \epsilon_s = g'(\kappa(a))\bar{\mu}(\tilde{\kappa}(l^*(a))) + C(\tilde{\kappa}(l^*(a))) + \epsilon_t \ ,$$

where $\epsilon_t \sim \mathcal{N}(0, \sigma^2 I_d)$. We used also in the second equality that κ induces a permutation of the blocks, so that $l^*(\kappa(a)) = \tilde{\kappa}(l^*(a))$.

We now decompose $X_t^{\pi'}$, using the expression defined Line 7 of Algorithm 7. Assuming that $\tilde{\kappa}(l^*(a)) = m \in [L]$, we have

$$\begin{aligned} X_t^{\pi'} =& \chi(m)\big[X_t^{\pi} - C(m)\big] + C(l^*(a)) \\ =& \chi(m)\big[g'(\kappa(a))\bar{\mu}(m) + \epsilon_t\big] + C(l^*(a)) \ . \end{aligned}$$

We develop and reorganize the terms, and we use the expression $\bar{\mu}_\kappa(l) = \bar{\mu}(\tilde{\kappa}(l))$,

$$\begin{aligned} X_t^{\pi'} =& g'(\kappa(a))\chi(m)\bar{\mu}(m) + \chi(m)\epsilon_t + C(l^*(a)) \\ =& g'(\kappa(a))\chi(m)\bar{\mu}_\kappa(l^*(a)) + \chi(m)\epsilon_t + C(l^*(a)) \ . \end{aligned}$$

As ϵ_t is symmetric with respect to 0, then $\epsilon'_t := \chi(m)\epsilon_t$ is distributed as a normal distribution $\mathcal{N}(0, \sigma^2 I_d)$. Besides, the $(\epsilon'_t)_t$ are independent. The arm a appears to π' to have a mean $C(l^*(a)) + \tilde{g}\bar{\mu}_\kappa(l^*(a))$, where $\tilde{g} = g'(\kappa(a))\chi(\tilde{\kappa}(l^*(a))) \in \{-1, 1\}$. It appears then that the data received by π' are distributed as $\nu(C'(\kappa, \chi), \bar{\mu}_\kappa)$, where, for all $(g, l) \in [L] \times \{-1, 1\}$,

$$C'(\kappa, \chi)_{l,g} = \{a \in [n]; l^*(a) = l, \text{ and } g'(\kappa(a))\chi(\tilde{\kappa}(l)) = g\} \ ,$$

which proves the first part of the lemma.

The second expression for $C'(\kappa, \chi)_{l,g}$ is now obtained using the fact that κ permutes the blocks, so that $l^*(\kappa(a)) = \tilde{\kappa}(l)$ and also that $\chi(\tilde{\kappa}(l)) \in \{-1, 1\}$.

$$\begin{aligned} \{a \in [n]; l^*(a) = l, \text{ and } g'(\kappa(a))\chi(\tilde{\kappa}(l)) = g\} =& \{a \in [n]; l^*(\kappa(a)) = \tilde{\kappa}(l), \text{ and } g'(\kappa(a)) = g\chi(\tilde{\kappa}(l))\} \\ =& \kappa^{-1}\left(C'_{\tilde{\kappa}(l), g\chi(\tilde{\kappa}(l))}\right) \ . \end{aligned}$$

Finally, if $\bar{\mu} \sim \gamma^{\otimes L}$, by exchangeability of the law of $\gamma^{\otimes L}$, and as $\tilde{\kappa}$ is a permutation of $[L]$, the vector $(\bar{\mu}(\tilde{\kappa}(l)))_{l\in[L]}$ is distributed as $(\bar{\mu}(l))_{l\in[L]}$. We also highlight that the event $\mathcal{Y} = \bigcap_{l\in[L]}\{\Delta/2 \leqslant \|\bar{\mu}(l)\| \leqslant \Delta(\beta-1)/2\} = \bigcap_{l\in[L]}\{\Delta/2 \leqslant \|\bar{\mu}_\kappa(l)\| \leqslant \Delta(\beta-1)/2\}$ remains the same, so that we have the equality of the laws

$$\mathbb{P}_{\pi, C'}(\cdot|\mathcal{Y}, \kappa, \chi) = \mathbb{P}_{\pi', C'(\kappa,\chi)}(\cdot|\mathcal{Y}) \ .$$

□

Correction of π. We now deduce that π is δ-correct on $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)$ —see Definition 3.B.6.

By construction of the algorithm, and with the definition of $C'(\kappa, \chi)$ given in Lemma 3.B.17,

we have conditionally on κ, χ, and $\bar{\mu}$,

$$\mathbb{P}_{\pi,C',\bar{\mu}}(\hat{\mathcal{C}}^{\pi} \sim C' | \kappa, \chi) = \mathbb{P}_{\pi',C'(\kappa,\chi),\bar{\mu}_\kappa}\left(\hat{\mathcal{C}}^{\pi'} \sim C'(\kappa,\chi)\right) \ .$$

If $\bar{\mu} \in \mathcal{Y}$, then we have $\bar{\mu}_\kappa \in \mathcal{Y}$ and environment $\nu(C'(\kappa,\chi), \bar{\mu}_\kappa)$ is in $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$. We recall that π is δ-correct on $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$, we then have

$$\mathbb{P}_{\pi',C'(\kappa,\chi),\bar{\mu}_\kappa}\left(\hat{\mathcal{C}}^{\pi'} \sim C'(\kappa,\chi)\right) \mathbb{1}_{\mathcal{Y}} \geqslant (1-\delta)\mathbb{1}_{\mathcal{Y}} \ .$$

The conclusion then follow by integrating over the law of κ, χ, and $\bar{\mu}$ to obtain $\mathbb{P}_{\pi,C'}(\hat{\mathcal{C}}^{\pi} \sim C' | \mathcal{Y}) \geqslant 1-\delta$, and π is indeed δ-correct on $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)$.

Symmetry of π. We want to prove that π is symmetric as defined in Definition 3.B.6. Take $a_1, a_2 \in [n]^2$ two arms and assume that $a_1 \in {C'}_{l_1,g_1}$ and $a_2 \in {C'}_{l_2,g_2}$.

First, we recall that $A_t^\pi = \kappa(A_t^{\pi'})$ so that,

$$N_{a_1}^\pi = \sum_{s=1}^{\mathcal{T}} \mathbb{1}_{A_s^\pi = a_1} = \sum_{s=1}^{\mathcal{T}} \mathbb{1}_{\kappa(A_s^{\pi'}) = a_1} = N_{\kappa^{-1}(a_1)}^{\pi'} \ .$$

We now use the expression of the uniform laws that follows κ,χ and Lemma 3.B.17,

$$\begin{aligned}\mathbb{E}_{\pi,C'}[N_{a_1}^\pi | \mathcal{Y}] =& \frac{1}{2^L \# \mathcal{S}} \sum_{\kappa \in \mathcal{S}} \sum_{\chi \in \{-1,1\}^L} \mathbb{E}_{\pi,C'}[N_{a_1}^\pi | \mathcal{Y}, \kappa, \chi] \\ =& \frac{1}{2^L \# \mathcal{S}} \sum_{\kappa \in \mathcal{S}} \sum_{\chi \in \{-1,1\}^L} \mathbb{E}_{\pi',C'(\kappa,\chi)}[N_{\kappa^{-1}(a_1)}^{\pi'} | \mathcal{Y}] \ .\end{aligned}$$

We construct $\kappa' \in \mathcal{S}$ a permutation which switches the blocks of a_1 and a_2, while switching a_1 and a_2, take

$$\begin{aligned}&\forall \epsilon \in \{-1,1\}, \kappa'({C'}_{l_1,\epsilon g_1}) = {C'}_{l_2,\epsilon g_2} \ ; && \forall \epsilon \in \{-1,1\}, \kappa'({C'}_{l_2,\epsilon g_2}) = {C'}_{l_1,\epsilon g_1} \ ; \\ &\kappa'(a_1) = a_2, \kappa'(a_2) = a_1 \ , \text{ and} && \forall c \in [n], \text{ if } l^*(c) \notin \{l_1, l_2\}, \kappa'(c) = c \ .\end{aligned}$$

The permutation κ' exists because the groups of C' have exactly the same size.

We also define $\chi' \in \{-1,1\}^L$ with

$$\chi'(l) = \chi(l) \text{ if } l \notin \{l_1, l_2\} \ , \qquad \chi'(l_1) = (g_1 g_2)\chi(l_2) \ , \qquad \text{and } \chi'(l_2) = (g_2 g_1)\chi(l_1) \ .$$

Note that $\kappa' \in \mathcal{S}$. When we consider $\mathcal{S}$ is a group of permutation we see that $\kappa'\mathcal{S} = \mathcal{S}$. Moreover, as the law of $\chi(1), \ldots, \chi(L)$ is exchangeable and symmetric with respect to 0, χ' and χ follow the same distribution.

It implies that we can use a change of variable in the sum,

$$\begin{aligned}\mathbb{E}_{\pi,C'}[N^{\pi}_{a_1}|\mathcal{Y}] =& \frac{1}{2^L \#\mathcal{S}} \sum_{\kappa\in\mathcal{S}} \sum_{\chi\in\{-1,1\}^L} \mathbb{E}_{\pi',C'(\kappa,\chi)}[N^{\pi'}_{\kappa^{-1}(a_1)}|\mathcal{Y}] \\ =& \frac{1}{2^L \#\mathcal{S}} \sum_{\kappa\in\mathcal{S}} \sum_{\chi\in\{-1,1\}^L} \mathbb{E}_{\pi',C'(\kappa'\kappa,\chi')}[N^{\pi'}_{(\kappa'\kappa)^{-1}(a_1)}|\mathcal{Y}] \ .\end{aligned}$$

Now, for any $\kappa \in \mathcal{S}$, $(\kappa'\kappa)^{-1}(a_1) = \kappa^{-1}(\kappa')^{-1}(a_1) = \kappa^{-1}(a_2)$ because κ' exchanges a_1 and a_2.

Then, fix χ and κ and consider the partition $C'(\kappa'\kappa, \chi')$. We want to prove that, $C'(\kappa'\kappa, \chi') = C'(\kappa, \chi)$. By definition (Lemma 3.B.17), we have to prove that $\forall b \in [n]$,

$$g'(\kappa(b))\chi(\tilde{\kappa}(l^*(b))) = g'(\kappa'\kappa(b))\chi'(\tilde{\kappa'}\tilde{\kappa}(l^*(b))) \ , \tag{3.24}$$

We prove Equation (3.24).

Take $\epsilon \in \{-1, 1\}$ and $b \in \kappa^{-1}(C'_{l_1,\epsilon g_1})$, by construction, $\tilde{\kappa'}$ is the transposition $(l_1\ l_2)$, and we have

$$\chi'(\tilde{\kappa'}\tilde{\kappa}(l^*(b))) = \chi'(\tilde{\kappa'}(l_1)) = \chi'((l_1\ l_2)(l_1)) = \chi'(l_2) = (g_1 g_2)\chi(l_1) \ .$$

Besides, we have $\chi(\tilde{\kappa}(l^*(b))) = \chi(l_1)$. Moreover, $\kappa(b) \in C'_{l_1,\epsilon g_1}$ and then $\kappa'(\kappa(b)) \in C'_{l_2,\epsilon g_2}$, i.e., $g'(\kappa'\kappa(b)) = \epsilon g_2$.

The equality in Equation (3.24) therefore holds for all b in $\kappa^{-1}(C'_{l_1,\epsilon g_1})$,

$$g'(\kappa(b))\chi(\tilde{\kappa}(l^*(b))) = \epsilon g_2(g_1 g_2)\chi(l_1) = \epsilon g_1 \chi(l_1) = g'(\kappa(b))\chi(\tilde{\kappa}(l^*(b))) \ .$$

The labels l_1 and l_2 play the symmetric role, so we also have the equality of Equation (3.24) for $b \in \kappa^{-1}(C'_{l_2,\epsilon g_2})$. Finally, if $l^*(\kappa(b)) \notin \{l_1, l_2\}$, then by construction of χ' and κ', we have $\kappa'(\kappa(b)) = \kappa(b)$ and $\chi'(\tilde{\kappa'}\tilde{\kappa}(l^*(b))) = \chi'(\tilde{\kappa}(l^*(b))) = \chi(\tilde{\kappa}(l^*(b)))$.

Equation (3.24) being proved, we have finally,

$$\begin{aligned}\mathbb{E}_{\pi,C'}[N^{\pi}_{a_1}|\mathcal{Y}] =& \frac{1}{2^L \#\mathcal{S}} \sum_{\kappa\in\mathcal{S}} \sum_{\chi\in\{-1,1\}^L} \mathbb{E}_{\pi',C'(\kappa'\kappa,\chi')}[N^{\pi'}_{(\kappa'\kappa)^{-1}(a_1)}|\mathcal{Y}] \\ =& \frac{1}{2^L \#\mathcal{S}} \sum_{\kappa\in\mathcal{S}} \sum_{\chi\in\{-1,1\}^L} \mathbb{E}_{\pi',C'(\kappa,\chi)}[N^{\pi'}_{\kappa^{-1}(a_2)}|\mathcal{Y}] \\ =& \mathbb{E}_{\pi,C'}[N^{\pi}_{a_2}|\mathcal{Y}] \ .\end{aligned}$$

This proves that $\mathbb{E}_{\pi,C'}[N^{\pi}_{a_1}|\mathcal{Y}]$ is independent of a_1 and equal to $\mathbb{E}_{\pi,C'}[\mathcal{T}|\mathcal{Y}]/n$. Now, using the same method as above with $\kappa' = (a\ b_a)$, we also deduce that $\mathbb{E}_{\pi,\mathcal{C}^*{}_{(a)}}[\mathcal{T}|\mathcal{Y}] = \mathbb{E}_{\pi,\mathcal{C}^*}[\mathcal{T}|\mathcal{Y}]$ does not depend on a.

This proves that π is symmetric as defined in Definition 3.B.6.

We have proved that π is δ-correct and symmetric on $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma)$. It remains to conclude for the proof of the lemma.

Budget of π. By construction of the algorithm, we have

$$\mathbb{E}_{\pi,\mathcal{C}^*}[\mathcal{T}_\pi|\mathcal{Y},\kappa,\chi] = \mathbb{E}_{\pi',\mathcal{C}^*(\kappa,\chi)}[\mathcal{T}_{\pi'}|\mathcal{Y}] \leqslant \sup_{\nu\in\mathcal{E}(\Delta,\theta,\sigma,n,K,d)} \mathbb{E}_{\pi',\nu}[\mathcal{T}_{\pi'}] \ ,$$

since, on the event $\mathcal{Y}$, we have $\nu(\mathcal{C}^*(\kappa,\chi),\bar{\mu}) \in \mathcal{E}(\Delta,\theta,\sigma,n,K,d)$. We now use the fact that π is in $\Pi_{Sym}(\delta,\mathcal{E}_{Sym}(\mathcal{C}^*,\gamma))$, so that

$$T^*_{Sym}(\delta,\mathcal{E}_{Sym}(\mathcal{C}^*,\gamma)) \leqslant \mathbb{E}_{\pi,\mathcal{C}^*}[\mathcal{T}_\pi|\mathcal{Y}] \leqslant \sup_{\nu\in\mathcal{E}(\Delta,\theta,\sigma,n,K,d)} \mathbb{E}_{\pi',\nu}[\mathcal{T}_{\pi'}] \ .$$

Finally, we prove Lemma 3.B.7 by taking the inf over $\pi' \in \Pi(\delta,\mathcal{E}(\Delta,\theta,\sigma,n,K,d)$.

Proofs from Section 3.B.3

Proof of Lemma 3.B.11. Let $l \in [L]$ and define $Z = \|\bar{\mu}(l)\|^2/\rho^2$. We have $\bar{\mu}(l) \sim \mathcal{N}(0,\rho^2)$ with $\rho^2 = \Delta^2/d$, then $Z \sim \chi_2(d)$ is a chi-square distribution with d degrees of freedom. We apply the Laurent-Massart inequality with $x = (3/8)^2 d$ — see 3.E.2,

$$\mathbb{P}\left(\|\bar{\mu}(l)\| < \frac{\Delta}{2}\right) = \mathbb{P}\left(Z - d < \frac{\Delta^2}{4\rho^2} - d\right) = \mathbb{P}\left(Z - d < -2\sqrt{d(3/8)^2 d}\right) \leqslant \exp(-(3/8)^2 d) \ .$$

Then, we notice that $\beta = 4$ satisfies $(\beta-1)^2/4 \geqslant 1 + 2(3/8)^2 + 2\sqrt{(3/8)^2}$, we have

$$\begin{aligned}\mathbb{P}\left(\|\bar{\mu}(l)\| > (\beta-1)\frac{\Delta}{2}\right) =& \mathbb{P}\left(Z - d > ((\beta-1)^2/4 - 1)d\right) \\ \leqslant& \mathbb{P}\left(Z - d > 2\sqrt{d(3/8)^2 d} + 2(3/8)^2 d\right) \ .\end{aligned}$$

Now, we use the other side of Laurent-Massart inequality with $x = (3/8)^2 d$ to obtain

$$\mathbb{P}\left(\|\bar{\mu}(l)\| > (\beta-1)\frac{\Delta}{2}\right) \leqslant \exp(-(3/8)^2 d) \ .$$

We recall that we assumed that $d \geqslant (8/3)^2 \log(K/\delta)$, and so $\exp(-(3/8)^2 d) \leqslant \delta/K$. A union bound on $L = K/2$ ensures that Lemma 3.B.11 holds. □

Proof of Lemma 3.B.15. Let $a \in {\mathcal{C}^*}_+$ be an arm labelled by $(l^*_a, 1)$ in $\mathcal{C}^*$ and let $Y, Z \sim \mathbb{P}_{-1}$ —see Definition 3.B.13. We fix an algorithm π for the clustering with bandit feedback problem on $\mathcal{E}(\mathcal{C}^*,\gamma_2)$. The algorithm π is characterized by three families of measurable functions $(\pi_s,\mathcal{T}_s,f_s)_{s\geqslant 1}$ where for all $s \geqslant 1$

— $A_s = \pi_s((A_1,X_1),\dots,(A_{s-1},X_{s-1});U_s)$

— $\mathcal{T} = \min\{t \geqslant 1 \,;\mathcal{T}_s((A_1,X_1),\dots,(A_s,X_s);U_s) = 1\}$

— $\hat{g} = f_{\mathcal{T}}((A_1,X_1),\dots,(A_{\mathcal{T}},X_{\mathcal{T}});U_{\mathcal{T}})$

Here, the sequence (U_s) captures the fact that π can use some external randomness to make decisions. We define $N_{s,a} = \sum_{u=1}^{s} \mathbb{1}_{\{A_u \in \{a,b_a\}\}}$ and $M_s = \sum_{u=1}^{s} \mathbb{1}_{\left\{A_u \in {\mathcal{C}^*}_{l_a^*}\right\}}$. We consider the event B_a on which the inequalities $N_{\mathcal{T},a} = N_a + N_{b_a} \leqslant t$ and $M_{l_a^*} \leqslant T$ holds. Then, the data collected $(X_1, \ldots, X_{\mathcal{T}})$ when π interacts with $\nu({\mathcal{C}^*}_{(a)}, \bar{\mu})$ and $\bar{\mu} \sim \gamma^{\otimes L}$ can be constructed with Y, Z, ϵ, U using the following coupling.

First, we create the observations from arms that belongs to a block different from the one of a, using the variables $(\epsilon_u)_{u \geqslant 1}$. We sample once and for all $(L-1)$ centers by defining for any $l \in [L] \setminus \{l_a^*\}$,

$$\bar{\mu}(l) = \rho \epsilon_l \ ,$$

we observe that $(\bar{\mu}(l))_{l \neq l_a^*} \sim \gamma_2^{\otimes (L-1)}$. Then, for any $s \geqslant 1$, if $A_s \in {\mathcal{C}^*}_l$ with $l \neq l_a^*$, we can create X_s with the expression

$$X_s = C(l) + g \bar{\mu}(l) + \sigma \epsilon_{s+L} \ .$$

Now, for $s \geqslant 1$, when $A_s \in {\mathcal{C}^*}_{l_a^*}$, we use Y, Z,

— $X_s = C(l_a^*) + Y_{N_{s,a}}$ if $A_s = a$

— $X_s = C(l_a^*) - Y_{N_{s,a}}$ if $A_s = b_a$

— $X_s = C(l_a^*) + g^*_{A_s} Z_{M_s}$ if $A_s \in {\mathcal{C}^*}_{l_a^*} \setminus \{a, b_a\}$

We highlight that the law of Y, Z is a marginal distribution that captures the fact that the data obtained from the block ${\mathcal{C}^*}_{l_a^*}$ are obtained using the prior γ for $\bar{\mu}(l_a^*)$.

From there, it is possible to give (explicitly) a function f_a measurable with respect to Y, Z, ϵ, U such that $A_a \cap B_a = f(Y, Z, \epsilon, U)$ where the equality holds in law with respect to $\mathbb{P}_{\pi, {\mathcal{C}^*}_{(a)}}$ (integrated with respect to $\bar{\mu}$). If we use the same measurable function f_a with $X, Y \sim \mathbb{P}_1$, then $A_a \cap B_a = f(Y, Z, \epsilon, U)$ where the equality holds with respect to $\mathbb{P}_{\pi, \mathcal{C}^*}$. □

Proof of Lemma 3.B.16 . We recall that π is a δ-correct algorithm for the problem of clustering with bandit feedback with an oracle. We recall that $A_a = \{\hat{\mathcal{C}} \sim {\mathcal{C}^*}_{(a)}\}$. By construction of the partitions ${\mathcal{C}^*}_{(a)}$, these partitions are not equivalent (for the relation $\sim$). We highlight that this is due to the fact that all the groups contain more than two arms. The events $(A_a)_a$ are disjoints, and $\sqcup_{a \in {\mathcal{C}^*}_+} (A_a \cap B_a) \subset \{\hat{\mathcal{C}} \not\sim \mathcal{C}^*\}$.

Now, we have directly

$$\mathbb{P}_{\pi, \mathcal{C}^*}(\cup_{a \in [n] \setminus S} A_a \cap B_a) \leqslant \mathbb{P}_{\pi, \mathcal{C}^*}(\hat{\mathcal{C}} \not\sim \mathcal{C}^*) \ .$$

By definition, π is δ-correct on $\mathcal{E}_{Sym}(\mathcal{C}^*, \gamma_2)$, we have

$$\begin{aligned} \mathbb{P}_{\pi, \mathcal{C}^*}(\hat{\mathcal{C}} \not\sim \mathcal{C}^*) &= \mathbb{P}_{\pi, \mathcal{C}^*}(\hat{\mathcal{C}} \not\sim \mathcal{C}^* | \mathcal{Y}) \mathbb{P}_{\gamma^{\otimes L}}(\mathcal{Y}) + \mathbb{P}_{\pi, \mathcal{C}^*}(\hat{\mathcal{C}} \not\sim \mathcal{C}^* | \mathcal{Y}^c) \mathbb{P}_{\gamma^{\otimes L}}(\mathcal{Y}^c) \\ &\leqslant \delta + \mathbb{P}_{\gamma^{\otimes L}}(\mathcal{Y}^c) \leqslant 2\delta \end{aligned}$$

For the second point of the lemma, we fix $a \in {\mathcal{C}^*}_+$.

$$\begin{aligned}\mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}((A_a \cap B_a)^c) &= \mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(A_a^c \cup B_a^c|\mathcal{Y})\mathbb{P}_{\gamma^{\otimes L}}(\mathcal{Y}) + \mathbb{P}_{\pi}(A_a^c \cup B_a^c|\mathcal{Y}^c)\mathbb{P}_{\gamma^{\otimes L}}(\mathcal{Y}^c) \\ &\leqslant \mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(A_a^c \cup B_a^c|\mathcal{Y}) + \mathbb{P}_{\gamma^{\otimes L}}(\mathcal{Y}^c) \\ &\leqslant \mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(A_a^c|\mathcal{Y}) + \mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(B_a^c|\mathcal{Y}) + \mathbb{P}_{\gamma^{\otimes L}}(\mathcal{Y}^c) \ .\end{aligned}$$

Now, π is δ-correct which implies that

$$\mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(A_a^c|\mathcal{Y}) = \mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(\hat{\mathcal{C}} \not\sim {\mathcal{C}^*}_{(a)}|\mathcal{Y}) \leqslant \delta \ .$$

For the second term, we use Markov inequality with respect to the distribution $\mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(\cdot|\mathcal{Y})$. We recall that π satisfies a symmetry property and that $t = 3\mathbb{E}_{\pi,{\mathcal{C}^*}_{(a)}}[N_a + N_{b_a}|\mathcal{Y}]$ and $T = 3\mathbb{E}_{\pi,{\mathcal{C}^*}_{(a)}}[M_{l_a^*}|\mathcal{Y}]$. We also recall that $B_a = \{N_a + N_{b_a} \leqslant t\} \cap \{M_{l_a^*} \leqslant T\}$. We have with Markov inequality

$$\mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(B_a^c|\mathcal{Y}) \leqslant \mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(N_a + N_{b_a} > t|\mathcal{Y}) + \mathbb{P}_{\pi,{\mathcal{C}^*}_{(a)}}(M_{l_a^*} > T|\mathcal{Y}) \leqslant \frac{1}{3} + \frac{1}{3} = \frac{2}{3} \ .$$

This concludes the proof of Lemma 3.B.16. □

Proof of Lemma 3.B.14. Let $g \in \{-1, 1\}$ and take $\mathbb{P}_g$ defined in Definition 3.B.13 with the Gaussian prior. We have $\mu \sim \mathcal{N}(0, \rho^2 I_d)$ and conditionally on μ,

— $Y_1, \dots, Y_t, Z_1, \dots, Z_T$ are independent;

— $\forall r \in [t]$, $Y_r \sim \mathcal{N}(g\mu, \sigma^2 I_d)$

— $\forall s \in [T]$, $Z_r \sim \mathcal{N}(\mu, \sigma^2 I_d)$.

First, $Y_1, \dots, Y_t, Z_1, \dots, Z_T$ have i.i.d coordinates and so has μ. Then, it is enough to prove Lemma 3.B.14 in dimension 1. The general case will be obtained by multiplying by d the result for dimension 1. We assume then that $d = 1$, and we want to prove that $\mathrm{KL}(\mathbb{P}_{-g}, \mathbb{P}_g) = \frac{2tT\rho^4}{\sigma^4 + \sigma^2\rho^2(T+t)}$.

Now, we specify the distribution of the vector Y, Z. As μ follows a Gaussian distribution, the vector $(X, Y) = Y_1, \dots, X_t, Z_1, \dots, Z_T$ is a Gaussian vector.

With the law of total variance, we have $Y, Z \sim \mathcal{N}(0, \Sigma_g)$ where Σ_g is the covariance (square) matrix of size $(T + t)$. The matrix Σ_g is defined as follows:

$$\Sigma_g = \sigma^2 I_{t+T} + \rho^2 \begin{pmatrix} J_{t,t} & gJ_{t,T} \\ gJ_{T,t} & J_{T,T} \end{pmatrix} =: \sigma^2 I_{(t+T)} + \rho^2 H_g \ ,$$

where $I_{(t+T)}$ is the identity matrix of size $(T + t)$, and we define $J_{t,T}$ being the rectangle matrix of size $t \times T$ where all entries are equal to 1.

We observe that H_g has a particular shape, in particular, $H_g^2 = (T + t)H_g$. As a consequence, it is easy to compute its inverse. We have:

$$\Sigma_g^{-1} = \frac{1}{\sigma^2} I_{(t+T)} + \frac{1}{\tilde{\rho}^2} H_g \ ;$$

with $\tilde{\rho}^2 = -\frac{\sigma^2}{\rho^2}(\sigma^2 + \rho^2(t+T))$.

Now,

$$\begin{aligned}\Sigma_g^{-1}\Sigma_{-g} - I_{(T+t)} &= \left(\frac{1}{\sigma^2}I_{(T+t)} + \frac{1}{\tilde{\rho}^2}H_g\right)\left(\sigma^2 I_{(T+t)} + \rho^2 H_{-g}\right) - I_{(T+t)} \\ &= \frac{\rho^2}{\sigma^2}H_{-g} + \frac{\sigma^2}{\tilde{\rho}^2}H_g + \frac{\rho^2}{\tilde{\rho}^2}H_g H_{-g} \ ,\end{aligned}$$

where we compute

$$H_g H_{-g} = (t-T)\begin{pmatrix} J_{t,t} & -gJ_{t,T} \\ gJ_{T,t} & -J_{T,T}\end{pmatrix} \ .$$

Finally, with the formula for the KL divergence between two multidimensional Gaussian distribution, we have

$$\begin{aligned}\mathrm{KL}(\mathbb{P}_{-g}, \mathbb{P}_g) &= \frac{1}{2}\left(\log\frac{|\Sigma_g|}{|\Sigma_{-g}|} + Tr(\Sigma_g^{-1}\Sigma_{-g} - I_{(T+t)}) + 0\Sigma_g^{-1}0\right) \\ &= \frac{1}{2}\left(\frac{\rho^2}{\sigma^2}Tr(H_{-g}) + \frac{\sigma^2}{\tilde{\rho}^2}Tr(H_g) + \frac{\rho^2}{\tilde{\rho}^2}Tr(H_g H_{-g})\right) \\ &= \frac{1}{2}\left(\frac{\rho^2(t+T)}{\sigma^2} - \frac{\rho^2(T+t)}{\sigma^2+\rho^2(T+t)} - \frac{\rho^4(T-t)^2}{\sigma^2(\sigma^2+\rho^2(t+T))}\right) \\ &= \frac{1}{2}\left(\frac{\rho^4((t+T)^2-(T-t)^2)}{\sigma^2(\sigma^2+\rho^2(T+t))}\right) = \frac{2tT\rho^4}{\sigma^4+\sigma^2\rho^2(T+t)} \ .\end{aligned}$$

This concludes the computation of $\mathrm{KL}(\mathbb{P}_{-g}, \mathbb{P}_g)$. □

3.C Analysis of ACB

In this section, we establish that ACB 4 is δ-correct, and we control its budget thereby proving the part of Theorem 3.4.1 pertaining to ACB.

Theorem 3.C.1. *Let $\delta > 0$. Let $\Delta > 0$, $\theta > 0$ be the two parameters used in the design of ACB, such that $\mathcal{E}(\Delta, \theta, \sigma, n, K, d) \neq \emptyset$. The ACB algorithm (4) is δ-correct on $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$. Moreover, define $\mathcal{T}_{\mathrm{ACB}}$ for the budget of ACB(δ, Δ, θ). There exist two universal constants c and c' (with c small), independent of all the parameters $\Delta, \theta, \sigma, n, K, d$ and such that for any environment ν in $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$, if we assume that $\frac{\log(K)}{\theta} \leqslant n$, then $\mathbb{E}_{\mathrm{ACB},\nu}[\mathcal{T}_{\mathrm{ACB}}] \leqslant cn + c'A$, and $\mathcal{T}_{\mathrm{ACB}} \leqslant cn + c'(A+B)$ almost surely, where*

$$\begin{aligned}A =& \frac{\sigma^2}{\Delta^2}\left[n\log(n/\delta) + \sqrt{dnK\log(n/\delta)} + \sqrt{d}\frac{\log(K)}{\theta}\right] \\ B =& \frac{\log(K/\delta)}{\theta} + \frac{\sigma^2}{\Delta^2}\frac{1}{\theta}\log\left(\frac{K}{\delta}\right)\left[\sqrt{d} + \log\log\left(\frac{1}{\theta\delta}\right)\right] \ .\end{aligned}$$

In fact, Theorem 3.C.1 is a straightforward consequence of the two following lemmas that separately consider the two sub-routines SRI and ADC.

Lemma 3.C.2 (Analysis of SRI)**.** *Let $\delta > 0$ be fixed, let $\Delta > 0$, let $1/K > \theta > 0$, and let $\hat{S} = \mathrm{SRI}(\delta, \Delta, \theta)$ be the output of Algorithm SRI applied to an environment in $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$. Let $\mathcal{T}_{\mathrm{SRI}}$ be the number of samples used by the SRI routine to compute $\hat{S}$. With probability higher than $1 - \delta$, it holds that $\hat{S}$ contains exactly one arm by group.*

Moreover, there exist two universal constant c and c' (independent of all the parameters) such that almost surely, we have

$$\mathcal{T}_{\mathrm{SRI}} \leqslant c\frac{1}{\theta}\log\left(\frac{K}{\delta}\right) + c'\frac{\sigma^2}{\Delta^2}\frac{1}{\theta}\log\left(\frac{K}{\delta}\right)\left[\log(K) + \sqrt{d} + \log\log\left(\frac{1}{\theta\delta}\right)\right] \ . \tag{3.25}$$

Also, the expected budget satisfies

$$\mathbb{E}_\nu[\mathcal{T}_{\mathrm{SRI}}] \leqslant c\frac{\log(K)}{\theta} + c'\frac{\sigma^2}{\Delta^2}\left[\frac{\log(K)}{\theta}\log\left(\frac{1}{\theta\delta}\right) + \frac{\log(K)}{\theta} + \sqrt{dK\frac{\log(K)}{\theta}\log\left(\frac{K}{\delta}\right)}\right] \ . \tag{3.26}$$

Lemma 3.C.3 (Analysis of ADC)**.** *Let ν be an environment in $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$. Let S be a set of K arms containing exactly one arm belonging to each of the K groups. Let $\hat{\mathcal{C}} = ADC(\delta, \Delta, S)$ be the output of the ACD routine, and $\mathcal{T}_{\mathrm{ADC}}$ be the budget of ADC, i.e., the number of samples used to compute $\hat{\mathcal{C}}$. First, with probability larger than $1 - \delta$, $\hat{\mathcal{C}}$ is a perfect clustering, that is*

$$\mathbb{P}_{\mathrm{ADC},\nu}(\hat{\mathcal{C}} \sim \mathcal{C}^*) \geqslant 1 - \delta \ .$$

Second, there exists a universal constant c such that

$$\mathcal{T}_{\mathrm{ADC}} \leqslant 2n + c\frac{\sigma^2}{\Delta^2}n\log\left(\frac{n}{\delta}\right) + c\frac{\sigma^2}{\Delta^2}\sqrt{dnK\log\left(\frac{n}{\delta}\right)} \ . \tag{3.27}$$

3.C.1 Analysis of the SRI subroutine

In this section, we prove theorem 3.C.2. We organize the proof in several steps. As a warm-up, we discuss the intuition behind the SRI routine in Section 3.C.1. We also further explanation, along with notation. Then, we provide guarantees on SRI which holds even if the parameters Δ, θ used to calibrate SRI are larger than the trues parameters Δ_*,θ_*. We bound the probability that SRI would reject a good candidate for being a representative, or add a bad one to the set S. Then, we prove Lemma 3.C.2 by proving its correction and bounding its budget. The proofs of some technical lemmas are postponed to Section 3.C.2. Finally, we establish Lemma 3.C.3 in Section 3.C.3.

Step 1: explanation and notation. In this section, we fix $\Delta > 0$, and $\theta > 0$ the two parameters used in the design of the SRI routine. Let $\delta > 0$ and $\sigma > 0$. Consider then the algorithm SRI $= \mathrm{SRI}(\delta, \Delta, \theta)$, where the parameters of the algorithm U, $(n_s)_s$, $n_{\max}$ and r are computed with σ,

Δ, θ and δ, using the expressions from Remark 3.C.4. Denote by $\mathbb{P}_\nu$ for the probability induced by $\mathrm{SRI}(\delta, \Delta, \theta)$ and an environment ν.

Let ν be an environment with a hidden partition $\mathcal{C}^* = \mathcal{C}_1^*, \dots, \mathcal{C}_K^*$ and the centers of the groups $\mu(1), \dots, \mu(K)$, with σ-subGaussian noises Assumption 3.2.1. Associate to $\mathcal{C}^*$ the labels $(k(a))_{a\in[n]}$ such that the mean of a is $\mu_a = \mu(k(a))$ and $a \in \mathcal{C}^*{}_{k(a)}$. Recall that Δ_* denotes the minimal gap of ν and θ_* is the proportion of arms in the smallest group. We study how $\mathrm{SRI} = \mathrm{SRI}(\delta, \Delta, \theta)$ behaves when it interacts with the environment ν. For now, ν denotes any environment in the hidden partition model, with subGaussian noises of parameters σ — see Assumption 3.2.1 and Assumption 3.2.3. In particular,for now, we do not assume anything about Δ_* and θ_*.

In the algorithm, there are some parameters defined in (3.10)-(3.12) that we recall here.

Remark 3.C.4. For any $s \geqslant 1$,

$$
\begin{aligned}
U &= \left\lceil \frac{8}{\theta} \log\left(\frac{8K}{\delta}\right) \right\rceil \quad , \\
r &= \lceil \log_2(\log(4U/\delta)) \rceil \quad , \\
n_s &= \left\lceil c_1 \frac{\sigma^2}{\Delta^2} \left(2^s + \log(12K)\right) \right\rceil \vee \left\lceil c_2 \frac{\sigma^2}{\Delta^2} \sqrt{d(2^s + \log(6))} \right\rceil \quad , \\
n_{\max} &= n_r \vee \left\lceil c_3 \frac{\sigma^2}{\Delta^2} \sqrt{d} \log(2K) \right\rceil \quad , \\
s_0 &= r \wedge \min\{s \geqslant 1; n_s \geqslant 2\} \quad ,
\end{aligned}
$$

where the universal constants c_1, c_2, c_3 are respectively defined by $c_1 = 32^2 \vee 8c_{\mathrm{hw}}$, $c_2 = 16\sqrt{c_{\mathrm{hw}}/2} \vee 32\sqrt{2}$, and $c_3 = 32\sqrt{2}$ where c_{hw} is the constant of Hanson-Wright inequality —see Appendix 3.E. Also, the maximum budget $T_{\max}$ (3.13) is defined as

$$
T_{\max} = 2K \left(n_{\max} + \sum_{s=s_0+1}^{r} n_s \right) + 2U n_{s_0} + 2U \sum_{s=s_0+1}^{r} \frac{n_s}{2^{s-4}}
$$

and thereby only depends on θ, Δ, K, and δ

We refer as an epoch of the algorithm, the successive passage in the u loop in the SRI routine. We introduce some notation, taking into account the dependency on u.

At the beginning of the u-th epoch, the arm a_u is taken randomly and uniformly on the set $[n]$ of arms (independently of everything else). We denote by S_u for the set of arms selected as representatives before the u-th epoch. Before the first epoch, we initialize $S_1 = \{a_0\}$. During the u-th epoch, the algorithm decides to add a_u to S_u or not by performing a sequence of tests — see Line 6 to 12 in SRI routine. If a_u is added to S_u, it computes (Line 12) two empirical means $\hat{\mu}_{a_u}$ and $\hat{\mu}'_{a_u}$ using $2n_{\max}$ samples.

We say that

— the arm a_u is bad if there exists $a \in S_u$ such that $\|\mu_{a_u} - \mu_a\| \leqslant \Delta/4$;

— the arm a_u is good if for any arm $a \in S_u$ then $\|\mu_{a_u} - \mu_a\| \geqslant \Delta$.

Remark 3.C.5. If $\Delta \geqslant \Delta_*$, it is possible that some arms are neither good nor bad. Nonetheless, if $\Delta_* \geqslant \Delta$, then all arms from ν are good or bad. Moreover, in this case, the arm a_u is bad if and only if a_u is already represented in S_u.

We want to add a_u to S_u if a_u is good, but we allow the algorithm to reject some good arms if it does not affect the budget (up to a numerical constant). Anyway, we want to reject every bad arm, and reject them as quickly as possible.

For $s \geqslant 1$, we define as ϕ_s^u for the output of RepresentedTest$\Big(a_u, (\hat{\mu}_b, \hat{\mu}'_b)_{b\in S_u}, \Delta, n_s\Big)$ computed during the u-th epoch and for the s-th step. We call it the test (u, s). We further write,

$$\phi_s^u := \mathbb{1}_{\left\{\min_{a\in S_u}\langle \bar{\mu}_{u,s}-\hat{\mu}_a, \bar{\mu}'_{u,s}-\hat{\mu}'_a\rangle \leqslant \frac{\Delta^2}{2}\right\}} \ ,$$

where $\bar{\mu}_{u,s}$ and $\bar{\mu}'_{u,s}$ denotes the two empirical means of arm a_u computed with $2n_s$ samples, when RepresentedTest$\Big(a_u, (\hat{\mu}_b, \hat{\mu}'_b)_{b\in S_u}, \Delta, n_s\Big)$ is called. Remark that these empirical means are only used for the test (u, s).

We start with some s_0 equal to $r \wedge \min\{s \geqslant 1; n_s \geqslant 2\}$ so that n_s strictly increases at each iteration $s \to s + 1$. If, at some test $s_0 \leqslant s \leqslant r$, it holds that $\phi_s^u = 1$, then a_u is rejected and considered as a bad arm (Line 8). If a_u is rejected, we denote by $\mathcal{T}_u$ for the time of rejection of a_u, $\mathcal{T}_u := \min\{s_0 \leqslant s \leqslant r \ ; \phi_s^u = 1\}$. If for all $s = s_0, \dots, r$, ϕ_s^u is equal to zero (False) (condition in Line 10), then a_u is added to S_u (Line 11) and considered as a new representative. If a_u is not rejected, $\mathcal{T}_u = +\infty$ by convention. The empirical mean $\hat{\mu}_{a_u}$ (resp. $\hat{\mu}'_a$) denotes the estimator of μ_{a_u} computed once and for all when a_u is added to S_u (Line 12), and used in every test that follows.

Remark 3.C.6. In RepresentedTest, the condition $\langle \bar{\mu}_{u,s}-\hat{\mu}_a, \bar{\mu}'_{u,s}-\hat{\mu}'_a\rangle \leqslant \frac{\Delta^2}{2}$ is natural, because $\mathbb{E}_\nu[\langle \bar{\mu}_{u,s} - \hat{\mu}_a, \bar{\mu}'_{u,s} - \hat{\mu}'_a\rangle] = \|\mu_{a_u} - \mu_a\|^2$, which is equal to zero if a and a_u are in the same group, and is larger than Δ_* else. This is a benefit of sub-sampling.

Step 2: Control the probability of rejecting a good arm or adding a bad arm to S. In order to use the subGaussian noise assumption— see Assumption 3.2.1, we define $\epsilon_a = \frac{\sqrt{n_{\max}}}{\sigma}(\hat{\mu}_a - \mu_a)$ and $\epsilon_{u,s} = \frac{\sqrt{n_s}}{\sigma}(\bar{\mu}_{u,s} - \mu_{a_u})$ (and respectively $\epsilon'_{u,s}$, ϵ'_a). We refer to Corollary 3.E.4 for concentration inequalities on these variables.

With this notation, we develop the statistic $\langle \bar{\mu}_{u,s} - \hat{\mu}_a, \bar{\mu}'_{u,s} - \hat{\mu}'_a\rangle$ as follows

$$\begin{aligned}\left\langle \bar{\mu}_{u,s} - \hat{\mu}_a, \bar{\mu}'_{u,s} - \hat{\mu}'_a \right\rangle = \|\mu_{a_u} - \mu_a\|^2 &+ \frac{\sqrt{2}\sigma}{\sqrt{n_{\max}}}\left\langle \frac{\epsilon_a + \epsilon'_a}{\sqrt{2}}, \mu_a - \mu_{a_u} \right\rangle + \frac{\sigma^2}{n_{\max}}\langle \epsilon_a, \epsilon'_a\rangle \\ &- \frac{\sigma^2}{\sqrt{n_{\max} n_s}}\langle \epsilon_{u,s}, \epsilon'_a\rangle - \frac{\sigma^2}{\sqrt{n_{\max} n_s}}\left\langle \epsilon'_{u,s}, \epsilon_a \right\rangle \\ &+ \frac{\sqrt{2}\sigma}{\sqrt{n_s}}\left\langle \frac{\epsilon_{u,s} + \epsilon'_{u,s}}{\sqrt{2}}, \mu_{a_u} - \mu_a \right\rangle + \frac{\sigma^2}{n_s}\left\langle \epsilon_{u,s}, \epsilon'_{u,s} \right\rangle \ .\end{aligned} \tag{3.28}$$

We will use concentration inequalities in order to control all deviations of $\left\langle \bar{\mu}_{u,s} - \hat{\mu}_a, \bar{\mu}'_{u,s} - \hat{\mu}'_a \right\rangle$ around its mean $\|\mu_{a_u} - \mu_a\|^2$.

Remark 3.C.7. In order to estimate the means of the representatives added to S, we compute once and for all $(\hat{\mu}_a, \hat{\mu}'_a)$ when arm a is added to S (see Line 12 of the SRI routine). It implies that the test statistics $(\phi^u_s)_{s,u}$ are not independent. This is why we condition on the event $\mathcal{Y}$ defined below, which controls once and for all the deviation of the random variables ϵ_a and ϵ'_a).

We define $\mathcal{Y}$ as the event:

$$\begin{aligned}\mathcal{Y} = &\left\{\forall a \in \hat{S}, \forall k \in [K] \setminus \{k(a)\},\ \left|\left\langle \frac{\epsilon_a + \epsilon'_a}{\sqrt{2}}, \frac{\mu_a - \mu(k)}{\|\mu_a - \mu(k)\|} \right\rangle\right| \leqslant \frac{1}{16}\sqrt{\frac{\Delta^2}{2\sigma^2}}\sqrt{n_{\max}}\right\} \\ &\bigcap \left\{\forall a \in \hat{S}, |\langle \epsilon_a, \epsilon'_a\rangle| \leqslant \frac{1}{16}\frac{\Delta^2}{\sigma^2} n_{\max}\right\} \\ &\bigcap \left\{\forall a \in \hat{S},\ \|\epsilon'_a\|^2 \vee \|\epsilon_a\|^2 - \mathbb{E}[\|\epsilon_a\|^2] \leqslant c_{\mathrm{hw}} \log(12K/\delta) \vee \sqrt{c_{\mathrm{hw}} d \log(12K/\delta)}\right\}\ ,\end{aligned} \tag{3.29}$$

where c_{hw} is the universal constant from Hanson-Wright inequality (Lemma 3.E.3).

Lemma 3.C.8. *For any environment ν, we have,*

$$\mathbb{P}_\nu(\mathcal{Y}) \geqslant 1 - \delta/4\ .$$

We leave the proof of this technical lemma to Section 3.C.2; it is a consequence of the concentration of subGaussian random variables, in particular Hanson-Wright inequality (Lemmas 3.E.3 and Corollary 3.E.4).

We now give an auxiliary lemma that will be used in the rest of the proof as an elementary brick. For every test (u, s), we define the event $\mathcal{Z}_{u,s}$ as

$$\begin{aligned}\mathcal{Z}_{u,s} = &\left\{\exists k \in [K] \setminus \{k(a_u)\}\ ;\ \left|\left\langle \frac{\epsilon_{u,s} + \epsilon'_{u,s}}{\sqrt{2}}, \frac{\mu_{a_u} - \mu(k)}{\|\mu_{a_u} - \mu(k)\|} \right\rangle\right| \geqslant \frac{1}{16}\frac{\Delta}{\sigma}\sqrt{\frac{n_s}{2}}\right\} \\ &\bigcup \left\{\left|\left\langle \epsilon_{u,s}, \epsilon'_{u,s}\right\rangle\right| \geqslant \frac{1}{16}\frac{\Delta^2}{\sigma^2} n_s\right\} \\ &\bigcup \left\{\exists a \in \hat{S}\ ;\ |\langle \epsilon_{u,s}, \rho'_a\rangle| + \left|\left\langle \epsilon'_{u,s}, \rho_a\right\rangle\right| \geqslant \frac{1}{4}\frac{\Delta^2}{\sigma^2}\sqrt{\frac{n_s n_{\max}}{4d + c_{\mathrm{hw}} l \vee \sqrt{c_{\mathrm{hw}} d l}}}\right\}\ ,\end{aligned} \tag{3.30}$$

with $l = \log(12K/\delta)$ and c_{hw} is the constant from Lemma 3.E.3.

Lemma 3.C.9. *The sequence of events $(\mathcal{Z}_{u,s})_{u \geqslant 1, s \geqslant s_0}$ satisfies four properties.*

1. *Conditionally on the random directions $(\rho_a, \rho'_a)_a := \left(\frac{\epsilon_a}{\|\epsilon_a\|}, \frac{\epsilon'_a}{\|\epsilon'_a\|}\right)_a$, with $a \in \hat{S}$ the events $\mathcal{Z}_{u,s}$ are independent (for all test (u, s)).*
2. *For all $u \geqslant 1$ and $\forall s_0 \leqslant s \leqslant r$, the inclusion $\mathcal{Y} \cap \{a_u$ is good and $\phi^u_s = 1\} \subset \mathcal{Z}_{u,s}$ holds.*
3. *For all $u \geqslant 1$ and $\forall s_0 \leqslant s \leqslant r$, the inclusion $\mathcal{Y} \cap \{a_u$ is bad and $\phi^u_s = 0\} \subset \mathcal{Z}_{u,s}$ also holds.*
4. *Finally, we have $\forall u \geqslant 1$ and $\forall s_0 \leqslant s \leqslant r$, $\mathbb{P}_\nu(\mathcal{Z}_{u,s}) \leqslant \exp(-2^s)$.*

This result is important to prove that the SRI routine actually rejects bad arms and add good arms to S. We recall that $\phi_s^u = 1$ implies that the test (u, s) would reject a_u.

Sketch of proof. The terminology bad and good was introduced in the previous paragraph. Let $u \geqslant 1$ and $s_0 \leqslant s \leqslant r$. The variables $(\epsilon_{u,s}, \epsilon'_{u,s})$ are mutually independent for any test (u, s), and the event $\mathcal{Z}_{u,s}$ is measurable with respect to $\left((\rho_a, \rho'_a)_{a\in\hat{S}}, \epsilon_{u,s}, \epsilon'_{u,s}\right)$, the first point of Lemma 3.C.9 is clear.

The construction of the event $\mathcal{Z}_{u,s}$ follows from the decomposition in Equation (3.28). We notice that if the event $\mathcal{Y}$ holds, the estimation of all centers (μ_a, μ'_a) for a in $\hat{S}$ are concentrated on the true centers. The points two and three follow from this observation. Moreover, the deviation of $\bar{\mu}_{u,s}$ around μ_{a_u} are subGaussian, the point 4 will follow from subGaussian concentration inequalities. We postpone the proof of this result to Section 3.C.2. □

Now, in the next lemma, we prove that r is large enough to ensure that, within the procedure, every bad arm is rejected with large probability.

Lemma 3.C.10. *Recall that $r = \lceil \log_2(\log(4U/\delta)) \rceil$. If $\mathcal{Y}$ holds, then, with probability higher than $1 - \delta/4$, we do not add bad arms to S, i.e.,*

$$\mathbb{P}_\nu \left(\{ \exists a, b \in \hat{S} \, , \|\mu_a - \mu_b\| \leqslant \Delta/4 \} \cap \mathcal{Y} \right) \leqslant \frac{\delta}{4} \ .$$

Proof. Within the procedure, the algorithm picks at most U arms (without counting a_0) — see Line 4 in SRI routine. If there exists $a, b \in \hat{S}$ such that $\|\mu_a - \mu_b\| \leqslant \Delta/4$, it means that there exists an epoch $1 \leqslant u \leqslant U$, where the last test statistic ϕ_r^u is equal to zero, although a_u is bad. If the events $\mathcal{Y}$ holds, using the third point of Lemma 3.C.9, the event $\mathcal{Z}_{u,r}$ holds.

In terms of probability, with a simple union bound, we have

$$\begin{aligned} \mathbb{P}_\nu \left(\{ \exists a, b \in \hat{S} \, , \|\mu_a - \mu_b\| \leqslant \Delta/4 \} \cap \mathcal{Y} \right) &\leqslant \mathbb{P}_\nu(\{ \exists 1 \leqslant u \leqslant U \, , a_u \text{ is bad } , \phi_r^u = 0 \} \cap \mathcal{Y}) \\ &\leqslant \sum_{u=1}^{U} \mathbb{P}_\nu(\mathcal{Z}_{u,r}) \ . \end{aligned}$$

We recall that the probability of $\mathcal{Z}_{u,r}$ is smaller than $\exp(-2^r)$, and we conclude with the expression of r.

$$\mathbb{P}_\nu \left(\{ \exists a, b \in \hat{S} \, , \|\mu_a - \mu_b\| \leqslant \Delta/4 \} \cap \mathcal{Y} \right) \leqslant U \exp(-2^r) \leqslant \frac{\delta}{4} \ .$$

□

Step 3: SRI is δ-correct. For all epochs $u \geqslant 1$ and $s_0 \leqslant s \leqslant r$, we denote $H_{s,u} := \sum_{v=1}^{u-1} \mathbb{1}_{\{s \leqslant \mathcal{T}_v \leqslant r\}}$ as the number of arms that are rejected with a time of rejection larger than s within the epochs $1, \ldots, u-1$. We highlight that $H_{s_0,u}$ is the total number of arms rejected before epoch u.

Now, we define $M = \inf\{u \geqslant 1; |S_u| = K \text{ or } u > U \text{ or Budget} > T_{\max}\}$ as the stopping time (i.e., the number of epochs) of SRI. It corresponds to the number of arms taken randomly from $[n]$, (namely $a_0, \dots, a_{M-1}$) to build the set $\hat{S}$. When M is reached, SRI outputs $\hat{S} = S_M$, whether or not it contains K arms.

We prove that, with probability higher than $1-\delta$, on $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$, it holds that $|S_M| = K$ and S_M contains one representative of each cluster.

Now, assume that $\theta_* \geqslant \theta$ and $\Delta_* \geqslant \Delta$. We use Section 3.C.1 to prove that the SRI routine outputs a set with exactly one arm by group when it interacts with the environment $\nu \in \mathcal{E}(\Delta, \theta, \sigma, n, K, d)$.

First, we define

$$\mathcal{X} := \bigcap_{s=s_0+1}^{r} \{H_{s,M} < \frac{1}{2^{s-4}} U\} \ . \tag{3.31}$$

The definition (3.13) of $T_{\max}$ ensures that, on the event $\mathcal{X}$, the stopping condition of the SRI routine reduces to the condition $\{|S_u| = K\} \cup \{u > U\}$ and then $M = (U+1) \wedge \min\{u \geqslant 1; |S_u| = K\}$. It turns out that the event $\mathcal{X}$ has a large probability when it is intersected with $\mathcal{Y}$.

Lemma 3.C.11. *Let $1 + s_0 \leqslant s \leqslant r$ and recall that $H_{s,M} = \#\{u \in [|1; M-1|], s \leqslant \mathcal{T}_u \leqslant r\}$. It holds that*

$$\mathbb{P}_\nu \left(\{H_{s,M} \geqslant \frac{1}{2^{s-4}} U\} \cap \mathcal{Y} \right) \leqslant \exp(-U/2) \ . \tag{3.32}$$

This implies that

$$\mathbb{P}_\nu(\mathcal{Y} \cap \mathcal{X}^c) \leqslant \frac{\delta}{8} \ .$$

Proof of Lemma 3.C.11. We start with the first statement of Lemma 3.C.11, take s such that $s_0 < s \leqslant r$.

If $s = 1, 2$ or 3, the inequality is trivial because $H_{s,M} \leqslant U$, we assume that $s > 3 \vee s_0$. Recall that $\nu \in \mathcal{E}(\Delta, \theta, \sigma, n, K, d)$, so that all arms are either good or bad. By definition of $\mathcal{T}_u$, if $s \leqslant \mathcal{T}_u \leqslant r$, it means that at some test $t \in [s, r]$, $\phi_t^u = 1$ but $\phi_t^u = 0$ for $t < s$. Moreover, each arm is either good or bad because $\Delta_* \geqslant \Delta$. The following inclusion holds then,

$$\begin{aligned} \{s \leqslant \mathcal{T}_u \leqslant r\} \cap \mathcal{Y} &= (\{s \leqslant \mathcal{T}_u \leqslant r\} \cap \{a_u \text{ is good } \} \cap \mathcal{Y}) \bigsqcup (\{s \leqslant \mathcal{T}_u \leqslant r\} \cap \{a_u \text{ is bad } \} \cap \mathcal{Y}) \\ &\subset (\cup_{s \leqslant t \leqslant r} \{\phi_t^u = 1\} \cap \{a_u \text{ is good}\} \cap \mathcal{Y}) \bigsqcup (\{\phi_{s-1}^u = 0\} \cap \{a_u \text{ is bad}\} \cap \mathcal{Y}) \ . \end{aligned}$$

We use the points 2 of Lemma 3.C.9 to get the inclusion $\{\phi_t^u = 1\} \cap \{a_u \text{ is good}\} \cap \mathcal{Y} \subset \mathcal{Z}_{u,t}$ valid for any $t \in [s, r]$. Using the point 3 of the same lemma with $s - 1 \geqslant s_0$, we have also $\{\phi_{s-1}^u = 0\} \cap \{a_u \text{ is bad}\} \cap \mathcal{Y} \subset \mathcal{Z}_{u,s-1}$. Then,

$$\{s \leqslant \mathcal{T}_u \leqslant r\} \cap \mathcal{Y} \subset \bigcup_{s-1 \leqslant t \leqslant r} \mathcal{Z}_{u,t} \ , \tag{3.33}$$

and we recall that the events $\left(\bigcup_{s-1 \leqslant t \leqslant r} \mathcal{Z}_{u,t}\right)_{u \geqslant 1}$ are independent according to Lemma 3.C.9 if we condition on the random directions $(\rho_a, rho'_a)_a$. Now, we use a union bound on t and the bound

$\mathbb{P}_\nu(\mathcal{Z}_{u,t}) \leqslant \exp(-2^t)$ valid for any $t \in [s_0, r]$, and we have,

$$\mathbb{P}_\nu\left(\bigcup_{s-1\leqslant t\leqslant r} \mathcal{Z}_{u,t}\right) \leqslant \exp(-2^{s-2}) \ .$$

From the inequality $H_{s,M} = \sum_{u=1}^{M-1} \mathbb{1}_{\{s\leqslant \mathcal{T}_u\leqslant r\}} \leqslant \sum_{u=1}^{U} \mathbb{1}_{\{s\leqslant \mathcal{T}_u\leqslant r\}}$, we deduce that $H_{s,M}\mathbb{1}_{\mathcal{Y}}$ is stochastically dominated by $\mathcal{B}(U, q_s)$ where $q_s := \exp(-2^{s-2})$.

We use Chernoff bound with $\alpha \geqslant \sqrt{q_s}$, we have

$$\begin{aligned}\mathbb{P}_\nu\left(H_{s,M}\mathbb{1}_{\mathcal{Y}} \geqslant (1+\alpha/q_s)q_sU\right) \leqslant& \left[\frac{e^{\alpha/q_s}}{(1+\alpha/q_s)^{1+\alpha/q_s}}\right]^{q_sU}\\ =& \exp\Big[\alpha U\Big(1-\log\Big(1+\tfrac{\alpha}{q_s}\Big)\,(1+\tfrac{q_s}{\alpha})\Big)\Big]\\ =&(1+\tfrac{\alpha}{q_s})^{-\alpha U/2}\exp[\alpha U(1-\tfrac{1}{2}\log(1+\tfrac{\alpha}{q_s}))-Uq_s\log(1+\tfrac{\alpha}{q_s})] \ .\end{aligned}$$

As $\alpha \geqslant \sqrt{q_s}$ and $s \geqslant 4$, we have $\frac{\alpha}{q_s} \geqslant \frac{1}{\sqrt{q_s}} = \exp(2^{s-3}) \geqslant e^2-1$ and then $1-\log(1+\alpha/q_s)/2 \leqslant 0$. It follows that

$$\begin{aligned}\mathbb{P}_\nu\left(H_{s,M}\mathbb{1}_{\mathcal{Y}} \geqslant 2\alpha U\right) \leqslant&(1+\alpha/q_s)^{-\alpha U/2}\\ \leqslant&(1+1/\sqrt{q_s})^{-\alpha U/2}\\ \leqslant& \exp\left(-\frac{\alpha U}{2}2^{s-3}\right) = \exp(-\alpha U 2^{s-4}) \ .\end{aligned}$$

Finally, taking $\alpha = 1/2^{s-3} \geqslant \exp(-2^{s-3}) = \sqrt{q_s}$, we deduce the first result of Lemma 3.C.11

$$\mathbb{P}_\nu\left(\{H_{s,M} \geqslant \frac{1}{2^{s-4}}U\}\cap\mathcal{Y}\right) \leqslant \mathbb{P}_\nu\left(H_{s,M}\mathbb{1}_{\mathcal{Y}} \geqslant \frac{1}{2^{s-4}}U\right) \leqslant \exp(-U/2) \ .$$

Directly,

$$\mathcal{Y}\cap\mathcal{X}^c = \bigcup_{s=1}^{r} \mathcal{Y}\cap\{H_{s,M} \geqslant \frac{1}{2^{s-4}}U\} \ ,$$

then, we use the first part of the lemma and a union bound,

$$\mathbb{P}\left(\mathcal{Y}\cap\mathcal{X}^c\right) \leqslant r\exp(-U/2) \leqslant \delta/8 \ ,$$

where we conclude with the expression of $U \geqslant \frac{8}{\theta}\log\left(\frac{8K}{\delta}\right) \geqslant 2\log\left(\frac{8r}{\delta}\right)$. The last inequality follows from the expression of r.

□

Now, we study the probability of adding K arms to S, before reaching the maximum number of epochs U.

Lemma 3.C.12. *Recall that $\hat{S}$ denotes the output of SRI, consider a group C_k^*, it holds that*

$$\mathbb{P}_\nu(\{\hat{S} \cap \mathcal{C}_k^* = \emptyset\} \cap \{\forall a, b \in \hat{S}, \mu_a \neq \mu_b\} \cap \mathcal{Y} \cap \mathcal{X}) \leqslant 2\exp\left(-\frac{U\theta}{8}\right) \ .$$

Proof of Lemma 3.C.12. Let $k \in [K]$, we study the event $\{\hat{S} \cap \mathcal{C}_k^* = \emptyset\}$, event where the group $\mathcal{C}_k^*$ is not represented in $\hat{S}$ (the output of the SRI routine). We use here the assumption that the groups are nonempty. If $\hat{S}$ contains arms from different groups but no arm from the group $\mathcal{C}_k^*$, it implies that $|\hat{S}| < K$ and then, if the event $\mathcal{X}$ also holds, the algorithm has passed every epoch from $u = 1$ to U, i.e., $M = U + 1$. In particular, the algorithm rejected every arm from $\mathcal{C}_k^* \cap \{a_1, \dots, a_U\}$. We have the inclusion between events,

$$\{\hat{S} \cap \mathcal{C}_k^* = \emptyset\} \cap \{\forall a \neq b \in \hat{S}, \mu_a \neq \mu_b\} \cap \mathcal{X} \subset \bigcap_{u \in B_k} \{a_u \text{ rejected}\} \ ,$$

where $B_k := \mathcal{C}_k^* \cap \{a_1, \dots, a_U\}$ and denote $X_k := |B_k|$. As $\{a_1, \dots, a_U\}$ are i.i.d and uniform on $[n]$ then X_k is binomial with parameters U and $\theta_k = |C_k|/n \geqslant \theta_* \geqslant \theta$. Using Hoeffding's bound and taking $\alpha \in (0, 1)$ to be specified later, it holds that

$$\mathbb{P}_\nu(X_k \leqslant \theta U(1-\alpha)) \leqslant \exp(-2\alpha^2 U\theta) \ . \tag{3.34}$$

Then, as $\Delta_* \geqslant \Delta$, the arms in $C_k \cap \{a_1, \dots, a_U\}$ are good until one of them is added to S. In particular, if none are added to S, they are all good. We then have

$$\{|\hat{S}| \cap \mathcal{C}_k^* = \emptyset\} \cap \{\forall a \neq b \in \hat{S}, \mu_a \neq \mu_b\} \cap \mathcal{Y} \cap \mathcal{X} \subset \bigcap_{u \in B_k} \{a_u \text{ good and rejected}\} \cap \mathcal{Y} \ .$$

If a_u is rejected, it means that $\phi_s^u = 1$ for some $s_0 \leqslant s \leqslant r$, we then have the inclusion $\{a_u \text{ is good and rejected}\} \subset \bigcup_{s \geqslant s_0} \{a_u \text{ is good and } \phi_s^u = 1\}$. According to the second point of Lemma 3.C.9, we have $\{a_u \text{ is good and rejected}\} \cap \mathcal{Y} \subset \cup_{s \geqslant s_0} \mathcal{Z}_{u,s}$. We denote $\mathcal{Z}_u = \cup_{s \geqslant s_0} \mathcal{Z}_{u,s}$.

In terms of probability, we have $\mathbb{P}_\nu\left(\cap_{u \in B_k} \{a_u \text{ rejected}\} \cap \mathcal{Y}\right) \leqslant \mathbb{P}_\nu\left(\cap_{u \in B_k} \mathcal{Z}_u\right)$.

Then, we also have $\mathbb{P}_\nu(\mathcal{Z}_{u,s}) \leqslant \exp(-2^s)$ for any $s \in [s_0, r]$, and with a union bound, $\mathbb{P}_\nu(\mathcal{Z}_u) \leqslant 1/2$. Moreover, with the first point of Lemma 3.C.9, we deduce that the events $(\mathcal{Z}_u)_u$ are independent (if we condition on ρ_a, ρ'_a).

If $X_k > \theta U(1-\alpha)$ and using the independence of the events $(\mathcal{Z}_u)_u$, we have

$$\mathbb{P}_\nu\left(\{X_k > \theta U(1-\alpha)\} \cap \left(\bigcap_{u \in B_k} \mathcal{Z}_u\right)\right) \leqslant \left(\frac{1}{2}\right)^{\theta U(1-\alpha)} \ .$$

Finally, with Equation (3.34), we have

$$\begin{aligned}
&\mathbb{P}_\nu(\{\hat{S}\cap\mathcal{C}_k^*=\emptyset\}\cap\{\forall a\neq b\in\hat{S},\mu_a\neq\mu_b\}\cap\mathcal{Y}\cap\mathcal{X})\\
\leqslant&\mathbb{P}_\nu\left(\{X_k>\theta U(1-\alpha)\}\cap\left(\bigcap_{u\in B_k}\mathcal{Z}_u\right)\right)+\mathbb{P}_\nu\left(X_k\leqslant\theta U(1-\alpha)\right)\\
\leqslant&\exp\left(-\log(2)\theta U(1-\alpha)\right)+\exp\left(-2\alpha^2U\theta\right)\\
\leqslant&2\exp(-U\theta/8)\ .
\end{aligned}$$

In the last line, we took $\alpha=1/2$. □

We now have all the tools that we need to prove that SRI(δ,Δ,θ) is δ-correct on the collection of environments $\mathcal{E}(\Delta,\theta,\sigma,n,K,d)$ for the representative identification problem. Indeed, with probability higher than $1-\delta$, SRI outputs a set of K representatives for environments that are $\mathcal{E}(\Delta,\theta,\sigma,n,K,d)$.

Proof of the first statement of Lemma 3.C.2.. Let $\nu\in\mathcal{E}(\Delta,\theta,\sigma,n,K,d)$.

As a direct consequence of Lemma 3.C.10, we know that with high probability, $\hat{S}$ does not contain two arms from the same group.

$$\mathbb{P}_\nu\left(\{\exists a,b\in\hat{S}\ ,\mu_a=\mu_b\}\cap\mathcal{Y}\right)\leqslant\mathbb{P}_\nu\left(\{\exists a,b\in\hat{S}\ ,\|\mu_a-\mu_b\|\leqslant\Delta/4\}\cap\mathcal{Y}\right)\leqslant\frac{\delta}{4}\ .$$

Now, with Lemma 3.C.12, for all $k\in[K]$,

$$\mathbb{P}_\nu\left(\{\hat{S}\cap\mathcal{C}_k^*=\emptyset\}\cap\{\forall a\neq b\in\hat{S},\mu_a\neq\mu_b\}\cap\mathcal{Y}\cap\mathcal{X}\right)\leqslant2\exp(-U\theta/8)\ .$$

Now, we recall that $U\geqslant\frac{8}{\theta}\log\left(\frac{8K}{\delta}\right)$ and then $\exp(-U\theta/8)\leqslant\delta/8K$. If the set $\hat{S}$ contains strictly less than K arms then at least one group is not represented. With a union bound on $k\in[K]$,

$$\mathbb{P}_\nu\left(\{|\hat{S}|<K\}\cap\{\forall a\neq b\in\hat{S},\mu_a\neq\mu_b\}\cap\mathcal{Y}\cap\mathcal{X}\right)\leqslant2K\exp(-U\theta/8)\leqslant\delta/4\ .$$

Finally, together with Lemmas 3.C.8 and 3.C.11, we conclude that

$$\begin{aligned}
\mathbb{P}_\nu\left(\forall k\in[K]\,,\exists!a\in\hat{S}\,;\mu_a=\mu(k)\right)\geq&\,1-\mathbb{P}_\nu(\mathcal{Y}^c)-\mathbb{P}_\nu(\mathcal{Y}\cap\mathcal{X}^c)\\
&-\mathbb{P}_\nu\left(|\hat{S}|<K\}\cap\{\forall a\neq b\in\hat{S},\mu_a\neq\mu_b\}\cap\mathcal{Y}\cap\mathcal{X}\right)\\
&-\mathbb{P}_\nu\left(\{\exists a,b\in\hat{S}\ ,\mu_a=\mu_b\}\cap\mathcal{Y}\right)\\
\geqslant&\,1-\delta\ .
\end{aligned}$$

In summary, we have proved that SRI is δ-correct on $\mathcal{E}(\Delta,\theta,\sigma,n,K,d)$ for the representatives' identification problem. □

Step 4: upper bound on the budget $\mathcal{T}_{\mathrm{SRI}}$. We establish here the bounds on the budget $\mathcal{T}_{\mathrm{SRI}}$ of Lemma 3.C.2.

Explanation on $T_{\max}$
By definition (3.13) of $T_{\max}$, we know that, almost surely, the total budget $\mathcal{T}_{\mathrm{SRI}}$ satisfies:

$$\mathcal{T}_{\mathrm{SRI}} \leq T_{\max} = 2K\left(n_{\max} + \sum_{s=s_0+1}^{r} n_s\right) + 2Un_{s_0} + 2U\sum_{s=s_0+1}^{r}\frac{n_s}{2^{s-4}}\ . \tag{3.35}$$

Although plugging the values of $n_{\max}$, n_s, U, and s_0, will lead to (3.25), we start by gently describing the budget of SRI in order to give intuition on the definition of $T_{\max}$.

To analyze the budget, we divide the budget in two parts, $\mathcal{T}_{\mathrm{SRI}} = \mathcal{T}^a_{\mathrm{SRI}} + \mathcal{T}^r_{\mathrm{SRI}}$, where $\mathcal{T}^a_{\mathrm{SRI}}$ is the number of samples used for arms that are added to $\hat{S}$ and $\mathcal{T}^r_{\mathrm{SRI}}$ is the number of samples uses for arms that are rejected.

First, we study $\mathcal{T}^a_{\mathrm{SRI}}$. From the algorithm, the arms that are selected in $\hat{S}$ are arms that pass successfully all the tests, and after the tests, they are sampled again $2n_{\max}$ times. In total, each arm in $\hat{S}$ is sampled $2n_{\max} + \sum_{s=s_0}^{r} 2n_s$, the factor 2 comes from the fact that we compute two empirical means at each time. We have then

$$\mathcal{T}^a_{\mathrm{SRI}} = 2|\hat{S}|n_{\max} + (|\hat{S}|-1)\sum_{s=s_0}^{r} 2n_s \leqslant 2K\left(n_{\max} + \sum_{s=s_0+1}^{r} n_s\right) + 2(|\hat{S}|-1)n_{s_0}\ . \tag{3.36}$$

Now, consider the budget spent for arms that are ultimately rejected during the procedure. For $s_0 \leqslant s \leqslant r$, we defined previously $H_{s,M} = \sum_{u=1}^{M-1} \mathbb{1}_{\{s\leqslant \mathcal{T}_u \leqslant r\}}$, as the number of arms rejected after at least s tests in the procedure. The algorithm outputs $\hat{S}$ after $M-1$ epochs. In particular, $H_{s_0,M} = M - |\hat{S}|$. Besides, if the candidate a_u is rejected, it is sampled $\sum_{s=s_0}^{\mathcal{T}_u} 2n_s$ times. This leads us to the equality

$$\mathcal{T}^r_{\mathrm{SRI}} = \sum_{u=1}^{M-1}\sum_{s=s_0}^{\mathcal{T}_u} 2n_s \mathbb{1}_{\{a_u \text{ is rejected }\}} = \sum_{s=s_0}^{r} 2H_{s,M}n_s = 2\left(M - |\hat{S}|\right)n_{s_0} + 2\sum_{s=s_0+1}^{r} H_{s,M}n_s\ . \tag{3.37}$$

This justifies the definition $T_{\max}$ (3.13), as under the large probability event $\mathcal{X}$, $\mathcal{T}_{\mathrm{SRI}} \leqslant T_{\max}$ is directly implied by (3.36) and (3.37).

Upper bound on $T_{\max}$
In order to upper bound $T_{\max}$ (3.13), we need the following lemmas, whose proofs are postponed to Section 3.C.2. These lemmas are direct consequences of the expressions of n_s, $n_{\max}$, U and r from equations (3.10),(3.12).

Lemma 3.C.13. *Using the explicit expression of r, n_s, $n_{\max}$ and s_0 from Remark 3.C.4, we have,*

up to a universal constant c, the inequality

$$K\left(n_{\max}+\sum_{s=s_0+1}^{r} n_s\right) \leqslant K + c\frac{\sigma^2}{\Delta^2}\left[\frac{\log(K)}{\theta}\log(K/\delta)+\sqrt{dK\frac{\log(K)}{\theta}\log(K/\delta)}\right]$$
$$\leqslant K + c\frac{\sigma^2}{\Delta^2}\frac{1}{\theta}\log\left(\frac{K}{\delta}\right)\left[\log(K)+\sqrt{d}\right] \ .$$

Lemma 3.C.14. *Using the explicit expression of r, U, n_s and s_0 from Remark 3.C.4, we have*

$$U\sum_{s=s_0+1}^{r}\frac{n_s}{2^{s-4}} \leqslant c\frac{\sigma^2}{\Delta^2}\frac{1}{\theta}\log\left(\frac{K}{\delta}\right)\left[\log(K)+\sqrt{d}+\log\log\left(\frac{1}{\theta\delta}\right)\right] \ ,$$

where c is a numerical constant.

Now, by combining (3.35) with Lemmas 3.C.13 and 3.C.14, and bounding Un_{s_0} by $U+A$ where A is the right side of Lemma 3.C.14, we conclude that

$$\mathcal{T}_{\mathrm{SRI}} \leq T_{\max} \leqslant 2(U+K) + c\frac{\sigma^2}{\Delta^2}\frac{1}{\theta}\log\left(\frac{K}{\delta}\right)\left[\log(K)+\sqrt{d}+\log\log\left(\frac{1}{\theta\delta}\right)\right] \ , \tag{3.38}$$

where c is a numerical constant. We have proved (3.25).

Upper bound on $\mathbb{E}[\mathcal{T}_{\mathrm{SRI}}]$

We now upper bound the expectation of $\mathcal{T}_{\mathrm{SRI}}$. For that purpose, we now assume that $\nu \in \mathcal{E}(\Delta,\theta,\sigma,n,K,d)$. We will prove (3.26).

With the same decomposition on the budget $\mathcal{T}_{\mathrm{SRI}} = \mathcal{T}^a_{\mathrm{SRI}} + \mathcal{T}^r_{\mathrm{SRI}}$, and by linearity of the expectation, we deduce from (3.36) and (3.37) that

$$\begin{aligned}
\mathbb{E}_\nu[\mathcal{T}_{\mathrm{SRI}}] &= \mathbb{E}_\nu[\mathcal{T}_{\mathrm{SRI}}\mathbb{1}_{\mathcal{Y}^c}] + \mathbb{E}_\nu[\mathcal{T}_{\mathrm{SRI}}\mathbb{1}_{\mathcal{Y}}] \\
&\leq T_{\max}\delta + 2n_{s_0}\mathbb{E}_\nu\left[(M-1)\mathbb{1}_{\mathcal{Y}}\right] + \sum_{s=s_0+1}^{r} 2n_s\mathbb{E}_\nu\left[H_{s,M}\mathbb{1}_{\mathcal{Y}}\right] \\
&+ 2\mathbb{E}_\nu[|\hat{S}|\mathbb{1}_{\mathcal{Y}}]\left(n_{\max}+\sum_{s=s_0+1}^{r} n_s\right) \\
&\leq T_{\max}\delta + 2K\left(n_{\max}+\sum_{s=s_0+1}^{r} n_s\right) + 2n_{s_0}\mathbb{E}_\nu\left[(M-1)\mathbb{1}_{\mathcal{Y}}\right] + \sum_{s=s_0+1}^{r} 2n_s\mathbb{E}_\nu\left[H_{s,M}\mathbb{1}_{\mathcal{Y}}\right] \ .
\end{aligned} \tag{3.39}$$

Let us focus on the terms $\mathbb{E}_\nu\left[(M-1)\mathbb{1}_{\mathcal{Y}}\right]$ and $\mathbb{E}_\nu\left[H_{s,M}\mathbb{1}_{\mathcal{Y}}\right]$.

Recall that $H_{s,M} = \sum_{u=1}^{M-1}\mathbb{1}_{\{s\leqslant\mathcal{T}_u\leqslant r\}}$. We also recall Equation (3.33), valid for $s > s_0$, and which is a consequence of Lemma 3.C.9 and the fact that $\Delta_* \geqslant \Delta$,

$$\{s\leqslant\mathcal{T}_u\leqslant r\}\cap\mathcal{Y} \subset \bigcup_{s-1\leqslant t\leqslant r}\mathcal{Z}_{u,t} \ .$$

Then,

$$\mathbb{E}_\nu\left[H_{s,M}\mathbb{1}_{\mathcal{Y}}\right] = \mathbb{E}_\nu\left[\sum_{u=1}^{M-1}\mathbb{1}_{\{\{s\leqslant\mathcal{T}_u\leqslant r\}\cap\mathcal{Y}\}}\right] \leqslant \mathbb{E}_\nu\left[\sum_{u=1}^{M-1}\mathbb{1}_{\left\{\bigcup_{s-1\leqslant t\leqslant r}\mathcal{Z}_{u,t}\right\}}\right] \ .$$

We can use Wald's equation. Indeed, if we condition on the direction of the estimated centers (ρ_a, ρ'_a), the random variables $\mathbb{1}_{\left\{\bigcup_{s-1\leqslant t\leqslant r}\mathcal{Z}_{u,t}\right\}}$ are independent and identically distributed for $u = 1, \dots, U$. Moreover, $\mathbb{P}_\nu(\bigcup_{s-1\leqslant t\leqslant r}\mathcal{Z}_{u,t}) \leqslant \exp(-2^{s-4})$ thanks to Lemma 3.C.9. We observe that M is a stopping time with respect to the filtration naturally associated to the sequence of epochs, and the sequence $\left(\bigcup_{s-1\leqslant t\leqslant r}\mathcal{Z}_{u,t}\right)_u$ is adapted to this filtration. With Wald's equation, we deduce that

$$\mathbb{E}_\nu\left[H_{s,M}\mathbb{1}_{\mathcal{Y}}\right] \leqslant \mathbb{E}_\nu[(M-1)\mathbb{1}_{\mathcal{Y}}]\exp(-2^{s-4}) \ . \tag{3.40}$$

Hence, we conclude that

$$\sum_{s=s_0+1}^{r} 2n_s\mathbb{E}_\nu\left[H_{s,M}\mathbb{1}_{\{\mathcal{Y}\}}\right] \leqslant \sum_{s=s_0+1}^{r} 2n_s\mathbb{E}_\nu[(M-1)\mathbb{1}_{\mathcal{Y}}]\exp(-2^{s-4}) \ . \tag{3.41}$$

It remains to bound $\mathbb{E}_\nu[(M-1)\mathbb{1}_{\mathcal{Y}}]$. We will control stochastically $M-1$ by a sum of geometric random variables. Recall that S_u is the state of the set of representatives at the beginning of the epoch u and $\hat{S} = S_M$. Define for $k \in [K-1]$, $M_k = \sum_{u=1}^{M-1}\mathbb{1}_{\{|S_u|=k\}}$. The number M_k is the number of epochs necessary to add the $(k+1)$-th arm to S. Once $|S_u| = K$, the algorithm stops, so that

$$M - 1 = \sum_{k=1}^{K-1} M_k \ .$$

Fix now $k \in [K-1]$. If $|\hat{S}| < k$, we have $M_k = 0$. We assume that $|\hat{S}| \geqslant k$, and we condition on $S^{(k)}$ and $\mathcal{Y}$, the set containing the k first arms that were added to $\hat{S}$. Let u such that $|S_u| = k$. Thanks to the second point of Lemma 3.C.9, it holds that

$$\{a_u \text{ is good }\} \cap \mathcal{Y} \cap \left(\cap_{s\in[s_0,r]}\mathcal{Z}^c_{u,s}\right) \subset \cap_{s\in[s_0,r]}\{\phi^u_s = 0\} = \{a_u \text{ is added to } S\} \ .$$

Then, the events $\{a_u \text{ is good }\}$, $\mathcal{Y}$ and $\cap_{s\in[r]}\mathcal{Z}^c_{u,s}$ are independent. Moreover, $\mathbb{P}_\nu(a_u \text{ is good }) \geqslant (K-k)\theta_* \geqslant (K-k)\theta$ because it remains at least $(K-k)$ groups not represented in $S^{(k)}$ and all these groups have a proportion larger than θ_*. We also have thanks to Lemma 3.C.8 and Lemma 3.C.9, $\mathbb{P}_\nu(\mathcal{Y}) \geqslant 1-\delta/4$ and $\mathbb{P}_\nu(\cap_{s\in[r]}\mathcal{Z}^c_{u,s}) \geqslant 1/2$.

Conditionally on $S_u = S^{(k)}$, and on the estimated centers of the representatives in $S^{(k)}$, the event $\{a_u \text{ is added to } S\}$ are independent and of probability larger than $(1-\delta/4)(K-k)\theta/2$. Then, M_k is stochastically dominated by a geometric random variable of parameter $(K-k)\theta/4$. Finally, $\mathbb{E}_\nu[M_k\mathbb{1}_{\mathcal{Y}}] \leqslant \frac{4}{(K-k)\theta}$ and

$$\mathbb{E}_\nu[(M-1)\mathbb{1}_{\mathcal{Y}}] = \sum_{k=1}^{K-1}\mathbb{E}_\nu[M_k\mathbb{1}_{\mathcal{Y}}] \leqslant \sum_{k=1}^{K-1}\frac{4}{(K-k)\theta} = \frac{4}{\theta}\sum_{k=1}^{K-1}\frac{1}{k} \leqslant \frac{4}{\theta}(1+\log(K)) \ . \tag{3.42}$$

Now, Equation (3.41) becomes

$$\sum_{s=s_0+1}^{r} 2n_s \mathbb{E}_\nu \left[H_{s,M} \mathbb{1}_{\mathcal{Y}} \right] \leqslant \sum_{s=s_0+1}^{r} 2n_s \left[\frac{4}{\theta}(1+\log(K)) \exp(-2^{s-4}) \right] \ .$$

We bound the previous expression, using the same computation as Lemma 3.C.14, and we state the bound as a lemma proved later.

Lemma 3.C.15. *Using the explicit expression of r, U, n_s, $n_{\max}$ and s_0 from Remark 3.C.4, we have*

$$\sum_{s=s_0}^{r} n_s \left[\frac{4}{\theta} \cdot \frac{1+\log(K)}{\exp(2^{s-4})} \right] \leqslant c \frac{\sigma^2}{\Delta^2} \frac{\log(K)}{\theta} \left[\log(K) + \sqrt{d} \right]$$

where c is a universal constant.

We come back to Equation (3.39), using Equations (3.41),(3.42), we have

$$\begin{aligned} \mathbb{E}_\nu[\mathcal{T}_{\mathrm{SRI}}] \leqslant & T_{\max}\delta + 2K \left(n_{\max} + \sum_{s=s_0+1}^{r} n_s \right) \\ & + \frac{8n_{s_0}}{\theta}(1+\log(K)) + \sum_{s=s_0+1}^{r} 2n_s \left[\frac{4}{\theta}(1+\log(K)) \exp(-2^{s-4}) \right] \ . \end{aligned}$$

Now, if we define

$$A' = \frac{\sigma^2}{\Delta^2} \left[\frac{\log(K)}{\theta} \log(1/(\theta\delta)) + \frac{\log(K)}{\theta}\sqrt{d} + \sqrt{dK \frac{\log(K)}{\theta} \log(K/\delta)} \right]$$

Lemma 3.C.13 and Lemma 3.C.15 implies that we can choose c large enough (and universal) such that

$$\sum_{s=s_0+1}^{r} 2n_s \left[\frac{4}{\theta}(1+\log(K)) \exp(-2^{s-4}) \right] + 2K \left(n_{\max} + \sum_{s=s_0+1}^{r} n_s \right) \leqslant 2K + cA' \ .$$

By definition of n_{s_0}, we can also see that

$$\frac{8n_{s_0}}{\theta}(1+\log(K)) \leqslant 16 \frac{\log(K)}{\theta} + cA' \ .$$

Finally, it follows from (3.38) and $\delta \log(1/\delta) \leq 1$ that

$$T_{\max}\delta \leq c' \frac{\log(K)}{\theta} + cA' \ .$$

In summary, we have the desired bound in expectation,

$$\mathbb{E}_\nu[\mathcal{T}_{\mathrm{SRI}}] \leqslant c' \frac{\log(K)}{\theta} + c \frac{\sigma^2}{\Delta^2} \left[\frac{\log(K)}{\theta} \log(1/(\theta\delta)) + \frac{\log(K)}{\theta}\sqrt{d} + \frac{\log^2(K)}{\theta} \right] \ .$$

3.C.2 Proofs of technical lemmas

Proof of Lemma 3.C.8. We want to prove that $\mathbb{P}_\nu(\mathcal{Y}^c) \leqslant \delta/4$, where $\mathcal{Y}^c$ is equal to

$$\mathcal{Y}^c = \left\{ \exists a \in \hat{S}, \exists k \in [K] \setminus \{k(a)\},\ \left| \left\langle \frac{\epsilon_a + \epsilon'_a}{\sqrt{2}}, \frac{\mu_a - \mu(k)}{\|\mu_a - \mu(k)\|} \right\rangle \right| > \frac{1}{16} \sqrt{\frac{\Delta^2}{2\sigma^2}} \sqrt{n_{\max}} \right\}$$
$$\bigcup \left\{ \exists a \in \hat{S}, |\langle \epsilon_a, \epsilon'_a \rangle| > \frac{1}{16} \frac{\Delta^2}{\sigma^2} n_{\max} \right\}$$
$$\bigcup \left\{ \exists a \in \hat{S},\ \|\epsilon'_a\|^2 \vee \|\epsilon_a\|^2 - \mathbb{E}[\|\epsilon_a\|^2] > c_{\mathrm{hw}} \log(12K/\delta) \vee \sqrt{c_{\mathrm{hw}} d \log(12K/\delta)} \right\} \ .$$

From the definition of n_s and $n_{\max}$ as given in (3.11), (3.12), it holds that $n_{\max} \geqslant n_r$, with

$$n_s \geqslant c_1 \frac{\sigma^2}{\Delta^2} \left(2^s + \log(12K)\right) \vee c_2 \frac{\sigma^2}{\Delta^2} \sqrt{d(2^s + \log(6))} \ .$$

With the definition of r in (3.10), we have also $2^r \geqslant \log(4U/\delta)$ and $U \geqslant 8K \log(K/\delta)$. We prove that, for $c_1 = 32^2 \vee 8c_{\mathrm{hw}}$ and $c_2 = 16\sqrt{c_{\mathrm{hw}}/2} \vee c_3$, then $n_{\max}$ is large enough to ensure Lemma 3.C.8 (the value of c_3 will be useful later).

We start with a union bound on $\hat{S} \times [K]$, where $|\hat{S}| \leqslant K$. It holds that

$$\mathbb{P}_\nu \left(\exists a \in \hat{S}, \exists k \in [K] \setminus \{k(a)\},\ \left| \left\langle \frac{\epsilon_a + \epsilon'_a}{\sqrt{2}}, \frac{\mu_a - \mu(k)}{\|\mu_a - \mu(k)\|} \right\rangle \right| > \sqrt{\frac{n_{\max}\Delta^2}{2 \cdot 16^2 \sigma^2}} \right)$$
$$\leqslant K^2 \mathbb{P}_\nu \left(\left| \left\langle \frac{\epsilon_a + \epsilon'_a}{\sqrt{2}}, \frac{\mu_a - \mu(k)}{\|\mu_a - \mu(k)\|} \right\rangle \right| > \sqrt{\frac{n_{\max}\Delta^2}{2 \cdot 16^2 \sigma^2}} \right) \ .$$

From the assumption on the noise (Assumption 3.2.1), it is easy to see that $\left\langle \frac{\epsilon_a + \epsilon'_a}{\sqrt{2}}, \frac{\mu_a - \mu(k)}{\|\mu_a - \mu(k)\|} \right\rangle$ is 1-subGaussian, proceeding as in the same proof of Corollary 3.E.4. With the standard concentration inequality Lemma 3.E.1 for subGaussian variables, we have

$$\mathbb{P}_\nu \left(\left| \left\langle \frac{\epsilon_a + \epsilon'_a}{\sqrt{2}}, \frac{\mu_a - \mu(k)}{\|\mu_a - \mu(k)\|} \right\rangle \right| > \sqrt{\frac{n_{\max}\Delta^2}{2 \cdot 16^2 \sigma^2}} \right) \leqslant 2 \exp\left(- \frac{n_{\max}\Delta^2}{(32\sigma)^2} \right) \ ,$$

Now, $c_1 \geqslant 32^2$ and $2^r \geqslant \log(4K/\delta)$ so that $n_{\max} \geqslant c_1 \frac{\sigma^2}{\Delta^2}(2^r + \log(12K)) \geqslant 32^2 \frac{\sigma^2}{\Delta^2} \log\left(\frac{48K^2}{\delta}\right)$. Finally, with the union bound above, we have

$$\mathbb{P}_\nu \left(\exists a \in \hat{S}, \exists k \in [K] \setminus \{k(a)\},\ \left| \left\langle \frac{\epsilon_a + \epsilon'_a}{\sqrt{2}}, \frac{\mu_a - \mu(k)}{\|\mu_a - \mu(k)\|} \right\rangle \right| > \sqrt{\frac{n_{\max}\Delta^2}{2 \cdot 16^2 \sigma^2}} \right) \leqslant \frac{\delta}{24} \ . \tag{3.43}$$

Now, as proved in Corollary 3.E.4, Hanson Wright inequality imply a bound for $\langle \epsilon_a, \epsilon'_a \rangle$ and

then with the constant c_{hw} from Lemma 3.E.3, we have:

$$\mathbb{P}_\nu\left(\exists a\in\hat{S},,\ |\langle\epsilon_a,\epsilon'_a\rangle|>\frac{\Delta^2 n_{\max}}{16\sigma^2}\right)\leqslant 2K\exp\left(-\frac{2}{c_{\mathrm{hw}}}\left(\frac{\Delta^2 n_{\max}}{16\sigma^2}\wedge\frac{\Delta^4 n_{\max}^2}{16^2\sigma^4 d}\right)\right)\leqslant\frac{\delta}{24}\ , \tag{3.44}$$

because the definition of c_1, c_2 and r, implies that

$$n_{\max}\geqslant 8c_{\mathrm{hw}}\frac{\sigma^2}{\Delta^2}\log\left(\frac{48K}{\delta}\right)\vee 16\sqrt{\frac{c_{\mathrm{hw}}}{2}}\frac{\sigma^2}{\Delta^2}\sqrt{d\log\left(\frac{48K}{\delta}\right)}\ .$$

Finally, a direct application of Hanson-Wright inequality (Lemma 3.E.3) ensures that

$$\begin{aligned}&\mathbb{P}_\nu\left(\exists a\in\hat{S},\ \|\epsilon'_a\|^2\vee\|\epsilon_a\|^2-\mathbb{E}[\|\epsilon_a\|^2]\geq c_{\mathrm{hw}}\log(12K/\delta)\vee\sqrt{c_{\mathrm{hw}}d\log(12K/\delta)}\right)\\&\leqslant 2K\exp(-\log(12K/\delta))\leqslant\frac{\delta}{6}\ ,\end{aligned} \tag{3.45}$$

This concludes the proof of Lemma 3.C.8, using a union bound and inequalities (3.43) to (3.45). □

Proof of Lemma 3.C.9 . We recall that in this lemma, ν is an environment with minimal gap Δ_* and balancedness θ_*, and ν is not necessary in $\mathcal{E}(\Delta,\theta,\sigma,n,K,d)$. Let $u\geqslant 1$ and $s\in[s_0,r]$.

The first point of the lemma is a direct consequence of the expression of $\mathcal{Z}_{u,s}$ and the mutual independence of the series of empirical means $(\bar{\mu}_{u,s},\bar{\mu}'_{u,s})$.

Second point of Lemma 3.C.9. We assume that a_u is a good arm rejected by the test (u,s), which by definition of the test statistic means that for all $a\in S_u$, then $|\mu_{a_u}-\mu_a|\geqslant\Delta$, while the test statistic ϕ_s^u is equal to 1. It implies that there exists $a\in S_u$ such that

$$\left\langle\bar{\mu}_{u,s}-\hat{\mu}_a,\bar{\mu}'_{u,s}-\hat{\mu}'_a\right\rangle\leqslant\frac{\Delta^2}{2}\ .$$

From the decomposition of (3.28), and conditionally on the event $\mathcal{Y}$ (3.29), we have:

$$\begin{aligned}\frac{\Delta^2}{2}\geqslant&\|\mu_{a_u}-\mu_a\|^2-\frac{\Delta}{16}\|\mu_{a_u}-\mu_a\|-\frac{\Delta^2}{16}\\&-\frac{\sigma^2}{\sqrt{n_{\max}n_s}}\langle\epsilon_{u,s},\rho'_a\rangle\|\epsilon'_a\|-\frac{\sigma^2}{\sqrt{n_{\max}n_s}}\left\langle\epsilon'_{u,s},\rho_a\right\rangle\|\epsilon_a\| &(3.46)\\&+\frac{\sqrt{2}\sigma}{\sqrt{n_s}}\left\langle\frac{\epsilon_{u,s}+\epsilon'_{u,s}}{\sqrt{2}},\mu_{a_u}-\mu_a\right\rangle+\frac{\sigma^2}{n_s}\left\langle\epsilon_{u,s},\epsilon'_{u,s}\right\rangle\ . &(3.47)\end{aligned}$$

It holds that, on $\mathcal{Y}$, $\|\epsilon_a\|^2\leqslant\mathbb{E}[\|\epsilon_a\|^2]+c_{\mathrm{hw}}l\vee\sqrt{c_{\mathrm{hw}}dl}$ with $l:=\log\left(\frac{12K}{\sigma}\right)$. Moreover, $\mathbb{E}[\|\epsilon_a\|^2]\leqslant 4d$ is a direct consequence of the subGaussian assumption (see Rigollet and Hütter, 2023, Lemma 1.4). We have

$$(3.46)\geqslant-\frac{\sigma^2}{\sqrt{n_{\max}n_s}}\left[\left|\left\langle\epsilon'_{u,s},\rho_a\right\rangle\right|+|\langle\epsilon_{u,s},\rho'_a\rangle|\right]\sqrt{4d+c_{\mathrm{hw}}l\vee\sqrt{c_{\mathrm{hw}}dl}}\ .$$

Also,

$$(3.47) \geqslant -\sqrt{\frac{2\sigma^2\|\mu_{a_u}-\mu_a\|^2}{n_s}}\left|\left\langle \frac{\epsilon_{u,s}+\epsilon'_{u,s}}{\sqrt{2}}, \frac{\mu_{a_u}-\mu_a}{\|\mu_{a_u}-\mu_a\|}\right\rangle\right| - \frac{\sigma^2}{n_s}\left|\left\langle \epsilon_{u,s}, \epsilon'_{u,s}\right\rangle\right| \ .$$

We recall that $\Delta \leqslant \|\mu_{a_u}-\mu_a\|$, because a_u is a good arm. Hence, we have $\|\mu_{a_u}-\mu_a\|^2 - \frac{\Delta}{16}\|\mu_{a_u}-\mu_a\| - \frac{\Delta^2}{16} - \frac{\Delta^2}{2} \geqslant \frac{3}{8}\|\mu_{a_u}-\mu_a\|^2 \geq \frac{3}{8}\Delta^2$.

From there, we state that if a_u is good and $\phi^u_s = 1$ then there exists $a \in \hat{S}$ such that at least one of the these three inequalities holds:

$$\begin{aligned}
&|\langle \epsilon_{u,s}, \rho'_a\rangle| + \left|\left\langle \epsilon'_{u,s}, \rho_a\right\rangle\right| \geqslant \frac{1}{4}\frac{\Delta^2}{\sigma^2}\sqrt{\frac{n_s n_{\max}}{4d+\sqrt{c_{\mathrm{hw}}dl}\vee c_{\mathrm{hw}}l}} \ , \\
&\left|\left\langle \epsilon_{u,s}, \epsilon'_{u,s}\right\rangle\right| \geqslant \frac{1}{16}\frac{\Delta^2}{\sigma^2}n_s \ , \\
&\left|\left\langle \frac{\epsilon_{u,s}+\epsilon'_{u,s}}{\sqrt{2}}, \frac{\mu_{a_u}-\mu_a}{\|\mu_{a_u}-\mu_a\|}\right\rangle\right| \geqslant \frac{1}{16}\frac{\Delta}{\sigma}\sqrt{\frac{n_s}{2}} \ .
\end{aligned}$$

By definition, $\mathcal{Z}_{u,s}$ is the event where one of these three inequalities hold for some $a \in \hat{S}$, so the inclusion $\mathcal{Y} \cap \{a_u \text{ is good and } \phi^u_s = 1\} \subset \mathcal{Z}_{u,s}$ is proved.

Third point of Lemma 3.C.9

We now assume that a_u is bad, but the s-th test accept a_u, i.e., $\phi^u_s = 0$. As a_u is bad, there exists $a \in S_u$ such that $\|\mu_{a_u}-\mu_a\| \leqslant \Delta/4$. As $\phi^u_s = 0$, for this specific arm a, we have: $\left\langle \bar{\mu}_{u,s}-\hat{\mu}_a, \bar{\mu}'_{u,s}-\hat{\mu}'_a\right\rangle > \frac{\Delta^2}{2}$.

Assume that $0 < \|\mu_{a_u}-\mu_a\| \leqslant \Delta/4$, with the same computation as in the first case, we have

$$\begin{aligned}
\frac{\Delta^2}{2} \leqslant& \|\mu_{a_u}-\mu_a\|^2 + \frac{\Delta}{16}\|\mu_{a_u}-\mu_a\| + \frac{\Delta^2}{16} \\
&+ \frac{\sigma^2}{\sqrt{n_{\max}n_s}}\sqrt{4d + c_{\mathrm{hw}}l \vee \sqrt{c_{\mathrm{hw}}dl}}\left(|\langle \epsilon_{u,s}, \rho'_a\rangle| + \left|\left\langle \epsilon'_{u,s}, \rho_a\right\rangle\right|\right) \\
&+ \sqrt{\frac{2\sigma^2\|\mu_{a_u}-\mu_a\|^2}{n_s}}\left|\left\langle \frac{\epsilon_{u,s}+\epsilon'_{u,s}}{\sqrt{2}}, \frac{\mu_{a_u}-\mu_a}{\|\mu_{a_u}-\mu_a\|}\right\rangle\right| + \frac{\sigma^2}{n_s}\left|\left\langle \epsilon_{u,s}, \epsilon'_{u,s}\right\rangle\right| \ .
\end{aligned}$$

In the last line, we upper-bound $\|\mu_{a_u}-\mu_a\|$ by $\Delta/4$. Now, consider the constant terms, $\frac{\Delta^2}{2} - \|\mu_{a_u}-\mu_a\|^2 - \frac{\Delta}{16}\|\mu_{a_u}-\mu_a\| - \frac{\Delta^2}{16} \geqslant \frac{\Delta^2}{2} - \frac{\Delta^2}{16} - \frac{\Delta^2}{64} - \frac{\Delta^2}{16} \geqslant \Delta^2\left(\frac{1}{4}+\frac{1}{16}+\frac{1}{4\cdot 16}\right)$. As above, we deduce that at least one of the three inequalities defining $\mathcal{Z}_{u,s}$ holds. If $\|\mu_{a_u}-\mu_a\| = 0$, there are simply fewer terms in the equality. This proves the inclusion $\mathcal{Y} \cap \{a_u \text{ is bad and } \phi^u_s = 0\} \subset \mathcal{Z}_{u,s}$.

Probability of $\mathcal{Z}_{u,s}$

We prove now that the probability of $\mathcal{Z}_{u,s}$ decrease exponentially fast with s. Fix $s_0 \leqslant s \leqslant r$.

We recall the expression of n_s, and $n_{\max}$

$$n_s \geqslant c_1\frac{\sigma^2}{\Delta^2}\left(2^s + \log(12K)\right) \vee c_2\frac{\sigma^2}{\Delta^2}\sqrt{d(2^s+\log(6))} \ ,$$

where $c_1 = 32^2 \vee 8c_{\mathrm{hw}}$, $c_2 = 16\sqrt{c_{\mathrm{hw}}/2} \vee 32\sqrt{2}$, $c_3 = 32\sqrt{2}$, and $n_{\max} \geqslant n_r \vee c_3 \frac{\sigma^2}{\Delta^2}\sqrt{d}\log(2K)$.

Let $k \in [K]$ such that $\mu_{a_u} \neq \mu(k)$, as a consequence of Assumption 3.2.1, the one-dimensional variable $\left\langle \frac{\epsilon_{u,s}+\epsilon'_{u,s}}{\sqrt{2}}, \frac{\mu_{a_u}-\mu(k)}{\|\mu_{a_u}-\mu(k)\|} \right\rangle$ is 1-subGaussian. With standard concentration of subGaussian variables (Lemma 3.E.1), we have

$$\mathbb{P}_\nu\left(\left|\left\langle \frac{\epsilon_{u,s}+\epsilon'_{u,s}}{\sqrt{2}}, \frac{\mu_{a_u}-\mu(k)}{\|\mu_{a_u}-\mu(k)\|}\right\rangle\right| \geqslant \frac{1}{16}\frac{\Delta}{\sigma}\sqrt{\frac{n_s}{2}}\right) \leqslant 2\exp\left(-\frac{\Delta^2}{\sigma^2}\frac{n_s}{32^2}\right) \leqslant \frac{1}{3K}\exp(-2^s) \ ,$$

because $n_s \geqslant 32^2\frac{\sigma^2}{\Delta^2}(2^s + \log(6K))$.

Now, with a union bound over $k \in [K]$, it holds that

$$\mathbb{P}\left(\exists k \in [K]\setminus\{k(a_u)\}\,;\left|\left\langle \frac{\epsilon_{u,s}+\epsilon'_{u,s}}{\sqrt{2}}, \frac{\mu_{a_u}-\mu(k)}{\|\mu_{a_u}-\mu(k)\|}\right\rangle\right| \geqslant \frac{1}{16}\frac{\Delta}{\sigma}\sqrt{\frac{n_s}{2}}\right) \leqslant \frac{1}{3}\exp(-2^s) \ , \tag{3.48}$$

Then, $\left\langle \epsilon_{u,s}, \epsilon'_{u,s}\right\rangle$ is the inner product of two independent vectors, for which the assumptions from Corollary 3.E.4 holds. We use this corollary of Hanson-Wright inequality, and obtain

$$\mathbb{P}_\nu\left(\left|\left\langle \epsilon_{u,s}, \epsilon'_{u,s}\right\rangle\right| \geqslant \frac{\Delta^2}{\sigma^2}\frac{n_s}{16}\right) \leqslant 2\exp\left(-\frac{2}{c_{\mathrm{hw}}}\left(\frac{\Delta^2}{\sigma^2}\frac{n_s}{16} \wedge \frac{1}{d}\frac{\Delta^4}{\sigma^4}\frac{n_s^2}{16^2}\right)\right) \leqslant \frac{1}{3}\exp(-2^s) \ , \tag{3.49}$$

where the last inequality follows from the definition of n_s (and c_1, c_2) where $n_s \geqslant 8c_{\mathrm{hw}}\frac{\sigma^2}{\Delta^2}(2^s + \log(6)) \vee 16\sqrt{\frac{c_{\mathrm{hw}}}{2}}\frac{\sigma^2}{\Delta^2}\sqrt{d(2^s+\log(6))}$.

Finally, we want to upper-bound the probability that the cross term between ϵ_a and $\epsilon_{u,s}$ is too large. By conditioning with respect to the random variables $(\rho_a, \rho'_a)_a$, we consider these variables as constants. We start with a union bound and the inequality $a + b \leqslant 2a \vee b$.

$$\begin{aligned}
&\mathbb{P}_\nu\left(\exists a \in \hat{S}\,;\left|\langle \epsilon_{u,s}, \rho'_a\rangle\right| + \left|\left\langle \epsilon'_{u,s}, \rho_a\right\rangle\right| \geqslant \frac{1}{4}\frac{\Delta^2}{\sigma^2}\sqrt{\frac{n_s n_{\max}}{4d + \sqrt{c_{\mathrm{hw}}dl} \vee c_{\mathrm{hw}}l}}\right)\\
\leqslant& 2K\mathbb{P}_\nu\left(|\,\langle \epsilon_{u,s}, \rho\rangle\,| \geqslant \frac{1}{8}\frac{\Delta^2}{\sigma^2}\sqrt{\frac{n_s n_{\max}}{4d + \sqrt{c_{\mathrm{hw}}dl} \vee c_{\mathrm{hw}}l}}\right) \ ,
\end{aligned}$$

with ρ of norm 1. Then $\langle \epsilon_{u,s}, \rho\rangle$ is a 1-dimensional subGaussian random variable. We use therefore the concentration inequality in Lemma 3.E.1, and we state that

$$\begin{aligned}
&\mathbb{P}_\nu\left(\exists a \in \hat{S}\,;\left|\langle \epsilon_{u,s}, \rho'_a\rangle\right| + \left|\left\langle \epsilon'_{u,s}, \rho_a\right\rangle\right| \geqslant \frac{1}{4}\frac{\Delta^2}{\sigma^2}\sqrt{\frac{n_s n_{\max}}{4d + \sqrt{c_{\mathrm{hw}}dl} \vee c_{\mathrm{hw}}l}}\right)\\
\leqslant& 4K\exp\left(-\frac{1}{2\cdot 8^2}\frac{\Delta^4}{\sigma^4}\frac{n_s n_{\max}}{4d + \sqrt{c_{\mathrm{hw}}dl} \vee c_{\mathrm{hw}}l}\right)\\
\leqslant& 4K\exp\left(-\frac{1}{16^2}\frac{\Delta^4}{\sigma^4}\frac{n_s n_{\max}}{4d \vee \sqrt{c_{\mathrm{hw}}dl} \vee c_{\mathrm{hw}}l}\right) \ ,
\end{aligned}$$

in the last line, we use again the inequality $a + b \leqslant 2a \vee b$.

Now, we need to bound this last expression by $\frac{1}{3}\exp(-2^s)$ by using the definition of n_s and $n_{\max}$ We recall that $l = \log(12K/\delta) \leqslant 2^r$.

Now, with our choice for c_1 and c_2, it holds that $n_s \geqslant 32^2 \frac{\sigma^2}{\delta^2}(2^s + \log(12K))$ and $n_{\max} \geqslant c_{\mathrm{hw}} l \vee \sqrt{c_{\mathrm{hw}} dl}$, so that

$$n_s n_{\max} \geqslant 16^2 \frac{\sigma^4}{\Delta^2}(\sqrt{c_{\mathrm{hw}} dl} \vee c_{\mathrm{hw}} l)(2^s + \log(12K)) \ .$$

We finally use the assumption that $c_2 \geqslant 32\sqrt{2}$ and $c_3 = 32\sqrt{2}$, so that

$$\begin{aligned} n_s &\geqslant 16\sqrt{2}\frac{\sigma^2}{\Delta^2}\sqrt{(4d)(2^s + \log(6))} \\ n_{\max} &\geqslant 16\sqrt{2}\frac{\sigma^2}{\Delta^2}\sqrt{4d}\sqrt{2^s + \log(6)} \\ n_{\max} &\geqslant 16\sqrt{2}\frac{\sigma^2}{\Delta^2}\sqrt{4d}\log(2K) \end{aligned}$$

Then, with the inequality $a \vee b \geqslant (a+b)/2$, we have

$$n_s n_{\max} \geqslant 16^2 \frac{\sigma^4}{\Delta^2}(4d)(2^s + \log(12K)) \ .$$

We combine these lower bound on $n_s n_{\max}$ to deduce that

$$n_s n_{\max} \geqslant 16^2 \frac{\sigma^4}{\Delta^2}(4d \vee \sqrt{c_{\mathrm{hw}} dl} \vee c_{\mathrm{hw}} l)(2^s + \log(12K)) \ .$$

This allows us to conclude that

$$\begin{aligned} &\mathbb{P}_\nu\left(\exists a \in \hat{S}\ ; |\langle \epsilon_{u,s}, \rho'_a\rangle| + \left|\left\langle \epsilon'_{u,s}, \rho_a \right\rangle\right| \geqslant \frac{1}{4}\frac{\Delta^2}{\sigma^2}\sqrt{\frac{n_s n_{\max}}{4d + \sqrt{c_{\mathrm{hw}} dl} \vee c_{\mathrm{hw}} l}}\right) \\ &\leqslant 4K\exp\left(-\frac{1}{16^2}\frac{\Delta^4}{\sigma^4}\frac{n_s n_{\max}}{4d \vee \sqrt{c_{\mathrm{hw}} dl} \vee c_{\mathrm{hw}} l}\right) \leqslant \frac{1}{3}\exp(-2^s) \ . \end{aligned} \tag{3.50}$$

We finish the proof with a union bound, gathering the inequalities (3.48) to (3.50),

$$\mathbb{P}_\nu(\mathcal{Z}_{u,s}) \leqslant \frac{1}{3}\exp(-2^s) + \frac{1}{3}\exp(-2^s) + \frac{1}{3}\exp(-2^s) = \exp(-2^s) \ .$$

□

Proof of Lemma 3.C.13. Throughout the proofs of Lemmas 3.C.13, 3.C.14, 3.C.15, c is a universal constant changing from one line to another. Also, we use that, by the definition of s_0 and n_s, it turns out, that if $s > s_0$ then

$$n_s \leqslant 2c_1\frac{\sigma^2}{\Delta^2}(2^s + \log(12K)) \vee 2c_2\frac{\sigma^2}{\Delta^2}\sqrt{d(2^s + \log(6))} \ . \tag{3.51}$$

We now bound $K\left(\sum_{s=s_0+1}^{r} n_s + n_{\max}\right)$. Relying on the expression of n_s above, and the sums $\sum_{s=1}^{r} 2^s \leqslant 2^{r+1}$ and $\sum_{s=1}^{r} \sqrt{2}^s \leqslant \sqrt{2}^{r+1}(1+\sqrt{2})$, we deduce that

$$K\sum_{s=s_0+1}^{r} n_s \leqslant 2c_1\frac{\sigma^2}{\Delta^2}K(2^{r+1}+\log(12K)r) \bigvee 2c_2\frac{\sigma^2}{\Delta^2}K\sqrt{d}\left(\sqrt{2^r}(2+\sqrt{2})+\sqrt{\log(6)}r\right) \ .$$

Now, from the expression of r and U, we have $2^r \leqslant 2\log(4U/\delta) \leqslant c\log(1/(\theta\delta))$. It leads to the bound

$$K\sum_{s=s_0+1}^{r} n_s \leqslant c\frac{\sigma^2}{\Delta^2}\left[K\log(1/(\theta\delta)) + K\log(K)\log\log(1/(\theta\delta)) + K\sqrt{d\log(1/(\theta\delta))}\right] \ .$$

As $1/\theta \geqslant K$, we have $K\log\left(\frac{1}{\theta}\right) \leqslant \frac{1}{\theta}\log(K)$, so that we can bound the term above by

$$K\sum_{s=s_0+1}^{r} n_s \leqslant c\frac{\sigma^2}{\Delta^2}\left[\frac{1}{\theta}\log(K/\delta) + \sqrt{dK\frac{1}{\theta}\log(K/\delta)}\right] \ .$$

We also compute $n_{\max}$, we have $Kn_{\max} \leqslant K + c\frac{\sigma^2}{\Delta^2}\left[Kn_r + K\log(K)\sqrt{d}\right]$, where Kn_r is upper bounded by the same bound as $K\sum_{s=s_0+1}^{r} n_s$.

Finally, it implies that

$$K\sum_{s=s_0+1}^{r} n_s + Kn_{\max} \leqslant K + c\frac{\sigma^2}{\Delta^2}\left[\frac{\log(K)}{\theta}\log(K/\delta) + \sqrt{dK\frac{\log(K)}{\theta}\log(K/\delta)}\right] \ .$$

The second inequality in Lemma 3.C.13 is clear, and the lemma is proved. □

Proof of Lemma 3.C.14. With the bound on n_s for $s > s_0$ from Equation (3.51), we simplify the terms in 2^s and obtain,

$$\begin{aligned}
\sum_{s=s_0+1}^{r} \frac{U}{2^{s-4}}n_s \leqslant& 2c_1\frac{16U\sigma^2}{\Delta^2}\sum_{s=s_0+1}^{r}\left(1+\frac{1}{2^s}\log(12K)\right) \bigvee 2c_2\frac{16U\sigma^2}{\Delta^2}\sum_{s=s_0+1}^{r}\left(\frac{\sqrt{d}}{\sqrt{2}^s}+\frac{1}{2^s}\sqrt{d\log(6)}\right) \\
\leqslant& 2c_1\frac{16U\sigma^2}{\Delta^2}\left(r+2\log(12K)\right) \bigvee 2c_2\frac{16U\sigma^2}{\Delta^2}\left((2+\sqrt{2})\sqrt{d}+2\sqrt{\log(6)}\sqrt{d}\right) \\
\leqslant& c\frac{\sigma^2}{\Delta^2}U\left[\log\log\left(\frac{1}{\theta\delta}\right)+\log(K)+\sqrt{d}\right] \ ,
\end{aligned}$$

because $\sum_{s\geqslant 1} 1/2^s \leqslant 2$ and $\sum_{s\geqslant 1} 1/\sqrt{2}^s = 2+\sqrt{2}$. We also use in the last inequality that $\log(U/\delta) \leqslant 2\log(8/\theta\delta)$, so that $r \leqslant \log(2\log(8/\theta\delta)) \leqslant c\log\log(1/\theta\delta)$.

From the previous bound, we conclude that

$$\sum_{s=s_0+1}^{r} \frac{U}{2^{s-4}} n_s \leqslant c\frac{\sigma^2}{\Delta^2} U \left[\log\log\left(\frac{1}{\theta\delta}\right) + \log(K) + \sqrt{d}\right] + c\frac{\sigma^2}{\Delta^2}\sqrt{d\log(K)KU} \ .$$

Moreover, we have by definition of U (3.10), $U \geqslant K\log(K)$, and then

$$\frac{\sigma^2}{\Delta^2}\sqrt{d\log(K)KU} \leqslant \frac{\sigma^2}{\Delta^2} U\sqrt{d} \ .$$

Finally, using the expression of U (3.10), we have

$$\sum_{s=s_0+1}^{r} \frac{U}{2^{s-4}} n_s \leqslant c\frac{\sigma^2}{\Delta^2}\frac{1}{\theta}\log(K/\delta)\left[\log(K) + \sqrt{d} + \log\log\left(\frac{1}{\theta\delta}\right)\right] \ .$$

□

Proof of Lemma 3.C.15. With the same computation as in Lemma 3.C.14, we obtain

$$\begin{aligned}\sum_{s=s_0+1}^{r} n_s \left[\frac{4}{\theta}(1+\log(K))\exp(-2^{s-4})\right] &\leqslant c\frac{\sigma^2}{\Delta^2}\frac{\log(K)}{\theta}\left[\log(K)+\sqrt{d}\right] + c\frac{\sigma^2}{\Delta^2}\sqrt{dK\frac{\log(K)}{\theta}\log(K)} \\ &\leqslant c\frac{\sigma^2}{\Delta^2}\frac{\log(K)}{\theta}\left[\log(K)+\sqrt{d}\right] \ ,\end{aligned}$$

where we use $K \leqslant 1/\theta$ in the last inequality. □

3.C.3 Proof of Lemma 3.C.3

In this section, we want to prove that the subroutine ADC outputs the exact partition with probability larger than $1-\delta$, for environments in $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$. Let ν be an environment with a minimal gap smaller than Δ, following Assumptions 3.2.1 and 3.2.3. We highlight that the algorithm ADC uses Δ, σ, n, K and d as parameters but not θ. Let $S = \{b_1, \dots, b_K\}$ be a set of K arms containing one representative by group. The objective is to find the groups $\mathcal{C}_1^*, \dots, \mathcal{C}_K^*$ up to permutation. Without loss of generality, we fix the label of the groups so that $\mathcal{C}^*{}_k = \{a \in [n], \mu_a = \mu_{b_k}\}$. We denote by $k(a)$ as the corresponding label of any arm a ($a \in \mathcal{C}^*{}_{k(a)}$). With this convention, making an error of clustering is equivalent of making an error of labeling.

