Polynomial time and private learning of unbounded Gaussian Mixture Models

Polynomial time and private learning of unbounded Gaussian Mixture Models

By
JAMIL ARBAS,
B.Eng.

A THESIS
SUBMITTED TO THE DEPARTMENT OF DEPARTMENT OF COMPUTING
AND
SOFTWARE
AND THE SCHOOL OF GRADUATE STUDIES
OF MCMASTER UNIVERSITY
IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

McMaster University
Hamilton, Ontario

Master of Science (2023)
Computing and Software
McMaster University
Hamilton, Ontario, Canada

TITLE: Polynomial time and private learning of unbounded Gaussian Mixture Models

AUTHOR:
Jamil Arbas,
B.Eng. (Electronics Engineering),

SUPERVISOR:
Dr. Hassan Ashtiani
Department of Computing and Software,
McMaster University, ON, Canada

SUPERVISORY COMMITTEE MEMBERS:
Dr. Shahab Asoodeh
Department of Computing and Software,
McMaster University, ON, Canada

Dr. Ryszard Janicki
Department of Computing and Software,
McMaster University, ON, Canada

NUMBER OF PAGES: viii, 50

Declaration

I Jamil Arbas, declare that this thesis titled "Polynomial time and private learning of unbounded Gaussian Mixture Models" is my work and is based on collaboration with Christopher Liaw and Hassan Ashtiani.

Lay Abstract

In parameter estimation, we are given a random sample generated from an unknown parameterized distribution, and we are supposed to estimate the parameters of that distribution. In many cases, this random sample may consist of sensitive information belonging to individuals. This sensitive information could be leaked by parameter estimation results. Therefore, we have to estimate the parameters of that distribution privately.

In this thesis, we study parameter estimation under the assumption that the data is generated from Gaussian Mixture Models. We try to develop a sample-efficient and computationally efficient algorithm to estimate the distribution parameters privately. Moreover, to insure the privacy of the information, we consider approaches that maintain privacy.

Abstract

We develop a technique for privately estimating the parameters of a mixture distribution by reducing the problem to its non-private counterpart. This technique allows us to privatize existing non-private algorithms in a BlackBox manner while only incurring a small overhead in sample complexity and running time.

As the main application of our framework, we develop an algorithm for privately learning mixtures of Gaussians using the non-private algorithm of Moitra and Valiant [MV10] as a BlackBox and incurs only a polynomial time overhead in the sample complexity and computational complexity. As a result, this gives the first sample complexity upper bound and the first polynomial time algorithm in d for learning the parameters of the Gaussian Mixture Models privately without requiring any boundedness assumptions on the parameters.

To prove the results we introduced Private Populous Estimator (PPE) which is a generalized version of the one used in [AL22] to achieve (ϵ, δ)-differential privacy. We also develop a new masking mechanism for a single Gaussian component. Then we introduce a general recipe to turn a masking mechanism for a component into a masking mechanism for mixtures.

Acknowledgements

I would like to give my warmest thanks to my supervisor Dr. Hassan Ashtiani who made this achievement possible. I was clueless student with no experience working on theoretical research problems. He taught me the basics till we reach advanced level. He never give up on me when i made mistakes. He guided me when i lost my way, and encouraged me when i do things right. His guidance and advises through all the stages built a solid foundation to proceed in my academic career.

I would also like to thank my co-author Christopher Liaw, who rounded all sharp points and problems that we faced in this collaborations.

I would also like to thank the other students in Hassan's research group. This includes Nima, Ishaq, Alireza, Qing, Ghazal, and Mohammad. It has been a pleasure learning from each of them through our weekly research meetings.

I would also like to give special thanks to my family for their continuous support and understanding.

Finally, I would like to thank my company Collins Aerospace for their financial support during my study.

Contents

Chapter 1

Introduction

The main concern for distribution learning is to design an algorithm. If we give this algorithm an independent and identically distributed samples generated from an unknown distribution parameterized by Y, it will output an estimation of the parameters $\widetilde{Y}$ that is "close" to Y. One drawback is that this estimation could reveal sensitive information which belongs to the given samples. Therefore, differential private distribution learning algorithms are necessary.

To guarantee privacy for such estimation tasks, a widely accepted framework called differential privacy was introduced by Dwork, McSherry, Nissim, and Smith [DMNS06]. It can hide the contribution of individual data points in the output (of the estimation). Intuitively, the outputs of a differentially private algorithm on two data sets that differ by only one data point have to be "statistically indistinguishable". Compared to non-private distribution learning, differentially private distribution learning needs a larger amount of data to hide each data point's contribution. Differentially private distribution learning algorithms are evaluated based on the number of samples needed to guarantee a small error which is defined by sample complexity and based on the running time of the algorithm which is called computational complexity.

One of the most famous and widely studied statistical models is the Gaussian Mixture Model. It is used in various applications mainly in social sciences. The problem of learning a Gaussian mixture model can be either density estimation or parameter estimation. In density estimation, given a sequence of i.i.d. samples from a density f, the goal is to output a density $\widetilde{f}$ as an estimate of f. On the other hand, estimating the underlying distribution parameters is called parameter estimation. For instance, we can represent the Gaussian Mixture Model by a set of k tuples $(w_i, \mu_i, \Sigma_i)_{i=1}^{k}$, where each tuple represents the mean, covariance matrix, and mixing weight of one of its components.

As the main application of our framework, we develop an algorithm to privately learn mixtures of Gaussians using the non-private algorithm of Moitra and Valiant [MV10] as a BlackBox and incurs only a polynomial time overhead in the sample complexity and computational complexity. As a result, this gives the first sample complexity upper bound and the first polynomial time algorithm with respect to dimension d for learning the parameters of the Gaussian Mixture Models privately without requiring any boundedness assumptions on the parameters.

1.1 Learning Unbounded Gaussian Mixture Model Privately

The problem of learning the parameters of an underlying Gaussian mixture model (GMM) distribution is a fundamental problem in statistics. It dates back as early as 1894 with the work of the mathematician Karl Pearson. A GMM is a distribution where each sample is drawn from one of a collection of fixed Gaussian distributions. Given only the samples, the goal is to recover the unknown collection of Gaussians. For instance, a GMM with k components in d dimensions can be represented with $(w_i, \mu_i, \Sigma_i)_{i=1}^k$ where $w_i \in [0,1]$ and $\sum_{i\in[k]} w_i = 1$, $\mu_i \in \mathbb{R}^d$, and $\Sigma_i \in \mathbb{R}^{d\times d}$. To draw a sample from this GMM, one first sample a component by choosing index i with probability w_i and then samples from the Gaussian distribution $\mathcal{N}(\mu_i, \Sigma_i)$. Given samples from this GMM, the goal is to approximately recover all the tuples of parameters that represent the components of the GMM.

The sample complexity and polynomial-time learnability for non-private learning of GMMs is well understood in [Das99], [SK01], [VW04], [AM05], [BV08], [KMV10], [FSO06], [BS09], [BS10], [BDJKKV22], [LM21], [LM22]. However, the estimated tuple of parameters that represent the components of the GMM could reveal sensitive information which belongs to the given samples. The goal of this thesis is to design a private algorithm for learning GMMs.

In this thesis, we work in a widely accepted rigorous framework known as differential privacy which was introduced by [DMNS06]. At high-level, differential privacy provides privacy by ensuring that the contribution of any individual's data has only a small effect on the output. Therefore, differentially private distribution learning needs a larger amount of data compared to a non-private settings, because it hides individual data point contributions.

[KSSU19] considered differentially private learning of the parameters of a GMM. They guarantee the privacy in the worst case. However, they require additional assumptions to guarantee utility such as strong separation[1] or boundedness [2] of the components. It is better not to rely on such assumptions on the underlying distribution. This raises a question about the main problem that we are tackling.

> Is there a polynomial time and differentially private algorithm for learning unbounded components of GMMs?

Since the problem of learning the parameters of a GMM has been extensively studied, it is natural to ask whether there is some reduction from private learning of GMMs to non-private learning of GMMs. Our key insight is that there is. This would help to avoid separation and boundedness assumptions on the parameters of the underlying distribution. However, it will add a reasonable overhead over its sample complexity and computational complexity. This paper will answer the following question.

> Is there a polynomial time and polynomial sample overhead reduction from private to non-private learning of mixtures?

The contribution of this work is unique because we prove that there is such a reduction.

Theorem 1.1.1 (**Reduction Algorithm, Informal**). *There is a reduction from learning the parameters of a GMM in the approximate differential privacy model to its non-private counterpart. Moreover, this reduction adds only polynomial time and sample overhead in terms of the dimension and number of components.*

Our reduction algorithm with the non-private learner of GMMs [MV10], gives the first sample complexity upper bound and the first polynomial time algorithm in d for learning the parameters of the Gaussian Mixture Models privately without requiring any boundedness assumptions on the parameters.

1.2 Overview of used Techniques

Our developed reduction framework is an extension of the framework in [AL22] to the

[1] It is possible to learn the mixture components if the following separation condition is satisfied $\forall i \neq j \;\; ||\mu_i - \mu_j||_2 \geq \widetilde{\Omega}\left(\sqrt{k} + \sqrt{\frac{1}{w_i} + \frac{1}{w_j}}\right) \cdot \max\left\{||\Sigma_i^{1/2}||, ||\Sigma_j^{1/2}||\right\}$, we can see that separation condition depends on k

[2] Assuming that there are known quantities $R, \sigma_{max}, \sigma_{min}$ such that $\forall i \in [k] \;\; \|\mu_i\|_2 \leq R$ and $\sigma_{min}^2 \leq ||\Sigma_i|| \leq \sigma_{max}^2$

mixture setting. Their reduction is simple and efficient for reduction from (ε, δ) differentially private (DP) statistical estimation to its non-private counterpart. It is based on the Propose-Test-Release framework [DL09] and the Subsample-And-Aggregate framework [NRS07]. Our reduction algorithm for learning Gaussian mixture distributions has two requirements. Firstly, we need a non-private algorithm. Secondly, a masking mechanism for a single-component. At a high-level, neighbouring datasets D_1, D_2 and a masking mechanism $\mathcal{B}$ is a mechanism such that $\mathcal{B}(D_1)$, $\mathcal{B}(D_2)$ are indistinguishable provided D_1, D_2 are sufficiently close (γ close Definition 4.2.1). We showed in this thesis that we can extend single-component masking to mixture masking.

At a very high level we follow [AL22], given a dataset that we split into t subsets indexed by i. The reason is that we want to run the non-private algorithm $\mathcal{A}$ on each subset to output Y_i (implementing Subsample-And-Aggregate). We privately check if most of the outcomes are close to each other (using Propose-Test-Release). After that our framework will take a different direction from [AL22]. In [AL22], they release an estimate by noising the computed average which is not applicable for mixtures, because the non-private algorithm will output a different sequence of components every time we run it on a different subset. Therefore, it is not possible to compute the average in mixtures case using their method. Whereas in our framework, we take the solution that is close to more than 60% to other solutions. If there are multiple answers, we will break the tie by choosing the solution with the smallest index.

We also generalized the framework in [AL22] from the notion of a convex semimetric to a weaker notion of semimetric space. They needed convexity and locality properties to prove the privacy of their framework whereas we do not need such assumptions.

As mentioned earlier, our framework requires a masking mechanism. Masking mechanism development steps start with introducing a masking mechanism for one component. We noise each component parameters' separately (We add noise to the mixing weight of a single-component using the Gaussian mechanism. Also, we add noise to the mean using empirically re-scaled Gaussian mechanism. In addition, we add noise to the covariance matrix using the noising mechanism described in [AL22]. They view the input covariance matrix as a d^2 vector then apply Gaussian mechanism that is scaled to the input covariance matrix itself). The final step is to turn the produced masking mechanism for one component into a masking mechanism for mixtures. Our main idea is to add noise to each of the components and then permute the output to make the output invariant to the original order of the components. For this task, we had to use advanced composition and our proved fact that if we can mask every component in a mixture then

we can mask the whole mixture for free without any additional privacy cost as shown in Lemma 2.2.4.

Our reduction framework and the non-private algorithm in [MV10] consider the distance between the GMMs with respect to $\mathrm{dist}_{\mathrm{GMM}}$ (The TV distance between the parameters of two GMMs which is defined in Definition 2.1.2 and Definition 2.1.3). Whereas our developed masking mechanism is with respect to $\mathrm{dist}_{\mathrm{PARAM}}$ (A distance between two GMMs which is defined in Section 6.2). Thus, we had to translate $\mathrm{dist}_{\mathrm{PARAM}}$ into TV distance using the bound mentioned in [DMR18]. This translation is for a single Gaussian so we extended that bound for mixtures. As a result, if two GMMs are close in $\mathrm{dist}_{\mathrm{GMM}}$, then they will be close in $\mathrm{dist}_{\mathrm{PARAM}}$ up to a constant factor.

1.3 Summary of Contributions

In the following, we will state a summary of our contributions.

1. We generalize the framework in [AL22] from the notion of a convex semimetric to a weaker notion of semimetric space.

2. We translate the parameterized distance between GMMs into TV distance using the bound mentioned in [DMR18]. This translation is for a single Gaussian so we extended that bound for mixtures.

3. We develop a new masking mechanism for a Gaussian component. We add noise to all component parameters'. Firstly, we add noise to the mixing weight of a single-component using a Gaussian mechanism. Secondly, we add noise to the mean of a single-component using an empirically re-scaled Gaussian mechanism where the empirical covariance matrix is used to shape the noise that we add to the mean. Finally, we add noise to the covariance matrix of a single-component using the noising mechanism described in [AL22, §5].

4. We develop a general approach to extend the masking mechanism from components to mixtures. The idea is simple, we add noise to each of the components and then permute the output.

5. We develop a masking mechanism for GMMs. Intuitively, randomly shuffling the components makes the outcome of masking insensitive to the order of components in the input of the masking.

6. We introduce a general private to non-private reduction framework for learning GMMs.

7. We introduce the first sample complexity upper bound and the first polynomial time algorithm in d for learning the parameters of the Gaussian Mixture Models privately without requiring any boundedness assumptions on the parameters.

1.4 Thesis Organization

Chapter 2 provides basic preliminaries, notations, definitions, well-known results, and facts that are used in this thesis classified by topic in probability, differential privacy, TV distance of Gaussian distributions, and distribution learning.

In Chapter 3, we go over related work which is categorized into four topics, privately estimating the parameters of mixtures, private density estimation of mixtures, learning Gaussians privately and efficient algorithms for learning Gaussians.

We show our main algorithm "Private Populous Estimator" in Chapter 4. In Chapter 5, we show how to mask a single Gaussian component including noising the mixing weights, mean, and covariance matrix.

In Section 6.1 we extend the masking from component to mixture, and we apply it to Gaussian mixtures in Section 6.2.

We state our final reduction theorem to learn GMMs in Section 7.1, and we show how it can be applied in Section 7.2.

In Chapter 8, we summarize the thesis and conclude with some open problems.

Chapter 2

Background

This chapter provides basic preliminaries, notations, definitions, well-known results, and facts that are used in this thesis classified by topic in probability, differential privacy, TV distance of Gaussian distributions, and distribution learning.

2.1 Preliminaries

We use $\|v\|_2$ to denote the Euclidean norm of a vector $v \in \mathbb{R}^d$ and $\|A\|_F$ (resp. $\|A\|$) to denote the Frobenius (resp. spectral) norm of a matrix $A \in \mathbb{R}^{d\times d}$.

In this thesis, we write $\mathcal{S}^d$ to denote the positive-definite cone in $\mathbb{R}^{d\times d}$. We will often abuse terminology and refer to a distribution via its probability density function (p.d.f.).

Let $\mathcal{G}(d) = \{\mathcal{N}(\mu, \Sigma) : \mu \in \mathbb{R}^d, \Sigma \in \mathcal{S}^d\}$ be the family of d-dimensional Gaussians. We can now define the class $\mathcal{G}(d, k)$ of mixtures of Gaussians as following.

Definition 2.1.1 (Gaussian Mixtures)**.** *The class of Gaussian k-mixtures in $\mathbb{R}^d$ is*

$$\mathcal{G}(d,k) \coloneqq \left\{ \sum_{i=1}^{k} w_i G_i : G_1, \ldots, G_k \in \mathcal{G}(d), w_1, \ldots, w_k > 0, \sum_{i=1}^{k} w_i = 1 \right\}.$$

We can represent the Gaussian Mixture Model (GMM) by a set of k tuples $(w_i, \mu_i, \Sigma_i)_{i=1}^k$, where each tuple represents the mean, covariance matrix, and mixing weight of one of

its components. Note that the order of components is important in our notation, since it may affect the privacy in a subsequent analysis.

