EXTENSIONS OF PRINCIPAL COMPONENTS ANALYSIS

A Thesis
Presented to
The Academic Faculty

by

S. Charles Brubaker

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Computer Science

Georgia Institute of Technology
August 2009

EXTENSIONS OF PRINCIPAL COMPONENTS ANALYSIS

Approved by:

Professor Santosh Vempala, Advisor
School of Computer Science
Georgia Institute of Technology

Professor Haesun Park
School of Computer Science
Georgia Institute of Technology

Professor Adam Kalai
School of Computer Science
Georgia Institute of Technology

Professor Vladimir Koltchinskii
School of Mathematics
Georgia Institute of Technology

Professor Ravi Kannan
Algorithms Research Group
Microsoft Research Labs., India

Date Approved: 17 June 2009

To my father

ACKNOWLEDGEMENTS

My journey through the PhD program has been a long one, taking me from robotics, to computer vision, to machine learning, and finally to theory. Making so many transitions slowed my graduation, no doubt, but it also allowed me to work with leaders across several research fields and has given me a better perspective on the research world. Ultimately, I am the better for it, and I am grateful to everyone I have had the opportunity to work with.

The most important man in my PhD career has been my advisor, Santosh Vempala, who brought out the best in me. Santosh's deep knowledge of theory and insight into my character allowed him to steer me toward the problems where I would enjoy myself and have success. Working with him closely has taught me how to think about math more intuitively and shown me how a truly great mind thinks. I will never forget his many acts of kindness and sometimes heroic efforts on my behalf. If I can emulate his advisement with my own students, I will consider myself a great success.

I also want to thank my two previous faculty advisors Frank Dellaert and Jim Rehg for the opportunity to work with them and for all that I learned under their guidance. Matt Mullin has also been a great resource on all matters mathematical. I am also obliged to some of the great teachers at Georgia Tech. A few that stand out in my memory are Vijay Vazirani, Eric Vigoda, Yan Ding, and Christopher Heil. My committee also deserves thanks for their feedback and thoughtful questions about my work.

Among my fellow graduate students, I am grateful to the members of the Theory Group and Wall Lab for their comradery through our shared struggles. I especially enjoyed my conversations with Jie Sun about our efforts at self-improvement and with

Raffay Hamid about deep questions outside of computer science. These conversations were often the highlight of my day.

Lastly, I want to thank my family. My parents more than anyone else cultivated the intellectual curiosity which has enriched my life and been a consolation through hard times. They and my wife Stephanie remained supportive throughout my PhD career. Stephanie has also been an excellent proofreader for all my papers. She and our two children, Charles and Sophie, have been remarkably patient through the late nights I spent in the lab and my bad humors. They mean the world to me.

TABLE OF CONTENTS

LIST OF FIGURES

SUMMARY

Principal Components Analysis is a standard tool in data analysis, widely used in data-rich fields such as computer vision, data mining, bioinformatics, and econometrics. For a set of vectors in $\mathbb{R}^n$ and a natural number $k < n$, the method returns a subspace of dimension k whose average squared distance to that set is as small as possible. Besides saving computation by reducing the dimension, projecting to this subspace can often reveal structure that was hidden in high dimension.

This thesis considers several novel extensions of PCA, which provably reveals hidden structure where standard PCA fails to do so. First, we consider Robust PCA, which prevents a few points, possibly corrupted by an adversary, from having a large effect on the analysis. The key idea is to alternate noise removal with projection to a constant fraction of the dimensions. When applied to noisy mixture models, the algorithm finds a subspace that is close to the pair of means that are furthest apart. By choosing and testing random directions in this subspace, the algorithm finds a partitioning hyperplane that does not cut any component and then recurses on the two resulting halfspaces. This strategy yields a learning algorithm for noisy mixtures of log-concave distributions that is only slightly weaker than the noiseless result (Chap. 5).

Second, we consider Isotropic PCA, which can go beyond the first two moments in identifying "interesting" directions in data. The algorithm first makes the distribution isotropic through an affine transformation. Then the algorithm reweights the data and computes the resulting first and second moments. In effect, this simulates a non-isotropic distribution, whose moments are sometimes meaningful. In the case of a mixture of Gaussians under a Gaussian reweighting, either the first moment or the

direction of maximum second moment can be used to partition the components of the mixture assuming that the components are sufficiently separated. This strategy leads to the first affine-invariant algorithm that can provably learn mixtures of Gaussians in high dimensions, improving significantly on known results (Chap. 6).

Thirdly, we define the "Subgraph Parity Tensor" of order r of a graph and reduce the problem of finding planted cliques in random graphs to the problem of finding the top principal component of this tensor (Chapter 7). This extends work by Frieze and Kannan, which considers only third order tensors. The intuition behind the result is that the entries in the block of the tensor corresponding to the clique will all have the same values, while the values in other blocks will be uncorrelated. This forces the top principal component of the tensor to "point" towards the clique. Using a previously known algorithm, the clique can be recovered.

CHAPTER I

INTRODUCTION

Intuitively, Principal Components Analysis (PCA) reveals in which directions a finite set of points is most stretched out. This concept is most familiar in the context of linear regression. Every high school student has plotted points in the plane and drawn a "best fit" line through them as part of a physics lab or math class. Similarly, for a set of points in three dimensions one can also imagine a best fit line, or by choosing an additional orthogonal direction, a best fit plane. In the language of PCA, after placing the origin at the mean of the points, the best fit line is the top principal component and the best fit plan is the span of the top two principal components. Analogously, for any point set in $\mathbb{R}^n$, we can define a top k principal components, whose span is a best fit k-dimensional subspace for the data.

These principal components are most easily characterized as eigenvectors of the covariance matrix. Thus, they can be defined for any distribution (not merely a finite point set) that has a bounded second moment. This type of eigenvector analysis plays an important role in algorithms for a broad set of problems, from the analysis of random graphs, to mixture models used in statistics, to applications such as data mining and computer vision. Often these algorithms using PCA are the best known. This thesis explores several extensions or modifications of PCA that produce provably better results than standard PCA for several problems.

1.1 A Brief History of PCA

If the essential concepts behind PCA are 1) that any distribution with bounded second moments should have a set of principal axes and 2) that these axes are revealed by eigenvectors of the covariance matrix, then PCA can be traced back as far as the

middle of the 18th century. In 1730, Leonhard Euler published his *Theoria motus corporum solidorum seu rigidorum*, which describes the motion of rigid bodies and introduces the idea of principal axes of rotation. A generation later, Lagrange recognized that these axes were the eigenvectors of the tensor of inertia, a close relative of the covariance matrix.[1] Replacing the rigid body (a uniform distribution with connected compact support) in $\mathbb{R}^3$ with a point set in $\mathbb{R}^n$ yields PCA.

Thoughout the nineteenth century, the ideas of principal axes and eigenvector decompositions proved fertile ground, particularly in the areas of quadric surfaces (Cauchy) and differential equations (Sturm-Louisville). It was not until the early twentieth century, however, that the idea was applied to data analysis in the work of Karl Pearson [42]. Pearson pointed out that in many practical applications the division between "independent" and "dependent" variables is arbitrary. For example, suppose that we measured the heights $\{h_i\}$ and leg lengths $\{\ell_i\}$ of a population and that we seek an affine relationship between the two quantities. In a traditional least squares approach, if we treat the heights as independent and leg lengths as dependent, then we obtain a function $\ell : \mathbb{R} \to \mathbb{R}$, mapping height to leg length. For pairs $\{(h_i, \ell_i)\}$, the least squares regression minimizes $\sum_i (\ell_i - \ell(h_i))^2$, taking no account for error in h_i. Conversely, if we switch the independent and dependent variables, then we obtain a function $h : \mathbb{R} \to \mathbb{R}$ which minimizes $\sum_i (h_i - h(\ell_i))^2$, taking no account for error in ℓ_i. Plotting h along the x-axis and ℓ along the y-axis, the first approach considers only vertical distances between a point and the "fit", while the second approach considers only horizontal distances. Pearson argues that the *minimum* distance between a point and the "fit" is the right quantity to consider. That is, letting $F = \{(x, y) \in \mathbb{R}^2 : ax + by + c = 0\}$ be a flat, we should choose the

[1] For a more detailed history see [27, 35].

flat that minimizes

$$\sum_i \min_{(x,y)\in F} (h_i - x)^2 + (\ell_i - y)^2.$$

As we will see in Chap. 2, this defines PCA.

It is worth noting that eigenvalues can be found by finding the roots of the characteristic polynomial of a matrix. Thus, it was practical to find eigenvalues for a matrix long before it was to find eigenvectors. The first numerical algorithm for finding eigenvectors of large matrices was the "power method", in which a random vector is multiplied repeated by the matrix until in converges. [2] Interestingly, the introduction of this algorithm predates the modern computer. The widely used QR algorithm was proposed independently by Francis [19] and Kublanovskaya [36] in 1961 .

1.2 Some Example Applications

In his 1901 paper, Pearson noted that calculation becomes "cumbersome if we have four, five, or more variables." Today, since the development of the modern computer and the fast algorithms for SVD, it has become practical to work with hundreds or thousands of variables. PCA has therefore become popular in fields like data mining, computer vision, econometrics, and psychometrics where there are a large number of variables and the relationships among them are not always clear.

Perhaps, the most common use is in data compression or simplification, in which a high dimensional data set is given a low dimensional representation. This is most effective when the data lies close to some k dimensional subspace (or flat). Let m be the number of samples and n the number of variables and let the columns of the n-by-k matrix V be the top k principal components. Then a data set stored in a n-by-m matrix M, can be summarized by the projection coefficients $C = V^T M$, a k-by-m matrix. The data can be approximately constructed from V and C as

[2] Credit for the algorithm usually goes to Von Mises (1929), though the idea of using a high power A^k appears in work by Muntz (1913) and Ostrowski and Werner Gautschi are the first to give a careful treatment. See [25] for more detailed chronology and sources.

$\hat{M} = VC = VV^T M$. Storing V and C, however, only requires $O(nk + km)$ space as opposed to $O(mn)$ space for the full data.

In a least squares sense, projecting to a PCA subspace changes the data as little as possible. Hence, one might expect that algorithms that run on the PCA coefficients instead of the original data might do *not too much worse*. In fact, however, it has often been observed that certain classification algorithms actually work *better*. Two of the most striking examples in computer science are latent semantic indexing and eigenfaces for face recognition.

In information retrieval and document analysis, documents are often represented according to the number of times certain words occur in them. If there are n such words, then each document is represented as a vector in $\mathbb{R}^n$ where the ith element is some function of the number of times word i occurs in the document. Typical tasks are retrieval ("find me documents similar to this one") and clustering ("organize these documents by topic"). For such methods to be effective, the notion of similarity or distance between documents need to correspond to our human understanding. A typical measure of similarity would be the correlation between to vectors. Interestingly, the application of PCA to a document corpus, called latent sematic indexing, goes a long way toward this end. This process of trading a large number of word counts for a small number of PCA coefficients has been shown to improve retrieval results [4, 15]. A theoretical analysis of this phenomenon based on a probabilistic model of the document corpus is given in [41].

A similar phenomenon has been observed in face recognition. Here we are given a corpus of individuals and sample images of each individual's face. Given a new image of one of these individuals' faces, we would like to identify the individual. Each image is represented as a vector in $\mathbb{R}^n$, where each coordinate reflects the intensity of a unique pixel. For this application, n is typically on the order of 10^5. It is intuitive, therefore, that the speed of retrieval would be improved if n could be replaced by a small number

of coefficients. In fact, however, a dramatic improvement in accuracy is also observed [46, 26]. In this case, PCA not only creates a more efficient representation, but it also has the effect of removing noise, while preserving the desired underlying signal. This same intuition underlies the application of PCA to learning mixture models.

1.3 Contributions

This thesis considers several novel extensions of PCA, which provably reveal hidden structure where standard PCA fails to do so. It presents two novel algorithms called Robust PCA and Isotropic PCA, which are outlined in Chap. 3, and applies them to learning mixture models. Robust PCA makes it possible learn noisy logconcave mixtures assuming only slightly more separation than necessary in the noiseless case. This work appeared in *Proceedings of the Symposium on Discrete Algorithms*, 2009 [5] and is presented in Chap. 5. Isotropic PCA yields an affine invariant method for learning well-separated mixtures of Gaussians. This is joint work with Santosh Vempala, appearing in *Building Bridges Between Mathematics and Computer Science* [6]. An earlier version also appeared at the *Symposium on Foundations of Computer Science*, 2008. The work is presented in Chap. 6.

The thesis also considers an extension of the idea of the top principal component to tensors (the analogue of matrices with more than two indices). Building on previous work of Frieze and Kannan [21], it shows that finding the top principal component of a tensor of order r makes in possible to find planted cliques of size $Cn^{1/r}$ in random graphs. This also is joint work with Santosh Vempala and it will appear in *Proceedings of the 13th International Workshop on Randomization and Computation*, 2009 [7]. It is presented in Chap. 7.

CHAPTER II

PRELIMINARIES

2.1 Standard PCA

For a collection of vectors $\{a_i\}_{i=1}^m$ in $\mathbb{R}^n$ such that $\sum_{i=1}^m a_i = 0$, the *principal components* are orthogonal vectors $v_1, \ldots v_n$ such that for every whole number $k < n$ the subspace $V_k = \text{span}\{v_1, \ldots, v_k\}$ minimizes

$$\sum_{i=1}^{m} d(a_i, V_k)^2, \tag{1}$$

where d is the distance between the point a_i and the subspace V_k, i.e.

$$d(a_i, V_k) = \inf_{x \in V_k} \|a_i - x\|.$$

The vectors $\{v_i\}$ are called the principal components, and the subset $v_1, \ldots, v_k$ are called the "top" k principal components.

Note that the requirement that the mean of the vectors $\{a_i\}$ be zero is trivial, since it can be achieved through a simple translation of the points. This process is called "centering the data." Taking V_0 to be the point at the origin, (1) holds for $k = 0$ as well. Thus, $V_0, \ldots, V_n$ form a set of nested subspaces, each differing from the next by the inclusion of one of the principal components.

These principal components can be found through the Singular Value Decomposition (SVD) of the m-by-n matrix A whose rows are the vectors a_i. Through SVD, the matrix A be written as a sum of rank-1 matrices

$$A = \sum_{i=1}^{\min\{m,n\}} \sigma_i u_i v_i^T,$$

where $\{\sigma_i\}$ non-negative reals, $\{u_i\}$ are orthogonal unit vectors in $\mathbb{R}^m$, and $\{v_i\}$ are the principal components of the rows of A.

Another equivalent characterization that is often useful is to say that the principal components are the eigenvectors of the symmetric matrix $M = A^T A/m$, i.e. the second moment matrix of the set $\{a_i\}$. When the origin is placed at the mean of the points, this becomes the eigenvectors of the covariance matrix. We can then characterize the principal components as

$$\begin{aligned} v_1 &= \arg\max_{v:\|v\|=1} v^T M v \\ v_2 &= \arg\max_{v:\|v\|=1, v_2 \perp v_1} v^T M v \\ &\vdots \\ v_n &= \arg\max_{v:\|v\|=1, v_n \perp v_1, \ldots, v_n \perp v_{n-1}} v^T M v \end{aligned} \tag{2}$$

This characterization of the top principal component can be extended to higher order tensors.

2.2 Generalization to Tensors

If vectors have one index, matrices have two, then for the purpose of this thesis "tensors" have more than two. We call the number of indices the order of the tensor. Although there is no clear analog to SVD or eigenvector analysis for tensors with order larger than two, we define the top principal component of a symmetric tensor A to be the unit vector that maximizes

$$A(x) = \sum_{k_1 \ldots k_r \in [n]^r} A_{k_1 \ldots k_r} x_{k_1} \ldots x_{k_r}. \tag{3}$$

This is the natural analogue to (2) and the maximum possible $A(x)$ defines the tensor norm.

CHAPTER III

EXTENSIONS OF MATRIX PCA

This section gives an exposition of Robust PCA and Isotropic PCA, illustrating them through some simple examples. Although these algorithms were originally developed for learning mixture models, they may be of broader interest and are presented independently here.

3.1 Robust PCA

Robust PCA addresses the problem of corrupted data. Given a set of points in $\mathbb{R}^n$ an adversary need only corrupt k data points to make the top k principal components of the corrupted data orthogonal to the top k components of the original data. This holds regardless of the number of data points. For instance, let $v_{n-k+1}, \ldots, v_n$ be the k smallest spectral components for a set of samples S. To this set add k noise points, $x_1 = cv_n, \ldots, x_k = cv_{n-k+1}$. For large values of c, the largest k principal components of $S \cup \{x_i\}_{i=1}^k$ will converge to $v_{n-k+1} \ldots v_n$, which are orthogonal to $v_1 \ldots v_k$.

The most immediate way to address this problem is to remove outliers. This is challenging in high dimensions because even random points sampled from small volumes tend to be far apart. Putting a ball around the uncorrupted data is not sufficient to preserve the relevent principal components.

We illustrate this challenge by way of example (see Fig. 1). Consider a set S of m points sampled uniformly from the stretched cube $\sqrt{6}[-1,1] \times \sqrt{3}[-1,1]^{n-1}$, which has variance 2 along the first coordinate and variance 1 along all others. Suppose that the point set is rotated in some arbitrary way, and we wish to recover the direction in which the cube is most stretched. Without corrupted data, the top principal component will point along this direction.

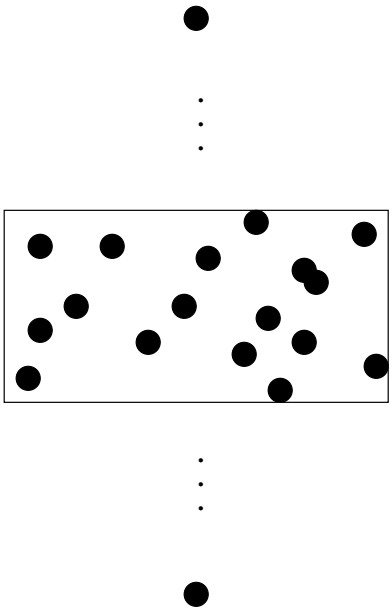


Figure 1: Samples from a rectangle mixed with malicious noise.

To the set S, add a set ϵm noise points at $\sqrt{n+1}e_2$ and an equal sized set at $-\sqrt{n+1}e_2$. Call these noise points N. Note that $E_S[\|x\|^2] = E_N[\|x\|^2] = n+1$–the points in N have exactly the same expected squared distance from the origin as points in S from the cube. This make the points difficult to distinguish. At least, we cannot simply put a ball around the uncorrupted data.

Despite not being outside the radius of the uncorrupted data, the noise points can have an unwanted effect on PCA. For just the points S sampled from the stretched cube, the top principle component should be along the long axis of the cube e_1. This can change with the addition of the noise points N. Ignoring the $(1+\epsilon)^{-1}$ factors, we have $E_{S\cup N}[x_1^2] = 2$ and, $E_{S\cup N}[x_2^2] = 1 + \epsilon(n+1)$. For ϵ as small as $1/n$, this means that

$$E_{S\cup N}[x_2^2] = 1 + \frac{n+1}{n} > 2 = E_{S\cup N}[x_1^2].$$

The addition of the points N has made e_2 the top principal component! By adding k more similar clusters of points along the coordinate axes $e_3 \ldots$, it is possible to produce the same effect on more coordinates, pushing e_1 to be the $(k+1)$th principal component for any $k < n$. Notice, however, that this requires that $\epsilon \geq k/n$.

The intuition behind the Robust PCA algorithm is that if ϵ is small, then it is "safe" to remove the bottom $n/2$ principal components (i.e. project to the top $n/2$ components). This projection has the effect of shrinking points in S toward the origin. After two such iterations, instead of having an expected squared norm of $n+1$, we now have $E_S[\|x\|^2] = n/4 + 1$ and no point will be as far as $\sqrt{3(n/4+1)}$ from the origin. Since $\sqrt{3(n/4+1)} < \sqrt{n+1}$, it is now possible to put a ball around the uncorrupted data and remove the noise points.

Algorithm 1 Robust PCA

Input:
1) Collection $\{Z_i\}$ of $\lceil \log_2 n \rceil$ sets of points in $\mathbb{R}^n$.
2) Integers k, r, scalar ξ.

Output: A subspace W of dimension k.

1. Let $W = \mathbb{R}^n$.
2. While $\dim(W) > k$,
   (a) Let $Z = \text{proj}_W(Z_i)$, where Z_i is the next set of samples.
   (b) For every $p \in Z$ find the point $q(p)$, defined to be rth furthest away point.
   (c) Find the point p_0 such that the distance $\|p_0 - q(p_0)\|$ is the rth largest distance in the set $\{\|p - q(p)\| : p \in Z\}$. Let $q_0 = q(p_0)$ and let $t(Z) = \|p_0 - q_0\|$.
   (d) Let $Z' = Z \cap B(p_0, \xi t(Z))$.
   (e) Let W be the span of the top $\lfloor (\dim(W) - k)/2 \rfloor + k$ eigenvectors of the matrix $\sum_{p \in Z'} (p - p_0)(p - p_0)^T$.

The remaining challenge is to put a ball around the good data. The key assumption is that there is some integer $r < |S|/2$ and real ξ such that in every iteration

1. r is larger than the number of corrupt points, and
2. within the uncorrupted set there are r points whose distance to the rth furthest away point is within a $2/\xi$ factor of the maximum distance between points.

For a point $p \in S \cup N$, let $q(p)$ be the rth furthest away point and let t be the rth

largest element in the set $\{\|p - q(p)\| : p \in S \cup N\}$. Let p_0 be a point such that $\|p_0 - q(p_0)\| = t$. The above conditions guarantee that

$$t \leq \max_{p,q \in S} \|p - q\| \leq \frac{\xi}{2} t.$$

Hence, for any $p \in S$, the ball $B(p, \xi/2 \cdot t) \supseteq S$. Unfortunately, we do not know that $p_0 \in S$, but we do know that there is some point $q_0 \in B(p, t) \cap S$. Therefore, $B(p, \xi t) \supseteq S$.

Thus, the algorithm Robust PCA (Alg. 1) uses the strategy of alternately denoising and projecting to the top $n/2$ components, until the dimension is reduced to the desired k.

3.2 Isotropic PCA

We now turn to another extension of PCA, which goes beyond the first and second moments to identify "important" directions. When the covariance matrix of the input (distribution or point set in R^n) is a multiple of the identity, then PCA reveals no information; the second moment along any direction is the same. Such inputs are called isotropic. Our extension, which we call *Isotropic PCA*, can reveal interesting information in such settings.

To illustrate the technique, consider the uniform distribution on the set $X = \{(x, y) \in \mathbb{R}^2 : x \in \{-1, 1\}, y \in [-\sqrt{3}, \sqrt{3}]\}$, which is isotropic. Suppose this distribution is rotated in an unknown way and that we would like to recover the original x and y axes. For each point in a sample, we may project it to the unit circle and compute the covariance matrix of the resulting point set. The x direction will correspond to the top principal component, the y direction to the other. See Figure 3.2 for an illustration. Instead of projection onto the unit circle, this process may also be thought of as importance weighting, a technique which allows for the simulation of one distribution by using another. In this case, we are simulating a distribution over

the set X, where the density function is proportional to $(1+y^2)^{-1}$, so that points near $(1,0)$ or $(-1,0)$ are more probable.

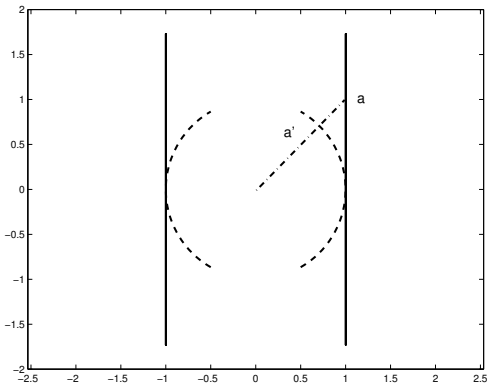


Figure 2: Mapping points to the unit circle and then finding the direction of maximum variance reveals the orientation of this isotropic distribution.

Algorithm 2 Isotropic PCA

Input:
1) A set X of points in $\mathbb{R}^n$.
2) Integer k, function $f : \mathbb{R} \to \mathbb{R}$

Output: Vectors u and $v_1, \ldots, v_k$.

1. Find the affine transformation that makes the point set X isotropic. Call the resulting points $\tilde{X}$.
2. Compute $u = \frac{1}{|X|} \sum_{x \in \tilde{X}} x f(\|x\|)$.
3. Compute $M = \frac{1}{|X|} \sum_{x \in \tilde{X}} x x^T f(\|x\|)$ and let $v_1, \ldots, v_k$ be its top k principal components.

This general approach of 1) making a point set isotropic, 2) reweighting, and 3) finding the resulting moments is outlined in Algorithm 2, which we call Isotropic PCA.

3.3 Future Directions

Currently, the guarantees about how Isotropic PCA and Robust PCA work apply only to learning mixtures of Gaussians and log-concave distributions respectively (see

Chapters 5 and 6). The ideal would be to prove as general a theorem as possible about the utility of these algorithms. For Isotropic PCA, this might mean characterizing all distributions for which the Unravel algorithm works. For Robust PCA, it might mean a claim about how close the Robust PCA subspace is to the PCA subspace with the noise removed. Short of these goal, finding new applications for the methods could make them interesting and useful to a wider group of researchers.

CHAPTER IV

MIXTURE MODELS

One of the most common modeling assumptions for data is that it is sampled from a mixture of known distributions. For instance, consider the set of newspaper articles appearing the in *The Washington Post* over the last year. A mixture model might approximate the true writing process as follows. First, the editor chooses a topic according to some random process, choosing topic i with probability w_i. Then the journalist writes the article according to some random process. If each article is represented as a vector in $\mathbb{R}^n$ (e.g. each coordinate might be the number of times a particular word appears), then this process induces a "mixture distribution" over $\mathbb{R}^n$

$$F = w_1 F_1 + \ldots + w_k F_k,$$

where F_i is the distribution given that topic i was assigned. Although mixture modeling is usually a poor approximation to what happens in the real world, it is often good enough to enable automated tasks such as document search ("find articles on the same topic as this one") or computational simplifications such as vector quantization (used in speech recognition, for example). Note that in these applications the point in $\mathbb{R}^n$ (the article) is known, but the component identifier (the topic) is not.

Despite their widespread use, little was known about when such mixture distributions are learnable until recently. Given samples from a mixture distribution, can the parameters of the distributions be recovered? The most obvious solution to learning these parameters is to figure out which samples came from which distribution and then learn the distributions individually. This partition of the data points is called clustering. The classical methods of "K-means Clustering" [39] and "Expectation-Maximization," [13], however, are local search methods and tend to become stuck in

suboptimal classifications.

