Settling the Polynomial Learnability of Mixtures of Gaussians

Ankur Moitra[*]
Electrical Engineering and Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139
moitra@mit.edu

Gregory Valiant[†]
Electrical Engineering and Computer Science
University of California, Berkeley
Berkeley, CA 94720
gvaliant@cs.berkeley.edu

***Abstract*—Given data drawn from a mixture of multivariate Gaussians, a basic problem is to accurately estimate the mixture parameters. We give an algorithm for this problem that has running time and data requirements polynomial in the dimension and the inverse of the desired accuracy, with provably minimal assumptions on the Gaussians. As a simple consequence of our learning algorithm, we we give the first polynomial time algorithm for proper density estimation for mixtures of k Gaussians that needs *no* assumptions on the mixture. It was open whether proper density estimation was even statistically possible (with no assumptions) given only polynomially many samples, let alone whether it could be computationally efficient.**

The building blocks of our algorithm are based on the work (Kalai *et al*, STOC 2010) [17] that gives an efficient algorithm for learning mixtures of two Gaussians by considering a series of projections down to one dimension, and applying the *method of moments* to each univariate projection. A major technical hurdle in [17] is showing that one can efficiently learn *univariate* mixtures of two Gaussians. In contrast, because pathological scenarios can arise when considering projections of mixtures of more than two Gaussians, the bulk of the work in this paper concerns how to leverage a weaker algorithm for learning univariate mixtures (of many Gaussians) to learn in high dimensions. Our algorithm employs *hierarchical clustering* and rescaling, together with methods for backtracking and recovering from the failures that can occur in our univariate algorithm.

Finally, while the running time and data requirements of our algorithm depend exponentially on the number of Gaussians in the mixture, we prove that such a dependence is necessary.

***Keywords*-learning; method of moments; mixture models;**

I. Introduction

Given access to random samples generated from a mixture of (multivariate) Gaussians, the algorithmic problem of learning the parameters of the underlying distribution is of fundamental importance in physics, biology, geology, social sciences – any area in which such finite mixture models arise (see [23], [30]). Starting with Dasgupta [9], a series of work in theoretical computer science has sought to find (or disprove the existence of) an efficient algorithm for this task [2], [11], [32], [1], [5], [3]. In this paper, we settle this problem: We give an algorithm for the problem of accurately estimating the parameters of the mixture which has running time and data requirements polynomial in the dimension and the inverse of the desired accuracy, with provably minimal assumptions on the Gaussians (specifically, that the mixing weights and the statistical distance between each pair of components are each bounded away from zero). We give a more precise definition for the learning problem in Section I-B. In fact, our estimate (on a component by component basis) converges at in inverse polynomial rate in a *statistical* sense to the true components of the mixture, and such statistical guarantees are much stronger than additive guarantees on the accuracy of the recovered parameters. These statistical guarantees are invariant under affine transformations and hence give guarantees even when an affine transformation is applied to the data before it is given to us.

As a simple consequence of our learning algorithm, we give the first polynomial time algorithm for proper density estimation for mixtures of k Gaussians without *any* assumptions on the mixture. It was open whether such an algorithm was even information theoretically possible.

In the remainder of this section, we briefly summarize previous work on this problem, formally state our main result, and then discuss the differences between learning mixtures of two Gaussians, and mixtures of many Gaussians. Additionally, we give a high-level outline of the main structure of our algorithm. We first define a Gaussian Mixture Model (GMM).

Consider a set of k *different* multinormal distributions, where each component is characterized by a mean $\mu_i \in \mathbb{R}^n$, and covariance matrix $\Sigma_i \in \mathbb{R}^{n \times n}$. Given a vector of k nonnegative weights $\vec{w}$, summing to one, we define the associated Gaussian Mixture Model (GMM) to be the distribution that results from choosing a component i according to the distribution $\vec{w}$, and then taking a sample from $\mathcal{N}(\mu_i, \Sigma_i)$.

A. A Brief History

The most popular solution for recovering reasonable estimates of the components of GMMs in practice is the EM algorithm given by Dempster, Laird and Rubin [12]. This algorithm is a local-search heuristic that converges to a

[*] This research was supported in part by a Fannie and John Hertz Foundation Fellowship.

[†] This research was supported in part by an NSF Graduate Research Fellowship. Part of this work was visiting Microsoft Research New England.

set of parameters that locally maximizes the probability of generated the observed samples. However, the EM algorithm is a heuristic, and makes no guarantees about converging to an estimate that is close to the true parameters. Worse still, the EM algorithm (even for univariate mixtures of just two Gaussians) has been observed to converge very slowly (see Redner and Walker for a thorough treatment [26]).

In order to even hope for an algorithm (not necessarily even polynomial time), we would need a uniqueness property – that two distinct mixtures of Gaussians must have different probability density functions. Teicher [29] demonstrated that a mixture of Gaussians can be uniquely identified (up to a relabeling of the components) by considering the probability density function at points sufficiently far from the centers (in the tails). However, such a result sheds little light on the *rate* of convergence of an estimator: If distinguishing Gaussian mixtures really required analyzing the tails of the distribution, then we would require an enormous number of data samples!

Dasgupta [9] introduced theoretical computer science to the algorithmic problem of *provably* recovering good estimates for the parameters in polynomial time (and polynomial sample complexity). His technique is based on projecting data down to a randomly chosen low-dimensional subspace and finding an accurate clustering from the low-dimensional data. Given enough sample points that are all accurately clustered, the empirical means and co-variances of each cluster will be a good estimate for the actual parameters of the corresponding component. Arora and Kannan [2] extended these ideas to work in the much more general setting in which the co-variances of each Gaussian component could be arbitrary, and not necessarily almost spherical as in [9]. Yet both of these techniques are based on a concentration of measure phenomenon which critically needs the assumption that the centers of the components before a random projection are separated by at least $\Omega(n^{1/4})$ times the largest variance. Vempala and Wong [32] and Achlioptas and McSherry [1] introduced the use of spectral techniques, and were able to overcome this barrier by choosing a subspace on which to project based on large principle components rather than choosing a subspace randomly. Brubaker and Vempala [5] later gave the first affine-invariant algorithm for learning mixtures of Gaussians, again based on clustering.

The above approaches for provably learning good estimates each require at the very least that the statistical overlap (i.e. one minus the statistical distance) between each pair of components is sub-constant. Yet the range in which our learning algorithms will work only requires that the statistical overlap be bounded away from one. Recently, Felman *et al* [13] gave a polynomial time algorithm for the related problem of density estimation (without any separation condition) for the special case of axis-aligned GMMs (GMMs where each component has principle coordinates aligned with the coordinate axes).

Belkin and Sinha [3] showed that one can efficiently learn each component in the special case that all components are identical spherical Gaussians. Belkin and Sinha [4] also recently gave an algorithm for learning mixtures of k Gaussians that only requires the desired precision to be polynomially smaller than the smallest additive gap in the parameters. Most similar to the present work is the recent work of Kalai *et al* [17], which gives a learning algorithm for the case of mixtures of two Gaussians with no separation assumptions.

