

# Separating Populations with Wide Data: a Spectral Analysis

Avrim Blum<sup>\*</sup>, Amin Coja-Oghlan<sup>\*\*</sup>, Alan Frieze<sup>\*\*\*</sup>, and Shuheng Zhou<sup>†</sup>

Carnegie Mellon University, Pittsburgh, PA 15213, USA

**Abstract.** In this paper, we consider the problem of partitioning a small data sample drawn from a mixture of  $k$  product distributions. We are interested in the case that individual features are of low average quality  $\gamma$ , and we want to use as few of them as possible to correctly partition the sample. We analyze a spectral technique that is able to approximately optimize the total data size—the product of number of data points  $n$  and the number of features  $K$ —needed to correctly perform this partitioning as a function of  $1/\gamma$  for  $K > n$ . Our goal is motivated by an application in clustering individuals according to their population of origin using markers, when the divergence between any two of the populations is small.

## 1 Introduction

We explore a type of classification problem that arises in the context of computational biology. The problem is that we are given a small sample of size  $n$ , e.g., DNA of  $n$  individuals (think of  $n$  in the hundreds or thousands), each described by the values of  $K$  *features* or *markers*, e.g., SNPs (Single Nucleotide Polymorphisms, think of  $K$  as an order of magnitude larger than  $n$ ). Our goal is to use these features to classify the individuals according to their population of origin. Features have slightly different probabilities depending on which population the individual belongs to, and are assumed to be independent of each other (i.e., our data is a small sample from a mixture of  $k$  very similar product distributions). The objective we consider is to minimize the total data size  $D = nK$  needed to correctly classify the individuals in the sample as a function of the “average quality”  $\gamma$  of the features, under the assumption that  $K > n$ . Throughout the paper, we use  $p_i^j$  and  $\mu_i^j$  as shorthands for  $p_i^{(j)}$  and  $\mu_i^{(j)}$  respectively.

**Statistical Model:** We have  $k$  probability spaces  $\Omega_1, \dots, \Omega_k$  over the set  $\{0, 1\}^K$ . Further, the components (*features*) of  $z \in \Omega_t$  are independent and  $\Pr_{\Omega_t}[z_i = 1] = p_t^i$  ( $1 \leq t \leq k, 1 \leq i \leq K$ ). Hence, the probability spaces  $\Omega_1, \dots, \Omega_k$  comprise the distribution of the features for each of the  $k$  populations. Moreover, the input of the algorithm consists of a collection (*mixture*) of  $n = \sum_{t=1}^k N_t$  unlabeled samples,  $N_t$  points from  $\Omega_t$ , and the algorithm is to determine for each data point from which of  $\Omega_1, \dots, \Omega_k$  it was chosen. In general we do *not* assume that  $N_1, \dots, N_t$  are revealed

<sup>\*</sup> Supported in part by the NSF under grant CCF-0514922.

<sup>\*\*</sup> Supported by the German Research Foundation under grant CO 646.

<sup>\*\*\*</sup> Supported in part by the NSF under grant CCF-0502793.

<sup>†</sup> Supported in part by the NSF under grants CCF-0625879 and CNF-0435382.

to the algorithm; but we do require some bounds on their relative sizes. An important parameter of the probability ensemble  $\Omega_1, \dots, \Omega_k$  is the *measure of divergence*

$$\gamma = \min_{1 \leq s < t \leq k} \frac{\sum_{i=1}^K (p_s^i - p_t^i)^2}{K} \quad (1)$$

between any two distributions. Note that  $\sqrt{K\gamma}$  measures the Euclidean distance between the means of any two distributions and thus represents their separation. Further, let  $N = n/k$  (so if the populations were balanced we would have  $N$  of each type) and assume from now on that  $kN < K$ . Let  $D = nK$  denote the size of the data-set. In addition, let  $\sigma^2 = \max_{i,t} p_t^i(1 - p_t^i)$  denote the maximum variance of any random bit.

The biological context for this problem is we are given DNA information from  $n$  individuals from  $k$  populations of origin and we wish to classify each individual into the correct category. DNA contains a series of markers called SNPs, each of which has two variants (alleles). Given the population of origin of an individual, the genotypes can be reasonably assumed to be generated by drawing alleles independently from the appropriate distribution. The following theorem gives a sufficient condition for a balanced ( $N_1 = N_2$ ) input instance when  $k = 2$ .

**Theorem 1. (Zhou 06 [25])** *Assume  $N_1 = N_2 = N$ . If  $K = \Omega(\frac{\ln N}{\gamma})$  and  $KN = \Omega(\frac{\ln N \log \log N}{\gamma^2})$  then with probability  $1 - 1/\text{poly}(N)$ , among all balanced cuts in the complete graph formed among  $2N$  sample individuals, the maximum weight cut corresponds to the partition of the  $2N$  individuals according to their population of origin. Here the weight of a cut is the sum of weights across all edges in the cut, and the edge weight equals the Hamming distance between the bit vectors of the two endpoints.*

Variants of the above theorem, based on a model that allows two random draws from each SNP for an individual, are given in [3, 25]. In particular, notice that edge weights based on the inner-product of two individuals' bit vectors correspond to the sample covariance, in which case the max-cut corresponds to the correct partition [25] with high probability. Finding a max-cut is computationally intractable; hence in the same paper [3], a hill-climbing algorithm is given to find the correct partition for balanced input instances but with a stronger requirement on the sizes of both  $K$  and  $nK$ .

**A Spectral Approach:** In this paper, we construct two simpler algorithms using spectral techniques, attempting to reproduce conditions above. In particular, we study the requirements on the parameters of the model (namely,  $\gamma$ ,  $N$ ,  $k$ , and  $K$ ) that allow us to classify every individual correctly and efficiently with high probability.

