Robustly Learning General Mixtures of Gaussians

ALLEN LIU and ANKUR MOITRA, Massachusetts Institute of Technology, USA

This work represents a natural coalescence of two important lines of work — learning mixtures of Gaussians and algorithmic robust statistics. In particular, we give the first provably robust algorithm for learning mixtures of any constant number of Gaussians. We require only mild assumptions on the mixing weights and that the total variation distance between components is bounded away from zero. At the heart of our algorithm is a new method for proving a type of dimension-independent polynomial identifiability — which we call robust identifiability — through applying a carefully chosen sequence of differential operations to certain generating functions that not only encode the parameters we would like to learn but also the system of polynomial equations we would like to solve. We show how the symbolic identities we derive can be directly used to analyze a natural sum-of-squares relaxation.

CCS Concepts: • **Theory of computation** → **Unsupervised learning and clustering**; *Design and analysis of algorithms*; • **Mathematics of computing** → **Multivariate statistics;**

Additional Key Words and Phrases: Mixtures of Gaussians, GMMs, robust statistics, method of moments, sum of squares, generating functions

ACM Reference format:
Allen Liu and Ankur Moitra. 2023. Robustly Learning General Mixtures of Gaussians. *J. ACM* 70, 3, Article 21 (May 2023), 53 pages.
https://doi.org/10.1145/3583680

1 INTRODUCTION

This work represents a natural coalescence of two important lines of work — learning mixtures of Gaussians and algorithmic robust statistics — that we describe next: In 1894 Karl Pearson [49] introduced mixture models and asked:

> *Is there a statistically efficient method for learning the parameters of a mixture of Gaussians from samples?*

Mixtures of Gaussians are natural models for when data is believed to come from two or more heterogenous sources. Since then they have found a wide variety of applications spanning statistics, biology, physics, and computer science. The textbook approach for fitting the parameters is to use the maximum likelihood estimator. However, it is not clear how many samples it requires to

This work was supported in part by a Microsoft Trustworthy AI Grant, NSF CAREER Award CCF-1453261, NSF Large CCF1565235, a David and Lucile Packard Fellowship and an ONR Young Investigator Award.

Authors' addresses: A. Liu, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 77 Massachusetts Ave., 38-401, office 32-G604, Cambridge MA 02139, USA; email: cliu568@mit.edu; A. Moitra, Department of Mathematics, Massachusetts Institute of Technology, 77 Massachusetts Ave, office 2-472, Cambridge MA 02139, USA; email: moitra@mit.edu.

0004-5411/2023/05-ART21 $15.00
https://doi.org/10.1145/3583680

estimate the parameters up to some desired accuracy. Even worse, it is hard to compute in high-dimensions [3].

In a seminal work, Sanjoy Dasgupta [19] introduced the problem to theoretical computer science and asked:

> *Is there an efficient algorithm for learning the parameters?*

Many early works were based on clustering the samples into which components generated them [1, 3, 16, 20, 42, 55]. However, when the components overlap non-trivially this is no longer possible. Nevertheless, Kalai, Moitra, and Valiant [38] gave an algorithm for learning the parameters of a mixture of two Gaussians that works even if the components are almost entirely overlapping. Their approach was based on reducing the high-dimensional problem to a series of one-dimensional problems and exploiting the structure of the moments. In particular, they proved that every mixture of two Gaussians is uniquely determined by its first six moments. Subsequently, Moitra and Valiant [47] and Belkin and Sinha [13] were able to give an algorithm for learning the parameters of a mixture of any constant number of Gaussians. These algorithms crucially made use of even higher moments along with several new ingredients like methods for separating out submixtures that are flat along some directions and difficult to directly learn. There are also approaches based on tensor decompositions [15, 26, 34] that get polynomial dependence on the number of components assuming that the parameters are non-degenerate and subject to some kind of smoothing. However, all of these algorithms break down when the data does not exactly come from a mixture of Gaussians. In fact, in Karl Pearson's original application [49], and in many others, mixtures of Gaussians are only intended as an approximation to the true data generating process.

The field of robust statistics was launched by the seminal works of John Tukey [52, 53] and Peter Huber [35] and seeks to address this kind of shortcoming by designing estimators that are provably robust to some fraction of their data being adversarially corrupted. The field had many successes and explicated some of the general principles behind what makes estimators robust [27, 36]. Provably robust estimators were discovered for fundamental tasks such as estimating the mean and covariance of a distribution and for linear regression. There are a variety of types of robustness guarantees but the crucial point is that these estimators can all tolerate a constant fraction of corruptions that is *independent of the dimension*. However, all of these estimators turn out to be hard to compute in high-dimensions [14, 28, 37].

Recently, Diakonikolas et al. [22] and Lai et al. [43] designed the first provably robust and computationally efficient estimators for the mean and covariance. They operate under some kind of assumption on the uncorrupted data — either that they come from a simple generative model like a single Gaussian or that they have bounded moments. To put this in perspective, without corruptions this is a trivial learning task because if you want to learn the mean and covariance for any distribution with bounded moments you can just use the empirical mean and empirical covariance, respectively. Algorithmic robust statistics has transformed into a highly active area [7, 8, 17, 18, 23–25, 31, 40, 41, 44, 50] with many successes. Since then, a much sought-after goal has been to answer the following challenge:

> *Is there a provably robust and computationally efficient algorithm for learning mixtures of Gaussians? Can we robustify the existing learning results?*

There has been steady progress on this problem. Diakonikolas et al. [22] gave a robust algorithm for learning mixtures of spherical Gaussians. In recent breakthroughs, Bakshi and Kothari [6] and Diakonikolas et al. [21] gave a robust algorithm for learning clusterable mixtures of Gaussians, and building on this, Kane [39] gave a robust algorithm for learning mixtures of two Gaussians. We note that these works do place some mild restrictions on the mixing weights and the variances.

In particular, they need the mixing weights to have bounded fractionality and the variances of all components to be nonzero in all directions.

The algorithms of Bakshi and Kothari [6] and Diakonikolas et al. [21] rely on the powerful sum-of-squares hierarchy [48]. One view is that it finds an operator, called the pseudo-expectation, that maps low degree polynomials to real values. Moreover, a large number of consistency constraints are imposed that force it to in some restricted sense behave like taking the expectation over a distribution on assignments to the variables. It gives a natural way to incorporate systems of polynomial constraints into a relaxation which can model complex primitives like selecting a large subset of the samples and enforcing that they have approximately the same types of moment bounds that hold for a single Gaussian. Of course, the real challenge is that you need some way to reason about the pseudo-expectation operator that only uses certain types of allowable steps that can be derived through the constraints that you enforced in the relaxation.

1.1 Key Challenges

It is believed that the sum-of-squares hierarchy might actually be the key to solving the more general problem of robustly learning a mixture of any constant number of Gaussians. However there are some obstacles that need to be overcome:

Robust Identifiability. Behind every polynomial time algorithm for learning the parameters of a mixture model is an argument for why there cannot be two mixtures with noticeably different parameters that produce almost the same distribution. In fact, we need quantitative bounds that say any two mixtures that are ϵ-different must be at least $\mathrm{poly}(\epsilon, 1/d)$ different according to some natural family of test functions, usually the set of low degree polynomials. Here d is the dimension. This is called *polynomial identifiability* [46, 51]. Because we allow a polynomial dependence on $1/d$, it often does not matter too much how we measure the differences between two mixtures, either in terms of some natural parameter distance between their components or in terms of the total variation distance, again between their components.

However, we need much stronger bounds when it comes to robust learning problems where we want to be able to tolerate a constant fraction of corruptions that is dimension independent. In particular, we need a family of test functions with the stronger quantitative property that whenever we have two mixtures whose components are ϵ-different in total variation distance there is some function in the family that has at least $\mathrm{poly}(\epsilon)$ discrepancy when we take the difference between its expectations with respect to the two distributions (and its variance must also be bounded). In particular, this relationship cannot involve dependence on the dimension d. We will call this *robust identifiability.* Recall that the non-robust learning algorithms for mixtures of Gaussians reduce to a series of one-dimensional problems. Unfortunately this strategy inherently introduces polynomial factors in d and it cannot give what we are after. For the special case of clusterable mixtures of Gaussians, Bakshi and Kothari [6] and Diakonikolas et al. [21] proved robust identifiability and their approach was based on classifying the ways in which two single Gaussians can have total variation distance close to one. When it comes to the more general problem of handling mixtures where the components can overlap non-trivially, it seems difficult to follow the same route because we can no longer match components from the two mixtures to each other and almost cancel them both out. To prove robust identifiability for general mixtures of Gaussians, we use a carefully chosen set of low-degree polynomials as test functions, namely the (a multivariate analog) of the Hermite polynomials. Our proof of robust identifiability is explained in more detail in the following subsection.

Reasoning About the Sum-of-Squares Relaxation. A proof of robust identifiability alone does not give an efficient algorithm. However, we leverage the now standard sum-of-squares framework [9] to convert our proof of robust identifiability into an actual learning algorithm. The sum-of-squares hierarchy is a general framework for coming up with large and powerful semidefinite programming relaxations that can be applied to many sorts of problems [10–12, 33]. The semidefinite programs automatically enforce a large family of constraints and proofs of statements that use only these types of sum-of-squares amenable constraints can be directly converted into algorithms. However, it is often quite challenging to understand whether or not it works and/or to identify, out of all the constraints that are enforced on the pseudo-expectation, which ones are actually useful in the analysis [30, 32]. What makes matters especially challenging in our setting is that it is clear the structure of the higher moments of a mixture of Gaussians must play a major role. But how exactly do we leverage them in our analysis?

1.2 Our Techniques and Main Result

Actually, we overcome both obstacles using the same approach. We store the relevant moments and variables we would like to solve for in certain generating functions. Then by manipulating the generating functions using differential operators, we are able to reason about an SOS relaxation of a natural polynomial system that allows us to solve for the parameters that we want to learn.

Let us describe the setup. For simplicity, assume that the mixture is in isotropic position. First, we have the unknown parameters of the mixture. Let

$$\mathcal{M} = w_1 N(\mu_1, I + \Sigma_1) + \cdots + w_k N(\mu_k, I + \Sigma_k)$$

where w_i are the mixing weights and μ_i and $I + \Sigma_i$ denote the mean and covariance of the ith component. Second, we have the indeterminates we would like to solve for. These will be denoted $\widetilde{\mu_i}$ and $\widetilde{\Sigma_i}$ and our intention is for these to be the means and covariances of a hypothetical mixture of Gaussians.[1] We will also guess the mixing weights of the hypothetical mixture $\widetilde{w_i}$. Finally, we have a d-dimensional vector $X = (X_1, \dots, X_d)$ of formal variables and one auxiliary formal variable y. These will mostly be used to help us organize everything in a convenient way. Roughly, we would like to solve for $\widetilde{\mu_i}$ and $\widetilde{\Sigma_i}$ so that the hypothetical mixture

$$\widetilde{\mathcal{M}} = \widetilde{w_1} N(\widetilde{\mu_1}, I + \widetilde{\Sigma_1}) + \cdots + \widetilde{w_k} N(\widetilde{\mu_k}, \widetilde{I + \Sigma_k})$$

is close to $\mathcal{M}$ on the family of test functions (which will be low-degree multivariate Hermite polynomials). It turns out that this amounts to solving a polynomial system for the indeterminates.

Now we explain in more detail how to actually reason about and solve the polynomial system. It will be useful to work with the following generating functions. First, let

$$F(y) = \sum_{i=1}^{k} w_i e^{\mu_i(X)y + \frac{1}{2}\Sigma_i(X)y^2}$$

Here we have used the notation that $\mu_i(X)$ denotes the inner product of μ_i with the d-dimensional vector X and that $\Sigma_i(X)$ denotes the quadratic form of X on Σ_i. Second, let

$$\widetilde{F}(y) = \sum_{i=1}^{k} \widetilde{w_i} e^{\widetilde{\mu_i}(X)y + \frac{1}{2}\widetilde{\Sigma_i}(X)y^2}$$

As is familiar from elementary combinatorics, we can tease out important properties of the generating function by applying carefully chosen operators that involve differentiation. This requires a lot more bookkeeping than usual because there are unknown parameters of the mixture, indeterminates, and formal variables. But it turns out that there are simple differential operators we can

[1] Technically our setup is slightly different in that we solve for vectors $u_1, \dots, u_k$ and $v_1, \dots, v_k$ that are supposed to form an orthonormal basis for the span of the $\{\widetilde{\mu_i}\}$ and $\{\widetilde{\Sigma_i}\}$, respectively. Regardless, the argument is conceptually the same.

apply which can isolate components. To gain some intuition, consider the operator

$$\mathcal{D}_i = \partial_y - (\mu_i(X) + \Sigma_i(X)y)\,.$$

Note that

$$\mathcal{D}_i \left(e^{\mu_i(X)y + \frac{1}{2}\Sigma_i(X)y^2} \right) = 0\,,$$

in other words, this operator annihilates the i^{th} component. Thus, by composing such operators, we can annihilate all but one of the components in F.[2] On the other hand, note that applying differential operators is just a rearrangement of the polynomials that show up in the infinite sum representation of the generating function but using differential operators and generating functions in exponential form gives a particularly convenient way to derive useful expressions that would otherwise be extremely complex to write down.

Ultimately we derive a symbolic identity

$$\widetilde{w_k} \prod_{i=1}^{k} (\widetilde{\Sigma_k}(X) - \Sigma_i(X))^{2^{i-1}} \prod_{i=1}^{k-1} (\widetilde{\Sigma_k}(X) - \widetilde{\Sigma_i}(X))^{2^{k+i-1}} = \sum_{i=1}^{m} P_i(X)(\widetilde{h_i}(X) - h_i(X)) \qquad (1)$$

where m is a function of k. In the above, the $h_i(X)$'s are the expectations of the multivariate Hermite polynomials for the true mixture $\mathcal{M}$ and the $\widetilde{h_i}$'s are the expectations of the multivariate Hermite polynomials for the hypothetical mixture $\widetilde{\mathcal{M}}$. A more detailed explanation of Hermite polynomials is given in Section 3, but for now we may simply think of them as modified moments. The reason that we use Hermite polynomials instead of standard moments is that they can be robustly estimated without losing dimension-dependent factors (see e.g., [39]).

The above identity (when combined with a few others of a similar form) allows us to deduce robust identifiability. Roughly, this is because if we have a mixture $\widetilde{\mathcal{M}}$ with means $\widetilde{\mu_i}$ and covariances $I + \widetilde{\Sigma_i}$ that don't match those of $\mathcal{M}$, then the LHS of (1) will be bounded away from 0, implying that some term on the RHS must also be bounded away from 0. This means that there must be some $i \leq m = O_k(1)$ such that $h_i(X)$ and $\widetilde{h_i(X)}$ are substantially different — i.e., there is some test function that is a low-degree multivariate Hermite polynomial that distinguishes the two mixtures.

Robust identifiability alone does not give us a polynomial time learning algorithm. However, it turns out that we can use SOS to obtain a polynomial time learning algorithm from the argument for robust identifiability i.e., we essentially get the learning algorithm "for free". The key point is that the $h_i(X)$ can be estimated using our samples and the coefficients of the $\widetilde{h_i}$ are explicit polynomials in the indeterminates that we can write down. Also the P_i's are polynomials in *everything*: the unknown parameters, the indeterminates, and the formal variables except for y. To set up an SOS system, we obtain robust estimates $\overline{h_i}$ for the expectations of the Hermite polynomials $h_i(X)$ for the true mixture that we can compute from existing techniques in the literature. We then enforce that the expectations of the Hermite polynomials for the hypothetical mixture $\widetilde{h_i}(X)$ are close to these robust estimates where closeness is defined in terms of the distance between their coefficient vectors.

It is not immediately clear why the expression in (1) ought to be useful for solving the SOS system that we set up. After all, we cannot explicitly compute it because it depends on things we do not have, like the true parameters, the μ_i's and Σ_i's. However, the sum-of-squares relaxation enforces that the pseudo-expectation operator assigns values to polynomials in the indeterminates in a way that behaves like an actual distribution on solutions when we are evaluating certain types of low degree expressions that contain the one above. So even though we do not know the actual

[2] There are some additional details because applying $\mathcal{D}_i$ to a different component creates an extra polynomial factor in front. Section 4.3.1 shows how to deal with this complication.

polynomials in the identity, they exist and the fact that they are enforced is enough to ensure that we can estimate the covariance of the kth component We stress that this is just the high-level picture and many more details are needed to fill it in.

Using these techniques, we come to the main result of our paper, which is a polynomial-time algorithm for robustly learning the parameters of a high-dimensional mixture of a constant number of Gaussians. Our main theorem is (informally) stated below. A formal statement can be found in Theorem 8.3.

Theorem 1.1. *Let k be a constant. Let $\mathcal{M} = w_1 G_1 + \cdots + w_k G_k$ be a mixture of Gaussians in $\mathbb{R}^d$ whose components are non-degenerate and such that the mixing weights have bounded fractionality and TV distances between different components are lower bounded. (Both of these bounds can be any function of k). Given $n = \text{poly}(d/\epsilon)$ samples from $\mathcal{M}$ that are ϵ-corrupted, there is an algorithm that runs in time $\text{poly}(n)$ and with high probability outputs a set of mixing weights $\widetilde{w_1}, \ldots, \widetilde{w_k}$ and components $\widetilde{G_1}, \ldots, \widetilde{G_k}$ that are $\text{poly}(\epsilon)$-close to the true components (up to some permutation).*

1.2.1 Discussion of Assumptions and Later Improvements.

Bounded Fractionality of Mixing Weights: The assumption of bounded fractionality stems from an issue in previous work [21] about learning clusterable mixtures of Gaussians i.e., when the components have essentially no overlap. We use some of their subroutines in our algorithm for clustering the mixture into submixtures where the components are not too far apart from each other (see Section 6). While [21] claims to handle general mixing weights, the analysis of the algorithm in [21] is only done in detail for the case of uniform mixing weights and the argument in Appendix C for reducing from general mixing weights to uniform mixing weights does not work. The authors of [21] along with the authors of [6] were able to fix these arguments to handle general mixing weights in [4]. The modified proof essentially rewrites the pre-existing argument but with non-uniform mixing weights instead of trying to go through a direct reduction. Plugging this improved clustering result into our analysis, instead of using the weaker guarantees for uniform mixing weights, we are able to straightforwardly remove the bounded fractionality assumption. See Section 6.3 for a formal statement and explanation.

Separation Assumption: While our original proof required constant separation between components, we show in a follow-up paper, [45], that this assumption can be replaced with a separation of $\epsilon^{\Omega_k(1)}$ (which is a qualitatively necessary assumption for parameter learning)[3] using standard tricks (see Theorem 9.2 in [45]). This argument works independently of the improvement for the mixing weight assumption. See Section 8.1 for a more formal statement and explanation.

Non-degeneracy of Components: This assumption was included so there are no bit complexity issues. In fact, dealing with potentially degenerate covariance matrices requires a formal specification of the model of computation.

To make the timeline of events clear, we stated our original result above. However, plugging in these improvements (for removing the bounded fractionality and separation assumption), we obtain an improved result, stated below. The formal statement can be found in Theorem 8.6. We emphasize that these modifications are *completely independent* of our main contributions, but are rather tools that we employ to reduce to the case where the components are not too far from

[3] Note that we cannot guarantee to learn the parameters to accuracy better than $\epsilon^{O(1/k)}$ because there are two mixtures whose components are all $\epsilon^{O(1/k)}$-separated but are ϵ-close in TV distance as mixtures [29] and are thus indistinguishable with ϵ-corruptions. Thus, if the assumed separation is less than $\epsilon^{O(1/k)}$, then we cannot hope to learn the mixing weights accurately. Alternatively, with no separation, we can still guarantee to learn a list that covers all of the components (see Theorem 9.2 in [45]).

each other. We reduce to this case by running a preprocessing step where we cluster the mixture into such submixtures. All of the assumptions stem from the clustering step, which is done via modifications to the techniques for learning fully clusterable mixtures in [6, 21].

Theorem 1.2. *Let k be a constant. Let $\mathcal{M} = w_1 G_1 + \cdots + w_k G_k$ be a mixture of Gaussians in $\mathbb{R}^d$ whose components are non-degenerate and such that the mixing weights are lower bounded by some function of k. Also assume that the TV distances between components are at least $\epsilon^{\Omega_k(1)}$. Then given $n = \text{poly}(d/\epsilon)$ samples from $\mathcal{M}$ that are ϵ-corrupted, there is an algorithm that runs in time $\text{poly}(n)$ and with high probability outputs a set of mixing weights $\widetilde{w_1}, \dots, \widetilde{w_k}$ and components $\widetilde{G_1}, \dots, \widetilde{G_k}$ that are $\text{poly}(\epsilon)$-close to the true components (up to some permutation).*

Remark. The improved clustering arguments of [4] are able to get polynomial dependence on the minimum mixing weight instead of exponential dependence so they are actually able to deal with mixing weights that are $\epsilon^{\Omega_k(1)}$. This improvement also plugs into our result as well.

1.3 Proof Overview

The proof of our main theorem can be broken down into several steps. We first present our main contribution, an algorithm for learning mixtures of Gaussians when no pair of components are too far apart. We introduce the necessary generating function machinery in Section 3 and then present our algorithm in Sections 4 and 5. Specifically, in Section 4 we show how to learn the parameters once we have estimates for the Hermite polynomials of the true mixture. And in Section 5, we show how to robustly estimate the Hermite polynomials, using similar techniques to [39].

To complete our full algorithm for learning general mixtures of Gaussians, we combine our aforementioned results with a clustering algorithm similar to [21]. Combining these algorithms, we prove that our algorithm outputs a mixture that is close to the true mixture in TV distance. This is done in Sections 6 and 7. We then prove identifiability in Section 8, implying that our algorithm actually learns the true parameters.

1.4 Concurrent and Subsequent Work

There are three main pieces of work that we discuss. The first by Bakshi et al. [4], was independent and concurrent to this one. The next is a subsequent work by Bakshi et al. [5] that improves their earlier result but also borrows techniques from our paper. We also discuss our follow-up work [45] that was after [4] but before [5] (and the contributions are essentially disjoint).

Bakshi et al. in [4] obtain a result that is similar to our main result Theorem 1.1, but using rather different techniques. There are a few key differences, which we discuss below. We learn the mixture to accuracy $\epsilon^{\Omega_k(1)}$ while their result only achieves accuracy $(1/\log(1/\epsilon))^{\Omega_k(1)}$, an exponentially worse guarantee. Also, our result solves parameter learning — i.e., we estimate the parameters of the true mixture — while their algorithm solves proper density estimation — i.e., it outputs a mixture of k Gaussians that is close to the true density. Their algorithm does not need any lower bound on the mixing weights or on the pairwise separation of the components. In fact, lower bounds on these quantities are necessary for parameter learning. However, our original result makes an even stronger assumption about the bounded fractionality of the mixing weights and pairwise separation.

Subsequently in [5], Bakshi et al. improve their earlier result to achieve parameter learning and accuracy $\epsilon^{\Omega_k(1)}$. In fact, the analysis of their new parameter learning algorithm relies crucially on the robust identifiability result from our paper (see Section 9 in [5]). Thus, all known algorithms for robust parameter learning go through our machinery and robust identifiability. Compared to our result, the main improvement in [5] is an improved, polynomial dependence on the minimum mixing weight.

Our follow-up work [45] improves the results here in a different direction. We are able to obtain an algorithm that solves density estimation to accuracy $\widetilde{O}(\epsilon)$ (instead of $\epsilon^{\Omega_k(1)}$). Note that parameter learning to this accuracy is information-theoretically impossible. The work in [45] does need a stronger assumption that the components have variances in all directions lower and upper bounded by a constant and the learning algorithm is improper, outputting a mixture of polynomials times Gaussians, instead of just a mixture of Gaussians. The main relevance of [45] to this paper is that as a first step, [45] shows how to improve the separation assumption in Theorem 1.1 from some constant to $\epsilon^{\Omega_k(1)}$.

2 PRELIMINARIES

2.1 Problem Setup

We use $N(\mu, \Sigma)$ to denote a Gaussian with mean μ and covariance Σ. We use $d_{\mathsf{TV}}(\mathcal{D}, \mathcal{D}')$ to denote the total variation distance between two distributions $\mathcal{D}, \mathcal{D}'$. We begin by formally defining the problem that we will study. First we define the contamination model. This is a standard definition from robust learning (see e.g., [21]).

Definition 2.1 (Strong Contamination Model). We say that a set of vectors $Y_1, \dots, Y_n$ is an ϵ-corrupted sample from a distribution $\mathcal{D}$ over $\mathbb{R}^d$ if it is generated as follows. First $X_1, \dots, X_n$ are sampled i.i.d. from $\mathcal{D}$. Then a (malicious, computationally unbounded) adversary observes $X_1, \dots, X_n$ and replaces up to ϵn of them with any vectors it chooses. The adversary may then reorder the vectors arbitrarily and output them as $Y_1, \dots, Y_n$.