B. Main Results

In this section we state our main results. We first consider what it means to "accurately recover the mixture components." We denote $D(F, F')$ as the statistical distance between F and F'. We provide a formal definition of statistical distance in Section II-A.

Definition 1. *Given two n-dimensional GMMs of k Gaussians, $F = \sum_i w_i \mathcal{N}(\mu_i, \Sigma_i)$ and $\hat{F} = \sum_i \hat{w}_i \mathcal{N}(\hat{\mu}_i, \hat{\Sigma}_i)$, we call $\hat{F}$ an ϵ-close estimate for F if there is permutation function $\pi : [k] \rightarrow [k]$ such that for all $i \in [k]$*

- $|w_i - \hat{w}_{\pi(i)}| \leq \epsilon$
- $D(\mathcal{N}(\mu_i, \Sigma_i), \mathcal{N}(\hat{\mu}_{\pi(i)}, \hat{\Sigma}_{\pi(i)})) \leq \epsilon,$

Note that the above definition of an ϵ-close estimate is affine invariant. This is more natural than defining a good estimate in terms of additive errors in the parameters, since in general, even estimating the mean of an arbitrary Gaussian to some fixed additive precision is impossible without restrictions on the covariance, as scaling the data will scale the error as well.

Before discussing our results, we first state three obvious lower bounds for learning an ϵ-close estimate of a GMM $F = \sum_{i=1}^{k} w_i F_i$. These examples will motivate our defintion of *ϵ-statistically learnable*.

1) Permuting the order of the components does not change the resulting density, so we can only hope to recover the parameter *set*, $\{(w_1, \mu_1, \Sigma_1), \ldots, (w_k, \mu_k, \Sigma_k)\}$.
2) We require at least $\Omega(1/\min_i(w_i))$ samples to estimate the parameters, since we need at least this many samples to ensure that we have seen, with reasonable probability, even just one sample from each component.
3) If $F_i = F_j$, then it is impossible to accurately estimate w_i, and in general we require at least $\Omega(1/D(F_i, F_j))$ samples to estimate w_i, where $D(F_i, F_j)$ denotes the statistical distance between the two distributions.

Definition 2. *We call a GMM $F = \sum_i w_i F_i$ ϵ-statistically learnable if $\min_i w_i \geq \epsilon$ and $\min_{i \neq j} D(F_i, F_j) \geq \epsilon$.*

Given just this provably necessary condition that F be ϵ-statistically learnable, we will be able to efficiently learn an ϵ-close estimate to F. We can now state our main theorem:

Theorem 1. *Given any n dimensional mixture of k Gaussians F that is ϵ-statistically learnable, there is an algorithm that, with probability at least $1 - \delta$, outputs an ϵ-close estimate $\hat{F}$ and the running time and data requirements of our algorithm (for any fixed k) are polynomial in n, $\frac{1}{\epsilon}$, and $\frac{1}{\delta}$.*

Throughout this paper, we favor clarity of proof and exposition above optimization of runtime. Since our main goal is show that these problems can be solved in polynomial time, we make very little effort to optimize the exponent. Our algorithms are polynomial in the dimension, the inverse of the success probability, and the inverse of the target accuracy for any fixed number of k Gaussians. However, the dependence on k is severe: the *degree* of our polynomials is $\Theta(k^3)$. Note that the exponent in the result due to Belkin and Sinha [4] is only guaranteed to be bounded for bounded k. In Section VI, we give a natural construction of two GMMs F, F' of k components that are each $1/k$-statistically learnable, for which F is not even a $1/4$-close estimate of F' and yet $D(F, F') \leq e^{-k}$. We would require an exponential in k number of samples to even distinguish the two mixtures in this example from each other. This demonstrates that exponential dependence on k is inevitable when learning mixtures of k Gaussians. We give a formal statement in Theorem 14.

C. Applications

We can leverage our main theorem to show that we can efficiently perform proper density estimation for *arbitrary* GMMs. By *proper* density estimation, we will require that our algorithm returns an estimate distribution that *is* a mixture of at most k Gaussians, in addition to the guarantee that our estimate distribution be statistically close as a mixture to to the true mixture. For density estimation—as opposed to parameter recovery—we do not need the components in our estimate to be close to the actual components. For example, if one component F_i in the mixture has negligible mixing weight, then our estimate can be statistically close as a distribution to the true distribution even if no component in our estimate is statistically close to F_i. Similarly, if two components F_i and F_j are only negligibly statistically different from each other, in our estimate we can merge these two components into a single component and our estimate can still be statistically close to the true mixture.

For these reasons, we can perform density estimation efficiently without the restriction that F be ϵ-statistically learnable, which was required as an assumption for the learning algorithm in Theorem 1.

Corollary 2. *For any $n \geq 1$, $\epsilon, \delta > 0$, and any n-dimensional GMM $F = \sum_{i=1}^{k} w_i F_i$, given access to independent samples from F, there is an algorithm that outputs $\hat{F} = \sum_{i=1}^{k} \hat{w}_i \hat{F}_i$ such that with probability at least $1 - \delta$ over the randomization in the algorithm and in selecting the samples, $D(F, \hat{F}) \leq \epsilon$. Additionally, the running time and data requirements of our algorithm (for any fixed k) are polynomial in n, $\frac{1}{\epsilon}$, and $\frac{1}{\delta}$.*

The above algorithm depends exponentially on k^3 because it uses the algorithm in Theorem 1 as a subroutine. Yet the problem of learning an ϵ-close estimate is harder than performing density estimation. In fact, our lower bound for algorithms which learn an ϵ-close estimate is not a lower bound for density estimation - if there are two mixtures of k Gaussians, F and F', for which $D(F, F')$ is exponentially small (in k) either one is a good statistical estimate as a distribution for the other. It remains a very interesting open problem to understand if proper density estimation for mixtures of k Gaussians can be performed in $poly(n, k, \frac{1}{\epsilon})$ samples (with no assumptions on the mixture).