A distribution learning method is an algorithm if given a sequence of i.i.d. samples from a distribution f, outputs a distribution $\widetilde{f}$ as an estimate of f. The specific measure of "closeness" between distributions that we use is the total variation (TV) distance.

Definition 2.1.2 (Total Variation Distance)**.** *Given two probability distributions $f(x), g(x)$ on $\mathbb{R}^d$, we define the TV distance between f and g as $d_{\mathrm{TV}}(f(x), g(x)) = \frac{1}{2}\int_{\mathbb{R}^d} |f(x) - g(x)|\,\mathrm{d}x$.*

The standard way to define the distance between two GMMs is as following.

Definition 2.1.3 (The distance between two GMMs [MV10] Definition 2)**.** *The distance between two GMMs is defined by*

$$\begin{aligned}&\mathrm{dist}_{GMM}\left((w_i,\mu_i,\Sigma_i)_{i=1}^k,(w_i',\mu_i',\Sigma_i')_{i=1}^k\right)\\&\quad=\min_{\pi}\max_{i\in[k]}\max\left\{|w_i-w'_{\pi(i)}|,d_{\mathrm{TV}}\left(\mathcal{N}(\mu_i,\Sigma_i),\mathcal{N}(\mu'_{\pi(i)},\Sigma'_{\pi(i)})\right)\right\}\end{aligned}$$

where π is chosen from the set of all permutations over $[k]$.

The max in Definition 2.1.3 appears because we are looking for the furthest component, and the min is required to make the distance invariant to the ordering of the components.

If X (resp. Y) is a random variable distributed according to f (resp. g), we write $d_{\mathrm{TV}}(X,Y) = d_{\mathrm{TV}}(f,g)$. We drop the reference to the p.d.f. of the random variable when it is clear or implicity from context.

Definition 2.1.4 (Distance between the means of two multidimensional Gaussians)**.** *To calculate the maximum distance between the means of two multi-dimensional Gaussians taking into account their covariance structure, we need to use the Mahalanobis distance. As following*

$$\mathrm{dist}_{MEAN}((\mu_1,\Sigma_1),(\mu_2,\Sigma_2)) = \max\{\|\mu_1-\mu_2\|_{\Sigma_1},\|\mu_1-\mu_2\|_{\Sigma_2}\}$$

where

$$\|\mu_1-\mu_2\|_{\Sigma_2} = \sqrt{(\mu_1-\mu_2)^T\Sigma_2^{-1}(\mu_1-\mu_2)}$$

2.2 Differential Privacy

Roughly speaking, differential privacy requires a method/mechanism to have similar output distributions given any two (ordered) neighboring data sets[1].

Definition 2.2.1 (Neighbouring Datasets)**.** *Let $\mathcal{X}, \mathcal{Y}$ denote sets and $n \in \mathbb{N}$. Two datasets $D = (X_1, \ldots, X_n), D' = (X_1, \ldots, X_n) \in \mathcal{X}^n$ are said to be* neighbouring *if $d_H(D, D') \leq 1$ where d_H denotes Hamming distance, i.e., $d_H(D, D') = |\{i \in [n] : X_i \neq X'_i\}|$.*

Definition 2.2.2 ((ε, δ)-indistinguishable)**.** *Let D, D' be two distributions defined on a set $\mathcal{Y}$. Then D, D' are said to be (ε, δ)-indistinguishable if for all measurable $S \subseteq \mathcal{Y}$, we have*

$$\mathbb{P}_{Y\sim D}[Y \in S] \leq e^{\varepsilon}\mathbb{P}_{Y\sim D'}[Y \in S] + \delta \quad \textit{and} \quad \mathbb{P}_{Y\sim D'}[Y \in S] \leq e^{\varepsilon}\mathbb{P}_{Y\sim D}[Y \in S] + \delta.$$

Definition 2.2.3 ((ε, δ)-differential privacy [DKMMN06; DMNS06])**.** *A randomized mechanism $\mathcal{M} \colon \mathcal{X}^n \to \mathcal{Y}$ is said to be (ε, δ)-differentially private if for all neighbouring datasets $D, D' \in \mathcal{X}^n$, $\mathcal{M}(D)$ and $\mathcal{M}(D')$ are (ε, δ)-indistinguishable.*

In this thesis, we use DP as shorthand for differentially private or differential privacy, depending on context.

Definition 2.2.4 ([AL22] Definition 2.7)**.** *Let $\mathcal{D}_1, \mathcal{D}_2$ be two continuous distributions defined on $\mathbb{R}^d$ and let f_1, f_2 be the respective density functions. We use $\mathcal{L}_{\mathcal{D}_1\|\mathcal{D}_2} \colon \mathbb{R}^d \to \mathbb{R}$ to denote the logarithm of the likelihood ratio, i.e. for any $x \in \mathbb{R}^d$,*

$$\mathcal{L}_{\mathcal{D}_1\|\mathcal{D}_2}(x) \coloneqq \ln \frac{f_1(x)}{f_2(x)}. \tag{2.1}$$

Below definition has D, D' which are different in single individual data and function f can capture the change in magnitude at the worst case.

Definition 2.2.5 (L_1-Sensitivity [DR+14], Definition 3.1)**.** *The L_1-sensitivity of a function $f \colon \mathcal{X}^n \to \mathbb{R}^k$ is defined as:*

$$\Delta(f) = \max_{D,D' \in \mathcal{X}^n \,:\, d_H(D,D') \leq 1} ||f(D) - f(D')||_1$$

[1] For sake of simplicity, we consider data sets to be ordered and therefore the neighboring data sets are defined based on their Hamming distances. However, one can easily translate guarantees proven for the ordered setting to the unordered one; see Proposition D.6 in [BGSUZ21].

where d_H is Hamming distance identified in Definition 2.2.1

The amount of noise necessary to ensure differential privacy for a given function depends on the sensitivity of the function. In other words, we can guarantee privacy using additive noise if the sensitivity of the function is bounded. The sensitivity of a function reflects the amount the function's output will change when its input changes.

Definition 2.2.6 (L_2-Sensitivity [DR+14], Definition 3.8)**.** *The L_2-sensitivity of a function $f\colon \mathcal{X}^n \to \mathbb{R}^k$ is defined as:*

$$\Delta_2(f) = \max_{D,D' \in \mathcal{X}^n \,:\, d_H(D,D') \leq 1} ||f(D) - f(D')||_2$$

where d_H is Hamming distance identified in Definition 2.2.1

Theorem 2.2.1 (Gaussian Mechanism [DR+14], Theorem 3.22)**.** *Let $\varepsilon \in (0,1)$ be arbitrary. For $c^2 > 2ln(1.25/\delta)$, the Gaussian Mechanism with parameter $\sigma \geq c\Delta_2 f/\varepsilon$ is (ε, δ)-differentially private.*

Definition 2.2.7 (Truncated Laplace distribution)**.** *It is denoted by* $\mathrm{TLap}(\Delta, \varepsilon, \delta)$ *whose probability density function is given by*

$$f_{\mathrm{TLap}(\Delta,\varepsilon,\delta)}(x) := \begin{cases} Be^{-|x|/\lambda} & x \in [-A, A] \\ 0 & x \notin [-A, A] \end{cases},$$

where $\lambda = \frac{\Delta}{\varepsilon}$, $A = \frac{\Delta}{\varepsilon} \ln\left(1 + \frac{e^{\varepsilon}-1}{2\delta}\right)$, $B = \frac{1}{2\lambda(1-e^{-A/\lambda})}$.

Theorem 2.2.2 ([GDGK18, Theorem 1])**.** *Suppose that $q\colon \mathcal{X} \to \mathbb{R}$ is a function with L_1-sensitivity Δ. Then the mechanism $q(x) + Y$ where $Y \sim \mathrm{TLap}(\Delta, \varepsilon, \delta)$ is (ε, δ)-DP.*

Theorem 2.2.3 (Advanced Composition [DRV10])**.** *Let $\mathcal{D}_1, \ldots, \mathcal{D}_k$ and $\mathcal{D}'_1, \ldots, \mathcal{D}'_k$ be probability densities such that $\mathcal{D}_j, \mathcal{D}'_j$ are (ε, δ)-indistinguishable for all $j \in [k]$. Let $\mathcal{D} = (\mathcal{D}_1, \ldots, \mathcal{D}_k)$ and $\mathcal{D}' = (\mathcal{D}'_1, \ldots, \mathcal{D}'_k)$. Then for every $\delta' > 0$, $\mathcal{D}, \mathcal{D}'$ are $(\varepsilon', k\delta + \delta')$-indistinguishable for*

$$\varepsilon' = \sqrt{2k \ln(1/\delta')\varepsilon} + k\varepsilon(e^{\varepsilon} - 1).$$

Lemma 2.2.4. *Let $\mathcal{D}_1, \ldots, \mathcal{D}_k$ and $\mathcal{D}'_1, \ldots, \mathcal{D}'_k$ denote probability distributions on a space $\mathcal{X}$. Suppose that for all $j \in [k]$, $\mathcal{D}_j$ and $\mathcal{D}'_j$ are (ε, δ)-indistinguishable. Let $w = (w_1, \ldots, w_k)$ be a probability vector, i.e. $w_j \geq 0$ for $j \in [k]$ and $\sum_{j\in[k]} w_j = 1$. Then the probability distributions $\sum_{j\in[k]} w_j \mathcal{D}_j$ and $\sum_{j\in[k]} w_j \mathcal{D}'_j$ are (ε, δ)-indistinguishable.*

Proof. Let $\mathcal{D} = \sum_{j\in[k]} w_j \mathcal{D}_j$ and $\mathcal{D}' = \sum_{j\in[k]} w_j \mathcal{D}'_j$. Fix a set $S \subseteq \mathcal{X}$. Then

$$\mathbb{P}_{x\sim\mathcal{D}}[x \in S] = \sum_{j=1}^{k} w_j \mathbb{P}_{x\sim\mathcal{D}_j}[x \in S] \leq \sum_{j=1}^{k} w_j \left[e^{\varepsilon} \cdot \mathbb{P}_{x\in\mathcal{D}'_j}[x \in S] + \delta\right] = e^{\varepsilon}\cdot\mathbb{P}_{x\sim\mathcal{D}'}[x \in S]+\delta,$$

as required. □

Lemma 2.2.5 ([AL22] Lemma 2.10)**.** *Let $\mathcal{D}_1, \mathcal{D}_2$ be continuous distributions defined on $\mathbb{R}^d$. If*

$$\mathbb{P}_{Y\sim\mathcal{D}_1}\left[\mathcal{L}_{\mathcal{D}_1\|\mathcal{D}_2}(Y) \geq \varepsilon\right] \leq \delta \quad \text{and} \quad \mathbb{P}_{Y\sim\mathcal{D}_2}\left[\mathcal{L}_{\mathcal{D}_2\|\mathcal{D}_1}(Y) \geq \varepsilon\right] \leq \delta$$

then $\mathcal{D}_1, \mathcal{D}_2$ are (ε, δ)-indistinguishable.

2.3 Standard Probability Facts

Fact 2.3.1. *Let X_1, X_2, Y_1, Y_2 be random variables such that X_1, X_2 (resp. Y_1, Y_2) are independent. Then $d_{\mathrm{TV}}((X_1, X_2), (Y_1, Y_2)) \leq d_{\mathrm{TV}}(X_1, Y_1) + d_{\mathrm{TV}}(X_2, Y_2)$.*

Fact 2.3.2. *Let X, Y be random variables. For any function f, $d_{\mathrm{TV}}(f(X), f(Y)) \leq d_{\mathrm{TV}}(X, Y)$.*

Fact 2.3.3. *Let $\mu_1, \mu_2 \in \mathbb{R}^d$ and $\Sigma_1, \Sigma_2 \succ 0$. Then*

$$D_{\mathrm{KL}}(\mathcal{N}(\mu_1, \Sigma_1) \parallel \mathcal{N}(\mu_2, \Sigma_2)) = \frac{1}{2}[\operatorname{tr}(\Sigma_2^{-1}\Sigma_1 - I) + (\mu_2 - \mu_1)^{\top}\Sigma_2^{-1}(\mu_2 - \mu_1) - \ln\det(\Sigma_2^{-1}\Sigma_1)].$$

Moreover, suppose that all the eigenvalues of $\Sigma_2^{-1}\Sigma_1$ are at least $\frac{1}{2}$. Then

$$D_{\mathrm{KL}}(\mathcal{N}(\mu_1, \Sigma_1) \parallel \mathcal{N}(\mu_2, \Sigma_2)) \leq \frac{1}{2}[\|\Sigma_2^{-1/2}\Sigma_1\Sigma_2^{-1/2} - I\|_F^2 + (\mu_2 - \mu_1)^{\top}\Sigma_2^{-1}(\mu_2 - \mu_1)]$$

Lemma 2.3.4 (Pinsker's Inequality)**.** *Let P and Q be two distributions for which KL-divergence is defined. Then $d_{\mathrm{TV}}(P, Q) \leq \sqrt{0.5 D_{\mathrm{KL}}(P \parallel Q)}$.*

Lemma 2.3.5 ([LM00, Lemma 1])**.** *Let $g_1, \ldots, g_k$ be i.i.d. $\mathcal{N}(0, 1)$ random variables. Then*

$$\mathbb{P}\left[\sum_{i=1}^{k} g_i^2 \geq k + 2\sqrt{kt} + 2t\right] \leq e^{-t}.$$

Lemma 2.3.6 ([AL22, Lemma D.2]). *Let $\mu_1, \mu_2 \in \mathbb{R}^d$ and let Σ_1, Σ_2 be full-rank $d \times d$ PSD matrices. Let $Y \sim \mathcal{N}(\mu_1, \Sigma_1)$. Then*

$$\begin{aligned}\mathcal{L}_{\mathcal{N}(\mu_1,\Sigma_1)\|\mathcal{N}(\mu_2,\Sigma_2)}(Y) \leq\ & D_{\mathrm{KL}}\left(\mathcal{N}(\mu_1,\Sigma_1) \parallel \mathcal{N}(\mu_2,\Sigma_2)\right) \\ & + 2\|\Sigma_1^{1/2}\Sigma_2^{-1}\Sigma_1^{1/2} - I_d\|_F \cdot \sqrt{\ln(2/\delta)} + 2\|\Sigma_1^{1/2}\Sigma_2^{-1}\Sigma_1^{1/2} - I_d\| \cdot \ln(2/\delta) \\ & + \|\Sigma_1^{1/2}\Sigma_2^{-1}\Sigma_1^{1/2}\| \cdot \|\Sigma_1^{-1/2} \cdot (\mu_2 - \mu_1)\|_2 \cdot \sqrt{2\ln(2/\delta)}\end{aligned} \tag{2.2}$$

with probability at least $1 - \delta$.

Fact 2.3.7. *For $x < \ln(2)$, we have $e^x \leq 1 + 2x$*

Proof. consider the function $f(x) = 1 + 2x - e^x$. Then $f''(x) = -e^x$ so f is concave. Note that $f(0) = 0$ and $f(\ln(2)) = 1 + 2\ln(2) - 2 > 0$ so $f(x) \geq 0$ for $x \in [0, \ln(2)]$ (by concavity) □

For following lemma, approximate triangle inequality terms are defined in Definition 4.1.1.

Lemma 2.3.8. *Let $\mathcal{S}^d$ be the set of all $d \times d$ positive definite matrices. For $A, B \in \mathcal{S}^d$ let $\mathrm{dist}(A, B) = \max\{\|A^{-1/2}BA^{-1/2} - I\|, \|B^{-1/2}AB^{-1/2} - I\|\}$. Then $(\mathcal{S}^d, \mathrm{dist})$ is a semimetric space which satisfies a $(3/2)$-approximate 1-restricted triangle inequality and 1-locality.*

Proof. A stronger version for convex semimetric space was proved in [AL22, Lemma 3.2], so what was applicable for convex semimetric space will be applicable for the weaker version of semimetric space.

