INRAE Universität Potsdam Université de Rennes

RESEARCH INTERNSHIP

MASTER'S DEGREE IN MATHEMATICS

Active clustering

Author:
Victor THUOT
Université de Rennes & ENS Rennes

Supervisors:
Alexandra CARPENTIER
Universität Potsdam
Nicolas Verzelen
INRAE Montpellier

Contents

Abstract

We investigate the recent problem of active clustering with bandit feedback. Unlike the classical and extensively studied problem of clustering, active clustering allows the learner to actively construct the sampling strategy. The learner get a sequence of noisy observations from a set of items that can be clustered. The learner has the ability to choose the order and the number of observations, and aim at recovering the hidden partition of the items with the minimal sampling cost. In this framework, the learner interacts with a stochastic multi-armed bandit with multi-dimensional feedback. The arms are divided into groups such that arms within the same group share the same distribution of feedback. . We tackle the problem in the fixed-confidence setting. The learner has to recover exactly the true partition of the arms with a fixed probability of error while minimizing the number of samples. The objective is to design computationally efficient algorithm with an optimal sampling budget. As a first step to achieve this objective, we focus on the Gaussian model and we propose an algorithm that we call ACB (Active Clustering with Bandit). We analyze and comment the number of samples (the budget) required to achieve the desired clustering accuracy. We highlight the benefits of using active sampling for the problem of clustering and we question the optimality of our budget.

Keywords
active clustering, active learning, clustering, stochastic bandit, pure exploration, hidden partition model, Gaussian model

1 Introduction

1.1 What is clustering ?

The task of clustering data is of major importance, as it has become one of the most fundamental problems in machine learning and has been extensively studied. The problem amounts to partitioning a data set into 'homogeneous' subgroups, where data within the same subgroup are similar while data in different subgroups are dissimilar. In the clustering problem, we make the assumption that the data can be partitioned into several homogeneous sub-populations. The goal of clustering algorithms is to uncover this unknown partition. We can highlight two main objective of clustering:

1. *Description of the data:* clustering can be used to describe, summarize, and understand the structure of a data set as in anomaly detection or image segmentation.

2. *Pre-processing of the data:* for statisticians, it is often easier to analyse homogeneous data, such as those derived from the same statistical model. However, this assumption is not reasonable with real-world data, particularly in the era of 'Big Data' where the collection of the data could involve multiple sources. A simple way to consider such heterogeneous data is to make a weaker assumption and assume a clustering structure of the data. By using a clustering algorithm, each subgroup can then be studied independently.

Some applications of clustering.
Clustering methods have been applied to various fields. In market analysis, clustering can be used to identify subgroups of consumers with common needs, with the goal of increasing potential profits or improving consumer satisfaction [CCGR97],[Gre77]. In social science, clustering can help describe interactions within a social network [BBA75]. In [HK00], clustering algorithms are compared for the diagnosis of heart disease. In [LHJ$^+$11], clustering methods are used to study the behavior of animals based on their movements.

A new framework: active clustering.

In the era of large-scale data, data collection often occurs sequentially and in some applications, the user can choose the order in which data are collected. In this report, we introduce the new framework of active clustering, which is specifically designed to handle such data. In this framework, a learner wants to cluster a set of items and for that, he can obtain several noisy observations of each item. The main characteristic of the problem is that the learner has the ability to actively choose the order of observations, which means that sequentially, the learner takes into account past observations and chooses the next item to observe. The learner also decides the number of observations that he needs and at some point he decides to stop sampling items and returns a partition of the items. The learner has two opposite objectives, he has to return a partition that is close to the true partition of the

items and he also has to use as few samples as possible. This situation can arise for instance in medical trials, where data collection is costly in terms of time and resources. The main objective is to study the benefits of using active learning to solve a clustering problem. In particular, the final objective behind this report is to find an optimal sampling strategy that would recover the clustering structure of the items with the minimal number of samples.

1.2 Organization

In Section 2.1, the new problem of active clustering is introduced and commented. This problem belongs to active learning and we use the formalism of bandit theory. The setting and notation considered in this report are also mostly introduced in this subsection. In Subsection 2.3, a short review on bandit theory highlights that active clustering is a pure exploration problem. In Subsection 2.3 some related papers on active clustering are presented.

In Section 3, we present the state-of-the-art on clustering and explain the differences with active clustering. The problem of clustering, can be seen as a special case of active clustering where we can only observe each item once. In this long section of bibliography, we will discuss in Subsection 3.1 some classical algorithms and methods used for clustering such as hierarchical clustering, the K-means paradigm, and convex relaxations of K-means. In Subsection 3.2, we introduce the Gaussian model that will also be considered in Section 4. The performance of clustering algorithms in the hidden partition Gaussian model are compared on the Subsection 3.2 focusing on conditions of separation. This will be used as a benchmark in order to study the benefits of using active sampling to cluster items in the Gaussian model.

In Section 4.1, we introduce a new and simple algorithm for active clustering with bandit feedback, which is an original contribution in this report. We explain the construction of this algorithm and analyse its performance in terms of the number of samples used to reach a fixed probability of error. A variant of this algorithm based on the doubling-trick is also introduced in this Section 4.2.

The reader can finally find in the Appendix a summary of notation A.1, some concentration inequalities A.2 and a heuristic introduction on supervised learning A.3.

2 The active clustering problem

In all this report, we consider a set of n items denoted by $\mathcal{A} = [n] = \{1, 2, \dots, n\}$. The learner get a sequence of observations $X_1, \dots, X_T$ such that X_t corresponds to the observation of the item A_t and T is the total number of samples. At each time t, the learner chooses a new item A_t and get an observation X_t distributed as $\mathcal{P}_{A_t}$ where $\mathcal{P}_{A_t}$ is a probability distribution associated to the arm A_t. We assume that there exists a hidden partition of these items such that two items in the same group share the same distribution of observations. From these observations, the learner's objective is to construct a partition of the items as close as possible to the hidden partition.

In the classical problem of clustering, the learner can not choose the order of the observations $(A_1, \dots, A_T)$ and he usually gets only one observation of each item. In the problem of active clustering, the situation is very different and at each time t, the learner can choose the next observed item A_t. The learner has to construct the sampling strategy in order to uncover as soon as possible the clustering structure of the items.

This paradigm where a learner can construct sequentially the sampling strategy based on past observation is called active learning. In our setting, we use the formalism of bandit theory: the items that we want to cluster are arms of a multi-armed bandit. We start with the definition of the multi-armed bandit environment. After that, we introduce the problem of active clustering with bandit feedback and the related notation. Then, we emphasis the connections of our setting with the bandit theory and we introduce some related papers on active clustering.

2.1 The active clustering problem: setting and notation

Multi-armed bandit environment

Our problem is a problem of active learning and we use the formalism of bandit theory: the items that we want to cluster are arms of a stochastic multi-armed bandit.

In the multi-armed stochastic bandit problem, a learner has access to a set of items (called arms) $\mathcal{A}$ each associated with an unknown distribution. At each step, the learner chooses an action of $\mathcal{A}$ and receives a feedback from the environment that depends on the chosen arm. Here is the definition of the multi-armed bandit, with the notation used in this report.

Definition 1 (Multi Armed Bandit (MAB))
In the multi-armed bandit problem, we consider a set of $n \in \mathbb{N}^*$ items (also called arm), we denote by $\mathcal{A} = [n]$ the set of available arms. Each arm is associated with a fixed probability distribution. For the arm $a \in \mathcal{A}$, we denote by $\mathcal{P}_a$ this distribution. The set $(\mathcal{A}, (\mathcal{P}_a)_a)$ is called the environment of the multi-armed stochastic bandit.

The learner do not know the environment but he can interact with it: the learner sequentially chooses one of these n arms and obtains from the environment a noisy observation also called feedback such that each observation of the arm a is distributed according to $\mathcal{P}_a$.

For $t \geqslant 1$, we denote by $A_t \in \mathcal{A}$ for the arm chosen at the step t by the learner and X_t for the associated observation. To choose A_t, the learner can use the past observations $(A_1, X_1, \ldots, A_{t-1}, X_{t-1})$. Conditionally on $A_1, X_1, \ldots, A_{t-1}, X_{t-1}, A_t$, the observation X_t is distributed according to $\mathcal{P}_{A_t}$.

Finally, we also denote by μ_a the mean of the arm $a \in \mathcal{A}$ and we assume that the observations are random vectors in $\mathbb{R}^d$ where $d \geqslant 1$ is the dimension. With this notation, we can write $X_t = \mu_{A_t} + \epsilon_t$ as the noisy observation of the vector $\mu_{A_t} \in \mathbb{R}^d$ with some (centered) noise ϵ_t.

Remark
We will study the case where d can be large, which differs from most of the literature on bandit theory in which $d = 1$.

Hidden partition bandit environment

We are now able to introduce the problem of active clustering with bandit feedback. In all this report, we will consider a stochastic bandit environment with n arms (definition 1) and d-dimensional feedback. We assume that there exists an underlying partition G_* of the arm set $\mathcal{A}$, unknown to the learner. This structure entails a partition of the arms into K groups, where arms within the same group share the same distribution, specifically the same mean vector. We call this model the hidden partition bandit environment.

Definition 2 (Hidden partition bandit environment)
Given a multi-armed bandit environment $(\mathcal{A}, (\mathcal{P}_a)_{a\in\mathcal{A}})$ (see definition 1), we say that there is an hidden partition when there exists a fixed partition of $\mathcal{A} = [n]$ denoted by $G^* = \{G_1^*, \ldots, G_K^*\}$ and $Q(1), \ldots, Q(K)$ a set of K probability distributions such that :

$$\forall k \in [K], \forall i \in G_k^*, P_i = Q(k) \ .$$

Remark
For identifiabilty reasons, we assume that the groups are all nonempty and that the the laws $Q(1), \ldots, Q(k)$ are distinct.

From the environment, the learner knows the number of arms n and the number of clusters $2 \leqslant K \leqslant n$ but the true partition G^* and the distributions $(\mathcal{P}_a)_{a\in\mathcal{A}}$ are unknown. The learner interacts with the environment and his goal is to recover the true partition of the arms.

We can now formalize what we call the player's strategy using the bandit notation. This strategy consists of an algorithm. We denote $(\mathcal{F}_t)_{t\geqslant 0}$ as the filtration $\mathcal{F}_t = \sigma(A_1, X_1, \ldots, A_t, X_t)$. We denote π as the player's strategy, which consists of three rules:

- A **selection rule** to choose the next arm A_t to play, based on the previously played arms and the obtained rewards. A_t is a $\mathcal{F}_t$-measurable random variable.
- A **stopping rule** to determine when the player should stop playing arms. This involves choosing a stopping time T with respect to the filtration $(\mathcal{F}_t)_t$. This stopping time corresponds to the number of arm sampled also referred as the budget.
- A **recommendation rule**, once the stopping time is reached, the player outputs an estimated partition of the arms $\hat{G}$. This partition is $\mathcal{F}_T$-measurable.

Definition 3 (Active clustering with bandit feedback.)
In the active clustering problem, a learner interact with a hidden partition bandit environment (definition 2) and his goal is to uncover the true partition G^* of the arms while using as few samples as possible. When the learner reach some stopping time T, he has to return a partition $\hat{G}$ of $\mathcal{A}$ into K groups which should be as close as possible to the hidden partition G^*.

The fixed confidence setting

Finally, we consider in the report what is called the fixed confidence setting. In this setting, the objective is to find an algorithm that stops in finite time and whose recommendation rule returns a clustering that is exact with probability larger than $1-\delta$. The main question is to find such an algorithm with a budget T (number of samples) as small as possible.

Definition 4
Exact recovery. For two partitions in K groups G, G', we write $G \sim G'$ if there exists $\sigma \in S_K$ such that for all $k \in [K], G_k = G'_{\sigma(k)}$.
We say that the estimation $\hat{G}$ is exact if $\hat{G} \sim G^*$ where G^* is the true partition.

Definition 5 (δ-PAC algorithm)
For a fixed confidence bound $\delta \in (0,1)$, an algorithm $\pi = \pi(\delta)$ is called δ-PAC (probably approximately correct) if for any environment :

1. the algorithm stops in finite time ($\mathbb{P}(T <= \infty) = 1$)
2. the algorithm returns the exact partition with probability larger than $1-\delta$ ($\mathbb{P}(\hat{G} \sim G^*) \geqslant 1-\delta$)

Remark
We do not consider here the fixed-budget setting where the learner has a fixed budget T and after T armed sampled has to estimate the partition with the best precision δ.

Remark
Now that all the setting is introduced, we highlight the fact that our objective is to study the budget that we need to uncover the true partition of the arms. The main questions raised by this setting are:

1. How to design a δ-PAC algorithm which uses a small number of samples?
2. What are the benefits of active sampling ?
3. What is the minimal number of samples that are needed to uncover the true partition ?

2.2 Bandit theory: a short review

In this subsection, we explain to what extent active clustering is a new problem in bandit theory and we give for that a short review on bandit theory. Especially, we will explain that active clustering is a problem of pure exploration.

The literature on bandit theory is extensive, and many active learning problems can be framed within the vocabulary of bandit. The bandit theory provides a powerful active learning framework. The simpler model used is the multi-armed bandit problem (see definition 1) with $d = 1$.