The second corollary that we obtain from Theorem 1 is for clustering. To define the problem of clustering, suppose that during the data sampling process, for each sample point $x_i \in \mathbb{R}^n$, a hidden label $y_i \in \{1, \ldots, k\}$ called the ground truth is generated based upon which component the point was sampled from. A clustering algorithm takes as input m points and outputs a *classifier* $C : \mathbb{R}^n \to \{1, \ldots, k\}$. The error of a classifier is the minimum, over all label permutations, of the probability that the permuted label agrees with the ground truth. Given the mixture parameters, it is easy to see that the optimal clustering algorithm will simply assign a label to each point based on which component has the largest posterior probability. We also obtain the following corollary:

Corollary 3. *For any $n \geq 1$, $\epsilon, \delta > 0$, and any n-dimensional GMM $F = \sum_{i=1}^{k} w_i F_i$, given access to independent samples from an ϵ-statistically learnable mixture F of k Gaussians, there is an algorithm that outputs a classifier C such that with probability at least $1 - \delta$ over the randomization in the algorithm and in selecting the samples, the error of C is at most ϵ larger than the error of the optimal classifier C_{OPT}. Additionally, the running time and data requirements of our algorithm (for any fixed k) are polynomial in n, $\frac{1}{\epsilon}$, and $\frac{1}{\delta}$.*

We will not describe the proofs of these corollaries in detail, because the proofs follow from the main theorem in a nearly identical manner to which the corresponding corollaries for mixtures of two Gaussians in [17] followed from the main theorem of [17]. Rather, our emphasis in this paper is on the problem of learning an ϵ-close estimate, and we use these corollaries to support the claim that the goal of learning an ϵ-close estimate is a strong goal that contains many other well-studied learning results as corollaries.

D. Hurdles to Moving Beyond Two Gaussians

This work leverages several key ideas initially presented in [17] which gave an efficient algorithm for learning mixtures of two Gaussians, with provably minimal assumptions.

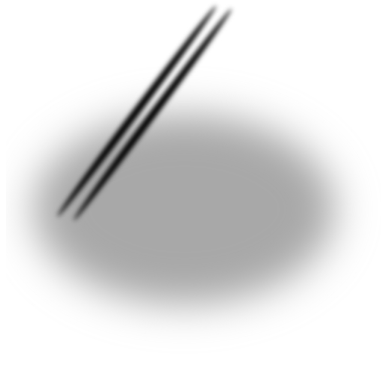

Figure 1. A challenging mixture of three Gaussians to learn: A projection onto a randomly chosen direction usually looks like a mixture of *two* Gaussians.

Nevertheless, additional high-level insights and technical details were required to extend the previous work to give an efficient learning algorithm for mixtures of many Gaussians, again with provably minimal assumptions. In this section we briefly summarize the algorithm for learning mixtures of two Gaussians given in [17], and then describe the hurdles to extending it to the general case. This discussion will provide insights and motivate the high-level structure of the algorithm presented in this paper, as well as clarify which components of the proof are new, and which are straightforward adaptations of ideas from [17].

Throughout this discussion, it will be helpful to refer to parameters $\epsilon_1, \epsilon_2, \epsilon_3$, which are polynomially related to each other, and satisfy $\epsilon_1 << \epsilon_2 << \epsilon_3$.

There are three key components to the proof that mixtures of two Gaussians can be learned efficiently: the 1-d Learnability Lemma, the Random Projection Lemma, and the Parameter Recovery Lemma. The 1-d Learnability Lemma states that given a mixture of two univariate Gaussians whose two components have nonnegligible statistical distance, one can efficiently recover accurate estimates of the parameters of the mixture. It is worth noting that in the univariate case, saying that the statistical distance between two Gaussians is non-negligible is "usually" equivalent to saying that the two sets of parameters are non-negligibly different, i.e. $|\mu - \mu'| + |\sigma^2 - \sigma'^2|$ is non-negligible .

The Random Projection Lemma states that given an n-dimensional ϵ-statistically learnable mixture of two Gaussians which is in isotropic position, (with high probability over the choice of a random unit vector r) the projection of the mixture onto r will yield a univariate mixture of two Gaussians that have nonnegligible statistical distance (say ϵ_3). Let $P_r[F]$ denote the projection of the mixture F onto the direction r.

The final component—the Parameter Recovery Lemma—states that, given a Gaussian G in n dimensions, if one has extremely accurate estimates (say to within some ϵ_1) of the mean and variance of G projected onto n^2 sufficiently distinct directions (directions that differ by at least $\epsilon_2 >> \epsilon_1$) one can accurately recover the multi-dimensional parameters of G.

Given these three pieces, the high-level algorithm for learning mixtures of two Gaussians is straight-forward:

1) Pick a random unit vector r.
2) Pick n^2 vectors $r_1, \ldots, r_{n^2}$, that are "close" to r, say $|r_i - r| \approx \epsilon_2$.
3) For each $i = 1, \ldots, n^2$, learn extremely accurate (to accuracy $\epsilon_1 << \epsilon_2$) univariate parameters $w_i, \mu_i, \sigma_i, \mu_i', \sigma_i'$ for the projection of the mixture onto the vector r_i.
4) Since $|r_i - r_j| \approx \epsilon_2$, the parameters of $P_{r_i}[F_1]$ and $P_{r_j}[F_1]$ must be very close - i.e. much closer than ϵ_3. By the Random Projection Lemma, $|\mu_i - \mu_i'| + |\sigma_i^2 - \sigma_i'^2| >> \epsilon_3$ so we can accurately match up which estimate parameters across different projections come from the same component. We can then apply the Parameter Recovery Lemma to obtain accurate multidimensional estimates of the parameters of each component.

Some of the above ideas are immediately applicable to the problem of learning mixtures of many Gaussians: we can clearly use the Parameter Recovery Lemma without modification. Additionally, we prove a generalization of the 1-d Learnability Lemma for mixtures of many Gaussians, provided each pair of components has non-negligible statistical distance (which, while technically tedious, employs the key idea from [17] of "deconvolving" by a suitably chosen Gaussian). Given this extension, if we were given a mixture of k Gaussians in isotropic position, and were guaranteed that the projection onto some vector r resulted in a univariate mixture of Gaussians for which all pairs of components either had reasonably different means or reasonably different variances, then we could piece together the parts more-or-less as in the case of mixtures of two Gaussians. For example, this would be the case if we assumed the desired precision was polynomially smaller than the additive gap in parameters.

Unfortunately, however, the Random Projection Lemma, ceases to hold in the general setting of mixtures of more than two Gaussians. There are simple mixtures that are ϵ-statistically learnable and are in isotropic position, but with high probability, the projection onto a randomly chosen unit vector r yields a distribution that is extremely close to a univariate mixture of *two* Gaussians. See Figure 1.

How can we recover an n-dimensional estimate that is a mixture of k Gaussians if, in all univariate projections, we see what appears to be a mixture of $k' < k$ components?

II. Algorithm Outline and Definitions

In this section, we explain the high-level structure of our algorithm. Our algorithm uses a similar approach to [17], in which the method of moments is applied to a series of univariate projections of the mixtures, from which the high-dimensional parameters are then reconstructed. This approach, however, is used many times within a larger recursive scheme that uses *hierarchical clustering*, and at intermediate stages of our algorithm an estimate mixture of $k' \leq k$ Gaussians is maintained.

How can an estimate mixture that has the wrong number of components be useful as an intermediate step?

We will say that a component $\hat{F}_i$ in our estimate is additively close to some set of components in the true mixture if it is additively close (in terms of parameter distance) to each component in the set, and additionally mixing weight $\hat{w}_i$ of $\hat{F}_i$ is close to the aggregate mixing weight of all the components in the set. At an intermediate stage, our estimate will be additively close to some partition (into k' sets) of the components in the true mixture.