In this paper, we study the following problem. There is an unknown mixture of Gaussians

$$\mathcal{M} = w_1 G_1 + \dots + w_k G_k$$

where $G_i = N(\mu_i, \Sigma_i)$. We receive an ϵ-corrupted sample $Y_1, \dots, Y_n$ from $\mathcal{M}$ where $n = \mathrm{poly}(d/\epsilon)$. The goal is to output a set of parameters $\widetilde{w_1}, \dots, \widetilde{w_k}$ and $(\widetilde{\mu_1}, \widetilde{\Sigma_1}), \dots, (\widetilde{\mu_k}, \widetilde{\Sigma_k})$ that are $\mathrm{poly}(\epsilon)$ close to the true parameters in the sense that there exists a permutation π on $[k]$ such that for all i

$$|w_i - \widetilde{w_{\pi(i)}}|, d_{\mathsf{TV}}\left(N(\mu_i, \Sigma_i), N(\widetilde{\mu_{\pi(i)}}, \widetilde{\Sigma_{\pi(i)}})\right) \leq \mathrm{poly}(\epsilon).$$

Throughout our paper, we will assume that all of the Gaussians that we consider have variance at least $\mathrm{poly}(\epsilon/d)$ and at most $\mathrm{poly}(d/\epsilon)$ in all directions i.e., they are not too flat. This implies that their covariance matrices are invertible so we may write expressions such as Σ_i^{-1}. We will also make the following assumptions about the mixture:

- The w_i are rational with denominator at most A
- For all $i \neq j$, $d_{\mathsf{TV}}(G_i, G_j) > b$

for some positive constants A, b. Note that a lower bound on the minimum mixing weight and a lower bound on the TV distance between components is necessary for parameter learning. Throughout this paper, we treat k, A, b as constants — i.e., A and b could be any function of k — and when we say polynomial, the exponent may depend on these parameters. We are primarily interested in dependence on ϵ and d (the dimension of the space).

2.2 Sum of Squares Proofs

We will make repeated use of the **Sum of Squares (SOS)** proof system. We review a few basic facts here (see [9] for a more extensive treatment). Our exposition here closely mirrors [21].

Definition 2.2 (Symbolic Polynomials). A degree-t symbolic polynomial P is a collection of indeterminates $\widehat{P}(\alpha)$, one for each multiset $\alpha \subseteq [n]$ of size at most t. We think of it as representing a polynomial $P : \mathbb{R}^n \to \mathbb{R}$ in the sense that

$$P(x) = \sum_{\alpha \subseteq [n], |\alpha| \leq t} \widehat{P}(\alpha) x^\alpha .$$

Definition 2.3 (SOS Proof). Let $x_1, \dots, x_n$ be indeterminates and let $\mathcal{A}$ be a set of polynomial inequalities

$$\{p_1(x) \geq 0, \dots, p_m(x) \geq 0\}$$

An SOS proof of an inequality $r(x) \geq 0$ from constraints $\mathcal{A}$ is a set of polynomials $\{r_S(x)\}_{S \subseteq [m]}$ such that each r_S is a sum of squares of polynomials and

$$r(x) = \sum_{S \subseteq [m]} r_S(x) \prod_{i \in S} p_i(x)$$

The degree of this proof is the maximum of the degrees of $r_S(x) \prod_{i \in S} p_i(x)$ over all S. We write

$$\mathcal{A} \vdash_k r(x) \geq 0$$

to denote that the constraints $\mathcal{A}$ give an SOS proof of degree k for the inequality $r(x) \geq 0$. Note that we can represent equality constraints in $\mathcal{A}$ by including $p(x) \geq 0$ and $-p(x) \geq 0$.

The dual objects to SOS proofs are pseudoexpectations. We will repeatedly make use of pseudoexpectations later on.

Definition 2.4. Let $x_1, \dots, x_n$ be indeterminates. A degree-k pseudoexpectation $\widetilde{\mathbb{E}}$ is a linear map

$$\widetilde{\mathbb{E}} : \mathbb{R}[x_1, \dots, x_n]_{\leq k} \to \mathbb{R}$$

from degree-k polynomials to $\mathbb{R}$ such that $\widetilde{\mathbb{E}}[p(x)^2] \geq 0$ for any p of degree at most $k/2$ and $\widetilde{\mathbb{E}}[1] = 1$. For a set of polynomial constraints $\mathcal{A} = \{p_1(x) \geq 0, \dots, p_m(x) \geq 0\}$, we say that $\widetilde{\mathbb{E}}$ satisfies $\mathcal{A}$ if

$$\widetilde{\mathbb{E}}[s^2(x) p_i(x)] \geq 0$$

for all polynomials $s(x)$ and $i \in [m]$ such that $s(x)^2 p_i(x)$ has degree at most k.

The key fact is that given a set of polynomial constraints, we can solve for a constant-degree pseudoexpectation that satisfies those constraints (or determine that none exist) in polynomial time as it reduces to solving a polynomially sized SDP.

Theorem 2.5 (SOS Algorithm [9]). *There is an algorithm that takes a natural number k and a satisfiable system of polynomial inequalities $\mathcal{A}$ in varibles $x_1, \dots, x_n$ with coefficients at most 2^n containing an inequality of the form $\|x\|^2 \leq M$ for some real number M and returns in time $n^{O(k)}$ a degree-k pseudoexpectation $\widetilde{\mathbb{E}}$ which satisfies $\mathcal{A}$ up to error 2^{-n}.*

Note that there are a few technical details with regards to only being able to compute a pseudoexpectation that nearly satisfies the constraints. These technicalities do not affect our proof (as 2^{-n} errors will be negligible) so we will simply assume that we can compute a pseudoexpectation that exactly satisfies the constraints. See [9] for more details about these technicalities.

Finally, we state a few simple inequalities for pseudoexpectations that will be used repeatedly later on.

Claim 2.6 (Cauchy Schwarz for Pseudo-distributions). *Let f, g be polynomials of degree at most k in indeterminates $x = (x_1, \dots, x_n)$. Then for any degree k pseudoexpectation,*

$$\widetilde{\mathbb{E}}[fg] \leq \sqrt{\widetilde{\mathbb{E}}[f^2]} \sqrt{\widetilde{\mathbb{E}}[g^2]}.$$

Corollary 2.7. *Let $f_1, g_1, \dots, f_m, g_m$ be polynomials of degree at most k in indeterminates $x = (x_1, \dots, x_n)$. Then for any degree k pseudoexpectation,*

$$\widetilde{\mathbb{E}}[f_1 g_1 + \cdots + f_m g_m] \leq \sqrt{\widetilde{\mathbb{E}}[f_1^2 + \cdots + f_m^2]}\sqrt{\widetilde{\mathbb{E}}[g_1^2 + \cdots + g_m^2]}.$$

Proof. Note

$$\widetilde{\mathbb{E}}[f_1 g_1 + \cdots + f_m g_m] \leq \sqrt{\widetilde{\mathbb{E}}[f_1^2]}\sqrt{\widetilde{\mathbb{E}}[g_1^2]} + \cdots + \sqrt{\widetilde{\mathbb{E}}[f_m^2]}\sqrt{\widetilde{\mathbb{E}}[g_m^2]} \leq \sqrt{\widetilde{\mathbb{E}}[f_1^2 + \cdots + f_m^2]}\sqrt{\widetilde{\mathbb{E}}[g_1^2 + \cdots + g_m^2]}$$

where the first inequality follows from Cauchy Schwarz for pseudoexpectations and the second follows from standard Cauchy Schwarz. □

3 FUN WITH GENERATING FUNCTIONS

We now introduce the generating function machinery that we will use in our learning algorithm. We begin with a standard definition.

Definition 3.1. Let $\mathcal{H}_m(x)$ be the univariate Hermite polynomials $\mathcal{H}_0 = 1, \mathcal{H}_1 = x, \mathcal{H}_2 = x^2 - 1 \cdots$ defined by the recurrence

$$\mathcal{H}_m(x) = x\mathcal{H}_{m-1}(x) - (m-1)\mathcal{H}_{m-2}(x)$$

Note that in $\mathcal{H}_m(x)$, the degree of each nonzero monomials has the same parity as m. In light of this, we can write the following:

Definition 3.2. Let $\mathcal{H}_m(x, y^2)$ be the homogenized Hermite polynomials e.g., $\mathcal{H}_2(x, y^2) = x^2 - y^2, \mathcal{H}_3(x, y^2) = x^3 - 3xy^2$.

It will be important to note the following fact:

Claim 3.3. *We have*

$$e^{xz - \frac{1}{2}y^2 z^2} = \sum_{m=0}^{\infty} \frac{1}{m!}\mathcal{H}_m(x, y^2) z^m$$

where the RHS is viewed as a formal power series in z whose coefficients are polynomials in x, y.

Now we define a multivariate version of the Hermite polynomials.

Definition 3.4. Let $H_m(X, z)$ be a formal polynomial in variables $X = X_1, \dots, X_d$ whose coefficients are polynomials in d variables $z_1, \dots, z_d$ that is given by

$$H_m(X, z) = \mathcal{H}_m(z_1 X_1 + \cdots + z_d X_d, X_1^2 + \cdots + X_d^2)$$

Note that H_m is homogeneous of degree m as a polynomial in $X_1, \dots, X_d$.

Definition 3.5. For a distribution D on $\mathbb{R}^d$, we let

$$h_{m,D}(X) = \mathbb{E}_{(z_1, \dots, z_d) \sim D}[H_m(X, z)]$$

where we take the expectation of H_m over $(z_1, \dots, z_d)$ drawn from D. Note that $h_{m,D}(X)$ is a polynomial in $(X_1, \dots, X_d)$. We will omit the D in the subscript when it is clear from context. Moreover, for a mixture of Gaussians

$$\mathcal{M} = w_1 N(\mu_1, \Sigma_1) + \dots w_k N(\mu_k, \Sigma_k)$$

we will refer to the Hermite polynomials $h_{m,\mathcal{M}}$ as the Hermite polynomials of the mixture.

We remark that if there is a mixture $\mathcal{M} = w_1 N(\mu_1, \Sigma_1) + \ldots w_k N(\mu_k, \Sigma_k)$ where instead of real numbers, the w_i, μ_i, Σ_i are given in terms of indeterminates, the Hermite polynomials will be polynomials in those indeterminates. We will repeatedly make use of this abstraction later on.

The first important observation is that the Hermite polynomials for Gaussians can be written in a simple closed form via generating functions.

Claim 3.6. *Let $D = N(\mu, I + \Sigma)$. Let $a(X) = \mu \cdot X$ and $b(X) = X^T \Sigma X$. Then*

$$e^{a(X)y + \frac{1}{2}b(X)y^2} = \sum_{m=0}^{\infty} \frac{1}{m!} \cdot h_{m,D}(X) y^m$$

as formal power series in y.

Proof. By Claim 3.3, we have

$$e^{a(X)y + \frac{1}{2}b(X)y^2} = \sum_{m=0}^{\infty} \frac{1}{m!} \mathcal{H}_m(a(X), -b(X)) y^m$$

It now suffices to verify that

$$\mathbb{E}_{(z_1, \ldots z_d) \sim D} \left[\mathcal{H}_m(z_1 X_1 + \cdots + z_d X_d, X_1^2 + \cdots + X_d^2) \right] = \mathcal{H}_m(a(X), -b(X))$$

This can be verified through straight-forward computations using the moment tensors of a Gaussian (see Lemma 2.7 in [39]). □

We now have two simple corollaries to the above.

Corollary 3.7. *Let $\mathcal{M} = w_1 N(\mu_1, I + \Sigma_1) + \ldots w_k N(\mu_k, I + \Sigma_k)$. Let $a_i(X) = \mu_i \cdot X$ and $b_i(X) = X^T \Sigma_i X$. Then*

$$\sum_{m=0}^{\infty} \frac{1}{m!} \cdot h_{m,\mathcal{M}}(X) y^m = w_1 e^{a_1(X)y + \frac{1}{2}b_1(X)y^2} + \cdots + w_k e^{a_k(X)y + \frac{1}{2}b_k(X)y^2}$$

Corollary 3.8. *Let $\mathcal{M} = w_1 N(\mu_1, I + \Sigma_1) + \ldots w_k N(\mu_k, I + \Sigma_k)$. Let $a_i(X) = \mu_i \cdot X$ and $b_i(X) = X^T \Sigma_i X$. Then the Hermite polynomials $h_{m,\mathcal{M}}(X)$ can be written as a linear combination of products of the $a_i(X), b_i(X)$ such that the number of terms in the sum, the number of terms in each product, and the coefficients in the linear combination are all bounded as functions of m, k.*

The next important insight is that the generating functions for the Hermite polynomials behave nicely under certain differential operators. We can use these differential operators to derive identities that the Hermite polynomials must satisfy and these identities will be a crucial ingredient in our learning algorithm.

The proceeding claims all follow from direct computation.

Claim 3.9. *Let ∂ denote the differential operator with respect to y. If*

$$f(y) = P(y, X) e^{a(X)y + \frac{1}{2}b(X)y^2}$$

where P is a polynomial in y of degree k (whose coefficients are polynomials in X) then

$$(\partial - (a(X) + yb(X))) f(y) = Q(y, X) e^{a(X)y + \frac{1}{2}b(X)y^2}$$

where Q is a polynomial in y with degree exactly $k - 1$ whose leading coefficient is k times the leading coefficient of P.

Corollary 3.10. *Let ∂ denote the differential operator with respect to y. If*

$$f(y) = P(y, X)e^{a(X)y + \frac{1}{2}b(X)y^2}$$

where P is a polynomial in y of degree k then

$$(\partial - (a(X) + yb(X)))^{k+1} f(y) = 0.$$

Claim 3.11. *Let ∂ denote the differential operator with respect to y. Let*

$$f(y) = P(y, X)e^{a(X)y + \frac{1}{2}b(X)y^2}$$

where P is a polynomial in y of degree k. Let the leading coefficient of P (viewed as a polynomial in y in the monomial basis) be $L(X)$. Let $c(X), d(X)$ be a linear and quadratic polynomial in the X variables, respectively, such that $\{a(X), b(X)\} \neq \{c(X), d(X)\}$. If $b(X) \neq d(X)$ then

$$(\partial - (c(X) + yd(X)))^{k'} f(y) = Q(y, X)e^{a(X)y + \frac{1}{2}b(X)y^2}$$

where Q is a polynomial of degree $k + k'$ in y with leading coefficient (viewed in the monomial basis)

$$L(x)(b(X) - d(X))^{k'}$$

and if $b(X) = d(X)$ then

$$(\partial - (c(X) + yd(X)))^{k'} f(y) = Q(y, X)e^{a(X)y + \frac{1}{2}b(X)y^2}$$

where Q is a polynomial of degree k in y with leading coefficient (viewed in the monomial basis)

$$L(X)(a(X) - c(X))^{k'}.$$

3.1 Polynomial Factorizations

The analysis of our SOS-based learning algorithm will rely on manipulations of Hermite polynomials. An important piece of our analysis is understanding how the coefficients of polynomials behave under addition and (polynomial) multiplication. Specifically, if we have two polynomials $f(X), g(X)$ and we have bounds on the coefficients of f and g, we now want to give bounds on the coefficients of the polynomials $f(X) + g(X)$ and $f(X)g(X)$. Most of these bounds are easy to obtain. The one that is somewhat nontrivial is lower bounding the coefficients of $f(X)g(X)$ i.e., if the coefficients of f and g are not all small, then the product $f(X)g(X)$ cannot have all of its coefficients be too small.

Definition 3.12. For a polynomial $f(X)$ in the d variables $X_1, \dots, X_d$ with real coefficients define $v(f)$ to be the vectorization of the coefficients. (We will assume this is done in a consistent manner so that the same coordinate of vectorizations of two polynomials corresponds to the coefficient of the same monomial.) We will frequently consider expressions of the form $\|v(f)\|$ i.e., the L^2 norm of the coefficient vector.

Definition 3.13. For a polynomial $A(X)$ of degree k in d variables $X_1, \dots, X_d$ and a vector $v \in \mathbb{R}^d$ with nonnegative integer entries summing to at most k, we use A_v to denote the corresponding coefficient of A.

First, we prove a simple result about the norm of the vectorization of a sum of polynomials.

Claim 3.14. *Let $f_1, \dots, f_m$ be polynomials in $X_1, \dots, X_d$ whose coefficients are polynomials in formal variables $u_1, \dots, u_n$ of degree $O_k(1)$. Then*

$$\|v(f_1 + \cdots + f_m)\|^2 \leq m(\|v(f_1)\|^2 + \cdots + \|v(f_m)\|^2)$$

Furthermore, the difference can be written as a sum of squares of polynomials of degree $O_k(1)$ in $u_1, \dots, u_n$.

Proof. Note

$$(a_1 + \cdots + a_m)^2 \leq m\left(a_1^2 + \cdots + a_m^2\right)$$

and the difference between the two sides can be written as a sum of squares

$$\sum_{i \neq j} (a_i - a_j)^2$$

The desired inequality can now be obtained by summing expressions of the above form over all coefficients. □

Next, we upper bound the norm of the vectorization of a product of polynomials.

Claim 3.15. *Let $f, g, h_1, \ldots, h_k$ be polynomials in $X_1, \ldots, X_d$ of degree at most k with coefficients that are polynomials in formal variables $u_1, \ldots, u_n$ of degree $O_k(1)$ Then for any pseudoexpectation $\widetilde{\mathbb{E}}$ of degree C_k for some sufficiently large constant C_k depending only on k,*

$$\widetilde{\mathbb{E}}[\|v(h_1)\|^2 \ldots \|v(h_k)\|^2 \|v(fg)\|^2] \leq O_k(1)\widetilde{\mathbb{E}}[\|v(h_1)\|^2 \ldots \|v(h_k)\|^2 \|v(f)\|^2 \|v(g)\|^2]$$

where the pseudoexpectation operates on polynomials in $u_1, \ldots, u_n$.

Proof. Note that each monomial in the product fg has degree at most $2k$ and thus can only be split in $O_k(1)$ ways. Specifically, each entry of $v(fg)$ can be written as a sum of $O_k(1)$ entries of $v(f) \otimes v(g)$ so

$$\|v(fg)\|^2 \leq O_k(1) \|v(f)\|^2 \|v(g)\|^2$$

where the difference between the two sides can be written as a sum of squares. This implies the desired inequality. □

Before we prove the final result in this section, we introduce a few definitions.

Definition 3.16. For a vector $v \in \mathbb{R}^d$ with integer coordinates, we define $\tau(v)$ to be the multiset formed by the coordinates of v. We call τ the type of v.

Definition 3.17. For a monomial say $X_1^{a_1} \ldots X_d^{a_d}$, we call $(a_1, \ldots, a_d) \in \mathbb{R}^d$ its degree vector.

Now we can prove a lower bound on the norm of the vectorization of the product of polynomials.

Claim 3.18. *Let $f, g, h_1, \ldots, h_k$ be polynomials in $X_1, \ldots, X_d$ of degree at most k with coefficients that are polynomials in formal variables $u_1, \ldots, u_n$ of degree $O_k(1)$. Then for any pseudoexpectation $\widetilde{\mathbb{E}}$ of degree C_k for some sufficiently large constant C_k depending only on k,*

$$\widetilde{\mathbb{E}}[\|v(h_1)\|^2 \ldots \|v(h_k)\|^2 \|v(fg)\|^2] \geq \Omega_k(1)\widetilde{\mathbb{E}}[\|v(h_1)\|^2 \ldots \|v(h_k)\|^2 \|v(f)\|^2 \|v(g)\|^2]$$

where the pseudoexpectation operates on polynomials in $u_1, \ldots, u_n$.

Proof. We will first prove the statement for $h_1 = \cdots = h_k = 1$.

Let S be the set of all types that can be obtained by taking the sum of two degree vectors for monomials of degree at most k and let T be the set of all types that can be obtained by taking the difference of two degree vectors for monomials of degree at most k. Note that $|S|, |T| = O_k(1)$. Now

$$\widetilde{\mathbb{E}}[\|v(fg)\|^2] = \widetilde{\mathbb{E}}\left[\sum_a \left(\sum_{u+v=a} f_u g_v\right)^2\right] = \widetilde{\mathbb{E}}\left[\sum_{u_1+v_1=u_2+v_2} f_{u_1} g_{v_1} f_{u_2} g_{v_2}\right]$$
$$= \widetilde{\mathbb{E}}\left[\sum_{u_1-v_2=u_2-v_1} f_{u_1} g_{v_1} f_{u_2} g_{v_2}\right] = \widetilde{\mathbb{E}}\left[\sum_b \left(\sum_{u-v=b} f_u g_v\right)^2\right]$$

where the sums in the above expression are over all a and all b that are vectors in $\mathbb{Z}^d$ for which the inner summands are nonempty. Let $T = \{t_1, \dots, t_n\}$ where the types $t_1, \dots, t_n$ are sorted in non-increasing order of their L^2 norm. Recall that T consists of all types that can be obtained by taking the difference of two degree vectors corresponding to monomials of degree at most k. Now first note

$$\widetilde{\mathbb{E}}[\|v(fg)\|^2] \geq \widetilde{\mathbb{E}}\left[\sum_{b,\,\tau(b)=t_1}\left(\sum_{u-v=b} f_u g_v\right)^2\right] = \widetilde{\mathbb{E}}\left[\sum_{b,\,\tau(b)=t_1}\left(\sum_{u-v=b} (f_u g_v)^2\right)\right]$$

since t_1 corresponds to the type $(k, -k)$ and each of the inner summands only contains one term. Now consider t_i for $i > 1$.

$$\widetilde{\mathbb{E}}\left[\sum_{b,\,\tau(b)=t_i}\left(\sum_{u-v=b} f_u g_v\right)^2\right] = \widetilde{\mathbb{E}}\left[\sum_{b,\,\tau(b)=t_i}\left(\sum_{u-v=b} (f_u g_v)^2\right)\right] + 2\widetilde{\mathbb{E}}\left[\sum_{b,\,\tau(b)=t_i}\left(\sum_{\substack{\{u_1,v_1\}\neq\{u_1,v_2\}\\ u_1-v_1=u_2-v_2=b}} f_{u_1} f_{u_2} g_{v_1} g_{v_2}\right)\right]$$

Note that in the second sum, either $u_1 - v_2 \in t_j$ for $j < i$ or $u_2 - v_1 \in t_j$ for $j < i$. To see this, let $a = v_1 - v_2$. Then $u_1 - v_2 = b + a$ and $u_2 - v_1 = b - a$. Now

$$\|b - a\|_2^2 + \|b + a\|_2^2 > \|b\|_2^2$$

since $a \neq 0$ so one of the differences must be of an earlier type.

Next, note that for a fixed u_1, v_2, there are at most $O_k(1)$ possible values for u_2', v_1' such that the term $f_{u_1} f_{u_2'} g_{v_1'} g_{v_2}$ appears. This is because we must have $u_1 + v_2 = u_2' + v_1'$ and there are only $O_k(1)$ ways to achieve this. Thus, by Cauchy Schwarz

$$\widetilde{\mathbb{E}}\left[\sum_{b,\,\tau(b)=t_i}\left(\sum_{u-v=b} f_u g_v\right)^2\right] \geq \widetilde{\mathbb{E}}\left[\sum_{b,\,\tau(b)=t_i}\left(\sum_{u-v=b} (f_u g_v)^2\right)\right]$$
$$-O_k(1)\sqrt{\widetilde{\mathbb{E}}\left[\sum_{j<i}\sum_{b,\,\tau(b)=t_j}\left(\sum_{u-v=b} (f_u g_v)^2\right)\right]} \cdot \sqrt{\widetilde{\mathbb{E}}[\|v(f)\|^2\,\|v(g)\|^2]}$$

Now combining the above with the fact that $|T| = n = O_k(1)$ and that

$$\widetilde{\mathbb{E}}[\|v(f)\|^2\,\|v(g)\|^2] = \widetilde{\mathbb{E}}\left[\sum_{i=1}^{n}\sum_{b,\,\tau(b)=t_i}\left(\sum_{u-v=b} (f_u g_v)^2\right)\right]$$

we can complete the proof. To see this, for each i, let

$$Q_i = \widetilde{\mathbb{E}}\left[\sum_{b,\,\tau(b)=t_i}\left(\sum_{u-v=b} (f_u g_v)^2\right)\right]$$
$$R_i = \widetilde{\mathbb{E}}\left[\sum_{b,\,\tau(b)=t_i}\left(\sum_{u-v=b} f_u g_v\right)^2\right]$$

Also, normalize so that

$$\widetilde{\mathbb{E}}[\|v(f)\|^2\,\|v(g)\|^2] = 1.$$

Let δ be some suitably chosen constant depending only on k. If $Q_1 \geq \delta$ then we are done. Otherwise, we have an upper bound on the square root terms that are subtracted in the expression for R_2. If $Q_2 \geq \Omega_1(k)\sqrt{\delta}$ then we are again done (since we have now reduced to the case where $Q_1 \leq \delta$). Iteratively repeating this procedure, we are done whenever one of the Q_i is sufficiently large compared to $Q_1, \dots, Q_{i-1}$. However, not all of $Q_1, \dots, Q_n$ can be small since their sum is 1. Choosing δ to be a sufficiently small constant but depending only on k we conclude that

$$\widetilde{\mathbb{E}}[\|v(fg)\|^2] \geq \Omega_k(1)\widetilde{\mathbb{E}}[\|v(f)\|^2 \|v(g)\|^2]$$

as desired.

For the general case when not all of the h_i are 1, we can multiply the insides of all of the pseudoexpectations above by $\|v(h_1)\|^2 \dots \|v(h_k)\|^2$ and the same argument will work. □

4 COMPONENTS ARE NOT FAR APART

Now we are ready to present our main contribution: an algorithm that learns the parameters of a mixture of Gaussians $\mathcal{M} = w_1G_1 + \dots + w_kG_k$ from an ϵ-corrupted sample when the components are not too far apart. In this section, we will assume that the mixture is in nearly isotropic position and that we have estimates for the Hermite polynomials. We will show how to learn the parameters from these estimates. In the next section, Section 5, we show how to actually place the mixture in isotropic position and obtain estimates for the Hermite polynomials.