Regret minimization

The usual objective for the player interacting with a MAB is to maximize the cumulative expected feedback, which is compared to the optimal solution in hindsight, see [ACBF02] for an analysis of that problem. The optimal

solution would be to pull the arm with the largest mean but the environment is unknown to the learner. In the fixed-budget setting, the learner has a budget T fixed, and his objective is to minimize the regret defined as:

$$R_T = T\mu^* - \mathbb{E}\left[\sum_{t=1}^{T} X_t\right]$$

where we denote by $\mu^* = \max_{a \in \mathcal{A}} \mu_a$ the largest mean. Usually, we assume that there is only one optimal arm a such that $\mu_a = \mu^*$. The learner has to explore the environment to try to identify the best arm but during this exploration the learner suffers a loss each time a sub-optimal arm is sample. In the opposite direction, the learner can choose to exploit the arms that seem to have large means to maximize the cumulative feedback but he can miss the optimal arm if this one was not sampled enough. The objective in the regret-minimization setting is to find the sampling strategy that balance optimally exploration and exploitation. Various algorithms have been studied such as the Explore Then Commit algorithm or the Upper Confidence Bound algorithm. Bandit theory exhibits numerous variations, including adversarial bandits and linear bandits. For a comprehensive review, see [LS20].

It is worth noting that some authors have explored settings where a clustering structure of the arms is assumed but in the context of regret minimization. The objective in these papers is not to learn the structure, but rather to use it strategically in order to maximize rewards. For example, in [CDJ21], the learner assumes prior knowledge of the clustering structure of the arms. A clustering structure (unknown) of the users in a contextual linear bandit is also assumed to exist in the paper [GLZ14], in the framework of regret minimization. Our problem belongs in fact to the family of pure exploration problems where the only objective is exploration and not exploitation.

Pure exploration problem

In a pure exploration bandit problem, the objective is to efficiently explore the environment to uncover specific proprieties of the environment. The focus is on actively selecting the sampling strategy to reveal the underlying properties of the environment using as few samples as possible.

The most famous problem of pure exploration is the best-arm-identification problem, where the algorithm has to identify the optimal arm. In the fix-confidence setting, the algorithm has to identify the optimal arm with as few samples as possible and with a probability of error fixed. In this setting, some algorithms are proved to achieve the lower bound on the number of samples (up to a log factor). Most of the algorithms are variants of an elimination algorithm ([KKS13],[dHCMC21]) or a UCB-like algorithm ([JMNB14]). A survey can be found on [JN14]. See also [KCG16] for results on the best arm identification.

The problem of active clustering with a clustered multi-armed bandit, is a new problem that belongs to the framework of pure exploration problems.

2.3 Related Work

The problem of active clustering combines active sampling and clustering. It is a pure exploration problem, where the learner aims at discovering the underlying structure of the arms with minimal sampling. The objective is similar to that of (batch) clustering, with the distinction that items can be sampled multiple times using active sampling. Some authors have recently studied some active clustering.

On the article [AOPY19], the problem of clustering a group of items using binary feedback obtained by asking closed questions to users is studied. In particular, the authors explore the possibility that the question may be chosen in an active manner.

The problem of active clustering with bandit feedback was initially introduced in the article by Yang, Zhong, and Tan [YZT22]. In this article, the authors consider the same environment, namely a bandit with a clustering structure of the arms that they want to recover. The authors tackle this problem with an exact clustering objective and a fixed confidence bound δ. In particular, they derive a environment-dependent lower bound on $\mathbb{E}[T]$ with T the number of samples of any δ-PAC algorithm. To our knowledge, this paper introduced the first and only results and algorithms on the active clustering with bandit feedback. We will discuss later these results and compare their algorithms and the associated performance to our contribution in Section 4.

Remark

It is worth mentioning that our problem should not be confused with the problem of online clustering, for example studied in [CAGKR21]. In online clustering, the data becomes available to the learner in a sequential manner,but with the order determined by factors beyond the learner's control and the learner has to cluster the new point as soon as it is available.

3 Preliminaries on Clustering

Before any investigations on the problem of active clustering, we give here a long section of bibliography about the clustering problem. In the usual clustering problem, each item is observed once and the observations are available all at the same time to the learner. While our setting allows for multiple and sequential observations of items, we can still discuss some methods of clustering that could potentially be used as a subroutine in an active clustering algorithm, as in [YZT22]. The other goal of this section is to introduce the Gaussian model and state-of-the-art results for clustering in the Gaussian model. This will be used as a benchmark in Section 4.

In the clustering problem, we consider a set of n data $X_1, \dots, X_n$ and we want to partition the data set into sub-groups such that points in the same group are similar and points in different groups are dissimilar.

Remark
For simplicity in the presentation, we consider that the data are points in the metric space $(R^d, \|\|_2)$ and the similarity between points is measured with the Euclidean distance, but most of the algorithms can be applied to various type of data as soon as a measure of 'proximity' between the data is available.

3.1 Classical Algorithms

Various methods and criteria are employed to derive a clustering from any set of points from $X_1, \dots, X_n$ [Gir21],[KR09]. In this Section, we will discuss some of these methods that can be applied to any set of data $X_1, \dots, X_n$.

Hierarchical clustering

In hierarchical clustering, the global idea is to progressively merge the closest groups together. Initially, each data point is treated as a separate cluster, and clusters are combined based on their proximity. The outcome can be represented as a dendogram, which visually depicts all the merges. The hierarchical methods were introduced in the 60's ([WJ63]). For more details,we refer to the book [KR09] or the survey paper [MC12].

Remark
The choice of the linkage method used to determine the closeness between clusters greatly influences the resulting cluster shapes, it highlights the general fact that there is no ground rule on how a set of data has to be cluster.

The K-means algorithm

In the famous K-mean paradigm, the objective is to find K points $\hat{\mu}(1), \dots, \hat{\mu}(K)$ called centers that would describe optimally the clustering structure. These centers are chosen as:

$$\hat{\mu}(1), \dots, \hat{\mu}(K) \in \underset{\mu(1),\dots,\mu(K)}{\operatorname{argmin}} \sum_{i\in[n]} \min_{k\in[K]} \|X_i - \mu(k)\|^2.$$

As we consider the Euclidean distance, the optimization problem associated with the K-means task comes to minimize the total within-cluster sum of squared distances. The associated partition of the points called $\hat{G}^{K-means}$ is the partition where each point is associated to the group with the closest center. The main problem of exact K-means is that this combinatorial optimisation problem is proven to be NP-hard [Vat09],[MNV12] so it can not be computed in a reasonable time.

Lloyd's algorithm, also known as the K-means algorithm, is a widely used method to approximate the optimization problem in K-means clustering. It was introduced by Lloyd in 1957 [Llo82]. The algorithm leverages two key observations. First, if the partition were known, the optimal centers would be the means of the data in each cluster. Secondly, with fixed centers, the partition that minimizes the sum of squared distances between the points and centers is achieved by assigning each point to its closest center. Lloyd's algorithm takes advantage of these observations and divides the optimization into a two steps optimization repeated until some criterion is reached. Starting with an initial set of centers, the algorithm alternatively assigns each point to the closest center and then updates the centers with the means of the new clusters. The quality of the method depends on the initialization of the centers, which can be done using techniques such as K-means++ introduced in 2007 in [AV07]. During the process, the target function, representing the sum of squared distances, decreases. However, it is important to note that the algorithm might converge to a local minimum rather than the global minimum or even not converge with some bad initialisation.

Convex relaxation of K-means

Another approach to approximate the partition of K-means is through the use of relaxed K-means and Semi Definite Programming, as introduced by Peng and Wei in [PW07]. The partition of K-means can be reformulated as the optimization of a linear function:

$$\hat{G}^{K-means} = \underset{G}{\operatorname{argmax}} \left\langle XX^T, B(G) \right\rangle_F \ ,$$

where $X \in \mathcal{M}_{n,p}(\mathbb{R})$ is the matrix whose i-th row is X_i and $B(G)$ is a matrix encoding the partition G. We also have:

$$\{B(G) : G \text{ partition}\} = \{B \in S_n^+ : B^2 = B, B1 = 1, Tr(B) = K, B \geqslant 0\} \ .$$

This set is given by linear and convex conditions except the condition $B^2 = B$. By relaxing this condition, Peng and Wei transform the problem into a semi-definite programming (SDP) problem that can be solved in polynomial time. Their objective becomes: $\hat{B}^{SDP} \in \operatorname{argmax}_{B \in \mathcal{C}} \left\langle XX^T, B \right\rangle_F$, where $\mathcal{C} = \{B \in S_n^+ : B1 = 1, Tr(B) = K, B \geqslant 0\}$. The matrix $\hat{B}^{SDP}$ obtained from the SDP relaxation of K-means does not directly yield a partition of $[n]$ and to obtain a partition, a rounding procedure is applied. It is worth noting that this method has computational advantages, as it can be implemented with polynomial time complexity.

Spectral clustering

Spectral clustering is an approach that involves projecting the data points onto a lower-dimensional space using eigenvalue decomposition, see [VW04]. Then, the projected points can be clustered with any clustering algorithms.

3.2 Hidden partition Gaussian Model

Although it does not cover the various type of data that can be clustered, we will consider the following stochastic model. We traduce the idea that there exists a partition $G^* = \{G_1^*, \dots, G_K^*\}$ such that two data in the same cluster are similar by assuming that they are random variables sharing the same distribution. Then, two data in the same cluster should be close according to the Euclidean distance and two data from different clusters should be distant.

Definition 6 (Hidden partition model)
We say that the data set $X_1, \dots, X_n$ follow the hidden partition model if:

- X_1, X_n are independent random variables ,
- there exists a fixed partition of $[n]$ denoted by $G^* = \{G_1^*, \dots, G_K^*\}$ and $Q_1, \dots, Q_K$ a set of K probability distributions such that:

$$\forall k \in [K], \forall i \in G_k^*, X_i \sim Q_k \ .$$

The number of clusters $2 \leqslant K \leqslant n$ is considered as a parameter known by the learner. In this model, the goal of clustering is to use the observed data $X_1, \dots, X_n$ to estimate the true partition G^*.

Remark
This model corresponds to a special case of active clustering with bandit feedback where the budget $T = n$ and $A_1 = 1, \dots, A_n = n$. In particular, in this model, the learner can not choose the sampling strategy. This model can also be seen as the result of the uniform sampling strategy where each item is observed the same number of times and then a clustering method is applied to the empirical means of the observations of the items, we will come back on this remark in Section 4.3.

A classical model is the hidden partition Gaussian model where the observations follow Gaussian distributions .

Definition 7 (Hidden partition Gaussian model)
In the clustering problem, the Hidden partition Gaussian model is the special case of the hidden partition model where there exists n means $\mu_1, \dots, \mu_n \in \mathbb{R}^d$ and $\sigma^2 > 0$ such that:

$$\forall i \in [n], X_i \sim \mathcal{N}(\mu_a, \sigma^2 I_d) \ .$$

For $k \in [K]$, we denote by $\mu(k)$ the common mean of all items in the cluster G_k, such that, $\forall a \in G_k, \mu_a = \mu(k)$. We call $(\mu(1), \dots, \mu(K))$ the centers of the clusters.

Remark
Another classical model is the Gaussian mixture model where, the only difference is that the true partition of the points is not arbitrary and fixed but random. The labels of the arms are hidden random variables $(Y_a)_{a\in\mathcal{A}}$. Then, conditionally on this random partition, the model is the model of definition 7. In this report, we will consider the true partition as fixed but our results can easily be adapted to the mixture model.

3.3 Clustering in the hidden partition Gaussian model

In this paragraph, we discuss the guarantees of clustering methods to uncover the true partition of data in the model of hidden partition Gaussian model (definition 7).

Let define the misclassification error rate relative to the estimation $\hat{G}$ of the true partition G^*,

$$err(\hat{G}, G^*) = \frac{1}{n} \min_{\sigma\in S_K} \sum_{k=1}^{K} \sum_{a\in G_k^*} \mathbb{1}_{\{a\neq \hat{G}_{\sigma(k)}\}} \ ,$$

where S_K is the set of permutation of $[K]$. It gives the proportion of arms that are misclassified in the partition $\hat{G}$, the minimization on S_K corresponds to the fact that we only want a partition up to the order of the clusters. Then, the clustering $\hat{G}$ is called exact if $\hat{G} \sim G^*$ which corresponds to $err(\hat{G}, G^*) < 1/n$.

We now define some parameters related to the Gaussian model that will naturally appear in the analysis of the methods of clustering. The first parameter is the covariance $\sigma^2 I_d$ that define how the observations are concentrated around their means. Then, we denote as Δ for the minimal distance between centers:

$$\Delta = \min_{k\neq k'\in[K]} \|\mu(k) - \mu(k')\| \ .$$

The size of the smallest cluster denoted by m is also relevant to the problem:

$$m = \min_{k\in[K]} |G_k| \ .$$

Given two positive numbers A and B, $A \gtrsim B$ (resp $\lesssim$) means that there exists a universal constant c such that $A \geqslant cB$ (resp $\leqslant$). We write $A \asymp B$ if $A \gtrsim B$ and $A \lesssim B$.

We can review some results regarding the clustering algorithms mentioned earlier when they are applied to the hidden partition Gaussian model. We will assume that all clusters have the same size ($m = n/K$).

The K-means criterion is natural, because $\hat{G}^{K-means}$ is the MLE estimator of G^* in this Gaussian model with common covariance $\sigma^2 I_d$. The corollary 3.1 of [LZ16] state that if $\hat{G}$ is the output of Lloyd's algorithm after $\lceil 4\log(n)\rceil$ iteration and initialized with a spectral algorithm, then with some conditions on k,n,$\frac{\Delta}{\sigma}$ and d, then the misclassification error decreases exponentially fast with $\frac{\Delta^2}{\sigma^2}$.

Royer uses in [Roy17] the relaxation of K-means into a convex SDP optimisation problem introduced in [PW07]. He obtains the following theorem:

> **Theorem 1 ([Roy17])**
> If the following separation condition holds:
>
> $$\frac{\Delta^2}{\sigma^2} \gtrsim (K + \log(n)) + \sqrt{(K + \log(n))\,\frac{dK}{n}}$$
>
> then the relaxed K-means returns an exact partition $\hat{G}^{SDP}$ with high probability.