How do we make progress when one component in the estimate mixture corresponds to a set of more than one component in the true mixture?

We show that the only way in which we end up with several of the original components corresponding to one of the recovered estimated components is if this estimate component has an extremely small variance in some direction — i.e. the minimum eigenvalue of the covariance matrix is extremely small. But in this case we can use this estimate productively: By projecting the samples onto the corresponding eigenvector, we will be able to accurately cluster the sample points in a manner consistent with some partition of the original clusters. Thus (with high probability) each cluster will now consist of samples that come from some mixture of $k'' < k$ components. We note that our algorithm actually works given just an upper bound on the number of components, and so we can now apply our algorithm recursively to each of the clusters.

Eventually we reach a base case in which the set of sample points given to our algorithm has been generated by a single Gaussian, and we can detect this condition. In this case the empirical mean and empirical co-variance of the sample points will be a statistically good estimate to the underlying Gaussian that generated the samples.

To illustrate the approach, consider the toy example of the mixture of three Gaussians given in Figure 1. In this example, we will need to learn an intermediate mixture of two Gaussians. This is because we cannot distinguish the two dark/skinny Gaussians at the current scale. We need to cluster out the samples generated by these two Gaussians, so that we can focus on just these two components and choose the right scale at which to distinguish between them.

In order to cluster out these two Gaussians, we need to first learn a direction in which the two Gaussians have small variance. This is precisely why we first learn an estimate mixture of two Gaussians (that is additively close to the true mixtures) – we can use this estimate to find a direction to project onto so that we can cluster out these two components.

A. Definitions

Definition 3. *Given two probability distributions $f(x), g(x)$ on $\Re^n$ we can define the statistical distance between these distributions as*

$$D(f(x), g(x)) = \frac{1}{2} \int_{\Re^n} |f(x) - g(x)| dx$$

We will also be interested in a related notion of the parameter distance between two univariate Gaussians:

Definition 4. *Given two univariate Gaussians, $F_1 = \mathcal{N}(\mu_1, \sigma_1^2), F_2 = \mathcal{N}(\mu_2, \sigma_2^2)$ we define the parameter distance as*

$$D_p(F_1, F_2) = |\mu_1 - \mu_2| + |\sigma_1^2 - \sigma_2^2|$$

In general, the parameter distance and the statistical distance between two univariate Gaussians can be unrelated. There are pairs of univariate Gaussians with arbitrarily small parameter distance, and yet statistical distance close to 1, and there are pairs of univariate Gaussians with arbitrarily small statistical distance, and yet arbitrarily large parameter distances. But these scenarios can only occur if the variances can be arbitrarily small or arbitrarily large. In many instances in this paper, we will have reasonable upper and lower bounds on the variances and this will allow us to move back and forth from statistical distance and parameter distance.

As we noted, there are simple examples of an ϵ-statistically learnable mixture of three Gaussians which as a distribution is in isotropic position, but for which with overwhelming probability in a projection onto a randomly chosen direction r, there will be some pair of univariate Gaussians that are arbitrarily close in parameter distance. In such a case, we must relax the goal of returning an accurate estimate which is a mixture of three Gaussians – Instead, our univariate algorithm will return a mixture which has only two components but is still in some sense a good estimate for the parameters of the projected mixture. To formalize this notion, we introduce what we call an ϵ-correct sub-division.

Definition 5. *Given a GMM of k Gaussians, $F = \sum_i w_i \mathcal{N}(\mu_i, \sigma_i^2)$ and a GMM of $k' \leq k$ Gaussians $\hat{F} = \sum_i \hat{w}_i \mathcal{N}(\hat{\mu}_i, \hat{\sigma}_i^2)$, we call $\hat{F}$ an ϵ-correct subdivision of F if there is a function $\pi : [k] \rightarrow [k']$ that is onto and*

- $\forall_{j \in [k']} |\sum_{i|\pi(i)=j} w_i - \hat{w}_j| \leq \epsilon$
- $\forall_{i \in [k]} D_p(F_i, \hat{F}_{\pi(i)}) \leq \epsilon$

When considering high-dimensional mixtures, we replace the above parameter distance by $\|\mu_i - \hat{\mu}_{\pi(i)}\| + \|\Sigma_i - \hat{\Sigma}_{\pi(i)}\|_F \leq \epsilon$, where $\|\|_F$ denotes the Frobenius norm.

Notationally, we will write $(\hat{F}, \pi) \in \mathcal{D}_\epsilon(F)$ as shorthand for the statement that $\hat{F}$ is an ϵ-correct subdivision for F and π is the (onto) function from k to k' that groups F into $\hat{F}$ as above.

Note that this definition, unlike the definition for an ϵ-close estimate, uses parameter distance as opposed to statistical distance.

III. A Robust Univariate Algorithm

In this section, we give a learning algorithm for univariate mixtures of Gaussians that will be the building block for our learning algorithm in n-dimensions. Unlike in the case of [17], the input to our univariate algorithm will not necessarily be a mixture of Gaussians for which all pairwise parameter distances are reasonably large. Instead, it could happen that we are given a mixture of (say) three Gaussians so that some pair has arbitrarily small parameter distance.

In the case in which we are guaranteed that all pairwise parameter distances are reasonably large, we can iterate the technical ideas in [17] to give an inductive proof that a simple brute force search algorithm will return a good estimate with the correct number of components. We call this algorithm the Basic Univariate Algorithm. From this, we build a General Univariate Algorithm that will return a good estimate regardless of the parameter distances, although in order to do so we will need to relax the notion of a good estimate to something weaker: the algorithm will return an ϵ-correct subdivision, and in this case the algorithm can return an estimate with strictly fewer components as long as this estimate is consistent with some partition of the original mixture.

A. Polynomially Robust Identifiability

In this section, we show that we can efficiently learn the parameters of univariate mixtures of Gaussians, provided that the components of the mixture have nonnegligible pairwise parameter distances. We emphasize again that this algorithm will return the correct number of components, because it is run with precision fine enough that all pairs of components look different. We refer to this algorithm as the Basic Univariate Algorithm. Such an algorithm will follow easily from Theorem 4—the polynomially robust identifiability of univariate mixtures. Throughout this section we will consider two univariate mixtures of Gaussians:

$$F(x) = \sum_{i=1}^{n} w_i \mathcal{N}(\mu_i, \sigma_i^2, x),\ F'(x) = \sum_{i=1}^{k} w_i' \mathcal{N}(\mu_i', \sigma_i'^2, x).$$

Definition 6. *We will call the pair F, F' ϵ-standard if $\sigma_i^2, \sigma_i'^2 \leq 1$ and if ϵ satisfies:*

- $w_i, w_i' \in [\epsilon, 1]$
- $|\mu_i|, |\mu_i'| \leq \frac{1}{\epsilon}$
- $|\mu_i - \mu_j| + |\sigma_i^2 - \sigma_j^2| \geq \epsilon$ *and* $|\mu_i' - \mu_j'| + |\sigma_i'^2 - \sigma_j'^2| \geq \epsilon$ *for all* $i \neq j$
- $\epsilon \leq \min_\pi \sum_i \left(|w_i - w'_{\pi(i)}| + |\mu_i - \mu'_{\pi(i)}| + |\sigma_i^2 - \sigma'^2_{\pi(i)}|\right)$, *where the minimization is taken over all mappings* $\pi : \{1, \ldots, n\} \to \{1, \ldots, k\}$.