□

2.4 TV Distance of Gaussian Distributions

Theorem 2.4.1 ([DMR18, Theorem 1.1]). *Let $\mu \in \mathbb{R}^d$, Σ_1, $\Sigma_2 \in \mathcal{S}^d$, The total variation distance between Gaussians with the same mean is bounded by*

$$\frac{\min\left\{1, \|\Sigma_1^{-1/2}\Sigma_2\Sigma_1^{-1/2} - I_d\|_F\right\}}{100} \leq d_{\mathrm{TV}}\left(\mathcal{N}(\mu,\Sigma_1), \mathcal{N}(\mu,\Sigma_2)\right)$$

Theorem 2.4.2 ([DMR18, Theorem 1.3])**.** *The total variation distance between one-dimensional Gaussians is bounded by*

$$\frac{1}{200}\min\left\{1,\max\left\{\frac{|\sigma_1^2-\sigma_2^2|}{\sigma_1^2},\frac{40|\mu_1-\mu_2|}{\sigma_1}\right\}\right\}\leq d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1,\sigma_1^2),\mathcal{N}(\mu_2,\sigma_2^2)\right)\leq\frac{3|\sigma_1^2-\sigma_2^2|}{2\sigma_1^2}+\frac{|\mu_1-\mu_2|}{2\sigma_1}.$$

Lemma 2.4.3. $d_{\mathrm{TV}}\left(\mathcal{N}(0,\Sigma_1),\mathcal{N}(0,\Sigma_2)\right)\leq 2\cdot d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1,\Sigma_1),\mathcal{N}(\mu_2,\Sigma_2)\right)$.

Proof. Let X_1,X_2,Y_1,Y_2 be independent random variables where $X_1,X_2\sim\mathcal{N}(\mu_1,\Sigma_1)$ and $Y_1,Y_2\sim\mathcal{N}(\mu_2,\Sigma_2)$. Applying Fact 2.3.1 gives

$$\begin{aligned}d_{\mathrm{TV}}\left((X_1,X_2),(Y_1,Y_2)\right)&\leq d_{\mathrm{TV}}\left(X_1,Y_1\right)+d_{\mathrm{TV}}\left(X_2,Y_2\right)\\&\leq d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1,\Sigma_1),\mathcal{N}(\mu_2,\Sigma_2)\right)+d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1,\Sigma_1),\mathcal{N}(\mu_2,\Sigma_2)\right)\end{aligned}$$

Now, let $X=(X_1,X_2)$ and $Y=(Y_1,Y_2)$, and define the function $f(X)=f(X_1,X_2)=(X_1-X_2)/\sqrt{2}$. Then by applying Fact 2.3.2 we have

$$d_{\mathrm{TV}}\left(f(X_1,X_2),f(Y_1,Y_2)\right)\leq d_{\mathrm{TV}}\left((X_1,X_2),(Y_1,Y_2)\right)\leq 2\cdot d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1,\Sigma_1),\mathcal{N}(\mu_2,\Sigma_2)\right)$$

Note that if $X_1,X_2\sim\mathcal{N}(\mu_1,\Sigma_1)$ then $f(X)\sim\mathcal{N}(0,\Sigma_1)$. Therefore we have

$$d_{\mathrm{TV}}\left(\mathcal{N}(0,\Sigma_1),\mathcal{N}(0,\Sigma_2)\right)\leq 2\cdot d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1,\Sigma_1),\mathcal{N}(\mu_2,\Sigma_2)\right),$$

as required. □

Lemma 2.4.4. *Let $\mu\in\mathbb{R}^d$. If $d_{\mathrm{TV}}\left(\mathcal{N}(0,I_d),\mathcal{N}(\mu,I_d)\right)\leq 3\alpha<1/200$ then $\|\mu\|_2\leq 15\alpha$.*

Proof. Let $g_1\sim\mathcal{N}(0,I_d)$, $g_2\sim\mathcal{N}(\mu,I_d)$ and $v=\mu/\|\mu\|_2$. Note that $v^\top g_1\sim\mathcal{N}(0,1)$ and $v^\top g_2\sim\mathcal{N}(\|\mu\|_2,1)$. Applying Fact 2.3.2 (with $f(x)=v^\top x$) we have

$$d_{\mathrm{TV}}\left(\mathcal{N}(0,1),\mathcal{N}(\|\mu\|_2,1)\right)\leq d_{\mathrm{TV}}\left(\mathcal{N}(0,I_d),\mathcal{N}(\mu,I_d)\right)\leq 3\alpha<1/200$$

Applying Theorem 2.4.2 on the left side, we have

$$\frac{1}{200}\min\left\{1,40\|\mu\|_2\right\}\leq d_{\mathrm{TV}}\left(\mathcal{N}(0,1),\mathcal{N}(\|\mu\|_2,1)\right)\leq d_{\mathrm{TV}}\left(\mathcal{N}(0,I_d),\mathcal{N}(\mu,I_d)\right)\leq 3\alpha<1/200.$$

Note that this implies $\min\{1, 40\|\mu\|_2\} = 40\|\mu\|_2 < 1$, therefore we conclude that $\|\mu\|_2 \leq 15\alpha$. □

Lemma 2.4.5. *Let $\mu_1, \mu_2 \in \mathbb{R}^d$ and Σ_1, Σ_2 be full-rank $d \times d$ PD matrices. Suppose that*

$$d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1, \Sigma_1), \mathcal{N}(\mu_2, \Sigma_2)\right) \leq \alpha < \frac{1}{600}.$$

Then (i) $\|\Sigma_1^{-1/2}\Sigma_2\Sigma_1^{-1/2} - I_d\|_F \leq 200\alpha$ and (ii) $\|\Sigma_1^{-1}(\mu_1 - \mu_2)\|_2 \leq 15\alpha$.

Proof. (i) Starting from the assumption

$$d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1, \Sigma_1), \mathcal{N}(\mu_2, \Sigma_2)\right) \leq \alpha < \frac{1}{600},$$

we apply Lemma 2.4.3 to obtain

$$d_{\mathrm{TV}}\left(\mathcal{N}(0, \Sigma_1), \mathcal{N}(0, \Sigma_2)\right) \leq 2\alpha < \frac{1}{300}.$$

Applying Theorem 2.4.1 gives

$$\min\left\{1, \|\Sigma_1^{-1/2}\Sigma_2\Sigma_1^{-1/2} - I_d\|_F\right\} \leq 100 \cdot d_{\mathrm{TV}}\left(\mathcal{N}(0, \Sigma_1), \mathcal{N}(0, \Sigma_2)\right) \leq 200\alpha < \frac{1}{3}.$$

Note that the inequality implies that $\min\left\{1, \|\Sigma_1^{-1/2}\Sigma_2\Sigma_1^{-1/2} - I_d\|_F\right\} = \|\Sigma_1^{-1/2}\Sigma_2\Sigma_1^{-1/2} - I_d\|_F$. We conclude that

$$\|\Sigma_1^{-1/2}\Sigma_2\Sigma_1^{-1/2} - I_d\|_F \leq 200\alpha.$$

This proves the first assertion.

(ii) By the triangle inequality, we have

$$\begin{aligned} d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1, \Sigma_1), \mathcal{N}(\mu_2, \Sigma_1)\right) &\leq d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1, \Sigma_1), \mathcal{N}(\mu_2, \Sigma_2)\right) + d_{\mathrm{TV}}\left(\mathcal{N}(\mu_2, \Sigma_1), \mathcal{N}(\mu_2, \Sigma_2)\right) \\ &= d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1, \Sigma_1), \mathcal{N}(\mu_2, \Sigma_2)\right) + d_{\mathrm{TV}}\left(\mathcal{N}(0, \Sigma_1), \mathcal{N}(0, \Sigma_2)\right). \end{aligned}$$

We know that

$$d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1, \Sigma_1), \mathcal{N}(\mu_2, \Sigma_2)\right) \leq \alpha < \frac{1}{600}.$$

Also, we know that

$$d_{\mathrm{TV}}\left(\mathcal{N}(0,\Sigma_1),\mathcal{N}(0,\Sigma_2)\right) \leq 2\alpha < \frac{1}{300}.$$

We conclude that

$$d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1,\Sigma_1),\mathcal{N}(\mu_2,\Sigma_1)\right) \leq 3\alpha < \frac{1}{200}.$$

Therefore,

$$\begin{aligned} d_{\mathrm{TV}}\left(\mathcal{N}(\mu_1,\Sigma_1),\mathcal{N}(\mu_2,\Sigma_1)\right) &= d_{\mathrm{TV}}\left(\mathcal{N}(\Sigma_1^{-1}\mu_1, I_d),\mathcal{N}(\Sigma_1^{-1}\mu_2, I_d)\right) \\ &= d_{\mathrm{TV}}\left(\mathcal{N}(0, I_d),\mathcal{N}(\Sigma_1^{-1}(\mu_1-\mu_2), I_d)\right). \end{aligned}$$

Finally, applying Lemma 2.4.4 gives $\|\Sigma_1^{-1}(\mu_1-\mu_2)\|_2 \leq 15\alpha$. □

Chapter 3

Related Work

We will discuss in this chapter all related work which is categorized into four topics, privately estimating the parameters of mixtures, private density estimation of mixtures, learning Gaussians privately, and efficient algorithms for learning Gaussians.

3.1 Privately Estimating the Parameters of Mixtures

We can represent the Gaussian Mixture Model (GMM) by a set of k tuples $(w_i, \mu_i, \Sigma_i)_{i=1}^{k}$, where each tuple represents the mean, covariance matrix, and mixing weight of one of its components. Estimating these parameters is called parameter estimation.

There are a few works on designing private algorithms for estimating the parameters of Gaussian mixtures. The work of Nissim, Raskhodnikova, and Smith [NRS07] was the first differential private algorithm for learning GMMs. It is an application of the Sample-and-Aggregate framework [NRS07]. They estimate the mean of each coordinate for a uniform mixture of spherical Gaussian distributions with separable components. However, they assume the variance of each coordinate is known.

In another related work that generalized to the unknown covariance, Kamath, Sheffet, Singhal, and Ullman [KSSU19] recovered the parameters of an unknown Gaussian mixture provided that the components are sufficiently well separated. Their technique is based on Principle Component Analysis PCA (to project the data into a low dimensional space to eliminate directions that do not contain meaningful information) and then clustering. They provided an algorithm that is efficient in sample complexity and computational complexity. However, the boundedness (the domain of the samples to be bounded) condition is still assumed for their technique.

3.2 Private Density Estimation of Mixtures

Given a sequence of i.i.d. samples from a density f, outputs a density $\widetilde{f}$ as an estimate of f. The specific measure of "closeness" between distributions that we use is the total variation (TV) distance. This estimation is called density estimation.

[BSKW19] studied the problem of differentially private density estimation. Under pure ε-DP, they introduced a technique to learn a class of distributions when the class admits a finite cover with respect to TV distance, which means that the entire class of distributions can be well-approximated in TV distance by a finite number of representative distributions. They applied the technique to learn bounded spherical Gaussian mixtures where each Gaussian component has a bounded mean and covariance matrix. However, the algorithm they provide runs in time exponential in both the dimension d and the number of components k. It also cannot provide sample complexity upper bounds for distributions that do not possess a finite cover. They also studied the problem under approximate differential privacy. Instead of requiring a finite cover, it requires a locally small cover, which means that each distribution in the class is well approximated by only a small number of elements within the cover.

In another related work by Acharya, Sun, and Zhang [ASZ21], they proved a sample complexity lower bound of $\Omega(kd/\alpha^2 + kd/\alpha\varepsilon)$ under pure differential privacy for GMMs with bounded mean and identity covariance matrix with respect to total variation distance. It matches the upper bound of [BSKW19] up to logarithmic factor. In their work, they developed and used differential private version techniques of Le Cam's method, Fano's inequality, and Assouad's lemma.

In a recent work of Aden-Ali, Ashtiani, and Liaw [AAL21], they developed a technique to learn a mixture of an unbounded axis aligned Gaussians with respect to the total variation distance. They reduced the problem of learning mixtures of distributions to the problem of list-decodable learning of a single distribution. As an application they prove that $\widetilde{O}(k^2 d \log^{3/2}(1/\delta)/\alpha^2\varepsilon)$ samples are sufficient for mixture density estimation.

3.3 Learning Gaussians Privately

One of the main milestones for learning univariate Gaussians privately is Karwa and Vadhan [KV17], who established a polynomial time and sample efficient method for learning both known and unknown variance. Their pure $(\varepsilon, 0)$-DP algorithms assume that the mean μ and variance σ^2 lie in a bounded interval. Their approximate (ε, δ)-DP

differentially private algorithms do not make any assumptions on mean μ and variance σ^2 so they can remain unbounded. This method can be used to learn axis-aligned Gaussians, by applying the learning to one-dimensional projection along each axis. In another work, Kamath, Li, Singhal, and Ullman [KLSU19] considered learning multivariate Gaussians privately for both pure DP and approximate DP. They provided a polynomial time algorithm and the sample complexity for general Gaussian. They transform the Gaussian to be nearly spherical. Making it possible to apply the methods of [KV17]. However, the sample complexity still depends logarithmically on the condition number of the covariance matrix, and requires a priori bounds on the range of the parameters.

Another approach introduced by Biswas, Dong, Kamath, and Ullman [BDKU20] presented differentially private estimators for the mean and covariance of multivariate Gaussians. It is empirically more accurate for small sample sizes compared to previous estimators. The sample complexity matches [KLSU19] and depends logarithmically on the condition number of the covariance matrix.

For the multivariate general Gaussian, Aden-Ali, Ashtiani, and Kamath [AAK21] utilized the private hypothesis selection (PHS) framework proposed in [BSKW19] to prove the first sample complexity bound that does not depend on the condition number or the size of the parameters. However, this approach is theoretical and computationally inefficient.

3.4 Efficient Algorithms for Learning Gaussians

In efficient algorithms for learning Gaussians. [KMV22] proposed an efficient algorithm for robust and (ε, δ) differentially private estimation of the mean, covariance matrix, and higher moments of distributions that satisfy either a certifiable subgaussianity or certifiable hypercontractivity. Their algorithm privacy guarantees are obtained by combining stability guarantees with noise injection mechanism in which noise scales with the eigenvalues of the estimated covariance and is verified using the sum-of-squares paradigm. They applied their algorithm to obtain an efficient robust and (ε, δ)-DP algorithm for learning a Gaussian in $\mathbb{R}^d$ with sample complexity $\Omega(d^8 \ln^4(1/\delta)/\alpha^4\varepsilon^4)$ where α is the corruption parameter and the desired accuracy.

In another work, [KMSSU22] proposed a polynomial time algorithm for Gaussian estimation which requires no prior knowledge about the distribution parameters. Their algorithm technique is built on the private preconditioning framework introduced in

[KLSU19]. However, they incurred additional cost on the sample complexity of order $O(d^{5/2}/\varepsilon)$ rather than $O(d^2/\varepsilon)$ which is the best-known sample complexity of [AAK21].

In another work, [AL22] introduced a general framework for reducing (ε, δ) differentially private statistical estimation to its non-private counterpart. They give a polynomial time and (ε, δ)-DP algorithm for learning Gaussian distributions in $\mathbb{R}^d$ with sample complexity $\widetilde{O}(d^2/\alpha^2 + d^2\sqrt{\ln(1/\delta)}/\alpha\varepsilon + d\ln(1/\delta)/\alpha\varepsilon)$ which match the best theoretical sample complexity of [AAK21]. Also, they provided a polynomial time (ε, δ)-DP algorithm for robust learning of Gaussians with sample complexity $\widetilde{O}(d^{3.5})$. The aggregation part of the FriendlyCore framework was applied in [AL22] with the only a small difference being that they compute a weighted average instead of computing a random core. [TCKMS22] introduced the FriendlyCore framework which is a general framework for preprocessing the data before the differentially private aggregation step. The target is to certify "friendly" or well-behaved data so the differentially private aggregation can be executed without unfriendly points. As a result, their algorithm allows for much less noise to be added in the aggregation step.

Chapter 4

Private Populous Estimator

In this chapter, we define our main reduction framework/algorithm which we call "Private Populous Estimator(PPE)". In addition, we define the PPE theorem which establishes the privacy and utility of the framework. We also define the theorem space which is a semimetric space, and we list its characteristics. Moreover, we define in general the masking mechanism in semimetric space.

4.1 The Notion of a Semimetric Space

First, we define the notion of a semimetric space. The important thing is that we relax the triangle inequality (from a regular metric space) to an *approximate* triangle inequality which is only required to hold if the points are sufficiently close together.

Definition 4.1.1 (Semimetric Space)**.** *We say* $(\mathcal{F}, \text{dist})$ *is a semimetric space if for every* $F, F_1, F_2, F_3 \in \mathcal{F}$*, the following conditions hold.*

1. ***Non Negativity.*** $\text{dist}(F, F) = 0$ *and* $\text{dist}(F_1, F_2) \geq 0$.
2. ***Symmetry.*** $\text{dist}(F_1, F_2) = \text{dist}(F_2, F_1)$.
3. z***-approximate*** r***-restricted triangle inequality.*** *If* $\text{dist}(F_1, F_2), \text{dist}(F_2, F_3) \leq r$ *then dist*$(F_1, F_3) \leq z \cdot (\text{dist}(F_1, F_2) + \text{dist}(F_2, F_3))$.