4.1 Learning Logconcave Mixture Models

Learning mixture models was popularized in the learning theory community by S. Dasgupta and Shulman [11, 12], beginning with mixtures of spherical Gaussians. The basic intuition for this early work was that points from the same component should be closer to each other than points from different components. If a point x is from component i and y is from component j, then the squared distance between them is roughly

$$\|x - y\|^2 = n(\sigma_i^2 + \sigma_j^2) + \|\mu_i - \mu_j\|^2 \pm C\sqrt{n}(\sigma_i + \sigma_j),$$

where μ_i is the mean of component i and σ_i^2 is the variance along a single direction for component i. To cluster, it is sufficient that the distance between the means dominate the variability in the distances (i.e. $C\sqrt{n}(\sigma_i + \sigma_j)$). Therefore, the separation requirement is that for every pair of components i, j,

$$\|\mu_i - \mu_j\| \geq n^{1/4}(\sigma_i + \sigma_j)\text{poly}(\log n).$$

Arora and Kannan [3] gave a more general notion of distance that handles a wider variety of cases, including the case where one Gaussian is "inside" another because of a large difference in their variances. For spherical Gaussians of similar radius, however, their result is comparable.

A major breakthrough in provable clustering results came from understanding the effect of applying PCA to the data [48, 29, 1]. If the means of the components of the mixture are reasonably well separated compared to the directional variances, then the principal components of the sample points will be close to the span of the means. Thus, projecting to the top k components preserves the separation between the components, while igoring other dimensions. If the distance between the means

is the signal, then projection to the PCA subspace reduces the noise by removing the orthogonal dimensions.

This idea first appears in work by Vempala and Wang [48], which shows only that

$$\|\mu_i - \mu_j\| \geq k^{1/4}(\sigma_i + \sigma_j)\text{poly}(\log n)$$

separation for k components. This work was later extended by Kannan, Salmasian, and Vempala [29] to mixtures of logconcave densities that are not necessarily isotropic. Achlioptas and McSherry [1] show similar results and explore the minimum necessary separation for clustering to be possible. In these works, the required separation is

$$\|\mu_i - \mu_j\| \geq (k^{3/2} + {w_{\min}}^{-1/2})(\sigma_i + \sigma_j)\text{poly}(\log n).$$

where $\sigma^2_{i,\max}$ is the maximum variance of the ith component in any direction.

A major shortcoming of this line of work is that large variances in directions orthogonal to the span of the means can cause the method to fail. In fact, every mixture where these algorithms work can be transformed to one where they fail by an affine transformation. The "Unravel" algorithm presented in [6] and summarized in Chap. 6 overcomes this shortcoming by going beyond first and second moments in identifying good separating directions for the clusters.

Another shortcoming of traditional PCA is that it can be subject to corruption of a few points. For instance, a single point can dramatically affect the top principal component. The Robust PCA algorithm presented in [5] and summarized in Chap. 5 is robust to this kind of noise and yields an algorithm for learning mixtures of log-concave distributions.

4.2 Learning Axis-Aligned Mixtures

A related area of work is on learning product distributions, where the coordinates are independent (e.g. a Gaussian would be axis-aligned). Here the goal is not necessarily to cluster data but to approximate the density of the mixture. Freund and Mansour

[20] first solved this problem for a mixture of two distributions of binary vectors , finding a model that approximates the true distribution in terms of Kullback-Leibler distance. Feldman and O'Donnell [17] extended this result to mixtures of any constant number of components and to discrete domains instead of binary vectors, i.e. $\{0, \ldots, b-1\}^n$ instead of $\{0,1\}^n$. Joined by Servedio in [18], they applied their technique to mixtures of a constant number of axis-aligned Gaussians, showing that they can be approximated without any separation assumption at all.

Another class of results on learning product distributions uses separation conditions which assume that the component centers be separated along many directions. Chaudhuri and Rao [9] note that results such as [29],[1] and [6] have a polynomial dependence on the inverse of the minimum mixing weight and reduce this to a logarithmic dependence by exploiting the independence of the coordinates. Beyond log-concave distributions, A. Dasgupta et al [10] consider a class of heavy-tailed product distributions and give separation conditions under which such distributions can be learned using an algorithm that is exponential in the number of samples. Chaudhuri and Rao [8] have recently given a polynomial algorithm to learn a related class of heavy-tailed product distributions.

CHAPTER V

CLUSTERING ON NOISY MIXTURES

5.1 Introduction

We consider the problem of learning a mixture from samples where the data includes some small miscellaneous component in addition to a well-behaved mixture. Equivalently, we may say that the sampling process for the well-behaved mixture has some noise, whereby with some small probability a point is replaced by a noise point about which can make no assumptions. The practical importance of robustness to the presence of noise should be apparent to anyone who has tried to file his bills, organize a closet, or set up a directory structure on his hard disk. Some things just don't belong to any large category. The presence of these "noisy" objects does not usually impede our ability to cluster or classify objects, suggesting that we should hold our algorithms to this standard as well. More concretely, in tasks such as document or web-page clustering it is unreasonable to assume that components will be well-separated with *absolutely nothing* in-between.

In our model, we assume that with probability at least $1 - \epsilon$ the sample source outputs a point from a mixture of log-concave distributions, but with the remaining probability it outputs a point about which we can make no assumptions. We call such a sample source ϵ-noisy. This is the natural analog to the malicious error models of [32, 47] for the clustering problem. Because the noise component is arbitrary, it may not be possible to cluster in the traditional sense. Indeed all noise points could be identical to one of the non-noise points, making them indistinguishable. Therefore, we set a different goal. Suppose the data set can be written as the disjoint union $S_1 \cup \ldots \cup S_k \cup N$, where S_i corresponds to the set of points from component i and

N to the set of noise points. Then we seek a collection of disjoint sets $C_1 \dots C_k$ such that for every S_i, there is a unique C_i where

$$S_i \subseteq C_i \subseteq S_i \cup N. \tag{4}$$

Although the sets $\{C_i\}$ may include some noise points, they induce a correct partition of the non-noise points $\{S_i\}$.

We present a polynomial time algorithm that given a noisy mixture of well-separated, logconcave distributions in $\mathbb{R}^n$, learns to separate the components of the mixture. That is, the algorithm finds a partition of $\mathbb{R}^n$ into k sets with disjoint interiors each of which contains almost all of the probability mass of a unique component of the mixture. The error of such a partition is the total mass that falls outside of the correct set. As a corollary, this algorithm makes it possible to cluster points from a noisy source in the sense of (4). The separation between the means necessary for the algorithm's success is only an $O^*(\log n)$ factor larger than the best analogous results without noise, treating k and $w_{\min}$ as constants.

The input to our algorithm is a source of samples LM, a natural number k and a scalar $w_{\min}$. The quantity k is the number of non-noise components in the mixture and $w_{\min}$ is a lower bound on the minimum mixing weight. For simplicity of the exposition, we state the results in terms of learning a partition of $\mathbb{R}^n$ (i.e. a classifier). This can easily be turned into a statement strictly about clustering a set of points through a slight modification of the algorithm.[1]

For a component i of the mixture, we define the following quantities : $\mu_i = E[x]$, $R_i^2 = E[\|x - \mu_i\|^2]$, and $\sigma_i^2 = \max_{v \in \mathbb{R}^n} E[(v \cdot (x - \mu_i))^2]/\|v\|^2$. We define the mixing weights w_i to be the probability that LM outputs a sample from component i. Thus, if LM outputs noise with probability ξ, then $\sum_{i=1}^k w_i = 1 - \xi$, effectively treating ξ

[1]For instance, we might divide the given points into overlapping blocks and cluster each block using the remaining points to simulate the sample source. The overlap of the blocks can be used to calculate to appropriate permutation of component indices for each block and thus obtain a clustering of the whole set.

as a mixing weight itself. We let $w_{\min}$ be the minimum mixing weight.

Our main result is summarized in the following theorem.

Theorem 1. *Let $\mathcal{F}$ be a mixture of k logconcave distributions with means $\{\mu_i\}$ and maximum variances $\{\sigma_i\}$. Let $\delta, \eta > 0$. There exist $\epsilon = \Omega(w_{\min} \log^{-2}(nk/(w_{\min}\delta\eta))$ and $\alpha = O(k^{3/2} {w_{\min}}^{-1} \log(nk/(w_{\min}\delta\eta)))$ such that if* LM *is an ϵ-noisy sample source for $\mathcal{F}$ and if for every pair of components i, j*

$$\|\mu_i - \mu_j\| \geq \alpha(\sigma_i + \sigma_j), \tag{5}$$

then the following holds. There is a polynomial algorithm that given access to at least $O(nk {w_{\min}}^{-1} \log^6(nk/\delta))$ samples from LM *with probability $1 - \delta$ returns a partition of $\mathbb{R}^n$ that correctly classifies a point from $\mathcal{F}$ with probability $1 - \eta$.*

5.2 A Robust Clustering Algorithm

In previous work [29], the approach is to first project the data onto its top k spectral components, then extract a single cluster, and repeat. This strategy succeeds because the projection onto the top k components preserves much of the intermean distances, while reducing the pairwise distance between points of the same component. The concentration of the pairwise distances is then exploited to remove a component.

In the presence of noise, however, this approach breaks down. In fact, only k well-chosen noise points are required to cause the intermean distances to become arbitrarily small after projection. To cope with this problem we first remove outliers. That is, we reduce the maximum distance between any two points to be $O(R_{\max} + \mu_{\max})$. Projection to the top k components still may not preserve the necessary intermean distances, but projection to the top $\lfloor (n-k)/2 \rfloor + k$ components will. By repeating this procedure through the Robust PCA algorithm (see Alg. 1), we reduce the dimension to k.

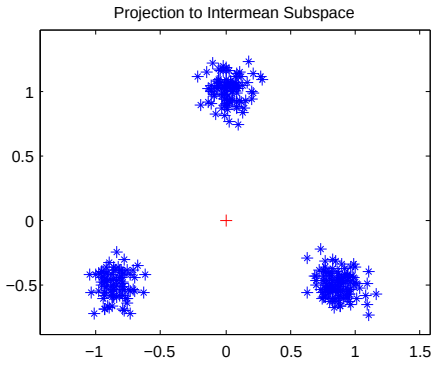
Robust PCA will preserve enough of the distance between the components whose

means are furthest apart so that the direction between their means can be approximated by a pair of samples, one coming from each component. Imagine projecting the entire mixture density onto this line. The concentration of the individual components implies that this density will be multimodal with large peaks and long flat valleys. By setting a threshold in the middle of a valley, we define a hyperplane that separates the components of the mixture. To determine the appropriate bucket width for the density estimation, we use the quantity $t(X)$ (defined line 2c of Robust PCA), which approximates the distance between the two furthest non-noise points.

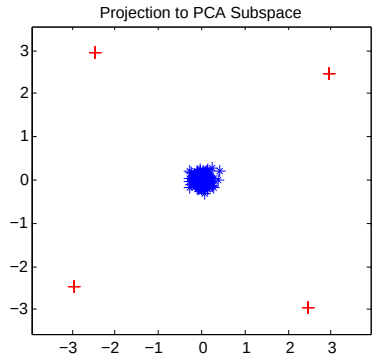
We then recurse on the two half-spaces defined by this hyperplane. At lower levels of the recursion tree, we are recursing on the intersection of these hyperplanes, i.e. a polyhedron. Ideally, we would like to recurse on a submixture, i.e. a subset of the original mixture's components. Fortunately, each component is far enough away from the support hyperplanes that the probability that a sample will appear on one side of a hyperplane while its component mean is on the other is vanishingly small. This enables us to simulate the desired submixture by rejecting samples until one from the correct part of $\mathbb{R}^n$ is obtained.

5.3 Empirical Illustrations

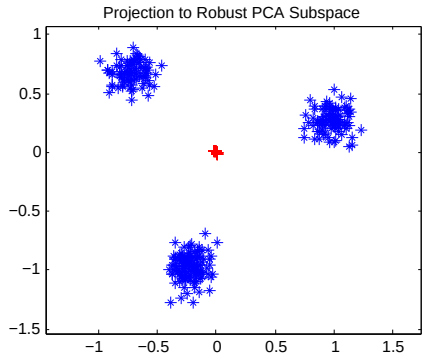
A Matlab implementation of the algorithm verifies the usefulness of Robust PCA, through an example shown in Fig. 3. Noise points (pictured by red '+' signs) were added to a set of points sampled from a mixture of three spherical Gaussians. The noise points make the top two principal components orthogonal to the intermean subspace, so that the PCA subspace collapses the components(3b). However, Robust PCA, is able to find the correct subspace by alternating denoising and projection (3). Note that in the example used for the figure, many noise points have the same location. A Matlab implementation of Robust PCA is printed in Appendix C.



(a) Intermean Subspace



(b) PCA Subspace



(c) Robust PCA Subspace

Figure 3: The PCA subspace does not preserve the separation of the mixtures, because the noise term dominates. Robust PCA, however, approximates the intermean subspace, making it possible to cluster.

Algorithm 3 Cluster Noisy Logconcave Mixture

Input:
1) Sampling source LM which generates point in $\mathbb{R}^n$.
2) Integer k, reals $\epsilon, w_{\min}$.
3) Polyhedron P. (Note $P = \mathbb{R}^n$ in the initial call.)

Output: A collection of k polyhedra.

1. For $i = 1$ to $\lceil \log n \rceil$ let Z_i a set m_Z points from LM.
2. Let W be the subspace returned by Robust PCA for the collection $\{Z_i \cap P\}$, $\xi = 16\beta$ and $r = \lfloor 2\epsilon m_Z \rfloor$.
3. Let $X = \text{proj}_W(X_0 \cap P)$, where X_0 is a set of m_X samples obtained from LM.
4. Let $d = t(X)/10k$ (Note t is defined in (6) and line 2c of Robust PCA)
5. Let $Y = \text{proj}_W(Y_0)$, where Y_0 is a set of m_Y samples from LM.
6. For every $(a, b) \in Y \times Y$
   (a) Let $v = (a - b)/\|a - b\|$.
   (b) Let $b_i = |\{x \in X : \text{proj}_v(x) \in [id, (i + 1)d)\}|$.
   (c) If there is a triple $i_1 < i_2 < i_3$ where $b_{i_1}, b_{i_3} > w_{\min} m_X/4$ and $b_{i_2} \leq 2\epsilon m_X$, then let $\gamma = (i_2+1/2)d$ and recurse with $P = P \cap H_{v,\gamma}$ and $P = P \cap H_{-v,-\gamma}$. Return the collection of polyhedra produced by these calls.
7. Return P.

5.4 Preliminaries

In our analysis, we will decompose a set Z obtained from the sample source LM into $S \cup N$ where S consists of the points drawn from $\mathcal{F}$ and N consists of the noise points. Further, we decompose the set S into $S_1 \cup \ldots \cup S_k$, where S_i consists of the points drawn from component i. For a point $p \in S$, we use $\ell(p)$ to denote the component from which p was drawn. We also use $\hat{\mu}_i$ to indicate the average of points from component i in a set. For a subspace W and polyhedron P, it will be convenient to define the following quantities. Let $I_P = \{i : \mu_i \in P\}$ and let $\mathcal{F}_P$ be the submixture

consisting of the components in I_P. Let

$$\begin{aligned} R_i^{(W)} &= E_i\left[\|\mathrm{proj}_W(x-\mu_i)\|^2\right]^{1/2} \\ R_{\max}^{(W,P)} &= \max_{i\in I_P} R_i^{(W)}. \\ \mu_{\max}^{(W,P)} &= \max_{i,j\in I_P} \|\mathrm{proj}_W(\mu_i-\mu_j)\| \\ {\sigma_{\max}}^{(P)} &= \max_{i\in I_p} \sigma_i. \end{aligned}$$

Note that E_i denotes an expectation with respect to the ith component of the mixture. When the superscript W is omitted, it may be assumed that $\mathbb{R}^n$ is meant. The polyhedron P is often clear from context and may be omitted as well.

Throughout the analysis we use the fact that the lower bound on the separation α is $\Theta(k^{3/2}{w_{\min}}^{-1}\log(nk/(w_{\min}\delta\eta)))$ and the upper bound on the noise ϵ is $\Theta(w_{\min}\log^{-2}(nk/(w_{\min}\delta\eta)))$.

5.4.1 Safe Polyhedra

The success of the algorithm depends on the fact that intersecting the sample set from LM with the polyhedron P in steps 2 and 3, of Algorithm 3 effectively simulates sampling from the submixture $\mathcal{F}_P$. That is, this intersection has the effect of including all points from components in I_P and excluding all points from other components. This motivates the following definition.

Definition 1. *A polyhedron P is* η-safe *for a mixture $\mathcal{F}$ if*

1. *For every $i \in I_P$, we have $\mathsf{P}[x \notin P] \leq \eta$, where x is a random point from component i.*

2. *For every $i \notin I_P$, we have $\mathsf{P}[x \in P] \leq \eta$, where x is a random point from component i.*

The concentration of logconcave distributions yields a simple criterion for showing that a halfspace is safe. We use the following theorem from [38].

Theorem 2. *Let* $R^2 = \max_{\|v\|=1} E[(v \cdot (x - \mu))^2]$ *for a random variable* x *from a logconcave distribution. Then*

$$\mathsf{P}(\|x - \mu\| > tR) < e^{-t+1}.$$

Restricting this to a single dimension gives the following corollary.

Corollary 1. *Let* $H_{v,\gamma} = \{x \in \mathbb{R}^n : v \cdot x \geq \gamma\}$ *be a halfspace in* $\mathbb{R}^n$*. For every* $\eta > 0$*, there is a factor* $\beta_{safe} = O(\log 1/\eta)$ *such that if for every component* i *in a logconcave mixture* $\mathcal{F}$*,*

$$|v \cdot \mu_i - \gamma| > \beta_{safe}\sigma_i,$$

then $H_{v,\gamma}$ *is* η*-safe for* $\mathcal{F}$*.*

Halfspaces are then easily combined into polyhedra.

Proposition 2. *If* P_1 *is* η_1*-safe for* $\mathcal{F}$ *and* P_2 *is* η_2*-safe for* $\mathcal{F}_{P_1}$*, then* $P_1 \cap P_2$ *is* $(\eta_1 + \eta_2)$*-safe for* $\mathcal{F}$*.*

Proof. Suppose component $i \in I_{P_1 \cap P_2}$ and let x be distributed according to component i. Then

$$\mathsf{P}[x \notin P_1 \cap P_2] \leq \mathsf{P}[x \notin P_1] + \mathsf{P}[x \notin P_2] \leq \eta_1 + \eta_2.$$

Now, suppose component $i \notin I_{P_1 \cap P_2}$. We distinguish two cases. If $i \notin I_{P_1}$, then

$$\mathsf{P}[x \in P_1 \cap P_2] \leq \mathsf{P}[x \in P_1] \leq \eta_1.$$

On the other hand, if $i \in I_{P_1 \setminus P_2}$, then

$$\mathsf{P}[x \in P_1 \cap P_2] \leq \mathsf{P}[x \in P_2] \leq \eta_2.$$

□

5.4.2 Properties of Sample Sets

As we will argue, the polyhedra obtained by the algorithm will be safe. Therefore, we expect that the polyhedra will contain the points from the components whose means are contained in the polyhedra. We also expect that no set chosen in steps 1 or 3 of Algorithm 3 will contain much more than its share of noise points and that the empirical means and variances will be close to those of the component distributions themselves. Our analysis rests on these sets obtained from LM in steps 1 and 3 having these and other key properties that are summarized in the following definition.

Definition 2. *A set $S_1 \cup \ldots \cup S_n \cup N$ of m points from* LM *is* good *for subspace W, polyhedron P, and scalar β if the following conditions hold.*

1. *For every component i, if $\mu_i \in P$, then $S_i \subseteq P$, and if $\mu_i \notin P$, then $S_i \cap P = \emptyset$.*
2. *$|S_i| \geq w_i m/2$ for all components i, and $|N| \leq 2\epsilon m$.*
3. *For every component $i \in I_P$ and every $p \in S_i$, $\|\text{proj}_W(p - \mu_i)\| \leq \beta R_i^{(W)}$*
4. *For every component $i \in I_P$ $\|\text{proj}_W(\mu_i - \hat{\mu}_i)\| \leq \frac{\sigma_i}{4}$.*
5. *For every component $i \in I_P$*

$$\frac{7}{8}R_i^{(W)} \leq \frac{1}{|S_i|}\sum_{p \in S_i} \|\text{proj}_W(p - \hat{\mu}_i)\|^2 \leq \frac{8}{7}R_i^{(W)}.$$

6. *For some pair $i, j \in I_P$ such that $\|\text{proj}_W(\mu_i - \mu_j)\| = \mu_{\max}^{(W,P)}$, it holds for all $p \in S_i \cup S_j$, that $\|\text{proj}_u(p - \mu_{\ell(p)})\| \leq \beta {\sigma_{\max}}^{(P)}$, where u is the unit vector along the direction* $\text{proj}_W(\mu_i - \mu_j)$.

For convenience, we will sometimes say the set $Z = \text{proj}_W(Z_0 \cap P)$ is "generated by a good set for W,P, and β." This is not really a property of the set Z itself, but rather of W,P, β and an implicit Z_0 (drawn from LM) such that $Z = \text{proj}_W(Z_0 \cap P)$.

It is important to note that a set is only good *for a particular subspace*. In our analysis on Robust PCA, we will require that Z_i from step 1 of Algorithm 3 be good for polyhedron P and the current subspace W in step 2a of Robust PCA, where $Z_i \cap P$ is used. Thus, Z_1 must be good for $\mathbb{R}^n$ and Z_2 must be good for the subspace obtained after one iteration in Robust PCA, etc. Finally, the set X_0 used in step 3 of Algorithm 3 must be good for W_k the subspace returned by Robust PCA. The following lemma shows that this happens with high probability.

Lemma 3. *Let Z_0 be a set of m points generated by ϵ-noisy sample source* LM *for a logconcave mixture. Let P be polyhedron that is $(\delta/2m)$-safe. Let W be any subspace of $\mathbb{R}^n$. There exist $M_{good} = O(n/w_{\min} \log^5 nk/\delta)$ and $\beta_{good} = O(\log(mk/\delta))$ such that with probability $1-\delta$ if $m \geq M_{good}$, then Z_0 is good for W, P, and any $\beta \geq \beta_{good}$.*

Proof. We consider the goodness properties in order. From the definition of η-safe, item 1 holds with probability $1 - \delta/2$. Item 2 follows from a Chernoff bound (recall that ϵ is an upper bound on the noise of LM and not the noise itself). The remaining items are standard results for logconcave distributions. See [29]. □

5.4.3 Bounds on t

It is important that we be able to approximate the greatest distance between two non-noise points. This enables us to put a ball around the non-noise data in Robust PCA so as to remove noise points that are far away from the non-noise points (step 2d of Robust PCA). It is also critical in determining d the resolution at which we look for valleys in the partitioning phase (step 4 of the clustering algorithm).

Lemma 4. *Suppose that $Z = S \cup N$ was generated by a good set from* LM *for W, P, and β. Then $t = t(Z)$ has the bounds*

$$\max\{\mu_{\max}^{(W,P)} - 2\beta\sigma_{\max}{}^{(P)}, R_{\max}^{(W,P)}/2\} \leq t \leq \max_{p,q \in S} \|p - q\| \leq \mu_{\max}^{(W,P)} + 2\beta R_{\max}^{(W,P)}. \quad (6)$$

Proof. By definition $Z = \mathrm{proj}_W(Z_0 \cap P)$, where Z_0 is go for P, W and β. For convenience, we partition Z into the non-noise points S and the noise points N, so that $Z = S \cup N$. To avoid cumbersome notation, we will drop the superscript W and P for the quantities $\mu_{\max}$ and $R_{\max}$. For the purpose of this proof, it will also be convenient to introduce the following notation. For a finite set $T \subset \mathbb{R}$, let sT denote the rth largest number in T. Thus,

$$t = s\{s\{\|p-q\|\}_{q\in S\cup N}\}_{p\in S\cup N}.$$

To obtain the upper bound on t, observe that for any $p \in S \cup N$,

$$s\{\|p-q\|\}_{q\in S\cup N} \le \max_{q\in S} \|p-q\|,$$

since there are at most $2\epsilon m$ elements in N. Similarly,

$$t \le s\{\max_{q\in S} \|p-q\|\}_{p\in S\cup N} \le \max_{p,q\in S} \|p-q\|.$$

But for any pair of points $p, q \in S$,

$$\|p-q\| \le \|p - \mathrm{proj}_W(\mu_{\ell(p)})\| + \|\mathrm{proj}_W(\mu_{\ell(p)} - \mu_{\ell(q)})\| + \|\mathrm{proj}_W(\mu_{\ell(q)}) - q\|$$

by the triangle inequality, where $\ell(p)$ is the index of the component from which p was drawn. Using the definition of "good" (Definition 2, item 3), we have that the first and last terms are bounded by $\beta R_{\max}$. Combining this with the definition of $\mu_{\max}$, we have

$$t \le \max_{p,q\in S} \|p-q\| \le \mu_{\max} + 2\beta R_{\max}.$$

Next we give a lower bound in terms of $\mu_{\max}$. For a pair of components i and j

$$\begin{aligned} t &= s\{s\{\|p-q\|\}_{q\in S\cup N}\}_{p\in S\cup N} \\ &\ge s\{s\{\|p-q\|\}_{q\in S_j}\}_{p\in S_i}. \end{aligned}$$

Note that these quantities are well defined, since $|S_j|, |S_i| > w_{\min} m/2 > \lfloor 2\epsilon m \rfloor$ by item 2 of Definition 2 and our choice of ϵ. We continue

$$\begin{aligned} s\{s\{\|p-q\|\}_{q\in S_j}\}_{p\in S_i} &\geq s\{\min_{q\in S_j} \|p-q\|\}_{p\in S_i} \\ &\geq \min_{p\in S_i, q\in S_j} \|p-q\|. \end{aligned}$$

Now suppose that i and j are the two components such that $\|\text{proj}_W(\mu_i - \mu_j)\| = \mu_{\max}$ and $\mu_i, \mu_j \in P$. Let u be the unit vector along the direction $\text{proj}_W(u_i - u_j)$. Then for any $p \in S_i$, $q \in S_j$ that are closest we have from the triangle inequality that

$$\begin{aligned} \|p-q\| &\geq |\text{proj}_u(p-q)| \\ &\geq |\text{proj}_u(\mu_i - \mu_j)| - |\text{proj}_u(\mu_i - p)| - |\text{proj}_u(\mu_j - q)|. \end{aligned}$$

Using the definition of "good" again (Definition 2, item 6), have that $|\text{proj}_u(\mu_i - p)|$ and $|\text{proj}_u(\mu_j - q)|$ are at most $\beta\sigma_{\max}$. At the same time, by construction $|\text{proj}_u(\mu_i - \mu_j)| = \mu_{\max}$. Thus,

$$t \geq \min_{p\in S_i, q\in S_j} \|p-q\| \geq \mu_{\max} - 2\beta\sigma_{\max}.$$

To give a lower bound in terms of $R_{\max}$, let i be a component such that $R_{\max} = S_i$. Then for any $p \in R_i$, by item 5 of Definition 2

$$\begin{aligned} \frac{7}{8} R_{\max} &\leq \frac{1}{|S_i|} \sum_{q\in S_i} \|\text{proj}_W(\hat{\mu_i}) - q\|^2 \\ &\leq \frac{1}{|S_i|} \sum_{q\in S_i} \|p-q\|^2 \\ &\leq s\{\|p-q\|^2\}_{q\in S_i} + \frac{2\epsilon m}{|S_i|} \max_{q\in S_i} \|p-q\|^2. \end{aligned}$$

Applying items 2 and 3 of Definition 2 to the last term, we have that $2\epsilon m/|S_i| \leq 4\epsilon m/w_{\min}$ and $\max_{q\in S_i} \|p-q\|^2 \leq 4\beta^2 R_{\max}^2$. Thus,

$$\frac{7}{8} R_{\max} \leq s\{\|p-q\|^2\}_{q\in S_i} + \frac{4\epsilon}{w_{\min}} \left(4\beta^2 R_{\max}^2\right)$$

Rearranging this yields,

$$s\{\|p-q\|^2\}_{q\in S_i} \geq \frac{7}{8} R_{\max} - \frac{4\epsilon}{w_{\min}} \left(4\beta^2 R_{\max}^2\right).$$

For an appropriate choice of $\epsilon = C_\epsilon w_{\min} \beta^{-2}$, we have the lower bound

$$s\{\|p-q\|^2\}_{q \in S_i} \geq R_{\max}^2/4,$$

which holds for every $p \in S_i$. Thus,

$$t^2 = s\{s\{\|p-q\|^2\}_{q \in S \cup N}\}_{p \in S \cup N} \geq s\{s\{\|p-q\|^2\}_{q \in S_i}\}_{p \in S_i} \geq R_{\max}^2/4.$$

□

5.4.4 A Spectral Lemma

For a matrix A, let $\lambda_j(A)$ be the jth largest eigenvalue of the matrix. When the matrix is clear from context, we may simply write λ_j. The following lemma will be useful in our analysis of Robust PCA.