We use the following conventions:

- The true means and covariances are given by $(\mu_1, I + \Sigma_1), \dots, (\mu_k, I + \Sigma_k)$
- The true mixing weights are $w_1, \dots, w_k$ and are all bounded below by some value $w_{\min}$
- Δ is an upper bound that we have on $\|\mu_i\|$ and $\|\Sigma_i\|$ i.e., the components are not too far separated.
- $||\mu_i - \mu_j||_2 + ||\Sigma_i - \Sigma_j||_2 \geq c$ for all $i \neq j$ i.e., no pair of components is too close
- We should think of $w_{\min}, c$ as being at least ϵ^r and Δ being at most ϵ^{-r} for some sufficiently small value of $r > 0$.
- Let the Hermite polynomials for the true mixture be given by $h_1 = h_{1,\mathcal{M}}, h_2 = h_{2,\mathcal{M}}, \dots$ where
$$\mathcal{M} = w_1N(\mu_1, I + \Sigma_1) + \dots + w_kN(\mu_k, I + \Sigma_k)$$

In this section we assume that we have the following:

- Estimates $\overline{h}_i(X)$ for the Hermite polynomials such that $||v(\overline{h_i}(X) - h_i(X))||^2 \leq \epsilon' = \text{poly}(\epsilon)$

and our only interaction with the actual samples is through these estimates. We will show how to obtain these estimates in Section 5 (closely mirroring the method in [39]).

The main theorem that we prove in this section is as follows.

Theorem 4.1. *Let ϵ' be a parameter that is sufficiently small in terms of k. There is a sufficiently small function $f(k)$ and a sufficiently large function $F(k)$ such that if*

$$\mathcal{M} = w_1N(\mu_1, I + \Sigma_1) + \dots + w_kN(\mu_k, I + \Sigma_k)$$

is a mixture of Gaussians with

- $\|\mu_i\|_2, \|\Sigma_i\|_2 \leq \Delta$ *for all i*
- $||\mu_i - \mu_j||_2 + ||\Sigma_i - \Sigma_j||_2 \geq c$ *for all $i \neq j$*
- $w_1, \dots, w_k \geq w_{\min}$

for parameters $w_{\min}, c \geq (\epsilon')^{f(k)}$ and $\Delta \leq (\epsilon')^{-f(k)}$ and we are given estimates $\overline{h}_i(X)$ for the Hermite polynomials for all $i \leq F(k)$ such that

$$\left\|v(\overline{h_i}(X) - h_i(X))\right\|^2 \leq \epsilon'$$

where h_i are the Hermite polynomials for the true mixture $\mathcal{M}$, then there is an algorithm that returns $\mathrm{poly}(1/\epsilon')^{O_1(k)}$ *candidate mixtures, at least one of which satisfies*

$$\|w_i - \widetilde{w_i}\| + \|\mu_i - \widetilde{\mu_i}\|_2 + ||\Sigma_i - \widetilde{\Sigma_i}||_2 \leq (\epsilon')^{f(k)}$$

for all i.

Informally, assuming that the parameters of the components of the mixture are bounded by $\mathrm{poly}(1/\epsilon')$ and that their separation is at least $\mathrm{poly}(\epsilon')$, given ϵ'-accurate estimates for the Hermite polynomials, we can learn the parameters of the mixture to within Frobenius error $\mathrm{poly}(\epsilon')$.

4.1 Reducing to All Pairs of Parameters Equal or Separated

We claim that it suffices to work under the following assumption. All pairs of parameters are either separated of equal. More specifically, for each pair of parameters μ_i, μ_j (and same for Σ_i, Σ_j), either $\mu_i = \mu_j$ or

$$||\mu_i - \mu_j||_2 \geq c$$

We now prove that it suffices to work with the above simplification. For any function $0 < f(k) < 1$ depending only on k, there is some $C \geq (f(k))^{k^2}$ such that there is no pair of parameters μ_i, μ_j or Σ_i, Σ_j whose distance is in the interval $[(\epsilon')^C, (\epsilon')^{f(k)C}]$. Now consider the graph on the k nodes where i, j are connected if and only if

$$||\mu_i - \mu_j|| \leq (\epsilon')^{f(k)C}$$

We now construct a new mixture $N(\mu_i', \Sigma_i')$. For each connected component say $\{i_1, \dots, i_j\}$, pick a representative and set $\mu'_{i_1} = \mu'_{i_2} = \cdots = \mu'_{i_j} = \mu_{i_1}$. Do this for all connected components and similar in the graph on covariance matrices. For all i, we have

$$\|\mu_i' - \mu_i\|, \|\Sigma_i' - \Sigma_i\| \leq O_k(1)(\epsilon')^C$$

because there is a path of length at most k connecting i to the representative in its component that it is rounded to, and all edges correspond to pairs within distance of $(\epsilon')^C$.

The Hermite polynomials of this new mixture satisfy

$$\|v(h_m' - h_m)\|^2 \leq O_k(1)\Delta^{O_k(1)}(\epsilon')^{\Omega_k(1)C}$$

as long as m is bounded as a function of k by Corollary 3.7. If we pretend that the new mixture is the true mixture, we have estimates $\overline{h}_m(X)$ such that

$$\left\|v(h_m' - \overline{h_m})\right\|^2 \leq O_k(1)\Delta^{O_k(1)}(\epsilon')^{\Omega_k(1)C}$$

and all pairs of parameters in the new mixture are either equal or $(\epsilon')^{f(k)C}$ separated. If we prove Theorem 4.1 with the assumption that the pairs of parameters are separated or equal, then we can choose $f(k)$ accordingly and then we deduce that the theorem holds in the general case (with worse, but still polynomial, bounds on $\Delta, c, w_{\min}$ and the accuracy of our output as a function of ϵ').

From now on we will work with the assumption that each pair of parameters is either equal or separated by c.

4.2 SOS Program Setup

Our algorithm for learning the parameters when given estimates of the Hermite polynomials involves solving an SOS program. Here we set up the SOS program that we will solve.

We will let $D = \binom{d}{2} + d$. We think of mapping between symmetric $d \times d$ matrices and $\mathbb{R}^D$ as

$$\begin{bmatrix} x_{11} & \dots & x_{1d} \\ \vdots & \ddots & \vdots \\ x_{d1} & \dots & x_{dd} \end{bmatrix} \leftrightarrow (x_{11}, 2x_{12}, 2x_{13}, \dots, x_{dd})$$

We will solve for an orthonormal basis for the span of the means μ_i and the span of the (vectorized) covariances Σ_i. We will have SOS variables for the elements of this basis. Note that regardless of the basis, since it is k-dimensional, we can brute-force search over the representations of the true means μ_i and covariances Σ_i in terms of elements of the basis. To set up our SOS-program, we will first guess these coefficients (which are real numbers) and then set up the SOS-program to solve for the desired bases i.e., there is a distinct program for each guess of the coefficients.

Definition 4.2 (Parameter Solving Program $\mathcal{S}$). We will have the following variables

- $u_1 = (u_{11}, \dots, u_{1d}), \dots, u_k = (u_{k1}, \dots, u_{kd})$
- $v_1 = (v_{1,(1,1)}, v_{1,(1,2)}, \dots, v_{1,(d,d)}), \dots, v_k = (v_{k,(1,1)}, v_{k,(1,2)}, \dots, v_{k,(d,d)})$

In the above $u_1, \dots, u_k \in \mathbb{R}^d$ and $v_1, \dots v_k \in \mathbb{R}^D$. Our goal will be to solve for these variables in a way so that the solutions form orthonormal bases for the span of the μ_i and the span of the Σ_i. Note $v_1, \dots, v_k$ live in $\mathbb{R}^D$ because the Σ_i must be symmetric.

We guess coefficients a_{ij}, b_{ij} where $i, j \in [k]$ expressing the means and covariances in this orthonormal basis. We ensure that the guesses satisfy the property that for every pair of vectors $A_i = (a_{i1}, \dots, a_{ik}), A_j = (a_{j1}, \dots, a_{jk})$ either $A_i = A_j$ or

$$||A_i - A_j||_2 \geq \frac{c}{2}$$

and similarly for B_i, B_j. We ensure that

$$||A_i||_2 \leq 2\Delta$$

Ensure similar conditions for the $\{B_i\}$. We also guess the mixing weights $\widetilde{w_1}, \dots, \widetilde{w_k}$ and ensure that our guesses are all at least $w_{\min}/2$. Note that these are all real numbers and not variables in the SOS program.

Now we set up the constraints. Let C be a sufficiently large integer depending only on k. Define $\widetilde{\mu_i} = a_{i1}u_1 + \dots + a_{ik}u_k$ and define $\widetilde{\Sigma_i}$ similarly. These are linear expressions in the variables that we are solving for (and not new variables in the SOS program). Now consider the hypothetical mixture with mixing weights $\widetilde{w_i}$, means $\widetilde{\mu_i}$, and covariances $I + \widetilde{\Sigma_i}$. The Hermite polynomials for this hypothetical mixture $\widetilde{h_i}(X)$ can be written as formal polynomials in $X = (X_1, \dots, X_d)$ with coefficients that are polynomials in u, v. Note that we can explicitly write down these Hermite polynomials. The set of constraints for our SOS system is as follows:

- $\|u_i\|_2^2 = 1$ for all $1 \leq i \leq k$
- $\|v_i\|_2^2 = 1$ for all $1 \leq i \leq k$
- $u_i \cdot u_j = 0$ for all $i \neq j$
- $v_i \cdot v_j = 0$ for all $i \neq j$
- For all $p = 1, 2, \dots, C$

$$\left\| v(\widetilde{h_p}(X) - \overline{h_p}(X)) \right\|^2 \leq 100\epsilon'$$

Note that we can explicitly write down the last set of constraints because we have estimates $\overline{h_i}$.

It is important to note that the $\widetilde{w_i}, A_i, B_i$ are real numbers. We will attempt to solve the system for each of our guesses and show that for some set of guesses, we obtain a solution from which we can recover the parameters. We can brute-force search over an ϵ'-net because there are only $O_k(1)$ parameters to guess. We call the SOS program that we set up $\mathcal{S}$.

4.3 Analysis

We now prove a set of properties that must be satisfied by any pseudoexpectation of degree C_k satisfying $\mathcal{S}$ where C_k is a sufficiently large constant depending only on k. What we would ideally want to show is that

- The span of the $\widetilde{\Sigma_i}$ is close to the span of the Σ_i
- The span of the $\widetilde{\mu_i}$ is close to the span of the μ_i

However, it appears to be difficult to prove a statement of the above form within an SOS framework. Instead, we will look at the pseudoexpectations of the matrices

$$M_i = \widetilde{\mathbb{E}}\left[\widetilde{\Sigma_i}\widetilde{\Sigma_i}^T\right]$$

(where $\widetilde{\Sigma_i}$ is viewed as a length-D vector so $\widetilde{\Sigma_i}\widetilde{\Sigma_i}^T$ is a $D \times D$ matrix.) The two key properties that we will prove about these matrices are in Lemmas 4.11 and 4.12.

Roughly Lemma 4.11 says that any singular vector that corresponds to a large singular value of M_i must be close to the span of the $\{\Sigma_i\}$. Lemma 4.12 says that any vector v that has large projection onto the subspace spanned by the $\{\Sigma_i\}$ must have the property that $v^T M_i v$ is large for some i. Putting these together, we can take the the top-k principal components of each of $M_1, \ldots, M_k$ and show that the span of these essentially contains the span of the $\{\Sigma_i\}$ (this last step is done outside the SOS framework). We can now brute-force over an ϵ'-net and guess the Σ_i (since we have narrowed them down to an $O_k(1)$-dimensional subspace). We can then plug in real values for the covariances and solve for the means using a similar method.

4.3.1 *Algebraic Identities.*

First we will prove several purely algebraic identities. We will slightly abuse notation and for $\mu \in \mathbb{R}^d$, we use $\mu(X)$ to denote the inner product of μ with the formal variables $(X_1, \ldots, X_d)$ and for $\Sigma \in \mathbb{R}^D$, we will use $\Sigma(X)$ to denote the quadratic form in formal variables $(X_1, \ldots, X_d)$ given by $X^T \Sigma X$ (when Σ is converted to a symmetric $d \times d$ matrix). It will be useful to consider the following two formal power series (in y)

$$F(y) = \sum_{i=1}^{k} w_i e^{\mu_i(X)y + \frac{1}{2}\Sigma_i(X)y^2}$$

$$\widetilde{F}(y) = \sum_{i=1}^{k} \widetilde{w_i} e^{\widetilde{\mu_i}(X)y + \frac{1}{2}\widetilde{\Sigma_i}(X)y^2}$$

We view these objects in the following way: the coefficients of $1, y, y^2, \cdots$ are formal polynomials in $(X_1, \ldots, X_d)$. In the first expression, the coefficients of these polynomials are (unknown) constants. In the second, the coefficients are polynomials in the variables $u_1, \ldots, u_k, v_1, \ldots, v_k$. In fact, the coefficients in the first power series are precisely $h_1, h_2, \ldots$ while the coefficients in the second power series are precisely $\widetilde{h_1}, \widetilde{h_2}, \ldots$. The key insight is the following:

> *After taking derivatives and polynomial combinations of either of the above formal power series, the coefficients can still be expressed as polynomial combinations of their respective Hermite polynomials.*

Definition 4.3. Let $\mathcal{D}_i$ denote the differential operator $(\partial - (\mu_i(X) + \Sigma_i(X)y))$ and $\widetilde{\mathcal{D}_i}$ denote the differential operator $(\partial - (\widetilde{\mu_i}(X) + \widetilde{\Sigma_i}(X)y))$. As usual, the partial derivatives are taken with respect to y.

To simplify the exposition, we make the following definition:

Definition 4.4. Consider a polynomial $P(X)$ that is a formal polynomial in $X_1, \dots, X_d$ whose coefficients are polynomials in the indeterminates $u_1, \dots, u_k, v_1, \dots, v_k$. We say P is m-simple if P can be written as a linear combination of a constant number of terms that are a product of some of $\{\mu_i(X)\}, \{\Sigma_i(X)\}, \{\widetilde{\mu_i}(X)\}, \{\widetilde{\Sigma_i}(X)\}$ where

(1) The coefficients in the linear combination are bounded by a constant depending only on m, k
(2) The number of terms in the sum depends only on m and k
(3) The number of terms in each product depends only on m and k.

Claim 4.5. *Consider the power series*

$$\widetilde{\mathcal{D}_{k-1}}^{2^{2k-2}} \dots \widetilde{\mathcal{D}_1}^{2^k} \mathcal{D}_k^{2^{k-1}} \dots \mathcal{D}_1^1(\widetilde{F})$$

For any m, the coefficient of y^m when the above is written as a formal power series can be written in the form

$$P_0(X) + P_1(X)\widetilde{h_1}(X) + \dots + P_{m'}(X)\widetilde{h_{m'}}(X)$$

where

- *m' depends only on m and k*
- *Each of the P_i is m-simple*
- *We have*
$$P_0(X) + P_1(X)h_1(X) + \dots + P_{m'}(X)h_{m'}(X) = 0$$
as an algebraic identity over formal variables $X_1, \dots, X_d, \{u_i\}, \{v_i\}$.

Proof. Note the coefficients of $\widetilde{F}$ (as a formal power series in y) are exactly given by the $\widetilde{h_i}$. Now the number of differential operators we apply is $O_k(1)$. The first two statements can be verified through straightforward computations since when applying each of the differential operators, we are simply multiplying the coefficients by some of $\{\mu_i(X)\}, \{\Sigma_i(X)\}, \{\widetilde{\mu_i}(X)\}, \{\widetilde{\Sigma_i}(X)\}$ and taking a linear combination. Next, note that by Corollary 3.10

$$\mathcal{D}_k^{2^{k-1}} \dots \mathcal{D}_1^1(F) = 0.$$

To see this, we prove by induction that the differential operator

$$\mathcal{D}_j^{2^{j-1}} \dots \mathcal{D}_1^1(F)$$

annihilates the components of F corresponding to Gaussians $N(\mu_1, I + \Sigma_1), \dots, N(\mu_j, I + \Sigma_j)$. The base case is clear. To complete the induction step, note that by Corollary 3.10, the above operator puts polynomials of degree at most $1 + 2 + \dots + 2^{j-1} = 2^j - 1$ in front of the other components. Thus, the operator

$$\mathcal{D}_{j+1}^{2^j} \dots \mathcal{D}_1^1(F)$$

annihilates the first $j + 1$ components, completing the induction. We now conclude that

$$\widetilde{\mathcal{D}_{k-1}}^{2^{2k-2}} \dots \widetilde{\mathcal{D}_1}^{2^k} \mathcal{D}_k^{2^{k-1}} \dots \mathcal{D}_1^1(F) = 0$$

implying that if the coefficients of $\widetilde{F}$ were $h_1, \dots, h_m$, then the result would be identically zero. □

Claim 4.6. *Consider the power series*

$$\mathcal{D}_{k-1}^{2^{2k-2}} \dots \mathcal{D}_1^{2^k} \widetilde{\mathcal{D}_k}^{2^{k-1}} \dots \widetilde{\mathcal{D}_1}^1(F)$$

For any m, the coefficient of y^m when the above is written as a formal power series can be written in the form

$$P_0(X) + P_1(X)h_1(X) + \cdots + P_{m'}(X)h_{m'}(X)$$

where

- *m' depends only on m and k*
- *Each of the P_i is m-simple*
- *We have*
$$P_0(X) + P_1(X)\widetilde{h_1}(X) + \cdots + P_{m'}(X)\widetilde{h_{m'}}(X) = 0$$
as an algebraic identity over formal variables $X_1, \dots, X_d, \{u_i\}, \{v_i\}$.

Proof. The proof is identical to the proof of Claim 4.5. □

Note that the polynomials P_i in Claims 4.5 and 4.6 are *not* necessarily the same.

4.3.2 Warm-up: All Pairs of Parameters are Separated. As a warm-up, we first analyze the case where all pairs of true parameters μ_i, μ_j and Σ_i, Σ_j satisfy $||\mu_i - \mu_j||_2 \geq c$ and $||\Sigma_i - \Sigma_j||_2 \geq c$. We will show how to deal with the general case where parameters may be separated or equal in Section 4.3.4.

We can assume that our guesses satisfy $||A_i - A_j||_2 \geq c/2$ and $||B_i - B_j||_2 \geq c/2$ for all i, j. The key expressions to consider are applying the following differential operators

$$\mathcal{D} = \widetilde{\mathcal{D}_k}^{2^{2k-1}-1}\widetilde{\mathcal{D}_{k-1}}^{2^{2k-2}} \dots \widetilde{\mathcal{D}_1}^{2^k}\mathcal{D}_k^{2^{k-1}} \dots \mathcal{D}_1^1$$
$$\widetilde{\mathcal{D}} = \mathcal{D}_k^{2^{2k-1}-1}\mathcal{D}_{k-1}^{2^{2k-2}} \dots \mathcal{D}_1^{2^k}\widetilde{\mathcal{D}_k}^{2^{k-1}} \dots \widetilde{\mathcal{D}_1}^1$$

to F and $\widetilde{F}$, respectively. The reason these differential operators are so useful is that $\mathcal{D}$ zeros out the generating function for the true mixture and also zeros out all but one component of the generating function for the hypothetical mixture with parameters $\widetilde{w_i}, \widetilde{\mu_i}, I + \widetilde{\Sigma}_i$. For the one component that is not zeroed out, only the leading coefficient remains and we can use Claim 3.11 to explicitly compute the leading coefficient. Thus, we can compare the results of applying these operators on the generating functions for the true and hypothetical mixtures and, using the fact that the Hermite polynomials for these mixtures must be close, we obtain algebraic relations that allow us to extract information about individual components.

We begin by explicitly computing the relevant leading coefficients.

Claim 4.7. *Write*

$$\widetilde{\mathcal{D}_k}^{2^{2k-1}-1}\widetilde{\mathcal{D}_{k-1}}^{2^{2k-2}} \dots \widetilde{\mathcal{D}_1}^{2^k}\mathcal{D}_k^{2^{k-1}} \dots \mathcal{D}_1^1(\widetilde{F})$$

as a formal power series in y. Its evaluation at $y = 0$ is

$$C_k\widetilde{w_k}\prod_{i=1}^{k}(\widetilde{\Sigma_k}(X) - \Sigma_i(X))^{2^{i-1}}\prod_{i=1}^{k-1}(\widetilde{\Sigma_k}(X) - \widetilde{\Sigma_i}(X))^{2^{k+i-1}}$$

where C_k is a constant depending only on k.

Proof. Write

$$\widetilde{F}(y) = \sum_{i=1}^{k}\widetilde{w_i}e^{\widetilde{\mu_i}(X)y + \frac{1}{2}\widetilde{\Sigma_i}(X)y^2}$$

When applying the differential operator, by Corollary 3.10, all of the terms become 0 except for

$$\widetilde{w_k}e^{\widetilde{\mu_k}(X)y + \frac{1}{2}\widetilde{\Sigma_i}(X)y^2}.$$

We now use Claims 3.11 and 3.9 to analyze what happens when applying the differential operator to this term. We know that

$$\widetilde{\mathcal{D}_{k-1}}^{2^{2k-2}} \dots \widetilde{\mathcal{D}_1}^{2^k} \mathcal{D}_k^{2^{k-1}} \dots \mathcal{D}_1^1(\widetilde{F}) = P(y)e^{\widetilde{\mu_k}(X)y+\frac{1}{2}\widetilde{\Sigma_i}(X)y^2}$$

where P has leading coefficient

$$\widetilde{w_k}\prod_{i=1}^{k}(\widetilde{\Sigma_k}(X)-\Sigma_i(X))^{2^{i-1}}\prod_{i=1}^{k-1}(\widetilde{\Sigma_k}(X)-\widetilde{\Sigma_i}(X))^{2^{k+i-1}}$$

and degree $2^{2k-1}-1$. Thus,

$$\begin{aligned}&\widetilde{\mathcal{D}_k}^{2^{2k-1}-1}\widetilde{\mathcal{D}_{k-1}}^{2^{2k-2}} \dots \widetilde{\mathcal{D}_1}^{2^k} \mathcal{D}_k^{2^{k-1}} \dots \mathcal{D}_1^1(\widetilde{F})\\ &= (2^{2k-1}-1)!\widetilde{w_k}\prod_{i=1}^{k}(\widetilde{\Sigma_k}(X)-\Sigma_i(X))^{2^{i-1}}\prod_{i=1}^{k-1}(\widetilde{\Sigma_k}(X)-\widetilde{\Sigma_i}(X))^{2^{k+i-1}}e^{\widetilde{\mu_k}(X)y+\frac{1}{2}\widetilde{\Sigma_i}(X)y^2}\end{aligned}$$

and plugging in $y=0$, we are done. □

Claim 4.8. *Write*

$$\mathcal{D}_k^{2^{2k-1}-1}\mathcal{D}_{k-1}^{2^{2k-2}} \dots \mathcal{D}_1^{2^k}\widetilde{\mathcal{D}_k}^{2^{k-1}} \dots \widetilde{\mathcal{D}_1}^1(F)$$

as a formal power series in y. Its evaluation at $y=0$ is

$$C_k w_k\prod_{i=1}^{k}(\Sigma_k(X)-\widetilde{\Sigma}_i(X))^{2^{i-1}}\prod_{i=1}^{k-1}(\Sigma_k(X)-\Sigma_i(X))^{2^{k+i-1}}$$

where C_k is a constant depending only on k.

Proof. This can be proved using the same method as Claim 4.7. □

Combining the previous two claims with Claims 4.5 and 4.6, we can write the expressions for the leading coefficients as polynomial combinations of the Hermite polynomials.

Lemma 4.9. *Consider the polynomial*

$$\widetilde{w_k}\prod_{i=1}^{k}(\widetilde{\Sigma_k}(X)-\Sigma_i(X))^{2^{i-1}}\prod_{i=1}^{k-1}(\widetilde{\Sigma_k}(X)-\widetilde{\Sigma_i}(X))^{2^{k+i-1}}$$

It can be written in the form

$$P_0(X)+P_1(X)\widetilde{h_1}(X)+\dots+P_m(X)\widetilde{h_m}(X)$$

where

- *m is a function of k*
- *Each of the P_i is m-simple*
- *We have*
$$P_0(X)+P_1(X)h_1(X)+\dots+P_m(X)h_m(X)=0$$
as an algebraic identity over formal variables $X_1,\dots,X_d,\{u_i\},\{v_i\}$.

Proof. Consider the power series

$$\widetilde{\mathcal{D}_k}^{2^{2k-1}-1}\widetilde{\mathcal{D}_{k-1}}^{2^{2k-2}} \dots \widetilde{\mathcal{D}_1}^{2^k} \mathcal{D}_k^{2^{k-1}} \dots \mathcal{D}_1^1(\widetilde{F})$$

Now using Claim 4.7 and repeating the proof of Claim 4.5, we get the desired identity. □

Similarly, we have:

Lemma 4.10. *Consider the polynomial*

$$w_k \prod_{i=1}^{k} (\Sigma_k(X) - \widetilde{\Sigma_i}(X))^{2^{i-1}} \prod_{i=1}^{k-1} (\Sigma_k(X) - \Sigma_i(X))^{2^{k+i-1}}$$

It can be written in the form

$$P_0(X) + P_1(X)h_1(X) + \cdots + P_m(X)h_m(X)$$

where

- *m is a function of k*
- *Each of the P_i is m-simple*
- *We have*

$$P_0(X) + P_1(X)\widetilde{h_1}(X) + \cdots + P_m(X)\widetilde{h_m}(X) = 0$$

as an algebraic identity over formal variables $X_1, \dots, X_d, \{u_i\}, \{v_i\}$.

Everything we've done so far has been symbolic manipulations and the claims in this section are all true as algebraic identities. We are now ready to analyze the SOS program. Note the polynomials $P_0, \dots, P_m$ in Lemma 4.9 are unknown because they depend on the true parameters. This is fine because we will simply use their existence to deduce properties of pseudoexpectations that solve the SOS-system $\mathcal{S}$.

Let U be the subspace spanned by the true $\mu_1, \dots, \mu_k$ and let V denote the subspace spanned by the true (flattened) $\Sigma_1, \dots, \Sigma_k$. We will use $\Gamma_V, \Gamma_{V^\perp}$ to denote projections onto V and the orthogonal complement of V (and similar for $U, U^\perp$). Note that these are linear maps.