Where $z \geq 1$ *and* $r > 0$.

4.2 Masking Mechanism According to Semimetric Space

A masking mechanism $\mathcal{B}$ is a mechanism such that $\mathcal{B}(D_1), \mathcal{B}(D_2)$ are indistinguishable (Definition 2.2.2) provided D_1, D_2 are sufficiently close (γ close Definition 4.2.1). Note

that a masking mechanism, in and of itself, is not a differentially private algorithm since it does not necessarily ensure utility. Now we define the masking mechanism according to the notion of a semimetric space.

Definition 4.2.1 (Masking mechanism)**.** *Let* $(\mathcal{F}, \text{dist})$ *be a semimetric space. A randomized function* $\mathcal{B}\colon \mathcal{F} \to \mathcal{F}$ *is a* $(\gamma, \varepsilon, \delta)$*-masking mechanism for* $(\mathcal{F}, \text{dist})$ *if for all* $F, F' \in \mathcal{F}$ *satisfying* $\text{dist}(F, F') \leq \gamma$*, we have that* $\mathcal{B}(F), \mathcal{B}(F')$ *are* (ε, δ)*-indistinguishable. Further,* $\mathcal{B}$ *is said to be* (α, β)*-concentrated if* $\mathbb{P}[\text{dist}(\mathcal{B}(F), F) > \alpha] \leq \beta$.

4.3 Private Populous Estimator Algorithm

Private Populous Estimator (PPE) is an efficient and general reduction algorithm from private parameter estimation to the non-private counterpart. We represent the non-private algorithm by $\mathcal{A}\colon \mathcal{X}^* \to \mathcal{Y}$ which takes samples from dataset D as inputs and outputs an element in $\mathcal{Y}$. PPE requires two assumptions. Firstly, We assume that $(\mathcal{Y}, \text{dist})$ is a semimetric space. Secondly, we assume that we have access to an efficient masking mechanism for $(\mathcal{Y}, \text{dist})$.

The version of PPE we introduce in this section can be seen as a somewhat generalized version of the one used in [AL22]. At a very high level we follow [AL22], given a dataset that we split into t subsets with index i. The reason is that we want to run the non-private algorithm $\mathcal{A}$ on each subset to output Y_i (implementing Subsample-And-Aggregate [DL09]). We privately check if most of the outcomes are close to each other (using Propose-Test-Release [NRS07]). In other words, we consider each point Y_j's within distance $r/2z$ from Y_i with respect to dist [1]. We calculate q_i, the fraction of Y_j's that are within distance $r/2z$. We release the final estimate that is close to more than 60% to other solutions ($q_i > 0.6$). If there are multiple answers, we will break the tie by choosing the solution with the smallest index. Compared to [AL22], they release an estimate by noising the computed average which is not applicable for mixtures, because the non-private algorithm will output a different sequence of components every time we run it on a different subset. Therefore, it is not possible to compute the average in mixtures case using their method.

To release the estimate, it is important to check the stability of the non-private algorithm $\mathcal{A}$. Making sure the outputs Y_i's are close to each other and not scattered.

[1] r and z are selected properly such that $(\mathcal{Y}, \text{dist})$ satisfies z-approximate r-restricted triangle inequality.

Therefore, we calculate the average Q of distance weights (q_i's). It should be larger than a threshold (0.8 + Truncated Laplace noise) otherwise we release a failure.

Algorithm 1 Private Populous Estimator

Input: Dataset $D = (X_1, \ldots, X_m)$, any algorithm $\mathcal{A}$: $\mathcal{X}^* \to \mathcal{Y}$, parameters $r, \varepsilon, \delta > 0, z \geq 1, t \in \mathbb{N}_{\geq 1}$.

1: Let $s \leftarrow \lfloor m/t \rfloor$.
2: For $i \in [t]$, let $Y_i \leftarrow \mathcal{A}(\{X_\ell\}_{\ell=(i-1)s+1}^{is})$.
3: For $i \in [t]$, let $q_i \leftarrow \frac{1}{t}|\{j \in [t] \,:\, \mathrm{dist}(Y_i, Y_j) \leq r/2z\}|$.
4: Let $Q \leftarrow \frac{1}{t}\sum_{i\in[t]} q_i$.
5: Let $Z \sim \mathrm{TLap}(2/t, \varepsilon, \delta)$.
6: Let $\widetilde{Q} \leftarrow Q + Z$.
7: If $\widetilde{Q} < 0.8 + \frac{2}{t\varepsilon}\ln\left(1 + \frac{e^\varepsilon - 1}{2\delta}\right)$, fail and return $\perp$.
8: $j = \min\{i \,:\, q_i > 0.6\}$.
9: Return $\widetilde{Y} = \mathcal{B}(Y_j)$.

The following theorem establishes the privacy and accuracy of Algorithm 1. The error of the algorithm is measured by $\mathrm{dist}(\widetilde{Y}, Y^*)$, where $\widetilde{Y}$ is our noisy estimation while Y^* is the truth.

Theorem 4.3.1. *Suppose that $(\mathcal{Y}, \mathrm{dist})$ satisfies a z-approximate r-restricted triangle inequality. Further, suppose that $\mathcal{B}$ is a (r, ε, δ)-masking mechanism.*

- **Privacy.** *For $t > 5$, Algorithm 1 is $(2\varepsilon, 4e^\varepsilon\delta)$-DP.*
- **Utility.** *Suppose that $\alpha \leq r/2z$, and for $t \geq (\frac{20}{\varepsilon}\ln\left(1 + \frac{e^\varepsilon - 1}{2\delta}\right)$. Let $\mathcal{B}$ be $(\alpha/2z, \beta)$-concentrated. If there exists Y^* with the property that for all $i \in [t], \mathrm{dist}(Y^*, Y_i) < \alpha/2z$, then $\mathbb{P}\left[\mathrm{dist}(\widetilde{Y}, Y^*) > \alpha\right] \leq \beta$.*

Proof of Theorem 4.3.1. **Proof of Privacy:** Let D and D' be two neighbouring datasets and let $\mathcal{A}$ denote the non-private algorithm specified in Algorithm 1. Note that the Q computed in Line 4 has sensitivity less than $\frac{2}{t}$. Since we use the Truncated Laplace mechanism in Line 7, we have (by Theorem 2.2.2)

$$\mathbb{P}\left[\mathcal{A}(D) = \perp\right] \leq e^\varepsilon \mathbb{P}\left[\mathcal{A}(D') = \perp\right] + \delta \tag{4.1}$$

We now show that for any $T \subseteq \mathcal{Y}$, we have

$$\mathbb{P}\left[\mathcal{A}(D) \in T\right] \leq e^{2\varepsilon}\mathbb{P}\left[\mathcal{A}(D') \in T\right] + 3e^\varepsilon\delta \quad \text{and} \tag{4.2}$$

$$\mathbb{P}\left[\mathcal{A}(D) \in T \cup \{\perp\}\right] \leq e^{2\varepsilon}\mathbb{P}\left[\mathcal{A}(D') \in T \cup \{\perp\}\right] + 4e^\varepsilon\delta \tag{4.3}$$

which establishes that Algorithm 1 is (ε, δ)-DP. To this end, we consider two different cases.

Case 1: $Q < 0.8$**.** In this case, $\widetilde{Q} < 0.8 + \frac{2}{t\varepsilon} \ln\left(1 + \frac{e^{\varepsilon}-1}{2\delta}\right)$ with probability 1 so $\mathbb{P}[\mathcal{A}(D) = \perp] = 1$. We now verify that Eq. (4.2) and Eq. (4.3) hold. For any $T \subseteq \mathcal{Y}$, we have $\mathbb{P}[\mathcal{A}(D) \in T] = 0$ so Eq. (4.2) is trivially satisfied. To check Eq. (4.3) holds, we apply Eq. (4.1) to see that

$$\mathbb{P}[\mathcal{A}(D) \in T \cup \{\perp\}] = \mathbb{P}[\mathcal{A}(D) = \perp] \leq e^{\varepsilon}\mathbb{P}[\mathcal{A}(D') = \perp] + \delta \leq e^{\varepsilon}\mathbb{P}[\mathcal{A}(D') \in T \cup \{\perp\}] + \delta.$$

Case 2: $Q \geq 0.8$**.** Let $Y_1, \ldots, Y_t$ and $Y'_1, \ldots, Y'_t$ be the outputs in Line 2 assuming the dataset is D, D', respectively. Let j, j' be the output of Line 8 assuming the dataset is D, D', respectively. Next, we show that $\mathrm{dist}(Y_j, Y'_j) \leq r$.

Let $S = \{\ell \in [t] : \mathrm{dist}(Y_j, Y_\ell) \leq r/2z\}$ and $S' = \{\ell \in [t] : \mathrm{dist}(Y'_j, Y'_\ell) \leq r/2z\}$. We know that $|S| > 0.6t$ and $|S'| > 0.6t$ (by definition of j in Line 8). By the inclusion-exclusion principle, we have $|S \cap S'| = |S| + |S'| - |S \cup S'| > 0.6t + 0.6t - t = 0.2t$. Thus, if $t \geq 5$, we have $|S \cap S'| > 1$ and since $|S \cap S'|$ is an integer, we must have $|S \cap S'| \geq 2$. Since D, D' differ only in a single datapoint, there is some $\ell \in S \cap S'$ such that $Y_\ell = Y'_\ell$. Thus, we conclude that

$$\mathrm{dist}(Y_j, Y'_j) \leq \mathrm{dist}(Y_j, Y_\ell) + \mathrm{dist}(Y_\ell, Y'_j) \leq z \cdot (r/2z + r/2z) = r,$$

where in the final inequality, we used that dist is a z-approximate r-restricted triangle inequality and that $\mathrm{dist}(Y_j, Y_\ell), \mathrm{dist}(Y_\ell, Y'_j) \leq r$.

We are now ready to verify that Eq. (4.2) and Eq. (4.3) hold. Let $\mathcal{M}$ denote the mechanism described in Algorithm 1. Fix any $T \subseteq \mathcal{Y}$. Then we have

$$\begin{aligned}
\mathbb{P}[\mathcal{M}(D) \in T] &= \mathbb{P}[\mathcal{M}(D) \neq \perp]\,\mathbb{P}[\mathcal{B}(Y_j) \in T] \\
&\leq (e^{\varepsilon}\mathbb{P}[\mathcal{M}(D') \neq \perp] + \delta)(e^{\varepsilon}\mathbb{P}\left[\mathcal{B}(Y'_{j'}) \in T\right] + \delta) \\
&= (e^{2\varepsilon}\mathbb{P}[\mathcal{M}(D') \neq \perp]\,\mathbb{P}\left[\mathcal{B}(Y'_{j'}) \in T\right] + 2e^{\varepsilon}\delta + \delta^2 \\
&\leq e^{2\varepsilon}\mathbb{P}[\mathcal{M}(D) \in T] + 3e^{\varepsilon}\delta
\end{aligned}$$

where in first inequality we used the fact that $\mathcal{B}$ is a (r, ε, δ)-masking mechanism, which satisfies Eq. (4.2). Next, we also have

$$\begin{aligned}\mathbb{P}\left[\mathcal{M}(D) \in \{\perp\} \cup T\right] &= \mathbb{P}\left[\mathcal{M}(D) = \perp\right] + \mathbb{P}\left[\mathcal{M}(D) \in T\right] \\ &\leq e^{\varepsilon}\mathbb{P}\left[\mathcal{M}(D') = \perp\right] + \delta + e^{2\varepsilon}\mathbb{P}\left[\mathcal{M}(D') \in T\right] + 3e^{\varepsilon}\delta \\ &\leq e^{2\varepsilon}\mathbb{P}\left[\mathcal{M}(D') \in \{\perp\} \cup T\right] + 4e^{\varepsilon}\delta.\end{aligned}$$

This completes the proof.

Proof of Utility.

We divide the proof into two parts.

1. First, we show that $\widetilde{Y}$ (the noisy output) concentrates around Y^*.
2. Second, we show that Algorithm 1 does not fail in Line 7.

For the first part, We know that $\mathcal{B}$ is $(\frac{\alpha}{2z}, \beta)$ concentrated. Furthermore, $\forall i \in [t]$, Y_i satisfies $\text{dist}(Y^*, Y_i) < \frac{\alpha}{2z}$. we have

$$\begin{aligned}&\mathbb{P}[\text{dist}(\widetilde{Y}, Y^*) > \frac{\alpha}{2} + \frac{\alpha}{2}] \leq \\ &\mathbb{P}[z.\,\text{dist}(\widetilde{Y}, Y_j) + z.\,\text{dist}(Y_j, Y^*) > \frac{\alpha}{2} + \frac{\alpha}{2}] \leq \\ &\mathbb{P}[\text{dist}(\widetilde{Y}, Y_j) + \text{dist}(Y_j, F^*) > \frac{\alpha}{2z} + \frac{\alpha}{2z}] \leq \\ &\mathbb{P}[\text{dist}(\widetilde{Y}, Y_j) > \frac{\alpha}{2z}] + \mathbb{P}[\text{dist}(Y_j, Y^*) > \frac{\alpha}{2z}] \leq \\ &\beta + 0 \leq \beta\end{aligned}$$

where the first inequality follows from the r-restricted z-approximate triangle inequality 3 (since $\alpha/2z < r/4z^2$ by assumption), and the first part of the last inequality follows the concentration of the masking mechanism. We get $\mathbb{P}[\text{dist}(\widetilde{Y}, Y_j) > \frac{\alpha}{2z}] = \beta$, because $\widetilde{Y}$ is just a masked version of Y_j. Also $\mathbb{P}[\text{dist}(Y_j, Y^*) > \frac{\alpha}{2z}] = 0$, because Y_j is selected from Y_i's, and none of them located in a distance larger than $\frac{\alpha}{2z}$ from Y^* based on our assumption.

For the second part, we start by guaranteeing that Q in Line 4 equals to 1. For that we need to ensure that for all $i, j \in [t]$, $\text{dist}(Y_i, Y_j) \leq \frac{r}{2z}$. To see this by triangle

inequality, we have

$$\text{dist}(Y_i, Y_j) \leq z.(\text{dist}(Y_i, Y^*) + \text{dist}(Y^*, Y_j))$$

since $\text{dist}(Y_i, Y^*), \text{dist}(Y^*, Y_j) \leq \frac{r}{4z^2}$. So we conclude that $Q = 1$.

Now we need to show that $\widetilde{Q} \leq 0.9$. From Line 6 $\widetilde{Q} = Q + Z$. Therefore it is enough to show $|Z| \leq 0.1$. We know that from Definition 2.2.7 $|Z| \leq \frac{2}{t\varepsilon} \ln\left(1 + \frac{e^{\varepsilon}-1}{2\delta}\right)$. By the assumption that $t \geq \frac{20}{\varepsilon} \ln\left(1 + \frac{e^{\varepsilon}-1}{2\delta}\right)$ we conclude that $\widetilde{Q} \leq 0.9$, so the Algorithm 1 does not fail in Line 7.

□

Remark 4.3.2. *Note that Algorithm 1 is a poly-time reduction from non-private estimation to private estimation. In particular, let $T_{\mathcal{A}}$ be the running time of the algorithm $\mathcal{A}$ in Line 2, T_{dist} be the time to compute $\text{dist}(Y_i, Y_j)$ for any $Y_i, Y_j \in \mathcal{Y}$ in Line 3, and $T_{\mathcal{B}}$ be the time to compute $\widetilde{Y}$ in Line 9. Then Algorithm 1 runs in time $O(t \cdot T_{\mathcal{A}} + t^2 \cdot T_{\text{dist}} + T_{\mathcal{B}})$.*

Remark 4.3.3. *$L1$-sensitivity for Algorithm 1 is bounded by $\frac{2}{t}$*

Proof. Consider q_i and q_i' computed on D and D' as per line 3 in Algorithm 1 while recalling the following:

$$q_i = \frac{1}{t}|\{j \in [t] \,:\, \text{dist}(Y_i, Y_j) \leq r/2z\}|$$

and assume one point has been changed in first subset so

$$||q_1 - q_1'||_1 \leq \frac{1}{t} \times (t-1) \leq 1$$

and for other points $\forall i \neq 1$

$$||q_i - q_i'||_1 \leq \frac{1}{t}$$

so

$$||Q - Q'||_1 = ||\frac{1}{t}\sum Q_i - Q_i'||_1 \leq \frac{(t-1)}{t^2} + \frac{1}{t} \leq \frac{2}{t}$$

so the $L1$-sensitivity for Algorithm 1 is bounded by $\frac{2}{t}$ □

To apply Algorithm 1 for private learning of GMMs, we need to introduce a masking mechanism for them. In order to do that, we start by defining a masking mechanism

for a single Gaussian component (presented in Section 5). We then show how one can convert a masking mechanism for a component to one for mixtures (Section 6.1). Finally, we apply this to come up with a masking mechanism for GMMs as shown in Section 6.2.

