Learning Mixtures of Product Distributions using Correlations and Independence

Kamalika Chaudhuri
Information Theory and Applications, UC San Diego
kamalika@soe.ucsd.edu

Satish Rao
Computer Science Division, UC Berkeley
satishr@cs.berkeley.edu

Abstract

We study the problem of learning mixtures of distributions, a natural formalization of clustering. A mixture of distributions is a collection of distributions $\mathcal{D} = \{D_1, \dots D_T\}$, and *mixing weights*, $\{w_1, \dots, w_T\}$ such that $\sum_i w_i = 1$. A sample from a mixture is generated by choosing i with probability w_i and then choosing a sample from distribution D_i. The problem of learning the mixture is that of finding the parameters of the distributions comprising $\mathcal{D}$, given only the ability to sample from the mixture. In this paper, we restrict ourselves to learning mixtures of product distributions.

The key to learning the mixtures is to find a *few* vectors, such that points from different distributions are sharply separated upon projection onto these vectors. Previous techniques use the vectors corresponding to the top few directions of highest variance of the mixture. Unfortunately, these directions may be directions of high noise and not directions along which the distributions are separated. Further, skewed mixing weights amplify the effects of noise, and as a result, previous techniques only work when the separation between the input distributions is large relative to the imbalance in the mixing weights.

In this paper, we show an algorithm which successfully learns mixtures of distributions with a separation condition that depends only logarithmically on the skewed mixing weights. In particular, it succeeds for a separation between the centers that is $\Theta(\sigma\sqrt{T \log \Lambda})$, where σ is the maximum directional standard deviation of any distribution in the mixture, T is the number of distributions, and Λ is polynomial in T, σ, $\log n$ and the imbalance in the mixing weights. For our algorithm to succeed, we require a *spreading condition*, that the distance between the centers be *spread* across $\Theta(T \log \Lambda)$ coordinates. Additionally, with arbitrarily small separation, *i.e.,* even when the separation is not enough for clustering, with enough samples, we can approximate the subspace containing the centers. Previous techniques failed to do so in polynomial time for non-spherical distributions regardless of the number of samples, unless the separation was large with respect to the maximum directional variance σ and polynomially large with respect to the imbalance of mixing weights.Our algorithm works for *Binary Product Distributions* and *Axis-Aligned Gaussians*. The spreading condition above is implied by the separation condition for binary product distributions, and is necessary for algorithms that rely on linear correlations.

Finally, when a stronger version of our spreading condition holds, our algorithm performs successful clustering when the separation between the centers is only $\Theta(\sigma_* \sqrt{T \log \Lambda})$, where σ_* is the maximum directional standard deviation in the subspace containing the centers of the distributions.

1 Introduction

Clustering, the problem of grouping together data points in high dimensional space using a similarity measure, is a fundamental problem of statistics with numerous applications in a wide variety of fields. A natural model for clustering is that of *learning mixtures of distributions*. A mixture of distributions is a collection of distributions $\mathcal{D} = \{D_1, \dots D_T\}$, and *mixing weights*, $\{w_1, \dots, w_T\}$ such that $\sum_i w_i = 1$. A sample from a mixture is generated by choosing i with probability w_i and choosing a sample from distribution D_i. The problem of learning the mixture is that of finding the parameters of the distributions comprising $\mathcal{D}$, given only the ability to sample from the mixture.

If the distributions $D_1, \dots, D_T$ are very close to each other, then even if we knew the parameters of the distributions, it would be impossible to classify the points correctly with high confidence. Therefore, Dasgupta [Das99] introduced the notion of a *separation condition*, which is a promise that each pair of distributions is sufficiently different according to some measure. Given points from a mixture of distributions and a separation condition, the goal is to find the parameters of the mixture $\mathcal{D}$, and cluster all but a small fraction of the points correctly. A commonly used separation measure is the distance between the centers of the distributions parameterized by the maximum directional variance, σ, of any distribution in the mixture.

A common approach to learning the mixtures and therefore, clustering the high-dimensional cloud of points is to find a *few* interesting vectors, such that points from different distributions are sharply separated upon projection onto these vectors. Various distance-based methods [AK01, Llo82, DLR77] are then applied to cluster in the resulting low-dimensional subspace. The state-of-the-art, in practice, is to use the vectors corresponding to the top few directions of *highest variance* of the mixture and to hope that it contains most of the separation between the centers. This is computed by a *Singular Value Decomposition*(SVD) of the matrix of samples. This approach has been theoretically analyzed by [VW02] for spherical distributions, and for more general distributions in [KSV05, AM05]. The latter show that the maximum variance directions are indeed the interesting directions when the separation is $\Theta(\frac{\sigma}{\sqrt{w_{\min}}})$, where $w_{\min}$ is the smallest mixing weight of any distribution.

This is the best possible result for SVD-based approaches; the directions of maximum variance may well not be the directions in which the centers are separated, but instead may be the directions of very high noise, as illustrated in Figure 1(b). This problem is exacerbated when the mixing weights w_i are skewed – because a distribution with low mixing weight diminishes the contribution to the variance along a direction that separates the centers.

This bound is suboptimal for two reasons. Although mixtures with skewed mixing weights arise naturally in practice(see [PSD00] for an example), given enough samples, mixing weights have no bearing on the separability of distributions. Consider two mixtures $\mathcal{D}'$ and $\mathcal{D}''$ of distributions D_1 and D_2: in $\mathcal{D}'$, $w_1 = w_2 = 1/2$, and in $\mathcal{D}''$, $w_1 = 1/4$ and $w_2 = 3/4$. Given enough computational resources, if we can learn $\mathcal{D}'$ from 50 samples, we should be able to learn $\mathcal{D}''$ from 100 samples. This does not necessarily hold for SVD-based methods. Secondly, regardless of σ, an algorithm, which has prior knowledge of the subspace containing the centers of the distributions, should be able to learn the mixture when the separation is proportional to σ_*, the maximum directional standard deviation of any distribution in the subspace containing the centers. An example in which σ and σ_* are significantly different is shown in Figure 1(b).

In this paper, we study the problem of learning mixtures of *product distributions*. A product distribution over $\mathbf{R}^n$ is one in which each coordinate is distributed independently of any others. In practice, mixtures of product distributions have been used as mathematical models for data and learning mixtures of product distributions specifically has been studied [FM99, FOS05, FOS06, DHKS05] – see the Related Work section for examples and details. However, even under this seemingly restrictive assumption, providing an efficient algorithm that does better than the bounds of [AM05, KSV05] turns out to be quite challenging. The main challenge is to find a low-dimensional subspace that contains most of the separation between the centers; although the independence assumption can (sometimes) help us identify which coordinates contribute to the distance between some pair of centers, the problem of actually finding the low-dimensional space still requires more involved techniques.

In this paper, we present an algorithm for learning mixtures of product distributions, which is stable in the presence of skewed mixing weights, and, under certain conditions, in the presence of high variance outside the subspace containing the centers. In particular, the dependence of the separation required by our algorithm on skewed mixing weights is only logarithmic. Additionally, with arbitrarily small separation, (*i.e.,* even when the separation is not enough for classification), with enough samples, we can approximate the subspace containing the centers. Previous techniques failed to do so for non-spherical distributions regardless of the number of samples, unless the separation was sufficiently large. Our algorithm works for binary product distributions and axis-aligned Gaussians. We require that the distance between the centers be *spread* across $\Theta(T \log \Lambda)$ coordinates, where Λ depends polynomially on the maximum distance between centers and w_{min}. For our algorithm to classify the samples correctly, we further need the separation between centers to be $\Theta(\sigma\sqrt{T\log\Lambda})$.

In addition, if a stronger version of the spreading condition is satisfied, then our algorithm requires a separation of only $\Theta(\sigma_*\sqrt{T\log\Lambda})$ to ensure correct classification of the samples. The stronger spreading condition, discussed in more detail later, ensures that when we split the coordinates randomly into two sets, the maximum directional variance of any distribution in the mixture along the projection of the subspace containing the centers into the subspaces spanned by the coordinate vectors in each set, is comparable to σ_*^2.

In summary, compared to [AM05, KSV05], our algorithm is much (exponentially) less susceptible to the imbalance in mixture weights and, when the stronger spreading condition holds, to high variance noise outside the subspace containing the centers. However, our algorithm requires a spreading condition and coordinate-independence, while [AM05, KSV05] are more general. We note that for perfectly spherical distributions, the results of [VW02] are better than our results – however, these results do not apply even for distributions with bounded eccentricity. Finally unlike the results of [Das99, AK01, DS00], which require the separation to grow polynomially with dimension, our separation only grows logarithmically with the dimension.

Our algorithm is based upon two key insights. The first insight is that if the centers are separated along several coordinates, then many of these coordinates are *correlated* with each other. To exploit this observation, we choose half the coordinates randomly, and search the space of this half for directions of high variance. We use the remaining half of coordinates to *filter* the found directions. If a found direction separates the centers, it is likely to have some correlation with coordinates in the remaining half, and therefore is preserved by the filter. If, on the other hand, the direction found is due to noise, coordinate independence ensures that there will be no correlation with the second half of coordinates, and therefore such directions get filtered away.