With the same method of relaxed K-means, in [GV19], Giraud and Verzelen introduce $s^2 = \frac{\Delta^2}{\sigma^2} \wedge \frac{n\Delta^4}{Kd\sigma^4}$ called the signal to noise ratio. They get the following theorem which gives the best known condition of separation if K is considered as a constant.

> **Theorem 2 ([GV19])**
> If the condition of separation:
>
> $$\frac{\Delta^2}{\sigma^2} \gtrsim \left(1 + \sqrt{\frac{d}{n}}\right) K$$

holds then with high probability,

$$err(\hat{G}^{SDP}, G^*) \leqslant c\exp(-c's^2).$$

In the theorem, c, c' are two universal constants. In fact, their result is more general and can be applied for groups with different size and variances. This gives the same condition for exact recovery as in [Roy17].

The condition of [GV19] is optimal in the case where there are two groups with symmetrical centers ($\mu(1) = -\mu(2)$) of equal size, as proved in [Nda18]. In the appendix A.3, we explain, in the Gaussian model with two balanced groups with symmetrical means, the nature of these separation condition and we give some heuristic on the definition of the SNR s^2 of [GV19].

Remark
In the previous results, high probability means that the probability converges to 1 when n goes to infinity.

In the next Section, we will use these results as a benchmark for the analysis of the active clustering problem.

4 Two algorithms for active clustering

In this Section, we come back to the problem of active clustering with bandit feedback, in the fixed confidence setting. We recall that our main objective is to design a δ-PAC algorithm with a small budget. We are also intersted in the optimal budget necessary to uncover the true partition.

We consider in this section a multi armed bandit environment with the hidden partition Gaussian model as defined in definition 2 and definition 7. The parameter σ^2 and the number of clusters K are assumed to be known by the learner. In this Gaussian model, the bandit instance is fully determined by the pair $(G^*, (\mu(1), \dots, \mu(K))$, which contains the arm partition and the centers of the clusters. We recall that in this model, for any $a \in \mathcal{A}$ such that $a \in G_k^*$ then $\mathcal{P}_a = \mathcal{N}(\mu(k), \sigma^2 I_d)$.

In Subsection 4.1, we design an original algorithm that we call ACB (1) for "active clustering for bandit", and we prove in Theorem 3 that it is δ-PAC (definition 5) in the Gaussian model. We comment the budget (number of arm samples) of our procedure. As in the condition of separation of Theorem 1, our budget is a function of Δ,σ,m,K,n and d. In particular, we compare it to the uniform sampling strategy and we highlight the benefits of active sampling. We partially discuss the questions of optimality of our result, but is still an open question that worth further investigations.

To run ACB algorithm, we assume that the learner knows the minimum gap between the centers Δ, and the the size of the smallest cluster m. This is a major drawback of the algorithm but thanks to a technique called the "doubling trick", we construct an algorithm adaptive in Δ as described in the algorithm 2 of 4.2. The proof of some technical lemmas based on concentration is postponed to Section 4.3.

4.1 ACB algorithm

4.1.1 Main result and theorem

It is now time to introduce the ACB algorithm. The algorithm can be divided in three steps:

- The first step of the algorithm consists on finding one arm by cluster. We first sample a subset of arms large enough so that each cluster is represented in this subset. Then we learn each of these arms uniformly so that with high probability, we could identify one member of each cluster.
- In the second step of the algorithm, we use the arms selected in the first part to get an estimation of the centers of the clusters.
- Finally, each arm is sampled the same number of times and associated to the closest estimated center, as in supervised learning (see the appendix A.3).

Remark
These three steps bring the algorithm close to what is done in bandit problems, such as the "explore then commit" (ETC) algorithm. In terms of regret analysis, ETC is optimal (up to a constant) if we choose the parameters of the learning phase optimally [LS20], so we can hope to achieve a similar result if we assume that Δ is known and if we consider a worst case scenario.

The following pseudo-code presents the ACB algorithm and introduces some notation.

Algorithm 1 Active clustering with bandit

Require: $\Delta, \sigma, \delta, m$

1: Take randomly $M := \left\lceil 3\frac{n}{m}\log\left(\frac{3K}{\delta}\right)\right\rceil$ arms denoted as $(b_j)_{j=1,\dots,M}$.
2: $l = \log(3M^2/\delta)$
3: Sample N times each arm b_j where $N = \frac{16\sigma^2}{\Delta^2}(16l + \sqrt{dl})$ and $j \in [M]$.
4: Compute the empirical means $\bar{\mu}_j$ of the arm b_j $(\forall j \in M)$.
5: $S = \{\}$
6: **for** $j = 1, \dots, M$ **do**
7: **if** $\exists\, i \in S$ such that $\|\bar{\mu}_i - \bar{\mu}_j\|^2 \leqslant \frac{\Delta^2}{2} + \frac{2\sigma^2}{N}d$ **then**
8: reject b_j
9: **else**
10: add b_j to S
11: **end if**
12: **end for**
13: $L = \log(12nK/\delta)$
14: Sample $J = 16(\sigma^2/\Delta^2)(8L\frac{n}{K} \vee \sqrt{dL})$ each arm in S and compute the empirical means $\hat{\mu}(1), \dots, \hat{\mu}(K)$.
15: **for** $a = 1, \dots, n$ **do**
16: Sample the arm a $I = 16(\sigma^2/\Delta^2)(8L \vee \sqrt{dL}\frac{K}{n})$ times and compute $\hat{\mu}_a$.
17: Classify a in the group $\hat{G}_{k(a)}$ such that $\hat{k}(a) \in \operatorname{argmin}_{k=1,\dots,K} \|\hat{\mu}_a - \hat{\mu}(k)\|$.
18: **end for**
19: **return** $\hat{G}_1, \dots, \hat{G}_K$

We can know state that this algorithm is δ-PAC, it returns (in finite time) an exact partition with the probability of error smaller than δ. This is the main result of this Section.

Theorem 3
If the learner knows Δ, σ and m, then the ACB algorithm (1) is δ-PAC. Moreover, the total number of samples T^{ACB} satisfies:

$$T^{ACB} \asymp \frac{\sigma^2}{\Delta^2}\left(n\log\left(\frac{n}{\delta}\right) + \left(\frac{n}{m}\log\left(\frac{K}{\delta}\right) \wedge n\right)\sqrt{d\log\left(\frac{n}{m\delta}\right)} + \sqrt{dKn\log\left(\frac{n}{\delta}\right)}\right) \ .$$

Before any comment on this theorem, we will explain further the algorithm and prove that the algorithm is indeed δ-PAC. The complete proof of Lemmas (1,2,3) is postpone to Section 4.3. As written earlier, we can consider three steps on the algorithm. The objective of the proof is to establish that with high probability, each of these steps behaves as expected. In particular the integers M, N, I, and J are chosen so that the resulting partition is correct with high probability (greater than to $1-\delta$) with the secondary objective of minimizing the total budget.

4.1.2 Proof of theorem 3

First step: representative identification

On the first step (lines 1 to 12 in the pseudo-code), the objective is to find a set S of K arms such that this set contains one arm by cluster. First of all, we take randomly M arms from $\mathcal{A}$ denoted as $b_1, \dots, b_M$. With Chernoff's bound, the choice $M := \left\lceil 3\frac{n}{m}\log\left(\frac{3K}{\delta}\right)\right\rceil$ is enough to find one arm of each cluster in each cluster with a probability greater than $1-\delta$.

Lemma 1
Recall that $M = \left\lceil 3\frac{n}{m}\log\left(\frac{3K}{\delta}\right)\right\rceil$, then

$$\mathbb{P}(\forall k \in [K], \exists\, j \in [M], k(b_j) = k) \geqslant 1 - \delta/3 \ .$$

Remark
If the clusters are all of the same size, $m = n/K$, then $M = \lceil 3K \log(3K/\delta) \rceil$.

Now, we sample uniformly these M arms, each arm is sampled N times and we compute the empirical means $\bar{\mu}_1, \dots, \bar{\mu}_M$. The number N is chosen such that with high probability :

$$\sup_{\substack{i,j \in [M]^2 \\ \mu_{b_i} = \mu_{b_j}}} \|\bar{\mu}_i - \bar{\mu}_j\|^2 \leqslant \frac{\Delta^2}{2} + \frac{2\sigma^2 d}{N} \leqslant \inf_{\substack{i,j \in [M]^2 \\ \mu_{b_i} \neq \mu_{b_j}}} \|\bar{\mu}_i - \bar{\mu}_j\|^2 \ . \tag{1}$$

The threshold $\frac{\Delta^2}{2} + \frac{2\sigma^2 d}{N}$ is natural because it is the middle between the expected square distance $\mathbb{E}\left[\|\bar{\mu_i} - \bar{\mu_i}\|^2\right]$ when b_i, b_j are in the same cluster and when b_i, b_j are in the two clusters with the closest centers. If this inequality holds, then the simple procedure of lines 5 to 12 returns K different representatives. In fact, if (1) holds, then the M arms can be clustered with a hierarchical clustering exactly and we can select easily one arm by cluster.

Lemma 2
If $N = \frac{16\sigma^2}{\Delta^2}(16 \log(3M^2/\delta) + \sqrt{d \log(3M^2/\delta)})$, then, with a probability larger than $1 - \frac{2}{3}\delta$, the first step of the algorithm identify one representative by cluster in the set S:

$$\mathbb{P}(\forall k \in [K],\ \exists! a \in S; a \in G_k^*) \geqslant 1 - \frac{2}{3}\delta \ .$$

During this first step, we used a budget of MN samples.

Second step: centers estimation

If the first step went well, we have S containing one representative by cluster. In the second step, we sample each arm in S a number J of times to get $\mu(1), \dots, \mu(K)$ an estimation of the centers (up to a permutation). This is line 14 of the algorithm.

Third step: supervised learning

Finally, we take each arm and we sample it I times to get the empirical mean of the arm $\hat{\mu}_a$. We choose then I, J simultaneously and optimally such that the distance-based classifier of line 17 returns the exact clustering of the arms. Without loss of generality, we can assume that the estimated center $\hat{\mu}(k)$ is an estimation of $\mu(k)$, the center of the cluster G_k^*. If $a \in G_k^*$, we want to learn enough the centers and the arm a so that:

$$\|\hat{\mu}_a - \hat{\mu}(k)\|^2 \leqslant \inf_{k' \neq k} \|\hat{\mu}_a - \hat{\mu}(k')\|^2 \ . \tag{2}$$

Lemma 3
If $I = 16(\sigma^2/\Delta^2)(8L \vee \sqrt{dL}\frac{K}{n})$ and $J = 16(\sigma^2/\Delta^2)(8L\frac{n}{K} \vee \sqrt{dL})$ where $L = \log(12nK/\delta)$ then,

$$\mathbb{P}(\hat{G} \sim G^*) \geqslant 1 - \delta \ .$$

Finally, the lemma 3 states that the algorithm is δ-PAC. From the algorithm, we can directly compute the (deterministic) budget T of the algorithm 1, we give the exact budget as a corollary.

Corollary 1
The total number of samples T^{ACB} used by algorithm 1 is equal to:

$$\begin{aligned} T^{ACB} = NM + KJ + NI =& 16\frac{\sigma^2}{\Delta^2} M \left(16 \log\left(\frac{3M^2}{\delta}\right) + \sqrt{d \log\left(\frac{3M^2}{\delta}\right)}\right) \\ &+ 64\frac{\sigma^2}{\Delta^2}\left(4n \log\left(\frac{12nK}{\delta}\right) \vee \sqrt{\frac{dKn}{2} \log\left(\frac{12nK}{\delta}\right)}\right) \ , \end{aligned}$$

where $M = \left\lceil 3\frac{n}{m}\log\left(\frac{3K}{\delta}\right)\right\rceil \wedge n$.

4.1.3 Comments on the ACB algorithm and Theorem 3

Benefits of active sampling compared to uniform sampling

Now that the algorithm is explained, we can comment Theorem 3 stated previously. For simplicity, we will comment the result in the situation where all the groups have the same size ($m = n/K$) and $\delta = 1/n^2$. Then the budget of ACB is:

$$\frac{\Delta^2}{\sigma^2}T^{ACB} \asymp n\log(n) + \sqrt{dK\log(n)}(\sqrt{n} + \sqrt{K}\log(n)).$$

We want to compare the budget to the strategy of uniform sampling (US) that we explain now. In the uniform sampling strategy, we sample each arm the same number of time T^{US}/n. Then, we apply a clustering algorithm to the empirical means $\hat{\mu}_1, \dots, \hat{\mu}_n$. The larger T^{US} is, the larger the SNR of the data $(\hat{\mu}_1, \dots, \hat{\mu}_n)$ is large. In fact, the uniform sampling strategy just change σ^2 into $\frac{\sigma^2 T^{US}}{n}$. We use the relaxation of K-means of [PW07] and the condition of separation for the exact clustering of Theorem 1). Then, to have an exact clustering with the uniform sampling strategy and the relaxed K-means, we must take

$$\frac{\Delta^2}{\sigma^2}T^{US} \asymp \left((nK + n\log(n)) + \sqrt{(K + \log(n))dKn}\right).$$

We highlight three different regimes:

- If $K \leqslant \log(n)$, then $\frac{\Delta^2}{\sigma^2}T^{ACB} \asymp \frac{\Delta^2}{\sigma^2}T^{US} \asymp n\log(n) + \sqrt{dKn\log(n)}$.
- If $\log(n) \leqslant K \leqslant \frac{n}{\log(n)^2}$, then $\frac{\Delta^2}{\sigma^2}T^{ACB} \asymp n\log(n) + \sqrt{dKn\log(n)}$ which is smaller that $\frac{\Delta^2}{\sigma^2}T^{US} \asymp nK + \sqrt{dK^2n}$.
- If $\frac{n}{\log(n)^2} \leqslant K$, then $\frac{\Delta^2}{\sigma^2}T^{ACB} \asymp n\log(n) + \sqrt{dK^2\log(n)^3}$ which is smaller that $\frac{\Delta^2}{\sigma^2}T^{US} \asymp nK + \sqrt{dK^2n}$.