Our goal now will be to show that V is essentially contained within the span of the union of the top k principal components of the matrices

$$\widetilde{\mathbb{E}}\left[\widetilde{\Sigma_1}\widetilde{\Sigma_1}^T\right], \dots, \widetilde{\mathbb{E}}\left[\widetilde{\Sigma_k}\widetilde{\Sigma_k}^T\right]$$

This gives us a k^2-dimensional space that essentially contains V and then we can guess the true covariance matrices via brute force search. In the first key lemma, we prove that the matrix $\widetilde{\mathbb{E}}[\widetilde{\Sigma_i}\widetilde{\Sigma_i}^T]$ lives almost entirely within the subspace V.

Lemma 4.11. *Let $\widetilde{\mathbb{E}}$ be a pseudoexpectation of degree C_k for some sufficiently large constant C_k depending only on k that solves $\mathcal{S}$. Consider the matrix*

$$M = \widetilde{\mathbb{E}}\left[\widetilde{\Sigma_k}\widetilde{\Sigma_k}^T\right]$$

where by this we mean we construct the $D \times D$ matrix $\widetilde{\Sigma_k}\widetilde{\Sigma_k}^T$ whose entries are quadratic in the variables $\{u\}, \{v\}$ and then take the entry-wise pseudoexpectation. Then

$$Tr_{V^\perp}(M) \leq (\epsilon')^{2^{-k}} O_k(1) \left(\frac{\Delta}{w_{\min} c}\right)^{O_k(1)}$$

where $Tr_{V^\perp}(M)$ denotes the trace of M on the subspace $V^\perp$.

Proof. Using Lemma 4.9, we may write

$$\widetilde{w_k} \prod_{i=1}^{k} (\widetilde{\Sigma_k}(X) - \Sigma_i(X))^{2^{i-1}} \prod_{i=1}^{k-1} (\widetilde{\Sigma_k}(X) - \widetilde{\Sigma_i}(X))^{2^{k+i-1}} = P_1(X)(\widetilde{h_1}(X) - h_1(X)) + \cdots + P_m(X)(\widetilde{h_m}(X) - h_m(X))$$

where $m = O_k(1)$

Now we bound

$$\widetilde{\mathbb{E}}\left[\left\|v\left(P_1(X)(\widetilde{h_1}(X)-h_1(X))+\cdots+P_m(X)(\widetilde{h_m}(X)-h_m(X))\right)\right\|^2\right]$$

Using Claims 3.15 and 3.14,

$$\begin{aligned}
&\widetilde{\mathbb{E}}\left[\left\|v\left(P_1(X)(\widetilde{h_1}(X)-h_1(X))+\cdots+P_m(X)(\widetilde{h_m}(X)-h_m(X))\right)\right\|^2\right]\\
&\leq O_k(1)\sum_{i=1}^{m}\widetilde{\mathbb{E}}\left[\|v(P_i(X))\|^2\cdot\left\|v(\widetilde{h_i}(X)-h_i(X))\right\|^2\right]\\
&\leq O_k(1)\sum_{i=1}^{m}\widetilde{\mathbb{E}}\left[\|v(P_i(X))\|^2\cdot 2\left(\left\|v(\widetilde{h_i}(X)-\overline{h_i}(X))\right\|^2+\left\|v(\overline{h_i}(X)-h_i(X))\right\|^2\right)\right]
\end{aligned}$$

Where the last step is true because Claim 3.14 allows us to write the difference between the two sides as a sum of squares.

Now $||v(\overline{h_i}(X)-h_i(X))||^2$ is just a real number and is bounded above by ϵ' by assumption. We also have the constraint that

$$\left\|v(\widetilde{h_i}(X)-\overline{h_i}(X))\right\|^2\leq 100\epsilon'$$

so

$$\widetilde{\mathbb{E}}\left[\left\|v\left(P_1(X)(\widetilde{h_1}(X)-h_1(X))+\cdots+P_m(X)(\widetilde{h_m}(X)-h_m(X))\right)\right\|^2\right]\leq O_k(1)\epsilon'\sum_{i=1}^{m}\widetilde{\mathbb{E}}\left[\|v(P_i(X))\|^2\right]$$

Now we use the properties from Lemma 4.9 that each of the P_i can be written as a linear combination of a constant number of terms that are a product of some of $\{\mu_i(X)\},\{\Sigma_i(X)\},\{\widetilde{\mu_i}(X)\},\{\widetilde{\Sigma_i}(X)\}$ where

- The coefficients in the linear combination are bounded by a constant depending only on k
- The number of terms in the sum depends only on k
- The number of terms in each product depends only on k.

Note for each $\widetilde{\mu_i}(X)$, since we ensured that our guesses for the coefficients that go with the orthonormal basis $u_1,\dots,u_k$ are at most Δ and we have the constraints $\|u_i\|_2^2=1, u_i\cdot u_j=0$, we have

$$\|v(\widetilde{\mu_i}(X))\|^2\dashv_{O(1)}O_k(1)\Delta^2$$

where $\dashv_{O(1)}$ means the difference can be written as a sum of squares of polynomials of constant degree. We can make similar arguments for $\widetilde{\Sigma_i}(X),\mu_i(X),\Sigma_i(X)$. Now using Claims 3.14 and 3.15 we can deduce

$$\widetilde{\mathbb{E}}\left[\|v(P_i(X))\|^2\right]\leq O_k(1)\Delta^{O_k(1)}$$

Overall, we have shown

$$\widetilde{\mathbb{E}}\left[\left\|v\left(P_1(X)(\widetilde{h_1}(X)-h_1(X))+\cdots+P_m(X)(\widetilde{h_m}(X)-h_m(X))\right)\right\|^2\right]\leq O_k(1)\epsilon'\Delta^{O_k(1)}$$

Now we examine the expression

$$\widetilde{\mathbb{E}}\left[\left\|v\left(\widetilde{w_k}\prod_{i=1}^{k}(\widetilde{\Sigma_k}(X)-\Sigma_i(X))^{2^{i-1}}\prod_{i=1}^{k-1}(\widetilde{\Sigma_k}(X)-\widetilde{\Sigma_i}(X))^{2^{k+i-1}}\right)\right\|^2\right]$$

By Claim 3.18 (recall $\widetilde{w_k}$ is a constant that we guess),

$$\widetilde{\mathbb{E}}\left[\left\|v\left(\widetilde{w_k}\prod_{i=1}^{k}(\widetilde{\Sigma_k}(X)-\Sigma_i(X))^{2^{i-1}}\prod_{i=1}^{k-1}(\widetilde{\Sigma_k}(X)-\widetilde{\Sigma}_i(X))^{2^{k+i-1}}\right)\right\|^2\right]$$
$$\geq \widetilde{w_k}\Omega_k(1)\widetilde{\mathbb{E}}\left[\prod_{i=1}^{k}\left(\left\|v(\widetilde{\Sigma_k}(X)-\Sigma_i(X))\right\|^2\right)^{2^{i-1}}\prod_{i=1}^{k-1}\left(\left\|v(\widetilde{\Sigma_k}(X)-\widetilde{\Sigma}_i(X))\right\|^2\right)^{2^{k+i-1}}\right]$$

Note that

$$\left\|v(\widetilde{\Sigma_k}(X)-\Sigma_i(X))\right\|^2 \vdash_{O(1)} \left\|\Gamma_{V^\perp}(\widetilde{\Sigma_k})\right\|^2$$

(recall that $\Gamma_{V^\perp}$ is a projection map with unknown but constant coefficients). Next, since we ensure that the coefficients B_i that we guess for the orthonormal basis satisfy $||B_i - B_j||_2 \geq \frac{c}{2}$, we have

$$\left\|v(\widetilde{\Sigma_k}(X)-\widetilde{\Sigma}_i(X))\right\|^2 \vdash_{O(1)} \frac{c^2}{4}$$

where we use the constraints in $\mathcal{S}$ that $\|v_i\|_2^2 = 1, v_i \cdot v_j = 0$. Overall, we conclude

$$\widetilde{\mathbb{E}}\left[\left\|v\left(\widetilde{w_k}\prod_{i=1}^{k}(\widetilde{\Sigma_k}(X)-\Sigma_i(X))^{2^{i-1}}\prod_{i=1}^{k-1}(\widetilde{\Sigma_k}(X)-\widetilde{\Sigma}_i(X))^{2^{k+i-1}}\right)\right\|^2\right] \geq \Omega_k(1)\widetilde{\mathbb{E}}\left[\left\|\Gamma_{V^\perp}(\widetilde{\Sigma_k})\right\|^{2^{k+1}-2}\right](\widetilde{w_k}c)^{O_k(1)}$$

Note

$$\widetilde{\mathbb{E}}\left[\left\|v\left(\widetilde{w_k}\prod_{i=1}^{k}(\widetilde{\Sigma_k}(X)-\Sigma_i(X))^{2^{i-1}}\prod_{i=1}^{k-1}(\widetilde{\Sigma_k}(X)-\widetilde{\Sigma}_i(X))^{2^{k+i-1}}\right)\right\|^2\right] =$$
$$\widetilde{\mathbb{E}}\left[\left\|v\left(P_1(X)(\widetilde{h}_1(X)-h_1(X))+\cdots+P_m(X)(\widetilde{h_m}(X)-h_m(X))\right)\right\|^2\right]$$

because the inner expressions are equal symbolically. Thus,

$$\widetilde{\mathbb{E}}\left[\left\|\Gamma_{V^\perp}(\widetilde{\Sigma_k})\right\|^{2^{k+1}-2}\right] \leq O_k(1)\epsilon'\left(\frac{\Delta}{w_{\min}c}\right)^{O_k(1)}$$

Thus,

$$\widetilde{\mathbb{E}}\left[\left\|\Gamma_{V^\perp}(\widetilde{\Sigma_k})\right\|^2\right] \leq (\epsilon')^{2^{-k}}O_k(1)\left(\frac{\Delta}{w_{\min}c}\right)^{O_k(1)}$$

It remains to note that

$$\mathrm{Tr}_{V^\perp}(M) = \widetilde{\mathbb{E}}\left[\left\|\Gamma_{V^\perp}(\widetilde{\Sigma_k})\right\|^2\right]$$

and we are done. □

In the next key lemma, we prove that any vector that has nontrivial projection onto V must also have nontrivial projection onto $\widetilde{\mathbb{E}}[\widetilde{\Sigma}_i\widetilde{\Sigma}_i^T]$ for some i.

Lemma 4.12. *Let $\widetilde{\mathbb{E}}$ be a pseudoexpectation of degree C_k for some sufficiently large constant C_k depending only on k that solves $\mathcal{S}$. Consider the matrix*

$$N = \sum_{i=1}^{k}\widetilde{\mathbb{E}}\left[\widetilde{\Sigma}_i\widetilde{\Sigma}_i^T\right]$$

where by this we mean we construct the $D \times D$ matrix whose entries are quadratic in the variables $\{u\}, \{v\}$ and then take the entry-wise pseudoexpectation. Then for any unit vector $z \in \mathbb{R}^D$,

$$z^T N z \geq \left(\frac{w_{\min}(z \cdot \Sigma_k)^{O_k(1)} - O_k(1)\epsilon' \Delta^{O_k(1)}}{O_k(1)\Delta^{O_k(1)}} \right)^2$$

as long as

$$w_{\min}(z \cdot \Sigma_k)^{O_k(1)} > O_k(1)\epsilon' \Delta^{O_k(1)}.$$

Proof. Using Lemma 4.10, we may write

$$w_k \prod_{i=1}^{k} (\Sigma_k(X) - \widetilde{\Sigma_i}(X))^{2^{i-1}} \prod_{i=1}^{k-1} (\Sigma_k(X) - \Sigma_i(X))^{2^{k+i-1}} = P_1(X)(\widetilde{h_1}(X) - h_1(X)) + \cdots + P_m(X)(\widetilde{h_m}(X) - h_m(X))$$

where $m = O_k(1)$.

Using the same method as the proof in Lemma 4.11, we have

$$\widetilde{\mathbb{E}}\left[\left\| v\left(P_1(X)(\widetilde{h_1}(X) - h_1(X)) + \cdots + P_m(X)(\widetilde{h_m}(X) - h_m(X)) \right) \right\|^2 \right] \leq O_k(1)\epsilon' \Delta^{O_k(1)}$$

Now by Claim 3.18,

$$\begin{aligned}
&\widetilde{\mathbb{E}}\left[\left\| v\left(w_k \prod_{i=1}^{k} (\Sigma_k(X) - \widetilde{\Sigma_i}(X))^{2^{i-1}} \prod_{i=1}^{k-1} (\Sigma_k(X) - \Sigma_i(X))^{2^{k+i-1}} \right) \right\|^2 \right] \\
&\geq w_k \widetilde{\mathbb{E}}\left[\prod_{i=1}^{k} \left(\left\| v\left(\Sigma_k(X) - \widetilde{\Sigma_i}(X) \right) \right\|^2 \right)^{2^{i-1}} \prod_{i=1}^{k-1} \left(\|(\Sigma_k(X) - \Sigma_i(X))\|^2 \right)^{2^{k+i-1}} \right] \\
&\geq w_k \widetilde{\mathbb{E}}\left[\prod_{i=1}^{k} \left((z \cdot \Sigma_k - z \cdot \widetilde{\Sigma_i})^2 \right)^{2^{i-1}} c^{O_k(1)} \right]
\end{aligned}$$

where the second inequality is true because

$$\left\| v\left(\Sigma_k(X) - \widetilde{\Sigma_i}(X) \right) \right\|^2 \vdash_{O(1)} (z \cdot \Sigma_k - z \cdot \widetilde{\Sigma_i})^2$$

Now we claim

$$\widetilde{\mathbb{E}}\left[\prod_{i=1}^{k} \left((z \cdot \Sigma_k - z \cdot \widetilde{\Sigma_i})^2 \right)^{2^{i-1}} \right] \geq (z \cdot \Sigma_k)^{O_k(1)} - O_k(1)\Delta^{O_k(1)} \sqrt{\widetilde{\mathbb{E}}\left[\sum_i (z \cdot \widetilde{\Sigma_i})^2 \right]}$$

To see this, first recall that $z \cdot \Sigma_k$ is just a constant. Next, we can expand the LHS into a sum of monomials in the $z \cdot \widetilde{\Sigma_i}$. In particular, we can write the expansion in the form

$$(z \cdot \Sigma_k)^{O_k(1)} + \sum_i (z \cdot \widetilde{\Sigma_i}) P_i(z \cdot \widetilde{\Sigma_1}, \ldots, z \cdot \widetilde{\Sigma_k})$$

where P is some polynomial in k variables. We can upper bound the coefficients of the polynomial in terms of Δ, k and we also know that

$$(z \cdot \widetilde{\Sigma_i})^2 \dashv_{O(1)} O_k(1)\Delta^{O(1)}$$

due to the constraints in our system. Thus, we can bound the pseudoexpectation

$$-\widetilde{\mathbb{E}}\left[\sum_i (z \cdot \widetilde{\Sigma_i}) P_i(z \cdot \widetilde{\Sigma_1}, \ldots, z \cdot \widetilde{\Sigma_k}) \right] \leq O_k(1)\Delta^{O_k(1)} \sqrt{\widetilde{\mathbb{E}}\left[\sum_i (z \cdot \widetilde{\Sigma_i})^2 \right]}$$

via Cauchy Schwarz. Putting everything together the same way as in Lemma 4.11, we deduce

$$\widetilde{\mathbb{E}}\left[\sum_i (z \cdot \widetilde{\Sigma}_i)^2\right] \geq \left(\frac{w_{\min}(z \cdot \Sigma_k)^{O_k(1)} - O_k(1)\epsilon' \Delta^{O_k(1)}}{O_k(1)\Delta^{O_k(1)}}\right)^2$$

and now we are done. □

Putting Lemmas 4.11 and 4.12 together, we now prove that V is essentially contained within the span of the union of the top principal components of $\widetilde{\mathbb{E}}[\widetilde{\Sigma}_i \widetilde{\Sigma}_i^T]$ over all i.

Lemma 4.13. *For each i, let M_i be the $D \times D$ matrix given by*

$$M_i = \widetilde{\mathbb{E}}\left[\widetilde{\Sigma}_i \widetilde{\Sigma_i^T}\right].$$

Assume that for a sufficiently small function f depending only on k,

$$\Delta \leq (\epsilon')^{-f(k)}$$
$$w_{\min}, c \geq (\epsilon')^{f(k)}$$

Let V_i be the subspace spanned by the top k singular vectors of M_i. Then for all i, the projection of the true covariance matrix Σ_i onto the orthogonal complement of $spn(V_1, \ldots, V_k)$ has length at most $(\epsilon')^{\Omega_k(1)}$.

Proof. Assume for the sake of contradiction that the desired statement is false for Σ_i. Let z be the projection of Σ_i onto the orthogonal complement of $\mathrm{spn}(V_1, \ldots, V_k)$. By Lemma 4.12,

$$\sum_j z^T M_j z \geq \left(\frac{w_{\min}(z \cdot \Sigma_i)^{O_k(1)} - O_k(1)\epsilon' \Delta^{O_k(1)}}{O_k(1)\Delta^{O_k(1)}}\right)^2 \tag{2}$$

so there is some j for which

$$z^T M_j z \geq \frac{1}{k}\left(\frac{w_{\min}(z \cdot \Sigma_i)^{O_k(1)} - O_k(1)\epsilon' \Delta^{O_k(1)}}{O_k(1)\Delta^{O_k(1)}}\right)^2$$

On the other hand, Lemma 4.11 implies that the sum of the singular values of M_j outside the top k is at most

$$(\epsilon')^{2^{-k}} O_k(1) \left(\frac{\Delta}{w_{\min} c}\right)^{O_k(1)}$$

Since z is orthogonal to the span of the top-k singular vectors of M_j, we get

$$z^T M_j z \leq (\epsilon')^{2^{-k}} O_k(1) \left(\frac{\Delta}{w_{\min} c}\right)^{O_k(1)} \|z\|_2^2 \tag{3}$$

Note $z \cdot \Sigma_i = \|z\|_2^2$ since z is a projection of Σ_i onto a subspace. Now combining (2) and (3) we get a contradiction unless

$$\|z\|_2 \leq (\epsilon')^{\Omega_k(1)}.$$

□

4.3.3 Finishing Up: Finding the Covariances and then the Means. Now we can brute-force search over the subspace spanned by the union of the top k singular vectors of $M_1, \dots, M_k$. Note that the SOS system $\mathcal{S}$ is clearly feasible as it is solved when the u_i, v_i form orthonormal bases for the true subspaces and the $\widetilde{w_i}, A_i, B_i$ are within $(\epsilon')^{O_k(1)}$ of the true values (i.e., the values needed to express the true means and covariances in the orthonormal basis given by the u_i, v_i).

Thus, brute forcing over an $(\epsilon')^{O_k(1)}$-net for the $\widetilde{w_i}, A_i, B_i$, we will find a feasible solution. By Lemma 4.12 and Lemma 4.13, once we find any feasible solution, we will be able to obtain a set of $(1/\epsilon')^{O_k(1)}$ estimates at least one of which, say $\overline{\Sigma_1}, \dots, \overline{\Sigma_k}$, satisfies

$$\left\|\Sigma_i - \overline{\Sigma_i}\right\|_2^2 \leq (\epsilon')^{\Omega_k(1)}$$

for all i. With these estimates we will now solve for the means. Note we can assume that our covariance estimates are exactly correct because we can pretend that the true mixture is actually $N(\mu_1, \overline{\Sigma_1}), \dots, N(\mu_k, \overline{\Sigma_k})$ and our estimates for the Hermite polynomials of this mixture will be off by at most $O_k(1)(\epsilon')^{\Omega_k(1)}$. Thus, making this assumption will only affect the dependence on ϵ' that we get at the end. From now on we can write Σ_i to denote the true covariances and treat these as known quantities.

Now we set up the same system as in Section 4.2 except we no longer have the variables $v_1, \dots, v_k$ and no longer have the $\widetilde{\Sigma}_i$. These will instead be replaced by real values from Σ_i. Formally:

Definition 4.14 (SOS Program for Learning Means). We will have the following variables

- $u_1 = (u_{11}, \dots, u_{1d}), \dots, u_k = (u_{k1}, \dots, u_{kd})$

In the above $u_1, \dots, u_k \in \mathbb{R}^d$. We guess coefficients a_{ij} where $i, j \in [k]$ expressing the means in this orthonormal basis. We ensure that the guesses satisfy the property that for every pair of vectors $A_i = (a_{i1}, \dots, a_{ik}), A_j = (a_{j1}, \dots, a_{jk})$ either $A_i = A_j$ or

$$||A_i - A_j||_2 \geq \frac{c}{2}$$

We ensure that

$$||A_i||_2 \leq 2\Delta$$

We also guess the mixing weights $\widetilde{w_1}, \dots, \widetilde{w_k}$ and ensure that our guesses are all at least $w_{\min}/2$.

Now we set up the constraints. Let C be a sufficiently large integer depending only on k. Define $\widetilde{\mu_i} = a_{i1}u_1 + \cdots + a_{ik}u_k$. These are linear expressions in the variables that we are solving for. Now consider the hypothetical mixture with mixing weights $\widetilde{w_i}$, means $\widetilde{\mu_i}$, and covariances $I + \Sigma_i$. The Hermite polynomials for this hypothetical mixture $\widetilde{h_i}(X)$ can be written as formal polynomials in $X = (X_1, \dots, X_d)$ with coefficients that are polynomials in u. Note that we can explicitly write down these Hermite polynomials. The set of constraints for our SOS system is as follows:

- $\|u_i\|_2^2 = 1$ for all $1 \leq i \leq k$
- $u_i \cdot u_j = 0$ for all $i \neq j$
- For all $p = 1, 2, \dots, C$

$$\left\|v(\widetilde{h_p}(X) - \overline{h_p}(X))\right\|^2 \leq 100\epsilon'$$

Now we can repeat the same arguments from Section 4.3.2 to prove that once we find a feasible solution, we can recover the span of the μ_i. The important generating functions are

$$F(y) = \sum_{i=1}^{k} w_i e^{\mu_i(X)y + \frac{1}{2}\Sigma_i(X)y^2}$$

$$\widetilde{F}(y) = \sum_{i=1}^{k} \widetilde{w_i} e^{\widetilde{\mu_i}(X)y + \frac{1}{2}\Sigma_i(X)y^2}$$

Define the differential operators as before except with $\widetilde{\Sigma_i}$ replaced with Σ_i. Let $\mathcal{D}_i$ denote the differential operator $(\partial - (\mu_i(X) + \Sigma_i(X)y))$ and $\widetilde{\mathcal{D}_i}$ denote the differential operator $(\partial - (\widetilde{\mu_i}(X) + \Sigma_i(X)y))$. All derivatives are taken with respect to y. The two key differential operators to consider are

$$\widetilde{\mathcal{D}_k}^{2^{2k-1}-2^{k-1}-1}\widetilde{\mathcal{D}_{k-1}}^{2^{2k-2}}\dots\widetilde{\mathcal{D}_1}^{2^k}\mathcal{D}_k^{2^{k-1}}\dots\mathcal{D}_1^1$$

$$\mathcal{D}_k^{2^{2k-1}-2^{k-1}-1}\mathcal{D}_{k-1}^{2^{2k-2}}\dots\mathcal{D}_1^{2^k}\widetilde{\mathcal{D}_k}^{2^{k-1}}\dots\widetilde{\mathcal{D}_1}^1$$

Note the change to $2^{2k-1} - 2^{k-1} - 1$ from $2^{2k-1} - 1$ in the exponent of the first term. This is because when operating on $P(y)e^{\mu_k(X)y+\frac{1}{2}\Sigma_k(X)y^2}$ for some polynomial P, the operator $\mathcal{D}_k$ reduces the degree of P by 1 while the operator $\widetilde{D_k}$ does not change the degree of P (whereas before this operator increased the degree of the formal polynomial P). Similar to Claims 4.7 and 4.8 in Section 4.3.2, we have

Claim 4.15. *Write*

$$\widetilde{\mathcal{D}_k}^{2^{2k-1}-2^{k-1}-1}\widetilde{\mathcal{D}_{k-1}}^{2^{2k-2}}\dots\widetilde{\mathcal{D}_1}^{2^k}\mathcal{D}_k^{2^{k-1}}\dots\mathcal{D}_1^1(\widetilde{F})$$

as a formal power series in y. Its evaluation at $y = 0$ is

$$C_k\widetilde{w_k}(\widetilde{\mu_k}(X) - \mu_k(X))^{2^{k-1}}\prod_{i=1}^{k-1}(\Sigma_k(X) - \Sigma_i(X))^{2^{i-1}}\prod_{i=1}^{k-1}(\Sigma_k(X) - \Sigma_i(X))^{2^{k+i-1}}$$

where C_k is a constant depending only on k.

Claim 4.16. *Write*

$$\mathcal{D}_k^{2^{2k-1}-2^{k-1}-1}\mathcal{D}_{k-1}^{2^{2k-2}}\dots\mathcal{D}_1^{2^k}\widetilde{\mathcal{D}_k}^{2^{k-1}}\dots\widetilde{\mathcal{D}_1}^1(F)$$

as a formal power series in y. Its evaluation at $y = 0$ is

$$C_k w_k(\widetilde{\mu_k}(X) - \mu_k(X))^{2^{k-1}}\prod_{i=1}^{k-1}(\Sigma_k(X) - \Sigma_i(X))^{2^{i-1}}\prod_{i=1}^{k-1}(\Sigma_k(X) - \Sigma_i(X))^{2^{k+i-1}}$$

where C_k is a constant depending only on k.