The second insight is that the tasks of searching for and filtering the directions can be simultaneously accomplished via a singular value decomposition of the matrix of covariances between the two halves of coordinates.In particular, we show that the top few directions of maximum variance of the covariance matrix approximately capture the subspace containing the centers. Moreover, we show that the covariance matrix has low singular value along any noise direction. By combining these ideas, we obtain an algorithm that is almost insensitive to mixing weights, a property essential for applications like population stratification [CHRZ07], and which can be implemented using the heavily optimized and thus, efficient, SVD procedure, and which works with a separation condition closer to the information theoretic bound.

Related Work

The first provable results for learning mixtures of Gaussians are due to Dasgupta [Das99] who shows how to learn mixtures of spherical Gaussians with a separation of $\Theta(\sigma\sqrt{n})$ in an n-dimensional space. An EM based algorithm by Dasgupta and Schulman [DS00] was shown to apply to more situations, and with a separation of $\Theta(\sigma n^{1/4})$. Arora and Kannan [AK01] show how to learn mixtures of distributions of arbitrary Gaussians whose centers are separated by $\Theta(n^{1/4}\sigma)$. Their results apply to many other situations, for example, *concentric* Gaussians with sufficiently different variance.

The first result that removed the dependence on n in the separation requirement was that of Vempala and Wang [VW02] who use SVD to learn mixtures of spherical Gaussians with $O(\sigma T^{1/4})$ separation. They project to a subspace of dimension T using an SVD and use a distance based method in the low dimensional space. If the separation is not enough for classification, [VW02] can also find, given enough samples, a subspace approximating the subgspace containing the centers. While the results of [VW02] are independent of the imbalance on mixing weights, they apply only to perfectly spherical Gaussians, and cannot be extended to Gaussians with bounded eccentricity. In further work Kannan, Salmasian, and Vempala[KSV05] and Achlioptas and McSherry [AM05] show how to cluster general Gaussians using SVD. While these results are weaker than ours, they apply to a mixture of general Gaussians, axis-aligned or not. We note that their analysis also applies to binary product distributions again with polynomial dependence on the imbalance in mixing weights[1]. In contrast, our separation requirement is $\Omega(\sigma_*\sqrt{T\log\Lambda})$, *i.e.,* is logarithmically dependent on the mixing weights and dimension and the maximum variance in noise directions.

There is also ample literature on specifically learning mixtures of product distributions. Freund and Mansour [FM99] show an algorithm which generates distributions that are ϵ-close to a mixture of two product distributions over $\{0,1\}^n$ in time polynomial in n and $1/\epsilon$. Feldman, O'Donnell, and Servedio show how to generate distributions that are ϵ-close to a mixture of T product distributions [FOS05] and axis-aligned Gaussians [FOS06]. Like [FM99], they have no separation requirements, but their algorithm takes $n^{O(T^3)}$ time. Dasgupta *et. al* [DHKS05] provide an algorithm for learning mixtures of heavy-tailed product distributions which works with a separation of $\Theta(R\sqrt{T})$, where R is the maximum half-radius of any distribution in the mixture.

[1] They do not directly address binary product distributions in their paper, but their techniques apply.

While their separation requirement does not depend polynomially on $\frac{1}{w_{\min}}$, their algorithm runs in time exponential in $\Theta(\frac{n}{w_{\min}})$. They also require a slope, which is comparable to our spreading condition. Chaudhuri *et al.* [CHRZ07] show an iterative algorithm for learning mixtures of two product distributions that implicitly uses the notion of co-ordinate independence to filter out noise directions. However, the algorithm heavily uses the two distribution restriction to find the appropriate directions, and does not work when $T > 2$.

More broadly, the problem of analyzing mixture models data has received a great deal of attention in statistics, see for example, [MB88, TSM85], and has numerous applications. We present three applications where data is modelled as a mixture of product distirbutions. First, the problem of population stratification in population genetics has been posed as learning mixtures of binary product distributions in [SRH07]. In their work, the authors develop an MCMC method for addressing the problem and their software embodiment is widely used. A second application is in speech recognition [Rey95, PFK02], which models acoustic features at a specific time point as a mixture of axis-aligned Gaussians. A third application is the widely used Latent Dirichlet Allocation model [BNJ03]. Here, documents are modelled as distributions over topics which, in turn, are distributions over words. Subsequent choices of topics and words are assumed to be *independent*. (For words, this is referred to as the “bag of words” assumption.) [BNJ03] develops variational techniques that provide interesting results for various corpora. Interestingly, the same model was used by Kleinberg and Sandler [KS04] to model user preferences for purchasing goods (users correspond to documents, topics to categories, and words to goods). Their algorithm, which provides provably good performance in this model, also uses SVD-like clustering algorithms as a subroutine.

Our clustering method also involves a Canonical Correlations Analysis of the samples, which seems to have connections with multiview learning[KF07] and co-training[AT98].

Discussion

The Spreading Condition. The spreading condition loosely states that the distance between each pair of centers is spread along about $\Theta(T \log \Lambda)$ coordinates. We demonstrate by an example, that a spread of $\Omega(T)$, is a natural limit for all methods that use linear correlations between coordinates, such as our methods and SVD based methods [VW02, KSV05, AM05]. We present, as an example, two distributions : a mixture $\mathcal{D}_1$ of T binary product distributions, and a single binary product distribution $\mathcal{D}_2$, which have exactly the same covariance matrix. Our example is based on the Hadamard code, in which a codeword for a k-bit message is 2^k bits long, and includes a parity bit for each subset of the bits of the message. The distributions comprising $\mathcal{D}_1$ are defined as follows. Each of the $T = 2^k$ centers is a codeword for a $k-$bit string appended by a string of length $n - k$ in which each coordinate has value $1/2$. Notice that the last $n - k$ bits are noise. Thus, the centers are separated by $T/2$ coordinates. $\mathcal{D}_2$ is the uniform distribution over the $n-$dimensional hypercube. As there are no linear correlations between any two bits in the Hadamard code, the covariance of $\mathcal{D}_1$ along any two directions is 0, and each direction has the same variance. As this is also the case for $\mathcal{D}_2$, any SVD-bsed or correlation-based algorithm will fail to distinguish between the two mixtures. We also note that learning binary product distributions with minimum separation 2 and average separation $1 + \frac{1}{2}\log T$ would allow one to learn parities of $\log T$ variables with noise. Finally, we note that when the spreading condition fails, one has only a few coordinates that contain most of the distance between centers. One could enumerate the set of possible coordinates to deal with this case, and is exponentional in $T \log n \log \Lambda$. [FOS05] on the other hand takes time exponential in $T^3 \log n$, and works with no separation requirement.

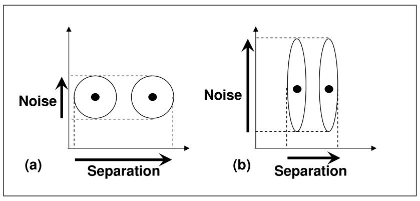

Figure 1: (a) Spherical Gaussians: Direction of maximum variance is the direction separating the centers (b) Arbitrary Gaussians: Direction of maximum variance is a noise direction.

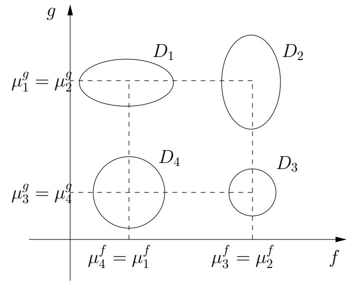

Figure 2: An Example where All Covariances are 0

2 A Summary of Our Results

We begin with some preliminary definitions about distributions drawn over n dimensional spaces. We use

$f, g, \ldots$ to range over coordinates, and $i, j, \ldots$ to range over distributions. For any $x \in \mathbf{R}^n$, we write x^f for the f-th coordinate of x. For any subspace $\mathcal{H}$ (resp. vector v), we use $\bar{\mathcal{H}}$ (resp. $\bar{v}$) to denote the orthogonal complement of $\mathcal{H}$ (resp. v). For a subspace $\mathcal{H}$ and a vector v, we write $\mathbf{P}_{\mathcal{H}}(v)$ for the projection of v onto the subspace $\mathcal{H}$. For any vector x, we use $||x||$ for the Euclidean norm of x. For any two vectors x and y, we use $\langle x, y \rangle$ for the dot-product of x and y.

Mixtures of Distributions. A *mixture of distributions* $\mathcal{D}$, is a collection of distributions, $\{D_1, \ldots, D_T\}$, over points in $\mathbf{R}^n$, and a set of mixing weights $w_1, \ldots, w_T$ such that $\sum_i w_i = 1$. In the sequel, n is assumed to be much larger than T. In a product distribution over $\mathbf{R}^n$, each coordinate is distributed independently of the others. When working with a mixture of binary product distributions, we assume that the f-th coordinate of a point drawn from distribution D_i is 1 with probability μ_i^f, and 0 with probability $1 - \mu_i^f$. When working with a mixture of axis-aligned Gaussian distributions, we assume that the f-th coordinate of a point drawn from distribution D_i is distributed as a Gaussian with mean μ_i^f and standard deviation σ_i^f.

Centers. We define the *center* of a distribution i as the vector μ_i, and the *center of mass of the mixture* as the vector $\bar{\mu}$ where $\bar{\mu}^f$ is the mean of the mixture for the coordinate f. We write $\mathcal{C}$ for the subspace containing $\mu_1, \ldots, \mu_T$.