We denote $\hat{\mathcal{C}}$ for the output of the ADC routine. The algorithm labels the arms in S so that $b_k \in \hat{\mathcal{C}}_k$ for $k \in [K]$ (see Line 8). Then, it labels each arm $a \in [n] \setminus S$ by $\hat{k}(a)$ defined (Equation (3.15)) by

$$\hat{k}(a) \in \operatorname*{argmin}_{j=1,\dots,K} \left\langle \hat{\mu}_a - \hat{\mu}(j), \hat{\mu}'_a - \hat{\mu}'(j) \right\rangle \ .$$

We have $\{\hat{\mathcal{C}} \sim \mathcal{C}^*\} = \{\exists a \in [n] \setminus S \, ; \hat{k}(a) \neq k(a)\}$.

Consider $j \in [K]$ a group and $a \in [n]$ an arm. As explained in the introduction, the statistic $\hat{d}^2_{a,j} := \left\langle \hat{\mu}_a - \hat{\mu}(j), \hat{\mu}'_a - \hat{\mu}'(j) \right\rangle$ is a natural non-biased estimator of $\|\mu_a - \mu(j)\|^2$ where $\mu(j) = \mu_{b_j}$ is

the center of ${\mathcal{C}^*}_j$. In the expression of $\hat{k}(a)$, $\hat{\mu}(j)$ [resp. $\hat{\mu}'(j)$] is the empirical mean of representative b_j computed with $J = \left\lceil c_4 \frac{\sigma^2}{\Delta^2} L \vee c_5 \frac{\sigma^2}{\Delta^2} \sqrt{dL \frac{n}{K}} \right\rceil$ samples —see Equation (3.14) and Line 7— and $L = \log(6nK/\delta)$. The random variable $\hat{\mu}_a$ [resp. $\hat{\mu}'_a$] is the empirical mean of the arm a computed with $I = \left\lceil c_4 \frac{\sigma^2}{\Delta^2} L \vee c_5 \frac{\sigma^2}{\Delta^2} \sqrt{dL \frac{K}{n}} \right\rceil$ samples — see Line 11. We emphasize that in high dimension, $J \asymp nI/K$ is much larger than I. We want to bound the probability of misclassification for a single arm in $[n] \setminus S$. Let $a \in [n] \setminus S$ such that a belongs to the group ${\mathcal{C}^*}_{k(a)}$. The misclassification probability for the arm a using the classifier $\hat{k}(a)$ of Equation (3.15) is

$$\begin{aligned}\mathbb{P}_\nu(\hat{k}(a) \neq k(a)) =& \mathbb{P}_\nu\Big(\exists j = 1, \dots, K \ , j \neq k(a);\ \hat{d}^2_{a,j} < \hat{d}^2_{a,k(a)}\Big) \\ &\leqslant \sum_{j \neq k(a)}^{K} \mathbb{P}_\nu\Big(\hat{d}^2_{a,j} < \hat{d}^2_{a,k(a)}\Big) \ . \end{aligned} \tag{3.52}$$

We used here a first union bound over $j \in [1; K] \setminus k(a)$, and now, we upper-bound each term on the sum.

Lemma 3.C.16. *For all $a \in [n] \setminus S$, and $j \in [K]$, if $\mu(j) \neq \mu_a$ then*

$$\mathbb{P}_\nu\Big(\hat{d}^2_{a,j} < \hat{d}^2_{a,k(a)}\Big) \leqslant \frac{\delta}{(K-1)(n-K)} \ .$$

This lemma easily leads to the desired result (Lemma 3.C.3) by a union bound on $a \in [n] \setminus S$. With Equation (3.52) and Lemma 3.C.16, we have indeed

$$\begin{aligned}\mathbb{P}_\nu\left(\hat{\mathcal{C}} \not\asymp \mathcal{C}^*\right) &= \mathbb{P}_\nu\left(\exists a \in [n] \setminus [S] \, ; \hat{k}(a) \neq k(a)\right) \\ &\leqslant \sum_{a \in [n] \setminus S} \sum_{j \in [K] \setminus \{k(a)\}} \mathbb{P}_\nu\Big(\hat{d}^2_{a,j} < \hat{d}^2_{a,k(a)}\Big) \leqslant \delta \ . \end{aligned}$$

Moreover, the budget $\mathcal{T}_{\mathrm{ADC}}$ used to compute ADC is deterministic and equal to $2(n-K)I + 2KJ$ with the notation of the algorithm which leads to the second part of the lemma directly.

We have indeed the (deterministic) bound on the budget of ADC

$$\mathcal{T}_{\mathrm{ADC}} = 2(n-K)I + 2KJ \leqslant 2n + 2c_4 \frac{\sigma^2}{\Delta^2} nL \vee 4c_5 \frac{\sigma^2}{\Delta^2} \sqrt{dnKL} \ .$$

It remains now to prove the auxiliary lemma.

Proof of Lemma 3.C.16. Without loss of generality, we assume that $\mu_a = \mu(1)$ and consider $j = 2$. We write

$$\hat{\mu}_a = \mu_a + \frac{\sigma}{\sqrt{I}} \varepsilon_a = \mu(1) + \frac{\sigma}{\sqrt{I}} \varepsilon_a \ ,$$

where $\varepsilon_a := \frac{\sqrt{I}}{\sigma}(\hat{\mu}_a - \mu_a)$. We define in the same way $\varepsilon(1) := \frac{\sqrt{J}}{\sigma}(\hat{\mu}(1) - \mu(1))$ and also $\varepsilon(2), \varepsilon'_a$, $\varepsilon'(1)$ and $\varepsilon'(2)$.

From direct computation, reorganizing the terms, we write the event $\{\hat{d}^2_{a,2} < \hat{d}^2_{a,1}\}$ as

$$\begin{aligned}&\left\langle \hat{\mu}_a - \hat{\mu}(2), \hat{\mu}'_a - \hat{\mu}'(2)\right\rangle < \left\langle \hat{\mu}_a - \hat{\mu}(1), \hat{\mu}'_a - \hat{\mu}'(1)\right\rangle \Leftrightarrow \\ &\frac{\sqrt{2}\sigma\|\mu(1)-\mu(2)\|}{\sqrt{I}}A + \frac{\sqrt{2}\sigma\|\mu(1)-\mu(2)\|}{\sqrt{J}}B + \frac{\sqrt{2}\sigma^2}{\sqrt{IJ}}(C+D) + \frac{\sigma^2}{J}(E+F) > \|\mu(1)-\mu(2)\|^2\end{aligned} \tag{3.53}$$

where

$$\begin{aligned}A &:= -\left\langle \frac{\mu(1)-\mu(2)}{\|\mu(1)-\mu(2)\|}, \frac{\varepsilon'_a + \varepsilon_a}{\sqrt{2}}\right\rangle; \qquad C := -\left\langle \varepsilon_a, \frac{\varepsilon'(1)-\varepsilon'(2)}{\sqrt{2}}\right\rangle; \quad E := -\left\langle \varepsilon(2), \varepsilon'(2)\right\rangle; \\ B &:= -\left\langle \frac{\mu(2)-\mu(1)}{\|\mu(2)-\mu(1)\|}, \frac{\varepsilon'(2)+\varepsilon(2)}{\sqrt{2}}\right\rangle; \quad D := -\left\langle \varepsilon'_a, \frac{\varepsilon(1)-\varepsilon(2)}{\sqrt{2}}\right\rangle; \qquad F := -\left\langle \varepsilon'(1), \varepsilon(1)\right\rangle \ .\end{aligned}$$

Let us control the variation of each of these terms.

First, by Assumption 3.2.1, as in the proofs of Section 3.C.1, A and B are subGaussian. With the concentration inequality (Lemma 3.E.1) for subGaussian (real) variables, we have

$$\mathbb{P}_\nu(A > \sqrt{2L}) \leqslant \exp(-L) \qquad \text{and} \qquad \mathbb{P}(B > \sqrt{2L}) \leqslant \exp(-L) \ .$$

For the other terms, we use Hanson-Wright inequality (Corollary 3.E.4) with c_{hw} the universal constant from the lemma. The scalar products C, D, E and F verifies all the assumptions for Corollary 3.E.4, and for instance,

$$\mathbb{P}_\nu\left(C > \frac{c_{\mathrm{hw}}L}{2} \vee \sqrt{c_{\mathrm{hw}}\frac{dL}{2}}\right) \leqslant \exp(-L) \ ,$$

and we have the same bound for D,E and F.

We recall the expression $L = \log\left(\frac{6nK}{\delta}\right)$ (3.14), in particular, $\exp(-L) \leqslant \frac{\delta}{6nK}$.

With a union bound on these 6 errors, it holds that with probability larger than $1-\delta/nK$ we have

$$\begin{aligned}&\frac{\sqrt{2}\sigma\|\mu(1)-\mu(2)\|}{\sqrt{I}}A + \frac{\sqrt{2}\sigma\|\mu(1)-\mu(2)\|}{\sqrt{J}}B + \frac{\sqrt{2}\sigma^2}{\sqrt{IJ}}(C+D) + \frac{\sigma^2}{J}(E+F) \\ \leqslant &\frac{\sqrt{2}\sigma\|\mu(1)-\mu(2)\|}{\sqrt{I}}\sqrt{2L} + \frac{\sqrt{2}\sigma\|\mu(1)-\mu(2)\|}{\sqrt{J}}\sqrt{2L} + \frac{2\sqrt{2}\sigma^2}{\sqrt{IJ}}\left(\frac{c_{\mathrm{hw}}L}{2} \vee \sqrt{c_{\mathrm{hw}}\frac{dL}{2}}\right) \\ &+ \frac{2\sigma^2}{J}\left(\frac{c_{\mathrm{hw}}L}{2} \vee \sqrt{c_{\mathrm{hw}}\frac{dL}{2}}\right) \ .\end{aligned}$$

The parameters I,J are defined as

$$I = \left\lceil \frac{\sigma^2}{\Delta^2} \left(c_4 L \vee c_5 \sqrt{\frac{K}{n} dL} \right) \right\rceil ; \quad J = \left\lceil \frac{\sigma^2}{\Delta^2} \left(c_4 L \vee c_5 \sqrt{\frac{n}{K} dL} \right) \right\rceil \ ,$$

with c_4 and c_5 two universal constants defined as $c_4 = 8^2 \vee 4\sqrt{2}c_{\rm hw}$ and $c_5 = 8\sqrt{c_{\rm hw}}$ with $c_{\rm hw}$ the universal constant in Hanson-Wright inequality (Lemma 3.E.3). Now, each term in the last sum is smaller than $\|\mu(1) - \mu(2)\|\Delta/4$, or $\Delta^2/4$. As $\nu \in \mathcal{E}(\Delta, \theta, \sigma, n, K, d)$, we have $\Delta_* \geqslant \Delta$ and $\|\mu(1) - \mu(2)\| \geqslant \Delta$. It implies that with probability larger than $1 - \delta/nK$, it holds that

$$\frac{\sqrt{2}\sigma\|\mu(1) - \mu(2)\|}{\sqrt{I}} A + \frac{\sqrt{2}\sigma\|\mu(1) - \mu(2)\|}{\sqrt{J}} B + \frac{\sqrt{2}\sigma^2}{\sqrt{IJ}}(C + D) + \frac{\sigma^2}{J}(E + F) \leqslant \|\mu(1) - \mu(2)\|^2 \ .$$

From there, eq. (3.53) assures that

$$\mathbb{P}_\nu\Big(\Big\langle \hat{\mu}_1 - \hat{\mu}(2), \hat{\mu}'_1 - \hat{\mu}'(2) \Big\rangle < \Big\langle \hat{\mu}_a - \hat{\mu}(k(a)), \hat{\mu}'_a - \hat{\mu}'(k(a)) \Big\rangle\Big) \leqslant \frac{\delta}{nK} \ .$$

□

3.D Analysis of ACB*

In this section, we prove the part of Theorem 3.4.1 pertaining to ACB*. In fact, this result is a straightforward consequence of the following theorem

Theorem 3.D.1. *Let $\delta > 0$. For any environment ν, ACB* Algorithm 5 is δ-correct. There exist positive numerical constants c, c', and c'' such that the following holds.*

$$\mathbb{P}_{\mathrm{ACB}^*,\nu}\Big[\mathcal{T}_{\mathrm{ACB}^*} \leqslant cn + c' \frac{\sigma^2}{\Delta_*^2 \theta_*} L_* \log\left(\frac{L_* K}{\delta}\right) \Big[\log(K) + \sqrt{d} + \log\log(L_*) + \log\log(n/\delta)\Big] + c' \frac{L_*}{\theta_*} \log\left(\frac{L_* K}{\delta}\right) + c'' \frac{\sigma^2}{\Delta_*^2} \Big[n \log(n/\delta) + \sqrt{dnK \log(n/\delta)}\Big] \Big] \geq 1 - \delta \tag{3.54}$$

where

$$L_* := \left\lceil \log_2 \left(\frac{1}{\theta_* K} \left[\left(\frac{\Delta_0^2}{\Delta_*^2} \vee 1 \right) \right] \right) \right\rceil \ . \tag{3.55}$$

We set the numerical constant c_6 in the definition (3.17) of n'_p as

$$c_6 = 2048 \vee 64 c_{\rm hw} \vee 92\sqrt{c_{\rm hw}} \tag{3.56}$$

where $c_{\rm hw}$ is the constant arising in Hanson-Wright inequality —see Lemma 3.E.3.

3.D.1 Analysis of SRI for $\Delta \leq 4\Delta_*$

We explained in Section 3.C.1 how the algorithm $\hat{S} = \mathrm{SRI}(\delta, \Delta, \theta)$ behaves for environments that are not in $\mathcal{E}(\Delta, \theta, \sigma, n, K, d)$. If $\Delta_* \geqslant \Delta$ then the identification of K representatives goes well but, if $\Delta_* \gg \Delta$, the budget will be unnecessarily large. If $\Delta_* \leqslant \Delta$, then the set of representative $\hat{S}$ may contain less than K representative. The following lemma summarizes the properties of SRI.

Lemma 3.D.2. *Take ν an environment with a minimal gap Δ_* and a balancedness θ_*. Consider $\hat{S} = \mathrm{SRI}(\delta, \Delta, \theta)$ the output of the SRI routine, designed with $\Delta > 0$ and $\theta > 0$. With probability $\mathbb{P}_{\mathrm{SRI},\nu}$ larger than $1 - \delta$, the following holds.*

- *The set $\hat{S}$ does not contain two arms from the same cluster.*
- *If $\Delta_* \leqslant \Delta/4$, then $\hat{S}$ contains strictly less than K arms.*
- *If $\Delta_* \geqslant \Delta$ and $\theta_* \geqslant \theta$ then $\hat{S}$ contains exactly one arm by group.*

Proof. The first point is a consequence of Lemma 3.C.8 and Lemma 3.C.10. The third point is exactly the result of Lemma 3.C.2.

For the second point, recall that by definition, a candidate a_u is bad if there exists an arm a in the set S such that $\|\mu_{a_u} - \mu_a\| \leqslant \Delta/4$. In Lemma 3.C.10, we prove that with probability larger than $1 - \delta$, no bad arms would be added to S. Moreover, if $\Delta_* \leqslant \Delta/4$, then there exists at least one group whose arms are bad during all the procedure, and hence, the second point is also a consequence of Lemma 3.C.8 and Lemma 3.C.10. □

3.D.2 Proof of Theorem 3.D.1

ACB* is δ-correct

Consider separately two cases $\Delta_* \leq \Delta_0$ and $\Delta_* > \Delta_0$. First focus on the case where $\Delta_* \leqslant \Delta_0$.

The procedure ACB_* consists on a sequence of calls for SRI, with different parameters, we remind these parameters as defined in (3.16), (3.17)

$$\Delta_0^2 = \sigma^2[\log(K) + \sqrt{d} + \log\log(6n/\delta)], \qquad \delta_l = \frac{\delta}{6(l+1)^3}$$
$$\theta_{p,l} = \frac{1}{K2^{l-p}}, \quad \Delta_p = \Delta_0\sqrt{\frac{1}{2^p}} \quad n'_p = \left\lceil c_6 \frac{\sigma^2}{\Delta_p^2}\left(\log(3K^2/\delta) + \sqrt{d\log(3K^2/\delta)}\right)\right\rceil .$$

For short, we write $\mathrm{SRI}(p, l)$ for SRI routine with parameters δ_l, Δ_p, and $\theta_{p,l}$. For $l \geqslant 0$ and $p = 0, \dots, l$, we define $\mathcal{E}_{p,l}$ as the event of probability larger than $1 - \delta_l$ under $\mathbb{P}_{\mathrm{SRI}(p,l),\nu}$ defined in Lemma 3.D.2. We write $\mathcal{E}$ for the intersection of these events.

From Lemma 3.D.2, the event $\mathcal{E}_{p,l}$ has a probability larger than $1 - \delta_l$. With a union bound, and the definition of δ_l (3.16), we deduce that

$$\mathbb{P}(\mathcal{E}) = \mathbb{P}\left(\bigcap_{p,l} \mathcal{E}_{p,l}\right) \geqslant 1 - \sum_{l\geqslant 1}\sum_{p=0}^{l} \delta_l = 1 - \sum_{l\geqslant 0} \frac{\delta}{6(l+1)^2} \geqslant 1 - \delta/3 \ .$$

We write (l', p') the first value of (l, p) in Algorithm 5 such that $|S_{l,p}| = K$. On the event $\mathcal{E}$, we have that $\hat{S} = S_{l',p'}$ contains exactly one arm by cluster — see again Lemma 3.D.2.

Even, if on the event $\mathcal{E}$, we know that $\Delta_* \geq \Delta_{p'}/4$ (see also Lemma 3.D.2). This lower bound on Δ_* could be used to parameterize the ADC, however, we prefer to estimate Δ_* directly in Algorithm 5 before applying the routine ADC.

Recall that $n'_p = c_6 \frac{\sigma^2}{\Delta_p^2}\left(\log(3K^2/\delta) + \sqrt{d\log(3K^2/\delta)}\right)$. We use $2Kn'_{p'}$ samples to estimate Δ_*—see $\widehat{\Delta}$ in Line 8 of Algorithm 5. Arguing as in the proof of Lemma 3.C.8, we deduce from the definition (3.56) of c_6, that, on the intersection of the event $\mathcal{E}$ with an event of probability higher than $1 - \delta/3$, we have

$$\frac{1}{4}\Delta_*^2 \leqslant \frac{1}{2}\hat{\Delta}^2 \leqslant \Delta_*^2$$

Since, on this event, we have $2^{-1/2}\hat{\Delta} \leq \Delta_*$, we are in position to apply Lemma 3.C.3 to $\mathrm{ADC}(\delta/3, 2^{-1/2}\hat{\Delta}, \hat{S})$. In summary, we have proved that ACB* is δ-correct.

Control of the budget of ACB*

We now bound the budget of ACB* under the same event as in the previous subsection. The key observation was proven page 22 of (Jamieson et al., 2016), it holds that

$$\{\theta \in (0, 1/K), \Delta \in (0, \Delta_0); \frac{\Delta_0^2}{K\theta\Delta^2} \leqslant 2^l\} \subset \bigcup_{p=0}^{l-1} \{(\theta, \Delta) : \theta \geqslant \theta_{p,l}, \Delta \geqslant \Delta_p\} \ .$$

In particular, if $2^l \geqslant \frac{\Delta_0^2}{K\theta_*\Delta_*^2}$, then, there exists $p \in [l-1]$ such that $\theta_{p,l} \leqslant \theta_*$ and $\Delta_{p,l} \leqslant \Delta_*$. From this result and from Lemma 3.D.2, we get that, on the event $\mathcal{E}$, the stopping time l' satisfies $l' \leqslant L_* = \left\lceil \log_2\left(\frac{\Delta_0^2}{\theta_* K \Delta_*^2}\right)\right\rceil$ —recall that L_* is defined in (3.55).

We write $\mathcal{T}_1$ at the total budget we have spent for computing $\widehat{S}$. Recall that the budget of the routine SRI is almost surely bounded by $T_{\max}$ —see (3.13) — and we upper-bounded $T_{\max}$ in (3.38). In order to emphasize the dependency of this budget on (δ, Δ, θ) we write $T_{\max}(\delta, \Delta, \theta)$ in the sequel.

By (3.38), on the event $\mathcal{E}$, we have

$$\begin{aligned}
\mathcal{T}_1 &\leqslant \sum_{l=0}^{L_*}\sum_{p=0}^{l} T_{\max}(\delta_l, \Delta_p, \theta_{p,l} \vee 1/n) \\
&\leqslant \sum_{l=0}^{L_*}\sum_{p=0}^{l} 2\left(\left\lceil \frac{8}{\theta_{p,l}}\log\left(\frac{8K}{\delta_l}\right)\right\rceil + K\right) + c'\frac{\sigma^2}{\Delta_p^2}\frac{1}{\theta_{p,l}}\log\left(\frac{K}{\delta_l}\right)\left[\log(K) + \sqrt{d} + \log\log\left(\frac{n}{\delta_l}\right)\right] \ .
\end{aligned}$$

We observe that, in ACB_* Line 3, we use SRI with $\theta_{p,l} \vee 1/n$ because any environment has necessary a balancedness larger than $1/n$. It allows us to bound the log log-term in Equation (3.38) by $\log\log(n/\delta)$.

Now, by definition (3.17), $\theta_{p,l} = \frac{1}{K2^{l-p}}$ and $\Delta_p^2 = \frac{\Delta_0^2}{2^p}$ so that $\frac{1}{\Delta_p^2}\frac{1}{\theta_{p,l}} = \frac{K2^l}{\Delta_0^2}$ and then

$$\begin{aligned}
\mathcal{T}_1 \leqslant & \sum_{l=0}^{L_*}\sum_{p=0}^{l}\left(16K\cdot 2^p\log\left(\frac{8K}{\delta_l}\right)+2(K+1)\right)\\
&+\sum_{l=0}^{L_*}\sum_{p=0}^{l}c'\frac{\sigma^2}{\Delta_0^2}K2^l\log\left(\frac{K}{\delta_l}\right)\left[\log(K)+\sqrt{d}+\log\log\left(\frac{n}{\delta_{L_*}}\right)\right]\\
\leqslant & 2(L_*+1)^2(K+1)+cK2^{L^*}\log\left(\frac{8K}{\delta_{L_*}}\right)\\
&+c'\frac{\sigma^2}{\Delta_0^2}K(L_*+1)2^{L_*}\log\left(\frac{K}{\delta_{L_*}}\right)\left[\log(K)+\sqrt{d}+\log\log\left(\frac{n}{\delta_{L_*}}\right)\right] \ .
\end{aligned}$$

Now, $2^{L_*} \leqslant 2\frac{\Delta_0^2}{\theta_* K\Delta_*^2}$, so that

$$\begin{aligned}
\mathcal{T}_1 \leqslant & 2(L_*+1)^2(K+1)+c\frac{\Delta_0^2}{\theta_*\Delta_*^2}\log\left(8\frac{K(L_*+1)^3}{\delta}\right)\\
&+c'(L_*+1)\frac{\sigma^2}{\theta_*\Delta_*^2}\log\left(\frac{6K(L_*+1)^3}{\delta}\right)\left[\log(K)+\sqrt{d}+\log\log\left(\frac{6n(L_*+1)^3}{\delta}\right)\right]\\
\leqslant & cL_*^2K+c'L_*\frac{\sigma^2}{\theta_*\Delta_*^2}\log\left(\frac{KL_*}{\delta}\right)\left[\log(K)+\sqrt{d}+\log\log\left(\frac{nL_*}{\delta}\right)\right] \ . \qquad (3.57)
\end{aligned}$$

In the last inequality, we used the expression of Δ_0^2 (3.16) which implies that

$$\frac{\Delta_0^2}{\theta_*\Delta_*^2}\log\left(8\frac{K(L_*+1)^3}{\delta}\right) \leqslant c'\frac{\sigma^2}{\theta_*\Delta_*^2}\log\left(\frac{KL_*}{\delta}\right)\left[\log(K)+\sqrt{d}+\log\log\left(\frac{n}{\delta}\right)\right] \ .$$

Let us now consider the budget $\mathcal{T}_2$ dedicated to the estimator of Δ_*. Since $\Delta_{p'}^{-2} \leq 2^{L_*}\Delta_0^{-2}$, we deduce that

$$\mathcal{T}_2 = 2Kn_{p'} \leq 2K + c\frac{\sigma^2}{\theta_*\Delta_*^2}\left(\log(3K^2/\delta)+\sqrt{d\log(3K^2/\delta)}\right) \ . \qquad (3.58)$$

Finally, as we are working under the event $\widehat{\Delta}^2/\Delta_*^2 \in [1/2,2]$, we deduce from Lemma 3.C.3 that the budget $\mathcal{T}_3$ incurred by ADC is smaller or equal to

$$\mathcal{T}_3 \leq 2n + c\frac{\sigma^2}{\Delta_*^2}n\log\left(\frac{n}{\delta}\right) + c\frac{\sigma^2}{\Delta_*^2}\sqrt{dnK\log\left(\frac{n}{\delta}\right)} \ . \qquad (3.59)$$

The total budget is obtained by summing the bounds (3.57), (3.58), and (3.59).

It remains to consider the case where $\Delta_* \geq \Delta_0$. In that case, under the events of the previous subsection, the first phase of the algorithm stops at the latest as $(l,p) = (L_*, 0)$, where $L_* =$

$\lceil \log_2(1/(\theta_* K)) \rceil$. Arguing as above, we deduce that $\mathcal{T}_1$ satisfies

$$\mathcal{T}_1 \leq cKL_*^2 + c' L_* \frac{1}{\theta_*} \log\left(\frac{KL_*}{\delta}\right) \quad . \tag{3.60}$$

Regarding the second step of the algorithm, we know that $p' \leq L_*$ so that $\Delta_{p'}^{-2} \leq 2^{L_*} \Delta_0^{-2} \leq \frac{\Delta_0^{-2}}{\theta_* K}$. We deduce that

$$\mathcal{T}_2 \leq 2K + c\frac{1}{\theta_*} \frac{\log(3K^2/\delta) + \sqrt{d \log(3K^2/\delta)}}{[\log(K) + \sqrt{d} + \log\log(6n/\delta)]} \leq 2K + c' L_* \frac{1}{\theta_*} \log\left(\frac{KL_*}{\delta}\right) \quad . \tag{3.61}$$

Finally, the budget $\mathcal{T}_3$ is still given by (3.59). Gathering (3.60), (3.61), and (3.59) allows us to conclude.

3.E Concentration inequalities

We now give a few concentration inequalities used in Chapter 3.

First, a consequence of the definition of σ-subGaussian random variables given in Assumption 3.2.1 is the following,

Lemma 3.E.1. *Let $Y \in \mathbb{R}$ be subGaussian, then for all $x > 0$,*

$$\mathbb{P}(X > x) \leqslant \exp(-\frac{x^2}{2}), \text{ and } \quad \mathbb{P}(X < -x) \leqslant \exp(-\frac{x^2}{2}) \ .$$

Here is Laurent and Massart' inequality, page 1325 of (Laurent and Massart, 2000).

Lemma 3.E.2 (Laurent & Massart). *Let $Z \sim \chi^2_d$ a chi-square distribution, where $d \geqslant 1$ is the degree of freedom, then for any $x > 0$,*

$$\mathbb{P}(Z \geqslant d + 2\sqrt{dx} + 2x) \leqslant \exp(-x), \text{ and } \quad \mathbb{P}(Z \leqslant d - 2\sqrt{dx}) \leqslant \exp(-x)$$

We now give the Hanson-Wright inequality for the concentration of scalar products of sub-Gaussian random variables — see (Rudelson and Vershynin, 2013) for the proof.

Lemma 3.E.3 (Hanson-Wright inequality)**.** *Let Y be a d-dimensional vector in $\mathbb{R}^d$ with independent, centered and 1-subGaussian components. Let A be a $d \times d$ matrix. Then, there exists a constant c_{hw} such that for any $x \geqslant 0$,*

$$\mathbb{P}(Y^T A Y - \mathbb{E}[Y^T A Y] > x) \leqslant \exp\left(-\frac{1}{c_{\text{hw}}}\left(\frac{x^2}{\|A\|_F^2} \wedge \frac{x}{\|A\|_{op}}\right)\right) \ ,$$

where $\|A\|_{op}$ is the operator norm of A, $\|A\|_F$ is the Frobenius norm.

We use the following corollary,

Corollary 3.E.4. *Let ν_1 and ν_2 be two probability distribution, with respective expectations μ_1 and μ_2. We assume that there exists Σ_1 and Σ_2 two symmetric $d \times d$ matrices such that, for $a = 1, 2$, under ν_a, $E = \Sigma_a^{-1/2}[X - \mu_a]$ is a vector with independent subGaussian random variables. Assume also that $\|\Sigma_1\|_{op} \leqslant \sigma^2$ and $\|\Sigma_2\|_{op} \leqslant \sigma^2$.*

Let $m_1 \in \mathbb{N}^*$ and $m_2 \in \mathbb{N}^*$ be two integers. Consider $X_{1,1}, \dots, X_{1,n_1}$ be i.i.d variables distributed as ν_1, and $X_{2,1}, \dots, X_{2,n_2}$ i.i.d variables distributed as ν_2, independently of the observations of a_2.

If $\epsilon_1 := \frac{\sqrt{n_1}}{\sigma}\left(\frac{1}{n_1}\sum_{i=1}^{n_1} X_{1,i} - \mu_1\right)$, and $\epsilon_2 := \frac{\sqrt{n_2}}{\sigma}\left(\frac{1}{n_2}\sum_{i=1}^{n_2} X_{2,i} - \mu_2\right)$, then, for any $x \geqslant 1$,

$$\mathbb{P}(\langle \epsilon_1, \epsilon_2 \rangle > x) \leqslant \exp\left(-\frac{2}{c_{\mathrm{hw}}}\left(\frac{x^2}{d} \vee x\right)\right) \ .$$

Similarly, for any $x > 0$, we have

$$\mathbb{P}\left(\langle \epsilon_1, \epsilon_2 \rangle > \frac{c_{\mathrm{hw}}}{2}x \vee \sqrt{\frac{c_{\mathrm{hw}}}{2}dx}\right) \leqslant \exp\left(-x\right) \ .$$

Proof. Let $a = 1, 2$. We specify the rotation Σ_a in the expression of ϵ_a,

$$\epsilon_a = \frac{\sqrt{n_a}}{\sigma}\left(\frac{1}{n_a}\sum_{i=1}^{n_a} X_{a,i} - \mu_a\right) = \frac{1}{\sigma}\Sigma_a^{1/2}\frac{1}{\sqrt{n_a}}\sum_{t=1}^{n_a}\Sigma_a^{-1/2}[X_{a,i} - \mu_a] \ .$$

Now, by assumption on the distribution ν_a, for all $i \in [n_a]$, the vector $\Sigma_a^{-1/2}[X_{a,i} - \mu_a]$ has independent and subGaussian entries. By independence of the random variables $(X_{a,1}, \dots, X_{a,n_a})$, the vector $\frac{1}{\sqrt{n_a}}\sum_{t=1}^{n_a}\Sigma_a^{-1/2}[X_{a,i} - \mu_a]$ has independent entries. By independence and using the definition of subGaussian variables given in Assumption 3.2.1, $Y_a := \frac{1}{\sqrt{n_a}}\sum_{t=1}^{n_a}\Sigma_a^{-1/2}[X_{a,i} - \mu_a]$ is composed of independent and subGaussian entries. It holds then that

$$\langle \epsilon_1, \epsilon_2 \rangle = Y_1^T \frac{\Sigma_1^{1/2}\Sigma_2^{1/2}}{\sigma^2} Y_2 = \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix}^T S \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} \ ,$$

where S is the following $2d \times 2d$ matrix

$$S := \frac{1}{2}\begin{bmatrix} 0 & \frac{\Sigma_1^{1/2}\Sigma_2^{1/2}}{\sigma^2} \\ \frac{\Sigma_1^{1/2}\Sigma_2^{1/2}}{\sigma^2} & 0 \end{bmatrix} \ .$$

We can then apply Lemma 3.E.3, noticing that $\mathbb{E}[\langle \epsilon_1, \epsilon_2 \rangle] = 0$, $\|S\|_{op} = \|\Sigma_1^{1/2}\Sigma_2^{1/2}\frac{1}{\sigma^2}\|_{op}/2 \leqslant 1/2$ and $\|S\|_F^2 \leqslant d/2$.

□

Chapter 4

Non-parametric Clustering with Bandit Feedback

Abstract. *Clustering with bandit feedback refers to the problem of partitioning a set of items, where the clustering algorithm can sequentially query the items to receive noisy observations. The problem is formally posed as the task of partitioning the arms of an n-armed stochastic bandit according to their underlying distributions, grouping two arms together if and only if they share the same distribution, using samples collected sequentially and adaptively. This setting has gained attention in recent years due to its applicability in recommendation systems and crowdsourcing. Existing works on clustering with bandit feedback rely on a strong assumption that the underlying distributions are sub-Gaussian. As a consequence, the existing methods mainly cover settings with linearly-separable clusters, which has little practical relevance.*

We introduce a framework of non-parametric clustering with bandit feedback, where the underlying arm distributions are not constrained to any parametric form, and hence, it is applicable for active clustering of real-world datasets. We adopt a kernel-based approach, which allows us to reformulate the non-parametric problem as the task of clustering the arms according to their kernel mean embeddings in a reproducing kernel Hilbert space (RKHS). Building on this formulation, we introduce the KACB algorithm with theoretical correctness guarantees and analyze its sampling budget. We introduce a notion of signal-to-noise ratio for this problem that depends on the maximum mean discrepancy (MMD) between the arm distributions and on their variance in the RKHS. Our algorithm is adaptive to this unknown quantity: it does not require it as an input yet achieves instance-dependent guarantees.

Related publication. This Chapter is a joint work with Sebastian Vogt [1], Debarghya Ghoshdastidar [2], and Nicolas Verzelen [3], available as a preprinted as (Thuot et al., 2026).

4.1 Introduction

We consider a non-parametric instance of the so-called clustering with bandit feedback problem introduced in (Yang et al., 2024) and (Thuot et al., 2025). In this pure exploration problem, the goal is to partition a set of unknown distributions, from which we can collect samples, and to provide guarantees on the partition returned by the learner. This problem captures various contemporary settings where data are collected sequentially and in an adaptive manner. In digi-

1. Equal contribution—Technical University of Munich, Munich, Germany.
2. Technical University of Munich, Munich, Germany.
3. INRAE, Mistea, Institut Agro, Univ Montpellier, Montpellier, France.

tal marketing, online platforms repeatedly interact with users and must quickly discover specific groups of customers (market segments) to personalize recommendations while limiting costly feedback collection. Beyond customer segmentation, our non-parametric, kernel-based formulation is well suited to modern high-dimensional or complex data for which parametric assumptions are often untenable, for example when clustering noisy biological or medical signals in adaptive trials. See (Yang et al., 2024) for further applications, e.g., in medical trials.

Clustering with bandit feedback. We adopt the stochastic multi-armed bandit model (see, e.g., Bubeck et al., 2012; Lattimore and Szepesvári, 2020), in which a learner interacts sequentially with an unknown environment with $n \in \mathbb{N}$ arms.

Each arm i is associated with an unknown distribution ν_i supported on a space $\mathcal{X}$. At each time step, the algorithm chooses an arm and observes a data sampled from its associated distribution. The algorithm interacts with the environment until a (random) stopping time $\mathcal{T}$– which it chooses. We consider the active clustering problem as in (Yang et al., 2024; Thuot et al., 2025). We assume that there exists an underlying partition of the arms into K groups such that two arms are in the same group if and only if they are associated with the same distribution. The learner's objective is to exactly recover this partition with probability of error at most a prescribed δ, in which case the algorithm is said to be δ-correct. Since data collection is costly, the goal is to design δ-correct algorithms, that make as few observations as possible.

Non-parametric formulation. In Yang et al. (2024) and Thuot et al. (2025), the authors restrict the possible arm distributions to sub-Gaussian distributions and cluster the arms according to their means in the original d-dimensional space $\mathbb{R}^d$. Our work aims to develop algorithms in the setting where no strong assumptions are imposed on the arm distributions, turning the problem into a non-parametric one.

Rather than working directly in the original space $\mathcal{X}$, we map distributions into a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ and represent each arm i through its kernel mean embedding (KME) in that space. Our kernel-based formulation allows us to drop assumptions in the original space: we only assume that the kernel g is bounded, characteristic and translation invariant. Importantly, the characteristic property of g implies that the non-parametric problem of clustering the arms according to their distributions is equivalent, in the RKHS, to clustering the arms according to their KMEs, which allows us to leverage kernel methods. Recently, the maximum mean discrepancy (MMD) have been extensively used for comparing and testing distributions in RKHS — see e.g., (Gretton et al., 2012; Muandet et al., 2017), we similarly use the MMD in our method.

Contributions. We introduce Kernel Active Bandit Clustering (KACB, Algorithm 9), a new sequential and adaptive algorithm for clustering with bandit feedback. First, KACB is δ-correct: it outputs the correct partition of the arms with error probability at most δ. Second, it leverages state-of-the-art concentration inequalities for empirical KMEs (Wolfer and Alquier, 2025) to adapt simultaneously to the unknown MMD between groups and to the unknown RKHS variances (see (4.3)).

Our main theoretical contribution is Theorem 4.3.1, which provides a non-asymptotic, high-probability upper bound budget of KACB and identifies a variance-aware signal-to-noise ratio governing the difficulty of the problem. At a high level, KACB is an adaptive algorithm that, at each iteration, runs variance-aware kernel two-sample tests based on empirical MMD and empirical variances for all pairs of arms, and increases the sampling budget until all tests can reliably decide whether two arms belong to the same group or not.

Related work on clustering with bandit feedback problems. The clustering with bandit feedback problem (CBP) has attracted increasing attention in recent years (Yang et al., 2024; Thuot et al., 2025; Chandran et al., 2025; Yavas et al., 2025; Graf et al., 2025). The problem was first formalized by Yang et al. (2024), who consider a parametric setting where arms are partitioned according to their d-dimensional means and the arm distributions are Gaussian. They introduce the BOC algorithm, a δ-correct procedure based on the Track-and-Stop method of Garivier and Kaufmann (2016), and establish that an expected budget is asymptotically optimal in the regime $\delta \to 0$. In turn, Thuot et al. (2025) study the same parametric CBP and provide a non-asymptotic characterization of the optimal complexity, combining tools from bandit pure exploration with high-dimensional hypothesis testing. A variant of the problem is considered by (Chandran et al., 2025), where the target partition is defined as the single-linkage clustering of the arm means. Yavas et al. (2025) study a family of distribution-matching problems, that encompasses the CBP under the assumption that the arm distributions lies on a finite alphabet. In a different direction, (Graf et al., 2025) study a similar clustering problem, but the learner is only able to sample partial information on each arm. All these works rely on strong parametric or distributional assumptions (e.g., Gaussian or sub-Gaussian arms, finite alphabets). In contrast, we consider here the active clustering problem in a genuinely non-parametric setting by working in an RKHS and clustering arms according to their kernel mean embeddings, under mild assumptions on the kernel rather than on the original distributions.

Related work on kernel methods. Recent advances in kernel methods have shown that kernel mean embeddings (KMEs) into RKHSs provide a powerful and versatile framework for statistical learning on distributions (Smola et al., 2007; Muandet et al., 2017). This framework has led to a wide range of applications, including kernel two-sample tests (Gretton et al., 2007a, 2012), independence testing (Gretton et al., 2007b), and many other distributional inference tasks; see Muandet et al. (2017); Berlinet and Thomas-Agnan (2011) for comprehensive reviews. From a more geometric point of view, Sriperumbudur et al. (2011) and Sriperumbudur et al. (2010) study when RKHS embeddings induce metrics on probability measures and characterize universality and characteristic kernels.

Within this framework, kernel two-sample tests compare empirical KMEs of two samples using the maximum mean discrepancy (MMD) as a test statistic, yielding non-parametric tests that avoid explicit density estimation. (Tolstikhin et al., 2017) study the minimax estimation of KMEs via their empirical counterparts and show that, for bounded kernels, the optimal rate depends only on kernel properties and not on the underlying distributions. (Tolstikhin et al.,

2016) extend this perspective to the minimax estimation of the MMD. More recently, (Wolfer and Alquier, 2025) derived state-of-the-art, variance-aware concentration bounds for the MMD between true and empirical KMEs. In particular, for distribution with small variance in the RKHS, the bounds from (Wolfer and Alquier, 2025) achieves better rates. Our analysis builds directly on these variance-aware KME and MMD concentration inequalities.

Related work on Kernel clustering. Another related line of work is the fruitful development of kernel methods for clustering. Kernel-based methods such as kernel k-means (Dhillon et al., 2004) and kernel spectral clustering (Ng et al., 2001) are widely used in practice, especially when cluster geometry is complex. More recently, (Vankadara et al., 2021) established separability conditions under which kernel-based clustering can recover the underlying true partition under non-parametric mixture models. In particular, their analysis highlights that these conditions can be expressed in terms of MMD between components. In a complementary line, kernel K-means has also been proposed for clustering distributional data by applying K-means directly to KMEs in RKHS and using MMD as the distance between probability measures (Baíllo et al., 2025). These works focus on batch clustering from a fixed sample. In contrast, we address an active, bandit-style setting where the learner adaptively decides which distributions (arms) to sample in order to recover the clustering.

Outline Section 4.2 introduces the problem setting and notation. Section 4.3 presents the KACB algorithm and our main upper bound on its sampling complexity.

Section 4.4 concludes with further comments and perspectives. The proof of our main theoretical contribution can be found in Appendix 4.A.

4.2 Setting and notation

Kernel and RKHS. Let $\mathcal{X}$ be a separable topological space. Let $g : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a continuous, positive definite kernel on $\mathcal{X}$. The kernel g induces a reproducing kernel Hilbert space (RKHS) $(\mathcal{H}, \langle \cdot, \cdot \rangle)$ and a feature map $\phi : \mathcal{X} \to \mathcal{H}$ such that

$$\forall x, y \in \mathcal{X} : \quad g(x, y) = \langle \phi(x), \phi(y) \rangle \ . \tag{4.1}$$

We assume that g is bounded, in the sense that $\sup_{x \in \mathcal{X}} g(x, x) < +\infty$. We define the supremum and range of g as

$$\bar{g} \coloneqq \sup_{x,y \in \mathcal{X}} g(x, y) \ , \qquad \tilde{g} \coloneqq \sup_{x,y \in \mathcal{X}} g(x, y) - \inf_{x,y \in \mathcal{X}} g(x, y) \ .$$

For a distribution ν on $\mathcal{X}$, its kernel mean embedding (KME) is defined as

$$\mu_\nu \coloneqq \mathop{\mathbb{E}}_{X \sim \nu} [\phi(X)] \in \mathcal{H} \ ,$$

where μ_ν is a Bochner integral ((Berlinet and Thomas-Agnan, 2011)). A kernel g is called characteristic if the KME map $\mu : \nu \mapsto \mu_\nu$ is injective. Finally, g is translation invariant if there exists a function $\Psi : \mathcal{X} \to \mathbb{R}$ such that $\forall x, x' \in \mathcal{X} \colon g(x, x') = \Psi(x - x')$ (Wolfer and Alquier, 2025).

For more background on RKHSs and kernel mean embeddings, see, e.g., (Gretton et al., 2012), (Muandet et al., 2017), or (Berlinet and Thomas-Agnan, 2011).

Model and unknown partition. We consider an n-armed stochastic bandit problem with $n \geq 2$ arms, indexed by $[n] = \{1, \dots, n\}$, where arm i is associated with an unknown distribution ν_i on $\mathcal{X}$. The environment is $\nu = \{\nu_1, \dots, \nu_n\}$. For any $i \in [n]$, we denote as

$$\mu_i := \mathop{\mathbb{E}}_{X \sim \nu_i} [\phi(X)] \in \mathcal{H} \ . \tag{4.2}$$

We assume that there exists a partition $\mathcal{C}^*$ of $[n]$ such that two arms belong to the same group if and only if they share the same kernel mean embedding (KME), and we denote by K the number of groups in $\mathcal{C}^*$. For any partition $\mathcal{C}$ of $[n]$, we write $C(i)$ for the cluster containing i. We say that $\mathcal{C}$ is correct if it groups together and only together arms with the same KME:

$$\forall i \neq j \in [n] \colon \quad \mu_i = \mu_j \iff C(i) = C(j) \ .$$

As usual in clustering problems, the true partition $\mathcal{C}^*$ is only defined up to permutation of the groups. When the kernel is characteristic, it corresponds exactly to the problem of clustering the arms according to their underlying distributions.

Sequential strategies and δ-correct objective. We work in an adaptive, sequential setting, where an algorithm interacts with the bandit environment—i.e., collects samples—in order to recover the clustering (up to permutation of the groups).

A strategy collects data sequentially from the environment in the following way: at each time $t \geq 1$, it selects an arm $A_t \in [n]$ and observes $X_t \sim \nu_{A_t}$. Let $\mathcal{F}_t = \sigma(A_1, X_1, \dots, A_t, X_t)$ denote the information available at time t. A strategy is specified by:

1. a selection rule choosing the next arm A_t based on the past $\mathcal{F}_{t-1}$;
2. a stopping time $\mathcal{T}$ with respect to $(\mathcal{F}_t)$ deciding when to stop;
3. a recommendation rule that outputs a clustering $\hat{\mathcal{C}}_\mathcal{T}$ based on $\mathcal{F}_\mathcal{T}$.

Given a confidence level $\delta \in (0, 1)$ and a class of environments $\mathcal{E}$, we call a strategy δ-correct for this problem if

$$\forall \nu \in \mathcal{E} \colon \quad \mathbb{P}\left(\hat{\mathcal{C}}_\mathcal{T} \text{ is correct}\right) \geq 1 - \delta \ .$$

Our goal is to design δ-correct algorithms whose sampling complexity (the sampling budget) is as small as possible. We consider for the problem the class of problems with exactly K nonempty groups, where the number of groups K is known by the learner.

We characterize this complexity in terms of three main factors: the confidence parameter δ, the kernel g, and the environment ν. On the kernel side, we will use the bound $\bar{g}$; on the environment side we consider distribution-dependent quantities in the RKHS. In particular, we measure

separation between arms via the maximum mean discrepancy $\|\mu_i - \mu_j\|$. Then, following (Wolfer and Alquier, 2025), we define the RKHS variance of arm i as

$$\mathcal{V}_i^* \coloneqq \mathop{\mathbb{E}}_{X\sim\nu_i}\left[\|\phi(X) - \mu_i\|^2\right] \ , \tag{4.3}$$

which we interpret as a noise level.

Having specified the model, and the δ-correct objective, we now turn to the design of our algorithm.

4.3 Algorithm

We introduce KACB (Kernel Active Bandit Clustering), an adaptive algorithm that recovers the K true clusters defined by equality of kernel mean embeddings (KME). The algorithm does not require any knowledge of the gaps between clusters or of the variances in the RKHS. The pseudocode is given in Algorithm 9.

Kernel two-sample testing for clustering. At a high level, KACB reduces clustering to deciding, for every pair of arms, whether they share the same KME or not. This is exactly the setting of kernel two-sample testing, which we now recall and adapt to our bandit scenario. In particular, we follow the work of (Wolfer and Alquier, 2025), and we use a variance-aware bound on empirical KME.

Consider two arms $i, j \in [n]$. The null hypothesis $H_0^{i,j} : \mu_i = \mu_j$ (same KME) is tested against $H_1^{i,j} : \mu_i \neq \mu_j$. Given two arms i and j and a per-arm budget $T \geq 1$, we draw i.i.d. samples $(X_1^i, \dots, X_T^i) \sim \nu_i$ and $(X_1^j, \dots, X_T^j) \sim \nu_j$ and form the empirical KMEs

$$\hat{\mu}_i \coloneqq \frac{1}{T}\sum_{t=1}^{T} g(X_t^i, \cdot), \qquad \hat{\mu}_j \coloneqq \frac{1}{T}\sum_{t=1}^{T} g(X_t^j, \cdot).$$

By the reproducing property, the squared empirical distance between $\hat{\mu}_i$ and $\hat{\mu}_j$ is computed using only the kernel:

$$\|\hat{\mu}_j - \hat{\mu}_i\|^2 = \frac{1}{T^2}\sum_{s,t=1}^{T}\left(g(X_s^i, X_t^i) - 2g(X_s^i, X_t^j) + g(X_s^j, X_t^j)\right). \tag{4.4}$$

As in kernel two-sample testing, we interpret $\|\hat{\mu}_j - \hat{\mu}_i\|$ as an empirical maximum mean discrepancy (MMD) between the distributions of arms i and j, and use it as a test statistic for the hypothesis $H_0^{i,j} : \mu_i = \mu_j$ versus $H_1^{i,j} : \mu_i \neq \mu_j$.

We reject $H_0^{i,j}$ (declare arms i, j in different clusters) if $\|\hat{\mu}_j - \hat{\mu}_i\| > \mathcal{B}^{i,j}(T, \delta')$, where $\mathcal{B}^{i,j}(T, \delta')$ is a threshold derived from non-asymptotic concentration bounds. The comparison $\|\hat{\mu}_i - \hat{\mu}_j\| \leqslant \mathcal{B}^{i,j}(T, \delta')$ serves as a proxy to decide whether (i, j) belong to the same group or not.

Standard MMD tests use kernel-uniform bounds $\mathcal{B}^{i,j} \asymp \sqrt{\bar{g}\log(1/\delta)/T}$ (Tolstikhin et al., 2017).

We instead leverage empirical RKHS variance estimators:

$$\hat{\mathcal{V}}_i \coloneqq \frac{1}{T-1}\sum_{t=1}^{T}\left(g(X_t^i, X_t^i) - \frac{1}{T}\sum_{s=1}^{T} g(X_t^i, X_s^i)\right). \tag{4.5}$$

We use then as threshold

$$\mathcal{B}^{i,j}(T,\delta') = \left(\sqrt{\hat{\mathcal{V}}_i} + \sqrt{\hat{\mathcal{V}}_j}\right)\sqrt{2\frac{\log\frac{8(n^2-n)}{\delta'}}{T}} + \frac{32}{3}\sqrt{\tilde{g}}\frac{\log\frac{8(n^2-n)}{\delta'}}{T} \quad . \tag{4.6}$$

Indeed, Lemma 4.A.3 from (Wolfer and Alquier, 2025) —see also Appendix 4.A— ensures that

$$\mathbb{P}\left[\left|\|\mu_i - \mu_j\| - \|\hat{\mu}_j - \hat{\mu}_i\|\right| \le \mathcal{B}^{i,j}(T,\delta')\right] \ge 1 - \frac{\delta'}{n^2-n} \quad .$$

Graph-based clustering. To transform these pairwise kernel two-sample tests into a clustering procedure, we introduce a graph-based subroutine that we call $\texttt{CLUSTER}(T,\delta')$ that takes as input a fixed per-arm sampling budget T and a confidence parameter $\delta' \in (0,1)$. The subroutine $\texttt{CLUSTER}(T,\delta')$ simultaneous performs two-sample testing across all $n(n-1)/2$ pairs via a graph-based construction (Algorithm 8). For any $\{i,j\}$, the edge $\{i,j\}$ is added to $G=(V,E)$ whenever $H_0^{i,j}$ is not rejected, that is $\|\hat{\mu}_j - \hat{\mu}_i\| \le \mathcal{B}^{i,j}$. The connected components of G form the clustering $\mathcal{C}$.

The *type I error* (false splits within clusters) will be controlled by construction: arms with identical KME remain connected with high probability. *Type II errors* (false merges across clusters) are controlled by the MMD concentration when T is large enough. Thus, CLUSTER either returns the true partition or fewer than K clusters.

Active choice of the Adaptive procedure. The minimal T ensuring reliable two-sample testing across all pairs depends on unknown gaps $\|\mu_i - \mu_j\|$ and variances $\mathcal{V}_i^*$. Hence, our main procedure KACB (Algorithm 9) applies the so-called "doubling trick". At iteration $k \ge 1$, define the per-arm sampling budget as

$$T_k = \left\lceil 2^k \log\Big(\frac{8(n^2-n)}{\delta_k}\Big)\right\rceil, \quad \delta_k = \frac{\delta}{4k^2}.$$

The algorithm calls the subroutine $\texttt{CLUSTER}(T_k,\delta_k)$ until $|\mathcal{C}_k| = K$, then outputs $\mathcal{C}_k$. Observe that the true number of clusters K is known by the learner, and is only used in the stopping condition. The quadratic decay of the confidence δ_k ensures $\sum_k \delta_k \le \delta$.

We establish in Theorem 4.3.1 that KACB is δ-correct, terminates almost surely, and that its sampling complexity $\mathcal{T}$ satisfies

$$\mathcal{T} \lesssim \frac{n}{s_*^2}\log\Big(\frac{n}{\delta}\Big),$$

where s_*^2 defined in (4.7) is interpreted as a signal-to-noise ratio.

Theorem 4.3.1 (KACB(δ, K) is δ-correct)**.** *Let g be a continuous, positive definite, characteristic, translation invariant, bounded kernel, and let $\delta \in (0,1)$. Define the (variance-aware) signal-to-noise ratio*

$$s_*^2(\nu) := \min_{\substack{i \neq j \in [n] \\ \mu_i \neq \mu_j}} \left(\frac{\|\mu_i - \mu_j\|^2}{\mathcal{V}_i^* \vee \mathcal{V}_j^*} \wedge \frac{2\|\mu_i - \mu_j\|}{\sqrt{\bar{g}}} \right) \quad . \tag{4.7}$$

Then:

1. *Algorithm 9, KACB(δ, K), is δ-correct, i.e., it outputs the true clustering with probability at least $1-\delta$;*
2. *with probability at least $1-\delta$, the total budget $\mathcal{T}$ of KACB(δ, K) satisfies*

$$\mathcal{T} \leqslant 8n \cdot \left(\frac{128}{s_*^2} \vee 1 \right) \cdot \left(\log \left(\frac{32(n^2-n)k_*^2}{\delta} \right) \right) \quad , \tag{4.8}$$

where $k_* = \left\lceil \log_2 \left(\frac{128}{s_*^2} \right) \right\rceil$ is a logarithmic term independent of δ.

The proof is postponed to Appendix 4.A.

Algorithm 8: CLUSTER(T, δ') Clustering with fixed budget $n \times T$

Input: T, δ'

$sample \leftarrow$ SAMPLE(T) ; ▷ Sample each arm T times

$(\|\hat{\mu}_j - \hat{\mu}_i\|)_{i \neq j \in [n]}$, $(\hat{\mathcal{V}}_i)_{i \in [n]} \leftarrow$ COMPUTESTATISTICS$(sample)$; ▷ See (4.4), (4.5)

$(\mathcal{B}^{i,j}(T, \delta'))_{i \neq j \in [n]} \leftarrow$ COMPUTETHRESHOLD(T, δ') ; ▷ See (4.6)

$V \leftarrow [n]$;

$E \leftarrow \emptyset$;

for $i \neq j \in [n]$ **do**

  if $\|\hat{\mu}_j - \hat{\mu}_i\| \leq \mathcal{B}^{i,j}(T, \delta')$ **then**

    $E \leftarrow E \cup \{\{i, j\}\}$; ▷ Do not reject $H_0^{i,j}$; connect i and j

  end

end

$G \leftarrow (V, E)$;

return GETCONNECTEDCOMPONENTS(G)

In Theorem 4.3.1, Algorithm 9 is shown to be δ-PAC while adapting to the instance-specific quantity s_*. We now comment on the sampling budget from Equation (4.8).

First, note that n appears as an overall multiplicative factor in the budget. This is because every call to CLUSTER (Algorithm 8) allocates samples uniformly across arms, so each arm is queried the same number of times. Such a linear dependence in n is unavoidable when the inter-cluster separations are all of the same order.

In total, each arm receives $\mathcal{T}/n$ samples, which scales as $s_*^{-2} \log(n/\delta)$. The quantity s_*^{-2} plays the role of a signal-to-noise ratio, matching the sample complexity needed to perform a non-parametric two-sample test based on the MMD with state-of-the-art procedures such as (Wolfer

Algorithm 9: KACB(δ, K) Kernel Active Bandit Clustering

Input: δ, K

for $k = 1, 2, \ldots$ **do**
  $\delta_k \leftarrow \frac{\delta}{4k^2}$; ▷ Per-iteration error
  $T_k \leftarrow \left\lceil 2^k \log \frac{8(n^2-n)}{\delta_k} \right\rceil$; ▷ Per-arm sampling budget
  $\mathcal{C}_k \leftarrow \textsc{Cluster}(T_k, \delta_k)$;
  if $|\mathcal{C}_k| = K$ **then**
    return $\mathcal{C}_k$; ▷ Stop when K clusters are identified
  end
end

and Alquier, 2025). The minimum in the definition of s_* reflects that, for every pair of arms i, j, one must test the hypotheses $\mu_i = \mu_j$ versus $\mu_i \neq \mu_j$with error probability at most $\delta/(n^2 - n)$,so that a union bound guarantees an exact clustering overall.

The term s_*^{-2} also appears inside the logarithm through an additive contribution of order $c\,\frac{n}{s_*^2} \log\log(s_*^{-2})$, reflecting the cost of adapting to the unknown value of s_*.

Although the linear kernel is unbounded, consider the special case of Gaussian distribution with a linear kernel, where $k(x, y) = \langle x, y\rangle$, and $\nu_i = \mathcal{N}(\mu_i, \sigma_i^2 I_d)$ on $\mathbb{R}^d$. To recover the bounded-kernel assumption, restrict to truncated Gaussians supported on a compact subset of $\mathcal{B}(0, \sqrt{\bar{g}})$. In this setting, our upper bound on the sampling budget scales as

$$\max_{\mu_i \neq \mu_j} \left(\frac{\sigma_i^2 \vee \sigma_j^2}{\|\mu_i - \mu_j\|_2^2} \vee \frac{\sqrt{\bar{g}}}{\|\mu_i - \mu_j\|_2} \right) \times n \log(n/\delta).$$

The bound from (Thuot et al., 2025), scales with

$$\max_{\mu_i \neq \mu_j} \frac{\sigma_i^2 \vee \sigma_j^2}{\|\mu_i - \mu_j\|_2^2} \times \left(n \log(n/\delta) \vee \sqrt{dKn \log(n/\delta)} \right),$$

and is optimal, known to be optimal, at least in the canonical regime where the clusters have comparable sizes. When $\|\mu_i - \mu_j\| \leq \sigma_j^2/\sqrt{\bar{g}}$ and the dimension dd is moderate, our bound is therefore analogous to this optimal Gaussian linear-kernel rate.

4.4 Discussion

Benefit of variance-aware bounds. In this work, we relied on the variance-aware bounds of (Wolfer and Alquier, 2025). As an alternative, we could have used sub-Gaussian type bounds –see Proposition A.1 in (Tolstikhin et al., 2017)–. For that purpose, we would only need to replace that threshold $\mathcal{B}^{i,j}(t, \delta')$ in Algorithm 8 by $\sqrt{\bar{g}/t}(\sqrt{\log(8(n^2 - n)/\delta)} + 2)$. The modified algorithm would still be δ-correct, but its budget $\mathcal{T}$ would now be bounded by $n\bar{g}[\min_{i \neq j} \|\mu_i - \mu_j\|]^{-2} \log(n/\delta)$. Since $\mathcal{V}_i^* \leqslant \bar{g}$ for every arm i, the variance-aware bounds of (Wolfer and Alquier, 2025) are never

worse and can substantially improve the budget whenever $\mathcal{V}_i^* \ll \bar{g}$.

Unknown number of clusters. In this work, we assume that the number of clusters is known to the learner, and our algorithm is adaptive to the unknown quantity s_*, which plays the role of a signal-to-noise ratio for the problem. The number of clusters is only used in the stopping condition of the procedure. If a lower bound $s_* \geqslant s_0$ is available, then, even without knowing K, a single call CLUSTER(T_0, δ) to Algorithm 8 with a per-arm sampling budget $T_0 \asymp s_0^{-2} \log(n/\delta)$ yields a correct partition with high probability $1 - \delta$. Observe that, when neither K nor s_* is known, the problem is ill-posed, since no algorithm can, in finite time, distinguish two arms that are arbitrarily close in MMD from two arms belonging to the same cluster.

Computational complexity. It is possible to implement KACB with a computational complexity of order $n^2 \log(n)$ by evaluating the quantities $\mathbf{1}\{\|\hat{\mu}_j - \hat{\mu}_i\| \leq \mathcal{B}^{ij}\}$ in a sequential fashion as done e.g.,in (Thuot et al., 2025), and by using efficient algorithms for computing connected components. As presented in Algorithm 9, the computational complexity of KACB is of the order $\log(n)\left[n^2 + n/s_*^2\right]$, this choice is made for the sake of clarity.

Adaptivity The algorithm is adaptive to the unknown quantity s_*. Leveraging classical techniques from bandit theory, it achieves a guarantee that scales as s_*^{-2}, without any prior knowledge of the environment, except for the number of clusters. One can also define a non-adaptive variant, which takes s_* as an input parameter. More precisely, consider the subroutine CLUSTER (Algorithm 8) with confidence parameter δ and per-arm sampling budget $T_* = 128\, s_*^{-2} \log(8(n^2-n)/\delta)$. Then Lemma 4.A.1 guarantees that the output of CLUSTER(T_*, δ) is correct with probability at least $1 - \delta$. The total budget of this non-adaptive procedure is therefore $c\, n\, s_*^{-2} \log(8(n^2 - n)/\delta)$. To the best of our knowledge, and using the MMD-based two-sample tests of (Wolfer and Alquier, 2025), this is the state-of-the-art sampling budget that ensures correct clustering of all arms with global error probability at most δ. Finally, note that the price of adaptivity is only an additional doubly logarithmic term $c\, \frac{n}{s_*^2} \log\log(s_*^{-2})$, which is negligible compared to the leading term, for instance if δ is small.

Appendix of Chapter 4

4.A Proofs of Theorem 4.3.1

In this appendix, we prove the correctness of KACB (Algorithm 9) and derive a high-probability upper bound on its budget.

4.A.1 Proof of Theorem 4.3.1

Recall from Section 4.3 that KACB repeatedly calls the subroutine `CLUSTER`(T_k, δ_k) with increasing per-arm budgets T_k and decreasing confidence levels δ_k. At iteration k, `CLUSTER` constructs a graph by performing variance-aware kernel two-sample tests between all pairs of arms and returns the connected components as a clustering $\mathcal{C}_k$. For any arm $i \in [n]$, let $C_k(i)$ denote the unique cluster in $\mathcal{C}_k$ containing i.

The analysis of KACB therefore reduces to understanding, for each fixed k, how `CLUSTER` behaves in terms of type I and type II errors under the thresholds $\mathcal{B}^{i,j}(T_k, \delta_k)$ introduced in Equation4.6.

Intuitively, we need that `CLUSTER`(T_k, δ_k) either returns fewer than K clusters or identifies the correct K clusters. The condition $\|\hat{\mu}_i - \hat{\mu}_j\| \leq \mathcal{B}^{i,j}(T_k, \delta_k)$ ensures that arms i and j with $\mu_i = \mu_j$ are clustered together with high probability. Moreover, there exists an iteration k such that arms with $\mu_i \neq \mu_j$ are separated (i.e., $\|\hat{\mu}_i - \hat{\mu}_j\| > \mathcal{B}^{i,j}(T_k, \delta_k)$) with high probability, yielding exactly K clusters. These key properties of `CLUSTER` are formalized in Lemma 4.A.1, whose proof—relying solely on concentration inequalities from Appendix 4.A.3—is deferred to Appendix 4.A.2.

Lemma 4.A.1. *For a fixed iteration $k \geq 1$, consider the partition $\mathcal{C}_k$ as the output of `CLUSTER` with parameters (T_k, δ_k), and let $C_k(i)$ denote the cluster containing arm i.*

Define the event type I error event $\mathcal{E}_{1,k}$ under which arms with identical KMEs are assigned to different clusters in $\mathcal{C}_k$:

$$\mathcal{E}_{1,k} = \bigcup_{\substack{i \neq j \in [n] \\ \mu_i = \mu_j}} \Big\{ C_k(i) \neq C_k(j) \Big\} \ . \tag{4.9}$$

Then,

$$\forall k \geqslant 1,\ \mathbb{P}(\mathcal{E}_{1,k}) \leqslant \delta_k \ .$$

Define the type II error event $\mathcal{E}_{2,k}$ under which arms with distinct KMEs are assigned to the same cluster in $\mathcal{C}_k$:

$$\mathcal{E}_{2,k} = \bigcup_{\substack{i \neq j \in [n] \\ \mu_i \neq \mu_j}} \Big\{ C_k(i) = C_k(j) \Big\} \ . \tag{4.10}$$

Let $s_* = s_*(\nu)$ be given by

$$s_*^2(\nu) := \min_{\substack{i \neq j \in [n] \\ \mu_i \neq \mu_j}} \frac{\|\mu_i - \mu_j\|^2}{\mathcal{V}_i^*} \wedge \frac{2\|\mu_i - \mu_j\|}{\sqrt{\bar{g}}} \ . \tag{4.11}$$

For any iteration $\forall k \geqslant 1$*, such that* $T_k \geqslant 128 \frac{1}{s_*^2} \log\left(\frac{8(n^2-n)}{\delta_k}\right)$*, then*

$$\mathbb{P}(\mathcal{E}_{2,k}) \leqslant \delta_k \ .$$

Proof of Theorem 4.3.1. Let $\delta \in (0,1)$ and assume the environment ν has n arms forming exactly K groups with distinct KMEs. We proceed in three parts: (i) correctness (outputting the true partition with probability at least $1-\delta$), (ii) almost-sure termination, and (iii) a high-probability bound on the sampling complexity.

(i) Correctness of KACB

Consider the call of $KACB(\delta, K)$ on the environment ν. Let $\mathcal{C}$ be the clustering returned by KACB(δ, K), obtained from `CLUSTER`(T_k, δ_k) at the stopping iteration $k \geq 1$. Observe that $k < +\infty$ happens almost surely (as we prove in (ii)), and the clustering $\mathcal{C}$ contains exactly K groups.

On the event $\mathcal{E}_{1,k}^c$ (Equation 4.9), all arms with identical KMEs are clustered together. By definition of the model (see Section 4.2), the true partition $\mathcal{C}^*$ is exactly the partition into equivalence classes of the relation $\mu_i = \mu_j$ and has cardinality K. As a consequence, any clustering with no type I error and exactly K nonempty clusters must coincide with $\mathcal{C}^*$ (up to permutation of the labels). Since the stopping iteration k is unknown, we proceed with a union bound

$$\begin{aligned}
\mathbb{P}\left(\{\mathcal{C} \text{ is incorrect}\}\right) =& \mathbb{P}\left(\{\exists k \geq 1 : \mathcal{C}_k \text{ is incorrect and } |\mathcal{C}_k| = K\}\right) \\
&\leq \sum_{k=1}^{\infty} \mathbb{P}\left(\mathcal{E}_{1,k}\right) \\
&\leq \sum_{k=1}^{\infty} \delta_k \leq \delta \ ,
\end{aligned}$$

where the final bound uses $\sum_{k=1}^{\infty} 1/k^2 = \pi^2/6$ and $\delta_k = \delta/(4k^2)$.

(ii) Almost-sure termination of KACB(δ, K)

On $\mathcal{E}_{1,k}^c \cap \mathcal{E}_{2,k}^c$, we have that $\mathcal{C}_k$ clusters the arms exactly according to their KMEs, and $\mathcal{C}_k$ equals the true partition. In particular $|\mathcal{C}_k| = K$, and the algorithm terminates. In other words, once T_k is large enough, the graph built by `CLUSTER` exactly matches the true equivalence relation $\mu_i = \mu_j$, and KACB stops at that iteration.

Then,

$$\mathbb{P}\left(KACB(\delta, K) \text{ never terminates}\right) \leqslant \inf_{k \to \infty} \mathbb{P}(\mathcal{E}_{1,k} \cup \mathcal{E}_{2,k}) \ .$$

Moreover, for k large enough, it holds that $T_k \geq 128 s_*^{-2} \log\left(8(n^2-n)/\delta_k\right)$, Lemma 4.A.1 then

implies that for k large enough, $\mathbb{P}(\mathcal{E}_{1,k} \cup \mathcal{E}_{2,k}) \leqslant \delta_k$, and $\lim_{k\to\infty} \mathbb{P}(\mathcal{E}_{1,k} \cup \mathcal{E}_{2,k}) = 0$. Finally,

$$\mathbb{P}\left(KACB(\delta, K) \text{ never terminates}\right) \leqslant \inf_{k\to\infty} \mathbb{P}(\mathcal{E}_{1,k} \cup \mathcal{E}_{2,k}) = 0 \ ,$$

and the algorithm terminates almost-surely.

(iii) Budget of KACB

Denote

$$k_* = \left\lceil \log_2\left(\frac{128}{s_*^2}\right)\right\rceil \vee 1 \ , \tag{4.12}$$

so that $T_k \geq 128 s_*^{-2} \log\left(8(n^2-n)/\delta_k\right)$ for all $k \geq k_*$. Lemma 4.A.1 then implies $\mathbb{P}(\mathcal{E}_{1,k} \cup \mathcal{E}_{2,k}) \leq \delta_k \leq \delta_{k_*}$ for $k \geq k_*$.

On $\mathcal{E}_{1,k_*}^c \cap \mathcal{E}_{2,k_*}^c$, the algorithm terminates at or before iteration k_*. The per-iteration budget is $\mathcal{T}_k = n \cdot T_k$, so the total budget $\mathcal{T}$ satisfies

$$\mathbb{P}\left(\mathcal{T} \leq \sum_{k=1}^{k_*} \mathcal{T}_k\right) \geq \mathbb{P}\left(\mathcal{E}_{1,k_*}^c \cap \mathcal{E}_{2,k_*}^c\right) \geq 1 - \delta_{k_*} \geqslant 1 - \delta \ .$$

Now,

$$\begin{aligned}
\sum_{k=1}^{k_*} \mathcal{T}_k =& n \sum_{k=1}^{k_*} T_k = n \sum_{k=1}^{k_*} \left\lceil 2^k \log \frac{8(n^2-n)}{\delta_k} \right\rceil \\
&\leq 2 \sum_{k=1}^{k_*} 2^k \cdot \log\left(\frac{8(n^2-n)}{\delta_{k_*}}\right) \\
&\leq 4n \cdot 2^{k_*} \cdot \log\left(\frac{8(n^2-n)}{\delta_{k_*}}\right) \\
&\leq 8n \cdot \left(\frac{128}{s_*^2} \vee 1\right) \cdot \left(\log\left(\frac{32(n^2-n)k_*^2}{\delta}\right)\right)
\end{aligned}$$

□

4.A.2 Proof of Lemma 4.A.1

Proof. We fix any iteration k, for which we call `CLUSTER`(T_k, δ_k). Recall that the algorithm `CLUSTER` constructs an undirected graph $G_k = ([n], E_k)$, whose vertices are the arms $[n]$, and whose set of edges is $E_k = \{\{i,j\} : \|\hat{\mu}_i - \hat{\mu}_j\| \leqslant \mathcal{B}^{i,j}(T_k, \delta_k)\}$. Then, the clustering $\mathcal{C}_k$ is defined as the connected components of G_k.

Type-I error: splitting arms with identical KME.

We consider the event $\mathcal{E}_{1,k}$ (Equation (4.9)), under which $\mathcal{C}_k$ assigns two arms with the same KME to different clusters. If $C_k(i) \neq C_k(j)$, then E_k cannot contain the edge $\{i,j\}$, because otherwise they would be in the same connected component, and thus in the same cluster. By definition of E_k, if $(i,j) \notin E_k$, that means that the comparison of $\|\hat{\mu}_j - \hat{\mu}_i\|$ to the decision

boundary yielded false. It follows by construction that

$$\mathcal{E}_{1,k} \subset \bigcup_{\substack{i \neq j \in [n] \\ \mu_i = \mu_j}} \left\{ \|\hat{\mu}_j - \hat{\mu}_i\| > \mathcal{B}^{i,j}(T_k, \delta_k) \right\},$$

where

$$\mathcal{B}^{i,j}(T_k, \delta_k) = \left(\sqrt{\hat{\mathcal{V}}_i} + \sqrt{\hat{\mathcal{V}}_j} \right) \sqrt{2 \frac{\log \frac{8(n^2-n)}{\delta_k}}{T_k}} + \frac{32}{3} \sqrt{\tilde{g}} \frac{\log \frac{8(n^2-n)}{\delta_k}}{T_k}.$$

Now, we bound the probability $\mathbb{P}(\mathcal{E}_{1,k})$, with a union bound, together with the concentration inequality from Theorem 4.A.3 with $\delta' = \frac{\delta_k}{n^2-n}$,

$$\begin{aligned} \mathbb{P}\left(\mathcal{E}_{1,k}\right) &\leq \sum_{\substack{i \neq j \in [n] \\ \mu_i = \mu_j}} \mathbb{P}\left(\left\{ \|\hat{\mu}_i - \hat{\mu}_j\| > \mathcal{B}^{i,j}(T_k, \delta_k) \right\} \right) \\ &\leq \sum_{\substack{i \neq j \in [n] \\ \mu_i = \mu_j}} \frac{\delta_k}{n^2 - n} \leqslant \delta_k \ . \end{aligned}$$

Type-II error: merging arms with different KMEs.

We now consider the event $\mathcal{E}_{2,k}$ (Equation 4.10), under which $\mathcal{C}_k$ assigns two arms with different KMEs to the same cluster. For simplicity, we note

$$\delta_{n,k} = \frac{\delta_k}{(n^2 - n)} = \frac{\delta}{4 \cdot (n^2 - n) \cdot k^2}$$

Assume that $\mu_i \neq \mu_j$, while $C_k(i) = C_k(j)$. By construction of $\mathcal{C}_k$, the condition $C_k(i) = C_k(j)$ implies that there exists a path in the graph $G_k = ([n], E_k)$ between i and j. Since there is a path between two arms with different means, somewhere on that path must be an edge connecting two arms with different means. Thus, it holds that

$$\mathcal{E}_{2,k} \subset \bigcup_{\substack{i,j \in [n] \\ \mu_i \neq \mu_j}} \left\{ \|\hat{\mu}_j - \hat{\mu}_i\| \leqslant \mathcal{B}^{i,j}(T_k, \delta_k) \right\} \ . \tag{4.13}$$

Let $i, j \in [n]$ be two arms such that $\mu_i \neq \mu_j$ Now we need to control the probability of the event $A_{i,j} \coloneqq \{\|\hat{\mu}_j - \hat{\mu}_i\| \leq \mathcal{B}^{i,j}(T_k, \delta_k)\}$.

Theorem 4.A.4 provides us that with probability at least $1 - \frac{\delta_{n,k}}{2}$

$$\|\hat{\mu}_j - \hat{\mu}_i\| \geq \|\mu_j - \mu_i\| - \left(\sqrt{\mathcal{V}_i^*} + \sqrt{\mathcal{V}_j^*} \right) \sqrt{2 \frac{\log \frac{8}{\delta_{n,k}}}{T_k}} - \frac{8}{3} \sqrt{\bar{g}} \frac{\log \frac{8}{\delta_{n,k}}}{T_k}.$$

It remains to prove that the assumption T_k ensures that (with high probability), the lower bound above from Theorem 4.A.4 will be larger than the threshold $\mathcal{B}^{i,j}(T_k, \delta_k)$ used for our classification.

Since $\mathcal{B}^{i,j}(T_k, \delta_k)$ contains the empirical variance, we first need to bound the empirical variance

by the true variance. We define the event

$$B_i := \left\{ \sqrt{\hat{\mathcal{V}}_i} \leq \sqrt{\mathcal{V}_i^*} + 2\sqrt{\frac{2\tilde{g}\log\frac{4}{\delta_{n,k}}}{T_k}} \right\} ,$$

whose probability is controlled by Theorem 4.A.5 as

$$\mathbb{P}\left(B_i^c\right) \leq \frac{\delta_{n,k}}{8}. \tag{4.14}$$

With a union bound on Equation 4.13, we have that

$$\begin{aligned}
\mathbb{P}(\mathcal{E}_{2,k}) &\leq \sum_{\substack{i,j\in[n]\\ \mu_i\neq\mu_j}} \mathbb{P}\left(A_{i,j}\right) \\
&\leq \sum_{\substack{i,j\in[n]\\ \mu_i\neq\mu_j}} \left(\mathbb{P}\left(A_{i,j}\cap B_i\cap B_j\right) + \mathbb{P}\left(B_i^c\right) + \mathbb{P}\left(B_j^c\right)\right) \\
&\leq \sum_{\substack{i,j\in[n]\\ \mu_i\neq\mu_j}} \left(\mathbb{P}\left(A_{i,j}\cap B_i\cap B_j\right) + \frac{\delta_{n,k}}{4}\right) ,
\end{aligned} \tag{4.15}$$

where the final inequality follows from Equation (4.14). Then, it remains to bound the probability of the events $A_{i,j}\cap B_i\cap B_j$.

Under $A_{i,j}\cap B_i\cap B_j$, it holds that:

$$\begin{aligned}
\|\hat{\mu}_j - \hat{\mu}_i\| &\leq \left(\sqrt{\hat{\mathcal{V}}_i} + \sqrt{\hat{\mathcal{V}}_j}\right)\sqrt{2\frac{\log\frac{8}{\delta_{n,k}}}{T_k}} + \frac{32}{3}\sqrt{\tilde{g}}\frac{\log\frac{8}{\delta_{n,k}}}{T_k} \\
&\leq \left(\sqrt{\mathcal{V}_i^*} + \sqrt{\mathcal{V}_j^*} + 4\sqrt{\frac{2\tilde{g}\log\frac{8}{\delta_{n,k}}}{T_k}}\right)\sqrt{2\frac{\log\frac{8}{\delta_{n,k}}}{T_k}} + \frac{32}{3}\sqrt{\tilde{g}}\frac{\log\frac{8}{\delta_{n,k}}}{T_k} \\
&= \left(\sqrt{\mathcal{V}_i^*} + \sqrt{\mathcal{V}_j^*}\right)\sqrt{2\frac{\log\frac{8}{\delta_{n,k}}}{T_k}} + \frac{56}{3}\sqrt{\tilde{g}}\frac{\log\frac{8}{\delta_{n,k}}}{T_k}
\end{aligned} \tag{4.16}$$

Now, assume that

$$T_k \geq \max_{\substack{i\neq j\in[n]\\ \mu_i\neq\mu_j}} \max\left\{128\frac{\mathcal{V}_i^*}{\|\mu_i-\mu_j\|^2}, \frac{112\sqrt{\tilde{g}}+16\sqrt{\bar{g}}}{3\|\mu_i-\mu_j\|^2}\right\}\log\frac{8}{\delta_{n,k}}$$

In particular, as $\tilde{g}\leqslant 2\bar{g}$, this will hold if $T_k \geqslant 128\frac{1}{s_*^2}\log\frac{8}{\delta_{n,k}}$.

Now, we derive from direct computation that

$$\begin{aligned}
&\left(\sqrt{\mathcal{V}_i^*}+\sqrt{\mathcal{V}_j^*}\right)\sqrt{2\frac{\log\frac{8}{\delta_{n,k}}}{T_k}}+\frac{56}{3}\sqrt{\bar{g}}\frac{\log\frac{8}{\delta_{n,k}}}{T_k} \\
&\leqslant \|\mu_j-\mu_i\|-\left(\sqrt{\mathcal{V}_i^*}+\sqrt{\mathcal{V}_j^*}\right)\sqrt{2\frac{\log\frac{8}{\delta_{n,k}}}{T_k}}-\frac{8}{3}\sqrt{\bar{g}}\frac{\log\frac{8}{\delta_{n,k}}}{T_k} \qquad (4.17)
\end{aligned}$$

Finally, we gather Equations (4.16), and (4.17) to control $\mathbb{P}\left(A_{i,j}\cap B_i\cap B_j\right)$,

$$\begin{aligned}
&\mathbb{P}\left(A_{i,j}\cap B_i\cap B_j\right) \\
&\leq \mathbb{P}\left(\|\hat{\mu}_j-\hat{\mu}_i\|\leq\left(\sqrt{\mathcal{V}_i^*}+\sqrt{\mathcal{V}_j^*}\right)\sqrt{2\frac{\log\frac{8}{\delta_{n,k}}}{T_k}}+\frac{56}{3}\sqrt{\bar{g}}\frac{\log\frac{8}{\delta_{n,k}}}{T_k}\right) \\
&\leq \mathbb{P}\left(\|\hat{\mu}_j-\hat{\mu}_i\|\leq\|\mu_j-\mu_i\|-\left(\sqrt{\mathcal{V}_i^*}+\sqrt{\mathcal{V}_j^*}\right)\sqrt{2\frac{\log\frac{8}{\delta_{n,k}}}{T_k}}-\frac{8}{3}\sqrt{\bar{g}}\frac{\log\frac{8}{\delta_{n,k}}}{T_k}\right) \\
&\leq \frac{\delta_{n,k}}{2},
\end{aligned}$$

where the last inequality results from Theorem 4.A.4 applied with $\delta'=\frac{\delta_{n,k}}{2}$.

Finally, we come back to our previous bound on $\mathbb{P}(\mathcal{E}_{2,k})$ in Equation 4.15, to conclude that

$$\begin{aligned}
\mathbb{P}(\mathcal{E}_{2,k}) &\leq \sum_{\substack{i,j\in[n]\\ \mu_i\neq\mu_j}}\left(\mathbb{P}\left(A_{i,j}\cap B_i\cap B_j\right)+\frac{2\delta_{n,k}}{8}\right) \\
&\leq \sum_{\substack{i,j\in[n]\\ \mu_i\neq\mu_j}}\left(\frac{\delta_{n,k}}{2}+\frac{\delta_{n,k}}{4}\right) \\
&\leqslant \delta_k \ ,
\end{aligned}$$

where the final inequality follows from the definition of $\delta_{n,k}=\frac{\delta_k}{(n^2-n)}$ □

4.A.3 Concentration inequalities

The design of the algorithm relies mostly on concentration inequalities for empirical KME, and empirical variances. For completeness, we recall in this subsection several inequalities whose proofs can be found in (Wolfer and Alquier, 2025).

Lemma 4.A.2 (Variance-aware empirical bound (Wolfer and Alquier, 2025))**.** *Let g be a continuous, positive definite, characteristic, translation invariant, bounded kernel. Assume arm $i\in[n]$ was sampled $t\in\mathbb{N}$ times to calculate the empirical KME $\hat{\mu}_i$ and the empirical variance $\hat{\mathcal{V}}_i$. Then*

it holds that

$$\forall \delta' \in (0,1) \colon \mathbb{P}\left(\left\{ \|\mu_i - \hat{\mu}_i\| \leq \sqrt{2\hat{\mathcal{V}}_i \frac{\log \frac{4}{\delta'}}{t}} + \frac{16}{3}\sqrt{\tilde{g}}\frac{\log \frac{4}{\delta'}}{t} \right\}\right) \geq 1 - \delta'.$$

For our application, we will prefer the following inequality, which follows directly from Theorem 4.A.2, by application of the triangle inequality.

Lemma 4.A.3 (Variance-aware empirical bound for the distance of two arms)**.** *Under the same assumptions as Theorem 4.A.2*

$$\forall \delta' \in (0,1) \colon \mathbb{P}\left(\left|\|\hat{\mu}_j - \hat{\mu}_i\| - \|\mu_j - \mu_i\|\right| \leq \mathcal{B}\right) \geq 1 - \delta' \ ,$$

for

$$\mathcal{B} \coloneqq \left(\sqrt{\hat{\mathcal{V}}_i} + \sqrt{\hat{\mathcal{V}}_j}\right)\sqrt{2\frac{\log \frac{8}{\delta'}}{t}} + \frac{32}{3}\sqrt{\tilde{g}}\frac{\log \frac{8}{\delta'}}{t} \ .$$

Lemma 4.A.4. *Let g be a continuous, positive definite and bounded kernel with supremum $\bar{g}$. Assume arms $i \in [n]$ was sampled t times to calculate the empirical KME $\hat{\mu}_i$. Then it holds that for all $\delta \in (0,1)$*

$$\mathbb{P}\left(\left|\, \|\hat{\mu}_j - \hat{\mu}_i\| - \|\mu_j - \mu_i\| \,\right| \leq \mathcal{B}_*\right) \geq 1 - \delta \ ,$$

for

$$\mathcal{B}_* \coloneqq \left(\sqrt{\mathcal{V}_i^*} + \sqrt{\mathcal{V}_j^*}\right)\sqrt{2\frac{\log \frac{4}{\delta}}{t}} + \frac{8}{3}\sqrt{\bar{g}}\frac{\log \frac{4}{\delta}}{t} \ .$$

Lemma 4.A.5 (Bound for empirical Variance (Wolfer and Alquier, 2025))**.** *Let g be a continuous, positive definite, characteristic, translation invariant, bounded kernel. Assume arm $i \in [n]$ was sampled $t \in \mathbb{N}$ times to calculate the empirical variance $\hat{\mathcal{V}}_i$. Let $b \in \{-1, 1\}$ Then it holds that*

$$\forall \delta \in (0,1) \colon \mathbb{P}\left(b\left[\sqrt{\hat{\mathcal{V}}_i} - \sqrt{\mathcal{V}_i^*}\right] \leq 2\sqrt{\frac{2\tilde{g}\log \frac{1}{\delta}}{t}}\right) \geq 1 - \delta.$$

Chapter 5

Clustering Items Through Bandit Feedback Finding the Right Feature out of Many

Abstract. *We study the problem of clustering a set of items based on bandit feedback. Each of the n items is characterized by a feature vector, with a possibly large dimension d. The items are partitioned into two unknown groups, such that items within the same group share the same feature vector. We consider a sequential and adaptive setting in which, at each round, the learner selects one item and one feature, then observes a noisy evaluation of the item's feature. The learner's objective is to recover the correct partition of the items, while keeping the number of observations as small as possible. We provide an algorithm which relies on finding a relevant feature for the clustering task, leveraging the Sequential Halving algorithm. With probability at least $1-\delta$, we obtain an accurate recovery of the partition and derive an upper bound on the budget required. Furthermore, we obtain an instance-dependent lower bound, which is tight in some relevant cases.*

Related publication. This Chapter is a joint work with Maximilian Graf [1], and Nicolas Verzelen [2], published in ICML 2025 (Graf et al., 2025).

5.1 Introduction

We consider a sequential and adaptive pure exploration problem, in which a learner aims to cluster a set of items, each represented by a feature vector in $\mathbb{R}^d$. The items are partitioned into two unknown groups such that items within the same group share the same feature vector. The learner sequentially selects an item and a feature, and then observes a noisy evaluation of the chosen feature of that item. Given a prescribed probability δ, the learner's objective is to collect enough information to recover the partition of the items with a probability of error at most δ.

This problem arises in crowdsourcing platforms, where complex labeling tasks are decomposed into simpler sub-tasks, typically involving answering specific questions about an item — see (Ariu et al., 2024). A motivating example is image labeling: a platform sequentially presents an image to a user along with a simple question such as "Is this a vehicle?" or "How many wheels can you see?". The learner leverages these answers to classify the images into categories. In this setting, the images correspond to items that must be clustered, while questions correspond to features. This problem is a special case of the model studied in (Ariu et al., 2024), where the authors

1. Equal contribution—Institut für Mathematik, Universität Potsdam, Potsdam, Germany
2. INRAE, Mistea, Institut Agro, Univ Montpellier, Montpellier, France

numerically demonstrate the advantage of an adaptive sampling scheme over non-adaptive ones. However, they do not establish the theoretical validity of their adaptive procedure.

From a theoretical perspective, our problem consists of clustering n items based on sequential and adaptive queries to some of their d features. Intuitively, the difficulty of the clustering task is driven by the differences between items in different groups across these d features. In particular, it depends both on the magnitude of these differences and on their sparsity—that is, the number of features on which the items differ significantly.

In this work, we precisely characterize the sample complexity of the two-group [3] clustering task, in a fully adaptive setting. Our **main contributions** are as follows:

— We introduce the `BanditClustering` procedure — Algorithm 13. On the one hand, it outputs the correct partition of the items with a prescribed probability $1-\delta$. On the other hand, it adapts to the unknown means of the groups in order to sample at most the informative features. In Theorem 5.3.1, we provide a tight, non-asymptotic upper bound on its sample complexity as a function of n, d, $\log(1/\delta)$, and the difference between the means.

— Conversely, we establish in Section 5.4 an information-theoretic lower bound on the budget, which entails the optimality of `BanditClustering`.

From a high-level perspective, our algorithm operates in three steps: first, it identifies a pair of representative belonging to different groups; second, it selects a feature that best discriminates between the groups; and finally, it leverages this discriminative feature to cluster all items.

Connection to good arm identification and adaptive sensing literature . One of the key challenges is to achieve the trade-off between the budget used for identifying a good discriminative feature, and the budget used for the clustering task. We borrow techniques from the best-arm identification literature, specifically employing the Sequential Halving algorithm (Karnin et al., 2013b) as a subroutine, leveraging its strong performance in settings where multiple arms are nearly optimal (Zhao et al., 2023; Katz-Samuels and Jamieson, 2020b; Chaudhuri and Kalyanakrishnan, 2019). The first identification step — finding two items belonging to distinct groups — is closely related to the adaptive sensing strategies for signal detection, as studied in (Castro, 2014), where the problem is framed as a sequential and adaptive hypothesis testing task. Furthermore, our approach incorporates ideas from (Castro, 2014; Saad et al., 2023) to efficiently identify the most informative features for clustering.

Connection to dueling bandits literature . Our bandit clustering problem is an instance of a pure bandit exploration problem, where one can sample interaction between items and features. In that respect, it is also related to ranking (Saad et al., 2023) and dueling bandits in the online literature (Ailon et al., 2014; Chen et al., 2020; Heckel et al., 2019; Jamieson and Nowak, 2011; Jamieson et al., 2015; Urvoy et al., 2013; Yue et al., 2012; Haddenhorst et al., 2021b) where the goal is to recover a partition of the items based on noisy pairwise comparisons. Some ranking procedures are based on estimating the Borda count (Heckel et al., 2019), some other procedures, such as (Saad et al., 2023), aim at adapting to the unknown form of the comparison matrix to

3. We focus on $K = 2$ groups. We discuss how to extend our ideas to $K > 2$ clusters in Section 5.6.

reduce the total budget. In essence, our approach is related, as it seeks to balance the trade-off between identifying relevant entries and exploiting them for efficient comparisons.

Connection to other bandit clustering problems . Recent works (Yang et al., 2024; Thuot et al., 2025; Yavas et al., 2025) have investigated clustering in a bandit setting, where items must be clustered based on noisy evaluations of their feature vectors. However, in these settings, the entire feature vector of the chosen item is observed at each sampling step, whereas in our framework, only a single feature of a given item is observed per step. Our observation scheme enables a more efficient allocation of the budget by focusing on the most relevant features — those that best discriminate between groups. The trade-off between exploring relevant features and exploiting them for classification is at the core of our work. This allows us to cluster the items with a much lower observation budget than in (Yang et al., 2024; Thuot et al., 2025) — see the discussion section. Other authors have previously introduced adaptive clustering problems for crowdsourcing (Ho et al., 2013; Gomes et al., 2011), although their settings does not directly relate to ours.

Connection to online clustering of bandits . We point out another line of works (Gentile et al., 2014; Li et al., 2019; Liu et al., 2022; Li et al., 2025) on the so-called *online clustering of bandits* problem — an instance of contextual linear bandit. This problem bears some resemblance to our setting, as it involves exploring a bandit environment with an underlying clustering structure among items. Still, there are two major differences with our problem: (1) the learner has no control over which items are presented at each time step, and (2) the algorithms (such as the CLUB Algorithm and its extensions) are designed and evaluated in a cumulative regret setting.

Organization. The model is introduced along with notation in the following Section 5.2. In Section 5.3, we describe our procedure and analyze its sample complexity. Section 5.4 provides matching lower bounds on the budget that imply the optimality of our procedure. Finally, we present numerical experiments in Section 5.5. We discuss the extension to $K > 2$ groups in Section 5.6. We finally discuss our results in Section 5.7.

5.2 Problem formulation and notation

Consider a set of n items, indexed by $[n] = \{1, \dots, n\}$. Each item is characterized by a feature vector of dimension d, where the number of features d may be large. Let $M \in \mathbb{R}^{n \times d}$ be the $n \times d$ matrix such that the i-th row of M contains the feature vector of item i. We denote the feature vector of item i as $M_{i,\cdot} = (M_{i,1}, \dots, M_{i,d})$. We assume that the n items are partitioned into two unknown groups, such that items within the same group share the same feature vector. The groups are assumed to be nonempty and non-overlapping. The objective is to recover these two groups.

Assumption 5.2.1 (Hidden partition)**.** *There exist two distinct vectors $\mu_0 \in \mathbb{R}^d$ and $\mu_1 \in \mathbb{R}^d$,*

and a non-constant label vector $g^* \in \{0,1\}^n$ *such that for any item* $i \in [n]$,

$$M_{i,\cdot} = \begin{cases} \mu_0 & \text{if } g^*(i) = 0 \ , \\ \mu_1 & \text{if } g^*(i) = 1 \ . \end{cases}$$

As is standard in clustering, (g^*, μ_0, μ_1) encodes the same matrix M as $(1-g^*, \mu_1, \mu_0)$. Therefore, we assume without loss of generality that $g^*(1) = 0$ in order to make the label vector g^* identifiable.

We consider a bandit setting, in which the learner sequentially and adaptively observes noisy entries of the matrix M. At each time step t, based on passed observations, the learner selects one item $I_t \in [n]$ and one feature $J_t \in [d]$. She then receives X_t, a noisy evaluation of M_{I_t,J_t}. Conditionally on the pair (I_t, J_t), X_t is an independent sample drawn from an unknown distribution ν_{I_t,J_t} with expectation M_{I_t,J_t}. The collection of distributions $(\nu_{i,j})_{i,j}$ is referred to as the environment.

We assume that the noise in observations is 1-subGaussian.

Assumption 5.2.2 (1-subGaussian noise)**.** *For any pair* $(i,j) \in [n] \times [d]$*, if* $X \sim \nu_{i,j}$*, then* $X - M_{i,j}$ *is* 1*-subGaussian, namely*

$$\mathbb{E}\left[\exp(x(X - M_{i,j}))\right] \leq \exp\left(x^2/2\right) \quad \forall x \in \mathbb{R} \ .$$

This subGaussian assumption is standard in the bandit literature (Lattimore and Szepesvári, 2020). It covers, for example, the emblematic case where observations follow Gaussian distributions with variance at most 1, as well as bounded random variables such as those following a Bernoulli distribution on $[0,1]$. By rescaling, the results can be extended to the case of σ-subGaussian noise.

We tackle this pure exploration problem in the fixed confidence setting, a common framework in the bandit literature (Lattimore and Szepesvári, 2020). In this setting, the learner must decide not only which observations to make but also when to stop. Given a prescribed error probability δ, the learner aims to recover the correct partition of the items with probability at least $1-\delta$, while minimizing the total number of observations. The learner sequentially collects observations until a stopping time T, after which it outputs an estimated label vector $\hat{g} \in \{0,1\}^n$ (satisfying $\hat{g}(1) = 0$). The total number of observations, given by T, is referred to as the budget of the procedure. Formally, T is a stopping time with respect to the natural filtration associated to the sequential model.

For any confidence level δ, we say then that a procedure $\mathcal{A}$ is δ-correct if

$$\mathbb{P}_{\mathcal{A},\nu}(\hat{g} = g^*) \geqslant 1 - \delta \ ,$$

where $\mathbb{P}_{\mathcal{A},\nu}$ denotes the probability distribution induced by the interaction between the environment ν and the algorithm $\mathcal{A}$.

The performance of a δ-correct algorithm is evaluated through its budget T, which should be as small as possible. In this paper, we derive upper bounds on T that hold with high probability—typically on an event of probability at least $1-\delta$, under which the algorithm returns a correct

clustering.

We introduce two key quantities for analyzing the problem. The gap vector is defined as

$$\Delta := \mu_1 - \mu_0 \in \mathbb{R}^d \ ,$$

which naturally captures the difficulty of the clustering task. By assumption, we have $\Delta \neq 0$ so that there is exactly two disjoint groups. The smaller the norm of Δ is, the more challenging the estimation of g becomes. In particular, we analyze the complexity of the clustering task with respect to the entry of the gap vector Δ ordered by decreasing absolute value, namely $\left|\Delta_{(1)}\right| \geq \left|\Delta_{(2)}\right| \geq \cdots \geq \left|\Delta_{(d)}\right|$.

Most intuitions behind our method rely on the sparse setting, where the two groups differ in exactly s entries with a constant gap h. This corresponds to the case where the gap vector Δ has exactly s nonzero entries equal to $h > 0$ and $d - s$ entries equal to 0. The smaller the sparsity level s, the more entries must be explored to detect a discriminative feature. The smaller the magnitude h, the more budget is required to distinguish between the two groups.

Besides, we define θ the balancedness of the partition g^*, that is the proportion of arms in the smallest group

$$\theta := \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}(g^*(i) = 0) \wedge \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}(g^*(i) = 1) \ .$$

Intuitively, the smaller θ is, the more unbalanced the partition is, and the more difficult it is to discover two items of distinct groups. For identifiability reasons, we assumed in Assumption 5.2.1 that the groups are nonempty — which implies in particular that $n \geqslant 2$, but also that $\theta \in [1/n; 1/2]$.

5.3 Algorithms

5.3.1 Introduction to our method

We introduce briefly our method, which contains two steps.

First, we fix arbitrary the item $r_0 = 1$ as a representative of the first group [4]. Then, we aim to identify a second representative item $r_1 \in [n]$ that belongs to the other group, and a feature $j \in [d]$ such that $M_{r_1,j}$ differs from $M_{r_0,j}$ significantly, that is such that $|M_{r_1,j} - M_{r_0,j}|$ is large. Our method balances the budget spent on identifying such discriminative feature, with the budget required for classifying all items based on this feature. Improving the budget compared to non-active settings requires over-sampling certain rows , which we refer to as representatives, following (Thuot et al., 2025). This step is crucial for accurately estimating some entries of the vectors μ_0 and μ_1 and, ultimately, for accelerating the clustering task.

Importantly, if we detect an entry (r_1, j) in the matrix where the gap $|M_{r_1,j} - M_{r_0,j}|$ is sufficiently large, we obtain two complementary pieces of information. This naturally leads us to organize the clustering task as a two-step procedure:

4. By symmetry, any randomly selected row could serve the same purpose.

1. If we can test that $|M_{r_1,j} - M_{r_0,j}| > 0$, then item r_1 can serve as a representative of the second group. Algorithm 11 identifies such an item r_1 with high probability. Once the two representatives r_0 and r_1 are known, we allocate a significant portion of the budget to these items in order to identify a discriminative feature in the gap-vector Δ.

2. If we identify a feature j such that $|M_{r_1,j} - M_{r_0,j}|$ not only differs from zero but also exceeds a certain threshold—specified later—then feature j is deemed sufficiently discriminative, and we concentrate the classification budget on this feature. We then estimate $|\Delta_j| = |M_{r_0,j} - M_{r_1,j}|$ with samples from entries (r_0, j) and (r_1, j), and classify the remaining items with a budget of order $O\left(\frac{n}{\Delta_j^2}\log(n/\delta)\right)$, uniformly allocated over items in the j-th column of M. This second step is detailed in Algorithm 12.

5.3.2 Warm-up: adaptation of Sequential Halving

As a subroutine, we introduce `CSH` for `CompareSequentialHalving`, which is detailed in Algorithm 10. It is a variant of the Sequential Halving (SH) algorithm, introduced in (Karnin et al., 2013b), which is very similar to Bracketing Sequential Halving described in (Zhao et al., 2023, Alg. 3). Building on recent advances in the analysis of SH applied to various best arm identification problems (Zhao et al., 2023), we analyze the performance of the method in a specific problem that we introduce. We provide an explicit guarantee for `CSH` in Lemma 5.B.1.

The algorithm takes 4 entries r_0, I, L, T. Given a fixed item $r_0 \in [n]$ and a subset of items $I \subset [n] \setminus r_0$, `CSH` outputs an item $i \in I$ and a feature $j \in [d]$ for which the absolute difference $|M_{i,j} - M_{r_0,j}|$ is as large as possible. Each time we run `CSH`, we allocate a budget T [5], that the algorithm can fully spend. That is, `CSH` operates under a fixed budget constraint.

Following the literature on best arm identification with multiple good arms (Berry et al., 1997; Katz-Samuels and Jamieson, 2020b; Jamieson et al., 2016; De Heide et al., 2021), we incorporate a sub-sampling mechanism. Initially, `CSH` selects randomly S_0, a subset of 2^L entries from $I \times d$, where L is a parameter specifying the sub-sampling size. Sequential Halving is then applied exclusively to the selected subset, rather than to the entire matrix. The optimal choice for L balances two factors. If L is small, the algorithm concentrates more budget per entry, enabling the detection of smaller gaps. If L is large, we increase the likelihood of including in S_0 a significant proportion of good entries, ensuring that the quality of the remaining entry is not limited by an unlucky draw.

We employ CSH as a subroutine in both Algorithm 11 and Algorithm 12. In Algorithm 11, we explore the entire matrix in order to detect a row r_1 that differs from the first row $r_0 = 1$, for this we use $I = [n] \setminus r_0$. In Algorithm 12, we focus the exploration on two rows $r_0 = 1$ and r_1 ($I = \{r_1\}$), aiming at detecting a feature that best separates the two groups represented by r_0 and r_1.

5. Actually, row r_0 is sampled half of the time, so that each sample from $\nu_{i,j}$ is compared to a new one from $\nu_{r_0,j}$. Any more refined bookkeeping of samples from row r_0 would at best improve the budget by 2.

Algorithm 10: `CompareSequentialHalving (CSH)`

Input: r_0 an item, $I \subset [n] \setminus \{r_0\}$ subset of items, L number of halving steps, $T \geq 2^{L+2}$ budget

Output: a couple $(i, j) \in I \times [d]$

Select $S_0 \leftarrow \{(i_1, j_1), \dots, (i_{2^L}, j_{2^L})\}$ uniformly with replacement from $I \times [d]$;
for $l = 1, \dots, L$ **do**
  $\tau_l \leftarrow \left\lfloor \frac{T}{2^{L-l+2}L} \right\rfloor$;
  for $(i, j) \in S_{l-1}$ **do**
    Draw $X_{r_0,j}^{(1)}, \dots, X_{r_0,j}^{(\tau_l)} \sim^{\text{i.i.d.}} \nu_{r_0,j}$;
    Draw $X_{i,j}^{(1)}, \dots, X_{i,j}^{(\tau_l)} \sim^{\text{i.i.d.}} \nu_{i,j}$;
    Store $\widehat{D}_{i,j} \leftarrow \frac{1}{\tau_l} \sum_{u=1}^{\tau_l} \left(X_{i,j}^{(u)} - X_{r_0,j}^{(u)} \right)$;
  end
  Keep in S_l the 2^{L-l} indices $(i, j) \in S_{l-1}$ with largest $|\widehat{D}_{i,j}|$;
end
return $(i, j) \in S_L$

5.3.3 First step: `CandidateRow`

We start our procedure by solving a sub-problem which consists on detecting an item r_1 which is not in the same group as the prefixed representative $r_0 = 1$. We perform this step in the following Algorithm 11. The guarantees of Algorithm 11 are proved in Appendix 5.C and are gathered in Proposition 5.C.1.

Algorithm 11: `CandidateRow (CR)`

Input: confidence parameter $\delta > 0$, item r_0

Output: row index $r_1 \in [n]$

Initialize $r_1 \leftarrow 0$, $k \leftarrow 1$;
while $r_1 = 0$ **do**
  for $1 \leq L \leq L_{\max}$ *such that* $L \cdot 2^L \leq 2^{k+1}$ **do**
    $(i, j) \leftarrow \texttt{CSH}([n], L, 2^{k+1})$;
    Draw $X_{r_0,j}^{(1)}, \dots, X_{r_0,j}^{(2^k)} \overset{\text{i.i.d.}}{\sim} \nu_{r_0,j}$, $X_{i,j}^{(1)}, \dots, X_{i,j}^{(2^k)} \overset{\text{i.i.d.}}{\sim} \nu_{i,j}$;
    if $\left| \sum_{t=1}^{2^k} \left(X_{i,j}^{(t)} - X_{r_0,j}^{(t)} \right) \right| > \sqrt{4 \cdot 2^k \log\left(\frac{k^3}{0.15\delta} \right)}$ **then**
      $r_1 \leftarrow i$;
    end
  end
  $k \leftarrow k + 1$;
end

In Algorithm 11, we perform multiple runs of the `CHS` subroutine, iteratively increasing the budget $T_k = 2^{k+1}$ allocated for each run. For a given run of $\texttt{CSH}(r_0, [n], L, T_k)$, we obtain an

entry (i, j) (Line 4). In Line 5, we use the same amount of observations 2^{k+1} to estimate the gap $|M_{i,j} - M_{r_0,j}|$. Finally, in Line 8, we perform a test based on a Hoeffding's bound to decide whether $|M_{i,j} - M_{r_0,j}| > 0$ or not. If at some point, this test concludes, Algorithm 11 outputs $r_1 = i$. The threshold chosen in the stopping condition from Line 6 is designed to assure that with a probability larger than δ, then the selected item r_1 belongs to another group as $r_0 = 1$.

In Line 3, we chose $L_{\max}$ as

$$L_{\max} \coloneqq \left\lceil \log_2 \left(16dn \log \left(\frac{4 \log(8nd)}{\delta} \right) \right) \right\rceil \ ,$$

which corresponds to the sub-sampling budget required, according to Lemma 5.B.1, when $\theta = 1/n$ takes the smallest possible value.

From Lemma 5.B.1, we know that for $\Delta_{(s)}^2 \leq 128L^2$, for some constant c, if the condition

$$T_k = 2^{k+1} \geqslant cL_{\max}^3 \frac{d\left(\log(1/\delta) + \log\log(nd)\right)}{\theta s \Delta_{(s)}^2} \ ,$$

holds, then, with high probability, there exists $L \leqslant L_{\max}$ such that $\texttt{CHS}(r_0, [n], L, T_k)$ outputs a pair (i, j) with $|M_{i,j} - M_{r_0,j}| \geq \left|\Delta_{(s)}\right|/2$. Besides, under this budget condition, the termination condition from Line 8 will be reached w.h.p. This condition is especially true for the sparsity $s \in [d]$, where the inequality above is tightest.

As in (Jamieson et al., 2016; Saad et al., 2023), the exponential grid $T_k = 2^k$ allows us to adapt the strategy, and reach a budget that scales up to log terms as $O\left(\min_{s\in[d]} \frac{d}{\theta s \Delta_{(s)}^2} \log(1/\delta)\right)$, even without prior knowledge of this quantity by the learner.

Interestingly, we can relate this quantity to the l^2 norm of Δ. For that, we define

$$s^* \in \operatorname{argmax}_{s\in[d]} s \cdot \Delta_{(s)}^2 \ . \tag{5.1}$$

This quantity, s^*, appears as an effective sparsity parameter, as observed in signal detection contexts. Actually, the following bound holds

$$\max_{s\in[d]} s \cdot \Delta_{(s)}^2 \leq \|\Delta\|_2^2 \leq \log(2d) \max_{s\in[d]} s \cdot \Delta_{(s)}^2 \ . \tag{5.2}$$

Finally, we prove that Algorithm 11 outputs an item r_1 which belongs to the second group with a probability larger than $1 - \delta$, using a budget that is smaller, up to logarithmic terms, than the quantity $\frac{d}{\theta\|\Delta\|_2^2} \log(1/\delta)$. We will see in Theorem 5.4.1 that this bound is optimal for both θ and Δ.

5.3.4 Second step: `ClusterByCandidates`

Consider the high probability event on which, after the first step of our procedure, Algorithm 11 provides an item $r_1 \in [n]$ such that $M_{r_1,\cdot} \neq M_{r_0,\cdot}$. Our next goal is to select a feature j such that $|\Delta_j|$ is large enough to allow a quick classification of each of the n items. We propose the procedure

Algorithm 12 which shares a similar structure with Algorithm 11. The guarantees of Algorithm 12 are proved in Appendix 5.C, in Proposition 5.D.1.

Algorithm 12: `ClusterByCandidates (CBC)`

Input: confidence $\delta > 0$, representative items $r_0, r_1 \in [n]$
Output: labels $\hat{g} \in \{0,1\}^n$

1 $\hat{g} \leftarrow (0,\dots,0)^T \in \{0,1\}^n$, $k \leftarrow \lceil \log_2(n) \rceil$;
2 **while** *True* **do**
3 **for** $1 \leq L \leq \tilde{L}_{\max}$ *such that* $L \cdot 2^L \leq 2^{k+1}$ **do**
4 $j \leftarrow \texttt{CSH}(r_0, r_1, L, 2^{k+1})$;
5 Draw $X^{(1)}_{r_0,j}, \dots, X^{(\lfloor 2^k/n \rfloor)}_{r_0,j} \sim^{\text{i.i.d.}} \nu_{r_0,j}$, $X^{(1)}_{r_1,j}, \dots, X^{(\lfloor 2^k/n \rfloor)}_{r_1,j} \sim^{\text{i.i.d.}} \nu_{r_1,j}$;
6 $\hat{D} \leftarrow \sum_{t=1}^{\lfloor 2^k/n \rfloor} (X_{r_1,j} - X_{r_0,j})$;
7 $\varepsilon \leftarrow \sqrt{4 \cdot \lfloor 2^k/n \rfloor \log(nk^3/0.15\delta)}$;
8 **if** $\left|\hat{D}\right| \geq 3 \cdot \varepsilon$ **then**
9 **for** $i \in [n]$ **do**
10 Draw $X^{(1)}_{r_0,j}, \dots, X^{(\lfloor 2^k/n \rfloor)}_{r_0,j} \sim^{\text{i.i.d.}} \nu_{r_0,j}$, $X^{(1)}_{i,j}, \dots, X^{(\lfloor 2^k/n \rfloor)}_{i,j} \sim^{\text{i.i.d.}} \nu_{i,j}$;
11 $\hat{D}_i \leftarrow \sum_{t=1}^{\lfloor 2^k/n \rfloor} (X_{i,j} - X_{r_0,j})$;
12 $\hat{g}(i) \leftarrow \mathbb{1}\left(\left|\hat{D}_i\right| \geq \varepsilon\right)$;
13 **end**
14 **output** $(\hat{g}(1), \dots, \hat{g}(d))$;
15 **end**
16 **end**
17 **end**

In Line 4, we call $\texttt{CSH}(r_0, \{r_1\}, L, 2^{k+1})$. We obtain a feature j and then estimate $|\Delta_j| = |M_{r_0,j} - M_{r_1,j}|$ in Line 6. For that, we take $\lfloor 2^k/n \rfloor$ samples from $\nu_{r_0,j}$ and $\nu_{r_1,j}$, and compute the sum of differences $\hat{D} = \sum_{t=1}^{\lfloor 2^k/n \rfloor} (X_{r_1,j} - X_{r_0,j})$. We deduce a high probability lower bound $|\widehat{\Delta}|_j = \frac{1}{\lfloor 2^k/n \rfloor}(\hat{D} - \epsilon) \leqslant |\Delta_j|$, where ϵ is defined in Line 7. Based on $|\widehat{\Delta}|_j$, we can classify (with high probability) each item by sampling the j-th feature $O\left(\frac{1}{\widehat{\Delta}^2} \log(n/\delta)\right)$. We can then assess whether the classification budget required for feature j is feasible given the budget T_k. If $T_k \leqslant \frac{cn}{\widehat{|\Delta|}_j^2} \log(n/\delta)$, it seems that with feature j, the classification budget exceeds T_k, we discard this feature and repeat `CSH` with larger sub-sampling size L or budget T.

We now bound the budget of our procedure thanks to Lemma 5.B.1. Assume that $\Delta^2_{(s)} \leq 128L^2$. If it holds for some constant c that

$$T_k = 2^{k+1} \geqslant cL^3_{\max} \frac{d\left(\log(1/\delta) + \log\log(d)\right)}{s\Delta^2_{(s)}} \quad .$$

Then, there exists $L \leqslant \tilde{L}_{\max}$ such that $\mathrm{CHS}(r_0, r_1, L, T_k)$ outputs a feature j such that

$$|M_{r_0 j} - M_{r_1 j}| \geq \left|\Delta_{(s)}\right| / 2 \ .$$

If we also have $T_k \geqslant c\frac{n}{\Delta^2_{(s)}} \log(n/\delta)$, then the algorithm stops, otherwise it would continue sampling. Overall, we prove that the total budget of the procedure, up to logarithmic factors, is no more than

$$\min_{s \in [d]} \left(\frac{d}{s} + n\right) \left(\frac{1}{\Delta^2_{(s)}} + 1\right) \log(1/\delta) \ .$$

5.3.5 Main Algorithm

To obtain a complete clustering procedure that is adaptive to Δ and θ, one simply has to combine Algorithm 11 and Algorithm 12. The overall clustering procedure is then given in Algorithm 13.

Algorithm 13: `BanditClustering`

Input: confidence parameter $\delta > 0$
Output: labels $\hat{g} \in \{0,1\}^n$

Fix $r_0 = 1$;
$r_1 \leftarrow \mathtt{CR}(\delta/2, r_0)$;
$\hat{g} \leftarrow \mathtt{CBC}(\delta/2, r_0, r_1)$;

Theorem 5.3.1. *For $\delta \in (0, 1/2e)$, consider Algorithm 13 with entry δ. Define*

$$H := \frac{d}{\theta}\left(\frac{1}{\|\Delta\|^2} + \frac{1}{s^*}\right) + \min_{s \in [d]} \left(\frac{d}{s} + n\right) \left(\frac{1}{\Delta^2_{(s)}} + 1\right) \ , \tag{5.3}$$

where s^* is the effective sparsity defined in (5.2).

With a probability of at least $1 - \delta$, Algorithm 13 returns $\hat{g} = g^*$ with a budget of at most

$$T \leqslant \tilde{C} \cdot \log\left(\frac{1}{\delta}\right) \cdot H,$$

where there exists a numerical constant C, and an index $\tilde{s} = s^* \vee (\lceil d/n \rceil \wedge |\{j \in [d] \ , \Delta_j \neq 0\}|)$, such that $\tilde{C}$ is a logarithmic factor smaller than

$$\begin{aligned} & C \cdot (\log\log(1/\delta) \vee 1)^4 \\ & \quad \cdot \log(dn)^5 \log(d) (\log_+ \log(1/\Delta^2_{(\tilde{s})}) \vee 1) \ . \end{aligned}$$

In order to understand the motivation behind our complexity H, we write our main theorem in the sparse setting — Δ is s-sparse with constant magnitude h.

Corollary 5.3.2. *For $\delta \in (0, 1/e)$ and $\Delta \in \{0, h\}^d$ with $0 < h < 1$, with a probability of at least $1 - \delta$, Algorithm 13 returns $\hat{g} = g^*$ with a budget of at most*

$$\tilde{C} \cdot \log(1/\delta) \cdot \left(\frac{d}{\theta \|\Delta\|^2} + \frac{n}{h^2} \right),$$

where $\tilde{C}$ is a logarithmic factor smaller than

$$C \cdot (\log\log(1/\delta) \vee 1)^4 \cdot \log(dn)^5 (\log_+ \log(1/h^2) \vee 1) \ ,$$

with a numerical constant $C > 0$.

Compared to the lower bound in Theorem 5.4.1, we prove that our procedure is optimal when the gap vector is s-sparse with a constant magnitude h. For a general gap vector Δ, we have good reasons to think that understanding the optimality of this trade-off for this simple example allows us to understand (at least intuitively) the optimality for general vectors.

We interpret H in (5.3) as a non-asymptotic sampling complexity, which depends on the instance-specific parameters of our model θ, Δ, n, and d. The complexity H can be decomposed as two terms:

First Term: $\frac{d}{\theta\|\Delta\|^2} \log(1/\delta)$, which correspond to the budget used to identify an item belonging to the second group. In the sparse setting, it scales as $\frac{d}{\theta s} \times \frac{1}{h^2}$, which is necessary. Indeed, we need to explore at least $\frac{d}{\theta s}$ entries to find a non-zero entry. Then, we need at least $1/h^2$ samples from each of these entries to decide if it is equal to 0 or not with a constant probability of error.

Second Term: $\min_{s \in [d]} \left(\frac{d}{s} + n \right) \left(\frac{1}{\Delta_{(s)}^2} + 1 \right)$. This term represents the best trade-off between the exploration of the gap vector Δ and its exploitation for clustering. Indeed, the term $\frac{n}{\Delta_{(s)}^2} \log(1/\delta)$ is the price for clustering if we use a feature with a gap $|\Delta_{(s)}|$ whereas $\frac{d}{s\Delta_{(s)}^2} \log(1/\delta)$ corresponds to the price for identifying a feature with a gap at least $|\Delta_{(s)}|$.