The two algorithms CLASSIFY and PARTITION compare as follows. Both algorithms are based on spectral methods originally developed in graph partitioning. More precisely, Theorem 2 is based on computing the singular vectors with the two largest singular values for each of the  $n \times K$  input random matrix. The procedure is conceptually simple, easy to implement, and efficient in practice. For simplicity, Procedure Classify assumes the separation parameter  $\gamma$  is known to decide which singular vector to examine; in practice, one can just try both singular vectors as we do in the simulations. Proof techniques for Theorem 2, however, are difficult to apply to cases of multiple populations, i.e.,  $k > 2$ . Procedure Partition is based on computing a rank- $k$

approximation of the input random matrix and can cope with a mixture of a constant number of populations. It is more intricate for both implementation and execution than CLASSIFY. It does not require  $\gamma$  as an input, while only requires that the constant  $k$  is given. We prove the following theorems.

**Theorem 2.** Let  $\omega = \frac{\min(N_1, N_2)}{n}$  and  $\omega_{\min}$  be a lower bound on  $\omega$ . Let  $\gamma$  be given. Assume that  $K > 2n \ln n$  and  $k = 2$ . Procedure CLASSIFY allows us to separate two populations w.h.p., when  $n \geq \Omega\left(\frac{\sigma^2}{\gamma \omega_{\min} \omega}\right)$ , where  $\sigma^2$  is the largest variance of any random bit, i.e.  $\sigma^2 = \max_{i,t} p_t^i(1 - p_t^i)$ . Thus if the populations are roughly balanced, then  $n \geq \frac{c}{\gamma}$  suffices for some constant  $c$ .

This implies that the data required is  $D = nK = O(\ln n \sigma^4 / \gamma^2 \omega^2 \omega_{\min}^2)$ . Let  $P_s = (p_s^i)_{i=1, \dots, K}$ , we have

$$\|P_1 - P_2\|_2 = \sqrt{K\gamma} = \sqrt{\sum_{i=1}^K (p_1^i - p_2^i)^2} \geq \frac{\sigma}{\omega_{\min} \omega} \sqrt{\ln n}. \quad (2)$$

**Theorem 3.** Let  $\omega = \frac{\min(N_1, \dots, N_k)}{n}$ . There is a polynomial time algorithm PARTITION that satisfies the following. Suppose that  $K > n \log n$  and  $n > \frac{C_k \sigma^2}{\gamma \omega}$  for some large enough constant  $C_k$  and that  $\omega = \Omega(1)$ . Then given the empirical  $n \times K$  matrix comprising the  $K$  features for each of the  $n$  individuals along with the parameter  $k$ , PARTITION separates the  $k$  populations correctly w.h.p.

**Summary and Future Direction:** Note that unlike Theorem 1, both Theorem 2 and Theorem 3 require a lower bound on  $n$ , even when  $k = 2$  and the input instance is balanced. We illustrate through simulations to show that this seems not to be a fundamental constraint of the spectral techniques; our experimental results show that even when  $n$  is small, by increasing  $K$  so that  $nK = \Omega(1/\gamma^2)$ , one can classify a mixture of two populations using ideas in Procedure CLASSIFY with success rate reaching an ‘‘oracle’’ curve, which is computed assuming that distributions are known, where success rate means the ratio between correctly classified individuals and  $N$ . Exploring the tradeoffs of  $n$  and  $K$  that are sufficient for classification, when sample size  $n$  is small, is both of theoretical interests and practical value.

## 1.1 Related Work

In their seminal paper [21], Pritchard, Stephens, and Donnelly presented a model-based clustering method to separate populations using genotype data. They assume that observations from each cluster are random from some parametric model. Inference for the parameters corresponding to each population is done jointly with inference for the cluster membership of each individual, and  $k$  in the mixture, using Bayesian methods.

The idea of exploiting the eigenvectors with the first two eigenvalues of the adjacency matrix to partition graphs goes back to the work of Fiedler [12], and has been used in the heuristics for various NP-hard graph partitioning problems (e.g., [13]). The main difference between graph partitioning problems and the classification problem

that we study is that the matrices occurring in graph partitioning are symmetric and hence diagonalizable, while our input matrix is rectangular in general. Thus, the contribution of Theorem 2 is to show that a conceptually simple and efficient algorithm based on singular value decompositions performs well in the framework of a fairly general probabilistic model, where probabilities for each of the  $K$  features for each of the  $k$  populations are allowed to vary. Indeed, the analysis of CLASSIFY requires exploring new ideas such as the Separation Lemma and the normalization of the random matrix  $X$ , for generating a large gap between top two singular values of the expectation matrix  $\mathcal{X}$  and for bounding the angle between random singular vectors and their static correspondents, details of which are included in Section 2 with analysis in full version.

Procedure Partition and its analysis build upon the spectral techniques of McSherry [18] on graph partitioning, and an extension due to Coja-Oghlan [4]. McSherry provides a comprehensive probabilistic model and presents a spectral algorithm for solving the partitioning problem on random graphs, provided that a separation condition similar to (2) is satisfied. Indeed, [18] encompasses a considerable portion of the prior work on Graph Coloring, Minimum Bisection, and finding Maximum Clique. Moreover, McSherry’s approach easily yields an algorithm that solves the classification problem studied in the present paper under similar assumptions as in Theorem 3, provided that the algorithm is given the parameter  $\gamma$  as an additional input; this is actually pointed out in the conclusions of [18]. In the context of graph partitioning, an algorithm that does not need the separation parameter as an input was devised in [4]. The main difference between PARTITION and the algorithm presented in [4] is that PARTITION deals with the asymmetric  $n \times K$  matrix of individuals/features, whereas [4] deals with graph partitioning (i.e., a symmetric matrix).

There are two streams of related work in the learning community. The first stream is the recent progress in learning from the point of view of clustering: given samples drawn from a mixture of well-separated Gaussians (component distributions), one aims to classify each sample according to which component distribution it comes from, as studied in [8, 9, 2, 23, 1, 15, 7]. This framework has been extended to more general distributions such as log-concave distributions in [1, 15] and heavy-tailed distributions in [7], as well as to more than two populations. These results focus mainly on reducing the requirement on the separations between any two centers  $P_1$  and  $P_2$ . In contrast, we focus on the sample size  $D$ . This is motivated by previous results [3, 25] stating that by acquiring enough attributes along the same set of dimensions from each component distribution, with high probability, we can correctly classify every individual.