Directional Variance. We define σ^2 as the maximum variance of any distribution in the mixture along any direction. We define σ_*^2 as the maximum variance of any distribution in the mixture along any direction in the subspace containing the centers of the distributions. We write $\sigma_{\max}^2$ as the maximum variance of the entire mixture in any direction. This may be more than σ^2 due to contribution from the separation between the centers.

Spread. We say that a unit vector v in $\mathbf{R}^n$ has spread $\mathcal{S}$ if $\sum_f (v^f)^2 \geq \mathcal{S} \cdot \max_f (v^f)^2$.

Distance. Given a subspace $\mathcal{K}$ of $\mathbf{R}^n$ and two points x, y in $\mathbf{R}^n$, we write $d_{\mathcal{K}}(x, y)$ for the square of the Euclidean distance between x and y projected along the subspace $\mathcal{K}$.

The Spreading Condition and Effective Distance. The spreading condition tells us that the distance between each μ_i and μ_j should not be concentrated along a few coordinates. One way to ensure this is to demand that for all i, j, the vector $\mu_i - \mu_j$ has high spread. This is comparable to the slope condition used in [DHKS05].

However, we do not need such a strong condition for dealing with mixtures with imbalanced mixing weights. Our *spreading condition* therefore demands that for each pair of centers μ_i, μ_j, the norm of the vector $\mu_i - \mu_j$ high, even if we ignore the contribution of the top few (about $T \log T$) coordinates. Due to technicalities in our proofs, the number of coordinates we can ignore needs to depend (logarithmically) on this distance.

We therefore define the spreading condition as follows. We define parameters c_{ij} and a parameter Λ as : $\Lambda > \frac{\sigma_{\max} T \log^2 n}{w_{\min} \cdot (\min_{i,j} c_{ij}^2)}$ and c_{ij} is the maximum value such that there are $49T \log \Lambda$ coordinates f with $|\mu_i^f - \mu_j^f| > c_{ij}$. We note that Λ is bounded by a polynomial in $T, \sigma_*, 1/w_{min}, 1/c_{ij}$ and logarithmic in n.

We define $c_{\min}$ to be the minimum over all pairs i, j of c_{ij}. Given a pair of centers i and j, let Δ_{ij} be the set of coordinates f such that $|\mu_i^f - \mu_j^f| > c_{ij}$, and let ν_{ij} be defined as: $\nu_{ij}^f = \mu_i^f - \mu_j^f$, if $f \notin \Delta_{ij}$, and $\nu_{ij}^f = c_{ij}$ otherwise. We define $\bar{d}(\mu_i, \mu_j)$, the effective distance between μ_i and μ_j to be the square of the L_2 norm of ν_{ij}. In contrast, the square of the norm of the vector $\mu_i - \mu_j$ is the actual distance between centers μ_i and μ_j, and is always greater than or equal to the effective distance between μ_i and μ_j. Moreover, given i and j and the subspace $\mathcal{K}$, we define $\bar{d}_{\mathcal{K}}(\mu_i, \mu_j)$ as the square of the norm of the vector ν_{ij} projected onto the subspace $\mathcal{K}$.

Under these definitions, our spreading condition now requires that $\bar{d}(\mu_i, \mu_j) \geq 49 c_{ij}^2 T \log \Lambda$ and our stronger spreading condition requires that every vector in $\mathcal{C}$ has spread $32T \log \frac{\sigma}{\sigma_*}$.

A Formal Statement of our Results. Our main contribution is Algorithm CORR-CLUSTER, a correlation based algorithm for learning mixtures of binary product distributions and axis-aligned Gaussians. The input to the algorithm is a set of samples from a mixture of distributions, and the output is a clustering of the samples.

The main component of Algorithm CORR-CLUSTER is Algorithm CORR-SUBSPACE, which, given samples from a mixture of distributions, computes an approximation to the subspace containing the centers of the distributions. The motivation for approximating the latter space is as follows. In the T-dimensional subspace containing the centers of the distributions, the distance between each pair of centers μ_i and μ_j is the same as their distance in $\mathbf{R}^n$; however, because of the low dimensionality, the magnitude of the noise is small. Therefore, provided the centers of the distributions are sufficiently separated, projection onto this subspace will sharply separate samples from different distributions.

SVD-based algorithms [VW02, AM05, KSV05] attempt to approximate this subspace by the top T singular vectors of the matrix of samples. However, for product distributions, our Algorithm CORR-SUBSPACE can approximate this subspace correctly under more restrictive separation conditions.

The properties of Algorithms CORR-SUBSPACE and

Corr-Cluster are formally summarized in Theorem 1 and Theorem 2 respectively.

Theorem 1 (Spanning centers) *Suppose we are given a mixture of distributions $\mathcal{D} = \{D_1, \ldots, D_T\}$, with mixing weights $w_1, \ldots, w_T$. Then with at least constant probability, the subspace $\mathcal{K}$ of dimension at most $2T$ output by Algorithm* Corr-Subspace *has the following properties.*

1. *If, for all i and j, $\bar{d}(\mu_i, \mu_j) \geq 49c_{ij}^2 T \log \Lambda$, then, for all pairs i, j,*
$$d_{\mathcal{K}}(\mu_i, \mu_j) \geq \frac{99}{100}(\bar{d}(\mu_i, \mu_j) - 49Tc_{ij}^2 \log \Lambda)$$
2. *If, in addition, every vector in $\mathcal{C}$ has spread $32T \log \frac{\sigma}{\sigma_*}$, then, with at least constant probability, the maximum directional variance in $\mathcal{K}$ of any distribution D_i in the mixture is at most $11\sigma_*^2$.*

The number of samples required by Algorithm Corr-Subspace *is polynomial in $\frac{\sigma}{\sigma_*}$, T, n,σ and $\frac{1}{w_{\min}}$, and the algorithm runs in time polynomial in n, T, and the number of samples.*

The subspace $\mathcal{K}$ computed by Algorithm Corr-Subspace approximates the subspace containing the centers of the distributions in the sense that the distance between each pair of centers μ_i and μ_j is high along $\mathcal{K}$. Theorem 1 states that Algorithm Corr-Subspace computes an approximation to the subspace containing the centers of the distributions, provided the spreading condition is satisfied. If the strong spreading condition is satisfied as well, then the maximum variance of each D_i along $\mathcal{K}$ is also close to σ_*^2.

Note that in Theorem 1, there is no absolute lower bound required on the distance between any pair of centers. This means that, so long as the spreading condition is satisfied, and there are sufficiently many samples, even if the distance between the centers is not large enough for correct classification, we can compute an approximation to the subspace containing the centers of the distributions. We also note that although we show that Algorithm Corr-Subspace succeeds with constant probability, we can make this probability higher at the expense of a more restrictive spreading condition, or by running the algorithm multiple times.

Theorem 2 (Clustering) *Suppose we are given a mixture of distributions $\mathcal{D} = \{D_1, \ldots, D_T\}$, with mixing weights $w_1, \ldots, w_T$. Then, Algorithm* Corr-Cluster *has the following properties.*

1. *If for all i and j, $\bar{d}(\mu_i, \mu_j) \geq 49Tc_{ij}^2 \log \Lambda$, and for all i, j we have:*
$$\bar{d}(\mu_i, \mu_j) > 59\sigma^2 T(\log \Lambda + \log n)$$
(for axis-aligned Gaussians)
$$\bar{d}(\mu_i, \mu_j) > 59T(\log \Lambda + \log n)$$
(for binary product distributions)
then with probability $1 - \frac{1}{n}$ over the samples and with constant probability over the random choices made by the algorithm, Algorithm Corr-Cluster *computes a correct clustering of the sample points.*
2. *For axis-aligned Gaussians, if every vector in $\mathcal{C}$ has spread at least $32T \log \frac{\sigma}{\sigma_*}$, and for all i, j:*
$$\bar{d}(\mu_i, \mu_j) \quad \geq \quad 150\sigma_*^2 T(\log \Lambda + \log n)$$
then, with constant probability over the randomness in the algorithm, and with probability $1 - \frac{1}{n}$ over the samples, Algorithm Corr-Cluster *computes a correct clustering of the sample points.*

Algorithm Corr-Cluster *runs in time polynomial in n and the number of samples required by Algorithm* Corr-Cluster *is polynomial in $\frac{\sigma}{\sigma_*}$, T, n, σ and $\frac{1}{w_{\min}}$.*

We note that because we are required to do classification here, we do require an absolute lower bound on the distance between each pair of centers in Theorem 2.

The second theorem follows from the first and the distance concentration Lemmas of [AM05] as described in detail in Chapter 3 of [Cha07]. The Lemmas show that once the points are projected onto the subspace computed in Theorem 1, a distance-based clustering method suffices to correctly cluster the points.