Let $M_i[F] = E_{x \sim F}[x^i]$ i.e. $M_i[F]$ is the i^{th} raw moment of the distribution F.

Theorem 4. *There is a constant $c > 0$ such that, for any ϵ-standard F, F' and any $\epsilon < c$,*

$$\max_{i \leq 2(n+k-1)} |M_i(F) - M_i(F')| \geq \epsilon^{O(k)}$$

We note that it is known that there are distinct mixtures of k Gaussians that can match exactly on the first linearly (in k) many raw moments.

Given the polynomially robust identifiability guaranteed by the above theorem, and simple concentration bounds on the i^{th} sample moment, it is easy to see that a brute-force search over a set of candidate parameter sets will yield an efficient algorithm that recovers the parameters for a univariate mixture of Gaussians whose components have pairwise parameter distance at least ϵ: roughly, the Basic Univariate Algorithm will take a polynomial number of samples, compute the first $4k-2$ sample moments empirically, and compare those with the first $4k-2$ moments (which are computed analytically) of each of the candidate parameter sets in a grid search. The algorithm then returns the parameter set whose moments most closely match the empirical moments. Theorem 4 guarantees that if the first $4k-2$ sample moments very closely match those of the chosen parameter set, then the parameter set must be nearly accurate. To conclude the proof, we argue that a polynomial-sized set of candidate parameters suffices to guarantee that at least one set of parameters will yield analytic moments very close to the emprical moments. We state the corollary below, and defer the details of the algorithm and the proof of its correctness to the full version of our paper.

Corollary 5. *Suppose we are given access to independent samples from a GMM $\sum_{i=1}^{k} w_i \mathcal{N}(\mu_i, \sigma_i^2, x)$ with mean 0 and variance in the interval $[1/2, 2]$, where $w_i \geq \epsilon$, and $|\mu_i - \mu_j| + |\sigma_i^2 - \sigma_j^2| \geq \epsilon$. There is an algorithm that, for any fixed k, has runtime and sample complexity at most $poly(\frac{1}{\epsilon}, \frac{1}{\delta})$ and with probability at least $1 - \delta$ will output mixture parameters $\hat{w}_i, \hat{\mu}_i, \hat{\sigma_i}^2$, so that there is a permutation $\pi : [k] \to [k]$ and for each $i = 1, \ldots, k$:*

$$|w_i - \hat{w}_{\pi(i)}| \leq \epsilon, \quad |\mu_i - \hat{\mu}_{\pi(i)}| \leq \epsilon, \quad |\sigma_i^2 - \hat{\sigma}^2_{\pi(i)}| \leq \epsilon$$

B. The General Univariate Algorithm

In this section we seek to extend the Basic Univariate Algorithm of Corollary 5 to the general setting of a univariate mixture of k Gaussians without any requirements that the components have significant pair-wise parameter distance. In particular, given some target accuracy ϵ, and access to independent samples from a mixture F of k univariate Gaussians, we want to efficiently compute a mixture F' of $k' \leq k$ Gaussians that is an ϵ-correct subdivision of F.

We say that a mixture is in near isotropic position if the mean is zero and the variance is between $\frac{1}{2}$ and 2.

Proposition 6. *There is an algorithm which, given $\epsilon, \delta > 0$, and access to a GMM of at most k Gaussians, $F = \sum_i w_i \mathcal{N}(\mu_i, \sigma_i^2)$ that is in near isotropic position and satisfies $w_i \geq \epsilon$ (for any fixed k) has runtime and sample complexity at most $poly(\frac{1}{\epsilon}, \frac{1}{\delta})$ and with probability at least $1 - \delta$ will output a GMM of $k' \leq k$ Gaussians $\hat{F}$ that is an ϵ-correct subdivision of F.*

The critical insight in building up such a GENERAL UNIVARIATE ALGORITHM is that if two components are actually close enough (in statistical distance), then because the BASIC UNIVARIATE ALGORITHM only requires a polynomial number of samples, these two components will look (to our algorithm) as if they were a single Gaussian. So given a target precision ϵ_1 for the BASIC UNIVARIATE ALGORITHM, there is some window that describes whether or not the algorithm will work correctly: If all pairwise parameter distances are either sufficiently large or sufficiently small, then the BASIC UNIVARIATE ALGORITHM will behave as if it were given sample access to a mixture that actually does meet the requirements of the algorithm. Groups of components for which all pairs are very close in parameter distance will look roughly the same as if we were to replace the entire group with a singe, appropriately chosen Gaussian. And all pairs of groups will be sufficiently different in parameter distance that the BASIC UNIVARIATE ALGORITHM will be able to tell them apart.

However, when there is some parameter distance that falls inside the BASIC UNIVARIATE ALGORITHM's window, we are not guaranteed that the BASIC UNIVARIATE ALGORITHM will fail safely. The idea, then, is to use many disjoint windows (each of which corresponds to running the BASIC UNIVARIATE ALGORITHM with some target precision). If we choose enough such windows, each pairwise parameter distance can only corrupt a single run of the BASIC UNIVARIATE ALGORITHM so a majority of the computations will be correct. We defer the algorithm and proof of correctness to the full version of our paper.

IV. PARTITION PURSUIT

In this section we demonstrate how to use the GENERAL UNIVARIATE ALGORITHM to obtain good additive approximations in n-dimensions. Roughly, we will project the n-dimensional mixture F onto many close-by directions, and run the GENERAL UNIVARIATE ALGORITHM on each projection. This is also how the algorithm in [17] is able to recover good additive estimates in n-dimensions. However we will have to cope with the additional complication that our univariate algorithm (the GENERAL UNIVARIATE ALGORITHM) does not necessarily return an estimate that is a mixture of k Gaussians.

We explain in detail how the algorithm in [17] is able to obtain additive approximation guarantees in n-dimensions, building on a univariate algorithm for learning mixtures of two Gaussians: Let $P_r[F]$ denote the univariate mixture that results from projecting F onto direction r. Let $\epsilon_3 >> \epsilon_2 >> \epsilon_1$. Given any ϵ-statistically learnable mixture of two Gaussians in n-dimensions that is in isotropic position, with high probability the parameter distance between the two univariate Gaussians in $P_r[F]$ that result from projecting on a randomly chosen direction r will be at least ϵ_3.