In each case, we get a smaller number of samples. We believe the gap between the Uniform strategy and the ACB algorithm highlights the benefit that we can get when it is possible to use active sampling for clustering. We think that the third regime is artificial and that it comes from a problem of method that could be avoid.

Optimality?

The question of optimality is still an open question and this is the next step on our investigation. There exists already one lower bound that can be found in [YZT22]. In this paper, the authors prove a lower bound that depends on the environment. We do not give the general bound here but only in the case of equal size of clusters and equidistant centers. In this case, for any δ-PAC algorithm with budget T (a stopping time) then

$$\mathbb{E}[T] \geqslant d_{KL}(\delta, 1-\delta)\frac{2(n+K)\sigma^2}{\Delta^2}$$

where $d_{KL}(a, b)$ is the Kullback?Leibler divergence between two Bernoulli's variable with parameter a and b and $d_{KL}(\delta, 1-\delta) \sim \log(1/\delta)$ in the asymptotic regime where $\delta \to 0$. In [YZT22], an algorithm is introduced and this algorithm achieve the lower bound when $\delta \to 0$. If n, K and d are considered as fixed, the asymptotic budget of ACB algorithm when $\delta \to 0$ is $T^{ACB} \underset{\delta\to 0}{\sim} c'\log(1/\delta)\frac{n\sigma^2}{\Delta^2}$ which is optimal, up to the large universal constant c'. In [YZT22], the authors get exactly the good constant but with the price of this asymptotic regime that hide for instance the dependency on the dimension.

For the optimality (if δ is arbitrary), we believe that the method we used in the first step of the algorithm can be improved and we hope to get rid of the term $\left(\frac{n}{m}\log\left(\frac{n}{m\delta}\right) \wedge n\right)\sqrt{d\log\left(\frac{n}{m\delta}\right)}$. After that, we hope that the budget $\frac{\sigma^2}{\Delta^2}\left(n\log\left(\frac{n}{\delta}\right) + \sqrt{dKn\log\left(\frac{n}{\delta}\right)}\right)$ is optimal.

4.2 Adaptive version of ACB

4.2.1 The doubling trick

In the ACB algorithm, we assumed that the minimal gap Δ was known to design the algorithm. However, this strong assumption is not reasonable and needs to be relaxed. In this Section, the minimal gap Δ is considered as unknown, we will use the "doubling trick" to construct an adaptive algorithm that does not require the knowledge of Δ. Up to a logarithmic term, we will achieve the same performance in terms of budget.

Explanation of the ACB+ algorithm

First, we explain the doubling-trick and the ideas behind the algorithm. The ACB+ algorithm is a modified version of ACB where we use the "doubling trick". The main difference between ACB and ACB+ is in the first step where we want to identify the set of K representatives.

Imagine that the ACB algorithm is applied with Δ' where Δ' is different from the true minimal gap Δ. If $\Delta' \leqslant \Delta$ then the identification of K representatives goes well but more samples than necessary are spent. On the contrary, if $\Delta' \geqslant \Delta$ then the set of representative S can contain less than K representative. The idea of the doubling trick is to perform the first step of the ACB algorithm with smaller and smaller gap $\Delta^{(0)} \geqslant \Delta^{(1)} \geqslant \cdots$ until the set S contains K arms. For $p \geqslant 0$, we take $\Delta^{(p)} = \frac{\Delta^{(p-1)}}{2}$ and we construct $S^{(p)}$, we call this part of ACB+ the p-th epoch. We denote by $\rho := \min\{p \geqslant 0 : |S^{(p)}| \geqslant K\}$ for the first epoch such that $S^{(p)}$ contains K arms . The consequence of dividing by two the gap at each epoch is that the total budget for all epochs is dominated by the budget used in the last epoch ρ. This is the doubling trick. For each epoch, we want to have a probability of error smaller than $\frac{2\delta^{(p)}}{3}$ and such that $\sum_p \delta^{(p)} \leqslant \delta$. The only drawback is that this will bring an additional logarithmic term in the total budget.

We will also prove that the gap $\Delta^{(\rho)}$ is really close to the true minimal gap Δ. In particular, we use the estimation $\hat{\Delta} = \frac{\Delta^{(\rho)}}{2}$ in the end of the algorithm to construct the estimation with $S^{(\rho)}$ and $\hat{\Delta}$. We finally return the estimation of the true partition denoted as $\hat{G}$

We give in 2 the pseudo-code of the ACB+ algorithm and in Theorem 4 the corresponding theorem.

Algorithm 2 ACB+

Require: δ, σ^2

Initialize $p=0$, $\Delta^{(0)}$, $S=\emptyset$
while $|S|<K$ **do**
  $\delta^{(p)} = \frac{\delta}{2(p+1)^2}$
  Take randomly $M^{(p)} = 3\frac{n}{m}\log\left(\frac{3K}{\delta^{(p)}}\right)$ arms denoted by $(b_j^{(p)})_{j=1,\dots,M^{(p)}}$.
  $l^{(p)} = \log\left(\frac{3(M^{(p)})^2}{\delta^{(p)}}\right)$
  $N^{(p)} = \left(\frac{16\sigma}{\Delta^{(p)}}\right)^2\left(l^{(p)} + \sqrt{dl^{(p)}}\right)$
  Sample $N^{(p)}$ times each arm in $(b_j^{(p)})_{j\in M^{(p)}}$.
  Compute the empirical means $\bar{\mu}_1^{(p)},\dots,\bar{\mu}_{M^{(p)}}^{(p)}$.
  $S=\emptyset$
  for $j=1,\dots,M^{(p)}$ **do**
    if $\forall b_i^{(p)}\in S, \|\bar{\mu}_i^{(p)} - \bar{\mu}_j^{(p)}\|^2 > \frac{(\Delta^{(p)})^2}{2} + \frac{2\sigma^2 d}{N^{(p)}}$ **then**
      Add $b_j^{(p)}$ to S
    end if
  end for
  Update $\Delta^{(p+1)} = \frac{\Delta^{(p)}}{2}$ and $p=p+1$
end while
$\hat{\Delta} := \Delta^{(p-1)}/2$
$L = \log(12nK/\delta)$
Sample $J_{\hat{\Delta}} = 32\frac{\sigma^2}{\hat{\Delta}^2}(4L\frac{n}{K} \vee \sqrt{\frac{ndL}{2K}})$ times each arm in S and compute the empirical means $\hat{\mu}(1),\dots,\hat{\mu}(K)$.
for $a=1,\dots,n$ **do**
  Sample the arm a $I_{\hat{\Delta}} = 32\frac{\sigma^2}{\hat{\Delta}^2}(4L \vee \sqrt{\frac{KdL}{2n}})$ times and compute $\hat{\mu}_a$.
  Classify a in the group $\hat{G}_k$ such that $k\in \operatorname{argmin}_{k=1,\dots,K}\|\hat{\mu}_a - \hat{\mu}(k)\|$.
end for
return $\hat{G}_1,\dots,\hat{G}_K$

Theorem 4
The ACB+ algorithm (2) is δ-PAC and with a probability larger than $1-\delta$,

$$T \leqslant c\frac{\sigma^2}{\Delta^2}\left(n\log\left(\frac{n}{\delta}\right) + \sqrt{dKn\log\left(\frac{n}{\delta}\right)}\right) + c\frac{\sigma^2}{\Delta^2}\left(\frac{n}{m}\log\left(\frac{K}{\delta}\frac{\sigma}{\Delta}\right)\wedge n\right)\left(\log\left(\frac{n}{m\delta}\frac{\sigma}{\Delta}\right) + \sqrt{d\log\left(\frac{n}{m\delta}\frac{\sigma}{\Delta}\right)}\right).$$

4.2.2 Proof of theorem 4

We can now explain more precisely the ACB+ algorithm, the "doubling-trick" and prove that the algorithm is indeed δ-PAC. We use in this proof the notation introduced in the algorithm and in the preceding explanation.

We consider an epoch $p \geqslant 0$. From Lemma 1, with probability larger than $1-\frac{\delta^p}{3}$, the set of arms selected at the epoch p, denoted as $(b_j^{(p)})_{j=1,\dots,M^{(p)}}$ contains one member of each cluster at least. Then, from the concentrations inequalities 11 and 12, we prove as in Lemma 2 (using both side of the concentration inequality 12), that with a probability larger than $1-\frac{\delta^{(p)}}{3}$ then $\forall i<j\in[M^{(p)}]^2$,

- if $\|\mu_{b_i^{(p)}} - \mu_{b_j^{(p)}}\|^2 \leqslant \frac{(\Delta^{(p)})^2}{4}$ then $\|\mu_{\bar{b}_i^{(p)}} - \mu_{\bar{b}_j^{(p)}}\|^2 \leqslant \frac{(\Delta^{(p)})^2}{2} + \frac{2\sigma^2 d}{N^{(p)}}$,
- if $\|\mu_{b_i^{(p)}} - \mu_{b_j^{(p)}}\|^2 \geqslant (\Delta^{(p)})^2$ then $\|\mu_{\bar{b}_i^{(p)}} - \mu_{\bar{b}_j^{(p)}}\|^2 \geqslant \frac{(\Delta^{(p)})^2}{2} + \frac{2\sigma^2 d}{N^{(p)}}$.

From this result, we can directly prove the following lemma:

Lemma 4
For any epoch $p \geqslant 0$, recall that $S^{(p)}$ contains the arms selected at the end of the p-th epoch. With a probability larger than $1 - \frac{2\delta^{(p)}}{3}$,

1. $S^{(p)}$ does not contain two arms from the same cluster,
2. if $\Delta^{(p)} \leqslant \Delta$ then $S^{(p)}$ contains exactly one arm by cluster,
3. if $\Delta^{(p)} \geqslant 2\Delta$ then $S^{(p)}$ contains strictly less than K arms.

We used $\delta^{(p)} = \frac{\delta}{(p+1)^2}$ at epoch p in order to have $\sum_p \delta^{(p)} \leqslant 1$. From Lemma 4 and a union bound on all epochs, it holds that with a probability larger than $1 - \frac{2}{3}\delta$ then:

1. $S^{(\rho)}$ contains exactly one arm by cluster,
2. $\Delta^{(\rho-1)} \geqslant \Delta$,
3. $\Delta^{(\rho)} \leqslant 2\Delta$.

Finally, we consider the estimation $\hat{\Delta} := \frac{\Delta^{(\rho)}}{2}$, which satisfies $\mathbb{P}(\frac{1}{4}\Delta \leqslant \hat{\Delta} \leqslant \Delta) \geqslant 1 - 2\delta/3$. From this estimation and the set $S^{(\rho)}$, we construct the estimation $\hat{G}$ with the same construction as ACB. A union bound and Lemma 3 ensures that with a probability larger than $1 - \delta$, the output of ACB+ $\hat{G}$ is the true partition.

It remains to study the number of sample T which, contrary to the ACB algorithm, is now a random variable. First of all, we have to check that $\mathbb{P}(T < \infty) = 1$. We have $\{T < \infty\} = \{\rho < \infty\}$. From Lemma 4, it appears that for any p such that $\Delta^{(p)} \leqslant \Delta$, $\mathbb{P}(|S^{(p)}| < K) \leqslant \delta^{(p)} \leqslant \delta$ and $\mathbb{P}(\rho < \infty) = 1$.