A Note on the Stronger Spreading Condition. The motivation for requiring the stronger spreading condition is as follows. Our algorithm splits the coordinates randomly into two sets $\mathcal{F}$ and $\mathcal{G}$. If $\mathcal{C}_{\mathcal{F}}$ and $\mathcal{C}_{\mathcal{G}}$ denote the restriction of $\mathcal{C}$ to the coordinates in $\mathcal{F}$ and $\mathcal{G}$ respectively, then our algorithm requires that the maximum directional variance of any distribution in the mixture is close to σ_* in $\mathcal{C}_{\mathcal{F}}$ and $\mathcal{C}_{\mathcal{G}}$ respectively. Notice that this does not follow from the fact that the maximum directional variance along $\mathcal{C}$ is σ_*^2: suppose $\mathcal{C}$ is spanned by $(0.1, 0.1, 1, 1)$ and $(0.1, 0.1, -1, 1)$, variances of D_1 along the axes are $(10, 10, 1, 1)$, and $\mathcal{F}$ is $\{1, 2\}$. Then, σ_*^2 is about 2.8, while the variance of D_1 along $\mathcal{C}_{\mathcal{F}}$ is 10. However, as Lemma 9 shows, the required condition is ensured by the strong spreading condition.

However, in general, the maximum directional variance of any D_i in the mixture along $\mathcal{C}_{\mathcal{F}}$ and $\mathcal{C}_{\mathcal{G}}$ may still be close to σ_*^2, even though strong spreading condition is far from being met. For example: if $\mathcal{C}$ is the space spanned by the first T coordinate vectors $e_1, \ldots, e_T$,then with probability $1 - \frac{1}{2^T}$, the maximum variance along $\mathcal{C}_{\mathcal{F}}$ and $\mathcal{C}_{\mathcal{G}}$ is also σ_*^2.

3 Algorithm Corr-Cluster

Our clustering algorithm follows the same basic framework as the SVD-based algorithms of [VW02, KSV05, AM05]. The input to the algorithm is a set S of samples, and the output is a pair of clusterings of the samples according to source distribution.

```
CORR-CLUSTER($S$)
1. Partition $S$ into $S_A$ and $S_B$ uniformly at random.
2. Compute: $\mathcal{K}_A = Corr - Subspace(S_A)$, $\mathcal{K}_B = Corr - Subspace(S_B)$
3. Project each point in $S_B$ (resp. $S_A$) on the subspace $\mathcal{K}_A$ (resp. $\mathcal{K}_B$).
4. Use a distance-based clustering algorithm [AK01] to partition the points in $S_A$ and $S_B$ after projection.
```

The first step in the algorithm is to use Algorithm CORR-SUBSPACE to find a $O(T)$-dimensional subspace $\mathcal{K}$ which is an approximation to the subspace containing the centers of the distributions. Next, the samples are projected onto $\mathcal{K}$ and a distance-based clustering algorithm is used to find the clusters.

We note that in order to preserve independence the samples we project onto $\mathcal{K}$ should be distinct from the ones we use to compute $\mathcal{K}$. A clustering of the complete set of points can then be computed by partitioning the samples into two sets A and B. We use A to compute $\mathcal{K}_A$, which is used to cluster B and vice-versa.

We now present our algorithm which computes a basis for the subspace $\mathcal{K}$. With slight abuse of notation we use $\mathcal{K}$ to denote the set of vectors that form the basis for the subspace $\mathcal{K}$.The input to CORR-SUBSPACE is a set S of samples, and the output is a subspace $\mathcal{K}$ of dimension at most $2T$.

Algorithm CORR-SUBSPACE**:**

Step 1: Initialize and Split Initialize the basis $\mathcal{K}$ with the empty set of vectors. Randomly partition the coordinates into two sets, $\mathcal{F}$ and $\mathcal{G}$, each of size $n/2$. Order the coordinates as those in $\mathcal{F}$ first, followed by those in $\mathcal{G}$.

Step 2: Sample Translate each sample point so that the center of mass of the set of sample points is at the origin. Let F (respectively G) be the matrix which contains a row for each sample point, and a column for each coordinate in $\mathcal{F}$ (respectively $\mathcal{G}$). For each matrix, the entry at row x, column f is the value of the f-th coordinate of the sample point x divided by $\sqrt{|S|}$.

Step 3: Compute Singular Space For the matrix $F^{\mathbf{T}}G$, compute $\{v_1, \ldots, v_T\}$, the top T left singular vectors, $\{y_1, \ldots, y_T\}$, the top T right singular vectors, and $\{\lambda_1, \ldots, \lambda_T\}$, the top T singular values.

Step 4: Expand Basis For each i, we abuse notation and use v_i (y_i respectively) to denote the vector obtained by concatenating v_i with the 0 vector in $n/2$ dimensions (0 vector in $n/2$ dimensions concatenated with y_i respectively). For each i, if the singular value λ_i is more than a threshold $\tau = O\left(\frac{w_{\min} c_{ij}^2}{T \log^2 n} \cdot \sqrt{\log \Lambda}\right)$, we add v_i and y_i to $\mathcal{K}$.

Step 5: Output Output the set of vectors $\mathcal{K}$.

The main idea behind our algorithm is to use half the coordinates to compute a subspace which approximates the subspace containing the centers, and the remaining half to validate that the subspace computed is indeed a good approximation. We critically use the coordinate independence property of product distributions to make this validation possible.

4 Analysis of Algorithm CORR-CLUSTER

This section is devoted to proving Theorems 1, and 2. We use the following notation.

Notation. We write $\mathcal{F}$-space (resp. $\mathcal{G}$-space) for the $n/2$ dimensional subspace of $\mathbf{R}^n$ spanned by the coordinate vectors $\{e_f \mid f \in \mathcal{F}\}$ (resp. $\{e_g \mid g \in \mathcal{G}\}$). We write $\mathcal{C}$ for the subspace spanned by the set of vectors μ_i. We write $\mathcal{C}_{\mathcal{F}}$ for the space spanned by the set of vectors $\mathbf{P}_{\mathcal{F}}(\mu_i)$. We write $\mathbf{P}_{\mathcal{F}}(\bar{\mathcal{C}}_{\mathcal{F}})$ for the orthogonal complement of $\mathcal{C}_{\mathcal{F}}$ in the $\mathcal{F}$-space. Moreover, we write $\mathcal{C}_{\mathcal{F} \cup \mathcal{G}}$ for the subspace of dimension $2T$ spanned by the union of a basis of $\mathcal{C}_{\mathcal{F}}$ and a basis of $\mathcal{C}_{\mathcal{G}}$. Next, we define a key ingredient of the analysis.

Covariance Matrix. Let N be a large number. We define $\hat{F}$ (resp. $\hat{G}$), the *perfect sample matrix* with respect to $\mathcal{F}$ (resp. $\mathcal{G}$) as the $N \times n/2$ matrix whose rows from $(w_1 + \ldots + w_{i-1})N + 1$ through $(w_1 + \ldots + w_i)N$ are equal to the vector $\mathbf{P}_{\mathcal{F}}(\mu_i)/\sqrt{N}$ (resp. $\mathbf{P}_{\mathcal{G}}(\mu_i)/\sqrt{N}$). For a coordinate f, let X_f be a random variable which is distributed as the f-th coordinate of the mixture $\mathcal{D}$. As the entry in row f and column g in the matrix $\hat{F}^{\mathbf{T}}\hat{G}$ is equal to $\mathbf{Cov}(X_f, X_g)$, the covariance of X_f and X_g, we call the matrix $\hat{F}^{\mathbf{T}}\hat{G}$ the *covariance matrix* of $\mathcal{F}$ and $\mathcal{G}$.

Proof Structure. The overall structure of our proof is as follows. First, we show that the centers of the distributions in the mixture have a high projection on the subspace of highest correlation between the coordinates. To do this, we first assume,in Section 4.1 that the input to the algorithm in Step 2 are the perfect sample matrices $\hat{F}$ and $\hat{G}$. Of course, we cannot directly feed in the matrices $\hat{F}, \hat{G}$, as the values of the centers are not known in advance. Next, we show in Section 4.2 that this holds even when the matrices F and G in Step 2 of Algorithm CORR-SUBSPACE are obtained by sampling. In Section 4.3, we combine these two results and prove Theorem 1. Finally, using results on distance concentration from [AM05, AK01], we complete the analysis by proving Theorem 2.

4.1 The Perfect Sample Matrix

The goal of this section is to prove Lemmas 3 and 7, which establish a relationship between directions of high correlation of the covariance matrix constructed from the perfect sample matrix, and directions which contain a lot of separation between centers. Lemma 3 shows that a direction which contains a lot of effective distance between some pair of centers, is also a direction of high correlation.

Lemma 7 shows that a direction $v \in \mathbf{P}_{\mathcal{F}}(\bar{\mathcal{C}}_{\mathcal{F}})$, which is perpendicular to the space containing the centers, is a direction with 0 correlation. In addition, we show in Lemma 8, another property of the perfect sample matrix – the covariance matrix constructed from the perfect sample matrix has rank at most T. We conclude this section by showing in Lemma 9 that when every vector in $\mathcal{C}$ has high spread, the directional variance of any distribution in the mixture along $\mathcal{F}$-space or $\mathcal{G}$-space is of the order of σ_*^2.

We begin by showing that if a direction v contains a lot of the distance between the centers, then, for most ways of splitting the coordinates, the magnitude of the covariance of the mixture along the projection of v on $\mathcal{F}$-space and the projection of v $\mathcal{G}$-space is high. In other words, the projections of v along $\mathcal{F}$-space and $\mathcal{G}$-space are directions of high correlation.