Then given such a direction r, we can choose n^2 different directions $r_{x,y}$ each of which are ϵ_2-close to r (i.e. $\|r - r_{x,y}\| \approx \epsilon_2$). We can bound how much the mean and variance of a component in $P_u[F]$ can change as we vary the direction u from r to $r_{x,y}$, and this will imply that for $\epsilon_2 << \epsilon_3$, we can consistently pair up estimates recovered from each projection, so that for each component we have n^2 different estimates of the projected mean and variance corresponding to the n^2 different directions. Each of these estimates are accurate to within ϵ_1 (i.e. this is the target precision that is given to the univariate algorithm).

Note that $P_r[\mathcal{N}(\mu, \Sigma)] = \mathcal{N}(r^T\mu, r^T\Sigma r)$ and so for any Gaussian, an estimate for the projected mean and the projected variance on a direction r gives a linear constraint on the mean vector μ and the co-variance matrix Σ. Taking the union of these linear constraints on μ and Σ for each of the n^2 directions, we obtain a system of constraints for μ and Σ which has condition number $poly(\frac{1}{\epsilon_2}, n)$ [17]. As a result, if $\epsilon_1 << \epsilon_2$ then the precision is much finer than the condition number of the system of linear constraints on μ, Σ and we can back-solve to obtain additively accurate estimates for μ and Σ in n-dimensions. We can directly use the bound on the condition number given in [17]:

Lemma 7. *[17] Let $\epsilon_2, \epsilon_1 > 0$. Suppose $|m^0 - \mu \cdot r|$,$|m^{ij} - \mu \cdot r^{ij}|$, $|v^0 - r^T\Sigma r|$,$|v^{ij} - (r^{ij})^T\Sigma r^{ij}|$ are all at most ϵ_1. Then there is an algorithm that outputs $\hat{\mu} \in \mathbf{R}^n$ and $\hat{\Sigma} \in \mathbf{R}^{n \times n}$ such that $\|\hat{\mu} - \mu\| < \frac{\epsilon_1\sqrt{n}}{\epsilon_2}$, and $\|\hat{\Sigma} - \Sigma\|_F \leq \frac{6n\epsilon_1}{\epsilon_2^2}$. Furthermore, $\hat{\Sigma} \succeq 0$ and $\hat{\Sigma}$ is symmetric.*

The algorithm to which this lemma refers is given [17] and in the full version of our paper.

However, the GENERAL UNIVARIATE ALGORITHM does not always return a mixture of k Gaussians, and can in fact return a mixture $\hat{F}^u$ of $k' < k$ Gaussians provided that this mixture is still an ϵ_1-correct subdivision of $P_u[F]$ (for some direction u). But then what happens if we consider two close-by directions, u and v and the number of Gaussians in the estimate $\hat{F}^u$ is different from the number of Gaussians in the estimate $\hat{F}^v$?

The key insight is that if we choose some direction r, and close-by directions $r_{x,y}$, we can ensure that if we see a different number of components in some close by direction, we will see strictly more. In such a case, if on some direction v we see strictly more components than for direction u, we can just restart our algorithm on direction v, throwing out all information we have observed so afar. But we have made progress because we have seen more components.

Suppose we reach a state in which in each (close by) direction, we observe mixtures of k' Gaussians. In each projection, the set of Gaussians in our estimate corresponds to a partition of the original mixture. Suppose every pair of Gaussians in every projection has a large parameter distance (say ϵ_3) and if all directions are sufficiently close (say $\epsilon_2 << \epsilon_3$), then the partitions (of the true components) to which each estimate corresponds must be the same across different directions. This is only true when the number of Gaussians in each estimate is exactly the same. We can finally apply the above bound on the condition number to obtain a multidimensional mixture of $k' \leq k$ Gaussians that is an ϵ-correct sub-division for F - i.e. is close to some partition of F.

We state our main proposition in this section, and defer the algorithm and proof to the full version of our paper.

Proposition 8. *Given access to an ϵ-statistically learnable GMM F of at most k Gaussians which as a mixture is in isotropic position, the* PARTITION PURSUIT ALGORITHM *(for any fixed k) has runtime and sample complexity at most $poly(\frac{1}{\epsilon}, \frac{1}{\delta})$ and with probability at least $1 - \delta$ will output an ϵ-correct sub-division $\hat{F}$ and if F has more than one component, $\hat{F}$ also has more than one component.*

V. Clustering and Recursion

In this section, we give an efficient algorithm for learning an estimate $\hat{F}$ that is ϵ-close to the actual mixture F. PARTITION PURSUIT assumes that the mixture F is in isotropic position, and even though F is not necessarily in isotropic position, we can first take enough samples to compute a transformation that places the mixture F in nearly isotropic position and then applying this transformation to each sample from the oracle.

The main technical challenge in this section is actually what to do when the mixture $\hat{F}$ returned by PARTITION PURSUIT is a good additive approximation to F (i.e. it is an ϵ_1-correct subdivision with $\epsilon_1 << \epsilon$), but is not ϵ-close to the mixture F. This can only happen if there is a component in F that has a very small variance in some direction (because otherwise additive guarantees yield statistical guarantees). Consider for example, two univariate Gaussians $\mathcal{N}(0, \gamma)$ and $\mathcal{N}(0, \gamma + \epsilon_1)$. Even if ϵ_1 is very small, if γ is much smaller, then the statistical distance between these two Gaussians can be arbitrarily close to 1.

The high-level idea is that if the estimate $\hat{F}$ returned by PARTITION PURSUIT is not ϵ-close to F (but $\hat{F}$ is an ϵ_1-correct subdivision of F for $\epsilon_1 << \epsilon$), then it must be the case that some component $\hat{F}_i$ of $\hat{F}$ has a co-variance matrix $\hat{\Sigma}_i$ with the property that for some direction v, $v^T \hat{\Sigma}_i v$ is very small. We can then use this direction v to make progress: If we project the mixture F onto v, we will be able to cluster accurately. There will be some partition of the components in F into two disjoint, non-empty sets of components S, T and some clustering scheme that can accurately clusters points sampled from F into points that originated from a component in S and points that originated from a component in T. We show that we can accurately cluster enough points sampled from F into sets of points that originated from components in S and sets of points that originated from components in T, so that we can then recursively run our learning algorithm on each of the two sets of samples, which now each correspond to samples from a mixture of strictly fewer components.

The main technical challenge is in showing that if there is some component of $\hat{F}$ with a small enough variance in some direction v, then we can accurately cluster points sampled from F (obviously provided that $\hat{F}$ is additively close to F). Given this, our main result follows almost immediately from an inductive argument.

A. How to Cluster

Here we formalize the notion of a clustering scheme. Additionally, we state the key lemmas that will be useful in showing that if $\hat{F}$ is not an ϵ-close estimate to F, we can use $\hat{F}$ to construct a good clustering scheme that makes progress on our learning problem.