Chapter 5

Masking a Single Gaussian Component

In a high level, a masking mechanism $\mathcal{B}$ is a mechanism such that $\mathcal{B}(D_1)$, $\mathcal{B}(D_2)$ are indistinguishable (Definition 2.2.2) provided D_1, D_2 are sufficiently close. In this chapter, we develop a masking mechanism for a single Gaussian component $\mathcal{F}_{\text{COMP}}$. To do that we introduced a distance $\text{dist}_{\text{COMP}}$ which is the maximum distance between all component parameters; weight w, mean μ, and covariance matrix Σ. Also, we make sure that $\text{dist}_{\text{COMP}}$ satisfies the z-approximate r-restricted triangle inequality.

Let $\mathcal{F}_{\text{COMP}} = \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^{d \times d}$ (corresponding to the weight w, mean μ, and covariance matrix Σ, respectively). Define $\text{dist}_{\text{COMP}} \colon \mathcal{F}_{\text{COMP}} \times \mathcal{F}_{\text{COMP}} \to \mathbb{R}_{\geq 0}$ as

$$\begin{aligned} &\text{dist}_{\text{COMP}}((w_1, \mu_1, \Sigma_1),(w_2, \mu_2, \Sigma_2)) \\ &\qquad = \max\{|w_1 - w_2|, \text{dist}_{\text{MEAN}}((\mu_1, \Sigma_1), (\mu_2, \Sigma_2)), \text{dist}_{\text{COV}}(\Sigma_1, \Sigma_2)\}, \end{aligned} \tag{5.1}$$

where

$$\begin{aligned} \text{dist}_{\text{COV}}(\Sigma_1, \Sigma_2) &= \max\{\|\Sigma_1^{1/2}\Sigma_2^{-1}\Sigma_1^{1/2} - I_d\|_F, \|\Sigma_2^{1/2}\Sigma_1^{-1}\Sigma_2^{1/2} - I_d\|_F\} \\ \text{dist}_{\text{MEAN}}((\mu_1, \Sigma_1), (\mu_2, \Sigma_2)) &= \max\{\|\mu_1 - \mu_2\|_{\Sigma_1}, \|\mu_1 - \mu_2\|_{\Sigma_2}\}. \end{aligned}$$

Lemma 5.0.1. $\text{dist}_{\text{COMP}}$ *satisfies a* 1*-restricted* $(3/2)$*-approximate triangle inequality.*

Proof. The absolute value in $\|\mu_1 - \mu_2\|_{\Sigma_1}$ and $\|\mu_1 - \mu_2\|_{\Sigma_2}$ satisfies triangle inequality. Also, dist_{COV} satisfies triangle inequality according to Lemma 2.3.8.

□

Lemma 5.0.2. *For $\gamma \leq \frac{\varepsilon\alpha}{C_2\sqrt{d(d+\ln(4/\beta))\cdot\ln(2/\delta)}}$, there exists a $(\gamma, 3\varepsilon, 3\delta)$-masking mechanism for $(\mathcal{F}_{\text{COMP}}, \text{dist}_{\text{COMP}})$ that is $(\alpha, 3\beta)$-concentrated, where C_2 is a universal constant.*

The rest of this chapter is dedicated to proving Lemma 5.0.2. In particular, we will introduce the masking mechanism $\mathcal{B}_{\text{COMP}}(w, \mu, \Sigma)$ that satisfies the conditions of Lemma 5.0.2. In order to add noise to a Gaussian component (w_i, μ_i, Σ_i) we perform a number of steps:

1. In Subsection 5.1, we discuss how to noise the mixing weight of a single-component. This is the most straightforward as we can simply use the Gaussian mechanism.

2. In Subsection 5.2, we discuss how to noise the mean of a single-component. To do this, we use an empirically re-scaled Gaussian mechanism where the empirical covariance matrix is used to shape the noise that we add to the mean. This is somewhat similar to the empirically re-scaled Gaussian mechanism used by [BGSUZ21].

3. In Subsection 5.3, we discuss how to noise the covariance matrix of a single-component. To do this, we use the noising mechanism described in [AL22, §5].

5.1 Noising the Mixing Weights

In this section, we prove that the mechanism $\mathcal{R}_{\text{W}}(w, \eta) = w + \eta g$ where $g \sim \mathcal{N}(0, 1)$ and $w, \eta \in \mathbb{R}$ can privatize the weights, we simply use the Gaussian mechanism.

Lemma 5.1.1. *Let $\alpha, \beta, \delta > 0$, $\eta = \frac{\alpha}{\sqrt{2+2\ln(1/\beta)}}$, and $\gamma \leq \frac{\alpha\varepsilon}{2\sqrt{2}\ln(2/\delta)\sqrt{1+\ln(1/\beta)}}$.*

1. *Let $w_1, w_2 \in \mathbb{R}$. If $|w_1 - w_2| \leq \gamma$ then $\mathcal{R}_W(w_1, \eta)$ and $\mathcal{R}_W(w_2, \eta)$ are (ε, δ)-indistinguishable.*

2. *Let $w \in \mathbb{R}$. Then $|\mathcal{R}_W(w, \eta) - w| \leq \alpha$ with probability at least $1 - \beta$.*

Proof. The first item is simply the guarantee of the Gaussian Mechanism Theorem 2.2.1 when substituting $\Delta_2 f, \sigma$ with γ, η respectively. The second item follows from standard tail bounds on a Gaussian random variable (e.g., Lemma 2.3.5). □

5.2 Noising the Mean

In this section, we prove that the mechanism $\mathcal{R}_{\text{MEAN}}(\mu, \Sigma, \eta) = \mu + \eta g$ where $g \sim \mathcal{N}(0, \Sigma)$ effectively privatizes the mean.

Lemma 5.2.1. *Let* $\alpha, \beta, \delta > 0$, $\eta = \sqrt{\frac{\alpha^2}{3(d+\ln(1/\beta))}}$ *and let* $\gamma \leq \min\{\frac{1}{2}, \frac{\varepsilon\alpha}{24\ln(2/\delta)\sqrt{d+\ln(1/\beta)}}\}$. *Let* $\mu_1, \mu_2 \in \mathbb{R}^d$ *and let* Σ_1, Σ_2 *be* $d \times d$ *positive-definite matrices. Suppose that*

1. $\max\{\|\Sigma_1^{1/2}\Sigma_2^{-1}\Sigma_1^{1/2} - I_d\|_F, \|\Sigma_2^{1/2}\Sigma_1^{-1}\Sigma_2^{1/2} - I_d\|_F\} \leq \gamma$; *and*
2. $\max\{\|\mu_1 - \mu_2\|_{\Sigma_1}, \|\mu_1 - \mu_2\|_{\Sigma_2}\} \leq \gamma$.

Then $\mathcal{R}_{\textsc{Mean}}(\mu_1, \Sigma_1, \eta)$ *and* $\mathcal{R}_{\textsc{Mean}}(\mu_2, \Sigma_2, \eta)$ *are* (ε, δ)*-indistinguishable. In addition, if we let* $\widetilde{\mu} = \mathcal{R}_{\textsc{Mean}}(\mu, \Sigma, \eta)$ *then* $\|\widetilde{\mu} - \mu\|_\Sigma \leq \alpha$ *with probability at least* $1 - \beta$.

First, we prove a bound on the privacy loss.

Lemma 5.2.2. *Let* $\eta > 0$ *and* $\gamma \in (0, 1/2]$. *Let* $\mu_1, \mu_2 \in \mathbb{R}^d$ *and let* Σ_1, Σ_2 *be* $d \times d$ *positive-definite matrices. Suppose that*

1. $\max\{\|\Sigma_1^{1/2}\Sigma_2^{-1}\Sigma_1^{1/2} - I_d\|_F, \|\Sigma_2^{1/2}\Sigma_1^{-1}\Sigma_2^{1/2} - I_d\|_F\} \leq \gamma$; *and*
2. $\max\{\|\mu_1 - \mu_2\|_{\Sigma_1}, \|\mu_1 - \mu_2\|_{\Sigma_2}\} \leq \gamma$.

Let $Y \sim \mathcal{N}(\mu_1, \eta^2\Sigma_1)$ *and define* $\mathcal{L} \coloneqq \mathcal{L}_{\mathcal{N}(\mu_1,\eta^2\Sigma_1)\|\mathcal{N}(\mu_2,\eta^2\Sigma_2)}(Y)$. *Then*

$$\mathcal{L} \leq \frac{\gamma^2}{2} + \frac{\gamma^2}{2\eta^2} + 2\gamma\sqrt{\ln(2/\delta)} + 2\gamma\ln(2/\delta) + 2\gamma\sqrt{2\ln(2/\delta)}/\eta \tag{5.2}$$

with probability at least $1 - \delta$.

Proof. We directly utilize Lemma 2.3.6 and bound each term in Eq. (2.2). For the first term, we have, using Fact 2.3.3 and that the eigenvalues of $\Sigma_2^{-1}\Sigma_1$ are at least $1/2$ by assumption (since $\gamma < 1/2$), we have[1]

$$\begin{aligned} D_{\mathrm{KL}}\left(\mathcal{N}(\mu_1, \eta\Sigma_1) \parallel \mathcal{N}(\mu_2, \eta\Sigma_2)\right) &\leq \frac{1}{2}\left[\|\Sigma_2^{-1/2}\Sigma_1\Sigma_2^{-1/2} - I_d\|_F^2 + (\mu_2 - \mu_1)^\top(\eta^2\Sigma_2)^{-1}(\mu_2 - \mu_1)\right] \\ &\leq \frac{1}{2}\left[\gamma^2 + \frac{\gamma^2}{\eta^2}\right] \end{aligned}$$

The second term in Eq. (2.2) is bounded by $2\gamma\sqrt{\ln(2/\delta)}$. The third term in Eq. (2.2) is bounded by $2\gamma\ln(2/\delta)$. Finally, the fourth term in Eq. (2.2) is bounded by $(1 + \gamma)\frac{\gamma}{\eta}\sqrt{2\ln(2/\delta)}$. □

The next lemma shows that $\mathcal{R}_{\textsc{Mean}}(\mu, \Sigma, \eta)$ concentrates tightly around μ w.r.t. Mahalanobis distance.

[1] Note that we use that $\Sigma_2^{-1}\Sigma_1, \Sigma_2^{-1/2}\Sigma_1\Sigma_2^{-1/2}, \Sigma_2^{1/2}\Sigma_1^{-1}\Sigma_2^{1/2}$ all have the same spectrum.

Lemma 5.2.3. *Let* $\widetilde{\mu} = \mathcal{R}_{\text{MEAN}}(\mu, \Sigma, \eta)$. *Then* $\mathbb{P}\left[\|\widetilde{\mu} - \mu\|_\Sigma^2 \geq 3\eta^2(d + \ln(1/\beta))\right] \leq \beta$.

Proof. Recall that $\widetilde{\mu} = \mu + \eta\Sigma^{1/2}g$ where $g \sim \mathcal{N}(0, I_d)$. Thus, $\|\widetilde{\mu} - \mu\|_\Sigma^2 = \eta^2\|g\|_2^2$. Applying Lemma 2.3.5 gives that

$$\begin{aligned}\mathbb{P}\left[\|\widetilde{\mu} - \mu\|_\Sigma^2 \geq 3\eta^2(d + \ln(1/\beta))\right] &= \mathbb{P}\left[\|g\|_2^2 \geq 3(d + \ln(1/\beta))\right] \\ &\leq \mathbb{P}\left[\|g\|_2^2 \geq d + 2\sqrt{d\ln(1/\beta)} + 2\ln(1/\beta)\right] \\ &\leq 1/\beta,\end{aligned}$$

where in the first inequality, we used that $2\sqrt{d\ln(1/\beta)} \leq d + \ln(1/\beta)$. □

Proof of Lemma 5.2.1. Note that

$$\gamma \leq \frac{\varepsilon\alpha}{24\ln(2/\delta)\sqrt{d + \ln(1/\beta)}} \leq \min\left\{\sqrt{\frac{\varepsilon}{2}}, \sqrt{\frac{\varepsilon\alpha^2}{6(d + \ln(1/\beta))}}, \frac{\varepsilon}{8\ln(2/\delta)}, \frac{\varepsilon\alpha}{24\sqrt{\ln(2/\delta)(d + \ln(1/\beta))}}\right\}$$

so the first claim follows by Lemma 2.2.5 and plugging γ and η into Lemma 5.2.2 to make each term in Eq. (5.2) is at most $\varepsilon/4$. Accuracy follows from Lemma 5.2.3 using our choice of η. □

5.3 Noising the Covariance Matrix

In this section, we prove that the mechanism $\mathcal{R}_{\text{Cov}}(\Sigma, \eta) = \Sigma^{1/2}(I_d + \eta G)(I_d + \eta G)^\top\Sigma^{1/2}$ where $G \in \mathbb{R}^{d\times d}$ is a matrix with independent $\mathcal{N}(0, 1)$ entries can privatizes the covariance matrix.

Define $\mathcal{R}_{\text{Cov}}(\Sigma, \eta) = \Sigma^{1/2}(I_d + \eta G)(I_d + \eta G)^\top\Sigma^{1/2}$ where $G \in \mathbb{R}^{d\times d}$ is a matrix with independent $\mathcal{N}(0, 1)$ entries. We require the following lemma which is paraphrased from Lemma 5.1 and Lemma 5.2 in [AL22].

Lemma 5.3.1 ([AL22, Lemma 5.1, Lemma 5.2])**.** *There are absolute constant* $C_1, C_2 > 0$ *such that the following holds. Let* $\varepsilon, \delta, \beta \in (0, 1]$ *and set* $\eta = \frac{\alpha}{C_1(\sqrt{d} + \sqrt{\ln(4/\beta)})}$.

- *Suppose that* $\gamma \leq \frac{\varepsilon\alpha}{C_2\sqrt{d(d+\ln(4/\beta))\cdot\ln(2/\delta)}}$. *If* Σ_1, Σ_2 *are positive-definite* $d \times d$ *matrices such that*

$$\max\{\|\Sigma_1^{1/2}\Sigma_2^{-1}\Sigma_1^{1/2} - I_d\|_F, \|\Sigma_2^{1/2}\Sigma_1^{-1}\Sigma_2^{1/2} - I_d\|_F\} \leq \gamma$$

then $\mathcal{R}_{Cov}(\Sigma_1, \eta)$ *and* $\mathcal{R}_{Cov}(\Sigma_2, \eta)$ *are* (ε, δ)*-indistinguishable.*

- *Let* $\widetilde{\Sigma} = \mathcal{R}_{Cov}(\Sigma, \eta)$*. Then*

$$\max\left\{\|\Sigma^{-1/2}\widetilde{\Sigma}\Sigma^{-1/2} - I_d\|_F, \|\widetilde{\Sigma}^{-1/2}\Sigma\widetilde{\Sigma}^{-1/2} - I_d\|_F\right\} \leq \alpha$$

with probability at least $1 - \beta$*.*

To prove Lemma 5.3.1, we require the following two lemmas from [AL22]. Note that Lemma 5.3.3 is slightly different than what is stated in [AL22] but follows easily from the proof.