Lemma 3 *Let v be any vector in $\mathcal{C}_{\mathcal{F}\cup\mathcal{G}}$ such that for some i and j, $\bar{d}_v(\mu_i, \mu_j) \geq 49Tc_{ij}^2 \log \Lambda$. If $v_{\mathcal{F}}$ and $v_{\mathcal{G}}$ are the normalized projections of v to $\mathcal{F}$-space and $\mathcal{G}$-space respectively, then, with probability at least $1 - \frac{1}{T}$ over the splitting step, for all such v, $v_{\mathcal{F}}^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} v_{\mathcal{G}} \geq \tau$ where $\tau = O\left(\frac{w_{\min} c_{ij}^2}{T \log^2 n} \cdot \sqrt{\log \Lambda}\right)$.*

A detailed proof, presented in [Cha07], is omitted due to lack of space. However, the main ingredient of the proof is Lemma 4.

Lemma 4 *Let v be a fixed vector in $\mathcal{C}$ such that for some i and j, $\bar{d}_v(\mu_i, \mu_j) \geq 49Tc_{ij}^2 \log \Lambda$. If $v_{\mathcal{F}}$ and $v_{\mathcal{G}}$ are the projections of v to $\mathcal{F}$-space and $\mathcal{G}$-space respectively, then, with probability at least $1 - \Lambda^{-2T}$ over the splitting step, $v_{\mathcal{F}}^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} v_{\mathcal{G}} \geq 2\tau$ where $\tau = O\left(\frac{w_{\min} c_{ij}^2}{T \log^2 n} \cdot \sqrt{\log \Lambda}\right)$.*

Let $\hat{F}_v$ ($\hat{G}_v$ respectively) be the $s \times n/2$ matrix obtained by projecting each row of $\hat{F}$ (respectively $\hat{G}$) on $v_{\mathcal{F}}$ (respectively $v_{\mathcal{G}}$). Then,

$$\begin{aligned} & v_{\mathcal{F}}^{\mathbf{T}} \hat{F}_v^{\mathbf{T}} \hat{G}_v v_{\mathcal{G}} \\ = & \sum_i w_i \langle v_{\mathcal{F}}, \mathbf{P}_{v_{\mathcal{F}}}(\mu_i - \bar{\mu}) \rangle \langle v_{\mathcal{G}}, \mathbf{P}_{v_{\mathcal{G}}}(\mu_i - \bar{\mu}) \rangle \\ = & v_{\mathcal{F}}^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} v_{\mathcal{G}} \end{aligned}$$

Moreover, for any pair of vectors x in $\mathcal{F}$-space and y in $\mathcal{G}$-space such that $\langle x, v_{\mathcal{F}} \rangle = 0$ and $\langle y, v_{\mathcal{G}} \rangle = 0$,

$$x^{\mathbf{T}} \hat{F}_v^{\mathbf{T}} \hat{G}_v y = \sum_i w_i \langle x, \mathbf{P}_{v_{\mathcal{F}}}(\mu_i - \bar{\mu}) \rangle \langle y, \mathbf{P}_{v_{\mathcal{G}}}(\mu_i - \bar{\mu}) \rangle = 0$$

Therefore, $\hat{F}_v^{\mathbf{T}} \hat{G}_v$ has rank at most 1.

The proof strategy for Lemma 4 is to show that if $d_v(\mu_i, \mu_j)$ is large then the matrix $\hat{F}_v^{\mathbf{T}} \hat{G}_v$ has high norm. We require the following notation. For each coordinate f we define a T-dimensional vector z_f as

$$z_f = [\sqrt{w_1}\mathbf{P}_v(\mu_1^f - \bar{\mu}^f), \ldots, \sqrt{w_T}\mathbf{P}_v(\mu_T^f - \bar{\mu}^f)]$$

Notice that for any two coordinates f,g:

$$\langle z_f, z_g \rangle = \mathbf{Cov}(\mathbf{P}_v(X_f), \mathbf{P}_v(X_g))$$

, computed over the entire mixture. We also observe that

$$\sum_f ||z_f||^2 = \sum_i w_i \cdot d_v(\mu_i, \bar{\mu})$$

The RHS of this equality is the weighted sum of the squares of the Euclidean distances between the centers of the distributions and the center of mass. By the triangle inequality, this quantity is at least $49 w_{\min} c_{ij}^2 T \log \Lambda$. We also a couple of technical lemmas – Lemmas 5 and 6, which are stated below. The proofs of these lemmas are omitted due to lack of space, but can be found in [Cha07].

Lemma 5 *Let A be a set of coordinates with cardinality more than $144T^2 \log \Lambda$ such that for each $f \in A$, $||z_f||$ is equal and $\sum_{f \in A} ||z_f||^2 = D$. Then, (1)*

$$\sum_{f,g \in A, f \neq g} \langle z_f, z_g \rangle^2 \geq \frac{D^2}{288T^2 \log \Lambda}$$

and (2) with probability $1 - \Lambda^{-2T}$ over the splitting of coordinates in Step 1,

$$\sum_{f \in \mathcal{F} \cap A, g \in \mathcal{G} \cap A} \langle z_f, z_g \rangle^2 \geq \frac{D^2}{1152T^2 \log \Lambda}$$

Lemma 6 *Let A be a set of coordinates such that for each $f \in A$, $||z_f||$ is equal and $\sum_{f \in A} ||z_f||^2 = D$. If $48T \log \Lambda + T < |A| \leq 144T^2 \log \Lambda$, then (1)*

$$\sum_{f,g \in A, f \neq g} \langle z_f, z_g \rangle^2 \geq \frac{D^2}{1152T^4 \log \Lambda}$$

and (2) with probability $1 - \Lambda^{-2T}$ over the splitting in Step 1,

$$\sum_{f \in \mathcal{F} \cap A, g \in \mathcal{G} \cap A} \langle z_f, z_g \rangle^2 \geq \frac{D^2}{4608T^4 \log \Lambda}$$

Proof:(Of Lemma 4) From the definition of effective distance, if the condition: $\bar{d}_v(\mu_i, \mu_j) > 49c_{ij}^2 T \log \Lambda$ holds then there are at least $49T \log \Lambda$ vectors z_f with total squared norm at least $98w_{\min}{c_{ij}}^2 T \log \Lambda$. In the sequel we will scale down each vector z_f with norm greater than $c_{ij}\sqrt{w_{\min}}$ so that its norm is exactly $c_{ij}\sqrt{w_{\min}}$. We divide the vectors into $\log n$ groups as follows: group B_k contains vectors which have norm between $\frac{c_{ij}\sqrt{w_{\min}}}{2^k}$ and $\frac{c_{ij}\sqrt{w_{\min}}}{2^{k-1}}$.

We will call a vector *small* if its norm is less than $\frac{\sqrt{w_{\min}c_{ij}}}{2\sqrt{\log n}}$, and otherwise, we call the vector *big*. We observe that there exists a set of vector B with the following properties: (1) the cardinality of B is more than $49T \log \Lambda$, (2) the total sum of squares of the norm of the vectors in B is greater than $\frac{49T \log \Lambda w_{\min} c_{ij}^2}{\log n}$, and, (3) the ratio of the norms of any two vectors in B is at most $2\sqrt{\log n}$.

Case 1: Suppose there exists a group B_k of small vectors the squares of whose norms sum to a value greater than $\frac{49T w_{\min} c_{ij}^2 \log \Lambda}{\log n}$. By definition, such a group has more than $49T \log \Lambda$ vectors, and the ratio is at most 2.

Case 2: Otherwise, there are at least $49T \log \Lambda$ big vectors. By definition, the sum of the squares of their norms exceeds $\frac{49T w_{\min} c_{ij}^2 \log \Lambda}{\log n}$. Due to the scaling, the ratio is at most $2\sqrt{\log n}$.

We scale down the vectors in B so that each vector has squared norm $\frac{w_{\min} c_{ij}^2}{2^k}$ in case 1, and, squared norm $\frac{w_{\min} c_{ij}^2}{4 \log n}$ in case 2. Due to (2) and (3), the total squared norm of the scaled vectors is at least $\frac{49T w_{\min} c_{ij}^2 \log \Lambda}{4 \log^2 n}$.

Due to (1), we can now apply Lemmas 5 and 6 on the vectors to conclude that for some constant a_1, with probability $1 - \Lambda^{-2T}$,

$$\sum_{f \in \mathcal{F}, g \in \mathcal{G}} \langle z_f, z_g \rangle^2 \geq a_1 \cdot \left(\frac{w_{\min}^2 c_{ij}^4 \log \Lambda}{T^2 \log^4 n} \right)$$

The above sum is the square of the Frobenius norm $|\hat{F}_v^{\mathbf{T}} \hat{G}_v|_{\mathbf{F}}$ of the matrix $\hat{F}_v^{\mathbf{T}} \hat{G}_v$. Since $\hat{F}_v^{\mathbf{T}} \hat{G}_v$ has rank at most 1, and the maximum singular value of a rank 1 matrix is its Frobenius norm [GL96], plugging in $\tau = O\left(\frac{w_{\min} c_{ij}^2}{T \log^2 n} \cdot \sqrt{\log \Lambda} \right)$ completes the proof. □

Next we show that a vector $x \in \mathbf{P}_{\mathcal{F}}(\bar{\mathcal{C}}_{\mathcal{F}})$ is a direction of 0 correlation. A similar statement holds for a vector $y \in \mathbf{P}_{\mathcal{G}}(\bar{\mathcal{C}}_{\mathcal{G}})$.