While our aim is different from those results, where  $n > K$  is almost universal and we focus on cases  $K > n$ , we do have one common axis for comparison, the  $\ell_2$ -distance between any two centers of the distributions. In earlier works [9, 2], the separation requirement depended on the number of dimensions of each distribution; this has recently been reduced to be independent of  $K$ , the dimensionality of the distribution for certain classes of distributions [1, 15]. This is comparable to our requirement in (2) for the discrete distributions. For example, according to Theorem 7 in [1], in order to separate the mixture of two Gaussians,

$$\|P_1 - P_2\|_2 = \Omega \left( \frac{\sigma}{\sqrt{\omega}} + \sigma \sqrt{\log n} \right) \quad (3)$$

is required. Besides Gaussian and Logconcave, a general theorem: Theorem 6 in [1] is derived that in principle also applies to mixtures of discrete distributions. The key difficulty of applying their theorem directly to our scenario is that it relies on a concentration property of the distribution (Eq. (10) of [1]) that need not hold in our case. In addition, once the distance between any two centers is fixed (i.e., once  $\gamma$  is fixed in the discrete distribution), the sample size  $n$  in their algorithms is always larger than  $\Omega(\frac{K}{\omega} \log^5 K)$  [1, 15] for log-concave distributions (in fact, in Theorem 3 of [15], they discard at least this many individuals in order to correctly classify the rest in the sample), and larger than  $\Omega(\frac{K}{\omega})$  for Gaussians [1], whereas in our case,  $n < K$  always holds. Hence, our analysis allows one to obtain a clean bound on  $n$  in the discrete case.

The second stream of work is under the PAC-learning framework, where given a sample generated from some target distribution  $Z$ , the goal is to output a distribution  $Z_1$  that is close to  $Z$  in Kullback-Leibler divergence:  $KL(Z||Z_1)$ , where  $Z$  is a mixture of product distributions over discrete domains or Gaussians [16, 14, 5, 6, 20, 10, 11]. They do not require a minimal distance between any two distributions, but they do not aim to classify every sample point correctly either, and in general require much more data.

## 2 A Simple Algorithm Using Singular Vectors

As described in Theorem 2, we assume we have a mixture of two product distributions. Let  $N_1, N_2$  be the number of individuals from each population class. Our goal is to correctly classify all individuals according to their distributions. Let  $n = 2N = N_1 + N_2$ , and refer to the case when  $N_1 = N_2$  as the balanced input case. For convenience, let us redefine “ $K$ ” to assume we have  $O(\log n)$  blocks of  $K$  features each (so the total number of features is really  $O(K \log n)$ ) and we assume that each set of  $K$  features has divergence at least  $\gamma$ . (If we perform this partitioning of features into blocks randomly, then with high probability this divergence has changed by only a constant factor for most blocks.) The high-level idea of the algorithm is now to repeat the following procedure for each block of  $K$  features: use the  $K$  features to create an  $n \times K$  matrix  $X$ , such that each row  $X_i, i = 1, \dots, n$ , corresponds to a feature vector for one sample point, across its  $K$  dimensions. We then compute the top two left singular vectors  $u_1, u_2$  of  $X$  and use these to classify each sample. This classification induces some probability of error  $f$  for each individual at each round, so we repeat the procedure for each of the  $O(\log n)$  blocks and then take majority vote over different runs. Each round we require  $K \geq n$  features, so we need  $O(n \log n)$  features total in the end.

In more detail, we repeat the following procedure  $O(\log n)$  times. Let  $T = \frac{15N}{32} \sqrt{3\omega_{\min}\gamma}$ , where  $\omega_{\min}$  is the lower bound on the minimum weight  $\min\{\frac{N_1}{2N}, \frac{N_2}{2N}\}$ , which is independent of an actual instance. Let  $s_1(X), s_2(X)$  be the top two singular values of  $X$ .

**Procedure Classify:** Given  $\gamma, N, \omega_{\min}$ . Assume that  $N \gg \frac{1}{\gamma}$ ,

- Normalization: use the  $K$  features to form a random  $n \times K$  matrix  $X$ ; Each individual random variable  $X_{i,j}$  is a *normalized* random variable based on the original Bernoulli r.v.  $b_{i,j} \in \{0, 1\}$  with  $\Pr[b_{i,j} = 1] = p_1^j$  for  $X_i \in P_1$  and  $\Pr[b_{i,j} = 1] = p_2^j$  for  $X_i \in P_2$ , such that  $X_{i,j} = \frac{b_{i,j}}{2}$ .

- Take top two left singular vectors  $u_1, u_2$  of  $X$ , where  $u_i = [u_{i,1}, \dots, u_{i,n}]$ ,  $i = 1, 2$ .
  1. If  $s_2(X) > T = \frac{15N}{32} \sqrt{3\omega_{\min}\gamma}$ , use  $u_2$  to partition the individuals with 0 as the threshold, i.e., partition  $j \in [n]$  according to  $u_{2,j} < 0$  or  $u_{2,j} \geq 0$ .
  2. Otherwise, use  $u_1$  to partition, with mixture mean  $M = \sum_{i=1}^n u_{1,n}$  as the threshold.

**Analysis of the Simple Algorithm:** Our analysis is based on comparing entries in the top two singular vectors of the normalized random  $n \times K$  matrix  $X$ , with those of a static matrix  $\mathcal{X}$ , where each entry  $\mathcal{X}_{i,j} = \mathbf{E}[X_{i,j}]$  is the expected value of the corresponding entry in  $X$ . Hence  $\forall i = 1, \dots, N_1$ ,  $\mathcal{X}_i = [\mu_1^1, \mu_1^2, \dots, \mu_1^K]$ , where  $\mu_1^j = \frac{1+p_1^j}{2}$ ,  $\forall j$ , and  $\forall i = N_1 + 1, \dots, n$ ,  $\mathcal{X}_i = [\mu_2^1, \mu_2^2, \dots, \mu_2^K]$ , where  $\mu_2^j = \frac{1+p_2^j}{2}$ ,  $\forall j$ . We assume the divergence is exactly  $\gamma$  among the  $K$  features that we have chosen in all calculations.