Now repeating the arguments in Lemmas 4.9, 4.10, 4.11, 4.12, and 4.13, we can prove that for any feasible solution, the subspace spanned by the top k singular vectors of each of $\widetilde{\mathbb{E}}[\widetilde{\mu_1}\widetilde{\mu_1}^T], \dots, \widetilde{\mathbb{E}}[\widetilde{\mu_k}\widetilde{\mu_k}^T]$ approximately contains all of $\mu_1, \dots, \mu_k$. We can now brute force search over this subspace (and since we are already brute-force searching over the mixing weights), we will output some set of candidate components that are close to the true components.

4.3.4 All Pairs of Parameters are Equal or Separated. In the case where some pairs of parameters may be equal (but pairs (μ_i, Σ_i) and (μ_j, Σ_j) cannot be too close), we can repeat essentially the same arguments from the previous section but with minor adjustments in the number of times we are applying each differential operator.

We can assume that our guesses for the coefficients A_i, B_i satisfy the correct equality pattern in the sense that $A_i = A_j$ if and only if $\mu_i = \mu_j$ and otherwise $||A_i - A_j|| \geq c/2$ and similar for the parameters B_i. This is because there are only $O_k(1)$ different equality patterns.

Now without loss of generality let $\{\Sigma_1, \dots, \Sigma_j\}$ $(j < k)$ be the set of covariance matrices that are equal to Σ_k. The key differential operators to consider are

$$\widetilde{\mathcal{D}_k}^{2^{2k-1}-1-2^k-\dots-2^{k+j}} \widetilde{\mathcal{D}_{k-1}}^{2^{2k-2}} \dots \widetilde{\mathcal{D}_1}^{2^k} \mathcal{D}_k^{2^{k-1}} \dots \mathcal{D}_1^1$$
$$\mathcal{D}_k^{2^{2k-1}-1-2^0-\dots-2^j} \mathcal{D}_{k-1}^{2^{2k-2}} \dots \mathcal{D}_1^{2^k} \widetilde{\mathcal{D}_k}^{2^{k-1}} \dots \widetilde{\mathcal{D}_1}^1$$

Similar to Claims 4.7 and 4.8, we get.

Claim 4.17. *Let $\{\Sigma_1, \dots, \Sigma_j\}$ $(j < k)$ be the set of covariance matrices that are equal to Σ_k. Note this also implies $\{\widetilde{\Sigma_1}, \dots, \widetilde{\Sigma_j}\}$ are precisely the subset of $\{\widetilde{\Sigma_i}\}$ that are equal to $\widetilde{\Sigma_k}$. Write*

$$\widetilde{\mathcal{D}_k}^{2^{2k-1}-1-2^k-\dots-2^{k+j-1}} \widetilde{\mathcal{D}_{k-1}}^{2^{2k-2}} \dots \widetilde{\mathcal{D}_1}^{2^k} \mathcal{D}_k^{2^{k-1}} \dots \mathcal{D}_1^1(\widetilde{F})$$

as a formal power series in y. Its evaluation at $y = 0$ is

$$C_k \widetilde{w_k} \prod_{i=1}^{k} (\widetilde{\Sigma_k}(X) - \Sigma_i(X))^{2^{i-1}} \prod_{i=1}^{j} (\widetilde{\mu_k}(X) - \widetilde{\mu_i}(X))^{2^{k+i-1}} \prod_{i=j+1}^{k-1} (\widetilde{\Sigma_k}(X) - \widetilde{\Sigma_i}(X))^{2^{k+i-1}}$$

where C_k is a constant depending only on k.

Claim 4.18. *Let $\{\Sigma_1, \dots, \Sigma_j\}$ $(j < k)$ be the set of covariance matrices that are equal to Σ_k. Note this also implies $\{\widetilde{\Sigma_1}, \dots, \widetilde{\Sigma_j}\}$ are precisely the subset of $\{\widetilde{\Sigma_i}\}$ that are equal to $\widetilde{\Sigma_k}$. Write*

$$\mathcal{D}_k^{2^{2k-1}-1-2^k-\dots-2^{k+j-1}} \mathcal{D}_{k-1}^{2^{2k-2}} \dots \mathcal{D}_1^{2^k} \widetilde{\mathcal{D}_k}^{2^{k-1}} \dots \widetilde{\mathcal{D}_1}^1(F)$$

as a formal power series in y. Its evaluation at $y = 0$ is

$$C_k \widetilde{w_k} \prod_{i=1}^{k} (\Sigma_k(X) - \widetilde{\Sigma}_i(X))^{2^{i-1}} \prod_{i=1}^{j} (\mu_k(X) - \mu_i(X))^{2^{k+i-1}} \prod_{i=j+1}^{k-1} (\Sigma_k(X) - \Sigma_i(X))^{2^{k+i-1}}$$

where C_k is a constant depending only on k.

Now we can repeat the arguments in Lemmas 4.9, 4.10, 4.11, 4.12, and 4.13. The key point is that the constraints in our SOS program give explicit values for

$$||v(\widetilde{\mu}_i(X) - \widetilde{\mu}_j(X))||^2$$
$$||v(\widetilde{\Sigma}_i(X) - \widetilde{\Sigma}_j(X))||^2$$

in terms of A_i, B_i (which are explicit real numbers). We can then repeat the arguments in Section 4.3.3 (with appropriate modifications to the number of times we apply each differential operator) to find the means.

5 ROBUST MOMENT ESTIMATION

In Section 4, we showed how to learn the parameters of a mixture of Gaussians $\mathcal{M}$ with components that are not too far apart when we are given estimates for the Hermite polynomials. In this section, we show how to estimate the Hermite polynomials from an ϵ-corrupted sample. Putting the results together, we will get a robust learning algorithm in the case when the components are not too far apart.

While the closeness of components in Section 4 is defined in terms of parameter distance, we will need to reason about TV-distance between components in order to integrate our results into our full learning algorithm. We begin with a definition.

Definition 5.1. We say a mixture of Gaussians $w_1 G_1 + \cdots + w_k G_k$ is δ-well-conditioned if the following properties hold:

(1) The graph $\mathcal{G}$ on $[k]$ constructed by connecting two nodes i, j if and only if $d_{\mathrm{TV}}(G_i, G_j) \leq 1 - \delta$ is a connected graph.
(2) $d_{\mathrm{TV}}(G_i, G_j) \geq \delta$ for all $i \neq j$
(3) $w_{\min} \geq \delta$

The main theorem that we will prove in this section is as follows.

Theorem 5.2. *There is a function $f(k) > 0$ depending only on k such that given an ϵ-corrupted sample from a δ-well-conditioned mixture of Gaussians*

$$\mathcal{M} = w_1 N(\mu_1, \Sigma_1) + \cdots + w_k N(\mu_k, \Sigma_k)$$

where $\delta \geq \epsilon^{f(k)}$, there is a polynomial time algorithm that outputs a set of $(1/\epsilon)^{O_k(1)}$ candidate mixtures $\{\widetilde{w_1} N(\widetilde{\mu_1}, \widetilde{\Sigma_1}) + \cdots + \widetilde{w_k} N(\widetilde{\mu_k}, \widetilde{\Sigma_k}\}$ and with high probability, at least one of them satisfies that for all i:

$$|w_i - \widetilde{w_i}| + d_{TV}(N(\mu_i, \Sigma_i), N(\widetilde{\mu_i}, \widetilde{\Sigma_i})) \leq \mathrm{poly}(\epsilon).$$

5.1 Distance between Gaussians

As mentioned earlier, we will first introduce a few tools for relating parameter distance and TV distance between Gaussians.

The following is a standard fact.

Claim 5.3. *For two Gaussians $N(\mu_1, \Sigma_1), N(\mu_2, \Sigma_2)$*

$$d_{TV}(N(\mu_1, \Sigma_1), N(\mu_2, \Sigma_2)) = O\left(\left((\mu_1 - \mu_2)^T \Sigma_1^{-1} (\mu_1 - \mu_2)\right)^{1/2} + \left\|\Sigma_1^{-1/2} \Sigma_2 \Sigma_1^{-1/2} - I\right\|_F\right)$$

Proof. See e.g., Fact 2.1 in [39]. □

Next, we will prove a bound in the opposite direction, that when Gaussians are not too far apart in TV distance, then their parameters also cannot be too far apart.

Lemma 5.4. *Let $\mathcal{M}$ be a mixture of k Gaussians that is δ-well conditioned. Let Σ be the covariance matrix of the mixture. Then*

(1) $\Sigma_i \preceq \mathrm{poly}(\delta)^{-1} \Sigma$ for all components of the mixture
(2) $\Sigma_i \succeq \mathrm{poly}(\delta) \Sigma$ for all components of the mixture
(3) For any two components i, j, we have $||\Sigma^{-1/2}(\mu_i - \mu_j)|| \leq \mathrm{poly}(\delta)^{-1}$
(4) For any two components i, j, we have $||\Sigma^{-1/2}(\Sigma_i - \Sigma_j)\Sigma^{-1/2}||_2 \leq \mathrm{poly}(\delta)^{-1}$

where the coefficients and degrees of the polynomials may depend only on k.

Proof. The statements are invariant under linear transformations so without loss of generality let $\Sigma = I$. Assume for the sake of contradiction that the first condition is failed. Then there is some direction v such that say

$$v^T\Sigma_1 v \geq \delta^{-10k}$$

There must be some $i \in [k]$ such that $v^T\Sigma_i v \leq 1$ since otherwise the variance of the mixture in direction v would be bigger than 1. Now we claim that i and 1 cannot be connected in $\mathcal{G}$, the graph defined in Definition 5.1. To see this, if they were connected, then there must be two vertices j_1, j_2 that are consecutive along the path between 1 and i such that

$$\frac{v^T\Sigma_{j_1}v}{v^T\Sigma_{j_2}v} \geq \delta^{-10}$$

But then $d_{\mathsf{TV}}(G_{j_1}, G_{j_2}) \geq 1 - \delta$. To see this, let $\sqrt{v^T\Sigma_{j_2}v} = c$. We can project both Gaussians onto the direction v and note that the Gaussian G_{j_1} is spread over width $\delta^{-5}c$ whereas the Gaussian G_{j_2} is essentially contained in a strip of width $O(\log 1/\delta)c$.

Now we may assume that the first condition is satisfied. Now we consider when the third condition is failed. Assume that

$$||(\mu_i - \mu_j)|| \geq k\delta^{-20k}$$

Now let v be the unit vector in direction $\mu_i - \mu_j$. Projecting the Gaussians G_i, G_j onto direction v and considering the path between them, we must find j_1, j_2 that are connected such that

$$||(\mu_{j_1} - \mu_{j_2})|| \geq \delta^{-20k}$$

Now, using the fact that the first condition must be satisfied (i.e., $v^T\Sigma_{j_1}v, v^T\Sigma_{j_2}v \leq \delta^{-10k}$) we get that $d_{\mathsf{TV}}(G_{j_1}, G_{j_2}) \geq 1 - \delta$, a contradiction.

Now we may assume that the first and third conditions are satisfied. Assume now that the second condition is not satisfied. Without loss of generality, there is some vector v such that

$$v^T\Sigma_1 v \leq (\delta/k)^{10^2 k}$$

If there is some component i such that

$$v^T\Sigma_i v \geq (\delta/k)^{50k}$$

then comparing the Gaussians along the path between i and 1 in the graph $\mathcal{G}$, we get a contradiction. Thus, we now have

$$v^T\Sigma_i v \leq (\delta/k)^{50}$$

for all components. Note that the covariance of the entire mixture is the identity. Thus, there must be two components with

$$|v \cdot \mu_i - v \cdot \mu_j| \geq \frac{1}{2k}.$$

Taking the path between i and j, we must be able to find two consecutive vertices j_1, j_2 such that

$$|v \cdot \mu_{j_1} - v \cdot \mu_{j_2}| \geq \frac{1}{2k^2}.$$

However, we then get $d_{\mathsf{TV}}(G_{j_1}, G_{j_2}) > 1 - \delta$, a contradiction.

Now we consider when the first three conditions are all satisfied. Using the first two conditions, we have bounds on the smallest and largest singular value of $\Sigma_i^{1/2}\Sigma_j^{-1/2}$ for all i, j. Thus,

$$||\Sigma_i - \Sigma_j||_2 \leq \text{poly}(\delta)^{-1}||I - \Sigma_i^{-1/2}\Sigma_j\Sigma_i^{-1/2}||_2$$

for all i, j. However, if for some i, j that are connected in $\mathcal{G}$, we have

$$||(\Sigma_i - \Sigma_j)||_2 \geq (k/\delta)^{10^4}$$

then we would have

$$||I - \Sigma_i^{-1/2}\Sigma_j\Sigma_i^{-1/2}||_2 \geq (k/\delta)^{10^3}$$

and this would contradict the assumption that $d_{\text{TV}}(G_i, G_j) \leq 1 - \delta$ (this follows from the same argument as in Lemma 3.2 of [39]). Now using triangle inequality along each path, we deduce that for all i, j

$$||(\Sigma_i - \Sigma_j)||_2 \leq (k/\delta)^{10^5}$$

completing the proof. □

As a corollary to the previous lemma, in a δ-well conditioned mixture, all component means and covariances are close to the mean and covariance of the overall mixture.

Corollary 5.5. *Let $\mathcal{M}$ be a mixture of k Gaussians that is δ-well conditioned. Let μ, Σ be the mean and covariance matrix of the mixture. Then we have for all i*

- $||\Sigma^{-1/2}(\mu - \mu_i)||_2 \leq \text{poly}(\delta)^{-1}$
- $||\Sigma^{-1/2}(\Sigma - \Sigma_i)\Sigma^{-1/2}||_2 \leq \text{poly}(\delta)^{-1}$

Proof. The statement is invariant under linear transformation so we may assume $\Sigma = I$ and $\mu = 0$. Then noting

$$\mu_i = \mu + w_1(\mu_i - \mu_1) + \cdots + w_k(\mu_i - \mu_k)$$

and using Lemma 5.4, we have proved the first part. Now for the second part, note $\Sigma = \sum_{i=1}^k w_i(\Sigma_i + \mu_i\mu_i^T)$ and hence we have

$$\Sigma = \Sigma_i + w_1(\Sigma_1 - \Sigma_i) + \cdots + w_k(\Sigma_k - \Sigma_i) + \sum_{i=1}^k w_i\mu_i\mu_i^T$$

and using Lemma 5.4 and the first part, we are done.

□

5.2 Hermite Polynomial Estimation

Now we show how to estimate the Hermite polynomials of a δ-well-conditioned mixture $\mathcal{M}$ if we are given an ϵ-corrupted sample (where $\delta \geq \epsilon^{f(k)}$ for some sufficiently small function $f(k) > 0$ depending only on k). Our algorithm will closely mirror the algorithm in [39].

The first step will be to show that we can robustly estimate the mean and covariance of the mixture $\mathcal{M}$ and then we will use these estimates to compute a linear transformation to place the mixture in isotropic position.

Lemma 5.6. *There is a sufficiently small function $f(k)$ depending only on k such that given a ϵ-corrupted sample from a δ-well-conditioned mixture of Gaussians $\mathcal{M}$ with true mean and covariance μ, Σ respectively, where $\delta \geq \epsilon^{f(k)}$, then with high probability we can output estimates $\widehat{\mu}$ and $\widehat{\Sigma}$ such that*

(1) $||\Sigma^{-1/2}(\widehat{\Sigma} - \Sigma)\Sigma^{-1/2}||_2 \leq \epsilon^{\Omega_k(1)}$
(2) $||\Sigma^{-1/2}(\widehat{\mu} - \mu)||_2 \leq \epsilon^{\Omega_k(1)}$.

Proof. This can be proven using a similar argument to Proposition 4.1 in [39]. First we will estimate the covariance of the mixture. Note that the statement is invariant under linear transformation (and the robust estimation algorithm that we will use, Theorem 2.4 in [39], is also invariant under linear transformation), so it suffices to consider when $\Sigma = I$. Let the components of the mixture be $G_1, \dots, G_k$. Note that by pairing up our samples, we have access to a 2ϵ-corrupted sample from the distribution $\mathcal{M} - \mathcal{M}'$ (i.e., the difference of two independent samples from $\mathcal{M}$). For each such sample say $Y \sim \mathcal{M} - \mathcal{M}'$, $\Sigma = 0.5\mathbb{E}[YY^T]$. We will now show that $Z = YY^T$ where Z is flattened into a vector, has bounded covariance. Note that we can view Y as being sampled from a mixture of $O(k^2)$ Gaussians $G_i - G_j$ (where we may have $i = j$). We now prove that

- For $Y \sim G_i - G_j$ and $Z = YY^T$, $\mathbb{E}[Z \otimes Z] - \mathbb{E}[Z] \otimes \mathbb{E}[Z] \leq \text{poly}(\delta)^{-1} I$
- For $Y \sim G_i - G_j, Y' \sim G_{i'} - G_{j'}$ and $Z = YY^T, Z' = Y'Y'^T$, $\|\mathbb{E}[Z - Z']\|_2^2 = \text{poly}(\delta)^{-1}$

Using Lemma 5.4 and Corollary 5.5, we have $\text{poly}(\delta)^{-1}$ bounds on $\|\mu_i\|_2$, $\|\Sigma_i\|_{\text{op}}$ and $\left\|\Sigma_i - \Sigma_j\right\|_2$ for all i, j. We can now follow the same argument as Proposition 4.1 in [39] to bound the above two quantities. With these bounds, by Theorem 2.4 in [39], we can robustly estimate the covariance. Once we have an estimate for the covariance $\widehat{\Sigma}$, we can apply the linear transformation $\widehat{\Sigma}^{-1/2}$ and robustly estimate the mean (which now has covariance close to identity). □

Using the above, we can place our mixture in isotropic position. This mirrors Proposition 4.2 in [39].

Corollary 5.7. *There is a sufficiently small function $f(k)$ depending only on k such that given a ϵ-corrupted sample from a δ-well-conditioned mixture of Gaussians $\mathcal{M} = w_1 G_1 + \cdots + w_k G_k$ with mean and covariance μ, Σ where $\delta \geq \epsilon^{f(k)}$, there is a polynomial time algorithm that with high probability outputs an invertible linear transformation L so that*

(1) $\|L(\mu)\|_2 \leq \text{poly}(\epsilon)$
(2) $\|I - L(\Sigma)\|_2 \leq \text{poly}(\epsilon)$.

Proof. We can first obtain estimates $\widehat{\mu}$ and $\widehat{\Sigma}$ using Lemma 5.6. We can then apply the linear transformation

$$L(x) = \widehat{\Sigma}^{-1/2}(x - \widehat{\mu})$$

It follows from direct computation that this transformation satisfies the desired properties. □

Once our mixture is placed in isotropic position, we will estimate the Hermite polynomials and then we will be able to use Theorem 4.1. The following lemma can be easily derived from the results in [39] (see Lemmas 2.7, 2.8, and 5.2 there).

Lemma 5.8. *Let $\mathcal{M}$ be a mixture of Gaussians $w_1 N(\mu_1, I + \Sigma_1) + \cdots + w_k N(\mu_k, I + \Sigma_k)$. Then*

$$\|\mathbb{E}_{z \sim \mathcal{M}}\left(v_X(H_m(X,z)) \otimes v_X(H_m(X,z))\right)]\|_2 = O_m(1 + \max\|\Sigma_i\|_2 + \max\|\mu_i\|)^{2m}$$

where $H_m(X,z)$ is defined as in Definition 3.4 and $v_X(H_m(X,z))$ denotes vectorizing as a polynomial in X so that the entries of the vector are polynomials in z.

Kane [39] works with Hermite polynomial tensors, which are tensorized versions of the Hermite polynomials we are using. It is clear that these two notions are equivalent up to $O_k(1)$ factors as long as m is $O_k(1)$ (writing them as formal polynomials instead of tensors simply collapses symmetric entries of the tensor but this collapses at most $O_m(1)$ entries together at once).

We can now combine everything in this section with Theorem 4.1 to complete the proof of Theorem 5.2.

Proof of Theorem 5.2. We can split the samples into $O(1)$ parts that are each $O(1)\epsilon$ corrupted samples. First, we use Corollary 5.7 to compute a transformation L that places the mixture in nearly isotropic position. Now Lemma 5.4 and Corollary 5.5 gives us bounds on how far each of the means is from 0 and how far each of the covariances is from I. We can apply Lemma 5.8 and standard results from robust estimation of bounded covariance distributions (see e.g., Theorem 2.2 in [39]) to obtain estimates $\overline{h_{m,L(\mathcal{M})}}(X)$ for the Hermite polynomials of the mixture $L(\mathcal{M})$ such that

$$\left\|v\left(\overline{h_{m,L(\mathcal{M})}}(X) - h_{m,L(\mathcal{M})}(X)\right)\right\|_2 \leq \text{poly}(\epsilon)$$

where m is bounded as a function of k. Now we must verify that the remaining hypotheses of Theorem 4.1 are satisfied with $\epsilon' = \text{poly}(\epsilon)$ for the transformed mixture $L(\mathcal{M})$.

- Corollary 5.5 gives the required upper bound on $\|L(\mu_i)\|$ and $\|L(\Sigma_i) - I\|$
- The first two conditions of Lemma 5.4, combined with Claim 5.3, imply the condition that no pair of components has essentially the same mean and covariance
- Finally, the mixing weights are unchanged by the linear transformation so the third condition is easily verified (since the original mixture is δ-well-conditioned)

Thus, by Theorem 4.1 we can obtain a list of $(1/\epsilon)^{O_k(1)}$ candidate mixtures at least one of which satisfies

$$||w_i - \widetilde{w_i}|| + ||L(\mu_i) - \widetilde{\mu_i}||_2 + ||L(\Sigma_i) - \widetilde{\Sigma_i}||_2 \leq \text{poly}(\epsilon)$$

for all i. By Claim 5.3, we know that the components we compute are close in TV to the true components. Now applying the inverse transformation L^{-1} to all of the components, we are done. □

6 ROUGH CLUSTERING

As mentioned earlier in the proof overview, the first step in our full algorithm will be to cluster the points. We present our clustering algorithm in this section. This section closely mirrors the work in [21]. We first define a measure of closeness between Gaussians that we will use throughout the paper.

Definition 6.1. We say that two Gaussians $N(\mu, \Sigma)$ and $N(\mu', \Sigma')$ are C-close if all of the following conditions hold:

(1) (mean condition) For all unit vectors $v \in \mathbb{R}^d$, we have $(v \cdot \mu - v \cdot \mu')^2 \leq C v^T(\Sigma + \Sigma')v$.
(2) (variance condition) For all unit vectors $v \in \mathbb{R}^d$, we have $\max(v^T \Sigma v, v^T \Sigma' v) \leq C \min(v^T \Sigma v, v^T \Sigma' v)$.
(3) (covariance condition) Finally, we have $||I - \Sigma'^{-1/2} \Sigma \Sigma'^{-1/2}||_2^2 \leq C$.

The main theorem that we aim to prove in this section is the following, which implies that if the true mixture can be well-clustered into submixtures, then we can recover this clustering with constant-accuracy.

Theorem 6.2. *Let k, D, γ, A be parameters. There exists a sufficiently small function f and a sufficiently large function F depending only on k, A, D, γ (with $f(k, A, D, \gamma), F(k, A, D, \gamma) > 0$) such that the following holds. Consider an unknown mixture of Gaussians $w_1 G_1 + \cdots + w_k G_k$ where the mixing weights w_i are all rational numbers with denominator bounded by a constant A and assume $A_1, \ldots, A_l$ is a partition of the components such that*

(1) For any j_1, j_2 in the same piece of the partition G_{j_1}, G_{j_2} are D-close
(2) For any j_1, j_2 in different pieces of the partition, G_{j_1}, G_{j_2} are not D'-close

where $D' > F(k, A, D, \gamma)$. Given an ϵ-corrupted sample $X_1, \ldots, X_n$ from the mixture, if we run Rough Clustering *(see Algorithm 1) with parameters $t > F(k, A, D, \gamma)$ and $\eta, \epsilon, \delta < f(k, A, D, \gamma)$, and $n \geq \text{poly}(1/\epsilon, 1/\eta, 1/\delta, d)^{O(k,A)}$, then with probability at least $1-\gamma$, one of the clusterings returned by the algorithm gives a γ-corrupted sample of each of the submixtures given by $A_1, \ldots, A_l$. Furthermore, the algorithm runs in time* poly(n).

Remark. Note that the last statement is well defined because the assumption about the partition essentially implies that all pairs of components in different submixtures are separated so γ-corrupted sample simply means correctly recovering a $1-\gamma$-fraction of the original points that were drawn from the corresponding submixture.

In this section, it will suffice to consider when the mixing weights are equal as we can subdivide one component into many identical ones so from now on we assume $w_1 = \cdots = w_k = 1/k$ and all dependencies on A become dependencies on k.

We begin with a few preliminaries. The following claim is a simple consequence of the definition.

Claim 6.3. *Let G_1, G_2, G_3 be Gaussians such that G_1 and G_2 are C-close and G_2 and G_3 are C-close. Then G_1 and G_3 are* $\mathrm{poly}(C)$*-close.*

Proof. The second condition follows immediately from the fact that G_1 and G_2 are C-close and G_2 and G_3 are C-close. Now we know that for all vectors v, $v^T\Sigma_1 v, v^T\Sigma_2 v, v^T\Sigma_3 v$ are all within a $\mathrm{poly}(C)$ factor of each other. This means that the singular values of $\Sigma_i^{1/2}\Sigma_j^{-1/2}$ are all bounded above and below by $\mathrm{poly}(C)$. From this and the triangle inequality, we get the first and third conditions. □

The next claim follows immediately from Lemma 3.6 in [21].