Lemma 7 *If at Step 2 of Algorithm* CORR-SUBSPACE, *the values of F and G are respectively $\hat{F}$ and $\hat{G}$, and for some k,the top k-th left singular vector is v_k and the corresponding singular value λ_k is more than τ, then for any vector x in $\mathbf{P}_{\mathcal{F}}(\bar{\mathcal{C}}_{\mathcal{F}})$, $\langle v_k, x \rangle = 0$.*

Proof: We first show that for any x in $\mathbf{P}_{\mathcal{F}}(\bar{\mathcal{C}}_{\mathcal{F}})$, and any y, $x^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} y = 0$.

$$x^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} y = \sum_{i=1}^{T} w_i \langle \mathbf{P}_{\mathcal{F}}(\mu_i), x \rangle \cdot \langle \mathbf{P}_{\mathcal{G}}(\mu_i), y \rangle$$

Since x is in $\mathbf{P}_{\mathcal{F}}(\bar{\mathcal{C}}_{\mathcal{F}})$, $\langle \mathbf{P}_{\mathcal{F}}(\mu_i), x \rangle = 0$, for all i, and hence $x^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} y = 0$ for all x in $\mathbf{P}_{\mathcal{F}}(\bar{\mathcal{C}}_{\mathcal{F}})$. We now prove the Lemma by induction on k.

Base case ($k = 1$). Let $v_1 = u_1 + x_1$, where $u_1 \in \mathcal{C}_{\mathcal{F}}$ and $x_1 \in \mathbf{P}_{\mathcal{F}}(\bar{\mathcal{C}}_{\mathcal{F}})$. Let y_1 be the top right singular vector of $\hat{F}^{\mathbf{T}} \hat{G}$, and let $|x_1| > 0$. Then, $v_1^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} y_1 = u_1^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} y_1$, and $u_1/|u_1|$ is a vector of norm 1 such that $\frac{1}{|u_1|} u_1^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} y_1 > v_1^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} y_1$, which contradicts the fact that v_1 is the top left singular vector of $\hat{F}^{\mathbf{T}} \hat{G}$.

Inductive case. Let $v_k = u_k + x_k$, where $u_k \in \mathcal{C}_{\mathcal{F}}$ and $x_k \in \mathbf{P}_{\mathcal{F}}(\bar{\mathcal{C}}_{\mathcal{F}})$. Let y_k be the top k-th right singular vector of $\hat{F}^{\mathbf{T}} \hat{G}$, and let $|x_k| > 0$. We first show that u_k is orthogonal to each of the vectors $v_1, \ldots, v_{k-1}$. Otherwise, suppose there is some j, $1 \leq j \leq k - 1$, such that $\langle u_k, v_j \rangle \neq 0$. Then, $\langle v_k, v_j \rangle = \langle x_k, v_j \rangle + \langle u_k, v_j \rangle = \langle u_k, v_j \rangle \neq 0$. This contradicts the fact that v_k is a left singular vector of $\hat{F}^{\mathbf{T}} \hat{G}$. Therefore, $v_k^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} y_k = u_k^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} y_k$, and $u_k/|u_k|$ is a vector of norm 1, orthogonal to $v_1, \ldots, v_{k-1}$ such that $\frac{1}{|u_k|} u_k^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} y_k > v_k^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} y_k$. This contradicts the fact that v_k is the top k-th left singular vector of $\hat{F}^{\mathbf{T}} \hat{G}$. The Lemma follows. □

Lemma 8 *The covariance matrix $\hat{F}^{\mathbf{T}} \hat{G}$ has rank at most T.*

The proof is omitted due to space constraints.

Finally, we show that if the spread of every vector in $\mathcal{C}$ is high, then with high probability over the splitting of coordinates in Step 1 of Algorithm CORR-SUBSPACE, the maximum directional variances of any distribution D_i in $\mathcal{C}_{\mathcal{F}}$ and $\mathcal{C}_{\mathcal{G}}$ are high. This means that there is enough information in both $\mathcal{F}$-space and $\mathcal{G}$-space for correctly clustering the distributions through distance concentration.

Lemma 9 *If every vector $v \in \mathcal{C}$ has spread at least $32T \log \frac{\sigma}{\sigma_*}$, then, with constant probability over the splitting of coordinates in Step 1 of Algorithm* CORR-SUBSPACE, *the maximum variance along any direction in $\mathcal{C}_{\mathcal{F}}$ or $\mathcal{C}_{\mathcal{G}}$ is at most $5\sigma_*^2$.*

Proof:(Of Lemma 9) Let v and v' be two unit vectors in $\mathcal{C}$, and let $v_{\mathcal{F}}$ (resp. $v'_{\mathcal{F}}$) and $v_{\mathcal{G}}$ (resp. $v'_{\mathcal{G}}$ denote the normalized projections of v (resp. v') on $\mathcal{F}$-space and $\mathcal{G}$-space respectively. If $||v_{\mathcal{F}} - v'_{\mathcal{F}}|| < \frac{\sigma_*}{\sigma}$, then, the directional variance of any D_i in the mixture along $v'_{\mathcal{F}}$ can be written as:

$$\begin{aligned} & \mathbf{E}[\langle v'_{\mathcal{F}}, x - \mathbf{E}[x] \rangle^2] \\ = & \ \mathbf{E}[\langle v_{\mathcal{F}}, x - \mathbf{E}[x] \rangle^2] + \mathbf{E}[\langle v'_{\mathcal{F}} - v_{\mathcal{F}}, x - \mathbf{E}[x] \rangle^2] \\ & +2\mathbf{E}[\langle v_{\mathcal{F}}, x - \mathbf{E}[x] \rangle] \mathbf{E}[\langle v'_{\mathcal{F}} - v_{\mathcal{F}}, x - \mathbf{E}[x] \rangle] \\ \leq & \ \mathbf{E}[\langle v_{\mathcal{F}}, x - \mathbf{E}[x] \rangle^2] + ||v_{\mathcal{F}} - v'_{\mathcal{F}}||^2 \sigma^2 \end{aligned}$$

Thus, the directional variance of any distribution in the mixture along v' is at most the directional variance along v, plus an additional σ_*^2. Therefore, to show this lemma, we need to show that if v is any vector on a $\frac{\sigma_*}{\sigma}$-cover of $\mathcal{C}$, then with high probability over the splitting of coordinates in Step 1 of Algorithm CORR-SUBSPACE, the directional variances of any D_i in the mixture along $v_{\mathcal{F}}$ and $v_{\mathcal{G}}$ are at most $4\sigma_*^2$.

We show this in two steps. First we show that for any v in a $\frac{\sigma_*}{\sigma}$-cover of $\mathcal{C}$, $\frac{1}{4} \leq \sum_{f \in \mathcal{F}} (v^f)^2 \leq \frac{3}{4}$. Then, we show that this condition means that for this vector v, the maximum directional variances along $v_{\mathcal{F}}$ and $v_{\mathcal{G}}$ are at most $4\sigma_*^2$.

Let v be any fixed unit vector in $\mathcal{C}$. We first show that with probability $1 - \left(\frac{\sigma_*}{\sigma}\right)^{2T}$ over the splitting of coordinates in Step 1 of Algorithm CORR-SUBSPACE, $\frac{1}{4} \leq \sum_{f \in \mathcal{F}} (v^f)^2 \leq \frac{3}{4}$. To show this bound, we apply the Method of Bounded Difference[PD05]. Since we split the coordinates into $\mathcal{F}$ and $\mathcal{G}$ uniformly at random, $\mathbf{E}[\sum_{f \in \mathcal{F}} (v^f)^2] = \frac{1}{2}$. Let γ_f be the change in $\sum_{f \in \mathcal{F}} (v^f)^2$ when the inclusion or exclusion of coordinate f in the set $\mathcal{F}$ changes. Then, $\gamma_f = (v^f)^2$ and $\gamma = \sum_f \gamma_f^2$. Since the spread of vector v is at least $32T \log \frac{\sigma}{\sigma_*}$, $\gamma = \sum_f (v^f)^4 \leq \frac{1}{32T \log \frac{\sigma}{\sigma_*}}$, and from the Method of Bounded Differences,

$$\begin{aligned} \Pr[| \sum_{f \in \mathcal{F}} (v^f)^2 - \mathbf{E}[\sum_{f \in \mathcal{F}} (v^f)^2]| > \frac{1}{4}] & \leq e^{-1/32\gamma} \\ & \leq \left(\frac{\sigma_*}{\sigma}\right)^{2T} \end{aligned}$$

By taking an union bound over all v on a $\frac{\sigma*}{\sigma}$-cover of $\mathcal{C}$, we deduce that for any such v, $\frac{1}{4} \leq \sum_{f \in \mathcal{F}} (v^f)^2 \leq \frac{3}{4}$.