Define the effective sparsity $\tilde{s}$ for which the minimum holds, and define the effective magnitude as $\Delta_{(\tilde{s})}$. In the sparse setting, this is exactly the sparsity level. Intuitively, we can argue that entries significantly larger than $\Delta_{(\tilde{s})}$ are too rare to be detected (otherwise, $\tilde{s}$ would be smaller), and entries much smaller than $\Delta_{(\tilde{s})}$ are too weak to be used for classification with a budget H. Our insight is that the problem is as hard as if the gap vector were $\tilde{s}$-sparse with a constant magnitude $\Delta_{(s^*)}$, a setting where we have matching lower and upper bounds (see Corollary 5.3.2,Theorem 5.4.1), leading to our complexity.

Finally, we mention that the remaining term $d/(\theta s^*)$ in H only dominates in the very specific setting where the non-zero entries of Δ are really large so that $\|\Delta^2\| \geq s^*$.

5.4 Lower bounds

In this section, we provide a lower bound on the budget of any δ-correct algorithm. The bound is instance-dependent, meaning it holds for a specific problem instance defined by the matrix M. We establish this result by constructing a family of alternative environments, each obtained by

slightly modifying the original matrix M. We then prove that any algorithm with a budget that is too small cannot perform well simultaneously across all these environments. For the lower bound, we consider Gaussian environments, for which Assumption 5.2.2 holds.

We define $\mathcal{E}_{\text{per}}(M)$ as the set of Gaussian environments constructed from M by permuting its rows and columns. Without loss of generality, we assume that $\mu_0 = 0$ and $\mu_1 = \Delta$. Formally, an environment $\tilde{\nu} \in \mathcal{E}_{\text{per}}(M)$ is defined using a permutation σ of $[n]$ and a permutation τ of $[d]$ as follows:

$$\tilde{\nu}_{i,j} = \begin{cases} \mathcal{N}(0,1) & \text{if } g^*(\sigma(i)) = 0 \\ \mathcal{N}(\Delta_{\tau_j}, 1) & \text{if } g^*(\sigma(i)) = 1 \end{cases}, \tag{5.4}$$

where $g^* \in \{0,1\}^n$ denotes the unknown labels associated to matrix M. Intuitively, permuting the rows and columns of M accounts for the fact that (a) the target labels g^* are not available to the learner, and (b) the structure of the gap vector Δ is also unknown.

Theorem 5.4.1. *Fix $\delta \in (0, 1/4)$. Assume that $\mathcal{A}$ is δ-correct for the clustering task, then, there exists $\tilde{\nu} \in \mathcal{E}_{\text{per}}(M)$ such that the $(1-\delta)$-quantile of the budget of algorithm $\mathcal{A}$ is bounded as follows*

$$\mathbb{P}_{\mathcal{A},\tilde{\nu}}\left(T \geqslant \frac{2(n-2)}{\Delta_{(1)}^2} \log\left(\frac{1}{4.8\delta}\right) \vee \frac{2d}{\theta\|\Delta\|_2^2} \log\frac{1}{6\delta}\right) \geqslant \delta \tag{5.5}$$

The lower bound contains two terms. The first term scales as $\frac{d}{\theta\|\Delta\|^2}\log(1/\delta)$, and can be interpreted as the budget required to identify one item from each group, while adapting to the unknown structure of the gap vector Δ. This term matches, up to logarithmic factors, the budget incurred in the first step of our algorithm for identifying a relevant item. In particular, it implies that the first step from Algorithm 11 is optimal.

The second term scales as $\frac{n}{\Delta_{(1)}^2}\log(1/\delta)$, and take into account the difficulty of clustering all items once a discriminative feature is identified. Specifically, if the most informative feature is provided by an oracle—i.e., a feature index $j \in [d]$ such that $|\Delta_j| = \Delta_{(1)}$ is maximal—then the problem reduces to performing n independent Gaussian hypothesis tests of the form $H_0 : X \sim \mathcal{N}(0,1)$ versus $H_1 : X \sim \mathcal{N}(\Delta_{(1)}, 1)$.

In the case where the gap vector Δ takes two values, this lower bound matches, up to poly-logarithmic terms, the upper bound from Corollary 5.3.2. In summary, when Δ only takes two values, our budget is optimal with respect to d, n, θ and $\log(1/\delta)$. For more general Δ, we conjecture that the trade-off in H in (5.3) is optimal and unavoidable.

5.5 Experiments

We support our theoretical results with numerical experiments on synthetic data. In the first experiment, we investigate the sample complexity of our algorithm in sparse regimes. We compare its performance to a uniform sampling baseline, where each of the $n \times d$ entries of the matrix M is sampled an equal number of times, and the resulting data is clustered using the K-means algorithm. In the second experiment, we evaluate the `BanditClustering` algorithm under fixed

sparsity while simultaneously increasing the parameters n and d, and we compare different choices of the confidence parameter $\delta > 0$. In a third experiment, we compare `BanditClustering` to `Adaptive Clustering` (Ariu et al., 2024) and see how a growing number of features impacts both algorithms performances. In a fourth experiment, we illustrate the influence of θ on the budget of `BanditClustering`. As shown in Corollary 5.3.2, the effect of $1/\theta$ is comparable to that of the sparsity s discussed in the first experiment.

The code we used for the simulations is available in a GitHub repository [6].

Experiment 1. In this experiment, we consider a small number $n = 20$ of items, and a large number $d = 1000$ of features. For $s \in 1, \dots, d$, define the vector $\Delta^s = (\underbrace{h_s, \dots, h_s}_{s \text{ times}}, 0, \dots, 0)$, where $h_s = 15/\sqrt{s}$. This choice ensures that Δ^s is s-sparse and satisfies $\|\Delta^s\|_2 = 15$ for any s. First, we construct the matrix M^s with half of its rows equal to 0 and the other half equal to Δ^s, and sample observations as $\nu_{i,j} \sim \mathcal{N}(M^s_{i,j}, 1)$. Then, we examine the behavior of our algorithms as s varies, which causes the magnitude h_s to vary accordingly.

We report separately the budgets required by the two steps of our algorithm. For $\delta = 0.8$, we run each of the algorithms `CR`$(\delta/2)$, `CBC`$(\delta/2, i^*)$ and `BanditCluster`(δ) a total of $\kappa = 5000$ times. We provide to `CBC` a representative item i^* in the second group as an oracle. In this setting, the observed error rate of `BanditClustering` remains close to 0.01 for all values of s. We depict the average budgets required by `BanditClustering`(δ), considering only the runs where the first step `CR` returns a valid candidate row (otherwise, we emergency-stop the algorithm). We also show the average budget required by `CR`$(\delta/2)$ and `CBC`$(\delta/2, i^*)$ in Figure 5.1. The figure includes the $(0.05, 0.95)$ quantiles across simulations.

As a benchmark, we compare our algorithm to a strategy that samples uniformly all entries of M^s, and then applies the `kmeans` algorithm from the Scikit-learn library (Pedregosa et al., 2011). Given a budget T, we sample $\tau = \lfloor T/nd \rfloor$ observations $X^{(t)}_{i,j} \sim^{\text{i.i.d.}} \mathcal{N}(M^s_{i,j}, 1)$ per entry, and compute $\bar{X}_{i,j} = \frac{1}{\tau}\sum_{t=1}^{\tau} X^{(t)}_{i,j}$. We then cluster the items by performing the K-means algorithm with the vectors $(\bar{X}_{1j})_{j=1,\dots,d}, \dots, (\bar{X}_{nj})_{j=1,\dots,d}$. For each sparsity s, the budget T is turned by an oracle, with a grid search, so that the observed error rate after $\kappa = 5000$ runs is below 0.01. The resulting budgets are reported in Figure 5.1.

6. `https://github.com/grafmaxi/bandit_two_clusters`

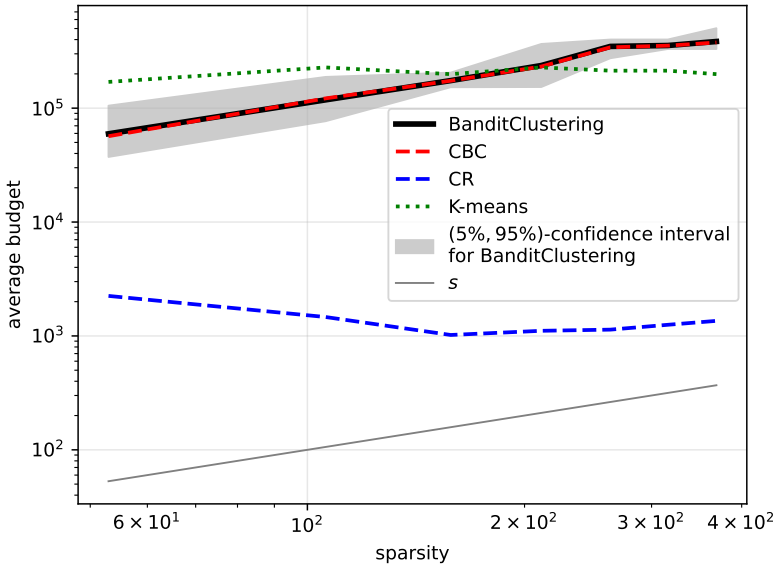


Figure 5.1 – Different budgets for Experiment 1, depending on the sparsity of Δ^s.

From Figure 5.1 we observe that in the sparse regime, our algorithm requires fewer observations than the uniform sampling approach used as baseline. Moreover, the budget required for `CR` appears to be mostly independent of the sparsity level s. In this setting, the overall sample complexity of `BanditClustering` is mainly driven by the cost of the `CBC` step, which grows approximately linearly with s. These empirical dependencies on s align with our theoretical results. The budget of `CR` in this case is, up to poly logarithmic factors, of order $d/\theta\|\Delta\|_2^2$, which is constant in s for fixed $\|\Delta\|_2$, while `CBC` requires a budget of order $n/(\Delta_i^s)^2 \sim n \cdot s$.

Experiment 2. In this experiment, we consider matrices of increasing size, with $n/100 = 1, 2, 5, 10, 20, 50$ and $d = 10 \cdot n$, so that the number of features grows proportionally with the number of items. For each value of n, we define the vector $\tilde{\Delta}^{(n)} \in \mathbb{R}^d$ as $\tilde{\Delta}^{(n)} = (\underbrace{5, \dots, 5}_{10 \text{ times}}, \underbrace{0, \dots, 0}_{d-10 \text{ times}}) \in \mathbb{R}^d$. We construct a matrix $M^{(n)}$ with $n/2$ rows equal to 0 and $n/2$ rows equal to $\tilde{\Delta}^{(n)}$, and we add Gaussian noise with unit variance. For each $\delta \in 0.8, 0.5, 0.2, 0.05$, we run `BanditClustering`$(\delta)$ over $\kappa = 5000$ independent trials. In Figure 5.2, we report the average budget required by `BanditClustering` for each configuration. For $\delta = 0.05$, we additionally report the 5th and 95th percentiles across the simulations.

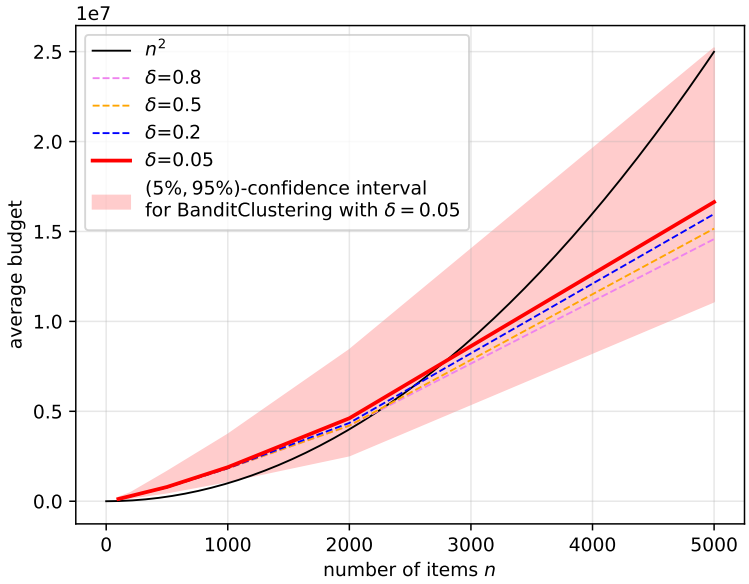


Figure 5.2 – Different budgets for Experiment 2, depending on the dimensionality of the problem n and $d = 10 \cdot n$.

If we allocate a budget smaller than $5n^2 = nd/2$ uniformly at random across the nd entries of M, then on average, each item i will have $d/2$ unobserved features. As both n and d increase, the probability that some item i is only sampled on coordinates j such that $\tilde{\Delta}_j^{(n)} = 0$ tends to one, rendering accurate clustering impossible. By contrast, Figure 5.2 shows that the budget required by our algorithm scales linearly with n. This matches the bounds from Corollary 5.3.2, which implies that when $d = 10 \cdot n$ and the parameters θ, s, and h are fixed, the total budget required (up to poly logarithmic factors) is of order n.

Experiment 3. In this third experiment, we compare our algorithm `BanditClustering` with the `Adaptive Clustering` algorithm introduced as Algorithm 2 in (Ariu et al., 2024). We fix the number of items to $n = 30$ and vary the number of features as $d_\gamma = 2^\gamma$ with $\gamma = 1, \dots, 5$. We set $\theta = 0.5$, and for each d_γ, we define $s_\gamma = d_\gamma/2$ features j such that $M_{i,j} = 0.25$ for items in the first cluster and $M_{i,j} = 0.75$ for items in the second cluster, yielding $h = 0.5$. For the remaining features, we set $M_{i,j} = 0.5$, independent of the item's cluster. Observations are sampled from Bernoulli distributions: $\nu_{i,j} = \text{Bern}(M_{i,j})$. We apply `BanditClustering` with $\delta = 0.8$ over $\kappa = 500$ runs. We then ran the `Adaptive Clustering` algorithm using the MATLAB code provided by (Ariu et al., 2024), under the same setting and a fixed budget of $T = 400{,}000$, also with $\kappa = 500$ runs. In Figure 5.3, we compare this fixed budget to the average budget used by `BanditClustering`, and report the corresponding error rates as a function of d_γ.

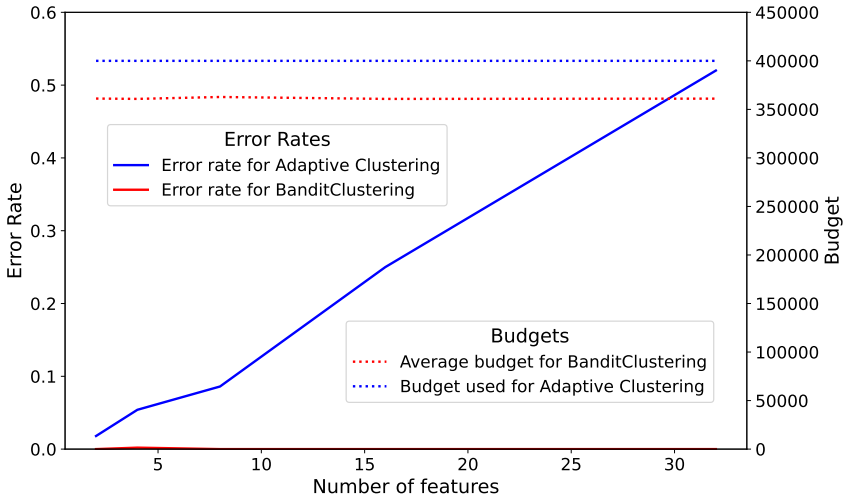


Figure 5.3 – Comparison of the performance of `BanditClustering` and `Adaptive Clustering` in Experiment 3, depending on the number of features d_γ.

As the number of features increases, we observe that the error of the fixed-budget algorithm `Adaptive Clustering` also increases. In contrast, the fixed-confidence algorithm `BanditClustering` maintains a consistently low error, and its average required budget remains largely unchanged. This behavior aligns with Corollary 5.3.2, as both $d_\gamma/\theta\|\Delta\|^2$ and n/h^2 are constant in our experimental setup. While our algorithm performs better in this specific scenario, it is important to note that the problem addressed in (Ariu et al., 2024) is more complex than the clustering task we consider. We believe their algorithm could be adapted to our setting; however, it remains unclear whether the influence of the number of features on the error rate can be mitigated.

Experiment 4. In this experiment we consider matrices with dimensions $n = d = 1000$. The gap vector Δ' is defined as

$$\Delta'_j = \begin{cases} 1.5 & \text{if } j \leq 100 \\ 0 & \text{else.} \end{cases},$$

The following simulations are run in $\kappa = 5000$ trials: For different values $\theta_\gamma = 2^\gamma/n$, $\gamma = 0, 1, \ldots, \lfloor \log_2(n) \rfloor - 1$, we run `CR` and `CBC` with confidence parameter $\delta = 0.4$. We also run `BanditClustering` with $\delta = 0.8$, analogously to experiment 1. The respective budgets are plotted in Figure 5.4. We also compare our procedure to the K-means algorithm. For that, we uniformly allocate budgets $T_\iota = \frac{2^\iota - 1}{2^{20}-1}(10^{10} - 10^6) + 10^6$, $\iota = 0, \ldots, 20$, by looking for each θ_γ for the smallest T_ι such that at most 0.01 (again, approximately the error rate of `BanditCLustering`(0.8) for all θ_γ) respective Monte Carlo iterations returned an incorrect cluster. We also illustrate these times in Figure 5.4.

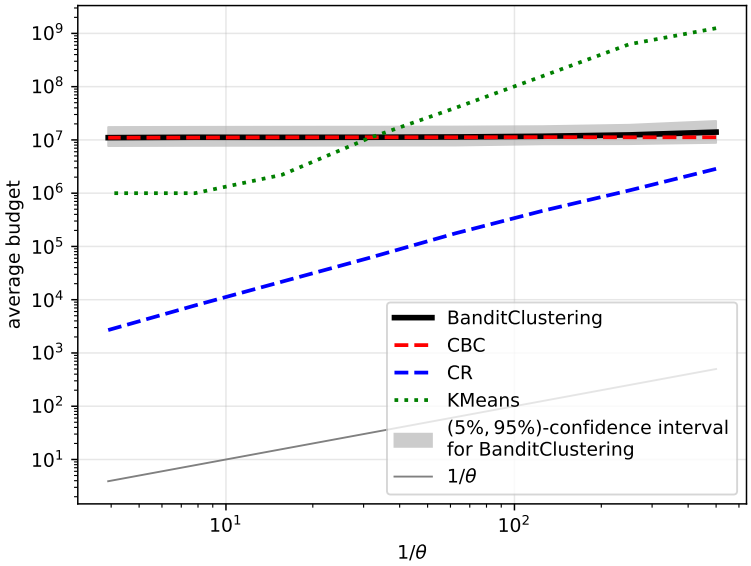


Figure 5.4 – Different budgets for Experiment 4, depending on θ_γ.

Again, one can see that the budget of `BanditClustering` is mainly spent on the `CBC` part of the algorithm. For small values of θ, our algorithm requires less budget than an equally accurate version of the K-means algorithm. We can see that the budget of `CBC` is not effected by θ, while the budget of `CR` seems to be almost proportional to $1/\theta$. Since n, d and Δ' are fixed, this is in line with the results in Corollary 5.3.2.

5.6 Extension to $K > 2$ clusters

In the main part of this Chapter, we focus on the case of two groups as it serves as an informative baseline for understanding the optimal trade-offs in clustering with bandit feedback. We introduce in this section an algorithm that extends our method to the general case where $K > 2$.

Building on the ideas of Algorithm 12, we aim to identify a set of representative items $(r_1, \dots, r_K) \in [n]^K$, with one item from each cluster. Given these representatives, the algorithm learns a discriminative feature for each pair of clusters, enabling perfect clustering in the general case with $K > 2$ groups. The algorithm is described in pseudocode in Algorithm 14.

Setting with $K > 2$ groups.

In this section, we assume that the items are partitioned into exactly $K \geqslant 2$ clusters, such that items within the same cluster share the same mean-vector. Specifically, there exists K centers in $\mathbb{R}^d$, denoted as $\mu_1, \dots, \mu_K$ such that, for any $i \in [n]$ then $M_{i,\cdot} \in \{\mu_1, \dots, \mu_K\}$. We further

assume that all the centers $\mu_1, \dots, \mu_K$ are pairwise distinct. For each $k \in [K]$, we denote as $G_k^* := \{i \in [n] \ ; \ M_{i,\cdot} = \mu_k\}$, and we assume that each cluster G_k^* is non-empty. As before, we consider sub-Gaussian noise and aim to identify the true partition $G^* = G_1^*, \dots, G_K^*$ in the δ-correct setting.

Note that the partition $G^* = G_1^* \sqcup \cdots \sqcup G_K^*$ is defined only up to a permutation of the cluster labels.

Description of Algorithm 14.

The algorithm uses the subroutines `CR` (Algorithm 11) and `CBC` (Algorithm 12), applied to subset of items.

For any $\delta \in (0,1)$, $r \in [n]$ and $G \subset [n]$, we denote as $\texttt{CR}(\delta, r; G)$ the call of Algorithm 11 where the search of representatives is restricted to a given set of items G (instead of $[n]$). We recall that $\texttt{CR}(\delta, r, G)$ is designed, to identify, if it exists, an item $s \in G$ such that $M_{s,\cdot} \neq M_{r,\cdot}$.

For any $\delta \in (0,1)$, $r \in [n]$, $s \in [n]$ and $G \subset [n]$, we write $\texttt{CBC}(\delta, r, s; G)$ for the run of `CBC` restricted to the set of items G. As we will prove later, with high probability $1-\delta$, $\texttt{CBC}(\delta, r, s; G)$ will output a partition of G into two groups, such that the items with mean $M_{r,\cdot}$ and $M_{s,\cdot}$ are well-separated. When calling $\texttt{CBC}(\delta, r, s; G)$, the items from the clusters of r and s within G are separated in two distinct sets, while other items might be split anywhere.

The algorithm takes as input the confidence parameter δ and the number of clusters K. It is important to note that, in the δ-correct setting, the number of clusters K must be known in advance. Indeed, consider the simpler case where there are only two items with means μ_1 and μ_2. Even for this simple problem, there is no finite time testing procedure that can decide between the hypotheses $\mu_1 = \mu_2$ (i.e., $K = 1$) and $\mu_1 \neq \mu_2$ (i.e., $K = 2$) without any prior knowledge on the separation $\mu_1 - \mu_2$.

First, we start with a partition $G^{(1)} = [n]$, where all items are grouped together, and we fix an arbitrary item $r_1 \in G^{(1)}$ as the first representative.

The algorithm proceeds in $K-1$ epochs, indexed by $e = 1, \dots, (K-1)$. In each epoch e, the algorithm identifies a new representative r_{e+1} and isolates all items sharing the same mean-vector $M_{r_{e+1},\cdot}$ into a new cluster.

At the beginning of the e-th epoch, the algorithm has access to a partition of $[n]$ into e groups $[n] = \cup_{k=1}^{e} \hat{G}_k^{(e)}$ together with e representatives $r_1, \dots, r_e$. If all the previous epochs were successful, then the representatives $r_1, \dots, r_e$ belong to different clusters, and the intermediate cluster $\hat{G}_k^{(e)}$ will contain all items with the same mean-vector as r_k. The remaining items (i.e., those from unrepresented clusters) may be mixed in the current partition $\hat{G}_1^{(e)}, \dots, \hat{G}_e^{(e)}$.

To identify a new representative, the algorithm calls `CR` with the representative r_k restricted to the group $\hat{G}_k^{(e)}$, for each $k = 1, \dots, e$, using confidence level δ_e. These e calls — denoted $\texttt{CR}(\delta_e, r_k; \hat{G}_k^{(e)})$ — are run in parallel. The first returned item from any successful call is selected as the new representative $re + 1$.

Next, for each $k \in [e]$, the algorithm runs $\texttt{CBC}(\delta_e, r_k, r_{e+1}; \hat{G}_k^{(e)})$ to split the group $\hat{G}_k^{(e)}$ based on the new representative. With high probability, r_k and r_{e+1} come from different clusters. In that case, `CBC` will divide $\hat{G}_k^{(e)}$ into two subgroups: $\hat{G}_k^{(e+1)}$, containing all items with mean $M_{r_k,\cdot}$,

and $\hat{R}_k^{(e+1)}$, containing all items with mean $M_{re+1,\cdot}$. We then define the new group $\hat{G}_{e+1}^{(e+1)} := \bigcup_{k=1}^{e} \hat{R}_k^{(e+1)}$, which should contain all items with mean $M_{r_{e+1},\cdot}$.

At the end of the $(K-1)$-th epoch, the partition $[n] = \cup_{k=1}^{K} \hat{G}_k^{(K)}$ is the output of the algorithm. With high probability, it should be exact.

Algorithm 14: K-`BanditClustering`

Input: confidence parameter $\delta > 0$, number of clusters K
Output: Clusters $\hat{G}_1, \dots, \hat{G}_K$

$\hat{G}_1^{(1)} \leftarrow [n]$;
Pick a first representative r_1 uniformly at random from $[n]$;
for $1 \leqslant e \leqslant K-1$ **do**
  $\delta_e \leftarrow \frac{\delta}{e(K-1)}$;
  Run in parallel $\texttt{CR}(\delta_e, r_k; G_k^{(e)})$ for $k \in [r]$ until one new representative r_{e+1} is identified.;
  for $k \in [e]$ **do**
    Call $\texttt{CBC}(r_k, r_{e+1}, \delta_e; G_k^{(e)})$ to cluster $G_k^{(e)}$ into two groups $\hat{G}_k^{e+1}$, $\hat{R}_k^{e+1}$ (with $r_k \in \hat{G}_k^{e+1}$);
  end
  Gather $\hat{G}_{e+1}^{e+1} = \cup_{k=1}^{e} \hat{R}_k^{e+1}$;
end
return $\hat{G}_1^{(K)}, \dots, \hat{G}_K^{(K)}$ partition of the items;

Theorem 5.6.1. *Let ν be an environment with n items. Assume that there exists $[n] = \bigsqcup_{k=1}^{K} G_k^*$ a partition of the items into K nonempty and disjoint groups such that all items in G_k^* share the same mean-vector μ_k. For any $k \in [K]$, denote as $\theta_k = \frac{|\{i \in [n] \,;\, M_{i,\cdot} = \mu_k\}|}{n}$ as the proportion of items with mean-vector μ_k.*

Define, for any $0 \neq \Delta \in \mathbb{R}^d$,

$$\tilde{H}(\Delta) := \min_{s \in [d]} \left[\left(\frac{d}{s} + n \right) \left(\frac{1}{\Delta_{(s)}^2} + 1 \right) \right] \quad .$$

For $\delta \in (0, 1/e)$, consider Algorithm 14 with entry δ and K.

With probability larger than $1 - \delta$, Algorithm 14 returns a partition $\hat{G}$ of $[n]$ equal to G^* (up to labelization of the clusters), with a budget of at most

$$\tilde{C} \log\left(\frac{1}{\delta}\right) \left[\sum_{k \in [K]} \max_{k' \neq k} \frac{Kd}{\theta_k \|\mu_k - \mu_{k'}\|^2} + \sum_{k \neq k'} \tilde{H}(\mu_k - \mu_{k'}) \right] \quad ,$$

where there exists a numerical constant C, such that $\tilde{C}$ is a logarithmic factor smaller than

$$C \cdot (\log\log(1/\delta) \vee 1)^4 \\ \cdot \log(dn)^5 \log(d)(\log_+ \log(d/\min_{k \neq k'} \|\mu_k - \mu'_k\|^2) \vee 1) \ .$$

The proof of Theorem 5.6.1 is postponed to Appendix 5.E. The proofs simply exploit the results obtained in the next section of this Appendix about subroutines `CR` and `CBC`.

Comments on Algorithm 14 and Theorem 5.6.1

In Algorithm 14, we extend our clustering approach to the general case with $K > 2$ clusters by reducing the problem to a sequence of binary classification tasks. This allows us to reuse the subroutines `CR` and `CBC`, originally designed for the case where there two groups, in a pipeline fashion. As shown in Theorem 5.6.1, this reduction yields a δ-correct algorithm for the general clustering problem.

The resulting sample complexity scales as K^2, since the algorithm performs one binary classification for each pair of clusters. This quadratic dependency is unavoidable in the worst case—for example, when the cluster means $\mu_1, \ldots, \mu_K$ are positioned in such a way that each pair of clusters must be treated independently. However, this approach may be suboptimal in general settings. For instance, if certain features allow simultaneous discrimination among all K clusters, then the sample complexity should not need to scale quadratically in K.

In other words, while our extension to $K > 2$ is straightforward and functional, it remains naive in terms of adaptivity. A more refined approach would aim to capture instance-dependent complexity, leveraging the joint geometry of the cluster centers $\mu_1, \ldots, \mu_K$ to potentially reduce the overall budget. Developing such adaptive strategies remains an open and interesting direction for future work.

5.7 Discussion

Comparison to other active clustering settings and batch clustering. In this work, we consider a bandit clustering setting where the learner can adaptively sample each item-feature pair. This contrasts with (Yang et al., 2024; Thuot et al., 2025; Yavas et al., 2025) where the authors have to sample all the features for each item and cannot focus on most relevant features. Rewriting their results in our setting, the optimal budget for the latter problem is, up to poly-logarithmic terms, of the order of

$$\frac{nd\log(1/\delta)}{\|\Delta\|^2} + \frac{d^{3/2}\sqrt{n\log(1/\delta)}}{\|\Delta\|^2} \ .$$

Comparing this with our main result (Theorem 5.3.1), we first observe that the ability to adaptively select features allows to remove the so-called high-dimensional terms $d^{3/2}\frac{\sqrt{n\log(1/\delta)}}{\|\Delta\|^2}$ that occurs when the number of features is large — $d \geq n\log(1/\delta)$. Second, the adaptive queries allow to drastically decrease the budget in situation where the vector Δ contains a few large

entries so that a few features are especially relevant to discriminate. To illustrate this, consider e.g., a setting as in Corollary 5.3.2 where $\Delta \in \{0, h\}^d$ takes s non-zero values where the partition is balanced so that $\theta = 1/2$. Then, our budget is of the order of

$$\log(1/\delta) \left[\frac{d}{sh^2} + \frac{n}{h^2} \right] ,$$

which represents a potential reduction by a factor $n \wedge \frac{d}{s}$ compared to (Yang et al., 2024; Thuot et al., 2025).

Extension to a larger number of groups. As discussed in Section 5.6, our algorithms can be used as subroutines to address the case where $K > 2$. We provide an algorithm that handles this extension, along with a (non-optimal) budget that scales with K^2. A key challenge in achieving optimality in this setting is determining whether the algorithm should focus on all $K(K-1)/2$ pairwise discriminative features or on a smaller, more informative subset. It is significantly more difficult to devise a strategy that adapts optimally to the relative positions of the centers of the K groups. We leave this question for future work.

Extension to heterogeneous groups. We also assumed throughout this work that all items within a group are perfectly similar, meaning their corresponding mean vectors μ_i are equal. This assumption could be relaxed by allowing the μ_i's within a group to be close, but not necessarily identical. For instance, suppose we have prior knowledge that, for any feature $j \in [d]$, the within-group variation satisfies, $\max_{g^*(i)=g^*(i')} |M_{i,j} - M_{i',j}| \leqslant c \min_{g^*(i)\neq g^*(i')} |M_{i,j} - M_{i',j}|$. If $c < 1/4$, our algorithm remains correct since the search for a single discriminative feature is still meaningful and enables classification. However, if the within-group heterogeneity becomes comparable to or larger than the intergroup differences in some features, our method could fail, and further investigation would be required. Note also that if $c = 1/2$, the problem becomes unidentifiable.

Appendix of Chapter 5

5.A Notation

To ease the reading, we gather the main notation below

— n number of items, d number of features

— $\mu_0 \neq \mu_1 \in \mathbb{R}^d$ feature vectors of the two groups

— $M_{i,\cdot} \in \{\mu_0, \mu_1\}$, $i \in [n]$ feature vector of item i

— M matrix with rows $(M_{i,\cdot})$ (with size $n \times d$)

— $g^* \in \{0,1\}^n$ true labels (fixing $g^*(1) = 0$)

— $X \sim \nu_{i,j}$ for $(i,j) \in [n] \times [d]$: $\mathbb{E}[X] = M_{i,j}$ with $X - M_{i,j}$ 1-sub-Gaussian

— $\delta \in (0,1)$ prescribed probability of error

— $\theta \coloneqq \frac{\sum_{i=1}^n \mathbb{1}(g^*(i)=0) \wedge \sum_{i=1}^n \mathbb{1}(g^*(i)=1)}{n}$ balancedness

— $\Delta \coloneqq \mu_1 - \mu_0 \neq 0$ gap vector

— $s^* \in \operatorname{argmax}_{s \in [d]} s \cdot \Delta_{(s)}^2$ effective sparsity

Moreover, as we repeatedly compare the entries of M with some row r_0 of our choice, we consider a fixed row r_0 for the following proofs and define

— $D_{i,j} \coloneqq M_{i,j} - M_{r_0,j}$, for $(i,j) \in [n] \times [d]$.

5.B Analysis of Algorithm 10

We analyze here the performance of `CSH`.

Lemma 5.B.1. *Consider $\delta \in (0,1)$, $s \in [d]$ and $h > 0$ such that $\left|\Delta_{(s)}\right| \geq h$. Consider $I \subset [n]$ and define the relative proportion of items in the second group as $\alpha = \frac{|\{i \in I;\, g^*(i)=1\}|}{|I|}$. Run Algorithm 10— `CSH`$(r_0, I, L, T)$ —with input r_0, I, L, T such that*

$$L = \left\lceil \log_2 \left(16 \frac{d}{\alpha s} \log \left(\frac{4 \log(8|I|d)}{\delta} \right) \right) \right\rceil \ , \tag{5.6}$$

$$T \geq 516 \frac{L^3 \cdot 2^L}{h^2} \vee 2^{L+1} L \ . \tag{5.7}$$

Then `CSH`(r_0, I, L, T) outputs a pair $(\hat{\imath}, \hat{\jmath})$ such that $|M_{\hat{\imath},\hat{\jmath}} - M_{r_0,\hat{\jmath}}| \geq h/2$ with probability $\geq 1 - \delta$.

Remark that for $I = [n]$, then $\alpha \geqslant \theta$. If $I = \{r_1\}$ where $g^*(r_1) \neq g^*(r_0)$, then $\alpha = 1$.

Throughout this section, we will prove Lemma 5.B.1 with $I = [n]$. The general result directly follows from the case where I contains all items. To see that, we just have to see that α is equal to the balancedness of the matrix $M|_I$ restricted to the rows in I, and we would replace n by $|I|$, and θ by α.

Therefore, we consider Algorithm 10 with input r_0, $I = [n]$, $L = \left\lceil \log_2 \left(16 \frac{d}{\theta s} \log \left(\frac{4 \log(8nd)}{\delta} \right) \right) \right\rceil$ and $T = 516 \frac{L^3 \cdot 2^L}{h^2} \vee 2^{L+1} L$. For simplicity in notation, we also fix $r_0 = 1$ for the proofs.

We recall that we use for the proofs the notation $D_{i,j} := M_{i,j} - M_{r_0,j} = M_{i,j} - M_{1,j}$ for any couple $(i,j) \in [n] \times [d]$ as the gap between the entries of M compared to the mean-vector of the fixed item $r_0 = 1$.

For the following proofs, define $\gamma := h/2L$, and for any halving step $l = l = 0, 1, \ldots, L$, we define U_l as the set of remaining entries (i,j) in S_l such that the gap $|D_{i,j}|$ exceeds $h - l\gamma$

$$U_l := \{(i,j) \in S_l : \ |D_{i,j}| \geq h - l\gamma\} \ .$$

Lemma 5.B.1 is a direct consequence of following statement:

Lemma 5.B.2. *With probability of at least $1 - \delta$, it holds*

$$\frac{|U_l|}{|S_l|} \geq 2^{-L+3} \log \left(\frac{4 \log(8nd)}{\delta} \right) \quad \forall \ l = 0, 1, \ldots, L \ .$$

Proof of Lemma 5.B.1. The first statement follows by Lemma 5.B.2. Indeed, at the last halving step, S_L contains only one pair of indices $(\hat{\imath}, \hat{\jmath})$. Lemma 5.B.2 implies that $U_L \subseteq S_L$ is nonempty with probability at least $1 - \delta$, so that $(\hat{\imath}, \hat{\jmath}) \in U_L$, that is, $D_{\hat{\imath},\hat{\jmath}} \geqslant h - L\gamma = h/2$. □

Proof of Lemma 5.B.2. We will prove via induction over l, that

$$\frac{|U_k|}{|S_k|} \geq 2^{-L+3} \log \left(\frac{4 \log(8nd)}{\delta} \right) \quad \forall \ k = 0, 1, \ldots, l$$

holds with probability at least

$$1 - (l+1) \left(\frac{\delta}{4 \log(8nd)} \right)^2 \ .$$

Let $\tilde{\delta} = \frac{\delta}{4 \log(8nd)}$. The statement follows then from

$$\begin{aligned}
(L+1) \cdot \tilde{\delta}^2 &\leq \left(2 \log \left(16 \frac{d}{\theta s} \log \left(\frac{1}{\tilde{\delta}} \right) \right) + 1 \right) \cdot \tilde{\delta}^2 \\
&\leq 3 \log \left(8 \frac{d}{\theta s} \right) \cdot \tilde{\delta}^2 + 2 \log \log \left(\frac{1}{\tilde{\delta}^2} \right) \cdot \tilde{\delta}^2 \\
&\leq \frac{3}{4} \frac{\delta^2}{\log(8nd)} + \tilde{\delta} \leq \delta \ ,
\end{aligned}$$

where we used that $\lceil \log_2(x) \rceil \leq 2 \log(x)$ for $x > 5$, and the last line is obtained by $8d/\theta s \leq 8nd$ and $2x \cdot \log\log(1/x) \leq \sqrt{x}$ for $x \in (0,1)$.

The base case $l = 0$ The initial set $S_0 = \{(i_1, j_1), \ldots, (i_{2^L}, j_{2^L})\}$ is constructed by picking 2^L entries uniformly at random (with replacement) from $[n] \times [d]$, as described in Line 1 in Algorithm 10. In the context of Lemma 5.B.2, the parameter α reduces to θ, since we consider

$I = [n]$ for the proof, and s, h are such that $|\Delta_{(s)} \geqslant h$. Consequently, the matrix M contains at least $\theta n \cdot s$ entries such that $|D_{i,j}| \geq h$. Then, the random variables

$$X_t^{(0)} \coloneqq \mathbb{1}\left(|D_{i_t,j_t}| \geq h\right), \quad t = 1, \ldots, 2^L$$

are i.i.d. Bernoulli random variables with $\mathbb{P}(X_t^{(0)} = 1) \geq \theta \frac{s}{d}$. In particular, we have

$$\mathbb{E}\left[\sum_{t=0}^{2^L} X_t^{(0)}\right] = 2^L \theta \frac{s}{d} \geq 16 \log\left(\frac{4\log(8nd)}{\delta}\right) \ .$$

Applying the second inequality in Lemma 5.G.1 (a standard Chernoff bound for Bernoulli distributions), we obtain:

$$\mathbb{P}\left(\sum_{t=0}^{2^L} X_t^{(0)} \leq 8\log\left(\frac{4\log(8nd)}{\delta}\right)\right) \leq \mathbb{P}\left(\sum_{t=0}^{2^L} X_t^{(0)} \leq \frac{1}{2}\mu_{(0)}\right) \leq \exp\left(-\frac{\mu_{(0)}}{8}\right) \leq \left(\frac{\delta}{4\log(8nd)}\right)^2 \ .$$

So we have

$$|U_0| = \sum_{t=0}^{2^L} X_t^{(0)} > 8\log\left(\frac{4\log(8nd)}{\delta}\right) = |S_0| 2^{-L+3} \log\left(\frac{4\log(8nd)}{\delta}\right)$$

with probability at least $1 - \left(\frac{\delta}{4\log(8nd)}\right)^2$.

Induction step: from l to $l+1$ Consider the event ξ_l, defined as

$$\frac{|U_k|}{|S_k|} \geq 2^{-L+3} \log\left(\frac{4\log(8nd)}{\delta}\right) \quad \forall\ k = 0, 1, \ldots, l \ .$$

We want to show

$$\mathbb{P}\left(\xi_l\right) \geq 1 - (l+1)\left(\frac{\delta}{4\log(8nd)}\right)^2 \quad \Rightarrow \quad \mathbb{P}\left(\xi_l\right) \geq 1 - (l+2)\left(\frac{\delta}{4\log(8nd)}\right)^2$$

Note that $\xi_{l+1} \subseteq \xi_l$, so showing

$$\mathbb{P}\left(\xi_{l+1} \mid \xi_l\right) \geq 1 - \left(\frac{\delta}{4\log(8nd)}\right)^2$$

suffices to conclude

$$\begin{aligned}\mathbb{P}\left(\xi_{l+1}\right) = \mathbb{P}\left(\xi_l\right) \cdot \mathbb{P}\left(\xi_{l+1} \mid \xi_l\right) &\geq \left(1 - (l+1)\left(\frac{\delta}{4\log(8nd)}\right)^2\right)\left(1 - \left(\frac{\delta}{4\log(8nd)}\right)^2\right) \\ &\geq 1 - (l+2)\left(\frac{\delta}{4\log(8nd)}\right)^2 \ .\end{aligned}$$

When we condition on the event ξ_l, this implies the condition

$$|U_l| \geq 2^{-L+3}|S_l| \log\left(\frac{4\log(8nd)}{\delta}\right) \ .$$

Recall that in line 5 of Algorithm 10, we first sample

$$X_{1,j}^{(1)}, \ldots, X_{1,j}^{(\tau_{l+1})} \sim^{\text{i.i.d.}} \nu_{1,j} \ , X_{i,j}^{(1)}, \ldots, X_{i,j}^{(\tau_{l+1})} \sim^{\text{i.i.d.}} \nu_{i,j}$$

for each $(i,j) \in S_l$ and store

$$\hat{D}_{i,j} = \frac{1}{\tau_{l+1}} \sum_{u=1}^{\tau_{l+1}} X_{i,j}^{(u)} - X_{1,j}^{(u)} \ .$$

Since we assumed that $X_{i,j}^{(u)} - M_{i,j} \in SG(1)$ and $X_{1,j}^{(u)} - M_{1,j} \in SG(1)$, this implies

$$\sum_{u=1}^{\tau_{l+1}} \left(X_{i,j}^{(u)} - X_{1,j}^{(u)} - D_{i,j}\right) \in SG(2\tau_{l+1}) \ ,$$

and we obtain

$$\mathbb{P}\left(\hat{D}_{i,j} - D_{i,j} \geq \gamma/2\right) = \left(\sum_{u=1}^{\tau_{l+1}} \left(X_{i,j}^{(u)} - X_{1,j}^{(u)} - D_{i,j}\right) \geq \tau_{l+1}\gamma/2\right) \leq \exp\left(-\frac{\tau_{l+1}\gamma^2}{16}\right) =: p_l \ , \quad (5.8)$$

and likewise

$$\mathbb{P}\left(D_{i,j} - \hat{D}_{i,j} \geq \gamma/2\right) \leq p_l \ ,$$

For $(i,j) \in U_l$, this implies that

$$\mathbb{P}\left(\left|\hat{D}_{i,j}\right| \leq h - (l+1/2)\gamma\right) \leq p_l \ .$$

So we can construct i.i.d. Bernoulli random variables $B_{i,j}$ with

$$\mathbb{P}(B_{i,j} = 1) = p_l$$

and

$$\left|\hat{D}_{i,j}\right| \leq h - (l+1/2)\gamma \quad \Rightarrow \quad B_{i,j} = 1$$

for $(i,j) \in U_l$. By Lemma 5.G.1 it follows (by letting $\kappa = \frac{1}{2p_l} - 1$)

$$\begin{aligned}
\mathbb{P}\left(\sum_{(i,j)\in U_l} B_{i,j} \geq |U_l|/2 \mid |U_l| = \eta\right) &\leq \exp\left(\kappa p_l \eta - (1+\kappa) p_l \log(1+\kappa)\right) \\
&\leq \exp\left(\eta\left(1/2 - p_l - \frac{\log(1/p_l) - \log(2)}{2}\right)\right) \\
&\leq \exp\left(\eta\left(1/2 - \exp(-\tau_{l+1}\gamma^2/16) - \frac{\tau_{l+1}\gamma^2/16 - \log(2)}{2}\right)\right) \\
&\leq \exp(-2^{l-2}\eta) \ .
\end{aligned}$$

The last inequality follows, since we assumed

$$T \geq 516 \cdot 2^L \cdot L^3/h^2 \vee 2^{L+1} L \geq 128 \cdot 2^L \cdot L/\gamma^2 \vee 2^{L+1} L \ ,$$

such that

$$\tau_{l+1} = \left\lfloor \frac{T}{2^{L-l+1} L} \right\rfloor \geq \left\lfloor 32 \frac{2^{l+1}}{\gamma^2} \vee 2^{l-1} \right\rfloor \geq 16 \frac{2^{l+1}}{\gamma^2}$$

(by $\lfloor x \rfloor \geq x/2$ for $x \geq 1$), from where we can conclude

$$1/2 - \exp(-\tau_{l+1}\gamma^2/16) - \frac{\tau_{l+1}\gamma^2/16 - \log(2)}{2} \leq -2^{l-2}$$

for $l \geq 0$. For

$$\eta \geq 2^{-L+3} |S_l| \log\left(\frac{4\log(8nd)}{\delta}\right) = 2^{-l+2} \log\left(\left(\frac{4\log(8nd)}{\delta}\right)^2\right) \ ,$$

this implies

$$\mathbb{P}\left(\sum_{(i,j)\in U_l} B_{i,j} \geq |U_l|/2 \mid |U_l| = \eta\right) \leq \left(\frac{\delta}{4\log(8nd)}\right)^2$$

and therefore

$$\mathbb{P}\left(\sum_{(i,j)\in U_l} B_{i,j} \geq |U_l|/2 \mid \xi_l\right) \leq \left(\frac{\delta}{4\log(8nd)}\right)^2 \tag{5.9}$$

Next, define

$$V_l \coloneqq \{(i,j) \in S_l : \ |D_{i,j}| < h - (l+1)\gamma\} \ .$$

Note that $S_{l+1} \setminus U_{l+1} \subseteq V_l$. So if

$$|V_l| < 2^{-L+3} |S_l| \log\left(\frac{4\log(8nd)}{\delta}\right) \ ,$$

this implies

$$\begin{aligned}|U_{l+1}| &= |S_{l+1}| - |S_{l+1} \setminus U_{l+1}| \\ &\geq |S_{l+1}| - |V_l| \\ &> |S_{l+1}| - 2^{-L+3}|S_l| \log\left(\frac{4\log(8nd)}{\delta}\right) \\ &= \left(1 - 2^{-L+2}\right)|S_{l+1}| \log\left(\frac{4\log(8nd)}{\delta}\right) \\ &\geq 2^{-L+3}|S_{l+1}| \log\left(\frac{4\log(8nd)}{\delta}\right) \ , \end{aligned} \tag{5.10}$$

since we have $L \geq \log_2(16) = 4$. Therefore, consider the nontrivial case

$$|V_l| \geq 2^{-L+3}|S_l| \log\left(\frac{4\log(8nd)}{\delta}\right) \ ,$$

Like before, we have from (5.8) that

$$\mathbb{P}\left(\left|\hat{D}_{i,j}\right| \geq h - (l+1/2)\gamma\right) \leq p_l$$

for $(i,j) \in V_l$. Note that we can again define Bernoulli random variables $C_{i,j}$ with

$$\mathbb{P}(C_{i,j} = 1) = p_l$$

and

$$|D_{i,j}| \geq h - (l+1/2)\gamma \quad \Rightarrow \quad C_{i,j} = 1$$

for all $(i,j) \in V_l$. We can again show that conditional $|V_l| = \eta$ with

$$\eta \geq 2^{-L+3}|S_l| \log\left(\frac{4\log(8nd)}{\delta}\right) \ ,$$

it holds

$$\sum_{(i,j)\in V_l} C_{i,j} \geq \eta/2 \tag{5.11}$$

with probability $1 - \left(\frac{\delta}{4\log(8nd)}\right)^2$.

Now if $\overline{\Delta}$ is the median of the $\hat{D}_{i,j}$, $(i,j) \in S_l$, it is either $\overline{\Delta} < h-(l+1/2)\gamma$ or $\overline{\Delta} \geq h-(l+1/2)\gamma$. In the case $\overline{\Delta} < h - (l+1/2)\gamma$, the bound (5.9) tells us that U_{l+1} contains at least half of the indices of U_l, in other words,

$$\frac{|U_{l+1}|}{|S_{l+1}|} \geq \frac{|U_l|}{2|S_{l+1}|} = \frac{|U_l|}{|S_l|} \ ,$$

with probability at least $1 - \delta$. In the case $\overline{\Delta} \geq h - (l+1/2)\gamma$, we either directly conclude the induction step from (5.10), or we know from (5.11) that the number of $(i,j) \in S_{l+1}$ with

$|D_{i,j}| \leq h - (l+1)\gamma$ is less than half the number of arms in V_l. Then,

$$\frac{|U_{l+1}|}{|S_{l+1}|} \geq 1 - \frac{|V_l|}{2|S_{l+1}|} = 1 - \frac{|V_l|}{|S_l|} = \frac{|U_l|}{|S_l|} \ ,$$

with probability at least $1 - \left(\frac{\delta}{4\log(8nd)}\right)^2$. Combining both cases yields the claim. □

5.C Analysis of Algorithm 11

Using the results of Lemma 5.B.1, we can now determine theoretical guarantees for Algorithm 11, $\texttt{CR}(\delta, r_0)$.

We present the individual guarantees offered by Algorithm 11 in the following proposition.

Proposition 5.C.1. *Let $\delta \in (0, 1/e)$. With probability larger than $1-\delta$, Algorithm 11— $\texttt{CR}(\delta, r_0)$ —returns an index r_1, such that it holds that $M_{r_1,\cdot} \neq M_{r_0,\cdot}$. Moreover, the total budget is upper bounded by*

$$\tilde{C} \cdot \log\left(\frac{1}{\delta}\right) \cdot \frac{d}{\theta}\left(\frac{1}{\|\Delta\|^2} + \frac{1}{s^*}\right) \ ,$$

where $\tilde{C}$ is a logarithmic factor smaller than

$$C \cdot (\log\log(1/\delta) \vee 1)^4 \cdot \log(dn)^5 \log(d)(\log_+ \log(1/\Delta^2_{(s^*)}) \vee 1) \ ,$$

with a numerical constant $C > 0$ and $\log_+(x) \coloneqq \log(x \vee 1)$ for $x \in \mathbb{R}$.

Proof of Proposition 5.C.1. Consider $s \in [d]$ and $k \geq 1$ minimal, such that for

$$L = \left\lceil \log_2\left(16\frac{d}{\theta s}\log\left(\frac{16\log(8nd)}{\delta}\right)\right)\right\rceil \ .$$

It holds

$$516\frac{L^3 2^L}{\Delta^2_{(s)}} \vee 2^{L+1}L \leq 2^k \ , \tag{5.12}$$

and

$$3714 \cdot \frac{\log(1/\delta) + \log_+\log\left(\frac{1}{\Delta^2_{(s)}}\right)}{\Delta^2_{(s)}} \vee 2 < 2^k \ . \tag{5.13}$$

The proof consists of three parts.

1. Algorithm $\texttt{CSH}(r_0, [n], L, 2^k)$ returns (r_1, j) with $|D_{r_1,j}| \geq \left|\Delta_{(s)}\right|/2$, with probability at least $1 - \delta/2$.
2. For $|D_{r_1,j}| \geq \left|\Delta_{(s)}\right|/2$, if we sample

$$X^{(1)}_{r_0,j}, \ldots, X^{(2^k)}_{r_0,j} \sim^{\text{i.i.d.}} \nu_{r_0,j} \quad \text{and} \quad X^{(1)}_{r_1,j}, \ldots, X^{(2^k)}_{r_1,j} \sim^{\text{i.i.d.}} \nu_{r_1,j} \ ,$$

it holds with probability of at least $1 - \frac{\delta}{0.3k^3}$ that

$$\left| \frac{1}{2^k} \sum_{t=1}^{2^k} X_{r_1,j}^{(t)} - X_{r_0,j}^{(t)} \right| > \sqrt{\frac{4}{2^k} \log\left(\frac{k^3}{0.15\delta}\right)} \ .$$

3. For any $k' \geq 1$, if $D_{i,j'} = 0$ and we sample

$$X_{r_0,j'}^{(1)}, \ldots, X_{r_0,j'}^{(2^{k'})} \sim^{\text{i.i.d.}} \nu_{r_0,j'} \quad \text{and} \quad X_{i,j'}^{(1)}, \ldots, X_{i,j'}^{(2^{k'})} \sim^{\text{i.i.d.}} \nu_{i,j'} \ ,$$

it holds with probability of at least $1 - \frac{\delta}{0.3k'^3}$ that

$$\left| \frac{1}{2^{k'}} \sum_{t=1}^{2^{k'}} X_{i,j'}^{(t)} - X_{r_0,j'}^{(t)} \right| \leq \sqrt{\frac{4}{2^{k'}} \log\left(\frac{k'^3}{0.15\delta}\right)} \ .$$

From point 1 and 2 one can conclude that Algorithm 11 terminates in the L^{th} step of the k^{th} iteration at the latest with probability at least $1 - \delta/2 - \delta/0.3k^3$. If it has not terminated before, by point 1, we obtain in line 4 (r_1, j) with $|D_{r_1,j}| \geq \left|\Delta_{(s)}\right|/2$ and by point 2, that for such (r_1, j), the algorithm terminates in line 6, returning r_1.

If the algorithm terminates for some $k' < k$ or in the k^{th} round, but for some other L', by line 6 this means that the algorithm returns some i such that for some j' it holds

$$\left| \frac{1}{2^{k'}} \sum_{t=1}^{2^{k'}} X_{i,j'}^{(t)} - X_{r_0,j'}^{(t)} \right| > \sqrt{\frac{4}{2^{k'}} \log\left(\frac{k'^3}{0.15\delta}\right)} \ .$$

Then, point 3 implies for each iteration k' and each L' we iterate over, that we do not return an index i with $M_{i,\cdot} = M_{r_0,\cdot}$, with probability at least $1 - \delta/0.3k^3$.

By the union bound, Algorithm 11 returns r_1 with $M_{r_1,\ldots} \neq M_{r_0,\ldots}$ with probability at least

$$1 - \delta/2 - \sum_{k' \leq k} \sum_{\substack{1 \leq L \leq L_{\max} \\ L2^L \leq 2^{k'}}} \delta/(0.3k'^3) \geq 1 - \delta/2 - \delta \cdot 0.3 \sum_{k \geq 1} \frac{1}{k^2} \geq 1 - \delta/2 - 0.3\frac{\pi^2}{6}\delta > 1 - \delta \ .$$

Finally, we are left with proving the three points.

Proof of 1 By Lemma 5.B.1 and inequality (5.12), calling $\texttt{CSH}(r_0, [n], L, 2^k)$ in line 4 of Algorithm 11 yields a pair (r_1, j) with $|D_{r_1,j}| \geq |\Delta_{(s)}|/2$ with probability at least $1 - \delta/2$.

Proof of 2 Note that for

$$\hat{D}_{r_1,j} \coloneqq \frac{1}{2^k} \sum_{t=1}^{2^k} X_{r_1,j}^{(t)} - X_{r_0,j}^{(t)} \ , \tag{5.14}$$

by an application of Hoeffding's inequality we know

$$\hat{D}_{r_1,j} \in \left[D_{r_1,j} - \sqrt{\frac{4}{2^k} \log\left(\frac{k^3}{0.15\delta}\right)}, D_{r_1,j} + \sqrt{\frac{4}{2^k} \log\left(\frac{k^3}{0.15\delta}\right)} \right] \tag{5.15}$$

with a probability of at least $1 - 0.3\delta/k^3$. Note that from inequality (5.13) we know by the monotonicity of $\log\log(x)/x$ for $x \geq e^2$ that

$$\begin{aligned}\frac{16}{2^k} \log\left(\frac{k^3}{0.15\delta}\right) &= \frac{16}{2^k}\left(\log(1/\delta) + 3\log\log(2^k) + 3\log\left(\frac{1}{\log(2)}\right) + \log(20/3)\right) \\ &\leq 80 \frac{\log(1/\delta) + \log\log\left(2^k\right)}{2^k} \ .\end{aligned}$$

We want to prove the bound

$$\frac{16}{2^k} \log\left(\frac{k^3}{0.15\delta}\right) \leq \Delta_{(s)}^2/4 \ . \tag{5.16}$$

Let us first consider the case $\Delta_{(s)}^2 \geq 1/e$. We can bound

$$\begin{aligned}80\frac{\log(1/\delta) + \log\log(2^k)}{2^k} &\leq 80\frac{\log(1/\delta) + \log(2^k)}{2^k} \\ &= 80\frac{\log(1/\delta) + \log(2^k\Delta_{(s)}^2) - \log(\Delta_{(s)}^2)}{2^k\Delta_{(s)}^2}\Delta_{(s)}^2 \\ &\leq 80\frac{2\log(1/\delta) + \log\log(2^k\Delta_{(s)}^2)}{2^k\Delta_{(s)}^2} \ .\end{aligned}$$

From inequality (5.12), we know

$$2^k\Delta_{(s)}^2 \geq 3714\log(1/\delta) \ ,$$

and we can therefore use that $x \mapsto \log(x)/x$ is decreasing for $x \geq e$ to obtain

$$\begin{aligned}80\frac{\log(1/\delta) + \log\log(2^k)}{2^k} &\leq 80\frac{2\log(1/\delta) + \log(3714\log(1/\delta))}{3714\log(1/\delta)}\Delta_{(s)}^2 \\ &\leq 80\frac{3 + \log(3714)}{3714}\Delta_{(s)}^2 \leq \Delta_{(s)}^2/4 \ .\end{aligned}$$

Next, consider the case $\Delta_{(s)}^2 \leq 1/e$. Then we know from inequality(5.13) that

$$2^k \geq 3714\frac{\log(1/\delta) + \log\log(\frac{1}{\Delta_{(s)}^2})}{\Delta_{(s)}^2} \ .$$

Because $x \mapsto \log\log(x)/x$ is decreasing for $x \geq e^2$, we can bound

$$80\frac{\log(1/\delta)+\log\log(2^k)}{2^k} \leq 80\frac{\log(1/\delta)+\log\log\left(3714\frac{\log(1/\delta)+\log\log(\frac{1}{\Delta_{(s)}^2})}{\Delta_{(s)}^2}\right)}{3714\left(\log(1/\delta)+\log\log\left(\frac{1}{\Delta_{(s)}^2}\right)\right)}\Delta_{(s)}^2 \ .$$

For $a, b \geq e$ it holds $\log\log(ab) \leq \log(2)+\log\log(a)+\log\log(b)$, so we can bound

$$\begin{aligned}
&\log\log\left(3714\frac{\log(1/\delta)+\log\log(\frac{1}{\Delta_{(s)}^2})}{\Delta_{(s)}^2}\right)\\
&\leq \log(2)+\log\log(\tfrac{1}{\Delta_{(s)}^2})+\log\log\left(3714\left(\log(1/\delta)+\log\log\left(\tfrac{1}{\Delta_{(s)}^2}\right)\right)\right)\\
&\leq \log(2\cdot 3714)+2\log\log\left(\tfrac{1}{\Delta_{(s)}^2}\right)+\log(1/\delta) \ .
\end{aligned}$$

This allows us to bound

$$\begin{aligned}
80\frac{\log(1/\delta)+\log\log(2^k)}{2^k} &\leq 80\frac{(2+\log(2\cdot 3714))\log(1/\delta)+3\log\log\left(\frac{1}{\Delta_{(s)}^2}\right)}{3714\left(\log(1/\delta)+\log\log\left(\frac{1}{\Delta_{(s)}^2}\right)\right)}\Delta_{(s)}^2\\
&\leq \frac{80(2+\log(2\cdot 3714))}{3714}\Delta_{(s)}^2 \leq \Delta_{(s)}^2/4 \ .
\end{aligned}$$

For $|D_{r_1,j}| \geq \left|\Delta_{(s)}\right|/2$, we have with high probability according to (5.15) that

$$\left|\hat{D}_{r_1,j}\right| \geq \sqrt{\frac{4}{2^k}\log\left(\frac{k^3}{0.15\delta}\right)} \ .$$

Proof of 3 Analogously to (5.14) and (5.15) we can use Hoeffding's inequality to show that for

$$\hat{D}_{i,j} \coloneqq \frac{1}{2^{k'}}\sum_{t=1}^{2^{k'}} X_{ij}^{(t)} - X_{1j}^{(t)}$$

it holds

$$\left|\hat{D}_{i,j}\right| \leq \sqrt{\frac{4}{2^{k'}}\log\left(\frac{k'^3}{0.15\delta}\right)}$$

with probability at least $1-\frac{\delta}{0.3k'^3}$.

Bounding the budget: First, we can bound

$$\begin{aligned}
L_{\max} &\leq 2\log\left(16nd\log\left(\frac{16\log(8nd)}{\delta}\right)\right) \\
&\leq 10\log\left(nd\log\left(\frac{16\log(8nd)}{\delta}\right)\right) \\
&\leq 10\left(\log(nd)+\log\log\left(\frac{16\log(8nd)}{\delta}\right)\right) \\
&\leq 10\left(\log(nd)+\log\left((nd)^4+\log(1/\delta)\right)\right) \\
&\leq 70\left(\log(nd)+\log\log(1/\delta)\right) \quad .
\end{aligned}$$

At the same time, we have that

$$\begin{aligned}
2^L &\leq 32\frac{d}{\theta s}\log\left(\frac{16\log(8nd)}{\delta}\right) \\
&\leq 32\frac{d}{\theta s}\left(\log\log(nd)+\log(64/\delta)\right) \\
&\leq 192\frac{d}{\theta s}\left(\log\log(nd)+\log(1/\delta)\right) \quad .
\end{aligned}$$

So, if we define $C \coloneqq 156\cdot 70^3\cdot 192$ and k^* the minimal $k\geq 1$ such that

$$\begin{aligned}
2^{k+1} \geq C\min_{s\in[d]}\Bigg(&\frac{\left(\log(nd)+\log\log(1/\delta)\right)^3\left(\log\log(nd)+\log(1/\delta)\right)d}{\theta s\Delta_{(s)}^2} \\
&+\frac{d\left(\log(nd)+\log\log(1/\delta)\right)\left(\log\log(nd)+\log(1/\delta)\right)}{\theta s} \\
&+\frac{d}{\theta s}\frac{\log(1/\delta)+\log_+\log(\frac{1}{\Delta_{(s)}^2})}{\Delta_{(s)}^2}\Bigg) \quad ,
\end{aligned}$$

we can see that by (5.12) and (5.13) Algorithm 11 terminates and returns r_1 such that $M_{r_1,\dots}\neq M_{r_0,\dots}$ with a probability of at least $1-\delta$. Moreover, on this event of high probability, the algorithm terminates after at most

$$\begin{aligned}
&\sum_{k=1}^{k^*}\sum_{1\leq L\leq L_{\max}:\ L\cdot 2^L\leq 2^{k+1}} 2\cdot 2^{k+1} \leq 8L_{\max}2^{k^*} \\
\leq C'\cdot L_{\max}\min_{s\in[d]}\Bigg(&\frac{\left(\log(nd)+\log\log(1/\delta)\right)^3\left(\log\log(nd)+\log(1/\delta)\right)d}{\theta s\Delta_{(s)}^2} \\
&+\frac{d\left(\log(nd)+\log\log(1/\delta)\right)\left(\log\log(nd)+\log(1/\delta)\right)}{\theta s} \\
&+\frac{d}{\theta s}\frac{\log(1/\delta)+\log_+\log(\frac{1}{\Delta_{(s)}^2})}{\Delta_{(s)}^2}\Bigg) \quad ,
\end{aligned}$$

by minimality of k^*, where $C' > 0$ is some numerical constant that might change. To obtain the claimed upper bound, note that by (5.2) we know $1/s^*\Delta^2_{(s^*)} \leq \log(2d)/\|\Delta\|_2^2$ and therefore the right-hand side of the above display can be bounded by

$$C''(\log(nd) + \log\log(1/\delta))^4 \log(nd) + \log(1/\delta)(\log_+ \log(1/\Delta^2_{s^*}) \vee 1)\frac{d}{\theta}\left(\frac{1}{\|\Delta\|^2} + \frac{1}{s^*}\right) \ ,$$

where $C'' > 0$ is some numerical constant. Reassembling the logarithmic terms yields the claim. □

5.D Analysis of Algorithm 12

Now, we prove the correctness and we upper-bound the budget of Algorithm 12.

Proposition 5.D.1. *Let $\delta \in (0, 1/e)$, let $r_1 \in [n]$ such that $M_{r_1,\cdot} \neq M_{r_0,\cdot}$. Then Algorithm 12—* `CBC`(δ, r_0, r_1) *—returns $\hat{g} = g^*$ (fixing arbitrary $g^*(r_0) = 0$), with probability at least $1 - \delta$, with a budget of at most*

$$\tilde{C} \cdot \log(1/\delta) \cdot \min_{s \in [d]} \left[\left(\frac{d}{s} + n\right)\left(\frac{1}{\Delta^2_{(s)}} + 1\right)\right] \ ,$$

where $\tilde{C}$ is a logarithmic factor smaller than

$$C \cdot (\log\log(1/\delta) \vee 1)^4 \cdot \log(d)^5 \cdot \log_+ \log\left(1/\Delta^2_{(\tilde{s})}\right) \ ,$$

with a numerical constant $C > 0$, and $\tilde{s} = \lceil d/n \rceil \wedge |\{j \in [d] \, , \Delta_j \neq 0\}|$.

The proof of Proposition 5.D.1 does not differ much from the proof of Proposition 5.C.1. Again, we have to bound the time when `CSH` returns an index pair for which the stopping condition is fulfilled with high probability. The main difference is, that we also need a guarantee for correct clustering using these indices, which also leads to a change of the stopping rule.

Proof. Consider $s \in [d]$ and $k \in \mathbb{N}$, $k > \log_2(n)$ minimal, such that for

$$L = \left\lceil \log_2\left(16\frac{d}{s}\log\left(\frac{16\log(8d)}{\delta}\right)\right)\right\rceil \ .$$

It holds

$$516\frac{L^3 2^L}{{\Delta_{(s)}}^2} \vee 2^{L+1}L \leq 2^k \ , \tag{5.17}$$

and

$$34423 \cdot \frac{\left(\log(1/\delta) + \log_+ \log\left(\frac{1}{\Delta^2_{(s)}}\right) + \log n\right) \cdot n}{{\Delta_{(s)}}^2} \vee 2n \leq 2^k \ . \tag{5.18}$$

The proof relies on the two following facts.

1. First, $\texttt{CSH}(r_0, [n], L, 2^k)$ returns (r_1, j) with $|D_{r_1,j}| \geq \left|\Delta_{(s)}\right|/2$, with probability at least $1 - \delta/2$.

2. Then, we have that jointly for all iterations $k' \geq 1$ and $1 \leq L \leq \tilde{L}_{\max}$ with $2^L L \leq 2^{k+1}$, for some $j' \in [d]$ (chosen each time in line 4) and all $1 < i \leq n$, when we draw

$$X^{(1)}_{r_0,j'}, \ldots, X^{(\lfloor 2^{k'}/n \rfloor)}_{r_0,j'} \sim^{\text{i.i.d.}} \nu_{r_0,j'} \quad \text{and} \quad X^{(1)}_{i,j'}, \ldots, X^{(\lfloor 2^{k'}/n \rfloor)}_{i,j'} \sim^{\text{i.i.d.}} \nu_{i,j'}$$

holds

$$\left| \sum_{t=1}^{\lfloor 2^{k'}/n \rfloor} \left(X^{(t)}_{i,j'} - X^{(t)}_{1,j'} - D_{i,j'} \right) \right| \leq \sqrt{4 \cdot \lfloor 2^{k'}/n \rfloor \log\left(\frac{nk'^3}{0.15\delta} \right)} \ ,$$

uniformly with probability at least $1 - \delta/2$.

Point 1 is a direct consequence of Lemma 5.B.1 and (5.17). Point 2 follows directly from Hoeffding's inequality and a union bound over all $k \geq 1$, $L \leq L_{\max}$ such that $L2^L \leq 2^{k+1}$ and $i \in [n]$. Indeed, each inequality for itself holds with probability at least $1 - 0.3\delta/nk'^3$, so the intersection must hold with a probability of at least

$$1 - \sum_{k' \geq 1} \sum_{\substack{1 \leq L \leq L_{\max} \\ L2^L \leq 2^{k'+1}}} \sum_{i=2}^{n} 0.3\delta/nk'^3 \geq 1 - 0.3\delta \sum_{k' \geq 1} \frac{1}{k'^2} \geq 1 - \delta/2 \ .$$

We will prove that, with probability at least $1 - \delta$, Algorithm 12 terminates at the latest in the L^{th} round of the k^{th} iteration. Moreover, it clusters correctly. This holds on the intersection of the high probability events of point 1 and 2 which we will call ξ_{cbc}.

Algorithm 12 terminates at the latest in the k^{th} iteration Assume we are on ξ_{cbc}. By point 1, we know that at round L of iteration k it holds $|D_{r_1,j}| \geq \left|\Delta_{(s)}\right|/2$ for j obtained in line 4. We want to prove

$$\frac{64}{\left\lfloor \frac{2^k}{n} \right\rfloor} \log\left(\frac{nk^3}{0.15\delta} \right) \leq \Delta^2_{(s)}/4 \ . \tag{5.19}$$

Note that by (5.18), it holds

$$\begin{aligned} \frac{64}{\left\lfloor \frac{2^k}{n} \right\rfloor} \log\left(\frac{nk^3}{0.15\delta} \right) &\leq \frac{128n}{2^k} \left(\log n + 3 \log\log 2^k + \log(20/3) + \log(1/\delta) \right) \\ &\leq c_1 n \frac{\log(1/\delta) + \log n + \log\log 2^k}{2^k} \ . \end{aligned}$$

Again, consider first the case $\Delta_{(s)}^2 \geq 1/e$. In this case, we know from (5.18) that

$$c_2 \left(\log(1/\delta) + \log n\right) \cdot n \leq 2^k \Delta_{(s)}^2 \ .$$

For presentation purpose, we write $c_1 = 640$ and $c_2 = 34423$.

We can use that $x \mapsto \log(x)/x$ is decreasing for $x \geq e$ and obtain

$$\begin{aligned}
c_1 n \frac{\log(1/\delta) + \log n + \log\log(2^k)}{2^k} &\leq c_1 n \frac{\log(1/\delta) + \log n + \log(2^k)}{2^k} \\
&= c_1 n \frac{\log(1/\delta) + \log n + \log(2^k \Delta_{(s)}^2) - \log(\Delta_{(s)}^2)}{2^k \Delta_{(s)}^2} \Delta_{(s)}^2 \\
&\leq c_1 n \frac{2\log(1/\delta) + \log n + \log(2^k \Delta_{(s)}^2)}{2^k \Delta_{(s)}^2} \Delta_{(s)}^2 \\
&\leq \frac{c_1}{c_2} \cdot \frac{2\log(1/\delta) + \log n + \log\left(c_2 n \log(1/\delta) + \log n\right)}{\log(1/\delta) + \log n} \Delta_{(s)}^2 \\
&\leq \frac{c_1}{c_2} \cdot \frac{2\log(1/\delta) + 2\log n + \log c_2 + \log(\log(1/\delta) + \log n)}{\log(1/\delta) + \log n} \Delta_{(s)}^2 \\
&\leq \frac{c_1 \cdot (3 + \log(c_2))}{c_2} \Delta_{(s)}^2 \leq \Delta_{(s)}^2/4 \ .
\end{aligned}$$

This proves (5.19) in the case $\Delta_{(s)}^2 \geq 1/e$.

Consider $\Delta_{(s)}^2 < 1/e$. Then, by (5.18), we know

$$2^k \geq c_2 n \frac{\log(1/\delta) + \log n + \log\log\left(\frac{1}{\Delta_{(s)}^2}\right)}{\Delta_{(s)}^2} \ .$$

We can apply that $x \mapsto \log\log(x)/x$ is decreasing for $x \geq e^2$ and obtain

$$c_1 n \frac{\log(1/\delta) + \log n + \log\log 2^k}{2^k} \leq \frac{c_1}{c_2} \frac{\log(1/\delta) + \log n + A}{\log(1/\delta) + \log n + \log\log\left(\frac{1}{\Delta_{(s)}^2}\right)} \Delta_{(s)}^2 \ ,$$

where A is the following,

$$\begin{aligned}
A &= \log\log\left(c_2 n \frac{\log(1/\delta) + \log n + \log\log\left(\frac{1}{\Delta_{(s)}^2}\right)}{\Delta_{(s)}^2}\right) \\
&\leq \log(2) + \log\log\left(\tfrac{1}{\Delta_{(s)}^2}\right) + \log\log\left(c_2 n\left(\log(1/\delta) + \log n + \log\log\left(\tfrac{1}{\Delta_{(s)}^2}\right)\right)\right) \\
&\leq \log(2\cdot c_2) + \log\log\left(\tfrac{1}{\Delta_{(s)}^2}\right) + \log(n) + \log\left(\log(1/\delta) + \log n + \log\log\left(\tfrac{1}{\Delta_{(s)}^2}\right)\right) \\
&\leq (\log(2\cdot c_2)+1)\log(1/\delta) + 2\log\log\left(\tfrac{1}{\Delta_{(s)}^2}\right) + 2\log(n) \ ,
\end{aligned}$$

where we used $\log\log(a\cdot b) \leq \log(2) + \log\log(a) + \log\log(b)$ for $a, b \geq e$. Thus, it holds

$$\begin{aligned}
c_1 n \frac{\log(1/\delta) + \log n + \log\log(2^k)}{2^k} &\leq \frac{c_1}{c_2} \frac{(2+\log(2\cdot c_2))\log(1/\delta) + 2\log n + 2\log\log\left(\frac{1}{\Delta_{(s)}^2}\right)}{\log(1/\delta) + \log n + \log\log(\frac{1}{\Delta_{(s)}^2})} \Delta_{(s)}^2 \\
&\leq \frac{c_1(2+\log(2\cdot c_2))}{c_2}\Delta_{(s)}^2 \leq \Delta_{(s)}^2/4 \ ,
\end{aligned}$$

which proves (5.19).

Inequality (5.19) implies

$$|D_{r_1,j}| \geq \left|\Delta_{(s)}\right|/2 \geq 4\cdot\sqrt{\frac{4}{\left\lfloor\frac{2^k}{n}\right\rfloor}\log\left(\frac{nk^3}{0.15\delta}\right)}$$

and by points 1 and 2 we have a guarantee that

$$\left|\hat{D}_{r_1,j}\right| \geq 3\cdot\sqrt{\frac{4}{\left\lfloor\frac{2^k}{n}\right\rfloor}\log\left(\frac{nk^3}{0.15\delta}\right)} \ .$$

By line 8 of Algorithm 12, this is sufficient for the algorithm to terminate after the L^{th} round of iteration k.

Algorithm 12 clusters correctly. Consider the first $k' \in \mathbb{N}$ with $k' > \log_2(n)$ such that for the samples

$$X_{r_0,j}^{(1)}, \ldots, X_{r_0,j}^{(\lfloor 2^{k'}/n\rfloor)} \sim^{\text{i.i.d.}} \nu_{r_0,j} \quad \text{and} \quad X_{r_1,j}^{(1)}, \ldots, X_{r_1,j}^{(\lfloor 2^{k'}/n\rfloor)} \sim^{\text{i.i.d.}} \nu_{r_1,j}$$

we have that

$$\frac{1}{\lfloor 2^{k'}/n \rfloor}\left|\sum_{t=1}^{\lfloor 2^{k'}/n \rfloor} X_{r_1,j}^{(t)} - X_{r_0,j}^{(t)}\right| > 3 \cdot \sqrt{\frac{4}{\lfloor \frac{2^{k'}}{n} \rfloor} \log\left(\frac{nk'^3}{0.15\delta}\right)} \ .$$

Then by line 8, we know that after completing the iteration Algorithm 12 terminates. From point 2 we know that on ξ_{cbc} it holds

$$|D_{r_1,j}| > 2 \cdot \sqrt{\frac{4}{\lfloor \frac{2^{k'}}{n} \rfloor} \log\left(\frac{nk'^3}{0.15\delta}\right)} \ .$$

So if for each $i \geq 2$ we sample again

$$X_{r_0,j}^{(1)}, \ldots, X_{r_0,j}^{(\lfloor 2^{k'}/n \rfloor)} \sim^{\text{i.i.d.}} \nu_{i,j} \quad \text{and} \quad X_{i,j}^{(1)}, \ldots, X_{i,j}^{(\lfloor 2^{k'}/n \rfloor)} \sim^{\text{i.i.d.}} \nu_{i,j} \ ,$$

then for the averages holds again by point 2 that

$$\frac{1}{\lfloor 2^{k'}/n \rfloor}\left|\sum_{t=1}^{\lfloor 2^{k'}/n \rfloor} X_{i,j}^{(t)} - X_{r_0,j}^{(t)}\right| > \sqrt{\frac{4}{\lfloor \frac{2^{k'}}{n} \rfloor} \log\left(\frac{nk'^3}{0.15\delta}\right)}$$

if and only if $D_{i,j} \neq 0$. So on ξ_{cbc}, the labeling in line 12 yields to a perfect clustering $\hat{g} = g^*$.

Bounding the budget:

Similar to the proof of Proposition 5.C.1, we can bound

$$\tilde{L}_{\max} \leq 70(\log(d) + \log\log(1/\delta)) \ ,$$

and

$$2^L \leq 192\frac{d}{s}(\log\log d + \log(1/\delta)) \ .$$

Defining $C \coloneqq 156 \cdot 70^3 \cdot 192$ and letting k^* being minimal such that

$$\begin{aligned} 2^{k+1} \geq C \min_{s\in[d]} \Bigg(& \frac{(\log d + \log\log(1/\delta))^3(\log\log d + \log(1/\delta))d}{s\Delta_{(s)}^2} \\ & + \frac{(\log d + \log\log(1/\delta))(\log\log d + \log(1/\delta))d}{s} \\ & + \frac{\left(\log(1/\delta) + \log_+\log\left(\frac{1}{\Delta_{(s)}^2}\right) + \log n\right) \cdot n}{\Delta_{(s)}^2} \Bigg) \end{aligned}$$

we know from (5.17) and (5.18) that with probability at least $1-\delta$, Algorithm 12 terminates and clusters correctly. Moreover, on the same event, it spends a budget of at most

$$\sum_{k=1}^{k^*} \sum_{1\leq L\leq \tilde{L}_{\max}:\ L\cdot 2^L\leq 2^{k+1}} 2\cdot 2^{k+1} \leq 8\tilde{L}_{\max}2^{k^*}$$

$$\leq C'(\log d+\log\log(1/\delta))\min_{s\in[d]}\left[\left(\frac{(\log d+\log\log(1/\delta))^3(\log\log d+\log(1/\delta))}{\Delta_{(s)}^2}+1\right)\frac{d}{s}\right.$$
$$\left.+\left(\frac{\log(1/\delta)+\log_+\log\left(\frac{1}{\Delta_{(s)}^2}\right)+\log n}{\Delta_{(s)}^2}+1\right)n\right] ,$$

where $C>0$ is a numerical constant. Inserting $\tilde{s}$ in the right-hand side and gathering the logarithmic terms like in the proof of Proposition 5.C.1 yields the claim. □

5.E Analysis of Algorithm 14

Come-back on sub-routines `CR` and `CBC`.

We can reformulate Proposition 5.C.1 in the following corollary.

Corollary 5.E.1. *Let $\delta\in(0,1/e)$, $r\in[n]$ and $G\subset[n]$. There exists an event of probability larger than $1-\delta$ such that*

1. *If $M_{i,\cdot}=M_{r,\cdot}$ $\forall i\in G$, then $\texttt{CR}(\delta,r,G)$ does not stop.*
2. *If $\exists i\in G$ such that $M_{i,\cdot}\neq M_{r,\cdot}$, then $\texttt{CR}(\delta,r,G)$ returns an item s such that $M_{r,\cdot}\neq M_{s,\cdot}$, with a budget T smaller than*

$$\tilde{C}\log\left(\frac{1}{\delta}\right)\min_{h>0}\left[\frac{d|G|}{|\{(i,j)\in G\times[d]\,;\,|M_{r,j}-M_{i,j}|\geqslant h\}|}\left(\frac{1}{h^2}+1\right)\right] ,$$

with $\tilde{C}$ a poly-logarithmic term defined in Proposition 5.C.1.

Proposition 5.D.1 can be formulated as follows:

Corollary 5.E.2. *Let $\delta\in(0,1/e)$, $r\in[n]$, $s\in[n]$ and $G\subset[n]$. Assume that $M_{r,\cdot}\neq M_{s,\cdot}$, then there exists an event of probability larger than $1-\delta$ such that $\texttt{CBC}(\delta,r,s;G)$ outputs a partition of G, $G=R\sqcup S$ with a budget T verifying the two following points.*

1. *$\{i\in G\,;\,M_{i,\cdot}=M_{r,\cdot}\}\subset R$ and $\{i\in G\,;\,M_{i,\cdot}=M_{s,\cdot}\}\subset S$*
2. *The budget T is smaller than*

$$\tilde{C}\log\left(\frac{1}{\delta}\right)\min_{h}\left[\left(\frac{d}{|\{j\in[d]\,;|M_{r,j}-M_{s,j}|\geqslant h\}|}+n\right)\left(\frac{1}{h^2}+1\right)\right] ,$$

with $\tilde{C}$ a poly-logarithmic term defined in Proposition 5.D.1.

Proof of Theorem 5.6.1. We use the notation from the pseudocode Algorithm 14 and from Section 5.6. The correction of the algorithm is a direct consequence of the following lemma, which states that, with high probability, all epochs behave as expected. The bound on the total budget given in Theorem 5.6.1 follows directly by summing the sample complexities of all `CR` and `CBC` calls across the $K-1$ epochs. These individual complexities are provided in Corollary 5.E.1 and Corollary 5.E.2.

Lemma 5.E.3. *There exists an event of probability larger than $1-\delta$ such that for each epoch $1 \leqslant e \leqslant K-1$,*

1. *Epoch e terminates using a finite budget.*
2. *The item r_{e+1} selected is a new representative: $M_{r_{e+1},\cdot} \notin \{M_{r_1,\cdot}, \dots, M_{r_e,\cdot}\}$*
3. *For all $k \in [e+1]$, $\{i \in [n];\ M_{i,\cdot} = M_{r_k,\cdot}\} \subset \hat{G}_k^{(e+1)}$*

Let $\mathcal{E}$ denote the event that, for all epochs $e = 1, \dots, K-1$, each call to `CR` and `CBC` behaves correctly — i.e., satisfies Points 1 and 2 of Corollary 5.E.2 and Corollary 5.E.1. In epoch e, the algorithm makes e calls to `CR` and e calls to `CBC`, each with a confidence level $\delta_e = \frac{\delta}{e(K-1)}$. By a union bound over all calls across all epochs, the probability of event $\mathcal{E}$ is at least $1-\delta$.

We prove by induction on $e \in [K-1]$ that the three points of Lemma 5.E.3 hold on $\mathcal{E}$.

Initially, we have a trivial partition $G^{(1)} = [n]$, and a representative r_1 is arbitrarily selected. Points 1–3 trivially hold at $e = 1$.

Assume that, at epoch e, $(r_1, \dots, r_e)$ are representatives from e distinct clusters, and for each $k \in [e]$, the set $\hat{G}_k^{(e)}$ contains all items with mean $M_{r_k,\cdot}$.

Since $e < K$, there exists at least one cluster not yet represented, and therefore, there exists some $i \in \hat{G}_k^{(e)}$ and some $k \in [e]$ such that $M_{i,\cdot} \neq M_{r_k,\cdot}$. On event $\mathcal{E}$, $\texttt{CR}(\delta_e, r_k; G_k^{(e)})$ will return an item in finite time, and Line 5 of the corresponding algorithm terminates. Denote r_{e+1} the item returned from this call. Then, by Corollary 5.E.1, we know that $M_{r_{e+1},\cdot} \neq M_{r_k,\cdot}$.

Moreover, for $k' \neq k$, $G_{k'}^{(e)}$ contains all items with mean $M_{r'_k,\cdot}$ so that $M_{r_{e+1},\cdot}$ is not in the set $\{M_{r_1,\cdot} \dots, M_{r_e,\cdot}\}$, and Point 2 also holds for the e-th epoch. Since all calls terminate in finite time on $\mathcal{E}$, Point 1 also holds.

Now, for each $k \in [e]$, $M_{r_{e+1},\cdot} \neq M_{r_k,\cdot}$, so that on $\mathcal{E}$, the partition of $\hat{G}_k^e$ into two groups $\hat{G}_k^{e+1} \sqcup \hat{R}_k^{e+1}$ will perfectly separate the items with mean $M_{r_k,\cdot}$ and $M_{r_{e+1},\cdot}$. By assumption, the partial cluster $\hat{G}_k^e$ already contains all items with mean $M_{r_k,\cdot}$, so $\{i \in [n]\ ;\ M_{i,\cdot} = M_{r_k,\cdot}\} \subset \hat{G}_k^e$. Similarly, as $\hat{G}_1^e, \dots, G_e^{(e)}$ is a partition of $[n]$, any item i with mean $M_{r_{e+1},\cdot}$ will be set in one of the sets $\hat{R}_k^{e+1}$ so that $\{i \in [n]\ ;\ M_{i,\cdot} = M_{r_{e+1},\cdot}\} \subset \cup_{k=1}^e \hat{R}_k^{e+1} = \hat{G}_{e+1}^{e+1}$. Thus, Point 3 also holds for epoch e.

By induction, all three points in the lemma hold for all $e \in [K-1]$ on event $\mathcal{E}$, which concludes the proof. □ □

5.F Proof of the lower bounds

The lower bound in Theorem 5.4.1 consists of two terms, which we prove separately. In the proofs, we use $T_{i,j}$ as the number of time a procedure selects the pair $(i,j) \in [n] \times [d]$.

Lemma 5.F.1. *The $(1-\delta)$-quantile of the budget of any δ-correct algorithm $\mathcal{A}$ is bounded as follows*

$$\max_{\tilde{\nu}\in\mathcal{E}_{\mathrm{per}}(M)} \mathbb{P}_{\tilde{\nu},\mathcal{A}}\left(T \geqslant \frac{2d}{\theta\|\Delta\|_2^2}\log\frac{1}{6\delta}\right) \geqslant \delta \ . \tag{5.20}$$

Proof of Lemma 5.F.1. Fix an algorithm $\mathcal{A}$, and let $\mathcal{E}_{\mathrm{per}}(M)$ denote the set of Gaussian environments obtained by permuting the rows and columns of M. For the purpose of the proof, we define $\mathbb{P}_{\sigma,\tau}$ as the probability distribution induced by the interaction between algorithm $\mathcal{A}$ and the environment defined in (5.4), where σ and τ are permutations of the rows and columns of M, respectively.

We permute the rows of M to reflect the fact that the learner has no access to the label vector g^*. In addition, we permute the columns of M to account for the algorithm's ignorance of the structure of the gap vector Δ, in particular, the identity of the feature with the largest gap.

Without loss of generality, we assume that $\mu_0 = \mathbf{0}$ and $\mu_1 = \Delta$, with the group associated with mean vector Δ being the smaller of the two.

Define χ as the smallest integer such that for all permutations σ and τ of $1,\dots,n$ and $1,\dots,d$, respectively, the following inequality holds:

$$\mathbb{P}_{\sigma,\tau}(T > \chi) \leqslant \delta \ . \tag{5.21}$$

Our goal is to derive a lower bound on χ. Intuitively, we show that for small χ, there exists a permutation of M for which it is impossible to detect a nonzero entry.

Introduce $\mathbb{P}_0$ as the probability distribution induced by $\mathcal{A}$, in an environment where all items belong to a single cluster, i.e., each $X_t \sim \mathcal{N}(0,1)$. We will prove that under this "null" environment, the algorithm $\mathcal{A}$ requires more than χ samples with probability at least $1-2\delta$.

To this end, consider an environment $\nu(g,\mu)$ consisting of two clusters with means $\mathbf{0}$ and μ, and let $\mu \to 0$. Since $\mathcal{A}$ is δ-correct, there exist two distinct partitions $g \neq g'$ and an event A such that

$$\mathbb{P}_{\nu(g,\mu)}(A, T \leqslant \chi) + \mathbb{P}_{\nu(g',\mu)}(A^c, T \leqslant \chi) \leqslant 2\delta \ .$$

For example, take $g(1)=0$, $g(2)=1$, and $g'(1)=0$, $g'(2)=0$, the event $\{\hat{g}(1)=\hat{g}(2)\}$ suffices. Then, conditionally on $T \leqslant \chi$, $\mathbb{P}_{\nu(g,\mu)}$ and $\mathbb{P}_{\nu(g',\mu)}$ converge in total variation to $\mathbb{P}_0$ as $\mu \to 0$. Consider an environment $\nu(g,\mu)$ consisting of two clusters with means $\mathbf{0}$ and μ, and let $\mu \to 0$.

$$\mathbb{P}_0(T \leqslant \chi) \leqslant 2\delta \ . \tag{5.22}$$

Applying the Bretagnolle–Huber inequality (see Lattimore and Szepesvári, 2020, Thm. 14.2), and combining (5.21) and (5.22), we obtain

$$\frac{1}{2}\exp\left(-\mathrm{KL}(\mathbb{P}_0,\mathbb{P}_{\sigma,\tau})\right) \leqslant \mathbb{P}_0(T \leqslant \chi) + \mathbb{P}_{\sigma,\tau}(T > \chi) \leqslant 3\delta \ ,$$

which implies

$$\log\frac{1}{6\delta} \leqslant \mathrm{KL}(\mathbb{P}_0,\mathbb{P}_{\sigma,\tau}) \ . \tag{5.23}$$

Next, using the decomposition of KL divergence for bandit models (see Lattimore and Szepesvári, 2020, Lemma. 15.1), and the Gaussian assumption, we have

$$\mathrm{KL}(\mathbb{P}_0, \mathbb{P}_{\sigma,\tau}) = \sum_{i,j} \mathbb{E}_0[T_{i,j}]\, \mathrm{KL}(\mathbb{P}_0^{i,j}, \mathbb{P}_{\sigma,\tau}^{i,j}) = \sum_{i,j} \mathbb{E}_0[T_{i,j}] \mathbb{1}_{g^*(\sigma(i))=1} \frac{\Delta_{\tau(j)}^2}{2} \ . \tag{5.24}$$

Averaging both sides of (5.23) over all permutations σ, τ, and using (5.23)(5.24), we get

$$\log \frac{1}{6\delta} \leqslant \frac{1}{n!}\frac{1}{d!} \sum_{\sigma,\tau} \mathbb{E}_0[T_{i,j}] \mathbb{1}_{g^*(\sigma(i))=1} \frac{\Delta_{\tau(j)}^2}{2} \ . \tag{5.25}$$

Now, observe that each element in $i \in \{1, \dots, n\}$ (resp. $j \in \{1, \dots, d\}$) appears exactly $(n-1)!$ (resp. $(d-1)!$) times in the multi-set $\{\sigma(i)\}_\sigma$ (resp. $\{\tau(j)\}_\tau$), so that

$$\begin{aligned}
\frac{1}{n!}\frac{1}{d!} \sum_{\sigma,\tau} \sum_{i,j} \mathbb{E}_0[T_{i,j}] \mathbb{1}_{g^*(\sigma(i))=1} \frac{\Delta_{\tau(j)}^2}{2} &= \frac{(n-1)!}{n!}\frac{(d-1)!}{d!} \sum_{k,l} \sum_{i,j} \mathbb{E}_0[T_{i,j}] \mathbb{1}_{g^*(k)=1} \frac{\Delta_l^2}{2} \\
&= \frac{1}{n} \sum_{k \in [n]} \mathbb{1}_{g^*(k)=1} \frac{\|\Delta\|_2^2}{2d} \mathbb{E}_0[T] \ .
\end{aligned}$$

Since the group associated with Δ is the smallest, $\frac{1}{n}\sum_{k\in[n]} \mathbb{1}_{g^*(k)=1} = \theta$. Using a modified algorithm $\mathcal{A}'$ that stops at $T \wedge \chi$, we can bound $\mathbb{E}_0[T] \leqslant \chi$. Finally, it follows that:

$$\chi \geqslant \frac{2d}{\theta \|\Delta\|_2^2} \log \frac{1}{6\delta} \ .$$

Since χ is the maximum over all permuted environments constructed with M of the $(1-\delta)$-quantile of the budget, this inequality concludes the proof of Lemma 5.F.1. □

Lemma 5.F.2. *Assume that $\delta < 1/2$. If $\mathcal{A}$ is δ-correct for the clustering problem, then for any environment ν,*

$$\mathbb{E}_{\mathcal{A},\nu}[T] \geqslant \frac{2(n-2)}{\Delta_{(1)}^2} \log\left(\frac{1}{2.4\delta}\right) \ , \tag{5.26}$$

where $|\Delta_{(1)}| = \max_{j\in[d]} |\Delta_j|$.

Proof of Theorem 5.4.1. Observe that Lemma 5.F.2 does not directly provide a high-probability lower bound on the budget. We now show how the expectation bound given by Lemma 5.F.2 implies a lower bound on the $(1-\delta)$-quantile of the budget.

Let $\mathcal{A}$ be any δ-correct algorithm. Assume, by contradiction, that

$$\mathbb{P}_{\nu,\mathcal{A}}\left(T \geqslant \frac{2(n-2)}{\Delta_{(1)}^2} \log\left(\frac{1}{4.8\delta}\right)\right) < \delta.$$

We modify $\mathcal{A}$ such that it stops at time

$$T' := T \wedge \frac{2(n-2)}{\Delta_{(1)}^2} \log\left(\frac{1}{4.8\delta}\right) \ .$$

If $\mathcal{A}$ reaches time $\frac{2(n-2)}{\Delta_{(1)}^2} \log\left(\frac{1}{4.8\delta}\right)$, it stops sampling and outputs an error. The resulting algorithm $\mathcal{A}'$ is 2δ-correct, with a budget satisfying

$$\mathbb{E}_{\mathcal{A}',\nu}[T'] \leqslant \frac{2(n-2)}{\Delta_{(1)}^2} \log\left(\frac{1}{2.4\delta}\right) \ .$$

However, this contradicts Lemma 5.F.2, applied to $\mathcal{A}'$ with $\delta' = 2\delta$. Thus, we have

$$\mathbb{P}_{\nu,\mathcal{A}}\left(T \geqslant \frac{2(n-2)}{\Delta_{(1)}^2} \log\left(\frac{1}{4.8\delta}\right)\right) \geqslant \delta \ .$$

□

Proof of Lemma 5.F.2. Let $\mathcal{A}$ be any δ-correct algorithm for the clustering problem, and consider the matrix M that parametrizes the Gaussian environment ν. We fix for all environments in this proof that $g^*(1) = 0$ and $g^*(2) = 1$. It implies intuitively that we assume that the algorithm knows one item from each group via an oracle.

For the Gaussian environment ν, let $i, j \in [n] \times [d]$. The observations follow a Gaussian distribution:

$$\nu_{i,j} = \begin{cases} \mathcal{N}(0,1), & \text{if } g^*(i) = 0, \\ \mathcal{N}(\Delta_j, 1), & \text{if } g^*(i) = 1 \ . \end{cases}$$

We aim to show that with a budget smaller than $\frac{cn}{\Delta_{(1)}^2} \log(1/\delta)$, a δ-correct algorithm cannot distinguish the environment ν from another environment where one item from ν has been switched to the other group. We construct now this alternative environment.

For any $k \in \{3, \dots, n\}$, define g^k as the vector of labels obtained from g by flipping the label of row k, and let ν^k denote the corresponding Gaussian environment. The lower bound follows from the information-theoretic cost of distinguishing ν from any ν^k.

To handle multiple environments, let $\mathbb{P}_{g^k}$ (resp. $\mathbb{P}_{g^*}$) denote the probability distribution induced by the interaction between algorithm $\mathcal{A}$ and environment ν^k (resp. ν).

For any $k \in \{3, \dots, n\}$, note that environments ν and ν^k differ only on row k. By decomposing the KL divergence and using the Gaussian KL formula, we have:

$$\mathrm{KL}(\mathbb{P}_{g^*}, \mathbb{P}_{g^k}) = \sum_{j=1}^{d} \mathbb{E}_{g^*}[T_{k,j}] \frac{\Delta_j^2}{2} \leqslant \sum_{j=1}^{d} \mathbb{E}_{g^*}[T_{k,j}] \frac{\Delta_{(1)}^2}{2} \ , \tag{5.27}$$

where we use that $|\Delta_{(1)}| = \max_{j \in [d]} |\Delta_j|$, and $T_{k,j}$ denotes the number of samples taken from row k and column j.

Since $\mathcal{A}$ is δ-correct for the clustering task, we have:

$$\mathbb{P}_{g^*}(\hat{g} \neq g^*) \leqslant \delta, \qquad \mathbb{P}_{g^k}(\hat{g} \neq g^k) \leqslant \delta \ .$$

Now, if $\delta \in (0, 1/2)$, by the monotonicity of the binary KL divergence kl, and using the data-processing inequality, we obtain:

$$\mathrm{kl}(\delta, 1-\delta) \leqslant \mathrm{kl}\left(\mathbb{P}_{g^*}(\hat{g} = g^k), \mathbb{P}_{g^k}(\hat{g} = g^k)\right) \leqslant \mathrm{KL}(\mathbb{P}_{g^*}, \mathbb{P}_{g^k}) \ . \tag{5.28}$$

Combining (5.27) and (5.28), and summing over $k \in \{3, \dots, n\}$, we get:

$$(n-2)\,\mathrm{kl}(\delta, 1-\delta) \leqslant \sum_{k=3}^{n} \sum_{j=1}^{d} \mathbb{E}_{g^*}[T_{k,j}] \frac{\Delta_{(1)}^2}{2} \leqslant \mathbb{E}_{g^*}[T] \frac{\Delta_{(1)}^2}{2} \ . \tag{5.29}$$

Finally, Lemma 5.F.2 follows by combining (5.29) with the inequality $\mathrm{kl}(\delta, 1-\delta) \geqslant \log\left(\frac{1}{2.4\delta}\right)$. □

5.G Technical Results

Lemma 5.G.1 (Chernoff-Bound for Binomial random variables)**.** *For $i = 1, \dots, n$, consider $X_1, X_2, \dots, X_n \sim^{\text{i.i.d.}} \mathrm{Bern}(p)$ with $p \in (0,1)$, denote $\mu \coloneqq np$ and consider $\kappa > 0$. We have*

$$\mathbb{P}\left(\sum_{i=1}^{n} X_i \geq (1+\kappa)\mu\right) \leq \frac{e^{\kappa\mu}}{(1+\kappa)^{(1+\kappa)\mu}}.$$

If $\kappa \in (0,1)$, we also have

$$\mathbb{P}\left(\sum_{i=1}^{n} X_i \leq (1-\kappa)\mu\right) \leq \exp\left(-\frac{\kappa^2\mu}{2}\right).$$

Chapter 6

The Sampling Complexity of Condorcet Winner Identification in Dueling Bandits

Abstract. *We study best-arm identification in large-scale stochastic dueling bandits under the sole assumption that a Condorcet winner exists, i.e., an arm that wins each noisy pairwise comparison with probability at least* $1/2$*. We introduce a new identification procedure that exploits the full gap matrix* $\Delta_{i,j} = q_{i,j} - \frac{1}{2}$ *(where* $q_{i,j}$ *is the probability that arm* i *beats arm* j*), rather than only the gaps between the Condorcet winner and the other arms. We derive high-probability, instance-dependent sample-complexity guarantees that (up to logarithmic factors) improve the best known ones by leveraging informative comparisons beyond those involving the winner. We complement these results with matching lower bounds that establish the optimality of our procedures in all regimes. Overall, our results reveal the general form of the sampling complexity, characterized by a trade-off between the cost of locating informative entries and the verification cost required to achieve the desired confidence. In particular, this complexity drastically differs from what is suggested by pure asymptotic results or by procedures that are tailored to Strongly Stochastic Transitive models.*

Related publication. This Chapter is a joint work with El Mehdi Saad [1], and Nicolas Verzelen [2], available as a preprint in (Saad et al., 2026).

6.1 Motivation and High-Level Overview

In many modern machine learning applications, obtaining trustworthy absolute feedback can be difficult, expensive, or systematically biased. By contrast, relative judgments are often easier to elicit and can be highly informative. This is especially apparent in information retrieval and recommendation systems, where users naturally compare two alternatives such as rankings or models rather than providing calibrated relevance scores (Joachims et al., 2007; Hofmann et al., 2016).

The dueling bandits framework formalizes this paradigm by allowing a learner to adaptively query pairs of arms and observe only a noisy binary outcome indicating which arm is preferred. At each round, the learner selects a pair of two arms $(i, j) \in [K] \times [K]$ and observes the outcome of their duel: the feedback is 1 if arm i is preferred to arm j, and 0 otherwise. This observation is modeled as a Bernoulli random variable with unknown parameter $q_{i,j} \in [0, 1]$. The collection

1. Equal contribution—UM6P College of Computing Rabat, Morocco.
2. INRAE, Mistea, Institut Agro, Univ Montpellier, Montpellier, France.

of pairwise preference probabilities is represented by the matrix $\boldsymbol{Q} = (q_{i,j})_{i,j\in[K]}$. Since self-comparisons are uninformative and preferences are anti-symmetric, namely $q_{i,j} = 1 - q_{j,i}$ for all $i, j \in [K]$, the unknown matrix $\boldsymbol{Q}$ satisfies a skew-symmetry condition. Equivalently, we define the gap matrix $\boldsymbol{\Delta} = (\Delta_{i,j})_{i,j\in[K]}$ by $\Delta_{i,j} := q_{i,j} - 1/2$, which satisfies $\Delta_{i,j} = -\Delta_{j,i}$ and $\Delta_{i,i} = 0$.

Stochastic dueling bandits have been studied extensively under a variety of structural assumptions on $\boldsymbol{Q}$ and with multiple notions of optimality (Bengs et al., 2021; Komiyama et al., 2015; Falahatgar et al., 2017, 2018; Ren et al., 2020; Jamieson et al., 2015; Zoghi et al., 2015a; Haddenhorst et al., 2021b). Unlike in the classical multi-armed bandit setting, defining an "optimal arm" is not immediate, which has led to several competing winner definitions; see the survey of Bengs et al. (2021). In this work, we focus on instances where a distinguished arm $i^* \in [K]$ defeats, in expectation, every other arm, i.e., $q_{i^*,j} > 1/2$ for all $j \in [K] \setminus \{i^*\}$. Such an arm is called a *Condorcet winner* (CW) and is unique; we also refer to it as the optimal arm. Most existing work on dueling bandits assumes the existence of a CW (Zoghi et al., 2014, 2015b; Li et al., 2020; Komiyama et al., 2015; Chen and Frazier, 2017; Saha and Gaillard, 2022; Saha and Gupta, 2022), or even imposes the stronger requirement that the arms admit a total order (Yue et al., 2012; Yue and Joachims, 2009; Chen and Frazier, 2017). Alternative notions of optimality are discussed in Section 6.7.

Objective: Condorcet winner identification. Given $\delta \in (0, 1)$, the learner must output the CW i^* with probability at least $1 - \delta$ by adaptively and sequentially choosing pairs (i, j) to be compared and choosing a stopping time. We evaluate an algorithm by its (random) number of duels N_δ, called the budget, and we seek instance-dependent guarantees in terms of the centered gaps $\Delta_{i,j} = q_{i,j} - \frac{1}{2}$, which encode both preference direction (e.g., $\Delta_{i,j} > 0$ means i beats j) and statistical difficulty.

State-of-the-art. Although CW identification has attracted quite a lot of attention (Komiyama et al., 2015; Ailon et al., 2014; Chen and Frazier, 2017; Saha and Gaillard, 2022; Peköz et al., 2022), the optimal budget for this task remains poorly understood. Maiti et al. (2024) developed an algorithm that exploits that the CW row is the unique one with only positive gaps. They obtained a high-probability guarantee of the budget of the order of $H_{\mathrm{cw}}(\delta)$ where

$$H_{\mathrm{cw}}(\delta) := \log(1/\delta) \sum_{i\neq i^*} \frac{1}{\Delta^2_{i^*,i}} \quad . \tag{6.1}$$

In the specific scenario where the CW is the strongest opponent of every suboptimal arm, that is,

$$i^* = \mathrm{argmin}_{j:\Delta_{i,j}<0} \Delta_{i,j} \quad , \forall i \neq i^* \quad , \tag{CW-SO}$$

the condition (6.1) is matched by the lower bound from Haddenhorst et al. (2021b) which implies that the sample complexity $H_{\mathrm{cw}}(\delta)$ is optimal. Although (CW-SO) is satisfied for Strong Stochastic

Transitive (SST) models [3], this condition is arguably quite strong and is not even guaranteed when there is a total order on the arms. In particular, the bound (6.1) can be overly conservative when some arm i is nearly tied with i^* as it largely ignores potentially informative comparisons among suboptimal arms.

Prior to that, Karnin (2016) introduced a verification-based approach. The expected budget of this procedure asymptotically satisfies

$$\mathbb{E}[N_\delta] \leq c \log\left(\frac{1}{\delta}\right) \sum_{i \neq i^*} \min_{j:\Delta_{i,j}<0} \frac{1}{\Delta_{i,j}^2} + \tilde{O}\left[\sum_{i \neq i^*} \frac{1}{\Delta_{i^*,i}^2} + \sum_{i \neq i^*} \sum_{j \neq i} \min\left(\frac{1}{\Delta_{i,j}^2}, \min_{j':\Delta_{i,j'}<0} \frac{1}{\Delta_{i,j'}^2}\right)\right] , \quad (6.2)$$

where c is a positive numerical constant and $\tilde{O}$ hides possible logarithmic dependencies. The $\log(1/\delta)$ term in the bound (6.2) interprets as the sum, over all non-CW arms i, of $\log(1/\delta)/[\min_j \Delta_{i,j}]^2$ which is the minimal budget required to check whether the row $\Delta_{i,\cdot}$ is non-negative if an oracle provides to the learner the information on the best opponent of i. Indeed, Theorem 5.2 of Haddenhorst et al. (2021b) establishes a lower bound of the order $\log(1/\delta)/[\min_j \Delta_{i,j}]^2$. However, in a large-scale problem where K is large compared to $\log(1/\delta)$, the right-hand side term in (6.2) is sizable as it scales at least like $K^2/(\max_{i \neq j} \Delta_{i,j}^2)$ and its optimality is questionable. Altogether, apart from the small-scale asymptotic regime (K fixed, $\delta \to 0$) or under the restrictive condition (CW-SO), the sample complexity remains to be understood.

Question 6. *What is the true sampling complexity of CW identification and how does it depend on gap matrix $\mathbf{\Delta}$ beyond the rows and columns of the Condorcet winner?*

This question falls within *structured* pure exploration, where the feedback is noisy but constrained by an underlying latent object (here, a skew-symmetric matrix possessing a positive row), so the goal is to exploit structure rather than estimate all entries. We emphasize that such problems require ideas and techniques that are at the crossroads of active learning and high-dimensional statistics. Related challenges arise in noisy payoff matrix games Maiti (2025), e.g., in pure Nash equilibrium identification.

Contributions. Our main contributions are threefold: (i) we introduce new elimination-based algorithms for both the fixed budget and the fixed confidence settings and provide non-asymptotic guarantees, (ii) we establish matching lower bounds. (iii) Overall, this allows us to highlight the trade-offs and the multiple strategies that underlie CW identification. For the sake of simplicity, we focus in the main text on the δ-PAC setting; we also treat the fixed-budget framework and provide analogous guarantees in Section 6.5 and 6.B. Our guarantees recover, up to logarithmic factors, the known optimal rate $H_{\mathrm{cw}}(\delta)$ in the favorable structured regimes such as (CW-SO). In contrast, when this restrictive assumption fails, our bounds can be drastically smaller than the existing $H_{\mathrm{cw}}(\delta)$-type guaranties of (6.1) or the bound (6.2) of Karnin (2016) by adapting to the full gaps matrix.

3. SST assumes a total order such that, whenever i is ranked above j and j above k, we have: $q_{i,k} \geq \max\{q_{i,j}, q_{j,k}\}$.

At a high level, our elimination-based procedure (FC-CWI), described in Algorithms (16) and (17), iteratively scores the current candidates for CW using subroutines that (i) *search* in the gap matrix $\mathbf{\Delta}$ for informative comparisons and exploit its skew-symmetric structure, and (ii) *estimate* the signs of the discovered entries with sufficient accuracy. Candidates are then ranked by these scores and a constant fraction of arms is eliminated at each round. Our analysis reveals a delicate dependence on the full gap matrix $\mathbf{\Delta}$. Indeed, providing evidence that i^* is the CW either amounts to showing that all the CW gaps $\{\Delta_{i^*,i}\}_{i\neq i^*}$ are positive or amounts to showing that all arms $i \neq i^*$ are not CW. The evidence of sub-optimality for a given arm $i \neq i^*$ is governed both by the number of negative entries in its row, $K_{i;<0} := |\{j : \Delta_{i,j} < 0\}|$, and by the magnitudes of these gaps, denoted by the ordered values $\Delta_{i,(1)} \leq \cdots \leq \Delta_{i,(K_{i;<0})} < 0$. For each $i \neq i^*$, fix an integer $s_i \leq K_{i;<0}$ and write $\boldsymbol{s} = (s_1, \ldots, s_K)$. The following results will involve a trade-off in $\boldsymbol{s}$. Our analysis decomposes the complexity into $H_{\mathrm{cw}}(\delta)$ –see (6.1)–, which corresponds to the cost of separating i^* from every competitor only relying on duels with i^*, as well as two new components:

— *Exploration/Selection cost.* This term quantifies the effort required to select a negative entry whose absolute value is at least $|\Delta_{i,(s_i)}|$ in each suboptimal row.

$$H_{\mathrm{explore}}(\boldsymbol{s}, \delta) := \max_{i\neq i^*} \frac{K \log(1/\delta)}{s_i \Delta^2_{i,(s_i)}} + \sum_{i\neq i^*} \frac{K}{s_i \Delta^2_{i,(s_i)}} \ , \tag{6.3}$$

Note that the second term of the right-hand-side expression is independent of δ and accounts for the fact that looking for an entry at least $|\Delta_{i,(s_i)}|$ out of K depends on both the number s_i of such entries and the magnitude $|\Delta_{i,(s_i)}|$. The $\log(1/\delta)$-dependency only arises for a single arm $i \neq i^*$.

— *Certification cost.* This term corresponds to the number of samples required to estimate the signs of the selected gaps (at the exploration step) at confidence level $1 - \delta$

$$H_{\mathrm{certify}}(\boldsymbol{s}, \delta) := \sum_{i\neq i^*} \frac{\log(1/\delta)}{\Delta^2_{i,(s_i)}} \ .$$

Our main upper bound shows that, with probability at least $1 - \delta$, the budget N_δ of FC-CWI satisfies

$$N_\delta \lesssim H_{\mathrm{cw}}(\delta) \wedge \min_{\substack{(s_i)_{i\neq i^*} \\ \forall i,\ s_i \leq K_{i;<0} \wedge K/8}} \{H_{\mathrm{certify}}(\boldsymbol{s}, \delta) + H_{\mathrm{explore}}(\boldsymbol{s}, \delta)\} \ , \tag{6.4}$$

where the notation $\lesssim$ hides logarithmic factors in $K, (\Delta_{i,(1)})_{i\neq i^*}$ and a $\log\log(1/\delta)$ factor. Under the scenario (CW-SO), our procedure still achieves budget smaller than $H_{\mathrm{cw}}(\delta)$ as in Maiti et al. (2024) but also achieves better guarantees for other gap matrices $\mathbf{\Delta}$, where the budget is driven by the right-hand side in (6.4). In the above infimum in (6.4), the smaller the s_i's are, the smaller $H_{\mathrm{certify}}(\boldsymbol{s}, \delta)$ is, but the exploration cost $H_{\mathrm{explore}}(\boldsymbol{s}, \delta)$ for localizing a good candidate can increase for small s_i's. In the following, we denote $\boldsymbol{s}^*_{\mathbf{\Delta}}$ as the vector $(s^*_i)_{i\neq i^*}$ achieving the best trade-off in Equation (6.4). We interpret $s^*_{\mathbf{\Delta}}$ as an effective sparsity of $\mathbf{\Delta}$, although it also depends on δ. Importantly, our algorithm does not take $\boldsymbol{s}^*_{\mathbf{\Delta}}$ as input and therefore *automatically* achieves the

best balance captured by (6.4).

In Section 6.4, we provide matching lower bounds supporting the optimality of our budget in all regimes with respect to δ and K, as well as for general forms of the matrix $\mathbf{\Delta}$. Importantly, this showcases that the exploration/certification trade-off unveiled in (6.4) is unavoidable and intrinsic to the sample complexity of CW-identification. The benefit of our procedure is also illustrated on numerical experiments.

Emblematic regimes. As the sample complexity is quite intricate in the general case, we discuss some specific regimes to emphasize key phenomena.

— CW has small gaps. If some entries of the i^*-th row of Δ are small, then $H_{\mathrm{cw}}(\delta)$ is not the minimum in (6.4) and our sample complexity is of the order of $H_{\mathrm{certify}}(\boldsymbol{s}^*_{\mathbf{\Delta}}, \delta) + H_{\mathrm{explore}}(\boldsymbol{s}^*_{\mathbf{\Delta}}, \delta)$ which can be arbitrarily smaller than the budget $H_{\mathrm{cw}}(\delta)$ of Maiti et al. (2024).

— Large-scale/fixed probability regime. Let us consider δ as a constant and K as large. In (6.4), when $H_{\mathrm{cw}}(\delta)$ is not the minimum, then the sample complexity is of the order [4] of

$$\min\left(\sum_{i\neq i^*}\frac{1}{\Delta^2_{i^*,i}}, \sum_{i\neq i^*}\frac{K}{\|\Delta_i^-\|_2^2}\right) ,$$

where $\Delta^-_{i,j} := \min(\Delta_{i,j}, 0)$. In comparison to the complexity of Karnin (2016) in (6.2), our sample complexity is always smaller and can be smaller by a factor as larger as K.

— Small probability regime. This last observation is more subtle. Similarly to Karnin (2016), consider the asymptotic regime where $\log(1/\delta)$ goes to infinity, while K and $\mathbf{\Delta}$ are fixed. In this asymptotic, our lower and upper bounds on the $(1-\delta)$-quantile of the budget are of the form

$$\log\left(\frac{1}{\delta}\right)\inf_{\boldsymbol{s}}\left[\sum_{i\neq i^*}\frac{1}{\Delta^2_{i,(s_i)}} + \max_{i\neq i^*}\frac{K}{s_i\Delta^2_{i,(s_i)}}\right] ,$$

whereas the bound in Karnin (2016) on the expected budget only involves the smaller quantity $\sum_{i\neq i^*}\frac{\log(1/\delta)}{\Delta^2_{i,(1)}}$. Since our bound is optimal in quantile, this establishes that there is a significant, yet intrinsic, gap between guarantees in expectation and in quantile of the budget. Especially, when there is heterogeneity in the $\Delta_{i,(s_1)}$, this gap can be as large as a factor K. This phenomenon is central for the analysis of fixed-budget algorithms - see Section 6.B.

All the way through these two extreme regimes (δ fixed, $\delta \to 0$), both (6.4) and the matching lower bound illustrate a trade-off between exploration and certification. Intuitively, when $\log(1/\delta)$ increases, the effective sparsity $s^*_{\mathbf{\Delta}}$ tends to decrease so the algorithm explores other arms more thoroughly to identify stronger opponents.

Technical Innovations. Our Algorithms 16 and 17 are based on a new iterative scoring strategy that builds on the selection, for each 'active' arm i, of a strong opponent as well as the estimation of

4. To show this, we observe that $\max_{s_i=1,\dots,K_{i;<0}} s_i\Delta^2_{i,(s_i)} \asymp \|\Delta_i^-\|_2^2$, see Lemma 6.F.1.

some quantile of the estimation $\Delta_{i,\cdot}$. For that purpose, we need to introduce a new active quantile estimation algorithm achieving optimal ϵ-error simultaneously for all ϵ –see Section 6.2. Our main lower bounds use novel approaches and techniques as our aim is to lower bound the $(1-\delta)$ quantile of the budget. For that purpose, we reduce the problem to an active multiple testing problem of the existence of negative entries within a vector of size $K-1$, our lower bound introduce techniques that go beyond classical MAB lower bounds are at the crossroad of high-dimensional statistics and bandit theory.

Organization. Section 6.3 presents our algorithms and fixed-confidence upper bounds; the fixed-budget guarantees are deferred to the appendix. Section 6.4 gives lower bounds for the fixed-confidence setting, while Section 6.5 provides lower bounds for the fixed-budget setting. Section 6.6 illustrates our results on numerical experiments. Section 6.7 concludes with implications and future directions. All proofs are provided in the appendix.

6.2 Intermediate result: Adaptive quantile estimation

In this section, we present an algorithm for quantile estimation with a fixed budget of samples, and a high-probability guarantee on the estimation error. This result is of independent interest and will be used as a key building block in the proofs of the main results.