Definition 7. *We will call $A, B \subset \Re^n$ a clustering scheme if $A \cap B = \emptyset$*

Definition 8. *For $A \subset \Re^n$, we will write $P[F_i, A]$ to denote $Pr_{x \sim F_i}[x \in A]$ - i.e. the probability that a randomly chosen sample from F_i is in the set A.*

The intuition is clearest in the case of mixtures of two Gaussians: Suppose one of the components, say $\hat{F}_1$, had small variance on direction v. If the entire mixture is in isotropic position, then the variance of the mixture when projected onto direction v is 1. This can only happen if either the difference in projected means $|v^T(\hat{\mu}_1 - \hat{\mu}_2)|$ is a constant or the variance of $\hat{F}_2$ on direction v is a constant. In the first case, we can choose an interval around each projected (estimate) mean $v^T \hat{\mu}_1$ and $v^T \hat{\mu}_2$ so that with high probability, any point sampled from F_1 is contained in the interval around $v^T \hat{\mu}_1$ and similarly for F_2.

If, instead, the variance of F_2 when projected onto v is a constant, then again a small interval around the point $v^T \hat{\mu}_1$ will contain most samples from F_1, but because the maximum density of $v^T F_2$ is never large and the interval around $v^T \hat{\mu}_1$ is very short, with high probability samples from F_2 will not be contained in the interval. This idea is the basis of our clustering lemmas. Although there will be additional complications when the mixture contains more than two Gaussians, the intuition is close to the same.

Let $(\hat{F}, \pi) \in \mathcal{D}_{\epsilon_1}(F)$. Suppose also that $\hat{F}$ is a mixture of k' components.

Lemma 9. *Suppose that for some direction v, for all i: $v^T \hat{\Sigma}_i v \leq \epsilon_2$, for $\epsilon_1 \leq \frac{\sqrt{\epsilon_2}}{2\epsilon_3}$. If there is some bi-partition $S \subset [k']$ s.t. $\forall_{i \in S, j \in [k']-S} |v^T \hat{\mu}_i - v^T \hat{\mu}_j| \geq \frac{3\sqrt{\epsilon_2}}{\epsilon_3}$ then there is a clustering scheme (A, B) (based only on $\hat{F}$) so that for all $i \in S, j \in \pi^{-1}(i)$, $P[F_i, A] \geq 1 - \epsilon_3$ and for all $i \notin S, j \in \pi^{-1}(i)$, $Pr[F_i, B] \geq 1 - \epsilon_3$.*

This lemma corresponds to the first case in the above thought exercise when there is some bi-partition of the components so that all pairs of projected means across the bi-partition are reasonably separated.

Lemma 10. *Suppose there is some direction v and some $i \in [k']$ such that: $v^T \hat{\Sigma}_i v \leq \epsilon_m$, for $\epsilon_m >> \epsilon_1$. If there is some bi-partition $S \subset [k']$ s.t.*

$$\frac{\min_{i \in S} v^T \hat{\Sigma}_i v}{\max(\max_{j \notin S} v^T \hat{\Sigma}_j v, \epsilon_m)} \geq \frac{1}{\epsilon_t}$$

(and $\epsilon_t << \epsilon_3^3$) then there is a clustering scheme A, B such that for all $i \in S, j \in \pi^{-1}(i)$, $P[F_i, A] \geq 1 - \epsilon_3$ and for all $i \notin S, j \in \pi^{-1}(i)$, $Pr[F_i, B] \geq 1 - \epsilon_3$.

This lemma corresponds to the second case, when there is some bi-partition of the components so that one side of the bi-partition has projected variances that are much larger than the other. The proofs of these lemmas are given in the full version of our paper.

B. Making Progress when there is a Small Variance

We state a lemma from [17] which formalizes the intuition that if there is no component in $\hat{F}$ has small variance in any direction, then $\hat{F}$ is a good statistical estimate to F:

Lemma 11. *[17] Suppose $\|\hat{\mu}_i - \mu_i\| \leq \epsilon_1$, $\|\hat{\Sigma}_i - \Sigma_i\|_F \leq \epsilon_1$, and $|\hat{w}_i - w_i| \leq \epsilon_1$, if either $\|\Sigma_i^{-1}\|_2 \leq \frac{1}{2\epsilon_m}$ or $\|\hat{\Sigma}_i^{-1}\|_2 \leq \frac{1}{2\epsilon_m}$ then $D(\hat{F}_i, F_i)^2 \leq \frac{2n\epsilon_1}{\epsilon_m} + \frac{\epsilon_1^2}{2\epsilon_m}$.*

Additionally, we argue that if we have an ϵ-statistically learnable mixture of at least two components, that is in isotropic position, then it must be the case that there are at least two components whose parameters differ non-negligibly on a random projection.

Lemma 12. *[Isotropic Projection Lemma] Given a mixture of k n-Dimensional Gaussians $F = \sum_i w_i F_i$ which is in isotropic position and is ϵ-statistically learnable, with probability $\geq 1 - \delta$ over a randomly chosen direction u, there is some pair of Gaussians F_i, F_j s.t. $D_p(P_u[F_i], P_u[F_j]) \geq \frac{\epsilon^5 \delta^2}{50 n^2}$.*

The above two lemmas serve as the building blocks of the following proposition, which, guarantees that our algorithm will either accurately recover an estimate mixture, or will be able to successfully partition the mixture and accurately cluster the set of samples. We defer the proof of the above lemma to the full version of our paper.

Proposition 13. *Given access to an ϵ-statistically learnable GMM F of at most k Gaussians which as a mixture is in isotropic position, there is an algorithm that (for any fixed k) has runtime and sample complexity at most $poly(\frac{1}{\epsilon}, \frac{1}{\delta})$ and with probability at least $1 - \delta$ will output either an ϵ-close statistical estimate $\hat{F}$ for F, or returns a clustering scheme A, B such that there is some bipartition $S \subset [k]$ such that for all $i \in S, j \in \pi^{-1}(i)$, $P[F_i, A] \geq 1 - \epsilon_3$ and for all $i \notin S, j \in \pi^{-1}(i)$, $Pr[F_i, B] \geq 1 - \epsilon_3$. And also $S, [k] - S$ are both non-emtpy.*

C. Recursion

Our learning algorithm will work recursively; given a set of samples that come from a mixture of *at most* k' Gaussian components, we put the samples into isotropic position and run the algorithm of Proposition 13, which will either return an ϵ-close estimate, or a nontrivial clustering (A, B) of the samples that respects some bipartition of the components of the mixture. We will choose the parameters such that the clustering error is so small that, with high probability, we can recursively run the algorithm on subsamples from the two clusters A and B. Each set of samples can then be viewed as coming from a mixture of strictly fewer components than the original set of samples. How do we know when we are done? As a consequence of Lemma 12, if we put the mixture into roughly isotropic position then project onto a random vector, with high probability the GENERAL UNIVARIATE ALGORITHM will return a single component if, and only if the high-dimensional mixture actually consisted of a single component.