Claim 6.4. *There is a decreasing function f such that $f(C) > 0$ for all $C > 0$ such that if two Gaussians G_1, G_2 are C-close then*

$$d_{TV}(G_1, G_2) \le 1 - f(C)$$

We will now show that either all pairs in the mixture are not too far apart, or there exists a nontrivial partition of the mixture into two parts that are separated in either mean, variance in some direction, or covariance. This parallels Corollary 3.7 in [21]. However, we require a slightly different statement because their paper specializes to the case where all pairs of components are separated. We use μ, Σ to denote the mean and covariance of the overall mixture.

Claim 6.5. *Let $C > 100$ be a constant. Let C_k be a sufficiently large constant depending only on C and k. Assume that there are $i, j \in [k]$ such that $N(\mu_i, \Sigma_i)$ and $N(\mu_j, \Sigma_j)$ are not C_k-close. Then there exists a partition of $[k]$ into two disjoint sets S, T such that for any $a \in S, b \in T$, $N(\mu_a, \Sigma_a)$ is not k^C-close to $N(\mu_b, \Sigma_b)$ and at least one of the following holds:*

(1) There is a direction v such that for all $a \in S, b \in T$,

$$((\mu_a - \mu_b)\cdot v) \ge \max\left(k^C(v^T(\Sigma_a + \Sigma_b)v), \frac{v^T\Sigma v}{k^2}\right)$$

(2) There is a direction v such that for all $a \in S, b \in T$,

$$\frac{v^T\Sigma_a v}{v^T\Sigma_b v} \ge k^C \text{ and } \frac{v^T\Sigma_a v}{v^T\Sigma v} \ge \frac{1}{k^{2Ck}}$$

(3) We have

$$||I - \Sigma_a^{-1/2}\Sigma_b\Sigma_a^{-1/2}||^2 \ge k^C \max\left(||\Sigma_a^{1/2}A_{ab}\Sigma_a^{1/2}||, ||\Sigma_b^{1/2}A_{ab}\Sigma_b^{1/2}||, ||\Sigma^{1/2}A_{ab}\Sigma^{1/2}||\right)$$

where $A_{ab} = \Sigma_a^{-1/2}\left(I - \Sigma_a^{-1/2}\Sigma_b\Sigma_a^{-1/2}\right)\Sigma_a^{-1/2}$.

Proof. We break into a few cases:

Case 1: Suppose that there is a v such that for some a, b

$$((\mu_a - \mu_b) \cdot v)^2 \geq 10k^2 \cdot k^C \max_i (v^T \Sigma_i v)$$

then we claim we are done. To see this, first observe that

$$v^T \Sigma v = \frac{1}{k^2} \sum_{i \neq j} ((\mu_i - \mu_j) \cdot v)^2 + \frac{1}{k} \sum v^T \Sigma_i v$$

so then choosing a, b such that $((\mu_a - \mu_b) \cdot v)^2$ is maximal, we have $((\mu_a - \mu_b) \cdot v)^2 \geq 0.1 v^T \Sigma v$. Now we can partition the components based on the value of $\mu_i \cdot v$. We can ensure that the gap between the clusters has size at least $\frac{(\mu_a - \mu_b) \cdot v}{k}$. This will imply for all $a \in S, b \in T$

$$((\mu_a - \mu_b) \cdot v)^2 \geq k^C v^T (\Sigma_a + \Sigma_b) v)$$

i.e., the corresponding components are not k^C-close. Since we can choose C_k sufficiently large, the first condition is also satisfied and we are done in this case.

Case 2: Alternatively suppose there is a v such that

$$\frac{\max_i (v^t \Sigma_i v)}{\min_i (v^T \Sigma_i v)} \geq k^{4Ck}$$

In this case, we can partition the components based on the value of $v^T \Sigma_i v$. Without loss of generality we have

$$v^T \Sigma_1 v \geq \cdots \geq v^T \Sigma_k v$$

Note that since we are not in the first case $v^T \Sigma_1 v \geq \frac{v^T \Sigma v}{20k^{2+C}}$. Next, because $\frac{v^T \Sigma_k v}{v^T \Sigma v} \leq \frac{1}{k^{2Ck}}$ there must be some consecutive $i, i+1$ such that

$$\left(\frac{v^T \Sigma_i v}{v^T \Sigma v} \right) \geq k^C \left(\frac{v^T \Sigma_{i+1} v}{v^T \Sigma v} \right) \text{ and } \left(\frac{v^T \Sigma_i v}{v^T \Sigma v} \right) \geq \frac{1}{k^{2Ck}}$$

partitioning into $S = \{1, 2, \dots, i\}$ and $T = \{i+1, \dots, k\}$, we immediately verify that the desired conditions (second condition) are satisfied.

Case 3: Finally, it remains to consider the situation where neither the condition in Case 1 nor the condition in Case 2 holds. Note that by assumption, there is some pair $a, b \in [k]$ for which $N(\mu_a \Sigma_a), N(\mu_b, \Sigma_b)$ are not C_k-close. Since we can choose

$$C_k > (kC)^{10kC}$$

this pair cannot fail the variance condition in any direction (second condition of Definition 6.1). This pair also cannot fail the mean condition in any direction (first condition of Definition 6.1) because then we would have

$$((\mu_a - \mu_b) \cdot v)^2 \geq C_k v^T \Sigma_a v \geq \frac{C_k}{k^{4Ck}} \max_i (v^T \Sigma_i v)$$

and we would be in the first case. Thus, we must actually have

$$\left\| I - \Sigma_a^{-1/2} \Sigma_b \Sigma_a^{-1/2} \right\|_2^2 \geq C_k$$

Next, we claim that for all i, j, $\Sigma_i^{1/2} \Sigma_j^{-1/2}$ has smallest and largest singular value in the interval

$$\mathcal{I} \triangleq \left[\frac{1}{k^{4Ck}}, k^{4Ck} \right]$$

If this were not true, without loss of generality we can find a unit vector v such that $\left\|\Sigma_i^{1/2}\Sigma_j^{-1/2}v\right\|_2 \geq k^{4Ck}$. But this implies

$$\frac{(\Sigma_j^{-1/2}v)^T\Sigma_i(\Sigma_j^{-1/2}v)}{(\Sigma_j^{-1/2}v)^T\Sigma_j(\Sigma_j^{-1/2}v)} \geq k^{8Ck}$$

meaning we are actually in Case 2. Similarly, we can show that $\Sigma_i^{1/2}\Sigma^{-1/2}$ has smallest and largest singular value in the interval $\mathcal{I}$ or else we would be in Case 1.

To complete the proof, let a_0, b_0 be indices corresponding to a pair of components that are not C_k-close and construct the following graph. Two nodes i, j are connected if and only if

$$\left\|\Sigma_{a_0}^{-1/2}\Sigma_j\Sigma_{a_0}^{-1/2} - \Sigma_{a_0}^{-1/2}\Sigma_i\Sigma_{a_0}^{-1/2}\right\|_2^2 \leq \frac{C_k}{k^2}$$

This graph must not be connected since otherwise there would be a path of length at most k between a_0 and b_0 and summing the above inequalities along this path, this would contradict the fact that

$$\left\|I - \Sigma_{a_0}^{-1/2}\Sigma_{b_0}\Sigma_{a_0}^{-1/2}\right\|_2^2 \geq C_k.$$

We claim that it suffices to take S and T to be two connected components of the graph. Indeed, for any $a \in S, b \in T$, we have

$$\left\|\Sigma_{a_0}^{-1/2}\Sigma_a\Sigma_{a_0}^{-1/2} - \Sigma_{a_0}^{-1/2}\Sigma_b\Sigma_{a_0}^{-1/2}\right\|_2^2 \geq \frac{C_k}{k^2}$$

Now observe

$$I - \Sigma_a^{-1/2}\Sigma_b\Sigma_a^{-1/2} = (\Sigma_a^{-1/2}\Sigma_{a_0}^{1/2})\left(\Sigma_{a_0}^{-1/2}\Sigma_a\Sigma_{a_0}^{-1/2} - \Sigma_{a_0}^{-1/2}\Sigma_b\Sigma_{a_0}^{-1/2}\right)(\Sigma_{a_0}^{1/2}\Sigma_a^{-1/2})$$

and combining with the singular value bounds we showed for $\Sigma_i^{1/2}\Sigma_j^{-1/2}$ and $\Sigma_i^{1/2}\Sigma^{-1/2}$, we have

$$\left\|I - \Sigma_a^{-1/2}\Sigma_b\Sigma_a^{-1/2}\right\|_2^2 \geq \max\left(k^C, k^C\,\|A_{ab}\|\right)$$

for any a, b on different sides of the partition. The other quantities in the third condition can be bounded similarly as long as C_k is chosen to be sufficiently large. □

6.1 SOS Program

To solve the clustering problem, we set up the same polynomial constraints as in Diakonikolas et al. [21]. Recall that Definition 2.4 gives a recipe for turning this into an SDP relaxation.

Definition 6.6 (Clustering Program $\mathcal{A}$, Restated from [21]). Let $X_1, \ldots, X_n \in \mathbb{R}^d$ represent the samples. Let $w_1, \ldots, w_n, z_1, \ldots, z_n, X_1', \ldots, X_n'$ and $\Sigma, \Sigma^{1/2}, \Sigma^{-1/2} \in \mathbb{R}^{d\times d}$ (we think of the Σ as $d \times d$ matrices whose entries are variables) be indeterminates that we will solve for in the system. We think of the w variables as weights on the points and the z variables as representing whether points are outliers. We will enforce that the subset of points weighted by w has moments that are approximately Gaussian. The full system of polynomial constraints is given below:

(1) We have parameters $t \in \mathbb{N}$ that is even and $\delta, \epsilon > 0$.
(2) Let $\mathcal{A}_{\text{corruptions}} = \{z_i^2 = z_i\}_{i\in[n]}, \{z_i(X_i - X_i') = 0\}_{i\in[n]}, \{\sum_{i\in[n]} z_i = (1-\epsilon)n/k\}$
(3) Let $\mathcal{A}_{\text{subset}} = \{w_i^2 = w_i\}_{i\in[n]}, \{\sum_{i\in[n]} w_i = n/k\}$
(4) Let $\mu(w) = \frac{k}{n}\sum_{i\in[n]} w_i X_i'$
(5) Let $\Sigma(w) = \frac{k}{n}\sum_{i\in[n]} w_i(X_i' - \mu(w))(X_i' - \mu(w))^T$
(6) Let $\mathcal{A}_{\text{matrices}} = \{(\Sigma^{1/2})^2 = \Sigma(w)\}, \{(\Sigma^{-1/2}\Sigma^{1/2})^2 = \Sigma^{-1/2}\Sigma^{1/2}\}, \{\Sigma^{-1/2}\Sigma^{1/2}w_i(X_i' - \mu(w)) = w_i(X_i' - \mu(w))\}_{i\in[n]}$

(7) Let $\mathcal{A}_{\text{moments}}$ be the following set of polynomial inequalities for all $s \leq t$

$$\left\| \frac{k}{n} \sum_{i \in [n]} w_i [\Sigma^{-1/2}(X_i' - \mu(w))]^{\otimes s} - M_s \right\|^2 \leq \delta d^{-2t}$$

where $M_s = \mathbb{E}_{g \in N(0,I)}[g^{\otimes s}]$ is the moment tensor of a standard Gaussian.

We will work with the same set of deterministic conditions on the samples as in Diakonikolas et al. [21]. These conditions hold with high probability for the uncorrupted points.

Definition 6.7 (Deterministic Conditions, Restated from [21]). Fix Gaussians $G_1, \dots, G_k$ on $\mathbb{R}^d$. For $\delta, \psi > 0$ and $t \in \mathbb{N}$. The (δ, ψ, t)-deterministic conditions with respect to $G_1, \dots, G_k$ on a set of samples $X_1, \dots, X_n \in \mathbb{R}^d$ are

(1) There is a partition of $\{X_1, \dots, X_n\}$ into k pieces $S_1, \dots, S_k$ each of size n/k such that for all $i \in [k]$ and $s \leq t$

$$\left\| \frac{k}{n} \sum_{j \in S_i} [\overline{\Sigma}_i^{-1/2}(X_j - \overline{\mu}_i)]^{\otimes s} - M_s \right\|_F^2 \leq d^{-2t}\delta$$

where $\overline{\Sigma}_i$ and $\overline{\mu}_i$ denote the empirical mean and covariance of the uniform distribution over elements of S_i and $M_s = \mathbb{E}_{g \in N(0,I)}[g^{\otimes s}]$ is the moment tensor of a standard Gaussian.

(2) For $a \in [k], v \in \mathbb{R}^d, A \in \mathbb{R}^{d \times d}$ we define
  (a) $E_a(v) = \{X_i \in S_a | ((X_i - \mu_a) \cdot v)^2 \leq O(1) \log(1/\psi) v^T \Sigma_a v\}$
  (b) $F_a(v) = \{(X_i, X_j) \in S_a^2 | ((X_i - X_j) \cdot v)^2 \geq \Omega(1) \cdot \psi v^T \Sigma_a v\}$
  (c) $G_a(A) = \{(X_i, X_j) \in S_a^2 | (X_i - X_j)^T A (X_i - X_j) = 2\langle \Sigma_a, A \rangle \pm O(1) \log(1/\psi) \cdot \|\Sigma_a A\|_F\}$.

  Then for every $v \in \mathbb{R}^d, A \in \mathbb{R}^{d \times d}$ we have
  - $|E_a(v)| \geq (1 - \psi)(n/k)$
  - $|F_a(v)|, |G_a(A)| \geq (1 - \psi)(n/k)^2$.

Claim 6.8 (Restated from [21]). *For all even t, if*

$$n \geq \log(1/\gamma)^{Ct} d^{10kt}/\delta^2$$

for some sufficiently large constant C and $\psi \geq \delta$, then $X_1, \dots, X_n$ drawn i.i.d from $\frac{1}{k}\sum_{i=1}^k G_i$ satisfy Definition 6.7 with probability at least $1 - \gamma$.

We will use the following key lemmas from [21]. The setup is exactly the same. Let $X_1, \dots, X_n \in \mathbb{R}^d$ satisfy the (δ, ψ, t)-deterministic conditions (Definition 6.7) with respect to Gaussians $G_1, \dots, G_k$. Let $S_1, \dots, S_k$ be the partition guaranteed in the definition. Let $Y_1, \dots, Y_n$ be an ϵ-corruption of $X_1, \dots, X_n$ and let $\mathcal{A}$ be the clustering program (Definition 6.6) for $Y_1, \dots, Y_n$. For indeterminates $w_1, \dots, w_n$, define

$$\alpha_i(w) = \sum_{j \in S_i} w_j.$$

Below we will assume ψ, τ are smaller than some universal constants $\psi_0, \tau_0 > 0$.

Recall in Claim 6.5 that there are essentially three different ways that two Gaussians can be separated in TV distance. We call these mean separation, variance separation, and covariance separation. The lemmas below roughly assert that if two Gaussians are separated in one of these ways, then a valid solution to the clustering program $\mathcal{A}$ cannot assign significant weight to both of them.

Lemma 6.9 (Mean Separation, Restated from [21]). *For every $\tau > 0$, there is $s = \widetilde{O}(1/\tau^2)$ such that if $\epsilon, \delta \leq s^{-O(s)} k^{-20}$ then for all $a, b \in [k]$, all $v \in \mathbb{R}^d$ and all sufficiently small $\rho > 0$, if*

$$\langle \mu_a - \mu_b, v \rangle^2 \geq \rho \mathbb{E}_{X, X' \sim \frac{1}{k}\sum G_i} \langle X - X', v \rangle^2,$$

then

$$\mathcal{A} \vdash_{O(s)} \left(\frac{\alpha_a(w)\alpha_b(w)}{n^2}\right)^s \leq (s \log 1/\psi)^{O(s)} \cdot \left(\frac{\langle v, \Sigma_a v\rangle + \langle v, \Sigma_b v\rangle}{\langle \mu_a - \mu_b, v\rangle^2}\right)^{\Omega(s)} + \rho^{-O(s)}(\tau^{\Omega(s)} + \epsilon^{\Omega(s)} k^{O(s)} s^{O(s^2)} + \psi^{\Omega(s)}).$$

Lemma 6.10 (Variance Separation, Restated from [21]). *For every $\tau > 0$, there is $s = \widetilde{O}(1/\tau^2)$ such that if $\epsilon, \delta \leq s^{-O(s)} k^{-20}$ then for all $a, b \in [k]$, all $v \in \mathbb{R}^d$ and all sufficiently small $\rho > 0$, if*

$$\langle v, \Sigma_b v\rangle \geq \rho \mathbb{E}_{X, X' \sim \frac{1}{k}\sum G_i} \langle X - X', v\rangle^2,$$

then

$$\mathcal{A} \vdash_{O(s)} \left(\frac{\alpha_a(w)\alpha_b(w)}{n^2}\right)^s \leq \psi^{-O(s)} \cdot \left(s^{O(s)} \left(\frac{\langle v, \Sigma_a v\rangle}{\langle v, \Sigma_b v\rangle}\right)^{\Omega(s)} + \rho^{-O(s)}(\tau^{\Omega(s)} + \epsilon^{\Omega(s)} k^{O(s)} s^{O(s^2)})\right) + \rho^{-O(s)} \psi^{\Omega(s)}.$$

Lemma 6.11 (Covariance Separation, Restated from [21]). *Let Σ be the covariance of the mixture $\frac{1}{k}\sum G_i$. If $\epsilon, \delta < k^{-O(1)}$, then for all $a, b \in [k]$ and $A \in \mathbb{R}^{d \times d}$,*

$$\mathcal{A} \vdash_{O(1)} \left(\frac{\alpha_a(w)\alpha_b(w)}{n^2}\right)^{16} \leq O(\log 1/\psi)^8 \cdot \frac{\left\|\Sigma^{1/2} A \Sigma^{1/2}\right\|_F^8 + \left\|\Sigma_a^{1/2} A \Sigma_a^{1/2}\right\|_F^8 + \left\|\Sigma_b^{1/2} A \Sigma_b^{1/2}\right\|_F^8}{\langle \Sigma_a - \Sigma_b A\rangle^8}$$
$$+O(\psi^4) + O(\epsilon^2 k^{20}).$$

6.2 Clustering Algorithm

We use essentially the same clustering algorithm as [21].

ALGORITHM 1: Rough Clustering

```
Input: $\epsilon$-corrupted samples $X_1, \ldots, X_n$ and parameters $t, \delta, \epsilon, k, \eta$
Initialize a list of subsets $L = \{\}$
for count $= 0, 1, \ldots, 100k \log 1/\eta$ do
    Let $\mathcal{A}$ be the clustering program (Definition 6.6) for $X_1, \ldots, X_n$
    Compute the pseudoexpectation $\widetilde{E}$ that satisfies the constraints $\mathcal{A}$ (Definition 6.6) and maximizes
        $\widetilde{E}\left[\sum_{i \notin \cup_{R \in L} R} w_i\right]$
    Choose a random $i \sim [n]$ with probability $p_i = \frac{\widetilde{\mathbb{E}}[w_i]}{\sum \widetilde{\mathbb{E}}[w_i]}$
    Create set $R$ by adding each element $j \in [n]$ independently with probability $\frac{\widetilde{\mathbb{E}}[w_i w_j]}{\widetilde{\mathbb{E}}[w_i]}$
    Add $R$ to the list $L$
Let $L = \{R_1, \ldots, R_m\}$
for all subsets $S \subset L$ do
    Recurse on $\cup_{i \in S} R_i$ for each of $k \to 1, 2, \ldots, k-1$ and $t, \delta, \epsilon, \eta$ unchanged
Return $\{X_1, \ldots, X_n\}$ (as one cluster) and all unions of some combination of the clusters returned
in each computation branch
```

Proof of Theorem 6.2. We can use Claim 6.8 to ensure that with $1 - \gamma/2$ probability, the deterministic conditions in Definition 6.7 are satisfied for all submixtures and the various values of δ, ψ, t that we will need.

First, if all pairs of components are D' close, then returning the entire sample as one cluster suffices. Now, we may assume that there is some pair that is not D'-close. We apply Claim 6.5 and let U, V be the partition of the components given by the claim. Let C be a sufficiently large function of k, D, γ that we will set later. We can do this as long as we ensure that D' is a sufficiently large function of k, C. We ensure that $k^C > D$. Note that each of the pieces $A_1, \dots, A_l$, must be entirely in U or in V because of our assumption about closeness between the components. We claim that

$$\widetilde{\mathbb{E}}\left[\left(\sum_{i \in \cup_{j \in U} S_j} w_i\right)\left(\sum_{i \in \cup_{j \in V} S_j} w_i\right)\right] \leq \gamma' n^2 \tag{4}$$

where we can make γ' sufficiently small in terms of γ, D, k by choosing D' and the functions f, F suitably.

Below we will let a, b be indices such that $a \in U$ and $b \in V$. If the first clause of Claim 6.5 is satisfied, then we can take $\rho = poly(1/k)$ and for τ sufficiently small in terms of γ, k, D, C, we have

$$\widetilde{\mathbb{E}}\left(\frac{\alpha_a(w)\alpha_b(w)}{n^2}\right)^s \leq k^{-C/2s}$$

Summing over all $a \in U, b \in V$, this gives (4).

If the second clause of Claim 6.5 is satisfied, then we can take

$$\rho = \min_{a \in U} \frac{v^T \Sigma_a v}{v^T \Sigma v} \geq \frac{1}{k^{2Ck}}$$

We choose τ sufficiently small in terms of γ, k, C, D and combining with the fact that $(v^t \Sigma_a v) \geq k^C (v^T \Sigma_b v)$ for all $a \in U, b \in V$ we get

$$\widetilde{\mathbb{E}}\left(\frac{\alpha_a(w)\alpha_b(w)}{n^2}\right)^s \leq k^{-\Omega(C)s}$$

Finally, when the third clause of Claim 6.5 is satisfied it follows similarly after setting $A = A_{ab}$. In all cases, we now have (4). The next step will be to analyze our random sampling to select the subset R. First note

$$\widetilde{\mathbb{E}}[|R|] = \frac{\sum_{i,j} \widetilde{\mathbb{E}}[w_i w_j]}{\sum_i \widetilde{\mathbb{E}}[\sum_i w_i]} = \frac{n}{k}$$

Next we analyze the intersections with the two sides of the partition U, V. We will slightly abuse notation and use $i \in U$ when $i \in \cup_{j \in U} S_j$ and it is clear from context that we are indexing the samples. Conditioned on the first index that is randomly chosen satisfying $i \in U$ then

$$\mathbb{E}[|R \cap V|] = \frac{\sum_{i_1 \in U, i_2 \in V} \widetilde{\mathbb{E}}[w_{i_1} w_{i_2}]}{\sum_{i \in U} \widetilde{\mathbb{E}}[w_i]} \leq \frac{\gamma' n^2}{\sum_{i \in U} \widetilde{\mathbb{E}}[w_i]}$$

repeating the same argument for when $i \in V$, we have $\mathbb{E}[\min(|R \cap U|, |R \cap V|)] \leq \gamma' kn$. Finally, we lower bound the expected number of new elements that R adds to the list L. This quantity is

$$\frac{\widetilde{\mathbb{E}}[\sum_{i \in [n], j \notin L} w_i w_j]}{n/k} = \sum_{j \notin L} \widetilde{\mathbb{E}}[w_j]$$

where by $j \notin L$ we mean j is not in the union of all previous subsets in the list L. Note that indicator functions of the components $S_1, \dots, S_k$ are all valid pseudoexpectations and since we are picking the pseudoexpectation that maximizes $\sum_{j \notin L} \widetilde{\mathbb{E}}[w_j]$, the expected number of new elements added to L is at least

$$\frac{n - |\cup_{R \in L} R|}{k}$$

Now we analyze the recombination step once we finalize $L = \{R_1, \dots, R_m\}$. For any sufficiently small function $h(k, \gamma, D)$, we claim that by choosing D' and the functions f, F appropriately, we can ensure with $1 - h(k, D, \gamma)$ probability, there is some recombination that gives a $1 - h(k, D, \gamma)$-corrupted sample of the submixture corresponding to U. To see this, it suffices to set $\eta < h(k, D, \gamma)$

and then look at the first $m' = 100k \log 1/h(k, D, \gamma)$ subsets in L. Their union has expected size $(1 - h(k, D, \gamma)^{100})n$. Next, among $R_1, \dots, R_{m'}$,

$$\mathbb{E}\left[\sum_{i=1}^{m'} \min(|R_i \cap U|, |R_i \cap V|)\right] \le \gamma' k m' n$$

If we ensure that γ' is sufficiently small in terms of γ, D, k, then using Markov's inequality, with $1-h(k, D, y)$ probability, there is some recombination that gives a $1-h(k, D, \gamma)$-corrupted sample of the submixture corresponding to U. We can make the same argument for V. Now we can recurse and repeat the argument because each of these submixtures only contains at most $k - 1$ true components. □

6.3 Improved Clustering Result from [4]

In [4], the authors obtain a rough clustering result similar to Theorem 6.2 but are able to remove the bounded fractionality assumption. Their result is restated using our notation below.

Theorem 6.12 ([4]). *Let k, D, γ be parameters. Assume we are given ϵ-corrupted samples from a mixture of Gaussians $w_1 G_1 + \cdots + w_k G_k$ where the mixing weights w_i are all at least $1/A$ for some constant A. Let $A_1, \dots, A_l$ be a partition of the components such that*

(1) For any j_1, j_2 in the same piece of the partition G_{j_1}, G_{j_2} are D-close
(2) For any j_1, j_2 in different pieces of the partition, G_{j_1}, G_{j_2} are not D'-close

where $D' > F(k, A, D, \gamma)$ for some sufficiently large function F. Assume that $t > F(k, A, D, \gamma)$ and $\eta, \epsilon, \delta < f(k, A, D, \gamma)$ for some sufficiently small function f. Then with probability at least $1 - \gamma$, if $X_1, \dots, X_n$ is an ϵ-corrupted sample from the mixture $w_1 G_1 + \cdots + w_k G_k$ with $n \ge \mathrm{poly}(1/\epsilon, 1/\eta, 1/\delta, d)^{O(k,A)}$, then there is an algorithm that runs in $\mathrm{poly}(n)$ time and returns $O_k(1)$ candidate clusterings, at least one of which gives a γ-corrupted sample of each of the submixtures given by $A_1, \dots, A_l$.