Finally, we compute T. In the algorithm, we took $(\Delta^{(0)})^2 = 256\sigma^2(l^{(0)} + \sqrt{dl^{(0)}})$ to have $N^{(0)} = 1$. Recall that $\Delta^{(p)} = \frac{\Delta^{(0)}}{2^p}$ and

$$N^{(p)} = \left(\frac{16\sigma}{\Delta^{(p)}}\right)^2 \left(l^{(p)} + \sqrt{dl^{(p)}}\right) = \left(\frac{16\sigma}{\Delta^{(0)}}\right)^2 4^p \left(l^{(p)} + \sqrt{dl^{(p)}}\right).$$

The total budget used to construct $S^{(\rho)}$ is:

$$\begin{aligned}
\sum_{p=0}^{\rho} N^{(p)} M^{(p)} &= \sum_{p=0}^{\rho} \left(\frac{16\sigma}{\Delta^{(0)}}\right)^2 4^p \left(l^{(p)} + \sqrt{dl^{(p)}}\right) M^{(p)} \\
&\leqslant \left(\frac{16\sigma}{\Delta^{(0)}}\right)^2 \left(l^{(\rho)} + \sqrt{dl^{(\rho)}}\right) M^{(\rho)} \sum_{p=0}^{\rho} 4^p \\
&\leqslant \left(\frac{16\sigma}{\Delta^{(0)}}\right)^2 \left(l^{(\rho)} + \sqrt{dl^{(\rho)}}\right) M^{(\rho)} \frac{4^{\rho+1}}{3}
\end{aligned}$$

Now, with probability larger than $1 - \delta$, $\Delta \leqslant 2\Delta^{(\rho)}$, and $4^\rho \leqslant \left(\frac{2\Delta^{(0)}}{\Delta}\right)^2$. In particular, $\rho \leqslant \log_4(\left(\frac{2\Delta^{(0)}}{\Delta}\right)^2)$. We notice that the budget for step 2 and 3 is almost the same as in ACB because with probability larger than $1 - \Delta$, $\hat{\Delta} \geqslant \Delta/4$. Finally, up to a universal constant, with a probability larger than $1 - \delta$,

$$\begin{aligned}
T \lesssim & \frac{\sigma^2}{\Delta^2}\left(n \log\left(\frac{n}{\delta}\right) + \sqrt{dKn \log\left(\frac{n}{\delta}\right)}\right) \\
& + \frac{\sigma^2}{\Delta^2}\left(\frac{n}{m} \log\left(\frac{K}{\delta}\frac{\sigma}{\Delta}\right) \vee n\right)\left(\log\left(\frac{n}{m\delta}\frac{\sigma}{\Delta}\right) + \sqrt{d \log\left(\frac{n}{m\delta}\frac{\sigma}{\Delta}\right)}\right) \quad .
\end{aligned}$$

4.3 Proofs of lemmas 1, 2 and 3

Proof of Lemma 1

Let $k \in [K]$, and for $j \in [M]$, $Y_j(k) := \mathbb{1}_{\{b_j \in G_k^*\}}$, we have: $\sum_{j \in [M]} Y_j(k) = |\{j \in [M] : b_j \in G_k^*\}|$. The random variables $(Y_j(k))_{j \in [M]}$ are independent Bernoulli's random variables with parameter $\theta(k) := \frac{|G_k^*|}{n} \geqslant \frac{m}{n}$. Chernoff's

inequality (Lemma 14) with $\alpha = \sqrt{2/3}$ and the expression of M gives:

$$\mathbb{P}\left(\sum_{j\in[M]} Y_j(k) < (1-\alpha)M\theta(k)\right) \leqslant \exp\left(-\frac{\alpha^2 M\theta(k)}{2}\right) \leqslant \exp\left(-\frac{\alpha^2 Mm}{2n}\right) \leqslant \frac{\delta}{3K} \ .$$

We notice that $(1-\alpha)M\theta(k) > 0$, and then $\mathbb{P}\left(\sum_{j\in[M]} Y_j(k) = 0\right) \leqslant\leqslant \frac{\delta}{3K}$. This exactly means that the probability that the cluster G_k^* is not represented in $(b_1, \dots, b_M)$ is smaller than $\delta/3K$.

Finally, a union bound on $k \in [K]$ concludes the proof ■

Proof of Lemma 2

Let $j \in [M]$ and S the set of arms selected before j by the algorithm. We have to choose N large enough such that, with high probability, the algorithm adds b_j to S if and only if the cluster of b_j is not represented in S yet. Consider b_i with $i < j$ and such that $b_i \in S$. We recall that we sample N times the arms b_i and b_j and $\bar{\mu}_i$, $\bar{\mu}_j$ are the corresponding empirical means.

First case: b_i and b_j have the same mean.

If $\mu_{b_i} = \mu_{b_j}$, then $\bar{\mu}_i - \bar{\mu}_j \sim \mathcal{N}(0, \frac{2\sigma^2}{N})$ and $\|\bar{\mu}_i - \bar{\mu}_j\|^2 \sim \frac{2\sigma^2}{N}\chi^2_d$. With the inequality of concentration of the χ^2 distribution in appendix (Lemma 12), with probability larger than $1 - \frac{2\delta}{3M^2}$:

$$\|\bar{\mu}_i - \bar{\mu}_j\|^2 \leqslant \frac{2\sigma^2}{N}(d + 2\sqrt{dl} + 2l) \text{ where } l = \log(3M^2/2\delta).$$

Our choice of N enforces

$$N \geqslant 4\frac{\sigma^2}{\Delta^2}(\sqrt{dl} + l). \tag{3}$$

which is equivalent to $\frac{2\sigma^2}{N}(d + 2\sqrt{dl} + 2l) \leqslant \frac{\Delta^2}{2} + \frac{2\sigma^2}{N}d$. This is exactly what we need to ensure that

$$\|\bar{\mu}_i - \bar{\mu}_j\|^2 \leqslant \frac{\Delta^2}{2} + \frac{2\sigma^2}{N}d \ .$$

Second case: different clusters.

Now, assume that $\mu_{b_i} \neq \mu_{b_j}$,

$$\begin{aligned}
\|\bar{\mu}_i - \bar{\mu}_j\|^2 =& \|(\bar{\mu}_i - \mu_{b_i}) - (\bar{\mu}_j - \mu_{b_j}) + (\mu_{b_i} - \mu_{b_j})\|^2 \\
=& \|(\bar{\mu}_i - \mu_{b_i}) - (\bar{\mu}_j - \mu_{b_j})\|^2 + \|\mu_{b_i} - \mu_{b_j}\|^2 \\
&+ 2\langle(\bar{\mu}_i - \mu_{b_i}) - (\bar{\mu}_j - \mu_{b_j})|\mu_{b_i} - \mu_{b_j}\rangle \\
=& \frac{2\sigma^2}{N}Z_1 + 2\sigma\sqrt{2/N}\|\mu_{b_i} - \mu_{b_j}\|Z_2 + \|\mu_{b_i} - \mu_{b_j}\|^2 \ ,
\end{aligned}$$

where $Z_1 \sim \chi^d_2$ and $Z_2 \sim \mathcal{N}(0,1)$.

We use the χ^2 concentration inequality (Lemma 12), the Gaussian concentration (11) and a union bound; with probability larger than $1 - 2\delta/3M^2$,

$$\|\bar{\mu}_i - \bar{\mu}_j\|^2 \geqslant \frac{2\sigma^2}{N}(d - 2\sqrt{d\tilde{l}}) - 2\sigma\sqrt{2/N}\|\mu_{b_i} - \mu_{b_j}\|\sqrt{2\tilde{l}} + \|\mu_{b_i} - \mu_{b_j}\|^2$$

where $\tilde{l} = \log(3M^2/\delta)$.

With the threshold chosen in the algorithm, we have to check that

$$\frac{2\sigma^2}{N}(d - 2\sqrt{d\tilde{l}}) - 2\sigma\sqrt{2/N}\|\mu_{b_i} - \mu_{b_j}\|\sqrt{2\tilde{l}} + \|\mu_{b_i} - \mu_{b_j}\|^2 \geqslant \frac{\Delta^2}{2} + 2\frac{\sigma^2}{N}d.$$

Or equivalently:

$$4\frac{\sigma^2}{N}\sqrt{d\tilde{l}} + 4\sigma\sqrt{\frac{\tilde{l}}{N}}\|\mu_{b_i} - \mu_{b_j}\| \leqslant \|\mu_{b_i} - \mu_{b_j}\|^2 - \frac{\Delta^2}{2}.$$

We highlight that $N\frac{16\sigma^2}{\Delta^2}(16\tilde{l}+\sqrt{d\tilde{l}})$ was chosen to satisfy the two following inequalities:

$$4\frac{\sigma^2}{N}\sqrt{d\tilde{l}} \leqslant \frac{\Delta^2}{4} \tag{4}$$

$$4\sigma\sqrt{\frac{\tilde{l}}{N}} \leqslant \frac{\Delta}{4}. \tag{5}$$

From equations 4 and 5, we can easily conclude that with a probability larger than $1-2\delta/3M^2$,

$$\|\bar{\mu}_i - \bar{\mu}_j\|^2 \geqslant \frac{\Delta^2}{2} + \frac{2\sigma^2}{N}d \ .$$

From these two case, we conclude with a union bound on $i<j\in[M]^2$ that with a probability larger than $1-\frac{\delta}{3}$,

$$\sup_{\substack{i,j\in[M]^2\\ \mu_{b_i}=\mu_{b_j}}} \|\bar{\mu}_i - \bar{\mu}_j\|^2 \leqslant \frac{\Delta^2}{2} + \frac{2\sigma^2 d}{N} \leqslant \inf_{\substack{i,j\in[M]^2\\ \mu_{b_i}\neq\mu_{b_j}}} \|\bar{\mu}_i - \bar{\mu}_j\|^2 \ .$$

If these inequalities hold, S contains one representative by cluster (if this cluster is represented in $(\beta_1,\dots,\beta_M)$). We finally conclude the proof with Lemma 1 and a union bound. ■

Proof of Lemma 3

In this last part of the algorithm, we essentially reduce the problem to a case of supervised classification in the context of a Gaussian model with unknown means. We consider the high-probability event from the previous lemma, which gives us the set S containing one representative from each group. We sample each of these K arms J times. Without loss of generality, we order the clusters of G_k^* such that $\hat{\mu}(1),\dots,\hat{\mu}(K)$, are the estimates of the centers $\mu(1),\dots,\mu(K)$.
Let a be an arm, and let $\hat{\mu}_a$ be the empirical mean of arm a after drawing it I times. We classify a in the cluster $G_{\hat{k}(a)}$ using the following classifier: $\hat{k}(a)\in \underset{k\in[K]}{\operatorname{argmin}} \|\hat{\mu}_a-\hat{\mu}(k)\|$. We assume that $a\in G_k^*$ meaning $\mu_a=\mu(k)$. Now, we examine the probability of error in the classification of arm a. To ensure that we can accurately classify each arm, we want the probability of making a mistake for the arm a to be less than $\delta/3N$.

$$\begin{aligned}\mathbb{P}(\hat{k}(a)\neq k) =& \mathbb{P}\left(\exists i=1,\dots,K, i\neq k;\ \|\hat{\mu}_a-\hat{\mu}(i)\|<\|\hat{\mu}_a-\hat{\mu}(k)\|\right)\\ \leqslant& \sum_{i\neq k}^{K}\mathbb{P}\left(\|\hat{\mu}_a-\hat{\mu}(i)\|<\|\hat{\mu}_a-\hat{\mu}(k)\|\right)\end{aligned}$$

Without loss of generality, we assume that $a\in G_1^*$ and we study the event $\{\|\hat{\mu}_a-\hat{\mu}(2)\|<\|\hat{\mu}_a-\hat{\mu}(1)\|\}$ Now, we observe the following geometric equality:

$$\{\|\hat{\mu}_a-\hat{\mu}(2)\|<\|\hat{\mu}_a-\hat{\mu}(1)\|\} = \left\{\left\langle \hat{\mu}_a - \frac{\hat{\mu}(1)+\hat{\mu}(2)}{2}, \frac{\hat{\mu}(1)-\hat{\mu}(2)}{2}\right\rangle <0\right\}$$

We have

$$\begin{cases}\hat{\mu}_a = \mu(1)+\frac{\sigma}{\sqrt{I}}\varepsilon_a\\ \hat{\mu}(1)=\mu(1)+\frac{\sigma}{\sqrt{J}}\varepsilon(1)\\ \hat{\mu}(2)=\mu(2)+\frac{\sigma}{\sqrt{J}}\varepsilon(2)\end{cases}$$

with $\varepsilon_a,\varepsilon(1)$ and $\varepsilon(2)$ independent and distributed as $\mathcal{N}(0,I_d)$. We denote u as the unit vector $\frac{\mu(1)-\mu(2)}{\|\mu(1)-\mu(2)\|}$, we can write:

$$\begin{aligned}\left\langle \hat{\mu}_a - \frac{\hat{\mu}(1)+\hat{\mu}(2)}{2}, \frac{\hat{\mu}(1)-\hat{\mu}(2)}{2}\right\rangle =& \left\langle \frac{\mu(1)-\mu(2)}{2}+\frac{\sigma}{\sqrt{I}}\varepsilon_a - \frac{\sigma}{2\sqrt{J}}(\varepsilon(1)+\varepsilon(2))\ ,\ \frac{\mu(1)-\mu(2)}{2}+\frac{\sigma}{2\sqrt{J}}(\varepsilon(1)-\varepsilon(2))\right\rangle\\ =& \frac{1}{4}\|\mu(1)-\mu(2)\|^2 + \frac{\sigma}{2\sqrt{I}}\|\mu(1)-\mu(2)\|\langle\varepsilon_a,u\rangle - \frac{\sigma}{2\sqrt{J}}\|\mu(1)-\mu(2)\|\langle\varepsilon(2),u\rangle\\ &- \frac{\sigma^2}{2J}\left\langle\frac{\varepsilon(1)+\varepsilon(2)}{\sqrt{2}}, \frac{\varepsilon(1)-\varepsilon(2)}{\sqrt{2}}\right\rangle + \frac{\sigma^2}{\sqrt{2IJ}}\left\langle\varepsilon_a, \frac{\varepsilon(1)-\varepsilon(2)}{\sqrt{2}}\right\rangle\end{aligned} \ .$$

We have isolated several terms and we aim to control their variations.