Since the maximum directional variance of any distribution D_i in the mixture in $\mathcal{C}$ is at most σ_*^2, $\sum_f (v^f)^2 (\sigma_i^f)^2 \leq \sigma_*^2$. Therefore the maximum variance along $v_{\mathcal{F}}$ as well as $v_{\mathcal{G}}$ can be computed as:

$$\frac{1}{||v_{\mathcal{F}}||^2} \sum_{f \in \mathcal{F}} (v^f)^2 (\sigma_i^f)^2 \leq \frac{1}{||v_{\mathcal{F}}||^2} \sum_f (v^f)^2 (\sigma_i^f)^2 \leq 4\sigma_*^2$$

The lemma follows. □

4.2 Working with Real Samples

In this section, we show that given sufficient samples, the properties of the matrix $F^{\mathbf{T}}G$, where F and G are generated by sampling in Step 2 of Algorithm CORR-CLUSTER are very close to the properties of the matrix $\hat{F}^{\mathbf{T}}\hat{G}$. The lemmas are stated below. The proofs are omitted due to space constraints, but can be found in [Cha07]. The proofs use the Method of Bounded Differences (when the input is a mixture of binary product distributions) and the Gaussian Concentration of Measure Inequality (for axis-aligned Gaussians).

The central lemma of this section is Lemma 10, which shows that, if there are sufficiently many samples, for any set of $2m$ vectors, $\{v_1, \ldots, v_m\}$ and $\{y_1, \ldots, y_m\}$, $\sum_k v_k^{\mathbf{T}} F^{\mathbf{T}} G y_k$ and $\sum_k v_k^{\mathbf{T}} \hat{F}^{\mathbf{T}} \hat{G} y_k$ are very close. This lemma is then used to prove Lemmas 11 and 12. Lemma 11 shows that the top few singular vectors of $F^{\mathbf{T}}G$ output by Algorithm CORR-SUBSPACE have very low projection on $\mathbf{P}_{\mathcal{F}}(\bar{\mathcal{C}}_{\mathcal{F}})$ or $\mathbf{P}_{\mathcal{G}}(\bar{\mathcal{C}}_{\mathcal{G}})$. Lemma 12 shows that the rank of the matrix $F^{\mathbf{T}}G$ is almost T, in the sense that the $T + 1$-th singular value of this matrix is very low.

Lemma 10 *Let $U = \{u_1, \ldots, u_m\}$, $Y = \{y_1, \ldots, y_m\}$ be any two sets of orthonormal vectors, and let F and G be the matrices generated by sampling in Step 2 of the algorithm. If the number of samples $|S|$ is greater than $\Omega(\frac{m^3 n^2 \log n \log(\sigma_{\max}/\delta)}{\delta^2})$ (for Binary Product Distributions), and $\Omega(\max(a_1, a_2))$ (for Axis-Aligned Gaussians), where $a_1 = \frac{\sigma^4 m^4 n^2 \log^2 n \log^2(\sigma_{\max}/\delta)}{\delta^2}$, and $a_2 = \frac{\sigma^2 \sigma_{\max}^2 m^3 n \log n \log(\sigma_{\max}/\delta)}{\delta^2}$, then, with probability at least $1 - 1/n$,*

$$|\sum_k u_k^{\mathbf{T}} (F^{\mathbf{T}} G - \mathbf{E}[F^{\mathbf{T}} G]) y_k| \leq \delta$$

Lemma 11 *Let F and G be the matrices generated by sampling in Step 2 of the algorithm, and let $v_1, \ldots, v_m$ be the vectors output by the algorithm in Step 4. If the number of samples $|S|$ is greater than $\Omega(\frac{m^3 n^2 \log n (\log \Lambda + \log \frac{1}{\epsilon})}{\tau^2 \epsilon^4})$ (for Binary Product Distributions), and $\max(a_1, a_2)$ (for Axis-Aligned Gaussians) where $a_1 = \frac{\sigma^4 m^4 n^2 \log^2 n \log^2(\Lambda/\epsilon)}{\tau^2 \epsilon^4}$, and $a_2 = \frac{\sigma^2 \sigma_{\max}^2 m^3 n \log n \log(\Lambda/\epsilon)}{\tau^2 \epsilon^4}$, then, for each k, and any x in $\mathbf{P}_{\mathcal{F}}(\bar{\mathcal{C}}_{\mathcal{F}})$, $\langle v_k, x \rangle \leq \epsilon$.*

Lemma 12 *Let F and G be the matrices generated by sampling in Step 2 of Algorithm* CORR-SUBSPACE*. If the number of samples $|S|$ is greater than $\Omega\left(\frac{T^3 n^2 \log n \log \Lambda}{\tau^2}\right)$ (for binary product distributions) and $\Omega\left(\max\left(\frac{\sigma^4 T^4 n^2 \log^2 \log \Lambda}{\tau^2}, \frac{\sigma_{\max}^2 \sigma^2 T^3 n \log n \log \Lambda}{\tau^2}\right)\right)$ for axis-aligned Gaussians, then, λ_{T+1}, the $T + 1$-th singular value of the matrix $F^{\mathbf{T}}G$ is at most $\tau/8$.*

4.3 The Combined Analysis

In this section, we combine the lemmas proved in Sections 4.1 and 4.2 to prove Theorem 1.

We begin with a lemma which shows that if every vector in $\mathcal{C}$ has spread $32T \log \frac{\sigma}{\sigma_*}$, then the maximum directional variance in $\mathcal{K}$, the space output by Algorithm CORR-SUBSPACE, is at most $11\sigma_*^2$.

Lemma 13 *Let $\mathcal{K}$ be the subspace output by the algorithm, and let v be any vector in $\mathcal{K}$. If every vector in $\mathcal{C}$ has spread $32T \log \frac{\sigma}{\sigma_*}$, and the number of samples $|S|$ is greater than $\Omega\left(\max\left(\frac{\sigma^6 T^4 n^2 \log^2 \log \Lambda}{\tau^2 \sigma_*^4}, \frac{\sigma_{\max}^2 \sigma^4 T^3 n \log n \log \Lambda}{\tau^2 \sigma_*^4}\right)\right)$ then for any i the maximum variance of D_i along v is at most $11\sigma_*^2$.*

The proof is omitted due to space constraints, and can be found in [Cha07].

The above Lemmas are now combined to prove Theorem 1.

Proof:(Of Theorem 1)

Suppose $\mathcal{K} = \mathcal{K}_L \cup \mathcal{K}_R$, where $\mathcal{K}_L = \{v_1, \ldots, v_m\}$, the top m left singular vectors of $F^{\mathbf{T}}G$ and $\mathcal{K}_R = \{y_1, \ldots, y_m\}$ are the corresponding right singular vectors. We abuse notation and use v_k to denote the vector v_k concatenated with a vector consisting of $n/2$ zeros, and use y_k to denote the vector consisting of $n/2$ zeros concatenated with y_k. Moreover, we use $\mathcal{K}$, $\mathcal{K}_L$, and $\mathcal{K}_R$ interchangeably to denote sets of vectors and the subspace spanned by those sets of vectors.

We show that with probability at least $1 - \frac{1}{T}$ over the splitting step, there exists no vector $v \in \mathcal{C}_{\mathcal{F}\cup\mathcal{G}}$ such that (1) v is orthogonal to the space spanned by the vectors $\mathcal{K}$ and (2) there exists some pair of centers i and j such that $\bar{d}_v(\mu_i, \mu_j) > 49Tc_{ij}^2 \log \Lambda$. For contradiction, suppose there exists such a vector v.

Then, if $v_{\mathcal{F}}$ and $v_{\mathcal{G}}$ denote the normalized projections of v onto $\mathcal{F}$-space and $\mathcal{G}$-space respectively, from Lemma 3, $v_{\mathcal{F}}^{\mathbf{T}} \hat{F}^{\mathbf{T}} G v_{\mathcal{G}} \geq \tau$ with probability at least $1 - \frac{1}{T}$ over the splitting step. From Lemma 10, if the number of samples $|S|$ is greater than $\Omega\left(\frac{T^3 n^2 \log n \log \Lambda}{\tau^2}\right)$ for binary product distributions, and if $|S|$ is greater than $\Omega\left(\max\left(\frac{\sigma^4 n^2 \log^2 \log \Lambda}{\tau^2}, \frac{\sigma^2 \sigma_{\max}^2 n \log n \log \Lambda}{\tau^2}\right)\right)$ for axis-aligned Gaussians, $v_{\mathcal{F}}^{\mathbf{T}} F^{\mathbf{T}} G v_{\mathcal{G}} \geq \frac{\tau}{2}$ with at least constant probability. Since v is orthogonal to the space spanned by $\mathcal{K}$, $v_{\mathcal{F}}$ is orthogonal to $\mathcal{K}_L$ and $v_{\mathcal{G}}$ is orthogonal to $\mathcal{K}_R$. As λ_{m+1} is the maximum value of $x^{\mathbf{T}} F^{\mathbf{T}} G y$ over all vectors x orthogonal to $\mathcal{K}_L$ and y orthogonal to $\mathcal{K}_R$, $\lambda_{m+1} \geq \frac{\tau}{2}$, which is a contradiction. Moreover, from Lemma 12, $\lambda_{T+1} < \frac{\tau}{8}$, and hence $m \leq T$.