Consider a classical K-armed bandit setting, where we are given K arms with means $(\mu_i)_{i\in[K]}$. We assume that the samples from the arms are bounded [5] by 1, and without loss of generality that the means are distinct. Denote by $\mu_{(1)} < \mu_{(2)} < \ldots < \mu_{(K)}$ the ordered means of the K arms. For two integers $d \leq u$ in $[K]$, our objective is to find a point in the interval $[\mu_{(d)}, \mu_{(u)}]$.
For this task, a learner is given a fixed budget of T queries, after which the learner outputs a quantity q_T. In this framework, we are targeting an 'adaptive' guarantee in the following sense. Given the budget T as input, we want the output to satisfy

$$\forall \epsilon > 0, \quad \mathbb{P}\left(q_T \notin \left[\mu_{(d)} - \epsilon, \mu_{(u)} + \epsilon\right]\right) \leq \exp\left(-c \cdot r\epsilon^2 \frac{T}{\log(T)}\right) ,$$

where c is a positive universal constant, and r is a positive quantity depending only on d, u and K.
The allocation strategy we develop requires a sufficiently large budget. Specifically, we assume that

$$T \geq \frac{128K}{u-d} \log_2\left(\frac{128K}{u-d}\right) .$$

When T falls below this threshold, the resulting guarantee becomes vacuous (the stated upper bound on the error probability exceeds 1). In this regime, we therefore resort to an arbitrary heuristic. Algorithm 15 implements this by explicitly branching between the small-budget case $T < \frac{128K}{u-d} \log_2\left(\frac{128K}{u-d}\right)$ and the regime where the budget is large enough for the analysis to be meaningful.

5. The result can be extended to sub-Gaussian variables

Solving this problem requires balancing the tasks of locating the arms whose means fall within the desired rank range, and estimating these means accurately. The algorithm runs a multi-step scheme indexed by ℓ. At each level ℓ, it draws a random multi-set $\mathcal{A}_\ell$ of arms large enough to contain, with good probability, representatives of the $[d, u]$ quantile range. Then, it allocates $\Theta(\epsilon_\ell^{-2})$ samples per selected arm so that empirical means are accurate up to ϵ_ℓ.
Then we form three empirical quantiles $\hat{t}_\ell^{(1)}, \hat{t}_\ell^{(2)}, \hat{t}_\ell^{(3)}$ corresponding to ranks slightly below, near the middle of, and slightly above $[d, u]$, yielding a noisy bracket around the target interval. Finally, a Lepski-type stability rule selects the earliest level $\bar{\ell}$ whose middle estimate $\hat{t}_\ell^{(2)}$ remains consistent with the brackets produced at all finer levels (up to the tolerance $2\epsilon_{\ell'}$), and returns $\hat{t}_{\bar{\ell}}^{(2)}$.
Theorem 6.2.1 states that with budget T, the output lies in $[\mu_{(d)} - \epsilon, \mu_{(u)} + \epsilon]$ with high probability for every $\epsilon \in (0, 1)$, and the failure probability decays essentially as $\exp(-\Theta(\epsilon^2 T / \log T))$ (up to the multiplicative factor $40 \log_2(T)$ and the extra $\log(\frac{16K}{u-d})$ term). The quantity

$$r = \min\left\{\frac{d}{K}, 1 - \frac{u}{K}\right\} \cdot \left(\frac{u-d}{K}\right)^2,$$

captures the difficulty of the target rank range as it decreases when the interval is narrower ($u - d$ small) or when it is close to the extremes (small d or large u). When d and u are constant fractions of K, we have $r = \Theta(1)$ and the bound simplifies to $\log(T)\exp(-c\epsilon^2 T / \log T)$.

Algorithm 15: Range-Quantile(K, d, u, T)

Input: K arms, budget T, integers $d \le u$ in $[K]$

$L \leftarrow \lfloor * \rfloor \log_2(T / \log_2(T))$, $\ell_{\min} \leftarrow \lceil * \rceil \log_2\left(\frac{16K}{u-d}\right)$;

if $T \le \frac{128K}{u-d} \log_2\left(\frac{128K}{u-d}\right)$ or $u = d$ **then**

  Allocate budget uniformly over arms and return the average empirical mean between ranks d and u; ▷ Small-budget fallback

end

for $\ell = \ell_{\min}, \ldots, L-1$ **do**

  $\epsilon_\ell \leftarrow 2 \cdot 2^{-(L-\ell)/2}$;

  Sample a multiset $\mathcal{A}_\ell$ of size $\lfloor * \rfloor \frac{\epsilon_\ell^2 T}{\log\left(\frac{16K}{u-d}\right)\log_2(T)}$ from $[K]$ with replacement;

  ▷ Duplicates treated as distinct

  Allocate $\lceil * \rceil \frac{\log\left(\frac{16K}{u-d}\right)}{2\epsilon_\ell^2}$ samples to each arm $a \in \mathcal{A}_\ell$ and compute $\hat{\mu}_a$;

  Rank empirical means in $\mathcal{A}_\ell$: $\hat{\mu}_{(1)} \le \cdots \le \hat{\mu}_{(|\mathcal{A}_\ell|)}$;

$$\hat{t}_\ell^{(1)} = \hat{\mu}_{(\lceil * \rceil \frac{3d+u}{4K}|\mathcal{A}_\ell|)}, \quad \hat{t}_\ell^{(2)} = \hat{\mu}_{(\lceil * \rceil \frac{d+u}{2K}|\mathcal{A}_\ell|)}, \quad \hat{t}_\ell^{(3)} = \hat{\mu}_{(\lceil * \rceil \frac{d+3u}{4K}|\mathcal{A}_\ell|)}. \tag{6.5}$$

end

$\bar{\ell} \leftarrow \min_{\ell \in [\![\ell_{\min}, L-1]\!]} \left\{\forall \ell' \in \{\ell, \ldots, L-1\} : \hat{t}_\ell^{(2)} \in \left[\hat{t}_{\ell'}^{(1)} - 2\epsilon_{\ell'}, \hat{t}_{\ell'}^{(3)} + 2\epsilon_{\ell'}\right]\right\}$;

return $\hat{t}_{\bar{\ell}}^{(2)}$; ▷ Lepski-type selected estimate

Theorem 6.2.1. *Consider Algorithm 15 with inputs (K, d, u, T), such that $u > d$. Then, the output satisfies for any $\epsilon \in (0,1)$:*

$$\mathbb{P}\left(\hat{t}_{\bar{\ell}}^{(2)} \notin [\mu_{(d)} - \epsilon, \mu_{(u)} + \epsilon]\right) \leq 40 \log_2(T) \exp\left(-c \cdot r \cdot \frac{\epsilon^2 T}{\log\left(\frac{16K}{u-d}\right) \log_2(T)}\right), \tag{6.6}$$

where $r = \min\left\{\frac{d}{K}, 1 - \frac{u}{K}\right\}\left(\frac{u-d}{K}\right)^2$ is a positive quantity depending only on d, u and K, and c is an absolute numerical constant.

Here, we did not try to optimize the constants. Next, we state a corollary that will be used in the proofs of the main theorems.

Corollary 6.2.2. *Consider Algorithm 15 with inputs $(K, \lceil K/8 \rceil, \lceil K/4 \rceil, T)$, where $T \geq 4$. Then, the output satisfies for any $\epsilon \in (0,1)$:*

$$\mathbb{P}\left(\hat{t}_{\bar{\ell}}^{(2)} \notin [\mu_{(\lceil K/8 \rceil)} - \epsilon, \mu_{(\lceil K/4 \rceil)} + \epsilon]\right) \leq \log(T) \exp\left(-c \cdot \frac{\epsilon^2 T}{\log(T)}\right),$$

where c is an absolute numerical constant smaller than 1.

6.3 Upper Bounds: Algorithm and Guarantees

This section presents our fixed-confidence identification procedure and the upper bound announced in Section 6.1. The algorithm is built from a budgeted elimination-and-certification subroutine, denoted FB-CWI (Algorithm 16). Given a tentative budget scale T, a confidence level δ, this subroutine performs one elimination run using $\mathcal{O}(T)$ duels and returns a candidate arm together with a Boolean certificate indicating whether the run can be safely stopped. The budget T should be viewed only as a guessed complexity scale: if it is too small for the instance, the subroutine is allowed to fail to certify rather than output a final answer.

The fixed-confidence algorithm FC-CWI (Algorithm 17) removes the need to know this scale in advance by calling FB-CWI with geometrically increasing values of T and stopping at the first successful certificate. Thus, the main algorithm is a genuine δ-PAC procedure: the stopping time is determined by the certification step, while the budgeted subroutine only provides the candidate-generation and elimination mechanism. We describe this subroutine first, and then state the high-probability sample-complexity guarantee for the resulting doubled procedure. The corresponding fixed-budget framework and guarantees are deferred to the appendix.

FB-CWI is an elimination procedure initialized with $A_1 = [K]$: at each round k, it assigns a score $S_k(\alpha)$ to every active arm $\alpha \in A_k$, ranks the arms accordingly, and discards the bottom $1/8$ fraction (so $|A_{k+1}| = \lfloor 7\,|A_k|\,/8 \rfloor$). Therefore, the number of rounds is $O(\log K)$.
The core of FB-CWI is the score computation, whose purpose is to keep the CW ranked above the elimination threshold. At round k, we split a budget of order $T/\log(K)$ across the active set A_k. For each $\alpha \in A_k$, we devote one quarter of its share to search for a strong opponent by running Sequential Halving (SH) (Karnin et al., 2013a) on the instance of the duels $\{(\beta, \alpha) : \beta \in [K] \backslash \{\alpha\}\}$,

yielding an opponent $\alpha^{(s)}$ that is likely to beat α, and another quarter to estimate the gap $\Delta_{\alpha,\alpha^{(s)}}$ via an empirical mean; this estimate defines the strong-opponent component of the score. We call $\alpha^{(s)}$ 'strong' because it is selected from all K arms (not only from A_k): this tends to penalize sub-optimal arms more sharply, at the price of higher uncertainty due to the larger search space. Relying only on the strong-opponent term can be brittle: if the CW is nearly tied with some arm, the selected opponent may yield a gap estimate close to zero and provide little separation with the elimination threshold. We therefore add a 'weak-opponent' term that yields adaptivity to larger gaps with the CW. More specifically, writing $\mathbf{\Delta}^{(k)} := \mathbf{\Delta}_{A_k \times A_k}$, skew-symmetry implies that at least half of the entries of $\mathbf{\Delta}^{(k)}$ are non-positive, and a simple pigeon-hole type argument implies that at least $|A_k|/4$ rows contain at least $|A_k|/4$ non-positive entries (Lemma 6.F.6 in the appendix). Accordingly, for each $\alpha \in A_k$ we estimate a point whose value lies between the 1/8- and 1/4-quantiles of the row $(\Delta_{\alpha,\beta})_{\beta \in A_k}$ (via RANGE-QUANTILE). This lower-tail statistic is typically negative for many sub-optimal arms pushing them into the bottom-1/8 region, while for the CW it remains positive and leverages the fact that most of its gaps can still be large.

Subroutines: Bracketed Sequential Halving and Range-Quantile. Our score construction relies on two subroutines. For the strong-opponent search we use Bracketed Sequential Halving (BSH) (Zhao et al., 2023), an anytime and parameter-free variant of SH, (Karnin et al., 2013a), designed for the data-poor regime, chosen for its adaptive guarantees on simple regret, which translate in our context into gap-dependent guarantees –see Zhao et al. (2023) and Section 6.2 of the appendix. For the weak-opponent choice, we introduce Range-Quantile (Algorithm 15), a general fixed-budget procedure revealing a point in a prescribed quantile range: given N arms with means $(\mu_i)_{i\in[N]}$ ordered as $\mu_{(1)} \leq \cdots \leq \mu_{(N)}$ and indices $d < u$, it returns an estimate $\hat{t}$ that falls between the d-th and u-th means (up to an additive error ε) with error probability decaying as $\exp(-\widetilde{\Theta}(\frac{(u-d)^2}{N^2} T\varepsilon^2))$ –see Theorem 6.2.1). Importantly, RANGE-QUANTILE does not require ε as input and is therefore simultaneously valid for any ϵ; in FB-CWI we instantiate it with $N = |A_k|$, $\varepsilon = \frac{1}{2}\Delta_{i^*,(\lceil |A_k|/8 \rceil)}$, $d = \lfloor |A_k|/8 \rfloor$ and $u = \lceil |A_k|/4 \rceil$ to obtain a value between the 1/8- and 1/4-quantiles of $(\Delta_{\alpha,\beta})_{\beta\in A_k}$. Note that Maiti et al. (2024) gives a fixed-confidence routine that, given (δ, ε), outputs a value in $[\mu_{(N/2)} - \varepsilon,\ \mu_{(N/4+1)} + \varepsilon]$ with probability at least $1-\delta$. Here, ε is instance-dependent and unknown, which motivates our adaptive RANGE-QUANTILE subroutine that does not take ε as input.

From a budgeted subroutine to fixed confidence. Algorithm 16 is used as a certificate-driven subroutine: for a tentative budget scale T and confidence level δ, it spends $\mathcal{O}(T)$ duels and returns a candidate I together with a success flag $\Psi = \phi_1 \vee \phi_2$. The direct certificate ϕ_2 verifies that I is the CW, leading to the classical $H_{\mathrm{cw}}(\delta)$ regime. The elimination certificate ϕ_1 instead checks that the elimination frontier remains safely negative, thereby exploiting informative comparisons among suboptimal arms. The analysis shows that, up to logarithmic factors, certification succeeds once

$$T \gtrsim H_{\mathrm{cw}}(\delta) \wedge \min_{\substack{(s_i)_{i\neq i^*} \\ s_i \leq K_{i;<0}}} \{H_{\mathrm{certify}}(\boldsymbol{s}, \delta) + H_{\mathrm{explore}}(\boldsymbol{s}, \delta)\} \ . \tag{6.7}$$

Algorithm 16: FB-CWI + Certification

Input: Fixed budget T, certification parameters (δ, T, c)

$k \leftarrow 1$, $A_1 \leftarrow [K]$, $n \leftarrow \log_2\left(\frac{T}{2K\log_{8/7}(K)}\right)$;

$\phi_1, \phi_2 \leftarrow$ `True`;

while $|A_k| > 1$ **do**

  $B_k \leftarrow \lfloor * \rfloor \frac{T}{|A_k|\log_{8/7}(K)}$;

  for $\alpha \in A_k$ **do**

    Run Bracketed SH with budget $\lceil B_k/4 \rceil$ on candidates $\{(\beta, \alpha) : \beta \in [K] \setminus \{\alpha\}\}$; let $(\alpha^{(s)}, \alpha)$ be the output; ▷ Find a strong opponent

    Query $\lceil B_k/4 \rceil$ samples of $(\alpha, \alpha^{(s)})$ and compute $Z_k^{(s)}(\alpha)$;

    Run Range-Quantile on duels between α and $A_k \setminus \{\alpha\}$ with budget $\lceil B_k/2 \rceil$ and quantiles $(\lceil |A_k|/8 \rceil, \lceil |A_k|/4 \rceil)$; let $Z_k^{(w)}(\alpha)$ be the output; ▷ Compute weak score component

  end

  Compute $S_k(\alpha) = \min\{Z_k^{(s)}(\alpha), 0\} + Z_k^{(w)}(\alpha)$ for each $\alpha \in A_k$;

  Rank arms in A_k by $S_k(\cdot)$ and set A_{k+1} to the top $|A_k| - \lceil |A_k|/8 \rceil$ arms;

  $\bar{\alpha} \leftarrow$ arm ranked $|A_k| - \lceil |A_k|/8 \rceil + 1$ according to $S_k(\cdot)$;

  $\phi_1 \leftarrow \phi_1 \cdot \mathbb{1}\left(S_k(\bar{\alpha}) < -\sqrt{\frac{2c\log(T)}{\lceil B_k/4 \rceil}\log\left(8K^2\log_{8/7}(K)\log(T)\cdot\frac{n(n+1)}{\delta}\right)}\right)$;

  ▷ Fixed-confidence check

  $k \leftarrow k + 1$;

end

$I \leftarrow$ unique arm in A_k;

$\phi_2 \leftarrow$ Test-CW(I, δ, T); ▷ Certify sign of all gaps for I

return $\phi_1 \vee \phi_2$, I; ▷ Return certificate and candidate

Algorithm 17: FC-CWI

Input: c, confidence parameter δ

$\varphi \leftarrow$ `False`, $T \leftarrow 8K\log(K)$;

while $\neg\varphi$ **do**

  $(\varphi, I) \leftarrow$ FB-CWI(δ, T, c); ▷ Run one budgeted certification round

  $T \leftarrow 2T$; ▷ Doubling schedule on the budget

end

return I; ▷ First certified candidate

Since these quantities are unknown, we run FB-CWI under a doubling schedule on T and stop as soon as a budgeted verification succeeds. It uses at most $2T$ queries per stage (in Algorithm 16, ϕ_1 is computed from the same samples as the scores, and ϕ_2 runs TEST-CW with at most $T/2$ extra queries). A direct verification is to certify that the returned arm I is Condorcet by testing $\Delta_{I,j} > 0$ for all $j \neq I$ at level $1-\delta$, which costs $\sum_{j\neq I} \log(1/\delta)/\Delta_{I,j}^2$ and matches the $H_{\mathrm{cw}}(\delta)$ regime. When the second term in (6.7) is dominant, our identification relies instead on certifying sub-optimality: many sub-optimal arms have scores $S_k(\cdot)$ that are typically negative while the CW remains positive. This motivates the second verification ϕ_1, which checks that the elimination frontier is on the negative side. We stop at the first success of ϕ_1 or ϕ_2, which guarantees δ-correctness. The complete proof of Theorem 6.3.1 is presented in Section 6.C of the appendix.

Theorem 6.3.1. *There exists a constant c_0 such that the following holds for any $\delta \in (0, 1/6)$. Under the assumption of the existence of a CW, the output of Algorithm 17, denoted ψ, with input (δ, c) where $c \geq c_0$ satisfies: $\mathbb{P}(\psi_\delta \neq i^*) \leq \delta$.*

Moreover, the total number of queries N_δ satisfies, with probability at least $1-\delta$,

$$N_\delta \leq \tilde{c} \cdot \Big(H_{cw}(\delta) \wedge \min_{\substack{(s_i)_{i\neq i^*} \\ \forall i,\ s_i \leq K_{i;<0}}} \big\{H_{certify}(\boldsymbol{s}, \delta) + H_{explore}(\boldsymbol{s}, \delta)\big\}\Big) \ ,$$

where $\tilde{c}$ is proportional to c and hides logarithmic factors in K and $(\Delta_{i,(1)})_{i\neq i^*}$ and a $\log\log(1/\delta)$ factor, and $H_{cw}, H_{certify}, H_{explore}$ are defined in Section 6.1 .

Remark 6.3.2 (On the constant c in the stopping rule). The stopping condition in Algorithm 16 involves a numerical constant c_0 inherited from the high-probability analysis of Corollary 6.2.2. This absolute value of c_0 can be made explicit by tracking constants in the proof. Since we did not optimize numerical factors, we keep c_0 symbolic for readability.

Guarantees: intuition. The proof of Theorem 6.3.1 analyzes one call of the budgeted subroutine FB-CWI at a tentative scale T. In the outer fixed-confidence algorithm, a call that does not certify is harmless: FC-CWI simply doubles T and tries again. Thus, the key point is to show that once T exceeds the relevant instance-dependent scale, the CW is not eliminated and one of the certificates succeeds with high probability. Conditionally on $i^* \in A_k$, the only way the elimination step can be harmful is if the CW score falls below the bottom-1/8 cutoff in some round. The analysis therefore controls the separation between the CW score and the elimination frontier across the $O(\log K)$ rounds.

The weak-opponent term gives a baseline separation. At round k, the CW benefits from a positive lower-tail gap of order $\Delta_{i^*,(\lceil * \rceil |A_k|/8)}$, estimated with $B_k = \Theta(T/(|A_k|\log K))$ samples. Concentration bounds, together with the RANGE-QUANTILE guarantee, yield an error of order $\exp(-\widetilde{\Theta}(B_k \Delta^2_{i^*,(\lceil |A_k|/8 \rceil)}))$. The worst round is controlled by

$$\max_k \frac{|A_k|/8}{\Delta^2_{i^*,(\lceil |A_k|/8 \rceil)}} \leq \max_{i\in[K-1]} \frac{i}{\Delta^2_{i^*,(i)}} \leq \sum_{i\neq i^*} \frac{1}{\Delta^2_{i,i^*}} \ ,$$

which leads, after the final CW verification, to the classical $H_{\mathrm{cw}}(\delta)$ regime.

The strong-opponent term yields the matrix-adaptive regime by actively finding negative witnesses for suboptimal arms. Fix $s_i \leq K_{i,<0}$. For each $i \neq i^*$, Sequential Halving needs budget of order $K/(s_i\Delta^2_{i,(s_i)})$ to locate an opponent whose gap is at most $\Delta_{i,(s_i)}$, while certifying the sign of this gap costs $1/\Delta^2_{i,(s_i)}$. The aggregate cost of locating such witnesses gives to $\sum_{i\neq i^*} \frac{K}{s_i\Delta^2_{i,(s_i)}}$ whereas the high-confidence part is governed by $\sum_{i\neq i^*} \frac{1}{\Delta^2_{i,(s_i)}} + \max_{i\neq i^*} \frac{K}{s_i\Delta^2_{i,(s_i)}}$. Thus, certification succeeds once T is larger than the bound stated in Theorem 6.3.1.

6.4 Fixed confidence Lower Bounds

Beyond the instance-dependent lower bound (6.2) from Haddenhorst et al. (2021b) that we complement with a corresponding quantile-bound and restate as Proposition 6.4.1, we establish in Theorem 6.4.2 the first lower bound that highlight the intrinsic cost of detecting informative entries for CWI. In Section 6.5, we also derive fixed-budget minimax lower bounds. Denote by $\mathbb{D}_{\mathrm{cw}}$ the class of dueling bandit environments that admits a CW [6] :

$$\mathbb{D}_{\mathrm{cw}} := \left\{ \boldsymbol{\Delta} \in [-\tfrac{1}{4}, \tfrac{1}{4}]^{K\times K} : \boldsymbol{\Delta} = -\boldsymbol{\Delta}^T \text{ and } \exists\, i^* \in [K] \text{ such that } \forall j \neq i^*,\ \Delta_{i^*,j} > 0 \right\} . \tag{6.8}$$

We say that an algorithm A is δ-correct for CW identification if, for any $\boldsymbol{\Delta} \in \mathbb{D}_{\mathrm{cw}}$, it identifies i^* with error probability at most δ, that is, $\mathbb{P}_{\boldsymbol{\Delta},A}(\hat{\imath} \neq i^*(\boldsymbol{\Delta})) \leqslant \delta$, where $\mathbb{P}_{\boldsymbol{\Delta},A}$ denotes the probability [7] induced by the interaction between A and the environment with gap matrix $\boldsymbol{\Delta}$.

Proposition 6.4.1. *Let $K \geqslant 2$ and $\delta \in (0, 1/6)$ and consider any gap matrix $\boldsymbol{\Delta} \in \mathbb{D}_{\mathrm{cw}}$. For any algorithm A that is δ-correct on $\mathbb{D}_{\mathrm{cw}}$, the budget N_δ satisfies*

$$\mathbb{E}_{\boldsymbol{\Delta},A}[N_\delta] \ \geqslant\ \frac{1}{4}\sum_{i\neq i^*} \frac{\log(1/(4\delta))}{\Delta^2_{i,(1)}} \ , \ \textit{and} \quad \mathbb{P}_{\boldsymbol{\Delta},A}\left(N_\delta \geqslant \frac{1}{3}\sum_{i\neq i^*} \frac{\log\left(1/(6\delta)\right)}{\Delta^2_{i,(1)}} \right) \geqslant \delta. \ . \tag{6.9}$$

The lower Bound of the expected budget is a particular case of Theorem 5.2 in Haddenhorst et al. (2021b). We complement it here with a quantile-bound. This proposition certifies optimality of the complexity $H_{\mathrm{cw}}(\delta)$ in the CW–SO regime. Indeed, under (CW-SO), the strongest negative entry of every suboptimal row is the comparison with the Condorcet winner, so $\Delta_{i,(1)} = \Delta_{i,i^\star} = -\Delta_{i^\star,i}$. Hence the lower bound in Proposition 6.4.1 is, up to constants, $\sum_{i\neq i^\star} \log(1/\delta)/\Delta^2_{i^\star,i} = H_{\mathrm{cw}}(\delta)$.

To characterize the optimality of (6.4), we establish a lower bound on the δ-quantile of any algorithm under a local class of instances. The budget condition (6.4) only depends on the gap matrix $\boldsymbol{\Delta}$ through three vectors: (i) the row i^* of the CW $\Delta_{i^*,\cdot}$, (ii) the effective sparsity $\boldsymbol{s}^*_{\boldsymbol{\Delta}}$, and (iii) the gaps at the sparsity level $\boldsymbol{s}^*$: $(\Delta_{i,(s^*_i)})_{i\neq i^*}$. Given any gap matrix $\boldsymbol{\Delta}$, we define the collection $\mathbb{D}(\boldsymbol{\Delta})$ of gap matrices $\tilde{\boldsymbol{\Delta}}$ that leave the Condorcet winner $i^*_{\boldsymbol{\Delta}}$, the effective sparsity $\boldsymbol{s}^*_{\boldsymbol{\Delta}}$,

6. The restriction to gaps in away from $-1/2$ and $1/2$ is standard in lower bounds with Bernoulli rewards.
7. denote $\mathbb{E}_{\boldsymbol{\Delta},A}$ for the corresponding expectation

and the gaps $(\Delta_{i,(s_i^*)})_{i\neq i^*}$ unchanged

$$\mathbb{D}(\boldsymbol{\Delta}) := \{\tilde{\boldsymbol{\Delta}} \text{ s.t. } i^*_{\tilde{\boldsymbol{\Delta}}} = i^*_{\boldsymbol{\Delta}},\ \boldsymbol{s}^*_{\tilde{\boldsymbol{\Delta}}} = \boldsymbol{s}^*_{\boldsymbol{\Delta}},\ (\tilde{\Delta}_{i,(s_i^*)})_{i\neq i^*} = (\Delta_{i,(s_i^*)})_{i\neq i^*}\}\ . \tag{6.10}$$

Theorem 6.4.2. *Let A be a δ-correct algorithm for CW identification, with $\delta \leqslant 1/12$, and let $\boldsymbol{\Delta} \in \mathbb{D}_{\mathrm{cw}}$. Assume that $\boldsymbol{\Delta}$ has no ties, that is, $\forall i \neq j$, $\Delta_{i,j} \neq 0$. For this matrix $\boldsymbol{\Delta}$, one can construct a matrix $\tilde{\boldsymbol{\Delta}}$ by permuting the entries of $\boldsymbol{\Delta}$ in such a way that $\tilde{\boldsymbol{\Delta}} \in \mathbb{D}(\boldsymbol{\Delta})$, and such that*

$$\mathbb{P}_{\tilde{\boldsymbol{\Delta}},A}\left(N_\delta \ \geqslant\ \frac{1}{3}\max_{i\neq i^*}\frac{K_{i;<0}}{\|\Delta_i^-\|^2}\log\left(\frac{1}{6\delta}\right) \vee \frac{1}{37\log(2K)}\sum_{i\neq i^*}\frac{K_{i;<0}}{\|\Delta_i^-\|^2}\right) \ \geqslant\ \delta\ . \tag{6.11}$$

Moreover, for all $i \neq i^*$, the rows satisfy $(\tilde{\Delta}_{i,(j)})_{j\leq K_{i;<0}} = (\Delta_{i,(j)})_{j\leq K_{i;<0}}$, i.e., $\tilde{\Delta}_{i,\cdot}$ and $\Delta_{i,\cdot}$ share the same $K_{i;<0}$ negative entries, up to permutation.

Remark 6.4.3. While the bound (6.9) from Proposition 6.4.1 captures the cost of certification, the lower bound (6.11) captures the intrinsic need of exploration. Indeed, from the classical bound of Lemma 6.F.1, we have $\min_{\boldsymbol{s}} H_{\mathrm{explore}}(\boldsymbol{s},\delta) = \max_{i\neq i^*} K\log(1/\delta)/\|\Delta_i^-\|^2 + \sum_{i\neq i^*} K/\|\Delta_i^-\|^2$, up to a factor $\log(2K)$. Importantly, the two lower bounds in Proposition 6.4.1 and Theorem 6.4.2 imply the following.

Corollary 6.4.4. *Let A be a δ-correct algorithm for CW identification, with $\delta \leqslant 1/12$, and let $\boldsymbol{\Delta} \in \mathbb{D}_{\mathrm{cw}}$. Then, one can construct $\tilde{\boldsymbol{\Delta}} \in \mathbb{D}(\boldsymbol{\Delta})$, such that*

$$\mathbb{P}_{\tilde{\boldsymbol{\Delta}},A}\left(N_\delta \gtrsim H_{certify}(\boldsymbol{s}^*,\delta) + H_{explore}(\boldsymbol{s}^*,\delta)\right) \geqslant \delta\ , \tag{6.12}$$

where $\gtrsim$ hides a log term in K and a numerical constant.

This corollary assesses the optimality of our upper bound (6.4) on the budget. Indeed, observe that $\tilde{\boldsymbol{\Delta}}$ has exactly the same sign structure as $\boldsymbol{\Delta}$ (i.e., $\mathrm{sign}(\boldsymbol{\Delta}) = \mathrm{sign}(\tilde{\boldsymbol{\Delta}})$), and that the permutation preserves, in each row, the multiset of negative magnitudes. Intuitively, $\tilde{\boldsymbol{\Delta}}$ has the same dueling structure as $\boldsymbol{\Delta}$, except that $\tilde{\boldsymbol{\Delta}}$ has been chosen in such a way that the complexity $H_{\mathrm{cw}}(\delta)$ for $\tilde{\boldsymbol{\Delta}}$ is larger than $H_{\mathrm{certify}}(\boldsymbol{s}^*,\delta) + H_{\mathrm{explore}}(\boldsymbol{s}^*,\delta)$. A more general version—Theorem 6.D.3, which also covers ties and provides an explicit construction—is provided in Appendix 6.D.3.

Proof Sketch Theorem 6.4.2. The proof reduces CW identification to a collection of active signal-detection tasks: for each non-winner row, the learner must certify the existence of at least one negative entry under a global error constraint. The key technical novelty is an adversarial permutation construction that hides the locations of negative gaps and reduces the problem to multiple hypothesis testing; this yields the complexity term $\frac{K_{i;<0}}{\|\Delta_i^-\|^2}\log(1/\delta)$ via active signal-detection lower-bound arguments. Finally, the two terms in (6.11) are obtained through new multiple-hypothesis techniques.

6.5 Minimax Fixed-Budget Lower Bounds

In this section, we establish lower bounds for the fixed-budget CW identification. In some way, these are the counterparts of the results of Section 6.4, only that we only derive minimax bounds, similar to Corollary 6.4.4.

We specify a class of distributions, parametrized by the row of the Condorcet Winner. Let $\underline{\Delta} = (\Delta_1, \Delta_2, \dots, \Delta_K)$, with $\Delta_1 = 0$ and $\Delta_i \in (0, 1/4)$ for $i \neq 1$. Define $\mathbb{D}^{(1)}(\underline{\Delta})$ as the set of dueling feedback distributions whose gap matrix $\boldsymbol{\Delta} \in \mathbb{D}_{\mathrm{cw}}$ is such that the row of the Condorcet winner i^*, namely $\Delta_{i^*,\cdot}$, is equal to $\underline{\Delta}$ up to a permutation σ. Formally,

$$\mathbb{D}^{(1)}(\underline{\Delta}) \coloneqq \Big\{ \boldsymbol{\Delta} \in \mathbb{D}_{\mathrm{cw}} : \exists\, \sigma \in \mathfrak{S}_K \text{ s.t. } i^*_{\boldsymbol{\Delta}} = \sigma(1), \text{ and } \Delta_{i^*,\cdot} = \sigma(\underline{\Delta}) \Big\}, \tag{6.13}$$

where $\mathfrak{S}_K$ is the set of permutations on $[K]$, and for any $x \in \mathbb{R}^K$, $\sigma(x) = (x_{\sigma(i)})_{i \in [K]}$. The following result is the counterpart of Theorem 6.4.1.

Theorem 6.5.1. *Let $K \geq 2$ and $T \in \mathbb{N}^*$ and consider any vector $\underline{\Delta}$ with $\Delta_1 = 0$, and $\Delta_i \in (0, 1/4)$ for $i \neq 1$. For any algorithm A with a fixed budget T, one has*

$$\max_{\boldsymbol{\Delta} \in \mathbb{D}^{(1)}(\underline{\Delta})} \mathbb{P}_{\boldsymbol{\Delta}, A} \left(\hat{\imath}_T \neq i^* \right) \geq \frac{1}{4} \exp\left(-22 \frac{T}{H_{\mathrm{cw}}} \right) ,$$

where $H_{\mathrm{cw}} = \sum_{i=2}^{K} \frac{1}{\Delta_i^2}$.

Remark 6.5.2. This theorem reveals that one can exhibit a matrix Δ for which the the exponential error decay scaling as $\exp(-T/H_{\mathrm{cw}})$ in Theorem 6.B.1 is tight. The proof can be found in Appendix 6.E.1.

We now turn to the counterpart of Theorem 6.4.2. Consider the following class of environments, for which the quantities $(\boldsymbol{s}^*_{\boldsymbol{\Delta}}, \Delta_{(\boldsymbol{s}^*)})$, defined in Section 6.4 are equal to a given couple $(\underline{\Delta}, \underline{s})$ up to a permutation of the arms. By convention, we extend $\boldsymbol{s}^*_{\boldsymbol{\Delta}} = (s^*_i)_{i \neq i^*}$ into a K-dimensional vector by fixing as 0 the i^*-th entry, that is $s^*_{i^*} \coloneqq 0$. We proceed similarly for $\Delta_{(s^*)}$.

$$\mathbb{D}^{(2)}(\underline{\Delta}, \underline{s}) \coloneqq \Big\{ \boldsymbol{\Delta} \in \mathbb{D}_{\mathrm{cw}} : \exists\, \sigma \in \mathfrak{S}_K \text{ s.t. } \boldsymbol{s}^*_{\boldsymbol{\Delta}} = \sigma(\underline{s}) \text{ and } \Delta_{(\boldsymbol{s}^*)} = \sigma(\underline{\Delta}) \Big\} , \tag{6.14}$$

Theorem 6.5.3. *Let $T \in \mathbb{N}^*$, K a multiple of 8, $\underline{\Delta} = (\Delta_1, \dots, \Delta_K)$ with $\Delta_1 = 0$ and $\Delta_i \in (0, 1/4)$ for $i \neq 1$, and $\underline{s} = (s_1, \dots, s_K)$ with $s_1 = 0$, $1 \leqslant s_i \leqslant K/4$ for $i = 2, \dots, K$. Then, any fixed-budget algorithm A satisfies*

$$\max_{\boldsymbol{\Delta} \in \mathbb{D}^{(2)}(\underline{\Delta}, \underline{s})} \mathbb{P}_{A, \boldsymbol{\Delta}}(\hat{\imath}_T \neq i^*(\boldsymbol{\Delta})) \geqslant \frac{1}{4} \exp\left(-5 \frac{T}{\max_{i=2}^{K/2} \frac{K}{s_i \Delta_i^2}} \right). \tag{6.15}$$

If, additionally $\Delta_i = \Delta > 0$ *and* $s_i = s \in [K/4]$ *for all* $i \in \{2, \dots, K/2\}$*, then*

$$\max_{\mathbf{\Delta} \in \mathbb{D}^{(2)}(\underline{\Delta}, \underline{s})} \mathbb{P}_{\mathbf{\Delta}, A}(\hat{\imath}_T \neq i^*(\mathbf{\Delta})) \geqslant \frac{1}{4} \exp\left(-10 \frac{T\Delta^2}{K}\right), \tag{6.16}$$

$$\max_{\mathbf{\Delta} \in \mathbb{D}^{(2)}(\underline{\Delta}, \underline{s})} \mathbb{P}_{\mathbf{\Delta}, A}(\hat{\imath}_T \neq i^*(\mathbf{\Delta})) \geqslant \frac{1}{2} - \sqrt{37 \frac{s\Delta^2}{K^2} T}. \tag{6.17}$$

Remark 6.5.4. Equation (6.17) shows that for matrices where half the rows equal (up to permutation) a vector with s entries at $-\Delta$ and the rest near zero, any algorithm must pay $\frac{K^2}{s\Delta^2}$ to reach a constant success probability. This is aligned with the probability-independent cost $H^{(0)}_{\text{explore}}(\underline{s})$) that suffers Algorithm 16 –see Theorem 6.B.1.

Equation (6.15) establishes probability of error is no smaller than $\exp(-T/H^{(1)}_{\text{explore}}(\underline{s}))$, while (6.16) implies that the quantity $\exp(-T/H_{\text{certify}}(\boldsymbol{s}))$ is required, at least for some specific highly structured matrices.

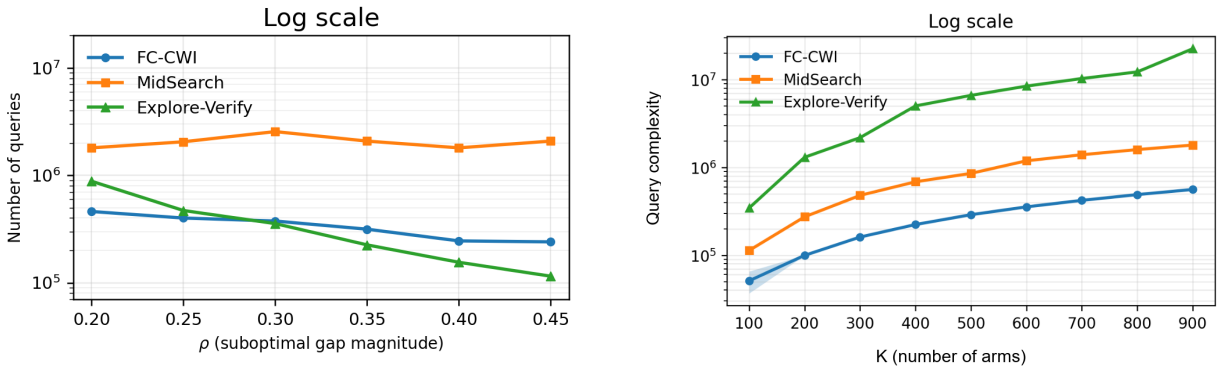
We refer to Appendix 6.E for the proofs, and a detailed discussion on the link between fixed-budget and fixed-confidence lower bounds.

6.6 Numerical simulations

We compare FC-CWI with two standard baselines: MidSearch (Maiti et al., 2024) and Explore–Verify (Karnin, 2016). Our experiments target two predictions of the theory. First, we vary the strength ρ of informative suboptimal-suboptimal comparisons, while keeping the direct CW gaps fixed, to isolate the benefit of exploiting the full gap matrix. Second, we study the scaling with the number of arms K in the same sparse informative regime. Each method is run with a fixed numerical prefactor, chosen once to compensate for conservative theoretical constants and then kept unchanged across all main-text experiments.

All instances are sparse skew-symmetric gap matrices with arm 0 as the Condorcet winner. The CW beats every other arm with margin at least γ, while the suboptimal block contains an α fraction of nonzero entries per row, with gaps equal to $\pm\rho$ and all remaining entries set to zero. Thus, γ controls the difficulty of direct CW verification, whereas ρ controls the usefulness of suboptimal-suboptimal comparisons.

We fix $K = 64$, $\alpha = 0.3$, and $\delta = 0.1$, and vary ρ in the range $[0.2, 0.45]$ with step 0.05. For the large-scale diagnostic, we fix $\gamma = 0.3$, $\rho = 0.4$, and $\alpha = 0.3$, and vary $K \in \{100, 200, \dots, 900\}$. Figure 6.1 summarizes the two diagnostics. In the left panel, FC-CWI remains more sample-efficient than MidSearch for different ρ values. Explore–Verify also improves as ρ increases and can become competitive in this small-scale setting, where finding and verifying strong opponents is cheap. This behavior is consistent with the theory: the cost of Explore–Verify can be favorable at small K, but its sample complexity carries a stronger dependence on K. This is visible in the right panel, where, as K grows, Explore–Verify scales much faster than FC-CWI, while FC-CWI keeps the most favorable query growth on the logarithmic scale.



(a) Informativeness sweep.

(b) Large-scale behavior.

Figure 6.1 – Left: query complexity in the sparse gaps matrix model with $K = 64$, $\alpha = 0.3$, and $\delta = 0.1$, as the suboptimal gap magnitude ρ varies. Right: query complexity as a function of K in the sparse model with $\gamma = 0.3$, $\rho = 0.4$, and $\alpha = 0.3$, shown on a logarithmic scale.

We report now two complementary experiments. The first is a Bradley–Terry difficulty sweep, which tests a classical structured regime with a total order. The second uses unstructured random sparse matrices, to evaluate robustness beyond the controlled circulant model used in the main text.

For the Bradley-Terry experiment, arm 0 is the Condorcet winner. We assign latent scores $\theta_i = K - i$ for $i = 0, \dots, K-1$ and set $q_{i,j} = \frac{1}{1+\exp(-\lambda_{\mathrm{BT}}(\theta_i - \theta_j))}$, $\Delta_{i,j} = q_{i,j} - \frac{1}{2}$. The parameter λ_{BT} controls the difficulty: smaller values produce smaller gaps and therefore harder instances. We fix $K = 200$, $\delta = 0.1$, and run 5 trials for $\lambda_{\mathrm{BT}} \in \{0.04, 0.05, 0.06, 0.07, 0.08\}$, with a cap of 7×10^6 pairwise queries.

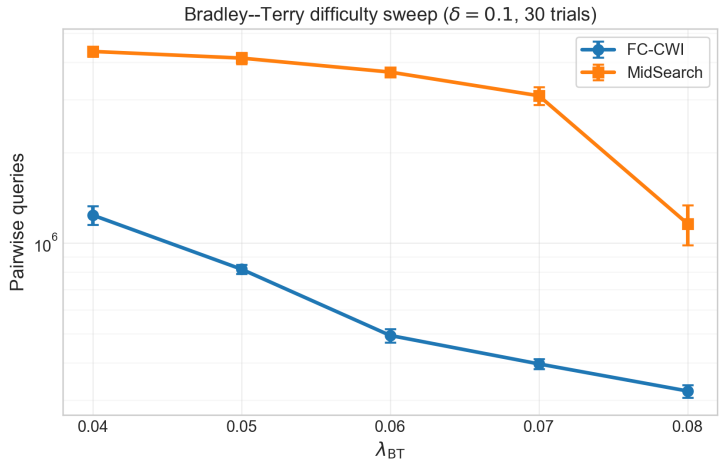


Figure 6.2 – Query complexity on Bradley–Terry instances as the difficulty parameter λ_{BT} varies. FC-CWI remains consistently more sample-efficient than MidSearch across the sweep.

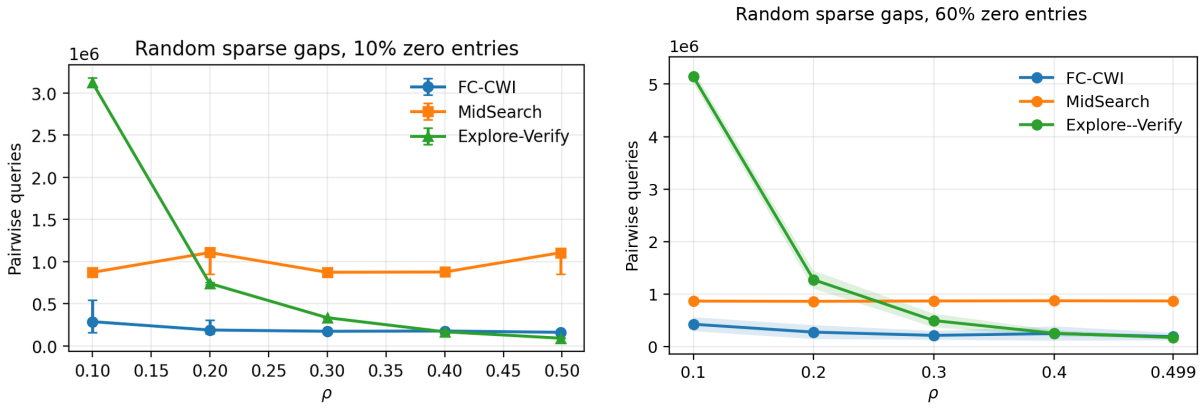


Figure 6.3 – Query complexity on random sparse gap matrices. Left: 10% zero entries. Right: 60% zero entries. FC-CWI is most effective for small and intermediate ρ, while Explore–Verify improves as stronger random witnesses become easier to find and certify.

Figure 6.2 shows that FC-CWI identifies the Condorcet winner in all runs and uses substantially fewer queries than MidSearch. The gap is most pronounced at intermediate difficulty; for instance, when $\lambda_{\mathrm{BT}} = 0.05$, FC-CWI uses about 8.4×10^5 queries on average, whereas MidSearch uses about 5.3×10^6 queries and reaches the query cap once.

We also evaluate random sparse gap matrices. Arm 0 is again the Condorcet winner, with

$$\Delta_{0,i} \sim \mathrm{Unif}(0.05, 0.1), \qquad \Delta_{i,0} = -\Delta_{0,i}.$$

For each pair of suboptimal arms, the entry is set to zero with probability p_0 and otherwise sampled from $\mathrm{Unif}[-\rho, \rho]$, with skew-symmetry imposed. We use $K = 50$, $\delta = 0.1$, and average over 10 trials for $\rho \in \{0.1, 0.2, 0.3, 0.4, 0.499\}$.

Overall, these experiments show that FC-CWI remains competitive in classical structured instances and is particularly effective when useful suboptimal-suboptimal comparisons exist but are sparse or heterogeneous.

Calibration of numerical constants. All three algorithms contain numerical constants that affect their empirical sampling budgets. Since the theoretical constants are conservative, we calibrate one scalar constant per method on a single representative instance and then keep these constants fixed in all subsequent experiments. The calibration instance is the sparse controlled model with

$$K = 128, \qquad \delta = 0.1, \qquad \alpha = 0.2, \qquad \gamma = 0.05, \qquad \rho = 0.3.$$

For FC-CWI, the scalar c_{FC} is the constant used in the elimination/certification threshold. For MidSearch, the scalar c_{MS} rescales the sampling budgets used inside the MidSearch subroutines. For Explore–Verify, the scalar c_{EV} rescales the initial budget in the doubling schedule. In all cases, the confidence radii and verification thresholds are left unchanged.

Table 6.1 – Selected calibration constants on the sparse controlled instance. All selected constants identify the Condorcet winner in all 10 paired trials.

Method	Selected constant	Errors / 10	Median queries	90% quantile
FC-CWI	0.5	0/10	1.95×10^6	2.14×10^6
MidSearch	0.3	0/10	8.42×10^6	8.75×10^6
Explore–Verify	4	0/10	4.00×10^6	4.03×10^6

To make the comparison insensitive to random fluctuations, we use paired trials: for a fixed method and trial index, all constants are evaluated on the same randomly permuted instance and with the same initial random seed. Each entry below is computed over 10 trials.

Table 6.2 – Robustness of the calibration. Query-complexity ranges are computed over local sweeps around the selected constants. All entries in the table have zero errors over 10 paired trials.

Method	Tested constants	Median query range	90% quantile range
FC-CWI	$\{0.5, 0.7, 0.9, 1.1, 1.3, 1.5\}$	$[1.95, 2.11]\times10^6$	$[2.14, 2.14]\times10^6$
MidSearch	$\{0.2, 0.3, 0.5\}$	$[6.09, 13.08]\times10^6$	$[6.44, 13.12]\times10^6$
Explore–Verify	$\{1, 2, 4\}$	$[4.00, 4.02]\times10^6$	$[4.03, 4.05]\times10^6$

The stability table shows that the empirical conclusions are not driven by a finely tuned constant. FC-CWI remains essentially unchanged over the tested range of c_{FC}, while Explore–Verify is also stable near the selected constant. MidSearch is more sensitive because its scalar directly multiplies several internal sampling allocations, but the selected value $c_{\mathrm{MS}} = 0.3$ lies in a clean zero-error regime and the resulting query complexity remains well separated from FC-CWI on this calibration instance.

6.7 Discussion

In this manuscript, we consider δ-PAC and fixed-budget Condorcet-winner identification in stochastic dueling bandits under the sole assumption that a CW exists. We derive instance-dependent, high-probability sample-complexity guarantees that exploit the full gap matrix, and complement them with new lower bounds covering both large-scale regimes, where K is large, and small probability regimes, where δ is small. We further improve over the state-of-the-art under additional structural assumptions such as weak stochastic transitivity (WST), both on the upper- and lower-bound sides– see the paragraph above. More broadly, CWI is a structured pure-exploration problem over a latent matrix, with close connections to noisy payoff matrix games and Nash equilibrium identification (Zhao et al., 2023; Maiti et al., 2024, 2025; Ito et al., 2025). We believe that the exploration-certification decomposition and the tail-oriented lower-bound viewpoint may help clarify analogous expectation-versus-high-probability separations in equilibrium learning, whose optimal instance-dependent query complexity remains largely open; see Maiti

(2025).

Total order implies a Condorcet winner. Weak stochastic transitivity (WST) is a standard sufficient condition for a latent total order: for any $i, j, k \in [K]$, $\Delta_{i,j} \geq 0$ and $\Delta_{j,k} \geq 0$ imply $\Delta_{i,k} \geq 0$. Thus (up to tie-breaking) preferences are transitive and admit a total order, whose maximal element is a Condorcet winner (CW). CW/optimal-arm identification in such total-order regimes is studied in, e.g., Falahatgar et al. (2017, 2018), which provide worst-case guarantees under WST (notably an $\mathcal{O}(K\varepsilon^{-2}\log(K/\delta))$ bound for (ε, δ)-maxing, i.e., returning an arm I such that $\mathbb{P}(\Delta_{I,i^*} \leq -\varepsilon) \leq \delta)$). Since WST is strictly stronger than merely assuming the existence of a CW, our CW-based bounds directly apply under WST. In particular, our bound for FC-CW1 in Theorem 6.3.1 improves over the state-of-the art under WST without relying on that assumption. Furthermore, our lower bounds in Theorems 6.4.1 and 6.4.2 provide novel distribution-dependent and minimax lower bounds under the WST assumption. Indeed, regarding Theorem 6.4.2, if the matrix $\mathbf{\Delta}$ satisfies WST, then the class $\mathbb{D}(\mathbf{\Delta})$ defined in (6.10) considered in that theorem only contains gap matrices satisfying WST. A stronger assumption is strong stochastic transitivity (SST), which adds magnitude constraints (for ordered $i \succ j \succ k$, $\Delta_{i,k} \geq \max\{\Delta_{i,j}, \Delta_{j,k}\}$); in particular, SST implies a CW and ensures that each suboptimal arm attains its largest loss against the CW. Instance-dependent sample complexity under SST (often with mild regularity such as STI) is characterized in Falahatgar et al. (2017); Ren et al. (2020), and our upper bounds recover these guarantees (of the order $H_{\mathrm{cw}}(\delta)$) up to logarithmic factors without even relying on the SST assumption. Finally, parametric random-utility models such as Bradley–Terry–Luce, Plackett–Luce, and Thurstone (Bradley and Terry, 1952; Luce et al., 1959; Plackett, 1975; Thurstone, 2017) are more restrictive than SST and therefore fall within our scope; in particular, our bounds recover existing guarantees for BTL (see, e.g., Ren et al. (2020)). See Bengs et al. (2021) for a broader overview.

Works on other types of winners. Beyond the CW objective (Komiyama et al., 2015; Ailon et al., 2014; Chen and Frazier, 2017; Saha and Gaillard, 2022; Peköz et al., 2022), alternative notions include the *Borda winner* (Chen et al., 2020; Jamieson et al., 2015) (maximizing average pairwise advantage) and the *Copeland winner* (Zoghi et al., 2015a; Komiyama et al., 2016) (maximizing the number of beaten opponents). Since the Borda and Condorcet winners can differ, Borda-specific guarantees do not transfer to our setting. Copeland winners always exist (possibly non-unique) and generalize the CW; fixed-confidence identification is studied in Zoghi et al. (2015a), but when specialized to CW instances the resulting complexity is $\mathcal{O}\left(\max_{i\in[K]} \sum_{j\neq i} \log(1/\delta)/\Delta_{i,j}^2\right)$, which is looser than our bounds.

Appendix of Chapter 6

6.A Proof of Section 6.2

6.A.1 Proof of Theorem 6.2.1

Since the right-hand side of (6.6) does not depend on the values of $\mu_{(1)}, \dots, \mu_{(K)}$, it suffices to treat the strictly ordered case where $\mu_{(1)} < \mu_{(2)} < \cdots < \mu_{(K)}$; the case where some values are perhaps identical follows by a continuity argument.

Proof. If $T \leq \frac{128K}{u-d} \log_2\left(\frac{128K}{u-d}\right)$ the bound is vacuous. Assume that $T \geq \frac{128K}{u-d} \log_2\left(\frac{128K}{u-d}\right)$ so that $\log_2(\log_2(T)) \geq 1$ and $\ell_{\min} \leq L-1$. We introduce the following additional notation, for each $\ell \in \{\ell_{\min}, \dots, L-1\}$ where $\ell_{\min} = \lceil * \rceil \log_2\left(\frac{16K}{u-d}\right)$, let

$$N_\ell := |\mathcal{A}_\ell| = \lfloor * \rfloor \frac{\epsilon_\ell^2 T}{\log\left(\frac{16K}{u-d}\right) \log_2(T)} \qquad \text{and} \qquad T_\ell := \lceil * \rceil \frac{\log\left(\frac{16K}{u-d}\right)}{2\epsilon_\ell^2} \,, \tag{6.18}$$

and let

$$r_0 := \frac{d}{K} \quad , \quad r_1 := \frac{3d+u}{4K} \quad , \quad r_2 := \frac{d+u}{2K} \quad , \quad r_3 := \frac{d+3u}{4K} \quad \text{and} \quad r_4 := \frac{u}{K} \,.$$

The proof follows the steps below

— We start by a sanity check, verifying that the total number of queries made by Algorithm 15 is at most T, and that $\bar{\ell}$ exists.

— Next, we show an intermediary result about the quantities $\hat{t}_\ell^{(i)}$ for $i \in \{1,2,3\}$ in the form of an upper-bound on

$$\mathbb{P}\left(\hat{t}_\ell^{(i)} \notin \left[\mu_{\left(\lceil * \rceil \frac{r_{i-1}+r_i}{2} \cdot K\right)} - \epsilon_\ell,\ \mu_{\left(\lceil * \rceil \frac{r_i+r_{i+1}}{2} \cdot K\right)} + \epsilon_\ell\right]\right) \,.$$

— Finally, we build on the obtained intermediary result to prove that the way $\bar{\ell}$ is defined allows to have the stated guarantees.

Sanity checks: Recall the expressions $L = \lfloor * \rfloor \log_2(T/\log_2(T))$, $\epsilon_\ell = 2.\ 2^{-(L-\ell)/2}$ and $|\mathcal{A}_\ell| = \lfloor * \rfloor \frac{\epsilon_\ell^2 T}{\log\left(\frac{16K}{u-d}\right) \log_2(T)}$. Algorithm 15 comprises $L - \ell_{\min}$ iterations, for each iteration $\ell \in \{\ell_{\min}, \dots, L-$

1} it makes $|\mathcal{A}_\ell|\, T_\ell$ queries. Thus, the total number of queries is

$$\begin{aligned}
\sum_{\ell=\ell_{\min}}^{L-1} |\mathcal{A}_\ell| \cdot \lceil * \rceil \, \frac{\log\left(\frac{16K}{u-d}\right)}{2\epsilon_\ell^2} &= \sum_{\ell=\ell_{\min}}^{L-1} \lfloor * \rfloor \, \frac{\epsilon_\ell^2 T}{\log\left(\frac{16K}{u-d}\right)\log_2(T)} \cdot \lceil * \rceil \, \frac{\log\left(\frac{16K}{u-d}\right)}{2\epsilon_\ell^2} \\
&\le \sum_{\ell=\ell_{\min}}^{L-1} \frac{\epsilon_\ell^2 T}{\log\left(\frac{16K}{u-d}\right)\log_2(T)} \cdot \left(\frac{\log\left(\frac{16K}{u-d}\right)}{2\epsilon_\ell^2} + 1 \right) \\
&= \sum_{\ell=\ell_{\min}}^{L-1} \frac{T}{2\log_2(T)} + \frac{\epsilon_\ell^2 T}{\log\left(\frac{16K}{u-d}\right)\log_2(T)} \\
&< \frac{T}{2} + \frac{T}{\log\left(\frac{16K}{u-d}\right)\log_2(T)} \sum_{\ell=\ell_{\min}}^{L-1} \epsilon_\ell^2 \\
&\le \frac{T}{2} + \frac{4T}{\log\left(\frac{16K}{u-d}\right)\log_2(T)} \le T \,,
\end{aligned}$$

where we used in the last line the threshold condition on T, giving

$$\log(16K/(u-d))\log_2(T) \ge 8\,.$$

For the definition of the quantity $\bar{\ell}$, note that the set over which the minimum is taken is not empty, since it always contains $L-1$.

Step 2: Per-level quantile control.

In this step we will prove that for every level $\ell \in \{\ell_{\min}, \dots, L-1\}$ and every $i \in \{1,2,3\}$

$$\mathbb{P}\Big(\hat{t}_\ell^{(i)} \notin C_{\ell,i}\Big) \le p_\ell\,, \tag{6.19}$$

where we define (for each ℓ and $i \in \{1,2,3\}$)

$$\begin{aligned}
C_{\ell,i} &:- \left[\mu\Big(\Big\lceil \tfrac{r_{i-1}+r_i}{2} K \Big\rceil\Big) - \epsilon_\ell,\ \mu\Big(\Big\lceil \tfrac{r_i+r_{i+1}}{2} K \Big\rceil\Big) + \epsilon_\ell \right], \\
\kappa_{d,u} &:- \min\Big\{\frac{d}{K}, 1-\frac{u}{K}\Big\}\Big(\frac{u-d}{60K}\Big)^2, \qquad p_\ell :- 4\exp(-\kappa_{d,u} N_\ell)\,.
\end{aligned}$$

Throughout this step, fix $\ell \in \{\ell_{\min}, \dots, L-1\}$ and $i \in \{1,2,3\}$. Define the two (random-sample) ranks

$$r_- :- \left\lceil \frac{r_{i-1}+2r_i}{3} N_\ell \right\rceil, \qquad r_+ :- \left\lceil \frac{2r_i+r_{i+1}}{3} N_\ell \right\rceil,$$

and define the two (population) bracket points

$$m_- :- \mu\Big(\Big\lceil \tfrac{r_{i-1}+r_i}{2} K \Big\rceil\Big), \qquad m_+ :- \mu\Big(\Big\lceil \tfrac{r_i+r_{i+1}}{2} K \Big\rceil\Big)\,.$$

Let $\gamma_1, \ldots, \gamma_{N_\ell}$ be the (random) true means of the sampled multiset $\mathcal{A}_\ell$, and let $\gamma_{(1)} \leq \cdots \leq \gamma_{(N_\ell)}$ be their order statistics (ties broken arbitrarily).

Next, introduce the event $E^{(i)}$ given by

$$E^{(i)} :- \left\{\gamma_{(r_-)} < m_-\right\} \ \cup \ \left\{\gamma_{(r_+)} > m_+\right\}.$$

Then, by a union bound,

$$\mathbb{P}\Big(\hat{t}_\ell^{(i)} \notin C_{\ell,i}\Big) \leq \underbrace{\mathbb{P}\Big(E^{(i)}\Big)}_{\text{Term 1}} + \underbrace{\mathbb{P}\Big(\hat{t}_\ell^{(i)} \notin C_{\ell,i} \text{ and } \neg E^{(i)}\Big)}_{\text{Term 2}}. \tag{6.20}$$

Next we will control the probability of $E^{(i)}$ (Term 1). Define the counts

$$M_1 :- \left|\{j \in [N_\ell] : \gamma_j < m_-\}\right|, \qquad M_2 :- \left|\{j \in [N_\ell] : \gamma_j > m_+\}\right|.$$

Since $\mathcal{A}_\ell$ is obtained by sampling arms i.i.d. with replacement from $[K]$, these are binomials:

$$M_1 \sim \mathrm{Bin}\left(N_\ell, \ \frac{\left\lceil \frac{r_{i-1}+r_i}{2} K \right\rceil - 1}{K}\right), \qquad M_2 \sim \mathrm{Bin}\left(N_\ell, \ 1 - \frac{\left\lceil \frac{r_i+r_{i+1}}{2} K \right\rceil}{K}\right).$$

Moreover, by definition of order statistics we have

$$\{\gamma_{(r_-)} < m_-\} \subseteq \{M_1 \geq r_-\}, \qquad \{\gamma_{(r_+)} > m_+\} \subseteq \{M_2 \geq N_\ell - r_+ + 1\}.$$

Therefore,

$$\mathbb{P}\Big(E^{(i)}\Big) \leq \mathbb{P}(M_1 \geq r_-) \ + \ \mathbb{P}(M_2 \geq N_\ell - r_+ + 1).$$

Let us bound the two terms in the upper bound above using binomial tail bounds. Using $\lceil x \rceil - 1 \leq x$ and $\lceil x \rceil \geq x$, we have the parameter bounds

$$\frac{\left\lceil \frac{r_{i-1}+r_i}{2} K \right\rceil - 1}{K} \ \leq \ \frac{r_{i-1}+r_i}{2}, \qquad 1 - \frac{\left\lceil \frac{r_i+r_{i+1}}{2} K \right\rceil}{K} \ \leq \ 1 - \frac{r_i + r_{i+1}}{2}.$$

Hence, M_1 and M_2 are stochastically dominated by $\mathrm{Bin}(N_\ell, \frac{r_{i-1}+r_i}{2})$ and $\mathrm{Bin}(N_\ell, 1 - \frac{r_i+r_{i+1}}{2})$, respectively. Also, by construction we have

$$\frac{r_{i-1}+2r_i}{3} - \frac{r_{i-1}+r_i}{2} = \frac{r_i - r_{i-1}}{6}, \qquad \Big(1 - \frac{2r_i + r_{i+1}}{3}\Big) - \Big(1 - \frac{r_i + r_{i+1}}{2}\Big) = \frac{r_{i+1} - r_i}{6}.$$

Applying Hoeffding's inequality to these dominating binomials yields

$$\mathbb{P}(M_1 \geq r_-) \leq \exp\left(-2N_\ell \Big(\frac{r_i - r_{i-1}}{6}\Big)^2\right),$$
$$\mathbb{P}(M_2 \geq N_\ell - r_+ + 1) \leq \exp\left(-2N_\ell \Big(\frac{r_{i+1} - r_i}{6}\Big)^2\right).$$

Since for each $j \in \{1,2,3,4\}$ one has $r_j - r_{j-1} \geq \frac{u-d}{4K}$, we obtain

$$\text{Term } 1 = \mathbb{P}\Big(E^{(i)}\Big) \leq 2\exp\left(-\frac{N_\ell (u-d)^2}{288K^2}\right). \tag{6.21}$$

Next, let us upper bound Term 2 in (6.20). On $\neg E^{(i)}$ we have $\gamma_{(r_-)} \geq m_-$ and $\gamma_{(r_+)} \leq m_+$, hence

$$[m_- - \epsilon_\ell,\ m_+ + \epsilon_\ell] \supseteq [\gamma_{(r_-)} - \epsilon_\ell,\ \gamma_{(r_+)} + \epsilon_\ell].$$

Therefore,

$$\begin{aligned}\text{Term } 2 &= \mathbb{P}\Big(\hat{t}_\ell^{(i)} \notin [m_- - \epsilon_\ell, m_+ + \epsilon_\ell] \text{ and } \neg E^{(i)}\Big) \\ &\leq \mathbb{P}\Big(\hat{t}_\ell^{(i)} \notin [\gamma_{(r_-)} - \epsilon_\ell,\ \gamma_{(r_+)} + \epsilon_\ell]\Big).\end{aligned}$$

By Lemma 6.A.2, this implies

$$\text{Term } 2 \leq 2\exp(-\kappa_{d,u} N_\ell). \tag{6.22}$$

Combining (6.20), (6.21) and (6.22), and using that $\kappa_{d,u} \leq (u-d)^2/(288K^2)$, we obtain

$$\mathbb{P}\Big(\hat{t}_\ell^{(i)} \notin C_{\ell,i}\Big) \leq 4\exp(-\kappa_{d,u} N_\ell) = p_\ell\,,$$

which is exactly (6.19).

Step 3: Conclusion.

If $\epsilon < 3\epsilon_{\ell_{\min}}$, then the upper bound of the theorem is greater than 1 and the bound is vacuous. Assume that $\epsilon \geq 3\epsilon_{\ell_{\min}}$. Let $\ell^\star$ be the largest level such that

$$3\epsilon_{\ell^\star} \leq \epsilon\ .$$

This implies in particular, since $\ell^\star + 1$ violates the condition above, that $\epsilon < 3\epsilon_{\ell^\star+1} = 3\sqrt{2}\epsilon_{\ell^\star}$, therefore

$$\epsilon_{\ell^\star} \geq \frac{\epsilon}{3\sqrt{2}}\ . \tag{6.23}$$

Next, we will prove that for any $\ell \in \{\ell_{\min}, \dots, L-1\}$, we have

$$\mathbb{P}(\hat{t}_{\bar{\ell}}^{(2)} \notin [\mu_{(d)} - 3\epsilon_\ell,\ \mu_{(u)} + 3\epsilon_\ell]) \leq 2(4L+1)p_\ell\ .$$

Let $\ell \in \{\ell_{\min}, \dots, L-1\}$, recall that by definition of $\bar{\ell}$, for $l \geqslant \bar{\ell}$, one has $\hat{t}_\ell^{(2)} \leqslant\ \hat{t}_{\ell'}^{(3)} + 2\epsilon_{\ell'}$. Then, it holds that

$$\mathbb{P}\left(\hat{t}_{\bar{\ell}}^{(2)} > \mu_{(u)} + 3\epsilon_\ell\right) \leq \mathbb{P}(\bar{\ell} > \ell) + \mathbb{P}(\hat{t}_\ell^{(3)} > \mu_{(u)} + \epsilon_\ell) \leq 4Lp_\ell + p_\ell\ ,$$

where we use Lemma 6.A.1, which ensures that for every $\ell \in \{\ell_{\min}, \dots, L-1\}$ we have $\mathbb{P}(\bar{\ell} > \ell) \leq$

$4Lp_\ell$, and we use the Bound 6.19 from step 2 with $i = 3$. The second bound

$$\mathbb{P}\left(\hat{t}^{(2)}_{\bar{\ell}} < \mu_{(d)} - 3\epsilon_\ell\right) \leq (4L+1)p_\ell \ ,$$

is proven using the same arguments (in particular Bound 6.19 with $i = 1$).
Applying this bound to $\ell^\star$, using $3\epsilon_{\ell^\star} \leqslant \epsilon$, we have

$$\begin{aligned}\mathbb{P}\left(\hat{t}^{(2)}_{\bar{\ell}} \notin [\mu_{(d)} - \epsilon,\ \mu_{(u)} + \epsilon]\right) &\leq \mathbb{P}\left(\hat{t}^{(2)}_{\bar{\ell}} \notin [\mu_{(d)} - 3\epsilon_{\ell^\star},\ \mu_{(u)} + 3\epsilon_{\ell^\star}]\right) \\ &\leq 2(4L+1)p_{\ell^\star} \ .\end{aligned}$$

Next, in order to upper bound $p_{\ell^\star}$ we use the following lower bound on $N_{\ell^\star}$

$$\begin{aligned}N_{\ell^\star} = \lfloor * \rfloor \frac{\epsilon^2_{\ell^\star} T}{\log\left(\frac{16K}{u-d}\right)\log_2(T)} &\geq \frac{1}{2}\frac{\epsilon^2_{\ell^\star} T}{\log\left(\frac{16K}{u-d}\right)\log_2(T)} \\ &\geq \frac{\epsilon^2 T}{36\log\left(\frac{16K}{u-d}\right)\log_2(T)} \ ,\end{aligned}$$

where we use the fact that from the assumption on the budget T, $N_{\ell^\star} \geqslant 2$, and $\epsilon_{\ell^\star} \geqslant \epsilon/3\sqrt{2}$ (see (6.23)). Therefore, using the definition of p_ℓ,

$$\begin{aligned}p_{\ell^\star} &= 4\exp(-\kappa_{d,u} \cdot N_{\ell^\star}) \\ &\leq 4\exp\left(-\min\left\{\frac{d}{K},\ 1-\frac{u}{K}\right\}\left(\frac{u-d}{60K}\right)^2 \cdot \frac{\epsilon^2 T}{36\log\left(\frac{16K}{u-d}\right)\log_2(T)}\right) \\ &= 4\exp\left(-\frac{cr\epsilon^2 T}{\log\left(\frac{16K}{u-d}\right)\log_2(T)}\right) \ ,\end{aligned}$$

where $c > 0$ is an absolute constant, and $r = \min\left\{\frac{d}{K},\ 1-\frac{u}{K}\right\}\left(\frac{u-d}{K}\right)^2$. Finally, given that $L \leq \log_2(T)$ and $2(4L+1)\cdot 4 \leq 40\log_2(T)$ for $T \geq 2$,

$$\mathbb{P}\left(\hat{t}^{(2)}_{\bar{\ell}} \notin [\mu_{(d)} - \epsilon,\ \mu_{(u)} + \epsilon]\right) \leq 40\log_2(T)\exp\left(-\frac{cr\epsilon^2 T}{\log\left(\frac{16K}{u-d}\right)\log_2(T)}\right) \ ,$$

which is the desired bound. □

Below are two technical lemmas deferred here to avoid cluttering the proof above.

Lemma 6.A.1. *For every $\ell \in \{\ell_{\min}, \dots, L-1\}$, we have $\mathbb{P}(\bar{\ell} > \ell) \leq 4Lp_\ell$.*

Proof. Suppose that $\bar{\ell} > \ell$, then using the definition of $\bar{\ell}$ we have necessarily that there exists

$\ell' \geqslant \ell$ such that $\hat{t}_\ell^{(2)} \notin I_{\ell'}$, with $I_{\ell'} = \left[\hat{t}_{\ell'}^{(1)} - 2\epsilon_{\ell'},\ \hat{t}_{\ell'}^{(3)} + 2\epsilon_{\ell'}\right]$. Therefore,

$$\mathbb{P}(\bar{\ell} > \ell) \leq \sum_{\ell' \geqslant \ell} \mathbb{P}\left(\hat{t}_\ell^{(2)} < \hat{t}_{\ell'}^{(1)} - 2\epsilon_{\ell'}\right) + \sum_{\ell' \geqslant \ell} \mathbb{P}\left(\hat{t}_\ell^{(2)} > \hat{t}_{\ell'}^{(3)} + 2\epsilon_{\ell'}\right) \ .$$

Let

$$m_- \coloneqq \mu_{\left(\lceil * \rceil \frac{r_1+r_2}{2} K\right)}, \quad m_+ \coloneqq \mu_{\left(\lceil * \rceil \frac{r_2+r_3}{2} K\right)} \ .$$

Let $\ell' \geqslant \ell$, the event $\hat{t}_\ell^{(2)} < \hat{t}_{\ell'}^{(1)} - 2\epsilon_{\ell'}$ implies $\hat{t}_{\ell'}^{(1)} > m_- + \epsilon_{\ell'}$ or $\hat{t}_\ell^{(2)} < m_- - \epsilon_{\ell'}$. Since we have $\epsilon_{\ell'} \geq \epsilon_\ell$ for $\ell' \geqslant \ell$, Bound 6.19 yields $\mathbb{P}(\hat{t}_{\ell'}^{(1)} > m_- + \epsilon_{\ell'}) \leq p_{\ell'}$ and

$$\begin{aligned}\mathbb{P}(\hat{t}_\ell^{(2)} < m_- - \epsilon_{\ell'}) &\leq \mathbb{P}(\hat{t}_\ell^{(2)} < m_- - \epsilon_\ell) \\ &\leq p_\ell \ .\end{aligned}$$

Therefore,

$$\mathbb{P}(\hat{t}_\ell^{(2)} < \hat{t}_{\ell'}^{(1)} - 2\epsilon_{\ell'}) \leq p_\ell + p_{\ell'} \ .$$

Similarly,

$$\mathbb{P}(\hat{t}_\ell^{(2)} > \hat{t}_{\ell'}^{(3)} + 2\epsilon_{\ell'}) \leq p_\ell + p_{\ell'} \ .$$

Therefore, for every $\ell' \geqslant \ell$,

$$\mathbb{P}(\hat{t}_\ell^{(2)} \notin I_{\ell'}) \leq 2p_\ell + 2p_{\ell'} \ .$$

Summing over $\ell' \geqslant \ell$,

$$\mathbb{P}(\bar{\ell} > \ell) \leq \sum_{\ell' \geqslant \ell} (2p_\ell + 2p_{\ell'}) \leq 2Lp_\ell + 2\sum_{\ell' \geqslant \ell} p_{\ell'} \ .$$

Since $N_{\ell'}$ increases with ℓ' (thus $p_{\ell'}$ is decreasing), we have $\sum_{\ell' \geqslant \ell} p_{\ell'} \leq Lp_\ell$, which gives $\mathbb{P}(\bar{\ell} > \ell) \leq 4Lp_\ell$. □

Lemma 6.A.2. *Let $\ell \in \{\ell_{\min}, \dots, L-1\}$ and consider the notation introduced in the proof of Theorem 6.2.1. For each $i \in \{1,2,3\}$, define the two indices*

$$r_- \coloneqq \lceil * \rceil \frac{r_{i-1} + 2r_i}{3} N_\ell \quad \textit{and} \quad r_+ \coloneqq \lceil * \rceil \frac{2r_i + r_{i+1}}{3} N_\ell \, .$$

Then

$$\mathbb{P}\left(\hat{t}_\ell^{(i)} \notin \left[\gamma_{(r_-)} - \epsilon_\ell, \gamma_{(r_+)} + \epsilon_\ell\right] \mid \mathcal{A}_\ell\right) \leq 2\exp\left(-\kappa_{d,u} \cdot N_\ell\right) \ .$$

Proof. Fix $\ell \in \{\ell_{\min}, \dots, L-1\}$ and $i \in \{1,2,3\}$. Define $q = \lceil r_i N_\ell \rceil$. Given that we have $T_\ell = \lceil \log(16K/(u-d))/(2\epsilon_\ell^2) \rceil$, let

$$\begin{aligned}\delta_\ell &\coloneqq \exp\left(-2T_\ell \epsilon_\ell^2\right) \\ &\leq \exp\left(-\log(16K/(u-d))\right) = \frac{u-d}{16K} \, . \end{aligned} \tag{6.24}$$

Moreover, for every arm j, given that samples are 1-range bounded and $\hat{\mu}_j$ is computed with T_ℓ samples, Hoeffding's inequality gives

$$\mathbb{P}(\hat{\mu}_j \leq \gamma_j - \epsilon_\ell \mid \mathcal{A}_\ell) \leq \delta_\ell, \quad \mathbb{P}(\hat{\mu}_j \geq \gamma_j + \epsilon_\ell \mid \mathcal{A}_\ell) \leq \delta_\ell \,. \tag{6.25}$$

Next, we prove the lower tail of the claimed bound. Recall that $\hat{t}_\ell^{(i)} = \hat{\mu}_{(q)}$ and $r_- :- \lceil * \rceil \frac{r_{i-1}+2r_i}{3} N_\ell$ and define the event

$$\mathcal{E}_- :- \{\hat{\mu}_{(q)} < \gamma_{(r_-)} - \epsilon_\ell\} \,.$$

If $\mathcal{E}_-$ occurs, then at least q empirical means are smaller than $\gamma_{(r_-)} - \epsilon_\ell$ (with $q \geqslant r_-$). Given that $(\gamma_{(k)})_k$ are in non-decreasing order, we conclude that among the set

$$\mathcal{G}_- :- \{j : \gamma_j \geq \gamma_{(r_-)}\} \,,$$

then at least $q - (r_- - 1)$ elements must satisfy $\hat{\mu}_j < \gamma_{(r_-)} - \epsilon_\ell$. For each element $j \in \mathcal{G}_-$ we have $\gamma_j \geq \gamma_{(r_-)}$, hence $\{\hat{\mu}_j < \gamma_{(r_-)} - \epsilon_\ell\} \subseteq \{\hat{\mu}_j < \gamma_j - \epsilon_\ell\}$, using (6.24) and (6.25) we have that, conditionally on $\mathcal{A}_\ell$, each such downward deviation has probability at most $(u-d)/(16K)$. Therefore, we conclude that

$$\mathbb{P}(\mathcal{E}_- \mid \mathcal{A}_\ell) \leq \mathbb{P}\left(\mathrm{Bin}(|\mathcal{G}_-|, (u-d)/(16K)) \geq q - (r_- - 1)\right) .$$

Using the definitions of q, r_- and r_i, we have

$$\begin{aligned} q - (r_- - 1) &= \lceil * \rceil \, r_i N_\ell - \left(\lceil * \rceil \, \frac{r_{i-1} + 2r_i}{3} N_\ell - 1 \right) \\ &\geq r_i N_\ell - \frac{r_{i-1} + 2r_i}{3} N_\ell \\ &= \frac{r_i - r_{i-1}}{3} N_\ell = \frac{u-d}{12K} N_\ell \\ &\geq \frac{u-d}{12K} |\mathcal{G}_-| \,. \end{aligned}$$

Applying Hoeffding's inequality for binomials yields

$$\begin{aligned} \mathbb{P}(\mathcal{E}_- \mid \mathcal{A}_\ell) &\leq \exp\left(-2 \, |\mathcal{G}_-| \left(\frac{u-d}{12K} - \frac{u-d}{16K} \right)^2 \right) \\ &\leq \exp\left(-2 \, |\mathcal{G}_-| \left(\frac{u-d}{48K} \right)^2 \right) . \end{aligned}$$

Finally, for $i \in \{1,2,3\}$ we have $\frac{r_{i-1}+2r_i}{3} \leq \frac{u}{K}$, hence $|\mathcal{G}_-| = N_\ell - r_- + 1 \geq (1 - u/K) N_\ell$. Therefore,

$$\begin{aligned} \mathbb{P}(\mathcal{E}_- \mid \mathcal{A}_\ell) &\leq \exp\left(-2 \left(1 - \frac{u}{K} \right) N_\ell \left(\frac{u-d}{48K} \right)^2 \right) \\ &\leq \exp\left(-\kappa_{d,u} \cdot N_\ell \right) . \end{aligned} \tag{6.26}$$

Let us show the upper tail of the claimed bound. We follow similar steps as in the lower tail proof. Consider $r_+ :- \lceil * \rceil \frac{2r_i + r_{i+1}}{3} N_\ell$ and define

$$\mathcal{E}_+ :- \{\hat{\mu}_{(q)} > \gamma_{(r_+)} + \epsilon_\ell\}\,.$$

If $\mathcal{E}_+$ occurs, then at least $N_\ell - q + 1$ empirical means exceed $\gamma_{(r_+)} + \epsilon_\ell$. At most $N_\ell - r_+$ arms can have true mean larger than $\gamma_{(r_+)}$, therefore at least $(N_\ell - q + 1) - (N_\ell - r_+) = r_+ - q + 1$ arms from the set

$$\mathcal{G}_+ :- \{j : \gamma_j \leq \gamma_{(r_+)}\}$$

must satisfy $\hat{\mu}_j > \gamma_{(r_+)} + \epsilon_\ell$. For each $j \in \mathcal{G}_+$ we have $\gamma_j \leq \gamma_{(r_+)}$, therefore $\{\hat{\mu}_j > \gamma_{(r_+)} + \epsilon_\ell\} \subseteq \{\hat{\mu}_j \geq \gamma_j + \epsilon_\ell\}$, using (6.24) and (6.25) we have that, conditionally on $\mathcal{A}_\ell$, each such upward deviation has probability at most $(u-d)/(16K)$ conditionally on $\mathcal{A}_\ell$. Thus, conditionally on $\mathcal{A}_\ell$, we have

$$\mathbb{P}(\mathcal{E}_+ \mid \mathcal{A}_\ell) \leq \mathbb{P}\left(\mathrm{Bin}(|\mathcal{G}_+|\,, (u-d)/(16K)) \geq r_+ - q + 1\right)\,.$$

Also, we have

$$\begin{aligned} r_+ - q + 1 &\geq \frac{2r_i + r_{i+1}}{3} N_\ell - r_i N_\ell \\ &= \frac{r_{i+1} - r_i}{3} N_\ell = \frac{u-d}{12K} N_\ell\,. \end{aligned}$$

Therefore, using the binomial Hoeffding bound we obtain

$$\begin{aligned} \mathbb{P}(\mathcal{E}_+ \mid \mathcal{A}_\ell) &\leq \exp\left(-2\,|\mathcal{G}_+| \left(\frac{u-d}{48K}\right)^2\right) \\ &\leq \exp\left(-2r_+ \left(\frac{u-d}{48K}\right)^2\right)\,. \end{aligned}$$

Moreover, for $i \in \{1,2,3\}$ we have $\frac{2r_i + r_{i+1}}{3} \geq \frac{d}{K}$, so $r_+ \geq \frac{d}{K} N_\ell$. Hence,

$$\begin{aligned} \mathbb{P}(\mathcal{E}_+ \mid \mathcal{A}_\ell) &\leq \exp\left(-2\frac{d}{K} N_\ell \left(\frac{u-d}{48K}\right)^2\right) \\ &\leq \exp\left(-\kappa_{d,u} \cdot N_\ell\right)\,. \end{aligned} \tag{6.27}$$

The conclusion follows by combining (6.26), and (6.27) which leads to the bound

$$\mathbb{P}(\hat{\mu}_{(q)} \notin [\gamma_{(r_-)} - \epsilon_\ell,\ \gamma_{(r_+)} + \epsilon_\ell]) \leq 2\exp\left(-\kappa_{d,u} N_\ell\right)\,.$$

□

6.A.2 Proof of Corollary 6.2.2

Proof of Corollary 6.2.2. Suppose $K \geq 5$, then $\lceil K/4 \rceil > \lceil K/8 \rceil$. Let $\epsilon > 0$ and $I = [\mu_{(\lceil K/8 \rceil)} - \epsilon, \mu_{(\lceil K/4 \rceil)} + \epsilon]$. We have

$$\min\left\{\frac{\lceil K/8 \rceil}{K}, 1 - \frac{\lceil K/4 \rceil}{K}\right\} \geq \min\left\{\frac{1}{8}, 1 - \frac{K/4+1}{K}\right\} \geq \frac{1}{8} \,. \tag{6.28}$$

Moreover, using $K \geq 5$ and $K = 8q + r$ where $q \in \mathbb{N}$ and $r \in \{0, \dots, 7\}$, we show that

$$\left(\frac{\lceil K/4 \rceil - \lceil K/8 \rceil}{K}\right)^2 \geq \frac{1}{144} \,. \tag{6.29}$$

Applying Theorem 6.2.1 with $u = \lceil K/4 \rceil$, $d = \lceil K/8 \rceil$ and using the bounds (6.28) and (6.29) we obtain

$$\begin{aligned}\mathbb{P}\left(\hat{t}^{(2)}_{\bar{\ell}} \notin I\right) &\leq \min\left\{1,\ 40\log_2(T)\exp\left(-c\,\frac{1}{8}\cdot\frac{1}{144}\cdot\frac{\epsilon^2 T}{3\log_2(T)}\right)\right\} \\ &\leq \log(T)\exp\left(-c'\frac{\epsilon^2 T}{\log(T)}\right) \,,\end{aligned}$$

where c' is a numerical constant. The last line follows by absorbing all numerical constants into $c' > 0$, using $T \geq 4$.

Suppose now that $K \in \{2, 3, 4\}$, then $\lceil K/4 \rceil = \lceil K/8 \rceil = 1$. In this case, Algorithm 15 allocates at least $T/4$ samples to each arm and outputs the minimal empirical mean. Let $a \in [K]$ denote the index corresponding to the arm with the smallest true mean, we therefore have (let $\hat{t}$ denote the output)

$$\begin{aligned}\mathbb{P}\Big(\hat{t} \notin [\mu_{(\lceil K/8 \rceil)} - \epsilon, \mu_{(\lceil K/4 \rceil)} + \epsilon]\Big) &= \mathbb{P}\Big(\hat{t} < \mu_{(1)} - \epsilon\Big) + \mathbb{P}\Big(\hat{t} > \mu_{(1)} + \epsilon\Big) \\ &\leq \mathbb{P}(\hat{\mu}_a > \mu_a + \epsilon) + \sum_{i=1}^{K} \mathbb{P}(\hat{\mu}_i < \mu_i - \epsilon) \,,\end{aligned}$$

where $\hat{\mu}_i$ denotes the empirical mean of arm i. We conclude using Hoeffding's inequality, with the fact that each arm receives at least $T/4$ samples that

$$\mathbb{P}\Big(\hat{t} \notin [\mu_{(\lceil K/8 \rceil)} - \epsilon, \mu_{(\lceil K/4 \rceil)} + \epsilon]\Big) \leq 5\exp\left(-\epsilon^2\frac{T}{2}\right) \,,$$

which corresponds to the result. □

6.A.3 A result on Sequential Halving Algorithm by Zhao et al. (2023)

Consider a K-armed bandit problem with Bernoulli reward with unknown means $\mu_1, \dots, \mu_K$. As in the previous subsection, we write $\mu_{(1)} \leq \dots \leq \mu_{(K)}$ for its ordered values. Sequential halving is a classical elimination scheme for pure-exploration problems (Karnin et al., 2013a). It proceeds

in at most $\lceil \log_2 K \rceil$ phases. Starting from the full set of K candidate arms, each phase spends approximately $\lfloor T/\log_2 K \rfloor$ samples by distributing them uniformly across the surviving arms, then ranks arms by their empirical means and discards the top half. Since our goal is to identify arms with the smallest mean, we retain the bottom-ranked half after each phase. This procedure is known to be adaptive for simple-regret minimization, in the sense formalized by Theorem 6.A.3 below.

Theorem 6.A.3. *[From Zhao et al. (2023), Theorem 6] Consider the Algorithm Bracketed SH with inputs T and K. The output I_T satisfies for any $\epsilon > 0$ and $m \in [K]$:*

$$\mathbb{P}\left(\mu_{I_T} \geq \mu_{(m)} + \epsilon\right) \leq \exp\left(-c\frac{m\epsilon^2\ T}{K\log^3(K)}\right),$$

where c is a positive absolute constant.

6.B Guarantees on FB CWI procedure (Algorithm 16) in the fixed budget setting

In order to structure the proofs, in this section we present guarantees about the output of Algorithm 16 when fed with input (δ, T). More precisely the output being $(\phi_1 \vee \phi_2, I)$, here we provide upper bounds on the probability of misidentification error for the arm candidate I. This corresponds to the typical kind of guarantees encountered in context of best arm identification in the fixed budget framework. In turn, we apply these results in Section 6.C to prove the guarantees presented in Theorem 6.3.1.

6.B.1 First Upper Bound

Theorem 6.B.1. *The output of FB CWI (Algorithm 16) with input T satisfies:*

$$\mathbb{P}\left(\psi_T \neq i^*\right) \leq 27K\log(K)\log(T)\cdot\exp\left(-c\cdot\frac{T}{\log(T)\log(K)H_{cw}}\right),$$

where c is a numerical constant, and we recall that H_{cw} is defined by

$$H_{cw} = \sum_{i\neq i^*}\frac{1}{\Delta_{i^*,i}^2}\ ,$$

if $\Delta_{i^*,i} > 0$ for all $i \in [K]\setminus\{i^*\}$ and $H_{cw} = +\infty$ otherwise.

Notation: Let $\mathbf{\Delta}^{(k)} \in [-1/2, 1/2]^{|A_k|\times|A_k|}$ denote the sub-matrix of $\mathbf{\Delta}$ restricted to rows and columns in A_k. For $\alpha \in A_k$, let $\left(\Delta^{(k)}_{\alpha,(i)}\right)_{i\in\{1,\dots,|A_k|-1\}}$ denote the ordered gaps between α and arms in $A_k \setminus \{\alpha\}$ such that:

$$\Delta^{(k)}_{\alpha,(1)} \leq \cdots \leq \Delta^{(k)}_{\alpha,(|A_k|-1)}\ .$$

Since the gaps sub-matrix for arms in A_k is skew-symmetric (i.e., $\forall i,j : \mathbf{\Delta}^{(k)}_{i,j} = -\mathbf{\Delta}^{(k)}_{j,i}$), the number of arms such that $\Delta^{(k)}_{\alpha,(\lceil * \rceil |A_k|/4)} \leq 0$ is at least $\lceil * \rceil |A_k|/4$ (see Lemma 6.F.6). Let $E_k \subset A_k$ denote the last set of arms:

$$E_k := \left\{ \alpha \in A_k : \Delta^{(k)}_{\alpha,(\lceil * \rceil |A_k|/4)} \leq 0 \right\} .$$

Finally, we remind the reader that for any $j \in [K]$, the quantities $\Delta_{j,(1)} \leq \cdots \leq \Delta_{j,(K-1)}$ correspond to the ordered gaps between j and all arms in $[K] \setminus \{j\}$.

Proof of Theorem 6.B.1. Suppose that $T \geq 8K \log_{8/7}(K)$, otherwise the bound is vacuous. Assume that $\Delta_{i^*,i} > 0$ for all $i \neq i^*$. Otherwise, if $\Delta_{i^*,j} = 0$ for some $j \neq i^*$, then $H_{\mathrm{cw}} = +\infty$ and the stated bound is trivial.

We start by bounding the probability of the event $\psi_T \neq i^*$ by the probabilities that i^* gets eliminated at some step k. Since $i^* \in A_1 = [K]$, the event $\{\psi_T \neq i^*\}$ implies that there exists a round $k \in \{1, \ldots, k_{\max} - 1\}$ such that $i^* \in A_k$ but $i^* \notin A_{k+1}$. Hence,

$$\begin{aligned}
\mathbb{P}(\psi_T \neq i^*) &= \mathbb{P}\Big(\bigcup_{k=1}^{k_{\max}-1} \{ i^* \in A_k,\ i^* \notin A_{k+1} \} \Big) \\
&\leq \sum_{k=1}^{k_{\max}-1} \mathbb{P}(i^* \in A_k,\ i^* \notin A_{k+1}) \\
&\leq k_{\max} \cdot \max_{k \in \{1,\ldots,k_{\max}-1\}} \mathbb{P}(i^* \notin A_{k+1} \mid i^* \in A_k) . \qquad (6.30)
\end{aligned}$$

Recall that $k_{\max} \leq \lceil \log_{8/7}(K) \rceil$.

Next, we upper-bound $\mathbb{P}(i^* \notin A_{k+1} \mid i^* \in A_k)$. Fix $k \in \{1, \ldots, k_{\max} - 1\}$ and condition on $\{i^* \in A_k\}$. If $i^* \notin A_{k+1}$, then by the definition of the next set (keeping only the top fraction), the number of arms in A_k with score smaller than $S_k(i^*)$ is at most $\lceil |A_k|/8 \rceil$. Equivalently, at least $|A_k| - \lceil |A_k|/8 \rceil$ arms in A_k have score at least $S_k(i^*)$.
By Lemma 6.F.7 applied to the skew-symmetric matrix $\mathbf{\Delta}^{(k)}$, we conclude that if $|A_k| \geq 3$ then the intersection between E_k and A_{k+1} (which have a size of $|A_k| - \lceil |A_k|/8 \rceil$) is non-empty, therefore

$$\exists \alpha \in E_k : \quad S_k(\alpha) \geq S_k(i^*).$$

Otherwise, if $|A_k| = 2$ and $i^* \in A_k$, we necessarily have $E_k = A_k \setminus \{i^*\}$. We conclude that

$$\begin{aligned}
\mathbb{P}(i^* \notin A_{k+1} \mid i^* \in A_k) &\leq \mathbb{P}\Big(\exists \alpha \in E_k :\ S_k(\alpha) \geq S_k(i^*) \Big) \\
&\leq \underbrace{\mathbb{P}\left(S_k(i^*) \leq \frac{1}{2} \Delta^{(k)}_{i^*,(\lceil |A_k|/8 \rceil)} \right)}_{\text{Term 1}} + \underbrace{\mathbb{P}\left(\exists \alpha \in E_k :\ S_k(\alpha) \geq \frac{1}{2} \Delta^{(k)}_{i^*,(\lceil |A_k|/8 \rceil)} \right)}_{\text{Term 2}} .
\end{aligned}$$

Denote $\Delta_k :- \Delta^{(k)}_{i^*,(\lceil |A_k|/8 \rceil)}$. By Lemma 6.B.2, Terms 1 and 2 satisfy

$$\begin{aligned}\mathbb{P}(i^* \notin A_{k+1} \mid i^* \in A_k) &\leq (K + 2\log(T)) \exp\left(-c \frac{\Delta_k^2}{\log(\lceil B_k/2 \rceil)} B_k\right) \\ &\quad + K \log(T) \exp\left(-c' \frac{\Delta_k^2}{\log(\lceil B_k/2 \rceil)} B_k\right) \\ &\leq 3K \log(T) \exp\left(-c_1 \frac{\Delta_k^2}{\log(\lceil B_k/2 \rceil)} B_k\right) , \end{aligned} \tag{6.31}$$

where $c_1 = \min\{c, c'\}$.

Next, we develop a bound on $\Delta^{(k)}_{i^*,(\lceil |A_k|/8 \rceil)}$ using $H_{\mathrm{cw}} = \sum_{i \neq i^*} \frac{1}{\Delta^2_{i,i^*}}$. Recall $B_k = \left\lfloor \frac{T}{|A_k| \log_{8/7}(K)} \right\rfloor$. We have

$$\begin{aligned}\Delta_k^2 B_k &= \Delta_k^2 \left\lfloor \frac{T}{|A_k| \log_{8/7}(K)} \right\rfloor \\ &\geq \Delta_k^2 \cdot \frac{T}{2\,|A_k| \log_{8/7}(K)} \\ &\geq \frac{\Delta_k^2}{\lceil |A_k|/8 \rceil} \cdot \frac{T}{16 \log_{8/7}(K)} \\ &\geq \frac{T}{16 \log_{8/7}(K) \cdot \sum_{i \neq i^*} \frac{1}{\Delta^2_{i^*,i}}} = \frac{T}{16 \log_{8/7}(K)\, H_{\mathrm{cw}}} , \end{aligned}$$

where we used in the second line the fact that $T \geq 8K \log_{8/7}(K)$ and Lemma 6.F.2 in the last line (using $\Delta_k :- \Delta^{(k)}_{i^*,(\lceil |A_k|/8 \rceil)}$). Plugging this into (6.31) yields

$$\begin{aligned}\mathbb{P}(i^* \notin A_{k+1} \mid i^* \in A_k) &\leq 3K \log(T) \cdot \exp\left(-c_1 \frac{T}{16 \log_{8/7}(K)\, \log(\lceil B_k/2 \rceil)\, H_{\mathrm{cw}}}\right) \\ &\leq 3K \log(T) \exp\left(-c_1' \frac{T}{\log(K) \log(T)\, H_{\mathrm{cw}}}\right) , \end{aligned} \tag{6.32}$$

for numerical constants $c_1, c_1' > 0$ (using $\log_{8/7}(K) = \Theta(\log K)$ and $\log(\lceil B_k/2 \rceil) \leq \log(T)$).

Finally, we combine the bounds (6.30) and (6.32), and use $k_{\max} \leq \left\lceil \log_{8/7}(K) \right\rceil$, we get

$$\mathbb{P}(\psi_T \neq i^*) \leq \lceil \log_{8/7}(K) \rceil \cdot 3K \log(T) \exp\left(-c_1' \frac{T}{\log(K) \log(T)\, H_{\mathrm{cw}}}\right) ,$$

which yields the claimed form

$$\mathbb{P}(\psi_T \neq i^*) \leq 27K \log(K) \log(T) \exp\left(-c \frac{T}{\log(T) \log(K) H_{\mathrm{cw}}}\right) ,$$

for a numerical constant $c > 0$.

It remains to prove the following technical lemma.

Lemma 6.B.2. *Consider step k in Algorithm 16 and assume that $i^* \in A_k$. Let $\Delta_k :- \Delta^{(k)}_{i^*,(\lceil |A_k|/8 \rceil)}$, then*

$$\mathbb{P}\left(S_k(i^*) \leq \frac{1}{2}\Delta_k\right) \leq (K + 2\log(T)) \exp\left(-c\frac{\Delta_k^2}{\log(\lceil B_k/2 \rceil)}\, B_k\right),$$
$$\mathbb{P}\left(\exists \alpha \in E_k :\ S_k(\alpha) \geq \frac{1}{2}\Delta_k\right) \leq K\log(T) \exp\left(-c'\ \frac{\Delta_k^2}{\log(\lceil B_k/2 \rceil)}\, B_k\right)\ ,$$

for numerical constants $c, c' > 0$.

Proof. Assume $T \geq 8K\log_{8/7}(K)$, this guarantees $B_k = \left\lfloor \frac{T}{|A_k|\log_{8/7}(K)} \right\rfloor \geq 8$. Let c_1 denote the constant from Corollary 6.2.2.

Proof of the first bound: Let i_s^* be the strong opponent chosen for i^* at step k of Algorithm 16. Recall

$$S_k(i^*) = \min\{Z_k^{(s)}(i^*), 0\} + Z_k^{(w)}(i^*)\,.$$

Therefore, we have

$$\mathbb{P}\Big(S_k(i^*) \leq \frac{1}{2}\Delta_k\Big) \leq \mathbb{P}\Big(Z_k^{(s)}(i^*) + Z_k^{(w)}(i^*) \leq \frac{1}{2}\Delta_k\Big) + \mathbb{P}\Big(Z_k^{(w)}(i^*) \leq \frac{1}{2}\Delta_k\Big)\ . \tag{6.33}$$

Recall that the event $\left\{Z_k^{(s)}(i^*) + Z_k^{(w)}(i^*) \leq \frac{1}{2}\Delta_k\right\}$ implies that

$$\left\{Z_k^{(s)}(i^*) \leq -\frac{1}{4}\Delta_k \ \text{ or } \ Z_k^{(w)}(i^*) \leq \frac{3}{4}\Delta_k\right\}\ ,$$

Combining with Inequality 6.33 we obtain

$$\mathbb{P}\left(S_k(i^*) \leq \frac{1}{2}\Delta_k\right) \leq \mathbb{P}\left(Z_k^{(s)}(i^*) \leq -\frac{1}{4}\Delta_k\right) + 2\mathbb{P}\left(Z_k^{(w)}(i^*) \leq \frac{3}{4}\Delta_k\right)\ . \tag{6.34}$$

We use Hoeffding's inequality (Lemma 6.F.10) to bound the first term in the upper bound above. For any fixed $i \in [K] \setminus \{i^*\}$ and $\epsilon > 0$, we have

$$\mathbb{P}\left(\hat{\Delta}_{i^*,i} - \Delta_{i^*,i} \leq -\epsilon\right) \leq \exp\left(-\frac{\epsilon^2 B_k}{2}\right)\ ,$$

where $\hat{\Delta}_{i^*,i}$ is the empirical mean of duels between (i^*, i) computed using $\lceil B_k/4 \rceil$ samples. There-

fore, applying the bound above with $\epsilon = \Delta_k/4$ and a union bound over the arms, we have

$$\begin{aligned}\mathbb{P}\left(Z_k^{(s)}(i^*) \leq -\frac{1}{4}\Delta_k\right) &\leq \mathbb{P}\left(Z_k^{(s)}(i^*) - \Delta_{i^*,i_s^*} \leq -\frac{1}{4}\Delta_k\right) \\ &\leq (K-1)\exp\left(-\frac{\Delta_k^2}{32}B_k\right) . \end{aligned} \tag{6.35}$$

where we used the fact that $\Delta_{i^*,j} \geq 0$ for all $j \in [K]$. Now, using Corollary 6.2.2 which gives a guarantee on the output $Z_k^{(w)}(i^*)$, we have

$$\begin{aligned}\mathbb{P}\left(Z_k^{(w)}(i^*) \leq \frac{3}{4}\Delta_k\right) &= \mathbb{P}\left(Z_k^{(w)}(i^*) \leq \Delta^{(k)}_{i^*,(\lceil |A_k|/8 \rceil)} - \frac{1}{4}\Delta_k\right) \\ &\leq \log\left(\lceil * \rceil \frac{B_k}{2}\right)\exp\left(-c_1\frac{\Delta_k^2}{32\log(\lceil B_k/2 \rceil)}B_k\right) . \end{aligned} \tag{6.36}$$

We conclude by combining (6.35), (6.36) and (6.34) that

$$\mathbb{P}\left(S_k(i^*) \leq \frac{1}{2}\Delta_k\right) \leq (K - 1 + 2\log(T))\exp\left(-c\frac{\Delta_k^2}{\log(\lceil B_k/2 \rceil)}B_k\right) ,$$

where c is a numerical constant.

Proof of the second bound: Fix $\alpha \in E_k$. By definition of E_k, we have

$$\Delta^{(k)}_{\alpha,(\lceil |A_k|/4 \rceil)} \leq 0 .$$

Moreover, $\min\{Z_k^{(s)}(\alpha), 0\} \leq 0$, hence

$$\begin{aligned}\mathbb{P}\left(S_k(\alpha) \geq \frac{1}{2}\Delta_k\right) &= \mathbb{P}\left(\min\{Z_k^{(s)}(\alpha), 0\} + Z_k^{(w)}(\alpha) \geq \frac{1}{2}\Delta_k\right) \\ &\leq \mathbb{P}\left(Z_k^{(w)}(\alpha) \geq \frac{1}{2}\Delta_k\right) \\ &\leq \mathbb{P}\left(Z_k^{(w)}(\alpha) \geq \Delta^{(k)}_{\alpha,(\lceil |A_k|/4 \rceil)} + \frac{1}{2}\Delta_k\right) , \end{aligned}$$

where the last step uses $\Delta^{(k)}_{\alpha,(\lceil |A_k|/4 \rceil)} \leq 0$. Applying Corollary 6.2.2 to $Z_k^{(w)}(\alpha)$ then yields

$$\mathbb{P}\left(S_k(\alpha) \geq \frac{1}{2}\Delta_k\right) \leq \log\left(\left\lceil \frac{B_k}{2} \right\rceil\right)\exp\left(-c_1\,\frac{\Delta_k^2}{8\log(\lceil B_k/2 \rceil)}\,B_k\right) . \tag{6.37}$$

Finally, union bound over $\alpha \in E_k$ gives

$$\mathbb{P}\left(\exists \alpha \in E_k :\ S_k(\alpha) \geq \frac{1}{2}\Delta_k\right) \leq |E_k| \log\left(\left\lceil \frac{B_k}{2} \right\rceil\right)\exp\left(-c'\,\frac{\Delta_k^2}{\log(\lceil B_k/2 \rceil)}B_k\right) ,$$

We may bound $|E_k|$ by K; hence the above yields the stated form

$$\mathbb{P}\left(\exists \alpha \in E_k : \; S_k(\alpha) \geq \frac{1}{2}\Delta_k\right) \leq K \log(T) \exp\left(-c' \; \frac{\Delta_k^2}{\log(\lceil B_k/2 \rceil)} \, B_k\right) \; ,$$

for a numerical constant $c' > 0$. This proves the second inequality and concludes the lemma. □

6.B.2 Second Upper Bound

For each $i \neq i^*$, let $\Delta_{i,(k)}$ denote the ordered gaps $(\Delta_{i,j})_{i \neq j}$ as

$$\Delta_{i,(1)} \leq \cdots \leq \Delta_{i,(K-1)} \; ,$$

Denote by $K_{i;<0}$ the number of j such that $\Delta_{i,j} < 0$. For each $i \in [K]$, let $s_i \leq K_{i;<0}$, and $\boldsymbol{s} = (s_1, \ldots, s_K)$. Here, we take the convention $K_{i^*;<0} = 0$. We recall the expressions of the quantities $H_{\text{certify}}(\boldsymbol{s})$, $H^{(0)}_{\text{explore}}(\boldsymbol{s})$ and $H^{(1)}_{\text{explore}}(\boldsymbol{s})$

$$H_{\text{certify}}(\boldsymbol{s}) = \sum_{i \neq i^*} \frac{1}{\Delta^2_{i,(s_i)}}, \quad H^{(1)}_{\text{explore}}(\boldsymbol{s}) = \max_{i \neq i^*} \frac{K}{s_i \Delta^2_{i,(s_i)}} \;\; \text{and} \;\; H^{(0)}_{\text{explore}}(\boldsymbol{s}) = \sum_{i \neq i^*} \frac{K}{s_i \Delta^2_{i,(s_i)}} \; .$$

Theorem 6.B.3. *For any $\boldsymbol{s}$ such that $1 \leq s_i \leq K_{i;<0}$, it holds that*

$$\mathbb{P}\left(\psi_T \neq i^*\right) \leq 47K \log(K) \log(T) \exp\left(-\frac{c_1}{\log^3(K)\log(T)} \frac{T - c_2 \log^5(H^{(0)}_{explore}(\boldsymbol{s})) \cdot H^{(0)}_{explore}(\boldsymbol{s})}{H^{(1)}_{explore}(\boldsymbol{s}) + H_{certify}(\boldsymbol{s})}\right),$$

where c_1 and c_2 are numerical constants.

We restate and extend the notation introduced in the last section.

Notation: Let $\boldsymbol{\Delta}^{(k)} \in [-1/2, 1/2]^{|A_k| \times |A_k|}$ denote the sub-matrix of $\boldsymbol{\Delta}$ restricted to lines and rows in A_k. For $\alpha \in A_k$, let $\left(\Delta^{(k)}_{\alpha,(i)}\right)_{i \in \{1,\ldots,|A_k|-1\}}$ denote the ordered gaps between α and arms in A_k such that:

$$\Delta^{(k)}_{\alpha,(1)} \leq \cdots \leq \Delta^{(k)}_{\alpha,(|A_k|-1)}.$$

Recall that since the gaps sub-matrix for arms in A_k is skew-symmetric (i.e., $\forall i,j : \boldsymbol{\Delta}^{(k)}_{i,j} = -\boldsymbol{\Delta}^{(k)}_{j,i}$), the number of arms such that $\Delta^{(k)}_{\alpha,(\lceil * \rceil |A_k|/4)} \leq 0$ is at least $\lceil * \rceil \, |A_k| \, /4$ (see Lemma 6.F.6). Let $E_k \subset A_k$ denote the last set of arms

$$E_k := \left\{\alpha \in A_k : \Delta^{(k)}_{\alpha,(\lceil |A_k|/4 \rceil)} \leq 0\right\} \; .$$

We rank the quantities $(\Delta_{\alpha,(s_\alpha)})_{\alpha \in E_k}$. We denote the ranked sequence with ties broken arbitrarily $(\Delta_{E_k:i})_{i \in [|E_k|]}$

$$\Delta_{E_k:1} \leq \cdots \leq \Delta_{E_k:|E_k|} \; .$$

Define the quantity $\bar{\Delta}_k$ by

$$\bar{\Delta}_k := \Delta_{E_k:\lceil * \rceil \frac{7}{8}|E_k|} \leq 0 \ . \tag{6.38}$$

Observe that when $\bar{\Delta}_k = 0$, we necessarily have $\Delta_{i,(s_i)} = 0$ for some $i \in [K] \setminus \{i^*\}$, this implies in particular that $H_{\text{certify}} = \infty$ and the bound becomes loose. Therefore, in the remainder of this proof, we assume that $\bar{\Delta}_k < 0$.

Let F_k denote the subset of arms in E_k such that $\Delta_{\alpha,(s_\alpha)} \leq \bar{\Delta}_k$.

$$F_k := \left\{ \alpha \in E_k : \Delta_{\alpha,(s_\alpha)} \leq \bar{\Delta}_k \right\} \ . \tag{6.39}$$

Finally, we denote for each $i \neq i^* : \ \Gamma_i := s_i \Delta^2_{i,(s_i)}$, and let $(\Gamma_{(i)})_{i \neq i^*}$ correspond to the ranked quantities $\Gamma_{(2)} \leq \cdots \leq \Gamma_{(K)}$, with ties broken arbitrarily.

Proof of Theorem 6.B.3. Fix $\boldsymbol{s}$ such that $1 \leq s_i \leq K_{i;<0}$ for all $i \neq i^*$ (and $K_{i^*;<0} = 0$ by convention). Note that by the assumption of the uniqueness of the Condorcet winner we have $K_{i;<0} \geq 1$ for any $i \neq i^*$. Let $c > 0$ be a numerical constant (chosen smaller than the constants appearing in Corollary 6.2.2 and Theorem 6.A.3). We assume that $T \geq 8K \log_{8/7}(K)$, otherwise the bound of the theorem is vacuous.

Similar to the proof of Theorem 6.B.1, we start by bounding the probability of the event $\psi_T \neq i^*$ by the probabilities that i^* gets eliminated at some step k. We have

$$\mathbb{P}(\psi_T \neq i^*) \leq \sum_{k=1}^{k_{\max}-1} \mathbb{P}(i^* \notin A_{k+1},\ i^* \in A_k) \leq k_{\max} \cdot \max_{k \leq k_{\max}-1} \mathbb{P}(i^* \notin A_{k+1} \mid i^* \in A_k) \, . \tag{6.40}$$

Recall $k_{\max} \leq \lceil \log_{8/7}(K) \rceil$. Fix $k \in \{1, \ldots, k_{\max}-1\}$, we will first consider the case where $|A_k| \geq 3$. The case where $|A_k| = 2$ is simple and is left to the end of this proof.

Next, we build the argument of our proof on the observation that given $i^* \in A_k$, the event $i^* \notin A_{k+1}$ implies in particular that the number of arms $\alpha \in A_k$ with a score $S_k(\alpha)$ larger than $S_k(i^*)$ is at least $|A_k| - \lceil |A_k|/8 \rceil$. Therefore, the event $i^* \notin A_{k+1}$ implies that the number of arms in F_k with a score $S_k(\cdot)$ larger than $S_k(i^*)$ is at least $|A_{k+1} \cap F_k| \geq \lceil |F_k|/3 \rceil$ as stated in the following lemma

Lemma 6.B.4. *Let $k \in \{1, \ldots, k_{\max} - 1\}$, recall the definition of F_k given in* (6.39)*. We have, if $|A_k| \geq 3$ then*

$$|A_{k+1} \cap F_k| \geq \lceil * \rceil \, \frac{1}{3} \, |F_k| \ .$$

This lemma implies that if i^* is eliminated at step k (i.e., $i^* \notin A_{k+1}$), then many "bad" arms in F_k beat i^*. More precisely, since A_{k+1} consists of the top-scoring arms at step k, every $\alpha \in A_{k+1}$ satisfies $S_k(\alpha) \geq S_k(i^*)$ whenever $i^* \notin A_{k+1}$. Therefore,

$$\{i^* \notin A_{k+1}\} \ \subseteq \ \left\{ \left| \{\alpha \in F_k : \ S_k(\alpha) \geq S_k(i^*)\} \right| \ \geq \ \left\lceil \frac{1}{3} \, |F_k| \right\rceil \right\} .$$

Introduce the threshold $\frac{1}{2}\bar{\Delta}_k$ defined by (6.38) and split

$$\begin{aligned}\mathbb{P}(i^* \notin A_{k+1} \mid i^* \in A_k) &\leq \mathbb{P}\left(\left|\{\alpha \in F_k : \ S_k(\alpha) \geq S_k(i^*)\}\right| \geq \left\lceil \frac{1}{3}\left|F_k\right| \right\rceil\right) \\ &\leq \underbrace{\mathbb{P}\left(S_k(i^*) \leq \frac{1}{2}\bar{\Delta}_k\right)}_{\text{Term 1}} + \underbrace{\mathbb{P}\left(\left|\{\alpha \in F_k : \ S_k(\alpha) \geq \frac{1}{2}\bar{\Delta}_k\}\right| \geq \left\lceil \frac{1}{3}\left|F_k\right| \right\rceil\right)}_{\text{Term 2}}.\end{aligned}$$

The following lemma is a key step in the proof, we postponed its proof to the next subsection.

Lemma 6.B.5. *Under the assumptions of Theorem 6.B.3, consider step k in Algorithm 16. Then, we have*

$$\mathbb{P}\left(S_k(i^*) \leq \frac{1}{2}\bar{\Delta}_k \mid i^* \in A_k\right) \leq (K + \log(T)) \exp\left(-c\frac{\bar{\Delta}_k^2}{\log(B_k)} B_k\right)$$

$$\mathbb{P}\left(\left|\{\alpha \in F_k : \frac{1}{2}\bar{\Delta}_k \leq S_k(\alpha)\}\right| \geq \lceil * \rceil \frac{1}{3}\left|F_k\right|\right) \leq \exp\left(-\frac{c}{\log^3(K)\log(T)} \cdot \frac{T - c' \log^5(H^{(0)}_{explore}(\boldsymbol{s})) \cdot H^{(0)}_{explore}(\boldsymbol{s})}{H^{(1)}_{explore}(\boldsymbol{s})}\right),$$

where c and c' are positives numerical constants.

A direct application of the lemma above gives (for numerical constants $c, c' > 0$)

$$\begin{aligned}\mathbb{P}(i^* \notin A_{k+1} \mid i^* \in A_k) &\leq (K + \log(T)) \exp\left(-c\,\frac{\bar{\Delta}_k^2}{\log(B_k)}\, B_k\right) \\ &\quad + \exp\left(-\frac{c}{\log^3(K)\log(T)} \cdot \frac{T - c' \log^5(H^{(0)}_{\text{explore}}(\boldsymbol{s}))\, H^{(0)}_{\text{explore}}(\boldsymbol{s})}{H^{(1)}_{\text{explore}}(\boldsymbol{s})}\right). \quad (6.41)\end{aligned}$$

Next, we will convert the dependence of the bound on $\bar{\Delta}_k$ into $H_{\text{certify}}(\boldsymbol{s})$.
Recall $|E_k| \geq \lceil |A_k|/4 \rceil$. Since $\bar{\Delta}_k = \Delta_{E_k:\lceil \frac{7}{8}|E_k| \rceil}$, and the sequence $(\Delta_{E_k:i})_i$ is non-decreasing and non-positive, the squared sequence $(\Delta^2_{E_k:i})_i$ is non-increasing. Applying Lemma 6.F.2 to $(\Delta^2_{E_k:i})_{i \in [|E_k|]}$ yields

$$|A_k| \cdot \frac{1}{\bar{\Delta}_k^2} \leq 4\,|E_k| \cdot \frac{1}{\bar{\Delta}_k^2} \leq 32 \cdot \left\lceil \frac{|E_k|}{8} \right\rceil \cdot \frac{1}{\Delta^2_{E_k:\lceil \frac{7}{8}|E_k| \rceil}} \leq 32 \sum_{\alpha \in E_k} \frac{1}{\Delta^2_{\alpha,(s_\alpha)}} \leq 32 H_{\text{certify}}(\boldsymbol{s}). \quad (6.42)$$

Using $B_k = \left\lfloor \frac{T}{|A_k| \log_{8/7}(K)} \right\rfloor \geq \frac{T}{2|A_k| \log_{8/7}(K)}$ (and $\log(B_k) \leq \log(T)$), we obtain

$$(K + \log(T)) \exp\left(-c\,\frac{\bar{\Delta}_k^2}{\log(B_k)}\, B_k\right) \leq 2(K + \log(T)) \exp\left(-c'\,\frac{T}{H_{\text{certify}}(\boldsymbol{s})\ \log(T)\ \log(K)}\right), \quad (6.43)$$

for a numerical constant $c' > 0$ (absorbing $\log_{8/7}(K) = \Theta(\log K)$ into constants).

Next combine (6.41) and (6.43), and using $\exp(-a)+\exp(-b) \le 2\exp(-\min\{a,b\})$, we get that if $|A_k| \ge 3$, we have

$$\mathbb{P}(i^* \notin A_{k+1} \mid i^* \in A_k) \le 3(K+\log(T)) \exp\left(-\frac{c}{\log^3(K)\log(T)} \cdot \frac{T - c_2 \log^5(H^{(0)}_{\mathrm{explore}}(\boldsymbol{s}))\, H^{(0)}_{\mathrm{explore}}(\boldsymbol{s})}{H^{(1)}_{\mathrm{explore}}(\boldsymbol{s}) + H_{\mathrm{certify}}(\boldsymbol{s})} \right),$$

for a numerical constant $c_2 > 0$ (renaming constants). To conclude we need to consider the edge case where $|A_k| = 2$ (last iteration). In this case we have $|E_k| = |F_k| =: \{\alpha\}$. Therefore,

$$\begin{aligned}\mathbb{P}(i^* \notin A_{k+1} \mid i^* \in A_k) &\le \mathbb{P}(S_k(\alpha) \ge S_k(i^*)) \\ &\le \mathbb{P}\left(S_k(i^*) \le \frac{1}{2}\bar{\Delta}_k\right) + \mathbb{P}\left(S_k(\alpha) \ge \frac{1}{2}\bar{\Delta}_k\right).\end{aligned}$$

The first term in the upper bound can be bounded using (6.B.5), the second term can be bounded using Lemma 6.B.6. The resulting bound is smaller than the one obtained when $|A_k| \ge 3$.