First, we need to control the first two terms, which correspond to the noise in the observation of the mean $\mu(2)$ and the mean μ_a in the direction given by u. Both of these terms are Gaussian and can be controlled with Lemma 11. Then, we need to control the cross-terms $\left\langle \frac{\varepsilon(1)+\varepsilon(2)}{\sqrt{2}}, \frac{\varepsilon(1)-\varepsilon(2)}{\sqrt{2}} \right\rangle$ and $\left\langle \varepsilon_a, \frac{\varepsilon(1)-\varepsilon(2)}{\sqrt{2}} \right\rangle$, which are dot products of independent standard Gaussian vectors. We use for that Corollary 2 which is a consequence of Hanson-Wright inequality. With a probability greater than $1-\frac{\delta}{3NK}$, and with $L = \log(12nK/\delta)$,the following four inequalities hold:

$$-\frac{\sigma}{2\sqrt{I}}\|\mu(1)-\mu(2)\|\langle \varepsilon_a, u\rangle \leqslant \frac{\sigma}{2\sqrt{I}}\|\mu(1)-\mu(2)\|\sqrt{2L}.$$

$$\frac{\sigma}{2\sqrt{J}}\|\mu(1)-\mu(2)\|\langle \varepsilon(2), u\rangle \leqslant \frac{\sigma}{2\sqrt{J}}\|\mu(1)-\mu(2)\|\sqrt{2L}$$

$$\frac{\sigma^2}{\sqrt{2IJ}}\left\langle \varepsilon_a, \frac{\varepsilon(2)-\varepsilon(1)}{\sqrt{2}} \right\rangle \leqslant \frac{\sigma^2}{\sqrt{2IJ}}(\sqrt{4dL}\vee 4L)$$

$$\frac{\sigma^2}{2J}\left\langle \frac{\varepsilon(1)-\varepsilon(2)}{\sqrt{2}}, \frac{\varepsilon(1)+\varepsilon(2)}{\sqrt{2}} \right\rangle \leqslant \frac{\sigma^2}{2J}(\sqrt{4dL}\vee 4L)\ .$$

Finally, we recall that in the algorithm, $I = 32\frac{\sigma^2}{\Delta^2}(4L \vee \sqrt{\frac{KdL}{2n}})$ and $J = 32\frac{\sigma^2}{\Delta^2}(4L\frac{n}{K} \vee \sqrt{\frac{ndL}{2K}})$. This choice was made to enforce that the sum of these four terms is smaller than $\frac{1}{4}\|\mu(1)-\mu(2)\|^2$ while minimizing the number of samples $nI + KJ$.

Finally, for any $a \in G_k^*$ and $k' \neq k$, we proved that with probability greater than $1-\frac{\delta}{3nK}$, then $\|\hat{\mu}_a - \mu(k)\| < \|\hat{\mu}_a - \hat{\mu}(k')\|$. A union bound on $a \in [n]$ and $k \in [K]$ and the lemma 2 conclude that the estimated partition is exact with a probability larger than δ. ■

5 Conclusion

In this report, we formalized the new problem of active clustering with bandit feedback in which there is a set of arms that can be clustered. In order to cluster the arms, the learner construct actively a sequence of observations. We tackled the problem in the fixed confidence setting and our problematic was to study the minimal number of samples that the learner needs to uncover the true partition. We proposed an algorithm called ACB and a variant ACB+ that are δ-PAC in the hidden partition gaussian model and we analysed the budget of the procedure. The performance of our algorithm highlights the benefits of using active sampling to solve a clustering problem and we used for that state-of-the art batch clustering as a benchmark.

The result presented in this report are a restitution of a work in progress and our algorithm suffers several drawbacks that we believe can be fixed in the future, we believe that the ACB(+) algorithm can be improved. In the first step of ACB+, the identification of the representatives can probably be modified in order to improve the total budget and we also need to get rid of the dependency in the minimal size of the cluster m. Our objective in the future is also to design an algorithm that is adaptive with more precision to the environment, for instance to the size of every cluster and all distance between cluster. We could for instance use elimination techniques or UCB-like techniques inspired by the best arm identification problem but anyway, we will have to use all the opportunities of active sampling.

We can easily generalize our results to the sub-Gaussian model, but the study of general models is also interesting. It is worth developing active sampling techniques for other clustering problems such as hierarchical clustering or in the stochastic block model.

The question of optimality will naturally be the next step of our investigation. We believe that active sampling could improve the computational limitation of clustering and to prove that, an important direction is to develop non asymptotic lower bounds. Our intuition from the clustering literature is that our algorithm will be almost optimal for a worst case bound.

6 Representative identification

6.1 Toy model $K = 2$

We assume for simplicity that $K = 2$. There are two groups G_1^* and G_2^* unknown but $\Delta = \|\mu(1)-\mu(2)\|$, $m = \min(\#G_1^*, \#G_2^*)$ and σ^2 are assumed to be known.

Algorithm 3 Sequential identification for $K = 2$

Require: δ,σ^2,m and Δ

Initialize $u = 0$, $S = \{\}$.
Take randomly $a_0 \in \mathcal{A}$ and add a_0 to S.
while $|S| < K = 2$ **do**
  Set $u = u + 1$, $r_u = \lceil \log_2 \left(\log \left(\frac{u^2}{\delta} \frac{\pi^2}{6} \right) \right) \rceil$ and $T_0 = 8\sigma^2/\Delta^2$.
  Take randomly one arm denoted by a_u.
  for $s = 1, \dots, r_u$ **do**
    Sample $N_s = (32 \times 2^s + 2\sqrt{d2^s})T_0$ times arms a_u and a_0 and compute the empirical means $\bar{\mu}_s^{(0)}$ and $\bar{\mu}_s^{(u)}$.
    if $\|\bar{\mu}_s^{(u)} - \bar{\mu}_s^{(0)}\|^2 \leqslant \frac{\Delta^2}{2} + \frac{2\sigma^2 d}{N_s}$ **then**
      Reject a_u.
    end if
  end for
  Add a_u to S.
end while
return S

Remark
Do we use independent samples from a_0 or always the same samples ?

Notation
For $u \geqslant 1$, We refer as the epoch u for the part of the algorithm where the arm a_u is taken randomly and the algorithm decide if a_u should be added to S or not. Let $s \geqslant 1$, we consider $\phi_s^u := \mathbb{1}_{\left\{ \|\bar{\mu}_s^{(u)} - \bar{\mu}_s^{(0)}\|^2 > \frac{\Delta^2}{2} + \frac{2\sigma^2 d}{N_s} \right\}}$ where $\bar{\mu}_s^{(u)}$ is computed with $N_s = (32 \times 2^s + 2\sqrt{d2^s})T_0$ samples of a_u and $T_0 = 8\sigma^2/\Delta^2$. We denote by τ_u for the time of rejection of a_u, $\tau_u = \min_{s \geqslant 1} \phi_s^u = 0$. If $\tau_u \leqslant r_u$ then a_u is eliminated after τ_u iterations. If $\tau_u > r_u$ then a_u is added to S after r_u iteration.
At each epoch, we consider the test $H_0^u : \mu_{a_0} = \mu_{a_u}$ versus $H_1^u : \|\mu_{a_0} - \mu_{a_u}\| \geqslant \Delta$ and we compute the statistic $\psi_u = \mathbb{1}_{\{\tau_u > r_u\}}$.
The total number of epoch is $M = \min\{u \leqslant 1 : \psi_u = 1\}$. Finally, the algorithm return $S = \{a_0, a_M\}$. The total budget is denoted by T.

Lemma 5
Let $u \geqslant 1$ and $s \geqslant 1$

$$\mathbb{P}_{H_0}(\phi_s^u = 1) \leqslant \exp(-2^s) \tag{6}$$
$$\mathbb{P}_{H_1}(\phi_s^u = 0) \leqslant 2\exp(-2^s) \quad . \tag{7}$$

Proof
First case under H_0^u.
Compute $\mathbb{P}_{H_0}(\phi_s^u = 1)$.

$$\begin{aligned}
\mathbb{P}_{H_0}(\phi_s^u = 1) &= \mathbb{P}_{H_0}\left(\|\bar{\mu}_s^{(u)} - \bar{\mu}_s^{(0)}\|^2 > \frac{\Delta^2}{2} + \frac{2\sigma^2 d}{N_s} \right) \\
&= \mathbb{P}\left((\|\epsilon\|^2 - d) > \frac{\Delta^2}{4\sigma^2} N_s \right) \\
&\leqslant \mathbb{P}\left((\|\epsilon\|^2 - d) > 2(2^s + \sqrt{d2^s}) \right) \\
&\leqslant \exp(-2^s)
\end{aligned}$$

Second case under H_1^u.
Compute $\mathbb{P}_{H_1}(\phi_s^u = 0)$.

$$
\begin{aligned}
\mathbb{P}_{H_1}(\phi_s^u = 0) &= \mathbb{P}_{H_1}\left(\|\bar{\mu}_s^{(u)} - \bar{\mu}_s^{(0)}\|^2 \leqslant \frac{\Delta^2}{2} + \frac{2\sigma^2 d}{N_s}\right) \\
&= \mathbb{P}\left(\left\|\frac{\mu_{a_u} - \mu_{a_0}}{\sigma}\sqrt{\frac{N_s}{2}} + \epsilon\right\|^2 - d \leqslant \frac{\Delta^2}{4\sigma^2}N_s\right) \\
&= \mathbb{P}\left(\|x_s^u + \epsilon\|^2 - d \leqslant 2\tilde{N}_s\right) \\
&= \mathbb{P}\left((\|\epsilon\|^2 - d) + 2\langle x_s^u, \epsilon\rangle \leqslant 2\tilde{N}_s - \|x_s^u\|^2\right) \\
&\leqslant \mathbb{P}\left((\|\epsilon\|^2 - d) \leqslant \tilde{N}_s - \|x_s^u\|^2/2\right) + \mathbb{P}\left(2\langle x_s^u, \epsilon\rangle \leqslant \tilde{N}_s - \|x_s^u\|^2/2\right) \\
&\leqslant \mathbb{P}\left((\|\epsilon\|^2 - d) \leqslant \tilde{N}_s - \|x_s^u\|^2/2\right) + \mathbb{P}\left(\epsilon \leqslant \frac{\tilde{N}_s}{2\|x_s^u\|} - \frac{\|x_s^u\|}{4}\right) \\
&\leqslant \mathbb{P}\left((\|\epsilon\|^2 - d) \leqslant -2\sqrt{d2^s}\right) + \mathbb{P}\left(\epsilon \leqslant -\sqrt{2^{s+1}}\right) \\
&\leqslant \exp(-2^s) + \exp(-2^s)
\end{aligned}
$$

where $x_s^u = \frac{\mu_{a_u} - \mu_{a_0}}{\Delta}\sqrt{4\tilde{N}_s}$ and $\tilde{N}_s = 32 \times 2^s + 2\sqrt{d2^s}$. We used concentration inequalities from the appendix and the inequality $\|x_s^u\|^2 \geqslant 4\tilde{N}_s$. ■

Now, from the inequality $\mathbb{P}_{H_0}(\phi_s^u = 1) \leqslant \exp(-2^s)$, we have the following lemma.

Lemma 6
Recall that $r_u = \lceil \log_2\left(\log\left(\frac{u^2}{\delta}\frac{\pi^2}{6}\right)\right)\rceil$, then:

$$\mathbb{P}(\mu_{a_M} = \mu_{a_0}) \leqslant \delta \tag{8}$$

Proof
From the expression of r_u and the last lemma,

$$\mathbb{P}_{H_0}(\phi_{r_u}^u = 1) \leqslant \frac{\delta}{u^2}\frac{6}{\pi^2} \tag{9}$$

+ union bound on u ■

Lemma 7

$$\mathbb{P}(T < +\infty) = 1 \tag{10}$$

Proof
At any epoch, the probability of rejecting a good arm a_u is smaller than a constant and the probability that a_u is good is larger than m/n. ■

Remark
At this step of the proof, we have checked that the algorithm return in finite time a solution S containing arms from two different groups with probability of error smaller than δ. Then, the procedure is δ-PAC.

Lemma 8
For any $u \geqslant 1$ and $s \geqslant 2$, conditionnaly on a_u being eliminated,

$$\mathbb{P}(\tau_u \geqslant s) \leqslant \exp(-2^{s-1}) \tag{11}$$

Proof
First case under H_0^u.
We assume that $\mu_{a_u} = \mu_{a_0}$. If $\tau_u \geqslant s$ then, at the iteration $s-1$, $\phi_s^u = 1$, or $\mathbb{P}_{H_0}(\phi_{s-1}^u = 1) \leqslant \exp(-2^s)$. Then, $\mathbb{P}_{H_0^u}(\tau_u \geqslant s) \leqslant \exp(-2^{s_1})$.
Second case under H_1^u.
We assume now that $\mu_{a_u} \neq \mu_{a_0}$ and that a_u is eliminated. If $\tau_u \geqslant s$ then, the arm a_u is eliminated between iteration s and r_u. $\mathbb{P}_{H_1}(s \leqslant \tau_u \leqslant r_u) = \mathbb{P}(\exists t \in [|s; r_u|], \phi_t^u = 0) \leqslant \sum_{t \geqslant s} 2\exp\left(-2^t\right)$. Then, $\mathbb{P}_{H_0^u}(\tau_u \geqslant s) \leqslant \exp(-2^{s-1})$.