Let us construct an orthonormal series of vectors $v_1, \ldots, v_m, \ldots$ which are *almost* in $\mathcal{C}_{\mathcal{F}}$ as follows. $v_1, \ldots, v_m$ are the vectors output by Algorithm CORR-SUBSPACE. We inductively define v_l as follows. Suppose for each k, $v_k = u_k + x_k$, where $u_k \in \mathcal{C}_{\mathcal{F}}$ and $x_k \in \mathbf{P}_{\mathcal{F}}(\bar{\mathcal{C}}_{\mathcal{F}})$. Let u_l be a unit vector in $\mathcal{C}_{\mathcal{F}}$ which is perpendicular to $u_1, \ldots, u_{l-1}$. Then, $v_l = u_l$. By definition, this vector is orthogonal to $u_1, \ldots, u_{l-1}$. In addition, for any $k \neq l$, $\langle v_l, v_k \rangle = \langle u_l, u_k \rangle + \langle u_l, x_k \rangle = 0$, and v_l is also orthogonal to $v_1, \ldots, v_{l-1}$. Moreover, if $\epsilon < \frac{1}{100T}$, $u_1, \ldots, u_m$ are linearly independent, and we can always find $\dim(\mathcal{C}_{\mathcal{F}})$ such vectors. Similarly, we construct a set of vectors $y_1, y_2, \ldots$. Let us call the combined set of vectors $\mathcal{C}^*$.

We now show that if there are sufficient samples, $d_{\bar{\mathcal{C}}^*}(\mu_i, \mu_j) \leq c_{ij}^2$. Note that for any unit vector v^* in $\mathcal{C}^*$, and any unit $x \in \bar{C}_{\mathcal{F}\cup\mathcal{G}}$, $\langle v, x \rangle \leq m\epsilon$. Also, note that for any u_k and u_l, $k \neq l$, $|\langle u_k, u_l \rangle| \leq \epsilon^2$, and $||u_k||^2 \geq 1 - \epsilon^2$. Let $v = \sum_k \alpha_k u_k$ be any unit vector in $\mathcal{C}_{\mathcal{F}\cup\mathcal{G}}$. Then, $1 = ||v||^2 = \sum_{k,k'} \alpha_k \alpha_{k'} \langle u_k, u_{k'} \rangle \geq \sum_k \alpha_k^2 ||u_k||^2 - \Omega(T^2 \epsilon^2)$.

The projection of v on C^* can be written as:

$$\begin{aligned} & \sum_k \langle v, v_k \rangle^2 = \sum_k \langle v, u_k \rangle^2 \\ = \quad & \sum_k \sum_l \alpha_l^2 \langle u_k, u_l \rangle^2 + 2 \sum_{l,l'} \alpha_l \alpha_{l'} \langle u_k, u_l \rangle \langle u_k, u_{l'} \rangle \\ \geq \quad & \sum_k \alpha_k^2 ||u_k||^4 - T^3 \epsilon^4 \geq 1 - \Omega(T^2 \epsilon^2) \end{aligned}$$

The last step follows because for each k, $||u_k||^2 \geq 1 - \epsilon^2$. If the number of samples $|S|$ is greater than $\Omega(\frac{m^3 n^2 \log n (\log \Lambda + \log 100T)}{\tau^2 T^4})$ (for Binary Product Distributions), and $\max\left(\frac{\sigma^4 m^4 n^2 \log^2 n \log^2(100T\Lambda)}{\tau^2 T^4}, \frac{\sigma_{\max}^2 \sigma^2 m^3 n \log\log(100T\Lambda)}{\tau^2 T^4}\right)$ (for axis-aligned Gaussians), then, $\epsilon < 1/100T$. Therefore,

$$d_{\bar{\mathcal{C}}^*}(\mu_i, \mu_j) \leq \frac{1}{100} d(\mu_i, \mu_j)$$

For any i and j,

$$d(\mu_i, \mu_j) = d_{\mathcal{K}}(\mu_i, \mu_j) + d_{\mathcal{C}^* \setminus \mathcal{K}}(\mu_i, \mu_j) + d_{\bar{\mathcal{C}}^*}(\mu_i, \mu_j)$$

Since vectors $v_{m+1}, \ldots$ and $y_{m+1}, \ldots$, all belong to $\mathcal{C}_{\mathcal{F}\cup\mathcal{G}}$ (as well as $\mathcal{C}^* \setminus \mathcal{K}$, there exists no $v \in \mathcal{C}^* \setminus \mathcal{K}$ with the Conditions (1) and (2) in the previous paragraph, and $\bar{d}_{\mathcal{C}_{\mathcal{F}\cup\mathcal{G}} \setminus \mathcal{K}}(\mu_i, \mu_j) \leq 49Tc_{ij}^2 \log \Lambda$. That is, the actual distance between μ_i and μ_j in $\mathcal{C}_{\mathcal{F}\cup\mathcal{G}} \setminus \mathcal{K}$ (as well as $\mathcal{C}^* \setminus \mathcal{K}$) is at most the contribution to $d(\mu_i, \mu_j)$ from the top $49Tc_{ij}^2 \log \Lambda$ coordinates, and the contribution to $d(\mu_i, \mu_j)$ from $\mathcal{K}$ and $\bar{C}^*$ is at least the contribution from the rest of the coordinates. Since $d_{\bar{\mathcal{C}}^*}(\mu_i, \mu_j) \leq \frac{1}{100} d(\mu_i, \mu_j)$, the distance between μ_i and μ_j in $\mathcal{K}$ is at least $\frac{99}{100} \bar{d}(\mu_i, \mu_j) - 49T \log \Lambda c_{ij}^2)$. The first part of the theorem follows.

The second part of the theorem follows directly from Lemma 13. □

References

[AK01] S. Arora and R. Kannan. Learning mixtures of arbitrary gaussians. In *Proceedings of 33rd ACM Symposium on Theory of Computing*, pages 247–257, 2001.

[AM05] D. Achlioptas and F. McSherry. On spectral learning of mixtures of distributions. In *Proceedings of the 18th Annual Conference on Learning Theory*, pages 458–469, 2005.

[AT98] A.Blum and T.Mitchell. Combining labeled and unlabeled data with co-training. In *Proc. of Conference on Learning Theory*, 1998.

[BNJ03] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, (3):993–1022, January 2003.

[Cha07] K. Chaudhuri. *Learning Mixtures of Distributions*. PhD thesis, University of California, Berkeley, 2007. UCB/EECS-2007-124.

[CHRZ07] K. Chaudhuri, E. Halperin, S. Rao, and S. Zhou. A rigorous analysis of population stratification with limited data. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, 2007.

[Das99] S. Dasgupta. Learning mixtures of gaussians. In *Proceedings of the 40th IEEE Symposium on Foundations of Computer S cience*, pages 634–644, 1999.

[DHKS05] A. Dasgupta, J. Hopcroft, J. Kleinberg, and M. Sandler. On learning mixtures of heavy-tailed distributions. In *Proceedings of the 46th IEEE Symposium on Foundations of Computer Science*, pages 491–500, 2005.

[DLR77] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm (with discussion). *Journal of the Royal Statistical Society B, 39*, pages 1–38, 1977.

[DS00] S. Dasgupta and L. Schulman. A two-round variant of em for gaussian mixtures. In *Sixteenth Conference on Uncertainty in Artificial Intelligence (UAI)*, 2000.

[FM99] Y. Freund and Y. Mansour. Estimating a mixture of two product distributions. In *COLT: Proceedings of the Workshop on Computational Learning Theory, Morgan Kaufmann Publishers*, 1999.

[FOS05] J. Feldman, R. O'Donnell, and R. Servedio. Learning mixtures of product distributions over discrete domains. In *Proceedings of FOCS*, 2005.

[FOS06] J. Feldman, R. O'Donnell, and R. Servedio. Learning mixtures of gaussians with no separation assumptions. In *Proceedings of COLT*, 2006.

[GL96] G. Golub and C. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, 1996.

[KF07] S. Kakade and D. Foster. Multi-view regression via canonical correlation analysis. In *Proc. of Conference on Learning Theory*, 2007.

[KS04] Jon M. Kleinberg and Mark Sandler. Using mixture models for collaborative filtering. In *STOC*, pages 569–578, 2004.

[KSV05] R. Kannan, H. Salmasian, and S. Vempala. The spectral method for general mixture models. In *Proceedings of the 18th Annual Conference on Learning Theory*, 2005.

[Llo82] S.P. Lloyd. Least squares quantization in pcm. *IEEE Trans. on Information Theory*, 1982.

[MB88] G.J. McLachlan and K.E. Basford. *Mixture Models: Inference and Applications to Clustering*. Marcel Dekker, 1988.

[PD05] A. Panconesi and D. Dubhashi. Concentration of measure for the analysis of randomised algorithms. Draft, 2005.

[PFK02] C. Pal, B. Frey, and T. Kristjansson. Noise robust speech recognition using Gaussian basis functions for non-linear likelihood function approximation. In *ICASSP '02: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages I–405–I–408, 2002.

[PSD00] J. K. Pritchard, M. Stephens, and P. Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155:954–959, June 2000.

[Rey95] D. Reynolds. Speaker identification and verification using gaussian mixture speaker models. *Speech Communications*, 1995.

[SRH07] Srinath Sridhar, Satish Rao, and Eran Halperin. An efficient and accurate graph-based approach to detect population substructure. In *RECOMB*, 2007.

[TSM85] D.M. Titterington, A.F.M. Smith, and U.E. Makov. *Statistical Analysis of Finite Mixture Distributions*. Wiley, 1985.

[VW02] V. Vempala and G. Wang. A spectral algorithm of learning mixtures of distributions. In *Proceedings of the 43rd IEEE Symposium on Foundations of Computer Science*, pages 113–123, 2002.