Finally, using (6.40) and $k_{\max} \le \lceil \log_{8/7}(K) \rceil$, and absorbing $\lceil \log_{8/7}(K) \rceil$ and additive logarithms into the prefactor, we obtain

$$\mathbb{P}(\psi_T \ne i^*) \le 47K\log(K)\log(T) \exp\left(-\frac{c_1}{\log^3(K)\log(T)} \cdot \frac{T - c_2 \log^5(H^{(0)}_{\mathrm{explore}}(\boldsymbol{s}))\, H^{(0)}_{\mathrm{explore}}(\boldsymbol{s})}{H^{(1)}_{\mathrm{explore}}(\boldsymbol{s}) + H_{\mathrm{certify}}(\boldsymbol{s})} \right),$$

which is the claim of Theorem 6.B.3.

6.B.3 Proofs of Technical Lemmas

6.B.3.1 Proof of Lemma 6.B.4

Proof. Recall $\bar{\Delta}_k = \Delta_{E_k:\lceil \frac{7}{8}|E_k| \rceil}$ and $F_k = \{\alpha \in E_k : \Delta_{\alpha,(s_\alpha)} \le \bar{\Delta}_k\}$, hence

$$|F_k| \;\ge\; \left\lceil \frac{7}{8}\,|E_k| \right\rceil. \tag{6.44}$$

Algorithm 16 keeps $|A_{k+1}| = |A_k| - \lceil * \rceil\, |A_k|/8$ arms, so for any $A_{k+1} \subseteq A_k$ and $F_k \subseteq A_k$,

$$|A_{k+1} \cap F_k| \;\ge |A_{k+1}| + |F_k| - |A_k| = |F_k| - \lceil * \rceil\, \frac{|A_k|}{8}\,. \tag{6.45}$$

Case 1: $|A_k| \ge 5$**.** By Lemma 6.F.6, $|E_k| \ge \lceil |A_k|/4 \rceil \ge 2$, hence $\lceil |A_k|/8 \rceil \le \lceil |E_k|/2 \rceil$. Moreover, for every integer $m \ge 2$,

$$2\left\lceil \frac{7m}{8} \right\rceil \ge 3\left\lceil \frac{m}{2} \right\rceil, \tag{6.46}$$

(which follows by writing $m = 8q + r$ and checking $r \in \{0, \dots, 7\}$; the only delicate residue $r = 1$ is harmless since then $q \ge 1$). Applying (6.46) with $m = |E_k|$ and using (6.44) gives

$\lfloor \frac{2}{3} |F_k| \rfloor \geq \lceil |E_k|/2 \rceil \geq \lceil |A_k|/8 \rceil$. Plugging into (6.45) yields

$$|A_{k+1} \cap F_k| \geq |F_k| - \left\lfloor \frac{2}{3} |F_k| \right\rfloor = \left\lceil \frac{1}{3} |F_k| \right\rceil .$$

Case 2: $|A_k| \in \{3, 4\}$**.** Here $\lceil |A_k|/8 \rceil = 1$. By skew-symmetry of $\Delta^{(k)}$, at most one row can have all off-diagonal entries > 0, hence at least $|A_k| - 1$ rows have $\Delta^{(k)}_{\alpha,(1)} \leq 0$, so $|E_k| \geq |A_k| - 1 \in \{2, 3\}$. Then $\lceil \frac{7}{8} |E_k| \rceil = |E_k|$, and since $F_k \subseteq E_k$, (6.44) implies $|F_k| = |E_k| \geq |A_k| - 1$. Using (6.45),

$$|A_{k+1} \cap F_k| \ \geq \ |F_k| - 1 ,$$

and for $|F_k| \in \{2, 3\}$ this satisfies $|F_k| - 1 \geq \lceil |F_k|/3 \rceil$.

Combining both cases proves that for every $|A_k| \geq 3$,

$$|A_{k+1} \cap F_k| \ \geq \ \left\lceil \frac{1}{3} |F_k| \right\rceil .$$

□

6.B.3.2 Proof of Lemma 6.B.5

Proof. Fix a round $k \in \{1, \ldots, k_{\max} - 1\}$ and recall $\bar{\Delta}_k \leq 0$ by construction.

Proof of the first bound

$$\mathbb{P}\left(S_k(i^*) \leq \frac{1}{2} \bar{\Delta}_k \right) \leq (K + \log(T)) \exp\left(-c \frac{\bar{\Delta}_k^2}{\log(B_k)} B_k \right) .$$

If $\bar{\Delta}_k = 0$ the bound is immediate. Assume $\bar{\Delta}_k < 0$. Recall $S_k(i^*) = \min\{Z_k^{(s)}(i^*), 0\} + Z_k^{(w)}(i^*)$. Then

$$\begin{aligned} \mathbb{P}\left(S_k(i^*) \leq \frac{1}{2} \bar{\Delta}_k \right) &\leq \mathbb{P}\left(Z_k^{(s)}(i^*) + Z_k^{(w)}(i^*) \leq \frac{1}{2} \bar{\Delta}_k \right) + \mathbb{P}\left(Z_k^{(w)}(i^*) \leq \frac{1}{2} \bar{\Delta}_k \right) \\ &\leq \mathbb{P}\left(Z_k^{(s)}(i^*) \leq \frac{1}{4} \bar{\Delta}_k \right) + 2\,\mathbb{P}\left(Z_k^{(w)}(i^*) \leq \frac{1}{4} \bar{\Delta}_k \right) , \end{aligned}$$

where the last step uses $\bar{\Delta}_k < 0$, hence $\{Z_k^{(w)}(i^*) \leq \frac{1}{2} \bar{\Delta}_k\} \subseteq \{Z_k^{(w)}(i^*) \leq \frac{1}{4} \bar{\Delta}_k\}$.

The first term in the last upper-bound is bounded using Hoeffding and a union bound over the opponent choice,

$$\mathbb{P}\left(Z_k^{(s)}(i^*) \leq \frac{1}{4} \bar{\Delta}_k \right) \leq (K - 1) \exp\left(-\frac{\bar{\Delta}_k^2}{32} B_k \right) .$$

The second term is bounded by Corollary 6.2.2,

$$\mathbb{P}\Big(Z_k^{(w)}(i^*) \le \frac{1}{4}\bar{\Delta}_k\Big) \le \mathbb{P}\Big(Z_k^{(w)}(i^*) \le \Delta^{(k)}_{i^*,(\lceil |A_k|/8\rceil)} + \frac{1}{4}\bar{\Delta}_k\Big)$$
$$\le \log(T)\exp\left(-c\,\frac{\bar{\Delta}_k^2}{\log(\lceil B_k/2\rceil)}\,B_k\right).$$

Finally, absorbing $\log(\lceil B_k/2\rceil)$ into $\log(B_k)$ and constants yields

$$\mathbb{P}\Big(S_k(i^*) \le \frac{1}{2}\bar{\Delta}_k\Big) \le (K+\log(T))\exp\left(-c\,\frac{\bar{\Delta}_k^2}{\log(B_k)}\,B_k\right).$$

Proof of the second bound

$$\mathbb{P}\Big(|\{\alpha\in F_k:\ S_k(\alpha)\ge \frac{1}{2}\bar{\Delta}_k\}| \ge \lceil |F_k|/3\rceil\Big) \le \exp\left(-\frac{c}{\log^3(K)\log(T)}\cdot\frac{T - c'\log^5(H^{(0)}_{\mathrm{explore}}(\boldsymbol{s}))\cdot H^{(0)}_{\mathrm{explore}}(\boldsymbol{s})}{H^{(1)}_{\mathrm{explore}}(\boldsymbol{s})}\right).$$

We start from the indicator-sum form

$$\mathbb{P}\Big(|\{\alpha\in F_k:\ S_k(\alpha)\ge \tfrac{1}{2}\bar{\Delta}_k\}| \ge \left\lceil\frac{|F_k|}{3}\right\rceil\Big) = \mathbb{P}\left(\sum_{\alpha\in F_k}\mathbb{1}\Big(S_k(\alpha)\ge\tfrac{1}{2}\bar{\Delta}_k\Big) \ge \left\lceil\frac{|F_k|}{3}\right\rceil\right). \tag{6.47}$$

Next, we keep only the hardest 3/4 of F_k. More formally, we rank $\Gamma_\alpha = s_\alpha\Delta^2_{\alpha,(s_\alpha)}$ over $\alpha\in F_k$ as $\Gamma_{F_k:1}\le\cdots\le\Gamma_{F_k:|F_k|}$, and let $F_k^{(3/4)}$ be the subset containing the top $\lceil 3\,|F_k|/4\rceil$ arms with largest Γ_α. Then $\left|F_k\setminus F_k^{(3/4)}\right| = \lfloor |F_k|/4\rfloor$, so

$$\left\{\sum_{\alpha\in F_k}\mathbb{1}(S_k(\alpha)\ge\tfrac{1}{2}\bar{\Delta}_k) \ge \left\lceil\frac{|F_k|}{3}\right\rceil\right\} \subseteq \left\{\sum_{\alpha\in F_k^{(3/4)}}\mathbb{1}(S_k(\alpha)\ge\tfrac{1}{2}\bar{\Delta}_k) \ge \left\lceil\frac{|F_k|}{12}\right\rceil\right\}.$$

Let us develop a uniform per-arm bound on $F_k^{(3/4)}$. Lemma 6.B.6 below gives such a bound

Lemma 6.B.6. *Let $\alpha\in F_k^{(3/4)}$. We have*

$$\mathbb{P}\left(S_k(\alpha)\ge\frac{1}{2}\bar{\Delta}_k\right) \le (\log(T)+K)\exp\left(-c"\,\frac{T}{\log^3(K)\log(T)\ \sum_{i\in E_k}\frac{K}{s_i\Delta^2_{i,(s_i)}}}\right), \tag{6.48}$$

where c" is a positive numerical constant. Moreover, if $T\ge c'\,H^{(0)}_{\mathrm{explore}}(\boldsymbol{s})\,\log^5(H^{(0)}_{\mathrm{explore}}(\boldsymbol{s}))$, where $c' := 10^3\vee\frac{960}{c"}\log^2(\frac{960}{c"})$, we have

$$\mathbb{P}\left(S_k(\alpha)\ge\frac{1}{2}\bar{\Delta}_k\right) \le \frac{1}{18}.$$

In the remainder of this proof we assume that the condition $T \geq c' \, H^{(0)}_{\text{explore}}(\boldsymbol{s}) \, \log^5\big(H^{(0)}_{\text{explore}}(\boldsymbol{s})\big)$ is satisfied, otherwise the upper bound stated by the theorem is greater than 1 and is thus vacuous. Denote by p_k the bound given by the lemma above

$$p_k :- \frac{1}{18} \wedge \ (\log(T) + K) \exp\left(-c" \, \frac{T}{\log^3(K) \log(T) \, \sum_{i \in E_k} \frac{K}{s_i \Delta^2_{i,(s_i)}}} \right) \, .$$

We use Lemma 6.F.9, which is purely technical and deferred to Section 6.F, to obtain the following upper bound

$$p_k \leq \exp\left(-\bar{c}_1 \cdot \frac{T}{\log^3(K) \log(T) \, \sum_{i \in E_k} \frac{K}{s_i \Delta^2_{i,(s_i)}}} \right) \, , \tag{6.49}$$

where $\bar{c}_1$ is a numerical constant depending only on $c"$. Therefore, by independence across arms in the construction of $S_k(\cdot)$ since the algorithm uses independent fresh samples per arm,

$$\sum_{\alpha \in F_k^{(3/4)}} \mathbb{1}\Big(S_k(\alpha) \geq \tfrac{1}{2} \bar{\Delta}_k \Big) \quad \text{is stochastically dominated by} \quad M_k \sim \text{Bin}\Big(\Big\lceil \frac{3}{4} \, |F_k| \Big\rceil \, , \, p_k \Big) \, .$$

Consequently, using the fact that $p_k \leq \frac{1}{18}$ implies $\left\lceil \frac{|F_k|}{12} \right\rceil - p_k \left\lceil \frac{3}{4} \, |F_k| \right\rceil \geq \frac{|F_k|}{24}$ we have

$$\begin{aligned} \mathbb{P}\left(\sum_{\alpha \in F_k^{(3/4)}} \mathbb{1}\Big(S_k(\alpha) \geq \tfrac{1}{2} \bar{\Delta}_k \Big) \geq \left\lceil \frac{|F_k|}{12} \right\rceil \right) &\leq \mathbb{P}\left(M_k \geq \left\lceil \frac{|F_k|}{12} \right\rceil \right) \\ &\leq \mathbb{P}\left(M_k - \mathbb{E}[M_k] \geq \frac{|F_k|}{24} \right) \, . \end{aligned} \tag{6.50}$$

Next, we use Lemma 6.F.12 which provides a deviation bound for binomial variables in regimes where the parameters can be small. Recall that M_k is a binomial distribution with parameters $(p_k, \lceil 3 \, |F_k| \, / 4 \rceil)$. We have

$$\mathbb{P}\left(M_k - \mathbb{E}[M_k] \geq \frac{|F_k|}{24} \right) \leq \exp\left(- \frac{|F_k|}{864 \, \phi(p_k)} \right) \, , \tag{6.51}$$

where ϕ is the function defined in Lemma 6.F.1. Since we have proved that $p_k \leq \frac{1}{18}$, the expression of $\phi(p_k)$ is therefore given by

$$\phi(p_k) = \frac{\frac{1}{2} - p_k}{\log(1 - p_k) - \log(p_k)} \, .$$

Since the function ϕ is increasing on $(0, 1/2)$ and, since by Lemma 6.F.5 we have, for any $y > 0$,

$$0 < \frac{\frac{1}{2} - \exp(-y)}{\log(1 - \exp(-y)) - \log(\exp(-y))} \leq \frac{1}{2y} \, ,$$

we conclude using the bound (6.49) that

$$\frac{1}{\phi(p_k)} \geq 2\frac{\bar{c}_1}{320\log(T)\log^3(K)} \cdot \frac{T}{\sum_{i\in E_k} \frac{K}{s_i\Delta^2_{i,(s_i)}}} \ .$$

Using the bound above with (6.51) and $|F_k| \geq \frac{7}{8}|E_k|$, we have

$$\begin{aligned}\mathbb{P}\left(M_k - \mathbb{E}[M_k] \geq \frac{|F_k|}{24}\right) &\leq \exp\left(-\frac{|F_k|}{864} \cdot \frac{\bar{c}_1}{320\log(T)\log^3(K)} \cdot \frac{T}{\sum_{i\in E_k} \frac{K}{s_i\Delta^2_{i,(s_i)}}}\right) \\ &\leq \exp\left(-\bar{c}_2 \cdot \frac{T\,|E_k|}{\log^3(K)\log(T)\sum_{i\in E_k} \frac{K}{s_i\Delta^2_{i,(s_i)}}}\right) \ ,\end{aligned}$$

where $\bar{c}_2$ is a numerical constant. We then use the fact that

$$\frac{|E_k|}{\sum_{i\in E_k} \frac{K}{s_i\Delta^2_{i,(s_i)}}} \ \geq \ \frac{1}{\max_{i\neq i^*} \frac{K}{s_i\Delta^2_{i,(s_i)}}} \ = \ \frac{1}{H^{(1)}_{\text{explore}}(\boldsymbol{s})} \ .$$

Plugging these two relations into (6.50) then (6.47) yields for a numerical constant $\bar{c}_3$

$$\mathbb{P}\Big(|\{\alpha \in F_k : \ S_k(\alpha) \geq \tfrac{1}{2}\bar{\Delta}_k\}| \geq \left\lceil \frac{|F_k|}{3} \right\rceil\Big) \leq \exp\left(-\bar{c}_3 \cdot \frac{T}{\log^3(K)\log(T)\,H^{(1)}_{\text{explore}}(\boldsymbol{s})}\right),$$

as soon as $T \geq c' H^{(0)}_{\text{explore}}(\boldsymbol{s})\log^5(H^{(0)}_{\text{explore}}(\boldsymbol{s}))$. Reintroducing the shift (to cover smaller T) gives the stated bound in Lemma 6.B.5. □

Proof. **of Lemma 6.B.6**. Fix $\alpha \in F_k^{(3/4)}$, let $\alpha^{(s)}$ denote the strong opponent chosen for α. We have

$$\begin{aligned}\mathbb{P}\left(S_k(\alpha) \geq \frac{1}{2}\bar{\Delta}_k\right) &= \mathbb{P}\left(\min\{Z_k^{(s)}(\alpha),\ 0\} + Z_k^{(w)}(\alpha) \geq \frac{1}{2}\bar{\Delta}_k\right) \\ &\leq \mathbb{P}\left(Z_k^{(s)}(\alpha) + Z_k^{(w)}(\alpha) \geq \frac{1}{2}\bar{\Delta}_k\right) \\ &\leq \mathbb{P}\left(Z_k^{(s)}(\alpha) - \Delta_{\alpha,\alpha^{(s)}} \geq -\frac{1}{4}\bar{\Delta}_k\right) \\ &\quad + \mathbb{P}\left(\Delta_{\alpha,\alpha^{(s)}} \geq \frac{7}{8}\bar{\Delta}_k\right) + \mathbb{P}\left(Z_k^{(w)}(\alpha) \geq -\frac{1}{8}\bar{\Delta}_k\right) \ . \qquad (6.52)\end{aligned}$$

Using Hoeffding's concentration inequality with a union bound over the possible choices of $\alpha^{(s)}$, we have

$$\mathbb{P}\left(Z_k^{(s)}(\alpha) - \Delta_{\alpha,\alpha^{(s)}} \geq -\frac{1}{4}\bar{\Delta}_k\right) \leq (K-1)\exp\left(-\frac{\bar{\Delta}_k^2}{32}B_k\right) \ . \qquad (6.53)$$

Since $\alpha \in F_k \subset E_k$, we have $\Delta^{(k)}_{\alpha,(\lceil |A_k|/4 \rceil)} \leq 0$. Therefore, by Corollary 6.2.2, we get

$$\begin{aligned}
\mathbb{P}\left(Z_k^{(w)}(\alpha) \geq -\frac{1}{8}\bar{\Delta}_k\right) &\leq \mathbb{P}\left(Z_k^{(w)}(\alpha) \geq \Delta^{(k)}_{\alpha,(\lceil |A_k|/4 \rceil)} - \frac{1}{8}\bar{\Delta}_k\right) \\
&\leq \log(T) \exp\left(-c \cdot \frac{\bar{\Delta}_k^2}{64 \log(\lceil B_k/2 \rceil)} \lceil * \rceil \frac{B_k}{2}\right) \\
&\leq \log(T) \exp\left(-c \cdot \frac{\bar{\Delta}_k^2}{128 \log(B_k)} B_k\right) \ . \qquad (6.54)
\end{aligned}$$

Since $\alpha \in F_k$, by definition of F_k given in (6.39), we have $\Delta_{\alpha,(s_\alpha)} \leq \bar{\Delta}_k \leq 0$. Therefore,

$$\mathbb{P}\left(\Delta_{\alpha,\alpha^{(s)}} \geq \frac{7}{8}\bar{\Delta}_k\right) \leq \mathbb{P}\left(\Delta_{\alpha,\alpha^{(s)}} \geq \frac{7}{8}\Delta_{\alpha,(s_\alpha)}\right) \ .$$

Then using Theorem 6.A.3 we have

$$\begin{aligned}
\mathbb{P}\left(\Delta_{\alpha,\alpha^{(s)}} \geq \frac{7}{8}\bar{\Delta}_k\right) &\leq \mathbb{P}\left(\Delta_{\alpha,\alpha^{(s)}} \geq \frac{7}{8}\Delta_{\alpha,(s_\alpha)}\right) \\
&= \mathbb{P}\left(\Delta_{\alpha,\alpha^{(s)}} \geq \Delta_{\alpha,(s_\alpha)} - \frac{1}{8}\Delta_{\alpha,(s_\alpha)}\right) \\
&\leq \exp\left(-c \cdot \frac{s_\alpha \Delta^2_{\alpha,(s_\alpha)}}{64K \log^3(K)} \lceil * \rceil \frac{B_k}{4}\right) \\
&\leq \exp\left(-\frac{c}{256} \cdot \frac{s_\alpha \Delta^2_{\alpha,(s_\alpha)}}{K \log^3(K)} B_k\right) \ . \qquad (6.55)
\end{aligned}$$

We conclude by plugging the bounds (6.53), (6.54) and (6.55) into (6.52)

$$\mathbb{P}\left(S_k(\alpha) \geq \frac{1}{2}\bar{\Delta}_k\right) \leq (K + \log(T)) \exp\left(-c' \ \min\left\{\frac{\Gamma_\alpha}{K \log^3(K)}, \frac{\bar{\Delta}_k^2}{\log(B_k)}\right\} B_k\right) \ ,$$

where $c' := \frac{c}{1024}$. Now it remains to prove that

$$\min\left\{\frac{\Gamma_{F_k:\lceil |F_k|/4 \rceil}}{K \log^3(K)}, \ \frac{\bar{\Delta}_k^2}{\log(B_k)}\right\} B_k \ \geq c' \frac{T}{\log^3(K) \log(T) \ \sum_{i \in E_k} \frac{K}{\Gamma_i}} \ .$$

Recall Lemma 6.F.2 gives

$$\frac{\lceil * \rceil \frac{|F_k|}{4}}{\Gamma_{F_k:\lceil * \rceil \frac{1}{4}|F_k|}} \leq \sum_{i \in F_k} \frac{1}{\Gamma_i} \leq \sum_{i \in E_k} \frac{1}{\Gamma_i} \ .$$

Therefore,

$$\frac{|F_k|}{4 \sum_{i \in E_k} \frac{1}{\Gamma_i}} \leq \Gamma_{F_k:\lceil * \rceil \frac{1}{4}|F_k|} \ .$$

Hence, using the bound above and the definition of B_k we obtain

$$\begin{aligned}\frac{\Gamma_{F_k:\lceil * \rceil \frac{1}{4}|F_k|}}{K\log^3(K)}B_k &\geq \frac{|F_k|}{4\sum_{i\in E_k}\frac{1}{\Gamma_i}}\cdot\frac{1}{K\log^3(K)}\cdot\frac{T}{2\,|A_k|\log_{8/7}(K)}\\ &= \frac{T}{8\log^3(K)\log_{8/7}(K)}\cdot\frac{1}{\sum_{i\in E_k}\frac{K}{\Gamma_i}}\cdot\frac{|F_k|}{|A_k|}\\ &\geq \frac{T}{138\log^4(K)\sum_{i\in E_k}\frac{K}{\Gamma_i}}\cdot\frac{|F_k|}{|A_k|}\,,\end{aligned}$$

Recall that $|F_k| \geq \lceil * \rceil \frac{7}{8}|E_k| \geq \lceil * \rceil \frac{3}{16}|A_k|$. Therefore, the bound above gives

$$\frac{\Gamma_{F_k:\lceil * \rceil \frac{1}{4}|F_k|}}{K\log^3(K)}B_k \geq \frac{T}{736\log^4(K)\sum_{i\in E_k}\frac{K}{\Gamma_i}}\,. \tag{6.56}$$

Moreover, we have

$$\begin{aligned}|A_k|\,\frac{1}{\bar{\Delta}_k^2} \leq 4\,|E_k|\cdot\frac{1}{\bar{\Delta}_k^2} &\leq 32\cdot\lceil * \rceil\,\frac{1}{8}\,|E_k|\frac{1}{\Delta^2_{E_k:\lceil * \rceil\frac{7}{8}|E_k|}}\\ &\leq 32\cdot\sum_{\alpha\in E_k}\frac{1}{\Delta^2_{\alpha,(s_\alpha)}}\,,\end{aligned} \tag{6.57}$$

where we used again Lemma 6.F.2 in the second line. Therefore, we have

$$\begin{aligned}\frac{\bar{\Delta}_k^2}{\log(B_k)}B_k &\geq \frac{\bar{\Delta}_k^2}{\log(B_k)}\frac{T}{2\,|A_k|\log_{8/7}(K)}\\ &\geq \frac{1}{\sum_{\alpha\in E_k}\frac{1}{\Delta^2_{\alpha,(s_\alpha)}}}\cdot\frac{T}{64\log(B_k)\log_{8/7}(K)}\,.\end{aligned} \tag{6.58}$$

Therefore, combining (6.56) and (6.58), we get,

$$\begin{aligned}\mathbb{P}\left(S_k(\alpha)\geq\frac{1}{2}\bar{\Delta}_k\right) &\leqslant l\exp\left(-c'\ \min\left\{\frac{\Gamma_{F_k:\lceil * \rceil\frac{1}{4}|F_k|}}{K\log^3(K)},\ \frac{\bar{\Delta}_k^2}{\log(B_k)}\right\}B_k\right)\\ &\leq l\exp\left(-\frac{c'}{736}\min\left\{\frac{1}{\log^3(K)\sum_{i\in E_k}\frac{K}{\Gamma_i}},\ \frac{1}{\sum_{\alpha\in E_k}\frac{1}{\Delta^2_{\alpha,(s_\alpha)}}\log(T)}\right\}\frac{T}{\log(K)}\right)\\ &\leq l\exp\left(-\frac{c'}{736}\ \frac{T}{\log^3(K)\log(T)\sum_{i\in E_k}\frac{K}{\Gamma_i}}\right)\,,\end{aligned} \tag{6.59}$$

where we used $\log(T)\geq\log(K)$ in the last line, and the pre-factor is $l=(\log(T)+K)$.

For the remainder of this proof, we denote $H :- H^{(0)}_{\text{explore}}(\boldsymbol{s})$. Let us prove the last claim.

Let $c" := 10^3 \vee \frac{960}{c'}\log^2(\frac{960}{c'})$, which implies that $c' \geq 960\frac{\log^2(c")}{c"}$. The function $T \mapsto (\log(T) + K)\exp\left(-\frac{c'}{736}\frac{T}{H\log^3(K)\log(T)}\right)$ is non-increasing on the interval $[c" \cdot H\log^5(H), +\infty)$. Therefore, we have using (6.59)

$$\begin{aligned}
\mathbb{P}\left(S_k(\alpha) \geq \frac{1}{2}\bar{\Delta}_k\right) &\leq (\log(c"H\log^5(H)) + K)\exp\left(-\frac{c'}{736}\cdot\frac{c"H\log^5(H)}{H\log^3(K)\log(c"H\log^5(H))}\right) \\
&\leq (\log(c"H\log^5(H)) + H)\exp\left(-\frac{c'}{736}\cdot\frac{c"\log^2(H)}{\log(c"H\log^5(H))}\right) \\
&\leq (\log(c"H\log^5(H)) + H)\exp\left(-4\log^2(c")\cdot\frac{\log^2(H)}{\log(c"H\log^5(H))}\right) . \qquad (6.60)
\end{aligned}$$

where we used in the second line the facts that $H \geq K\min_{i,j}\Delta_{i,j}^{-2} \geq 4K$ (since $|\Delta_{i,j}| \leq \frac{1}{2}$) and that $c' \geq 960\frac{\log^2(c")}{c"}$ by definition of $c"$. Next, we show that

$$2\frac{\log^2(c")\log^2(H)}{\log(c"H\log^5(H))} \geq \log(c"H) ,$$

this bound is derived just by studying the variations of a function and using $c" \geq 10^3$ by definition and $H \geq 4K \geq 8$, the proof is deferred to Lemma 6.F.8 in Section 6.F. Combining the bound above with (6.60), we obtain

$$\begin{aligned}
\mathbb{P}\left(S_k(\alpha) \geq \frac{1}{2}\bar{\Delta}_k\right) &\leq (\log(c"H\log^5(H)) + H)\exp\left(-2\log(c"H)\right) \\
&\leq \frac{\log(c"H\log^5(H)) + H}{(c"H)^2} \leq \frac{1}{18} ,
\end{aligned}$$

where we used $100\log(c"H\log^5(H)) \leq c"H^2$ and $36H \leq (c"H)^2$, given that $H \geq 8$ and $c" \geq 10^3$. □

6.C Proof of Theorem 6.3.1

This routine, presented in Algorithm 18, serves as one of the two certification sub-procedures in the fixed-confidence algorithm. Given a confidence level δ, a query budget T, and a candidate Condorcet winner I, it sequentially tests whether one can certify—using at most T comparisons—that all pairwise gaps $(\Delta_{I,i})_{i\neq I}$ are positive with probability at least $1-\delta$. The budget is allocated uniformly across these gaps, and the procedure terminates as soon as either (i) a negative gap is detected, (ii) all gaps are certified positive, or (iii) the budget T is exhausted.

6.C.1 Proof of δ-correctness

Let c_0 denote the absolute numerical constant corresponding to the one appearing in the upper bound of Corollary 6.2.2. Theorem 6.3.1 states that Algorithm 17 with input $\delta \in (0,1)$ and

Algorithm 18: Test-CW

Input: $I \in [K], \delta, T$

$C \leftarrow [K] \setminus \{I\}$, $\tilde{\Delta}_{I,j} \leftarrow 0$ for all $j \in C$, $t \leftarrow 1$;

$n \leftarrow \log_2\left(\frac{T}{4K \log_{8/7}(K)}\right)$;

$N_{I,j} \leftarrow 0$ for all $j \in [K]$; ▷ Query counts

while $C \neq \emptyset$ and $t \leq T$ **do**

  Sample duel (I, j) with $j \in \operatorname{argmin}_{j \in C} N_{I,j}$ and update empirical means;

  $N_{I,j} \leftarrow N_{I,j} + 1$, $t \leftarrow t + 1$;

  for $j \in C$ **do**

    if $\tilde{\Delta}_{I,j} \geq \sqrt{\frac{\log\left(KN_{I,j}^2 \frac{n(n+1)}{\delta}\right)}{N_{I,j}}}$ **then**

      $C \leftarrow C \setminus \{j\}$; ▷ Certified positive gap

    end

    else if $\tilde{\Delta}_{I,j} \leq -\sqrt{\frac{\log\left(KN_{I,j}^2 \frac{n(n+1)}{\delta}\right)}{N_{I,j}}}$ **then**

      break; ▷ Negative gap detected

    end

  end

end

if $C = \emptyset$ **then**

  return True; ▷ Candidate certified

end

return False; ▷ Budget exhausted or contradiction

$c \geq 2/c_0$, it outputs an arm different from the CW with probability at most δ. Let ψ_δ denote the output of Algorithm 17 when the input is δ. We will prove that

$$\mathbb{P}(\psi_\delta \neq i^*) \leq \delta \ .$$

To prove this claim, we introduce the following notation. In the n-th iteration (i.e., the n-th call to Algorithm 16), denote by $\bar{\alpha}^{(n)}, \phi_1^{(n)}, \phi_2^{(n)}, I^{(n)}$ and $T^{(n)}$ the corresponding values of $\bar{\alpha}, \phi_1, \phi_2, I$ and T, and let $\varphi^{(n)} := \phi_1^{(n)} \vee \phi_2^{(n)}$. For convenience define, for all $n \geq 1$,

$$\delta_n := \frac{\delta}{8K^2 \log_{8/7}(K)\ \log(T^{(n)})\, n(n+1)} \ .$$

Let $S_k^{(n)}(\cdot)$ denote the score used at round k within the n-th call to Algorithm 16, and let $Z_k^{(w,n)}(\cdot)$ denote its weak component. Recall that each call to Algorithm 16 has at most $k_{\max} \leq \lceil \log_{8/7}(K) \rceil$ rounds. Finally, in the n-th call to Test-CW (Algorithm 18) with inputs $(I^{(n)}, \delta, T^{(n)})$, let $\tilde{\Delta}_{I^{(n)},j}$ denote the final empirical estimate of $\Delta_{I^{(n)},j}$ for each $j \neq I^{(n)}$.

If $\psi_\delta \neq i^*$, then for some $n \geq 1$ the algorithm must have certified an incorrect candidate, namely

$\{I^{(n)} \neq i^*\}$ and $\{\varphi^{(n)} = \texttt{True}\}$. Hence, by a union bound,

$$\begin{aligned}\mathbb{P}(\psi_\delta \neq i^*) &\leq \mathbb{P}\Big(\exists n \geq 1 :\ I^{(n)} \neq i^*,\ \varphi^{(n)} = \texttt{True}\Big) \\ &\leq \sum_{n=1}^{\infty} \mathbb{P}\Big(I^{(n)} \neq i^*,\ \phi_1^{(n)} = \texttt{True}\Big) + \sum_{n=1}^{\infty} \mathbb{P}\Big(I^{(n)} \neq i^*,\ \phi_2^{(n)} = \texttt{True}\Big)\ . \end{aligned} \tag{6.61}$$

We first bound the contribution of ϕ_1. On the event $\{I^{(n)} \neq i^*,\ \phi_1^{(n)} = \texttt{True}\}$, during the n-th run of Algorithm 16 the true Condorcet winner i^* must have been eliminated at some round $k < k_{\max}$ (otherwise the procedure would return $I^{(n)} = i^*$). By the definition of the selection $\bar{\alpha}^{(n)}$ and the condition $\phi_1^{(n)} = \texttt{True}$, this implies that for some $k < k_{\max}$,

$$S_k^{(n)}(i^*) < -\sqrt{\frac{2c\log(T^{(n)})\log(1/\delta_n)}{\lceil B_k^{(n)}/4\rceil}}.$$

Using $S_k^{(n)}(i^*) = \min\{\hat{\Delta}_{i^*,u}^{(k,n)}, 0\} + Z_k^{(w,n)}(i^*)$ (for the opponent u queried at that round) and the fact that $\min\{x, 0\} + y < -\eta$ implies $(x < -\eta/2)$ or $(y < -\eta/2)$, we get

$$\begin{aligned}\mathbb{P}\Big(I^{(n)} \neq i^*,\ \phi_1^{(n)} = \texttt{True}\Big) &\leq \sum_{k<k_{\max}} \mathbb{P}\left(\exists u \neq i^* :\ \min\{\hat{\Delta}_{i^*,u}^{(k,n)}, 0\} + Z_k^{(w,n)}(i^*) < -\sqrt{\frac{2c\log(T^{(n)})\log(1/\delta_n)}{\lceil B_k^{(n)}/4\rceil}}\right) \\ &\leq \sum_{k<k_{\max}} \sum_{u\neq i^*} \mathbb{P}\left(\hat{\Delta}_{i^*,u}^{(k,n)} < -\sqrt{\frac{c\log(T^{(n)})\log(1/\delta_n)}{2\lceil B_k^{(n)}/4\rceil}}\right) \\ &\quad + \sum_{k<k_{\max}} \mathbb{P}\left(Z_k^{(w,n)}(i^*) < -\sqrt{\frac{c\log(T^{(n)})\log(1/\delta_n)}{2\lceil B_k^{(n)}/4\rceil}}\right)\ . \end{aligned} \tag{6.62}$$

For the first term, since $\Delta_{i^*,u} > 0$ for all $u \neq i^*$, we can center and apply Hoeffding's inequality: for $N = \lceil B_k^{(n)}/4\rceil$,

$$\begin{aligned}\mathbb{P}\left(\hat{\Delta}_{i^*,u}^{(k,n)} < -\sqrt{\frac{c\log(T^{(n)})\log(1/\delta_n)}{2N}}\right) &\leq \mathbb{P}\left(\hat{\Delta}_{i^*,u}^{(k,n)} - \Delta_{i^*,u} < -\sqrt{\frac{\log(1/\delta_n)}{2N}}\right) \\ &\leq \exp\left(-2N\cdot\frac{\log(1/\delta_n)}{2N}\right) \leq \delta_n\ . \end{aligned} \tag{6.63}$$

For the second term in (6.62), Corollary 6.2.2 (with constant c_0) gives

$$\begin{aligned}\mathbb{P}\left(Z_k^{(w,n)}(i^*) < -\sqrt{\frac{c\log(T^{(n)})\log(1/\delta_n)}{2\lceil B_k^{(n)}/4\rceil}}\right) &\le \mathbb{P}\left(Z_k^{(w,n)}(i^*) < \Delta^{(k)}_{i^*,(\lceil|A_k|/8\rceil)} - \sqrt{\frac{c\log(T^{(n)})\log(1/\delta_n)}{2\lceil B_k^{(n)}/4\rceil}}\right)\\ &\le \log\left(\lceil * \rceil\,\frac{B_k^{(n)}}{2}\right)\exp\left(-c_0\cdot\frac{c\log(T^{(n)})\log(1/\delta_n)}{2\log(\left\lceil B_k^{(n)}/2\right\rceil)}\right)\\ &\le \log\left(\lceil * \rceil\,\frac{B_k^{(n)}}{2}\right)\delta_n, \end{aligned} \tag{6.64}$$

where we used $c \ge 2/c_0$, and $B_k^{(n)} \leqslant T^{(n)}$. Plugging (6.63) and (6.64) into (6.62), summing over $k < k_{\max}$ and using $\log(\lceil B_k^{(n)}/2\rceil) \le \log(T^{(n)})$ and $k_{\max} \le \lceil \log_{8/7}(K)\rceil$, we obtain

$$\begin{aligned}\sum_{n=1}^{\infty}\mathbb{P}\Big(I^{(n)} \neq i^*,\ \phi_1^{(n)} = \texttt{True}\Big) &\le \sum_{n=1}^{\infty}\Big((k_{\max}(K-1))\delta_n + k_{\max}\log(T^{(n)})\delta_n\Big)\\ &\le \sum_{n=1}^{\infty}\frac{\delta}{2n(n+1)} \le \frac{\delta}{2}. \end{aligned} \tag{6.65}$$

We now bound the contribution of ϕ_2 in (6.61). On the event $\{I^{(n)} \neq i^*,\ \phi_2^{(n)} = \texttt{True}\}$, the n-th call to Test-CW returns `True` although $I^{(n)}$ is not the Condorcet winner. In particular, for some $N \ge 1$ the test must have accepted the comparison against i^*, meaning that

$$\tilde{\Delta}_{I^{(n)},i^*} > \sqrt{\frac{\log\Big(KN^2\frac{n(n+1)}{\delta}\Big)}{N}}.$$

Hence, by a union bound over $n \ge 1$, $N \ge 1$ and all $i \neq i^*$, and since $\Delta_{i,i^*} \le 0$ when $i \neq i^*$,

$$\begin{aligned}\sum_{n=1}^{\infty}\mathbb{P}\Big(I^{(n)} \neq i^*,\ \phi_2^{(n)} = \texttt{True}\Big) &\le \sum_{n=1}^{\infty}\sum_{N=1}^{\infty}\sum_{i\neq i^*}\mathbb{P}\left(\tilde{\Delta}_{i,i^*} > \sqrt{\frac{\log\Big(KN^2\frac{n(n+1)}{\delta}\Big)}{N}}\right)\\ &\le \sum_{n=1}^{\infty}\sum_{N=1}^{\infty}\sum_{i\neq i^*}\mathbb{P}\left(\tilde{\Delta}_{i,i^*} - \Delta_{i,i^*} > \sqrt{\frac{\log\Big(KN^2\frac{n(n+1)}{\delta}\Big)}{N}}\right)\\ &\le \sum_{n=1}^{\infty}\sum_{N=1}^{\infty}\sum_{i\neq i^*}\frac{\delta}{4K\,N^2\,n(n+1)} \ \le\ \frac{\delta}{2}\,, \end{aligned} \tag{6.66}$$

where the last inequality follows from Hoeffding's inequality and $\sum_{N\ge1} 1/N^2 \le 2$.

Finally, combining (6.61) with (6.65) and (6.66) yields

$$\mathbb{P}(\psi_\delta \neq i^*) \le \frac{\delta}{2} + \frac{\delta}{2} = \delta\,,$$

which concludes the proof.

6.C.2 Proof of Theorem 6.3.1 (sample complexity statement)

We build on the guarantees established for Algorithm 16 (fixed-budget elimination with certification) to prove the second statement of Theorem 6.3.1. We use the notation: for each $i \neq i^*$, $\Delta_{i,(1)} \leq \cdots \leq \Delta_{i,(K-1)}$ denotes the ordered list of gaps $(\Delta_{i,j})_{j\neq i}$, and $K_{i;<0} := \big|\{j : \Delta_{i,j} < 0\}\big|$. Fix any vector $\boldsymbol{s} = (s_1, \ldots, s_K)$ such that $s_i \leq K_{i;<0}$ for all $i \neq i^*$ (and $K_{i^*;<0} = 0$ by convention), and recall

$$H_{\mathrm{certify}}(\boldsymbol{s}) = \sum_{i\neq i^*} \frac{1}{\Delta^2_{i,(s_i)}}, \qquad H^{(1)}_{\mathrm{explore}}(\boldsymbol{s}) = \max_{i\neq i^*} \frac{K}{s_i \Delta^2_{i,(s_i)}}, \qquad H^{(0)}_{\mathrm{explore}}(\boldsymbol{s}) = \sum_{i\neq i^*} \frac{K}{s_i \Delta^2_{i,(s_i)}} \ .$$

Let c_1 be the numerical constant in Theorem 6.B.1 and let c_2, c_3 be the numerical constants in Lemma 6.B.5. Let c_0 be the numerical constant in Corollary 6.2.2, and assume $c \geq 2/c_0$ as in the statement of Theorem 6.3.1. For concision, define

$$\begin{aligned}
G_{1,\delta} &:= \frac{32}{c_1} H_{\mathrm{cw}} \log(K) \log\Big(\frac{32}{c_1}\frac{K H_{\mathrm{cw}}}{\delta}\Big) \log\Big(c_1^{-1} H_{\mathrm{cw}} \log(K/\delta)\Big) \ , \\
G_{2,\delta} &:= \frac{512c}{c_2} H_{\mathrm{certify}}(\boldsymbol{s}) \ \log^3(K) \ \log\Big(\frac{32c}{c_2}\frac{K H^{(0)}_{\mathrm{explore}}(\boldsymbol{s})}{\delta}\Big) \ , \\
G_{3,\delta} &:= \frac{32c}{c_2} H^{(1)}_{\mathrm{explore}}(\boldsymbol{s}) \ \log^3(K) \ \log\left(\frac{2Kk_{\max}}{\delta}\right) \log\left(\frac{32c}{c_2} H^{(1)}_{\mathrm{explore}}(\boldsymbol{s}) \log^3(K) \log\left(\frac{2Kk_{\max}}{\delta}\right)\right) \ , \\
G_0 &:= \frac{2c_3}{c_2} \ H^{(0)}_{\mathrm{explore}}(\boldsymbol{s}) \ \log^5\Big(\frac{2c_3}{c_2} H^{(0)}_{\mathrm{explore}}(\boldsymbol{s})\Big) \ .
\end{aligned}$$

Algorithm 17 doubles the budget parameter T at each unsuccessful iteration; therefore, if one can show that whenever

$$T \in \Big[M_\delta, \ 2M_\delta\Big], \qquad \text{where } M_\delta := \min\Big\{G_{1,\delta}, \ G_{2,\delta} + G_{3,\delta} + G_0\Big\} \ , \tag{6.67}$$

the call to Algorithm 16 with inputs (δ, T, c) returns $\phi_1 \vee \phi_2 = \texttt{True}$ with probability at least $1 - 6\delta$, then it follows that the total number of queries N_δ used by Algorithm 17 is at most a universal constant multiple of M_δ with probability at least $1 - 6\delta$ (since the sum of a doubling schedule up to the first successful budget is at most 2 times that budget). We now verify this success probability for any T satisfying (6.67), distinguishing two regimes depending on which term attains the minimum.

Regime 1: $G_{1,\delta} \leq G_{2,\delta} + G_{3,\delta} + G_0$**.** Then, we focus on the regime $T \in [G_{1,\delta}, 2G_{1,\delta}]$. In this regime we certify correctness through the fixed-budget guarantee of Theorem 6.B.1. Indeed, for

$T \in [G_{1,\delta},\ 2G_{1,\delta}]$, we have

$$\begin{aligned}\mathbb{P}(I \neq i^*) &\leq 27K\log(K)\log(T)\exp\Big(-c_1\,\frac{T}{\log(T)\log(K)H_{\mathrm{cw}}}\Big)\\ &\leq 27K\log(K)\log(2G_{1,\delta})\exp\Big(-c_1\,\frac{G_{1,\delta}}{\log(2G_{1,\delta})\log(K)H_{\mathrm{cw}}}\Big)\,. \end{aligned} \tag{6.68}$$

We now use the explicit definition of $G_{1,\delta}$ and the crude upper bound

$$\log(2G_{1,\delta}) \leq 16\log\Big(c_1^{-1}H_{\mathrm{cw}}\log(K/\delta)\Big)\,, \tag{6.69}$$

which results from the expression of $G_{1,\delta}$, $K, H_{\mathrm{cw}} \geq 2$ and $\delta \in (0, 1/6)$. Using the bound (6.69) and plugging it back in (6.68), we obtain

$$\begin{aligned}\mathbb{P}\left(I \neq i^*\right) &\leq 432 \cdot K\log(K)\log(c_1^{-1}H_{\mathrm{cw}}\log(K/\delta)) \cdot \exp\Big(-2\cdot\log(32c_1^{-1}KH_{\mathrm{cw}}/\delta)\Big)\\ &\leq \delta \cdot \frac{432K\log(K)\log(c_1^{-1}H_{\mathrm{cw}}\log(K/\delta))\cdot\delta}{(32c_1^{-1}KH_{\mathrm{cw}})^2}\\ &\leq \delta\,.\end{aligned}$$

where in the last line we used the fact that $K, H_{\mathrm{cw}} \geq 2, \delta \in (0, 1/6)$ and $c_1 \in (0, 1)$. It remains to argue that, conditional on $I = i^*$, the auxiliary certification Test-CW (Algorithm 18) returns `True` with probability at least $1-2\delta$, hence overall $\mathbb{P}(\phi_1 \vee \phi_2 = \texttt{True}) \geq 1-3\delta$ in this regime. Run Test-CW with inputs (i^*, δ, T) and let T_i be the number of comparisons allocated to pair (i^*, i). By construction, $\sum_{i\neq i^*} T_i \leq T$. If Test-CW returns `False`, then either it exhausted the budget without eliminating all opponents, or it triggered a negative-deviation stopping rule. Formally, define

$$\mathcal{E}_1 := \Big\{\sum_{i\neq i^*} T_i = T\Big\}, \qquad \mathcal{E}_2 := \Big\{\exists j \neq i^*,\ \exists N \geq 1 :\ \tilde{\Delta}_{i^*,j}(N) \leq -\sqrt{\tfrac{\log(KN^2\,\frac{n(n+1)}{\delta})}{N}}\Big\},$$

so that $\{\phi_2 = \texttt{False}\} \subseteq \mathcal{E}_1 \cup \mathcal{E}_2$. Since $\Delta_{i^*,j} \geq 0$ for all $j \neq i^*$, Hoeffding's inequality and a union bound give

$$\begin{aligned}\mathbb{P}(\mathcal{E}_2) &\leq \sum_{j\neq i^*}\sum_{N\geq 1}\mathbb{P}\left(\tilde{\Delta}_{i^*,j}(N) - \Delta_{i^*,j} \leq -\sqrt{\tfrac{\log(KN^2\,\frac{n(n+1)}{\delta})}{N}}\right)\\ &\leq \sum_{j\neq i^*}\sum_{N\geq 1}\frac{\delta}{K\,n(n+1)\,N^2} \;\leq\; \delta\,,\end{aligned} \tag{6.70}$$

where we used in the last line the fact that $n(n+1) \geq 2 \geq \pi^2/6$.

Next, for each $i \neq i^*$ define

$$\bar{T}_i := \frac{16}{\Delta_{i^*,i}^2}\log\left(\frac{32Kn(n+1)}{\delta\Delta_{i^*,i}^2}\right)\,.$$

Lemma 6.C.1 below ensures that $\sum_{i\neq i^*} \bar{T}_i < G_{1,\delta} \leq T$, hence

$$\begin{aligned}\mathbb{P}(\mathcal{E}_1) = \mathbb{P}\left(\sum_{i\neq i^*} T_i = T\right) &\leq \mathbb{P}\left(\sum_{i\neq i^*} T_i \geq G_{1,\delta}\right) \leq \mathbb{P}\Big(\exists i \neq i^* : T_i > \bar{T}_i\Big) \\ &\leq \sum_{i\neq i^*} \mathbb{P}(T_i > \bar{T}_i)\ . \end{aligned} \tag{6.71}$$

If $T_i > \bar{T}_i$, then at time $N = \bar{T}_i$ the arm i was not eliminated, meaning

$$\tilde{\Delta}_{i^*,i}(\bar{T}_i) \ < \ \sqrt{\frac{\log\big(K\bar{T}_i^2\,\frac{n(n+1)}{\delta}\big)}{\bar{T}_i}}\ .$$

By Lemma 6.C.1, the RHS is at most $\Delta_{i^*,i} - \sqrt{\frac{\log(K\bar{T}_i^2\,\frac{n(n+1)}{\delta})}{2\bar{T}_i}}$, hence

$$\begin{aligned}\mathbb{P}(T_i > \bar{T}_i) &\leq \mathbb{P}\left(\tilde{\Delta}_{i^*,i}(\bar{T}_i) - \Delta_{i^*,i} < -\sqrt{\frac{\log(K\bar{T}_i^2\,\frac{n(n+1)}{\delta})}{2\bar{T}_i}}\right) \\ &\leq \sum_{N\geq 1} \frac{\delta}{K\,n(n+1)\,N^2} \ \leq\ \frac{\delta}{K}\ , \end{aligned} \tag{6.72}$$

where the last line uses Hoeffding and a union bound over $N \geq 1$. Combining (6.71) and (6.72) yields $\mathbb{P}(\mathcal{E}_1) \leq \delta$. Together with (6.70), we obtain $\mathbb{P}(\phi_2 = \texttt{False}) \leq 2\delta$, hence $\mathbb{P}(\phi_2 = \texttt{True}) \geq 1-2\delta$ when $I = i^*$. This completes Regime 1.

Regime 2: $G_{1,\delta} > G_{2,\delta} + G_{3,\delta} + G_0$. Then $T \in [G_{2,\delta} + G_{3,\delta} + G_0,\ 2(G_{2,\delta} + G_{3,\delta} + G_0)]$. We show that, for such T, the certification variable ϕ_1 in Algorithm 16 remains `True` with probability at least $1 - 2\delta$. Let $\bar{\alpha}_k$ be the arm ranked $|A_k| - \lceil |A_k|/8 \rceil + 1$ at round k according to scores $S_k(\cdot)$. Define

$$L_{k,\delta} := \sqrt{\frac{2c\log(T)}{\lceil B_k/4\rceil}\log\Big(\frac{1}{\delta_{n,K}}\Big)}, \qquad \delta_{n,K} := \frac{\delta}{8K^2\log_{8/7}(K)\log(T)\,n(n+1)}\ .$$

By the update rule for ϕ_1, the event $\{\phi_1 = \texttt{False}\}$ implies that for some $k \leq k_{\max}$, $S_k(\bar{\alpha}_k) \geq -L_{k,\delta}$. The definition of $\bar{\alpha}_k$ entails that at most $\lceil * \rceil\,|A_k|/8$ arms have score not larger than $-L_{k,\delta}$, i.e.

$$\{S_k(\bar{\alpha}_k) \geq -L_{k,\delta}\} \ \subseteq\ \left\{\sum_{\alpha\in A_k} \mathbb{1}\big(S_k(\alpha) < -L_{k,\delta}\big) \leq \lceil * \rceil\,\frac{|A_k|}{8}\right\}\ . \tag{6.73}$$

We now relate $-L_{k,\delta}$ to the threshold $\frac{1}{2}\bar{\Delta}_k$ used in Lemma 6.B.5. Recall the definitions (as in the proof of Theorem 6.B.3): let

$$E_k := \Big\{\alpha \in A_k :\ \Delta^{(k)}_{\alpha,(\lceil |A_k|/4\rceil)} \leq 0\Big\}, \qquad |E_k| \geq \lceil * \rceil\,|A_k|/4\ ,$$

and define the 7/8-quantile $\bar{\Delta}_k := \Delta_{E_k:\lceil(7/8)|E_k|\rceil} \leq 0$ and the subset

$$F_k := \left\{\alpha \in E_k : \ \Delta_{\alpha,(s_\alpha)} \leq \bar{\Delta}_k\right\} .$$

Recall that by definition we have $\bar{\Delta}_k \leq 0$. Moreover, $\bar{\Delta}_k \to 0$ implies that $H^{(\boldsymbol{s})}_{\text{certify}} \to \infty$ and the bound resulting on the choice of $\boldsymbol{s}$ are vacuous in this case. We therefore suppose that $\bar{\Delta}_k < 0$. Lemma 6.C.2 ensures that for all $k \leq k_{\max}$ and all T in the present regime,

$$-L_{k,\delta} \ \geq \ \frac{1}{2}\,\bar{\Delta}_k \ . \tag{6.74}$$

We assume that $|A_k| \geq 3$, the case $|A_k| = 2$ is treated in the end. Using (6.74) inside (6.73) gives

$$\begin{aligned}\mathbb{P}(S_k(\bar{\alpha}_k) \geq -L_{k,\delta}) &\leq \mathbb{P}\left(\sum_{\alpha\in A_k} \mathbb{1}\left(S_k(\alpha) < \tfrac{1}{2}\bar{\Delta}_k\right) \leq \lceil * \rceil\, \frac{|A_k|}{8}\right) \\ &= \mathbb{P}\left(\sum_{\alpha\in A_k} \mathbb{1}\left(S_k(\alpha) \geq \tfrac{1}{2}\bar{\Delta}_k\right) \geq |A_k| - \lceil * \rceil\, \frac{|A_k|}{8}\right) \ . \end{aligned} \tag{6.75}$$

Since $|A_k| - \lceil * \rceil\, |A_k|/8 = |A_{k+1}|$ and Lemma 6.B.4 gives $|A_{k+1} \cap F_k| \geq \lceil * \rceil\, |F_k|/3$, the RHS of (6.75) is upper bounded by

$$\mathbb{P}\left(\sum_{\alpha\in F_k} \mathbb{1}\left(S_k(\alpha) \geq \tfrac{1}{2}\bar{\Delta}_k\right) \geq \lceil * \rceil\, \frac{|F_k|}{3}\right) \ .$$

Lemma 6.B.5 then yields, for a numerical constant $c_2 > 0$,

$$\mathbb{P}(S_k(\bar{\alpha}_k) \geq -L_{k,\delta}) \leq \exp\left(-c_2 \frac{T - c_3 H^{(0)}_{\text{explore}}(\boldsymbol{s}) \log^5(H^{(0)}_{\text{explore}}(\boldsymbol{s}))}{\log^3(K)\log(T)} \cdot \frac{1}{H^{(1)}_{\text{explore}}(\boldsymbol{s})}\right) \ . \tag{6.76}$$

Next, we use $T \geq G_{3,\delta} + G_0$ with Lemma 6.C.3, which turns the last inequality into a bound on the exponent term of (6.76) leading to

$$\mathbb{P}(S_k(\bar{\alpha}_k) \geq -L_{k,\delta}) \leq \frac{\delta}{2K k_{\max}} \ . \tag{6.77}$$

which is the desired per-round bound.

Suppose that $|A_k| = 2$, then we have $E_k = F_k :- \{\alpha\}$. Therefore

$$\begin{aligned}\mathbb{P}(S_k(\bar{\alpha}_k) \geq -L_{k,\delta}) &\leq \mathbb{P}\left(S_k(\alpha) \geq \frac{1}{2}\bar{\Delta}\right) \\ &\leq \exp\left(-c_2 \frac{T - c_3 H^{(0)}_{\mathrm{explore}}(\boldsymbol{s}) \log^5(H^{(0)}_{\mathrm{explore}}(\boldsymbol{s}))}{\log^3(K)\log(T)} \cdot \frac{1}{H^{(1)}_{\mathrm{explore}}(\boldsymbol{s})}\right) \\ &\leq \frac{\delta}{2K k_{\max}} \ .\end{aligned}$$

where in the second line we used Lemma 6.B.6 (which provides a smaller upper bound than the one given above).

Finally, since $\{\phi_1 = \texttt{False}\} \subseteq \bigcup_{k \leq k_{\max}} \{S_k(\bar{\alpha}_k) \geq -L_{k,\delta}\}$, a union bound and (6.77) yield

$$\mathbb{P}(\phi_1 = \texttt{False}) \ \leq \ \sum_{k=1}^{k_{\max}} \frac{\delta}{2K\, k_{\max}} \ \leq \ \delta \ ,$$

which completes Regime 2.

Conclusion. In either regime, for any T satisfying (6.67) the call to Algorithm 16 returns $\phi_1 \vee \phi_2 = \texttt{True}$ with probability at least $1-6\delta$. Since Algorithm 17 doubles T until this event occurs, its total number of queries N_δ is at most a universal constant multiple of $M_\delta = \min\{G_{1,\delta}, G_{2,\delta} + G_{3,\delta} + G_0\}$ with probability at least $1-6\delta$. Absorbing numerical constants into $\bar{c}_1, \bar{c}_2$ yields the stated bounds of Theorem 6.3.1.

The lemmas below are technical.

Lemma 6.C.1. *Consider the notation introduced in the proof of Theorem 6.3.1. Then we have*

$$\sum_{i \neq i^*} \bar{T}_i < G_{1,\delta} \ .$$

Moreover, for all $i \neq i^*$

$$2\sqrt{\frac{\log\left(K\bar{T}_i^2 \frac{n(n+1)}{\delta}\right)}{\bar{T}_i}} < \Delta_{i^*,i} \ .$$

Proof. We have

$$\begin{aligned}\sum_{i \neq i^*} \bar{T}_i &= \sum_{i \neq i^*} \frac{16}{\Delta^2_{i^*,i}} \log\left(\frac{32Kn(n+1)}{\delta \Delta^2_{i^*,i}}\right) \\ &\leq \sum_{i \neq i^*} \frac{16}{\Delta^2_{i^*,i}} \log\left(\frac{32Kn(n+1)}{\delta} H_{\mathrm{cw}}\right) \\ &\leq 16 H_{\mathrm{cw}}(\log(32K H_{\mathrm{cw}}/\delta) + \log(n(n+1))) \ . \end{aligned} \tag{6.78}$$

Therefore we only need to prove that

$$16H_{\rm cw}(\log(32KH_{\rm cw}/\delta)+\log(n(n+1)))\le G_{1,\delta}\,,$$

which is equivalent to

$$\log\left(\frac{32KH_{\rm cw}}{\delta}\right)+\log(n(n+1))\le\frac{32}{16c_1}\log(K)\log(KH_{\rm cw}/\delta)\log(c_1^{-1}H_{\rm cw}\log(K/\delta))\,.$$

Observe that to prove the bound above we just need an upper bound on $\log(n(n+1))$, more precisely, given that $\log(K)\log(c_1^{-1}H_{\rm cw}\log(K/\delta)\ge 2$ and $c_1<\frac{1}{2}$, it suffices the show that

$$\log(n(n+1))\le\frac{1}{c_1}\log(K)\log(KH_{\rm cw}/\delta) \tag{6.79}$$

We have from the definition of $n=\log_2\left(\frac{T}{4K\log_{8/7}(K)}\right)$ and $T\le 2G_{1,\delta}$ that

$$\begin{aligned}
n&\le\log_2\left(\frac{2G_{1,\delta}}{2K\log_{8/7}(K)}\right)\\
&\le\log_2\left(\frac{32\log(8/7)}{c_1}\frac{H_{\rm cw}}{K}\log\left(\frac{32KH_{\rm cw}}{c_1\delta}\right)\log\left(c_1^{-1}H_{\rm cw}\log(K/\delta)\right)\right)\\
&\le\log_2\left(\frac{6}{c_1^2}\frac{H_{\rm cw}^2}{K}\log^2\left(\frac{32KH_{\rm cw}}{c_1\delta}\right)\right)\\
&\le 2\log_2\left(\frac{6}{c_1}H_{\rm cw}\log\left(\frac{32KH_{\rm cw}}{c_1\delta}\right)\right)
\end{aligned}$$

This gives

$$\begin{aligned}
\log(n(n+1))&\le 2\log(n+1)\\
&\le 2\log\left(2\log_2\left(\frac{12}{c_1}\frac{H_{\rm cw}}{K}\log\left(\frac{32KH_{\rm cw}}{c_1\delta}\right)\right)\right)\,,
\end{aligned}$$

which gives (6.79), and leads to the first claim of the lemma.
The second claim of the lemma is equivalent to $\sqrt{\frac{\log\left(K\bar{T}_i^2\frac{n(n+1)}{\delta}\right)}{\bar{T}_i}}<\frac{\Delta_{i^*,i}}{2}$ which in turn is implied by $\log\left(K\bar{T}_i^2\frac{n(n+1)}{\delta}\right)<4\cdot\log\left(\frac{32Kn(n+1)}{\delta\Delta_{i^*,i}^2}\right)$, which is verified given the definition of $\bar{T}_i$.

□

Lemma 6.C.2. *Consider the notation introduced in the proof of Theorem 6.3.1. If $\bar{\Delta}_k<0$ and*

$$T\in[G_{2,\delta}+G_{3,\delta}+G_0,\ 2(G_{2,\delta}+G_{3,\delta}+G_0)],$$

then

$$-L_{k,\delta}\ge\frac{1}{2}\bar{\Delta}_k\ .$$

Proof. Let

$$\delta_{n,K} = \frac{\delta}{8K^2 \log_{8/7}(K) \log(T)\, n(n+1)}.$$

Assume $\bar{\Delta}_k < 0$. Since $L_{k,\delta} \geq 0$, the inequality $-L_{k,\delta} \geq \frac{1}{2}\bar{\Delta}_k$ is equivalent to $L_{k,\delta} \leq -\frac{1}{2}\bar{\Delta}_k$, i.e.

$$L_{k,\delta}^2 \;\leq\; \frac{\bar{\Delta}_k^2}{4}. \tag{6.80}$$

Recalling

$$L_{k,\delta}^2 = \frac{2c\log(T)}{\lceil B_k/4 \rceil}\, \log\Big(\frac{1}{\delta_{n,K}}\Big), \quad B_k = \lfloor * \rfloor \frac{T}{|A_k| \log_{8/7}(K)}\ ,$$

we have (6.80) is implied by

$$\log(T)\, \log\Big(\frac{1}{\delta_{n,K}}\Big) \;\leq\; \frac{\bar{\Delta}_k^2}{8c}\, \frac{T}{|A_k| \log_{8/7}(K)}\ . \tag{6.81}$$

Next, using the inequality (6.57)

$$\frac{|A_k|}{\bar{\Delta}_k^2} \;\leq\; 32 \sum_{\alpha \in E_k} \frac{1}{\Delta_{\alpha,(s_\alpha)}^2}\ ,$$

and the fact that $i^* \notin E_k$ (since all $\Delta_{i^*,j} > 0$ so $\Delta^{(k)}_{i^*,(\lceil |A_k|/4 \rceil)} > 0$), we have $\sum_{\alpha \in E_k} \frac{1}{\Delta^2_{\alpha,(s_\alpha)}} \leq H_{\text{certify}}(\boldsymbol{s})$. Hence

$$\frac{\bar{\Delta}_k^2}{|A_k|} \;\geq\; \frac{1}{32\, H_{\text{certify}}(\boldsymbol{s})}\ . \tag{6.82}$$

Moreover, using the expression of $\delta_{n,K}$ with $n \leq \log_2\left(\frac{T}{2K \log_{8/7}(K)}\right)$, we have

$$\begin{aligned}
\log\Big(\frac{1}{\delta_{n,K}}\Big) &\leq \log\Big(\frac{8K^2 \log_{8/7}(K)}{\delta}\Big) + \log\log(T) + \log(n(n+1)) \\
&\leq \log\Big(\frac{8K^2 \log_{8/7}(K)}{\delta}\Big) + \log\log(T) + 2\log\log_2\left(\frac{2T}{K \log_{8/7}(K)}\right)\ .
\end{aligned} \tag{6.83}$$

Combining (6.82), (6.83) with (6.81) we conclude that we only need that T satisfies the bound

$$\log(T)\left[\log\Big(\frac{8K^2 \log_{8/7}(K)}{\delta}\Big) + \log\log(T) + 2\log\log_2\left(\frac{2T}{K \log_{8/7}(K)}\right)\right] \leq \frac{T}{512c \log_{8/7}(K) H_{\text{certify}}(\boldsymbol{s})}\ . \tag{6.84}$$

Give that $T \geq G_{2,\delta} + G_0$, using the expressions of $G_{2,\delta}$ and G_0, with the statement of the technical Lemma 6.C.4, we conclude that (6.84) is satified, which concludes the proof. □

Lemma 6.C.3. *Suppose that $T \geq G_0 + G_{3,\delta}$. Then we have*

$$\exp\left(-c_2 \frac{T - c_3 H^{(0)}_{explore}(\boldsymbol{s}) \log^5(H^{(0)}_{explore})(\boldsymbol{s})}{\log(K)\log(T)} \frac{1}{H^{(1)}_{explore}(\boldsymbol{s})}\right) \leq \frac{\delta}{2Kk_{\max}} \ .$$

Proof. The desired inequality is equivalent to

$$\frac{T - c_3 H^{(0)}_{\text{explore}}(\boldsymbol{s}) \log^5(H^{(0)}_{\text{explore}}(\boldsymbol{s}))}{\log(T)} \ \geq \ \frac{1}{c_2} H^{(1)}_{\text{explore}}(\boldsymbol{s}) \log^3(K) \log\left(\frac{2Kk_{\max}}{\delta}\right) \ . \tag{6.85}$$

Define

$$A :- 32 \frac{c}{c_2} H^{(1)}_{\text{explore}}(\boldsymbol{s}) \log^3(K) \log\left(\frac{2Kk_{\max}}{\delta}\right) \ .$$

By assumption,

$$T \ \geq \ 2\frac{c_3}{c_2} H^{(0)}_{\text{explore}}(\boldsymbol{s}) \log^5(H^{(0)}_{\text{explore}}(\boldsymbol{s})) \ + \ A \log(A) \ .$$

Since $c_2 \leq 1$ (as is the case for the numerical constant c_2 coming from the preceding bounds), the first term implies $T \geq 2H^{(0)}_{\text{explore}}(\boldsymbol{s}) \log^5(H^{(0)}_{\text{explore}}(\boldsymbol{s}))$, hence

$$T - c_3 H^{(0)}_{\text{explore}}(\boldsymbol{s}) \log^5(H^{(0)}_{\text{explore}}(\boldsymbol{s})) \geq \ \frac{T}{2}. \tag{6.86}$$

Next, the function $f(x) := x/\log x$ for $x > e$ is increasing. Moreover, we have $\log(2Kk_{\max}/\delta) > 1$ and $H^{(1)}_{\text{explore}}(\boldsymbol{s}) \geq 4$ (since $|\Delta|_{i,(s_i)} \leq \frac{1}{2}$ and $s_i \leq K$), and with $c \geq 1$ and $c_2 \leq 1$ this yields $A \geq 32 \cdot 4 \cdot (\log 2)^3 \cdot \log 4 > e$. Therefore, $A \log A > e$ and in particular $\log(A \log A) > 0$.

Since $T \geq A \log A$ and f is increasing on (e, ∞), we obtain

$$\frac{T}{\log T} \ = \ f(T) \ \geq \ f(A \log A) \ = \ \frac{A \log A}{\log(A \log A)}. \tag{6.87}$$

Finally, because $A > e$, we have $\log(A \log A) = \log A + \log\log A \leq \log A + \log A = 2 \log A$, and therefore

$$\frac{A \log A}{\log(A \log A)} \ \geq \ \frac{A \log A}{2 \log A} \ = \ \frac{A}{2}. \tag{6.88}$$

Combining (6.86), (6.87), and (6.88) gives

$$\frac{T - c_3 H^{(0)}_{\text{explore}}(\boldsymbol{s}) \log^5(H^{(0)}_{\text{explore}}(\boldsymbol{s}))}{\log T} \ \geq \ \frac{1}{2} \cdot \frac{T}{\log T} \ \geq \ \frac{1}{2} \cdot \frac{A}{2} \ = \ \frac{A}{4}.$$

By the definition of A,

$$\frac{A}{4} \ = \ 8 \frac{c}{c_2} H_1 \log^3(K) \ \log(2Kk_{\max}/\delta) \ \geq \ \frac{1}{c_2} H_1 \log^3(K) \ \log(2Kk_{\max}/\delta),$$

where we used $c \geq 1$. This proves (6.85), and hence

$$\exp\left(-c_2 \frac{T - c_3 H^{(0)}_{\text{explore}}(\boldsymbol{s}) \log^5(H^{(0)}_{\text{explore}}(\boldsymbol{s}))}{\log^3(K)\log(T)\, H_1}\right) \leq e^{-\log(2Kk_{\max}/\delta)} = \frac{\delta}{2Kk_{\max}} \ .$$

□

Lemma 6.C.4. *Let $K \geq 2$, $\delta \in (0,1)$, if $T \geq G_0 + G_{2,\delta}$, then*

$$\log(T)\left[\log\Big(\frac{8K^2 \log_{8/7}(K)}{\delta}\Big) + \log\log(T) + 2\log\log_2\left(\frac{2T}{K\log_{8/7}(K)}\right)\right] \leq \frac{T}{512c\log_{8/7}(K) H_{certify}(\boldsymbol{s})} \ .$$

Proof. Let $L := \log_{8/7}(K)$, $H := H_{\text{certify}}(\boldsymbol{s})$, and set

$$M := 512cLH, \qquad B := \log\Big(\frac{8K^2 L}{\delta}\Big) + 3\log(8M) + 10 \ .$$

By the definition of G_0 and $G_{2,\delta}$, and given $c_2 < 1/8$ the assumption $T \geq G_0 + G_{2,\delta}$ implies in particular that

$$T \geq T_0 := 2048 \cdot cLHB \log(2048 \cdot cLHB) \qquad \text{and} \qquad T \geq e^2.$$

Moreover, we have for $T \geq 2$,

$$\log\log_2\Big(\frac{2T}{KL}\Big) \leq \log\log(T) + 2 \ ,$$

Therefore the left-hand side is at most

$$u(T) := \log(T)\left[\log\Big(\frac{8K^2 L}{\delta}\Big) + 3\log\log T + 4\right] \ .$$

The function $u(T)/T$ is decreasing on $[e^2, \infty)$. Hence for all $T \geq T_0$,

$$u(T) \leq \frac{T}{T_0}\, u(T_0) \ .$$

Now $T_0 = 2048 \cdot cLHB \log(2048 \cdot cLHB)$ (given that $G_{2,\delta} \geq 2048 \cdot cLHB \log(2048 \cdot cLHB)$) gives

$$\log T_0 = \log(2048 \cdot cLHB) + \log\log(2048 \cdot cLHB) \leq 2\log(2048 \cdot cLHB) \ ,$$

and

$$\log\log T_0 \leq \log\log(2048 \cdot cLHB) + 1 \ ,$$

so

$$u(T_0) \leq 2\log(2048 \cdot cLHB)\big(\log\Big(\frac{8K^2 L}{\delta}\Big) + 3\log\log(2048 \cdot cLHB) + 7\big).$$

Moreover,

$$\begin{aligned}\log\log(2048\cdot cLHB) &\le \log(2048\cdot cLHB)\\ &= \log(2048\cdot cLH)+\log B\\ &\le \log(2048\cdot cLH)+B\ ,\end{aligned}$$

and $B=\log\left(\frac{8K^2L}{\delta}\right)+3\log(2048\cdot cLH)+10$, hence

$$\begin{aligned}\log\left(\frac{8K^2L}{\delta}\right)+3\log\log(2048\cdot cLHB)+7 &\le \log\left(\frac{8K^2L}{\delta}\right)+3\log(2048\cdot cLH)+3B+7\\ &= 4B-3\le 4B\ .\end{aligned}$$

Therefore

$$u(T_0)\le 2\log(2048\cdot cLHB)\cdot 4B = 8B\log(2048\cdot cLHB)=\frac{T_0}{512\cdot cLH}.$$

Combining the previous displays yields $u(T)\le T/M$ for all $T\ge T_0$, i.e.

$$\log(T)\left[\log\left(\frac{8K^2L}{\delta}\right)+\log\log(T)+2\log\log_2\Big(\frac{2T}{KL}\Big)\right]\le\frac{T}{512\,cLH}\ ,$$

which is exactly the desired inequality. □

6.D Proofs of Section 6.4

In this section, we provide all proofs for the instance-dependent fixed-confidence lower bounds.

We begin with a short roadmap in Subsection 6.D.1 describing the classical change-of-measure arguments underlying all our constructions (along the way, we fix some notation). In Subsection 6.D.2, we prove Proposition 6.4.1, separating the expected budget bound (Subsection 6.D.2.1) from the high-probability quantile bound (Subsection 6.D.2.2).

In Subsection 6.D.3, we explain the construction leading to Theorem 6.4.2 and state a more precise formulation in Theorem 6.D.3, proved in Subsection 6.D.4. Corollary 6.4.4 follows in Subsection 6.D.5.

Finally, Subsection 6.D.6 discusses lower bounds preserving CW row structure.

6.D.1 Roadmap on change-of-measure lower bounds

All our proofs follow a common three-step structure.

First step: reference and alternative instances. We fix a reference instance $\boldsymbol{\Delta}\in\mathbb{D}_{\mathrm{cw}}$, that is, a gap matrix admitting a (unique) Condorcet winner $i^*(\boldsymbol{\Delta})$. In the fixed-confidence regime, our lower bounds are instance-dependent, so all constructions are built directly from the given $K\times K$

matrix

$$\boldsymbol{\Delta} = (\Delta_{i,j})_{i,j\in[K]}.$$

For some results (e.g., Theorem 6.4.2), we also consider a local class of instances obtained from $\boldsymbol{\Delta}$ by permuting the negative entries of each row, while preserving prescribed structural features (CW, sign structure, multiset of negative entries, effective sparsity, and so on). This leads to families $\{\boldsymbol{\Delta}^{(\pi)}\}_\pi$ indexed by permutations π, but the reference object remains $\boldsymbol{\Delta}$.

For each suboptimal arm $k \neq i^*$, we construct an alternative instance

$$\boldsymbol{\Delta}^{(k)}$$

in such a way that i^* is no longer the (strong) Condorcet winner, while k becomes the CW or at least a weak CW, in the sense that the k-th row of $\boldsymbol{\Delta}^{(k)}$ contains only non-negative entries. A typical construction consists in modifying only the k-th row of $\boldsymbol{\Delta}$, for example by setting all its negative entries to a small constant $\epsilon \geq 0$, and updating the k-th column so as to preserve symmetry. In more refined arguments, we first permute the negative entries within each row.

Second step: total variation. We then exploit the properties of a given algorithm A in order to exhibit, for each $k \neq i^*$, a separating event B_k on which the two laws assign very different probabilities,

$$\mathbb{P}_{\boldsymbol{\Delta},A}(B_k) \text{ is large (typically } \geq 1-\delta), \qquad \mathbb{P}_{\boldsymbol{\Delta}^{(k)},A}(B_k) \text{ is small (typically } \leq \delta).$$

Here $\mathbb{P}_{\boldsymbol{\Delta},A}$ denotes the law of all observations (and internal randomness) when algorithm A interacts with environment $\boldsymbol{\Delta}$. A natural choice, when A is δ-correct for CW identification, is the event $\{\hat{\imath} = k\}$, where $\hat{\imath}$ is the recommendation output by A. For quantile (high-probability) lower bounds, we additionally introduce the $(1-\delta)$-quantile χ of the budget N_δ under $\mathbb{P}_{\boldsymbol{\Delta},A}$, and we consider events such as $\{N_\delta \leq \chi\}$ or their intersections with identification events. The precise choice of B_k varies from theorem to theorem, but the goal is always to produce a set on which the two laws $\mathbb{P}_{\boldsymbol{\Delta},A}$ and $\mathbb{P}_{\boldsymbol{\Delta}^{(k)},A}$ have very different probabilities, thereby enforcing a large total-variation distance:

$$\mathrm{TV}(\mathbb{P}_{\boldsymbol{\Delta},A}, \mathbb{P}_{\boldsymbol{\Delta}^{(k)},A}) \;\geq\; \left|\mathbb{P}_{\boldsymbol{\Delta},A}(B_k) - \mathbb{P}_{\boldsymbol{\Delta}^{(k)},A}(B_k)\right|.$$

The total-variation distance is then controlled from above through a standard data-processing inequality: we use either Pinsker's inequality, the Bretagnolle–Huber inequality, or a Fano-type inequality (see Lemma 6.F.4) to relate TV to the Kullback–Leibler divergence. For example,

$$\mathrm{TV}(P,Q) \;\leq\; \sqrt{\tfrac{1}{2}\,\mathrm{KL}(P,Q)}, \qquad \text{or} \qquad 1-\mathrm{TV}(P,Q) \;\geq\; \tfrac{1}{2}\exp\big(-\mathrm{KL}(P,Q)\big).$$

The choice depends on the error-probability regime: Bretagnolle–Huber is convenient in the very small-δ regime, while Pinsker is often sharper in moderate-error regimes.

Third step: decomposition and control of the KL divergence. The last step is to decompose the Kullback–Leibler divergence between the laws induced by A under the two environments. Let $N_{i,j}$ denote the total number of observed duels of the ordered pair (i,j) between time 1 and the stopping time N_δ, and let $N_{\{i,j\}} = N_{i,j} + N_{j,i}$ be the total number of observations of the unordered pair $\{i,j\}$. A standard KL-decomposition for adaptive bandit algorithms (see, e.g., Lattimore and Szepesvári, 2020, Lemma 15.1) yields

$$\mathrm{KL}(\mathbb{P}_{\boldsymbol{\Delta},A}, \mathbb{P}_{\boldsymbol{\Delta}^{(k)},A}) \;=\; \sum_{1\le i<j\le K} \mathbb{E}_{\boldsymbol{\Delta},A}[N_{\{i,j\}}]\, \mathrm{KL}(\nu_{\Delta_{i,j}}, \nu_{\Delta_{i,j}^{(k)}}),$$

where $\nu_{\Delta_{i,j}}$ denotes the Bernoulli distribution with parameter $1/2+\Delta_{i,j}$. In all our constructions, $\boldsymbol{\Delta}$ and $\boldsymbol{\Delta}^{(k)}$ differ only on a small set of pairs (typically those involving arm k), so the sum reduces to those indices. The Bernoulli KL-divergence admits the classical upper bound, for $p,q\in(0,1)$,

$$\mathrm{kl}(p,q) \;\le\; \frac{(p-q)^2}{q(1-q)},$$

which, in our setting, leads to bounds of the form

$$\mathrm{KL}(\mathbb{P}_{\boldsymbol{\Delta},A}, \mathbb{P}_{\boldsymbol{\Delta}^{(k)},A}) \;\lesssim\; \sum_{\{i,j\}\ \text{modified}} \mathbb{E}_{\boldsymbol{\Delta},A}[N_{\{i,j\}}]\, (\Delta_{i,j} - \Delta_{i,j}^{(k)})^2.$$

Combining this upper bound with the lower bound on total variation from the third step yields a constraint that any δ-correct algorithm A must satisfy. Rearranging these inequalities then produces a lower bound on the sample complexity, typically expressed in terms of the expected budget $\mathbb{E}_{\boldsymbol{\Delta},A}[N_\delta]$ or on the $(1-\delta)$-quantile of N_δ, and involving the instance-dependent hardness parameters (such as $\Delta_{i,(1)}$ or $\|\Delta_i^-\|_2^2/K_{i;<0}$).

6.D.2 Proof of Proposition 6.4.1

We prove separately the bound in expectation (paragraph 6.D.2.1) and the bound on the $(1-\delta)$-quantile of N_δ (paragraph 6.D.2.2). The two arguments share the same change-of-measure structure (see Section 6.D.1) and differ only in the way we exploit either the identification rule or the stopping rule to construct a separating event. The expectation bound 6.89 already appears as Theorem 5.2 in Haddenhorst et al. (2021b), but we provide here a proof for completeness.

6.D.2.1 Bound in expectation from Proposition 6.4.1

Fix $K\geqslant 2$ and $\delta\in(0,1)$. Consider an algorithm A that is δ-correct over the entire class $\mathbb{D}_{\mathrm{cw}}$. We fix any matrix $\boldsymbol{\Delta}\in\mathbb{D}_{\mathrm{cw}}$ such that $i^*=1$ and prove the following expected lower bound that holds for this specific instance:

$$\mathbb{E}_{\boldsymbol{\Delta},A}[N_\delta] \;\geqslant\; \frac{1}{4}\sum_{i\neq i^*} \frac{\log\left(\frac{1}{4\delta}\right)}{\Delta_{i,(1)}^2}\,. \tag{6.89}$$

Sketch of proof. We follow the three-step roadmap of Section 6.D.1. (i) Reference and alternative instances. Fix any $\boldsymbol{\Delta} \in \mathbb{D}_{\mathrm{cw}}$ with CW $i^* = 1$; this is our reference instance. For each suboptimal arm $k \neq 1$, we construct an alternative instance $\boldsymbol{\Delta}^{(k)}$ by lifting all non-positive entries in row k to a small constant $\epsilon > 0$ (and adjusting the k-th column to preserve symmetry), so that k becomes the CW in $\boldsymbol{\Delta}^{(k)}$. (ii) Separating event and total variation. Since the algorithm is δ-correct for CW identification, it must distinguish $\boldsymbol{\Delta}$ from each $\boldsymbol{\Delta}^{(k)}$ with error at most δ when deciding between i^* and k as CW. Using the test event $B_k = \{\hat{\imath} = k\}$, we obtain that the total-variation distance between $\mathbb{P}_{\boldsymbol{\Delta},A}$ and $\mathbb{P}_{\boldsymbol{\Delta}^{(k)},A}$ is at least $1 - 2\delta$, which, via the Bretagnolle–Huber inequality, yields a lower bound on $\mathrm{KL}(\mathbb{P}_{\boldsymbol{\Delta},A}, \mathbb{P}_{\boldsymbol{\Delta}^{(k)},A})$. (iii) KL decomposition. We then decompose this KL along unordered pairs. The two instances $\boldsymbol{\Delta}$ and $\boldsymbol{\Delta}^{(k)}$ differ only on duels involving arm k, and for each such pair the Bernoulli parameters differ by at most a constant of order $|\Delta_{k,(1)}|$. A Bernoulli KL upper bound, combined with the decomposition, forces $\mathbb{E}_{\boldsymbol{\Delta},A}[N_k] \gtrsim \Delta_{k,(1)}^{-2} \log(1/\delta)$, where N_k counts duels involving k. Summing this constraint over all $k \neq 1$ yields the desired lower bound (6.89).

Proof. Step 1: reference and perturbed instances.

The argument is fully instance-dependent: we fix an arbitrary gap matrix $\boldsymbol{\Delta} \in \mathbb{D}_{\mathrm{cw}}$ with Condorcet winner $i^* = 1$, and work throughout with this specific instance. We denote $\mathbb{P}$ the probability induced by the interaction between $\boldsymbol{\Delta}$ and A.

Let $\epsilon > 0$ be a constant, arbitrary small. Let $k \neq 1$ be an arm that is not the CW under $\boldsymbol{\Delta}$. A simple way to modify $\boldsymbol{\Delta}$ so that k becomes the CW, is to make all non-positive entries in the k-th row of $\boldsymbol{\Delta}$ equal to ϵ.

Construct the gap matrix $\boldsymbol{\Delta}^{(k)}$ as follows. For all $i, j \notin \{1, k\}$, set $\Delta_{i,j}^{(k)} = \Delta_{i,j}$. Set $\Delta_{k,1}^{(k)} = \epsilon$ and $\Delta_{1,k}^{(k)} = -\epsilon$. Finally, for each $j \notin \{1, k\}$, define

$$\Delta_{k,j}^{(k)} = \begin{cases} \Delta_{k,j}, & \text{if } \Delta_{k,j} > 0, \\ \epsilon, & \text{if } \Delta_{k,j} \leqslant 0, \end{cases} \qquad \Delta_{j,k}^{(k)} = -\Delta_{k,j}^{(k)}. \tag{6.90}$$

For ϵ small enough, the modified matrix $\boldsymbol{\Delta}^{(k)}$ can be represented as

$$\boldsymbol{\Delta}^{(k)} = \begin{pmatrix} 0 & \Delta_{1,2} & \cdots & \Delta_{1,k-1} & -\epsilon & \Delta_{1,k+1} & \cdots & \Delta_{1,K} \\ \Delta_{2,1} & 0 & \cdots & \Delta_{2,k-1} & -(\epsilon \vee \Delta_{2,k}) & \Delta_{2,k+1} & \cdots & \Delta_{2,K} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & & \vdots \\ \Delta_{k-1,1} & \Delta_{k-1,2} & \cdots & 0 & -(\epsilon \vee \Delta_{k-1,k}) & \Delta_{k-1,k+1} & \cdots & \Delta_{k-1,K} \\ \epsilon & \epsilon \vee \Delta_{k,2} & \cdots & \epsilon \vee \Delta_{k,k-1} & 0 & \epsilon \vee \Delta_{k,k+1} & \cdots & \epsilon \vee \Delta_{k,K} \\ \Delta_{k+1,1} & \Delta_{k+1,2} & \cdots & \Delta_{k+1,k-1} & -(\epsilon \vee \Delta_{k+1,k}) & 0 & \cdots & \Delta_{k+1,K} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \ddots & \vdots \\ \Delta_{K,1} & \Delta_{K,2} & \cdots & \Delta_{K,k-1} & -(\epsilon \vee \Delta_{K,k}) & \Delta_{K,k+1} & \cdots & 0 \end{pmatrix},$$

where the blue entries indicate the differences with respect to the reference $\mathbf{\Delta}$. In fact, only non-positive entries of row k are modified between $\mathbf{\Delta}$ and $\mathbf{\Delta}^{(k)}$.

Since $\epsilon > 0$, the k-th row $\Delta^{(k)}_{k,\cdot}$ is positive (aside from $\Delta^{(k)}_{k,k}$). Hence, the CW of $\mathbf{\Delta}^{(k)}$ is k. As standard in this line of work, our construction is motivated by the fact that the instance $\mathbf{\Delta}^{(k)}$ is hard to distinguish from the reference gap matrix $\mathbf{\Delta}$.

For $k \geq 2$, denote by $\mathbb{P}^{(k)}$ the distribution of the data when the underlying gap matrix is $\mathbf{\Delta}^{(k)}$. Formally, when the algorithm queries a pair (i, j) with $i < j$, it receives a sample $X_{i,j} \sim \mathcal{B}(\Delta^{(k)}_{i,j} + \frac{1}{2})$, where $\mathcal{B}(p)$ denotes the Bernoulli distribution with parameter p.

Step 2: information-theoretic arguments.

Let $\hat{\imath}$ denote the output of algorithm A, which is assumed to be δ-correct over $\mathbb{D}_{\mathrm{cw}}$. When the gap matrix is $\mathbf{\Delta}^{(k)}$ the CW is k, so δ-correctness implies

$$\forall k \neq i^*,\ \mathbb{P}^{(k)}(\hat{\imath} = k) \geq 1 - \delta \quad \text{and} \quad \mathbb{P}(\hat{\imath} = k) \leqslant \delta\ .$$

By the definition of the total variation distance,

$$\mathrm{TV}(\mathbb{P}, \mathbb{P}^{(k)}) \geq \left|\mathbb{P}(\hat{\imath} = k) - \mathbb{P}^{(k)}(\hat{\imath} = k)\right| \geq 1 - 2\delta\ . \tag{6.91}$$

From (6.91), the Bretagnolle–Huber inequality (Theorem 14.2 in Lattimore and Szepesvári, 2020) yields

$$1 - 2\delta \leqslant \mathrm{TV}(\mathbb{P}, \mathbb{P}^{(k)}) \leqslant 1 - \frac{1}{2}\exp\Big\{-\mathrm{KL}(\mathbb{P}, \mathbb{P}^{(k)})\Big\}\,. \tag{6.92}$$

Step 3: computing the KL divergence and concluding on the budget.

For any unordered pair $\{i, j\}$ with $i \neq j$, denote by $N_{\{i,j\}}$ the number of duels involving either (i, j) or (j, i), that is,

$$N_{\{i,j\}} \coloneqq N_{i,j} + N_{j,i},$$

where

$$N_{i,j} \coloneqq |\{t \in [N_\delta] : (I_t, J_t) = (i, j)\}|,$$

and N_δ is the stopping time of the algorithm.

Using the divergence decomposition lemma (Lemma 15.1 in Lattimore and Szepesvári, 2020) and the fact that the two instances $\mathbf{\Delta}$ and $\mathbf{\Delta}^{(k)}$ differ only on pairs involving arm k, we obtain

$$\begin{aligned}\mathrm{KL}(\mathbb{P}, \mathbb{P}^{(k)}) &= \sum_{1 \leqslant i < j \leqslant K} \mathbb{E}\Big[N_{\{i,j\}}\Big]\ \mathrm{KL}(\mathbb{P}_{i,j}, \mathbb{P}^{(k)}_{i,j}) \\ &= \sum_{\substack{i=1 \\ i \neq k}}^{K} \mathbb{E}\Big[N_{\{k,i\}}\Big]\ \mathrm{KL}(\mathbb{P}_{k,i}, \mathbb{P}^{(k)}_{k,i}).\end{aligned} \tag{6.93}$$

We now upper bound $\mathrm{KL}(\mathbb{P}_{k,i}, \mathbb{P}^{(k)}_{k,i})$. Let $i \in [K]$ with $i \neq k$. If $\Delta_{k,i} > 0$, then by construction

$\mathbb{P}_{k,i} = \mathbb{P}^{(k)}_{k,i}$. Otherwise, if $\Delta_{k,i} < 0$, the corresponding Bernoulli feedback distributions satisfy

$$\mathbb{P}_{k,i} = \mathcal{B}\Big(\tfrac{1}{2} + \Delta_{k,i}\Big)\,, \qquad \mathbb{P}^{(k)}_{k,i} = \mathcal{B}\Big(\tfrac{1}{2} + \epsilon\Big)\,,$$

so that

$$\begin{aligned}
\mathrm{KL}(\mathbb{P}_{k,i}, \mathbb{P}^{(k)}_{k,i}) &= \mathrm{kl}\Big(\Delta_{k,i} + \tfrac{1}{2},\ \epsilon + \tfrac{1}{2}\Big) \\
&\leqslant \frac{\big(\Delta_{k,i} + \frac{1}{2} - (\epsilon + \frac{1}{2})\big)^2}{(-\epsilon + \frac{1}{2})(\epsilon + \frac{1}{2})} \qquad (6.94) \\
&\leqslant \frac{(\epsilon - \Delta_{k,(1)})^2}{(-\epsilon + \frac{1}{2})(\epsilon + \frac{1}{2})}\ . \qquad (6.95)
\end{aligned}$$

Here, (6.94) follows from the standard upper bound $\mathrm{kl}(p,q) \leqslant (p-q)^2/[q(1-q)]$ for $p, q \in (0,1)$, while (6.95) uses that $\Delta^2_{k,i} \leqslant \Delta^2_{k,(1)}$ by definition of $\Delta_{k,(1)}$ as the smallest negative entry in row k. Similarly, if $\Delta_{k,j} = 0$,

$$\mathbb{P}_{k,j} = \mathcal{B}\Big(\tfrac{1}{2}\Big)\,, \qquad \mathbb{P}^{(k)}_{k,j} = \mathcal{B}\Big(\tfrac{1}{2} + \epsilon\Big)\,,$$

so that the same computation gives

$$\mathrm{KL}(\mathbb{P}_{k,j}, \mathbb{P}^{(k)}_{k,j}) \leqslant \frac{\epsilon^2}{(-\epsilon + \frac{1}{2})(\epsilon + \frac{1}{2})}\ ,$$

which vanishes to 0 with $\epsilon \to 0$.

Combining these bounds with (6.93), and taking the limit $\epsilon \to 0$, we obtain

$$\mathrm{KL}(\mathbb{P}, \mathbb{P}^{(k)}) \leqslant 4 \left(\sum_{\substack{i=1 \\ i \neq k}}^{K} \mathbb{1}_{\{\Delta_{k,i} < 0\}}\, \mathbb{E}\Big[N_{\{k,i\}}\Big] \right) \Delta^2_{k,(1)}.$$

Using the Bretagnolle–Huber inequality (6.92) from Step 3 , we then get

$$\sum_{\substack{i=1 \\ i \neq k}}^{K} \mathbb{1}_{\{\Delta_{k,i} < 0\}}\, \mathbb{E}\Big[N_{\{k,i\}}\Big] \geq \frac{1}{4\,\Delta^2_{k,(1)}} \log \frac{1}{4\delta}\ .$$

Summing over $k \geq 2$, we conclude that the total number of queries N_δ satisfies

$$\begin{aligned}
\mathbb{E}\left[N_\delta\right] &\geq \sum_{k \neq 1}^{K} \left(\sum_{\substack{i=1 \\ i \neq k}}^{K} \mathbb{1}_{\{\Delta_{k,i} < 0\}}\, \mathbb{E}\Big[N_{\{k,i\}}\Big] \right) \\
&\geq \frac{1}{4} \sum_{k=2}^{K} \frac{1}{\Delta^2_{k,(1)}} \log \frac{1}{4\delta}\ ,
\end{aligned}$$

which is exactly the claimed lower bound (6.89). □

6.D.2.2 Bound in quantile in Proposition 6.4.1

In this paragraph, we prove the quantile bound from Proposition 6.4.1, namely

$$\mathbb{P}_{\boldsymbol{\Delta},A}\left(N_\delta \geqslant \frac{1}{3}\sum_{i\neq i^*}\frac{\left(\frac{1}{6\delta}\right)}{\Delta_{i,(1)}^2}\right) \geqslant \delta\ . \tag{6.96}$$

Sketch of proof. Even though expectation lower bounds are naturally weaker than quantile bounds, deducing a quantile lower bound from its expectation counterpart is nontrivial. Still, the arguments largely mirror the expectation proof (paragraph 6.D.2.1), we follow the same three-step roadmap. (i) Reference instance and alternative instances: we fix $\boldsymbol{\Delta} \in \mathbb{D}_{\mathrm{cw}}$ with CW i^* and, for each $k \neq i^*$, construct $\boldsymbol{\Delta}^{(k)}$ by zeroing the negative entries in row/column k. In particular, $\boldsymbol{\Delta}^{(k)}$ has two nonnegative rows (those of 1 and k) and therefore does not belong to $\mathbb{D}_{\mathrm{cw}}$. (ii) Separating event and total variation: we now use the stopping rule instead of the recommendation and consider the event $B = \{N_\delta > Q\}$, where Q is the $(1-\delta)$-quantile of N_δ under $\boldsymbol{\Delta}$. This event has small probability under $\mathbb{P}$, but (after a continuity argument using perturbations of $\boldsymbol{\Delta}^{(k)}$) it has large probability under $\mathbb{P}^{(k)}$, since A cannot quickly decide between $\hat{\imath} = 1$ and $\hat{\imath} = k$. (iii) KL decomposition: as before, we decompose the KL along pairs and use that $\boldsymbol{\Delta}$ and $\boldsymbol{\Delta}^{(k)}$ differ only on duels involving k. The only extra ingredient is that, since B depends only on the first Q observations, we introduce a truncated algorithm $\tilde{A}$ that stops at time Q, apply the KL decomposition to $\tilde{A}$, and then reinterpret the resulting inequality as a lower bound on Q, i.e., on the $(1-\delta)$-quantile of N_δ.

Proof. Let A be any δ-correct algorithm on $\mathbb{D}_{\mathrm{cw}}$.

Step 1: Reference and alternative instances.

As for the expectation bound, we fix any $\boldsymbol{\Delta}$ as the reference instance, with $i^* = 1$. Denote $\mathbb{P}_A$ for the probability induced by the interaction between A and $\boldsymbol{\Delta}$. Then, for each suboptimal arm $k \in \{2, \dots, K\}$, construct $\boldsymbol{\Delta}^{(k)}$ by setting to zero all entries (k, j) with $\Delta_{k,j} < 0$, that is, $\boldsymbol{\Delta}^{(k)}$ is defined by Equation (6.90) with $\epsilon = 0$.

The matrices $\boldsymbol{\Delta}$ and $\boldsymbol{\Delta}^{(k)}$ differ only in row/column k. By construction, rows 1 and k of $\boldsymbol{\Delta}^{(k)}$ contain only nonnegative entries, then $\boldsymbol{\Delta}^{(k)}$ contains two weak CW, and 1 and k are tied. In particular, $\boldsymbol{\Delta}^{(k)} \notin \mathbb{D}_{\mathrm{cw}}$. Denote $\mathbb{P}_A^{(k)}$ the distribution induced by A interacting with $\boldsymbol{\Delta}^{(k)}$.

Step 2: bound in total variation and reduction to fixed budget.

Consider the recommendation rule $\hat{\imath}$ and the budget N_δ of A. Define Q as the $(1-\delta)$-quantile of the budget under $\mathbb{P}$:

$$Q = \inf\{x > 0 \text{ s.t. } \mathbb{P}(N_\delta \geqslant x) \leqslant \delta\}. \tag{6.97}$$

Consider the event $B := \{N_\delta \leq \chi\}$ where the budget is smaller than χ.

We define a truncated version $\tilde{A}$ of A with budget at most χ as follows: run A for $t = 1, \dots, \chi$; if A stops before time χ, return $\tilde{\imath} = \hat{\imath}$; else stop at time χ and return $\tilde{\imath} = 0$. Let $\tilde{N}_\delta$, $\tilde{\imath}$ be $\tilde{A}$'s

budget/recommendation, $\mathbb{P}_{\tilde{A}}$ (resp. $\mathbb{P}_{\tilde{A}}^{(k)}$) its law under $\boldsymbol{\Delta}$ (resp. $\boldsymbol{\Delta}^{(k)}$). By construction,

$$B = \{\tilde{\imath} \neq 0\} \ ,$$

so this event is measurable with respect to the observations of algorithm $\tilde{A}$. Moreover, it has the same probability under A and $\tilde{A}$.
We now lower bound the total variation distance between $\mathbb{P}_{\tilde{A}}$ and $\mathbb{P}_{\tilde{A}}^{(k)}$.

By the definition of Q in (6.97), we have

$$\mathbb{P}_{\tilde{A}}(\tilde{\imath} = 0) = \mathbb{P}_A(N_\delta > Q) \ \leqslant \ \delta. \tag{6.98}$$

Under $\boldsymbol{\Delta}^{(k)}$, the instance does not belong to $\mathbb{D}_{\mathrm{cw}}$, so we cannot directly invoke δ-correctness. We therefore approximate $\boldsymbol{\Delta}^{(k)}$ by nearby instances in $\mathbb{D}_{\mathrm{cw}}$. More precisely, define $\boldsymbol{\Delta}^{(k,\epsilon)}$ as (6.90), i.e., by lifting all zero entries in row k of $\boldsymbol{\Delta}^{(k)}$ to $\epsilon > 0$ (and adjusting the k-th column to preserve symmetry). For $0 < \epsilon \leqslant 1/4$, the matrix $\boldsymbol{\Delta}^{(k,\epsilon)}$ lies in $\mathbb{D}_{\mathrm{cw}}$ and admits k as its CW. Similarly, define $\boldsymbol{\Delta}^{(k,-\epsilon)}$ by subtracting ϵ to all zero entries int the k-th row of $\boldsymbol{\Delta}^{(k)}$ (and adjusting the k-th column to preserve symmetry); so that $\boldsymbol{\Delta}^{(k,-\epsilon)} \in \mathbb{D}_{\mathrm{cw}}$ and admits 1 as its CW.

Let $\mathbb{P}_A^{(k,\epsilon)}$ and $\mathbb{P}_A^{(k,-\epsilon)}$ denote the laws of A under $\boldsymbol{\Delta}^{(k,\epsilon)}$ and $\boldsymbol{\Delta}^{(k,-\epsilon)}$, respectively. Since A is δ-correct on $\mathbb{D}_{\mathrm{cw}}$, we have

$$\mathbb{P}_A^{(k,\epsilon)}(\hat{\imath} \neq k) \leqslant \delta, \qquad \mathbb{P}_A^{(k,-\epsilon)}(\hat{\imath} \neq 1) \leqslant \delta \ . \tag{6.99}$$

Moreover, $\mathbb{P}_A^{(k,\epsilon)}$ converges in total variation to $\mathbb{P}_A^{(k)}$ as $\epsilon \to 0$. Using these facts and letting $\epsilon \to 0$, we obtain

$$\begin{aligned}
\mathbb{P}_A^{(k)}(N_\delta \leqslant \chi) &= \mathbb{P}_A^{(k)}(\hat{\imath} = 1, N_\delta \leqslant \chi) + \mathbb{P}_A^{(k)}(\hat{\imath} \neq 1, N_\delta \leqslant \chi) \\
&= \lim_{\epsilon \to 0} \left[\mathbb{P}_A^{(k,\epsilon)}(\hat{\imath} = 1, N_\delta \leqslant \chi) + \mathbb{P}_A^{(k,-\epsilon)}(\hat{\imath} \neq 1, N_\delta \leqslant \chi) \right] \\
&\leqslant 2\delta \ ,
\end{aligned} \tag{6.100}$$

where the last inequality follows from (6.99).

Since $\tilde{A}$ and A coincide up to time Q, we also have $\mathbb{P}_{\tilde{A}}^{(k)}(\tilde{\imath} \neq 0) = \mathbb{P}_A^{(k)}(N_\delta \leqslant Q)$, so (6.100) implies

$$\mathbb{P}_{\tilde{A}}^{(k)}(\tilde{\imath} \neq 0) \leqslant 2\delta.$$

combining (6.98) with this inequality, and writing $B = \{\tilde{\imath} = 0\}$, we obtain

$$\mathrm{TV}(\mathbb{P}_{\tilde{A}}, \mathbb{P}_{\tilde{A}}^{(k)}) \geqslant \mathbb{P}_{\tilde{A}}^{(k)}(B^c) - \mathbb{P}_{\tilde{A}}(B) \ \geqslant 1 - 3\delta \ . \tag{6.101}$$

Remark 6.D.1. Intuitively, one may think of $\tilde{A}$ as a fixed-budget algorithm with budget Q. This can be viewed as a reduction: from any δ-correct algorithm A that enjoys a high-probability control on its budget (namely, $\mathbb{P}(N_\delta \leqslant Q) \geq 1 - \delta$), we construct a fixed-budget algorithm $\tilde{A}$ with budget Q that inherits the same distinguishing power between the reference instance and its

perturbations.

Step 3: computing the KL divergence.

By the Bretagnolle–Huber inequality (see, e.g., Lattimore and Szepesvári, 2020), we have

$$1 - 3\delta \leqslant \mathrm{TV}(\mathbb{P}_{\tilde{A}}, \mathbb{P}_{\tilde{A}}^{(k)}) \leqslant 1 - \frac{1}{2}\exp\Big\{-\mathrm{KL}(\mathbb{P}_{\tilde{A}}, \mathbb{P}_{\tilde{A}}^{(k)})\Big\} \ .$$

In particular,

$$\mathrm{KL}(\mathbb{P}_{\tilde{A}}, \mathbb{P}_{\tilde{A}}^{(k)}) \ \geqslant \ \log\left(\frac{1}{6\delta}\right). \tag{6.102}$$

We now decompose $\mathrm{KL}(\mathbb{P}_{\tilde{A}}, \mathbb{P}_{\tilde{A}}^{(k)})$. For $i \neq j$ in $[K]$, recall that $N_{i,j}$ denotes the number of duels (i,j), while $N_{\{i,j\}} = N_{i,j} + N_{j,i}$ is the number of duels with unordered pair $\{i,j\}$. By the standard KL decomposition for adaptive procedures (Lattimore and Szepesvári, 2020),

$$\begin{aligned}\mathrm{KL}(\mathbb{P}_{\tilde{A}}, \mathbb{P}_{\tilde{A}}^{(k)}) &= \sum_{i \neq j} \mathbb{E}_{\tilde{A}}[N_{i,j}]\, \mathrm{KL}(\mathbb{P}_{i,j}, \mathbb{P}_{i,j}^{(k)}) \\ &= \sum_{j:\Delta_{k,j}<0} \mathbb{E}_{\tilde{A}}[N_{\{k,j\}}]\, \mathrm{KL}(\mathbb{P}_{k,j}, \mathbb{P}_{k,j}^{(k)}) \ , \end{aligned} \tag{6.103}$$

since $\boldsymbol{\Delta}$ and $\boldsymbol{\Delta}^{(k)}$ differ only on duels $\{k,j\}$ with $\Delta_{k,j} < 0$.

For j with $\Delta_{k,j} < 0$, $\mathbb{P}_{k,j}^{(k)} = \mathcal{B}(1/2)$, $\mathbb{P}_{k,j} = \mathcal{B}(1/2 + \Delta_{k,j})$ with $\Delta_{k,j} \in [-1/4, 0]$. Thus,

$$\begin{aligned}\mathrm{KL}(\mathbb{P}_{k,j}, \mathbb{P}_{k,j}^{(k)}) &= \frac{1}{2}\log\left(\frac{\frac{1}{2}}{\frac{1}{2} + \Delta_{k,j}}\right) + \frac{1}{2}\log\left(\frac{\frac{1}{2}}{\frac{1}{2} - \Delta_{k,j}}\right) \\ &= -\frac{1}{2}\log(1 - 4\Delta_{k,j}^2) \ \leqslant \ 8\log\left(\frac{4}{3}\right)\Delta_{k,j}^2 \ , \end{aligned} \tag{6.104}$$

where (6.104) follows from $\sup_{x \in [0,1/4]} \frac{-\log(1-x)}{x} \leqslant 4\log\left(\frac{4}{3}\right)$, applied with $x = 4\Delta_{k,j}^2 \in [0, 1/4]$.

Plugging these bounds into (6.103), we obtain

$$\begin{aligned}\mathrm{KL}(\mathbb{P}_{\tilde{A}}, \mathbb{P}_{\tilde{A}}^{(k)}) &\leqslant 8\log\left(\frac{4}{3}\right) \sum_{j:\Delta_{k,j}<0} \mathbb{E}_{\tilde{A}}[N_{\{k,j\}}]\, \Delta_{k,j}^2 \\ &\leqslant 8\log\left(\frac{4}{3}\right) \left(\sum_{j:\Delta_{k,j}<0} \mathbb{E}_{\tilde{A}}[N_{\{k,j\}}]\right) \Delta_{k,(1)}^2 \ , \end{aligned} \tag{6.105}$$

since $\Delta_{k,j}^2 \leqslant \Delta_{k,(1)}^2$ whenever $\Delta_{k,j} < 0$. Combining (6.102)–(6.105), we obtain

$$\frac{1}{8\log(4/3)} \frac{1}{\Delta_{k,(1)}^2} \log\left(\frac{1}{6\delta}\right) \leqslant \sum_{j:\Delta_{k,j}<0} \mathbb{E}_{\tilde{A}}[N_{\{k,j\}}] \ . \tag{6.106}$$

Since $\tilde{A}$ uses at most χ duels, we have almost surely, under $\mathbb{P}_{\tilde{A}}$,

$$\sum_{i \neq i^*} \sum_{j:\Delta_{k,j}<0} N_{\{k,j\}} \;\leqslant\; \tilde{N}_\delta \leqslant \chi \;,$$

in particular, the same bound also holds in expectation $\mathbb{E}_{\tilde{A}}$.

Summing in (6.106) over $k \neq i^*$ yields

$$\frac{1}{8\log(4/3)} \sum_{k \neq i^*} \frac{1}{\Delta_{k,(1)}^2} \log\left(\frac{1}{6\delta}\right) \leqslant \sum_{k \neq i^*} \sum_{j:\Delta_{k,j}<0} \mathbb{E}_{\tilde{A}}[N_{\{k,j\}}] \leqslant \chi \;.$$

Using the numerical bound $8\log(4/3) < 3$ and the definition of Q as the $(1-\delta)$-quantile of N_δ then gives the high-probability lower bound (6.96).

□

6.D.3 Proof of Theorem 6.4.2

In this subsection, we prove the high-probability lower bound from Theorem 6.4.2. We start with a sketch of proof.

Proof Sketch of (6.11)**.** The key idea is to reduce CW identification to multiple active signal detection problems (Castro, 2014): for each $i \neq i^*$, we need to certify that the row $\Delta_{k,\cdot}$ has at least a negative entry, this with a probability of error at most δ. Along the way, we have to improve state-of-the-art lower bounds for such detection problems.

Consider any δ-correct algorithm A. We introduce a collection $\mathbf{\Delta}^{(\pi)}$ of gap matrices that differ from $\mathbf{\Delta}$ as we permute, on each row $i \neq i^*$, the position of the negative entries by some collection $\pi = (\pi_i)_{i \neq i^*}$ of permutations while preserving the skew symmetry. Then, we fix a specific arm $i \neq i^*$ and construct the gap matrices $\mathbf{\Delta}^{(\pi,i)}$, by setting to 0 the negative entries of the i-th row of $\mathbf{\Delta}^{(\pi)}$ while preserving skew symmetry. As $\mathbf{\Delta}^{(\pi,i)}$ contains two rows with non-negative entries, one easily deduces that the budget of A under $\mathbf{\Delta}^{(\pi,i)}$ is arbitrarily large with probability $1-\delta$. In contrast, under $\mathbf{\Delta}$, A finishes before χ—the $(1-\delta)$-quantile of N_δ under $\mathbb{P}_{\mathbf{\Delta},A}$. Hence, we reduce A to an active testing problem with budget χ for any unknown gap matrix $\tilde{\mathbf{\Delta}}$ of the hypotheses

$$H_0^{(i)} : \tilde{\mathbf{\Delta}} = \mathbf{\Delta}^{(\pi,i)} \text{ for some perm. } \pi \quad \text{vs.} \quad H_1^{(i)} : \tilde{\mathbf{\Delta}} = \mathbf{\Delta}^{(\pi)} \text{ for some perm. } \pi \;.$$

Since the permutation π is unknown to the learner, this allows to improve over (6.9) by accounting for the fact that the algorithm must explore all possible positions of the negative entries in row i.

Then, by a convexity argument, we deduce $\chi \gtrsim \frac{K_{i;<0}}{\|\Delta_i^-\|^2}\log(1/\delta)$. Optimizing over $i \neq i^*$ yields the first part of the bound $\chi \gtrsim \max_{i \neq i^*} \frac{K_{i;<0}}{\|\Delta_i^-\|^2}\log(1/\delta)$.

The second term $\sum_{i \neq i^*} \frac{K_{i;<0}}{\|\Delta_i^-\|^2}$ interprets as the total cost for testing all hypotheses $H_0^{(i)}$ against $H_1^{(i)}$ with a constant error probability. We develop new arguments for this multiple-hypotheses problem. Unlike the involved technique in Simchowitz et al. (2017), we reduce w.l.o.g. to the case

where all tests have similar complexity $\frac{\|\Delta_i^-\|^2}{K_{i;<0}}$ and we write β^2 for this common value. Then, we build upon the symmetry of our problem to reduce to the case where each row receives the same sampling effort χ/K. Relying again on lower bounds on active signal detection, we get $\chi/K \gtrsim \beta^{-2}$, which leads to the desired result. We believe that these arguments can generalize to other multiple active testing problems. The full proof is given in Appendix 6.D.3.

We first introduce additional notation and state a more precise version of the result, in particular, we consider here the case where $\mathbf{\Delta}$ might contains some ties.

Notation and precise formulation Consider a matrix $\mathbf{\Delta}$ with entries in $[-\frac{1}{4}, \frac{1}{4}]$ that admits a unique CW $i^*(\mathbf{\Delta})$, that is, $\mathbf{\Delta} \in \mathbb{D}_{\mathrm{cw}}$. Without loss of generality, we assume that $i^*(\mathbf{\Delta}) = 1$, and we keep this matrix fixed throughout this section. We now define a family of environments obtained from $\mathbf{\Delta}$ by permuting its entries in a specific way. This will lead to a more precise formulation of Theorem 6.4.2.

Fix an antisymmetric matrix $\mathbf{\Sigma} = (\sigma_{i,j})_{1 \leqslant i,j \leqslant K}$ defined, for any $1 \leqslant i \neq j \leqslant K$, by

$$\sigma_{i,j} = \begin{cases} \mathrm{sign}(\Delta_{i,j}) \ , & \text{if } \Delta_{i,j} \neq 0 \ , \\ f_{\mathrm{tb}}(i,j) \ , & \text{if } \Delta_{i,j} = 0 \ , \end{cases} \tag{6.107}$$

where $f_{\mathrm{tb}} : [K]^2 \to \{-1, 0, 1\}$ is an antisymmetric function used as a tie-breaking convention. In other words, $\mathbf{\Sigma}$ records the sign pattern of the gap matrix $\mathbf{\Delta}$, with a fixed rule when $\Delta_{i,j} = 0$. Note that $\mathbf{\Sigma}$ is antisymmetric, since $\mathbf{\Delta}$ is antisymmetric and f_{tb} is antisymmetric by assumption. For now, we do not specify how we fix this convention, as the precise choice will arise naturally at the very end of our proofs.

For a suboptimal arm $i \in \{2, \dots, K\}$, denote by Σ_i^- the set of arms that beat i (with ties broken according to f_{tb}),

$$\Sigma_i^- \coloneqq \{j \in [K] \setminus \{i\} : \sigma_{i,j} = -1\} \ .$$

For any $i \in \{2, \dots, K\}$, define $\Pi_i(\mathbf{\Sigma})$ as the set of permutations of Σ_i^-:

$$\Pi_i(\mathbf{\Sigma}) \coloneqq \{\pi_i : \Sigma_i^- \to \Sigma_i^- \,|\, \pi_i \text{ is a bijection}\} \ ,$$

and set

$$\Pi(\mathbf{\Sigma}) \coloneqq \Pi_2(\mathbf{\Sigma}) \times \cdots \times \Pi_K(\mathbf{\Sigma}) \ . \tag{6.108}$$

Fix a permutation $\pi \coloneqq (\pi_i)_{i=2}^K \in \Pi(\mathbf{\Sigma})$. For each row $i \neq 1$, we permute the entries of $\mathbf{\Delta}$ indexed by Σ_i^- according to π_i. To preserve antisymmetry, we apply the same permutation π_i to the corresponding entries in column i. We thus define the matrix $\mathbf{\Delta}^{(\pi)}$ as follows: for all $(i,j) \in [K]^2$,

$$\Delta_{i,j}^{(\pi)} \coloneqq \begin{cases} \Delta_{i,\pi_i(j)}, & \text{if } \sigma_{i,j} = -1 \ , \\ -\Delta_{j,\pi_j(i)}, & \text{if } \sigma_{i,j} = 1 \ , \\ \Delta_{i,j}, & \text{else} \ , \end{cases} \tag{6.109}$$

the last condition might happen when $\sigma_{i,j} = 0$, which only happens for ties.

The permutations in $\Pi(\mathbf{\Sigma})$ destroy any exploitable ordering structure between arms while preserving, in each row, the multiset of non-positive entries and hence the row-wise hardness parameters $(K_{i;<0}, \|\Delta_i^-\|_2^2)$.

We now list properties that are preserved by the permutation π. In the following lemma, a row j is said to be a weak CW for a given gap matrix if all entries in its row are nonnegative.

Lemma 6.D.2. *For any $\mathbf{\Sigma}$ satisfying* (6.107) *and any $\pi \in \Pi(\mathbf{\Sigma})$, we have*

$$1.\ \mathbf{\Delta}^{(\pi)} = -(\mathbf{\Delta}^{(\pi)})^T \quad \textit{(antisymmetry)}$$

$$2.\ \forall j \neq 1, \Delta^{(\pi)}_{1,j} \geqslant 0 \quad \textit{(row 1 is a weak CW)}$$

$$3.\ \forall i \neq 1,\ K_{i,<0}(\mathbf{\Delta}) = K_{i,<0}(\mathbf{\Delta}^{(\pi)}) \text{ and } K_{i,\leqslant 0}(\mathbf{\Delta}) = K_{i,\leqslant 0}(\mathbf{\Delta}^{(\pi)}) \quad \textit{(same sign structure)}$$

$$4.\ \forall i \neq 1, \forall s \leqslant K_{i,\leqslant 0},\ \Delta^{(\pi)}_{i,(s)} = \Delta_{i,(s)} \quad \textit{(same ordered nonpositive entries)}$$

In particular, if $\mathbf{\Delta}$ has no ties (that is, $\forall i \neq j$, $\Delta_{i,j} \neq 0$), then $\mathbf{\Delta}^{(\pi)} \in \mathbb{D}_{\mathrm{cw}}$. Moreover, for any $\boldsymbol{s}$, $H_{explore}(\boldsymbol{s}, \delta)$ and $H_{certify}(\boldsymbol{s}, \delta)$ remains unchained under any permutation $\pi \in \Pi(\mathbf{\Sigma})$.

Proof of Lemma 6.D.2. 1. Anti-symmetry Let $(i, j) \in [K]^2$ with $i \neq j$.
Assume that $(\sigma_{i,j} = -1)$ which is equivalent to $j \in \Sigma_i^-$. By the first line in Equation (6.109), one has $\Delta^{(\pi)}_{i,j} = \Delta^{(\pi)}_{i,\pi_i(j)}$. Then, by the second line applied with (j, i) $(\sigma_{j,i} = -1)$, one has $\Delta^{(\pi)}_{j,i} = -\Delta_{i,\pi_i(j)} = -\Delta^{(\pi)}_{i,j}$.
The case $\sigma_{i,j} = 1$ is treated similarly.
For $(\sigma_{i,j} = 0)$, one also have $\sigma_{j,i} = 0$ and $\Delta^{\pi}_{i,j} = \Delta_{i,j} = -\Delta_{j,i} = -\Delta^{\pi}_{j,i}$. It proves the anti-symmetry of $\mathbf{\Delta}^{(\pi)}$.