Lemma 5. *Let $A = M + C$ where M and C are symmetric positive semi-definite $n \times n$ matrices and* $\operatorname{rank}(M) = k$. *Then for $j > k$,*

$$\lambda_j(A) \leq \frac{1}{j-k} \sum_{i=1}^{j} \lambda_i(C).$$

Proof of Lemma 5. We use the following well-known theorem (see Theorem 4.8 of [45] for example).

Theorem 3. *Let $A = M + C$ where M and C are symmetric n-by-n matrices. Then*

$$\begin{aligned} \sum_{i=1}^{j} \lambda_i(M) + \lambda_{n-j+i}(E) &\leq \sum_{i=1}^{j} \lambda_i(M+E) \\ &\leq \sum_{i=1}^{j} \lambda_i(M) + \lambda_i(E). \end{aligned}$$

Thus,

$$\sum_{i=1}^{k} \lambda_i(M) \leq \sum_{i=1}^{k} \lambda_i(A)$$

and

$$\sum_{i=1}^{j} \lambda_i(A) \leq \sum_{i=1}^{j} \lambda_i(M) + \lambda_i(C).$$

Using the first of these inequalities shows that

$$\begin{aligned}(j-k)\lambda_j(A) &\leq \sum_{i=k+1}^{j} \lambda_i(A) \\ &\leq \sum_{i=k+1}^{j} \lambda_i(A) + \sum_{i=1}^{k} \lambda_i(A) - \sum_{i=1}^{k} \lambda_i(M) \\ &= \sum_{i=1}^{j} \lambda_i(A) - \sum_{i=1}^{k} \lambda_i(M).\end{aligned}$$

The second then shows

$$\sum_{i=1}^{j} \lambda_i(A) - \sum_{i=1}^{k} \lambda_i(M) \leq \sum_{i=1}^{j} \lambda_i(C),$$

since $\lambda_i(M) = 0$ for $i > k$. □

5.5 Analysis

We now turn to the major portion of our analysis. In Section 5.5.1, we analyze the effect of Robust PCA, showing that it preserves much of the distance between at least two means. In Section 5.6, we show the correctness of the partitioning step. Finally, we synthesize the whole argument in Section 5.5.2 to give the main theorem.

The essential parameters of the algorithm and analysis are $m_Z, m_X, m_Y, \beta, \epsilon$, and α. In terms of the quantities n,$w_{\min}$,δ, η, and k, these are

$$\begin{aligned}m_Z &= C_Z n {w_{\min}}^{-1} \log^5(nk/\delta) \\ m_X &= C_X n {w_{\min}}^{-1} \log^5(nk/\delta) \\ m_Y &= C_Y {w_{\min}}^{-1} \log(k/\delta) \\ \beta &= C_\beta \log((m_X + m_Y + m_Z)k \log(n)/(\delta\eta)) \\ &= O(\log(nk/(w_{\min}\delta\eta))) \\ \epsilon &= C_\epsilon w_{\min}/\beta^2 \\ &= \Omega(w_{\min} \log^{-2}(nk/(w_{\min}\delta\eta)) \\ \alpha &= C_\alpha k^{3/2} {w_{\min}}^{-1} \beta \log n \\ &= O(k^{3/2} {w_{\min}}^{-1} \log(nk/(w_{\min}\delta\eta)))\end{aligned}$$

where the leading factor is an appropriate constant. We will exercise the choice of these constants in the course of the analysis. The reader will find it useful to refer to these equations in following the proof. Without loss of generality, we may assume that η is a polynomial factor smaller than δ.

5.5.1 Robust PCA

Here we show that Robust PCA preserves most of the distance between the two components that are furthest part (Lemma 6). We accomplish this by showing that only a small fraction of this distance is lost as the dimension of the data is halved in each iteration (Lemma 7).

This result rests on two key claims. Claim 8 shows that the diameter of the non-noise data can be approximated in the presence of noise and that this permits the algorithm to place a relatively tight ball around the non-noise data, excluding noise points that are far away. The estimated diameter (roughly the parameter t) can neither be too small (or non-noise points will be excluded), nor too large (or noise points at the edge of the ball may have too large of an effect on the eigenvectors).

The other key claim is Claim 9 which bounds the maximum variance of the data in the subspace that is thrown out in an iteration. Recall that Robust PCA projects to $\lfloor(\dim(W) - k)/2\rfloor + k$ dimensions, removes outliers outside the ball $B(p_0, 16\beta t)$, and repeats until a k dimensional subspace is found.

To illustrate why simply removing outliers and using standard PCA to project to a k dimensional subspace is inadequate, we define the following matrices. Let N' be the set of noise points after outliers are removed and assume that no non-noise points are removed. The remaining points are therefore $S \cup N'$. Assume p_0 be the origin, that $W = \mathbb{R}^n$ and consider the matrix computed in step 2e of Robust PCA

$$A = \frac{1}{m'} \sum_{p \in S \cup N'} pp^T,$$

where N' consists of the noise points that were not removed and $m' = |S \cup N'|$. Using

the sample means $\hat{u}_i$ and covariance $\hat{\Sigma}_i$, we can decompose this matrix as the sum of

$$\begin{aligned} M &= \frac{1}{m'} \sum_{i=1}^{k} |S_i| \hat{\mu}_i \hat{\mu}_i^T \\ C &= \frac{1}{m'} \sum_{i=1}^{k} |S_i| \hat{\Sigma}_i \\ E &= \frac{1}{m'} \sum_{p \in N'} pp^T. \end{aligned}$$

The matrix M is the mixture of the outer product of the means, C is the mixture of the covariances, and E is the noise contribution.

Without noise, the second moment matrix A is just $M + C$. The rank of M is k and its eigenvectors are the subspace that we would ideally like to find, i.e. the span of the means. The matrix C can be viewed as a perturbation, which may cause the eigenvectors of $M + C$ to differ from those of M. The 2-norm of C is bounded from above by ${\sigma_{\max}}^2$, while the 2-norm of M is bounded from below in terms of $\mu_{\max}^2$. For an adequate separation of the component means the matrix M dominates so that applying PCA to $M + C$ gives a k dimension subspace that is close to the span of the means.

In the presence of noise, however, we must account not only for the perturbation caused by C but that caused by E (the noise component) as well. At first, it may seem that the noise component cannot have a large effect. As we will show in the proof of Claim 9, the sum of the eigenvalues of E is comparable to that of C. Recall that the 2-norm is the largest eigenvalue. While the sum of the eigenvalues of C may be on the order of $n{\sigma_{\max}}^2$, this is spread out over all n eigenvectors, so that no one eigenvalue is larger that ${\sigma_{\max}}^2$. We have no such guarantee for E; the sum may be concentrated in a single eigenvalue and therefore a single eigenvector. Even worse, it could be spread out over a constant fraction of the eigenvectors, each challenging the dominance of the eigenvectors of M. Hence, some constant fraction of the dimension must be preserved in order to avoid removing the eigenvectors of M, i.e. the span of

the means. Claim 9 shows that half of the dimension is adequate to preserve most of the distance between the means.

In our analysis, we often will identify a subspace by giving its dimension as a subscript. For instance, we will use W_k to denote the subspace returned by Robust PCA and W_ℓ for intermediate subspaces within Robust PCA. The main result of this section is the following lemma.

Lemma 6. *Suppose that every set Z obtained in step 2a of Robust PCA was generated by a good set for the current subspace W, P, and β. Then, letting W_k be the final subspace,*

$$\mu_{\max}^{(W_k,P)^2} \geq \frac{1}{2}{\mu_{\max}^{(P)}}^2.$$

Proof. This lemma is proved by applying the following lemma to each successive projection, until $n = k$.

Lemma 7. *Let P be a polyhedron in $\mathbb{R}^n$ and let W be a subspace in $\mathbb{R}^n$ with dimension greater than k, where $\mu_{\max}^{(W,P)} \geq \mu_{\max}^{(P)}/2$. Suppose that the set Z in step 2a of Robust PCA is generated by a good set for W,P and β. Let W_ℓ be the subspace of dimension $\lfloor(\dim(W) - k)/2\rfloor + k$ obtained in step 2e. Then for all pairs of components $i, j \in I_P$*

$$\|\mathrm{proj}_{W_\ell}(\mu_i - \mu_j)\|^2 \geq \|\mathrm{proj}_W(\mu_i - \mu_j)\|^2 - (\lfloor(\dim(W) - k)/2\rfloor + 1)^{-1}\frac{1}{4}{\mu_{\max}^{(P)}}^2 - \frac{32k}{w_{\min}}{\sigma_{\max}}^{(P)^2}.$$

Continuing with the proof of Lemma 6, let μ_i and μ_j be the means of two components such that $\|\mu_i - \mu_j\| = \mu_{\max}^{(P)}$. We observe that the quantity $\mu_{\max}^{(W,P)}$ can only decrease as the dimension of W is reduced. Therefore, when we unravel the recurrence relation implied by Lemma 7, we may simplify the bound to

$$\|\mathrm{proj}_{W_k}(\mu_i - \mu_j)\|^2 \geq \|\mu_i - \mu_j\|^2 - \frac{{\mu_{\max}^{(P)}}^2}{8}\sum_{j=0}^{\lceil\log_2(n-k)\rceil}\frac{1}{2^j} - \sum_{j=0}^{\lceil\log_2(n-k)\rceil}\frac{32}{w_{\min}}k{\sigma_{\max}}^{(P)^2},$$

By our choice of the pair μ_i, μ_j, we have $\|\mu_i - \mu_j\|^2 = {\mu_{\max}^{(P)}}^2$. Clearly, the first sum is bounded by ${\mu_{\max}^{(P)}}^2/4$. The second sum becomes $32k{w_{\min}}^{-1}{\sigma_{\max}}^2\lceil\log_2(n-k)\rceil$. By

the choice of α in Theorem 1, however, we may assume that this is no larger than ${\mu_{\max}^{(P)}}^2/4$ either. Thus,

$$\mu_{\max}^{(W,P)^2} \geq \|\text{proj}_{W_k}(\mu_i - \mu_j)\|^2 \geq \frac{1}{2}{\mu_{\max}^{(P)}}^2.$$

□

Proof of Lemma 7. Let $Z = S \cup N$ be a set generated by a good set of m_Z samples from LM. Let p_0, q_0 be a pair of points in $S \cup N$ satisfying $\|p_0 - q_0\| = t$. Because the denoising step removes all points outside of the ball $B(p_0, 16\beta t)$, we define the sets of remaining points $S' = S \cap B(p_0, 16\beta t)$ and $N' = N \cap B(p_0, 16\beta t)$. For convenience, we define $m' = |S' \cup N'|$.

We first claim that no non-noise points are eliminated (i.e. $S = S'$) and give a bound on the radius of the ball $B(p_0, 16\beta t)$.

Claim 8. *Suppose $p_0, q_0 \in S \cup N$ satisfy $\|p_0 - q_0\| = t$. Then*

$$S \subseteq B(p_0, 16\beta t) \subseteq B(p_0, 32\beta^2(\mu_{\max} + R_{\max})).$$

Thus, the second moment matrix used for the spectral analysis becomes

$$A = \frac{1}{m'} \sum_{p \in S \cup N'} (p - p_0)(p - p_0)^T.$$

This matrix has the following critical property.

Claim 9. *Suppose that $p_0, q_0 \in S \cup N$ satisfy $\|p_0 - q_0\| = t$. Then for $\ell = \lfloor (\dim(W) - k)/2 \rfloor + k$,*

$$\lambda_{\ell+1}(A) \leq \frac{w_{\min}}{64}(\lfloor (\dim(W) - k)/2 \rfloor + 1)^{-1}\mu_{\max}^{(W,P)^2} + 4k{\sigma_{\max}}^{(P)^2}.$$

By definition W_ℓ is the span of the top ℓ components of the matrix A. Let $\bar{W}_\ell$ be the complementary subspace in W. Consider a pair of means μ_i, μ_j. We will establish an upper bound on $\|\text{proj}_{\bar{W}_\ell}(\mu_i - \mu_j)\|^2$ and thus a lower bound on

$$\|\text{proj}_{W_\ell}(\mu_i - \mu_j)\|^2 = \|\text{proj}_W(\mu_i - \mu_j)\|^2 - \|\text{proj}_{\bar{W}_\ell}(\mu_i - \mu_j)\|^2 \tag{7}$$

to prove the lemma.

Let v denote unit vector in the direction of $\text{proj}_{\bar{W}_\ell}(\mu_i - \mu_j)$. Let $e = \mu_i - \mu_j - (\hat{\mu}_i - \hat{\mu}_i)$. Then

$$\begin{aligned} \|\text{proj}_{\bar{W}_\ell}(\mu_i - \mu_j)\|^2 &= (v^T(\mu_i - \mu_j))^2 \\ &\leq 2(v^T(\hat{\mu}_i - \hat{\mu}_j))^2 + 2\|e\|^2. \end{aligned}$$

By item 4 of Definition 2, the term $\|e\|^2$ is bounded from above by ${\sigma_{\max}}^{(P)^2}/4$. To bound the remaining term we use the fact that

$$1 \leq \frac{2}{w_{\min}} \frac{|S_i|}{m} \leq \frac{2}{w_{\min}} \frac{|S_i|}{m'}$$

from item 2 of Definition 2 to argue

$$\begin{aligned} (v^T(\hat{\mu}_i - \hat{\mu}_j))^2 &\leq 2((v^T(\hat{\mu}_i - p_0))^2 + (v^T(\hat{\mu}_j - p_0))^2) \\ &\leq 2\sum_{i=1}^{k} (v^T(\hat{\mu}_i - p_0))^2 \\ &\leq \frac{4}{w_{\min}} \cdot \frac{1}{m'} \sum_{i=1}^{k} |S_i|(v^T(\hat{\mu}_i - p_0))^2. \end{aligned}$$

For each i,

$$|S_i|(v^T(\hat{\mu}_i - p_0))^2 \leq \sum_{p \in S_i} (v^T(p - p_0))^2.$$

Including the points from N', we then have

$$\frac{1}{m'} \sum_{i=1}^{k} |S_i|(v^T(\hat{\mu}_i - p_0))^2 \leq v^T \left(\frac{1}{m'} \sum_{p \in S \cup N'} (p - p_0)(p - p_0)^T \right) v \leq \lambda_{\ell+1}(A).$$

since v is the in the subspace $\bar{W}_\ell$.

From Claim 9, we have

$$\begin{aligned} \|\text{proj}_{\bar{W}_\ell}(\mu_i - \mu_j)\|^2 &\leq \frac{4}{w_{\min}} \lambda_{\ell+1}(A) + \frac{{\sigma_{\max}}^{(P)^2}}{4} \\ &\leq \lfloor (\dim(W) - k)/2 \rfloor^{-1} \frac{1}{8} {\mu_{\max}^{(W,P)}}^2 + \frac{32}{w_{\min}} k {\sigma_{\max}}^{(P)^2}. \end{aligned}$$

Combined with (7), this proves the lemma. □

Proof of Claim 8. Since Lemma 7 assumes that $\mu_{\max}^{(W,P)} \geq \mu_{\max}^{(P)}/2$, we have

$$2\beta{\sigma_{\max}}^{(P)} \leq \alpha{\sigma_{\max}}^{(P)}/4 \leq \mu_{\max}^{(P)}/4 \leq \mu_{\max}^{(W,P)}/2,$$

using a suitable α and the separation assumption of (5).

By Lemma 4 then

$$t \geq \mu_{\max}^{(W,P)} - 2\beta{\sigma_{\max}}^{(P)} \geq \mu_{\max}^{(W,P)}/2.$$

Also, by the same lemma $t \geq R_{\max}^{(W,P)}/2$. Without loss of generality assume $\beta \geq 1$. Then

$$\max_{p,q\in S} \|p-q\| \leq \mu_{\max}^{(W,P)} + 2\beta R_{\max}^{(W,P)} \leq 2t + 4\beta t < 8\beta t.$$

Thus, no two points in S can be further than $8\beta t$ apart.

Now let p be an arbitrary point in S and let $q \in S \cap B(p_0, t)$. Note that such a point q exists because by definition of t, $B(p_0, t)$ contains $(1 - 2\epsilon m) > |N|$ points. We have by the triangle inequality and Lemma 4 that

$$\|p_0 - p\| \leq \|p_0 - q\| + \|q - p\| \leq t + 8\beta t < 16\beta t \leq 32\beta^2(\mu_{\max}^{(W,P)} + R_{\max}^{(W,P)})).$$

Thus, $S \subseteq B(p_0, 16\beta t) \subseteq B(p_0, 32\beta^2(\mu_{\max} + R_{\max}))$. □

Proof of Claim 9. Without loss of generality, let us assume that p_0 is the origin. Thus, the matrix from line 6 of the algorithm becomes $A = \frac{1}{m'}\sum_{p\in S\cup N'} pp^T$. Using the sample means $\hat{\mu}_i$, we can decompose this matrix as the sum of

$$\begin{aligned} M &= \frac{1}{m'}\sum_{i=1}^{k} |S_i| \mathrm{proj}_W(\hat{\mu}_i)\mathrm{proj}_W(\hat{\mu}_i)^T \\ C &= \frac{1}{m'}\sum_{i=1}^{k}\sum_{p\in S_i} \mathrm{proj}_W(p-\hat{\mu}_i)\mathrm{proj}_W(p-\hat{\mu}_i)^T \\ E &= \frac{1}{m'}\sum_{p\in N'} pp^T. \end{aligned}$$

Our strategy will be to bound $\sum_{i=1}^{\ell+1} \lambda_i(C+E)$ and apply Lemma 5 to bound $\lambda_{\ell+1}(A)$.

$$\begin{aligned}
\sum_{i=1}^{n} \lambda_i(C) &\leq \frac{1}{m'} \sum_{i=1}^{k} \sum_{p \in S_i} \|\mathrm{proj}_W(p - \hat{\mu}_i)\|^2 \\
&= \sum_{i=1}^{k} \frac{|S_i|}{m'} \frac{1}{|S_i|} \sum_{p \in S_i} \|\mathrm{proj}_W(p - \hat{\mu}_i)\|^2 \\
&\leq \max_i \frac{1}{|S_i|} \sum_{p \in S_i} \|\mathrm{proj}_W(p - \hat{\mu}_i)\|^2 \\
&\leq \frac{8}{7} {R_{\max}^{(W,P)}}^2.
\end{aligned}$$

By Claim 8, $N' \subseteq B(0, 16\beta t) \subseteq B(32\beta^2(\mu_{\max}^{(W,P)} + R_{\max}^{(W,P)}))$, so

$$\sum_{i=1}^{n} \lambda_i(E) \leq 2\epsilon \max_{p \in N'} \|p\|^2 \leq \epsilon 64 \beta^2 ({\mu_{\max}^{(W,P)}}^2 + {R_{\max}^{(W,P)}}^2) \leq \frac{w_{\min}}{64} ({\mu_{\max}^{(W,P)}}^2 + {R_{\max}^{(W,P)}}^2),$$

for an appropriate choice of $\epsilon = C_\epsilon w_{\min} \beta^{-2}$ from Theorem 1. Combining these bounds

$$\begin{aligned}
\sum_{i=1}^{n} \lambda_i(C) + \lambda_i(E) &\leq \frac{w_{\min}}{64} {\mu_{\max}^{(W,P)}}^2 + \left(\frac{w_{\min}}{64} + \frac{8}{7} \right) {R_{\max}^{(W,P)}}^2 \\
&\leq \frac{w_{\min}}{64} {\mu_{\max}^{(W,P)}}^2 + 2 \dim(W) {\sigma_{\max}}^2.
\end{aligned}$$

Lemma 5 then gives the bound

$$\begin{aligned}
\lambda_{\ell+1}(A) &\leq \frac{1}{\ell + 1 - k} \sum_{i=1}^{\ell+1} \lambda_i(C+E) \\
&\leq \frac{1}{\ell + 1 - k} \sum_{i=1}^{n} \lambda_i(C) + \lambda_i(E) \\
&\leq (\lfloor (\dim(W) - k)/2 \rfloor + 1)^{-1} \\
&\quad \left(\frac{w_{\min}}{64} {\mu_{\max}^{(W,P)}}^2 + 2 \dim(W) {\sigma_{\max}}^2 \right). \qquad (8)
\end{aligned}$$

We note that $2\dim(W)/(\lfloor (\dim(W) - k)/2 \rfloor + 1) \leq 4k$. If $\dim(W) < 2k$, then this is trivial. On the other hand, if $\dim(W) \geq 2k$, then

$$\frac{2\dim(W)}{\lfloor (\dim(W) - k)/2 \rfloor + 1} \leq \frac{4\dim(W)}{\dim(W) - k} \leq 8 \leq 4k.$$

The bound of (8) then becomes

$$\lambda_{\ell+1}(A) \leq (\lfloor(\dim(W) - k)/2\rfloor + 1)^{-1} \frac{w_{\min}}{64} {\mu_{\max}^{(W,P)}}^2 + 4k{\sigma_{\max}}^2.$$

□

5.5.2 Partitioning Components

We show that Algorithm 3 successfully partitions the components. The algorithm tries many directions in the subspace W until it finds a one with a "valley" corresponding to the intuition given in Section 5.2. We capture this notion formally in the following definition.

Definition 3. *Let X be a set of m_X points in $\mathbb{R}$. For $i \in \mathbb{Z}$ let $b_i = |\{x \in X : x \in (id, (i+1)d]\}|$. We say that X has a* valley *if there is a triple $i_1 < i_2 < i_3$ such that $b_{i_1}, b_{i_3} > w_{\min} m_X/8$ and $b_{i_2} \leq 2\epsilon m_X$. We define the point $d(i_2 + 1/2)$ to be the* middle *of the valley.*

Assuming that d is well-chosen, the existence of a valley is ensured by the fact that the means are well-separated compared to the width of the widest component. If d is too small, then we are likely to find valleys within the point set of a single component. If d is too large, then the whole mixture might fit into a single unit of resolution or "bucket." When d is chosen correctly, non-noise points from two components fill the outer buckets, while only noise points fill the middle one.

The following two Lemmas show that with high probability the algorithm succeeds in a given node in the recursion tree. The first applies for internal nodes in the tree where components need to be separated, the second applies to the leaves where the division of space terminates.

Lemma 10. *Let $\delta > 0$. Suppose that P is η-safe for $\mathcal{F}$ and that $|I_P| > 1$. Then with probability $1 - \delta$, the halfspaces $H_{v,\gamma}$ and $H_{-v,-\gamma}$ are (η/k)-safe and each contains at least one component mean of I_P.*

Lemma 11. *Let $\delta > 0$. Suppose that P is η-safe for $\mathcal{F}$ and that $|I_P| = 1$. Then with probability $1 - \delta$, no valley will be found by Algorithm 3 in step 6c.*

Proof of Lemma 10. We first consider the set Y_0 obtained in line 6. Let $Y_0 = S_1 \cup \ldots S_k \cup N$ disjointly, where N is the set of noise points and S_i is the set of points from component i. We show that with probability $1 - \delta/2$ the set Y_0 has the following two properties.

1. Y_0 contains at least one point from every component.
2. For every component i and every $p \in S_i$, $\|\text{proj}_W(p - \mu_i)\| \leq \beta R_i^{(W)}$.

The probability that no point from component i is in a set of m_Y points is at most $(1 - w_{\min})^m \leq \exp(-w_{\min} m_Y) \leq \frac{\delta}{4k},$, where we have used the fact that $m_Y \geq {w_{\min}}^{-1} \log(4k/\delta)$ in the last step. Taking a union bound over all k components shows that the probability that no samples are taken from some component is at most $\delta/4$.

To show the second property, we consider projection of the mixture distribution onto the subspace W. Component i of this distribution has mean $\text{proj}_W(\mu_i)$ and the component is logconcave, being the projection of a logconcave distribution. Applying theorem 2 gives the result, since β is larger than the requirement of $\beta \geq O(\log(m_Y/\delta)$.

Without loss of generality, we assume that $\eta \leq \delta(4(\lceil \log n \rceil + 1)(m_Z + m_X))^{-1}$. Thus, by Lemma 3 we may argue that with probability $1 - \delta/2$ every set used by Robust PCA is generated by a good set (Definition 2) as required by Lemma 6 and the set X_0 is good as well. Overall, the collection of sample sets has the desired properties with probability $1 - \delta$.

Assuming that all sets given to Robust PCA are good, Lemma 6 guarantees that after projection to W

$$\mu_{\max}^{(W,P)} \geq \frac{\mu_{\max}^{(P)}}{2}.$$

With an appropriate choice of α, this implies that in the subspace W

$$\mu_{\max}^{(W,P)} \geq \mu_{\max}^{(P)}/2 \geq \frac{\alpha}{2}{\sigma_{\max}}^{(P)} \geq 41k^{3/2}\beta{\sigma_{\max}}^{(P)} \geq 41k\beta R_{\max}^{(P)}.$$

This fact enables us to use the following claim.