Lemma 5.3.2 ([AL22, Lemma 5.1])**.** *Let* $d \in \mathbb{N}, \eta > 0, \varepsilon \in (0, 1], \delta \in (0, 1], \gamma > 0$ *and suppose that*

$$\gamma \leq \min\left\{\sqrt{\frac{\varepsilon}{2d(d + 1/\eta^2)}}, \frac{\varepsilon}{8d\sqrt{\ln(2/\delta)}}, \frac{\varepsilon}{8\ln(2/\delta)}, \frac{\varepsilon\eta}{12\sqrt{d}\sqrt{\ln(2/\delta)}}\right\}. \tag{5.3}$$

Let Σ_1, Σ_2 *be two positive-definite* $d \times d$ *matrices. Suppose that*

$$\max\{\|\Sigma_1^{1/2}\Sigma_2^{-1}\Sigma_1^{1/2} - I_d\|_F, \|\Sigma_2^{1/2}\Sigma_1^{-1}\Sigma_2^{1/2} - I_d\|_F\} \leq \gamma.$$

Define $\mathcal{R}_{Cov}(\Sigma, \eta) = \Sigma^{1/2}(I + \eta G)(I + \eta G)^\top \Sigma^{1/2}$ *where* $G \sim \mathbb{R}^{d\times d}$ *is a matrix with independent* $\mathcal{N}(0, 1)$ *entries. Then* $\mathcal{R}_{Cov}(\Sigma_1, \eta)$ *and* $\mathcal{R}_{Cov}(\Sigma_2, \eta)$ *are* (ε, δ)*-indistinguishable.*

Lemma 5.3.3 ([AL22, Lemma 5.2])**.** *There is a sufficiently large constant* $C > 0$ *such that the following holds. Let* $\beta > 0$ *and* Σ *be a positive-definite* $d \times d$ *matrix and set* $\eta = \frac{\alpha}{C(\sqrt{d}+\sqrt{\ln(4/\beta)})}$*. If* $\widetilde{\Sigma} = \mathcal{R}_{Cov}(\Sigma, \eta)$ *then*

$$\max\left\{\|\Sigma^{-1/2}\widetilde{\Sigma}\Sigma^{-1/2} - I_d\|_F, \|\widetilde{\Sigma}^{-1/2}\Sigma\widetilde{\Sigma}^{-1/2} - I_d\|_F\right\} \leq \alpha$$

with probability at least $1 - \beta$*.*

Proof of Lemma 5.3.1. For the first assertion, it suffices to show that the inequality in Eq. (5.3) holds. Since

$$\gamma \leq \frac{\varepsilon\alpha}{C_2\sqrt{d(d + \ln(4/\beta))}\ln(2/\delta)}, \tag{5.4}$$

it is clear that γ is bounded above by the second and third terms of Eq. (5.3) provided C_2 is sufficiently large. Next, we prove that γ is bounded above by the first term in

Eq. (5.3). Indeed, we have

$$\eta^2 = \frac{\alpha^2}{C_1^2(\sqrt{d}+\sqrt{\ln(4/\beta)})^2} \geq \frac{\alpha^2}{2C_1^2(d+\ln(4/\beta))},$$

where in the last inequality we used the fact that $(a+b)^2 \leq 2a^2+2b^2$ for any real numbers a, b. Plugging this bound of η^2 into Eq. (5.3) and some calculations give that

$$\sqrt{\frac{\varepsilon}{2d(d+1/\eta^2)}} \geq \sqrt{\frac{\varepsilon\alpha^2}{C_3 d(d+\ln(4/\beta))}}, \tag{5.5}$$

for some constant $C_3 > 0$. Thus, if C_2 is large enough then the right side of Eq. (5.4) is upper bounded by the right side of Eq. (5.5).

Finally, it is straightforward to check that γ is at most the last term in Eq. (5.4) by plugging in the value of η. □

5.4 Masking a Single Gaussian Component

Now we use the previous three subsections to devise a masking mechanism for masking a single-component. Let $\eta_{\mathrm{W}} = \frac{\alpha}{\sqrt{2+2\ln(1/\beta)}}$, $\eta_{\mathrm{MEAN}} = \frac{\alpha}{\sqrt{3(d+\ln(1/\beta))}}$ and $\eta_{\mathrm{COV}} = \frac{\alpha}{C_1(\sqrt{d}+\sqrt{\ln(4/\beta)}}$ for a sufficiently large constant C_1. Consider the mechanism

$$\mathcal{B}_{\mathrm{COMP}}(w,\mu,\Sigma) = (\mathcal{R}_{\mathrm{W}}(w,\eta_{\mathrm{W}}), \mathcal{R}_{\mathrm{MEAN}}(\mu,\Sigma,\eta_{\mathrm{MEAN}}), \mathcal{R}_{\mathrm{COV}}(\Sigma,\eta_{\mathrm{COV}})) \tag{5.6}$$

Proof. of Lemma 5.0.2 The fact that $\mathcal{B}_{\mathrm{COMP}}$ is a $(\gamma, 3\varepsilon, 3\delta)$-masking follow from Lemma 5.1.1, Lemma 5.2.1, and Lemma 5.3.1 along with basic composition. That $\mathcal{B}_{\mathrm{COMP}}$ is $(\alpha, 3\beta)$-concentrated also follow from Lemma 5.1.1, Lemma 5.2.1, and Lemma 5.3.1 along with a union bound. □

We can conclude this section by stating that we develop a masking mechanism for a single Gaussian component, which is a combination of masking all component parameters; weight w, mean μ, and covariance matrix Σ. In the next chapter, we will turn this masking mechanism into a masking mechanism for mixtures.

Chapter 6

Turning a Masking Mechanism for a Component to a Masking Mechanism for Mixtures

The goal of this chapter is to show how to "lift" a masking mechanism for a single-component to a masking mechanism for mixtures. Then we show how to mask a mixture of k Gaussians.

6.1 A General Recipe

The goal of this section is to show how to "lift" a masking mechanism for a single-component to a masking mechanism for mixtures. The idea is simple: we add noise to each of the components and randomly permute the output components.

Formally, let $\mathcal{F}$ denote a space and let $\mathcal{F}^k = \mathcal{F} \times \ldots \times \mathcal{F}$ (k times). The following definition is useful in defining the distance between two mixtures, as it is invariant to the order of components.

Definition 6.1.1. *Let* dist *denote a distance function on* $\mathcal{F}$*. The distance function* $\mathrm{dist}^k \colon \mathcal{F}^k \times \mathcal{F}^k \to \mathbb{R}_{\geq 0}$ *is defined by*

$$\mathrm{dist}^k((F_1, \ldots, F_k), (F'_1, \ldots, F'_k)) \coloneqq \min_{\pi} \max_{i \in [k]} \mathrm{dist}(F_i, F'_{\pi(i)}),$$

where the minimization is taken over all permutations π*.*

The following definition is useful for extending a masking mechanism for a component to a masking mechanism for a mixture. The important thing is that the components are

shuffled randomly, therefore that the outcome is invariant to the original order of the components.

Definition 6.1.2. *Suppose that $\mathcal{B}$ is a $(\gamma, \varepsilon, \delta)$-masking mechanism for $\mathcal{F}$, and the mechanism $\mathcal{B}^k$ as $(\mathcal{B}_1, \ldots, \mathcal{B}_k)$, then the mechanism $\mathcal{B}^k_\sigma$ is defined by $\mathcal{B}^k_\sigma(F_1, \ldots, F_k) = (\mathcal{B}(F_{\sigma(1)}), \ldots, \mathcal{B}(F_{\sigma(k)}))$, where σ is one selected uniformly random permutation.*

Suppose that $\mathcal{B}$ is an (α, β)-concentrated $(\gamma, \varepsilon, \delta)$-masking mechanism for $\mathcal{F}$. The next lemma shows that $\mathcal{B}^k_\sigma$ is indeed a masking mechanism w.r.t. $(\mathcal{F}^k, \mathrm{dist}^k)$.

Lemma 6.1.1. *If $\mathcal{B}$ is an (α, β)-concentrated $(\gamma, \varepsilon, \delta)$-masking mechanism for $(\mathcal{F}, \mathrm{dist})$ then, for any $\delta' > 0$, $\mathcal{B}^k_\sigma$ is an $(\alpha, k\beta)$-concentrated $(\gamma, \varepsilon', k\delta + \delta')$-masking mechanism for $(\mathcal{F}^k, \mathrm{dist}^k)$ where*

$$\varepsilon' = \sqrt{2k \ln(1/\delta')}\varepsilon + k\varepsilon(e^\varepsilon - 1).$$

Proof. First, we prove privacy. Let $F = (F_1, \ldots, F_k) \in \mathcal{F}^k$ and $F' = (F'_1, \ldots, F'_k) \in \mathcal{F}_k$ be such that $\mathrm{dist}^k(F, F') \leq \gamma$. In other words, there exists a permutation π such that $\mathrm{dist}(F_i, F'_{\pi(i)}) \leq \gamma$ for all $i \in [k]$. Since $\mathcal{B}$ is a $(\gamma, \varepsilon, \delta)$-masking mechanism, we now that $\mathcal{B}(F_i), \mathcal{B}(F'_{\pi(i)})$ are (ε, δ)-indistinguishable. Thus, by advanced composition (Theorem 2.2.3), $(\mathcal{B}(F_1), \ldots, \mathcal{B}(F_k))$ and $(\mathcal{B}(F'_{\pi(1)}), \ldots, \mathcal{B}(F'_{\pi(k)}))$ are $(\varepsilon', k\delta + \delta')$-indistinguishable with ε' as stated in the lemma. Since $\mathcal{B}^k_\sigma((F'_1, \ldots, F'_k))$ has the same distribution has $\mathcal{B}^k_\sigma((F'_{\pi(1)}, \ldots, F'_{\pi(k)}))$, we conclude, using the fact that permutation preserves privacy (see Lemma 2.2.4), that $\mathcal{B}^k_\sigma(F)$ and $\mathcal{B}^k_\sigma(F')$ are $(\varepsilon', k\delta + \delta')$-indistinguishable.

Finally, it remains to prove accuracy (i.e. that $\mathcal{B}^k_\sigma$ is $(\alpha, k\beta)$-concentrated). Indeed, given $F = (F_1, \ldots, F_k) \in \mathcal{F}^k$, we know that $\mathrm{dist}(\mathcal{B}(F_i), F_i) \leq \alpha$ with probability at least $1 - \beta$. Thus, by a union bound $\mathrm{dist}(\mathcal{B}(F_i), F_i) \leq \alpha$ for all $i \in [k]$ with probability at least $1 - k\beta$. We conclude that $\mathrm{dist}(\mathcal{B}(F), F) \leq \alpha$ with probability at least $1 - k\beta$. □

Recall that Theorem 4.3.1 requires that the distance function satisfies an r-restricted z-approximate. The following lemma shows that dist^k indeed does satisfy this property provided that dist does.

Lemma 6.1.2. *If* dist *satisfies an r-restricted z-approximate triangle inequality then so does* dist^k.

Proof. Let $F, F', F'' \in \mathcal{F}^k$. We need to show that if $\mathrm{dist}^k(F, F') \leq r$ and $\mathrm{dist}^k(F', F'') \leq r$ then $\mathrm{dist}^k(F, F'') \leq z \cdot (\mathrm{dist}^k(F, F') + \mathrm{dist}^k(F', F''))$. To that end, let $\pi_1^* \in \arg\min_\pi \max_{i \in [k]} (F_i, F'_{\pi(i)})$

and let $\pi_2^* \in \arg\min_\pi \max_{\in[k]}(F'_{\pi_1^*(i)}, F''_{\pi(i)})$. Since dist satisfies r-restricted z-approximate triangle inequality and for any i, $\text{dist}(F_i, F'_{\pi_1^*(i)}), \text{dist}(F'_{\pi_1^*(i)}, F''_{\pi_2^*(i)}) \leq r$, we have

$$\begin{aligned}\text{dist}(F_i, F''_{\pi_2^*(i)}) &\leq z \cdot \left(\text{dist}(F_i, F'_{\pi_1^*(i)}) + \text{dist}(F'_{\pi_1^*(i)}, F''_{\pi_2^*(i)})\right) \\ &\leq z \cdot \left(\text{dist}^k(F, F') + \text{dist}^k(F', F'')\right).\end{aligned}$$

In particular

$$\text{dist}^k(F, F'') \leq \max_{i\in[k]} \text{dist}(F_i, F''_{\pi_2^*(i)}) \leq z \cdot \left(\text{dist}^k(F, F') + \text{dist}^k(F', F'')\right), \tag{6.1}$$

as required. □

The following two lemmas show that both $\mathcal{B}_\sigma^k$ and dist^k can be computed with polynomial (in k) overhead.

Lemma 6.1.3. *If $\mathcal{B}_{\text{SINGLE}}$ is the running time of $\mathcal{B}$ then $\mathcal{B}_\sigma^k$ can be computed in time $T_{\mathcal{B}} = O(k \cdot \mathcal{B}_{\text{SINGLE}} + k \log k)$.*

Proof. Computing $\mathcal{B}_\sigma^k$ only requires computing $\mathcal{B}_{\text{SINGLE}}$ a total of k times and finding permutation. The former takes time $O(k \cdot \mathcal{B}_{\text{SINGLE}})$ and the latter takes time $O(k \log k)$ (say by sampling k uniform random numbers in $[0, 1]$ and then sorting). □

Lemma 6.1.4. *If $\text{dist}_{\text{SINGLE}}$ is the running time to compute* dist *then dist^k can be computed in time $T_{\text{dist}} = O(k^2 \, \text{dist}_{\text{SINGLE}} + k^3 \log k)$.*

Proof. The plan is to reduce the problem of computing dist^k to binary search and checking if a bipartite graph has a perfect matching.

First, we compute $\text{dist}(F_i, F_j)$ for every $i, j \in [k]$. This takes time $k^2 \, \text{dist}_{\text{SINGLE}}$. Note that

$$\text{dist}^k((F_1, \ldots, F_k), (F'_1, \ldots, F'_k))$$

must be one of these k^2 values. In addition, observe that we can determine if

$$\text{dist}^k((F_1, \ldots, F_k), (F'_1, \ldots, F'_k)) \leq x$$

for any number x by consider the following bipartite graph. The disjoint node sets are $\{F_1, \ldots, F_k\}$ and $\{F'_1, \ldots, F'_k\}$ and there is an edge between F_i, F'_j if and only if $\text{dist}(F_i, F'_j) \leq x$. We then determine if there is a complete bipartite matching on

this graph, which takes time at most $O(k^3)$ (e.g. by using the Hungarian algorithm). Thus, we can simply combine this with a binary search on the sorted values given by $\{\text{dist}(F_i, F'_j)\}_{i,j'}$ to compute dist^k. □

6.2 A Masking Mechanism for GMMs

In this section, we show how to mask a mixture of k Gaussians. Let $\mathcal{F}_{\text{GMM}} = \mathcal{F}_{\text{COMP}} \times \ldots \times \mathcal{F}_{\text{COMP}}$ (k times). Note we drop k from $\mathcal{F}_{\text{GMM}}$ (and related notation below) since k is fixed and implied from context. Let $\text{dist}_{\text{COMP}}$ be as defined in Eq. (5.1) and define the distance

$$\text{dist}_{\text{PARAM}}(\{(w_i, \mu_i, \Sigma_i)\}_{i\in[k]}, \{(w'_i, \mu'_i, \Sigma'_i)\}_{i\in[k]}) = \min_{\pi} \max_{i\in[k]} \text{dist}_{\text{COMP}}((w_{\pi(i)}, \mu_{\pi(i)}, \Sigma_{\pi(i)}), (w'_i, \mu'_i, \Sigma'_i)).$$

where π is chosen from the set of all permutations over $[k]$. Now define the masking mechanism

$$\mathcal{B}_{\text{GMM}}(\{(w_i, \mu_i, \Sigma_i)\}_{i\in[k]}) = \{\mathcal{B}_{\text{COMP}}(w_{\sigma(i)}, \mu_{\sigma(i)}, \Sigma_{\sigma(i)})\}_{i\in[k]}$$

where $\mathcal{B}_{\text{COMP}}$ is defined in Eq. 5.6, and σ is a permutation *chosen uniformly at random* from the set of all permutations over $[k]$. Intuitively, randomly shuffling the components makes the outcome of masking insensitive to the order of components in the input of the masking.

Now we define the main Lemma for masking mechanism for GMMs.