The inspiration for this approach is based on the following lemma, whose proof is built upon a theorem that is presented in a lecture note by Spielman [22]. For a  $n \times K$  matrix  $A$ , let  $s_1(A) \geq s_2(A) \geq \dots \geq s_n(A)$  be singular values of  $A$ . Let  $u_1, \dots, u_n, v_1, \dots, v_n$ , be the  $n$  left and right singular vectors of  $X$ , corresponding to  $s_1(X), \dots, s_n(X)$  such that  $\|u_i\|_2 = 1, \|v_i\|_2 = 1, \forall i$ . We denote the set of  $n$  left and right singular vectors of  $\mathcal{X}$  with  $\bar{u}_1, \dots, \bar{u}_n, \bar{v}_1, \dots, \bar{v}_n$ .

**Lemma 4.** *Let  $X$  be the random  $n \times K$  matrix and  $\mathcal{X}$  its expected value matrix. Let  $A = X - \mathcal{X}$  be the zero-mean random matrix. Let  $\theta_i$  be the angle between two vectors:  $[u_i, v_i], [\bar{u}_i, \bar{v}_i]$ , where  $\|[u_i, v_i]\|_2 = \|[\bar{u}_i, \bar{v}_i]\|_2 = 2$  and  $[u, v]$  represents a vector that is the concatenation of two vectors  $u, v$ .*

$$\|u_i - \bar{u}_i\|_2 \leq \|[u_i, v_i] - [\bar{u}_i, \bar{v}_i]\|_2 \approx 2\theta_i \approx 2 \sin(\theta_i) \leq \frac{4s_1(A)}{\text{gap}(i, \mathcal{X})}, \quad (4)$$

where  $\text{gap}(i, \mathcal{X}) = \min_{j \neq i} |s_i(\mathcal{X}) - s_j(\mathcal{X})|$ .

We first bound the largest singular value  $s_1(A) = s_1(X - \mathcal{X})$  of  $(a_{i,j})$  with independent zero-mean entries, which defines the Euclidean operator norm

$$\|(a_{i,j})\| := \sup \left\{ \sum_{i,j} a_{i,j} x_i y_j : \sum x_i^2 \leq 1, \sum y_i^2 \leq 1 \right\}. \quad (5)$$

The behavior of the largest singular value of an  $n \times m$  random matrices  $A$  with i.i.d. entries is well studied. Latala [17] shows that the weakest assumption for its regular behavior is boundedness of the fourth moment of the entries, even if they are not identically distributed. Combining Theorem 5 of Latala with the concentration Theorem 6 by Meckes [19] proves Theorem 7 that we need <sup>1</sup>.

**Theorem 5. (Bounded Norm of Random Matrices [17])** *For any finite  $n \times m$  matrix  $A$  of independent mean zero r.v.'s  $a_{i,j}$  we have, for an absolute constant  $C$ ,*

$$\mathbf{E} \|(a_{i,j})\| \leq C \left( \max_i \sqrt{\sum_j \mathbf{E} a_{i,j}^2} + \max_j \sqrt{\sum_i \mathbf{E} a_{i,j}^2} + \left( \sum_{i,j} \mathbf{E} a_{i,j}^4 \right)^{\frac{1}{4}} \right). \quad (6)$$

<sup>1</sup> One can also obtain an upper bound of  $O(\sqrt{n+K})$  on  $s_1(A)$  using a theorem on by Vu [24], through the construction a  $(n+K) \times (n+K)$  square matrix out of  $A$ .

**Theorem 6. (Concentration of Largest Singular Value: Bounded Range [19])** For any finite  $n \times m$ , where  $n \leq m$ , matrix  $A$ , such that entries  $a_{i,j}$  are independent r.v. supported in an interval of length at most  $D$ , then, for all  $t$ ,

$$\Pr[|s_1(A) - \mathbb{M}s_1(A)| \geq t] \leq 4e^{-t^2/4D^2}. \quad (7)$$

**Theorem 7. (Largest Singular Value of a Mean-zero Random Matrix)** For any finite  $n \times K$ , where  $n \leq K$ , matrix  $A$ , such that entries  $a_{i,j}$  are independent mean zero r.v. supported in an interval of length at most  $D$ , with fourth moment upper bounded by  $B$ , then

$$\Pr[s_1(A) \geq CB^{1/4}\sqrt{K} + 4D\sqrt{\pi} + t] \leq 4e^{-t^2/4} \quad (8)$$

for all  $t$ . Hence  $\|A\| \leq C_1B^{1/4}\sqrt{K}$  for an absolute constant  $C_1$ .

## 2.1 Generating a Large Gap in $s_1(\mathcal{X})$ , $s_2(\mathcal{X})$

In order to apply Lemma 4 to the top two singular vectors of  $X$  and  $\mathcal{X}$  through

$$\|u_1 - \bar{u}_1\|_2 \leq \frac{4s_1(X - \mathcal{X})}{|s_1(\mathcal{X}) - s_2(\mathcal{X})|} \quad (9)$$

$$\|u_2 - \bar{u}_2\|_2 \leq \frac{4s_1(X - \mathcal{X})}{\min(|s_1(\mathcal{X}) - s_2(\mathcal{X})|, |s_2(\mathcal{X})|)}, \quad (10)$$

we need to first bound  $|s_1(\mathcal{X}) - s_2(\mathcal{X})|$  away from zero, since otherwise, RHSs on both (9) and (10) become unbounded. We then analyze  $\text{gap}(2, \mathcal{X}) = \min(|s_1(\mathcal{X}) - s_2(\mathcal{X})|, |s_2(\mathcal{X})|)$ .

Let us first define values  $a, b, c$  that we use throughout the rest of the paper:

$$a = \sum_{k=1}^K (\mu_1^k)^2, \quad b = \sum_{k=1}^K \mu_1^k \mu_2^k, \quad c = \sum_{k=1}^K (\mu_2^k)^2. \quad (11)$$

For the following analysis, we can assume that  $a, b, c \in [K/4, K]$ , given that  $X$  is normalized in Procedure Classify.