2. CW row By definition, $i^* \in \Sigma_i^-$ for any $i \neq i^*$ (and $i^* = 1$ by assumption). In particular, $\Delta^{(\pi)}_{1,i} = \Delta_{\pi_i(1),i} \geqslant 0$, the non-negativity comes from the fact that $\pi_i(1) \in \Sigma_i^-$, that is $\pi_i(1)$ also beats i (or is even with i in the case where Δ contains ties).

3. Sign structure. For every non-CW arm $i \neq 1$, the set of indices Σ_i^- contains only entries j such that $\Delta_{i,j} \leqslant 0$, and (6.109) only permutes the entries in that set. Hence the multiset of nonpositive entries in row i is preserved, and so are the counts $K_{i,<0}$ and $K_{i,\leqslant 0}$.

4. Order statistics. Since the multiset of nonpositive entries in row i is preserved up to permutation, the ordered sequence $(\Delta_{i,(s)})_{s \leqslant K_{i,\leqslant 0}}$ is unchanged, which gives $\Delta^{(\pi)}_{i,(s)} = \Delta_{i,(s)}$ for all $s \leqslant K_{i,\leqslant 0}$. In particular, for any vector $\boldsymbol{s}$ with $s_i \leqslant K_{i;<0}$ (a condition independent of π), the gaps $\Delta^{(\pi)}_{i,(s_i)}$ coincide with $\Delta_{i,(s_i)}$, so that $H_{\mathrm{explore}}(\boldsymbol{s}, \delta)$ and $H_{\mathrm{certify}}(\boldsymbol{s}, \delta)$ remain unchanged under any $\pi \in \Pi(\mathbf{\Sigma})$. □

Now we are ready to state Theorem 6.D.3, which directly implies Theorem 6.4.2 and provides a more constructive formulation.

Theorem 6.D.3. *Let A be a δ-correct algorithm over the class $\mathbb{D}_{\mathrm{cw}}$, with $\delta \leqslant 1/12$. Let $\mathbf{\Delta} \in \mathbb{D}_{\mathrm{cw}}$ be such that $i^*(\mathbf{\Delta}) = 1$.*

Define χ as the smallest positive number such that, for any sign convention $\mathbf{\Sigma}$ (satisfying (6.107)) and any $\pi \in \Pi(\mathbf{\Sigma})$, one has $\mathbb{P}_{\mathbf{\Delta}^{(\pi)},A}(N_\delta > \chi) \leqslant \delta$. Equivalently,

$$\chi \coloneqq \inf\left\{x > 0 : \sup_{\mathbf{\Sigma}} \sup_{\pi\in\Pi(\mathbf{\Sigma})} \mathbb{P}_{\mathbf{\Delta}^{(\pi)},A}(N_\delta > x) \leqslant \delta\right\} \ . \tag{6.110}$$

In other words, χ is a uniform $(1-\delta)$-quantile of the budget, taken in the worst case over all admissible sign conventions and permutations.

Then, χ satisfies

$$\chi \geqslant \frac{1}{8\log(4/3)} \max_{i=2}^{K} \frac{K_{i;\leqslant 0}}{\|\Delta_i^-\|^2} \log\left(\frac{1}{6\delta}\right), \tag{6.111}$$

$$\chi \geqslant \frac{1}{128\log(4/3)} \cdot \frac{1}{\log(2K)} \sum_{i=2}^{K} \frac{K_{i;\leqslant 0}}{\|\Delta_i^-\|^2} \ . \tag{6.112}$$

We postpone the proof of Theorem 6.D.3 to the following Subsection 6.D.4, which is divided between the proof of Equation (6.111) and Equation (6.112). We first explain how this result directly implies Theorem 6.4.2.

Proof of Theorem 6.4.2. Let A be a δ-correct algorithm over the class $\mathbb{D}_{\mathrm{cw}}$, with $\delta \leqslant 1/12$. Fix any matrix $\mathbf{\Delta} \in \mathbb{D}_{\mathrm{cw}}$ with CW $i^* = 1$. Consider χ as defined in Equation (6.110).

Combining the numerical bounds $8\log(4/3) < 3$ and $128\log(4/3) < 37$ with the definition of χ and the lower bounds (6.111) and (6.112), we obtain that there exist a sign convention $\mathbf{\Sigma}$ and a permutation $\pi \in \Pi(\mathbf{\Sigma})$ such that

$$\mathbb{P}_{\mathbf{\Delta}^{(\pi)},A}\left(N_\delta \geqslant \frac{1}{3} \cdot \max_{i\neq i^*} \frac{K_{i;\leqslant 0}}{\|\Delta_i^-\|^2} \log\left(\frac{1}{6\delta}\right) \vee \frac{1}{37\log(2K)} \sum_{i\neq i^*} \frac{K_{i;\leqslant 0}}{\|\Delta_i^-\|^2}\right) \geqslant \delta \ .$$

By Lemma 6.D.2, if $\mathbf{\Delta}$ contains no ties, then the matrix $\mathbf{\Delta}^\pi$ admits a Condorcet winner and verifies all properties required of the matrix $\tilde{\mathbf{\Delta}}$ from Theorem 6.4.2. This proves Theorem 6.4.2 in the no-ties case. □

Remark 6.D.4. Observe that, by the second point of Lemma 6.D.2, it is possible that $\mathbf{\Delta}^{(\pi)}$ admits no CW, when $\mathbf{\Delta}$ admit some ties. Yet it still admits a weak Condorcet winner, in the same that $\Delta^\pi_{1,\cdot}$ admits only non-negative entries, while all the other rows admit at least one negative entry. Then, one can construct a matrix $\mathbf{\Delta}^{\pi,\epsilon}$ as close as we need to $\mathbf{\Delta}^{(\pi)}$ and such that $\mathbf{\Delta}^{(\pi,\epsilon)}$ admit 1 as CW. We can for instance add a small constant $\epsilon > 0$ to the first row of $\mathbf{\Delta}^{(\pi,\epsilon)}$ [and $-\epsilon$ to the first column]. For ϵ small enough, the given matrix $\mathbf{\Delta}^{(\pi,\epsilon)}$ shares most properties of Δ.

6.D.4 Proof of Theorem 6.D.3

6.D.4.1 Proof of Equation (6.111) in Theorem 6.D.3

Let $\delta \leqslant 1/12$. Let A be any δ-correct algorithm over the entire class $\mathbb{D}_{\mathrm{cw}}$. Fix $\mathbf{\Delta} \in \mathbb{D}_{\mathrm{cw}}$ with CW $i^*(\mathbf{\Delta}) = 1$. Recall the definition

$$\chi \coloneqq \inf\left\{ x > 0 : \sup_{\Sigma} \sup_{\pi \in \Pi(\mathbf{\Sigma})} \mathbb{P}_{\mathbf{\Delta}^{(\pi)}, A}(N_\delta > x) \leqslant \delta \right\} .$$

We prove Equation (6.111), namely

$$\chi \geqslant \frac{1}{8\log(4/3)} \max_{i=2}^{K} \frac{K_{i;\leqslant 0}}{\|\Delta_i^-\|^2} \log\left(\frac{1}{6\delta}\right).$$

Sketch of proof. We follow the three-step roadmap of Section 6.D.1. The arguments are very similar to the proof of the quantile proof of Proposition 6.4.1 in paragraph 6.D.2.2, except that we work with a local class of permuted instances and average over all permutations of $\mathbf{\Delta}$.

(i) Reference and alternative instances. We fix $\mathbf{\Delta} \in \mathbb{D}_{\mathrm{cw}}$ with $i^*(\mathbf{\Delta}) = 1$ and consider the local class of reference $(\Delta^{(\pi)})_{\pi \in \Pi}$ obtained by permuting, within each non-CW row, the positions of its negative entries according to π. For each suboptimal arm k, we construct the alternative instance $\mathbf{\Delta}^{(\pi,k)}$ by lifting to 0 all negative entries in row k, so that rows 1 and k become nonnegative and the instance has two weak CWs (hence lies outside $\mathbb{D}_{\mathrm{cw}}$). (ii) Separating event and total variation. As in Subsection 6.D.2.2, we truncate A at the worst-case $(1-\delta)$-quantile χ to obtain a fixed-budget algorithm $\tilde{A}$, and we use the event $\{N_\delta > \chi\}$ as a separating event. This yields a total-variation lower bound $\mathrm{TV}(\mathbb{P}_{\tilde{A}}^{(\pi)}, \mathbb{P}_{\tilde{A}}^{(\pi,k)}) \geq 1-3\delta$, hence a KL lower bound of order $\log(1/(6\delta))$ for each k. (iii) KL decomposition and extraction of $K_{k;\leqslant 0}/\|\Delta_k^-\|_2^2$. We decompose the KL along pairs and use that $\mathbf{\Delta}^{(\pi)}$ and $\mathbf{\Delta}^{(\pi,k)}$ differ only on duels involving k. Averaging over permutations $\pi_k \in \Pi_k(\mathbf{\Sigma})$ symmetrizes the contribution of all negative entries in row k, and yields a lower bound $\chi \gtrsim K_{k;\leqslant 0}\|\Delta_k^-\|_2^{-2}\log(1/\delta)$. Finally, choosing the row k that maximizes $K_{k;\leqslant 0}/\|\Delta_k^-\|_2^2$ gives (6.111). (iv) Tie-breaking convention. In the last step, we a tie-breaking convention to conclude.

Proof. Step 1: Reference and alternative instances.
For now, fix a tie-breaking convention f_{tb} and a sign matrix $\mathbf{\Sigma}$ as in (6.107). The specific choice of $\mathbf{\Sigma}$ will be made in the final step of the proof.

Fix a permutation $\pi \in \Pi(\mathbf{\Sigma})$ (see Equation (6.108)), and consider the corresponding matrix $\mathbf{\Delta}^{(\pi)}$ defined in Equation (6.109). We denote by $\mathbb{P}_A^{(\pi)}$ the distribution of the observations induced by the interaction between algorithm A and the environment with gap matrix $\mathbf{\Delta}^{(\pi)}$.

Fix a suboptimal arm $k \in \{2, \dots, K\}$. The precise choice of k will be made explicit in the last step of the proof. We construct the gap matrix $\mathbf{\Delta}^{(\pi,k)}$ by setting to zero all entries (k,j) with $j \in \Sigma_k^- \coloneqq \{j \in [K] : \sigma_{k,j} = -1\}$. Recall that, by the definition of $\mathbf{\Sigma}$ (see (6.107)), if $j \in \Sigma_k^-$ then $\Delta_{k,j}^{(\pi)} \leq 0$, that is Σ_k^- contains all arms that beat strictly k, together with some arms that are tied

with k. We define

$$\boldsymbol{\Delta}^{(\pi,k)}_{i,j} := \begin{cases} \Delta^{(\pi)}_{i,j}, & \text{if } i \neq k \text{ and } j \neq k, \\ \Delta^{(\pi)}_{i,j}, & \text{if } (i = k \text{ and } \sigma_{k,j} = 1) \text{ or } (j = k \text{ and } \sigma_{k,i} = 1), \\ 0, & \text{if } (i = k \text{ and } \sigma_{k,j} = -1) \text{ or } (j = k \text{ and } \sigma_{k,i} = -1). \end{cases} \tag{6.113}$$

The matrices $\boldsymbol{\Delta}^{(\pi)}$ and $\boldsymbol{\Delta}^{(\pi,k)}$ differ only in row/column k. Moreover, by construction, rows 1 and k of $\boldsymbol{\Delta}^{(\pi,k)}$ contain only nonnegative entries, so that $\boldsymbol{\Delta}^{(\pi,k)} \notin \mathbb{D}_{\mathrm{cw}}$. We denote by $\mathbb{P}^{(\pi,k)}_A$ the distribution of the observations induced by the interaction between algorithm A and the environment with gap matrix $\boldsymbol{\Delta}^{(\pi,k)}$.

Step 2: information-theoretic arguments.

This step is identical to Step 2 in the proof of the quantile bound (6.96) (Section 6.D.2.2). Consider the event $B := \{N_\delta \leqslant \chi\}$, and define the truncated algorithm $\tilde{A}$ as in that proof: run A up to time χ, returning $\tilde{i} = \hat{\imath}$ if A stops before χ, and $\tilde{i} = 0$ otherwise.

By the definition of χ in (6.110)—a uniform upper bound on the $(1-\delta)$-quantile of N_δ under any $\mathbb{P}^{(\pi)}_A$—we have

$$\mathbb{P}^{(\pi)}_{\tilde{A}}(\tilde{i} = 0) = \mathbb{P}^{(\pi)}_A(N_\delta > \chi) \leqslant \delta \ .$$

Since $\boldsymbol{\Delta}^{(\pi,k)}$ can be approximated by CW instances admitting 1 or k as CW (as in (6.99)–(6.100)), we obtain

$$\mathbb{P}^{(\pi,k)}_{\tilde{A}}(\tilde{i} \neq 0) \leqslant 2\delta \ .$$

Writing $B := \{\tilde{i} = 0\}$, this yields

$$\mathrm{TV}(\mathbb{P}^{(\pi)}_{\tilde{A}}, \mathbb{P}^{(\pi,k)}_{\tilde{A}}) \ \geqslant \ (1 - 2\delta) - \delta \ = \ 1 - 3\delta. \tag{6.114}$$

Remark 6.D.5. The result from Step 2 can be interpreted as a reduction scheme. Consider the signal detection problem of testing $H_0 : \mu = \mathbf{0}$ versus $H_1(\pi_k) : \mu = (\Delta_{k,\pi_k(\ell)})_{\ell \in \Sigma^-_k}$ in a bandit setting. Equation (6.114) shows that, for any permutation π_k, $\tilde{A}$ is 2δ-correct for this signal detection problem, with a budget bounded by χ (independently of the permutation). This reduction is the main novelty of our proof technique. The remaining arguments in Step 3 build upon the literature on active signal detection (Castro, 2014; Saad et al., 2023; Graf et al., 2025).

Step 3: computing the KL divergence.

By the Bretagnolle–Huber inequality (see, e.g., Lattimore and Szepesvári, 2020), the conclusion of Step 2 (6.114) implies

$$\mathrm{KL}(\mathbb{P}^{(\pi,k)}_{\tilde{A}}, \mathbb{P}^{(\pi)}_{\tilde{A}}) \ \geqslant \ \log\left(\frac{1}{6\delta}\right). \tag{6.115}$$

We now compute $\mathrm{KL}(\mathbb{P}^{(\pi,k)}_{\tilde{A}}, \mathbb{P}^{(\pi)}_{\tilde{A}})$. Observe that we take the law under the alternative instance $\boldsymbol{\Delta}^{(\pi,k)}$ in the left side of the divergence: this ensures that the expectation $\mathbb{E}^{(\pi,k)}_{\tilde{A}}[N_{\{k,j\}}]$ appearing in the KL decomposition (below) does not depend on π_k, which will allow us to average over permutations $\pi_k \in \Pi_k(\boldsymbol{\Sigma})$.

By the standard decomposition of the Kullback–Leibler divergence for adaptive procedures (see, e.g., Lattimore and Szepesvári, 2020, Lemma 15.1),

$$\begin{aligned}\mathrm{KL}(\mathbb{P}_{\tilde{A}}^{(\pi,k)}, \mathbb{P}_{\tilde{A}}^{(\pi)}) &= \sum_{i \neq j} \mathbb{E}_{\tilde{A}}^{(\pi,k)}[N_{i,j}] \, \mathrm{KL}(\mathbb{P}_{i,j}^{(\pi,k)}, \mathbb{P}_{i,j}^{(\pi)}) \\ &= \sum_{j \in \Sigma_k^-} \mathbb{E}_{\tilde{A}}^{(\pi,k)}[N_{\{k,j\}}] \, \mathrm{KL}(\mathbb{P}_{k,j}^{(\pi,k)}, \mathbb{P}_{k,j}^{(\pi)}) \ , \end{aligned} \tag{6.116}$$

since the two instances differ only on pairs (k,j) or (j,k) with $j \in \Sigma_k^-$, by construction of $\mathbf{\Delta}^{(\pi,k)}$.

Now fix $j \in \Sigma_k^-$. By the definitions of $\mathbf{\Delta}^{(\pi)}$ and $\mathbf{\Delta}^{(\pi,k)}$ (Equations (6.109) and (6.113)), we have

$$\mathbb{P}_{k,j}^{(\pi,k)} = \mathcal{B}\Big(\tfrac{1}{2}\Big) , \qquad \mathbb{P}_{k,j}^{(\pi)} = \mathcal{B}\Big(\tfrac{1}{2} + \Delta_{k,\pi_k(j)}\Big) ,$$

where $\Delta_{k,\pi_k(j)} \in [-1/4, 0]$. Hence, using the bound on kl from (6.104),

$$\mathrm{KL}(\mathbb{P}_{k,j}^{(\pi,k)}, \mathbb{P}_{k,j}^{(\pi)}) \leqslant 8 \log\left(\frac{4}{3}\right) \Delta_{k,\pi_k(j)}^2 \ ,$$

Plugging these bounds into (6.116), we obtain

$$\mathrm{KL}(\mathbb{P}_{\tilde{A}}^{(\pi,k)}, \mathbb{P}_{\tilde{A}}^{(\pi)}) \leqslant 8 \log\left(\frac{4}{3}\right) \sum_{j \in \Sigma_k^-} \mathbb{E}_{\tilde{A}}^{(\pi,k)}[N_{\{k,j\}}] \, \Delta_{k,\pi_k(j)}^2 \ .$$

A key property of our construction is that $\mathbf{\Delta}^{(\pi,k)}$ does not actually depend on π_k: all entries permuted by π_k in row k of $\mathbf{\Delta}^{(\pi)}$ are set to 0 under $\mathbf{\Delta}^{(\pi,k)}$. Consequently, $\mathbb{E}_{\tilde{A}}^{(\pi,k)}$ does not depend on π_k. Given $\pi_k' \in \Pi_k(\mathbf{\Sigma})$, we denote $\pi' = (\pi_2, \dots, \pi_{k-1}, \pi_k', \pi_{k+1}, \dots)$ where we only change π_k. Averaging the previous inequality over $\pi_k' \in \Pi_k(\mathbf{\Sigma})$ while keeping $(\pi_l)_{l \neq k}$ fixed, we get

$$\begin{aligned}\frac{1}{|\Pi_k(\mathbf{\Sigma})|} \sum_{\pi_k' \in \Pi_k(\mathbf{\Sigma})} \mathrm{KL}(\mathbb{P}_{\tilde{A}}^{(\pi,k)}, \mathbb{P}_{\tilde{A}}^{(\pi')}) &\leqslant 8 \log\left(\frac{4}{3}\right) \frac{1}{|\Pi_k(\mathbf{\Sigma})|} \sum_{\pi_k' \in \Pi_k(\mathbf{\Sigma})} \sum_{j \in \Sigma_k^-} \mathbb{E}_{\tilde{A}}^{(\pi,k)}[N_{\{k,j\}}] \, \Delta_{k,\pi_k'(j)}^2 \\ &= 8 \log\left(\frac{4}{3}\right) \frac{1}{|\Sigma_k^-|} \left(\sum_{j \in \Sigma_k^-} \mathbb{E}_{\tilde{A}}^{(\pi,k)}[N_{\{k,j\}}] \right) \left(\sum_{j \in \Sigma_k^-} \Delta_{k,j}^2 \right) \ , \end{aligned} \tag{6.117}$$

where we used Lemma 6.F.3 to symmetrize over all permutations π_k' of Σ_k^-.

By definition of Σ_k^-, it holds that $\Sigma_k^- \subset \{j \in [K] \setminus \{k\} : \Delta_{k,j} \leqslant 0\}$, so that

$$\sum_{j \in \Sigma_k^-} \Delta_{k,j}^2 = \|\Delta_k^-\|^2 \ .$$

Moreover, by construction, the modified algorithm $\tilde{A}$ has a budget upper bounded by χ, and

$$\sum_{j \in \Sigma_k^-} \mathbb{E}_{\tilde{A}}^{(\pi,k)}[N_{\{k,j\}}] \leqslant \chi \ .$$

From there, we get

$$\frac{1}{|\Pi_k(\mathbf{\Sigma})|} \sum_{\pi'_k \in \Pi_k(\mathbf{\Sigma})} \mathrm{KL}(\mathbb{P}_{\tilde{A}}^{(\pi,k)}, \mathbb{P}_{\tilde{A}}^{(\pi')}) \leqslant 8 \log\left(\frac{4}{3}\right) \frac{\|\Delta_k^-\|^2}{|\Sigma_k^-|} \chi \ . \tag{6.118}$$

Finally, combining (6.118) with the Bretagnolle–Huber bound (6.115), we obtain

$$\chi \geqslant \frac{1}{8 \log(4/3)} \frac{|\Sigma_k^-|}{\|\Delta_k^-\|^2} \log\left(\frac{1}{6\delta}\right) \ . \tag{6.119}$$

Step 4: choice of convention $\mathbf{\Sigma}$ and conclusion.

It remains to choose an appropriate arm k, and a convention f_{tb} in the definition of the sign matrix $\mathbf{\Sigma}$ (see Equation (6.107)). Consider

$$k^* \in \operatorname*{argmax}_{k=2}^{K} \frac{K_{k;\leqslant 0}}{\|\Delta_k^-\|^2} \ ,$$

as a suboptimal arm for which detecting a negative entry in its row is the most costly.

Fix a tie-breaking convention $f_{\mathrm{tb}} : [K]^2 \to \{-1, 0, 1\}$ such that $f_{\mathrm{tb}}(k^*, i) = -1$ for any $i \neq k^*$. For this choice, we have

$$\frac{|\Sigma_{k^*}^-|}{\|\Delta_{k^*}^-\|^2} = \frac{K_{k^*;\leqslant 0}}{\|\Delta_{k^*}^-\|^2} = \max_{i \neq i^*} \frac{K_{i;\leqslant 0}}{\|\Delta_i^-\|^2} \ .$$

Plugging this into (6.119) yields

$$\chi \geqslant \frac{1}{8 \log(4/3)} \max_{i \neq i^*} \frac{K_{i;\leqslant 0}}{\|\Delta_i^-\|^2} \log\left(\frac{1}{6\delta}\right) \ ,$$

which establishes the first inequality (6.111) on χ. □

6.D.4.2 Proof of Equation (6.112) in Theorem 6.D.3

Let $\delta \leqslant 1/12$. Let A be any δ-correct algorithm over the entire class $\mathbb{D}_{\mathrm{cw}}$, and fix $\mathbf{\Delta} \in \mathbb{D}_{\mathrm{cw}}$ with CW $i^*(\mathbf{\Delta}) = 1$. In this subsection, we prove Equation (6.112) from Theorem 6.D.3, using the same instance construction as in the proof of bound (6.111). Recall this bound:

$$\chi \geqslant \frac{1}{64 \log(4/3)} \frac{1}{\log(2K)} \sum_{i=2}^{K} \frac{K_{i;\leqslant 0}}{\|\Delta_i^-\|^2}.$$

Sketch of proof. We follow the three-step roadmap of Section 6.D.1, reusing the instance construction from the proof of (6.111) but with a more refined separating event. The key differences are (ii) a refined event B_k that also controls the number of duels involving arm k, and (iii) the use of Pinsker's inequality and Jensen to average over multiple arms simultaneously.

(i) Reference and alternative instances. As before, we consider the local class $(\mathbf{\Delta}^{(\pi)})_{\pi\in\Pi(\mathbf{\Sigma})}$ and, for each $k \in \{2,\dots,K\}$, the alternative $\mathbf{\Delta}^{(\pi,k)}$. We define the row-wise hardness $\beta_k^2 = \|\Delta_k^-\|^2/|\Sigma_k^-|$, assumed ordered increasingly. Let I be the index that maximizes $(k-1)/\beta_k^2$, corresponding to the worst-case average hardness over the first k rows. (ii) Separating event and total variation. Instead of $\{N_\delta \le \chi\}$, we use the event $B_k = \{N_\delta \leqslant \chi,\, N_{k,\cdot} \leqslant 4\chi/(I-1)\}$, where $N_{k,\cdot}$ counts duels between k and opponents in Σ_k^-. Define a truncation $\tilde{A}_k$ that outputs $\mathbb{1}_{B_k}$, using budget at most $4\chi/(I-1)$ on k. Under $\mathbf{\Delta}^{(\pi,k)}$, $\mathbb{P}(\mathbb{1}_{B_k}=1) \leqslant 2\delta$. Under $\mathbf{\Delta}^{(\pi)}$, we use a pigeonhole argument and the average probability is larger than $3/4-\delta$. This yields averaged TV larger than $1/2$—the key novelty of this proof technique. (iii) KL decomposition and extraction of the sum. By Pinsker's inequality, the averaged TV lower bound implies an averaged KL lower bound. Each KL is upper bounded as previously, and now we average over $k=2,\dots,I$ and use the truncation constraint $N_{k,\cdot} \leqslant 4\chi/(I-1)$ to get $\chi \gtrsim (I-1)/\beta_I^2$. By the definition of I, this gives $\chi \gtrsim \sum_{i=2}^K K_{i;\leqslant 0}/\|\Delta_i^-\|_2^2$, up to logarithmic factors.

Proof. Step 1: construction of instances. Fix a tie-breaking convention f_{tb} and a sign matrix $\mathbf{\Sigma}$ (see (6.107)). Again, they will be chosen in the last step of the proof. Fix a permutation $\pi \in \Pi(\mathbf{\Sigma})$ and use $\mathbf{\Delta}^{(\pi)}$ as the reference matrix (see (6.109)). Define, for each $k \in \{2,\dots,K\}$, the row-wise hardness

$$\beta_k^2(\Sigma) \coloneqq \frac{\|\Delta_k^-\|^2}{|\Sigma_k^-|} \quad . \tag{6.120}$$

Without loss of generality, assume that the arms are ordered so that $\beta_2^2 \leqslant \beta_3^2 \leqslant \cdots \leqslant \beta_K^2$. Define

$$I \coloneqq \operatorname*{argmax}_{k=2,\dots,K} \frac{k-1}{\beta_k^2} \quad , \tag{6.121}$$

the index that captures the worst-case average hardness over the first k rows.

As alternative instances, we consider the family $\{\mathbf{\Delta}^{(\pi,k)}\}_{k=2,\dots,I}$.

Step 2: information-theoretic arguments. In this step, we construct an event under which algorithm A should behave differently depending on whether it interacts with $\mathbf{\Delta}^{(\pi)}$ or $\mathbf{\Delta}^{(\pi,k)}$. To capture the sum lower bound (6.112), we need a more refined event than in the proof of (6.111). For $k \in \{2,\dots,I\}$, define

$$B_k \coloneqq \{N_\delta \leqslant \chi\} \cap \left\{N_{k,\cdot} \leqslant \frac{4\chi}{I-1}\right\}, \tag{6.122}$$

where

$$N_{k,\cdot} \coloneqq \sum_{i\in\Sigma_k^-} N_{\{k,i\}}$$

denotes the total number of duels involving arm k against opponents in Σ_k^-.

Remark 6.D.6. The event B_k is designed as follows. The bound (6.111) shows that the quantity β_k^{-2} characterizes the budget needed to find a negative entry in row k of $\mathbf{\Delta}^{(\pi)}$, uniformly over all permutations π. . Identifying the CW $i^* = 1$ amounts to solving simultaneously $K-1$ such signal detection problems, one per suboptimal row. By the definition of I in (6.121) and Lemma 6.F.1, it is natural to think of the simplified regime where $(\beta_2^{-2}, \dots, \beta_I^{-2})$ are of the same order, so that arms $2, \dots, I$ are equally hard to eliminate. In that case, any reasonable algorithm should allocate its samples roughly uniformly across rows $2, \dots, I$, and the event B_k describes this expected behavior for a δ-correct algorithm A.

We now compute the event B_k with a truncated procedure $\tilde{A}_k$ that uses a total budget at most χ, and at most $4\chi/(I-1)$ comparisons involving arm k against an opponent in Σ_k^-.

Define the following procedure $\tilde{A}_k$. For $t = 1, \dots, \chi$, run algorithm A. If A stops before time χ, compute $N_{k,\cdot}$ and output $\psi_k \coloneqq \mathbb{1}_{\{N_{k,\cdot} \leqslant 4\chi/(I-1)\}}$. Otherwise, stop at time $t = \chi$ and set $\psi_k = 0$. By construction, the binary decision ψ_k computed by $\tilde{A}_k$ satisfies $\psi_k = \mathbb{1}_{B_k}$.

We write $\mathbb{P}_{\tilde{A}_k}^{(\pi)}$ (resp. $\mathbb{P}_{\tilde{A}_k}^{(\pi,k)}$) for the distribution induced by the interaction between algorithm $\tilde{A}_k$ and the environment with gap matrix $\mathbf{\Delta}^{(\pi)}$ (resp. $\mathbf{\Delta}^{(\pi,k)}$). We now lower bound the total variation distance between $\mathbb{P}_{\tilde{A}_k}^{(\pi)}$ and $\mathbb{P}_{\tilde{A}_k}^{(\pi,k)}$ using the event B_k.

First, $B_k \subset \{N_\delta \leqslant \chi\}$. Since A is δ-correct over $\mathbb{D}_{\mathrm{cw}}$ and $\mathbf{\Delta}^{(\pi,k)}$ can be approximated by instances in $\mathbb{D}_{\mathrm{cw}}$ as in the proof of (6.100), we obtain, for any $k \neq 1$,

$$\mathbb{P}_{\tilde{A}_k}^{(\pi,k)}(B_k) \leqslant \mathbb{P}_{\tilde{A}_k}^{(\pi,k)}(N_\delta \leqslant \chi) \leqslant 2\delta \ . \tag{6.123}$$

Next, consider $\mathbb{P}_{\tilde{A}_k}^{(\pi)}$ and the complement B_k^c. We have

$$B_k^c = \{N_\delta > \chi\} \ \cup \ \{N_\delta \leqslant \chi, \ N_{k,\cdot} > 4\chi/(I-1)\}.$$

Since A is δ-correct and $\mathbf{\Delta}^{(\pi)} \in \mathbb{D}_{\mathrm{cw}}$, the definition of χ implies

$$\mathbb{P}_{\tilde{A}_k}^{(\pi)}(N_\delta > \chi) = \mathbb{P}_A^{(\pi)}(N_\delta > \chi) \leqslant \delta \ . \tag{6.124}$$

We now average the second term of B_k^c over $k \in \{2, \dots, I\}$:

$$\frac{1}{I-1} \sum_{k=2}^{I} \mathbb{P}_{\tilde{A}_k}^{(\pi)}\big(N_\delta \leqslant \chi, \ N_{k,\cdot} > 4\chi/(I-1)\big) = \mathbb{E}_A^{(\pi)}\left[\frac{1}{I-1} \sum_{k=2}^{I} \mathbb{1}_{\{N_\delta \leqslant \chi, \ N_{k,\cdot} > 4\chi/(I-1)\}}\right].$$

Since $\sum_{k=2}^{K} N_{k,\cdot} \leqslant N_\delta$, on the event $\{N_\delta \leqslant \chi\}$ at most $(I-1)/4$ indices $k \in \{2, \dots, I\}$ can satisfy $N_{k,\cdot} > 4\chi/(I-1)$. Hence,

$$\frac{1}{I-1} \sum_{k=2}^{I} \mathbb{1}_{\{N_\delta \leqslant \chi, \ N_{k,\cdot} > 4\chi/(I-1)\}} \leqslant \frac{1}{4},$$

which yields

$$\frac{1}{I-1}\sum_{k=2}^{I}\mathbb{P}_{\tilde{A}_k}^{(\pi)}\big(N_\delta \leqslant \chi,\ N_{k,\cdot} > 4\chi/(I-1)\big) \leqslant \frac{1}{4}\ . \tag{6.125}$$

Combining (6.123), (6.124), and (6.125), we obtain

$$\begin{aligned}\frac{1}{I-1}\sum_{k=2}^{I}\mathrm{TV}\big(\mathbb{P}_{\tilde{A}_k}^{(\pi,k)},\mathbb{P}_{\tilde{A}_k}^{(\pi)}\big) &\geqslant \frac{1}{I-1}\sum_{k=2}^{I}\Big(\mathbb{P}_{\tilde{A}_k}^{(\pi,k)}(B_k^c)-\mathbb{P}_{\tilde{A}_k}^{(\pi)}(B_k^c)\Big)\\ &\geqslant 1-2\delta-\frac{1}{I-1}\sum_{k=2}^{I}\mathbb{P}_{\tilde{A}_k}^{(\pi)}(B_k^c)\\ &\geqslant (1-2\delta)-\left(\delta+\frac{1}{4}\right)\\ &\geqslant \frac{1}{2}\ ,\end{aligned}$$

where the last inequality uses the assumption $\delta \leqslant 1/12$. Finally, we apply a data-processing inequality. In this regime which does not depend on δ, we use Pinsker's inequality which implies that

$$\frac{1}{2} \leqslant \frac{1}{I-1}\sum_{k=2}^{I}\mathrm{TV}\big(\mathbb{P}_{\tilde{A}_k}^{(\pi,k)},\mathbb{P}_{\tilde{A}_k}^{(\pi)}\big) \leqslant \frac{1}{I-1}\sum_{k=2}^{I}\sqrt{\frac{1}{2}\,\mathrm{KL}\big(\mathbb{P}_{\tilde{A}_k}^{(\pi,k)},\mathbb{P}_{\tilde{A}_k}^{(\pi)}\big)}\ . \tag{6.126}$$

Remark 6.D.7. We can again interpret this result as a reduction argument. Averaging (6.123), (6.124), and (6.125) over $\pi \sim \mathcal{U}\mathrm{nif}(\Pi(\mathbf{\Sigma}))$, we obtain that there exists some $k \in \{2,\dots,I\}$ (independent of π) such that $\tilde{A}_k$ is 1/2-correct, with budget at most $4\chi/(I-1)$, for the active signal detection problem

$$H_0 : \mu = \mathbf{0} \quad \text{vs} \quad H_1 : \mu = \big(\Delta_{k,\pi_k(\ell)}\big)_{\ell\in\Sigma_k^-}, \quad \pi_k \sim \mathcal{U}\mathrm{nif}(\Pi_k(\mathbf{\Sigma}))\,.$$

Step 3: computing the KL divergence. We now conclude by computing the KL divergence above. Fix $k \in \{2,\dots,I\}$. Given $\pi'_k \in \Pi_k(\mathbf{\Sigma})$, we write $\pi' = (\pi_2,\dots,\pi_{k-1},\pi'_k,\pi_{k+1},\dots,)$. Using the same computation as in Equation (6.118), we obtain

$$\begin{aligned}\frac{1}{|\Pi_k(\mathbf{\Sigma})|}\sum_{\pi_k\in\Pi_k(\mathbf{\Sigma})}\mathrm{KL}\big(\mathbb{P}_{\tilde{A}_k}^{(\pi,k)},\mathbb{P}_{\tilde{A}_k}^{(\pi')}\big) &\leqslant 8\log\left(\frac{4}{3}\right)\sum_{j\in\Sigma_k^-}\mathbb{E}_{\tilde{A}_k}^{(\pi,k)}[N_{\{k,j\}}]\,\beta_k^2\\ &\leqslant 8\log\left(\frac{4}{3}\right)\cdot\frac{4\chi}{I-1}\cdot\beta_I^2\ ,\end{aligned}$$

where the last inequality uses the facts that $\sum_{j\in\Sigma_k^-}\mathbb{E}_{\tilde{A}_k}^{(\pi,k)}[N_{\{k,j\}}] \leqslant 4\chi/(I-1)$ by construction of $\tilde{A}_k$, and that, by our ordering assumption, $\beta_k^2 \leqslant \beta_I^2$ for all $k \in \{2,\dots,I\}$. Averaging additionally over all permutations $\pi = (\pi_2,\dots,\pi_K) \in \Pi(\mathbf{\Sigma})$, we obtain

$$\frac{1}{|\Pi(\mathbf{\Sigma})|}\sum_{\pi\in\Pi(\mathbf{\Sigma})}\frac{1}{I-1}\sum_{k=2}^{I}\mathrm{KL}\big(\mathbb{P}_{\tilde{A}_k}^{(\pi,k)},\mathbb{P}_{\tilde{A}_k}^{(\pi)}\big) \leqslant 8\log\left(\frac{4}{3}\right)\cdot\frac{4\chi}{I-1}\cdot\beta_I^2\ . \tag{6.127}$$

Finally, combining Pinsker's inequality (6.126) with Jensen's inequality, we get

$$\begin{aligned}\frac{1}{2} &\leqslant \frac{1}{|\Pi(\mathbf{\Sigma})|}\sum_{\pi\in\Pi(\mathbf{\Sigma})}\frac{1}{I-1}\sum_{k=2}^{I}\sqrt{\frac{1}{2}\mathrm{KL}\big(\mathbb{P}_{\tilde{A}_k}^{(\pi,k)},\mathbb{P}_{\tilde{A}_k}^{(\pi)}\big)} \\ &\leqslant \sqrt{\frac{1}{2}\cdot\frac{1}{|\Pi(\mathbf{\Sigma})|}\sum_{\pi\in\Pi(\mathbf{\Sigma})}\frac{1}{I-1}\sum_{k=2}^{I}\mathrm{KL}\big(\mathbb{P}_{\tilde{A}_k}^{(\pi,k)},\mathbb{P}_{\tilde{A}_k}^{(\pi)}\big)} \\ &\leqslant \sqrt{\frac{1}{2}\cdot 8\log\Big(\frac{4}{3}\Big)\cdot\frac{4\chi}{I-1}\cdot\beta_I^2}\ ,\end{aligned}$$

where the last inequality follows from (6.127). Rearranging yields

$$\chi \ \geqslant\ \frac{1}{64\log(4/3)}\frac{I-1}{\beta_I^2} = \frac{1}{64\log(4/3)}\max_{i=2,\dots,K}\frac{i-1}{\beta_i^2}\ , \tag{6.128}$$

where the second equality follows from the definition of I (see (6.121)). From Lemma 6.F.1, we have

$$\max_{i=2,\dots,K}\frac{i-1}{\beta_i^2} \geqslant \frac{1}{\log(2K)}\sum_{i=2}^{K}\frac{1}{\beta_i^2} = \frac{1}{\log(2K)}\sum_{i=2}^{K}\frac{|\Sigma_i^-|}{\|\Delta_i^-\|^2}\ .$$

Step 4: choice of convention $\mathbf{\Sigma}$ and conclusion.

We claim that there exists a tie-breaking convention f_{tb} that satisfies

$$\sum_{i=2}^{K}\frac{|\Sigma_i^-|}{\|\Delta_i^-\|^2} \geqslant \frac{1}{2}\sum_{i=2}^{K}\frac{K_{i;\leqslant 0}}{\|\Delta_i^-\|^2}\ . \tag{6.129}$$

Then, the conclusion (6.112) directly follows from the (6.128) together with (6.129).

We finally finish with a technical construction of a tie-breaking that satisfies (6.129). Consider any suboptimal arm $i \neq 1$. For any $j \neq i$, it holds that

$$\big(j \in \Sigma_i^-\big) \iff \big(\Delta_{i,j} < 0\big) \text{ or } \big(\Delta_{i,j} = 0,\ f_{\mathrm{tb}}(i,j) = -1\big),$$

so that

$$|\Sigma_i^-| = \sum_{\substack{j=1\\ j\neq i}}^{K}\mathbb{1}_{\{\Delta_{i,j}<0\}} + \mathbb{1}_{\{\Delta_{i,j}=0\}}\mathbb{1}_{\{f_{\mathrm{tb}}(i,j)=-1\}}\ .$$

Summing over $i = 2, \dots, K$ gives

$$\begin{aligned}\sum_{i=2}^{K} \frac{|\Sigma_i^-|}{\|\Delta_i^-\|^2} &= \sum_{i=2}^{K} \sum_{\substack{j=1 \\ j\neq i}}^{K} \frac{1}{\|\Delta_i^-\|^2} \left(\mathbb{1}_{\{\Delta_{i,j}<0\}} + \mathbb{1}_{\{\Delta_{i,j}=0\}} \mathbb{1}_{\{f_{\mathrm{tb}}(i,j)=-1\}} \right) \\ &= \sum_{i=2}^{K} \sum_{\substack{j=1 \\ j\neq i}}^{K} \frac{1}{\|\Delta_i^-\|^2} \mathbb{1}_{\{\Delta_{i,j}<0\}} + \sum_{i=2}^{K} \frac{1}{\|\Delta_i^-\|^2} \mathbb{1}_{\{\Delta_{i,1}=0\}} \mathbb{1}_{\{f_{\mathrm{tb}}(i,1)=-1\}} \\ &\quad + \sum_{2\leqslant i<j\leqslant K} \mathbb{1}_{\{\Delta_{i,j}=0\}} \left(\frac{\mathbb{1}_{\{f_{\mathrm{tb}}(i,j)=-1\}}}{\|\Delta_i^-\|^2} + \frac{\mathbb{1}_{\{f_{\mathrm{tb}}(i,j)=1\}}}{\|\Delta_j^-\|^2} \right) \ , \end{aligned} \tag{6.130}$$

where the last equality follows from a simple reorganization of the sum.

We now choose f_{tb} to make the second and third terms as large as possible. Let f_{tb} be the antisymmetric function $f_{\mathrm{tb}} : [K]^2 \to \{-1, 0, 1\}$, defined for any $i < j$ by

$$f_{\mathrm{tb}}(i,j) \coloneqq \begin{cases} 1, & \text{if } i = 1,\ j > 1, \\ \mathbb{1}_{\{\|\Delta_i^-\| > \|\Delta_j^-\|\}} - \mathbb{1}_{\{\|\Delta_i^-\| < \|\Delta_j^-\|\}}, & \text{if } 2 \leq i < j \text{ and } \|\Delta_i^-\| \neq \|\Delta_j^-\|, \\ 1, & \text{if } 2 \leq i < j \text{ and } \|\Delta_i^-\| = \|\Delta_j^-\|. \end{cases}$$

The convention f_{tb} for the row of the CW implies that

$$\sum_{i=2}^{K} \frac{1}{\|\Delta_i^-\|^2} \mathbb{1}_{\{\Delta_{i,1}=0\}} \mathbb{1}_{\{f_{\mathrm{tb}}(i,1)=-1\}} = \sum_{i=2}^{K} \frac{1}{\|\Delta_i^-\|^2} \mathbb{1}_{\{\Delta_{i,1}=0\}} \ . \tag{6.131}$$

Moreover, the expression of $f_{\mathrm{tb}}(i,j)$ for $2 \leqslant i < j \leqslant K$ is chosen so that

$$\begin{aligned}\sum_{2\leqslant i<j\leqslant K} \mathbb{1}_{\{\Delta_{i,j}=0\}} \left(\frac{\mathbb{1}_{\{f_{\mathrm{tb}}(i,j)=-1\}}}{\|\Delta_i^-\|^2} + \frac{\mathbb{1}_{\{f_{\mathrm{tb}}(i,j)=1\}}}{\|\Delta_j^-\|^2} \right) &= \sum_{2\leqslant i<j\leqslant K} \mathbb{1}_{\{\Delta_{i,j}=0\}} \left(\frac{1}{\|\Delta_i^-\|^2} \vee \frac{1}{\|\Delta_j^-\|^2} \right) \\ &\geqslant \frac{1}{2} \sum_{2\leqslant i<j\leqslant K} \mathbb{1}_{\{\Delta_{i,j}=0\}} \left(\frac{1}{\|\Delta_i^-\|^2} + \frac{1}{\|\Delta_j^-\|^2} \right) \ . \end{aligned} \tag{6.132}$$

Finally, gathering Equations (6.130), (6.131), and (6.132), we get

$$\begin{aligned}\sum_{i=2}^{K} \frac{|\Sigma_i^-|}{\|\Delta_i^-\|^2} &\geqslant \sum_{i=2}^{K} \sum_{\substack{j=1 \\ j\neq i}}^{K} \frac{1}{\|\Delta_i^-\|^2} \left(\mathbb{1}_{\{\Delta_{i,j}<0\}} + \mathbb{1}_{\{j=1,\ \Delta_{i,1}=0\}} + \frac{1}{2} \mathbb{1}_{\{j\neq 1,\ \Delta_{i,j}=0\}} \right) \\ &\geqslant \frac{1}{2} \sum_{i=2}^{K} \frac{K_{i;\leqslant 0}}{\|\Delta_i^-\|^2} \ , \end{aligned}$$

which proves Equation (6.129), hence finishing the proof. □

6.D.5 Proof of Corollary 6.4.4

Proof. Without loss of generality, assume $i^* = 1$. Let $\underline{s} = (s_i)_{i\neq 1}$ with $s_i \in [K/8]$ for all $i \neq i^*$, and let $\underline{\Delta} = (\Delta_i)_{i\neq i^*}$ with $\Delta_i \in (0, 1/4)$. Let $\epsilon > 0$ be small.

For simplicity, assume K is a multiple of 8, and set $d = K/2$. We construct the $K \times K$ antisymmetric matrix $M^\epsilon = M^\epsilon(\underline{s}, \underline{\Delta})$:

$$M^\epsilon = \begin{pmatrix} A & -D \\ D^\top & \Lambda \end{pmatrix}, \tag{6.133}$$

where A, D, and Λ are $d \times d$ matrices defined below.

The matrix A is the $d \times d$ antisymmetric matrix with first row $A_{1,\cdot} = (0, \epsilon, \dots, \epsilon) \in \mathbb{R}^d$, and for $i = 2, \dots, d$,

$$A_{i,\cdot} = (-\epsilon, \dots, -\epsilon, \underbrace{0}_{i\text{-th}}, \epsilon, \dots, \epsilon) \in \mathbb{R}^d.$$

The matrix D is the $d \times d$ matrix with nonnegative entries where $D_{1,\cdot} = (\epsilon, \dots, \epsilon) \in \mathbb{R}^d$, and for $i = 2, \dots, d$,

$$D_{i,\cdot} = (\underbrace{\Delta_i, \dots, \Delta_i}_{s_i \text{ times}}, \epsilon, \dots, \epsilon) \in \mathbb{R}^d,$$

which is possible since $s_i \leqslant d$.

To construct Λ, recall d is a multiple of 4 and $s_i \in \{1, \dots, d/4\}$. Define Λ as the block matrix:

$$\Lambda = \begin{pmatrix} J_\epsilon & -\Lambda^{(0)} & \epsilon\mathbf{1} & \Lambda^{(3)} \\ \Lambda^{(0)} & J_\epsilon & -\Lambda^{(1)} & \epsilon\mathbf{1} \\ -\epsilon\mathbf{1} & \Lambda^{(1)} & J_\epsilon & -\Lambda^{(2)} \\ -\Lambda^{(3)} & -\epsilon\mathbf{1} & \Lambda^{(2)} & J_\epsilon \end{pmatrix},$$

where $\mathbf{1}$ is the $(d/4) \times (d/4)$ all-ones matrix, J_ϵ is the $(d/4) \times (d/4)$ antisymmetric matrix with ϵ above the diagonal, and for $l \in \{0, \dots, 3\}$, $\Lambda^{(l)}$ is the $(d/4) \times (d/4)$ matrix where the i-th row is

$$\Lambda^{(l)}_{i,\cdot} = (\underbrace{\Delta_j, \dots, \Delta_j}_{s_j \text{ times}}, \epsilon, \dots, \epsilon) \in \mathbb{R}^{d/4}, \quad j = d + \tfrac{d}{4}l + i.$$

The matrix $M^\epsilon = M(\underline{s}, \underline{\Delta}, \epsilon)$ is clearly antisymmetric. Its first row is $(0, \epsilon, \dots, \epsilon)$, so $i^* = 1$ and $M^\epsilon \in \mathbb{D}_{\text{cw}}$.

For each arm $i = 2, \dots, K$, row i of M^ϵ contains exactly s_i entries of magnitude $-\Delta_i$, with all other negative entries equal to $-\epsilon$. For sufficiently small $\epsilon > 0$, the optimal sparsity $\boldsymbol{s}^*_{M^\epsilon}$ achieving the minimum in (6.4) equals $\underline{s}$, with associated gaps $(M^\epsilon_{i,(s_i)})_{i\neq i^*} = (-\Delta_i)_{i\neq i^*}$. Moreover, M^ϵ has no ties since $\Delta_i \neq 0$ and $\epsilon > 0$.

Consider Corollary 6.4.4. Let $\delta \leqslant 1/12$. For $\boldsymbol{\Delta} \in \mathbb{D}_{\text{cw}}$, construction yields $\epsilon > 0$ small such that $M^\epsilon(\boldsymbol{s}^*_{\boldsymbol{\Delta}}, \boldsymbol{\Delta}_{(s^*)}) \in \mathbb{D}(\boldsymbol{\Delta})$ with no ties.

Theorem 6.4.2 applied to M^ϵ gives $\tilde{\mathbf{\Delta}} \in \mathbb{D}(M^\epsilon) = \mathbb{D}(\mathbf{\Delta})$ satisfying

$$\mathbb{P}_{\tilde{\mathbf{\Delta}},A}\left(N_\delta \geqslant \frac{1}{3}\max_{i\neq i^*}\frac{K^\epsilon_{i;<0}}{\|(M^\epsilon_i)^-\|^2}\log\left(\frac{1}{6\delta}\right) \vee \frac{1}{37\log(2K)}\sum_{i\neq i^*}\frac{K^\epsilon_{i;<0}}{\|(M^\epsilon_i)^-\|^2}\right) \geqslant \delta,$$

where $K^\epsilon_{i;<0}$ counts negative entries of row i. By construction, $K^\epsilon_{i;<0} \geqslant K/8$ for $i = 2,\dots,K$. For small ϵ,

$$s_i^*\Delta^2_{i,(s_i^*)} \leqslant \|(M^\epsilon_i)^-\|^2 \leqslant s_i^*\Delta^2_{i,(s_i^*)} + (K - s_i^*)\epsilon^2, \leqslant 2s_i^*\Delta^2_{i,(s_i^*)}$$

yielding

$$\mathbb{P}_{\tilde{\mathbf{\Delta}},A}\left(N_\delta \geqslant \frac{1}{48}\max_{i\neq i^*}\frac{K}{s_i^*\Delta^2_{i,(s_i^*)}}\log\left(\frac{1}{6\delta}\right) \vee \frac{1}{592\log(2K)}\sum_{i\neq i^*}\frac{K}{s_i^*\Delta^2_{i,(s_i^*)}}\right) \geqslant \delta.$$

This scales as $H_{\text{explore}}(\boldsymbol{s}^*)$ up to $\log K$ factors, proving the first part of Corollary 6.4.4.

The $H_{\text{certify}}(\boldsymbol{s}^*, \delta)$ term follows from the quantile bound in Proposition 6.4.1: by construction of M^ϵ, we have $\tilde{\Delta}_{i,(1)} = \tilde{\Delta}_{i,(s_i^*)}$ for all $i \neq i^*$, so

$$H_{\text{certify}}(\boldsymbol{s}^*, \delta) = \sum_{i\neq i^*}\frac{1}{\Delta_{i,(s_i^*)}} = \sum_{i\neq i^*}\frac{1}{\tilde{\Delta}_{i,(1)}},$$

and the lower bound applies directly. □

6.D.6 Lower Bounds Preserving CW Row Structure

Consider $\mathbf{\Delta} \in \mathbb{D}_{\text{cw}}$. For simplicity, assume that $\mathbf{\Delta}$ has no ties.[8] Let $\mathbf{\Sigma}$ be its sign matrix as in Equation (6.107), and let $\pi \in \Pi(\mathbf{\Sigma})$ (see (6.108)) with associated matrix $\mathbf{\Delta}^\pi$ defined in Equation (6.109).

By Lemma 6.D.2, $\mathbf{\Delta}^\pi$ has the same gap structure as $\mathbf{\Delta}$: it preserves all signs (hence all pairwise preferences) and gap magnitudes up to reordering. However, $\mathbf{\Delta}^\pi$ may alter the Condorcet winner row, so in general $H_{\text{cw}}(\mathbf{\Delta}^\pi) \neq H_{\text{cw}}(\mathbf{\Delta})$, and the construction can even drastically increase it: $H_{\text{cw}}(\mathbf{\Delta}^\pi) \gg H_{\text{cw}}(\mathbf{\Delta})$.

In this section, we explain how the lower bound techniques from the proof of Theorem 6.4.2 can be adapted to also preserve the CW row.

To this end, define $\tilde{\Pi}(\mathbf{\Sigma}) \subset \Pi(\mathbf{\Sigma})$ as the subset of permutations preserving the CW row. For each $i \in \{2,\dots,K\}$, let $\tilde{\Pi}_i(\mathbf{\Sigma})$ be permutations of Σ_i^- that fix $i^* = 1$:

$$\tilde{\Pi}_i(\mathbf{\Sigma}) \coloneqq \left\{\pi_i : \Sigma_i^- \to \Sigma_i^- \,\middle|\, \pi_i \text{ is a bijection and } \pi_i(1) = 1\right\}, \tag{6.134}$$

and set $\tilde{\Pi}(\mathbf{\Sigma}) \coloneqq \tilde{\Pi}_2(\mathbf{\Sigma}) \times \cdots \times \tilde{\Pi}_K(\mathbf{\Sigma})$. For $\pi \in \tilde{\Pi}(\mathbf{\Sigma})$, construct $\mathbf{\Delta}^{(\pi)}$ via Equation (6.109). In addition to properties 1, 3, and 4 of Lemma 6.D.2, we have:

8. If $\mathbf{\Delta}$ contains ties, we fix the convention $f_{\text{tb}} \equiv 0$, i.e., we never permute zero entries, hence keeping them uninformative.

Lemma 6.D.8. *For any $\pi \in \tilde{\Pi}(\mathbf{\Sigma})$, $\mathbf{\Delta}^{(\pi)}$ satisfies:*

2' $\Delta^{(\pi)}_{1,\cdot} = \Delta_{1,\cdot}$ (CW row preservation).

We can then derive the following theorem, analogous to Theorem 6.4.2:

Theorem 6.D.9. *Let A be a δ-correct algorithm over $\mathbb{D}_{\mathrm{cw}}$, with $\delta \leqslant 1/12$. Let $\mathbf{\Delta} \in \mathbb{D}_{\mathrm{cw}}$ have no ties. Define*

$$\tilde{\chi} \coloneqq \inf \left\{ x > 0 : \sup_{\pi \in \tilde{\Pi}(\mathbf{\Sigma})} \mathbb{P}_{\mathbf{\Delta}^{(\pi)},A}(N_\delta > x) \leqslant \delta \right\}. \tag{6.135}$$

Then,

$$\tilde{\chi} \geqslant \frac{1}{16 \log(4/3)} \max_{i \neq i^*} \left(\frac{1}{\Delta^2_{i^*,i}} \wedge \frac{K_{i;<0}}{\|\Delta_i^-\|^2} \right) \log\left(\frac{1}{6\delta}\right), \tag{6.136}$$

$$\tilde{\chi} \geqslant \frac{1}{128 \log(4/3)} \frac{1}{\log(2K)} \sum_{i \neq i^*} \left(\frac{1}{\Delta^2_{i^*,i}} \wedge \frac{K_{i;<0}}{\|\Delta_i^-\|^2} \right). \tag{6.137}$$

Similarly to Section 6.4, we define a subclass of $\mathbb{D}(\mathbf{\Delta})$ (see (6.10)) that additionally preserves the CW row $\Delta_{i^*,\cdot}$:

$$\mathbb{D}_0(\mathbf{\Delta}) = \{ \tilde{\mathbf{\Delta}} \in \mathbb{D}(\mathbf{\Delta}) \text{ s.t. } (\tilde{\Delta}_{i^*,i})_{i \neq i^*} = (\Delta_{i^*,i})_{i \neq i^*} \} \ . \tag{6.138}$$

Corollary 6.D.10. *Let A be a δ-correct algorithm over $\mathbb{D}_{\mathrm{cw}}$, with $\delta \leqslant 1/12$. Let $\mathbf{\Delta} \in \mathbb{D}_{\mathrm{cw}}$. Then there exists $\tilde{\mathbf{\Delta}} \in \mathbb{D}_0(\mathbf{\Delta})$ such that, with $\mathbb{P}_{\tilde{\mathbf{\Delta}},A}$-probability at least δ, the budget N_δ satisfies*

$$N_\delta \gtrsim \sum_{i \neq i^*} \frac{\log(1/\delta)}{\Delta^2_{i^*,i} \vee \Delta^2_{i,(s_i^*)}} + \max_{i \neq i^*} \frac{\log(1/\delta)}{\Delta^2_{i^*,i} \vee \frac{\|\Delta_{i,(s_i^*)}\|^2}{K_{i;<0}}} + \sum_{i \neq i^*} \frac{1}{\Delta^2_{i^*,i} \vee \frac{\|\Delta_{i,(s_i^*)}\|^2}{K_{i;<0}}}, \tag{6.139}$$

where $\gtrsim$ hides logarithmic K factors and numerical constants.

Proof. The proof follows by taking M^ϵ as in the proof of Corollary 6.4.4 (see Appendix 6.D.5), except with the first row fixed as $\Delta_{i^*,\cdot}$. This constructs $M^\epsilon \in \mathbb{D}_0(\mathbf{\Delta})$ where each row i has $\geqslant K_{i;<0}$ negative entries. The corollary then follows from Theorem 6.D.9 and the quantile bound in Proposition 6.4.1. □

Remark 6.D.11. This reveals the fundamental trade-off between eliminating suboptimal arms against the CW versus finding better competitors among them. We identify three regimes.

When the CW is the strongest opponent (CW-SO), H_{cw} was already proved from Proposition 6.4.1 to be high-probability optimal, achieved by Algorithm (17) and Maiti et al. (2024). Actually, Karnin (2016) proves that it is even optimal for expectation of the budget, at least in the asymptotic regime of $\delta \to 0$.

In the CW-uniformly-poor-opponent regime,

$$\forall i \neq i^*, \quad \Delta^2_{i^*,i} \leqslant \frac{\|\Delta_i^-\|^2}{K_{i;<0}}, \tag{CW-PO}$$

we have $H_{\mathrm{cw}} \geqslant H_{\mathrm{certify}}(\boldsymbol{s}^*, \delta) + H_{\mathrm{explore}}(\boldsymbol{s}^*, \delta)$. Our bound (6.4) improves Maiti (2025), and Corollary 6.D.10 proves minimax optimality of $H_{\mathrm{certify}}(\boldsymbol{s}^*, \delta) + H_{\mathrm{explore}}(\boldsymbol{s}^*, \delta)$ over $\mathbb{D}_0(\boldsymbol{\Delta})$.

In the CW-intermediate-opponent regime,

$$\forall i \neq i^*, \quad \frac{\|\Delta_i^-\|^2}{K_{i;<0}} \leqslant \Delta_{i^*,i}^2 \leqslant \Delta_{i,(s_i^*)}^2, \tag{CW-IO}$$

a transition occurs between constant-δ (where the lower bound matches H_{cw}) and $\delta \to 0$ regimes (where it can be much smaller). A finer combinatorial analysis is needed to pinpoint the exact trade-off.

Proof of Theorem 6.D.9. Assume without loss of generality that $i^* = 1$. We start with the proof of Equation (6.136), which follows the proof of Equation (6.111). From careful inspection, Steps 1, 2, and 3 apply verbatim, replacing χ by $\tilde{\chi}$ and Π by $\tilde{\Pi}$.

Fix $k \neq i^*$. With the same notation and construction, one constructs $\tilde{A}$ with budget upper bounded by $\tilde{\chi}$ such that

$$\log\left(\frac{1}{6\delta}\right) \leqslant 8 \log\left(\frac{4}{3}\right) \frac{1}{|\tilde{\Pi}_k(\boldsymbol{\Sigma})|} \sum_{\pi_k \in \tilde{\Pi}_k(\boldsymbol{\Sigma})} \sum_{j \in \Sigma_k^-} \mathbb{E}_{\tilde{A}}^{(\pi,k)}[N_{\{k,j\}}] \Delta_{k,\pi_k(j)}^2, \tag{6.140}$$

with the only difference in the subsequent computation.

For $\pi \in \tilde{\Pi}_k$, we have $\pi_k(1) = 1$ and $\pi_k|_{\Sigma_k^- \setminus \{1\}}$ is a bijection, so $|\tilde{\Pi}_k(\boldsymbol{\Sigma})| \simeq \mathfrak{S}_{k_i}$ where $k_i := |\Sigma_k^- \setminus \{1\}| = K_{k;<0} - 1$. Separating the role of 1 and the rest of Σ_k^- in (6.140),

$$\begin{aligned}
&\frac{1}{|\tilde{\Pi}_k(\boldsymbol{\Sigma})|} \sum_{\pi_k \in \tilde{\Pi}_k(\boldsymbol{\Sigma})} \sum_{j \in \Sigma_k^-} \mathbb{E}_{\tilde{A}}^{(\pi,k)}[N_{\{k,j\}}] \Delta_{k,\pi_k(j)}^2 \\
&= \mathbb{E}_{\tilde{A}}^{(\pi,k)}[N_{\{k,1\}}] \Delta_{k,1}^2 + \frac{1}{|\mathfrak{S}_{k_i}|} \sum_{\pi_k \in \tilde{\Pi}_k(\boldsymbol{\Sigma})} \sum_{j \in \Sigma_k^- \setminus \{1\}} \mathbb{E}_{\tilde{A}}^{(\pi,k)}[N_{\{k,j\}}] \Delta_{k,\pi_k(j)}^2,
\end{aligned}$$

where $\mathbb{E}_{\tilde{A}}^{(\pi,k)}$ is independent of π_k. By Lemma 6.F.3,

$$\begin{aligned}
\frac{1}{8\log(4/3)} \log\left(\frac{1}{6\delta}\right) &\leqslant \mathbb{E}_{\tilde{A}}^{(\pi,k)}[N_{\{k,1\}}] \Delta_{k,1}^2 + \frac{1}{K_{k;<0} - 1} \sum_{j \in \Sigma_k^- \setminus \{1\}} \mathbb{E}_{\tilde{A}}^{(\pi,k)}[N_{\{k,j\}}] \sum_{j \in \Sigma_k^- \setminus \{1\}} \Delta_{k,j}^2 \\
&\leqslant \Delta_{k,1}^2 \tilde{\chi} + \frac{\|\Delta_k^-\|^2 - \Delta_{k,1}^2}{K_{k;<0} - 1} \tilde{\chi},
\end{aligned}$$

using $\sum_{j \in \Sigma_k^- \setminus \{1\}} \mathbb{E}_{\tilde{A}}^{(\pi,k)}[N_{\{k,j\}}] \leqslant \tilde{\chi}$ and $\sum_{j \in \Sigma_k^- \setminus \{1\}} \Delta_{k,j}^2 = \|\Delta_k^-\|^2 - \Delta_{k,1}^2$. Finally,

$$\Delta_{k,1}^2 + \frac{\|\Delta_k^-\|^2 - \Delta_{k,1}^2}{K_{k;<0} - 1} \leqslant 2 \left(\Delta_{k,1}^2 \vee \frac{\|\Delta_k^-\|^2}{K_{k;<0}} \right).$$

Taking the maximum over $k \neq i^*$ yields

$$\tilde{\chi} \geqslant \frac{1}{16 \log(4/3)} \max_{k \neq i^*} \frac{1}{\Delta_{k,1}^2 \vee \frac{\|\Delta_k^-\|^2}{K_{k;<0}}} \log\Big(\frac{1}{6\delta}\Big),$$

which is Equation (6.136).

The proof of (6.137) follows the proof of (6.112) step-by-step, highlighting differences below.

Step 1: Define $\tilde{\beta}_k = \Delta_{k,1}^2 \vee \frac{\|\Delta_k^-\|^2}{K_{k;<0}}$ and $\tilde{I} := \operatorname{argmax}_{k=2}^K \frac{k-1}{\tilde{\beta}_k}$.

Step 2: The same event (using $\tilde{I}$) bounds the probabilities, so (6.126) holds.

Step 3: The KL upper bound computation adapts as above, yielding

$$\frac{1}{|\tilde{\Pi}|} \sum_{\pi \in \tilde{\Pi}} \frac{1}{I-1} \sum_{k=2}^{I} \mathrm{KL}\big(\mathbb{P}_{\tilde{A}_k}^{(\pi,k)}, \mathbb{P}_{\tilde{A}_k}^{(\pi)}\big) \leqslant 8 \log\Big(\frac{4}{3}\Big) \cdot \frac{4\tilde{\chi}}{I-1} \cdot \tilde{\beta}_I^2. \tag{6.141}$$

which conclude from rearranging.

Step 4 is unnecessary since $\mathbf{\Sigma}$ avoids ties by convention.

□

6.E Proofs of Section 6.5

Comparison with Fixed Confidence lower bounds. The proofs for fixed-confidence and fixed-budget settings are remarkably similar. This reflects the strong connection between fixed-budget algorithms and fixed-confidence algorithms with high-probability budget bounds. However, fixed-budget minimax lower bounds cannot be directly deduced from the instance-dependent fixed confidence lower bounds derived earlier. We conjecture that obtaining instance-dependent lower bounds in the fixed-budget setting is considerably more challenging—if not outright impossible.

The fundamental reason lies in the nature of the algorithms themselves. Any δ-correct fixed-confidence algorithm must incorporate an internal stopping criterion that checks that the identified arm $\hat{\imath}$ is indeed the CW, under the assumption that such a unique CW exists in $\mathbb{D}_{\mathrm{cw}}$. Consider now what happens when applying this algorithm to a modified environment with two "weak" Condorcet winner candidates i_1^* and i_2^*, where both rows satisfy $\Delta_{i_1^*,\cdot} \geqslant \mathbf{0}$ and $\Delta_{i_2^*,\cdot} \geqslant \mathbf{0}$. The algorithm cannot distinguish between these candidates with probability $1 - \delta$, forcing an infinite expected stopping time as the verification step never confidently resolves the ambiguity.

Fixed-budget algorithms fundamentally lack such stopping rules, rendering this infinite-budget argument inapplicable. Instead, our lower bounds rely on symmetry arguments: for matrices where multiple suboptimal arms have identical "difficulty profiles" (i.e., identical multisets of negative gaps), any reasonable algorithm must select uniformly at random among the ambiguous best arms. Deriving such lower bounds requires constructing highly symmetric matrices that exploit this randomization, a significantly more restrictive condition than the instance-dependent constructions used in fixed confidence.

In Sections 6.D and 6.E, we propose two complete constructions respectively for each setting.

The proofs for fixed-budget results are similar to the fixed-confidence ones, except that they are applied to specific highly symmetric matrices for the reasons described above.

In Subsection 6.E.1, we prove Theorem 6.5.1. Theorem 6.5.3 follows in Subsection 6.E.2.

Roadmap for Fixed-Budget Lower Bounds The proofs in this section follow the same three-step change-of-measure pattern introduced for the fixed-confidence case in Subsection 6.D.1. However, the fixed-budget setting requires three important adaptations, which we now describe in detail. In the fixed-budget setting, we aim to lower bound the worst-case error

$$\inf_A \sup_{M \in \mathbb{D}} \mathbb{P}_{A,M}(\hat{\imath}_T \neq i^*),$$

where the infimum is over all algorithms A with fixed budget T, and $\mathbb{D}$ is some class of instances.

Step 1: Reference and alternative instances. In the fixed-confidence setting, the reference instance can be any arbitrary gap matrix $\mathbf{\Delta} \in \mathbb{D}_{\mathrm{cw}}$. Here, we instead construct a highly symmetric reference matrix $M \in \mathbb{D}$ that serves as a "hard instance" for the minimax bound. For each suboptimal arm $k \neq i^*$, we construct an alternative instance $M^{(k)}$ by setting all negative entries in row k (and corresponding column entries to preserve antisymmetry) to zero. This ensures $M^{(k)} \notin \mathbb{D}_{\mathrm{cw}}$.

Step 2: Separating event and total variation bound. Unlike fixed-confidence algorithms, which use a stopping time N_δ to construct events on which the two distributions disagree, fixed-budget algorithms have no stopping rule. Instead, we exploit the symmetry of the reference matrix M, together with the recommendation rule.

Step 3: KL decomposition. This step is conceptually identical to the fixed-confidence proofs. The KL divergence decomposes as

$$\mathrm{KL}(\mathbb{P}_M, \mathbb{P}_{M^{(k)}}) = \sum_{k<j} \mathbb{E}_M[N_{\{k,j\}}]\, \mathrm{kl}(\Delta_{k,j}, \Delta^{(k)}_{k,j}),$$

where $N_{\{k,j\}}$ counts unordered duels between arms k and j. The key difference is that the fixed budget T directly bounds $\sum_{k<j} \mathbb{E}_M[N_{\{k,j\}}] \leqslant T$, whereas fixed-confidence proofs require truncation at the $(1-\delta)$-quantile χ and reinterpretation. Combining the total variation lower bound from Step 2 with this KL upper bound from Step 3 yields the desired lower bound on ϵ_T.

6.E.1 Proof of Theorem 6.5.1

Let $\underline{\Delta} = (\Delta_i)_{i \in [K]}$ be a K-dimensional vector such that $\Delta_1 = 0$ and $\Delta_i \in (0, 1/4)$ for $i \neq 1$, and assume without loss of generality that $\Delta_2 \leqslant \Delta_3 \leqslant \cdots \leqslant \Delta_K$. Recall that $\mathbb{D}^{(1)}(\underline{\Delta})$ (see Definition 6.13) denotes the class of gap matrices that admit a Condorcet winner whose row is equal to $\underline{\Delta}$ up to permutation.

Fix an algorithm A with a fixed budget $T \in \mathbb{N}^*$. Define the worst-case error probability of A over the class $\mathbb{D}^{(1)}(\underline{\Delta})$ as

$$\epsilon_T \coloneqq \max_{M \in \mathbb{D}^{(1)}(\underline{\Delta})} \mathbb{P}_{M,A}(\hat{\imath}_T \neq i^*(M)), \tag{6.142}$$

where $\hat{\imath}_T$ denotes the recommendation of algorithm A after T queries.

We aim to prove Theorem 6.5.1, namely that

$$\epsilon_T \;\geq\; \frac{1}{4}\exp\left(-\frac{T}{\frac{1}{22}\sum_{i=2}^{K}\frac{1}{\Delta_i^2}}\right).$$

Sketch of proof. The proof follows the three-step roadmap of Subsection 6.D.1. (i) Reference and perturbed instances $M^{(1)}, M^{(k)}$. Since we seek a minimax bound, we can construct a specific reference matrix $M^{(1)} \in \mathbb{D}^{(1)}(\underline{\Delta})$ with Condorcet winner $i^* = 1$, where arm 1 is the strongest opponent of every suboptimal arm. For each suboptimal $k \neq 1$, $M^{(k)}$ modifies row k's negative entries of $M^{(1)}$ to make k the CW while preserving $M^{(k)} \in \mathbb{D}^{(1)}(\underline{\Delta})$. (ii) Separating event and TV bound. Algorithm A is ϵ_T-correct over $\mathbb{D}^{(1)}(\underline{\Delta})$, so $\mathbb{P}_{M^{(k)}}(\hat{\imath}_T = k) \geq 1 - \epsilon_T$. The event $B_k = \{\hat{\imath}_T \neq k\}$ thus satisfies $\mathbb{P}_{M^{(1)}}(B_k) \geq 1 - \epsilon_T$ and $\mathbb{P}_{M^{(k)}}(B_k) \leq \epsilon_T$, yielding $\mathrm{TV}(\mathbb{P}_{M^{(1)}}, \mathbb{P}_{M^{(k)}}) \geq 1 - 2\epsilon_T$. (iii) KL decomposition and computation. Similar computation as in the proof of Theorem 6.4.1.

Proof of Theorem 6.5.1. Step 1: construction of the reference instance $M^{(1)}$.
We construct within the class $\mathbb{D}^{(1)}(\underline{\Delta})$ a highly structured instance. The key structural property of this matrix is that the Condorcet winner $k^* = 1$ is the strongest opponent of every other arm.

Define a gap matrix $M^{(1)} \in \mathbb{D}_{\mathrm{cw}}$ by setting $M^{(1)}_{1,\cdot} = \underline{\Delta}$, and, for $i, j \neq 1$,

$$M^{(1)}_{i,j} \coloneqq \begin{cases} \Delta_j, & \text{if } i < j, \\ -\Delta_i, & \text{if } i > j, \\ 0, & \text{if } i = j. \end{cases}$$

Thus $M^{(1)}$ has the form

$$M^{(1)} = \begin{pmatrix} 0 & \Delta_2 & \Delta_3 & \Delta_4 & \cdots & \Delta_{K-1} & \Delta_K \\ -\Delta_2 & 0 & \Delta_3 & \Delta_4 & \cdots & \Delta_{K-1} & \Delta_K \\ -\Delta_3 & -\Delta_3 & 0 & \Delta_4 & \cdots & \Delta_{K-1} & \Delta_K \\ -\Delta_4 & -\Delta_4 & -\Delta_4 & 0 & \cdots & \Delta_{K-1} & \Delta_K \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ -\Delta_{K-1} & -\Delta_{K-1} & -\Delta_{K-1} & -\Delta_{K-1} & \cdots & 0 & \Delta_K \\ -\Delta_K & -\Delta_K & -\Delta_K & -\Delta_K & \cdots & -\Delta_K & 0 \end{pmatrix}.$$

By construction, we have $M^{(1)} \in \mathbb{D}^{(1)}(\underline{\Delta})$, and its Condorcet winner is $i^*(M^{(1)}) = 1$.

For each $k \geqslant 2$, define the matrix $M^{(k)}$ as follows:

— For $i, j \neq k$, set $M^{(k)}_{i,j} = M^{(1)}_{i,j}$.

— For $j < k$, set $M^{(k)}_{k,j} = \Delta_{j+1}$ and $M^{(k)}_{j,k} = -\Delta_{j+1}$.

— For $j \geqslant k$, set $M^{(k)}_{k,j} = M^{(1)}_{k,j}$ and $M^{(k)}_{j,k} = M^{(1)}_{j,k}$.

The matrix $M^{(k)}$ can be written as

$$M^{(k)} = \begin{pmatrix} 0 & \Delta_2 & \cdots & \Delta_{k-1} & -\Delta_2 & \Delta_{k+1} & \cdots & \Delta_K \\ -\Delta_2 & 0 & \cdots & \Delta_{k-1} & -\Delta_3 & \Delta_{k+1} & \cdots & \Delta_K \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & & \vdots \\ -\Delta_{k-1} & -\Delta_{k-1} & \cdots & 0 & -\Delta_k & \Delta_{k+1} & \cdots & \Delta_K \\ \Delta_2 & \Delta_3 & \cdots & \Delta_k & 0 & \Delta_{k+1} & \cdots & \Delta_K \\ -\Delta_{k+1} & -\Delta_{k+1} & \cdots & -\Delta_{k+1} & -\Delta_{k+1} & 0 & \cdots & \Delta_K \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \ddots & \vdots \\ -\Delta_K & -\Delta_K & \cdots & -\Delta_K & -\Delta_K & -\Delta_K & \cdots & 0 \end{pmatrix},$$

where the blue entries indicate the differences with respect to $M^{(1)}$.

It is straightforward to check that, for each k, $M^{(k)} \in \mathbb{D}_{\mathrm{cw}}$. These matrices have three key properties: (i) $M^{(k)}$ does not have the same Condorcet winner as $M^{(1)}$, indeed $i^*(M^{(k)}) = k$; (ii) we have $M^{(k)} \in \mathbb{D}^{(1)}(\underline{\Delta})$, indeed the k-th row $M^{(k)}_{k,\cdot}$ is equal to $\underline{\Delta}$ up to a permutation; and (iii) the environment with gap matrix $M^{(k)}$ is difficult to distinguish from the one defined by $M^{(1)}$ in terms of KL divergence.

For $k \geqslant 2$, denote by $\mathbb{P}^{(k)}$ the distribution of the data when the underlying gap matrix is $M^{(k)}$.

Step 2: TV bound.

Let A be a δ-correct algorithm over $\mathbb{D}^{(1)}(\underline{\Delta})$, and let $\hat{\imath}$ denote its output. For any $k \geqslant 1$, when the true gap matrix is $M^{(k)}$ the Condorcet winner is k, and $M^{(k)} \in \mathbb{D}^{(1)}(\underline{\Delta})$. Then, the definition of ϵ_T (see (6.142)) implies

$$\forall k \in [K], \quad \mathbb{P}^{(k)}(\hat{\imath} \neq k) \leqslant \epsilon_T \ .$$

In particular, we have

$$1 - 2\epsilon_T \leqslant \mathbb{P}^{(1)}(\hat{\imath} \neq k) - \mathbb{P}^{(k)}(\hat{\imath} \neq k) \leqslant \mathrm{TV}\big(\mathbb{P}^{(1)}, \mathbb{P}^{(k)}\big) \ .$$

Then, with Bretagnolle–Huber inequality, we have

$$1 - 2\epsilon_T \leqslant \mathrm{TV}\big(\mathbb{P}^{(1)}, \mathbb{P}^{(k)}\big) \leqslant 1 - \tfrac{1}{2}\exp\big\{-\mathrm{KL}(\mathbb{P}^{(1)}, \mathbb{P}^{(k)})\big\}.$$

Step 3: computing the KL divergence and concluding.

For $i < j$ in $[K]$, let $N_{\{i,j\}}$ denote the total number of observed duels between i and j under algorithm A. Using the divergence decomposition lemma (Lemma 15.1 in Lattimore and Szepesvári,

2020), we have

$$\begin{aligned}\mathrm{KL}(\mathbb{P}^{(1)}, \mathbb{P}^{(k)}) &= \sum_{1 \leqslant i < j \leqslant K} \mathbb{E}^{(1)}\Big[N_{\{i,j\}}\Big]\ \mathrm{KL}(\mathbb{P}^{(1)}_{i,j}, \mathbb{P}^{(k)}_{i,j}) \\ &= \sum_{i=1}^{k-1} \mathbb{E}^{(1)}[N_{\{k,i\}}]\, \mathrm{KL}(\mathbb{P}^{(1)}_{k,i}, \mathbb{P}^{(k)}_{k,i})\ , \end{aligned} \tag{6.143}$$

since the two instances differ only on pairs involving arm k.

For all $i < k$, the corresponding Bernoulli feedback distributions satisfy

$$\mathbb{P}^{(1)}_{k,i} = \mathcal{B}\Big(\tfrac{1}{2} - \Delta_k\Big)\,, \qquad \mathbb{P}^{(k)}_{k,i} = \mathcal{B}\Big(\tfrac{1}{2} + \Delta_{i+1}\Big)\,,$$

so that, with the same sequence of inequalities as in (6.94), (6.95), we obtain

$$\begin{aligned}\mathrm{KL}(\mathbb{P}^{(1)}_{k,i}, \mathbb{P}^{(k)}_{k,i}) &= \mathrm{kl}(-\Delta_k + \tfrac{1}{2},\ \Delta_{i+1} + \tfrac{1}{2}) \\ &\leqslant \frac{(-\Delta_k + \frac{1}{2} - (\Delta_{i+1} + \frac{1}{2}))^2}{(-\Delta_{i+1} + \frac{1}{2})(\Delta_{i+1} + \frac{1}{2})} \\ &\leqslant \frac{(2\Delta_k)^2}{3/16}\ \leq\ 22\,\Delta_k^2\ , \end{aligned}$$

where we used that $(\Delta_j)_{j \in \{2,\dots,K\}}$ is nondecreasing and $k \geqslant i+1$, which yields $\Delta_k \geqslant \Delta_{i+1}$. Also, we use that $\Delta_{i+1} \in (0, 1/4)$ and thus the denominator is bounded below by $3/16$.

Now, from (6.143) and the last inequality, we obtain

$$\mathrm{KL}(\mathbb{P}^{(1)}, \mathbb{P}^{(k)}) \leqslant 22 \sum_{i=1}^{k-1} \mathbb{E}^{(1)}\Big[N_{\{k,i\}}\Big]\, \Delta_k^2\ .$$

From Step 3, this bound implies

$$\sum_{i=1}^{k-1} \mathbb{E}^{(1)}\Big[N_{\{k,i\}}\Big] \geqslant \frac{1}{22\Delta_k^2}\ \log\frac{1}{4\epsilon_T}\ .$$

Summing over $k \geqslant 2$, and using that A has a budget T, we conclude that

$$T \geqslant \sum_{k=2}^{K} \sum_{i=1}^{k-1} \mathbb{E}^{(1)}\left[N_{\{k,i\}}\right] \geqslant \frac{1}{22} \sum_{k=2}^{K} \frac{1}{\Delta_k^2} \log\frac{1}{4\epsilon_T}\ ,$$

which, after rearranging, is exactly the claimed lower bound. □

6.E.2 Proof of Theorem 6.5.3

Before proving Theorem 6.5.3, we extend the notion of Condorcet winner to a broader class of matrices. Let $\mathbf{\Delta}$ be an antisymmetric matrix. We denote as a weak Condorcet winner, an arm

$i^* \in [K]$ such that

$$\forall i \neq i^*, \Delta_{i^*,i} \geqslant 0 \text{ and, } \forall i \neq i^*, \min_{j \neq i} \Delta_{i,j} < 0 \ . \tag{6.144}$$

Note that such arm is unique if it exists. Consider $\mathbb{D}_{\text{wcw}}$ as the class of dueling bandit environments for which there exists such weak Condorcet winner

$$\mathbb{D}_{\text{wcw}} := \Big\{ \boldsymbol{\Delta} \in [-\tfrac{1}{4}, \tfrac{1}{4}]^{K \times K} : \exists! \, i^* \in [K] \text{ such that } \forall j, \ \Delta_{i^*,j} \geqslant 0 \Big\}. \tag{6.145}$$

Observe that $\mathbb{D}_{\text{cw}} \subset \mathbb{D}_{\text{wcw}}$, and that naturally, any Condorcet winner is a weak Condorcet winner. Moreover, for a matrix $\boldsymbol{\Delta} \in \mathbb{D}_{\text{wcw}}$, the quantities $(\boldsymbol{s}^*_{\boldsymbol{\Delta}}, \Delta_{(\boldsymbol{s}^*)})$ defined in Section 6.4 are still well defined. Indeed, the minimum in Equation (6.4) for which $\boldsymbol{s}^*_{\boldsymbol{\Delta}}$ is defined as the argmin is still well defined under weak Condorcet winner assumption.

As a variant of the class $\mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$ introduced in (6.14), we introduce the class:

$$\mathbb{D}^{(3)}(\underline{\Delta}, \underline{s}) := \Big\{ \boldsymbol{\Delta} \in \mathbb{D}_{\text{wcw}} : \exists \, \sigma \in \mathfrak{S}_K \text{ s.t. } \boldsymbol{s}^*_{\boldsymbol{\Delta}} = \sigma(\boldsymbol{s}) \text{ and } \Delta_{(s^*)} = \sigma(\Delta_{(s^*)}) \Big\}. \tag{6.146}$$

The introduction of this definition is motivated by the fact that we want to consider in the lower bounds, matrices where the Condorcet winner may have some ties. The following lemma implies that this is feasible.

Lemma 6.E.1. *Let A be an algorithm with a fixed budget T. Then, we have*

$$\max_{\boldsymbol{\Delta} \in \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})} \mathbb{P}_{A, \boldsymbol{\Delta}}(\hat{\imath}_T \neq i^*(\boldsymbol{\Delta})) = \max_{\boldsymbol{\Delta} \in \mathbb{D}^{(2)}(\underline{\Delta}, \underline{s})} \mathbb{P}_{A, \boldsymbol{\Delta}}(\hat{\imath}_T \neq i^*(\boldsymbol{\Delta}))$$

Proof of Lemma 6.E.1. Let $\boldsymbol{\Delta} \in \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$, with a weak Condorcet winner i^*. By definition, for any $i \neq i^*$, $\Delta_{i,\cdot}$ contains at least one negative entry. Let $\epsilon > 0$. Consider the modified matrix $\boldsymbol{\Delta}^\epsilon$ obtained from $\boldsymbol{\Delta}$ by lifting to $\epsilon > 0$ all off-diagonal null entries of on CW row i^* [and $-\epsilon$ to (j, i^*)], so that, for ϵ small enough, $\boldsymbol{\Delta}^\epsilon$ admits i^* as a (strong) Condorcet winner, and $\boldsymbol{\Delta}^\epsilon \in \mathbb{D}_{\text{cw}}$. Now, for ϵ small enough, it holds that $\boldsymbol{s}^*_{\boldsymbol{\Delta}^\epsilon} = \boldsymbol{s}^*_{\boldsymbol{\Delta}}$ and $\Delta^\epsilon_{(s^*)} = \Delta_{(s^*)}$. Then,

$$\mathbb{P}_{A, \boldsymbol{\Delta}^\epsilon}(\hat{\imath}_T \neq i^*(\boldsymbol{\Delta})) \leqslant \max_{\boldsymbol{\Delta} \in \mathbb{D}^{(2)}(\underline{\Delta}, \underline{s})} \mathbb{P}_{A, \boldsymbol{\Delta}}(\hat{\imath}_T \neq i^*(\boldsymbol{\Delta})).$$

Since A has a fixed budget T, one can take the limit $\epsilon \to 0$ in the inequality above. Taking a maximum over $\boldsymbol{\Delta} \in \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$, one therefore obtains

$$\max_{\boldsymbol{\Delta} \in \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})} \mathbb{P}_{A, \boldsymbol{\Delta}}(\hat{\imath}_T \neq i^*(\boldsymbol{\Delta})) \leqslant \max_{\boldsymbol{\Delta} \in \mathbb{D}^{(2)}(\underline{\Delta}, \underline{s})} \mathbb{P}_{A, \boldsymbol{\Delta}}(\hat{\imath}_T \neq i^*(\boldsymbol{\Delta})) \ .$$

We have $\mathbb{D}^{(2)}(\underline{\Delta}, \underline{s}) \subset \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$, so the other side of the inequality is clear. □

Now, we are ready to prove Theorem 6.5.3. Assume that K is a multiple of 8, and denote $d = K/2$.

Let $(\underline{\Delta}, \underline{s})$ be such that $\underline{\Delta} = (\Delta_i)_{i \in [K]}$ with $\Delta_1 = 0$ and $(\Delta_2, \dots, \Delta_K) \in (0, 1/4)^{K-1}$. Let $\underline{s} = (s_1, s_2, \dots, s_K)$ with $s_1 = 0$, $1 \leqslant s_i \leqslant K/4$ for $i = 2, \dots, K$.

Consider the class $\mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$ as defined in Equation (6.146). Fix an algorithm A with a fixed budget T, and define the maximum error of A across $\mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$ as

$$\epsilon_T \coloneqq \max_{\boldsymbol{\Delta} \in \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})} \mathbb{P}_{A, \boldsymbol{\Delta}}(\hat{\imath}_T \neq i^*(\boldsymbol{\Delta})) \ . \tag{6.147}$$

The proof of Theorem 6.5.3 is divided in three lemmas, corresponding to the three terms in the lower bound.

Lemma 6.E.2. *We have*

$$\epsilon_T \geqslant \frac{1}{4} \exp\left(-16 \log(4/3) \frac{T}{\max_{i=2}^{d} \frac{K}{s_i \Delta_i^2}} \right) \ . \tag{6.148}$$

Lemma 6.E.3. *If $(\underline{\Delta}, \underline{s})$ are constants on the indices $i \in \{2, \dots, d\}$, that is, $\exists (\mu, s)$ such that, for any $i \in \{1, \dots, d\}$, $\Delta_i = \mu$, and $s_i = s$, then*

$$\epsilon_T \geqslant \frac{1}{2} - \sqrt{128 \log(4/3) \frac{T}{\frac{K^2}{s\mu^2}}} \ . \tag{6.149}$$

Lemma 6.E.4. *With the same assumption as Lemma 6.E.3, then*

$$\epsilon_T \geqslant \frac{1}{4} \exp\left(-32 \log(4/3) \frac{T}{\frac{K}{\mu^2}} \right) \ . \tag{6.150}$$

Proof of Theorem 6.5.3. Recall that ϵ_T (see (6.147)) is the maximum error over $\mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$. From Lemma 6.E.1, it is also equal to the maximum error over $\mathbb{D}^{(2)}(\underline{\Delta}, \underline{s})$. Together, Lemmas 6.E.2, 6.E.3, and 6.E.4 directly imply Theorem 6.5.3. □

Proof of Lemma 6.E.2. Recalling the definition of ϵ_T from (6.147), $\epsilon_T = \max_{\boldsymbol{\Delta} \in \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})} \mathbb{P}_{A, \boldsymbol{\Delta}}(\hat{\imath}_T \neq i^*(\boldsymbol{\Delta}))$, we want to prove the following bound, equivalent to (6.148)

$$T \geqslant \frac{1}{16 \log(4/3)} \max_{i=2}^{d} \frac{K}{s_i \Delta_i^2} \log\left(\frac{1}{4\epsilon_T} \right) \ .$$

Step 1: reference matrix M^π. For reference, we consider the same matrix as in Subsection 6.D.5, taking $\epsilon = 0$. For completeness, we recall this construction here.

We assumed for simplicity that K is a multiple of 8, and denote $d = K/2$. Consider the $K \times K$ antisymmetric matrix M defined by

$$M = \begin{pmatrix} \mathbf{0} & -D \\ D^\top & \Lambda \end{pmatrix} \ , \tag{6.151}$$

where D and Λ are two $d \times d$ matrices specified below.

The matrix D is the $d \times d$ matrix with nonnegative entries such that the first row is $D_{1,\cdot} = (0, \dots, 0) \in \mathbb{R}^d$, and for any $i = 2, \dots, d$,

$$D_{i,\cdot} = (\underbrace{\Delta_i, \dots, \Delta_i}_{s_i \text{ times}}, 0, \dots, 0) \in \mathbb{R}^d \ ,$$

which is possible since $s_i \leqslant d$ for $i = 2, \dots, d$.

To construct Λ, recall that d is assumed to be a multiple of 4 and that we assumed $s_i \in \{1, \dots, d/4\}$ for all $i \in \{d+1, \dots, K\}$. Define Λ as the following block matrix:

$$\Lambda = \begin{pmatrix} 0 & -\Lambda^{(0)} & 0 & \Lambda^{(3)} \\ \Lambda^{(0)} & 0 & -\Lambda^{(1)} & 0 \\ 0 & \Lambda^{(1)} & 0 & -\Lambda^{(2)} \\ -\Lambda^{(3)} & 0 & \Lambda^{(2)} & 0 \end{pmatrix} ,$$

where, for $l \in \{0, \dots, 3\}$, the sub-matrix $\Lambda^{(l)}$ is the $d/4 \times d/4$ matrix such that, for $i \in \{1, \dots, d/4\}$, the i-th row of $\Lambda^{(l)}$ is

$$\Lambda^{(l)}_{i,\cdot} = (\underbrace{\Delta_j, \dots, \Delta_j}_{s_j \text{ times}}, 0, \dots, 0) \in \mathbb{R}^{d/4} \quad \text{with } j = d + \frac{d}{4} l + i \ . \tag{6.152}$$

Overall, M is clearly antisymmetric by construction. Moreover, for each arm $i = 2, \dots, K$, the i-th row of M contains exactly s_i negative entries of magnitude Δ_i. The first row is equal to 0, so that $M \in \mathbb{D}_{\text{wcw}}$ (see(6.144)) and the (weak) Condorcet winner is $i^* = 1$. Finally, since in each row the negative entries are constant, we have $s^*_M = (s_1, \dots, s_K)$ and $M \in \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$.

We use the same permutation construction as in the proof of Theorem 6.4.2. We recall this construction here. Let Π be the set of permutations, where $\pi = (\pi_1, \dots, \pi_d) \in \Pi$ if π_i is a permutation of $\{1, \dots, d\}$ for any $i \in [d]$.

From any $\pi \in \Pi$, define M^π as the matrix obtained by permuting the d first columns of D according to π in the following way:

$$M^\pi = \begin{pmatrix} \mathbf{0} & -D^\pi \\ (D^\pi)^\top & \Lambda \end{pmatrix} \ , \tag{6.153}$$

where, for any $(i, j) \in [d]^2$,

$$D^\pi_{i,j} = D_{i, \pi_i(j)} \ .$$

By construction, for any $\pi \in \Pi$, we still have $M^\pi \in \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$.

Alternative instance $M^{(\pi,k)}$. Fix a suboptimal arm $k \in \{2, \dots, d\}$.

Construct the gap matrix $M^{(\pi,k)}$ by setting to zero all entries in the k-th row and the k-th column of M^π. By construction, rows 1 and k of $M^{(\pi,k)}$ only scontain zero entries, so that $M^{(\pi,k)}$ does not admit a unique Condorcet winner. We denote by $\mathbb{P}^{(\pi,k)}_A$ the distribution of the observations induced by the interaction between algorithm A and the environment with gap matrix $M^{(\pi,k)}$.

Step 2: information-theoretic arguments.

Consider the recommendation rule $\hat{\imath}$ and the budget T of algorithm A. By definition of ϵ_T, A can be considered as ϵ_T-correct over $\mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$.

Denote again by $\hat{\imath}$ the recommendation of algorithm A. Observe that we always have $\{\hat{\imath} \neq 1\}$ or $\{\hat{\imath} \neq k\}$. Therefore,

$$\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_A^{(\pi,k)}(\hat{\imath}\neq 1)\geqslant\frac{1}{2}\quad\text{or}\quad\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_A^{(\pi,k)}(\hat{\imath}\neq k)\geqslant\frac{1}{2}.$$

Without loss of generality, we assume that

$$\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_A^{(\pi,k)}(\hat{\imath}\neq 1)\geqslant\frac{1}{2}\ . \tag{6.154}$$

In the other case, we should consider as reference matrix the matrix $\tilde{M}$ obtained by exchanging rows 1 and k of M so that $i^*(\tilde{M}) = k$. Observe that we still have $\tilde{M} \in \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$. The rest of the proof is the same up to minor modifications.

Consider the event

$$B := \{\hat{\imath} \neq 1\}\ .$$

For any π, we have $M^\pi \in \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$ and $i^*(M^\pi) = 1$. We can then use the fact that A is ϵ_T-correct over this class, by definition of ϵ_T ((6.147)), to get

$$\mathbb{P}_A^\pi(B) = \mathbb{P}_A^\pi(\hat{\imath}\neq 1)\leqslant\epsilon_T\ . \tag{6.155}$$

Now we use Lemma 6.F.4, a Fano-type inequality presented as Proposition 4 in Gerchinovitz et al. (2020), to obtain

$$\mathbb{P}_A^{(\pi,k)}(B)\leqslant\frac{\mathrm{KL}(\mathbb{P}_A^{(\pi,k)},\mathbb{P}_A^\pi)+\log(2)}{-\log(\mathbb{P}_A^\pi(B))}\ .$$

Averaging over $\pi \in \Pi$ and using (6.154) and (6.155), we get

$$\frac{1}{2}\leq\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_A^{(\pi,k)}(B)\leqslant\frac{\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathrm{KL}(\mathbb{P}_A^{(\pi,k)},\mathbb{P}_A^\pi)+\log(2)}{\log(1/\epsilon_T)}\ .$$

Observe that, for any $\pi \in \Pi$, M^π and $M^{(\pi,k)}$ differ only in row k, and that that $M^{(\pi,k)}$ does not depend on the permutation π_k. Denote as $\pi^{(-k)}$ the vector of permutations obtained from $\pi = (\pi_1, \ldots, \pi_d) \in \Pi$ by removing the k-th component– $\pi^{(-k)} = (\pi_1, \ldots, \pi_{k-1}, \pi_{k+1}, \ldots, \pi_d)$. Denote as $\Pi^{(-k)}$ as the family $\{\pi^{(-k)}\}_{\pi\in\Pi}$. Observe that $M^{(\pi,k)}$ does not depend on π_k. For a fixed $\pi^{(-k)} \in \Pi^{(-k)}$, we have $M^{(\pi,k)} = M^{(\pi^{(-k)},k)}$. Then, we write the inequality above as

$$\frac{1}{2} \leqslant \frac{\frac{1}{|\Pi^{(-k)}|}\sum_{\pi^{(-k)}\in\Pi^{(-k)}} \frac{1}{|\mathfrak{S}_d|}\sum_{\pi'_k\in\mathfrak{S}_d} \mathrm{KL}\big(\mathbb{P}_A^{(\pi^{(-k)},k)}, \mathbb{P}_A^{\pi'}\big) + \log(2)}{\log(1/\epsilon_T)} \ , \tag{6.156}$$

where inside the sum, we denote as π' for the permutation obtained from $\pi^{(k)}$ and π'_k by $\pi' = (\pi_1, \dots, \pi_{k-1}, \pi'_k, \pi_{k+1}, \dots, \pi_d)$.

Step 3: computing the KL divergence. We now bound, for a fixed $\pi^{(-k)} \in \Pi^{(-k)}$,

$$\frac{1}{|\mathfrak{S}_d|}\sum_{\pi'_k\in\mathfrak{S}_d} \mathrm{KL}\big(\mathbb{P}_A^{(\pi^{(-k)},k)}, \mathbb{P}_A^{\pi'}\big).$$

This computation has already been carried out in the proof of Theorem 6.4.2, see Equation (6.118), and one has

$$\frac{1}{|\mathfrak{S}_d|}\sum_{\pi_k\in\mathfrak{S}_d} \mathrm{KL}\big(\mathbb{P}_A^{(\pi^{(-k)},k)}, \mathbb{P}_A^{\pi'}\big) \leqslant 8\log\left(\frac{4}{3}\right)\frac{\|M_{k,\cdot}\|^2}{d}T \ , \tag{6.157}$$

where, by construction of M, we have $\|M_{k,\cdot}\|^2 = s_k\Delta_k^2$.

Averaging over $\Pi^{(-k)}$ (6.157), and combining Equation (6.156), we get

$$T \geqslant \frac{1}{16\log(4/3)}\frac{d}{s_k\Delta_k^2}\log\left(\frac{1}{4\epsilon_T}\right) \ ,$$

which holds for any $k \in \{2, \dots, K\}$. This is exactly the desired bound (6.148). □

Proof of Lemma 6.E.3. Assume additionally that there exist $\mu > 0$ and $s \in [d]$ such that, for every $i \in \{2, \dots, d\}$, $\Delta_i = \mu$ and $s_i = s$. We want to prove Bound (6.149), that is

$$\epsilon_T \geqslant \frac{1}{2} - \sqrt{128\log(4/3)\frac{s\mu^2}{K^2}T} \ .$$

Step 1: reference and alternative instances.

Consider $\bar{M}$ as the matrix defined by

$$\bar{M} = \begin{pmatrix} \mathbf{0} & -\bar{D} \\ \bar{D}^\top & \Lambda \end{pmatrix} \ , \tag{6.158}$$

where Λ is as in (6.152), and $\bar{D}$ is the $d \times d$ matrix such that, for any $i = 1, \dots, d$,

$$\bar{D}_{i,\cdot} = (\underbrace{\mu, \dots, \mu}_{s \text{ times}}, 0, \dots, 0) \in \mathbb{R}^d \ .$$

Observe that the first d rows of $\bar{M}$ are equal, and that $\bar{M}$ does not admit a Condorcet winner; in particular, $\bar{M} \notin \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$.

Again, for any $\pi \in \Pi$, define $\bar{M}^\pi$ as the matrix obtained by permuting the d first rows of $\bar{M}$ according to $\pi_1, \dots, \pi_d$ as in (6.153). For this part of the proof, we denote by $\mathbb{P}_A^\pi$ the distribution of the observations induced by the interaction between algorithm A and the environment with gap matrix $\bar{M}^\pi$.

Construction of perturbed instances $\bar{M}^{(\pi,k)}$. Consider any arm $k \in [d]$.

We construct $\bar{M}^{(\pi,k)}$ as the matrix obtained from $\bar{M}^\pi$ by setting to zero all entries in the k-th row and the k-th column. By construction, $\bar{M}^{(\pi,k)} \in \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$ and $i^*(\bar{M}^{(\pi,k)}) = k$. We denote by $\mathbb{P}_A^{(\pi,k)}$ the distribution of the observations induced by the interaction between algorithm A and the environment with gap matrix $\bar{M}^{(\pi,k)}$.

Step 2: bound on the total variation distance.

Consider the recommendation rule $\hat{\imath}$ and the budget T of algorithm A, which is ϵ_T-correct over $\mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$, by definition of the maximum error ϵ_T.

Intuitively, under $\bar{M}$ there is no Condorcet winner among the first d arms, so algorithm A cannot systematically decide in favour of a specific subset of them, and it must make a large error on at least half of these arms. Indeed, it always holds that $\{\hat{\imath} \notin [\lfloor 1; d/2 \rfloor]\}$ or $\{\hat{\imath} \notin [\lfloor d/2+1; d \rfloor]\}$. Therefore,

$$\frac{1}{|\Pi|}\sum_{\pi\in\Pi} \mathbb{P}_A^\pi(\hat{\imath} \notin [1; d/2]) \geqslant \frac{1}{2} \quad \text{or} \quad \frac{1}{|\Pi|}\sum_{\pi\in\Pi} \mathbb{P}_A^\pi(\hat{\imath} \notin [d/2+1; d]) \geqslant \frac{1}{2}.$$

Without loss of generality [9] we assume that

$$\frac{1}{|\Pi|}\sum_{\pi\in\Pi} \mathbb{P}_A^\pi(\hat{\imath} \notin [1; d/2]) \geqslant \frac{1}{2}\ . \tag{6.159}$$

For any $k \in [d/2]$, consider the event

$$B_k \coloneqq \{\hat{\imath} = k\} \cup \{N_{\{k,\cdot\}} > \tfrac{16T}{K}\}\ ,$$

$N_{\{k,\cdot\}}$ denotes the number of duels involving arm k and an adversary in $[d+1; K]$ between time $t = 1$ and time T, that is,

$$N_{\{k,\cdot\}} = |\Big\{t \in [T]\ :\ \exists j \in [d+1; K] \text{ with } \{I_t, J_t\} = \{k, j\}\Big\}|\ .$$

Observe first that, for any fixed π, $\mathbb{P}_A^\pi$ does not depend on k, so that

$$\frac{1}{d/2}\sum_{k=1}^{d/2} \mathbb{P}_A^\pi(\hat{\imath} = k) = \frac{1}{d/2}\,\mathbb{P}_A^\pi(\hat{\imath} \in [1; d/2])\ .$$

Averaging over $\pi \in \Pi$ and using (6.159), we obtain

$$\frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi} \mathbb{P}_A^\pi(\hat{\imath} = k) \leqslant \frac{1}{d}\ . \tag{6.160}$$

9. in the other case, we consider $k \in [d/2+1, d]$ and run the same arguments

Now, by definition, the family $\big(N_{\{k,\cdot\}}\big)_{k\in[d/2]}$ counts duels with pairwise disjoint sets of arms, for time-steps between 1 and T, so that

$$\sum_{k=1}^{d/2} N_{\{k,\cdot\}} \leqslant T \ .$$

From this upper bound, a simple counting argument implies that at most a fraction $1/4$ of the arms in $[d/2]$ can satisfy $N_{\{k,\cdot\}} > \frac{16T}{K} = \frac{4T}{d/2}$. Hence,

$$\frac{1}{d/2}\sum_{k=1}^{d/2} \mathbb{1}_{\{N_{\{k,\cdot\}} > \frac{16T}{K}\}} \leqslant \frac{1}{4} \ .$$

Taking expectation with respect to the probability $\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_A^\pi$, we obtain

$$\frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_A^\pi\Big(N_{\{k,\cdot\}} > \tfrac{16T}{K}\Big) \leqslant \frac{1}{4} \ . \tag{6.161}$$

Combining (6.160) and (6.161), and using $d \geqslant 4$, we get

$$\frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_A^\pi(B_k) \leqslant \frac{1}{4} + \frac{1}{d} \leqslant \frac{1}{2} \ . \tag{6.162}$$

Now, consider B_k under $\mathbb{P}_A^{(\pi,k)}$. Observe that $B_k^c \subset \{\hat{\imath} \neq k\}$. The environment $\bar{M}^{(\pi,k)}$ admits k as Condorcet winner and belongs to $\mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$. Using that A is ϵ_T-correct over this class, by definition of ϵ_T, we obtain, for any $\pi \in \Pi$,

$$\mathbb{P}_A^{(\pi,k)}(B_k^c) \leqslant \mathbb{P}_A^{(\pi,k)}(\hat{\imath} \neq k) \leqslant \epsilon_T \, . \tag{6.163}$$

The event B_k has the additional property that it is measurable by an algorithm which runs A but uses at most $\frac{16T}{K}$ duels involving arm k. Define the following procedure $\tilde{A}_k$. For $t = 1, \dots, T$, run algorithm A. At each time t, compute $N_{\{k,\cdot\}}(t)$ as the number of duels involving k before time t, if $N_{\{k,\cdot\}}(t) > 16T/K$, stop sampling, and return $\psi_k = 1$. If the algorithm has not stopped by time T –that is, if $N_{\{k,\cdot\}} = N_{\{k,\cdot\}}(T) \leqslant 16T/K$ – compute $\hat{\imath}_T$ and output $\psi_k \coloneqq \mathbb{1}_{\hat{\imath}_T = k}$.

By construction, the decision ψ_k produced by $\tilde{A}_k$ satisfies $\psi_k = \mathbb{1}_{B_k}$. Moreover, for any envi-

ronment ν, we have $\mathbb{P}_{\tilde{A}_k,\nu}(B_k) = \mathbb{P}_{A,\nu}(B_k)$. From these observations, we deduce

$$\begin{aligned}
&\mathrm{TV}\left(\frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_{\tilde{A}_k}^{(\pi,k)}, \frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_{\tilde{A}_k}^{\pi}\right) \\
&\quad\geqslant \frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_{\tilde{A}_k}^{(\pi,k)}(B_k) - \frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_{\tilde{A}_k}^{\pi}(B_k) \\
&\quad= \frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_{A}^{(\pi,k)}(B_k) - \frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_{A}^{\pi}(B_k)\ .
\end{aligned}$$

Using (6.162) and (6.163), we obtain

$$\mathrm{TV}\left(\frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_{\tilde{A}}^{(\pi,k)}, \frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_{\tilde{A}}^{\pi}\right) \geqslant 1 - \epsilon_T - \frac{1}{2} = \frac{1}{2} - \epsilon_T \tag{6.164}$$

Finally, using the convexity of the total variation distance together with Pinsker's inequality and (6.164), we get

$$\begin{aligned}
\frac{1}{2} - \epsilon_T &\leqslant \mathrm{TV}\left(\frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_{\tilde{A}_k}^{(\pi,k)}, \frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathbb{P}_{\tilde{A}_k}^{\pi}\right) \\
&\leqslant \sqrt{\frac{1}{2}\frac{1}{d/2}\sum_{k=1}^{d/2}\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathrm{KL}\big(\mathbb{P}_{\tilde{A}_k}^{(\pi,k)}, \mathbb{P}_{\tilde{A}_k}^{\pi}\big)}\ .
\end{aligned} \tag{6.165}$$

Step 3: computing the KL divergence.

An important property of procedure $\tilde{A}_k$ is that the budget spent on duels with arm k is upper bounded by $16T/K$; precisely, for any environment ν,

$$\mathbb{E}_{\tilde{A}_k,\nu}\left[\sum_{i=d/2}^{d} N_{\{k,i\}}\right] \leqslant \frac{16T}{K}\ . \tag{6.166}$$

We now upper bound

$$\frac{1}{|\Pi|}\sum_{\pi\in\Pi}\mathrm{KL}\big(\mathbb{P}_{\tilde{A}_k}^{(\pi,k)}, \mathbb{P}_{\tilde{A}_k}^{\pi}\big).$$

As in previous proofs, we fix $\pi_1, \ldots, \pi_{k-1}, \pi_{k+1}, \ldots$ and we average over the π_k's. Hence, for any

$k \in [d/2]$, we get

$$\frac{1}{|\mathfrak{S}_d|} \sum_{\pi_k \in \mathfrak{S}_d} \mathrm{KL}(\mathbb{P}^{(\pi,k)}_{\bar{A}_k}, \mathbb{P}^{\pi}_{\bar{A}_k}) \leqslant 8 \log\left(\frac{4}{3}\right) \frac{\|\bar{M}_{k,\cdot}\|^2}{d} \mathbb{E}^{(\pi,k)}_{\bar{A}_k}\left[\sum_{i=d/2}^{d} N_{\{k,i\}}\right]$$
$$\leqslant 8 \log\left(\frac{4}{3}\right) \frac{s\mu^2}{d} \frac{16T}{K} \ ,$$

where we use $\|\bar{M}_{k,\cdot}\|^2 = s\mu^2$ and Equation (6.166).

Gathering Equation (6.165) with the bound above, and rearranging (using $d = K/2$), we obtain

$$\epsilon_T \geqslant \frac{1}{2} - \sqrt{128 \log(4/3) \frac{s\mu^2}{K^2} T} \ ,$$

which is exactly the desired bound (6.149). □

Proof of Lemma 6.E.4. Consider again the constant case where there exist $\mu > 0$ and $s \in [d]$ such that, for every $i \in \{1, \dots, d\}$, $\Delta_i = \mu$ and $s_i = s$. We want to prove the following bound, equivalent to (6.150):

$$T \geqslant \frac{1}{32 \log(4/3)} \frac{K}{\mu^2} \log\left(\frac{1}{4\epsilon_T}\right) .$$

Step 1. Again, assume that $\underline{\Delta}$ and $\underline{s}$ are constant. Take the matrix $\bar{M}$ defined in (6.158).

Fix for now $k \in \{1, \dots, d/2\}$. Consider $\bar{M}^{(k)}$, the matrix obtained from $\bar{M}$ by setting to zero the k-th row and the k-th column of $\bar{M}$.

Recall that $\bar{M}$ does not admit a Condorcet winner, while $\bar{M}^{(k)} \in \mathbb{D}^{(3)}(\underline{\Delta}, \underline{s})$ with $i^*(\bar{M}^{(k)}) = k$. We denote by $\mathbb{P}_A$ (resp. $\mathbb{P}^{(k)}_A$) the distribution of the observations induced by the interaction between algorithm A and the environment with gap matrix $\bar{M}$ (resp. $\bar{M}^{(k)}$).

Step 2: bound on the total variation distance.

Consider the event

$$B \coloneqq \{\hat{\imath} \in [d/2]\} \ .$$

As in the proof of (6.149), we can assume without loss of generality [10] that $\mathbb{P}_A(\hat{\imath} \notin [d/2]) \geqslant \frac{1}{2}$, so that

$$\mathbb{P}_A(B) \leqslant \mathbb{P}_A(\hat{\imath} \in [d/2]) \leqslant \frac{1}{2} \ . \tag{6.167}$$

Now, consider B under $\mathbb{P}^{(k)}_A$. Observe that $B^c \subset \{\hat{\imath} \neq k\}$. By definition of the maximum error ϵ_T, A is ϵ_T-correct over $\mathbb{D}^{(3)}$, so that

$$\mathbb{P}^{(k)}_A(B^c) \leqslant \mathbb{P}^{(k)}_A(\hat{\imath} \neq k) \leqslant \epsilon_T \, . \tag{6.168}$$

10. Otherwise, choose $B = \{\hat{\imath} \in [d/2; d]\}$, and take k in $[d/2; d]$ everywhere.

Now, by the Fano-type inequality from Lemma 6.F.4, it holds that

$$\mathbb{P}_A(B^c) \leqslant \frac{\mathrm{KL}(\mathbb{P}_A, \mathbb{P}_A^{(k)}) + \log(2)}{-\log(\mathbb{P}_A^{(k)}(B^c))} \ ,$$

and using (6.167) and (6.168), we obtain

$$\frac{1}{2} \leqslant \mathbb{P}_A(B^c) \leqslant \frac{\mathrm{KL}(\mathbb{P}_A, \mathbb{P}_A^{(k)}) + \log(2)}{-\log(\mathbb{P}_A^{(k)}(B^c))} \leqslant \frac{\mathrm{KL}(\mathbb{P}_A, \mathbb{P}_A^{(k)}) + \log(2)}{-\log(\epsilon_T)} \ . \tag{6.169}$$

Step 3: computing the KL divergence.

We now upper bound $\mathrm{KL}\left(\mathbb{P}_A, \mathbb{P}_A^{(k)}\right)$.

From the decomposition of the KL divergence and the definition of $\bar{M}^{(k)}$, we have

$$\begin{aligned}\mathrm{KL}\left(\mathbb{P}_A, \mathbb{P}_A^{(k)}\right) &= \sum_{i=d/2}^{d} \mathbb{E}_A[N_{\{k,i\}}]\, \mathrm{kl}\Big(\tfrac{1}{2} + \bar{M}_{k,i}, \tfrac{1}{2}\Big) \\ &\leqslant \sum_{i=d/2}^{d} \mathbb{E}_A[N_{\{k,i\}}]\, 8\log(4/3)\, \mu^2 \ ,\end{aligned}$$

where we use the fact that the row $\bar{M}_{k,\cdot}$ only takes values in $\{0, \mu\}$.

Combining (6.169) with the bound above and rearranging, we obtain

$$\begin{aligned}\frac{1}{16\log(4/3)}\, \frac{1}{\mu^2}\, \log\Big(\frac{1}{4\epsilon_T}\Big) &\leqslant \frac{1}{8\log(4/3)}\, \frac{1}{\mu^2}\, \mathrm{KL}\left(\mathbb{P}_A, \mathbb{P}_A^{(k)}\right) \\ &\leqslant \sum_{i=d/2}^{d} \mathbb{E}_A[N_{\{k,i\}}] \ .\end{aligned}$$

Summing over $k \in [d/2]$, we obtain

$$\frac{K}{32\log(4/3)}\, \frac{1}{\mu^2}\, \log\Big(\frac{1}{4\epsilon_T}\Big) \leqslant \sum_{k=1}^{d/2} \sum_{i=d/2}^{d} \mathbb{E}_A[N_{\{k,i\}}] \ \leqslant T \ ,$$

which is the desired bound (6.150). □

6.F Technical Results

6.F.1 Deterministic bounds

Lemma 6.F.1 (Section 6.1 of Audibert and Bubeck (2010))**.** *Let $x_1, \dots, x_K$ denote a decreasing sequence of positive numbers. We have:*

$$\max_{k\in\{1,\dots,K\}} kx_k^2 \leq \sum_{i=1}^{K} x_i^2 \leq \log(4K) \max_{k\in\{1,\dots,K\}} kx_k^2.$$

Lemma 6.F.2. *Let $x_1, \dots x_n$ denote a sequence of positive numbers such that $x_1 \leq \cdots \leq x_n$. Then we have for any $p \in (0,1)$*

$$\frac{pn}{x_{(\lceil * \rceil pn)}} \leq \sum_{i=1}^{n} \frac{1}{x_i} \ .$$

Proof. We have $\frac{1}{x_n} \leq \cdots \leq \frac{1}{x_1}$. Therefore

$$\frac{pn}{x_{\lceil * \rceil pn}} \leq \frac{\lceil * \rceil \, pn}{x_{\lceil * \rceil pn}} \leq \sum_{i=1}^{\lceil * \rceil pn} \frac{1}{x_i} \leq \sum_{i=1}^{n} \frac{1}{x_i} \ .$$

□

Lemma 6.F.3. *Let $\mathfrak{S}_d$ denote the set of permutations of $\{1, \dots, d\}$. Let $a_1, \dots, a_d$ and $b_1, \dots, b_d$ be two sequences of numbers. We have*

$$\frac{1}{d!} \sum_{\sigma\in\mathfrak{S}_d} \sum_{i=1}^{d} a_i b_{\sigma(i)} = \frac{1}{d} \left(\sum_{i=1}^{d} a_i\right) \left(\sum_{i=1}^{d} b_i\right) \ .$$

Proof. The result is just a consequence of summation manipulation. We have

$$\sum_{\sigma} \sum_{i=1}^{d} a_i b_{\sigma(i)} = \sum_{i=1}^{d} a_i \sum_{\sigma} b_{\sigma(i)} = \sum_{i=1}^{d} a_i \frac{d!}{d} \sum_{i=1}^{d} b_i = (d-1)! \left(\sum_{i=1}^{d} a_i\right) \left(\sum_{i=1}^{d} b_i\right) \ .$$

□

Below we present a useful Fano-type inequality presented as Proposition 4 in Gerchinovitz et al. (2020)

Lemma 6.F.4. *Let $\mathbb{P}$ and $\mathbb{Q}$ be two probability distributions, and let A be an event such that $\mathbb{Q}(A) \in (0,1)$. We have*

$$\mathbb{P}(A) \leq \frac{\mathrm{KL}(\mathbb{P}, \mathbb{Q}) + \log(2)}{-\log(\mathbb{Q}(A))} \ .$$

More generally, for all probability pairs $\mathbb{P}_i, \mathbb{Q}_i$ and all events A_i, where $i \in \{1, \dots, N\}$, with

$0 < \frac{1}{N}\sum_{i=1}^{N} \mathbb{Q}_i(A_i) < 1$, *we have*

$$\frac{1}{N}\sum_{i=1}^{N} \mathbb{P}_i(A_i) \leq \frac{\frac{1}{N}\sum_{i=1}^{N} \mathrm{KL}\left(\mathbb{P}_i, \mathbb{Q}_i\right) + \log(2)}{-\log\left(\frac{1}{N}\sum_{i=1}^{N} \mathbb{Q}_i(A_i)\right)} \ .$$

Lemma 6.F.5. *For any $x > 0$, we have*

$$0 < \frac{\frac{1}{2} - e^{-x}}{\log(1-e^{-x}) - \log(e^{-x})} < \frac{1}{2x} \, .$$

Proof. Let $y = e^{-x} \in (0,1)$ and define

$$h(y) := \log\Big(\frac{1-y}{y}\Big) = \log(1-y) - \log y, \qquad R(y) := \frac{\frac{1}{2} - y}{h(y)} \quad (y \neq \tfrac{1}{2}) \ .$$

Let us start with a proof of the positivity of the middle expression. The function h is strictly decreasing on $(0,1)$ and satisfies $h(\frac{1}{2}) = 0$, hence $\mathrm{sign}(h(y)) = \mathrm{sign}(\frac{1}{2} - y)$. Therefore $R(y) > 0$ for all $y \neq \frac{1}{2}$. At $y = \frac{1}{2}$, both numerator and denominator vanish, by l'Hôpital's rule,

$$\lim_{y \to 1/2} R(y) = \frac{-1}{h'(1/2)} = \frac{-1}{-(\frac{1}{1-y} + \frac{1}{y})\big|_{y=1/2}} = \frac{1}{4} \ d.$$

Thus the middle expression is well-defined by continuity and is strictly positive for all $x > 0$.