Lemma 9
Let $s \geqslant 1$ and let M_s be the number of arms rejected after s iteration, we have $M_s = \#\{u \in [|1; M-1|], \tau_u \geqslant s\}$. For any $x \in \mathbb{N}$ and if $s \geqslant 1$,

$$\mathbb{P}\left(M_s \geqslant 2^{-s+3} s(x-1) \middle| M \leqslant x\right) \leqslant \exp(-s(x-1)/2) \tag{12}$$

Proof
If $s = 1, 2, 3$, the inequality is trivial because $M_s \leqslant x - 1$. Assumr know that $s \geqslant 4$, From the expression $M_s = \sum_{u=1}^{M-1} \mathbb{1}_{\{\tau_u \geqslant s\}}$, M_s is the sum of $M-1$ independent Bernouilli's variables such that $\mathbb{P}(\tau_u \geqslant s) \leqslant \exp(-2^{s-1}) =: q_s$ and M_s is stochastically dominated by $\mathcal{B}(x-1, q_s)$ if $M \leqslant x$. From Chernoff's bound and with $\alpha \geqslant \sqrt{q_s}$, we have:

$$\begin{aligned}
\mathbb{P}\left(M_s \geqslant (1+\alpha/q_s)(x-1)q_s | M \leqslant x\right) &\leqslant \left[\frac{e^{\alpha/q_s}}{(1+\alpha/q_s)^{1+\alpha/q_s}}\right]^{(x-1)q_s} \\
&= \exp\left[(x-1)\alpha(1 - \log(1+\alpha/q_s)(1+q_s/\alpha)\right] \\
&= (1+\alpha/q_s)^{-(x_1)\alpha/2} \exp[(x-1)\alpha(1 - \log(1+\alpha/q_s)/2 - (x-1)q_s \log(1+\alpha/q_s)] \ .
\end{aligned}$$

As $\alpha \geqslant \sqrt{q_s}$ and $s \geqslant 3$, we have $\frac{\alpha}{q_s} \geqslant \frac{1}{\sqrt{q_s}} = \exp(2^{s-2}) \geqslant e^2 - 1$ and $1 - \log(1\alpha/q_s)/2 \leqslant 0$ and :

$$\begin{aligned}
\mathbb{P}\left(M_s \geqslant 2\alpha(x-1) | M \leqslant x\right) &\leqslant (1+\alpha/q_s)^{-(x_1)\alpha/2} \\
&\leqslant (1+1/\sqrt{q_s})^{-(x_1)\alpha/2} \\
&\leqslant \exp\left(-\frac{(x_1)\alpha}{2} 2^{s-2}\right) = \exp(-(x-1)\alpha 2^{s-3}) \ .
\end{aligned}$$

■

Finally, taking $\alpha = s/2^{s-2} \geqslant \exp(-2^{s-2}) = \sqrt{q_s}$, we finally have

$$\mathbb{P}\left(M_s \geqslant \frac{s}{2^{s-3}}(x-1) \middle| M \leqslant x\right) \leqslant \exp(-s(x-1)/2) \ .$$

Lemma 10
For any $x \geqslant 1$, with $c = \log(2)/2$,

$$\mathbb{P}(M \geqslant x) \leqslant 2\exp(-c\theta(x-1))) \tag{13}$$

Proof
Let $x \geqslant 1$ and let M_b be the number of good arms rejected during the procedure, $M_b := \#\{u \in [|1; M-1|]; \mu_{a_u} \neq \mu_{a_0}\}$. Conditiopnnaly on $M \geqslant x$, $M_b = \sum_{u=1}^{M-1} \mathbb{1}_{\{\mu_{a_u} \neq \mu_{a_0}\}}$ stochastically dominates $\mathcal{B}(x-1, \theta)$ where $\theta = m/n$. With Chernoff's bound and $\alpha \in (0,1)$,

$$\mathbb{P}(M_b \leqslant \theta(x-1)(1-\alpha) | M \geqslant x) \leqslant \exp(-\frac{\alpha^2(x-1)\theta}{2})$$

We recall for any arm a_u, $\mathbb{P}_{H_1^u}(\Psi_u = 0) \leqslant \sum_{s \geqslant 1} \exp(-2^s) \leqslant 1/2$ and then, by independence of the epochs, we have $\mathbb{P}(M_b \geqslant y) \leqslant \exp(-\log(2)y)$ for any $y \in \mathbb{N}$. We can now controle M,

$$\begin{aligned}
\mathbb{P}(M \geqslant x) &= \mathbb{P}(M \geqslant x, M_b \leqslant \theta(x-1)(1-\alpha)) + \mathbb{P}(M \geqslant x, M_b > \theta(x-1)(1-\alpha)) \\
&\leqslant \exp(-\alpha^2(x-1)\theta/2) + \exp(-\log(2)\theta(x-1)(1-\alpha))
\end{aligned}$$

■

Theorem 5
There exists a universal constant c such that with a probability larger than $1-\delta$ then the budget T_1 for identifying one representative by cluster with algo 4 is controlled as following:

$$T_1 \leqslant \frac{\sigma^2}{\Delta^2}\frac{1}{\theta}\sqrt{d}\log\left(1/\delta\right)\left(\log\left(\log\left(1/\theta\delta\right)\right)\right)^2 \tag{14}$$

Proof

$$\begin{aligned}
\frac{\Delta^2}{\sigma^2}T &\asymp \sum_{u=1}^{M-1}\sum_{s=1}^{\tau_u}(2^s+\sqrt{d2^s}) + K(2^{r_M}+\sqrt{d2^{r_M}}) \\
&\asymp \sum_{s=1}^{r_M}\sum_{u=1}^{M-1}\mathbb{1}_{\{\tau_u \geqslant s\}}(2^s+\sqrt{d2^s}) + K(2^{r_M}+\sqrt{d2^{r_M}}) \\
&\asymp \sum_{s=1}^{r_M} M_s(2^s+\sqrt{d2^s}) + K(2^{r_M}+\sqrt{d2^{r_M}})
\end{aligned}$$

Let $x \geqslant 1$. From ?THM? **??** and a union bound we have $\mathbb{P}\left(\exists s \in [|1; r_x|], \frac{M_s}{x-1} \geqslant 2^{-s+3}s \middle| M \leqslant x\right) \leqslant \sum_{s=1}^{r_x}\exp(-s(x-1)/2) \leqslant 2\exp(-(x-1)/2)$.

Then, we have

$$\begin{aligned}
T &\gtrsim \frac{\sigma^2}{\Delta^2}\left[\sum_{s=1}^{r_x}(x-1)s2^{-s+3}\left(2^s+\sqrt{d2^s}\right) + K\left(2^{r_x}+\sqrt{d2^{r_x}}\right)\right] \\
&\gtrsim \frac{\sigma^2}{\Delta^2}\left[xr_x^2\sqrt{d} + K\left(2^{r_x}+\sqrt{d2^{r_x}}\right)\right]
\end{aligned}$$

with a probability smaller than $3\exp(-\theta(x-1)/2)$.
Finally, by taking $x = 1 + 2/\theta\log(3/\delta)$ then with a probability smaller than δ, we have

$$T \gtrsim \frac{\sigma^2}{\Delta^2}\left[(x-1)r_x^2\sqrt{d} + K\left(2^{r_x}+\sqrt{d2^{r_x}}\right)\right] \asymp \frac{\sigma^2}{\Delta^2}\left[\frac{1}{\theta}\log\left(\frac{1}{\delta}\right)\sqrt{d}\log^2\left(\log\left(1/\theta\delta\right)\right)\right] \tag{15}$$

■

6.2 General case.

We assume $K \geqslant 2$, Δ, σ and $\theta = m/n$ known.

Algorithm 4 Sequential identification

Require: δ,σ^2,m and Δ
1: Initialize $u = 0$ and $T_0 = 8\sigma^2/\Delta^2$.
2: Take randomly $b_1 \in \mathcal{A}$ and set $k = 1$ and $S_1 = \{b_1\}$.
3: **while** $k < K$ **do**
4: Set $u = u + 1$, $r_u = \lceil \log_2\left(\log\left(\frac{u^2}{\delta}\frac{\pi^2}{6}\right)\right)\rceil$.
5: Take randomly one arm denoted by a_u.
6: **for** $s = 1, \ldots, r_u$ **do**
7: Sample $N_s = (32 \times 2^s + 2\sqrt{d2^s})T_0$ times arms a_u and compute the empirical mean and $\bar{\mu}_s^{(u)}$. For $b \in S_k$, compute $\bar{\mu}_s$
8: **if** $\|\bar{\mu}_s^{(u)} - \bar{\mu}_s^{(0)}\|^2 \leqslant \frac{\Delta^2}{2} + \frac{2\sigma^2 d}{N_s}$ **then**
9: Reject a_u.
10: **end if**
11: **end for**
12: Add a_u to S.
13: **end while**
14: **return** S

A Appendix

A.1 notation

- n: the number of arms
- $[n] = \{1, 2, \ldots, n\}$
- K: the number of clusters
- d: the dimension
- $\sigma^2 I_d$: the common covariance for the noise on the observations
- $\mathcal{A}$: the set of arms, sometimes we write $\mathcal{A} = [n]$
- $(\mathcal{P}_a)_{a\in\mathcal{A}}$: the distribution of feedback associated to the arms
- G^* is the true partition of $[n]$ into K groups. The group k for $k \in [K]$ is written G_k^* and is assumed to be nonempty.
- $k : [n] \to [K]$ is the function that encodes the true partition such the k-th cluster is given by $G_k^* = \{a \in [n]; k(a) = k\}$.
- For $k \in [K]$, $\mu(k)$ denotes the common mean of each arm in the k-th group. It is a vector in $\mathbb{R}^d$ and referred to as the center of the group k.
- For $a \in \mathcal{A}$, μ_a represents the mean of arm a, given by $\mu_a = \mu(k(a))$
- Δ: the minimal euclidean distance between two centers,

$$\Delta = \min_{k\neq k'\in[K]} \|\mu(k) - \mu(k')\|$$

- m: the size of the smallest cluster: $m = \min_{k\in[K]} |G_k^*|$.
- X_t: the t-th observation
- A_t: the t-th arm chosen
- T: the stopping time, number of samples
- δ: the fixed confidence bound for the probability of error

A.2 Concentration inequalities

Lemma 11
If $X \sim \mathcal{N}(0, 1)$ then for all $x > 0$:

$$\mathbb{P}(X \geqslant x) \leqslant \exp(-\frac{x^2}{2}).$$

Here is the inequality of Laurent and Massart, given in page 1325 of [LM00]. This is the tighter version of a concentration inequality on the chi-square distribution. The proof uses the Cramer Chernoff method.

Lemma 12
Laurent & Massart [LM00]
Let $Z \sim \chi_d^2$, where $d \geqslant 1$ is the degree of freedom, then for any $x > 0$,

$$\mathbb{P}(Z \geqslant d + 2\sqrt{dx} + 2x) \leqslant \exp(-x)$$

$$\mathbb{P}(Z \leqslant d - 2\sqrt{dx}) \leqslant \exp(-x)$$

Now, we give the inequality of Hanson-Wright in a version that is not general but enough for the use of this report and the Gaussian model. See [RV13] for the general version for sub-Gaussian variables and arbitrary matrices.

Lemma 13
Hanson-Wright
Let $\varepsilon \sim \mathcal{N}(0, I_d)$ and S a symmetrical real matrix. Then $\forall x \geqslant 0$,

$$\mathbb{P}\left(\varepsilon^T S\varepsilon - \mathrm{Tr}(S) > \sqrt{8\|S\|_F^2 x} \vee 8\|S\|_{Op} x\right) \leqslant \exp(-x)$$

Corollary 2
If $\varepsilon, \varepsilon'$ are and distributed as $\mathcal{N}(0, I_d)$ and $A \in M_n(\mathbb{R})$, then $\forall x \geqslant 0$,

$$\mathbb{P}(\varepsilon^T A\varepsilon' > \sqrt{4\|A\|_F^2 x} \vee 4|A|_{0p} x) \leqslant \exp(-x)$$

Proof
We can notice that $\varepsilon^T A\varepsilon' = \begin{bmatrix}\varepsilon \\ \varepsilon'\end{bmatrix}^T S \begin{bmatrix}\varepsilon \\ \varepsilon'\end{bmatrix}$ where the matrix $S := \frac{1}{2}\begin{bmatrix}0 & A \\ A^T & 0\end{bmatrix}$ is real and symmetrical. We can then apply the precedent lemma $|S|_{Op} = |A|_{Op}/2$ and $\|S\|_F^2 = \|A\|_F^2/2$. ■

Remark
If $A = I_d$, we get:

$$\mathbb{P}\left(\|\varepsilon\|^2 - d > \sqrt{8dx} \vee 8x\right) \leqslant \exp(-x)$$

$$\mathbb{P}(\varepsilon^T\varepsilon' > \sqrt{4dx} \vee 4x) \leqslant \exp(-x)$$

Lemma 14
Chernoff's inequality
Let $(Y_i)_{i=1}^M$ be a sequence of *iid* Bernoulli's variables $\mathcal{B}(\theta)$ with $\theta \in [0,1]$ and $M \in \mathbb{N}^*$. We denote $Y = \sum_{i=1}^M Y_i$. Then:

1. for $\alpha > 0$, $\mathbb{P}(Y > (1+\alpha)M\theta) \leqslant \left(\frac{e^\alpha}{(1+\alpha)^{1+\alpha}}\right)^{M\theta}$
2. pour $\alpha > 0$, $\mathbb{P}(Y < (1-\alpha)M\theta) \leqslant \left(\frac{e^{-\alpha}}{(1-\alpha)^{1-\alpha}}\right)^{M\theta}$
3. there exists a constant $c > 0.7$ such that for $\alpha > 1$, $\mathbb{P}(Y > (1+\alpha)M\theta) \leqslant \exp(-M\theta\alpha\log(\alpha)c)$
4. pour $0 < \alpha < 1$, $\mathbb{P}(Y < (1-\alpha)M\theta) \leqslant \exp\left(-\frac{\alpha^2 M\theta}{2}\right)$

A.3 Benchmark: supervised classification

Bayes classifier in the parametric Gaussian classification.

In this Section, we introduce the problem of Supervised Classification in a Gaussian mixture model. It is useful as a benchmark and to get some heuristics on the conditions of separation introduced in Section 3. An introduction to supervised classification can be find in the 11-th chapter of [Gir21].