We first show that normalization of  $X$  as described in Procedure Classify guarantees that not only  $|s_1(\mathcal{X}) - s_2(\mathcal{X})| \neq 0$ , but there also exists a  $\Theta(\sqrt{NK})$  amount of gap between  $s_1(\mathcal{X})$  and  $s_2(\mathcal{X})$  in Proposition 8:

$$\text{gap}(\mathcal{X}) := |s_1(\mathcal{X}) - s_2(\mathcal{X})| = \Theta(\sqrt{NK}). \quad (12)$$

**Proposition 8.** For a normalized random matrix  $X$ , its expected value matrix  $\mathcal{X}$  satisfies  $\frac{4c_0\sqrt{2NK}}{5} \leq \text{gap}(\mathcal{X}) \leq \sqrt{2NK}$ , where  $c_0 = \frac{|b|\sqrt{ac}}{K(a+c)}$  is a constant, given that  $a, b, c \in [K/4, K]$  as defined in (11). In addition,

$$\sqrt{\frac{KN}{4}} \leq s_1(\mathcal{X}) \leq \sqrt{2NK}, \text{ and } \sqrt{\frac{NK}{2}} \leq s_1(\mathcal{X}) + s_2(\mathcal{X}) \leq \sqrt{2NK}. \quad (13)$$

We next state a few important results that justify Procedure Classify. Note that the left singular vectors  $\bar{u}_i, \forall i$  of  $\mathcal{X}$  are of the form  $[x_i, \dots, x_i, y_i, \dots, y_i]^T$ :

$$\bar{u}_1 = [x_1, \dots, x_1, y_1, \dots, y_1]^T, \text{ and } \bar{u}_2 = [x_2, \dots, x_2, y_2, \dots, y_2]^T, \quad (14)$$

where  $x_i$  repeats  $N_1$  times and  $y_i$  repeats  $N_2$  times. We first show Proposition 9 regarding signs of  $x_i, y_i, i = 1, 2$ , followed by a lemma bounding the separation of  $x_2, y_2$ . We then state the key Separation Lemma that allows us to conclude that least one of top two left singular vectors of  $X$  can be used to classify data at each round. It can be extended to cases when  $k > 2$ .

**Proposition 9.** *Let  $b$  as defined in (11): when  $b > 0$ , entries  $x_1, y_1$  in  $\bar{u}_1$  have the same sign while  $x_2, y_2$  in  $\bar{u}_2$  have opposite signs.*

**Lemma 10.**  $|x_2 - y_2|^2 \leq \frac{C_{\max}}{2N}$  where  $C_{\max} = \left(\sqrt{\frac{1}{\omega_1}} + \sqrt{\frac{1}{\omega_2}}\right)^2 \leq \frac{4}{\omega_{\min}}$ ;  $|x_2|^2 \geq \frac{C_{x \min}}{2N}$  where  $C_{x \min} = \frac{\omega_2}{4\omega_1^2 + \omega_1\omega_2}$ ;  $|y_2|^2 \geq \frac{C_{y \min}}{2N}$  where  $C_{y \min} = \frac{\omega_1}{4\omega_2^2 + \omega_1\omega_2}$ .

**Lemma 11. (Separation Lemma)**  $K\gamma = s_1(\mathcal{X})^2(x_1 - y_1)^2 + s_2(\mathcal{X})^2(x_2 - y_2)^2$ .

*Proof.* Let  $\Delta := P_1 - P_2$  as in Theorem 2, and  $\mathbf{b} = [1, 0, \dots, 0, -1, 0, \dots, 0]^T$ , where 1 appears in the first and  $-1$  appears in the  $N_1 + 1^{\text{st}}$  positions. Then  $\Delta = X^T \mathbf{b} = [\mu_1^1 - \mu_2^1, \mu_1^2 - \mu_2^2, \dots, \mu_1^K - \mu_2^K]$ . Given  $\mathcal{X} = s_1(\mathcal{X})\bar{u}_1\bar{v}_1^T + s_2(\mathcal{X})\bar{u}_2\bar{v}_2^T$ , we thus rewrite  $\Delta$  as:  $\Delta = \mathcal{X}^T \mathbf{b} = s_1(\mathcal{X})\bar{v}_1\bar{u}_1^T \mathbf{b} + s_2(\mathcal{X})\bar{v}_2\bar{u}_2^T \mathbf{b} = s_1(\mathcal{X})\bar{v}_1(x_1 - y_1) + s_2(\mathcal{X})\bar{v}_2(x_2 - y_2)$ . The lemma follows from the fact that  $\|\Delta\|_2 = \sqrt{K}\gamma$  and  $\bar{v}_1, \bar{v}_2$  are orthonormal. ■

Combining Proposition 9, Lemma 10, (13), and Lemma 11, we have

**Corollary 12.**  $s_2(\mathcal{X}) \leq \frac{\sqrt{2N}K\gamma}{\sqrt{C_{x \min}} + \sqrt{C_{y \min}}}$ , and hence  $\text{gap}(2, \mathcal{X}) = \min(s_2(\mathcal{X}), |s_1(\mathcal{X}) - s_2(\mathcal{X})|) = s_2(\mathcal{X})$  for a sufficiently small  $\gamma$ .

Finally, we show that the probability of error at each round for each individual is at most  $f = 1/10$ , given the sample size  $n$  as specified in Theorem 2. Hence by taking majority vote over the different runs for each sample, our algorithm will find the correct partition with probability  $1 - 1/n^2$ , given that at each round we take a set of  $K > n$  independent features. We leave the detailed analysis in full version.