Lemma 6.2.1. *Let $\varepsilon < \ln(2)/3$. There is a sufficiently large constant C_2 such that for $\gamma \leq \frac{\varepsilon\alpha}{C_2\sqrt{k\ln(2/\delta)}\sqrt{d(d+\ln(12k/\beta))\cdot\ln(12k/\delta)}}$, $\mathcal{B}_{GMM}$ is a $(\gamma, \varepsilon, \delta)$-masking mechanism with respect to $(\mathcal{F}_{GMM}, \text{dist}_{PARAM})$. Moreover, $\mathcal{B}_{GMM}$ is (α, β)-concentrated.*

Proof. Applying Lemma 6.1.1 for masking mixtures (with ε, δ in Lemma 6.1.1 replaced by $3\varepsilon, 3\delta$, respectively), we have, for every $\delta' > 0$, that $\mathcal{B}_{\text{GMM}}$ is a $(\gamma, \varepsilon', 3k\delta + \delta')$-masking mechanism where

$$\varepsilon' = 3\sqrt{2k\ln(1/\delta')}\varepsilon + 3k\varepsilon(e^{3\varepsilon} - 1).$$

a $(\gamma, 3\sqrt{2k\ln(1/\delta')}\varepsilon + 3k\varepsilon(e^{3\varepsilon} - 1), 3k\delta + \delta')$-masking mechanism. As this is true for any δ', we can take $\delta' = 3k\delta$ and applying the numeric inequality $e^x \leq 1 + 2x$, valid for $x < \ln(2)$ (see Fact 2.3.7) to get that

$$\varepsilon' \leq 3\sqrt{2k\ln(1/3k\delta)}\varepsilon + 18k\varepsilon^2,$$

Finally, to prove the accuracy part ($\mathcal{B}_{\text{GMM}}$ is $(\alpha, 3k\beta)$-concentrated), we apply the accuracy part of Lemma 6.1.1 for masking mixtures which was proved by union bound for all $i \in [k]$. Also defining the distance to be the maximum between all three component parameters; weight w, mean μ, and covariance matrix Σ.

$$\text{dist}_{\text{PARAM}}(\{(w_i, \mu_i, \Sigma_i)\}_{i\in[k]}, \{(w'_i, \mu'_i, \Sigma'_i)\}_{i\in[k]}) = \min_{\pi} \max_{i\in[k]} \text{dist}_{\text{COMP}}((w_{\pi(i)}, \mu_{\pi(i)}, \Sigma_{\pi(i)}), (w'_i, \mu'_i, \Sigma'_i)).$$

We can conclude that

$$\text{dist}_{\text{PARAM}}(\mathcal{B}_{\text{GMM}}(\{(w_i, \mu_i, \Sigma_i)\}_{i\in[k]}), \{(w_i, \mu_i, \Sigma_i)\}_{i\in[k]}) \leq \alpha$$

with probability at least $1 - 3k\beta$.

Now we have $\mathcal{B}_{\text{GMM}}$ is a $(\gamma, 3\sqrt{2k\ln(1/3k\delta)}\varepsilon + 18k\varepsilon^2, 6k\delta)$-masking mechanism with respect to $(\mathcal{F}_{\text{GMM}}, \text{dist}_{\text{PARAM}})$. Moreover, $\mathcal{B}_{\text{GMM}}$ is $(\alpha, 3k\beta)$-concentrated.

To simplify it, let $\varepsilon' < \ln(2)/3, \delta' < 1, \alpha' < 1, \beta' < 1$ be parameters. We set $\delta = \delta'/6k$, $\beta = \beta'/3k$, $\alpha = \alpha'$ and $\varepsilon = \min\left\{\frac{\varepsilon'}{6\sqrt{2k\ln(1/3k\delta)}}, \sqrt{\frac{\varepsilon'}{36k}}\right\} \geq \frac{\varepsilon'}{\sqrt{72k\ln(2/\delta')}}$. Then for sufficiently large constant C such that if $\gamma \leq \frac{\varepsilon'\alpha'}{C_2\sqrt{k\ln(2/\delta')}\sqrt{d(d+\ln(12k/\beta'))\cdot\ln(12k/\delta')}}$, $\mathcal{B}_{\text{GMM}}$ is a $(\gamma, \varepsilon', \delta')$-masking mechanism that is (α', β')-concentrated. This proves the claim.

□

We can use the masking mechanism $\mathcal{B}_{\text{GMM}}$ in Lemma 6.2.1, and apply it into PPE Algorithm 1 saying that $\mathcal{B}_{\text{GMM}}(Y_1)$, $\mathcal{B}_{\text{GMM}}(Y_2)$ are indistinguishable (Definition 2.2.2) provided Y_1, Y_2 (the output of non-private algorithm $\mathcal{A}$) are γ close with respect to $\text{dist}_{\text{PARAM}}$.

The $\text{dist}_{\text{PARAM}}$ in Lemma 6.2.1 has to satisfy approximate triangle inequality. The following lemma shows that $\text{dist}_{\text{PARAM}}$ indeed does satisfy this property provided that $\text{dist}_{\text{COMP}}$ and dist^k does.

Lemma 6.2.2. dist_{PARAM} *satisfies a* 1*-restricted* $(3/2)$*-approximate triangle inequality.*

Proof. Lemma 5.0.1 implies that $\text{dist}_{\text{COMP}}$ satisfies 1-restricted $(3/2)$-approximate triangle inequality. Therfore, applying Lemma 6.1.2 $\text{dist}_{\text{PARAM}}$ satisfies 1-restricted $(3/2)$-approximate triangle inequality. □

Algorithm 2 GMM Masking Mechanism

Input: GMM defined by $Y_j = (\mathcal{F}_{\text{COMP}}(w_1, \mu_1, \Sigma_1), \ldots, \mathcal{F}_{\text{COMP}}(w_k, \mu_k, \Sigma_k))$; Parameters $w \in (0, 1)$

function $\mathcal{R}_{\text{W}}(w, \eta_{\text{W}})$
  Let $g \sim \mathcal{N}(0, 1)$
  $\eta_{\text{W}} \leftarrow \frac{\alpha}{\sqrt{2+2\ln(1/\beta)}}$ ▷ Lemma 5.1.1
  Return $w + \eta_{\text{W}} g$
function $\mathcal{R}_{\text{MEAN}}(\mu, \Sigma, \eta_{\text{MEAN}})$
  Let $g \sim \mathcal{N}(0, \Sigma)$
  $\eta_{\text{MEAN}} \leftarrow \frac{\alpha}{\sqrt{3(d+\ln(1/\beta))}}$ ▷ Lemma 5.2.1
  Return $\mu + \eta_{\text{MEAN}} g$
function $\mathcal{R}_{\text{COV}}(\Sigma, \eta_{\text{COV}})$
  Let $G \in \mathbb{R}^{d \times d}$ matrix with independent $\mathcal{N}(0, 1)$ entries; c_1 is a sufficiently large constant
  $\eta_{\text{COV}} \leftarrow \frac{\alpha}{C_1(\sqrt{d}+\sqrt{\ln(4/\beta)})}$ ▷ Lemma 5.3.1
  Return $\Sigma^{1/2}(I_d + \eta_{\text{COV}} G)(I_d + \eta_{\text{COV}} G)^\top \Sigma^{1/2}$
function $\mathcal{B}_{\text{COMP}}(w, \mu, \Sigma)$
  Return $(\mathcal{R}_{\text{W}}(w, \eta_{\text{W}}), \mathcal{R}_{\text{MEAN}}(\mu, \Sigma, \eta_{\text{MEAN}}), \mathcal{R}_{\text{COV}}(\Sigma, \eta_{\text{COV}}))$
function $\mathcal{B}_{\text{GMM}}(\{(w_i, \mu_i, \Sigma_i)\}_{i \in [k]})$
  Return $\{\mathcal{B}_{\text{COMP}}(w_{\sigma(i)}, \mu_{\sigma(i)}, \Sigma_{\sigma(i)})\}_{i \in [k]}$ ▷ where σ is a uniformly random permutation

Chapter 7

Private to Non-Private Reduction for Learning GMMs and Applications

In this chapter, we show and prove our main result which is a general theorem to reduce the learning GMMs parameters' from private to its non-private counterpart. After that, we apply it to a non-private algorithm. We specifically pick [MV10] non-private algorithm. As a result, we introduce the first sample complexity upper bound and the first polynomial time algorithm in d for learning the parameters of the Gaussian Mixture Models privately without requiring any boundedness assumptions on the parameters.

7.1 Private to Non-Private Reduction for Learning GMMs

In this section, we show and prove our main result which is a general theorem to reduce the learning GMMs parameters' from private to its non-private counterpart. This theorem allows us to privatize existing non-private algorithms in a BlackBox manner while only incurring a small overhead in sample complexity and running time.

Before we present the private to non-private reduction for learning GMMs. We need to remark that our developed masking mechanism Lemma 6.2.1 considers the distance between the GMMs with respect to $\text{dist}_{\text{PARAM}}$. Whereas our reduction framework and the non-private algorithm in [MV10] consider the distance between the GMMs with respect to dist_{GMM}. Thus, we had to translate $\text{dist}_{\text{PARAM}}$ into TV distance using the bound mentioned in [DMR18] as we show in Lemma 7.1.1. This translation is for a single Gaussian so we extended that bound for mixtures as we show in Lemma 7.1.2. As

a result, if two GMMs are close in dist_{GMM}, then they will be close in $\text{dist}_{\text{PARAM}}$ up to a constant factor.

First, we need to bound the total variation (TV) distance between Gaussians as follows.

Lemma 7.1.1. *Let $\mu_1, \mu_2 \in \mathbb{R}^d$ and Σ_1, Σ_2 be full-rank $d \times d$ PD matrices. Suppose that $d_{\text{TV}}(\mathcal{N}(\mu_1, \Sigma_1), \mathcal{N}(\mu_2, \Sigma_2)) < \frac{1}{600}$. Let*

$$\Delta = \max\left\{\|\Sigma_1^{-1/2}\Sigma_2\Sigma_1^{-1/2} - I_d\|_F, \|\Sigma_1^{-1}(\mu_1 - \mu_2)\|_2\right\}$$

Then

$$\frac{1}{200}\Delta \leq d_{\text{TV}}(\mathcal{N}(\mu_1, \Sigma_1), \mathcal{N}(\mu_2, \Sigma_2)) \leq \frac{1}{\sqrt{2}}\Delta$$

Proof. The lower bound follows from Lemma 2.4.5.

Now we prove the upper bound. By Lemma 2.4.5(i) the eigenvalues of $\Sigma_2^{-1}\Sigma_1$ are strictly larger than $1/2$. Therefore, using Fact 2.3.3 we know that

$$D_{\text{KL}}(\mathcal{N}(\mu_1, \Sigma_1) \parallel \mathcal{N}(\mu_2, \Sigma_2)) \leq \frac{1}{2}[\|\Sigma_2^{-1/2}\Sigma_1\Sigma_2^{-1/2} - I\|_F^2 + (\mu_2 - \mu_1)^\top\Sigma_2^{-1}(\mu_2 - \mu_1)].$$

Using Pinsker's inequality (Lemma 2.3.4) we have

$$d_{\text{TV}}(\mathcal{N}(\mu_1, \Sigma_1), \mathcal{N}(\mu_2, \Sigma_2)) \leq \frac{1}{2}\sqrt{[\|\Sigma_2^{-1/2}\Sigma_1\Sigma_2^{-1/2} - I\|_F^2 + (\mu_2 - \mu_1)^\top\Sigma_2^{-1}(\mu_2 - \mu_1)]} \leq \frac{\Delta}{\sqrt{2}}$$

which concludes the proof. □

Next step is to bound the total variation (TV) distance between GMMs with same number of components as following.

Lemma 7.1.2. *Let $F = (w_i, \mu_i, \Sigma_i)_{i=1}^k$ and $F' = (w_i', \mu_i', \Sigma_i')_{i=1}^k$ be two d-dimensional GMMs where Σ_i and Σ_i' are PD matrices. Suppose that $\text{dist}_{GMM}(F, F') < \frac{1}{600}$. Then*

$$\frac{1}{200}\,\text{dist}_{PARAM}(F, F') \leq \text{dist}_{GMM}(F, F') \leq \frac{1}{\sqrt{2}}\,\text{dist}_{PARAM}(F, F')$$

Proof. Recall from Definition 2.1.3 that dist_{GMM} is defined as $\text{dist}_{\text{GMM}}\left((w_i, \mu_i, \Sigma_i)_{i=1}^k, (w_i', \mu_i', \Sigma_i')_{i=1}^k\right)$

$$= \min_{\pi}\max_{i\in[k]}\max\left\{|w_i - w'_{\pi(i)}|, d_{\text{TV}}\left(\mathcal{N}(\mu_i, \Sigma_i), \mathcal{N}(\mu'_{\pi(i)}, \Sigma'_{\pi(i)})\right)\right\} \tag{7.1}$$

where π is chosen from the set of all permutations over $[k]$.

Mapping Lemma 7.1.1 for every component $i \in k$ for F, F' will keep the same inequalities, so that

$$\frac{1}{200}\Delta_k \leq \min_{\pi} \max_{i\in[k]} \max \left\{ |w_i - w'_{\pi(i)}|, d_{\mathrm{TV}} \left(\mathcal{N}(\mu_i, \Sigma_i), \mathcal{N}(\mu'_{\pi(i)}, \Sigma'_{\pi(i)}) \right) \right\} \leq \frac{1}{\sqrt{2}}\Delta_k \quad (7.2)$$

where

$$\Delta_k = \min_{\pi} \max_{i\in[k]} \max\{|w_i - w'_{\pi(i)}|, \mathrm{dist}_{\textsc{Mean}}((\mu_i, \Sigma_i), (\mu'_{\pi(i)}, \Sigma'_{\pi(i)})), \mathrm{dist}_{\textsc{Cov}}(\Sigma_i, \Sigma'_{\pi(i)})\}$$

which makes Δ_k has the same definition of $\mathrm{dist}_{\textsc{Param}}(F, F')$ in Section 6.2 which defined by

$$\mathrm{dist}_{\textsc{Param}}(F, F') = \min_{\pi} \max_{i\in[k]} \mathrm{dist}_{\textsc{Comp}}((w_{\pi(i)}, \mu_{\pi(i)}, \Sigma_{\pi(i)}), (w'_i, \mu'_i, \Sigma'_i)).$$

where

$$\mathrm{dist}_{\textsc{Comp}}((w_i, \mu_i, \Sigma_i),(w'_i, \mu'_i, \Sigma'_i)) = \max\{|w_i - w'_i|, \mathrm{dist}_{\textsc{Mean}}((\mu_i, \Sigma_i), (\mu'_i, \Sigma'_i)), \mathrm{dist}_{\textsc{Cov}}(\Sigma_i, \Sigma'_i)\}$$

So $\Delta_k = \mathrm{dist}_{\textsc{Param}}(F, F')$, applying that in Equation 7.2, we will have

$$\frac{1}{200}\mathrm{dist}_{\textsc{Param}}(F, F') \leq \mathrm{dist}_{\textsc{Gmm}}(F, F') \leq \frac{1}{\sqrt{2}}\mathrm{dist}_{\textsc{Param}}(F, F')$$

which conclude the proof □

Now we define the PAC learning of parameters of GMMs.

Definition 7.1.1 (PAC Learning of Parameters of GMMs)**.** *Let* $\mathcal{F} = \left\{ \left(w_i^j, \mu_i^j, \Sigma_i^j \right)_{i=1}^{k} \right\}^j$ *be any class of d-dimensional GMMs with k components*[1]*. Let* $\mathcal{A}$ *be function that receives a sequences S of instances in* $\mathbb{R}^d$ *and outputs a mixture* $\hat{F} = (\hat{w}_i, \hat{\mu}_i, \hat{\Sigma}_i)_{i=1}^k$*. Let* $m : (0,1)^2 \to \mathbb{N}$*. We say* $\mathcal{A}$ *learns* $\mathcal{F}$ *with m samples if for every* $\alpha, \beta \in (0,1)$ *and every*

[1]For examples, it is standard to pick $\mathcal{F}$ to be those GMMs that are separable/identifiable.

$F \in \mathcal{F}$, if S is an i.i.d. sample of size $m(\alpha, \beta)$ from F, then $\text{dist}_{GMM}(F, \hat{F}) < \alpha$ *with probability at least* $1 - \beta$.

Finally we introduce our general theorem to reduce the learning of GMMs parameters' from private to its non-private counterpart.