Claim 12. *If* $|I_P| > 1$ *and*

$$\mu_{\max}^{(W,P)} \geq 41k\beta R_{\max}^{(W,P)},$$

then the quantity $d = t/10k$ *satisfies the bounds*

$$4\beta R_{\max}^{(W,P)} \leq d \leq \frac{\mu_{\max}^{(W,P)}}{5k}.$$

Proof of Claim 12. We first derive a lower bound. Since $t \geq \mu_{\max}^{(W,P)} - 2\beta R_{\max}^{(W,P)}$ by Lemma 4 and $\mu_{\max}^{(W,P)} > 41k\beta R_{\max}^{(W,P)}$, we have

$$d = \frac{t}{10k} \geq \frac{\mu_{\max}^{(W,P)} - 2\beta R_{\max}^{(W,P)}}{10k} \geq \frac{41k\beta R_{\max}^{(W,P)} - 2\beta R_{\max}^{(W,P)}}{10k} \geq 4\beta R_{\max}^{(W,P)}.$$

To show the upper bound, we observe that since $\mu_{\max}^{(W,P)} \geq 41k\beta R_{\max}$, we have that $2\mu_{\max}^{(W,P)} \geq \mu_{\max}^{(W,P)} + 2\beta R_{\max}^{(W,P)}$. Thus, by Lemma 4

$$d = \frac{t}{10k} \leq \frac{\mu_{\max}^{(W,P)} + 2\beta R_{\max}^{(W,P)}}{10k} \leq \frac{2\mu_{\max}^{(W,P)}}{10k} = \frac{\mu_{\max}^{(W,P)}}{5k}.$$

□

The remainder of the proof rests on the following two claims. The first shows that any valley that is found produces a (η/k)-safe halfspace. The second claim shows that a valley will indeed be found.

Claim 13. *For any direction* $v \in W_k$*, if* $\mathrm{proj}_v(X)$ *has a valley with midpoint* γ*, then* $H_{v,\gamma}$ *and* $H_{-v,-\gamma}$ *are* (η/k)*-safe for* $\mathcal{F}_P$*.*

Proof. Because the set X is generated by a good set, the points from a single component j must be contained in an interval that is centered about the component mean of size $2\beta R_{\max}^{(W,P)}$. By Claim 12 this is at most $d/2$, half of the width of a bucket.

Suppose that a point $d(i + 1/2)$ falls into one of these intervals for some $i \in \mathbb{Z}$. Then the entire interval must be contained in the bucket i, i.e.

$$[\mu_j - \beta R_{\max}^{(W,P)}, \mu_j + \beta R_{\max}^{(W,P)}] \subseteq [di, (d+1)).$$

But then, $b_i \geq w_{\min} m_Z/2$. For an appropriate choice of ϵ, however, $w_{\min} m_Z/2 > 2\epsilon m_Z$, and hence i cannot be the middle part of the valley. We conclude that if $\text{proj}_v(W)$ has valley with middle γ, then for all $i \in I_P$

$$|v \cdot \mu_i - \gamma| > \beta R_{\max}^{(W,P)} \geq \beta \sigma_i.$$

Hence by Proposition 1, the halfspaces $H_{v,\gamma}$ and $H_{-v,-\gamma}$ are (η/k)-safe for $\mathcal{F}_P$, since we may choose $\beta \geq \beta_1 = O(\log(k/\eta))$. □

Claim 14. *There is a pair $(a, b) \in Y \times Y$ such that the unit vector v in the direction $a - b$ has a valley.*

Proof. Let μ_i and μ_j be two components such that $\|\text{proj}_W(\mu_i - \mu_j)\| = \mu_{\max}^{(W,P)}$. Let $a \in S_i$ and let $b \in S_j$, where $Y = S_1 \cup \ldots \cup S_k \cup N$. Define v to be the unit vector along $a - b$.We will show that μ_i and μ_j are far apart v.

Because we have assumed Y is generated by a good set, $\|(a-b) - \text{proj}_W(\mu_i - \mu_j)\| \leq 2\beta R_{\max}^{(W,P)}$. Thus,

$$\begin{aligned}
|\text{proj}_v(\mu_i - \mu_j)| &= \frac{|(a-b) \cdot \text{proj}_W(\mu_i - \mu_j)|}{\|(a-b)\|} \\
&\geq \|\text{proj}_W(\mu_i - \mu_j)\| \left(1 - \frac{\|(a-b) - \text{proj}_W(\mu_i - \mu_j)\|^2}{\|\text{proj}_W(\mu_i - \mu_j)\|^2}\right)^{1/2} \\
&\geq \mu_{\max}^{(W,P)} \left(1 - \frac{4\beta^2 {R_{\max}^{(W,P)}}^2}{{\mu_{\max}^{(W,P)}}^2}\right)^{1/2}
\end{aligned}$$

Because W has only k dimensions ${R_{\max}^{(W,P)}}^2 \leq k {\sigma_{\max}}^{(P)^2}$. As argued above by Lemma 6, $\mu_{\max}^{(W,P)} \geq \mu_{\max}^{(P)}/2 \geq \alpha {\sigma_{\max}}^{(P)}/2$. Therefore,

$$\begin{aligned}
|\text{proj}_v(\mu_i - \mu_j)| &\geq \mu_{\max}^{(W,P)} \left(1 - \frac{\beta^2 k {\sigma_{\max}}^{(P)^2}}{\alpha^2 {\sigma_{\max}}^{(P)^2}}\right)^{1/2} \\
&\geq \frac{\mu_{\max}^{(W,P)}}{2},
\end{aligned}$$

for $\alpha \geq \beta\sqrt{2k}$.

By Claim 12, $d \leq \mu_{\max}^{(W,P)}/(5k)$, so we have $\text{proj}_v(\mu_i - \mu_j) \geq 5kd/2$.

We now turn our attention to the set $X = S_1 \cup \ldots \cup S_k \cup N$ (the set Y will not be referred to again). Because X is good, every set $\text{proj}_v(S_i)$ must be contained in an interval centered around $\text{proj}_v(\mu_i)$ of length $2\beta R_{\max}$. By the lower bound on d from Claim 12, the width of this interval is at most $d/2$. Since there are k of these, this leaves $5kd/2 - kd/2 = 2kd$ of "empty" space between $\text{proj}_v(\mu_i)$ and $\text{proj}_v(\mu_j)$, in which only noise point can fall. This space can be cut into at most $k-1$ pieces, meaning that at least one piece must have length $2d$. An interval $[d\ell, d(\ell+1))$ must be contained in one of these pieces, and this will form the middle of a valley, with buckets containing $\text{proj}_v(S_i)$ and $\text{proj}_v(S_j)$ serving as the other buckets. □

□

Proof of Lemma 11. Without loss of generality, we may assume that $\eta \leq \delta/(4m_X)$. By Lemma 3, with probability $1 - \delta/2$ the set X_0 is good for P, W_k, β.

Assuming that X_0 is good we can derive a lower bound on d. Since $t \geq R_{\max}^{(W,P)}/2$ by Lemma 4, we have that

$$d \geq \frac{R_{\max}^{(W,P)}}{20k}.$$

Also assuming that X_0 is good, we have that $X = \text{proj}_{W_k}(X_0 \cap P)$ consists only of points from a single component S_j and a set of noise points N. The set N consists of no more than $2\epsilon m_X$ points. Thus, for any direction u generated by $Y \times Y$, we have

$$b_i = |\{x \in S_j \cup N : \text{proj}_u(x) \in [di, d(i+1))\}|.$$

For purposes of analysis, We define

$$b'_i = |\{x \in S_j : \text{proj}_u(x) \in [di, d(i+1))\}|.$$

Suppose that $i_1 < i_2 < i_3$ form a valley. This implies that $b_{i_1} \geq w_{\min} m_X/4$ and that

$$b'_{i_1} \geq b_{i_1} - |N| \geq w_{\min} m_X/4 - 2\epsilon m_X \geq w_{\min} m_Z/8.$$

choosing ϵ appropriately. The same bound holds for b_{i_3}. On the other hand,

$$b'_{i_2} \leq b_{i_2} \leq 2\epsilon m_X \leq w_{\min} m_X/32,$$

for an appropriate choice of ϵ. Since $m_X = C_X n w_{\min}{}^{-1} \log^5(nk/\delta)$, we argue that this event has probability less than $\delta/2$, using the following claim.

Claim 15. *Let $\xi, \delta > 0$. Consider a logconcave distribution $\mathcal{F}$ in one dimension with variance σ^2 and let $d \geq C\sigma$. Let S be a sample set of m points drawn from $\mathcal{F}$ and let $b_i = |\{p \in S : p \in [di, d(i+1))\}|$. There is a constant C' such that if $m \geq C'\xi^{-1} \log(\log(m)/C\delta)$, then with probability $1-\delta$ the following holds for every $i \in \mathbb{Z}$ and $\xi' \geq \xi$.*

1. *If $b_i > 2\xi' m$, then $\mathsf{P}[x \in [di, d(i+1)]) > \xi'$.*
2. *If $b_i < \xi' m/2$, then $\mathsf{P}[x \in [di, d(i+1)]) < \xi'$.*

Proof. We first observe that with probability $1 - \delta/2$ no point will be further than $\sigma \log(6m/\delta)$ away from the mean, using a trivial application (1 dimension only) of Theorem 2. Therefore, all but $2C^{-1} \log 6m/\delta$ buckets will be empty. For a single bucket, a Chernoff bound shows $m > 12\xi^{-1} \log(1/\delta')$ ensures that the desired property holds with probability $1 - \delta'$. With $\delta' = \delta C/(4 \log(6m/\delta))$, we may apply a union bound to prove the lemma. □

□

5.6 Proof of the Main Theorem

Proof of Theorem 1. Consider one node in the recursion tree of Algorithm 3, and suppose that P has $j < k$ support hyperplanes and that P is $(\eta j/k)$-safe for $\mathcal{F}$. Note that in the root of the tree this is true because $\mathbb{R}^n$ is 0-safe. In the case where the polyhedron contains more than one component mean (i.e. $|I_P| > 1$), Lemma 10 shows that with probability $1 - \delta'$ the half-space $H_{v,\gamma}$ obtained in line 9 excludes at least one

component mean and is (η/k)-safe for $\mathcal{F}_P$. Proposition 2 then shows that $P \cap H_{v,\gamma}$ is $(\eta(j+1)/k)$-safe for $\mathcal{F}$.

On the other hand, if the polyhedron P contains only one mean (i.e. $|I_P| = 1$), then with probability $1 - \delta'$ the algorithm does not find a valley and returns the polyhedron by Lemma 11. Thus, with probability $1 - 2k\delta'$ the algorithm returns a set of k polyhedra, each containing exactly one component mean and each η-safe for $\mathcal{F}$. Thus, we have by definition of η-safe that the collection of polyhedra induce a classifier that is correct with probability $1 - \eta$, as the theorem claims. $\square$

CHAPTER VI

AFFINE-INVARIANT CLUSTERING

6.1 Introduction

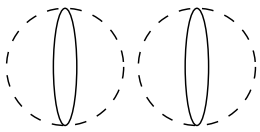
Prior to the introduction of Isotropic PCA, the representative hard case for learning mixtures of Gaussians was two parallel "pancakes", i.e., two Gaussians that are spherical in $n-1$ directions and narrow in the last direction, so that a hyperplane orthogonal to the last direction separates the two. The spectral approach of [29, 1] requires a separation that grows with their largest standard deviation which is unrelated to the distance between the pancakes (their means). Because there is a subspace where the Gaussians are separable, the separation requirement should depend only on the dimension of this subspace and the components' variances in it. The Unravel algorithm gives such a result.

We assume we are given a lower bound $w_{\min}$ on the minimum mixing weight and k, the number of components. With high probability, our algorithm Unravel returns a partition of space by hyperplanes so that each part (a polyhedron) encloses almost all of the probability mass of a single component and almost none of the other components. The error of such a set of polyhedra is the total probability mass that falls outside the correct polyhedron.

We first state our result for two Gaussians in a way that makes clear the relationship to previous work that relies on separation.

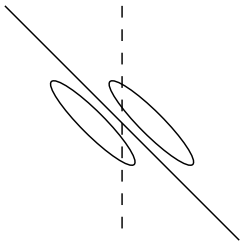
Theorem 4. *Let w_1, μ_1, Σ_1 and w_2, μ_2, Σ_2 define a mixture of two Gaussians. There is an absolute constant C such that, if there exists a direction v such that*

$$|\mathrm{proj}_v(\mu_1 - \mu_2)| \geq C \left(\sqrt{v^T \Sigma_1 v} + \sqrt{v^T \Sigma_2 v} \right) {w_{\min}}^{-2} \log^{1/2} \left(\frac{1}{w_{\min} \delta} + \frac{1}{\eta} \right),$$



(a) Distance Concentration Separability

(b) Hyperplane Separability



(c) Intermean Hyperplane and Fisher Hyperplane.

Figure 4: Previous work requires distance concentration separability which depends on the maximum directional variance (a). Our results require only hyperplane separability, which depends only on the variance in the separating direction(b). For non-isotropic mixtures the best separating direction may not be between the means of the components(c).

Then with probability $1-\delta$ *algorithm* UNRAVEL *returns two complementary halfspaces that have error at most* η *using time and a number of samples that is polynomial in* $n, {w_{\min}}^{-1}, \log(1/\delta)$.

The requirement is that in *some direction* the separation between the means must be comparable to the standard deviation. This separation condition of Theorem 4 is affine-invariant and much weaker than conditions of the form $\|\mu_1 - \mu_2\| \gtrsim \max\{\sigma_{1,\max}, \sigma_{2,\max}\}$ used in previous work. See Figure 4(a). The dotted line shows how previous work effectively treats every component as spherical. We require only hyperplane separability (Figure 4(b)), which is a weaker condition. We also note that the separating direction does not need to be the intermean direction as illustrated in Figure 4(c). The dotted line illustrates the hyperplane induced by the intermean direction, which may be far from the optimal separating hyperplane shown by the solid line.

It will be insightful to state this result in terms of the Fisher discriminant, a standard notion from Pattern Recognition [14, 22] that is used with labeled data. In words, the Fisher discriminant along direction p is

$$J(p) = \frac{\text{the intra-component variance in direction } p}{\text{the total variance in direction } p}$$

Mathematically, this is expressed as

$$J(p) = \frac{E\left[\|\text{proj}_p(x - \mu_{\ell(x)})\|^2\right]}{E\left[\|\text{proj}_p(x)\|^2\right]} = \frac{p^T(w_1\Sigma_1 + w_2\Sigma_2)p}{p^T(w_1(\Sigma_1 + \mu_1\mu_1^T) + w_2(\Sigma_2 + \mu_2\mu_2^T))p}$$

for x distributed according to a mixture distribution with means μ_i and covariance matrices Σ_i. We use $\ell(x)$ to indicate the component from which x was drawn.

Theorem 5. *There is an absolute constant* C *for which the following holds. Suppose that* $\mathcal{F}$ *is a mixture of two Gaussians such that there exists a direction* p *for which*

$$J(p) \leq C{w_{\min}}^3 \log^{-1}\left(\frac{1}{\delta w_{\min}} + \frac{1}{\eta}\right).$$

With probability $1-\delta$*, algorithm* UNRAVEL *returns a halfspace with error at most* η *using time and sample complexity polynomial in* $n, {w_{\min}}^{-1}, \log(1/\delta)$.

There are several ways of generalizing the Fisher discriminant for $k = 2$ components to greater k [22]. In all cases, however, instead of a single line, we seek a $(k-1)$-dimensional subspace in which to separate the components. Intuitively, we would like this subspace to minimize the distance between points and their component means relative to the distance between the means. For simplicity, we adapt the definition of the Fisher subspace to the isotropic case. Recall that an isotropic distribution has the identity matrix as its covariance and the origin as its mean. Therefore,

$$\sum_{i=1}^{k} w_i \mu_i = 0 \text{ and } \sum_{i=1}^{k} w_i (\Sigma_i + \mu_i \mu_i^T) = I.$$

It is well known that any distribution with bounded covariance matrix (and therefore any mixture) can be made isotropic by an affine transformation.

Under isotropy, the denominator of the Fisher discriminant is always 1. Thus, the discriminant is just the expected squared distance between the projection of a point and the projection of its mean, where projection is onto some direction p. The generalization to $k > 2$ is natural, as we may simply replace projection onto direction p with projection onto a $(k-1)$-dimensional subspace S. For convenience, let

$$\Sigma = \sum_{i=1}^{k} w_i \Sigma_i.$$

Let the vector $p_1, \ldots, p_{k-1}$ be an orthonormal basis of S and let $\ell(x)$ be the component from which x was drawn. We then have under isotropy

$$J(S) = E[\|\text{proj}_S(x - \mu_{\ell(x)})\|^2] = \sum_{j=1}^{k-1} p_j^T \Sigma p_j$$

for x distributed according to a mixture distribution with means μ_i and covariance matrices Σ_i. As Σ is symmetric positive definite, it follows that the smallest $k-1$ eigenvectors of the matrix are optimal choices of p_j. The Fisher subspace is the span of these vectors.

Definition 4. *Let* $\{(w_i, \mu_i, \Sigma_i)\}$ *be the weights, means, and covariance matrices for an isotropic mixture distribution where* $\dim(\text{span}\{\mu_1, \ldots, \mu_k\}) = k - 1$. *Let* $\ell(x)$ *be the component from which* x *was drawn. The* Fisher subspace F *is defined as the* $(k-1)$*-dimensional subspace that minimizes*

$$J(S) = E[\|\text{proj}_S(x - \mu_{\ell(x)})\|^2].$$

over subspaces S *of dimension* $k - 1$.

Note that $\dim(\text{span}\{\mu_1, \ldots, \mu_k\})$ is only $k-1$ because isotropy implies $\sum_{i=1}^k w_i \mu_i = 0$.

The next lemma provides a simple alternative characterization of the Fisher subspace as the span of the means of the components (after transforming to isotropic position).

Lemma 16. *Suppose* $\{w_i, \mu_i, \Sigma_i\}_{i=1}^k$ *defines an isotropic mixture in* $\mathbb{R}^n$. *Let* $\lambda_1 \geq \ldots \geq \lambda_n$ *be the eigenvalues of the matrix* $\Sigma = \sum_{i=1}^k w_i \Sigma_i$ *and let* $v_1, \ldots, v_n$ *be the corresponding eigenvectors. If the dimension of the span of the means of the components is* $k - 1$*, then the Fisher subspace*

$$F = \text{span}\{v_{n-k+2}, \ldots, v_n\} = \text{span}\{\mu_1, \ldots, \mu_k\}.$$

Our algorithm attempts to find the Fisher subspace (or one close to it) and succeeds in doing so, provided that the components do not "overlap" much in the following sense.

Definition 5. *The* overlap *of a mixture given as in Definition 4 is*

$$\phi = \min_{S:\dim(S)=k-1} \max_{p \in S} p^T \Sigma p. \tag{9}$$

It is a direct consequence of the Courant-Fisher min-max theorem that ϕ is the $(k-1)$th smallest eigenvalue of the matrix Σ and the subspace achieving ϕ is the Fisher subspace, i.e.,

$$\phi = \left\| E[\text{proj}_F(x - \mu_{\ell(x)}) \text{proj}_F(x - \mu_{\ell(x)})^T] \right\|_2.$$

We can now state our main theorem for $k > 2$.

Theorem 6. *There is an absolute constant C for which that following holds. Suppose that $\mathcal{F}$ is a mixture of k Gaussian components where the overlap satisfies*

$$\phi \leq C {w_{\min}}^3 k^{-3} \log^{-1}\left(\frac{nk}{\delta w_{\min}} + \frac{1}{\eta}\right)$$

With probability $1-\delta$, algorithm UNRAVEL *returns a set of k polyhedra that have error at most η using time and a number of samples that is polynomial in $n, {w_{\min}}^{-1}, \log(1/\delta)$.*

In words, the algorithm successfully unravels arbitrary Gaussians provided there exists a $(k-1)$-dimensional subspace in which along every direction, the expected squared distance of a point to its component mean is smaller than the expected squared distance to the overall mean by roughly a $\text{poly}(k, 1/w_{\min})$ factor. There is no dependence on the largest variances of the individual components, and the dependence on the ambient dimension is logarithmic. This means that the addition of extra dimensions (even where the distribution has large variance) has little impact on the success of our algorithm.

6.2 The Unravel Algorithm

The algorithm has three major components: an initial affine transformation, a reweighting step, and identification of a direction close to the Fisher subspace and a hyperplane orthogonal to this direction which leaves each component's probability mass almost entirely in one of the halfspaces induced by the hyperplane. The key insight is that the reweighting technique will either cause the mean of the mixture to shift in the intermean subspace, or cause the top $k-1$ principal components of the second moment matrix to approximate the intermean subspace. In either case, we obtain a direction along which we can partition the components.

We first find an affine transformation W which when applied to $\mathcal{F}$ results in an isotropic distribution. That is, we move the mean to the origin and apply a

linear transformation to make the covariance matrix the identity. We apply this transformation to a new set of m_1 points $\{x_i\}$ from $\mathcal{F}$ and then reweight according to a spherically symmetric Gaussian $\exp(-\|x\|^2/(2\alpha))$ for $\alpha = \Theta(n/w_{\min})$. We then compute the mean $\hat{u}$ and second moment matrix $\hat{M}$ of the resulting set.

After the reweighting, the algorithm chooses either the new mean or the direction of maximum second moment and projects the data onto this direction h. By bisecting the largest gap between points, we obtain a threshold t, which along with h defines a hyperplane that separates the components. Using the notation $H_{h,t} = \{x \in \mathbb{R}^n : h^T x \geq t\}$, to indicate a halfspace, we then recurse on each half of the mixture. Thus, every node in the recursion tree represents an intersection of half-spaces. To make our analysis easier, we assume that we use different samples for each step of the algorithm. The reader might find it useful to read Section 6.2.1, which gives an intuitive explaination for how the algorithm works on parallel pancakes, before reviewing the details of the algorithm.

Algorithm 4 Unravel

Input: Integer k, scalar $w_{\min}$. Initialization: $P = \mathbb{R}^n$.

1. (Isotropy) Use samples lying in P to compute an affine transformation W that makes the distribution nearly isotropic (mean zero, identity covariance matrix).
2. (Reweighting) Use m_1 samples in P and for each compute a weight $e^{-\|x\|^2/(\alpha)}$ (where $\alpha > n/w_{\min}$).
3. (Separating Direction) Find the mean of the reweighted data $\hat{\mu}$. If $\|\hat{\mu}\| > \sqrt{w_{\min}}/(32\alpha)$, let $h = \hat{\mu}$. Otherwise, find the second moment matrix $\hat{M}$ of the reweighted points and let h be its top principal component.
4. (Recursion) Project m_2 sample points to h and find the largest gap between points in the interval $[-1/2, 1/2]$. If this gap is less than $1/4(k-1)$, then return P. Otherwise, set t to be the midpoint of the largest gap, recurse on $P \cap H_{h,t}$ and $P \cap H_{-h,-t}$, and return the union of the polyhedra produces by these recursive calls.

6.2.1 Parallel Pancakes

The following special case, which represents an open problem in previous work, will illuminate the intuition behind the new algorithm. Suppose $\mathcal{F}$ is a mixture of two Gaussians that are spherical with variance 1 in the $n-1$ dimensions orthogonal to the intermean direction. Along the intermean direction the variance is some small quantity $\epsilon \ll 1$, and the distance between the means is much larger than $\sqrt{\epsilon}$. This mixture may be visualized as parallel pancakes.

We consider two cases, one where the mixing weights are equal and another where they are imbalanced. When the mixing weights are equal, the means of the components will be equally spaced at a distance of $1-\phi$ on opposite sides of the origin. For imbalanced weights, the origin will still lie on the intermean direction but will be much closer to the heavier component, while the lighter component will be much further away. In both cases, this transformation makes the variance of the mixture 1 in every direction, so the principal components give us no insight into the inter-mean direction.

Consider next the effect of the reweighting on the mean of the mixture. For the case of equal mixing weights, symmetry assures that the mean does not shift at all. For imbalanced weights, however, the heavier component, which lies closer to the origin will become heavier still. Thus, the reweighted mean shifts toward the mean of the heavier component, allowing us to detect the intermean direction.

Finally, consider the effect of reweighting on the second moments of the mixture with equal mixing weights. Because points closer to the origin are weighted more, the second moment in every direction is reduced. However, in the intermean direction, where part of the moment is due to the displacement of the component means from the origin, it shrinks less. Thus, the direction of maximum second moment is the intermean direction.

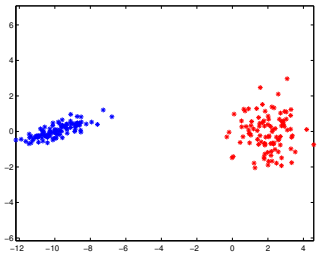
6.3 Empirical Illustrations

A Matlab implementation has been used to verify the effectiveness of our algorithm. Figure 5 illustrates the effect of enforcing isotropy. Notice how the intermean direction is not always a good separating direction for the non-isotropic case, but it is for the isotropic case.

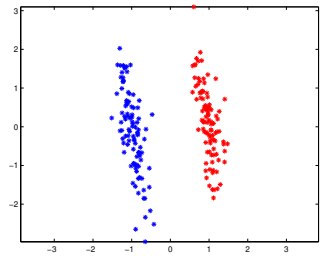
Figure 6 illustrates the effectiveness of the algorithm as a whole. In the example, three Gaussians in forty dimensions are given smaller variances in the intermean subspace and larger variances in the orthogonal subspace. One can think of each Gaussian being shaped like an egg with the narrow dimensions in the intermean subspace. Random projection does not work since a random vector will be almost orthogonal to the intermean subspace. PCA does not work because the larger variances (corresponding to the length of the egg) counter the effect of the separation of the means. Isotropic PCA, however, reveals the intermean direction.

6.4 Overview of the Analysis

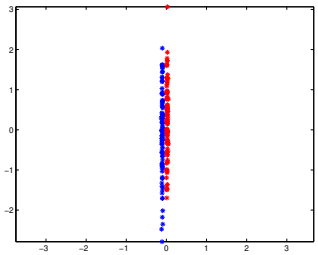
To analyze the algorithm in the general case we will proceed as follows. Section 6.5 shows that under isotropy the Fisher subspace coincides with the intermean subspace (Lemma 16) and relates overlap to a more conventional notion of separation (Prop. 20). Section 6.6 then gives some convenient approximations to the first and second moments of the reweighted mixture. Section 6.7 gives the necessary sampling convergence lemmas to ensure that these moments can be efficiently learned from data. Section 6.8 then combines the approximations of Sec. 6.6 with a perturbation lemma due to Stewart to show that the vector h (either the mean shift or the largest principal component) lies close to the intermean subspace. Finally, Section 6.9 shows the correctness of the recursive aspects of the algorithm.