### 3 The Algorithm PARTITION

As in Section 2, by repeating the partitioning process  $\log n$  times, we may restrict our attention to the problem of classifying a constant fraction of the individuals correctly. Let  $V = \{1, \dots, n\}$  be the set of all  $n$  individuals, and let  $\psi : V \rightarrow \{1, \dots, k\}$  be the map that assigns to each individual the population it belongs to. Further, set  $V_t = \psi^{-1}(t)$ , define  $N_t = |V_t|$ ,  $\Gamma = K\gamma$ , and  $\lambda = \sqrt{K}\sigma$ . In addition, let  $A = (a_{vi})$  denote the empirical  $n \times K$  input matrix. Then the assumption from Theorem 3 can be rephrased as  $n_{\min}K\gamma > C_k\lambda^2$ .

If  $X = (x_{ij})_{1 \leq i \leq n, 1 \leq j \leq K}$  is a  $n \times K$  matrix, then we let  $\|X\| = \max_{\|\xi\|=1} \|X\xi\|$  signify the operator norm of  $X$ , while  $\|X\|_F = \left(\sum_{i,j} x_{ij}^2\right)^{\frac{1}{2}}$  denotes the Frobenius

norm. The algorithm PARTITION computes a *rank  $k$  approximation*  $\widehat{A}$  of the input matrix  $A$ . That is,  $\widehat{A}$  is a  $n \times K$  matrix of rank at most  $k$ , and if  $B$  is any  $n \times K$  matrix of rank at most  $k$ , then  $\|A - \widehat{A}\| \leq \|A - B\|$ . Such an  $\widehat{A}$  can be computed in polynomial time via singular value decomposition. Let  $\widehat{A}_v$  denote the  $v$ -row of  $\widehat{A}$ .

**Algorithm 13.** PARTITION( $A, k$ )

*Input:* A  $n \times K$  matrix  $A$  and the parameter  $k$ . *Output:* A partition  $S_1, \dots, S_k$  of  $V$ .

1. Compute a rank  $k$  approximation  $\widehat{A}$  of  $A$ .
- For  $j = 1, \dots, 2 \log K$  do
  2. Let  $\Gamma_j = K2^{-j}$  and compute  $Q^{(j)}(v) = \{w \in V : \|\widehat{A}_w - \widehat{A}_v\|^2 \leq 0.01\Gamma_j^2\}$  for all  $v \in V$ .  
Then, determine sets  $Q_1^{(j)}, \dots, Q_k^{(j)}$  as follows: for  $i = 1, \dots, k$  do
    3. Pick  $v \in V \setminus \bigcup_{l=1}^{i-1} Q_l^{(j)}$  such that  $|Q^{(j)}(v) \setminus \bigcup_{l=1}^{i-1} Q_l^{(j)}|$  is maximum.  
Set  $Q_i^{(j)} = Q^{(j)}(v) \setminus \bigcup_{l=1}^{i-1} Q_l^{(j)}$  and  $\xi_i^{(j)} = \frac{1}{|Q_i^{(j)}|} \sum_{w \in Q_i^{(j)}} \widehat{A}_w$ .
  4. Partition the entire set  $V$  as follows: first, let  $S_i^{(j)} = Q_i^{(j)}$  for all  $1 \leq i \leq k$ .  
Then, add each  $v \in V \setminus \bigcup_{l=1}^k Q_l^{(j)}$  to a set  $S_i^{(j)}$  such that  $\|\widehat{A}_v - \xi_i^{(j)}\|$  is minimum.  
Set  $r_j = \sum_{i=1}^k \sum_{v \in S_i^{(j)}} \|\widehat{A}_v - \xi_i^{(j)}\|^2$ .
5. Let  $J$  be such that  $r^* = r_J$  is minimum. Return  $S_1^{(J)}, \dots, S_k^{(J)}$ .

The basic idea behind PARTITION is to classify each individual  $v \in V$  according to its row vector  $\widehat{A}_v$  in the rank  $k$  approximation  $\widehat{A}$ . That is, two individuals  $v, w$  are deemed to belong to the same population iff  $\|\widehat{A}_v - \widehat{A}_w\|^2 \leq 0.01\Gamma^2$ . Hence, PARTITION tries to determine sets  $S_1, \dots, S_k$  such that for any two  $v, w$  in the same set  $S_j$  the distance  $\|\widehat{A}_v - \widehat{A}_w\|$  is small. To see why classifying the individuals according to their corresponding row vectors in  $\widehat{A}$  is a good idea, we consider an auxiliary matrix  $\mathbb{E} = (\mathbb{E}_{vi})$  with entries  $\mathbb{E}_{vi} = p_{\psi(v)}^i$ . Thus, the entries of  $\mathbb{E}$  equal the expectations of the entries of  $A$ .

**Lemma 14.** *There is a constant  $C > 0$  such that  $\sum_{v \in V} \|\widehat{A}_v - \mathbb{E}_v\|^2 \leq Ck\lambda^2$  whp.*

*Proof.* Recall that  $\widehat{A}$  and  $\mathbb{E}$  both have rank  $\leq k$ , we obtain

$$\sum_{v \in V} \|\widehat{A}_v - \mathbb{E}_v\|^2 = \|\widehat{A} - \mathbb{E}\|_F^2 \leq 2k\|\widehat{A} - \mathbb{E}\| \leq 8k\|A - \mathbb{E}\|^2 \leq Ck\lambda^2,$$

where the last inequality follows from Theorem 7. ■

Observe that Lemma 14 implies that for *most*  $v$  we have  $\|\widehat{A}_v - \mathbb{E}_v\|^2 \leq 10^{-6}\Gamma$ , say. For letting  $z = |\{v : \|\widehat{A}_v - \mathbb{E}_v\|^2 > 10^{-6}\Gamma\}|$ , we get  $10^{-6}\Gamma z \leq \sum_{v \in V} \|\widehat{A}_v - \mathbb{E}_v\|^2 \leq Ck\lambda^2$ , whence  $z \ll n_{\min}$  due to our assumption that  $n_{\min}\Gamma \gg k\lambda^2$ . Thus, most rows of  $\widehat{A}$  are close to the corresponding rows of the *expected* matrix  $\mathbb{E}$ . Since  $\Gamma$  is not given to the algorithm as an input parameter, PARTITION has to estimate  $\Gamma$  on its own.