Theorem 7.1.3 (Private to Non-Private Reduction)**.** *Let $\mathcal{F}$ be any subclass of GMMs with k components in $\mathbb{R}^d$. Let $\mathcal{A}$ be a non-private Algorithm that PAC learns $\mathcal{F}$ with respect to* dist_{GMM} *using* $m_{\text{NON-PRIVATE}}(\alpha, \beta, k, d)$ *samples. Then for every* $\varepsilon < \ln(2)/3$, $\delta \in (0, 1)$, $\gamma \leq \frac{\varepsilon\alpha}{C_2\sqrt{k\ln(2/\delta)}\sqrt{d(d+\ln(12k/\beta))}\cdot\ln(12k/\delta)}$ *for a sufficiently large constant C and* $t = \max\{5, \lceil \frac{20}{\varepsilon}\ln(1 + \frac{e^{\varepsilon}-1}{2\delta})\rceil\}$, *there is a learner* $\mathcal{A}_{\text{PRIVATE}}$ *with the following properties:*

1. $\mathcal{A}_{\text{PRIVATE}}$ *is* $(2\varepsilon, 4e^{\varepsilon}\delta)$*-DP.*
2. $\mathcal{A}_{\text{PRIVATE}}$ *PAC learns* $\mathcal{F}$ *using* $O(m_{\text{NON-PRIVATE}}(\gamma, \beta/2t, k, d)\log(1/\delta)/\varepsilon)$ *samples.*
3. $\mathcal{A}_{\text{PRIVATE}}$ *runs in time* $O((\log(1/\delta)/\varepsilon)\cdot T_{\mathcal{A}} + (\log(1/\delta)/\varepsilon)^2 \cdot (k^2d^3 + k^3\log k))$, *where* $T_{\mathcal{A}}$ *is the running time for the non-private algorithm.*

Proof. In order to use Algorithm 1 we need to define a masking mechanism, so we use the masking mechanism $\mathcal{B}_{\text{GMM}}$ that defined in Lemma 6.2.1. However, α in Lemma 6.2.1 will be replaced with $\alpha/2z$ from Theorem 4.3.1, also β in Lemma 6.2.1 will be replaced with $\beta/2t$, because probability is divided over subsets, and also divided between the non-private algorithm and the concentration in masking mechanism $\mathcal{B}_{\text{GMM}}$.

1. The hypothesis of Theorem 4.3.1 holds so it is true.
2. $m_{\text{PRIVATE}} \geq \max\{5, \lceil \frac{20}{\varepsilon}\ln(1 + \frac{e^{\varepsilon}-1}{2\delta})\rceil\} \cdot m_{\text{NON-PRIVATE}}(\gamma, \beta/2t, k, d) =$

$$O(m_{\text{NON-PRIVATE}}(\gamma, \beta/2t, k, d)\log(1/\delta)/\varepsilon) \tag{7.3}$$

3. Recall from Remark 4.3.2, $\mathcal{A}_{\text{PRIVATE}}$ runs in time $O(t \cdot T_{\mathcal{A}} + t^2 \cdot T_{\text{dist}} + T_{\mathcal{B}})$.

   We start proving T_{dist}, running time to multiply two $d \times d$ matrices is $O(d^3)$, this implies $\text{dist}_{\text{SINGLE}}$ in Equation 5.1 is computed in time of $O(d^3)$, applying Lemma 6.1.4 implies that we can compute dist^k in $O(k^2d^3 + k^3\log k)$.

   Now we prove $T_{\mathcal{B}}$, masking each component takes time $O(d^3)$ because of matrix multiplication in Algorithm 2, to compute the masking on GMM we apply Lemma 6.1.3 so it takes time of $O(k \cdot d^3 + k\log k)$.

Finally, $t = \max\{5, \lceil \frac{20}{\varepsilon} \ln(1 + \frac{e^{\varepsilon}-1}{2\delta}) \rceil\} = O(\log(1/\delta)/\varepsilon)$ which is the number of subsets.

which concludes the proof.

□

7.2 Applications in Private Learning of GMMs

In this section, we apply the general theorem to reduce the learning GMMs parameters' from private to its non-private counterpart. We specifically pick [MV10] non-private algorithm. As a result, we introduce the first sample complexity upper bound and the first polynomial time algorithm in d for learning the parameters of the Gaussian Mixture Models privately without requiring any boundedness assumptions on the parameters.

Definition 7.2.1 (α-statistically learnable [MV10])**.** *We say a GMM $F = (w_i, \mu_i, \Sigma_i)_{i=1}^k$ is γ-statistically learnable if $\min_i w_i \geq \gamma$ and $\min_{i \neq j} d_{\mathrm{TV}}(\mathcal{N}(\mu_i, \Sigma_i), \mathcal{N}(\mu_j, \Sigma_j)) \geq \gamma$.*

If a GMM is γ-statistically learnable, means they are far from each other with certain distance γ. We will be able to recover its components accurately.

Theorem 7.2.1 (Non-private learning of GMMs [MV10])**.** *There exist an algorithm $\mathcal{A}(D, \alpha, \beta)$ that has the following property: for any fixed $k \in \mathbb{N}$, given an i.i.d. sample D of size $m_{\mathcal{A}}(d, k, \alpha, \beta)$ generated from F^*, where F^* is any α-statistically learnable d-dimensional GMM with k components, $\mathcal{A}$ returns $\hat{F}$ such that with probability at least $1-\beta$, F^* and $\hat{F}$ are α-close with respect to dist_{GMM}. Moreover, for any fixed $k \in \mathbb{N}$, the sample complexity, $m_{\mathcal{A}}(d, k, \alpha, \beta)$, and the running time are polynomial in $d, 1/\alpha$ and $1/\beta$.*

Our reduction algorithm with non-private learner of GMMs [MV10], gives the first sample complexity upper bound and the first polynomial time algorithm in d for learning the parameters of the Gaussian Mixture Models privately without requiring any boundedness assumptions on the parameters.

Corollary 7.2.2. *There exist an algorithm $\mathcal{A}(D, \alpha, \beta, \varepsilon, \delta)$ that if given an i.i.d sample D of size $m_{\mathcal{A}}(d, k, \alpha, \beta, \varepsilon, \delta)$ generated from $\mathcal{F}^*$, where $\mathcal{F}^*$ is any α-statistically learnable subclass of GMMs with k components in $\mathbb{R}^d$ for any fixed $k \in \mathbb{N}$, then $\mathcal{A}$ will privately PAC learns $\mathcal{F}^*$ and returns $\hat{F}$ with respect to dist_{GMM} with propability at least $1-\beta$ such*

that F^* *and* $\hat{F}$ *are* α*-close with respect to* dist_{GMM}*. Moreover, for any fixed* $k \in \mathbb{N}$*, the sample complexity,* $m_{\mathcal{A}}(d, k, \alpha, \beta, \varepsilon, \delta)$*, and the running time are polynomial in* $d, 1/\alpha$*,* $1/\beta$*,* $1/\varepsilon$*,* $\log(1/\delta)$*.*

Proof. We could plug in non-private Theorem 7.2.1 into the reduction Theorem 7.1.3 to make it private. □

In conclusion, we showed a general theorem to reduce the learning of parameters of GMMs from private to there non-private counterpart. And we applied on non-private algorithm [MV10]. As a result, We could reach the first sample complexity upper bound and the first polynomial time algorithm in d for learning the parameters of the Gaussian Mixture Models privately without requiring any boundedness assumptions on the parameters. We incurred a small overhead in sample complexity and running time over the algorithm in [MV10].

Chapter 8

Conclusion

8.1 Summary

In this thesis, we develop a technique that allows us to privatize existing non-private algorithms in a BlackBox manner while only incurring a small overhead in sample complexity and running time. We further show that we can learn the unbounded Gaussian Mixture Model privately.

To prove the results we introduced a Private Populous Estimator (PPE) which is a generalized version of the one used in [AL22]. We also simplified the notion from a convex semimetric space to semimetric space and lessen convexity and locality properties.

We develop a new masking mechanism for a single Gaussian component, which requires adding noise to all component parameters. Firstly, noise the mixing weight of a single component using a Gaussian mechanism. Secondly, noise the mean of a single component using empirically re-scaled Gaussian mechanism where the empirical covariance matrix is used to shape the noise that we add to the mean. Finally, noise the covariance matrix of a single component using the noising mechanism described in [AL22, §5].

As a major achievement, we introduced a general recipe to turn a masking mechanism for a component into a masking mechanism for mixtures. The idea is simple, we add noise to each of the components and then permute the output. Then we applied it to mask a mixture of k Gaussians.

In the results, we introduced our Private to Non-Private Reduction theorem for learning GMMs. We also applied this reduction on [MV10] non-private algorithm, to get the first sample complexity upper bound and the first polynomial time algorithm in d for learning the parameters of the Gaussian Mixture Models privately without requiring any

boundedness assumptions on the parameters. We incurred a small overhead in sample complexity and running time over the algorithm in [MV10].

8.2 Future Work

More ambitiously, we can investigate if there is an efficient algorithm for learning unbounded GMMs robustly, where a small fraction of the samples are arbitrarily corrupted by an adversary. First, we draw m i.i.d. samples from a GMM. Then, the adversary chooses at most αm samples and modifies them arbitrarily. This raises the question below.

> Is there a polynomial time and polynomial sample reduction from private and robust learning to non-private and robust learning of mixtures?

If so, it is possible to obtain the first polynomial time algorithm for private and robust learning of unbounded GMMs.

Another possible work as an extension to this thesis is to extend the reduction from GMMs to more general cases, such as mixtures of exponential distributions and laplacian mixture modeling. The target would be to privately estimate the parameters of these mixtures by reducing the problem to its non-private counterpart, and utilizing existing non-private parameter estimation algorithms in a BlackBox manner while only incurring a small overhead in sample complexity and running time. Our PPE framework could work efficiently for all other types of mixtures. The only drawback is that we need a new masking mechanism for each type.

Bibliography

[AAK21] Ishaq Aden-Ali, Hassan Ashtiani, and Gautam Kamath. "On the sample complexity of privately learning unbounded high-dimensional gaussians". In: *Algorithmic Learning Theory*. PMLR. 2021, pp. 185–216.

[AAL21] Ishaq Aden-Ali, Hassan Ashtiani, and Christopher Liaw. "Privately learning mixtures of axis-aligned gaussians". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 3925–3938.

[AL22] Hassan Ashtiani and Christopher Liaw. "Private and polynomial time algorithms for learning Gaussians and beyond". In: *Proceedings of Thirty Fifth Conference on Learning Theory*. Ed. by Po-Ling Loh and Maxim Raginsky. Vol. 178. Proceedings of Machine Learning Research. PMLR, Feb. 2022, pp. 1075–1076.

[AM05] Dimitris Achlioptas and Frank McSherry. "On spectral learning of mixtures of distributions". In: *International Conference on Computational Learning Theory*. Springer. 2005, pp. 458–469.

[ASZ21] Jayadev Acharya, Ziteng Sun, and Huanyu Zhang. "Differentially private assouad, fano, and le cam". In: *Algorithmic Learning Theory*. PMLR. 2021, pp. 48–78.

[BDJKKV22] Ainesh Bakshi, Ilias Diakonikolas, He Jia, Daniel M Kane, Pravesh K Kothari, and Santosh S Vempala. "Robustly learning mixtures of k arbitrary gaussians". In: *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*. 2022, pp. 1234–1247.

[BDKU20] Sourav Biswas, Yihe Dong, Gautam Kamath, and Jonathan Ullman. "Coinpress: Practical private mean and covariance estimation". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 14475–14485.

[BGSUZ21] Gavin Brown, Marco Gaboardi, Adam Smith, Jonathan Ullman, and Lydia Zakynthinou. "Covariance-aware private mean estimation without private covariance estimation". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 7950–7964.

[BS09] Mikhail Belkin and Kaushik Sinha. "Learning Gaussian mixtures with arbitrary separation". In: *arXiv preprint arXiv:0907.1054* (2009).

[BS10] Mikhail Belkin and Kaushik Sinha. "Polynomial learning of distribution families". In: *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*. IEEE. 2010, pp. 103–112.

[BSKW19] Mark Bun, Thomas Steinke, Gautam Kamath, and Zhiwei Steven Wu. "Private hypothesis selection". In: *Advances in Neural Information Processing Systems* 32 (2019).

[BV08] S Charles Brubaker and Santosh S Vempala. "Isotropic PCA and affine-invariant clustering". In: *Building Bridges*. Springer, 2008, pp. 241–281.

[Das99] Sanjoy Dasgupta. "Learning mixtures of Gaussians". In: *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)*. IEEE. 1999, pp. 634–644.

[DKMMN06] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. "Our data, ourselves: Privacy via distributed noise generation". In: *Annual international conference on the theory and applications of cryptographic techniques*. Springer. 2006, pp. 486–503.

[DL09] Cynthia Dwork and Jing Lei. "Differential privacy and robust statistics". In: *Proceedings of the forty-first annual ACM symposium on Theory of computing*. 2009, pp. 371–380.

[DMNS06] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. "Calibrating Noise to Sensitivity in Private Data Analysis". In: vol. Vol. 3876. Jan. 2006, pp. 265–284. ISBN: 978-3-540-32731-8.

[DMR18] Luc Devroye, Abbas Mehrabian, and Tommy Reddad. "The total variation distance between high-dimensional Gaussians". In: *arXiv preprint arXiv:1810.08693* (2018).

[DR+14] Cynthia Dwork, Aaron Roth, et al. "The algorithmic foundations of differential privacy". In: *Foundations and Trends® in Theoretical Computer Science* 9.3–4 (2014), pp. 211–407.

[DRV10] Cynthia Dwork, Guy N. Rothblum, and Salil P. Vadhan. "Boosting and Differential Privacy". In: *2010 IEEE 51st Annual Symposium on Foundations of Computer Science* (2010), pp. 51–60.

[FSO06] Jon Feldman, Rocco A Servedio, and Ryan O'Donnell. "PAC learning axis-aligned mixtures of Gaussians with no separation assumption". In: *International Conference on Computational Learning Theory*. Springer. 2006, pp. 20–34.

[GDGK18] Quan Geng, Wei Ding, Ruiqi Guo, and Sanjiv Kumar. "Truncated Laplacian mechanism for approximate differential privacy". In: *arXiv preprint arXiv:1810.00877* (2018).

[KLSU19] Gautam Kamath, Jerry Li, Vikrant Singhal, and Jonathan Ullman. "Privately learning high-dimensional distributions". In: *Conference on Learning Theory*. PMLR. 2019, pp. 1853–1902.

[KMSSU22] Gautam Kamath, Argyris Mouzakis, Vikrant Singhal, Thomas Steinke, and Jonathan Ullman. "A Private and Computationally-Efficient Estimator for Unbounded Gaussians". In: *Proceedings of Thirty Fifth Conference on Learning Theory*. Ed. by Po-Ling Loh and Maxim Raginsky. Vol. 178. Proceedings of Machine Learning Research. PMLR, Feb. 2022, pp. 544–572.

[KMV10] Adam Tauman Kalai, Ankur Moitra, and Gregory Valiant. "Efficiently learning mixtures of two Gaussians". In: *Proceedings of the forty-second ACM symposium on Theory of computing*. 2010, pp. 553–562.

[KMV22] Pravesh Kothari, Pasin Manurangsi, and Ameya Velingker. "Private robust estimation by stabilizing convex relaxations". In: *Conference on Learning Theory*. PMLR. 2022, pp. 723–777.

[KSSU19] Gautam Kamath, Or Sheffet, Vikrant Singhal, and Jonathan Ullman. "Differentially private algorithms for learning mixtures of separated gaussians". In: *Advances in Neural Information Processing Systems* 32 (2019).

[KV17] Vishesh Karwa and Salil Vadhan. "Finite sample differentially private confidence intervals". In: *arXiv preprint arXiv:1711.03908* (2017).

[LM00] Beatrice Laurent and Pascal Massart. "Adaptive estimation of a quadratic functional by model selection". In: *Annals of Statistics* (2000), pp. 1302–1338.

[LM21] Allen Liu and Ankur Moitra. "Settling the robust learnability of mixtures of gaussians". In: *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*. 2021, pp. 518–531.

[LM22] Allen Liu and Ankur Moitra. "Learning gmms with nearly optimal robustness guarantees". In: *Conference on Learning Theory*. PMLR. 2022, pp. 2815–2895.

[MV10] Ankur Moitra and Gregory Valiant. "Settling the polynomial learnability of mixtures of gaussians". In: *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*. IEEE. 2010, pp. 93–102.

[NRS07] Kobbi Nissim, Sofya Raskhodnikova, and Adam D. Smith. "Smooth sensitivity and sampling in private data analysis". In: *STOC '07*. 2007.

[SK01] Arora Sanjeev and Ravi Kannan. "Learning mixtures of arbitrary gaussians". In: *Proceedings of the thirty-third annual ACM symposium on Theory of computing*. 2001, pp. 247–257.

[TCKMS22] Eliad Tsfadia, Edith Cohen, Haim Kaplan, Yishay Mansour, and Uri Stemmer. "Friendlycore: Practical differentially private aggregation". In: *International Conference on Machine Learning*. PMLR. 2022, pp. 21828–21863.

[VW04] Santosh Vempala and Grant Wang. "A spectral algorithm for learning mixture models". In: *Journal of Computer and System Sciences* 68.4 (2004), pp. 841–860.