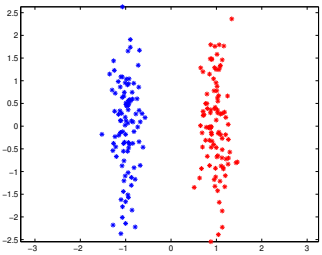
(a) Eg 1: Before Isotropy



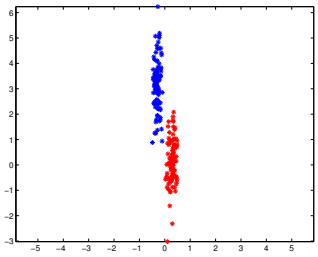
(b) Eg 1: After Isotropy



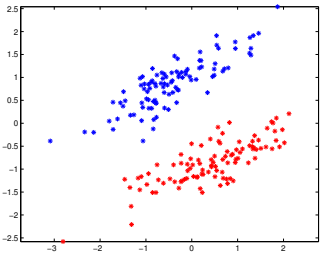
(c) Eg 2: Before Isotropy



(d) Eg 2: After Isotropy

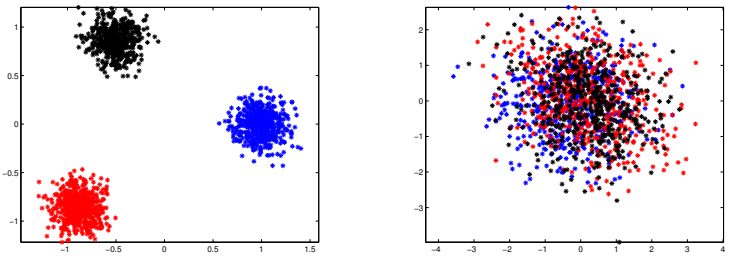


(e) Eg 3: Before Isotropy



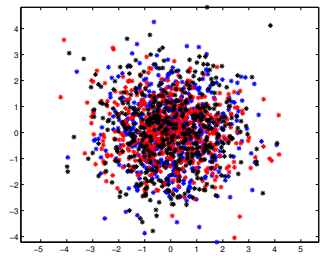
(f) Eg 3: After Isotropy

Figure 5: Enforcing Isotropy will squeeze components together if they are apart (a,b) or stretch them away from each other if they are close (c,d). It also has the effect of making the intermean direction the best choice for separating the components (e,f).

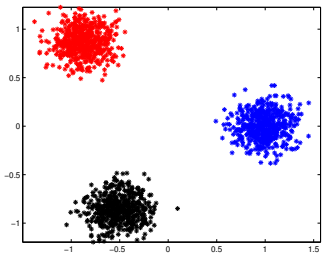


(a) Fisher Subspace

(b) Random Subspace



(c) PCA Subspace



(d) Isotropic PCA Subspace

Figure 6: Random Projection (b) and PCA (c) collapse the components, but Isotropic PCA find the Fisher subspace where the components can be separated.

6.5 Preliminaries

6.5.1 Matrix Properties

For a matrix Z, we will denote the ith largest eigenvalue of Z by $\lambda_i(Z)$ or just λ_i if the matrix is clear from context. Unless specified otherwise, all norms are the 2-norm. For symmetric matrices, this is $\|Z\|_2 = \lambda_1(Z) = \max_{x \in \mathbb{R}^n} \|Zx\|_2 / \|x\|_2$.

The following two facts from linear algebra will be useful in our analysis.

Fact 17. *Let $\lambda_1 \geq \ldots \geq \lambda_n$ be the eigenvalues for an n-by-n symmetric positive definite matrix Z and let $v_1, \ldots v_n$ be the corresponding eigenvectors. Then*

$$\lambda_n + \ldots + \lambda_{n-k+1} = \min_{S:\dim(S)=k} \sum_{j=1}^{k} p_j^T Z p_j,$$

where $\{p_j\}$ is any orthonormal basis for S. If $\lambda_{n-k} > \lambda_{n-k+1}$, then $\operatorname{span}\{v_n, \ldots, v_{n-k+1}\}$ is the unique minimizing subspace.

Recall that a matrix Z is positive semi-definite if $x^T Z x \geq 0$ for all non-zero x.

Fact 18. *Suppose that the matrix*

$$Z = \begin{bmatrix} A & B^T \\ B & D \end{bmatrix}$$

is symmetric positive semi-definite and that A and D are square submatrices. Then $\|B\| \leq \sqrt{\|A\|\|D\|}$.

Proof. Let y and x be the top left and right singular vectors of B, so that $y^T B x = \|B\|$. Because Z is positive semi-definite, we have that for any real γ,

$$0 \leq [\gamma x^T \; y^T] Z [\gamma x^T \; y^T]^T = \gamma^2 x^T A x + 2\gamma y^T B x + y^T D y.$$

This is a quadratic polynomial in γ that can have only one real root. Therefore the discriminant must be non-positive:

$$0 \geq 4(y^T B x)^2 - 4(x^T A x)(y^T D y).$$

We conclude that

$$\|B\| = y^T Bx \leq \sqrt{(x^T Ax)(y^T Dy)} \leq \sqrt{\|A\|\|D\|}.$$

□

6.5.2 The Fisher Criterion and Isotropy

We begin with the proof of the lemma that for an isotropic mixture the Fisher subspace is the same as the intermean subspace.

Proof of Lemma 16. By Definition 4 for an isotropic distribution, the Fisher subspace minimizes

$$J(S) = E[\|\text{proj}_S(x - \mu_{\ell(x)})\|^2] = \sum_{j=1}^{k-1} p_j^T \Sigma p_j,$$

where $\{p_j\}$ is an orthonormal basis for S.

By Fact 17, one minimizing subspace is the span of the smallest $k-1$ eigenvectors of the matrix Σ, i.e. $v_{n-k+2}, \ldots, v_n$. Because the distribution is isotropic,

$$\Sigma = I - \sum_{i=1}^{k} w_i \mu_i \mu_i^T.$$

and these vectors become the largest eigenvectors of $\sum_{i=1}^{k} w_i \mu_i \mu_i^T$. Clearly, we have $\text{span}\{v_{n-k+2}, \ldots, v_n\} \subseteq \text{span}\{\mu_1, \ldots, \mu_k\}$, but both spans have dimension $k-1$ making them equal. This also implies that

$$1 - \lambda_{n-k+2}(\Sigma) = v_{n-k+2}^T \sum_{i=1}^{k} w_i \mu_i \mu_i^T v_{n-k+2} > 0.$$

Thus, $\lambda_{n-k+2}(\Sigma) < 1$. On the other hand v_{n-k+1}, must be orthogonal every μ_i, so $\lambda_{n-k+1}(\Sigma) = 1$. Therefore, $\lambda_{n-k+1}(\Sigma) > \lambda_{n-k+2}(\Sigma)$ and by Fact 17 $\text{span}\{v_{n-k+2}, \ldots, v_n\} = \text{span}\{\mu_1, \ldots, \mu_k\}$ is the unique minimizing subspace. □

It follows directly that under the conditions of Lemma 16, the overlap may be characterized as

$$\phi = \lambda_{n-k+2}(\Sigma) = 1 - \lambda_{k-1}\left(\sum_{i=1}^{k} w_i \mu_i \mu_i^T\right).$$

For clarity of the analysis, we will assume that Step 1 of the algorithm produces a perfectly isotropic mixture. Theorem 7 gives a bound on the required number of samples to make the distribution nearly isotropic, and as our analysis shows, our algorithm is robust to small estimation errors.

We will also assume for convenience of notation that the the unit vectors along the first $k-1$ coordinate axes $e_1, \ldots e_{k-1}$ span the intermean (i.e. Fisher) subspace. That is, $F = \text{span}\{e_1, \ldots, e_{k-1}\}$. When considering this subspace it will be convenient to be able to refer to projection of the mean vectors to this subspace. Thus, we define $\tilde{\mu_i} \in R^{k-1}$ to be the first $k-1$ coordinates of μ_i; the remaining coordinates are all zero. In other terms,

$$\tilde{\mu_i} = \begin{bmatrix} I_{k-1} & 0 \end{bmatrix} \mu_i \ .$$

In this coordinate system the covariance matrix of each component has a particular structure, which will be useful for our analysis. For the rest of this paper we fix the following notation: an isotropic mixture is defined by $\{w_i, \mu_i, \Sigma_i\}$. We assume that $\text{span}\{e_1, \ldots, e_{k-1}\}$ is the intermean subspace and A_i,B_i, and D_i are defined such that

$$w_i \Sigma_i = \begin{bmatrix} A_i & B_i^T \\ B_i & D_i \end{bmatrix} \tag{10}$$

where A_i is a $(k-1) \times (k-1)$ submatrix and D_i is a $(n-k+1) \times (n-k+1)$ submatrix.

Lemma 19 (Covariance Structure). *Using the above notation,*

$$\|A_i\| \leq \phi \ , \|D_i\| \leq 1 \ , \|B_i\| \leq \sqrt{\phi}$$

for all components i.

Proof of Lemma 19. Because $\text{span}\{e_1, \ldots, e_{k-1}\}$ is the Fisher subspace

$$\phi = \max_{v \in \mathbb{R}^{k-1}} \frac{1}{\|v\|^2} \sum_{i=1}^{k} v^T A_i v = \left\| \sum_{i=1}^{k} A_i \right\|_2 .$$

Also $\sum_{i=1}^{k} D_i = I$, so $\|\sum_{i=1}^{k} D_i\| = 1$. Each matrix $w_i\Sigma_i$ is positive definite, so the principal minors A_i,D_i must be positive definite as well. Therefore, $\|A_i\| \leq \phi$, $\|D_i\| \leq 1$, and $\|B_i\| \leq \sqrt{\|A_i\|\|D_i\|} = \sqrt{\phi}$ using Fact 18. □

For small ϕ, the covariance between intermean and non-intermean directions, i.e. B_i, is small. For $k = 2$, this means that all densities will have a "nearly parallel pancake" shape. In general, it means that $k-1$ of the principal axes of the Gaussians will lie close to the intermean subspace.

We conclude this section with a proposition connecting, for $k = 2$, the overlap to a standard notion of separation between two distributions, so that Theorem 4 becomes an immediate corollary of Theorem 5.

Proposition 20. *If there exists a unit vector p such that*

$$|p^T(\mu_1 - \mu_2)| > t(\sqrt{p^T w_1 \Sigma_1 p} + \sqrt{p^T w_2 \Sigma_2 p}),$$

then the overlap $\phi \leq J(p) \leq (1 + w_1 w_2 t^2)^{-1}$.

Proof of Proposition 20. Since the mean of the distribution is at the origin, we have $w_1 p^T \mu_1 = -w_2 p^T \mu_2$. Thus,

$$\begin{aligned} |p^T\mu_1 - p^T\mu_2|^2 &= (p^T\mu_1)^2 + (p^T\mu_2)^2 + 2|p^T\mu_1||p^T\mu_2| \\ &= (w_1 p^T\mu_1)^2 \left(\frac{1}{w_1^2} + \frac{1}{w_2^2} + \frac{2}{w_1 w_2}\right), \end{aligned}$$

using $w_1 + w_2 = 1$. We rewrite the last factor as

$$\frac{1}{w_1^2} + \frac{1}{w_2^2} + \frac{2}{w_1 w_2} = \frac{w_1^2 + w_2^2 + 2w_1 w_2}{w_1^2 w_2^2} = \frac{1}{w_1^2 w_2^2} = \frac{1}{w_1 w_2}\left(\frac{1}{w_1} + \frac{1}{w_2}\right).$$

Again, using the fact that $w_1 p^T \mu_1 = -w_2 p^T \mu_2$, we have that

$$\begin{aligned} |p^T\mu_1 - p^T\mu_2|^2 &= \frac{(w_1 p^T\mu_1)^2}{w_1 w_2}\left(\frac{1}{w_1} + \frac{1}{w_2}\right) \\ &= \frac{w_1(p^T\mu_1)^2 + w_2(p^T\mu_2)^2}{w_1 w_2}. \end{aligned}$$

Thus, by the separation condition

$$w_1(p^T\mu_1)^2 + w_2(p^T\mu_2)^2 = w_1w_2|p^T\mu_1 - p^T\mu_2|^2 \geq w_1w_2t^2(p^Tw_1\Sigma_1p + p^Tw_2\Sigma_2p).$$

To bound $J(p)$, we then argue

$$\begin{aligned} J(p) &= \frac{p^Tw_1\Sigma_1p + p^Tw_2\Sigma_2p}{w_1(p^T\Sigma_1p + (p^T\mu_1)^2) + w_2(p^T\Sigma_2p + (p^T\mu_2)^2)} \\ &= 1 - \frac{w_1(p^T\mu_1)^2 + w_2(p^T\mu_2)^2}{w_1(p^T\Sigma_1p + (p^T\mu_1)^2) + w_2(p^T\Sigma_2p + (p^T\mu_2)^2)} \\ &\leq 1 - \frac{w_1w_2t^2(w_1p^T\Sigma_1p + w_2p^T\Sigma_2p)}{w_1(p^T\Sigma_1p + (p^T\mu_1)^2) + w_2(p^T\Sigma_2p + (p^T\mu_2)^2)} \\ &\leq 1 - w_1w_2t^2J(p), \end{aligned}$$

and $J(p) \leq 1/(1 + w_1w_2t^2)$. □

6.6 Approximation of the Reweighted Moments

Our algorithm works by computing the first and second reweighted moments of a point set from $\mathcal{F}$. In this section, we examine how the reweighting affects the moments of a single component and then give some approximations for the first and second moments of the entire mixture.

6.6.1 Single Component

The first step is to characterize how the reweighting affects the moments of a single component. Specifically, we will show for any function f (and therefore x and xx^T in particular) that for $\alpha > 0$,

$$E\left[f(x)\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right] = \sum_i w_i\rho_iE_i\left[f(y_i)\right],$$

Here, $E_i[\cdot]$ denotes expectation taken with respect to the component i, the quantity $\rho_i = E_i\left[\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right]$, and y_i is a Gaussian variable with parameters slightly perturbed from the original ith component.

Claim 21. *If $\alpha = n/w_{\min}$, the quantity $\rho_i = E_i\left[\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right]$ is at least $1/2$.*

Proof. Because the distribution is isotropic, for any component i, $w_i E_i[\|x\|^2] \leq n$. Therefore,

$$\rho_i = E_i\left[\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right] \geq E_i\left[1 - \frac{\|x\|^2}{2\alpha}\right] \geq 1 - \frac{1}{2\alpha}\frac{n}{w_i} \geq \frac{1}{2}.$$

□

Lemma 22 (Reweighted Moments of a Single Component). *For any $\alpha > 0$, with respect to a single component i of the mixture*

$$E_i\left[x\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right] = \rho_i(\mu_i - \frac{1}{\alpha}\Sigma_i\mu_i + f)$$

and

$$E_i\left[xx^T\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right] = \rho(\Sigma_i + \mu_i\mu_i^T - \frac{1}{\alpha}(\Sigma_i\Sigma_i + \mu_i\mu_i^T\Sigma_i + \Sigma_i\mu_i\mu_i^T) + F)$$

where $\|f\|, \|F\| = O(\alpha^{-2})$.

We first establish the following claim.

Claim 23. *Let x be a random variable distributed according to the normal distribution $N(\mu, \Sigma)$ and let $\Sigma = Q\Lambda Q^T$ be the singular value decomposition of Σ with $\lambda_1, \ldots, \lambda_n$ being the diagonal elements of Λ. Let $W = \mathrm{diag}(\alpha/(\alpha+\lambda_1), \ldots, \alpha/(\alpha+\lambda_n))$. Finally, let y be a random variable distributed according to $N(QWQ^T\mu, QW\Lambda Q^T)$. Then for any function $f(x)$,*

$$E\left[f(x)\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right] = \det(W)^{1/2}\exp\left(-\frac{\mu^T QWQ^T\mu}{2\alpha}\right)E\left[f(y)\right].$$

Proof of Claim 23. We assume that $Q = I$ for the initial part of the proof. From the definition of a Gaussian distribution, we have

$$E\left[f(x)\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right]$$
$$= \det(\Lambda)^{-1/2}(2\pi)^{-n/2}\int_{\mathbb{R}^n} f(x)\exp\left(-\frac{x^Tx}{2\alpha} - \frac{(x-\mu)^T\Lambda^{-1}(x-\mu)}{2}\right). \quad (11)$$

Because Λ is diagonal, we may write the exponents on the right hand side as

$$\sum_{i=1}^{n} x_i^2\alpha^{-1} + (x_i - \mu_i)^2\lambda_i^{-1} = \sum_{i=1}^{n} x_i^2(\lambda^{-1} + \alpha^{-1}) - 2x_i\mu_i\lambda_i^{-1} + \mu_i^2\lambda_i^{-1}.$$

Completing the square gives the expression

$$\sum_{i=1}^{n} \left(x_i - \mu_i\frac{\alpha}{\alpha+\lambda_i}\right)^2 \left(\frac{\lambda_i\alpha}{\alpha+\lambda_i}\right)^{-1} + \mu_i^2\lambda_i^{-1} - \mu_i^2\lambda_i^{-1}\frac{\alpha}{\alpha+\lambda_i}.$$

The last two terms can be simplified to $\mu_i^2/(\alpha+\lambda_i)$. In matrix form the exponent becomes

$$(x - W\mu)^T (W\Lambda)^{-1} (x - W\mu) + \mu^T W\mu\alpha^{-1}.$$

For general Q, this becomes

$$\left(x - QWQ^T\mu\right)^T Q(W\Lambda)^{-1}Q^T \left(x - QWQ^T\mu\right) + \mu^T QWQ^T\mu\alpha^{-1}.$$

Now recalling the definition of the random variable y, we see

$$\begin{aligned} E\left[f(x)\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right] &= \det(\Lambda)^{-1/2}(2\pi)^{-n/2}\exp\left(-\frac{\mu^T QWQ^T\mu}{2\alpha}\right) \\ &\int_{\mathbb{R}^n} f(x)\exp\left(-\frac{1}{2}\left(x - QWQ^T\mu\right)^T Q(W\Lambda)^{-1}Q^T\left(x - QWQ^T\mu\right)\right) \\ &= \det(W)^{1/2}\exp\left(-\frac{\mu^T QWQ^T\mu}{2\alpha}\right) E\left[f(y)\right]. \end{aligned}$$

□

The proof of Lemma 22 is now straightforward.

Proof of Lemma 22. For simplicity of notation, we drop the subscript i from ρ_i, μ_i, Σ_i with the understanding that all statements of expectation apply to a single component. Using the notation of Claim 23, we have

$$\rho = E\left[\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right] = \det(W)^{1/2}\exp\left(-\frac{\mu^T QWQ^T\mu}{2\alpha}\right).$$

A diagonal entry of the matrix W can expanded as

$$\frac{\alpha}{\alpha+\lambda_i} = 1 - \frac{\lambda_i}{\alpha+\lambda_i} = 1 - \frac{\lambda_i}{\alpha} + \frac{\lambda_i^2}{\alpha(\alpha+\lambda_i)},$$

so that

$$W = I - \frac{1}{\alpha}\Lambda + \frac{1}{\alpha^2}W\Lambda^2.$$

Thus,

$$\begin{aligned} E\left[x\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right] &= \rho(QWQ^T\mu) \\ &= \rho(QIQ^T\mu - \frac{1}{\alpha}Q\Lambda Q^T\mu + \frac{1}{\alpha^2}QW\Lambda^2Q^T\mu) \\ &= \rho(\mu - \frac{1}{\alpha}\Sigma\mu + f), \end{aligned}$$

where $\|f\| = O(\alpha^{-2})$.

We analyze the perturbed covariance in a similar fashion.

$$\begin{aligned} E\left[xx^T\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right] &= \rho\left(Q(W\Lambda)Q^T + QWQ^T\mu\mu^TQWQ^T\right) \\ &= \rho\left(Q\Lambda Q^T - \frac{1}{\alpha}Q\Lambda^2Q^T + \frac{1}{\alpha^2}QW\Lambda^3Q^T \right. \\ &\quad \left. +(\mu - \frac{1}{\alpha}\Sigma\mu + f)(\mu - \frac{1}{\alpha}\Sigma\mu + f)^T\right) \\ &= \rho\left(\Sigma + \mu\mu^T - \frac{1}{\alpha}(\Sigma\Sigma + \mu\mu^T\Sigma + \Sigma\mu\mu^T) + F\right), \end{aligned}$$

where $\|F\| = O(\alpha^{-2})$. □

6.6.2 Mixture Moments

The second step is to approximate the first and second moments of the entire mixture distribution. Let ρ be the vector where $\rho_i = E_i\left[\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right]$ and let $\bar{\rho}$ be the average of the ρ_i. We also define

$$u \equiv E\left[x\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right] = \sum_{i=1}^{k} w_i\rho_i\mu_i - \frac{1}{\alpha}\sum_{i=1}^{k} w_i\rho_i\Sigma_i\mu_i + f \tag{12}$$

$$\begin{aligned} M &\equiv E\left[xx^T\exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right] \\ &= \sum_{i=1}^{k} w_i\rho_i(\Sigma_i + \mu_i\mu_i^T - \frac{1}{\alpha}(\Sigma_i\Sigma_i + \mu_i\mu_i^T\Sigma_i + \Sigma_i\mu_i\mu_i^T)) + F \end{aligned} \tag{13}$$

with $\|f\| = O(\alpha^{-2})$ and $\|F\| = O(\alpha^{-2})$. We denote the estimates of these quantities computed from samples by $\hat{u}$ and $\hat{M}$ respectively.

Lemma 24. *Let* $v = \sum_{i=1}^{k} \rho_i w_i \mu_i$. *Then*

$$\|u - v\|^2 \leq \frac{4k^2}{\alpha^2 w_{\min}} \phi.$$

Proof of Lemma 24. We argue from (10) and (12) that

$$\begin{aligned} \|u - v\| &= \frac{1}{\alpha} \left\| \sum_{i=1}^{k} w_i \rho_i \Sigma_i \mu_i \right\| + O(\alpha^{-2}) \\ &\leq \frac{1}{\alpha \sqrt{w_{\min}}} \sum_{i=1}^{k} \rho_i \|(w_i \Sigma_i)(\sqrt{w_i}\mu_i)\| + O(\alpha^{-2}) \\ &\leq \frac{1}{\alpha \sqrt{w_{\min}}} \sum_{i=1}^{k} \rho_i \|[A_i, B_i^T]^T\| \|(\sqrt{w_i}\mu_i)\| + O(\alpha^{-2}). \end{aligned}$$

From isotropy, it follows that $\|\sqrt{w_i}\mu_i\| \leq 1$. To bound the other factor, we argue

$$\|[A_i, B_i^T]^T\| \leq \sqrt{2} \max\{\|A_i\|, \|B_i\|\} \leq \sqrt{2\phi}.$$

Therefore,

$$\|u - v\|^2 \leq \frac{2k^2}{\alpha^2 w_{\min}} \phi + O(\alpha^{-3}) \leq \frac{4k^2}{\alpha^2 w_{\min}} \phi,$$

for sufficiently large n, as $\alpha \geq n/w_{\min}$. □

Lemma 25. *Let*

$$\Gamma = \begin{bmatrix} \sum_{i=1}^{k} \rho_i (w_i \tilde{\mu}_i \tilde{\mu}_i{}^T + A_i) & 0 \\ 0 & \sum_{i=1}^{k} \rho_i D_i - \frac{\rho_i}{w_i \alpha} D_i^2 \end{bmatrix}.$$

If $\|\rho - 1\bar{\rho}\|_\infty < 1/(2\alpha)$, *then*

$$\|M - \Gamma\|_2^2 \leq \frac{16^2 k^2}{{w_{\min}}^2 \alpha^2} \phi.$$

Before giving the proof, we summarize some of the necessary calculation in the following claim.

Claim 26. *The matrix of second moments*

$$M = E\left[xx^T \exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right] = \begin{bmatrix} \Gamma_{11} & 0 \\ 0 & \Gamma_{22} \end{bmatrix} + \begin{bmatrix} \Delta_{11} & \Delta_{21}^T \\ \Delta_{21} & \Delta_{22} \end{bmatrix} + F,$$

where

$$\begin{aligned}
\Gamma_{11} &= \sum_{i=1}^{k} \rho_i (w_i \tilde{\mu}_i \tilde{\mu}_i{}^T + A_i) \\
\Gamma_{22} &= \sum_{i=1}^{k} \rho_i D_i - \frac{\rho_i}{w_i \alpha} D_i^2 \\
\\
\Delta_{11} &= -\sum_{i=1}^{k} \frac{\rho_i}{w_i \alpha} B_i^T B_i + \frac{\rho_i}{w_i \alpha} \left(w_i \tilde{\mu}_i \tilde{\mu}_i{}^T A_i + w_i A_i \tilde{\mu}_i \tilde{\mu}_i{}^T + A_i^2 \right) \\
\Delta_{21} &= \sum_{i=1}^{k} \rho_i B_i - \frac{\rho_i}{w_i \alpha} \left(B_i (w_i \tilde{\mu}_i \tilde{\mu}_i{}^T) + B_i A_i + D_i B_i \right) \\
\Delta_{22} &= -\sum_{i=1}^{k} \frac{\rho_i}{w_i \alpha} B_i B_i^T,
\end{aligned}$$

and $\|F\| = O(\alpha^{-2})$.