To this end, the outer loop goes through  $2 \log K$  “candidate values”  $\Gamma_j$ . These values are then used to obtain a partition  $Q_1^{(1)}, \dots, Q_1^{(k)}$  in Steps 2–4, which are similar to the algorithm presented in [18]. In addition, Step 4 computes the error parameter  $r_j$ . Finally Step 5 outputs the partition that minimizes the error parameter  $r_j$ . More precisely, Step 2

uses  $\Gamma_j$  to compute for each  $v \in V$  the set  $Q(v)$  of elements  $w$  such that  $\|\widehat{A}_w - \widehat{A}_v\| \leq 0.01\Gamma_j^2$ . Then, Step 3 tries to compute “big” disjoint  $Q_1^{(j)}, \dots, Q_k^{(j)}$ , where each  $Q_i^{(j)}$  results from some  $Q(v_i)$ . Further, Step 4 assigns all elements  $v$  not covered by  $Q_1^{(j)}, \dots, Q_k^{(j)}$  to that  $Q_i^{(j)}$  whose “center vector”  $\xi_i^{(j)}$  is closest to  $\widehat{A}_v$ .

Thus, we need to show that eventually picking the partition whose error term  $r_j$  is minimum yields a good approximation to the ideal partition  $V_1, \dots, V_k$ . The basic reason why this is true is that  $\xi_i^{(j)}$  should approximate the expectation  $\mathbb{E}^{V_i}$  for class  $V_i$  well iff  $Q_i^{(j)}$  is a good approximation of  $V_i$ . Hence, if  $Q_1^{(j)}, \dots, Q_k^{(j)}$  is “close” to  $V_1, \dots, V_k$ , then  $r_j = \sum_{i=1}^k \sum_{v \in S_i^{(j)}} \|\widehat{A}_v - \xi_i^{(j)}\|^2 \approx \|\widehat{A} - \mathbb{E}\|_F^2$  will be about as small as  $\|\widehat{A} - \mathbb{E}\|_F^2$  (cf. Lemma 14). Furthermore, Lemma 16 shows that any partition such that  $r_j$  is small yields a good approximation of  $V_1, \dots, V_k$ . Theorem 3 is an immediate consequence of Lemmas 15 and 16.

**Lemma 15.** *If  $\frac{1}{2}\Gamma \leq \Gamma_j \leq \Gamma$ , then  $r_j \leq C_0 k^3 \lambda^2$  for a certain constant  $C_0 > 0$ .*

**Lemma 16.** *Let  $S_1, \dots, S_k$  be a partition and  $\xi_1, \dots, \xi_k$  a sequence of vectors such that  $\sum_{i=1}^k \sum_{v \in S_i} \|\xi_i - A_v^*\|^2 \leq C_0 k^3 \lambda^2$ . Then there is a bijection  $\Xi : \{1, \dots, k\} \rightarrow \{1, \dots, k\}$  such that the following holds.*

1.  $\|\xi_i - \mathbb{E}^{V_{\Xi(i)}}\|^2 \leq 0.001\Gamma^2$  for all  $i = 1, \dots, k$ , and
2.  $\sum_{i=1}^k |S_i \Delta V_{\Xi(i)}| < 0.001n_{\min}$ .

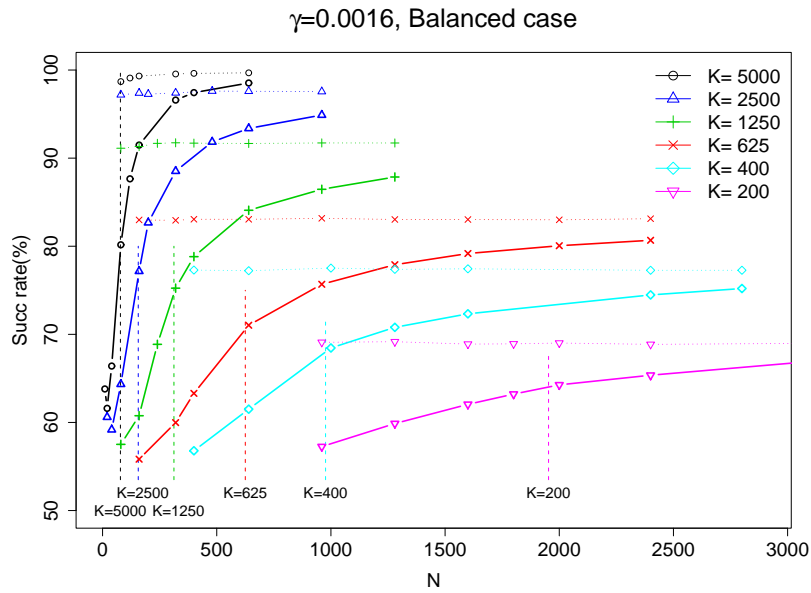
## 4 Experiments

We illustrate the effectiveness of spectral techniques using simulations. In particular, we explore the case when we have a mixture of two populations; we show that when  $NK > 1/\gamma^2$  and  $K > 1/\gamma$ , either the first or the second left singular vector of  $X$  shows an approximately correct partitioning, meaning that the success rate is well above 1/2. The entry-wise expected value matrix  $\mathcal{X}$  is: among  $K/2$  features,  $p_1^i > p_2^i$  and for the other half,  $p_1^i < p_2^i$  such that  $\forall i, p_1^i, p_2^i \in \{\frac{1+\alpha}{2} + \frac{\epsilon}{2}, \frac{1-\alpha}{2} + \frac{\epsilon}{2}\}$ , where  $\epsilon = 0.1\alpha$ . Hence  $\gamma = \alpha^2$ . We report results on balanced cases only, but we do observe that unbalanced cases show similar tradeoffs. For each population  $P$ , the success rate is defined as the number of individuals that are correctly classified, i.e., they belong to a group that  $P$  is the majority of that group, versus the size of the population  $|P|$ .