Our main theorem now follows by induction on the upper bound on the number of components in our mixture–essentially verifying that the recursive structure of our algorithm is correct. We defer the proof to the full version.

Theorem 1. Given an n dimensional mixture of k Gaussians F that is ϵ-statistically learnable, there is an algorithm that, with probability at least $1 - \delta$, outputs an ϵ-close estimate $\hat{F}$ and the running time and data requirements of our algorithm (for any fixed k) are polynomial in n, $\frac{1}{\epsilon}$, and $\frac{1}{\delta}$.

VI. EXPONENTIAL DEPENDENCE ON k IS INEVITABLE

We also give a lower bound, showing that exponential dependence on the number of components is necessary for learning ϵ-close estimates, even for mixtures in just one dimension. We show this by giving a simple construction of two univariate distributions, D_1, D_2 that are $1/(2m)$-standard. Specifically, each distribution is a mixture of at most m Gaussians, the weight of each component in each mixture is at least $1/(2m)$, and the parameter distance between the distributions is at least $1/(2m)$, but $||D_1 - D_2||_1 \leq e^{-m/30}$ for sufficiently large m. The construction hinges on the inverse exponential (in $k \approx \sqrt{m}$) statistical distance between $\mathcal{N}(0, 2)$, and a mixture of infinitely many Gaussians of unit variance whose components are centered at multiples of $1/k$, where the weight assigned to the component centered at i/k is given by $N(0, 1, i/k)$. Verifying this claim is an exercise in Fourier analysis. We then modify the example slightly so that it meets the conditions of being $1/(4k^2 + 2)$-standard.

Theorem 14. *There exists a pair D_1, D_2 of $1/(4k^2 + 2)$-standard distributions that are each mixtures of $k^2 + 1$ Gaussians such that*

$$||D_1 - D_2||_1 \leq 11ke^{-k^2/24}.$$

Acknowledgment

We are indebted to Adam Tauman Kalai, for introducing us to this beautiful line of research, and for all his insights, encouragement, and practical jokes.

References

[1] D. Achlioptas and F. McSherry. On spectral learning of mixtures of distributions. In *COLT*, pages 458–469, 2005.

[2] S. Arora and R. Kannan. Learning mixtures of arbitrary Gaussians. In *STOC*, pages 247–257, 2001.

[3] M. Belkin and K. Sinha. Learning Gaussian mixtures with arbitrary separation. In *COLT*, 2010, to appear.

[4] M. Belkin and K. Sinha. Polynomial learning of distribution families. In *FOCS*, 2010, this proceedings.

[5] S. C. Brubaker and S. Vempala. Isotropic PCA and affine-invariant clustering. In *FOCS*, pages 551–560, 2008.

[6] K. Chaudhuri and S. Rao. Learning mixtures of product distributions using correlations and independence. In *COLT*, pages 9–20, 2008.

[7] K. Chaudhuri and S. Rao. Beyond Gaussians: Spectral methods for learning mixtures of heavy-tailed distributions. In *COLT*, pages 21–32, 2008.

[8] A. Dasgupta, J. Hopcroft, J. Kleinberg, and M. Sandler. On learning mixtures of heavy-tailed distributions. In *FOCS*, pages 491–500, 2005.

[9] S. Dasgupta. Learning mixtures of Gaussians. In *FOCS*, pages 634–644, 1999.

[10] S. Dasgupta, A. T. Kalai, and C. Monteleoni. Analysis of perceptron-based active learning. In *COLT*, pages 249–263, 2005.

[11] S. Dasgupta and L. J. Schulman. A two-round variant of EM for Gaussian mixtures. In *UAI*, pages 152–159, 2000.

[12] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM Algorithm. *J. Roy. Statist. Soc. Ser. B*, 39:1–38, 1977.

[13] J. Feldman, R. A. Servedio, and R. O'Donnell. PAC learning axis-aligned mixtures of Gaussians with no separation assumption. In *COLT*, pages 20–34, 2006.

[14] A. A. Giannopoulos and V. D. Milman. Concentration property on probability spaces. *Adv. Math.*, 156:77–106, 2000.

[15] P. J. Huber. Projection pursuit. *Ann. Statist.* 13:435–475, 1985.

[16] R. A. Hummel and B. C. Gidas. Zero crossings and the Heat Equation. *Technical Report Number 111, Courant Institute of Mathematical Sciences at NYU*, 1984.

[17] A. T. Kalai, A. Moitra, and G. Valiant. Efficiently learning mixtures of two Gaussians. In *STOC*, 2010, to appear.

[18] R. Kannan, H. Salmasian, and S. Vempala. The spectral method for general mixture models. *SIAM J. Comput.*, 38(3):1141–1156, 2008.

[19] M. J. Kearns, Y. Mansour, D. Ron, R. Rubinfeld, R. E. Schapire, and L. Sellie. On the learnability of discrete distributions. In *STOC*, pages 273–282, 1994.

[20] L. Leindler. On a certain converse of Hölder's Inequality ii. *Acta Sci. Math. Szeged*, 33:217–223, 1972.

[21] B. Lindsay. *Mixture models: theory, geometry and applications*. American Statistical Association, 1995.

[22] L. Lovász and S. Vempala. The geometry of logconcave functions and sampling algorithms. *Random Struct. Algorithms*, 30(3):307–358, 2007.

[23] G.J. McLachan and D. Peel, *Finite Mixture Models* (2009), Wiley.

[24] K. Pearson. Contributions to the mathematical theory of evolution. *Phil. Trans. R. Soc. Lond. A*, 1894.

[25] A. Prékopa. Logarithmic concave measures and functions. *Acta. Sci. Math. Szeged*, 34:335–343, 1973.

[26] R. A. Redner and H. F. Walker. Mixture densities, maximum likelihood and the EM algorithm. *SIAM Rev.* , 26(2):195-239, 1984.

[27] M. Rudelson. Random vectors in the isotropic position. *J. Funct. Anal*, 164:60–72, 1999.

[28] M. Rudelson and R. Vershynin. Sampling from large matrices: An approach through geometric functional analysis. *J. ACM*, 54(4): 2007.

[29] H. Teicher. Identifiability of mixtures. *Ann. Math. Statist.*, 32(1):244–248, 1961.

[30] D.M. Titterington, A.F.M. Smith, and U.E. Makov. *Statistical analysis of finite mixture distributions* (1985), Wiley.

[31] L. Valiant. A theory of the learnable. *Comm. ACM*, 27(11):1134–1142, 1984.

[32] S. Vempala and G. Wang. A spectral algorithm for learning mixture models. *J. Comput. Syst. Sci.*, 68(4):841–860, 2004.

[33] C. F. J. Wu. On the convergence properties of the EM Algorithm. *Ann. Stat.*, 11(1):95–103, 1983.