Proof. The calculation is straightforward. □

Proof of Lemma 25. We begin by bounding the 2-norm of each of the blocks. Since $\|w_i \tilde{\mu}_i \tilde{\mu}_i{}^T\| < 1$ and $\|A_i\| \leq \phi$ and $\|B_i\| \leq \sqrt{\phi}$, we can bound

$$\begin{aligned}
\|\Delta_{11}\| &= \max_{\|y\|=1} \sum_{i=1}^{k} \frac{\rho_i}{w_i \alpha} y^T B_i^T B_i y^T - \frac{\rho_i}{w_i \alpha} y^T \left(w_i \tilde{\mu}_i \tilde{\mu}_i{}^T A_i + w_i A_i \tilde{\mu}_i \tilde{\mu}_i{}^T + A_i^2 \right) y + O(\alpha^{-2}) \\
&\leq \sum_{i=1}^{k} \frac{\rho_i}{w_i \alpha} \|B_i\|^2 + \frac{\rho_i}{w_i \alpha} (2\|A\| + \|A\|^2) + O(\alpha^{-2}) \\
&\leq \frac{4k}{w_{\min} \alpha} \phi + O(\alpha^{-2}).
\end{aligned}$$

By a similar argument, $\|\Delta_{22}\| \leq k\phi/(w_{\min}\alpha) + O(\alpha^{-2})$. For Δ_{21}, we observe that

$\sum_{i=1}^{k} B_i = 0$. Therefore,

$$\begin{aligned}
\|\Delta_{21}\| &\leq \left\|\sum_{i=1}^{k}(\rho_i - \bar{\rho})B_i\right\| + \left\|\sum_{i=1}^{k}\frac{\rho_i}{w_i\alpha}\left(B_i(w_i\tilde{\mu}_i\tilde{\mu}_i^T) + B_iA_i + D_iB_i\right)\right\| + O(\alpha^{-2}) \\
&\leq \sum_{i=1}^{k}|\rho_i - \bar{\rho}|\|B_i\| + \sum_{i=1}^{k}\frac{\rho_i}{w_i\alpha}\left(\|B_i(w_i\tilde{\mu}_i\tilde{\mu}_i^T)\| + \|B_iA_i\| + \|D_iB_i\|\right) + O(\alpha^{-2}) \\
&\leq k\|\rho - 1\bar{\rho}\|_\infty\sqrt{\phi} + \sum_{i=1}^{k}\frac{\rho_i}{w_i\alpha}(\sqrt{\phi} + \phi\sqrt{\phi} + \sqrt{\phi}) + O(\alpha^{-2}) \\
&\leq k\|\rho - 1\bar{\rho}\|_\infty\sqrt{\phi} + \frac{3k\bar{\rho}}{w_{\min}\alpha}\sqrt{\phi} \\
&\leq \frac{7k}{2w_{\min}\alpha}\sqrt{\phi} + O(\alpha^{-2}).
\end{aligned}$$

Thus, we have $\max\{\|\Delta_{11}\|, \|\Delta_{22}\|, \|\Delta_{21}\|\} \leq 4k\sqrt{\phi}/(w_{\min}\alpha) + O(\alpha^{-2})$, so that

$$\begin{aligned}
\|M - \Gamma\| &\leq \|\Delta\| + O(\alpha^{-2}) \\
&\leq 2\max\{\|\Delta_{11}\|, \|\Delta_{22}\|, \|\Delta_{21}\|\} \leq \frac{8k}{w_{\min}\alpha}\sqrt{\phi} + O(\alpha^{-2}) \leq \frac{16k}{w_{\min}\alpha}\sqrt{\phi}.
\end{aligned}$$

for sufficiently large n, as $\alpha \geq n/w_{\min}$. □

6.7 Sample Convergence

We now give some bounds on the convergence of the transformation to isotropy ($\hat{\mu} \to 0$ and $\hat{\Sigma} \to I$) and on the convergence of the reweighted sample mean $\hat{u}$ and sample matrix of second moments $\hat{M}$ to their expectations u and M. For the convergence of second moment matrices, we use the following lemma due to Rudelson [43], which was presented in this form in [44].

Lemma 27. *Let y be a random vector from a distribution D in R^n, with $\sup_D \|y\| = M$ and $\|\mathsf{E}(yy^T)\| \leq 1$. Let $y_1, \ldots, y_m$ be independent samples from D. Let*

$$\eta = CM\sqrt{\frac{\log m}{m}}$$

where C is an absolute constant. Then,

(i) If $\eta < 1$, then

$$\mathsf{E}\left(\|\frac{1}{m}\sum_{i=1}^{m} y_i y_i^T - \mathsf{E}(yy^T)\|\right) \leq \eta.$$

(ii) For every $t \in (0,1)$,

$$\mathsf{P}\left(\|\frac{1}{m}\sum_{i=1}^{m} y_i y_i^T - \mathsf{E}(yy^T)\| > t\right) \leq 2e^{-ct^2/\eta^2}.$$

This lemma is used to show that a distribution can be made nearly isotropic using only $O^*(kn)$ samples [43, 38]. The isotropic transformation is computed simply by estimating the mean and covariance matrix of a sample, and computing the affine transformation that puts the sample in isotropic position.

Theorem 7. *There is an absolute constant C such that for an isotropic mixture of k logconcave distributions, with probability at least $1-\delta$, a sample of size*

$$m > C\frac{kn\log^2(n/\delta)}{\epsilon^2}$$

gives a sample mean $\hat{\mu}$ and sample covariance $\hat{\Sigma}$ so that

$$\|\hat{\mu}\| \leq \epsilon \quad \text{and} \quad \|\hat{\Sigma} - I\| \leq \epsilon.$$

We now consider the reweighted moments.

Lemma 28. *Let $\epsilon, \delta > 0$ and let $\hat{\mu}$ be the reweighted sample mean of a set of m points drawn from an isotropic mixture of k Gaussians in n dimensions, where*

$$m \geq \frac{2n\alpha}{\epsilon^2}\log\frac{2n}{\delta}.$$

Then

$$\mathsf{P}\left[\|\hat{u} - u\| > \epsilon\right] \leq \delta$$

Proof. We first consider only a single coordinate of the vector $\hat{u}$.

$$y = x_1 \exp\left(-\|x\|^2/(2\alpha)\right) - u_1$$

and observe that

$$\left| x_1 \exp\left(-\frac{\|x\|^2}{2\alpha}\right)\right| \leq |x_1| \exp\left(-\frac{x_1^2}{2\alpha}\right) \leq \sqrt{\frac{\alpha}{e}} < \sqrt{\alpha}.$$

Thus, each term in the sum $m\hat{u}_1 = \sum_{j=1}^m y_j$ falls the range $[-\sqrt{\alpha} - u_1, \sqrt{\alpha} - u_1]$. We may therefore apply Hoeffding's inequality to show that

$$\mathsf{P}\left[|\hat{u}_1 - u_1| \geq \epsilon/\sqrt{n}\right] \leq 2\exp\left(-\frac{2m^2(\epsilon/\sqrt{n})^2}{m \cdot (2\sqrt{\alpha})^2}\right) \leq 2\exp\left(-\frac{m\epsilon^2}{2\alpha n}\right) \leq \frac{\delta}{n}.$$

Taking the union bound over the n coordinates, we have that with probability $1 - \delta$ the error in each coordinate is at most $\epsilon/\sqrt{n}$, which implies that $\|\hat{u} - u\| \leq \epsilon$. □

Lemma 29. *Let $\epsilon, \delta > 0$ and let $\hat{M}$ be the reweighted sample matrix of second moments for a set of m points drawn from an isotropic mixture of k Gaussians in n dimensions, where*

$$m \geq C_1 \frac{n\alpha}{\epsilon^2} \log \frac{n\alpha}{\delta}.$$

and C_1 is an absolute constant. Then

$$\mathsf{P}\left[\left\|\hat{M} - M\right\| > \epsilon\right] < \delta.$$

Proof. We will apply Lemma 27. Define $y = x \exp\left(-\|x\|^2/(2\alpha)\right)$. Then,

$$y_i^2 \leq x_i^2 \exp\left(-\frac{\|x\|^2}{\alpha}\right) \leq x_i^2 \exp\left(-\frac{x_i^2}{\alpha}\right) \leq \frac{\alpha}{e} < \alpha.$$

Therefore $\|y\| \leq \sqrt{\alpha n}$.

Next, since M is in isotropic position (we can assume this w.l.o.g.), we have for any unit vector v,

$$\mathsf{E}((v^T y)^2)) \leq \mathsf{E}((v^T x)^2) \leq 1$$

and so $\|\mathsf{E}(yy^T)\| \leq 1$.

Now we apply the second part of Lemma 27 with $\eta = \epsilon\sqrt{c/\ln(2/\delta)}$ and $t = \eta\sqrt{\ln(2/\delta)/c}$. This requires that

$$\eta = \frac{c\epsilon}{\ln(2/\delta)} \leq C\sqrt{\alpha n}\sqrt{\frac{\log m}{m}}$$

which is satisfied for our choice of m. □

Lemma 30. *Let X be a collection of m points drawn from a Gaussian with mean μ and variance σ^2. With probability $1-\delta$,*

$$|x-\mu| \leq \sigma\sqrt{2\log m/\delta}.$$

for every $x \in X$.

6.8 Finding a Vector near the Fisher Subspace

In this section, we use the approximations of Section 6.6 to show that the direction h chosen by step 3 of the algorithm is close to the intermean subspace. Finding such a direction is the most challenging part of the classification task and represents the main contribution of this work.

We first assume zero overlap and that the sample reweighted moments behave exactly according to expectation. In this case, the mean shift $\hat{u}$ becomes

$$v \equiv \sum_{i=1}^{k} w_i \rho_i \mu_i.$$

We can intuitively think of the components that have greater ρ_i as gaining mixing weight and those with smaller ρ_i as losing mixing weight. As long as the ρ_i are not all equal, we will observe some shift of the mean in the intermean subspace, i.e. Fisher subspace. Therefore, we may use this direction to partition the components. On the other hand, if all of the ρ_i are equal, then $\hat{M}$ becomes

$$\Gamma \equiv \begin{bmatrix} \sum_{i=1}^{k} \rho_i (w_i \tilde{\mu}_i \tilde{\mu}_i^T + A_i) & 0 \\ 0 & \sum_{i=1}^{k} \rho_i D_i - \frac{\rho_i}{w_i \alpha} D_i^2 \end{bmatrix} = \bar{\rho} \begin{bmatrix} I & 0 \\ 0 & I - \frac{1}{\alpha} \sum_{i=1}^{k} \frac{1}{w_i} D_i^2 \end{bmatrix}.$$

Notice that the second moments in the subspace $\text{span}\{e_1, \ldots, e_{k-1}\}$ are maintained while those in the complementary subspace are reduced by $\text{poly}(1/\alpha)$. Therefore, the top eigenvector will be in the intermean subspace, which is the Fisher subspace.

We now argue that this same strategy can be adapted to work in general, i.e., with nonzero overlap and sampling errors, with high probability. A critical aspect of

this argument is that the norm of the error term $\hat{M} - \Gamma$ depends only on ϕ and k and not the dimension of the data. See Lemma 25 and the supporting Lemma 19 and Fact 18.

Since we cannot know directly how imbalanced the ρ_i are, we choose the method of finding a separating direction according the norm of the vector $\|\hat{u}\|$. Recall that when $\|\hat{u}\| > \sqrt{w_{\min}}/(32\alpha)$ the algorithm uses $\hat{u}$ to determine the separating direction h. Lemma 31 guarantees that this vector is close to the Fisher subspace. When $\|\hat{u}\| \leq \sqrt{w_{\min}}/(32\alpha)$, the algorithm uses the top eigenvector of the covariance matrix $\hat{M}$. Lemma 32 guarantees that this vector is close to the Fisher subspace.

Lemma 31 (Mean Shift Method). *Let $\epsilon > 0$. There exists a constant C such that if $m_1 \geq Cn^4\text{poly}(k, {w_{\min}}^{-1}, \log n/\delta)$, then the following holds with probability $1 - \delta$. If $\|\hat{u}\| > \sqrt{w_{\min}}/(32\alpha)$ and*

$$\phi \leq \frac{{w_{\min}}^2\epsilon}{2^{14}k^2},$$

then

$$\frac{\|\hat{u}^T v\|}{\|\hat{u}\|\|v\|} \geq 1 - \epsilon.$$

Lemma 32 (Spectral Method). *Let $\epsilon > 0$. There exists a constant C such that if $m_1 \geq Cn^4\text{poly}(k, {w_{\min}}^{-1}, \log n/\delta)$, then the following holds with probability $1 - \delta$. Let $v_1, \ldots, v_{k-1}$ be the top $k-1$ eigenvectors of $\hat{M}$. If $\|\hat{u}\| \leq \sqrt{w_{\min}}/(32\alpha)$ and*

$$\phi \leq \frac{{w_{\min}}^2\epsilon}{640^2k^2}$$

then

$$\min_{v \in \text{span}\{v_1,\ldots,v_{k-1}\}, \|v\|=1} \|\text{proj}_F(v)\| \geq 1 - \epsilon.$$

6.8.1 Mean Shift

Proof of Lemma 31. We will make use of the following claim.

Claim 33. *For any vectors $a, b \neq 0$,*

$$\frac{|a^T b|}{\|a\|\|b\|} \geq \left(1 - \frac{\|a-b\|^2}{\max\{\|a\|^2, \|b\|^2\}}\right)^{1/2}.$$

By the triangle inequality, $\|\hat{u} - v\| \leq \|\hat{u} - u\| + \|u - v\|$. By Lemma 24,

$$\|u - v\| \leq \sqrt{\frac{4k^2}{\alpha^2 w_{\min}}\phi} = \sqrt{\frac{4k^2}{\alpha^2 w_{\min}} \cdot \frac{{w_{\min}}^2 \epsilon}{2^1 4k^2}} \leq \sqrt{\frac{w_{\min}\epsilon}{2^{12}\alpha^2}}.$$

By Lemma 28, for large m_1 we obtain the same bound on $\|\hat{u} - u\|$ with probability $1 - \delta$. Thus,

$$\|\hat{u} - v\| \leq \sqrt{\frac{w_{\min}\epsilon}{2^{10}\alpha^2}}.$$

Applying the claim gives

$$\begin{aligned}
\frac{\|\hat{u}^T v\|}{\|\hat{u}\|\|v\|} &\geq 1 - \frac{\|\hat{u} - v\|^2}{\|\hat{u}\|^2} \\
&\geq 1 - \frac{w_{\min}\epsilon}{2^{10}\alpha^2} \cdot \frac{32^2\alpha^2}{w_{\min}} \\
&= 1 - \epsilon.
\end{aligned}$$

□

Proof of Claim 33. Without loss of generality, assume $\|u\| \geq \|v\|$ and fix the distance $\|u - v\|$. In order to maximize the angle between u and v, the vector v should be chosen so that it is tangent to the sphere centered at u with radius $\|u - v\|$. Hence, the vectors u,v,$(u - v)$ form a right triangle where $\|u\|^2 = \|v\|^2 + \|u - v\|^2$. For this choice of v, let θ be the angle between u and v so that

$$\frac{u^T v}{\|u\|\|v\|} = \cos\theta = (1 - \sin^2\theta)^{1/2} = \left(1 - \frac{\|u - v\|^2}{\|u\|^2}\right)^{1/2}.$$

□

6.8.2 Spectral Method

We first show that the smallness of the mean shift $\hat{u}$ implies that the coefficients ρ_i are sufficiently uniform to allow us to apply the spectral method.

Claim 34 (Small Mean Shift Implies Balanced Second Moments). *If* $\|\hat{u}| \leq \sqrt{w_{\min}}/(32\alpha)$ *and*

$$\sqrt{\phi} \leq \frac{w_{\min}}{64k},$$

then

$$\|\rho - 1\bar{\rho}\|_2 \leq \frac{1}{8\alpha}.$$

Proof. Let $q_1, \ldots, q_k$ be the right singular vectors of the matrix $U = [w_1\mu_1, \ldots, w_k\mu_k]$ and let $\sigma_i(U)$ be the ith largest singular value. Because $\sum_{i=1}^k w_i\mu_i = 0$, we have that $\sigma_k(U) = 0$ and $q_k = 1/\sqrt{k}$. Recall that ρ is the k vector of scalars $\rho_1, \ldots, \rho_k$ and that $v = U\rho$. Then

$$\begin{aligned}
\|v\|^2 &= \|U\rho\|^2 \\
&= \sum_{i=1}^{k-1} \sigma_i(U)^2 (q_i^T\rho)^2 \\
&\geq \sigma_{k-1}(U)^2 \|\rho - q_k(q_k^T\rho)\|_2^2 \\
&= \sigma_{k-1}(U)^2 \|\rho - 1\bar{\rho}\|_2^2.
\end{aligned}$$

Because $q_{k-1} \in \text{span}\{\mu_1, \ldots, \mu_k\}$, we have that $\sum_{i=1}^k w_i q_{k-1}^T \mu_i \mu_i^T q_{k-1} \geq 1 - \phi$. Therefore,

$$\begin{aligned}
\sigma_{k-1}(U)^2 &= \|Uq_{k-1}\|^2 \\
&= q_{k-1}^T \left(\sum_{i=1}^k w_i^2 \mu_i \mu_i^T \right) q_{k-1} \\
&\geq w_{\min} q_{k-1}^T \left(\sum_{i=1}^k w_i \mu_i \mu_i^T \right) q_{k-1} \\
&\geq w_{\min}(1 - \phi).
\end{aligned}$$

Thus, we have the bound

$$\|\rho - 1\bar{\rho}\|_\infty \leq \frac{1}{\sqrt{(1-\phi)w_{\min}}} \|v\| \leq \frac{2}{\sqrt{w_{\min}}} \|v\|.$$

By the triangle inequality $\|v\| \leq \|\hat{u}\| + \|\hat{u} - v\|$. As argued in Lemma 24,

$$\|\hat{u} - v\| \leq \sqrt{\frac{4k^2}{\alpha^2 w_{\min}} \phi} = \sqrt{\frac{4k^2}{\alpha^2 w_{\min}} \cdot \frac{{w_{\min}}^2}{64^2 k^2}} =\leq \frac{\sqrt{w_{\min}}}{32\alpha}.$$

Thus,

$$\begin{aligned}\|\rho - 1\bar{\rho}\|_\infty &\leq \frac{2\bar{\rho}}{\sqrt{w_{\min}}}\|v\| \\ &\leq \frac{2\bar{\rho}}{\sqrt{w_{\min}}}\left(\frac{\sqrt{w_{\min}}}{32\alpha} + \frac{\sqrt{w_{\min}}}{32\alpha}\right) \\ &\leq \frac{1}{8\alpha}.\end{aligned}$$

□

We next show that the top $k-1$ principal components of Γ span the intermean subspace and put a lower bound on the spectral gap between the intermean and non-intermean components.

Lemma 35 (Ideal Case). *If $\|\rho - 1\bar{\rho}\|_\infty \leq 1/(8\alpha)$, then*

$$\lambda_{k-1}(\Gamma) - \lambda_k(\Gamma) \geq \frac{1}{4\alpha},$$

and the top $k-1$ eigenvectors of Γ span the means of the components.

Proof of Lemma 35. We first bound $\lambda_{k-1}(\Gamma_{11})$. Recall that

$$\Gamma_{11} = \sum_{i=1}^{k} \rho_i(w_i\tilde{\mu}_i\tilde{\mu}_i{}^T + A_i).$$

Thus,

$$\begin{aligned}\lambda_{k-1}(\Gamma_{11}) &= \min_{\|y\|=1} \sum_{i=1}^{k} \rho_i y^T(w_i\tilde{\mu}_i\tilde{\mu}_i{}^T + A_i)y \\ &\geq \bar{\rho} - \max_{\|y\|=1} \sum_{i=1}^{k} (\bar{\rho} - \rho_i) y^T(w_i\tilde{\mu}_i\tilde{\mu}_i{}^T + A_i)y.\end{aligned}$$

We observe that $\sum_{i=1}^{k} y^T(w_i\tilde{\mu}_i\tilde{\mu}_i{}^T + A_i)y = 1$ and each term is non-negative. Hence the sum is bounded by

$$\sum_{i=1}^{k} (\bar{\rho} - \rho_i) y^T(w_i\tilde{\mu}_i\tilde{\mu}_i{}^T + A_i)y \leq \|\rho - 1\bar{\rho}\|_\infty,$$

so,

$$\lambda_{k-1}(\Gamma_{11}) \geq \bar{\rho} - \|\rho - 1\bar{\rho}\|_\infty.$$

Next, we bound $\lambda_1(\Gamma_{22})$. Recall that

$$\Gamma_{22} = \sum_{i=1}^{k} \rho_i D_i - \frac{\rho_i}{w_i \alpha} D_i^2$$

and that for any $n-k$ vector y such that $\|y\| = 1$, we have $\sum_{i=1}^{k} y^T D_i y = 1$. Using the same arguments as above,

$$\begin{aligned} \lambda_1(\Gamma_{22}) &= \max_{\|y\|=1} \bar{\rho} + \sum_{i=1}^{k} (\rho_i - \bar{\rho}) y^T D_i y - \frac{\rho_i}{w_i \alpha} y^T D_i^2 y \\ &\leq \bar{\rho} + \|\rho - 1\bar{\rho}\|_\infty - \min_{\|y\|=1} \sum_{i=1}^{k} \frac{\rho_i}{w_i \alpha} y^T D_i^2 y. \end{aligned}$$

To bound the last sum, we observe that $\rho_i - \bar{\rho} = O(\alpha^{-1})$. Therefore

$$\sum_{i=1}^{k} \frac{\rho_i}{w_i \alpha} y^T D_i^2 y \geq \frac{\bar{\rho}}{\alpha} \sum_{i=1}^{k} \frac{1}{w_i} y^T D_i^2 y + O(\alpha^{-2}).$$

Without loss of generality, we may assume that $y = e_1$ by an appropriate rotation of the D_i. Let $D_i(\ell, j)$ be element in the ℓth row and jth column of the matrix D_i. Then the sum becomes

$$\begin{aligned} \sum_{i=1}^{k} \frac{1}{w_i} y^T D_i^2 y &= \sum_{i=1}^{k} \frac{1}{w_i} \sum_{j=1}^{n} D_j(1,j)^2 \\ &\geq \sum_{i=1}^{k} \frac{1}{w_i} D_j(1,1)^2. \end{aligned}$$

Because $\sum_{i=1}^{k} D_i = I$, we have $\sum_{i=1}^{k} D_i(1,1) = 1$. From the Cauchy-Schwartz inequality, it follows

$$\left(\sum_{i=1}^{k} w_i \right)^{1/2} \left(\sum_{i=1}^{k} \frac{1}{w_i} D_i(1,1)^2 \right)^{1/2} \geq \sum_{i=1}^{k} \sqrt{w_i} \frac{D_i(1,1)}{\sqrt{w_i}} = 1.$$

Since $\sum_{i=1}^{k} w_i = 1$, we conclude that $\sum_{i=1}^{k} \frac{1}{w_i} D_i(1,1)^2 \geq 1$. Thus, using the fact that $\bar{\rho} \geq 1/2$, we have

$$\sum_{i=1}^{k} \frac{\rho_i}{w_i \alpha} y^T D_i^2 y \geq \frac{1}{2\alpha}$$

Putting the bounds together

$$\lambda_{k-1}(\Gamma_{11}) - \lambda_1(\Gamma_{22}) \geq \frac{1}{2\alpha} - 2\|\rho - 1\bar{\rho}\|_\infty \geq \frac{1}{4\alpha}.$$

□

We now combine the facts that $\hat{M}$ is close to Γ and that Γ has a large eigenvalue gap between $k-1$ and k to prove Lemma 32. We require the following theorem due to Stewart [45].

Lemma 36 (Stewart's Theorem). *Suppose A and $A+E$ are n-by-n symmetric matrices and that*

$$A = \begin{array}{c} \\ \left[\begin{array}{cc} D_1 & 0 \\ 0 & D_2 \end{array}\right] \\ \begin{array}{cc} r & n-r \end{array} \end{array} \begin{array}{c} \\ \begin{array}{c} r \\ n-r \end{array} \\ \\ \end{array} \qquad E = \begin{array}{c} \\ \left[\begin{array}{cc} E_{11} & E_{21}^T \\ E_{21} & E_{22} \end{array}\right] \\ \begin{array}{cc} r & n-r \end{array} \end{array} \begin{array}{c} \\ \begin{array}{c} r \\ n-r \end{array} \\ \\ \end{array} .$$

Let the columns of V be the top r eigenvectors of the matrix $A+E$ and let P_2 be the matrix with columns $e_{r+1}, \ldots, e_n$. If $d = \lambda_r(D_1) - \lambda_1(D_2) > 0$ and

$$\|E\| \leq \frac{d}{5},$$

then

$$\|V^T P_2\| \leq \frac{4}{d}\|E_{21}\|_2.$$

The proof of Lemma 32 follows.

Proof of Lemma 32. Define $d = \lambda_{k-1}(\Gamma) - \lambda_k(\Gamma)$ and $E = \hat{M} - \Gamma$. We assume that the mean shift satisfies $\|\hat{u}\| \leq \sqrt{w_{\min}}/(32\alpha)$ and that ϕ is small. By Lemma 35, this implies that

$$d = \lambda_{k-1}(\Gamma) - \lambda_k(\Gamma) \geq \frac{1}{4\alpha}. \tag{14}$$

To bound $\|E\|$, we use the triangle inequality $\|E\| \leq \|\Gamma - M\| + \|M - \hat{M}\|$. Lemma 25 bounds the first term by

$$\|M - \Gamma\| \leq \sqrt{\frac{16^2k^2}{{w_{\min}}^2\alpha^2}\phi} = \sqrt{\frac{16^2k^2}{{w_{\min}}^2\alpha^2} \cdot \frac{{w_{\min}}^2\epsilon}{640^2k^2}} \leq \frac{1}{40\alpha}\sqrt{\epsilon}.$$

By Lemma 29, we obtain the same bound on $\|M - \hat{M}\|$ with probability $1-\delta$ for large enough m_1. Thus,

$$\|E\| \leq \frac{1}{20\alpha}\sqrt{\epsilon}.$$

Combining the bounds of (14) and (6.8.2), we have

$$\sqrt{1-(1-\epsilon)^2}d - 5\|E\| \geq \sqrt{1-(1-\epsilon)^2}\frac{1}{4\alpha} - 5\frac{1}{20\alpha}\sqrt{\epsilon} \geq 0,$$

as $\sqrt{1-(1-\epsilon)^2} \geq \sqrt{\epsilon}$. This implies both that $\|E\| \leq d/5$ and that $4\|E_{21}|/d < \sqrt{1-(1-\epsilon)^2}$, enabling us to apply Stewart's Lemma to the matrix pair Γ and $\hat{M}$.

By Lemma 35, the top $k-1$ eigenvectors of Γ, i.e. $e_1, \ldots, e_{k-1}$, span the means of the components. Let the columns of P_1 be these eigenvectors. Let the columns of P_2 be defined such that $[P_1, P_2]$ is an orthonormal matrix and let $v_1, \ldots, v_k$ be the top $k-1$ eigenvectors of $\hat{M}$. By Stewart's Lemma, letting the columns of V be $v_1, \ldots, v_{k-1}$, we have

$$\|V^T P_2\|_2 \leq \sqrt{1-(1-\epsilon)^2},$$

or equivalently,

$$\min_{v \in \text{span}\{v_1,\ldots,v_{k-1}\}, \|v\|=1} \|\text{proj}_F v\| = \sigma_{k-1}(V^T P_1) \geq 1-\epsilon.$$

□

6.9 Recursion

In this section, we show that for every direction h that is close to the intermean subspace, the "largest gap clustering" step produces a pair of complementary halfspaces that partitions $\mathbb{R}^n$ while leaving only a small part of the probability mass on the wrong side of the partition, small enough that with high probability, it does not affect the samples used by the algorithm.