Observe that Theorem 6.12 is the same as Theorem 6.2 but with the bounded fractionality assumption removed. Theorem 6.2 is the only source of the bounded fractionality assumption in our paper. In the subsequent sections, replacing all uses of Theorem 6.2 with Theorem 6.12 allows us to remove the bounded fractionality assumption from our main result.

7 PUTTING EVERYTHING TOGETHER

We can now combine our clustering results and our results for learning mixtures of Gaussians that are not too separated to get a learning algorithm in the fully general case. Our main theorem is stated below.

Theorem 7.1. *Let $k, A, b > 0$ be constants. There is a sufficiently large function G and a sufficiently small function g depending only on k, A, b (with $G(k, A, b), g(k, A, b) > 0$) such that given an ϵ-corrupted sample $X_1, \dots, X_n$ from a mixture of Gaussians $\mathcal{M} = w_1 G_1 + \cdots + w_k G_k \in \mathbb{R}^d$ where the G_i have variance at least $\mathrm{poly}(\epsilon/d)$ and at most $\mathrm{poly}(d/\epsilon)$ in all directions and*

- *The w_i are all rational with denominator at most A*
- *$d_{TV}(G_i, G_j) \ge b$*

and $n \ge (d/\epsilon)^{G(k,A,b)}$, then there is an algorithm that runs in time $\mathrm{poly}(n)$ and with 0.99 probability outputs a mixture

$$\widetilde{\mathcal{M}} = \widetilde{w_1}\widetilde{G_1} + \cdots + \widetilde{w_k}\widetilde{G_k}$$

such that $d_{TV}(\widetilde{\mathcal{M}}, \mathcal{M}) \le \epsilon^{g(k,A,b)}$.

7.1 Distance Between Gaussians

We will need to prove a few preliminary results. The main lemma we prove in this section is the following, which gives a stronger bound than the triangle inequality for TV distance between Gaussians.

Lemma 7.2. *Let λ be a constant. Let A, B, C be Gaussian distributions. Assume that $d_{TV}(A, B) \leq 1-\lambda$. If $d_{TV}(A, C) \geq 1 - \epsilon$ and ϵ is sufficiently small then*

$$d_{TV}(B, C) \geq 1 - \text{poly}(\epsilon)$$

(where the RHS may depend on λ).

Note that this result is not true for arbitrary distributions A, B, C. We actually need to exploit the fact that A, B, C are Gaussian.

Our proof will parallel results in Section 8 of [21]. First, a definition:

Definition 7.3. For two distributions P, Q let

$$h(P, Q) = -\log(1 - d_{\text{TV}}(P, Q)).$$

The key ingredient is the following result from [21]:

Lemma 7.4 (Restated from [21]). *Let A and B be two Gaussians with $h(A, B) = O(1)$. If $D \in \{A, B\}$ then*

$$P_{x\sim D}\left[\epsilon \leq \frac{A(x)}{B(x)} \leq \frac{1}{\epsilon}\right] \geq 1 - \text{poly}(\epsilon)$$

where $A(x), B(x)$ denote the probability density functions of the respective Gaussians at x.

Proof of Lemma 7.2. Note that

$$P_{x\sim A}\left[\epsilon^{0.5} \leq \frac{A(x)}{C(x)} \leq \frac{1}{\epsilon^{0.5}}\right] \leq \epsilon^{0.5}$$

If this weren't the case, then A and C would have more than ϵ overlap, contradicting our assumption. Next, by Lemma 7.4,

$$P_{x\sim A}\left[\epsilon^{0.1} \leq \frac{A(x)}{B(x)} \leq \frac{1}{\epsilon^{0.1}}\right] \geq 1 - \text{poly}(\epsilon) \tag{5}$$

Combining the above two inequalities, we deduce

$$P_{x\sim A}\left[\epsilon^{0.4} \leq \frac{C(x)}{B(x)} \leq \frac{1}{\epsilon^{0.4}}\right] \leq \text{poly}(\epsilon) \tag{6}$$

Let $0 < c < 0.1$ be a constant such that the RHS of (6) is at most ϵ^c. By Lemma 7.4

$$P_{x\sim B}\left[\epsilon^{c/2} \leq \frac{A(x)}{B(x)} \leq \frac{1}{\epsilon^{c/2}}\right] \geq 1 - \text{poly}(\epsilon)$$

and combining with (6), we deduce

$$P_{x\sim B}\left[\epsilon^{0.4} \leq \frac{C(x)}{B(x)} \leq \frac{1}{\epsilon^{0.4}}\right] \leq \text{poly}(\epsilon)$$

which implies $d_{\text{TV}}(B, C) \geq 1 - \text{poly}(\epsilon)$. □

7.2 Full Algorithm

We are now ready to prove Theorem 7.1. We begin by describing the algorithm. Our full algorithm consists of several phases.

(1) Cluster with constant accuracy into constant-separated submixtures
(2) Learn parameters of submixtures to constant accuracy
(3) Recluster all points and form new poly(ϵ)-separated submixtures
(4) Learn parameters of submixtures to poly(ϵ) accuracy

The algorithm Learn Parameters (well-conditioned) for learning the parameters of a well-conditioned mixture of Gaussians (see Theorem 5.2) is summarized in Algorithm 2.

ALGORITHM 2: Learn Parameters (well-conditioned)

```
Input: $\epsilon$-corrupted sample $X_1, \dots, X_n$ from $\delta$-well-conditioned mixture of Gaussians $\mathcal{M} = w_1 G_1 + \cdots + w_k G_k$
Estimate Hermite polynomials of $\mathcal{M}$
Solve for parameters using SOS (see Section 4)
```

Our full algorithm is described in the next block Algorithm 3.

ALGORITHM 3: Full Algorithm

```
Input: $\epsilon$-corrupted sample $X_1, \dots, X_n$ from mixture of Gaussians $\mathcal{M} = w_1 G_1 + \cdots + w_k G_k$
Run Rough Clustering Algorithm to split sample into subsamples for submixtures where all pairs are $D$-close for constant $D$
for each candidate clustering do
    Run Learn Parameters (well-conditioned) for each submixture
    Output candidate components
for each set of candidate components $\widetilde{G_1}, \dots \widetilde{G_k}$ do
    Assign samples to components according to maximum likelihood to form sets of samples $\{\widetilde{S_1}, \dots, \widetilde{S_k}\}$
    for all partitions of $[k]$ into sets $R_1, \dots, R_l$ do
        Run Learn Parameters (well-conditioned) on each of $\cup_{i \in R_j} S_i$ for all $j \in [l]$
        Output candidate components
Hypothesis test over all candidate components to find a mixture $\widetilde{\mathcal{M}}$ that is poly($\epsilon$)-close to $\mathcal{M}$
```

7.3 Analysis of Full Algorithm

The first step will be to show that among the first set of candidate components that we output, there are some that are within constant distance (say $< c(k)$ for some sufficiently small function c) of the true components.

Lemma 7.5. *Let $k, A, b > 0$ be constants and θ be a desired accuracy. There is a sufficiently large function G and a sufficiently small function g depending only on k, A, b, θ such that given an ϵ-corrupted sample $X_1, \dots, X_n$ from a mixture of Gaussians $\mathcal{M} = w_1 G_1 + \cdots + w_k G_k \in \mathbb{R}^d$ where*

- *The w_i are all rational with denominator at most A*
- *$d_{TV}(G_i, G_j) \geq b$*

and

- *$\epsilon < g(k, A, b, \theta)$*
- *$n \geq (d/\epsilon)^{G(k, A, b, \theta)}$*

then there is an algorithm that runs in time poly(n) *and with* 0.999 *probability outputs a set of* $(1/\theta)^{G(k,A,b,\theta)}$ *candidate mixtures at least one of which satisfies*

$$\max\left(d_{TV}(\widetilde{G_1},G_1),\dots,d_{TV}(\widetilde{G_k},G_k)\right)\leq\theta$$
$$\max\left(|\widetilde{w_1}-w_1|,\dots,|\widetilde{w_k}-w_k|\right)\leq\theta$$

Proof. We will use Theorem 6.2 to argue that the clustering algorithm finds some set of candidate clusters that can then be used to learn the parameters via Theorem 5.2. The main thing we need to prove is that we can find the D, D' satisfying the hypotheses of Theorem 6.2. In the argument below, all functions may depend on k, A, b, θ but we may omit writing some of these variables in order to highlight the important dependences.

Note that Claim 6.4 combined with Theorem 5.2 imply that if we have a γ-corrupted sample of a submixture of $\mathcal{M}$ where all pairs are D-close and $\gamma < f(D,\theta)$ for some sufficiently small function f then we can learn the components of the submixture to the desired accuracy. Now if the separation condition of Theorem 6.2 were satisfied with $\gamma = f(D,\theta)$ and $D' > F(k, D, \gamma)$ then we would be done.

We now show that there is some constant D depending only on k, A, b, θ for which this is true. Assume that the condition does not hold for some value of D_0. Then construct a graph G_{D_0} on nodes $1, 2, \dots, k$ where two nodes are connected if and only if they are D-close. Take the connected components in this graph. Note that by Claim 6.3, all pairs in the same connected component are poly(D_0)-close. Thus, there must be an edge between two components such that G_i and G_j are D_1-close for

$$D_0 < D_1 < F(k, \mathrm{poly}(D_0), f(\mathrm{poly}(D_0),\theta))$$

Now the graph G_{D_1} has one less connected component than G_{D_0}. Starting from say $D_0 = 2$, we can iterate this argument and deduce that the entire graph will be connected for some constant value of D depending only on k, A, b, θ. Now by Claim 6.3 it suffices to treat the entire mixture as one mixture and we can apply Claim 6.4 and Theorem 5.2 to complete the proof. □

Our next step is to show that if our algorithm starts with component estimates that are accurate within some constant and guesses a good set of clusters, then the resulting subsamples (after assigning according to maximum likelihood) are equivalent to poly(ϵ)-corrupted samples from the corresponding submixtures. First, we prove a preliminary claim which implies that a good set of clusters exists.

Claim 7.6. *Let $\mathcal{M} = w_1G_1 + \cdots + w_kG_k$ be a mixture of Gaussians. For any constant $c > 0$ and parameter ϵ, there exists a function $f(c,k)$ such that there exists a partition (possibly trivial) of $[k]$ into sets $R_1, \dots, R_l$ such that*

- *If we draw edges between all i, j such that $d_{TV}(G_i, G_j) \leq 1 - \epsilon^{c\kappa}$ then each piece of the partition is connected*
- *For any i, j in different pieces of the partition $d_{TV}(G_i, G_j) \geq 1 - \epsilon^{\kappa}$*

and $f(c,k) < \kappa < 1$.

Proof. For a real number f, let $\mathcal{G}_f$ be the graph on $[k]$ obtained by connecting two nodes i, j if and only if $d_{\mathrm{TV}}(G_i, G_j) \leq 1 - f$. Consider $\mathcal{G}_{\epsilon^{c^k}}$. Consider the partition formed by taking all connected components in this graph. If this partition does not satisfy the desired condition, then there are some two G_i, G_j in different components such that

$$d_{\mathrm{TV}}(G_i, G_j) \leq 1 - \epsilon^{c^{k-1}}$$

Thus, the graph $\mathcal{G}_{\epsilon^{c^{k-1}}}$ has strictly fewer connected components than $\mathcal{G}_{\epsilon^{c^k}}$. We can now repeat this argument on $\mathcal{G}_{\epsilon^{c^{k-1}}}$. However, the number of connected components in $\mathcal{G}_{\epsilon^{c^k}}$ is at most k so we conclude that there must be some $c^k < \kappa < 1$ for which the desired condition is satisfied. □

We will also need the following results about the VC-dimension of hypotheses obtained by comparing the density functions of two mixtures of Gaussians. The reason we need these VC dimension bounds is that we will need to argue that given *any* constant-accuracy estimates, we can obtain a clustering that is poly(ϵ) accurate. While naively this would require union bounding over infinitely many possibilities for the initial estimates, the VC dimension bound allows to get around this and obtain uniform convergence over all possible initial estimates.

Technically for our clustering result, we only need the VC dimension bound for single Gaussians (instead of mixtures of k Gaussians). However, we will need the VC dimension bound for mixtures of Gaussians later when we do hypothesis testing so we state the full result below. First we need a definition.

Definition 7.7. Let $\mathcal{F}$ be a family of distributions on some domain $\mathcal{X}$. Let $\mathcal{H}_{\mathcal{F},a}$ be the set of functions of the form $f_{\mathcal{M}_1,\mathcal{M}_2,\dots,\mathcal{M}_a}$ where $\mathcal{M}_1,\mathcal{M}_2,\dots,\mathcal{M}_a \in \mathcal{F}$ and

$$f_{\mathcal{M}_1,\mathcal{M}_2,\dots,\mathcal{M}_a}(x) = \begin{cases} 1 \text{ if } \mathcal{M}_1(x) \geq \mathcal{M}_2(x),\dots,\mathcal{M}_a(x) \\ 0 \text{ otherwise} \end{cases}$$

where $\mathcal{M}_i(x)$ denotes the pdf of the corresponding distribution at x.

Lemma 7.8 (Theorem 8.14 in [2]). *Let $\mathcal{F}_k$ be the family of distributions that are a mixture of at most k Gaussians in $\mathbb{R}^d$. Then the VC dimension of $\mathcal{H}_{\mathcal{F}_k,a}$ is* poly(d,a,k).

It is a standard result in learning theory that for a hypothesis class with bounded VC dimension, taking a polynomial number of samples suffices to get a good approximation for all hypotheses in the class.

Lemma 7.9 ([54]). *Let $\mathcal{H}$ be a hypothesis class of functions from some domain $\mathcal{X}$ to $\{0,1\}$ with VC dimension V. Let $\mathcal{D}$ be a distribution on $\mathcal{X}$. Let $\epsilon,\delta > 0$ be parameters. Let S be a set of $n =$* poly$(V, 1/\epsilon, \log 1/\delta)$ *i.i.d samples from $\mathcal{D}$. Then with $1-\delta$ probability, for all $f \in \mathcal{H}$*

$$\left|\mathbb{E}_{x\sim S}[f(x)] - \mathbb{E}_{x\sim\mathcal{D}}[f(x)]\right| \leq \epsilon\,.$$

Now we can prove our lemma about obtaining a poly(ϵ)-accurate clustering into submixtures when given constant-accuracy estimates for the components.

Lemma 7.10. *Let $\mathcal{M} = w_1G_1 + \cdots + w_kG_k \in \mathbb{R}^d$ be a mixture of Gaussians where*

- *The w_i are all rational with denominator at most A*
- *$d_{TV}(G_i,G_j) \geq b$*

There exists a sufficiently small function $g(k,A,b) > 0$ depending only on k,A,b such that the following holds. Let $X_1,\dots,X_n$ be an ϵ-corrupted sample from the mixture $\mathcal{M}$ where $\epsilon < g(k,A,b)$ and $n =$ poly(d/ϵ) *for some sufficiently large polynomial. Let $S_1,\dots,S_k \subset \{X_1,\dots,X_n\}$ denote the sets of samples from each of the components $G_1,\dots,G_k$, respectively. Let $R_1,\dots,R_l$ be a partition such that for $i_1 \in R_{j_1}, i_2 \in R_{j_2}$ with $j_1 \neq j_2$,*

$$d_{TV}(G_{i_1},G_{i_2}) \geq 1-\epsilon'$$

where $\epsilon \leq \epsilon' \leq g(k,A,b)$. Let $\widetilde{G_1},\dots,\widetilde{G_k}$ be any Gaussians such that $d_{TV}(G_i,\widetilde{G_i}) \leq g(k,A,b)$ for all i. Let $\widetilde{S_1},\dots,\widetilde{S_k} \subset \{X_1,\dots,X_n\}$ be the subsets of samples obtained by assigning each sample to the component $\widetilde{G_i}$ that gives it the maximum likelihood. Then with probability at least 0.999,

$$\left|\left(\cup_{i\in R_j}S_i\right) \cap \left(\cup_{i\in R_j}\widetilde{S_i}\right)\right| \geq (1-\text{poly}(\epsilon'))\left|\left(\cup_{i\in R_j}S_i\right)\right|$$

for all j.

Proof. First, we will upper bound the expected number of uncorrupted points that are misclassified for each $j \in [l]$ when the Gaussians $\widetilde{G_1}, \dots, \widetilde{G_k}$ are fixed. This quantity can be upper bounded by

$$\sum_{j_1 \neq j_2} \sum_{\substack{i_1 \in R_{j_1} \\ i_2 \in R_{j_2}}} \int 1_{\widetilde{G_{i_1}}(x) > \widetilde{G_{i_2}}(x)} dG_{i_2}(x)$$

Clearly, we can ensure $d_{\mathsf{TV}}(G_i, \widetilde{G_i}) \leq 1/2$. Thus, by Lemma 7.2 and the assumption about $R_1, \dots, R_l$, $d_{\mathsf{TV}}(\widetilde{G_{i_1}}, G_{i_2}) \geq 1 - \text{poly}(\epsilon')$ for all G_{i_2} where i_2 is not in the same piece of the partition as i_1. Let c be such that

$$d_{\mathsf{TV}}(\widetilde{G_{i_1}}, G_{i_2}) \geq 1 - \epsilon'^c$$

By Lemma 7.4,

$$\Pr_{x \in G_{i_2}} \left[\epsilon'^{c/2} \leq \frac{\widetilde{G_{i_2}}(x)}{G_{i_2}(x)} \leq \epsilon'^{c/2} \right] \geq 1 - \text{poly}(\epsilon')$$

and combining the above two inequalities, we deduce

$$\int 1_{\widetilde{G_{i_1}}(x) > \widetilde{G_{i_2}}(x)} dG_{i_2}(x) \leq \text{poly}(\epsilon')$$

Since we are only summing over $O_k(1)$ pairs of components, as long as ϵ' is sufficiently small compared to k, A, b, the expected fraction of misclassified points is $\text{poly}(\epsilon')$.

Next, note that the clustering depends only on the comparisons between the values of the pdfs of the Gaussians $\widetilde{G_1}, \dots, \widetilde{G_k}$ at each of the samples $X_1, \dots, X_n$. Since $n = \text{poly}(d/\epsilon)$ for some sufficiently large polynomial, applying Lemma 7.8 and Lemma 7.9 completes the proof (note that the fraction of corrupted points is at most ϵ so it does not matter how they are clustered). □

Combining Lemma 7.5, Claim 7.6, Lemma 7.10, and Theorem 5.2, we can show that at least one of the sets of candidate parameters that our algorithm outputs is close to the true parameters.

Lemma 7.11. *Let $k, A, b > 0$ be constants. There is a sufficiently large function G and a sufficiently small function g depending only on k, A, b such that given an ϵ-corrupted sample $X_1, \dots, X_n$ from a mixture of Gaussians $\mathcal{M} = w_1 G_1 + \dots + w_k G_k \in \mathbb{R}^d$ where*

- *The w_i are all rational with denominator at most A*
- *$d_{TV}(G_i, G_j) \geq b$*

and $n \geq (d/\epsilon)^{G(k,A,b)}$, with 0.999 probability, among the set of candidates output by Full Algorithm, *there is some $\{\widetilde{w_1}, \widetilde{G_1}, \dots, \widetilde{w_k}, \widetilde{G_k}\}$ such that for all i we have*

$$|w_i - \widetilde{w_i}| + d_{TV}(G_i, \widetilde{G_i}) \leq \text{poly}(\epsilon)$$

Proof. This follows from combining Lemma 7.5, Claim 7.6, Lemma 7.10, and finally applying Theorem 5.2. Note we can choose c in Claim 7.6 so that when combined with Lemma 7.10, the resulting accuracy that we get on each submixture is high enough that we can then apply Theorem 5.2 (we can treat the subsample corresponding to each submixture as a $\text{poly}(\epsilon')$-corrupted sample). We apply Lemma 7.10 with $\epsilon' = \epsilon^{\kappa}$ where the κ is obtained from Claim 7.6. □

We have shown that our algorithm recovers a list of candidate mixtures, at least one of which is close to the true mixture. The last result that we need is a hypothesis testing routine. This is similar to the hypothesis testing result in [39]. However, there is a subtle difference that the samples we use to hypothesis test may not be independent of the hypotheses. This is because the adversary sees all of the data points and may corrupt the data in a way to affect the list of hypotheses that we output. Thus, we must prove that given an ϵ-corrupted sample and *any* list of hypotheses with the promise that at least one of the hypotheses is close to the true distribution, we must output a hypothesis that is close to the true distribution.

Lemma 7.12. *Let $\mathcal{F}$ be a family of distributions on some domain $\mathcal{X}$ with explicitly computable density functions that can be efficiently sampled from. Let V be the VC dimension of $\mathcal{H}_{\mathcal{F},2}$ (recall Definition 7.7). Let $\mathcal{D}$ be an unknown distribution in $\mathcal{F}$. Let m be a parameter. Let $X_1, \dots, X_n$ be an ϵ-corrupted sample from $\mathcal{D}$ with $n \geq \mathrm{poly}(m, \epsilon, V)$ for some sufficiently large polynomial. Let $H_1, \dots, H_m$ be distributions in $\mathcal{F}$ given to us by an adversary with the promise that*

$$\min(d_{TV}(\mathcal{D}, H_i)) \leq \epsilon\,.$$

Then there exists an algorithm that runs in time $\mathrm{poly}(n, \epsilon)$ and outputs an i with $1 \leq i \leq m$ such that with 0.999 probability

$$d_{TV}(\mathcal{D}, H_i) \leq O(\epsilon)\,.$$

Proof. The proof will be very similar to the proof in [39] except we will use the VC dimension bound and Lemma 7.9 to obtain a bound over all possible hypothesis distributions given to us by the adversary.

For each i, j, define $A_{i,j}$ to be the subset of $\mathcal{X}$ where $H_i(x) \geq H_j(x)$ (where we abuse notation and use H_i, H_j to denote their respective probability density functions). Note $d_{\mathrm{TV}}(H_i, H_j) = |H_i(A_{i,j}) - H_j(A_{i,j})|$. By Lemma 7.9, we can ensure that with high probability, for all i, j, the empirical estimates of $A_{i,j}$ are close to their true values, i.e.,

$$|\mathcal{D}(A_{i,j}) - X(A_{i,j})| \leq 2\epsilon\,.$$

Now, since the distributions $H_1, \dots, H_m$ can be efficiently sampled from, we can obtain estimates $\widehat{H_l}(A_{i,j})$ that are within ϵ of $H_l(A_{i,j})$ for all i, j, l. Now, it suffices to return any l such that for all i, j,

$$|X(A_{i,j}) - \widehat{H_l}(A_{i,j})| \leq 4\epsilon.$$

Note that any l such that $d_{\mathrm{TV}}(\mathcal{D}, H_l) \leq \epsilon$ must satisfy the above by the triangle inequality. Next, we argue that any such l must be sufficient. To see this, let l' be such that $d_{\mathrm{TV}}(\mathcal{D}, H_{l'}) \leq \epsilon$. Then

$$\begin{aligned} d_{\mathrm{TV}}(\mathcal{D}, H_l) &\leq \epsilon + d_{\mathrm{TV}}(H_l, H_{l'}) = \epsilon + |H_l(A_{l,l'}) - H_{l'}(A_{l,l'})| \leq 2\epsilon + |H_l(A_{l,l'}) - \mathcal{D}(A_{l,l'})| \\ &\leq 2\epsilon + |X(A_{l,l'}) - \mathcal{D}(A_{l,l'})| + |X(A_{l,l'}) - \widehat{H_l}(A_{l,l'})| + |\widehat{H_l}(A_{l,l'}) - H_l(A_{l,l'})| = O(\epsilon). \end{aligned}$$ □

We can now complete the proof of our main theorem.

Proof of Theorem 7.1. Combining Lemma 7.11, Lemma 7.12, and Lemma 7.8, we immediately get the desired bound. □

8 IDENTIFIABILITY

Theorem 7.1 implies that we can learn a mixture that is close to the true mixture in TV distance. In order to prove that we recover the individual components, it suffices to prove identifiability. In this section we prove the following.

Theorem 8.1. *Let $\mathcal{M} = w_1 G_1 + \cdots + w_{k_1} G_{k_1}$ and $\mathcal{M}' = w_1' G_1' + \cdots + w_{k_2}' G_{k_2}'$ be mixtures of Gaussians such that $TV(\mathcal{M}, \mathcal{M}') \leq \epsilon$ and the G_i, G_i' have variance at least $\mathrm{poly}(\epsilon/d)$ and at most $\mathrm{poly}(d/\epsilon)$ in all directions. Further assume,*

- $d_{TV}(G_i, G_j) \geq b, d_{TV}(G_i', G_j') \geq b$ *for all $i \neq j$*
- $w_i, w_i' \geq w_{\min}$

where $w_{\min} \geq f(k)$, $b \geq \epsilon^{f(k)}$ where $k = \max(k_1, k_2)$ and $f(k) > 0$ is sufficiently small function depending only on k. Then $k_1 = k_2$ and there exists a permutation π such that

$$|w_i - w'_{\pi(i)}| + d_{TV}(G_i, G'_{\pi(i)}) \leq \text{poly}(\epsilon)$$

While technically, we do not need to prove identifiability in an algorithmic manner, our proof will mirror our main algorithm. We will first prove identifiability in the case where the two mixtures are δ-well conditioned for $\delta = \text{poly}(\epsilon)$.