Now let us sprove the stated upper bound. Since $x = \log(1/y) > 0$, we need to show

$$R(y) < \frac{1}{2\log(1/y)} \ .$$

If $y < \frac{1}{2}$ (so $h(y) > 0$) we need to prove that

$$2\log(1/y)\Big(\tfrac{1}{2} - y\Big) < h(y) \ .$$

which is equivalent to $2y\log(1/y) + \log(1-y) > 0$, define $g(y) :- 2y\log(1/y) + \log(1-y)$, therefore we need to show that

$$\forall y \in (0, 1/2), \quad g(y) > 0 \ .$$

If $y > \frac{1}{2}$ (so $h(y) < 0$), the same manipulation yields the equivalent condition $g(y) < 0$. Hence it suffices to prove $g(y) > 0$ on $(0, \frac{1}{2})$ and $g(y) < 0$ on $(\frac{1}{2}, 1)$.

A direct computation shows

$$g''(y) = -\frac{2}{y} - \frac{1}{(1-y)^2} < 0 \qquad (y \in (0,1)),$$

so g is strictly concave. Moreover,

$$\lim_{y \downarrow 0} g(y) = 0 \quad \text{and} \quad g(1/2) = 2 \cdot \frac{1}{2} \log 2 + \log(1/2) = 0.$$

By strict concavity, this implies $g(y) > 0$ for all $y \in (0, 1/2)$. Finally,

$$g'(y) = 2\log(1/y) - 2 - \frac{1}{1-y}, \qquad g'(1/2) = 2\log 2 - 4 < 0.$$

Since g is concave, g' is non-increasing, so $g'(y) \leq g'(1/2) < 0$ for all $y \geq 1/2$. Thus, g is strictly decreasing on $[1/2, 1)$, and hence $g(y) < 0$ for all $y \in (1/2, 1)$.
Therefore, $R(y) < 1/(2\log(1/y)) = 1/(2x)$ for all $x > 0$, concluding the proof. □

Lemma 6.F.6. *Let d be an integer greater than 1. Let $M \in \mathbb{R}^{d \times d}$ such that M is skew symmetric (i.e., $\forall i, j \in [d] : M_{i,j} = -M_{j,i}$). Then, the number of lines of M with at least $\lceil * \rceil\, (d+1)/4$ non-positive entries is at least $\lceil * \rceil\, (d+1)/4$.*

Proof. For $i \in [d]$ define

$$s_i := |\{j \in [d] : \ M_{i,j} \leq 0\}| \ .$$

Since the matrix M is skew symmetric (i.e. $M_{i,j} = -M_{j,i}$ for any i, j), for every unordered pair $\{i, j\}$ where $i \neq j$, at least one of the two quantities $M_{i,j}$ or $M_{j,i}$ is nonpositive. Therefore, the number of off-diagonal non-positive entries is at least $\binom{d}{2}$. We conclude by taking into account the diagonal entries that

$$\sum_{i=1}^{d} s_i \geq \binom{d}{2} + d = \frac{d(d+1)}{2} \ .$$

The conclusion follows by a simple contradiction argument. □

Lemma 6.F.7. *Let $n \geq 3$ and M be an $n \times n$ skew-symmetric matrix, let E denote the set of rows such that the number of non-positive entries is at least $\lceil n/4 \rceil + 1$, then the intersection of E with any subset of $\{1, \ldots, n\}$ of cardinality $n - \lceil n/8 \rceil$ is non-empty.*

Proof. Assume $n \geq 3$ and set $a := \lceil n/8 \rceil$, $b := \lceil n/4 \rceil$. Suppose by contradiction that there exists $S \subseteq [n]$ with $|S| = n - a$ and $S \cap E = \varnothing$. Then each row $i \in S$ has at most b non-positive entries, hence at least $n - b$ positive entries. Let $P := |\{(i, j) : M_{ij} > 0\}|$ be the total number of positive entries. Summing over rows in S gives

$$P \ \geq \ |S|\,(n-b) = (n-a)(n-b).$$

On the other hand, skew-symmetry implies that for each unordered pair $\{i, j\}$ with $i \neq j$, at most one of M_{ij}, M_{ji} is positive, hence

$$P \ \leq \ \binom{n}{2} = \frac{n(n-1)}{2}.$$

Thus $(n-a)(n-b) \le \frac{n(n-1)}{2}$. But for $n \ge 5$, using $\lceil x \rceil \le x+1$,

$$n-a \ge \frac{7(n-1)}{8}, \qquad n-b \ge \frac{3(n-1)}{4} \quad \Rightarrow \quad (n-a)(n-b) \ge \frac{21}{32}(n-1)^2 > \frac{n(n-1)}{2},$$

a contradiction; and the remaining cases $n = 3, 4$ are checked directly: $(n-a)(n-b) = 4 > 3$ and $9 > 6$, respectively. Hence no such S exists, i.e. every S with $|S| = n - \lceil n/8 \rceil$ intersects E. □

Lemma 6.F.8. *Let $x > 10^3$ and $y > 8$. Then*

$$2\,\frac{\log^2(x)\,\log^2(y)}{\log\bigl(xy(\log y)^5\bigr)} \;\ge\; \log(xy).$$

Proof. Since $x > 10^3$ and $y > 8$, we have $\log x > \log(10^3) > 6$ and $\log y > \log 8 > 2$. Hence

$$\log x\,\log y - (\log x + \log y) = (\log x - 1)(\log y - 1) - 1 > 0\ ,$$

so

$$\log x\,\log y \ge \log(xy)\ . \tag{6.170}$$

Moreover,

$$\log\bigl(xy(\log y)^5\bigr) = \log(xy) + 5\log\log y.$$

Since $y > 8$ implies $\log y > 2 > 1$, we have $\log\log y \le \log y$, and thus

$$\log\bigl(xy(\log y)^5\bigr) \le \log(xy) + 5\log y = \log x + 6\log y.$$

Therefore,

$$\begin{aligned} 2\log x\,\log y - (\log x + 6\log y) &= (2\log y - 1)\log x - 6\log y \\ &\ge 6(2\log y - 1) - 6\log y \\ &= 6(\log y - 1) > 0\ , \end{aligned}$$

so

$$2\log x\,\log y \;\ge\; \log\bigl(xy(\log y)^5\bigr). \tag{6.171}$$

Multiplying (6.170) and (6.171) yields

$$2\log^2(x)\,\log^2(y) \;\ge\; \log(xy)\,\log\bigl(xy(\log y)^5\bigr).$$

Since $\log\bigl(xy(\log y)^5\bigr) > 0$, dividing by it gives the claim. □

Lemma 6.F.9. *Let $K \ge 2$, $H \ge 4$, and $T \ge 8K\log_{8/7}(K)$. Define*

$$p_k := \frac{1}{18} \wedge (\log T + K)\exp\Bigl(-c\,\frac{T}{\log^3(K)\log(T)\,H}\Bigr).$$

Assume $T \geq c_0\, H \log^5(H)$ for some numerical constant c_0 large enough (depending only on c). Then there exists a numerical constant $c' > 0$ (e.g. $c' = c/2$) such that

$$p_k \leq \exp\Big(-c' \frac{T}{\log^3(K)\log(T)\, H}\Big).$$

Proof. Let us introduce the following notation

$$x \coloneqq \frac{T}{\log^3(K)\log(T)\, H}, \qquad A \coloneqq \log T + K\ .$$

Then $p_k \leq Ae^{-cx}$. Since $T \geq 8K\log_{8/7}(K)$ and $K \geq 2$, we have $A = \log T + K \leq T$, so $\log A \leq \log T$ and

$$p_k \leq \exp(-cx + \log T).$$

Thus it suffices to prove $\log T \leq \frac{c}{2}x$, i.e.

$$\frac{T}{(\log T)^2} \ \geq\ \frac{2H\log^3(K)}{c}. \tag{6.172}$$

Since $T \geq 8K\log_{8/7}(K) > K$, we have $\log^3(K) \leq (\log T)^3$. Therefore (6.172) follows from

$$\frac{T}{(\log T)^2} \ \geq\ \frac{2H}{c}(\log T)^3\ ,$$

which is equivalent to

$$\frac{T}{(\log T)^5} \ \geq\ \frac{2H}{c}.$$

Let $g(t) \coloneqq t/(\log t)^5$, which is increasing for $t \geq e^5$. Choose c_0 large enough so that $T \geq c_0 H\log^5(H) \geq e^5$ for all $H \geq 4$. Then, with $T_0 \coloneqq c_0 H\log^5(H)$, we have $g(T) \geq g(T_0)$ and

$$g(T_0) = \frac{c_0 H\log^5(H)}{\big(\log(c_0 H\log^5(H))\big)^5}.$$

For $H \geq 4$, we have $\log\log H \leq \log H$, so

$$\begin{aligned}\log(c_0 H\log^5(H)) &= \log c_0 + \log H + 5\log\log H \leq \log c_0 + 6\log H\\ &\leq \Big(6 + \frac{\log c_0}{\log 4}\Big)\log H =: C_0\log H\ .\end{aligned}$$

Hence $g(T_0) \geq \frac{c_0}{C_0^5}H$. Taking c_0 large enough so that $\frac{c_0}{C_0^5} \geq \frac{2}{c}$ yields $g(T) \geq g(T_0) \geq \frac{2H}{c}$, proving (6.172). Therefore $p_k \leq \exp(-(c/2)x)$, i.e. the claim with $c' = c/2$. □

6.F.2 Concentration inequalities

Below is Hoeffding concentration inequality.

Lemma 6.F.10. *Let $X_1, \dots, X_n$ be independent random variables such that $a_i \leq X_i \leq b_i$ almost surely. Let $S_n = \sum_{i=1}^n X_i$. Then we have for all $t > 0$:*

$$\mathbb{P}\left(S_n - \mathbb{E}[S_n] \leq -t\right) \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^n (b_i - a_i)^2}\right).$$

Below we restate two results on the concentration of the sum of independent binary random variable from Buldygin and Moskvichova (2013). First, let us introduce some notation. Let ξ denote a sub-Gaussian random variable, its sub-Gaussian standard is defined by:

$$\tau(\xi) := \inf\left\{a \geq 0 : \mathbb{E}\left[\exp\left(\lambda\xi\right)\right] \leq \exp\left(\frac{a^2\lambda^2}{2}\right), \lambda \in \mathbb{R}\right\}.$$

Lemma 6.F.11 (Theorem 2.1 in Buldygin and Moskvichova (2013))**.** *Let X denote a Bernoulli random variable with parameter $p \in [0,1]$. Then we have*

$$\tau^2\left(X - p\right) = \phi(p),$$

where $\phi(.)$ is the function defined by:

$$\phi(p) = \begin{cases} 0, & p \in \{0,1\}; \\ \frac{1}{4}, & p = \frac{1}{2}; \\ \frac{\frac{1}{2}-p}{\log(1-p)-\log(p)}, & p \in (0,1) \setminus \left\{\frac{1}{2}\right\}. \end{cases}$$

The lemma below gives a concentration bound on the binomial random variables.

Lemma 6.F.12. *Let X_j for $j \in \{1, \dots, n\}$ denote a sequence of independent Bernoulli random variables with parameter $p \in [0,1]$. Define $S_n = \sum_{j=1}^n (X_j - p)$. Then, we have for all $x > 0$*

$$\mathbb{P}\left(S_n \geq x\right) \leq \exp\left(\frac{-x^2}{2n\phi(p)}\right),$$

where ϕ is defined in Lemma 6.F.11. We also have for all $x > 0$

$$\mathbb{P}\left(S_n \leq -x\right) \leq \exp\left(\frac{-x^2}{2n\phi(p)}\right).$$

Proof. This is a direct consequence of Chernoff's bound with Lemma 6.F.11. □

Chapter 7

The Sample Complexity of Multiple Change Point Identification under Bandit Feedback

Abstract. *We study multiple change point localization under bandit feedback. An unknown piecewise-constant function on a compact interval can be queried sequentially at adaptively chosen inputs, and each query returns a noisy evaluation of the function. The goal is to identify a prescribed number of discontinuities, known as change points, within a target precision η and confidence level $1-\delta$, while using as few samples as possible. We propose an adaptive algorithm that first detects intervals likely to contain change points and then refines their locations to precision η. We establish non-asymptotic upper bounds on its sample budget, together with corresponding lower bounds. Prior work shows that jump magnitudes alone determine the asymptotic sample complexity as $\delta \to 0$. We reveal that this picture is incomplete beyond this regime. We demonstrate, both empirically and theoretically, that for general δ and η, the complexity is jointly governed by the jumps and the relative positions of the change points.*

Related publication. This Chapter is a joint work with Maximilian Graf[1], available as a preprint in (Graf and Thuot, 2026).

7.1 Introduction

Many scientific and engineering systems can be modeled as piecewise-constant functions of a continuous input, and identifying the thresholds at which their behavior changes abruptly is a fundamental task known as change point detection. In materials science, one seeks the control parameters (e.g., temperature and pressure) at which a material undergoes a phase transition (Nikolaev et al., 2016; Sebastián et al., 2011); in queuing and processing systems, one aims to detect abrupt changes in waiting times or throughput (Hung and Michailidis, 2007); in machine learning, one studies the sensitivity of black-box models to hyperparameter changes (Lan et al., 2009; Hayashi et al., 2019). In all these settings, evaluations are costly or time-consuming, and the input domain is continuous, so adaptive and sequential sampling strategies are particularly relevant. Moreover, reliable guarantees are required in practice: both the estimation precision and the probability of error must be controlled. This motivates a bandit formulation in which a learner queries the function at adaptively chosen inputs and aims to identify a prescribed number of change points, at a given precision η and confidence level $1-\delta$, while minimizing the total number of evaluations.

1. Equal contribution—Institut für Mathematik, Universität Potsdam, Potsdam, Germany

In this manuscript, we consider the multiple change point detection problem, as introduced in (Hayashi et al., 2019) and recently studied in (Lazzaro and Pike-Burke, 2025b). To fix notation before discussing related work (a more detailed presentation is deferred to Section 7.2), we consider a piecewise-constant function $f : [0,1] \to \mathbb{R}$ with m ($m \geqslant 1$) change points at positions $0 < x_1^* < \cdots < x_m^* < 1$, so that f is constant on $[x_{l-1}^*, x_l^*)$ for $l \in \{1, \ldots, m+1\}$, with $x_0^* = 0$ and $x_{m+1}^* = 1$. The learner sequentially and adaptively queries points $x_t \in [0,1]$ and observes $y_t = f(x_t) + \varepsilon_t$, where ε_t is sub-Gaussian noise; in particular, each new query may depend on past observations. Given a target number of change points $N \leqslant m$, a precision $\eta \in (0,1)$, and an error rate $\delta \in (0,1)$, the goal is to output N estimates within distance η of true change points with probability at least $1-\delta$. The sample complexity, denoted by $\mathcal{T}$, is the total number of evaluations required to achieve this guarantee.

In (Lazzaro and Pike-Burke, 2025b), the authors propose `MCPI`, a method based on the stopping conditions from (Garivier and Kaufmann, 2016), and prove its asymptotic optimality in expectation. Their analysis is conducted in a discrete action space with K arms, which corresponds to a precision of $\eta \simeq 1/K$ in our continuous setting. They provide non-asymptotic upper bounds, which contain at least a linear [2] dependence on K, while the focus of their analysis lies on the asymptotic regime $\delta \to 0$. Indeed, `MCPI` identifies N change points with expected sample complexity satisfying $\lim_{\delta \to 0} \frac{\mathbb{E}[\mathcal{T}]}{\log(1/\delta)} = 8H^{(N)}_{\text{localize}}$, where

$$H^{(N)}_{\text{localize}} = \sum_{l=1}^{N} \frac{1}{\Delta^2_{(l)}} \quad , \tag{7.1}$$

with $\Delta_i = f(x_i^*) - f(x_{i-1}^*)$ for $i \in \{1, \ldots, m\}$ denoting the jump at the i-th change point, and $|\Delta_{(1)}| \geqslant \cdots \geqslant |\Delta_{(m)}|$ denoting the decreasing rearrangement of jump magnitudes. Thus, the complexity is driven by the N-largest jumps in the asymptotic regime $\delta \to 0$, while dependence on K disappears in the limit. The analysis of large-scale problems (K large) remains open.

For comparison, consider the batch change point detection problem, in which the learner evaluates f once at each point of a fixed grid (Lai, 2001; Yu, 2020). In that setting, the relevant difficulty measure is the energy of each change point, defined as $\mathcal{E}_i^2 := s_i \Delta_i^2$, where s_i is the minimal spacing to neighboring change points (see Equation 7.4 in Section 7.2). One of our contributions is to show that energy also matters in the active setting. For instance, the guarantees in (Verzelen et al., 2023) imply a uniform-sampling detection threshold, and hence a sample-complexity term of order

$$H^{(m)}_{\text{detect}} = \max_{i=1,\ldots,m} \frac{1}{\mathcal{E}_i^2}, \tag{7.2}$$

which is always at least as large as $\frac{1}{2} H^{(N)}_{\text{localize}}$; it can in fact be much larger. For instance, consider change points with comparable jump size Δ and local spacing s (with $s \leqslant 1/m$). Then $H^{(m)}_{\text{detect}} \asymp \frac{1}{s\Delta^2}$, whereas $H^{(N)}_{\text{localize}} \asymp \frac{N}{\Delta^2}$. When s is small, detection may dominate the budget, showing that the rate (7.1) can miss an important non-asymptotic effect. This suggests a richer interplay between jump magnitudes and local spacings than what is captured by the asymptotic jump-only rate

2. Additionally, a quadratic dependence on K (namely, K^2) appears to be hidden in the non-explicit terms of their Prop. 5.6.

in (7.1).

We consider a continuous action space that avoids the restrictive assumption $\eta < \min_{i=1}^{m} s_i$ required by discrete methods such as (Lazzaro and Pike-Burke, 2025b). In the discrete setting, this induces a linear dependence on $1/\eta$, which can become prohibitive when high localization precision is required. Our algorithm adapts to local spacings s_i, which is practically significant when change points may be arbitrarily close and the spacing is unknown. By decomposing detection and localization, we show that the dependence on precision η is logarithmic, namely $\log(1/\eta)$, which is exponentially better than the linear dependence on the discretization grid $K \simeq 1/\eta$ inherent to discrete approaches. Deriving guarantees that explicitly capture the optimal dependence on both η and δ in all regimes is an open question and a key motivation for our work.

We illustrate the energy-based complexity in **Experiment 1** (Figure 7.1) with two change points ($\Delta_1 = -\Delta_2 = 1$), standard noise, and varying spacing s. We compare our Algorithm 20 (`LCP`) to `MCPI` from (Lazzaro and Pike-Burke, 2025b).[3] The results validate the theory: as s increases, sample complexity of `LCP` decreases roughly as $1/s$, matching the prediction $H_{\text{detect}} \asymp 1/(s\Delta^2)$. In contrast, `MCPI` is insensitive to s because its rate is dominated by a constant term proportional to $1/\eta$. Then, adaptation to local spacings is essential, as it allows us to outperform `MCPI` for moderate values of s.

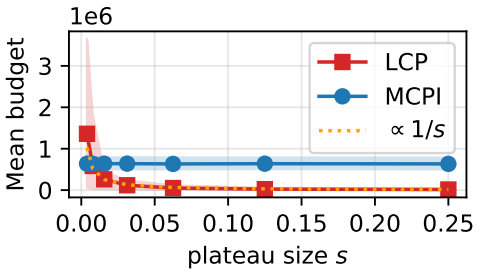


Figure 7.1 – **Experiment 1**. We consider two change points. We set $x_1^* \sim \mathcal{U}(0, 1/2)$ for each Monte-Carlo run and $x_2^* = x_1^* + s$ with s varying and 1000 Monte Carlo runs. We chose $\Delta_1 = -\Delta_2 = 1$ and compare `LCP` (Algorithm 20) with parameters $\eta = 2^{-11}$, $\delta = 0.05$ and $\delta_{\text{explore}} = 1$ to algorithm `MCPI`.

Open questions. These observations motivate the present work and raise the following fundamental questions, relative to the sample complexity of multiple change point identification:

— How does the sample complexity depend on jump magnitudes and the local spacings between change points? When is it governed by the energies $\mathcal{E}_i$ rather than only by the jumps Δ_i?

— What is the optimal achievable dependence on the precision η and confidence level δ? How does the problem scale for high precision (η small)?

Contributions. In this manuscript, we answer these questions.

1. We propose `LocalizeChangePoints` (Algorithm 20), a new algorithm for multiple change point identification under bandit feedback. It decomposes the task into (i) detecting intervals likely to contain change points and (ii) sequential refinement by binary search. We prove

3. In Appendix 7.D we show how to transform our setting into a discrete problem, for comparison with the `MCPI` Algorithm.

non-asymptotic sample-complexity guarantees that depend on both jump magnitudes and local spacings through $\mathcal{E}_i$ and Δ_i. In particular, Theorem 7.3.1 states that the budget $\mathcal{T}$ for localizing N change points among m is bounded, up to logarithmic terms independent of δ and η,[4] as

$$\mathbb{E}[\mathcal{T}] \underset{\log}{\leqslant} H^{(m)}_{\text{detect}} + H^{(N)}_{\text{localize}} \log\left(\frac{1}{\delta\eta}\right), \qquad \text{see (7.1), (7.2)}$$
$$\mathcal{T} \underset{\log}{\leqslant} H^{(m)}_{\text{detect}} \log\left(\frac{1}{\delta}\right) + H^{(N)}_{\text{localize}} \log\left(\frac{1}{\delta\eta}\right) \qquad \text{w.h.p. } 1-\delta.$$

2. Additionally, we derive information-theoretic lower bounds (Theorem 7.4.1). On a local-minimax class of instances, we prove that the bounds above are rate-optimal in all parameters, including δ and η. This shows that the asymptotic rate in (7.1) is overly optimistic.
3. Finally, we validate our results numerically on synthetic data.

Batch change point detection. Change point detection is one of the classical problems in statistics, with a rich literature spanning cumulative-sum (CUSUM) procedures (Hinkley, 1970), binary segmentation (Scott and Knott, 1974), wild binary segmentation (Fryzlewicz, 2014), and penalized least-squares methods (Wang et al., 2020; Verzelen et al., 2023); see (Lai, 2001; Yu, 2020; Truong et al., 2020) for comprehensive reviews. Our algorithm is directly inspired by two classical ideas from this literature: multiscale testing, to identify candidate intervals likely to contain a change point (Kovács et al., 2023; Pilliat et al., 2023), and binary search, to localize it to precision η. It is also well established that the relevant difficulty measure is the energy $\mathcal{E}_i^2 = s_i\Delta_i^2$ of each change point (Yu, 2020; Verzelen et al., 2023), an observation that we extend to the bandit setting. The problem has also been studied in the multivariate setting (Wang and Samworth, 2018) and in kernel-based frameworks (Arlot et al., 2019).

Change point detection in piecewise-stationary bandits. A related but distinct stream of literature studies piecewise-stationary bandits (Garivier and Moulines, 2011), where the reward distributions of a multi-armed bandit change at unknown time steps. Two main tasks are considered: quickest detection, which aims to raise an alarm as soon as possible after a change occurs (Veeravalli et al., 2024; Zhang and Mei, 2023; Xu and Mei, 2023), and regret minimization, where the goal is to adapt to the new environment in order to minimize cumulative regret (Cao et al., 2019; Besson et al., 2022; Mukherjee, 2022). Our setting differs from this line of work in two fundamental respects: our model is stationary, and the change points are spatial features of a fixed function f rather than temporal.

Active change point detection. Active and sequential change point detection studies settings in which the sampling locations are chosen adaptively rather than fixed in advance. Early foundations for this perspective appear in adaptive change-point estimation (Hall and Molchanov, 2003; Lan et al., 2009), where multistage sampling is used to improve localization rates. In the

4. Typically, $\underset{\log}{\leqslant}$ hides here at most a $\log(H^{(m)}_{\text{detect}})$ multiplicative factor.

bandit setting, (Hayashi et al., 2019) formulates active change point detection explicitly, and recent fixed-budget and fixed-confidence analyses for piecewise-constant bandits (Lazzaro and Pike-Burke, 2025a,b) provide instance-dependent guarantees based on jump magnitudes. In particular, (Lazzaro and Pike-Burke, 2025b) combines this formulation with Track-and-Stop ideas from (Garivier and Kaufmann, 2016) and proves asymptotic rates proportional to $\sum_{l=1}^{N} 1/\Delta_{(l)}^2$. Our work complements this line by focusing on non-asymptotic guarantees and by highlighting the additional role of spacing between CP. We highlight the strong connection to other structured bandit problems, such as monotone bandits (Cheshire et al., 2020) or clustering with bandit (Yang et al., 2024; Graf et al., 2025).

Outline. The rest of the paper is organized as follows. In Section 7.2, we introduce the formal setting and notation. In Section 7.3, we present our algorithm and its guarantees. In Section 7.4, we derive information-theoretic lower bounds. In Section 7.5, we illustrate our results numerically on synthetic data. Finally, in Section 7.6, we discuss the implications of our results, possible extensions, and limitations. All proofs are deferred to the appendix.

7.2 Problem formulation and notation

We consider an unknown piecewise-constant function $f : [0,1] \to \mathbb{R}$ with $m \geqslant 1$ change points at positions $0 < x_1^* < \cdots < x_m^* < 1$. The change points are the discontinuities of f, so f is constant on each interval $[x_{l-1}^*, x_l^*)$ for $l \in \{1, \ldots, m+1\}$, with the conventions $x_0^* = 0$ and $x_{m+1}^* = 1$. Both f and the true number of change points m are unknown to the learner.

We parametrize f as

$$f(x) = \mu_0 + \sum_{i=1}^{m} \Delta_i \mathbb{1}_{x \geq x_i^*} \,, \tag{7.3}$$

where $\mu_0 \in \mathbb{R}$ is the baseline level and $\Delta_i = f(x_i^*) - f(x_{i-1}^*)$ for $i \in \{1, \ldots, m\}$ is the jump at the i-th change point. For any compact interval $I = [l(I), r(I)]$, we introduce the notation $\Delta(I) = f(r(I)) - f(l(I))$ for the jump magnitude across the interval I. We assume $\Delta_i \neq 0$ for all i, so every change point is identifiable. Let $|\Delta_{(1)}| \geqslant \cdots \geqslant |\Delta_{(m)}|$ denote the decreasing ordering of jump magnitudes. These jumps govern asymptotic sample-complexity rates in (Lazzaro and Pike-Burke, 2025b), through the complexity $H_{\text{localize}}^{(N)} = \sum_{i=1}^{N} 1/\Delta_{(i)}^2$ (Equation 7.1). Additionally, we assume that the jumps are bounded by 1, i.e., $|\Delta_i| \leq 1$ for all $i \in \{1, \ldots, m\}$. This assumption is without loss of generality, as one can always rescale the problem by a known bound on the jumps. It is made for clarity of presentation and to avoid unnecessary constants in the bounds, but it can easily be removed.

Spacing between change points also plays a central role. For $i \in \{1, \ldots, m-1\}$, let $\vartheta_i = x_{i+1}^* - x_i^*$ denote the spacing after the i-th change point. By convention, define $\vartheta_0 = \vartheta_m = 1$. We then define the local spacing by $s_i = \vartheta_{i-1} \wedge \vartheta_i$ for $i = 1, \ldots, m$, and the associated energies by

$$\mathcal{E}_i^2 = s_i \Delta_i^2 \qquad \text{for } i = 1, \ldots, m. \tag{7.4}$$

The quantity s_i plays the role of sparsity level, and the quantity $\mathcal{E}_i$ captures the effect of local spac-

ing on the difficulty of detecting change point i, in line with the batch literature (Yu, 2020). The interplay between these quantities is captured in the complexity term $H^{(m)}_{\text{detect}} = \max_{i=1,\dots,m} 1/\mathcal{E}_i^2$ (Equation 7.2). Observe that for $m = 1$, one has $H_{\text{detect}} = H_{\text{localize}}$.

We work in a bandit setting: at each round $t \geq 1$, the learner chooses $x_t \in [0,1]$ and observes

$$y_t = f(x_t) + \varepsilon_t \ ,$$

where $(\varepsilon_t)_{t\geq 1}$ are i.i.d. 1-subGaussian random variables, i.e., $\mathbb{E}[\varepsilon_t] = 0$ and $\mathbb{E}[\exp(\lambda\varepsilon_t)] \leq \exp(\lambda^2/2)$ for all $\lambda \in \mathbb{R}$. This includes, for instance, Gaussian and bounded noise. [5] The pair (f, ε) defines an environment, denoted by ν.

Sampling is sequential and adaptive: each query x_t may depend on past observations. An algorithm π consists of three components: (i) a sampling rule for choosing next query, (ii) a stopping rule, and (iii) a decision rule that outputs estimated change points. These components may be randomized. The (random) sample complexity of π, denoted by $\mathcal{T}_\pi$, is the total number of evaluations of f. We write $\mathbb{P}_{\pi,\nu}$ and $\mathbb{E}_{\pi,\nu}$ for probability and expectation under environment ν and algorithm π.

The learner is given a precision $\eta \in (0,1)$, a confidence parameter $\delta \in (0,1)$, and a target number of change points $N \in \{1,\dots,m\}$. The objective is to identify N distinct change points within distance η of true change points with probability at least $1-\delta$, while minimizing the sample complexity. Note that the learner can select any of the m change points, as long as it localizes N distinct change points.

Definition 7.2.1 ((δ,η,N)-correctness)**.** An algorithm π is (δ,η,N)-correct if, for every environment ν with at least N change points,

$$\mathbb{P}_{\pi,\nu}\left(\exists 1 \leqslant i_1 < \cdots < i_N \leqslant m : \ \forall l \in [N], |\hat{x}_l - x^*_{i_l}| \leqslant \eta\right) \geq 1-\delta \ , \tag{7.5}$$

where $\hat{x}_1,\dots,\hat{x}_N$ are the estimates returned by π.

7.3 Algorithmic method and guarantees

7.3.1 Detection of Change Points

We describe the detection subroutine at the core of our method. Given a confidence level δ and a budget T, `DetectIntervals` (Algorithm 19) returns a set $\mathcal{I}$ of intervals with disjoint interiors. Using multiscale endpoint tests, the procedure guarantees that every returned interval contains at least one change point, with failure probability at most δ (Lemma 7.B.1).

Procedure. For each depth $d \in \{1,\dots,d_{\max}\}$ (with $d_{\max}$ defined in Line 1), the domain is partitioned into $n_d = 2^d$ dyadic intervals of equal length. Each endpoint is sampled T_d times (Line 3). Interval $[i/2^d, (i+1)/2^d]$ is flagged when the empirical jump between its endpoints exceeds the Hoeffding threshold β_d (Line 8), which accounts for the multiple testing across depths,

5. The analysis extends to σ-subGaussian noise by a standard rescaling.

intervals, and endpoints.[6] Flagged intervals are aggregated across depths; if a newly flagged interval is strictly contained in an existing one, the larger one is removed (Line 10). This pruning keeps $\mathcal{I}$ disjoint while favoring the finest detected intervals.

Algorithm 19: `DetectIntervals`

Input: δ confidence parameter, T budget
Result: $\mathcal{I}$ set of disjoint intervals

1 $\mathcal{I} \leftarrow \emptyset$, $d_{\max} \leftarrow \lfloor \log_2(T/\log(1/\delta)) \rfloor$;
2 **for** $d = 1, \dots, d_{\max}$ **do** ▷ Multiple-scale loop
3 $\quad n_d \leftarrow 2^d$, $T_d \leftarrow \left\lfloor \frac{T}{d_{\max}(n_d+1)} \right\rfloor$, $\beta_d \leftarrow \sqrt{\frac{8}{T_d} \log\left(\frac{2d_{\max}(n_d+1)}{\delta}\right)}$;
4 $\quad$ **for** $i = 0, 1, \dots, n_d$ **do**
5 $\quad\quad$ sample T_d times from $x_i^{(d)} = i/n_d$, store means as $\hat{y}_i^{(d)}$ ▷ Sample uniformly at endpoints
6 $\quad$ **end**
7 $\quad$ **for** $i = 1, \dots, n_d$ **do**
8 $\quad\quad$ **if** $|\hat{y}_i^{(d)} - \hat{y}_{i-1}^{(d)}| > \beta_d$ **then** ▷ Test for presence of jump between adjacent endpoints
9 $\quad\quad\quad$ remove from $\mathcal{I}$ any set containing $[x_{i-1}^{(d)}, x_i^{(d)}]$ ▷ Pruning step
10 $\quad\quad\quad$ add $[x_{i-1}^{(d)}, x_i^{(d)}]$ to $\mathcal{I}$, ▷ Update $\mathcal{I}$
11 $\quad\quad$ **end**
12 $\quad$ **end**
13 **end**

Guarantees and energy scaling. The threshold β_d in Algorithm 19 is set so that, with probability at least $1 - \delta$, every flagged interval I satisfies $\Delta(I) \neq 0$ and therefore contains at least one change point. Moreover, Lemma 7.B.1 shows that if $T \gtrsim \mathcal{E}_i^{-2} \log(1/\delta)$ (up to logarithmic factors in $\mathcal{E}_i^{-2}$), then change point i is detected with probability at least $1 - \delta$. For intuition, consider for change point i its natural depth $d_i^\star \asymp \log_2(1/s_i)$: at this depth, one dyadic interval isolates x_i^* and has left-right jump $|\Delta_i|$. Detection requires $|\Delta_i| \gtrsim \beta_{d_i^\star}$, i.e., $T_{d_i^\star} \gtrsim \Delta_i^{-2} \log(1/\delta)$. Since $T_{d_i^\star} \asymp s_i T$ up to logarithmic factors, this yields $T \gtrsim \mathcal{E}_i^{-2} \log(1/\delta)$, which explains the $H_{\text{detect}}^{(m)}$ term in the upper bound. Precise statements are given in Lemma 7.B.1 (Appendix).

7.3.2 Main algorithm and guarantees

Algorithm 20 takes as input the target number of change points N, the confidence level δ, and the precision η. In addition, it uses a tuning parameter δ_{explore}, which controls the probability of failure of the exploration phase and, therefore, the budget. No other problem-dependent parameter is required.

The algorithm follows a doubling schedule. At stage k (initialized in Line 1), it is given budget $T_k = 2^{k+2}$ and runs four steps that mimic the oracle strategy described above. If, at the end of the

6. If $T_d = 0$, β_d is set to $+\infty$ by convention.

stage, all N candidates are certified with error probability at most $\delta^{(k)}$, the algorithm returns the estimates $\mathcal{C} = \{c_1^{(k)}, \dots, c_N^{(k)}\}$ (Line 13). Otherwise, it moves to stage $k+1$, doubles the budget, and repeats (Line 17). Before stating the main guarantees of Algorithm 20, we describe each step.

At a generic stage with budget T_k, the four steps are the following.

(i) Detection. In Line 3, `DetectIntervals` (Algorithm 19) constructs a set $\mathcal{I}$ of disjoint intervals, each certified to contain at least one change point. This step is run with confidence level $\delta_{\text{explore}}/4$ and is described in Subsection 7.3.1.

(ii) Jump estimation. In Line 5, `EstimateJumps` (Algorithm 21) estimates the jump magnitudes on the candidate intervals and keeps the N most informative ones. The procedure is designed so that, with high probability, these estimates are accurate up to a constant multiplicative factor. It returns N intervals $\mathcal{J}^{(k)} = \{J_1^{(k)}, \dots, J_N^{(k)}\}$ together with the corresponding jump estimates $\mathcal{G}^{(k)} = \{\hat{\Delta}_1^{(k)}, \dots, \hat{\Delta}_N^{(k)}\}$, where $\hat{\Delta}_v^{(k)}$ estimates $|\Delta(J_v^{(k)})|$. The intervals are ordered by decreasing estimated jump, so that $\hat{\Delta}_1^{(k)} \geqslant \hat{\Delta}_2^{(k)} \geqslant \cdots \geqslant \hat{\Delta}_N^{(k)}$. `EstimateJumps` is a standard adaptive estimation routine. We defer its description to Appendix 21; its guarantees are stated in Lemma 7.B.3.

(iii) Refinement of localization. We use `SHB` (Sequential Halving with Backtracking) from (Lazzaro and Pike-Burke, 2025a), which localizes a single change point up to precision η under a fixed budget constraint. The method consists of a binary-search procedure, with a backtracking mechanism as in (Cheshire et al., 2020). In particular, when applied to an interval containing a single change point of jump magnitude Δ, a budget of order $\Delta^{-2}\log(1/(\delta\eta))$ suffices to return an estimate within distance η of the true change point with probability at least $1-\delta$—(Lazzaro and Pike-Burke, 2025a). For completeness, we restate these guarantees in Lemma 22, and the routine itself in Algorithm 22.

In Line 9, we apply `SHB` independently to each selected interval. The budget is allocated proportionally to the inverse squared estimated jump, so interval $J_v^{(k)}$ receives budget $T_v^{(k)}$ with

$$T_v^{(k)} = \left\lfloor \alpha_v^{(k)} \cdot 2^k \right\rfloor \vee 1 \text{ , with } \qquad \alpha_v^{(k)} = \frac{\left(\hat{\Delta}_v^{(k)}\right)^{-2}}{\sum_{v'=1}^{N} \left(\hat{\Delta}_{v'}^{(k)}\right)^{-2}} \ . \tag{7.6}$$

(iv) Verification. In Line 10, `VerifyCP` (Algorithm 23) tests whether the candidates $(c_v^{(k)})_{v\in[N]}$ lie within distance η of N distinct change points. For each v, it performs a two-sample test at the points

$$l_v^{(k)} := \max\left(\min J_v^{(k)}, c_v^{(k)} - \eta\right), \qquad r_v^{(k)} := \min\left(\max J_v^{(k)}, c_v^{(k)} + \eta\right), \tag{7.7}$$

using confidence level $\delta^{(k)} = \frac{3\delta}{2\pi^2 N k^2}$. It draws $T_v^{(k)}/2$ samples at each endpoint and compares the empirical jump to the threshold $\sqrt{\frac{32}{T_v^{(k)}}\log\left(\frac{2}{\delta^{(k)}}\right)}$. If the test is positive, then $c_v^{(k)}$ is certified to be within distance η of a change point with probability at least $1-\delta^{(k)}$. The routine is given in Algorithm 23, and its guarantees are stated in Lemma 7.B.5.

Theorem 7.3.1. *Let $N \leqslant m$, $0 < \delta < 1/4$, $0 < \eta < 1/4$, and $\delta_{\text{explore}} \leqslant 1/4$. Consider any environment ν with $m \geqslant N$ change points, and run Algorithm 20 with input parameters $(N, \delta, \eta, \delta_{\text{explore}})$.*

1. *(Correctness) Algorithm 20 is (δ, η, N)-correct (see (7.5)). With probability at least $1-\delta$,*

Algorithm 20 returns N points $0 \le c_1 < c_2 < \cdots < c_N \le 1$ such that $|c_i - x^*_{l_i}| \le \eta$ for $i = 1, \dots, N$, and $1 \le l_1 < l_2 < \cdots < l_N \le m$.

2. *(High-probability bound) If we set $\delta_{\text{explore}} = \delta$, then there exists an event ξ of probability at least $1 - \delta$ on which the output is correct (Definition 7.5), and the budget $\mathcal{T}$ satisfies*

$$\mathcal{T} \le c \cdot \left(\omega_1 H^{(m)}_{\text{detect}} \left(\log\left(\frac{1}{\delta}\right) + \omega_2 \right) + H^{(N)}_{\text{localize}} \left(\log\left(\frac{1}{\delta\eta}\right) + \omega_3 \right) \right).$$

3. *(Expectation bound) If we set $\delta_{\text{explore}} = 1/4$, then the expected budget $\mathbb{E}[\mathcal{T}]$ is bounded by*

$$\mathbb{E}[\mathcal{T}] \le c \cdot \left(\omega_1 H^{(m)}_{\text{detect}} \omega_2 + H^{(N)}_{\text{localize}} \left(\log\left(\frac{1}{\delta\eta}\right) + \omega_3 \right) \right).$$

In these bounds, $c > 0$ is a numerical constant, and $\omega_1, \omega_2, \omega_3$ are logarithmic terms depending on the environment, defined as $\omega_1 = \log(H^{(m)}_{\text{detect}})$, $\omega_2 = \max_{i=1}^{m} \left\{ \log\left(\log\left(\mathcal{E}_i^{-2} \right) \vee e \right) + \log\left(1/s_i\right) \right\}$, and $\omega_3 = \log\left(N \cdot \left(\log\left(H^{(N)}_{\text{localize}} \right) \vee 1 \right) \right)$. See (7.2), (7.1) for definitions of $H^{(m)}_{\text{detect}}$ and $H^{(N)}_{\text{localize}}$.

Algorithm 20: `LocalizeChangePoints`

Input: N number of change points, δ confidence parameter for correctness, η precision parameter, δ_{explore} confidence parameter for budget control

Result: $\mathcal{C}$ estimated change points

$k \leftarrow \lceil \log_2(2N) \rceil$ ▷ Initialize doubling schedule
while *true* **do**
  $\mathcal{I}^{(k)} \leftarrow$ `DetectIntervals` $\left(\frac{\delta_{\text{explore}}}{4}, 2^k \right)$ ▷ *(i) Detect candidate intervals*
  if $|\mathcal{I}^{(k)}| \ge N$ **then**
    $\left(\mathcal{J}^{(k)}, \mathcal{G}^{(k)} \right) \leftarrow$ `EstimateJumps` $\left(\mathcal{I}^{(k)}, \frac{\delta_{\text{explore}}}{4}, 2^k, N \right)$ ▷ *(ii) Estimate jumps*
    if $|\mathcal{J}^{(k)}| \ge N$ **then** ▷ If N intervals detected and estimated
      Order $\mathcal{J}^{(k)} = (J_1^{(k)}, \dots, J_N^{(k)})$ according to estimate jump from $\mathcal{G}^{(k)}$
      for $v = 1, \dots, N$ **do**
        $c_v^{(k)} \leftarrow$ `SHB` $\left(J_v^{(k)}, T_v^{(k)}, \eta \right)$ ▷ *(iii) Refine to precision η; for $T_v^{(k)}$ see (7.6)*
        $\text{ok}_v^{(k)} \leftarrow$ `VerifyCP` $\left(l_v^{(k)}, r_v^{(k)}, \delta^{(k)}, T_v^{(k)} \right)$ ▷ *(iv) Verify; for l_v, r_v see (7.7)*
      end
      if $\forall v \in [N]$, $\text{ok}_v^{(k)}$ *is true* **then** ▷ If all change points verified
        return $\mathcal{C} \leftarrow \left\{ c_1^{(k)}, \dots, c_N^{(k)} \right\}$ ▷ Stop and output estimates
      end
    end
  end
  $k \leftarrow k + 1$, and $\delta^{(k)} \leftarrow \frac{3\delta}{2\pi^2 N k^2}$ ▷ Double budget and retry
end

Intuition. To interpret $H^{(N)}_{\text{localize}} = \sum_{i=1}^{N} \Delta_{(i)}^{-2}$, consider an oracle that knows the true jumps $(\Delta_i)_{i=1}^m$ and the midpoints $c_i^* = (x_{i-1}^* + x_i^*)/2$. This yields N disjoint intervals, each containing one target change point. Running N parallel binary searches with budget proportional to $\Delta_{(i)}^{-2}$ then incurs a total cost of order $H^{(N)}_{\text{localize}} \log(1/(\delta\eta))$ — the cost of pure localization. The main challenge is that these intervals are unknown. Our key message is that they can be acquired adaptively, at an additional cost: detecting that a region contains a change point requires $\mathcal{E}_i^{-2} \log(1/\delta)$ samples rather than Δ_i^{-2}. The total complexity therefore splits into a detection cost $H^{(m)}_{\text{detect}}$ and a localization cost $H^{(N)}_{\text{localize}} \log(1/(\delta\eta))$. We provide here proof sketch, complete proofs are given in Section 7.B.

Sketch of proof. By step (iv), each stage-k acceptance via `VerifyCP` fails with probability at most $\delta^{(k)}$ (Lemma 7.B.5), so the union bound $\sum_k N\delta^{(k)} \le \delta$ implies (δ, η, N)-correctness.
We now sketch the budget bound by following the four-step procedure at stage k.
(i) Detection. By Lemma 7.B.1, if $T_k \gtrsim \omega_1 H^{(m)}_{\text{detect}}\big(\log(1/\delta_{\text{explore}}) + \omega_2\big)$, then with probability at least $1 - \delta_{\text{explore}}/4$, `DetectIntervals` returns m disjoint intervals that isolate each of the m CP.
(ii) Jump estimation. On the same event, and with the same budget, Lemma 7.B.3 implies that `EstimateJumps` returns N intervals and constant-factor accurate jump estimates. Therefore, the allocation weights $\alpha_v^{(k)}$ are within constants of the oracle proportions $\Delta_{(v)}^{-2}/H^{(N)}_{\text{localize}}$.
(iii) Refinement. Given these allocations, Lemma 7.B.4 (from (Lazzaro and Pike-Burke, 2025a)) yields that the v-th change point is localized to accuracy η once it receives budget of order $\Delta_{(v)}^{-2} \log(1/(\delta^{(k)}\eta))$. Summing over $v \in [N]$ gives a sufficient condition for the refinement step to succeed with probability at least $1 - \delta^{(k)}$: $T_k \gtrsim H^{(N)}_{\text{localize}}\big(\log(1/(\delta^{(k)}\eta)) + \omega_3\big)$.
(iv) Verification. By Lemma 7.B.5, when $[l_v^{(k)}, r_v^{(k)}]$ contains the v-th change point, with jump magnitude $\Delta_{(v)}$, as long as $T_v^{(k)} \gtrsim \Delta_{(v)}^{-2} \log(1/(\delta^{(k)}\eta))$, `VerifyCP` returns `True` with high probability. Thus, once (i)- (iii) hold, the stage is successful with high probability, and the algorithm stops.
Combining (i)- (iv), a sufficient stage condition, implying both bounds is

$$T_k \gtrsim \omega_1 H^{(m)}_{\text{detect}}\big(\log(1/\delta_{\text{explore}}) + \omega_2\big) + H^{(N)}_{\text{localize}}\big(\log(1/(\delta^{(k)}\eta)) + \omega_3\big) \ .$$

7.4 Lower Bound

In this section, we derive information-theoretic lower bounds for the budget $\mathcal{T}$ of any (δ, η, m)-correct algorithm π for the active change point detection problem in the case where $N = m$.

Consider an environment ν with $m \geqslant 1$ change points, for which we have to estimate every change point (i.e., $N = m$). As we consider different environments obtained by modification of ν, we use $H^{(m,\nu)}_{\text{detect}}$ (see 7.2) and $H^{(m,\nu)}_{\text{localize}}$ (see 7.1) to indicate their dependence on the environment ν.

Theorem 7.4.1. *Let $\delta \in (0, 1/4)$ and $\eta \in (0, 1/8)$, and consider $m = N$. There exists an environment ν' such that $H^{(m,\nu)}_{\text{detect}} \leqslant H^{(m,\nu')}_{\text{detect}} \leqslant 4H^{(m,\nu)}_{\text{detect}}$, $\frac{1}{2}H^{(m,\nu)}_{\text{localize}} \leqslant H^{(m,\nu')}_{\text{localize}} \leqslant H^{(m,\nu)}_{\text{localize}}$, and such*

that

$$\mathbb{P}_{\pi,\nu'}\left(\mathcal{T}_\pi \geqslant \tfrac{1}{4}H^{(m,\nu)}_{\text{detect}}\log\left(\frac{1}{8\delta}\right) + \tfrac{1}{2}H^{(m,\nu)}_{\text{localize}}\log\left(\frac{1}{8\delta}\right) + \tfrac{1}{2}\sum_{i=1}^{m}\frac{1}{\Delta_i^2}\log_+\left(\frac{s_i}{16\eta}\right)\right) \geqslant \delta \ .$$

Sketch of proof. The proof is information-theoretic. The high-probability statement combines two ingredients, proved separately in Theorems 7.C.1 and 7.C.2 .

(i) Detection term. To obtain a $(1-\delta)$-quantile lower bound of order $H^{(m,\nu)}_{\text{detect}}\log(1/\delta)$, we reduce the problem to signal detection. For intuition, consider an instance with two opposite jumps $\pm\Delta$ separated by spacing s, and compare it with a null environment with no change point. If an algorithm detects a change with probability at least $1-\delta$ using budget quantile T, it must sample sufficiently often inside the true plateau, whose location is unknown among about $1/s$ candidates shifts. This yields $T \gtrsim \frac{1}{s\Delta^2}\log\left(\frac{1}{\delta}\right)$, which is exactly the energy-driven detection scaling. Our reduction extends this intuition to the general case.
(ii) Localization at precision η. To obtain the term $\sum \Delta_i^{-2}\log(s_i/\eta)$, we construct a family of alternatives by shifting each change point x_i^* over $\alpha_i \asymp s_i/8\eta$ candidate positions. Distinguishing these alternatives induces the factor $\log_+(s_i/\eta)$ in the quantile bound. We exploit symmetry techniques to reveal this multiple-testing cost. The term $H^{(m,\nu)}_{\text{localize}}\log(1/\delta)$ follows from standard change-of-measure arguments, as in Theorem 5.2 in (Lazzaro and Pike-Burke, 2025b).
By standard Markov-type arguments, the high-probability lower bound of Theorem 7.4.1 implies the expectation lower bound stated in the following corollary. See Appendix 7.C for the full proof. □

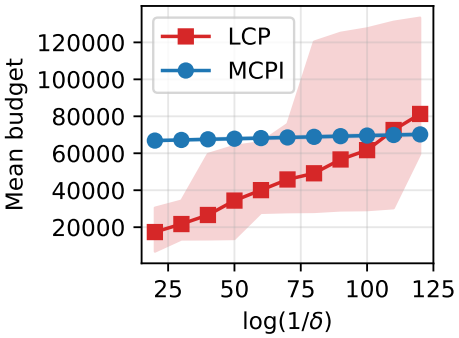
Corollary 7.4.2. *Let $\delta \in (0, 1/16)$ and $\eta \in (0, 1/8)$. Assume that all change points of ν are spaced by at least 2η ($\forall i = 2, \dots, m-1,\ \vartheta_i > 2\eta$). Then, there exists an environment ν' such that $H^{(m,\nu)}_{\text{detect}} \leqslant H^{(m,\nu')}_{\text{detect}} \leqslant 4H^{(m,\nu)}_{\text{detect}}$, $\frac{1}{2}H^{(m,\nu)}_{\text{localize}} \leqslant H^{(m,\nu')}_{\text{localize}} \leqslant H^{(m,\nu)}_{\text{localize}}$, and such that*

$$\mathbb{E}_{\pi,\nu'}[\mathcal{T}_\pi] \geqslant c\left(H^{(m,\nu)}_{\text{detect}} + H^{(m,\nu)}_{\text{localize}}\log\left(\frac{1}{4\delta}\right) + \sum_{i=1}^{m}\frac{1}{\Delta_i^2}\log_+\left(\frac{s_i}{16\eta}\right)\right) \ ,$$

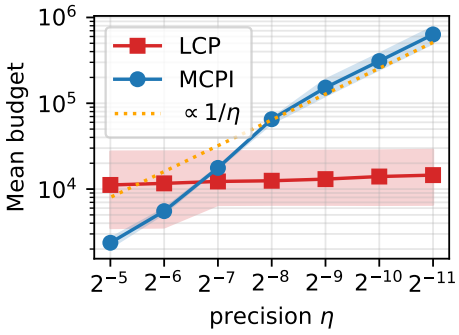
where $c > 0$ is a numerical constant.

7.5 Numerical experiments

Our theoretical findings are supported by simple numerical experiments on simulated data, where we model the underlying noise to be standard normal. In the following two experiments, we consider an environment with two change points, $x_1^* \sim \mathcal{U}(0, 1/2)$ and $x_2^* = x_1^* + s$ with $s = 1/4$, and jump sizes $\Delta_1 = -\Delta_2 = 1$. Each experiment is averaged over 1000 Monte-Carlo iterations. We compare our method, `LCP`, to `MCPI` from (Lazzaro and Pike-Burke, 2025b). Recall that `MCPI` is designed for discrete bandit change point problems, so we apply it to a discretization of the continuous problem (see Appendix 7.D). `MCPI` follows a track-and-stop approach with a forced exploration phase. We will see that while the number of arms, here of order $1/\eta$, does not affect



(a) **Experiment 2**. Average budget and $(0.05, 0.95)$ quantiles for localizing two CPs with varying δ.



(b) **Experiment 3**. Average budget and $(0.05, 0.95)$ quantiles for localizing two CPs with varying η.

Figure 7.2 – Numerical experiments

the asymptotic sample complexity as $\delta \to 0$, it strongly influences the algorithm's performance in practice.

In **Experiment 2**, we set $\eta = 2^{-8}$. With these configurations, we run `LCP` (again we treat δ_{explore} as tuning parameter and set $\delta_{\text{explore}} = 1$) and `MCPI` on a discretization for varying values of δ, that is, $\log(1/\delta) \in \{20, 30, \dots, 120\}$. We plot the averaged budget in Figure 7.2a. We can see that in this setting, the slope of the budget for `LCP` with respect to $\log(1/\delta)$ is larger than the very small slope of the budget required for `MCPI`. Still, it takes $\delta \approx e^{-110}$ until the average budget of our method is larger.

In **Experiment 3**, we choose $\delta = 0.05$ and run `LCP` (again, with $\delta_{\text{explore}} = 1$) and `MCPI` for $\eta = 2^{-i}$, $i = 5, 6, \dots, 11$. We plot the results of our experiment in Figure 7.2b. For `LCP`, there is only a small increase of the budget when η decreases, compared to `MCPI`. For the latter, we can see that for small values of η, the budget increases nearly proportionally to $1/\eta$.

Both experiments illustrate that for moderate choices of δ, asking for a small precision (or considering many arms in the discrete case) does not limit the performance of `LCP`, in contrast to `MCPI`.

In **Experiment 4**, illustrated in Figure 7.3, we consider one change point x_1^* and $\Delta_1 = 1$. Again, we consider standard normal noise. In 1000 Monte-Carlo runs, we run `LCP` (with $\delta_{\text{explore}} = 1$) and `SCPI` with precision $\eta = 2^{-7}$ and varying δ. `SCPI` is an adaptation of `MCPI`, also provided in (Lazzaro and Pike-Burke, 2025b) for identification of a single change point. Like in Experiment 2, we can see that while the budget of our method increases stronger for $\log(1/\delta)$ growing, we still achieve better performance for moderate choices of δ.

In **Experiment 5**, illustrated in Figure 7.4, we consider a problem with $m = 10$ change points and $\Delta_i = 1$ for i odd and $\Delta_i = -1$ for i even. The noise is again supposed to be standard normal. We run `MCPI` and `LCP` (with $\delta_{\text{explore}} = 1$) with varying $\log(1/\delta) \in \{20, 40, 60, 80, 100\}$ and $\eta = 0.0025 \cdot 2^{-i}$, $i = 1, 2, 3$. For `LCP` we observe that only the choice of δ has a visible but mild

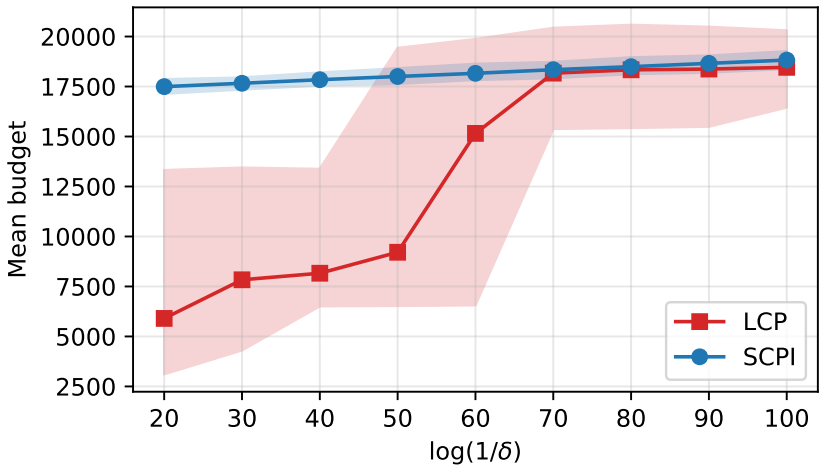


Figure 7.3 – **Experiment 4**. Average budget for localizing one change point with varying δ.

effect on the average budget. For `MCPI`, increasing η by a factor 4 leads to a decrease of the budget by a factor 1/4. Meanwhile, the slope seems to depend almost linearly on $\log(1/\delta)$.

All experiments were run on a MacBook Air (Apple M1, 8 GB RAM), using 4 parallel worker processes. The runtime is documented in table 7.1.

Table 7.1 – Computational resources used for each experiment (runtimes in sec.)

	# MC iterations	Total runtime	Runtime `LCP`	Runtime `MCPI`
Experiment 1	1000	6667.64	433.76	26277.82
Experiment 2	1000	1035.69	145.15	3987.08
Experiment 3	1000	1634.18	37.19	6487.09
Experiment 4	1000	599.06	37.31	2354.73
Experiment 5	100	2619.28	163.92	10281.14

7.6 Discussion

Comparison with (Lazzaro and Pike-Burke, 2025b). Prior expected-budget bounds of order $H^{(N)}_{\text{localize}} \log(1/\delta)$ effectively decompose the problem into N independent single-change-point localizations. This reduction misses a key phenomenon when $m \geqslant 2$: estimating several closely located jumps induces an additional detection cost. Our analysis isolates this term as $H^{(m)}_{\text{detect}}$, and shows that in expectation it enters additively and does not depend on δ; in moderate-confidence regimes, it can dominate the total budget. We also sharpen the dependence on precision: the localization component scales as $H^{(N)}_{\text{localize}} \log(1/\eta)$ rather than polynomially in the number of arms $K = 1/\eta$ in (Lazzaro and Pike-Burke, 2025b). More generally, the guarantees are non-asymptotic in both (η, δ) and adaptive to local spacing s_i, in contrast to approaches that require a global

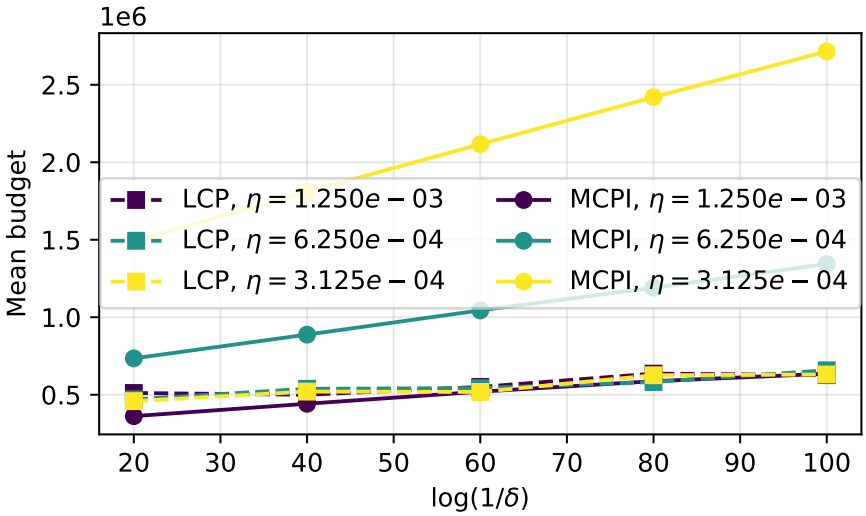


Figure 7.4 – **Experiment 5**. Average budget for localizing 10 change points for three values of η and varying δ.

separation condition $\min_i \vartheta_i > \eta$.

Optimality. When $N = m$, our upper and lower bounds match up to logarithmic factors, yielding order-optimal sample complexity. In the limit $\delta \to 0$, our expectation bound scales as $H^{(N)}_{\text{localize}} \log(1/\delta)$ up to a numerical constant, so the algorithm is rate-optimal in this regime. The case $N < m$ is open: we expect the detection-localization trade-off to persist, but characterizing the optimal rate would require new lower-bound techniques.

Fixed budget setting. Our procedure targets the fixed-confidence setting, but one stage of the doubling schedule in Algorithm 20 could potentially be adapted to fixed-budget. Doing so would require a parameter-free exploration phase, whereas the current method relies on a tuning parameter δ_{explore} to control failure probability. We leave this extension for future work.

Extension to multi-dimensional change point detection. Our framework extends to multivariate change-point and kernel-based models by replacing the one-dimensional test at each stage (including in SHB) with an appropriate two-sample procedure (Wang and Samworth, 2018). Adapting to sparse multidimensional change points, however, requires a careful exploration-phase design, which we leave for future work.

Limitations. Our lower bounds are limited to the case $N = m$, so the optimal trade-off when $N < m$ is still an open question, which is highly non-trivial. Algorithmically, our procedure relies on a doubling schedule, which is not always desirable in practice. Adaptations are required to derive fully-adaptive counterpart of the method.

Appendix of Chapter 7

7.A List of Notation

Table 7.2 – Summary of notation.

Symbol	Description
f	Unknown step function
m	Total number of change points
x_i^*	Location of the i^{th} change point
x_t	Location, chosen by a learner at time t
ε_t	Sub-Gaussian noise at time t
y_t	Feedback $y_t = f(x_t) + \varepsilon_t$ that the learner observes at time t
N	Number of change points the learner wants to localize
η	Precision parameter
δ	Confidence parameter for correctness
δ_{explore}	Confidence parameter for exploration
$\mathcal{T}$	Sample complexity / budget
Δ_i	Jump at the i^{th} change point, $\Delta_i = f(x_i^*) - f(x_{i-1}^*)$
$\Delta_{(i)}$	Jump ordered by its magnitude $\lvert\Delta_{(1)}\rvert \geq \cdots \geq \lvert\Delta_{(m)}\rvert$
ϑ_i	Spacing after the i^{th} change point, $\vartheta_i = x_{i+1}^* - x_i^*$ for $i = 1, \ldots, m-1$, with convention $\vartheta_0 = \vartheta_m = 1$
s_i	Local spacing between the i^{th} change point and its neighboring change points, $s_i = \vartheta_{i-1} \wedge \vartheta_i$ for $i = 1, \ldots, m$,
$\mathcal{E}_i^2$	Energy of the i^{th} change point, $\mathcal{E}_i^2 = s_i \Delta_i^2$
$H_{\text{detect}}^{(m)}$	Complexity term for change point detection, $H_{\text{detect}}^{(m)} = \max_{i=1,\ldots,m} \mathcal{E}_i^{-2}$
$H_{\text{localize}}^{(N)}$	Complexity term for change point localization, $H_{\text{localize}}^{(N)} = \sum_{i=1}^{N} 1/\Delta_{(i)}^2$
$l(I)$, $r(I)$	Boundaries of a compact interval $I = [l(I), r(I)]$
$\Delta(I)$	Jump between interval boundaries, $\Delta(I) = f(r(I)) - f(l(I))$

7.B Proofs of upper bounds

7.B.1 Multiple Change Point Detection

Lemma 7.B.1. *Let $0 < \delta < 1/4$. On an event ξ with $\mathbb{P}(\xi) \geq 1 - \delta$, Algorithm 19 only outputs intervals that contain at least one change point.*

In addition, for any change point x_i^* (with $i \in \{1, \dots, m\}$), if

$$T \geq c \cdot \omega_i^a(\delta) \cdot \frac{1}{\mathcal{E}_i^2} \ ,$$

with

$$\omega_i^a(\delta) = (\log(\mathcal{E}_i^{-2}) \vee 1) \cdot \log\left(\frac{\log(\mathcal{E}_i^{-2}) \vee e}{s_i \cdot \delta}\right) \ ,$$

then on the same event ξ, there exists an interval $I \in \mathcal{I}$ such that $x_i^* \in I$ and $|I| \leq \frac{s_i}{2}$.

In particular, if $T \geqslant \max_{i=1}^m c \cdot \omega_i^a(\delta) \cdot \frac{1}{\mathcal{E}_i^2}$, then on the event ξ, for any change point $i \in [m]$, there exists a unique interval $I \in \mathcal{I}$ containing x_i^*, and this interval has a length smaller than $\frac{s_i}{2}$.

Proof of Lemma 7.B.1. We refer to Section 7.3.1 and Algorithm 19 for the introduction of the notation used in this proof. Fix one depth $d \in \{1, \dots, d_{\max}\}$ and one index $j \in \{0, \dots, n_d\}$. Let $y_j^{(d)} = f(x_j^{(d)})$ be the true function value at $x_j^{(d)}$. By assumption, the noise is 1-sub-Gaussian, so Hoeffding's inequality yields, for any $\varepsilon > 0$,

$$\mathbb{P}\left(|\hat{y}_j^{(d)} - y_j^{(d)}| \geq \varepsilon\right) \leq 2\exp\left(-\frac{T_d \varepsilon^2}{2}\right) \ .$$

From a union bound, first over all $n_d + 1$ endpoints of depth d, and then over each depth $d = 1, \dots, d_{\max}$, we obtain an event ξ with $\mathbb{P}(\xi) \geq 1 - \delta$ such that for all $d = 1, \dots, d_{\max}$ and all $j = 0, \dots, n_d$,

$$|\hat{y}_j^{(d)} - y_j^{(d)}| \leq \sqrt{\frac{2}{T_d} \log\left(\frac{2d_{\max}(n_d + 1)}{\delta}\right)} = \beta_d/2. \tag{7.8}$$

First, on ξ, intervals without change points are not added to $\mathcal{I}$. Indeed, consider any depth d and any interval $[x_{j-1}^{(d)}, x_j^{(d)}]$ that does not contain a change point. Then $y_{j-1}^{(d)} = y_j^{(d)}$, and therefore, on ξ (see (7.8)),

$$|\hat{y}_j^{(d)} - \hat{y}_{j-1}^{(d)}| \leq |\hat{y}_j^{(d)} - y_j^{(d)}| + |y_{j-1}^{(d)} - \hat{y}_{j-1}^{(d)}| \leq \beta_d \ ,$$

and the interval is not added to $\mathcal{I}$ (Line 8). This guarantees that every interval in $\mathcal{I}$ contains at least one change point, which proves the first part of the statement.

Second, on ξ, change points are detected if the budget is large enough. Fix any change point x_i^*, with $i \in \{1, \dots, m\}$, jump size Δ_i, and minimal spacing s_i to the other change points. Recall

that $d_{\max} = \lfloor \log_2(T/\log(1/\delta)) \rfloor$, $n_d = 2^d$, and define the change point-specific depth

$$d_i^* = \left\lceil \log_2\left(\frac{2}{s_i}\right) \right\rceil \leq d_{\max} \tag{7.9}$$

The latter inequality ensures that depth d_i^* is considered by the algorithm. It follows from

$$T \geq 2e \cdot \mathcal{E}_i^{-2} \cdot \log(1/\delta) \geq \frac{2e \cdot \log(1/\delta)}{s_i} \ ,$$

recalling that $\mathcal{E}_i^2 = s_i \Delta_i^2 \leq s_i$.

Consider depth d_i^* and the index j such that $x_{j-1}^{(d_i^*)} < x_i^* \leq x_j^{(d_i^*)}$. By (7.9), we have $2^{-d_i^*} \leq s_i/2$, so the definition of s_i ensures that no other change point is contained in $[x_{j-1}^{(d_i^*)}, x_j^{(d_i^*)}]$, and $y_j^{(d_i^*)} - y_{j-1}^{(d_i^*)} = \Delta_i$. Moreover, on ξ, using (7.8) and the reverse triangle inequality,

$$|\hat{y}_j^{(d_i^*)} - \hat{y}_{j-1}^{(d_i^*)}| \geq |\Delta_i| - \beta_{d_i^*} \ .$$

Now, if T is large enough to ensure $\Delta_i \geq 2\beta_{d_i^*}$, then the test in Line 8 of Algorithm 19 is positive, and $[x_{j-1}^{(d_i^*)}, x_j^{(d_i^*)}]$ is added to $\mathcal{I}$ during the iteration at depth d_i^*. In particular, x_i^* is contained in an interval of $\mathcal{I}$ with length at most $2^{-d_i^*} \leq s_i/2$. Besides, at the following iterations at depth $d > d_i^*$, the algorithm maintains in $\mathcal{I}$ exactly one interval containing x_i^*, and whose length is at most $s_i/2$.

Indeed, consider any depth $d > d_i^*$ and an interval $[x_{j-1}^{(d)}, x_j^{(d)}]$ included in $[x_{j-1}^{(d_i^*)}, x_j^{(d_i^*)}]$. If it does not contain a change point, it is not added to $\mathcal{I}$, as the event ξ holds (Equation 7.8). Otherwise, $[x_{j-1}^{(d_i^*)}, x_j^{(d_i^*)}]$ contains x_i^* (as it is the unique change point in that interval). It may be added to $\mathcal{I}$, in which case the interval from $\mathcal{I}$ containing x_i^* is removed. Therefore, at the end of Algorithm 19, x_i^* is contained in exactly one interval of $\mathcal{I}$, with length at most $2^{-d_i^*} \leq s_i/2$.

It remains to determine a condition on T that guarantees $\Delta_i \geq 2\beta_{d_i^*}$.

Intuitively, by definition of d_i^*, we have $T_{d_i^*} \underset{\log}{\simeq} s_i T$ up to logarithmic factors, so $\Delta_i \geq 2\beta_{d_i^*}$ is roughly equivalent to $T \geq c\,\mathcal{E}_i^{-2} \log(1/\delta)$ up to logarithmic factors. The precise formulation is given in technical Lemma 7.B.2, which concludes the proof. □

Lemma 7.B.2. *Let $0 < \delta < 1/4$, $\Delta_i \leq 1$. If*

$$T \geq 7.4 \cdot 10^5 \cdot (\log(\mathcal{E}_i^{-2}) \vee 1) \cdot \log\left(\frac{\log(\mathcal{E}_i^{-2}) \vee 1}{s_i \cdot \delta}\right) \cdot \mathcal{E}_i^{-2} \ ,$$

then $T_{d_i^*} \geqslant 1$, and $\Delta_i \geqslant 2\beta_{d_i^*}$.

Proof of Lemma 7.B.2. Assume that $T \geq c \cdot (\log(\mathcal{E}_i^{-2}) \vee 1) \cdot \log\left(\frac{\log(\mathcal{E}_i^{-2}) \vee 1}{s_i \cdot \delta}\right) \cdot \mathcal{E}_i^{-2}$ for a constant c we want to determine.

First, we want to verify that $T_{d_i^*} \geqslant 1$. By definition of the budget allocated to depth d_i^*, we

have

$$T_{d_i^*} = \left\lfloor \frac{T}{d_{\max}(n_{d_i^*}+1)} \right\rfloor \ .$$

We have (since $\delta < 1/4$)

$$d_{\max} \leq \log_2(T/\log(1/\delta)) \leq \frac{\log(T)}{\log(2)}$$

and

$$n_{d_i^*} + 1 = 2^{\left\lceil \log_2\left(\frac{2}{s_i}\right)\right\rceil} + 1 \leq \frac{5}{s_i} \ ,$$

so

$$\frac{d_{\max}(n_{d^*}+1)}{T} \leq \frac{5}{\log(2)} \cdot \frac{\log(T)}{s_i \cdot T} \ .$$

Note that $s_i \leq 1/2$ (otherwise, we could simply pick any $T \geq 60$, which could be relevant in the cases $i = 1$ and $i = N$). Observe that $\mathcal{E}_i^2 = s_i\Delta_i^2 \leq s_i$, so that by assumption on T, we have

$$T \geq c_1 \cdot \frac{\log\left(\frac{1}{s_i}\right)}{s_i}. \tag{7.10}$$

Then we obtain from $\log(x)/x$ being decreasing for $x \geq e$ that holds

$$\begin{aligned}\frac{5}{\log(2)} \cdot \frac{\log(T)}{s_i \cdot T} &\leq \frac{5}{\log(2)} \cdot \frac{\log(c_1) + \log\left(\frac{1}{s_i}\right) + \log\log\left(\frac{1}{s_i}\right)}{c_1 \log\left(\frac{1}{s_i}\right)} \\ &\leq \frac{5}{\log(2)^2} \cdot \frac{\log(c_1)+2}{c_1} \leq 1 \ ,\end{aligned}$$

where the last inequality holds for $c_1 \geqslant 65$.

Now, if $c_1 \geq 65$, then $T_{d_i^*} \geqslant 1$, in which case $\beta_{d_i^*}$ is well-defined. Then, $\lfloor x \rfloor \geq x/2$ for $x \geq 1$, we can therefore obtain

$$\frac{32}{T_{d^*}} \leq 64\frac{d_{\max}(n_{d^*}+1)}{T} \leq \frac{320}{\log(2)} \cdot \frac{\log(T/\log(1/\delta))}{s_i \cdot T}$$

The objective $\Delta_i \geqslant 2\beta_{d_i^*}$ is equivalent to the condition

$$4\beta_{d_i^*}^2 = \frac{32}{T_{d_i^*}} \log\left(\frac{2d_{\max}(n_{d_i^*}+1)}{\delta}\right) \leq \Delta_i^2 \ ,$$

which holds as long as

$$\frac{320}{\log(2)} \cdot \frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log\left(\frac{10\log(T/\log(1/\delta))}{\log(2) \cdot s_i \cdot \delta}\right) \leq 1 \ .$$

Note that since $\delta < \frac{1}{4}$, we have

$$\begin{aligned}
&\frac{320}{\log(2)} \cdot \frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log\left(\frac{10\log(T/\log(1/\delta))}{\log(2) \cdot s_i \cdot \delta}\right) \\
\leq &\frac{320}{\log(2)} \cdot \frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \\
&\quad \times \left[\left(\frac{5}{\log(2)^2} + 1\right)\log(1/\delta) + \log\left(\frac{1}{s_i}\right) + \log\log(T/\log(1/\delta))\right] \ .
\end{aligned}$$

Assume first, that $\mathcal{E}_i^{-2} \geq e$. We want to determine $c_2, c_3, c_4 > 0$ such that

$$\frac{1600 + 320\log(2)^2}{\log(2)^3} \cdot \frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log(1/\delta) \leq 1/3 \qquad (7.11)$$
$$\text{for } T \geq c_2 \cdot \mathcal{E}_i^{-2} \cdot \log(\mathcal{E}_i^{-2}) \cdot \log(1/\delta) \ ;$$

and

$$\frac{320}{\log(2)} \cdot \frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log\left(\frac{1}{s_i}\right) \leq 1/3 \qquad (7.12)$$
$$\text{for } T \geq c_3 \cdot \mathcal{E}_i^{-2} \cdot \log(\mathcal{E}_i^{-2}) \cdot \log\left(\frac{1}{s_i}\right) \cdot \log\left(e \vee \log\left(\frac{1}{s_i}\right)\right) \ ;$$

and

$$\frac{320}{\log(2)} \cdot \frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log\log(T/\log(1/\delta)) \leq 1/3 \qquad (7.13)$$
$$\text{for } T \geq c_4 \cdot \mathcal{E}_i^{-2} \cdot \log(\mathcal{E}_i^{-2}) \cdot \log(\log(\mathcal{E}_i^{-2}) \vee e) \ .$$

By the monotonicity of $\log(x)/x$ for $x \geq e$, bounding T in inequality (7.11) yields

$$\begin{aligned}
\frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log(1/\delta) &\leq \frac{\log(c_2) + \log(\mathcal{E}_i^{-2}) + \log\log(\mathcal{E}_i^{-2})}{c_2 \cdot \log(\mathcal{E}_i^{-2})} \\
&\leq \frac{\log(c_2) + 2}{c_2} \leq \frac{\log(2)^3}{4800 + 960\log(2)^2}
\end{aligned}$$

for $c_2 \geq 2.3 \cdot 10^5$.

A similar approach for inequality (7.12) yields

$$\begin{aligned}
&\frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log\left(\frac{1}{s_i}\right) \\
&\leq \frac{\log(c_3) + \log(\mathcal{E}_i^{-2}) + \log\log(\mathcal{E}_i^{-2}) + \log\log\left(\frac{1}{s_i}\right) + \log\log\left(e \vee \log\left(\frac{1}{s_i}\right)\right)}{c_3 \cdot \log(\mathcal{E}_i^{-2}) \cdot \log\left(e \vee \log\left(\frac{1}{s_i}\right)\right)} \\
&\leq \frac{\log(c_3) + 4}{c_3} \leq \frac{\log(2)}{960}
\end{aligned}$$

for $c_3 \geq 2 \cdot 10^4$.

In the same way, we obtain for inequality (7.13), using

$$\log\log(xy) \leq \log 2(\log x \vee y) \leq \log\log x + 2\log y \quad \text{for } x > e,\ y \geq 2 \tag{7.14}$$

that for $c_4 \geq 2$

$$\begin{aligned}
&\frac{\log(T/\log(1/\delta)) \cdot \mathcal{E}_i^{-2}}{T} \cdot \log\log(T/\log(1/\delta)) \\
&\leq \frac{\left[\log(c_4) + 3\log(\mathcal{E}_i^{-2})\right] \cdot \left[5\log\log(\mathcal{E}_i^{-2}) + 2\log(c_4)\right]}{c_4 \cdot \log(\mathcal{E}_i^{-2}) \cdot \log(\log(\mathcal{E}_i^{-2}) \vee e)} \\
&\leq \frac{2\log(c_4)^2 + 11\log(c_4) + 15}{c_4} \leq \frac{\log(2)}{960}
\end{aligned}$$

for $c_4 \geq 7.4 \cdot 10^5$.

In the case $\mathcal{E}_i^{-2} \leq e$, we want to prove

$$\begin{aligned}
\frac{(1600 + 320\log(2)^2)e}{\log(2)^3} \cdot \frac{\log(T/\log(1/\delta))}{T} \cdot \log(1/\delta) \leq 1/3 \\
\text{for } T \geq c_2' \cdot \log(1/\delta)\ ;
\end{aligned} \tag{7.15}$$

and

$$\begin{aligned}
\frac{320e}{\log(2)} \cdot \frac{\log(T/\log(1/\delta))}{T} \cdot \log\left(\frac{1}{s_i}\right) \leq 1/3 \\
\text{for } T \geq c_3' \cdot \log\left(\frac{1}{s_i}\right) \cdot \log\left(e \vee \log\left(\frac{1}{s_i}\right)\right)\ ;
\end{aligned} \tag{7.16}$$

and

$$\begin{aligned}
\frac{320e}{\log(2)} \cdot \frac{\log(T/\log(1/\delta))}{T} \cdot \log\log(T/\log(1/\delta)) \leq 1/3 \\
\text{for } T \geq c_4'\ .
\end{aligned} \tag{7.17}$$

Indeed, for inequality (7.15) we have

$$\frac{\log(T/\log(1/\delta))}{T} \cdot \log(1/\delta) \leq \frac{\log(c_2')}{c_2'} \leq \frac{\log(2)^3}{3e \cdot (1600 + 320\log(2)^2)}$$

for $c_2' \geq 5.7 \cdot 10^5$.

For inequality (7.16) we obtain

$$\frac{\log(T/\log(1/\delta))}{T} \cdot \left(\frac{1}{s_i}\right) \leq \frac{\log(c_3') + 2}{c_3'} \leq \frac{\log(2)}{960e}$$

for $c_3' \geq 4.9 \cdot 10^4$.

We finally observe that inequality (7.17) is true for $c_4' \geq 1.1 \cdot 10^5$, since $\delta < 1/4$.

All together, we have shown that if $T \geq c \cdot (\log(\mathcal{E}_i^{-2}) \vee 1) \cdot \log\left(\frac{\log(\mathcal{E}_i^{-2}) \vee 1}{s_i \cdot \delta}\right) \cdot \mathcal{E}_i^{-2}$ for $c \geq \max\{c_1, c_2, c_3, c_4, c_2', c_3', c_4'\}$, then $T_{d_i^*} \geqslant 1$ and $\Delta_i \geqslant 2\beta_{d_i^*}$. □

7.B.2 Estimation of the Jumps

Algorithm 21 takes as input a set $\mathcal{I}$ of candidate intervals, a confidence level δ, a budget T, and a target number N of jump estimates. It outputs a subset $\mathcal{J} \subset \mathcal{I}$ of accepted intervals, and a set $\mathcal{G}$ of estimated jump magnitudes for the accepted intervals.

Description of the procedure The procedure proceeds in rounds. At round k, each active interval $I \in \mathcal{I} \setminus \mathcal{J}$ receives 2^{k-1} samples at each endpoint $x_l(I)$ and $x_r(I)$ (Lines 3–4), allowing to compute empirical means $\hat{y}_l^{(k)}(I)$ and $\hat{y}_r^{(k)}(I)$. The empirical jump estimate is

$$\hat{\Delta}^{(k)}(I) = |\hat{y}_r^{(k)}(I) - \hat{y}_l^{(k)}(I)|$$

(Line 5).

An interval is accepted when this estimate exceeds a confidence threshold

$$\sqrt{2^{-(k-1)} \log\left(\frac{\pi^2 |\mathcal{I}| k^2}{3\delta}\right)}$$

(Line 6). When I is accepted at round k, its estimate $\hat{\Delta}^{(k)}(I)$ is added to $\mathcal{G}$ and the interval is added to $\mathcal{J}$ (Line 7), so it is no longer sampled in later rounds. The running budget counter τ is updated after each batch of samples (Line 9), and the while-loop (Line 2) stops once either more than N intervals are accepted ($|\mathcal{G}| \geq N$) or the next round would exceed the total budget T ($\tau + |\mathcal{I} \setminus \mathcal{J}| \cdot 2^k > T$).

This classic design yields an adaptive multiple estimation strategy: large jumps are typically accepted early, while smaller jumps receive additional rounds of sampling, under a global budget constraint. This procedure is quite standard, and is very similar to existing elimination algorithms for `TopM` identification in multi-armed bandits, see e.g., (Bubeck et al., 2013; Kalyanakrishnan

et al., 2012).

Algorithm 21: `EstimateJumps`

Input: $\mathcal{I}$ set of intervals, δ confidence parameter, T budget, N number of desired estimates

Result: $\mathcal{J}$ subset of $\mathcal{I}$, $\mathcal{G}$ estimated jumps for intervals in $\mathcal{J}$

$\hat{\Delta}(I) \leftarrow 0$ for $I \in \mathcal{I}$, $k \leftarrow 1$, $\tau \leftarrow 0$, $\mathcal{G} \leftarrow \emptyset$, $\mathcal{J} \leftarrow \emptyset$;

while $|\mathcal{G}| < N$ *and* $\tau + |\mathcal{I} \setminus \mathcal{J}| \cdot 2^k \leq T$ **do** ▷ Sample until N estimates or budget exhausted

  for $[x_l(I), x_r(I)] = I \in \mathcal{I} \setminus \mathcal{J}$ **do** ▷ Sample active intervals

    sample 2^{k-1} times from $x_l(I)$ and $x_r(I)$, store means as $\hat{y}_l^{(k)}(I)$ and $\hat{y}_r^{(k)}(I)$

    $\hat{\Delta}^{(k)}(I) \leftarrow |\hat{y}_r^{(k)}(I) - \hat{y}_l^{(k)}(I)|$ ▷ Estimate jump at endpoints

    if $\hat{\Delta}^{(k)}(I) \geq \sqrt{2^{-(k-5)} \log\left(\frac{\pi^2 \cdot |\mathcal{I}| \cdot k^2}{3\delta}\right)}$ **then** ▷ Accept if estimate exceeds threshold

      $\mathcal{G} \leftarrow \mathcal{G} \cup \{\hat{\Delta}^{(k)}(I)\}$, $\mathcal{J} \leftarrow \mathcal{J} \cup \{I\}$ ▷ Add to accepted set

    end

    $\tau \leftarrow \tau + 2^k$ ▷ Update budget

  end

  $k \leftarrow k + 1$ ▷ Double samples for next round

end

Lemma 7.B.3. . *Let $M = |\mathcal{I}|$ and number the intervals $I \in \mathcal{I}$ in a way, such that if $I = [l(I), r(I)]$ and $\Delta(I) = f(r(I)) - f(l(I))$, then $|\Delta(I_1)| \geq |\Delta(I_2)| \geq \ldots |\Delta(I_M)|$. Let $\delta < 1/2$ and consider the output of Algorithm 21 $\mathcal{J} = \{J_1, \ldots, J_{N'}\}$ and $\mathcal{G} = \left\{\hat{\Delta}(J_1), \ldots, \hat{\Delta}(J_{N'})\right\}$ such that $\hat{\Delta}(J_1) \geq \cdots \geq \hat{\Delta}(J_{N'})$.*

Then there exists an event of probability at least $1 - \delta$ such that, for every $v = 1, \ldots, N'$,

$$0 < \frac{1}{2}\hat{\Delta}(J_v) \leq |\Delta(I_v)| \leq \frac{3}{2}\hat{\Delta}(J_v) \ .$$

and

$$0 < \frac{1}{2} \cdot \hat{\Delta}(J_v) \leq |\Delta(J_v)| \leq \frac{3}{2} \cdot \hat{\Delta}(J_v) \ .$$

Additionally, if

$$T \geq c \cdot \left[\sum_{v=1}^{N} \frac{1}{\Delta(I_v)^2} \cdot \log\left(\frac{M \cdot (\log(1/\Delta(I_v)^2) \vee 1)}{\delta}\right) + (M - N) \cdot \frac{1}{\Delta(I_N)^2} \cdot \log\left(\frac{M \cdot (\log(1/\Delta(I_N)^2) \vee 1)}{\delta}\right) \right] ,$$

for $c \geq 223$, then on the same event $N' \geq N$.

Proof of Lemma 7.B.3. Let $k \geqslant 1$ be some loop iteration of the algorithm. We define the quantity $\beta_k = \sqrt{2^{-(k-3)} \log\left(\frac{\pi^2 \cdot M \cdot k^2}{3\delta}\right)}$ for twice the threshold in Line 6.

For a given interval I, the empirical means $\hat{y}_l^{(k)}(I)$ and $\hat{y}_r^{(k)}(I)$ are computed from 2^{k-1} independent samples from $f(l(I))$ and $f(r(I))$, respectively (Line 5). By Hoeffding's inequality, using the sub-Gaussian noise assumption, for any fixed interval I and iteration k, we have

$$\mathbb{P}\left(\left|\hat{\Delta}^{(k)}(I) - \Delta(I)\right| > \beta_k\right) \leq 2\exp\left(-\frac{2^{k-2} \cdot \beta_k^2}{2}\right) = \frac{6\delta}{\pi^2 \cdot M \cdot k^2} \ .$$

Since $\sum_{k\geq 1} \frac{1}{k^2} = \frac{\pi^2}{6}$, a union bound yields on an event ξ with $\mathbb{P}(\xi) \geq 1 - \delta$, on which

$$\left|\hat{\Delta}^{(k)}(I) - \Delta(I)\right| \leq \beta_k \tag{7.18}$$

for all iterations k and all intervals I considered still active at iteration k.

First, we want to show that on ξ, for every accepted interval $J_v \in \mathcal{J}$, we have $\frac{1}{2}\hat{\Delta}(J_v) \leq |\Delta(J_v)| \leq \frac{3}{2}\hat{\Delta}(J_v)$. Note that if $J_v \in \mathcal{J}$, then there exists an iteration $k^*(J_v)$ where the algorithm stops considering the interval and accepts it. In particular, at iteration $k^*(J_v)$ the condition in line 6 is satisfied, so $\hat{\Delta}(J_v) = \hat{\Delta}^{(k^*(J_v))}(J_v)$. By inequality (7.18), we know from line 6

$$\begin{aligned} &\hat{\Delta}(J_v) \geq 2\beta_{k^*(J_v)} \\ \Rightarrow &\left|\hat{\Delta}(J_v) - |\Delta(J_v)|\right| \leq \frac{1}{2}\hat{\Delta}(J_v) \ . \end{aligned} \tag{7.19}$$

This proves the second inequality of Lemma 7.B.3 and if $|\Delta(J_v)| = |\Delta(I_v)|$, this coincides with the first inequality.

So assume that $|\Delta(J_v)| < |\Delta(I_v)|$. Then by inequality (7.19), we know that

$$\frac{1}{2}\hat{\Delta}(J_v) \leq |\Delta(J_v)| < |\Delta(I_v)| \ .$$

On the other hand, the condition $|\Delta(J_v)| < |\Delta(I_v)|$ means that there exists an $w \leq v$ such that either $I_w \notin \mathcal{J}$ or $\hat{\Delta}(I_w) \leq \hat{\Delta}(J_v)$. If $I_w \notin \mathcal{J}$ this means that the condition in line 6 is false for any iteration, in particular at iteration $k^*(J_v)$. By the concentration inequality (7.18), we obtain

$$|\Delta(I_v)| \leq |\Delta(I_w)| \leq |\hat{\Delta}(I_w)^{(k^*(J_v))} - \Delta(I_w)| + \hat{\Delta}(I_w)^{(k^*(J_v))} < 3\beta_{k^*(J_v)} \leq \frac{3}{2}\hat{\Delta}(J_v) \ .$$

If $I_w \in \mathcal{J}$ but $\hat{\Delta}(I_w) \leq \hat{\Delta}(J_v)$, then there exists an iteration $k^*(I_w)$ where Algorithm 21 stops considering the interval and the condition

$$\hat{\Delta}(I_w) \geq 2\beta_{k^*(I_w)}$$

holds. If $k^*(I_w) > k^*(J_v)$, we can use the same argumentation as for the case $I_w \notin \mathcal{J}$, so assume

without loss of generality that $k^*(I_w) \leq k^*(J_v)$. Then again by inequality (7.18) we know that

$$\begin{aligned}|\Delta(I_v)| &\leq |\Delta(I_w)| \leq |\Delta(I_w) - \hat{\Delta}(I_w)| + \hat{\Delta}(I_w) \\ &\leq \beta_{k^*(I_w)} + \hat{\Delta}(I_w) \leq \frac{3}{2}\hat{\Delta}(I_w) \leq \frac{3}{2}\hat{\Delta}(J_v) \ ,\end{aligned}$$

which proves the claimed inequality.

We can prove the claim in the case $|\Delta(J_v)| > |\Delta(I_v)|$ with the same arguments. Note that in this case, there is $w \geq v$ with $\hat{\Delta}(I_w) \geq \hat{\Delta}(J_v)$.

Now, let us prove the second part of the Lemma. Consider some interval $I \in \mathcal{I}$. Assume that $3\beta_k \leq |\Delta(I)|$. Together with inequality (7.18) this implies $\hat{\Delta}(I) \geq 2\beta_k$. This means, that if we reach the smallest k, such that

$$\frac{32}{9 \cdot 2^k} \log\left(\frac{\pi^2 \cdot M \cdot k^2}{3\delta}\right) \leq \Delta(I)^2 \ ,$$

before the algorithm terminates, we will have $I \in \mathcal{J}$ and I will not be considered in later sampling steps. We have assumed, that $\delta < 1/2$, so we can rewrite

$$\log\left(\frac{\pi^2 \cdot M \cdot k^2}{3\delta}\right) \leq 2\log_2(k) + 3\log\left(\frac{M}{\delta}\right) \ .$$

If we choose

$$k = \left\lceil \log_2\left(c \cdot \frac{\log\left(\frac{M \cdot (\log(1/\Delta(I)^2) \vee 1)}{\delta}\right)}{\Delta(I)^2}\right)\right\rceil$$

for some $c > 0$ large enough, we obtain with inequality (7.14) and $\log_2(\lceil \log_2(x) \rceil) \leq 1 + \frac{1}{\log(2)} \log(\log(x) \vee 1)$ that

$$\begin{aligned}&\frac{32}{9 \cdot \Delta(I)^2} \cdot \frac{\log\left(\frac{\pi^2 \cdot M \cdot k^2}{3\delta}\right)}{2^k} \\ &\leq \frac{32}{9} \cdot \frac{4\log\left(\frac{M}{\delta}\right) + \frac{2}{\log(2)} \cdot \log\left(\log\left(c \cdot \frac{\log\left(\frac{M \cdot (\log(1/\Delta(I)^2) \vee 1)}{\delta}\right)}{\Delta(I)^2}\right) \vee 1\right)}{c \cdot \left(\log\left(\frac{M}{\delta}\right) + \log\left(\log(1/\Delta(I)^2) \vee 1\right)\right)} \\ &\leq \cdot \frac{32(4\log(2)+4)\log\left(\frac{M}{\delta}\right) + 2\log(\log(1/\Delta(I)^2) \vee 1) + 4\log(c)}{9\log(2) \cdot c \cdot \left(\log\left(\frac{M}{\delta}\right) + \log\left(\log(1/\Delta(I)^2) \vee 1\right)\right)} \\ &\leq \frac{128(\log(2)+1)}{9\log(2)} \frac{1 + \log(c)}{c} \leq 1\end{aligned}$$

for $c \geq 223$.

So for sampling the arms corresponding to interval I, we use a budget of at most

$$4c \cdot \frac{\log\left(\frac{M \cdot (\log(1/\Delta(I)^2) \vee 1)}{\delta}\right)}{\Delta(I)^2}$$

and spend the remaining budget on the intervals in $\mathcal{I} \setminus \mathcal{J}$. The algorithm stops, when we added N arms to $\mathcal{J}$, which yields the claimed bound of the total budget. □

7.B.3 Single Change Point Localization

Once we have detected an interval I containing a change point, we want to localize it, at a precision of η. We adapt for that the `Sequential halving with backtracking` (`SHB`) algorithm introduced in (Lazzaro and Pike-Burke, 2025a) as their Algorithm 2, and recall their optimal guarantees for single change point localization, with fixed budget.

Description of the procedure The algorithm takes as input a compact interval $I = [l(I), r(I)]$, a budget T, and a precision parameter $\eta > 0$. It outputs a potential change point c.

The algorithm maintains a tuple of five point arms $(l(I), l_d, c_d, r_d, r(I))$, where the boundary arms $l(I)$ and $r(I)$ are fixed throughout. At each round d, it collects τ samples from each arm and compares the resulting empirical means to decide whether to zoom into the left sub-interval (l_d, c_d), the right sub-interval (c_d, r_d). It may also backtrack to the parent window when the evidence suggests the change point lies outside the current candidate region (Lazzaro and Pike-Burke, 2025a). The procedure runs for $d_{\max} = \lceil 6 \log(|I|/\eta) \rceil$ rounds, distributing the budget evenly across rounds and arms.

The only (and minor) difference with the original `SHB` algorithm is that we stop the procedure early if $|I| \leq 2\eta$, and return the midpoint of the interval as the change point estimate. This is because in this case, we are already guaranteed to be within η of the true change point, so there is no need to further zoom in. Moreover, we adapt the procedure to an interval of size $|I|$ instead of $[0, 1]$, which only requires a rescaling of the arms and the precision parameter η.

Lemma 7.B.4 ((Lazzaro and Pike-Burke, 2025a))**.** *Assume that $x^*_{i-1} \leq l(I) < x^*_i < r(I) < x^*_{i+1}$. If $|I| \leq 2\eta$, Algorithm 22 will return $c = \frac{l(I)+r(I)}{2}$ with $|x^*_i - c| \leq \eta$.*

Else, for a budget

$$T \geq \frac{600}{\Delta_i^2}\left(\log\left(\frac{1}{\delta}\right) + 13\log\left(\frac{|I|}{4\eta}\right)\right)$$

Algorithm 22 will return c with $|x^*_i - c| \leq \eta$ with probability at least $1 - \delta$.

Proof. The first part of the lemma is immediate from the definition of the algorithm. For the second part, we can directly apply Theorem 1 from (Lazzaro and Pike-Burke, 2025a), which states that under the given budget condition, the algorithm will return an estimate c such that $|x^*_i - c| \leq \eta$ with probability at least $1 - \delta$. □

Algorithm 22: `SHB (Sequential Halving with Backtracking)`

Input: $I = [l(I), r(I)]$ compact interval, T budget, $\eta > 0$ precision parameter
Result: potential change point c

$c_1 \leftarrow \frac{l(I)+r(I)}{2}$, $l_1 \leftarrow l(I)$, $r_1 \leftarrow r(I)$;
if $|I| \leq 2\eta$ **then**
  return c_1 ▷ Return interval midpoint if target precision already achieved
end
else
  $d_{\max} \leftarrow \lceil 6 \log(|I|/\eta) \rceil$, $\tau \leftarrow \lfloor T/5d_{\max} \rfloor$, $\mathcal{A}^1 \leftarrow (l(I), l_1, c_1, r_1, r(I))$;
  for $d = 1, \ldots, d_{\max}$ **do**
    sample τ times from arms in $\mathcal{A}^d$, store means as $\hat{y}^{(d)}_{l(I)}$, $\hat{y}^{(d)}_{l_d}$, $\hat{y}^{(d)}_{c_d}$, $\hat{y}^{(d)}_{r_d}$, $\hat{y}^{(d)}_{r(I)}$;
    if $\left| \frac{\hat{y}^{(d)}_{l(I)} + \hat{y}^{(d)}_{l_d}}{2} - \frac{\hat{y}^{(d)}_{r_d} + \hat{y}^{(d)}_{r(I)}}{2} \right| \geq \max\left(\left| \frac{\hat{y}^{(d)}_{l(I)} + \hat{y}^{(d)}_{l_d} + \hat{y}^{(d)}_{r_d}}{3} - \hat{y}^{(d)}_{r(I)} \right|, \left| \hat{y}^{(d)}_{l(I)} - \frac{\hat{y}^{(d)}_{l_d} + \hat{y}^{(d)}_{r_d} + \hat{y}^{(d)}_{r(I)}}{3} \right| \right)$ **then** ▷ If Jump seems outside middle region
      $\mathcal{A}^{d+1} \leftarrow P(\mathcal{A}^d)$ ▷ Backtrack: retreat to parent region
    end
    else if $|\hat{y}^{(d)}_{l_d} - \hat{y}^{(d)}_{c_d}| \leq |\hat{y}^{(d)}_{c_d} - \hat{y}^{(d)}_{r_d}|$ **then** ▷ Jump seems closer to right
      $\mathcal{A}^{d+1} \leftarrow R(\mathcal{A}^d) = (l(I), c_d, (c_d + r_d)/2, r_d, r(I))$ ▷ Zoom right
    else ▷ Jump seems closer to left
      $\mathcal{A}^{d+1} \leftarrow L(\mathcal{A}^d) = (l(I), l_d, (l_d + c_d)/2, c_d, r(I))$ ▷ Zoom left
    end
  end
  return c_{d+1} ▷ Return final midpoint
end

7.B.4 Verification of the Localized Change Points

Description of the procedure. `VerifyCP` is given in Algorithm 23, and is a very standard two-sample test. It takes as input a candidate change point location x, an interval I containing x, a confidence parameter δ, a precision parameter η, and a budget T. It performs a simple two-sample test to check for the presence of a change point in the neighborhood of x, given by the interval $[x_-, x_+]$ where $x_- = (x - \eta) \vee \min I$ and $x_+ = (x + \eta) \wedge \max I$, which corresponds to an η-neighborhood around x included in I. As we see in Lemma 7.B.5, if there is no actual jump ($f(x_+) = f(x_-)$), the algorithm correctly returns `False` with high probability $1 - \delta$. Conversely, if there is a jump of size $\Delta = |f(x_+) - f(x_-)|$, and the budget T is sufficiently large (scaling as $1/\Delta^2$), the algorithm will correctly return `True` with high probability.

Algorithm 23: `VerifyCP`

Input: x_- left point, x_+ right point, δ confidence parameter, T budget
Result: `detection` (boolean)

sample $\lfloor T/2 \rfloor$ times from x_- and x_+, store averages as $\hat{y}_-$ and $\hat{y}_+$;
if $T \geq 2$ *and* $|\hat{y}_+ - \hat{y}_-| > \sqrt{\frac{16}{T}\log(2/\delta)}$ **then**
  `detection` $\leftarrow$ `True`
else
  `detection` $\leftarrow$ `False`;
end

Lemma 7.B.5. *Let $\delta < 1/2$ and consider the output of Algorithm 23 with input x_-, x_+, δ, and T. Let $\Delta := f(x_+) - f(x_-)$.*

1. *If $\Delta = 0$, with probability at least $1 - \delta$ Algorithm 23 returns* `False`.
2. *If $\Delta \neq 0$ and $T \geq 64 \cdot \frac{\log\left(\frac{2}{\delta}\right)}{\Delta^2}$, Algorithm 23 returns* `True` *with probability at least $1 - \delta$.*

Proof of Lemma 7.B.5. By Hoeffding's inequality, we know that for

$$\beta = \sqrt{\frac{16}{T}\log(2/\delta)}$$

holds

$$\mathbb{P}\left(|(\hat{y}_+ - \hat{y}_-) - (y_+ - y_-)| \geq \beta\right) \leq \delta \ .$$

So if $y_+ = y_-$, then $|\hat{y}_+ - \hat{y}_-| < \beta$ and Algorithm 23 returns `False`, with probability at least $1 - \delta$. If on the other hand $\Delta = |y_+ - y_-| \geq 2\beta$ this guarantees, that $|\hat{y}_+ - \hat{y}_-| > \beta$ and Algorithm 23 will return `True`. This yields the condition

$$\Delta \geq 2 \cdot \sqrt{\frac{16}{T}\log(2/\delta)} \quad \Leftrightarrow \quad T \geq 64 \cdot \frac{\log(2/\delta)}{\Delta^2} \ . \qquad \square$$

7.B.5 Proof of the Main Theorem

Proof of Theorem 7.3.1. In this proof, we gather the guarantees on our four subroutines through the following Lemmas: (i) Lemma 7.B.1 for `DetectIntervals`, (ii) Lemma 7.B.3 for `EstimateJumps`, (iii) Lemma 7.B.4 for `SHB`, (iv) Lemma 7.B.5 for `VerifyCP`.

First, we justify the correctness guarantee in point 1 of the theorem. Then, we fix a stage index k in the doubling schedule and analyse the four steps of the procedure at this stage, deriving conditions on k under which they provide an accurate estimate of N change points and the stopping condition is reached. Finally, we derive a high-probability bound on the required budget when $\delta_{\text{explore}} = \delta$, and an expectation bound when $\delta_{\text{explore}} = 1/4$.

Consider δ, η and N as in the statement of the theorem, and any environment ν with $m \geq N$ change points. We run Algorithm 20 with input parameters $(N, \delta, \eta, \delta_{\text{explore}})$, with $\delta_{\text{explore}} \in \{\delta, 1/4\}$.

Correction The correctness guarantee relies only on the verification step and is independent of the choice of δ_{explore}. Let us verify that if Algorithm 20 stops at some stage k, the set $\mathcal{C}^{(k)}$ returned by the algorithm contains N points $c_1^{(k)} < \cdots < c_N^{(k)}$ with $|x^*_{l_v} - c_v^{(k)}| \leq \eta$ for N distinct change points $1 \leqslant l_1 < l_2 < \cdots < l_N \leqslant m$.

If not, then there exists a stage index k, and some estimated change point $c_v^{(k)} \in J_v^{(k)}$, with $J_v^{(k)} \in \mathcal{J}^{(k)}$, such that $[c_v^{(k)} - \eta, c_v^{(k)} + \eta] \cap J_v^{(k)}$ does not contain any change point, but $\text{ok}_v^{(k)} = \texttt{VerifyCP}\left(l_v^{(k)}, r_v^{(k)}, \delta^{(k)}, \left\lfloor \alpha_v^{(k)} \cdot 2^k \right\rfloor \vee 1\right)$ is True.

Consider such a $c_v^{(k)}$. Then, in `VerifyCP`, for $l_v^{(k)} = (c_v^{(k)} - \eta) \vee \min(J_v^{(k)})$ and $r_v^{(k)} = (c_v^{(k)} + \eta) \wedge \max(J_v^{(k)})$, we have $f(l_v^{(k)}) = f(r_v^{(k)})$. By Lemma 7.B.5, there exists an event $\xi^{(k)}_{\text{verify},v}$ with probability at least $1-\delta^{(k)}$ such that `VerifyCP` in line 10 returns `False` for $c_v^{(k)}$. By a union bound, the probability of returning an incorrect set $C^{(k)}$ in step k is therefore bounded by $N\delta^{(k)} = \frac{\delta}{4} \cdot \frac{6}{\pi^2 (k)^2}$. Now, since

$$\sum_{k'=\lceil \log_2(N) \rceil}^{+\infty} \frac{\delta}{4} \cdot \frac{6}{\pi^2 (k')^2} \leq \frac{1}{4}\delta \ ,$$

another union bound tells us that, with probability at least $1-\delta$, the set $\mathcal{C}$ returned by Algorithm 20 contains estimates of N distinct change points at uniform precision η. This proves the correctness guarantee in point 1 of Theorem 7.3.1.

Construction of a good event for a fixed stage index k Fix for now a stage index k in the doubling schedule (the index of the global while loop), at which the budget used is $T_k = 4 \times 2^k$. We analyse the four steps of the procedure at this stage, and we derive conditions on k under which they provide an accurate estimate of N change points and the stopping condition is reached with probability at least $1 - \delta_{\text{explore}}$.

(i) Detection of changes in intervals of interest.

For a fixed stage index k, consider the set $\mathcal{I}^{(k)}$ of intervals returned by `DetectIntervals` in line 3 of Algorithm 20, with confidence parameter $\delta_{\text{explore}}/4$, and budget $T_k/4 = 2^k$.

Consider $\omega_i^a(\delta_{\text{explore}}/4)$, and $c_1 = 7.4{\cdot}10^5$ numerical constant from Lemma 7.B.1. By Lemma 7.B.1, under the condition

$$2^k \geq c_1 \cdot \max_{i=1,\dots,N} \omega_i^a \left(\frac{\delta_{\text{explore}}}{4}\right) \cdot \mathcal{E}_i^{-2} \ , \tag{7.20}$$

there is an event $\xi_{\text{detect}}^{(k)}$ with $\mathbb{P}\left(\xi_{\text{detect}}^{(k)}\right) \geq 1 - \delta_{\text{explore}}/4$, on which the set $\mathcal{I}^{(k)}$ consists of m disjoint intervals, each containing exactly one change point. Denote by $I_i^{(k)}$ the interval in $\mathcal{I}^{(k)}$ with $x_i^* \in I_i^{(k)}$; it follows from the lemma that $|I_i^{(k)}| \leqslant \frac{s_i}{2}$.

Note that since $\delta_{\text{explore}} < 1/4$ we have

$$\max_{i=1,\dots,N} \omega_i^a \left(\frac{\delta_{\text{explore}}}{4}\right) \cdot \mathcal{E}_i^{-2} \leq 2\left(\omega_1 H_{\text{detect}}^{(m)} \left(\log\left(\frac{1}{\delta_{\text{explore}}}\right) + \omega_2\right)\right) \ ,$$

so that the condition (7.20) is satisfied as soon as

$$2^k \geq 2c_1 \cdot \omega_1 H_{\text{detect}}^{(m)} \left(\log\left(\frac{1}{\delta_{\text{explore}}}\right) + \omega_2\right) \ . \tag{7.21}$$

Assume for what follows that the event $\xi_{\text{detect}}^{(k)}$ holds, and that condition (7.20) is satisfied.

(ii) Estimation of the jumps.

Consider the subset $\mathcal{J}^{(k)} \subset \mathcal{I}^{(k)}$ of intervals returned by `EstimateJumps` in line 5 of Algorithm 20, with confidence parameter $\delta_{\text{explore}}/4$ and budget $T_k/4 = 2^k$, and number of arms N. Let $\mathcal{G}^{(k)}$ denote the set of estimated jump magnitudes on the intervals in $\mathcal{J}^{(k)} = \{J_v^{(k)}\}_{v=1}^{N'}$, ordered decreasingly, so that $\hat{\Delta}(J_1^{(k)}) \geqslant \cdots \geqslant \hat{\Delta}(J_{N'}^{(k)})$, with $N' = |\mathcal{J}^{(k)}|$. On $\xi_{\text{detect}}^{(k)}$, the intervals in $\mathcal{I}^{(k)}$ are disjoint and each contains exactly one change point. In particular, the true jumps on these intervals are exactly the jump magnitudes of the m change points of f.[7] Therefore, we can apply Lemma 7.B.3 to the set of intervals $\mathcal{I}^{(k)}$ returned by `DetectIntervals` and to the corresponding jumps $\Delta_1, \dots, \Delta_m$.

By Lemma 7.B.3, there exists an event $\xi_{\text{estimate}}^{(k)}$ with $\mathbb{P}(\xi_{\text{estimate}}^{(k)}) \geq 1 - \delta_{\text{explore}}/4$, so that, with c_2 a numerical constant from Lemma 7.B.3, under the condition

$$\begin{aligned} 2^k \geq c_2 \cdot \Bigg[& \sum_{i=1}^{N} \frac{1}{\Delta_{(i)}^2} \cdot \log\left(\frac{4m \cdot (\log(1/\Delta_{(i)}^2) \vee 1)}{\delta_{\text{explore}}}\right) \\ & + (m - N) \cdot \frac{1}{\Delta_{(N)}^2} \cdot \log\left(\frac{4m \cdot (\log(1/\Delta^2(I_{(N)})) \vee 1)}{\delta_{\text{explore}}}\right) \Bigg] \end{aligned} \tag{7.22}$$

`EstimateJumps` returns a set $\mathcal{G}^{(k)} = \{\hat{\Delta}(J_v^{(k)}),\ v = 1, \dots, N'\}$, with $N' \geq N$, with $\hat{\Delta}(J_1^{(k)}) \geq$

7. Recall that we defined $\Delta(I) = f(l) - f(r)$ for the jump across an interval $I = [l, r]$.

$\hat{\Delta}(J_2^{(k)}) \geq \cdots \geq \hat{\Delta}(J_{N'}^{(k)})$, and such that for $v = 1, \ldots, N$,

$$\frac{1}{2}\hat{\Delta}(J_v^{(k)}) \leq |\Delta(J_v^{(k)})| \leq \frac{3}{2}\hat{\Delta}(J_v^{(k)}) \tag{7.23}$$

$$\frac{1}{2}\hat{\Delta}(J_v^{(k)}) \leq |\Delta_{(v)}| \leq \frac{3}{2}\hat{\Delta}(J_v^{(k)}) \ . \tag{7.24}$$

These inequalities will allow us to verify that the proportion $\alpha_v^{(k)}$ of the budget allocated to interval $J_v^{(k)}$ is proportional to $\Delta_{(v)}^{-2}$, which we interpret as the optimal allocation for localizing the change point in $J_v^{(k)}$.

Let us now examine condition 7.22. Naturally, estimating the jumps once they have been detected is less costly than the detection step itself. First, the energy $\mathcal{E}_i^2$ is the product of Δ_i^2 and a spacing term, so $\Delta_i^{-2} \leq \mathcal{E}_i^{-2}$ for all i. This implies that all logarithmic terms in the condition 7.22 are upper bounded by $2(\omega_2 + \log(1/(\delta_{\text{explore}})))$, with ω_2 defined in the statement of the theorem. Secondly, one has

$$\sum_{i=1}^{N} \frac{1}{\Delta_{(i)}^2} + (m-N) \cdot \frac{1}{\Delta_{(N)}^2} \leqslant \sum_{i=1}^{m} \frac{1}{\Delta_{(i)}^2} \leqslant \sum_{i=1}^{m} \frac{1}{s_i \Delta_{(i)}^2} s_i \leq \left(\max_{j=1}^{m} \mathcal{E}_j^{-2} \right) \sum_{i=1}^{m} s_i \leqslant 2H_{\text{detect}}^{(m)} \ ,$$

where the last inequality holds since $\sum_{i=1}^m s_i \leq 2$, and $H_{\text{detect}}^{(m)} = \max_{i=1}^{m} \mathcal{E}_i^{-2}$.

All together, condition 7.22 is implied by

$$2^k \geq 2c_2 \cdot H_{\text{detect}}^{(m)} \cdot (\omega_2 + \log(1/\delta_{\text{explore}})) \tag{7.25}$$

Now, we assume that both events $\xi_{\text{detect}}^{(k)}$ and $\xi_{\text{estimate}}^{(k)}$ hold, and that conditions (7.20) and (7.22) are satisfied. We are now in a position to analyse the last two steps of the procedure.

(iii) Localization of the change points.

Let $v \in \{1, \ldots, N\}$, and consider the interval $J_v^{(k)}$ returned by `EstimateJumps`, with the v-th largest estimated jump $\hat{\Delta}(J_v^{(k)})$. From the guarantees on `DetectIntervals`, we know that $J_v^{(k)}$ contains exactly one change point, with jump $\Delta(J_v^{(k)})$, and that the if this unique change point is $x_{j_v}^*$, then $|\Delta(J_v^{(k)})| = |\Delta_{j_v}|$, and $|J_v^{(k)}| \leq s_{j_v}$.

Consider the proportion $\alpha_v^{(k)}$ of the budget allocated to interval $J_v^{(k)}$ for localization, defined in Line 9 of Algorithm 20. Let $c_v^{(k)}$ denote the output of the SHB algorithm (Algorithm 22) applied to $J_v^{(k)}$ with budget $\lfloor \alpha_v^{(k)} \cdot 2^k \rfloor \vee 1$, where $\alpha_v^{(k)}$ is defined in 7.6.

If $|J_v^{(k)}| \leq 2\eta$, Algorithm 22 takes the midpoint of the interval as the change-point estimate $c_v^{(k)}$, and we have the guarantee $|c_v^{(k)} - x_{j_v}^*| \leq \eta$. Otherwise, by Lemma 7.B.4 from (Lazzaro and Pike-Burke, 2025a), there exists an event $\xi_{\text{loc},v}^{(k)}$ of probability at least $1 - \delta_{\text{explore}}/4N$ such that, under the condition

$$\lfloor \alpha_v^{(k)} \cdot 2^k \rfloor \vee 1 \geq \frac{7200}{\Delta_{j_v}^2} \log\left(\frac{N}{\eta \cdot \delta_{\text{explore}}} \right) \ , \tag{7.26}$$

an estimate $c_v^{(k)} \in J_v^{(k)}$ with $|c_v^{(k)} - x_{j_v}^*| \leq \eta$.

Let us now find a condition on k that implies the condition (7.26) for any $v = 1, \ldots, N$. From Equations (7.23) and (7.24), we have $|\Delta(J_v^{(k)})| \geq \frac{1}{2}\hat{\Delta}(J_v^{(k)}) \geqslant \frac{1}{2}\frac{2}{3}\Delta_{(v)} = \frac{1}{3}\Delta_{(v)}$, where $\Delta_{(v)}$ is the v-th largest true jump. In particular, this implies that

$$H_{\text{localize}}^{(N)} = \sum_{w=1}^{N} \Delta_{(w)}^{-2} \geq \frac{1}{9} \sum_{w=1}^{N} \Delta(J_w^{(k)})^{-2} \ .$$

Moreover, from inequality (7.23), we also obtain for any $v = 1, \ldots, N$,

$$\alpha_v^{(k)} = \frac{\left(\hat{\Delta}(J_v)\right)^{-2}}{\sum_{w=1}^{N}\left(\hat{\Delta}(J_w)\right)^{-2}} \geq \frac{(2\Delta_{j_v})^{-2}}{\sum_{w=1}^{N}\left(\frac{2}{3}\Delta_{j_w}\right)^{-2}} = \frac{1}{9}\frac{\Delta_{j_v}^{-2}}{\sum_{w=1}^{N}\Delta_{j_w}^{-2}} \ .$$

Now, under the assumption

$$2^k \geq 1263600 \log\left(\frac{N}{\eta \cdot \delta_{\text{explore}}}\right) \sum_{w=1}^{N} \frac{1}{\Delta_{(v)}^2} \geq 140400 \log\left(\frac{N}{\eta \cdot \delta_{\text{explore}}}\right) \sum_{w=1}^{N} \frac{1}{\Delta_{j_w}^2}$$

we obtain

$$\left\lfloor \alpha_{i_v}^{(k)} \cdot 2^k \right\rfloor \geq \frac{7200}{\Delta_{j_v}^2} \log\left(\frac{N}{\eta \cdot \delta_{\text{explore}}}\right) \ .$$

By a union bound, we obtain an event $\xi_{\text{localize}}^{(k)} = \cap_{v=1}^{N} \xi_{\text{loc},v}^{(k)}$ with $\mathbb{P}(\xi_{\text{localize}}^{(k)}) \geq 1 - \delta/4$, on which $|c_v^{(k)} - x_{j_v}^*| \leq \eta$. This yields following sufficient condition for localization:

$$2^k \geq 1263600 \cdot H_{\text{localize}}^{(N)} \cdot \left(\log\left(\frac{1}{\delta_{\text{explore}}\eta}\right) + \omega_3\right) \ , \tag{7.27}$$

using $\omega_3 = \log\left(N \cdot \log(H_{\text{localize}}^{(N)})\right)$ as defined in the theorem.