Lemma 37. *Let $\delta, \delta' > 0$, where $\delta' \leq \delta/(2m_2)$, and let m_2 satisfy $m_2 \geq n/k \log(2k/\delta)$. Suppose that h is a unit vector such that*

$$\|\text{proj}_F(h)\| \geq 1 - \frac{w_{\min}}{2^{10}(k-1)^2 \log \frac{1}{\delta'}}.$$

Let $\mathcal{F}$ be a mixture of $k > 1$ Gaussians with overlap

$$\phi \leq \frac{w_{\min}}{2^9(k-1)^2} \log^{-1} \frac{1}{\delta'}.$$

Let X be a collection of m_2 points from $\mathcal{F}$ and let t be the midpoint of the largest gap in set $\{h^T x : x \in X\}$. With probability $1-\delta$, the halfspace $H_{h,t}$ has the following property. For a random sample y from $\mathcal{F}$ either

$$y, \mu_{\ell(y)} \in H_{h,t} \text{ or } y, \mu_{\ell(y)} \notin H_{h,t}$$

with probability $1-\delta'$.

Proof of Lemma 37. The idea behind the proof is simple. We first show that two of the means are at least a constant distance apart. We then bound the width of a component along the direction h, i.e. the maximum distance between two points belonging to the same component. If the width of each component is small, then clearly the largest gap must fall between components. Setting t to be the midpoint of the gap, we avoid cutting any components.

We first show that at least one mean must be far from the origin in the direction h. Let the columns of P_1 be the vectors $e_1, \ldots, e_{k-1}$. The span of these vectors is also the span of the means, so we have

$$\begin{aligned}
\max_i (h^T \mu_i)^2 &= \max_i (h^T P_1 P_1^T \mu_i)^2 \\
&= \|P_1^T h\|^2 \max_i \left(\frac{(P_1^T h)^T}{\|P_1 h\|} \tilde{\mu}_i \right)^2 \\
&\geq \|P_1^T h\|^2 \sum_{i=1}^{k} w_i \left(\frac{(P_1^T h)^T}{\|P_1 h\|} \tilde{\mu}_i \right)^2 \\
&\geq \|P_1^T h\|^2 (1-\phi) \\
&> \frac{1}{2}.
\end{aligned}$$

Since the origin is the mean of the means, we conclude that the maximum distance between two means in the direction h is at least $1/2$. Without loss of generality, we assume that the interval $[0, 1/2]$ is contained between two means projected to h.

We now show that every point x drawn from component i falls in a narrow interval when projected to h. That is, x satisfies $h^T x \in b_i$, where $b_i = [h^T \mu_i - (8(k -$

$1))^{-1}, h^T\mu_i + (8(k-1))^{-1}]$. We begin by examining the variance along h. Let $e_k, \ldots, e_n$ be the columns of the matrix n-by-$(n-k+1)$ matrix P_2. Recall from (10) that $P_1^T w_i \Sigma_i P_1 = A_i$, that $P_2^T w_i \Sigma_i P_1 = B_i$, and that $P_2^T w_i \Sigma_i P_2 = D_i$. The norms of these matrices are bounded according to Lemma 19. Also, the vector $h = P_1 P_1^T h + P_2 P_2^T h$. For convenience of notation we define ϵ such that $\|P_1^T h\| = 1 - \epsilon$. Then $\|P_2^T h\|^2 = 1 - (1-\epsilon)^2 \leq 2\epsilon$. We now argue

$$\begin{aligned} h^T w_i \Sigma_i h &\leq \left(h^T P_1 A_i P_1^T h + 2h^T P_2 B_i P_1 h + h^T P_2^T D_i P_2 h\right) \\ &\leq 2\left(h^T P_1 A_i P_1^T h + h^T P_2 D_i P_2^T h\right) \\ &\leq 2(\|P_1^T h\|^2 \|A_i\| + \|P_2^T h\|^2 \|\|D_i\|) \\ &\leq 2(\phi + 2\epsilon). \end{aligned}$$

Using the assumptions about ϕ and ϵ, we conclude that the maximum variance along h is at most

$$\max_i h^T \Sigma_i h \leq \frac{2}{w_{\min}} \left(\frac{w_{\min}}{2^9 (k-1)^2} \log \frac{1}{\delta'} + 2 \frac{w_{\min}}{2^{10}(k-1)^2} \log \frac{1}{\delta'} \right) \leq \left(2^7 (k-1)^2 \log 1/\delta'\right)^{-1}.$$

We now translate these bounds on the variance to a bound on the difference between the minimum and maximum points along the direction h. By Lemma 30, with probability $1 - \delta/2$

$$|h^T(x - \mu_{\ell(x)})| \leq \sqrt{2h^T \Sigma_i h \log(2m_2/\delta)} \leq \frac{1}{8(k-1)} \cdot \frac{\log(2m_2/\delta)}{\log(1/\delta')} \leq \frac{1}{8(k-1)}.$$

Thus, with probability $1 - \delta/2$, every point from X falls into the union of intervals $b_1 \cup \ldots \cup b_k$ where $b_i = [h^T\mu_i - (8(k-1))^{-1}, h^T\mu_i + (8(k-1))^{-1}]$. Because these intervals are centered about the means, at least the equivalent of one interval must fall outside the range $[0, 1/2]$, which we assumed was contained between two projected means. Thus, the measure of subset of $[0, 1/2]$ that does not fall into one of the intervals is

$$\frac{1}{2} - (k-1)\frac{1}{4(k-1)} = \frac{1}{4}.$$

This set can be cut into at most $k-1$ intervals, so the smallest possible gap between these intervals is $(4(k-1))^{-1}$, which is exactly the width of an interval.

Because $m_2 = k/w_{\min}\log(2k/\delta)$ the set X contains at least one sample from every component with probability $1-\delta/2$. Overall, with probability $1-\delta$ every component has at least one sample and all samples from component i fall in b_i. Thus, the largest gap between the sampled points will not contain one of the intervals $b_1, \ldots, b_k$. Moreover, the midpoint t of this gap must also fall outside of $b_1 \cup \ldots \cup b_k$, ensuring that no b_i is cut by t.

By the same argument given above, any single point y from $\mathcal{F}$ is contained in $b_1 \cup \ldots \cup b_k$ with probability $1-\delta'$ proving the Lemma. □

In the proof of the main theorem for large k, we will need to have every point sampled from $\mathcal{F}$ in the recursion subtree classified correctly by the halfspace, so we will assume δ' considerably smaller than m_2/δ.

The second lemma shows that all submixtures have smaller overlap to ensure that all the relevant lemmas apply in the recursive steps.

Lemma 38. *The removal of any subset of components cannot induce a mixture with greater overlap than the original.*

Proof of Lemma 38. Suppose that the components $j+1, \ldots k$ are removed from the mixture. Let $\omega = \sum_{i=1}^{j} w_i$ be a normalizing factor for the weights. Then if $c = \sum_{i=1}^{j} w_i\mu_i = -\sum_{i=j+1}^{k} w_i\mu_i$, the induced mean is $\omega^{-1}c$. Let T be the subspace that minimizes the maximum overlap for the full k component mixture. We then argue that the overlap $\tilde{\phi}^2$ of the induced mixture is bounded by

$$\begin{aligned}\tilde{\phi} &= \min_{\dim(S)=j-1} \max_{v\in S} \frac{\omega^{-1}v^T\Sigma v}{\omega^{-1}\sum_{i=1}^{j} w_i v^T(\mu_i\mu_i^T - cc^T + \Sigma_i)v} \\ &\leq \max_{v\in \mathrm{span}\{e_1,\ldots,e_{k-1}\}\setminus\mathrm{span}\{\mu_{j+1},\ldots,\mu_k\}} \frac{\sum_{i=1}^{j} w_i v^T\Sigma_i v}{\sum_{i=1}^{j} w_i v^T(\mu_i\mu_i^T - cc^T + \Sigma_i)v}.\end{aligned}$$

Every $v \in \text{span}\{e_1, \ldots, e_{k-1}\} \setminus \text{span}\{\mu_{j+1}, \ldots, \mu_k\}$ must be orthogonal to every μ_ℓ for $j+1 \leq \ell \leq k$. Therefore, v must be orthogonal to c as well. This also enables us to add the terms for $j+1, \ldots, k$ in both the numerator and denominator, because they are all zero.

$$\begin{aligned}
\tilde{\phi} &\leq \max_{v \in \text{span}\{e_1,\ldots,e_{k-1}\} \setminus \text{span}\{\mu_{j+1},\ldots,\mu_k\}} \frac{v^T \Sigma v}{\sum_{i=1}^k w_i v^T (\mu_i \mu_i^T + \Sigma_i) v} \\
&\leq \max_{v \in \text{span}\{e_1,\ldots,e_{k-1}\}} \frac{v^T \Sigma v}{\sum_{i=1}^k w_i v^T (\mu_i \mu_i^T + \Sigma_i) v} \\
&= \phi.
\end{aligned}$$

□

The proofs of the main theorems are now apparent. Consider the case of $k = 2$ Gaussians first. As argued in Section 6.7, using $m_1 = \omega(kn^4 {w_{\min}}^{-3} \log(n/\delta w_{\min}))$ samples to estimate $\hat{u}$ and $\hat{M}$ is sufficient to guarantee that the estimates are accurate. For a well-chosen constant C, the condition

$$\phi \leq J(p) \leq C {w_{\min}}^3 \log^{-1}\left(\frac{1}{\delta w_{\min}} + \frac{1}{\eta}\right)$$

of Theorem 5 implies that

$$\sqrt{\phi} \leq \frac{w_{\min}\sqrt{\epsilon}}{640 \cdot 2},$$

where

$$\epsilon = \frac{w_{\min}}{2^9} \log^{-1}\left(\frac{2m_2}{\delta} + \frac{1}{\eta}\right).$$

The arguments of Section 6.8 then show that the direction h selected in step 3 satisfies

$$\|P_1^T h\| \geq 1 - \epsilon = 1 - \frac{w_{\min}}{2^9} \log^{-1}\left(\frac{m_2}{\delta} + \frac{1}{\eta}\right).$$

Already, for the overlap we have

$$\sqrt{\phi} \leq \frac{w_{\min}\sqrt{\epsilon}}{640 \cdot 2} \leq \sqrt{\frac{w_{\min}}{2^9 (k-1)^2}} \log^{-1/2} \frac{1}{\delta'}.$$

so we may apply Lemma 37 with $\delta' = (m_2/\delta + 1/\eta)^{-1}$. Thus, with probability $1 - \delta$ the classifier $H_{h,t}$ is correct with probability $1 - \delta' \geq 1 - \eta$.

We follow the same outline for $k > 2$, with the quantity $1/\delta' = m_2/\delta + 1/\eta$ being replaced with $1/\delta' = m/\delta + 1/\eta$, where m is the total number of samples used. This is necessary because the half-space $H_{h,t}$ must classify every sample point taken below it in the recursion subtree correctly. This adds the n and k factors so that the required overlap becomes

$$\phi \leq C{w_{\min}}^3 k^{-3} \log^{-1}\left(\frac{nk}{\delta w_{\min}} + \frac{1}{\eta}\right)$$

for an appropriate constant C. The correctness in the recursive steps is guaranteed by Lemma 38. Assuming that all previous steps are correct, the termination condition of step 4 is clearly correct when a single component is isolated.

CHAPTER VII

THE SUBGRAPH PARITY TENSOR

We now turn away from learning mixture models and consider applying principal components analysis in a different way to a different problem. Recall from Sec. 2.2 that the idea of the top principal component extends naturally from matrices to arbitrary order tensors. In this chapter, we define the subgraph parity tensor for graph and show that the top principal component reveals of this tensor reveals large planted cliques in random graphs.

7.1 Introduction

It is well-known that a random graph $G(n, 1/2)$ almost surely has a clique of size $(2 + o(1)) \log_2 n$ and a simple greedy algorithm finds a clique of size $(1 + o(1)) \log_2 n$. Finding a clique of size even $(1 + \epsilon) \log_2 n$ for some $\epsilon > 0$ in a random graph is a long-standing open problem posed by Karp in 1976 [31] in his classic paper on probabilistic analysis of algorithms.

In the early nineties, a very interesting variant of this question was formulated by Jerrum [28] and by Kucera [37]. Suppose that a clique of size p is planted in a random graph, i.e., a random graph is chosen and all the edges within a subset of p vertices are added to it. Then for what value of p can the planted clique be found efficiently? It is not hard to see that $p > c\sqrt{n \log n}$ suffices since then the vertices of the clique will have larger degrees than the rest of the graph, with high probability [37]. This was improved by Alon et al [2] to $p = \Omega(\sqrt{n})$ using a spectral approach. This was refined by McSherry [40] and considered by Feige and Krauthgamer in the more general semi-random model [16]. For $p \geq 10\sqrt{n}$, the following simple algorithm works: form a matrix with 1's for edges and -1's for nonedges; find the largest

eigenvector of this matrix and read off the top p entries in magnitude; return the set of vertices that have degree at least $3p/4$ within this subset.

The reason this works is the following: the top eigenvector of a symmetric matrix A can be written as

$$\max_{x:\|x\|=1} x^T A x = \max_{x:\|x\|=1} \sum_{ij} A_{ij} x_i x_j,$$

maximizing a quadratic polynomial over the unit sphere. The maximum value is the spectral norm or 2-norm of the matrix. For a random matrix with $1, -1$ entries, the spectral norm (largest eigenvalue) is $O(\sqrt{n})$. In fact, as shown by Füredi and Komlós [23, 49], a random matrix with i.i.d. entries of variance at most 1 has the same bound on the spectral norm. On the other hand, after planting a clique of size $\sqrt{n}$ times a sufficient constant factor, the indicator vector of the clique (normalized) achieves a higher norm. Thus the top eigenvector points in the direction of the clique (or very close to it).

Given the numerous applications of eigenvectors (principal components), a well-motivated and natural generalization of this optimization problem to an r-dimensional tensor is the following: given a symmetric tensor A with entries $A_{k_1 k_2 \ldots k_r}$, find

$$\|A\|_2 = \max_{x:\|x\|=1} A(x, \ldots, x),$$

where

$$A(x^{(1)}, \ldots, x^{(r)}) = \sum_{i_1 i_2 \ldots i_r} A_{i_1 i_2 \ldots i_r} x^{(1)}_{i_1} x^{(2)}_{i_2} \ldots x^{(r)}_{i_r}.$$

The maximum value is the spectral norm or 2-norm of the tensor. The complexity of this problem is open for any $r > 2$, assuming the entries with repeated indices are zeros.

A beautiful application of this problem was given recently by Frieze and Kannan [21]. They defined the following tensor associated with an undirected graph $G = (V, E)$:

$$A_{ijk} = E_{ij} E_{jk} E_{ki}$$

where E_{ij} is 1 is $ij \in E$ and -1 otherwise, i.e., A_{ijk} is the parity of the number of edges between i, j, k present in G. They proved that for the random graph $G_{n,1/2}$, the 2-norm of the random tensor A is $\tilde{O}(\sqrt{n})$, i.e.,

$$\sup_{x:\|x\|=1} \sum_{i,j,k} A_{ijk} x_i x_j x_k \leq C\sqrt{n} \log^c n$$

where c, C are absolute constants. This implied that if such a maximizing vector x could be found (or approximated), then we could find planted cliques of size as small as $n^{1/3}$ times polylogarithmic factors in polynomial time, improving substantially on the long-standing threshold of $\Omega(\sqrt{n})$.

Frieze and Kannan ask the natural question of whether this connection can be further strengthened by going to r-dimensional tensors for $r > 3$. The tensor itself has a nice generalization. For a given graph $G = (V, E)$ the r-parity tensor is defined as follows. Entries with repeated indices are set to zero; any other entry is the parity of the number of edges in the subgraph induced by the subset of vertices corresponding to the entry, i.e.,

$$A_{k_1,\ldots,k_r} = \prod_{1 \leq i < j \leq r} E_{k_i k_j}.$$

Frieze and Kannan's proof for $r = 3$ is combinatorial (as is the proof by Füredi and Komlós for $r = 2$), based on counting the number of subgraphs of a certain type. It is not clear how to extend this proof.

Here we prove a nearly optimal bound on the spectral norm of this random tensor for any r. This substantially strengthens the connection between the planted clique problem and the tensor norm problem. Our proof is based on a concentration of measure approach. In fact, we first reprove the result for $r = 3$ using this approach and then generalize it to tensors of arbitrary dimension. We show that the norm of the subgraph parity tensor of a random graph is at most $f(r)\tilde{O}(\sqrt{n})$ whp. More precisely, our main theorem is the following.

Theorem 8. *There is a constant C_1 such that with probability at least $1 - n^{-1}$ the norm of the r-dimensional subgraph parity tensor $A : [n]^r \to \{-1, 1\}$ for the random graph $G_{n,1/2}$ is bounded by*

$$\|A\|_2 \leq C_1^r r^{(5r-1)/2} \sqrt{n} \log^{(3r-1)/2} n.$$

The main challenge to the proof is the fact that the entries of the tensor A are not independent. Bounding the norm of the tensor where every entry is independently 1 or -1 with probability $1/2$ is substantially easier via a combination of an ϵ-net and a Hoeffding bound. In more detail, we approximate the unit ball with a finite (exponential) set of vectors. For each vector x in the discretization, the Hoeffding inequality gives an exponential tail bound on $A(x, \ldots, x)$. A union bound over all points in the discretization then completes the proof. For the parity tensor, however, the Hoeffding bound does not apply as the entries are not independent. Moreover, all the $\binom{n}{r}$ entries of the tensor are fixed by just the $\binom{n}{2}$ edges of the graph. In spite of this heavy interdependence, it turns out that $A(x, \ldots, x)$ does concentrate. Our proof is inductive and bounds the norms of vectors encountered in a certain decomposition of the tensor polynomial. It is not clear whether the bound of Theorem 8 is optimal, though a lower bound of $\|A\|_2 = \Omega(\max\{\sqrt{n}, (2 \log n)^{r/2}\})$ is trivial.

Using Theorem 8, we can show that if the norm problem can be solved for tensors of dimension r, one can find planted cliques of size as low as $Cn^{1/r}\text{poly}(r, \log n)$. While the norm of the parity tensor for a random graph remains bounded, the norm becomes at least $p^{r/2}$ when a clique of size p is planted (using the indicator vector of the clique). Therefore, p only needs to be a little larger than $n^{1/r}$ in order for the the clique to become the dominant term in the maximization of $A(x, \ldots, x)$. More precisely, we have the following theorem.

Theorem 9. *Let G be random graph $G_{n,1/2}$ with a planted clique of size p, and let A be the r-parity tensor for G. For $\alpha \leq 1$, let $T(n, r)$ be the time to compute a vector*

x such that $A(x,\dots,x) \geq \alpha^r \|A\|_2$ whp. Then, for p such that

$$n \geq p > C_0 \alpha^{-2} r^5 n^{1/r} \log^3 n,$$

the planted clique can be recovered with high probability in time $T(n,r) + \mathrm{poly}(n)$, where C_0 is a fixed constant.

On one hand, this highlights the benefits of finding an efficient (approximation) algorithm for the tensor problem. On the other, given the lack of progress on the clique problem, this is perhaps evidence of the hardness of the tensor maximization problem even for a natural class of random tensors. For example, if finding a clique of size $\tilde{O}(n^{1/2-\epsilon})$ is hard, then by setting $\alpha = n^{1/2r+\epsilon/2-1/4}$ we see that even a certain polynomial approximation to the norm of the parity tensor is hard to achieve.

Corollary 39. *Let G be random graph $G_{n,1/2}$ with a planted clique of size p, and let A be the r-parity tensor for G. Let $\epsilon > 0$ be a small constant and let $T(n,r)$ be the time to compute a vector x such that $A(x,\dots,x) \geq n^{1/2+r\epsilon/2-r/4}\|A\|_2$. Then, for*

$$p \geq C_0 r^5 n^{\frac{1}{2}-\epsilon} \log^3 n,$$

the planted clique can be recovered with high probability in time $T(n,r) + \mathrm{poly}(n)$, where C_0 is a fixed constant.

7.1.1 Overview of analysis

The majority of the chapter is concerned with proving Theorem 8. In Section 7.2.1, we first reduce the problem of bounding $A(\cdot)$ over the unit ball to bounding it over a discrete set of vectors that have the same value in every non-zero coordinate. In Section 7.2.2, we further reduce the problem to bounding the norm of an off-diagonal block of A, using a method of Frieze and Kannan. This enables us to assume that if $(k_1,\dots,k_r)$ is a valid index, then the random variables E_{k_i,k_j} used to compute $A_{i_1,\dots,i_r}$ are independent. In Section 7.2.3, we prove a large deviation inequality (Lemma 42)

that allows us to bound norms of vectors encountered in a certain decomposition of the tensor polynomial. This inequality gives us a considerably sharper bound than the Hoeffding or McDiarmid inequalities in our context. We then apply this lemma to bound $\|A\|_2$ for $r = 3$ as a warm-up and then give the proof for general r in Section 7.3.

In Section 7.4 we prove Theorem 9. The key idea is that any vector x that comes close to maximizing $A(\cdot)$ must have an indicator decomposition (see Definition 6) where the support of one of the vectors has a large intersection with the clique (Lemma 50). This intersection is large enough that the clique can be recovered.

7.2 Preliminaries

7.2.1 Discretization

The analysis of $A(x, \ldots, x)$ is greatly simplified when x is proportional to some indicator vector. Fortunately, analyzing these vectors is sufficient, as any vector can be approximated as a linear combination of relatively few indicator vectors.

For any vector x, we define $x^{(+)}$ to be vector such that $x_i^{(+)} = x_i$ if $x_i > 0$ and $x_i^{(+)} = 0$ otherwise. Similarly, let $x_i^{(-)} = x_i$ if $x_i < 0$ and $x_i^{(-)} = 0$ otherwise. For a set $S \subseteq [n]$, let χ^S be the indicator vector for S, where the ith entry is 1 if $i \in S$ and 0 otherwise.

Definition 6 (Indicator Decomposition). *For a unit vector x, define the sets $S_1, \ldots$ and $T_1, \ldots$ through the recurrences*

$$S_j = \left\{ i \in [n] : (x^{(+)} - \sum_{k=1}^{j-1} 2^{-k} \chi^{S_k})_i > 2^{-j} \right\}.$$

and

$$T_j = \left\{ i \in [n] : (x^{(-)} - \sum_{k=1}^{j-1} 2^{-k} \chi^{S_k})_i < -2^{-j} \right\}.$$

Let $y_0(x) = 0$. For $j \geq 1$, let $y^{(j)}(x) = 2^{-j}\chi^{S_j}$ and let $y^{(-j)}(x) = -2^{-j}\chi^{T_j}$. We call the set $\{y^{(j)}(x)\}_{-\infty}^{\infty}$ the indicator decomposition of x.

Clearly,

$$\|y^{(i)}(x)\| \leq \max\{\|x^{(+)}\|, \|x^{(-)}\|\} \leq 1.$$

and

$$\left\| x - \sum_{j=-N}^{N} y^{(j)}(x) \right\| \leq \sqrt{n} 2^{-N}. \tag{15}$$

We use this decomposition to prove the following theorem.

Lemma 40. *Let*

$$U = \{k|S|^{-1/2}\chi^S : S \subseteq [n], k \in \{-1, 1\}\}.$$

For any tensor A over $[n]^r$ where $\|A\|_\infty \leq 1$

$$\max_{x^{(1)},\ldots,x^{(r)} \in B(0,1)} A(x^{(1)}, \ldots x^{(r)}) \leq (2\lceil r \log n \rceil)^r \max_{x^{(1)},\ldots,x^{(r)} \in U} A(x^{(1)}, \ldots, x^{(r)})$$

Proof. Consider a fixed set of vectors $x^{(1)}, \ldots, x^{(r)}$ and let $N = \lceil r \log_2 n \rceil$. For each i, let

$$\hat{x^{(i)}} = \sum_{j=-N}^{N} y^{(j)}(x^{(i)}).$$

We first show that replacing $x^{(i)}$ with $\hat{x^{(i)}}$ gives a good approximation to the value $A(x^{(1)}, \ldots, x^{(r)})$. Letting ϵ be the maximum difference between an $x^{(i)}$ and its approximation, we have from (15) that

$$\max_{i \in [r]} \|x^{(i)} - \hat{x^{(i)}}\| = \epsilon \leq \frac{n^{r/2}}{2r}$$

Because of the multilinear form of $A(\cdot)$ we have

$$|A(x^{(1)}, \ldots, x^{(r)}) - A(\hat{x^{(1)}}, \ldots, \hat{x^{(r)}})| \leq \sum_{i=1}^{r} \epsilon^i r^i \|A\| \leq \frac{\epsilon r}{1 - \epsilon r} \|A\| \leq 1.$$

Next, we bound $A(\hat{x^{(1)}}, \ldots, \hat{x^{(r)}})$. For convenience, let $Y^{(i)} = \cup_{j=-N}^{N} y^{(j)}(x^{(i)})$. Then using the multlinear form of $A(\cdot)$ and bounding the sum by its maximum term, we have

$$\begin{aligned} A(\hat{x}^{(1)}, \ldots, \hat{x}^{(r)}) &\leq (2N)^r \max_{v^{(1)} \in Y^{(1)},\ldots,v^{(r)} \in Y^{(r)}} A(v^{(1)}, \ldots, v^{(r)}) \\ &\leq (2N)^r \max_{v^{(1)},\ldots,v^{(r)} \in U} A(v^{(1)}, \ldots, v^{(r)}). \end{aligned}$$

□

7.2.2 Sufficiency of off-diagonal blocks

Analysis of $A(x^{(1)}, \ldots, x^{(r)})$ is complicated by the fact that all terms with repeated indices are zero. Off-diagonal blocks of A are easier to analyze because no such terms exist. Thankfully, as Frieze and Kannan [21] have shown, analyzing these off-diagonal blocks suffices. Here we generalize their proof to $r > 3$.