Lemma 8.2. *Let $\mathcal{M} = w_1G_1 + \cdots + w_{k_1}G_{k_1}$ and $\mathcal{M}' = w'_1G'_1 + \cdots + w'_{k_2}G'_{k_2}$ be two δ-well conditioned mixtures of Gaussians such that $d_{TV}(\mathcal{M}, \mathcal{M}') \leq \epsilon$ and $\delta \geq \epsilon^{f(k)}$ where $k = \max(k_1, k_2)$ and $f(k) > 0$ is sufficiently small function depending only on k. Then $k_1 = k_2$ and there exists a permutation π such that*

$$|w_i - w'_{\pi(i)}| + d_{TV}(G_i, G'_{\pi(i)}) \leq \text{poly}(\epsilon).$$

Proof. Let $\mu, \Sigma, \mu', \Sigma'$ be the means and covariances of the mixtures $\mathcal{M}$ and $\mathcal{M}'$. Let $\mu_i, \Sigma_i, \mu'_i, \Sigma'_i$ be the means and covariances of the respective components. Without loss of generality we may assume $\mu = 0$, $\Sigma = I$. The results in Section 5, namely Corollary 5.7, imply that

$$\|I - \Sigma'\| = \text{poly}(\epsilon)$$
$$\|\mu'\| = \text{poly}(\epsilon)$$

This is because we can simulate an ϵ-corrupted sample from $\mathcal{M}'$ by just sampling from $\mathcal{M}$ (since $d_{\text{TV}}(\mathcal{M}, \mathcal{M}') \leq \epsilon$) and then robustly estimate the mean and covariance of this sample. Thus, by Corollary 5.5, we have for all i,

$$\|\mu_i\|, \|\mu'_i\| \leq \text{poly}(\delta)^{-1}$$
$$\|\Sigma_i - I\|, \|\Sigma'_i - I\| \leq \text{poly}(\delta)^{-1}$$

Now, we can use Lemma 5.8 to estimate the Hermite polynomials of the mixtures $\mathcal{M}, \mathcal{M}'$. Since we can robustly estimate the means of bounded-covariance distributions (see Theorem 2.2 in [39], Lemma 5.8), we must have

$$\|v\,(h_{m,\mathcal{M}}(X) - h_{m,\mathcal{M}'}(X))\|_2 \leq \text{poly}(\epsilon)$$

Also note that since each of the mixtures is δ-well conditioned, using Claim 5.3 and Lemma 5.4 implies that

$$||\mu_i - \mu_j||_2 + ||\Sigma_i - \Sigma_j||_2 \geq \text{poly}(\delta)$$

and is similar for the components of the mixture $\mathcal{M}'$. Repeating the argument in Section 4.1, it suffices to prove the lemma in the case when all pairs of parameters are separated or equal i.e., among the sets $\{\mu_i\} \cup \{\mu'_i\}$ and $\{\Sigma_i\} \cup \{\Sigma'_i\}$, each pair of parameters is either equal or separated by at least poly(δ). If we prove this. we can then deduce the statement of the lemma in the general case with worse, but still polynomial dependencies on ϵ.

Now we consider the generating functions

$$F = \sum_{i=1}^{k_1} w_i e^{\mu_i(X)y + \frac{1}{2}\Sigma_i(X)y^2} = \sum_{m=0}^{\infty} \frac{1}{m!} h_{m,\mathcal{M}}(X) y^n$$
$$F' = \sum_{i=1}^{k_2} w'_i e^{\mu'_i(X)y + \frac{1}{2}\Sigma'_i(X)y^2} = \sum_{m=0}^{\infty} \frac{1}{m!} h_{m,\mathcal{M}'}(X) y^n$$

where similar to in Section 4, $\mu_i(X) = \mu_i \cdot X, \Sigma_i(X) = X^T \Sigma_i X$. Consider the pair $(\mu'_{k_2}, \Sigma'_{k_2})$. We claim that there must be some i such that

$$(\mu_i, \Sigma_i) = \left(\mu'_{k_2}, \Sigma'_{k_2}\right)$$

Assume for the sake of contradiction that this is not the case. Let S_1 be the subset of $[k_1]$ such that $\Sigma_i = \Sigma_{k'_2}$ and let S_2 be the subset of $[k_2 - 1]$ such that $\Sigma'_j = \Sigma'_{k_2}$. Define the differential operators

$$\mathcal{D}_i = \partial - \mu_i(X) - \Sigma_i(X)y$$
$$\mathcal{D}'_i = \partial - \mu'_i(X) - \Sigma'_i(X)y$$

where as before, partial derivatives are taken with respect to y. Now consider the differential operator

$$\mathcal{D} = \left(\mathcal{D}'_{k_2-1}\right)^{2^{k_1+k_2-2}} \left(\mathcal{D}'_1\right)^{2^{k_1}} \mathcal{D}_{k_1}^{2^{k_1-1}} \mathcal{D}_1^1$$

Note by Claim 3.9, $\mathcal{D}(F) = 0$. Using Claim 3.11,

$$\mathcal{D}(F') = P(y, X) e^{\mu'_{k_2}(X)y + \frac{1}{2}\Sigma'_{k_2}(X)y^2}$$

where P is a polynomial of degree

$$\deg(P) = 2^{k_1+k_2-1} - 1 - \sum_{i \in S_1} 2^{i-1} - \sum_{i \in S_2} 2^{k_1+i-2}$$

and has leading coefficient

$$C_0 = w'_{k_2} \prod_{i \in [k_1] \backslash S_1} (\Sigma'_{k_2} - \Sigma_i)^{2^{i-1}} \prod_{i \in S_1} (\mu'_{k_2} - \mu_i)^{2^{i-1}} \prod_{i \in [k_2-1] \backslash S_2} (\Sigma'_{k_2} - \Sigma'_i)^{2^{k_1+i-2}} \prod_{i \in S_2} (\mu'_{k_2} - \mu'_i)^{2^{k_1+i-2}}.$$

If there is no i such that $(\mu_i, \Sigma_i) = (\mu'_{k_2}, \Sigma'_{k_2})$ then

$$C_0 \geq \delta^{O_k(1)}$$

We can now compare

$$\left(\mathcal{D}'_{k_2}\right)^{\deg(P)} \mathcal{D}(F)$$
$$\left(\mathcal{D}'_{k_2}\right)^{\deg(P)} \mathcal{D}(F')$$

evaluated at $y = 0$. The first quantity is 0 because $\mathcal{D}(F)$ is identically 0 as a formal power series. The second expression is equal to $\Omega_k(1) C_0$. However, the coefficients of the formal power series F, F' are the Hermite polynomials $h_{m,\mathcal{M}}(X)$ and $h_{m,\mathcal{M}'}(X)$. We assumed that

$$||v(h_{m,\mathcal{M}}(X) - h_{m,\mathcal{M}'}(X))||_2 \leq \text{poly}(\epsilon)$$

so this is a contradiction as long as ϵ is smaller than $\delta^{F(k)}$ for some sufficiently large function F depending only on k. Thus, there must be some component of the mixture $\mathcal{M}$ that matches each component of $\mathcal{M}'$. We can then repeat the argument in reverse to conclude that $\mathcal{M}$ and $\mathcal{M}'$ have the same components. Finally, assume that we have two mixtures $\mathcal{M} = w_1 G_1 + \cdots + w_k G_k$ and $\mathcal{M}' = w'_1 G_1 + \cdots + w'_k G_k$ on the same set of components. WLOG

$$w_1 - w'_1 < \cdots < w_l - w'_l < 0 < w_{l+1} - w'_{l+1} < \cdots < w_k - w'_k$$

Then we can consider

$$(w'_1 - w_1)G_1 + \cdots + (w'_l - w_l)G_l \text{ and}$$
$$(w_{l+1} - w'_{l+1})G_{l+1} + \cdots + (w_k - w'_k)G_k$$

each treated as a mixture. If

$$\sum_{i=1}^{k} |w_i - w_i'| > \epsilon^{\zeta}$$

for some sufficiently small ζ depending only on k, we can then normalize each of the above into a mixture (i.e., make the mixing weights sum to 1) and repeat the same argument, using the fact that pairs of components cannot be too close, to obtain a contradiction. Thus, actually the components and mixing weights of the two mixtures must be poly(ϵ)-close and this completes the proof. □

To complete the proof of Theorem 8.1, we will prove a sort of cluster identifiability that mirrors our algorithm and then combine with Lemma 8.2.

Proof of Theorem 8.1. Let c be a sufficiently small constant that we will set later. We apply Claim 7.6 on the mixture $\mathcal{M}$ with parameter c to find a partition $R_1, \dots, R_l$. Let κ be the parameter obtained from the statement of Claim 7.6 i.e., κ depends on k and c. First, we claim that each of the components $G_1', \dots, G_{k_2}'$ must be essentially contained within one of the clusters. To see this, for each $j \in [k_2]$ there must be some i such that

$$d_{\mathrm{TV}}(G_i, G_j') \le 1 - \frac{w_{\min}}{2k} \le 1 - \Omega_k(1)$$

without loss of generality $i \in R_1$. Then by Lemma 7.2, for all $a \notin R_1$,

$$d_{\mathrm{TV}}(G_a, G_j') \ge 1 - \mathrm{poly}(\epsilon^{\kappa})$$

where the polynomial may depend on k *but does not depend on* c. The above implies that we can match each of the components $G_1', \dots, G_{k_2}'$ uniquely to one of the clusters $R_1, \dots, R_l$ where it has constant overlap with $\cup_{i \in R_j} G_i$. Let S_1 be the subset of $[k_2]$ corresponding to the components among $G_1', \dots, G_{k_2}'$ that are matched to R_1. Consider the mixtures

$$\mathcal{M}_1 = \frac{\sum_{i \in R_1} w_i G_i}{\sum_{i \in R_1} w_i}$$
$$\mathcal{M}_1' = \frac{\sum_{i \in S_1} w_i' G_i'}{\sum_{i \in S_1} w_i'}$$

The above (combined with our assumed lower bound on the minimum mixing weight) implies that

$$d_{\mathrm{TV}}(\mathcal{M}_1, \mathcal{M}_1') \le \mathrm{poly}(\epsilon^{\kappa})$$

where again the polynomial may depend on k but not c. Now if we choose c sufficiently small, we can apply Lemma 8.2 to deduce that the components and mixing weights of $\mathcal{M}_1, \mathcal{M}_1'$ must be close. We can then repeat this argument for all of the clusters $R_1, \dots, R_l$ to complete the proof. □

Combing Theorem 7.1 and Theorem 8.1. we have.

Theorem 8.3. *Let $k, A, b > 0$ be constants. There is a sufficiently large function G and a sufficiently small function g depending only on k, A, b (with $G(k, A, b), g(k, A, b) > 0$) such that given an ϵ-corrupted sample $X_1, \dots, X_n$ from a mixture of Gaussians $\mathcal{M} = w_1 G_1 + \dots + w_k G_k \in \mathbb{R}^d$ where the G_i have variance at least* $\mathrm{poly}(\epsilon/d)$ *and at most* $\mathrm{poly}(d/\epsilon)$ *in all directions and*

- *The w_i are all rational with denominator at most A*
- $d_{TV}(G_i, G_j) \ge b$

and $n \geq (d/\epsilon)^{G(k,A,b)}$, *then there is an algorithm that runs in time* $\text{poly}(n)$ *and with* 0.99 *probability outputs a set of components* $\widetilde{G_1}, \ldots, \widetilde{G_k}$ *and mixing weights* $\widetilde{w_1}, \ldots, \widetilde{w_k}$ *such that there exists a permutation* π *on* $[k]$ *with*

$$|w_i - \widetilde{w}_{\pi(i)}| + d_{TV}(G_i, \widetilde{G}_{\pi(i)}) \leq \epsilon^{g(k,A,b)}$$

for all i.

8.1 Improving the Separation Assumption

With simple modifications to the analysis, we obtain the following improvement of Theorem 7.1 in [45].

Theorem 8.4 ([45]). *Let* $k, A > 0$ *be constants. There is a sufficiently large function* G *and a sufficiently small function* g *depending only on* k, A *such that given an* ϵ*-corrupted sample* $X_1, \ldots, X_n$ *from a mixture of Gaussians* $\mathcal{M} = w_1 G_1 + \cdots + w_k G_k \in \mathbb{R}^d$ *where* $\epsilon < g(k,A)$, *the* w_i *are all rational with denominator at most* A, *and* $n \geq (d/\epsilon)^{G(k,A)}$, *there is an algorithm that runs in time* $\text{poly}(n)$ *and with* 0.999 *probability, outputs a set of* $(1/\epsilon)^{O_{k,A}(1)}$ *candidate mixtures such that for at least one of these candidates,* $\{\widetilde{w_1}, \widetilde{G_1}, \ldots, \widetilde{w_k}, \widetilde{G_k}\}$, *we have*

$$|w_i - \widetilde{w_i}| + d_{TV}(G_i, \widetilde{G_i}) \leq \epsilon^{g(k,A)}$$

for all $i \in [k]$.

To go from Theorem 7.1 to 8.4, the main idea to remove the constant separation assumption is just that we can find a scale δ such that all pairs of components either have $d_{\text{TV}}(G_i, G_j) \leq \delta$ or $d_{\text{TV}}(G_i, G_j) \geq \delta'$ for $\delta' \gg \delta$. This is possible because the number of components k is a constant. We can then merge components whose TV distance is less than δ, treating them as the same component and the remaining components will be sufficiently separated. See [45] for more details.

The results of [45] were obtained using the previous clustering subroutine of [21]. If we instead plug in the updated clustering results of [4] (see Theorem 6.12 vs Theorem 6.2), we can remove the bounded fractionality assumption on the mixing weights. Also, we can ensure that the algorithm outputs a unique mixture instead of a list by running the hypothesis testing routine in Lemma 7.12.

Theorem 8.5. *Let* $k, A > 0$ *be constants. There is a sufficiently large function* G *and a sufficiently small function* g *depending only on* k, A *such that given an* ϵ*-corrupted sample* $X_1, \ldots, X_n$ *from a mixture of Gaussians* $\mathcal{M} = w_1 G_1 + \cdots + w_k G_k \in \mathbb{R}^d$ *where* $\epsilon < g(k,A)$, *the* w_i *are all at least* $1/A$, *and* $n \geq (d/\epsilon)^{G(k,A)}$, *there is an algorithm that runs in time* $\text{poly}(n)$ *and with* 0.999 *probability, outputs a mixture* $\widetilde{M} = \widetilde{w_1}\widetilde{G_1} + \cdots + \widetilde{w_k}, \widetilde{G_k}$, *such that*

$$d_{TV}(\mathcal{M}, \widetilde{\mathcal{M}}) \leq \epsilon^{g(k,A)}\,.$$

Finally, combining the above with identifiability (Theorem 8.1), we immediately get an improved version of our main theorem for parameter learning.

Theorem 8.6. *Let* $k, A > 0$ *be constants. There is a sufficiently large function* G *and a sufficiently small function* g *depending only on* k, A *(with* $G(k,A), g(k,A) > 0$*) such that given an* ϵ*-corrupted sample* $X_1, \ldots, X_n$ *from a mixture of Gaussians* $\mathcal{M} = w_1 G_1 + \cdots + w_k G_k \in \mathbb{R}^d$ *where the* G_i *have variance at least* $\text{poly}(\epsilon/d)$ *and at most* $\text{poly}(d/\epsilon)$ *in all directions and*

- *The* w_i *are all at least* $1/A$
- $d_{TV}(G_i, G_j) \geq \epsilon^{g(k,A)}$

and $n \geq (d/\epsilon)^{G(k,A)}$, *then there is an algorithm that runs in time* $\text{poly}(n)$ *and with* 0.99 *probability outputs a set of components* $\widetilde{G_1}, \ldots, \widetilde{G_k}$ *and mixing weights* $\widetilde{w_1}, \ldots, \widetilde{w_k}$ *such that there exists a* *permutation π on $[k]$ with*

$$|w_i - \widetilde{w}_{\pi(i)}| + d_{TV}(G_i, \widetilde{G}_{\pi(i)}) \leq \epsilon^{g(k,A)}$$

for all i.

REFERENCES

[1] Dimitris Achlioptas and Frank McSherry. 2005. On spectral learning of mixtures of distributions. In *Learning Theory*. Springer, 458–469.

[2] Martin Anthony and Peter L. Bartlett. 2009. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press.

[3] Sanjeev Arora and Ravi Kannan. 2001. Learning mixtures of arbitrary Gaussians. In *Proceedings of the Thirty-third Annual ACM Symposium on Theory of Computing*. 247–257.

[4] Ainesh Bakshi, Ilias Diakonikolas, He Jia, Daniel M. Kane, Pravesh K. Kothari, and Santosh S. Vempala. 2020. Robustly learning mixtures of k arbitrary Gaussians. *arXiv preprint arXiv:2012.02119* (2020). This version may be found at https://arxiv.org/abs/2012.02119v2.

[5] Ainesh Bakshi, Ilias Diakonikolas, He Jia, Daniel M. Kane, Pravesh K. Kothari, and Santosh S. Vempala. 2020. Robustly learning mixtures of k arbitrary Gaussians. *arXiv preprint arXiv:2012.02119* (2020). This version may be found at https://arxiv.org/abs/2012.02119v3.

[6] Ainesh Bakshi and Pravesh Kothari. 2020. Outlier-robust clustering of non-spherical mixtures. *arXiv preprint arXiv:2005.02970* (2020).

[7] Ainesh Bakshi and Adarsh Prasad. 2020. Robust linear regression: Optimal rates in polynomial time. *arXiv preprint arXiv:2007.01394* (2020).

[8] Sivaraman Balakrishnan, Simon S. Du, Jerry Li, and Aarti Singh. 2017. Computationally efficient robust sparse estimation in high dimensions. In *Conference on Learning Theory*. 169–212.

[9] Boaz Barak. [n.d.]. Proofs, beliefs, and algorithms through the lens of sum-of-squares. ([n. d.]).

[10] Boaz Barak, Jonathan A. Kelner, and David Steurer. 2014. Rounding sum-of-squares relaxations. In *Proceedings of the Forty-sixth Annual ACM Symposium on Theory of Computing*. 31–40.

[11] Boaz Barak, Jonathan A. Kelner, and David Steurer. 2015. Dictionary learning and tensor decomposition via the sum-of-squares method. In *Proceedings of the Forty-seventh Annual ACM Symposium on Theory of Computing*. 143–151.

[12] Boaz Barak and Ankur Moitra. 2016. Noisy tensor completion via the sum-of-squares hierarchy. In *Conference on Learning Theory*. 417–445.

[13] Mikhail Belkin and Kaushik Sinha. 2010. Polynomial learning of distribution families. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*. IEEE, 103–112.

[14] Thorsten Bernholt. 2006. *Robust Estimators are Hard to Compute*. Technical Report.

[15] Aditya Bhaskara, Moses Charikar, Ankur Moitra, and Aravindan Vijayaraghavan. 2014. Smoothed analysis of tensor decompositions. In *Proceedings of the Forty-sixth Annual ACM Symposium on Theory of Computing*. 594–603.

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

[17] Moses Charikar, Jacob Steinhardt, and Gregory Valiant. 2017. Learning from untrusted data. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*. ACM, 47–60.

[18] Sitan Chen, Frederic Koehler, Ankur Moitra, and Morris Yau. 2020. Online and distribution-free robustness: Regression and contextual bandits with huber contamination. *arXiv preprint arXiv:2010.04157* (2020).

[19] Sanjoy Dasgupta. 1999. Learning mixtures of Gaussians. In *Foundations of Computer Science, 1999. 40th Annual Symposium on*. IEEE, 634–644.

[20] Sanjoy Dasgupta and Leonard Schulman. 2013. A two-round variant of EM for Gaussian mixtures. *arXiv preprint arXiv:1301.3850* (2013).

[21] Ilias Diakonikolas, Samuel B. Hopkins, Daniel Kane, and Sushrut Karmalkar. 2020. Robustly learning any clusterable mixture of Gaussians. *arXiv preprint arXiv:2005.06417* (2020).

[22] Ilias Diakonikolas, Gautam Kamath, Daniel Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. 2019. Robust estimators in high-dimensions without the computational intractability. *SIAM J. Comput.* 48, 2 (2019), 742–864.

[23] Ilias Diakonikolas, Gautam Kamath, Daniel Kane, Jerry Li, Jacob Steinhardt, and Alistair Stewart. 2019. Sever: A robust meta-algorithm for stochastic optimization. In *International Conference on Machine Learning*. 1596–1606.

[24] Ilias Diakonikolas, Gautam Kamath, Daniel M. Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. 2017. Being robust (in high dimensions) can be practical. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR, 999–1008.

[25] Ilias Diakonikolas and Daniel M. Kane. 2019. Recent advances in algorithmic high-dimensional robust statistics. *arXiv preprint arXiv:1911.05911* (2019).

[26] Rong Ge, Qingqing Huang, and Sham M. Kakade. 2015. Learning mixtures of Gaussians in high dimensions. In *Proceedings of the Forty-seventh Annual ACM Symposium on Theory of Computing*. 761–770.

[27] Frank R. Hampel, Elvezio M. Ronchetti, Peter J. Rousseeuw, and Werner A. Stahel. 2011. *Robust Statistics: The Approach based on Influence Functions*. Vol. 196. John Wiley & Sons.

[28] Moritz Hardt and Ankur Moitra. 2013. Algorithms and hardness for robust subspace recovery. In *Conference on Learning Theory*. 354–375.

[29] Moritz Hardt and Eric Price. 2014. Tight Bounds for Learning a Mixture of Two Gaussians. https://doi.org/10.48550/ARXIV.1404.4997

[30] Samuel B. Hopkins, Pravesh K. Kothari, Aaron Potechin, Prasad Raghavendra, Tselil Schramm, and David Steurer. 2017. The power of sum-of-squares for detecting hidden structures. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 720–731.

[31] Samuel B. Hopkins and Jerry Li. 2018. Mixture models, robustness, and sum of squares proofs. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*. ACM, 1021–1034.

[32] Samuel B. Hopkins, Tselil Schramm, Jonathan Shi, and David Steurer. 2016. Fast spectral algorithms from sum-of-squares proofs: Tensor decomposition and planted sparse vectors. In *Proceedings of the Forty-eighth Annual ACM Symposium on Theory of Computing*. 178–191.

[33] Samuel B. Hopkins, Jonathan Shi, and David Steurer. 2015. Tensor principal component analysis via sum-of-square proofs. In *Conference on Learning Theory*. 956–1006.

[34] Daniel Hsu and Sham M. Kakade. 2013. Learning mixtures of spherical Gaussians: Moment methods and spectral decompositions. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science*. ACM, 11–20.

[35] Peter J. Huber. 1964. Robust estimation of a location parameter. *The Annals of Mathematical Statistics* (1964), 73–101.

[36] Peter J. Huber. 2004. *Robust Statistics*. Vol. 523. John Wiley & Sons.

[37] David S. Johnson and Franco P. Preparata. 1978. The densest hemisphere problem. *Theoretical Computer Science* 6, 1 (1978), 93–107.

[38] Adam Tauman Kalai, Ankur Moitra, and Gregory Valiant. 2010. Efficiently learning mixtures of two Gaussians. In *Proceedings of the 42nd ACM Symposium on Theory of Computing*. ACM, 553–562.

[39] Daniel M. Kane. 2020. Robust learning of mixtures of Gaussians. *arXiv preprint arXiv:2007.05912* (2020).

[40] Adam Klivans, Pravesh K. Kothari, and Raghu Meka. 2018. Efficient algorithms for outlier-robust regression. In *Conference on Learning Theory*. 1420–1430.

[41] Pravesh K. Kothari, Jacob Steinhardt, and David Steurer. 2018. Robust moment estimation and improved clustering via sum of squares. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*. ACM, 1035–1046.

[42] Amit Kumar and Ravindran Kannan. 2010. Clustering with spectral norm and the k-means algorithm. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*. IEEE, 299–308.

[43] Kevin A. Lai, Anup B. Rao, and Santosh Vempala. 2016. Agnostic estimation of mean and covariance. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS'16)*. IEEE, 665–674.

[44] Jerry Zheng Li. 2018. *Principled Approaches to Robust Machine Learning and Beyond*. Ph.D. Dissertation. Massachusetts Institute of Technology.

[45] Allen Liu and Ankur Moitra. 2021. Learning GMMs with nearly optimal robustness guarantees. *arXiv preprint arXiv:2104.09665* (2021).

[46] Ankur Moitra. 2018. *Algorithmic Aspects of Machine Learning*. Cambridge University Press.

[47] Ankur Moitra and Gregory Valiant. 2010. Settling the polynomial learnability of mixtures of Gaussians. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*. IEEE, 93–102.

[48] Pablo A. Parrilo. 2000. *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*. Ph.D. Dissertation. California Institute of Technology.

[49] Karl Pearson. 1894. Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London. A* 185 (1894), 71–110.

[50] Jacob Steinhardt. 2018. *Robust Learning: Information Theory and Algorithms*. Ph.D. Dissertation. Stanford University.

[51] Henry Teicher. 1961. Identifiability of mixtures. *The Annals of Mathematical Statistics* 32, 1 (1961), 244–248.

[52] John W. Tukey. 1960. A survey of sampling from contaminated distributions. *Contributions to Probability and Statistics* (1960), 448–485.

[53] John W. Tukey. 1975. Mathematics and the picturing of data. In *Proceedings of the International Congress of Mathematicians, Vancouver, 1975*, Vol. 2. 523–531.

[54] Vladimir N. Vapnik and A. Ya Chervonenkis. 2015. On the uniform convergence of relative frequencies of events to their probabilities. In *Measures of Complexity*. Springer, 11–30.

[55] Santosh Vempala and Grant Wang. 2004. A spectral algorithm for learning mixture models. *J. Comput. System Sci.* 68, 4 (2004), 841–860.

Received 25 July 2021; revised 4 January 2023; accepted 19 January 2023