For now, assume that we are on the event $\xi_{\text{detect}}^{(k)} \cap \xi_{\text{estimate}}^{(k)} \cap \xi_{\text{localize}}^{(k)}$.

(iv) Verification of the localized change points.

Finally, we seek a condition on k such that, if the previous steps go well, the verification step also succeeds and the algorithm stops. This will hold under the following condition:

$$2^k = c' \cdot \log\left(\frac{N \cdot \left(\log\left(\sum_{w=1}^{N} \frac{1}{\Delta_{(w)}^2}\right) \vee 1\right)}{\delta}\right) \cdot \sum_{w=1}^{N} \frac{1}{\Delta_{(w)}^2},$$

where $c' > 0$ is computed later. If we can show that

$$\alpha_{i_v}^{(k)} \cdot 2^k \geq 128 \cdot \frac{\log\left(\frac{4\pi^2 \cdot N \cdot k^2}{3 \cdot \delta}\right)}{\Delta_{j_v}^2} \quad , \quad i = 1, \dots, N \ ,$$

then

$$\left\lfloor \alpha_{i_v}^{(k)} \cdot 2^k \right\rfloor \geq 64 \cdot \frac{\log\left(\frac{4\pi^2 \cdot N \cdot k^2}{3 \cdot \delta}\right)}{\Delta_{j_v}^2} \quad , \quad v = 1, \dots, N \ ,$$

and Lemma 7.B.5 together with a union bound show that, on the event $\xi_{\text{verify}}^{(k)}$ with $\mathbb{P}(\xi_{\text{verify}}^{(k)}) \geq 1 - \frac{3\delta}{2\pi^2 k^2}$ (already introduced in the correctness proof), `VerifyCP` in line 10 returns `True` for all $i = 1, \dots, N$.

Note that we have by the inequalities (7.23) and (7.24) that

$$\begin{aligned}
&\alpha_{i_v}^{(k)} \cdot c' \cdot \log\left(\frac{N \cdot \left(\log\left(\sum_{w=1}^N \frac{1}{\Delta_{(w)}^2}\right) \vee 1\right)}{\delta}\right) \cdot \sum_{w=1}^N \frac{1}{\Delta_{(w)}^2} \\
&\geq \frac{c'}{9\Delta_{j_v}^2} \cdot \log\left(\frac{N \cdot \left(\log\left(\frac{1}{9}\sum_{i=1}^N \frac{1}{\Delta_{j_w}^2}\right) \vee 1\right)}{\delta}\right) \\
&\geq \frac{c'}{18\Delta_{j_v}^2} \cdot \log\left(\frac{N \cdot \left(\log\left(\sum_{i=1}^N \frac{1}{\Delta_{j_w}^2}\right) \vee 1\right)}{\delta}\right) \ .
\end{aligned}$$

The last inequality follows from the fact that for $\delta < 1/4$ and $x > 0$, we can bound

$$\log\left(\frac{N \cdot (\log(\frac{1}{9}x) \vee 1)}{\delta}\right) \geq \frac{1}{2}\log\left(\frac{N \cdot (\log(x) \vee 1)}{\delta}\right) \ ,$$

since

$$\begin{aligned}
&\log\left(\frac{N \cdot (\log(\frac{1}{9}x) \vee 1)}{\delta}\right) - \frac{1}{2}\log\left(\frac{N \cdot (\log(x) \vee 1)}{\delta}\right) \\
&= \frac{1}{2}\log\left(\frac{N}{\delta}\right) + \log\left(\log\left(\frac{1}{9}x\right) \vee 1\right) - \frac{1}{2}\log\left(\log\left(x\right) \vee 1\right) \\
&\geq \log(2) + \log\left(\left(\log\left(x\right) - \log(9)\right) \vee 1\right) - \frac{1}{2}\log\left(\log\left(x\right) \vee 1\right) \quad =: g(x).
\end{aligned}$$

The function g is continuous $[0, \infty)$ with $g(x) = \log(2)$ on $[0, e]$, $g(x)$ decreasing on $(e, 9e)$ with $g(9e) = \log(2) - \frac{1}{2}\log(1 + \log(9)) \geq 0$ and increasing on $(9e, \infty)$, so we have indeed $g(x) \geq 0$.

Therefore, we obtain

$$\frac{\log\left(\frac{4\pi^2\cdot N\cdot k^2}{3\cdot\delta}\right)}{\alpha_{i_v}^{(k)}\cdot 2^k\cdot\Delta_{j_v}^2} \leq \frac{182\log\left(\frac{N}{\delta}\right)+102\log\left(\log\left(\sum_{w=1}^N\frac{1}{\Delta_{(w)}^2}\right)\vee 1\right)+72\log(c')}{c'\cdot\left[\log\left(\log\left(\sum_{w=1}^N\frac{1}{\Delta_{j_w}^2}\right)\vee 1\right)+\log\left(\frac{N}{\delta}\right)\right]}$$
$$\leq \frac{182+72\log(c_4)}{c_4} \leq \frac{1}{128}$$

when $c_4 > 1.4\cdot 10^5$. For the second inequality, we used Equation (7.14) to obtain

$$\log\log(2^k) \leq \log\left(\log\left(\sum_{w=1}^N\frac{1}{\Delta_{(w)}^2}\right)\vee 1\right) + 2\log\left(c'\cdot\log\left(\frac{N\cdot\left(\log\left(\sum_{w=1}^N\frac{1}{\Delta_{(w)}^2}\right)\vee 1\right)}{\delta}\right)\right)$$
$$\leq 2\log\left(\frac{N}{\delta}\right)+3\log\left(\log\left(\sum_{w=1}^N\frac{1}{\Delta_{(w)}^2}\right)\vee 1\right)+2\log(c') \ .$$

All together, using $\omega_3 = \log\left(N\cdot\left(\log\left(H_{\text{localize}}^{(N)}\right)\vee 1\right)\right)$, we have shown that under the condition

$$2^k \geq 1.4\cdot 10^5\cdot(\omega_3+\log(1/\delta))\cdot H_{\text{localize}}^{(N)} \ , \tag{7.28}$$

then on the event $\xi_{\text{verify}}^{(k)}$, `VerifyCP` in line 10 returns `True` for all $v = 1,\dots,N$.

Bound in high probability. Now, we are ready to derive a high probability bound on the required budget when $\delta_{\text{explore}} = \delta$. We have a constant c such that for any $k \geq \lceil\log_2(N)\rceil$, if k satisfies

$$2^k \geq c\cdot\left(\omega_1 H_{\text{detect}}^{(m)}\left(\log\left(\frac{1}{\delta}\right)+\omega_2\right)+H_{\text{localize}}^{(N)}\left(\log\left(\frac{1}{\delta\eta}\right)+\omega_3\right)\right) \ , \tag{7.29}$$

then all conditions (7.21),(7.25),(7.27),(7.28) are also verified (with $\delta_{\text{explore}} = \delta$). From the previous paragraph, under this budget, on the intersection $\xi^{(k)} := \xi_{\text{detect}}^{(k)}\cap\xi_{\text{estimate}}^{(k)}\cap\xi_{\text{localize}}^{(k)}\cap\xi_{\text{verify}}^{(k)}$, the stopping condition at the k-th stage of Algorithm 20 is reached. It is in particular sufficient to choose k_* as the smallest integer such that this bound (7.29) holds. Now, on the event $\xi = \xi_{\text{detect}}^{(k_*)}\cap\xi_{\text{estimate}}^{(k_*)}\cap\xi_{\text{localize}}^{(k_*)}\cap(\cap_{k'=1}^{k_*}\xi_{\text{verify}}^{(k')})$, then the algorithm stops at stage k_* (or before), and the output is correct.

Finally, one has $\mathbb{P}(\xi^{(k_*)}) \geq 1-\frac{3}{4}\delta-\sum_{k'=1}^{k_*}\frac{6}{\pi^2k'^2}\cdot\frac{\delta}{4} \geqslant 1-\delta$. Moreover, on ξ, the total budget used by the algorithm can be bounded as $\sum_{k'=\lceil\log_2(N)\rceil}^{k_*}4\cdot 2^{k'} < 8\cdot 2^{k_*}$, which gives us the claimed bound on the total budget. This concludes the proof of the high probability bound in point 2 of Theorem 7.3.1.

Bound in expectation. Finally, we can also derive a bound on the expected budget. Fix $\delta_{\text{explore}} = 1/4$.

From the analysis of the four steps of the procedure, there exists a universal constant c such that for any $k \geq \lceil \log_2(N) \rceil$, if k satisfies

$$2^k \geq c \cdot \left(\omega_1 H^{(m)}_{\text{detect}} \left(\log(4) + \omega_2 \right) + H^{(N)}_{\text{localize}} \left(\log(4/\delta\eta) + \omega_3 \right) \right) \quad , \tag{7.30}$$

then all conditions (7.21),(7.25),(7.27),(7.28) are verified. Under this budget constraint, on the intersection $\xi^{(k)} := \xi^{(k)}_{\text{detect}} \cap \xi^{(k)}_{\text{estimate}} \cap \xi^{(k)}_{\text{localize}} \cap \xi^{(k)}_{\text{verify}}$, the stopping condition at the k-th stage of Algorithm 20 is reached. Moreover, $\mathbb{P}(\xi^{(k)}) \geq 1 - \frac{3}{4} \cdot \frac{1}{4} - \sum_{k'=\lceil \log_2(N) \rceil}^{k} \frac{6}{\pi^2 k'^2} \cdot \frac{\delta}{4} \geq \frac{3}{4}$, as $\delta \leqslant 1/4$ and $\delta_{\text{explore}} = 1/4$.

Now, consider k_0, the smallest integer such that condition (7.30) is satisfied. Then, on the event $\xi^{(k_0)}$, for any $k \geq k_0$, the stopping condition at the k-th stage of Algorithm 20 is reached, and the output is correct with probability at least $\frac{3}{4}$. Therefore, if we denote by K the random variable corresponding to the stage at which the algorithm stops, then $(K - k_0) \vee 0$ is stochastically dominated by a geometric random variable with parameter $3/4$, and $\mathbb{E}[2^{(K-k_0)\vee 0}] \leqslant \sum_{l=0}^{\infty} 2^l (1/4)^l (3/4) = \frac{3}{2}$.

Moreover, the total budget used by the algorithm can be bounded as

$$\mathcal{T} \leqslant \sum_{k'=\lceil \log_2(N) \rceil}^{K} 4 \cdot 2^{k'} < 8 \cdot 2^{k_0} \cdot 2^{(K-k_0)\vee 0} \quad ,$$

and

$$\begin{aligned} \mathbb{E}[\mathcal{T}] &\leq 8 \cdot 2^{k_0} \cdot \mathbb{E}[2^{(K-k_0)\vee 0}] \leq 12 \cdot 2^{k_0} \\ 2^{k_0} &\leq 2c \cdot \left(\omega_1 H^{(m)}_{\text{detect}} \left(\log(4) + \omega_2 \right) + H^{(N)}_{\text{localize}} \left(\log(4/\delta\eta) + \omega_3 \right) \right) \quad . \end{aligned}$$

This concludes the proof of the bound in expectation in point 3 of Theorem 7.3.1. □

7.C Proofs of Lower Bounds

7.C.1 Proof of Theorem 7.4.1 and Corollary 7.4.2

Before proving Theorem 7.4.1, we first state two lemmas, which are the main building blocks of the proof. The first is a lower bound showing the optimality of the localization complexity $H^{(m)}_{\text{localize}}$, and the second is a lower bound scaling with $H^{(m)}_{\text{detect}}$.

Fix $\delta \in (0, 1/4)$ and $\eta \in (0, 1/8)$. For both lemmas, we consider any (δ, η, m)-correct algorithm π for the active change point detection problem, and any valid environment ν with $m \geqslant 1$ change points. We consider the case where all change points have to be localized, so that $N = m$. In all this section, we may omit the superscript (m) in $H^{(m)}_{\text{detect}}$ and $H^{(m)}_{\text{localize}}$ for readability, and simply write H_{detect} and H_{localize}. We will add the dependence of these quantities on the environment ν when needed, and write H^{ν}_{detect} and $H^{\nu}_{\text{localize}}$.

Lemma 7.C.1. *There exists an environment ν' such that (1) ν' and ν have the same jump magnitudes $\Delta_1, \dots, \Delta_m$ — so that $H_{\mathrm{localize}}^{(m,\nu')} = H_{\mathrm{localize}}^{(m,\nu)}$; (2) $\forall i \in 1, \dots, m$, $\frac{1}{4}s_i \leqslant s_i' \leqslant \frac{3}{4}s_i$ — so that $\frac{4}{3}H_{\mathrm{detect}}^{(m,\nu)} \leqslant H_{\mathrm{detect}}^{(m,\nu')} \leqslant 4H_{\mathrm{detect}}^{(m,\nu)}$; and*

$$\mathbb{P}_{\pi,\nu'}\left(\mathcal{T}_\pi \geqslant H_{\mathrm{localize}}^{(m,\nu)} \log\left(\frac{1}{8\delta}\right) + \sum_{i=1}^{m} \frac{1}{\Delta_i^2} \log_+\left(\frac{s_i}{16\eta}\right)\right) \geqslant \delta \ .$$

Lemma 7.C.2. *Assume that ν contains at least 2 change points, $N = m \geqslant 2$. Then, there exists an environment ν' such that $H_{\mathrm{detect}}^{(m,\nu)} \leqslant H_{\mathrm{detect}}^{(m,\nu')} \leqslant 4H_{\mathrm{detect}}^{(m,\nu)}$, $\frac{1}{2}H_{\mathrm{localize}}^{(m,\nu)} \leqslant H_{\mathrm{localize}}^{(m,\nu')} \leqslant H_{\mathrm{localize}}^{(m,\nu)}$, and*

$$\mathbb{P}_{\pi,\nu'}\left(\mathcal{T}_\pi \geqslant \frac{1}{2}H_{\mathrm{detect}}^{(m,\nu)} \log\left(\frac{1}{6\delta}\right)\right) \geqslant \delta \ .$$

From these two lemmas, we can now deduce Theorem 7.4.1.

proof of Theorem 7.4.1. The proof is a direct consequence of Lemmas 7.C.1 and 7.C.2.

First, assume that $N = m = 1$. Recall that in this case there is a single change point with energy $\mathcal{E}_1 = \Delta_1^2$ and sparsity $s_1 = 1$, so $H_{\mathrm{detect}} = H_{\mathrm{localize}}$. We apply Lemma 7.C.1 to obtain an environment ν' such that $H_{\mathrm{localize}}^{(m,\nu)} = H_{\mathrm{localize}}^{(m,\nu')}$ (and similarly for H_{detect}) and such that

$$\mathbb{P}_{\pi,\nu'}\left(\mathcal{T}_\pi \geqslant \frac{1}{\Delta_1^2} \log\left(\frac{1}{8\delta}\right) + \frac{1}{\Delta_1^2} \log_+\left(\frac{1}{16\eta}\right)\right) \geqslant \delta \ ,$$

and the claimed bound follows from the fact that $H_{\mathrm{detect}} = H_{\mathrm{localize}}$ in this case.

Second, assume that $N = m \geqslant 2$. We apply Lemma 7.C.2, and Lemma 7.C.1 to obtain an environment ν' such that $H_{\mathrm{detect}}^{(m,\nu)} \leqslant H_{\mathrm{detect}}^{(m,\nu')} \leqslant 4H_{\mathrm{detect}}^{(m,\nu)}$, $\frac{1}{2}H_{\mathrm{localize}}^{(m,\nu)} \leqslant H_{\mathrm{localize}}^{(m,\nu')} \leqslant H_{\mathrm{localize}}^{(m,\nu)}$, and $\mathbb{P}_{\pi,\nu'}\left(\mathcal{T}_\pi \geqslant \chi\right) \geqslant \delta$, with

$$\begin{aligned}\chi &= \max\left(\frac{1}{2}H_{\mathrm{detect}}^{(m,\nu)} \log\left(\frac{1}{6\delta}\right), H_{\mathrm{localize}}^{(m,\nu)} \log\left(\frac{1}{8\delta}\right) + \sum_{i=1}^{m} \frac{1}{\Delta_i^2} \log_+\left(\frac{s_i}{16\eta}\right)\right) \\ &\geqslant \frac{1}{4}H_{\mathrm{detect}}^{(m,\nu)} \log\left(\frac{1}{8\delta}\right) + \frac{1}{2}H_{\mathrm{localize}}^{(m,\nu)} \log\left(\frac{1}{8\delta}\right) + \frac{1}{2}\sum_{i=1}^{m} \frac{1}{\Delta_i^2} \log_+\left(\frac{s_i}{16\eta}\right) \ ,\end{aligned}$$

which concludes the proof. □

Now, we can deduce Corollary 7.4.2 from Theorem 7.4.1, and Theorem 5.2 from (Lazzaro and Pike-Burke, 2025b).

Proof of Corollary 7.4.2. First, observe that for any environment ν with $m \geqslant 1$ change points, the condition $\vartheta_i > 2\eta$ for $i = 2, \dots, m-1$ allows us to apply Theorem 5.2 from (Lazzaro and Pike-Burke, 2025b), which states that for any (δ, η, m)-correct algorithm π,

$$\mathbb{E}_{\pi,\nu}[\mathcal{T}_\pi] \geqslant 4H_{\mathrm{localize}}^{(m,\nu)} \log\left(\frac{1}{4\delta}\right) \ . \tag{7.31}$$

Second, we obtain the lower bound in $\mathbb{E}[\mathcal{T}_\pi]$ independent of δ by applying Theorem 7.4.1 with $\delta = 1/16$, observing that ν is in particular $(1/16, \eta, m)$-correct, as $\delta \leqslant 1/16$. From Theorem 7.4.1, there exists an environment ν' such that $H^{(m,\nu)}_{\text{detect}} \leqslant H^{(m,\nu')}_{\text{detect}} \leqslant 4H^{(m,\nu)}_{\text{detect}}$, $\frac{1}{2}H^{(m,\nu)}_{\text{localize}} \leqslant H^{(m,\nu')}_{\text{localize}} \leqslant H^{(m,\nu)}_{\text{localize}}$, and such that

$$\mathbb{P}_{\pi,\nu'}\left(\mathcal{T}_\pi \geqslant \tfrac{1}{4}H^{(m,\nu)}_{\text{detect}}\log(2) + \tfrac{1}{2}H^{(m,\nu)}_{\text{localize}}\log(2) + \tfrac{1}{2}\sum_{i=1}^{m}\frac{1}{\Delta_i^2}\log_+\left(\frac{s_i}{16\eta}\right)\right) \geqslant 1/16 \ .$$

Now, from Markov's inequality, we can lower bound $\mathbb{E}_{\pi,\nu'}[\mathcal{T}_\pi]$ as

$$\mathbb{E}_{\pi,\nu'}[\mathcal{T}_\pi] \geqslant \frac{1}{16}\left(\frac{1}{4}H^{(m,\nu)}_{\text{detect}}\log(2) + \frac{1}{2}H^{(m,\nu)}_{\text{localize}}\log(2) + \frac{1}{2}\sum_{i=1}^{m}\frac{1}{\Delta_i^2}\log_+\left(\frac{s_i}{16\eta}\right)\right) \tag{7.32}$$

Now, the final lower bound from Corollary 7.4.2 follows from Equations (7.31),(7.32), and the standard inequality $\max(a,b) \geqslant (a+b)/2$ for $a, b \geqslant 0$.

□

7.C.2 Proof of Lemma 7.C.1

Proof of Lemma 7.C.1. **Proof roadmap.** The proof has four steps.

First, we construct a family $(\nu_J)_J$, indexed by $J = (j_1, \dots, j_m) \in \prod_{i=1}^m \{0, \dots, \alpha_i - 1\}$, by shifting each change point x_i^* over $\alpha_i \sim s_i/8\eta$ candidate positions. We pay attention to preserve roughly the length of the change points, so that the complexities $H^{(m,\nu)}_{\text{detect}}$ is preserved (up to a numerical constant) under any of the considered shifts. Second, for each $i \in \{1, \dots, m\}$, we define an alternative environment $\nu_{J^{(-i)}}$ by removing from ν_J the i-th gap. Third, we compare the behavior of the algorithm under ν_J and $\nu_{J^{(-i)}}$. For that, we consider the event $\xi_{i,j_i} = \{\mathcal{T}_\pi \leqslant \chi,\ |\hat{x}_i - x_i^J| \leqslant \eta\}$, where x_i^J is the position of the i-th change point in environment ν_J, and χ is the largest $(1-\delta)$-quantile of the budget under the family of environments $(\nu_J)_J$. We show that ξ_{i,j_i} is likely under ν_J and unlikely on average under $\nu_{J^{(-i)}}$. These bounds allow us to lower bound the KL between the respective probability distributions, thanks to data-processing inequalities. Finally, we compute the KL divergence between the probability under ν_J and $\nu_{J^{(-i)}}$, which leads to a lower bound on the quantile budget χ, which translates Lemma 7.C.1.

Reference class of environments.

Consider the parameters of the environment ν given by the gaps $\Delta_1, \dots, \Delta_m$, the initial mean μ_0, and the change point positions $x_1^*, \dots, x_m^*$, and the function f defined by Equation 7.3. Recall that $x_0^* = 0$ and $x_{m+1}^* = 1$. Consider the length of the change points $\vartheta_0 = 1$, for $i = 1, \dots, m-1$ $\vartheta_i = x_{i+1}^* - x_i^*$, and $\vartheta_m = 1$.

Fix $\eta' > \eta$, which will be used to shift the change points. For $i = 1, \dots, m$, define

$$\alpha_i := 1 \vee \left\lfloor \frac{s_i}{8\eta'} \right\rfloor \ ,$$

as the number of possible positions for the change point i in the environments that we will

construct. Observe that $\alpha_i \geqslant 1$ for all $i = 1, \dots, m$.

Let $J = (j_1, \dots, j_m)$ be a vector of size m, so that for each $i = 1, \dots, m$, $0 \leqslant j_i \leqslant \alpha_i - 1$. For each $i = 1, \dots, m$, define

$$x_i^J = \frac{1}{2} x_i^* + j_i \cdot 2\eta' \tag{7.33}$$

By convention, $x_0^J := 0$ and $x_{m+1}^J := 1$.

For each $J = (j_1, \dots, j_m)$, define ν_J as the environment with the same gaps as ν, and where the position of the change points are given by the vector $(x_i^J)_{i=1,\dots,m}$. Note that ν_J has also m change points and that the gaps are unchanged. Moreover, for each $i = 1, \dots, m-1$, the change points x_i^J and x_{i+1}^J are spaced by at least $\vartheta_i/4$ and at most $(3/4)\vartheta_i$, so that the complexities $H_{\mathrm{detect}}^{(m,\nu)}$ are preserved (up to a numerical constant) under any of the considered shifts. We point to Lemma 7.C.3 for a formal statement of this fact.

Denote as $\mathbb{P}_{\pi,J}$ the probability distribution induced by the interaction between an algorithm π, and the environment is ν_J, and by $\mathbb{E}_{\pi,J}$ the associated expectation.

Alternative environments.

Let $i \in \{1, \dots, m\}$ and fix for now $j_i \in \{0, \dots, \alpha_i - 1\}$. Fix any $J \in \prod_{k=1}^m \{0, \dots, \alpha_k - 1\}$, such that j_i is the i-th component of J. Denote as $J^{(-i)}$ the vector obtained from J by removing the i-th component j_i. Define $\nu_{J^{(-i)}}$ as the environment obtained from ν_J by pulling to zero the gap Δ_i, so that $\nu_{J^{(-i)}}$ has $m-1$ change points. Note that $\nu_{J^{(-i)}}$ and ν_J only differ for arms in the interval $[x_i^J, x_{i+1}^J)$, and are identical elsewhere. Observe moreover that $\nu_{J^{(-i)}}$ does not depend on the index j_i, as the change point i is pulled to zero.

Denote as $\mathbb{P}_{\pi,J^{(-i)}}$ the probability distribution of the observations when the algorithm is π, and the environment is $\nu_{J^{(-i)}}$ and by $\mathbb{E}_{\pi,J^{(-i)}}$ the associated expectation.

Bound on total variation.

Let π be an algorithm for this problem, which is (δ, η)-correct, and let $\hat{x}_1, \dots, \hat{x}_m$ be the corresponding estimates of the change points. Denote as $\mathcal{T}_\pi$ the stopping time of π. Define χ as the supremum of the $(1-\delta)$-quantile of the budget over the family of probability distributions $\mathbb{P}_{\pi,J}$ for $J \in \prod_{i=1}^m \{0, \dots, \alpha_i - 1\}$. Namely,

$$\chi = \inf\{t > 0 : \forall J \in \prod_{i=1}^m \{0, \dots, \alpha_i - 1\}, \mathbb{P}_{\pi,J}(\mathcal{T}_\pi \geqslant t) \leqslant \delta\} \ . \tag{7.34}$$

Let $i \in \{1, \dots, m\}$. Let J be a vector of m indices, whose i-th component is j_i. Define the event

$$\xi_{i,j_i} = \{\mathcal{T}_\pi \leqslant \chi, \ |\hat{x}_i - x_i^J| \leqslant \eta\} \ .$$

Recall that the environment ν_J contains m change points, and the i-th change point is x_i^J. Then, it follows from the (δ, η)-correctness of π that

$$\mathbb{P}_{\pi,J}(|\hat{x}_i - x_i^J| > \eta) \leqslant \delta \ .$$

Moreover, by definition of χ (see (7.34)), it holds that

$$\mathbb{P}_{\pi,J}(\mathcal{T}_\pi \geqslant \chi) \leqslant \delta \ .$$

Therefore, by union bound, we have that, for all J such that j_i is the i-th component of J,

$$\mathbb{P}_{\pi,J}(\xi_{i,j_i}) \geqslant 1 - 2\delta \ . \tag{7.35}$$

Consider now the environment $\nu_{J^{(-i)}}$, which differs from ν_J by the fact that the gap at change point i is pulled to zero. Under this environment, the change point i is not identifiable, and the algorithm should not be able to identify it in finite time. In particular, we have the following bound, stated as Lemma 7.C.4, whose proof is deferred at the end of this section, and which hold as long as $\eta < 1/8$.

$$\text{For all } J^{(-i)}, \text{ it holds that} \quad \mathbb{P}_{\pi,J^{(-i)}}(\mathcal{T}_\pi \leqslant \chi) \leqslant 2\delta \ . \tag{7.36}$$

Let $j_i \in \{0, \dots \alpha_i - 1\}$ and recall that the position of the change point i in the environment ν_J is $x_i^J = x_i^* + j_i \cdot 2\eta'$. In particular, for any $\tilde{j}_i \in \{0, \dots, \alpha_i - 1\}$, with $\tilde{j}_i \neq j_i$, if $\tilde{J}$ is such that its i-th component is $\tilde{j}_i$, then, one has $|x_i^J - x_i^{\tilde{J}}| \geqslant 2\eta' > 2\eta$. Therefore, the events ξ_{i,j_i} and $\xi_{i,\tilde{j}_i}$ are disjoint. Moreover, observe that the environment $\nu_{J^{(-i)}}$ does not depend on the index j_i, as the change point i is pulled to zero.

Therefore, we can write, for a fixed i and for a fixed vector of indices $J^{(-i)}$,

$$\begin{aligned}
\frac{1}{\alpha_i} \sum_{j_i=0}^{\alpha_i - 1} \mathbb{P}_{\pi,J^{(-i)}}(\xi_{i,j_i}) &= \frac{1}{\alpha_i} \mathbb{E}_{\pi,J^{(-i)}} \left[\sum_{j_i=0}^{\alpha_i - 1} \mathbf{1}_{\xi_{i,j_i}} \right] = \frac{1}{\alpha_i} \mathbb{P}_{\pi,J^{(-i)}} \left[\bigsqcup_{j_i=0}^{\alpha_i - 1} \xi_{i,j_i} \right] \\
&\leqslant \frac{1}{\alpha_i} \mathbb{P}_{\pi,J^{(-i)}}(\mathcal{T}_\pi \leqslant \chi) \\
&\leqslant \frac{2\delta}{\alpha_i} \ ,
\end{aligned}$$

where we use that, $\forall j_i$, $\xi_{i,j_i} \subset \{\mathcal{T}_\pi \leqslant \chi\}$, and the last inequality follows from Equation (7.36).

On the other hand, it follows from Equation (7.35), averaging over j_i, that,

$$\frac{1}{\alpha_i} \sum_{j_i=0}^{\alpha_i - 1} \mathbb{P}_{\pi,J}(\xi_{i,j_i}) \geqslant 1 - 2\delta \ ,$$

where we point out that the distribution $\mathbb{P}_{\pi,J}$ on the left sum depends on the index j_i.

From these two bounds, using monotonicity of $\mathrm{kl}(\cdot,\cdot)$ in each argument (valid as soon as $2\delta/\alpha_i \leqslant 1 - 2\delta$, which holds for $\delta \leqslant 1/4$ and $\alpha_i \geqslant 1$), and joint convexity of the kl divergence

(e.g., Section 4.1 in (Polyanskiy and Wu, 2014)), we write

$$\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha_i}\right) \leqslant \mathrm{kl}\left(\frac{1}{\alpha_i}\sum_{j_i=0}^{\alpha_i-1} \mathbb{P}_{\pi,J}(\xi_{i,j_i}), \frac{1}{\alpha_i}\sum_{j_i=0}^{\alpha_i-1} \mathbb{P}_{\pi,J^{(-i)}}(\xi_{i,j_i})\right)$$
$$\leqslant \frac{1}{\alpha_i}\sum_{j_i=0}^{\alpha_i-1} \mathrm{kl}\left(\mathbb{P}_{\pi,J}(\xi_{i,j_i}), \mathbb{P}_{\pi,J^{(-i)}}(\xi_{i,j_i})\right) \quad . \tag{7.37}$$

Observe that the bound (7.37) holds for any components $J^{(-i)} = (j_k)_{k\neq i}$. In particular, we can average over $J^{(-i)}$ to write

$$\begin{aligned}\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha_i}\right) &\leqslant \frac{1}{\alpha_i}\sum_{j_i=0}^{\alpha_i-1} \frac{1}{\prod_{k\neq i}\alpha_k} \sum_{J^{(-i)}\in\prod_{k\neq i}\{0,\dots,\alpha_k-1\}} \mathrm{kl}\left(\mathbb{P}_{\pi,J}(\xi_{i,j_i}), \mathbb{P}_{\pi,J^{(-i)}}(\xi_{i,j_i})\right)\\ &= \frac{1}{\prod_{k=1}^m \alpha_k} \sum_{J\in\prod_{k=1}^m\{0,\dots,\alpha_k-1\}} \mathrm{kl}\left(\mathbb{P}_{\pi,J}(\xi_{i,j_i}), \mathbb{P}_{\pi,J^{(-i)}}(\xi_{i,j_i})\right)\\ &= \mathbb{E}_{J\sim\mathcal{U}}\left[\mathrm{kl}\left(\mathbb{P}_{\pi,J}(\xi_{i,j_i}), \mathbb{P}_{\pi,J^{(-i)}}(\xi_{i,j_i})\right)\right] \quad ,\end{aligned}$$

where the last equality is the notation for the expectation over J when J is uniformly distributed in $\prod_{k=1}^m\{0,\dots,\alpha_k-1\}$.

Now, ξ_{i,j_i} is measurable with respect to the observations of π up to time χ, so that the probability of ξ_{i,j_i} under $\mathbb{P}_{\pi,J}$ and $\mathbb{P}_{\pi,J^{(-i)}}$ only depends on the distribution of the observations up to time χ. Consider then the procedure $\tilde{\pi}$ defined as follows: for $t = 1,\dots,\chi$, $\tilde{\pi}$ behaves like π. If at time χ, the algorithm π did not stop, then $\tilde{\pi}$ stops anyway and returns an error. For any environment, the distribution of the observations of $\tilde{\pi}$ up to time χ is the same as the distribution of the observations of π up to time χ. Moreover, the algorithm $\tilde{\pi}$ stops at time χ at the latest, so that the budget of $\tilde{\pi}$ is at most χ. Therefore, we can rewrite the above inequality, and use the data-processing inequality for the KL divergence, to write

$$\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha_i}\right) \leqslant \mathbb{E}_{J\sim\mathcal{U}}\left[\mathrm{KL}\left(\mathbb{P}_{\tilde{\pi},J}, \mathbb{P}_{\tilde{\pi},J^{(-i)}}\right)\right] \quad . \tag{7.38}$$

Bound on the KL divergence.

Fix $i \in \{1,\dots,m\}$ and $J \in \prod_{k=1}^m\{0,\dots,\alpha_k-1\}$. We want to bound the KL divergence $\mathrm{KL}\left(\mathbb{P}_{\tilde{\pi},J}, \mathbb{P}_{\tilde{\pi},J^{(-i)}}\right)$.

Now, observe that the environments ν_J and $\nu_{J^{(-i)}}$ only differ by a constant magnitude Δ_i on the interval $[x_i^J, x_{i+1}^J)$, and are identical elsewhere. Denote as f_J for the reward function characterizing ν_J, and $f_{J^{(-i)}}$ for the reward function characterizing $\nu_{J^{(-i)}}$. One has

$$f_J - f_{J^{(-i)}} = \Delta_i \mathbb{I}\{x \in [x_i^J, x_{i+1}^J)\} \quad .$$

Recall that the noise is assumed to be Gaussian with variance 1.

Therefore, we can use the decomposition of the KL divergence for bandit problems, under a

continuous action space, to write

$$\begin{aligned}\mathrm{KL}\left(\mathbb{P}_{\tilde{\pi},J},\mathbb{P}_{\tilde{\pi},J^{(-i)}}\right) &= \mathbb{E}_{\tilde{\pi},J}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}}\mathrm{KL}\left(\mathcal{N}(f_J(x_t),1),\mathcal{N}(f_{J^{(-i)}}(x_t),1)\right)\right]\\ &= \mathbb{E}_{\tilde{\pi},J}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}}\frac{(f_J(x_t)-f_{J^{(-i)}}(x_t))^2}{2}\right]\\ &= \frac{\Delta_i^2}{2}\mathbb{E}_{\tilde{\pi},J}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}}\mathbb{I}\{x_t\in[x_i^J,x_{i+1}^J)\}\right] \ .\end{aligned}$$

Now, let average over J uniformly distributed in $\prod_{k=1}^m\{0,\dots,\alpha_k-1\}$, from Equation (7.38), we obtain

$$\frac{2}{\Delta_i^2}\,\mathrm{kl}\left(1-2\delta,\frac{2\delta}{\alpha_i}\right)\leqslant\mathbb{E}_{J\sim\mathcal{U}}\left[\mathbb{E}_{\tilde{\pi},J}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}}\mathbb{I}\{x_t\in[x_i^J,x_{i+1}^J)\}\right]\right] \ .$$

Observe that the averaging measure $\mathbb{E}_{J\sim\mathcal{U}}\mathbb{E}_{\tilde{\pi},J}$ is the same for each index i. Then, we sum over $i=1,\dots,m$

$$\begin{aligned}\sum_{i=1}^m\frac{2}{\Delta_i^2}\,\mathrm{kl}\left(1-2\delta,\frac{2\delta}{\alpha_i}\right) &\leqslant \sum_{i=1}^m\mathbb{E}_{J\sim\mathcal{U}}\left[\mathbb{E}_{\tilde{\pi},J}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}}\mathbb{I}\{x_t\in[x_i^J,x_{i+1}^J)\}\right]\right]\\ &= \mathbb{E}_{J\sim\mathcal{U}}\left[\mathbb{E}_{\tilde{\pi},J}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}}\sum_{i=1}^m\mathbb{I}\{x_t\in[x_i^J,x_{i+1}^J)\}\right]\right]\\ &\leqslant \mathbb{E}_{J\sim\mathcal{U}}\left[\mathbb{E}_{\tilde{\pi},J}\left[\mathcal{T}_{\tilde{\pi}}\right]\right]\\ &\leqslant \chi \ ,\end{aligned}$$

where the second inequality follows from the fact that the intervals $[x_i^J,x_{i+1}^J)$ are disjoint, and the last inequality follows from the fact that, under $\tilde{\pi}$, the budget $\mathcal{T}_{\tilde{\pi}}$ is at most χ.

By definition of χ as an infimum, this means that there exists $J\in\prod_{i=1}^m\{0,\dots,\alpha_i-1\}$ such that

$$\mathbb{P}_{\pi,J}\left(\mathcal{T}_\pi\geqslant 2\sum_{i=1}^m\Delta_i^{-2}\,\mathrm{kl}\left(1-2\delta,\frac{2\delta}{\alpha_i}\right)\right)\geqslant\delta \ ,$$

in particular, there exists an environment $\nu'=\nu_J$ such that the above holds. Moreover, by construction, the gaps of ν' are the same as the gaps of ν, and for each $i\in\{1,\dots,m-1\}$, $(1/4)\vartheta_i\leqslant\vartheta_i'\leqslant(3/4)\vartheta_i$, as verified in Lemma 7.C.3.

Finally, use the bound $\mathrm{kl}\left(1-2\delta,\frac{2\delta}{\alpha_i}\right)\geqslant\frac{1}{2}\log\left(\frac{\alpha_i}{8\delta}\right)$ from technical Lemma 7.C.5, and with the bound $\log(\alpha_i)\geqslant\log_+\left(\frac{s_i}{16\eta'}\right)$, which follows from the definition of α_i. As $\eta'>\eta$ is arbitrary, we can conclude by taking the limit $\eta'\to\eta$. □

Lemma 7.C.3. *For all $J \in \prod_{i=1}^{m}\{0,\dots,\alpha_i - 1\}$, it holds that ν_J is a valid environment with m change points, with the same jumps at ν, and with spacing $(1/4)\vartheta_i \leqslant \vartheta_i^J \leqslant (3/4)\vartheta_i$ for each $i = 1,\dots,m-1$. In particular, $H^{\nu_J}_{\text{localize}} = H^{\nu}_{\text{localize}}$, and $\frac{4}{3}H^{\nu}_{\text{detect}} \leqslant H^{\nu_J}_{\text{detect}} \leqslant 4H^{\nu}_{\text{detect}}$.*

Proof. First, we verify that the environment ν_J is valid and contains m change points. For validity, we only need the positions of the change points $(x_i^J)_{i=1}^m$ to be distinct and to belong to $(0,1)$.

Consider $x_m^J = \frac{1}{2}x_m^* + j_m \cdot 2\eta'$. One has $j_m \cdot 2\eta' \leqslant \alpha_m \cdot 2\eta' \leqslant \frac{s_m}{8\eta'} \cdot 2\eta' \leqslant 1/4$, and $x_m^* \leqslant 1$, so that $x_m^J < 1$.

Moreover, for each $i = 1,\dots,m-1$, the change points x_i^J and x_{i+1}^J are spaced by at least $\vartheta_i/4$ and at most $(3/4)\vartheta_i$. Indeed, one has $x_{i+1}^J - x_i^J = \frac{1}{2}(x_{i+1}^* - x_i^*) + (j_{i+1} - j_i)\cdot 2\eta' = \frac{1}{2}\vartheta_i + (j_{i+1} - j_i)\cdot 2\eta'$. We will show that $|j_{i+1} - j_i| \cdot 2\eta' \leqslant \frac{\vartheta_i}{4}$, which will conclude the proof.

By assumption on J, we have that $j_{i+1} - j_i \in \{-(\alpha_i - 1),\dots,\alpha_{i+1} - 1\}$, so that $|j_{i+1} - j_i| \cdot 2\eta' \leqslant ((\alpha_i - 1) \vee (\alpha_{i+1} - 1)) \cdot 2\eta'$. Now, by definition of α_i and α_{i+1}, we have $\alpha_i \vee \alpha_{i+1} - 1 = 1 \vee \left\lfloor \frac{\vartheta_{i-1}\wedge\vartheta_i}{8\eta'} \right\rfloor \vee \left\lfloor \frac{\vartheta_i \wedge \vartheta_{i+1}}{8\eta'} \right\rfloor - 1 \leqslant 0 \vee \left(\frac{\vartheta_i}{8\eta'} - 1\right)$. Therefore, $|j_{i+1} - j_i| \cdot 2\eta' \leqslant 0 \vee \left(\frac{\vartheta_i}{4} - 2\eta'\right) \leqslant \frac{\vartheta_i}{4}$, which conclude the proof.

□

Lemma 7.C.4.

$$\textit{For all } J^{(-i)}, \textit{ it holds that} \quad \mathbb{P}_{\pi, J^{(-i)}}(\mathcal{T}_\pi \leqslant \chi) \leqslant 2\delta \ .$$

Proof of Lemma 7.C.4. We can prove it by considering two environments, with one vanishing change point of jump magnitude ϵ. Denote as $\nu_\epsilon^{(1)}$ the environment obtained from $\nu_{J^{(-i)}}$ by adding a signal of magnitude ϵ on the interval $[x_m^* + l', 1)$, and as $\nu_\epsilon^{(2)}$ the environment obtained from $\nu_{J^{(-i)}}$ by adding a signal of magnitude ϵ on the interval $[1 - l', 1)$. Here, we take $0 < l' < 1/8 - \eta$, which is possible by assumption on $\eta < 1/8$. These choice imply that the m-th change point under $\nu_\epsilon^{(1)}$, and the m-th change point under $\nu_\epsilon^{(2)}$ are spaced by at least 2η. Indeed, $x_m^* + l' + \eta < 1 - l' - \eta$, which hold for $x_m^* \leqslant 3/4 \leqslant 1 - l' - 2\eta$. Note that $\nu_\epsilon^{(1)}$ and $\nu_\epsilon^{(2)}$ have m valid change points.

Consider the decomposition of the event $\{\mathcal{T}_\pi \leqslant \chi\}$ as

$$\{\mathcal{T}_\pi \leqslant \chi\} = \{\mathcal{T}_\pi \leqslant \chi,\ \hat{x}_m \leqslant \frac{1}{2}(x_m^* + 1)\} \cup \{\mathcal{T}_\pi \leqslant \chi,\ \hat{x}_m > \frac{1}{2}(x_m^* + 1)\} \ . \tag{7.39}$$

Under $\nu_\epsilon^{(1)}$, the change point m is at position $x_m^* + l' < \frac{1}{2}(x_m^* + 1) - \eta$. Then, by the (δ,η)-correctness of π, it holds that

$$\mathbb{P}_{\pi,\nu_\epsilon^{(1)}}\left(\hat{x}_m > \frac{1}{2}(x_m^* + 1)\right) \leqslant \mathbb{P}_{\pi,\nu_\epsilon^{(1)}}(|\hat{x}_m - (x_m^* + l')| > \eta) \leqslant \delta \ . \tag{7.40}$$

Similarly, under $\nu_\epsilon^{(2)}$, the change point m is at position $1 - l' > \frac{1}{2}(x_m^* + 1) + \eta$, so that

$$\mathbb{P}_{\pi,\nu_\epsilon^{(2)}}\left(\hat{x}_m \leqslant \frac{1}{2}(x_m^* + 1)\right) \leqslant \mathbb{P}_{\pi,\nu_\epsilon^{(2)}}(|\hat{x}_m - (1 - l')| > \eta) \leqslant \delta \ . \tag{7.41}$$

Moreover, when ϵ goes to zero, the total variation distance between $\mathbb{P}_{\pi,\nu_\epsilon^{(1)}}$ and $\mathbb{P}_{\pi,J^{(-i)}}$, goes to zero when we restrict to events measurable with respect to the observations up to the finite

time χ. Consider any event A measurable with respect to the observations up to time χ. Then, we can write, by decomposition of the KL divergence,

$$\begin{aligned}\mathrm{KL}(\mathbb{P}_{\pi,\nu_\epsilon^{(1)}}(A), \mathbb{P}_{\pi,J^{(-i)}}(A)) &\leqslant \mathbb{E}_{\pi,\nu_\epsilon^{(1)}}\left[\sum_{t=1}^{\chi} \mathrm{KL}(\mathcal{N}(f_\epsilon^{(1)}(x_t),1), \mathcal{N}(f_{J^{(-i)}}(x_t),1))\right] \\ &= \frac{\epsilon^2}{2}\mathbb{E}_{\pi,\nu_\epsilon^{(1)}}\left[\sum_{t=1}^{\chi} \mathbb{I}\{x_t \in [x_m^* + l', 1-l')\}\right] \\ &\leqslant \frac{\epsilon^2}{2}\chi \ ,\end{aligned}$$

Now, from the Pinsker's inequality, we have that

$$|\mathbb{P}_{\pi,\nu_\epsilon^{(1)}}(A) - \mathbb{P}_{\pi,J^{(-i)}}(A)| \leqslant \sqrt{\frac{1}{2}\mathrm{KL}(\mathbb{P}_{\pi,\nu_\epsilon^{(1)}}(A), \mathbb{P}_{\pi,J^{(-i)}}(A))} \leqslant \epsilon\sqrt{\chi}/2 \ ,$$

which goes to zero as ϵ goes to zero. We proceed similarly for $\nu_\epsilon^{(2)}$.

Now, using the decomposition of the event $\{\mathcal{T}_\pi \leqslant \chi\}$ given by Equation (7.39), we can write

$$\begin{aligned}\mathbb{P}_{\pi,J^{(-i)}}(\mathcal{T}_\pi \leqslant \chi) &= \mathbb{P}_{\pi,J^{(-i)}}(\mathcal{T}_\pi \leqslant \chi,\ \hat{x}_m \leqslant \tfrac{1}{2}(x_m^*+1)) + \mathbb{P}_{\pi,J^{(-i)}}(\mathcal{T}_\pi \leqslant \chi,\ \hat{x}_m > \tfrac{1}{2}(x_m^*+1)) \\ &= \lim_{\epsilon\to 0}\mathbb{P}_{\pi,\nu_\epsilon^{(2)}}(\mathcal{T}_\pi \leqslant \chi,\ \hat{x}_m \leqslant \tfrac{1}{2}(x_m^*+1)) \\ &\quad + \lim_{\epsilon\to 0}\mathbb{P}_{\pi,\nu_\epsilon^{(1)}}(\mathcal{T}_\pi \leqslant \chi,\ \hat{x}_m > \tfrac{1}{2}(x_m^*+1)) \\ &\leqslant \lim_{\epsilon\to 0}\mathbb{P}_{\pi,\nu_\epsilon^{(2)}}(\hat{x}_m \leqslant \tfrac{1}{2}(x_m^*+1)) + \lim_{\epsilon\to 0}\mathbb{P}_{\pi,\nu_\epsilon^{(1)}}(\hat{x}_m > \tfrac{1}{2}(x_m^*+1)) \\ &\leqslant 2\delta \ ,\end{aligned}$$

where the last inequality follows from the bounds (7.40),(7.41) we obtained under $\nu_\epsilon^{(1)}$ and $\nu_\epsilon^{(2)}$. This proves Equation (7.36). □

Lemma 7.C.5. *For $\delta \in (0, 1/4)$ and $\alpha \geqslant 1$, one has*

$$\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha}\right) \geqslant \frac{1}{2}\log\left(\frac{\alpha}{8\delta}\right) \ .$$

Proof of Lemma 7.C.5. We can write

$$\begin{aligned}\mathrm{kl}\left(1-2\delta, \frac{2\delta}{\alpha}\right) &\geqslant \mathrm{kl}\left(1/2, \frac{2\delta}{\alpha}\right) \\ &= \frac{1}{2}\log\left(\frac{1/2}{2\delta/\alpha}\right) + \frac{1}{2}\log\left(\frac{1/2}{1-2\delta/\alpha}\right) \\ &= \frac{1}{2}\log\left(\frac{\alpha}{8\delta}\right) + \frac{1}{2}\log\left(\frac{1}{(1-2\delta/\alpha)}\right) \\ &\geqslant \frac{1}{2}\log\left(\frac{\alpha}{8\delta}\right) \ ,\end{aligned}$$

where the first inequality follows from the fact that $1 - 2\delta \geqslant 1/2 \geqslant 2\delta/\alpha$, and the last inequality follows from the fact that $1 - 2\delta/\alpha \leqslant 1$. □

7.C.3 Proof of Lemma 7.C.2

Proof roadmap. The proof proceeds in four steps. First, we construct a family of environments $(\tilde{\nu}^{(s)})_{s\in[0,1/4]}$ from ν, and show that H_{detect} and H_{localize} are preserved up to absolute constants (Lemma 7.C.8). Second, we introduce an alternative environment $\bar{\nu}$ (with two removed gaps). Third, we consider the event $\xi = \{\mathcal{T}_\pi \leqslant \chi\}$, where χ is the maximum $(1-\delta)$-quantile of the budget under all the environments $\tilde{\nu}^{(s)}$. This event has a probability at least δ under $\nu' = \tilde{\nu}^{(s)}$ for all $s \in [0, 1/4]$. We show in Lemma 7.C.9 that it has a probability smaller than 2δ under $\bar{\nu}$, because $\bar{\nu}$ contains less than $m = N$ change points, so that the algorithm should not stop. Fourth, we convert this total-variation lower bound into a KL lower bound using the Bretagnolle-Huber inequality. Finally, we upper-bound the averaged KL by the expected number of pulls in the interval where $\bar{\nu}$ and $\tilde{\nu}^{(s)}$ differ, which yields a lower bound on χ; the conclusion follows by selecting $\nu' = \tilde{\nu}^{(s)}$ for a suitable s.

Proof of Lemma 7.C.2 . **Reference class of environments.**

Consider an environment ν with $m \geqslant 2$ change points, with parameters μ_0, $\Delta_1, \dots, \Delta_m$, $\vartheta_0, \dots, \vartheta_m$.

Observe that,

$$H^{\nu}_{\text{detect}} := H^{(m,\nu)}_{\text{detect}} = \max_{i=1,\dots,m} \frac{1}{\mathcal{E}_i^2} = \max_{i=1,\dots,m-1} \frac{1}{\vartheta_i} \frac{1}{\Delta_i^2 \wedge \Delta_{i+1}^2} \ .$$

The quantity $\frac{1}{\vartheta_i}\frac{1}{\Delta_i^2\wedge\Delta_{i+1}^2}$ measures the difficulty of detecting a point in the interval between x_i^* and x_{i+1}^*. Consider $i^* = \arg\max_{i=1,\dots,m-1} \frac{1}{\vartheta_i}\frac{1}{\Delta_i^2\wedge\Delta_{i+1}^2}$, and assume without loss of generality [8] that $|\Delta_{i^*}| \leqslant |\Delta_{i^*+1}|$, so that $H^{\nu}_{\text{detect}} = \frac{1}{\vartheta_{i^*}}\frac{1}{\Delta_{i^*}^2}$.

From ν, we construct a family of environments $\tilde{\nu}^{(s)}$ parametrized by some $s \in [0, 1/4]$. Define $\tilde{\Delta} = \sqrt{2}(|\Delta_{i^*}| \wedge |\Delta_{i^*+1}|) = \sqrt{2}|\Delta_{i^*}|$ and $\tilde{\vartheta} = \vartheta_{i^*}/2$.

Let $s \in [0, 1/4]$, define the environment $\tilde{\nu}^{(s)}$ with parameters $\tilde{\Delta}_1, \dots, \tilde{\Delta}_m$, the initial mean μ_0 and the length of the change points denoted as $\tilde{\vartheta}_0^{(s)}, \dots, \tilde{\vartheta}_m^{(s)}$, where

— the initial mean is μ_0 and initial position is $\tilde{x}_1^* = x_1^*/2$;

— for $i \in \{1, \dots, m\} \setminus \{i^*, i^*+1\}$, $\tilde{\Delta}_i = \Delta_i$;

— $\tilde{\Delta}_{i^*} = -\tilde{\Delta}_{i^*+1} = \tilde{\Delta}$;

— for $i \in \{1, \dots, m-1\} \setminus \{i^*-1, i^*, i^*+1\}$, $\tilde{\vartheta}_i^{(s)} = \vartheta_i/2$;

— $\tilde{\vartheta}_{i^*}^{(s)} = \vartheta_{i^*}/2$; $\tilde{\vartheta}_{i^*-1}^{(s)} = \vartheta_{i^*-1}/2 + s$ and $\tilde{\vartheta}_{i^*+1}^{(s)} = \vartheta_{i^*+1}/2 + \frac{1}{2} - s$

For notational purposes, denote as $\tilde{l} = \sum_{j=0}^{i^*-1} \tilde{\vartheta}_j^{(0)}$ for the position of the change point i^* under $\tilde{\nu}^{(0)}$. Observe that by definition, $\tilde{l} = \sum_{j=0}^{i^*-1} \vartheta_j/2 = x_{i^*}^*/2$. Recall that $\tilde{\vartheta} = \vartheta_{i^*}/2$. Define

8. The other case follows from a symmetric construction

also $\tilde{r} = \tilde{l} + \tilde{\vartheta} + 1/4$ for the position of the change point $i^* + 1$ under $\tilde{\nu}^{(1/4)}$. Observe that $\tilde{r} = \sum_{j=0}^{i^*} \tilde{\vartheta}^{(1/4)} = \sum_{j=0}^{i^*} \vartheta_j/2 + 1/4 = x^*_{i^*+1}/2 + 1/4$. Moreover, one has $|\tilde{r} - \tilde{l}| = \tilde{\vartheta} + 1/4$. We denote as $\mathbb{P}_{\pi,s}$ the probability distribution induced by the interaction between algorithm π and the environment $\tilde{\nu}^{(s)}$. We will also denote by $\mathbb{E}_{\pi,s}$ the corresponding expectation.

Remark 7.C.6. The family of environments $(\tilde{\nu}^{(s)})_s$ is designed to preserve, up to a multiplicative constant 2 the complexities H_{detect} and H_{localize}, that is $H^{\tilde{\nu}^{(s)}}_{\text{detect}}$ and $H^{\tilde{\nu}^{(s)}}_{\text{localize}}$ are of the same order as H^{ν}_{detect} and $H^{\nu}_{\text{localize}}$, respectively. This is proved in Lemma 7.C.8 and is a direct consequence of the construction of $\tilde{\nu}^{(s)}$ and the definition of H_{detect} and H_{localize}.

Alternative environment. Define $\bar{\nu}$ as the environment obtained from $\tilde{\nu}^{(0)}$ by pulling to zero the gaps $\tilde{\Delta}_{i^*}$ and $\tilde{\Delta}_{i^*+1}$, so that $\bar{\nu}$ has $m-2$ change points. Note that $\bar{\nu}$ does not depend on the parameter s. Interestingly, the environment $\tilde{\nu}^{(s)}$ and $\bar{\nu}$ only differ for arms in the interval $[\tilde{l}, \tilde{r}]$.

Denote as $\bar{\mathbb{P}}_\pi$ the probability distribution of the observations when the environment is $\bar{\nu}$ and by $\bar{\mathbb{E}}_\pi$ the associated expectation, with some algorithm π.

Remark 7.C.7. This construction is a reduction scheme. We reduce the problem of detecting m change points to the signal detection problem of detecting the presence of a signal of magnitude $\tilde{\Delta}$, on an interval of length $\tilde{\vartheta}$, planted somewhere in the interval $[\tilde{l}, \tilde{r}]$.

Bound in total variation. Let π be an algorithm, which is (δ, η)-correct for the m-change point detection problem, and let $\hat{x}_1, \dots, \hat{x}_m$ be the corresponding estimates of the change points. Denote as $\mathcal{T}_\pi$ the stopping time of π. Define χ as the supremum of the $(1-\delta)$-quantile of the budget over the family of probability distributions $\mathbb{P}_{\pi,s}$ for $s \in [0, 1/4]$. Namely,

$$\chi = \inf \left\{ t > 0 : \sup_{s \in [0,1/4]} \mathbb{P}_{\pi,s}(\mathcal{T}_\pi > t) \leqslant \delta \right\} \ .$$

Define the event

$$\xi = \{\mathcal{T}_\pi \leqslant \chi\} \ .$$

By definition of χ, it holds that $\mathbb{P}_{\pi,s}(\xi^c) \leqslant \delta$ for all $s \in [0, 1/4]$. Then, under $\bar{\nu}$, the algorithm is not able to identify m change points in finite time, as $\bar{\nu}$ only contains $m-2$ change points. In particular, one can prove the following bound, proved in Lemma 7.C.9 on the probability of the event ξ under $\bar{\nu}$:

$$\bar{\mathbb{P}}_\pi(\xi) \leqslant 2\delta \ . \tag{7.42}$$

This bound is a consequence of the fact that, under $\bar{\nu}$, the algorithm π should typically not stop, as it cannot identify m change points. In particular, the event ξ is an event on which π stops, and therefore should have a small probability under $\bar{\nu}$. The proof follows from proving that $\bar{\nu}$ is as close as possible to two different environments with m change points, under which the position of the change points are different.

Now, observe that the event ξ is measurable with respect to the first χ observations. Consider the modified algorithm $\tilde{\pi}$ defined as follows. For $t = 1, \dots, \chi$, $\tilde{\pi}$ uses the rules of π. If at time χ, the algorithm π did not stop, $\tilde{\pi}$ stops anyway and returns an error. Observe that ξ is exactly the

event on which $\tilde{\pi}$ does not return an error. In particular, the event ξ is measurable with respect to the observations of $\tilde{\pi}$, and has the same probability under π and $\tilde{\pi}$ (facing any environment). Therefore, $\bar{\mathbb{P}}_{\tilde{\pi}}(\xi) \leqslant 2\delta$ and $\mathbb{P}_{\tilde{\pi},s}(\xi) \geqslant 1-\delta$ for all $s \in [0, 1/4]$.

We can bound the total variation distance between $\bar{\mathbb{P}}_{\tilde{\pi}}$ and $\mathbb{P}_{\tilde{\pi},s}$ for $s \in [0, 1/4]$ as

$$\begin{aligned}\mathrm{TV}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s}) &\geqslant \bar{\mathbb{P}}_{\tilde{\pi}}(\xi^c) - \mathbb{P}_{\tilde{\pi},s}(\xi^c) \\ &\geqslant 1 - 2\delta - \delta = 1 - 3\delta \ .\end{aligned}$$

Data processing inequality. Classically, one can bound the total variation distance between $\bar{\mathbb{P}}_{\tilde{\pi}}$ and $\mathbb{P}_{\tilde{\pi},s}$ by the Kullback-Leibler divergence between these two distributions, using the Bretagnolle-Huber inequality as

$$\mathrm{TV}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s}) \leqslant 1 - \frac{1}{2}\exp\Big(-\mathrm{KL}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s})\Big) \ .$$

Rearranging this inequality,and using the previous bound on the total variation distance, we obtain, for any $s \in [0, 1/4]$,

$$\mathrm{KL}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s}) \geqslant \log\left(\frac{1}{2(1-\mathrm{TV}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s}))}\right) \geqslant \log\left(\frac{1}{6\delta}\right) \ . \tag{7.43}$$

Lower bound on the budget. Finally, we can bound the Kullback-Leibler divergence between $\bar{\mathbb{P}}_{\tilde{\pi}}$ and $\mathbb{P}_{\tilde{\pi},s}$, using the decomposition of the KL divergence for bandit problems, under a continuous action space. Observe that the environments $\bar{\nu}$ and $\tilde{\nu}^{(s)}$ only differ by a constant magnitude $\tilde{\Delta}$ on the interval $[\tilde{l}+s, \tilde{l}+s+\tilde{\vartheta})$, and are identical elsewhere. For notation, denote as $\bar{f}$ and $\tilde{f}^{(s)}$ the mean reward functions of $\bar{\nu}$ and $\tilde{\nu}^{(s)}$, respectively. Recall that the reward distributions are Gaussian with variance 1. Denote also as a_t the arm pulled at time t by $\tilde{\pi}$. Then, the reward distribution at time t under $\bar{\nu}$ is $\mathcal{N}(\bar{f}(a_t), 1)$ and the reward distribution at time t under $\tilde{\nu}^{(s)}$ is $\mathcal{N}(\tilde{f}^{(s)}(a_t), 1)$.

Therefore, we can write

$$\begin{aligned}\mathrm{KL}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s}) &= \bar{\mathbb{E}}_{\tilde{\pi}}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}} \mathrm{KL}\left(\mathcal{N}(\bar{f}(a_t), 1), \mathcal{N}(\tilde{f}^{(s)}(a_t), 1)\right)\right] \\ &= \bar{\mathbb{E}}_{\tilde{\pi}}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}} \frac{\left(\tilde{f}^{(s)}(a_t) - \bar{f}(a_t)\right)^2}{2}\right] \\ &= \frac{\tilde{\Delta}^2}{2}\bar{\mathbb{E}}_{\tilde{\pi}}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}} \mathbb{I}\{a_t \in [\tilde{l}+s, \tilde{l}+s+\tilde{\vartheta})\}\right] \ .\end{aligned}$$

Now, let average over s uniformly distributed in $[0, 1/4]$, we obtain

$$\begin{aligned}\frac{1}{1/4}\int_0^{1/4} \mathrm{KL}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s})ds &= \frac{1}{1/4}\frac{\tilde{\Delta}^2}{2}\bar{\mathbb{E}}_{\tilde{\pi}}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}}\int_0^{1/4} \mathbb{I}\{a_t \in [\tilde{l}+s, \tilde{l}+s+\tilde{\vartheta})\}ds\right] \\ &= 2\tilde{\Delta}^2\bar{\mathbb{E}}_{\tilde{\pi}}\left[\sum_{t=1}^{\mathcal{T}_{\tilde{\pi}}}\int_0^{1/4} \mathbb{I}\{s \in [a_t-\tilde{l}-\tilde{\vartheta}, a_t-\tilde{l})\}ds\right] \\ &\leqslant 2\tilde{\vartheta}\tilde{\Delta}^2\bar{\mathbb{E}}_{\tilde{\pi}}\left[\mathcal{T}_{\tilde{\pi}}\right] \\ &,\end{aligned}$$

where the last inequality follows from the fact that the length of the interval $[a_t - \tilde{l} - \tilde{\vartheta}, a_t - \tilde{l})$ is at most $\tilde{\vartheta}$. Now, using the fact that, under $\tilde{\pi}$, the budget $\mathcal{T}_{\tilde{\pi}}$ is at most χ, we can write

$$\log\left(\frac{1}{6\delta}\right) \leqslant \frac{1}{1/4}\int_0^{1/4} \mathrm{KL}(\bar{\mathbb{P}}_{\tilde{\pi}}, \mathbb{P}_{\tilde{\pi},s})ds \leqslant 2\tilde{\vartheta}\tilde{\Delta}^2\chi \ , \tag{7.44}$$

where we used the bound on the KL divergence from Equation (7.43) in the first inequality.

Rearranging yields

$$\chi \geqslant \frac{1}{2\tilde{\vartheta}\tilde{\Delta}^2}\log\left(\frac{1}{6\delta}\right) = \frac{1}{2\vartheta_{i^*}}\frac{1}{\Delta_{i^*}^2 \wedge \Delta_{i^*+1}^2}\log\left(\frac{1}{6\delta}\right) = \frac{1}{2}H^{\nu}_{\mathrm{detect}}\log\left(\frac{1}{6\delta}\right) \ . \tag{7.45}$$

By definition of χ, this means that there exists $s \in [0, 1/4]$ such that

$$\mathbb{P}_{\pi,s}\left(\mathcal{T}_{\pi} \geqslant \frac{1}{2}H^{\nu}_{\mathrm{detect}}\log\left(\frac{1}{6\delta}\right)\right) \geqslant \delta \ .$$

The proof of Lemma 7.C.2 is concluded by observing that, by Lemma 7.C.8, the environment $\tilde{\nu}^{(s)}$ has the desired properties. It remains to prove Lemma 7.C.9 and Lemma 7.C.8. □

Lemma 7.C.8. *For all $s \in [0, 1/4]$ it holds that $\tilde{\nu}^{(s)}$ is a valid environment with m change points spaced by at least 2η and that*

$$H^{\nu}_{\mathrm{detect}} \leqslant H^{\tilde{\nu}^{(s)}}_{\mathrm{detect}} \leqslant 2H^{\nu}_{\mathrm{detect}} \quad \textit{and} \quad \frac{1}{2}H^{\nu}_{\mathrm{localize}} \leqslant H^{\tilde{\nu}^{(s)}}_{\mathrm{localize}} \leqslant H^{\nu}_{\mathrm{localize}} \ .$$

Proof of the Lemma 7.C.8. It is clear that $\tilde{\nu}^{(s)}$ is a valid environment with m change points, and that the change points are spaced by at least 2η, because $\tilde{\vartheta}_i^{(s)} \geqslant \vartheta_i/2 \geqslant 2\eta$ for all $i = 1, \ldots, m-1$.

By definition of $\tilde{\nu}^{(s)}$, it holds that $\tilde{\Delta}_{i^*} = -\tilde{\Delta}_{i^*+1} = \sqrt{2}\Delta_{i^*}$ and $\tilde{\vartheta}_{i^*}^{(s)} = \vartheta_{i^*}/2$. Then, one has

$$H^{\tilde{\nu}^{(s)}}_{\mathrm{localize}} = \sum_{i=1}^{m}\frac{1}{\tilde{\Delta}_i^2} = \sum_{\substack{i=1 \\ i\neq i^*, i^*+1}}^{m}\frac{1}{\Delta_i^2} + \frac{2}{\tilde{\Delta}^2} = \sum_{\substack{i=1 \\ i\neq i^*, i^*+1}}^{m}\frac{1}{\Delta_i^2} + \frac{1}{\Delta_{i^*}^2} \ .$$

Moreover, it holds that $\frac{1}{2}\left(\frac{1}{\Delta_{i^*}^2} + \frac{1}{\Delta_{i^*+1}^2}\right) \leqslant \frac{1}{\Delta_{i^*}^2} \leqslant \frac{1}{\Delta_{i^*}^2} + \frac{1}{\Delta_{i^*+1}^2}$, because $|\Delta_{i^*}| \leqslant |\Delta_{i^*+1}|$. Then,

one has $\frac{1}{2}H^{\nu}_{\text{localize}} \leqslant H^{\tilde{\nu}^{(s)}}_{\text{localize}} \leqslant H^{\nu}_{\text{localize}}$.

Recall that we assumed that $\Delta_{i^*} \leqslant \Delta_{i^*+1}$, so that

$$H^{\nu}_{\text{detect}} = \max_{i=1,\dots,m-1} \frac{1}{\vartheta_i} \frac{1}{\Delta_i^2 \wedge \Delta_{i+1}^2} = \frac{1}{\vartheta_{i^*}} \frac{1}{\Delta_{i^*}^2}$$
$$H^{\tilde{\nu}^{(s)}}_{\text{detect}} = \max_{i=1,\dots,m-1} \frac{1}{\tilde{\vartheta}_i^{(s)}} \frac{1}{\tilde{\Delta}_i^2 \wedge \tilde{\Delta}_{i+1}^2} \ .$$

We consider the different cases for $i = 1, \dots, m-1$ to control $\frac{1}{\tilde{\vartheta}_i^{(s)}} \frac{1}{\tilde{\Delta}_i^2 \wedge \tilde{\Delta}_{i+1}^2}$. Assume that $i \neq i^*-1, i^*, i^*+1$, so that $\tilde{\vartheta}_i^{(s)} = \vartheta_i/2$ and $\tilde{\Delta}_i = \Delta_i$ and $\tilde{\Delta}_{i+1} = \Delta_{i+1}$. Then, one has

$$\frac{1}{\tilde{\vartheta}_i^{(s)}} \frac{1}{\tilde{\Delta}_i^2 \wedge \tilde{\Delta}_{i+1}^2} = \frac{2}{\vartheta_i} \frac{1}{\Delta_i^2 \wedge \Delta_{i+1}^2} \leqslant 2H^{\nu}_{\text{detect}} \ .$$

If $i = i^*-1$, one has $\tilde{\vartheta}^{(s)}_{i^*-1} = \vartheta_{i^*-1}/2 + s$, $\tilde{\Delta}_{i^*-1} = \Delta_{i^*-1}$ and $\tilde{\Delta}_{i^*} = \sqrt{2}\Delta_{i^*}$. Then, one has

$$\frac{1}{\tilde{\vartheta}^{(s)}_{i^*-1}} \frac{1}{\tilde{\Delta}^2_{i^*-1} \wedge \tilde{\Delta}^2_{i^*}} = \frac{1}{(\vartheta_{i^*-1}/2 + s)} \frac{1}{\Delta^2_{i^*-1} \wedge (\sqrt{2}\Delta_{i^*})^2} \leqslant \frac{2}{\vartheta_{i^*-1}} \frac{1}{\Delta^2_{i^*-1} \wedge \Delta^2_{i^*}} \leqslant 2H^{\nu}_{\text{detect}} \ ,$$

as $s \geqslant 0$ and $\sqrt{2}|\Delta_{i^*}| \geqslant |\Delta_{i^*}|$.
If $i = i^*$, one has $\tilde{\vartheta}^{(s)}_{i^*} = \vartheta_{i^*}/2$, $\tilde{\Delta}_{i^*} = \sqrt{2}\Delta_{i^*}$ and $\tilde{\Delta}_{i^*+1} = -\sqrt{2}\Delta_{i^*}$. Then,

$$\frac{1}{\tilde{\vartheta}^{(s)}_{i^*}} \frac{1}{\tilde{\Delta}^2_{i^*} \wedge \tilde{\Delta}^2_{i^*+1}} = \frac{1}{(\vartheta_{i^*}/2)} \frac{1}{(\sqrt{2}\Delta_{i^*})^2} = \frac{1}{\vartheta_{i^*}} \frac{1}{\Delta^2_{i^*}} = H^{\nu}_{\text{detect}} \ .$$

Finally, if $i = i^*+1$, one has $\tilde{\vartheta}^{(s)}_{i^*+1} = \vartheta_{i^*+1}/2 + 1/2 - s \geqslant \frac{1}{4}$ because $s \leqslant 1/4$. Also, $\tilde{\Delta}_{i^*+1} = -\sqrt{2}\Delta_{i^*}$ and $\tilde{\Delta}_{i^*+2} = \Delta_{i^*+2}$. Then,

$$\begin{aligned}\frac{1}{\tilde{\vartheta}^{(s)}_{i^*+1}} \frac{1}{\tilde{\Delta}^2_{i^*+1} \wedge \tilde{\Delta}^2_{i^*+2}} &= \frac{1}{(\vartheta_{i^*+1}/2 + 1/2 - s)} \frac{1}{(-\sqrt{2}\Delta_{i^*})^2 \wedge \Delta^2_{i^*+2}} \\ &\leqslant \frac{4}{\Delta^2_{i^*} \wedge \Delta^2_{i^*+2}} \leqslant 4 \max_{i=1}^{m} \frac{1}{\Delta_i^2} \leqslant 4H^{\nu}_{\text{detect}} \ ,\end{aligned}$$

where one uses that $\max_{i=1}^{m} \frac{1}{\Delta_i^2} \leqslant H^{\nu}_{\text{detect}}$ because $\vartheta_i \leqslant 1$ for all $i = 1, \dots, m-1$.

The inequality $H^{\nu}_{\text{detect}} \leqslant H^{\tilde{\nu}^{(s)}}_{\text{detect}} \leqslant 2H^{\nu}_{\text{detect}}$ follows from the previous inequalities, observing that the case $i = i^*$ gives the lower bound and the other cases give the upper bound. □

Lemma 7.C.9. *It holds that*

$$\bar{\mathbb{P}}_{\pi}(\xi) \leqslant 2\delta \ .$$

Proof of Lemma 7.C.9. Let $e = (1/8 - \eta) > 0$, which is positive because $\eta \leqslant 1/8$. By definition of $\tilde{l}$ and $\tilde{r}$, one has $\tilde{r} - \tilde{l} = \tilde{\vartheta} + 1/4 > 2\eta + 2e$. In particular, the points $\tilde{l} + e$ and $\tilde{r} - e$ are spaced by at least 2η, so that $\tilde{l} + e + \eta < \frac{1}{2}(\tilde{l} + \tilde{r}) < \tilde{r} - e - \eta$.

Let $\epsilon > 0$ and consider $\bar{\nu}_\epsilon^{(1)}$ as the environment obtained from $\bar{\nu}$ by adding a plateau of height ϵ, and length e on the interval $[\tilde{l}, \tilde{l}+e]$. That is, we consider a change point of jump magnitude $+\epsilon$ at $\tilde{l}$, and another one of jump $-\epsilon$ at $\tilde{l}+e$. Consider also $\bar{\nu}_\epsilon^{(2)}$ as the environment obtained from $\bar{\nu}$ by adding a plateau of height ϵ, and length e on the interval $[\tilde{r}-e, \tilde{r}]$. Note that $\bar{\nu}_\epsilon^{(1)}$ and $\bar{\nu}_\epsilon^{(2)}$ have m valid change points. Moreover, when ϵ goes to zero, the total variation distance between $\bar{\nu}_\epsilon^{(1)}$ and $\bar{\nu}$, and between $\bar{\nu}_\epsilon^{(2)}$ and $\bar{\nu}$ goes to zero [9]. Denote as $\mathbb{P}_\epsilon^{(1)}$ and $\mathbb{P}_\epsilon^{(2)}$ the probability distributions of the observations when the environment is $\bar{\nu}_\epsilon^{(1)}$ and $\bar{\nu}_\epsilon^{(2)}$, respectively, and when the algorithm is π.

Consider the decomposition of the event ξ as

$$\xi = \left\{ \mathcal{T}_\pi \leqslant \chi,\ \frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} \leqslant \frac{\tilde{l}+\tilde{r}}{2} \right\} \sqcup \left\{ \mathcal{T}_\pi \leqslant \chi,\ \frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} > \frac{\tilde{l}+\tilde{r}}{2} \right\}$$

Under $\bar{\nu}_\epsilon^{(1)}$, the change point i^*+1 is at position $\tilde{l}+e$. Moreover, it holds that $\tilde{l}+e+\eta < \frac{\tilde{l}+\tilde{r}}{2}$. Then, by the (δ, η, m)-correctness of π, it holds that

$$\mathbb{P}_\epsilon^{(1)} \left(\frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} > \frac{\tilde{l}+\tilde{r}}{2} \right) \leqslant \mathbb{P}_\epsilon^{(1)} \left(\hat{x}_{i^*+1} > (\tilde{l}+e) + \eta \right) \leqslant \delta \ . \tag{7.46}$$

Similarly, under $\bar{\nu}_\epsilon^{(2)}$, the change point i^* is at position $\tilde{r}-e$. Moreover, it holds that $\tilde{r}-e-\eta \geqslant \frac{\tilde{l}+\tilde{r}}{2}$. Then, by the (δ, η)-correctness of π, it holds that

$$\mathbb{P}_\epsilon^{(2)} \left(\frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} \leqslant \frac{\tilde{l}+\tilde{r}}{2} \right) \leqslant \mathbb{P}_\epsilon^{(2)} \left(\hat{x}_{i^*} < (\tilde{r}-e) - \eta \right) \leqslant \delta \ . \tag{7.47}$$

Overall, we have that, taking the limit $\epsilon \to 0$,

$$\begin{aligned}
\bar{\mathbb{P}}_\pi(\xi) &= \lim_{\epsilon \to 0} \left\{ \mathbb{P}_\epsilon^{(1)} \left(\mathcal{T}_\pi \leqslant \chi, \frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} > \frac{\tilde{l}+\tilde{r}}{2} \right) + \mathbb{P}_\epsilon^{(2)} \left(\mathcal{T}_\pi \leqslant \chi, \frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} \leqslant \frac{\tilde{l}+\tilde{r}}{2} \right) \right\} \\
&\leqslant \lim_{\epsilon \to 0} \mathbb{P}_\epsilon^{(1)} \left(\frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} > \frac{\tilde{l}+\tilde{r}}{2} \right) + \lim_{\epsilon \to 0} \mathbb{P}_\epsilon^{(2)} \left(\frac{\hat{x}_{i^*} + \hat{x}_{i^*+1}}{2} \leqslant \frac{\tilde{l}+\tilde{r}}{2} \right) \\
&\leqslant 2\delta \ ,
\end{aligned}$$

which concludes the proof of Lemma 7.C.9. □

7.D Relating the Continuous and the Discrete Bandit Change Point Problem

Assume we have an algorithm that is able to localize N change points in a K armed bandit change point problem. This means that if we have K arms with corresponding 1-subGaussian

9. Using the same argument as in Lemma 7.C.4

distribution $\tilde{\nu}_k$ and mean $\tilde{\mu}_k$ for $k \in [K]$ and there is $\mathcal{C} = \{1 \leq k_1^* < k_2^* < \cdots < k_m^* < K\}$ such that $\tilde{\mu}_k \neq \tilde{\mu}_{k+1}$ for $k \in \mathcal{C}$, the algorithm is able to identify a subset of $\mathcal{C}$ with cardinality m.

When we are now dealing with a continuous bandit change point problem as introduced in Section 7.2, we can still tackle it by using such an algorithm.

If we want to localize change points in $[0, 1]$ with precision η, let us first build a discrete bandit, by considering $\tilde{\nu}_k = \nu_{(k-1)\eta}$ for $k = 1, \ldots, \lfloor 1/\eta \rfloor + 1$. If we now localize a change point k^*, it means that $\tilde{\mu}_{k^*} \neq \tilde{\mu}_{k^*+1}$ and therefore $\mu_{(k^*-1)\eta} \neq \mu_{k^*\eta}$. This implies, that in the continuous setting there must be a change point x^* with $(k^* - 1)\eta < x^* \leq k^*\eta$, and using $c = (k^* - 1)\eta$ guarantees that $|x^* - c| \leq \eta$.

While we use this discretization throughout our experiments, we want to remark that conversely it is possible to transform a discrete change point identification problem for K arms to a continuous problem by constructing a respective step function. Here, localizing change points with precision $\eta < 1/2K$ leads to exact change point identification in the discrete case.

BIBLIOGRAPHY

Emmanuel Abbe. Community detection and stochastic block models: recent developments. Journal of Machine Learning Research, 18(177):1–86, 2018.

Nir Ailon, Zohar Karnin, and Thorsten Joachims. Reducing dueling bandits to cardinal bandits. In International Conference on Machine Learning, pages 856–864. PMLR, 2014.

Daniel Aloise, Amit Deshpande, Pierre Hansen, and Preyas Popat. NP-hardness of euclidean sum-of-squares clustering. Machine learning, 75(2):245–248, 2009.

Samaneh Aminikhanghahi and Diane J Cook. A survey of methods for time series change point detection. Knowledge and information systems, 51(2):339–367, 2017.

Kaito Ariu, Narae Ryu, Se-Young Yun, and Alexandre Proutière. Regret in online recommendation systems. Advances in Neural Information Processing Systems, 33:21141–21150, 2020.

Kaito Ariu, Kenshi Abe, and Alexandre Proutière. Thresholded lasso bandit. In International Conference on Machine Learning, pages 878–928. PMLR, 2022.

Kaito Ariu, Jungseul Ok, Alexandre Proutiere, and Seyoung Yun. Optimal clustering from noisy binary feedback. Machine Learning, 113(5):2733–2764, 2024.

Sylvain Arlot, Alain Celisse, and Zaid Harchaoui. A kernel multiple change-point algorithm via model selection. Journal of machine learning research, 20(162):1–56, 2019.

David Arthur and Sergei Vassilvitskii. K-means++ the advantages of careful seeding. In Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, pages 1027–1035, 2007.

Jean-Yves Audibert and Sébastien Bubeck. Best arm identification in multi-armed bandits. In COLT-23th Conference on learning theory-2010, pages 13–p, 2010.

Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. Machine learning, 47(2):235–256, 2002.

Martin Azizyan, Aarti Singh, and Larry Wasserman. Minimax theory for high-dimensional Gaussian mixtures with sparse mean separation. Advances in Neural Information Processing Systems, 26, 2013.

Tavor Baharav and David Tse. Ultra fast medoid identification via correlated sequential halving. Advances in Neural Information Processing Systems, 32, 2019.

Amparo Baíllo, Jose R Berrendero, and Martín Sánchez-Signorini. Kernel k-means clustering of distributional data. arXiv preprint arXiv:2509.18037, 2025.

Viktor Bengs, Róbert Busa-Fekete, Adil El Mesaoudi-Paul, and Eyke Hüllermeier. Preference-based online learning with dueling bandits: A survey. Journal of Machine Learning Research, 22(7):1–108, 2021.

Alain Berlinet and Christine Thomas-Agnan. Reproducing kernel Hilbert spaces in probability and statistics. Springer Science & Business Media, 2011.

Donald A Berry, Robert W Chen, Alan Zame, David C Heath, and Larry A Shepp. Bandit problems with infinitely many arms. The Annals of Statistics, 25(5):2103–2116, 1997.

Michael W Berry, Azlinah Mohamed, and Bee Wah Yap. Supervised and unsupervised learning for data science, volume 10. Springer, 2020.

Lilian Besson, Emilie Kaufmann, Odalric-Ambrym Maillard, and Julien Seznec. Efficient change-point detection for tackling piecewise-stationary bandits. Journal of Machine Learning Research, 23(77):1–40, 2022.

Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. Biometrika, 39(3/4):324–345, 1952.

Sébastien Bubeck, Rémi Munos, and Gilles Stoltz. Pure exploration in multi-armed bandits problems. In International conference on Algorithmic learning theory, pages 23–37. Springer, 2009.

Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. Foundations and Trends® in Machine Learning, 5(1):1–122, 2012.

Séebastian Bubeck, Tengyao Wang, and Nitin Viswanathan. Multiple identifications in multi-armed bandits. In International Conference on Machine Learning, pages 258–265. PMLR, 2013.

V Buldygin and K Moskvichova. The sub-gaussian norm of a binary random variable. Theory of probability and mathematical statistics, 86:33–49, 2013.

T Tony Cai, Zongming Ma, and Yihong Wu. Sparse PCA: optimal rates and adaptive estimation. The Annals of Statistics, 41(6):3074–3110, 2013.

Yang Cao, Zheng Wen, Branislav Kveton, and Yao Xie. Nearly optimal adaptive procedure with change detection for piecewise-stationary bandit. In Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics, volume 89 of Proceedings of Machine Learning Research, pages 418–427. PMLR, 16–18 Apr 2019. URL `https://proceedings.mlr.press/v89/cao19a.html`.

ALEXANDRA CARPENTIER. Testing the regularity of a smooth signal. Bernoulli, pages 465–488, 2015.

Alexandra Carpentier and Andrea Locatelli. Tight (lower) bounds for the fixed budget best arm identification bandit problem. In Conference on Learning Theory, pages 590–604. PMLR, 2016.

Alexandra Carpentier and Michal Valko. Simple regret for infinitely many armed bandits. In International Conference on Machine Learning, pages 1133–1141. PMLR, 2015.

RM Castro. Adaptive sensing performance lower bounds for sparse signal detection and support estimation. Bernoulli, 20(4):2217–2246, 2014.

M Emre Celebi, Hassan A Kingravi, and Patricio A Vela. A comparative study of efficient initialization methods for the k-means clustering algorithm. Expert systems with applications, 40(1): 200–210, 2013.

G Dhinesh Chandran, Kota Srinivas Reddy, and Srikrishna Bhashyam. Online clustering with bandit information. In 2025 IEEE International Symposium on Information Theory (ISIT), pages 1–6. IEEE, 2025.

Arghya Roy Chaudhuri and Shivaram Kalyanakrishnan. Pac identification of a bandit arm relative to a reward quantile. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 31, 2017.

Arghya Roy Chaudhuri and Shivaram Kalyanakrishnan. Pac identification of many good arms in stochastic multi-armed bandits. In International Conference on Machine Learning, pages 991–1000. PMLR, 2019.

Bangrui Chen and Peter I Frazier. Dueling bandits with weak regret. In International Conference on Machine Learning, pages 731–739. PMLR, 2017.

Lijie Chen and Jian Li. On the optimal sample complexity for best arm identification. arXiv preprint arXiv:1511.03774, 2015.

Shouyuan Chen, Tian Lin, Irwin King, Michael R Lyu, and Wei Chen. Combinatorial pure exploration of multi-armed bandits. Advances in neural information processing systems, 27, 2014.

Wei Chen, Yihan Du, Longbo Huang, and Haoyu Zhao. Combinatorial pure exploration for dueling bandit. In International Conference on Machine Learning, pages 1531–1541. PMLR, 2020.

Xiaohui Chen and Yun Yang. Hanson–Wright inequality in Hilbert spaces with application to K-means clustering for non-Euclidean data. Bernoulli, 27(1):586 – 614, 2021. doi: 10.3150/20-BEJ1251. URL `https://doi.org/10.3150/20-BEJ1251`.

James Cheshire, Pierre Ménard, and Alexandra Carpentier. The influence of shape constraints on the thresholding bandit problem. In Conference on Learning Theory, pages 1228–1275. PMLR, 2020.

Shein-Chung Chow and Mark Chang. Adaptive design methods in clinical trials–a review. Orphanet journal of rare diseases, 3(1):11, 2008.

Vincent Cohen-Addad, Benjamin Guedj, Varun Kanade, and Guy Rom. Online k-means clustering. In International Conference on Artificial Intelligence and Statistics, pages 1126–1134. PMLR, 2021.

Olivier Collier and Arnak S Dalalyan. Multidimensional linear functional estimation in sparse gaussian models and robust estimation of the mean. 2019.

Sanjoy Dasgupta. Learning mixtures of gaussians. In 40th annual symposium on foundations of computer science (Cat. No. 99CB37039), pages 634–644. IEEE, 1999.

Rianne De Heide, James Cheshire, Pierre Ménard, and Alexandra Carpentier. Bandits with many optimal arms. Advances in Neural Information Processing Systems, 34:22457–22469, 2021.

Rémy Degenne and Wouter M Koolen. Pure exploration with multiple correct answers. Advances in Neural Information Processing Systems, 32, 2019.

Rémy Degenne, Wouter M Koolen, and Pierre Ménard. Non-asymptotic pure exploration by solving games. Advances in Neural Information Processing Systems, 32, 2019.

Inderjit S Dhillon, Yuqiang Guan, and Brian Kulis. Kernel k-means: spectral clustering and normalized cuts. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 551–556, 2004.

Ilias Diakonikolas, Daniel M. Kane, and Alistair Stewart. Statistical query lower bounds for robust estimation of high-dimensional gaussians and gaussian mixtures. In 2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS), pages 73–84, 2017. doi: 10.1109/FOCS.2017.16. URL https://ieeexplore.ieee.org/abstract/document/8104048.

Ilias Diakonikolas, Daniel M. Kane, and Alistair Stewart. List-decodable robust mean estimation and learning mixtures of spherical gaussians. In Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, page 1047–1060. Association for Computing Machinery, 2018. ISBN 9781450355599. doi: 10.1145/3188745.3188758. URL https://doi.org/10.1145/3188745.3188758.

Ilias Diakonikolas, Daniel M. Kane, Thanasis Pittas, and Nikos Zarifis. Sq lower bounds for learning mixtures of separated and bounded covariance gaussians. In Gergely Neu and Lorenzo Rosasco, editors, Proceedings of Thirty Sixth Conference on Learning Theory, volume 195 of Proceedings of Machine Learning Research, pages 2319–2349. PMLR, 12–15 Jul 2023. URL https://proceedings.mlr.press/v195/diakonikolas23b.html.

Bertrand Even, Christophe Giraud, and Nicolas Verzelen. Computation-information gap in high-dimensional clustering. In The Thirty Seventh Annual Conference on Learning Theory, pages 1646–1712. PMLR, 2024.

Eyal Even-Dar, Shie Mannor, Yishay Mansour, and Sridhar Mahadevan. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. Journal of machine learning research, 7(6), 2006.

Moein Falahatgar, Alon Orlitsky, Venkatadheeraj Pichapati, and Ananda Theertha Suresh. Maximum selection and ranking under noisy comparisons. In International Conference on Machine Learning, pages 1088–1096. PMLR, 2017.

Moein Falahatgar, Ayush Jain, Alon Orlitsky, Venkatadheeraj Pichapati, and Vaishakh Ravindrakumar. The limits of maxing, ranking, and preference learning. In International conference on machine learning, pages 1427–1436. PMLR, 2018.

Yingjie Fei and Yudong Chen. Hidden integrality of SDP relaxations for sub-Gaussian mixture models. In Conference On Learning Theory, pages 1931–1965. PMLR, 2018.

Piotr Fryzlewicz. Wild binary segmentation for multiple change-point detection. The Annals of Statistics, 42(6):2243–2281, 2014.

Aurélien Garivier and Emilie Kaufmann. Optimal best arm identification with fixed confidence. In Conference on Learning Theory, pages 998–1027. PMLR, 2016.

Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for switching bandit problems. In International conference on algorithmic learning theory, pages 174–188. Springer, 2011.

Claudio Gentile, Shuai Li, and Giovanni Zappella. Online clustering of bandits. In Eric P. Xing and Tony Jebara, editors, Proceedings of the 31st International Conference on Machine Learning, volume 32 of Proceedings of Machine Learning Research, pages 757–765, Bejing, China, 22–24 Jun 2014. PMLR. URL https://proceedings.mlr.press/v32/gentile14.html.

Sébastien Gerchinovitz, Pierre Ménard, and Gilles Stoltz. Fano's inequality for random variables. Statistical Science, 35(2):178–201, 2020.

Christophe Giraud. Introduction to high-dimensional statistics. Chapman and Hall/CRC, 2021.

Christophe Giraud and Nicolas Verzelen. Partial recovery bounds for clustering with the relaxed k-means. Mathematical Statistics and Learning, 1(3):317–374, 2019.

Ryan Gomes, Peter Welinder, Andreas Krause, and Pietro Perona. Crowdclustering. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K.Q. Weinberger, editors, Advances in Neural Information Processing Systems, volume 24. Curran Associates, Inc., 2011. URL https://proceedings.neurips.cc/paper_files/paper/2011/file/c86a7ee3d8ef0b551ed58e354a836f2b-Paper.pdf.

Maximilian Graf and Victor Thuot. The sample complexity of multiple change point identification under bandit feedback, 2026. URL https://arxiv.org/abs/2605.13252. (Author's preprint, inspire Chapter 7).

Maximilian Graf, Thuot Victor, and Nicolas Verzelen. Clustering items through bandit feedback: Finding the right feature out of many. In International Conference on Machine Learning, pages 20296–20325. PMLR, 2025. (Author's publication, inspire Chapter 5).

Arthur Gretton, Karsten M Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alexander J Smola. A kernel approach to comparing distributions. In Proceedings of the national conference on artificial intelligence, volume 22, page 1637. AAAI Press, 2007a.

Arthur Gretton, Kenji Fukumizu, Choon Teo, Le Song, Bernhard Schölkopf, and Alex Smola. A kernel statistical test of independence. In Advances in neural information processing systems, volume 20, 2007b.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. The journal of machine learning research, 13(1):723–773, 2012.

Björn Haddenhorst, Viktor Bengs, Jasmin Brandt, and Eyke Hüllermeier. Testification of condorcet winners in dueling bandits. In Uncertainty in Artificial Intelligence, pages 1195–1205. PMLR, 2021a.

Björn Haddenhorst, Viktor Bengs, and Eyke Hüllermeier. Identification of the generalized condorcet winner in multi-dueling bandits. Advances in Neural Information Processing Systems, 34:25904–25916, 2021b.

Peter Hall and Ilya Molchanov. Sequential methods for design-adaptive estimation of discontinuities in regression curves and surfaces. The Annals of Statistics, 31(3):921–941, 2003.

Shogo Hayashi, Yoshinobu Kawahara, and Hisashi Kashima. Active change-point detection. In Asian Conference on Machine Learning, pages 1017–1032. PMLR, 2019.

Reinhard Heckel, Nihar B Shah, Kannan Ramchandran, and Martin J Wainwright. Active ranking from pairwise comparisons and when parametric assumptions do not help. The Annals of Statistics, 47(6):3099–3126, 2019.

David V Hinkley. Inference about the change-point in a sequence of random variables. Biometrika, pages 1–17, 1970.

Chien-Ju Ho, Shahin Jabbari, and Jennifer Wortman Vaughan. Adaptive task assignment for crowdsourced classification. In International conference on machine learning, pages 534–542. PMLR, 2013.

Katja Hofmann, Lihong Li, and Filip Radlinski. Online evaluation for information retrieval. Found. Trends Inf. Retr., 10:1–117, 2016. URL https://api.semanticscholar.org/CorpusID:34529647.

YC Hung and G Michailidis. Modeling, analysis and simulation of switched processing systems. ACM Transactions on Modeling, Analysis and Simulation, 2007.

Shinji Ito, Haipeng Luo, Taira Tsuchiya, and Yue Wu. Instance-dependent regret bounds for learning two-player zero-sum games with bandit feedback. arXiv preprint arXiv:2502.17625, 2025.

Anil K Jain, M Narasimha Murty, and Patrick J Flynn. Data clustering: a review. ACM computing surveys (CSUR), 31(3):264–323, 1999.

Kevin Jamieson and Robert Nowak. Best-arm identification algorithms for multi-armed bandits in the fixed confidence setting. In 2014 48th annual conference on information sciences and systems (CISS), pages 1–6. IEEE, 2014.

Kevin Jamieson, Matthew Malloy, Robert Nowak, and Sébastien Bubeck. lil'ucb: An optimal exploration algorithm for multi-armed bandits. In Conference on Learning Theory, pages 423–439. PMLR, 2014.

Kevin Jamieson, Sumeet Katariya, Atul Deshpande, and Robert Nowak. Sparse dueling bandits. In Artificial Intelligence and Statistics, pages 416–424. PMLR, 2015.

Kevin G Jamieson and Robert Nowak. Active ranking using pairwise comparisons. Advances in neural information processing systems, 24, 2011.

Kevin G Jamieson, Daniel Haas, and Benjamin Recht. The power of adaptivity in identifying statistical alternatives. Advances in Neural Information Processing Systems, 29, 2016.

Di Jin, Zhizhi Yu, Pengfei Jiao, Shirui Pan, Dongxiao He, Jia Wu, Philip S Yu, and Weixiong Zhang. A survey of community detection approaches: From statistical modeling to deep learning. IEEE Transactions on knowledge and data engineering, 35(2):1149–1170, 2021.

Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, Filip Radlinski, and Geri Gay. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. ACM Transactions on Information Systems (TOIS), 25(2):7–es, 2007.

Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. Pac subset selection in stochastic multi-armed bandits. In ICML, volume 12, pages 655–662, 2012.

Zohar Karnin, Tomer Koren, and Oren Somekh. Almost optimal exploration in multi-armed bandits. In International conference on machine learning, pages 1238–1246. PMLR, 2013a.

Zohar Karnin, Tomer Koren, and Oren Somekh. Almost optimal exploration in multi-armed bandits. In Sanjoy Dasgupta and David McAllester, editors, Proceedings of the 30th International Conference on Machine Learning, Proceedings of Machine Learning Research, pages 1238–1246, Atlanta, Georgia, USA, 17–19 Jun 2013b. PMLR. URL `https://proceedings.mlr.press/v28/karnin13.html`.

Zohar S Karnin. Verification based solution for structured mab problems. Advances in Neural Information Processing Systems, 29, 2016.

Nikolai Karpov and Qin Zhang. Batched coarse ranking in multi-armed bandits. In Conference on Neural Information Processing Systems (NeurIPS), 2020.

Sumeet Katariya, Lalit Jain, Nandana Sengupta, James Evans, and Robert Nowak. Adaptive sampling for coarse ranking. In International Conference on Artificial Intelligence and Statistics, pages 1839–1848. PMLR, 2018.

Julian Katz-Samuels and Kevin Jamieson. The true sample complexity of identifying good arms. In International Conference on Artificial Intelligence and Statistics, pages 1781–1791. PMLR, 2020a.

Julian Katz-Samuels and Kevin Jamieson. The true sample complexity of identifying good arms. In Silvia Chiappa and Roberto Calandra, editors, Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, volume 108 of Proceedings of Machine Learning Research, pages 1781–1791. PMLR, 26–28 Aug 2020b.

Leonard Kaufman and Peter J Rousseeuw. Finding groups in data: an introduction to cluster analysis. John Wiley & Sons, 2009.

Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On the complexity of a/b testing. In Conference on Learning Theory, pages 461–481. PMLR, 2014.

Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On the complexity of best-arm identification in multi-armed bandit models. The Journal of Machine Learning Research, 17(1):1–42, 2016.

Junpei Komiyama, Junya Honda, Hisashi Kashima, and Hiroshi Nakagawa. Regret lower bound and optimal algorithm in dueling bandit problem. In Conference on learning theory, pages 1141–1154. PMLR, 2015.

Junpei Komiyama, Junya Honda, and Hiroshi Nakagawa. Copeland dueling bandit problem: Regret lower bound, optimal algorithm, and computationally efficient algorithm. In International Conference on Machine Learning, pages 1235–1244. PMLR, 2016.

Pravesh K. Kothari and Jacob Steinhardt. Better Agnostic Clustering Via Relaxed Tensor Norms. CoRR, abs/1711.07465, 2017. URL `http://arxiv.org/abs/1711.07465`.

Solt Kovács, Peter Bühlmann, Housen Li, and Axel Munk. Seeded binary segmentation: a general methodology for fast and optimal changepoint detection. Biometrika, 110(1):249–256, 2023.

Jeongyeol Kwon and Constantine Caramanis. The em algorithm gives sample-optimality for learning mixtures of well-separated gaussians. In Jacob Abernethy and Shivani Agarwal, editors, Proceedings of Thirty Third Conference on Learning Theory, volume 125 of Proceedings

of Machine Learning Research, pages 2425–2487. PMLR, 09–12 Jul 2020. URL https://proceedings.mlr.press/v125/kwon20a.html.

Tze Leung Lai. Sequential analysis: some classical problems and new challenges. Statistica Sinica, pages 303–351, 2001.

Yan Lan, Moulinath Banerjee, and George Michailidis. Change-point estimation under adaptive sampling. The Annals of Statistics, 37(4):1752–1791, 2009.

Tor Lattimore and Csaba Szepesvári. Bandit algorithms. Cambridge University Press, 2020.

Beatrice Laurent and Pascal Massart. Adaptive estimation of a quadratic functional by model selection. Annals of Statistics, pages 1302–1338, 2000.

Joseph Lazzaro and Ciara Pike-Burke. Fixed-budget change point identification in piecewise constant bandits. In The 28th International Conference on Artificial Intelligence and Statistics, 2025a. URL https://openreview.net/forum?id=g8DVAMyn1j.

Joseph Lazzaro and Ciara Pike-Burke. Fixed-confidence multiple change point identification under bandit feedback. In Proceedings of the 42nd International Conference on Machine Learning, volume 267 of Proceedings of Machine Learning Research, pages 32675–32703. PMLR, 13–19 Jul 2025b. URL https://proceedings.mlr.press/v267/lazzaro25a.html.

Thibault Lesieur, Caterina De Bacco, Jess Banks, Florent Krzakala, Cris Moore, and Lenka Zdeborová. Phase transitions and optimal algorithms in high-dimensional Gaussian mixture clustering. In 2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton), pages 601–608. IEEE, 2016.

Chang Li, Ilya Markov, Maarten De Rijke, and Masrour Zoghi. Mergedts: A method for effective large-scale online ranker evaluation. ACM Transactions on Information Systems (TOIS), 38(4): 1–28, 2020.

Shuai Li, Wei Chen, Shuai Li, and Kwong-Sak Leung. Improved algorithm on online clustering of bandits. In Proceedings of the 28th International Joint Conference on Artificial Intelligence, IJCAI'19, page 2923–2929. AAAI Press, 2019. ISBN 9780999241141.

Zhuohua Li, Maoli Liu, Xiangxiang Dai, and John Lui. Demystifying online clustering of bandits: Enhanced exploration under stochastic and smoothed adversarial contexts. In ICLR 2025. PMLR, 2025.

Allen Liu and Jerry Li. Clustering mixtures with almost optimal separation in polynomial time. In Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2022, page 1248–1261, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450392648. doi: 10.1145/3519935.3520012. URL https://doi.org/10.1145/3519935.3520012.

Haoyang Liu, Chao Gao, and Richard J Samworth. Minimax rates in sparse, high-dimensional change point detection. The Annals of Statistics, 49(2):1081–1112, 2021.

Xutong Liu, Haoru Zhao, Tong Yu, Shuai Li, and John C.S. Lui. Federated online clustering of bandits. In James Cussens and Kun Zhang, editors, Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence, volume 180 of Proceedings of Machine Learning Research, pages 1221–1231. PMLR, 01–05 Aug 2022. URL https://proceedings.mlr.press/v180/liu22a.html.

Stuart Lloyd. Least squares quantization in PCM. IEEE transactions on information theory, 28 (2):129–137, 1982.

Andrea Locatelli, Maurilio Gutzeit, and Alexandra Carpentier. An optimal algorithm for the thresholding bandit problem. In International Conference on Machine Learning, pages 1690–1698. PMLR, 2016.

Jie Lu, Dianshuang Wu, Mingsong Mao, Wei Wang, and Guangquan Zhang. Recommender system application developments: a survey. Decision support systems, 74:12–32, 2015.

Yu Lu and Harrison H. Zhou. Statistical and Computational Guarantees of Lloyd's Algorithm and its Variants. ArXiv e-prints, December 2016. URL https://arxiv.org/abs/1612.02099.

Yu Lu and Harrison H Zhou. Statistical and computational guarantees of lloyd's algorithm and its variants. arXiv preprint arXiv:1612.02099, 2016.

R Duncan Luce et al. Individual choice behavior, volume 4. Wiley New York, 1959.

Arnab Maiti. Open problem: Optimal instance-dependent sample complexity for finding nash equilibrium in two player zero-sum matrix games. Proceedings of Machine Learning Research vol, 291:1–5, 2025.

Arnab Maiti, Ross Boczar, Kevin Jamieson, and Lillian Ratliff. Near-optimal pure exploration in matrix games: A generalization of stochastic bandits & dueling bandits. In International Conference on Artificial Intelligence and Statistics, pages 2602–2610. PMLR, 2024.

Arnab Maiti, Ross Boczar, Kevin Jamieson, and Lillian Ratliff. Query-efficient algorithm to find all nash equilibria in a two-player zero-sum matrix game. ACM Transactions on Economics and Computation, 13(3):1–18, 2025.

James B McQueen. Some methods of classification and analysis of multivariate observations. In Proc. of 5th Berkeley Symposium on Math. Stat. and Prob., pages 281–297, 1967.

Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, and Bernhard Schölkopf. Kernel mean embedding of distributions: A review and beyond. Foundations and Trends® in Machine Learning, 10(1-2):1–141, 2017.

Subhojyoti Mukherjee. Safety aware changepoint detection for piecewise i.i.d. bandits. In The 38th Conference on Uncertainty in Artificial Intelligence, 2022. URL `https://openreview.net/forum?id=rgZGGvLi5eq`.

Mohamed Ndaoud. Sharp optimal recovery in the two component gaussian mixture model. The Annals of Statistics, 50(4):2096–2126, 2022.

Andrew Ng, Michael Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. Advances in neural information processing systems, 14, 2001.

Pavel Nikolaev, Daylond Hooper, Frederick Webber, Rahul Rao, Kevin Decker, Michael Krein, Jason Poleski, Rick Barto, and Benji Maruyama. Autonomy in materials research: a case study in carbon nanotube growth. npj Computational Materials, 2(1):1–6, 2016.

Yue S Niu, Ning Hao, and Heping Zhang. Multiple change-point detection: a selective overview. Statistical Science, pages 611–623, 2016.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12:2825–2830, 2011.

Erol Peköz, Sheldon M Ross, and Zhengyu Zhang. Dueling bandit problems. Probability in the Engineering and Informational Sciences, 36(2):264–275, 2022.

Jiming Peng and Yu Wei. Approximating k-means-type clustering via semidefinite programming. SIAM journal on optimization, 18(1):186–205, 2007.

Emmanuel Pilliat, Alexandra Carpentier, and Nicolas Verzelen. Optimal multiple change-point detection for high-dimensional data. Electronic Journal of Statistics, 17(1):1240–1315, 2023.

Emmanuel Pilliat, Alexandra Carpentier, and Nicolas Verzelen. Optimal rates for ranking a permuted isotonic matrix in polynomial time. In Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pages 3236–3273. SIAM, 2024.

Robin L Plackett. The analysis of permutations. Journal of the Royal Statistical Society Series C: Applied Statistics, 24(2):193–202, 1975.

Yury Polyanskiy and Yihong Wu. Lecture notes on information theory. Lecture Notes for ECE563 (UIUC) and, 6(2012-2016):7, 2014.

Vikas C Raykar, Shipeng Yu, Linda H Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. Learning from crowds. Journal of machine learning research, 11(4), 2010.

Oded Regev and Aravindan Vijayaraghavan. On learning mixtures of well-separated gaussians. In Proceedings of 58th Annual IEEE Symposium on the Foundations of Computer Science, 2017.

Wenbo Ren, Jia Liu, and Ness Shroff. The sample complexity of best-k items selection from pairwise comparisons. In International Conference on Machine Learning, pages 8051–8072. PMLR, 2020.

Philippe Rigollet and Jan-Christian Hütter. High-dimensional statistics. arXiv preprint arXiv:2310.19244, 2023.

Herbert Robbins. Some aspects of the sequential design of experiments. Bulletin of the American Mathematical Society, 58(5):527–535, 1952.

Elad Romanov, Tamir Bendory, and Or Ordentlich. On the role of channel capacity in learning gaussian mixture models. Proceedings of Machine Learning Research vol 178:1–50, 2022. URL `https://proceedings.mlr.press/v178/romanov22a.html`.

Mark Rudelson and Roman Vershynin. Hanson-Wright inequality and sub-gaussian concentration. Electron. Commun. Probab, 18(82):1–9, 2013.

El Mehdi Saad, Nicolas Verzelen, and Alexandra Carpentier. Active ranking of experts based on their performances in many tasks. In International Conference on Machine Learning, pages 29490–29513. PMLR, 2023.

El Mehdi Saad, Victor Thuot, and Nicolas Verzelen. The sampling complexity of condorcet winner identification in dueling bandits. arXiv preprint arXiv:2603.15189, 2026. (Author's preprint, inspire Chapter 6).

Aadirupa Saha and Pierre Gaillard. Versatile dueling bandits: Best-of-both world analyses for learning from relative preferences. In International Conference on Machine Learning, pages 19011–19026. PMLR, 2022.

Aadirupa Saha and Shubham Gupta. Optimal and efficient dynamic regret algorithms for non-stationary dueling bandits. In International Conference on Machine Learning, pages 19027–19049. PMLR, 2022.

A.J. Scott and Martin Knott. A cluster analysis method for grouping means in the analysis of variance. Biometrics, pages 507–512, 1974.

Nerea Sebastián, MR De La Fuente, DO López, MA Pérez-Jubindo, J Salud, S Diez-Berart, and MB Ros. Dielectric and thermodynamic study on the liquid crystal dimer α-(4-cyanobiphenyl-4'-oxy)-ω-(1-pyreniminebenzylidene-4'-oxy) undecane (cbo11o $\cdot$ py). The Journal of Physical Chemistry B, 115(32):9766–9775, 2011.

Nimrod Segol and Boaz Nadler. Improved convergence guarantees for learning Gaussian mixture models by EM and gradient EM. Electronic journal of statistics, 15(2):4510–4544, 2021.

Nihar B Shah and Martin J Wainwright. Simple, robust and optimal ranking from pairwise comparisons. Journal of machine learning research, 18(199):1–38, 2018.

Max Simchowitz, Kevin Jamieson, and Benjamin Recht. The simulator: Understanding adaptive sampling in the moderate-confidence regime. In Conference on Learning Theory, pages 1794–1834. PMLR, 2017.

Aleksandrs Slivkins. Introduction to multi-armed bandits. Foundations and Trends® in Machine Learning, 12(1-2):1–286, 2019.

Alex Smola, Arthur Gretton, Le Song, and Bernhard Schölkopf. A hilbert space embedding for distributions. In International conference on algorithmic learning theory, pages 13–31. Springer, 2007.

Bharath K Sriperumbudur, Arthur Gretton, Kenji Fukumizu, Bernhard Schölkopf, and Gert RG Lanckriet. Hilbert space embeddings and metrics on probability measures. The Journal of Machine Learning Research, 11:1517–1561, 2010.

Bharath K. Sriperumbudur, Kenji Fukumizu, and Gert R. G. Lanckriet. Universality, characteristic kernels and rkhs embedding of measures. J. Mach. Learn. Res., 12(null):2389–2410, July 2011. ISSN 1532-4435.

Ambuj Tewari and Susan A Murphy. From ads to interventions: Contextual bandits in mobile health. In Mobile health: sensors, analytic methods, and applications, pages 495–517. Springer, 2017.

William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. Biometrika, 25(3/4):285–294, 1933.

Victor Thuot, Alexandra Carpentier, Christophe Giraud, and Nicolas Verzelen. Clustering with bandit feedback: breaking down the computation/information gap. In Gautam Kamath and Po-Ling Loh, editors, Proceedings of The 36th International Conference on Algorithmic Learning Theory, volume 272 of Proceedings of Machine Learning Research, pages 1221–1284. PMLR, 24–27 Feb 2025. URL `https://proceedings.mlr.press/v272/thuot25a.html`. (Author's publication, inspire Chapter 3).

Victor Thuot, Sebastian Vogt, Debarghya Ghoshdastidar, and Nicolas Verzelen. Nonparametric kernel clustering with bandit feedback. arXiv preprint arXiv:2601.07535, 2026. (Author's preprint, inspire Chapter 4).

Louis L Thurstone. A law of comparative judgment. In Scaling, pages 81–92. Routledge, 2017.

Andrea Tirinzoni and Rémy Degenne. On elimination strategies for bandit fixed-confidence identification. Advances in Neural Information Processing Systems, 35:18586–18598, 2022.

Ilya Tolstikhin, Bharath K Sriperumbudur, and Krikamol Muandet. Minimax estimation of kernel mean embeddings. Journal of Machine Learning Research, 18(86):1–47, 2017.

Ilya O Tolstikhin, Bharath K Sriperumbudur, and Bernhard Schölkopf. Minimax estimation of maximum mean discrepancy with radial kernels. Advances in Neural Information Processing Systems, 29, 2016.

Charles Truong, Laurent Oudre, and Nicolas Vayatis. Selective review of offline change point detection methods. Signal processing, 167:107299, 2020.

Tanguy Urvoy, Fabrice Clerot, Raphael Féraud, and Sami Naamane. Generic exploration and k-armed voting bandits. In International Conference on Machine Learning, pages 91–99. PMLR, 2013.

Leena C Vankadara and Debarghya Ghoshdastidar. On the optimality of kernels for high-dimensional clustering. In International conference on artificial intelligence and statistics, pages 2185–2195. PMLR, 2020.

Leena C Vankadara, Sebastian Bordt, Ulrike von Luxburg, and Debarghya Ghoshdastidar. Recovery guarantees for kernel-based clustering under non-parametric mixture models. In International Conference on Artificial Intelligence and Statistics, pages 3817–3825. PMLR, 2021.

Venugopal V Veeravalli, Georgios Fellouris, and George V Moustakides. Quickest change detection with controlled sensing. IEEE Journal on Selected Areas in Information Theory, 5:1–11, 2024.

Santosh Vempala and Grant Wang. A spectral algorithm for learning mixtures of distributions. In The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. Proceedings., pages 113–122. IEEE, 2002.

Santosh Vempala and Grant Wang. A spectral algorithm for learning mixture models. Journal of Computer and System Sciences, 68(4):841–860, 2004.

Nicolas Verzelen, Magalie Fromont, Matthieu Lerasle, and Patricia Reynaud-Bouret. Optimal change-point detection and localization. The Annals of Statistics, 51(4):1586–1610, 2023.

Daren Wang, Yi Yu, and Alessandro Rinaldo. Univariate mean change point detection: Penalization, cusum and optimality. Electronic Journal of Statistics, 14:1917–1961, 2020.

Po-An Wang, Ruo-Chun Tzeng, and Alexandre Proutiere. Best arm identification with fixed budget: A large deviation perspective. Advances in Neural Information Processing Systems, 36: 16804–16815, 2023.

Tengyao Wang and Richard J Samworth. High dimensional change point estimation via sparse projection. Journal of the Royal Statistical Society Series B: Statistical Methodology, 80(1): 57–83, 2018.

Joe H Ward Jr. Hierarchical grouping to optimize an objective function. Journal of the American statistical association, 58(301):236–244, 1963.

Geoffrey Wolfer and Pierre Alquier. Variance-aware estimation of kernel mean embedding. Journal of Machine Learning Research, 26(57):1–48, 2025.

Qunzhi Xu and Yajun Mei. Asymptotic optimality theory for active quickest detection with unknown postchange parameters. Sequential analysis, 42(2):150–181, 2023.

Junwen Yang, Zixin Zhong, and Vincent YF Tan. Optimal clustering with bandit feedback. Journal of Machine Learning Research, 25(186):1–54, 2024.

Recep Can Yavas, Yuqi Huang, Vincent YF Tan, and Jonathan Scarlett. A general framework for clustering and distribution matching with bandit feedback. IEEE Transactions on Information Theory, 71(3):2116–2139, 2025.

Yi Yu. A review on minimax rates in change point detection and localisation. arXiv preprint arXiv:2011.01857, 2020.

Yisong Yue and Thorsten Joachims. Interactively optimizing information retrieval systems as a dueling bandits problem. In Proceedings of the 26th Annual International Conference on Machine Learning, pages 1201–1208, 2009.

Yisong Yue, Josef Broder, Robert Kleinberg, and Thorsten Joachims. The k-armed dueling bandits problem. Journal of Computer and System Sciences, 78(5):1538–1556, 2012.

Se-Young Yun and Alexandre Proutière. Optimal sampling and clustering in the stochastic block model. Advances in Neural Information Processing Systems, 32, 2019.

Wanrong Zhang and Yajun Mei. Bandit change-point detection for real-time monitoring high-dimensional data under sampling control. Technometrics, 65(1):33–43, 2023.

Yao Zhao, Connor Stephens, Csaba Szepesvári, and Kwang-Sung Jun. Revisiting simple regret: Fast rates for returning a good arm. In International Conference on Machine Learning, pages 42110–42158. PMLR, 2023.

Xujuan Zhou, Yue Xu, Yuefeng Li, Audun Josang, and Clive Cox. The state-of-the-art in personalized recommender systems for social networking. Artificial Intelligence Review, 37(2):119–132, 2012.

Masrour Zoghi, Shimon A Whiteson, Maarten De Rijke, and Remi Munos. Relative confidence sampling for efficient on-line ranker evaluation. In Proceedings of the 7th ACM international conference on Web search and data mining, pages 73–82, 2014.

Masrour Zoghi, Zohar S Karnin, Shimon Whiteson, and Maarten De Rijke. Copeland dueling bandits. Advances in neural information processing systems, 28, 2015a.

Masrour Zoghi, Shimon Whiteson, and Maarten de Rijke. Mergerucb: A method for large-scale online ranker evaluation. In Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, pages 17–26, 2015b.

Titre : Apprentissage actif non supervisé

Mot clés : bandits, exploration pure, apprentissage non supervisé, clustering, vainqueur de Condorcet, détection de ruptures

Résumé : Cette thèse étudie l'apprentissage actif non supervisé dans des modèles de bandit. Elle examine plusieurs problèmes d'exploration pure dans lesquels l'objectif est de retrouver une structure cachée dans un environnement de bandit, avec un niveau de confiance prescrit. Le fil conducteur est la quantification du coût d'exploration nécessaire pour identifier des observations informatives, ainsi que du coût requis pour reconstruire la structure inconnue tout en en certifiant la correction.

Nous analysons cinq problèmes : le clustering actif paramétrique, le clustering non paramétrique, le clustering avec sélection adaptative de variables, l'identification du vainqueur de Condorcet en *dueling bandits*, et la localisation de multiples points de rupture. Pour chacun, nous proposons des algorithmes efficaces avec garanties en espérance et en quantiles, ainsi que des bornes inférieures sur le budget d'échantillonnage.

L'accent est mis sur des régimes de grande dimension et de grande échelle. Nos résultats mettent en évidence des phénomènes invisibles dans les analyses asymptotiques : un compromis fondamental exploration-certification, des écarts structurels entre bornes en espérance et en quantiles, et des gains significatifs dus à l'adaptation, à la fois en budget d'échantillonnage et en complexité computationnelle.

Title: Unsupervised Active Learning

Keywords: bandits, pure exploration, unsupervised learning, clustering, Condorcet winner, change point detection

Abstract: This thesis studies unsupervised active learning under bandit feedback. It investigates several pure-exploration problems in which the goal is to recover an underlying structure in a bandit environment with a prescribed confidence level. The common thread is the quantification of the exploration cost required to identify informative observations, together with the cost required to recover the unknown structure while certifying correctness.

We investigate five problems: parametric clustering with bandit feedback, non-parametric clustering via kernel embeddings, clustering with adaptive feature selection, Condorcet winner identification in dueling bandits, and multiple change-point localization. For each setting, we design computationally efficient algorithms with expected and high-probability guarantees, and derive matching information-theoretic lower bounds.

The focus is on high-dimensional and large-scale regimes. Our results reveal phenomena that are hidden in asymptotic analyses: a fundamental exploration-certification trade-off, structural gaps between expectation and quantile guarantees, and significant gains from adaptivity in both sampling efficiency and computational complexity.