In supervised learning, we get a data set $\mathcal{L}$ recording the label (or class) Y of n data points $X_1, \ldots, X_n$ and the goal is to deduce a function $x \mapsto h(x)$ that predicts the label of any new point. To compare to the problem of active clustering discussed in this report, let introduce a parametric set-up of supervised classification in the Gaussian model. Suppose that we observe a set of N labelled random variables denoted by $\mathcal{L} = (X_1, Y_1), \ldots, (X_n, Y_n)$ where the variable $(X_i, Y_i) \sim \mathbb{P}$ is composed of a label Y_i and a point X_i in $\mathbb{R}^d$. There are K groups so Y_i takes values

on $[K]$ and we denote $\theta(k)$ the proportion of point labelled by k, $\theta(k) = \mathbb{P}(Y = k)$. In the Gaussian mixture, we finally suppose that for all $k \in [K]$,

$$\mathcal{L}(X_i|Y_i = k) = \mathcal{N}(\mu(k), \Sigma)$$

with $\mu(k)$ the common mean of all variables X_i labelled by $Y_i = k$ and Σ the covariance matrix (same for all points). The goal is to find a function $h(x)$ called classifier that given a point $x \in \mathbb{R}^d$ return a label $h(x)$. A classifier $h : (\mathbb{R}^d, \mathcal{B}(\mathbb{R}^d)) \to [K]$ is a $\sigma(\mathcal{L})$-measurable function and we measure the prediction accuracy of this classifier using the probability of misclassification (or $0-1$-error) given by

$$R(f) = \mathbb{E}[\mathbb{1}_{\{h(X)\neq Y\}}]$$

where X, Y is distributed according to the probability $\mathbb{P}$ independently of $\mathcal{L}$. The classifier achieving the minimal risk is called Bayes classifier.

Lemma 15
The Bayes classifier is given by:

$$h(x) = \underset{k\in[K]}{\operatorname{argmax}}\, \mathbb{P}(Y = k|X = x) = \underset{k\in[K]}{\operatorname{argmax}} \left\langle \Sigma^{-1}\mu(k) \middle| x - \frac{\mu(k)}{2} \right\rangle + \ln(\theta(k))$$

.

Geometrically, a point x is classified according to the relative position to the set of affine hyper-planes orthogonal to $\Sigma^{-1}(\theta(k) - \theta(k'))$ with offset $\log(\pi(k)/\pi(k'))$ for all $k \neq k' \in [K]$.

This is the optimal classifier according to the probability of misclassification but it is unavailable to the learner who does not know $(\mu(k), \theta(k))_{k\in[K]}$ which parameterize the law of (X, Y).

Using a plug-in method leads to the classifier of Linear Data Analysis,

$$h_{LDA}(x) = \underset{k\in[K]}{\operatorname{argmax}} \left\langle \hat{\Sigma}^{-1}\hat{\mu}(k)|x - \frac{\hat{\mu}(k)}{2} \right\rangle + \ln(\hat{\theta}(k))$$

where $\hat{\mu}(k)$ is the empirical mean of the learning points with label k, $\hat{\theta}(k)$ is the empirical proportion of points with label k and $\hat{\Sigma}$ is the empirical covariance matrix of the data.

Binary classification

For easy explanation, we focus on the case of binary classification where $K = 2$, $Y_i \in \{-1, 1\}$. If there are two balanced group with symmetric means $\mu(\pm 1) = \pm\Delta$ then the Bayes classifier is $h(X) = \text{sign}(\langle \delta, X\rangle)$, the classification by the relative position to the hyperplane orthogonal to Δ is minimax optimal which is natural in the symmetric and balanced case.

Lemma 16
In the balanced and symmetric case $K = 2, \mu(1) = -\mu(-1), \theta(1) = -\theta(-1)$, the probability of misclassification of the Bayes classifier is:

$$\mathbb{P}(h(X) \neq Y) \leqslant \exp\left(-\frac{\|\Delta\|^2}{2\sigma^2}\right).$$

This result explain why the quantity $\frac{\Delta}{\sigma}$ is intrinsic and is called in the literature as the SNR for signal-noise ratio. Then, a first idea to measure the hardness to separate points in a given set-up can be sum-up with a so called condition of. These kind of conditions gives the minimum distance between centers that we need to be able to separate points.

From this simple result, we get a first and intrinsic separation condition which state that to be able to classify well a new point with binary supervised classification, we need $\frac{\|\Delta\|}{\sigma} \geqslant 1$.

As explained before, we do not know the mean Δ, so we do not have access to the Bayes classifier. The plug-in estimator $\hat{h}(X) = \text{sign}\left(\langle X, \hat{\Delta}\rangle\right) = \text{sign}(\langle \sum_{i=1}^{n} Y_i X_i, X\rangle)$ is not only natural but also minimax, if we add a "bayesian-type" hypothesis for Δ as in [Gir21] and [GV19].

Theorem 6
In the binary classification with two symmetrical and balanced groups with mean $\pm\Delta$, let suppose that $\Delta \sim \mathcal{U}(S)$ where $S = \mathcal{B}(0, \|\Delta\|)$. Then, the $\sigma(\mathcal{L})$-measurable classifier minimizing the probability of misclassification is:

$$\hat{h}(x) = \text{sign}\left(\langle X, \hat{\Delta}\rangle\right) = \text{sign}(\mathbb{P}(Y = 1|X = x, \mathcal{L}) - \mathbb{P}(Y = -1|X = x, \mathcal{L})).$$

Moreover,

$$\mathbb{P}(Y_{new} \neq \hat{f}(X_{new})) \asymp \exp\left(-\frac{\|\Delta\|^2}{8\sigma^2} \wedge \frac{n\|\Delta\|^4}{16d\sigma^4}\right)$$

Remark
We can deduce from this misclassification probability that if we do not know the mean Δ, then the separation condition become

$$\frac{\|\Delta\|^2}{\sigma^2} \geqslant 1 \vee \sqrt{\frac{d}{n}}.$$

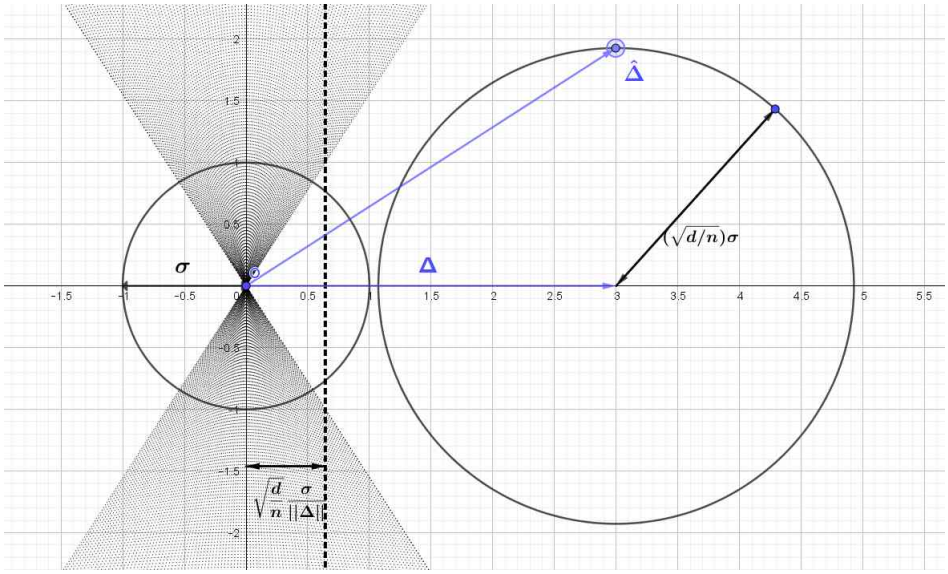

Figure 1: Heuristic on the SNR

Heuristic.

We give some heuristic to explain this separation condition, using the figure 1. The square distance $||\hat{\Delta} - \Delta||_2^2$ is distributed as $\frac{\sigma^2}{n}\chi_d^2$. With Lemma 12 in Appendix, it holds that with a probability larger than $1 - 2\delta$ we have $\left|||\hat{\Delta} - \Delta||_2^2 - \frac{d\sigma^2}{n}\right| \leqslant \frac{2\sigma^2}{n}\sqrt{d\log(1/\delta)} + \frac{2\sigma^2}{n}\log(1/\delta)$, which basically means that $||\hat{\Delta} - \Delta||_2^2$ is very concentrated around its mean $\sigma^2 d/n$ and $\hat{\Delta}$ lives on the circle of center Δ and radius $\sigma\sqrt{d/n}$. Now, $\sup_{\|x\|=1} |\langle x, \hat{\Delta} - \Delta\rangle| \leqslant \|\hat{\Delta} - \Delta\|_2 \lesssim \sqrt{d/n}\sigma$. In high dimension, the extreme case where $\hat{\Delta} - \Delta$ is orthogonal to Δ is very likely. Let $x \in \mathbb{R}^d$ such that $\|x\| = 1$ and $\langle x, \Delta\rangle > 0$, we have $\langle x, \hat{\Delta}\rangle = \langle x, \Delta\rangle + \langle x, \hat{\Delta} - \Delta\rangle \geqslant \langle x, \Delta\rangle - \sqrt{d/n}\sigma$. To have $\langle x, \hat{\Delta}\rangle \geqslant 0$ and classify the point well, we then need $\langle x, \Delta\rangle \geqslant \sqrt{d/n}\sigma$. In the figure, the zone where there is a problem is in grey. Finally, the condition $|\langle x, \Delta\rangle| \geqslant \sqrt{d/n}\sigma$ is only possible for $\|\Delta\| \geqslant \sqrt{d/n}\sigma$, with the previous condition for $h(x) = \langle x, \Delta\rangle$, $\|\Delta\| \geqslant \sigma$, we finally the following intrinsic condition:

$$\frac{\|\Delta\|}{\sigma} \geqslant 1 \vee \sqrt{\frac{d}{n}}.$$

When this condition is fulfilled, the probability of misclassification is decreasing exponentially fast with $s^2 = \frac{\Delta^2}{\sigma^2}$.

References

[ACBF02] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47:235–256, 2002.

[AOPY19] Kaito Ariu, Jungseul Ok, Alexandre Proutiere, and Se-Young Yun. Optimal clustering from noisy binary feedback. *arXiv preprint arXiv:1910.06002*, 2019.

[AV07] 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.

[BBA75] Ronald L Breiger, Scott A Boorman, and Phipps Arabie. An algorithm for clustering relational data with applications to social network analysis and comparison with multidimensional scaling. *Journal of Mathematical Psychology*, 12(3):328–383, 1975.

[CAGKR21] 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.

[CCGR97] Anil Chaturvedi, J Douglas Carroll, Paul E Green, and John A Rotondo. A feature-based approach to market segmentation via overlapping k-centroids clustering. *journal of Marketing Research*, 34(3):370–377, 1997.

[CDJ21] Emil Carlsson, Devdatt Dubhashi, and Fredrik D Johansson. Thompson sampling for bandits with clustered arms. *arXiv preprint arXiv:2109.01656*, 2021.

[dHCMC21] 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.

[Gir21] Christophe Giraud. *Introduction to high-dimensional statistics*. CRC Press, second edition edition, 2021.

[GLZ14] Claudio Gentile, Shuai Li, and Giovanni Zappella. Online clustering of bandits. In *International Conference on Machine Learning*, pages 757–765. PMLR, 2014.

[Gre77] Paul E Green. A new approach to market segmentation. *Business Horizons*, 20(1):61–73, 1977.

[GV19] Christophe Giraud and Nicolas Verzelen. Partial recovery bounds for clustering with the relaxed k-means. *Mathematical Statistics and Learning*, 1(3):317–374, 2019.

[HK00] Khaled Hammouda and Fakhreddine Karray. A comparative study of data clustering techniques. *University of Waterloo, Ontario, Canada*, 1, 2000.

[JMNB14] 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.

[JN14] 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.

[KCG16] Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On the complexity of best arm identification in multi-armed bandit models. *Journal of Machine Learning Research*, 17:1–42, 2016.

[KKS13] 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 2013. PMLR.

[KR09] Leonard Kaufman and Peter J Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons, 2009.

[LHJ+11] Zhenhui Li, Jiawei Han, Ming Ji, Lu-An Tang, Yintao Yu, Bolin Ding, Jae-Gil Lee, and Roland Kays. Movemine: Mining moving object data for discovery of animal movement patterns. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(4):1–32, 2011.

[Llo82] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

[LM00] Beatrice Laurent and Pascal Massart. Adaptive estimation of a quadratic functional by model selection. *Annals of Statistics*, pages 1302–1338, 2000.

[LS20] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[LZ16] Yu Lu and Harrison H Zhou. Statistical and computational guarantees of lloyd's algorithm and its variants. *arXiv preprint arXiv:1612.02099*, 2016.

[MC12] Fionn Murtagh and Pedro Contreras. Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(1):86–97, 2012.

[MNV12] Meena Mahajan, Prajakta Nimbhorkar, and Kasturi Varadarajan. The planar k-means problem is np-hard. *Theoretical Computer Science*, 442:13–21, 2012.

[Nda18] Mohamed Ndaoud. Sharp optimal recovery in the two-component gaussian mixture model. *arXiv preprint arXiv:1812.08078*, 2018.

[PW07] Jiming Peng and Yu Wei. Approximating k-means-type clustering via semidefinite programming. *SIAM journal on optimization*, 18(1):186–205, 2007.

[Roy17] Martin Royer. Adaptive clustering through semidefinite programming. *Advances in Neural Information Processing Systems*, 30, 2017.

[RV13] Mark Rudelson and Roman Vershynin. Hanson-wright inequality and sub-gaussian concentration, 2013.

[Vat09] Andrea Vattani. The hardness of k-means clustering in the plane. *Manuscript, accessible at http://cseweb. ucsd. edu/avattani/papers/kmeans_hardness. pdf*, 617, 2009.

[VW04] Santosh Vempala and Grant Wang. A spectral algorithm for learning mixture models. *Journal of Computer and System Sciences*, 68(4):841–860, 2004.

[WJ63] Joe H Ward Jr. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association*, 58(301):236–244, 1963.

[YZT22] Junwen Yang, Zixin Zhong, and Vincent YF Tan. Optimal clustering with bandit feedback. *arXiv preprint arXiv:2202.04294*, 2022.