Each point on the SVD curve corresponds to an average rate over 100 trials. Since we are interested in exploring the tradeoffs of  $N, K$  in all ranges (e.g., when  $N \ll K$  or  $N \gg K$ ), rather than using the threshold  $T$  in Procedure Classify that is chosen in case both  $N, K > 1/\gamma$ , to decide which singular vector to use, we try both  $u_1$  and  $u_2$  and use the more effective one to measure the success rate at each trial. For each data point, the distribution of  $X$  is fixed across all trials and we generate an independent  $X_{2N \times K}$  for each trial to measure success rate based on the more effective classifier between  $u_1$  and  $u_2$ .

One can see from the plot that when  $K < 1/\gamma$ , i.e., when  $K = 200$  and  $400$ , no matter how much we increase  $N$ , the success rate is consistently low. Note that 50/100 of success rate is equivalent to a total failure. In contrast, when  $N$  is smaller than  $1/\gamma$ ,

as we increase  $K$ , we can always classify with a high success rate, where in general,  $NK > 1/\gamma^2$  is indeed necessary to see a high success rate. In particular, the curves for  $K = 5000, 2500, 1250$  show the sharpness of the threshold behavior for increasing sample size  $n$  from below  $1/K\gamma^2$  to above. For each curve, we also compute the best possible classification one could hope to make if one knew in advance which features satisfied  $p_1^i > p_2^i$  and which satisfied  $p_1^i < p_2^i$ . These are the horizontal(ish) dotted lines above each curve. The fact that the solid curves are approaching these information-theoretic upper bounds shows that the spectral technique is correctly using the available information.



**Fig. 1.** Plots show success rate as a function of  $N$  for several values of  $K$ , when  $\gamma = (0.04)^2$ . Each point is an average over 100 trials. Horizontal lines (“oracles”) indicate the information-theoretically best possible success rate for that value of  $K$  (how well one could do if one knew in advance which features satisfied  $p_1^i > p_2^i$  and which satisfied  $p_1^i < p_2^i$ ; they are not exactly horizontal because they are also an average over 100 runs). Vertical bars indicate the value of  $N$  for which  $NK = 1/\gamma^2$ .

## 5 Acknowledgments

We thank John Lafferty, Frank McSherry, Roman Vershynin and Larry Wasserman for many helpful discussions on this work.

## References

1. D. Achlioptas and F. McSherry. On spectral learning of mixtures of distributions. In *Proceedings of the 18th Annual COLT*, pages 458–469, 2005. (Version in <http://www.cs.ucsc.edu/> op-

tas/papers/).

2. S. Arora and R. Kannan. Learning mixtures of arbitrary gaussians. In *Proceedings of 33rd ACM Symposium on Theory of Computing*, pages 247–257, 2001.
3. K. Chaudhuri, E. Halperin, S. Rao, and S. Zhou. A rigorous analysis of population stratification with limited data. In *Proceedings of the 18th ACM-SIAM SODA*, 2007.
4. A. Coja-Oghlan. An adaptive spectral heuristic for partitioning random graphs. In *Proceedings of the 33rd ICALP*, 2006.
5. M. Cryan. *Learning and approximation Algorithms for Problems motivated by evolutionary trees*. PhD thesis, University of Warwick, 1999.
6. M. Cryan, L. Goldberg, and P. Goldberg. Evolutionary trees can be learned in polynomial time in the two state general markov model. *SIAM J. of Computing*, 31(2):375–397, 2002.
7. A. Dasgupta, J. Hopcroft, J. Kleinberg, and M. Sandler. On learning mixtures of heavy-tailed distributions. In *Proceedings of the 46th IEEE FOCS*, pages 491–500, 2005.
8. S. Dasgupta. Learning mixtures of gaussians. In *Proceedings of the 40th IEEE Symposium on Foundations of Computer Science*, pages 634–644, 1999.
9. S. Dasgupta and L. J. Schulman. A two-round variant of em for gaussian mixtures. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI)*, 2000.
10. J. Feldman, R. O’Donnell, and R. Servedio. Learning mixtures of product distributions over discrete domains. In *Proceedings of the 46th IEEE FOCS*, 2005.
11. J. Feldman, R. O’Donnell, and R. Servedio. PAC learning mixtures of Gaussians with no separation assumption. In *Proceedings of the 19th Annual COLT*, 2006.
12. M. Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, pages 298–305, 1973.
13. P. Fjallstrom. Algorithms for graph partitioning: a survey. Technical report, Linkoping University Electroni Press, 1998.
14. Y. Freund and Y. Mansour. Estimating a mixture of two product distributions. In *Proceedings of the 12th Annual COLT*, pages 183–192, 1999.
15. R. Kannan, H. Salmasian, and S. Vempala. The spectral method for general mixture models. In *Proc. of the 18th Annual COLT*, 2005.
16. M. Kearns, Y. Mansour, D. Ron, R. Rubinfeld, R. Schapir, and L. Sellie. On the learnability of discrete distributions. In *Proceedings of the 26th ACM STOC*, pages 273–282, 1994.
17. R. Latala. Some estimates of norms of random matrices. In *Proceedings of the American Mathematical Society*, volume 133, pages 1273–1282, 2005.
18. Frank McSherry. Spectral partitioning of random graphs. In *Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science*, pages 529–537, 2001.
19. M. Meckes. Concentration of norms and eigenvalues of random matrices. *J. Funct. Anal.*, 211(2):508–524, 2004.
20. E. Mossel and S. Roch. Learning nonsingular phylogenies and hidden markov models. In *Proceedings of the 37th ACM STOC*, 2005.
21. J. K. Pritchard, M. Stephens, and P. Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155:954–959, June 2000.
22. D. Spielman. The behavior of algorithms in practice, 2002. Lecture notes.
23. V. Vempala and G. Wang. A spectral algorithm of learning mixtures of distributions. In *Proceedings of the 43rd IEEE FOCS*, pages 113–123, 2002.
24. V. Vu. Spectral norm of random matrices. In *Proceedings of 37th ACM STOC*, pages 423–430, 2005.
25. S. Zhou. *Routing, Disjoint Paths, and Classification*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, 2006. CMU Technical Report, CMU-PDL-06-109.