For a collection $\{V_1, V_2, \ldots, V_r\}$ of subsets of $[n]$, we define

$$A|_{V_1 \times \ldots \times V_r}(x^{(1)}, \ldots, x^{(r)}) = \sum_{k_1 \in V_1, \ldots, k_r \in V_r} A_{k_1 \ldots k_r} x^{(1)}_{i_1} x^{(2)}_{i_2} \ldots x^{(r)}_{i_r}$$

Lemma 41. *Let P be the class of partitions of $[n]$ into r equally sized sets $V_1, \ldots, V_r$ (assume wlog that r divides n). Let $V = V_1 \times \ldots \times V_r$. Let A be a random tensor over $[n]^r$ where each entry is in $[-1, 1]$ and let $R \subseteq B(0, 1)$. If for every fixed $(V_1, \ldots V_r) \in P$, it holds that*

$$\mathsf{P}[\max_{x^{(1)}, \ldots, x^{(r)} \in R} A|_V(x^{(1)}, \ldots, x^{(r)}) \geq f(n)] \leq \delta,$$

then

$$\mathsf{P}[\max_{x^{(1)}, \ldots, x^{(r)} \in R} A(x^{(1)}, \ldots, x^{(r)}) \geq 2r^r f(n)] \leq \frac{\delta n^{r/2}}{f(n)},$$

Proof of Lemma 41. Each r-tuple appears in an equal number of partitions and this number is slightly more than a r^{-r} fraction of the total. Therefore,

$$\begin{aligned} \left|A(x^{(1)}, \ldots A(x^{(r)})\right| &\leq \frac{r^r}{|P|} \left| \sum_{\{V_1, \ldots, V_r\} \in P} A|_V(x^{(1)}, \ldots A(x^{(r)}) \right| \\ &\leq \frac{r^r}{|P|} \sum_{\{V_1, \ldots, V_r\} \in P} \left|A|_V(x^{(1)}, \ldots A(x^{(r)})\right| \end{aligned}$$

We say that a partition $\{V_1, \ldots, V_r\}$ is good if

$$\max_{x^{(1)}, \ldots, x^{(r)} \in R} A|_V(x^{(1)}, \ldots, x^{(r)}) < f(n).$$

Let the good partitions be denoted by G and let $\bar{G} = P \setminus G$. Although the f upper bound does not hold for partitions in $\bar{G}$, the trivial upper bound of $n^{r/2}$ does (recall

that every entry in the tensor is in the range $[-1, 1]$ and $R \subseteq B(0, 1)$). Therefore

$$\left|A(x^{(1)}, \ldots A(x^{(r)})\right| \leq r^r(\frac{|G|}{|P|}f + \frac{|\bar{G}|}{|P|}n^{r/2}).$$

Since $E[|G|/|P|] = \delta$ by hypothesis, Markov's inequality gives

$$\mathsf{P}[\frac{|G|}{|P|}n^{r/2} > f] \leq \frac{\delta n^{r/2}}{f}$$

and thus proves the result. □

7.2.3 A concentration bound

The following concentration bound is a key tool in our proof of Theorem 8. We apply it for $t = \tilde{O}(N)$.

Lemma 42. *Let $\{u^{(i)}\}_{i=1}^N$ and $\{v^{(i)}\}_{i=1}^N$ be collections of vectors of dimension N' where each entry of $u^{(i)}$ is 1 or -1 with probability $1/2$ and $\|v^{(i)}\|_2 \leq 1$. Then for any $t \geq 1$,*

$$\mathsf{P}[\sum_{i=1}^{N}(u^{(i)} \cdot v^{(i)})^2 \geq t] \leq e^{-t/18}(4\sqrt{e\pi})^N.$$

Before giving the proof, we note that this lemma is stronger than what a naive application of standard theorems would yield for $t = \tilde{O}(N)$. For instance, one might treat each $(u^{(i)} \cdot v^{(i)})^2$ as an independent random variable and apply a Hoeffding bound. The quantity $(u^{(i)} \cdot v^{(i)})^2$ can vary by as much as N', however, so the bound would be roughly $\exp(-ct^2/NN'^2)$ for some constant c. Similarly, treating each $u_j^{(i)}$ as an independent random variable and applying McDiarmid's inequality, we find that every $u_j^{(i)}$ can affect the sum by as much as 1 (simultaneously). For instance suppose that every $v_j^{(i)} = 1/\sqrt{N'}$ and every $u_j^{(i)} = 1$. Then flipping $u_j^{(i)}$ would have an effect of $|N' - ((N' - 2)/\sqrt{N'})^2| \approx 4$, so the bound would be roughly $\exp(-ct^2/NN')$ for some constant c.

Proof of Lemma 42. Observe that $\sqrt{\sum_{i=1}^N (u^{(i)} \cdot v^{(i)})^2}$ is the length of the vector whose ith coordinate is $u^{(i)} \cdot v^{(i)}$. Therefore, this is also equivalent to the maximum projection

of this vector onto a unit vector:

$$\sqrt{\sum_{i=1}^{N}(u^{(i)} \cdot v^{(i)})^2} = \max_{y \in B(0,1)} \sum_{i=1}^{N}\sum_{j=1}^{N'} y_i u_j^{(i)} v_j^{(i)}.$$

We will use an ϵ-net to approximate the unit ball and give an upper bound for this quantity. Let $\mathcal{L}$ be the lattice $\left(\frac{1}{2\sqrt{N}}\mathbb{Z}\right)^N$.

Claim 43. *For any vector x,*

$$\|x\|_2 \leq 2 \max_{y \in \mathcal{L} \cap B(0,3/2)} y \cdot x.$$

Thus,

$$\sqrt{\sum_{i=1}^{N}(u^{(i)} \cdot v^{(i)})^2} \leq 2 \max_{y \in \mathcal{L} \cap B(0,3/2)} \sum_{i=1}^{N} y_i \sum_{j=1}^{N'} u_j^{(i)} v_j^{(i)}.$$

Consider a *fixed* $y \in \mathcal{L} \cap B(0, 3/2)$. Each $u_i^{(j)}$ is 1 or -1 with equal probability, so the expectation for each term is zero. The difference between the upper and lower bounds for a term is

$$2|2y_j u_j^{(i)} v(i)_j| = 4|y_j v(i)_j|$$

Therefore,

$$16\sum_{i=1}^{N}\sum_{j=1}^{N'}(y_i u_j^{(i)} v(i)_j)^2 \leq 16\sum_{i=1}^{N} y^2 \sum_{j=1}^{N'}(v(i)_j)^2 = 36.$$

Applying the Hoeffding bound gives that

$$\mathsf{P}[\sum_{i=1}^{N}(u^{(i)} \cdot v^{(i)})^2 \geq t] \leq \mathsf{P}[2\sum_{i=1}^{N} y_i \sum_{j=1}^{N'} u_j^{(i)} v(i)_j \geq \sqrt{t}] \leq e^{-t/18}.$$

The result follows by taking a union bound over $\mathcal{L} \cap B(0, 3/2)$, whose cardinality is bounded according to Claim 44. □

Claim 44. *The number of lattice points in $\mathcal{L} \cap B(0, 3/2)$ is at most $(4\sqrt{e\pi})^N$*

Proof of Claim 44. Consider the set of hypercubes where each cube is centered on a distinct point in $\mathcal{L} \cap B(0, 3/2)$ and each has side length of $(2\sqrt{n})^{-1}$. These cubes are

disjoint and their union contains the ball $B(0, 3/2)$. Their union is also contained in the ball $B(0, 2)$. Thus,

$$\begin{aligned} |\mathcal{L} \cap B(0, 3/2)| &\leq \frac{\mathrm{Vol}(B(0,2))}{(2\sqrt{N})^{-N}} \\ &\leq \frac{\pi^{N/2} 2^N}{\Gamma(N/2+1)} 2^N N^{N/2} \\ &\leq (4\sqrt{e\pi})^N. \end{aligned}$$

□

Proof of Claim 43. Without loss of generality, we assume that x is a unit vector. Let y be the closest point to x in the lattice. In each coordinate i, we have $|x_i - y_i| \leq (4\sqrt{n})^{-1}$, so overall $\|x - y\| \leq 1/4$.

Letting θ be the angle between x and y, we have

$$\frac{x \cdot y}{\|x\|\|y\|} = \cos\theta = \sqrt{1 - \sin^2\theta} \geq \left(1 - \frac{\|x-y\|^2}{\max\{\|x^2\|, \|y\|^2\}}\right)^{1/2} \geq \sqrt{\frac{15}{16}}.$$

Therefore,

$$x \cdot y \geq \|y\| \sqrt{\frac{15}{16}} \geq \frac{3}{4}\sqrt{\frac{15}{16}} \geq \frac{1}{2}.$$

□

7.3 A bound on the norm of the parity tensor

In this section, we prove Theorem 8. First, however, we consider the somewhat more transparent case of $r = 3$ using the same proof technique.

7.3.1 Warm-up: third order tensors

For $r = 3$ the tensor A is defined as follows:

$$A_{k_1 k_2 k_3} = E_{k_1 k_2} E_{k_2 k_3} E_{k_1 k_3}.$$

Theorem 10. *There is a constant C_1 such that with probability $1 - n^{-1}$*

$$\|A\| \leq C_1 \sqrt{n} \log^4 n.$$

Proof. Let V_1, V_2, V_3 be a partition of the n vertices and let $V = V_1 \times V_2 \times V_3$. The bulk of the proof consists of the following lemma.

Lemma 45. *There is some constant C_3 such that*

$$\max_{x^{(1)},x^{(2)},x^{(3)} \in U} A|_V(x^{(1)}, x^{(2)}, x^{(3)}) \leq C_3 \sqrt{n} \log n$$

with probability $1 - n^{-7}$.

If this bound holds, then Lemma 40 then implies that there is some C_2 such that

$$\max_{x^{(1)},x^{(2)},x^{(3)} \in B(0,1)} A|_V(x^{(1)}, x^{(2)}, x^{(3)}) \leq C_2 \sqrt{n} \log^4 n.$$

And finally, Lemma 41 implies that for some constant C_1

$$\max_{x^{(1)},x^{(2)},x^{(3)} \in B(0,1)} A(x^{(1)}, x^{(2)}, x^{(3)}) \leq C_1 \sqrt{n} \log^4 n$$

with probability $1 - n^{-1}$. □

Proof of Lemma 45. Define

$$U_k = \{x \in U : |\mathrm{sup}(x)| = k\} \tag{16}$$

and consider a fixed $n \geq n_1 \geq n_2 \geq n_3 \geq 1$. We will show that for some constant C_3,

$$\max_{(x^{(1)},x^{(2)},x^{(3)}) \in U_{n_1} \times U_{n_2} \times U_{n_3}} A|_V(x^{(1)}, x^{(2)}, x^{(3)}) \leq C_3 \sqrt{n} \log n$$

with probability n^{-10}. Taking a union bound over the n^3 choices of n_1, n_2, n_3 then proves the lemma.

We bound the cubic form as

$$\begin{aligned}
&\max_{(x^{(1)},x^{(2)},x^{(3)}) \in U_{n_1} \times U_{n_2} \times U_{n_3}} A|_V(x^{(1)}, x^{(2)}, x^{(3)}) \\
&= \max_{(x^{(1)},x^{(2)},x^{(3)}) \in U_{n_1} \times U_{n_2} \times U_{n_3}} \sum_{k_1 \in V_1, k_2 \in V_2, k_3 \in V_3} A_{k_1 k_2 k_3} x^{(1)}_{k_1} x^{(2)}_{k_2} x^{(3)}_{k_3} \\
&\leq \max_{(x^{(2)},x^{(3)}) \in U_{n_2} \times U_{n_3}} \sqrt{\sum_{k_1 \in V_1} \left(\sum_{k_2 \in V_2, k_3 \in V_3} A_{k_1 k_2 k_3} x^{(2)}_{k_2} x^{(3)}_{k_3} \right)^2} \\
&= \max_{(x^{(2)},x^{(3)}) \in U_{n_2} \times U_{n_3}} \sqrt{\sum_{k_1 \in V_1} \left(\sum_{k_2 \in V_2} E_{k_1 k_2} x^{(2)}_{k_2} \sum_{k_3 \in V_3} E_{k_2 k_3} x^{(3)}_{k_3} E_{k_1 k_3} \right)^2}.
\end{aligned}$$

Note that each of the inner sums (over k_2 and k_3) are the dot product of a random $-1, 1$ vector (the $E_{k_1k_2}$ and $E_{k_2k_3}$ terms) and another vector. Our strategy will be to bound the norm of this other vector and apply Lemma 42.

To this end, we define the $-1, 1$ vectors $u_{k_3}^{(k_2)} = E_{k_2k_3}$ and $u_{k_2}^{(k_1)} = E_{k_1k_2}$, and the general vectors

$$v^{(k_1k_2)}(x^{(3)})_{k_3} = x_{k_3}^{(3)} E_{k_1k_3}$$

and

$$v^{(k_1)}(x^{(2)}, x^{(3)})_{k_2} = x_{k_2}^{(2)}(u^{(k_2)} \cdot v^{(k_1k_2)}(x^{(3)})).$$

Thus, for each k_1,

$$\begin{aligned} &\sum_{k_2 \in V_2} E_{k_1k_2} x_{k_2}^{(2)} \sum_{k_3 \in V_3} E_{k_2k_3} x_{k_3}^{(3)} E_{k_1k_3} \\ &= \sum_{k_2 \in V_2} E_{k_1k_2} x_{k_2}^{(2)}(u^{(k_2)} \cdot v^{(k_1k_2)}(x^{(3)})) \\ &= u^{(k_1)} \cdot v^{(k_1)}(x^{(2)}, x^{(3)}). \end{aligned} \tag{17}$$

Clearly, the u's play the role of the random vectors and we will bound the norms of the v's in the application of Lemma 42.

To apply Lemma 42 with k_1 being the index i, $u_{k_2}^{k_1} = E_{k_1k_2}$ above, we need a bound for every $k_1 \in V_1$ on the norm of $v^{(k_1)}(x^{(2)}, x^{(3)})$. We argue

$$\begin{aligned} &\sum_{k_2} \left(x_{k_2}^{(2)} \sum_{k_3 \in V_3} E_{k_2k_3} x_{k_3}^{(3)} E_{k_1k_3} \right)^2 \\ &\leq \max_{k_1 \in V_1} \max_{x^{(2)} \in U_{n_2}} \max_{x^{(3)} \in U_{n_3}} \frac{1}{n_2} \sum_{k_2 \in \sup(x^{(x_2)})} \left(\sum_{k_3} E_{k_2k_3} x_{k_3}^{(3)} E_{k_1k_3} \right)^2 \\ &= F_1^2 \end{aligned}$$

Here we used the fact that $\|x^{(2)}\|_\infty \leq n_2^{-1/2}$. Note that F_1 is a function of the random variables $\{E_{ij}\}$ only.

To bound F_1, we observe that we can apply Lemma 42 to the expression being

maximized above, i.e.,

$$\sum_{k_2}\left(\sum_{k_3} E_{k_2k_3}\left(x^{(3)}_{k_3}E_{k_1k_3}\right)\right)^2$$

over the index k_2, with $u^{k_2}_{k_3} = E_{k_2k_3}$. Now we need a bound, for every k_2 and k_1 on the norm of the vector $v^{(k_1k_2)}(x^{(3)})$. We argue

$$\begin{aligned}\sum_{k_3}\left(x^{(3)}_{k_3}E_{k_1k_3}\right)^2 &\leq ||x^{(3)}||^2_\infty \sum_{k_3} E^2_{k_1k_3}\\ &\leq 1.\end{aligned}$$

Applying Lemma 42 for a fixed $k_1, x^{(2)}$ and $x^{(3)}$ implies

$$\frac{1}{n_2}\sum_{k_2\in\sup(x^{(2)})}\left(\sum_{k_3} E_{k_2k_3}x^{(3)}_{k_3}E_{k_1k_3}\right)^2 > C_3\log n$$

with probability at most

$$\exp(-\frac{C_3n_2\log n}{18})(4\sqrt{e\pi})^{n_2}.$$

Taking a union bound over the $|V_1| \leq n$ choices of k_1, and the at most $n^{n_2}n^{n_3}$ choices for $x^{(2)}$ and $x^{(3)}$, we show that

$$\mathsf{P}[F_1^2 > C_3\log n] \leq \exp(-\frac{C_3n_2\log n}{18})(4\sqrt{e\pi})^{n_2}nn^{n_2}n^{n_3}.$$

This probability is at most $n^{-10}/2$ for a large enough constant C_3.

Thus, for a fixed $x^{(2)}$ and $x^{(3)}$, we can apply Lemma 42 to (17) with $F_1^2 = C_3\log n$ to get:

$$\sum_{k_1\in V_1}\left(\sum_{k_2\in V_2} E_{k_1k_2}\left(x^{(2)}_{k_2}\sum_{k_3\in V_3} E_{k_2k_3}x^{(3)}_{k_3}E_{k_1k_3}\right)\right)^2 > F_1^2C_3n\log n$$

with probability at most $\exp(-C_3n\log n/18)(4\sqrt{e\pi})^n$. Taking a union bound over the at most $n^{n_2}n^{n_3}$ choices for $x^{(2)}$ and $x^{(3)}$, the bound holds with probability

$$\exp(-C_3n\log n/18)(4\sqrt{e\pi})^n n^{n_2}n^{n_3} \leq n^{-10}/2$$

for large enough constant C_3.

Thus, we can bound the squared norm:

$$\begin{aligned}
&\max_{(x^{(1)},x^{(2)},x^{(3)})\in U_{n_1}\times U_{n_2}\times U_{n_3}} A|_V(x^{(1)},x^{(2)},x^{(3)})^2 \\
\leq\ & \sum_{k_1\in V_1}\left(\sum_{k_2\in V_2} E_{k_1k_2}\left(x^{(2)}_{k_2}\sum_{k_3\in V_3} E_{k_2k_3}x^{(3)}_{k_3}E_{k_1k_3}\right)\right)^2 \\
\leq\ & C_3^2 n_1 \log^2 n
\end{aligned}$$

with probability $1-n^{-10}$. □

7.3.2 Higher order tensors

Let the random tensor A be defined as follows.

$$A_{k_1,\ldots,k_r} = \prod_{1\leq i<j\leq r} E_{k_ik_j}$$

where E is an $n\times n$ matrix where each off-diagonal entry is -1 or 1 with probability $1/2$ and every diagonal entry is 1.

For most of this section, we will consider only a single off-diagonal cube of A. That is, we index over $V_1\times\ldots\times V_r$ where V_i are an equal partition of $[n]$. We denote this block by $A|_V$. When k_i is used as an index, it is implied that $k_i\in V_i$.

The bulk of the proof consists of the following lemma.

Lemma 46. *There is some constant C_3 such that*

$$\max_{x^{(1)},\ldots x^{(r)}\in U} A|_V(x^{(1)},\ldots,x^{(r)})^2 \leq n(C_3 r\log n)^{r-1}$$

with probability $1-n^{-9r}$.

The key idea is that Lemma 42 can be applied repeatedly to collections of u's and v's in a way analogous to (17). Each sum over $k_r,\ldots,k_2$ contributes a $C_3 r\log n$ factor and the final sum over k_1 contributes the factor of n.

If the bound holds, then Lemma 40 implies that there is some C_2 such that

$$\max_{x^{(1)},x^{(2)},x^{(3)}\in B(0,1)} A|_V(x^{(1)},x^{(2)},x^{(3)})^2 \leq C_2^r r^{2r+r-1} n\log^{2r+(r-1)} n.$$

And finally, Lemma 41 implies that for some constant C_1

$$\begin{aligned}\max_{x^{(1)},x^{(2)},x^{(3)}\in B(0,1)} A(x^{(1)},x^{(2)},x^{(3)}) &\leq C_1^r r^{2r+2r+(r-1)} n \log^{2r+r-1} n \\ &= C_1^r r^{5r-1} n \log^{3r-1} n.\end{aligned}$$

with probability $1 - n^{-1}$.

Proof of Lemma 46. We define the set U_k as in (16). It suffices to show that the bound

$$\max_{(x^{(1)},\ldots x^{(r)})\in U_{n_1}\times\ldots\times U_{n_r}} A|_V(x^{(1)},\ldots,x^{(r)})^2 \leq n(C_3 r\log n)^{r-1}$$

holds with probability $1-n^{-10r}$ for some constant C_3, since we may then take a union bound over the n^r choices of $n \geq n_1 \geq \ldots \geq n_r \geq 1$.

For convenience of notation, we define a family of tensors as follows

$$B^{(k_1,\ldots,k_\ell)}_{k_{\ell+1},\ldots,k_r} = \prod_{i,j:i,\ell<j} E_{k_i k_j} \tag{18}$$

where the superscript indexes the family of tensors and the subscript indexes the entries. Note that for every $k_1,\ldots,k_r \in V_1\times\ldots\times V_r$, we have $B^{(k_1,\ldots,k_r)} = 1$, since the product is empty.

Note that the tensor $B^{(k_1,\ldots,k_\ell)}$ depends only a subset of E. In particular, any such tensor of order $r-\ell$ will depend only on the blocks of E

$$F_\ell = \{E|_{V_i\times V_j} : i,\ell<j\}.$$

Clearly, $F_r = \emptyset$, F_1 contains all blocks, and $F_\ell \setminus F_{\ell+1} = \{E|_{V_i\times V_{\ell+1}} : i \leq \ell\}$.

We bound the rth degree form as

$$\begin{aligned}&\max_{x^{(1)},\ldots,x^{(r)}\in U_{n_1}\times\ldots\times U_{n_r}} A|_V(x^{(1)},\ldots,x^{(r)}) \\ &= \max_{x^{(1)},\ldots,x^{(r)}\in U_{n_1}\times\ldots\times U_{n_r}} \sum_{k_1\in V_1} x^{(1)}_{k_1} B^{(k_1)}(x^{(2)},\ldots x^{(r)}) \\ &\leq \max_{x^{(2)},\ldots x^{(r)}\in U_{n_2}\times\ldots\times U_{n_r}} \sqrt{\sum_{k_1\in V_1} B^{(k_1)}(x^{(2)},\ldots x^{(r)})^2}. \end{aligned}\tag{19}$$

Observe that for a general ℓ,

$$B^{(k_1,\ldots,k_\ell)}(x^{(\ell+1)},\ldots,x^{(r)}) = \sum_{k_{\ell+1}\in V_{\ell+1}} E_{k_\ell k_{\ell+1}} v^{(k_1,\ldots,k_\ell)}(x^{(\ell+1)},\ldots,x^{(r)})_{k_{\ell+1}}, \tag{20}$$

where

$$v^{(k_1,\ldots,k_\ell)}(x^{(\ell+1)},\ldots,x^{(r)})_{k_{\ell+1}} = x^{(\ell+1)}_{k_{\ell+1}} B^{(k_1,\ldots,k_{\ell+1})}(x^{(\ell+2)},\ldots,x^{(r)}) \prod_{i<\ell} E_{k_i k_{\ell+1}}. \tag{21}$$

It will be convenient to think of $B^{(k_1,\ldots,k_\ell)}(x^{(\ell+1)},\ldots,x^{(r)})$ as the dot product of a random vector $u^{(k_\ell)}$, where $u^{(k_\ell)}_{k_{\ell+1}} = E_{k_\ell k_{\ell+1}}$ and $v^{(k_1,\ldots,k_\ell)}(x^{(\ell+1)},\ldots,x^{(r)})_{k_{\ell+1}}$, so that

$$B^{(k_1,\ldots,k_\ell)}(x^{(\ell+1)},\ldots,x^{(r)}) = u^{(k_\ell)} \cdot v^{(k_1,\ldots,k_\ell)}(x^{(\ell+1)},\ldots,x^{(r)}). \tag{22}$$

The sum over $k_1 \in V_1$ from (19) can therefore be expanded as

$$\sum_{k_1\in V_1} B^{(k_1)}(x^{(2)},\ldots x^{(r)})^2 = \sum_{k_1\in V_1} \left(u^{(k_1)} \cdot v^{(k_1)}(x^{(2)},\ldots,x^{(r)})\right)^2.$$

Our goal is to bound $\|v^{(k_1)}(x^{(2)},\ldots,x^{(r)})\|$ and apply Lemma 42. Notice that for general ℓ

$$\begin{aligned}
&\left\|v^{(k_1,\ldots,k_\ell)}(x^{(\ell+1)},\ldots,x^{(r)})\right\|_2^2 \\
&\qquad = \frac{1}{n^{\ell+1}} \sum_{k_{\ell+1}\in \sup(x^{(\ell+1)})} B^{(k_1,\ldots,k_{\ell+1})}(x^{(\ell+2)},\ldots,x^{(r)})^2 \\
&\qquad\qquad \le \max_{k_1,\ldots,k_\ell} \max_{x^{(\ell+1)}\in U_{n_{\ell+1}} \ldots x^{(r)}\in U_{n_r}} \\
&\qquad\qquad \frac{1}{n_{\ell+1}} \sum_{k_{\ell+1}\in \sup(x^{(\ell+1)})}) B^{(k_1,\ldots,k_{\ell+1})}(x^{(\ell+2)},\ldots,x^{(r)})^2 = f_\ell^2.
\end{aligned} \tag{23}$$

Note that the quantity f_ℓ (define above) depends only on the blocks $F_{\ell+1}$.

The following claims will establish a probabilistic bound on f_1.

Claim 47. *The quantity*

$$f_{r-1} = 1.$$

Proof. Trivially, every $B^{(k_1,\ldots,k_r)}()^2 = 1$. Therefore, for every subset $S_r \subseteq V_r$ such that $|S_r| = n_r$

$$\frac{1}{n_r}\sum_{k_r \in S_r} B^{(k_1,\ldots,k_r)}()^2 = 1.$$

□

Claim 48. *There is a constant C_3 such that for any $\ell \in 1 \ldots r-2$*

$$\mathsf{P}[f_\ell^2 > C_3 r f_{\ell+1}^2 \log n] \leq n^{-12r}.$$

We postpone the proof of Claim 48 and argue that by induction we have that

$$f_1^2 \leq (C_3 r \log n)^{r-2}$$

with probability $1 - n^{-12r} r \geq 1 - n^{-11r}$.

Assuming that this bound holds,

$$v^{(k_1)}(x^{(2)}, \ldots, x^{(r)}) \leq (C_3 r \log n)^{r-2}$$

for all $k_1 \in V_1$ and $x^{(2)} \ldots, x^{(r)}$. By Lemma 42 then

$$\begin{aligned}\sum_{k_1 \in V_1} B^{(k_1)}(x^{(2)}, \ldots x^{(r)})^2 &= \sum_{k_1 \in V_1} \left(u^{(k_\ell)} \cdot v^{(k_1)}(x^{(3)}, \ldots, x^{(r)})\right)^2 \\ &> n(C_3 r \log n)^{r-1}\end{aligned}$$

with probability at most

$$\exp\left(-\frac{C_3 r n \log n}{18}\right)(4\sqrt{e\pi})^n$$

which is at most n^{-11r} for a suitably large C_3.

Altogether the bound of the lemma holds with probability $1 - 2n^{-11r} \geq 1 - n^{-10r}$. □

Proof of Claim 48. Consider a fixed choice of the following: 1) $k_1, \ldots k_\ell$ and 2) $x^{(\ell+1)} \in U_{n_{\ell+1}}, \ldots x^{(r)} \in U_{n_r}$. From (23), we have from definition that for every $k_{\ell+1} \in V_{\ell+1}$

$$\|v^{(k_1 \ldots k_{\ell+1})}(x^{(\ell+2)}, \ldots, x^{(r)})\|_2^2 \leq f_{\ell+1}^2.$$

Therefore, by Lemma 42

$$\begin{aligned}
&\sum_{k_{\ell+1}\in\sup(x^{(\ell+1)})} B^{(k_1,\ldots,k_{\ell+1})}(x^{(\ell+2)},\ldots,x^{(r)})^2 \\
&= \sum_{k_{\ell+1}\in\sup(x^{(\ell+1)})} \left(u^{(\ell+1)}\cdot v^{(k_1\ldots k_{\ell+1})}(x^{(\ell+2)},\ldots,x^{(r)})\right)^2 \\
&> C_3 r f_{\ell+1}^2 n_{\ell+1}\log n
\end{aligned}$$

with probability at most

$$\exp\left(-\frac{C_3 r n_{\ell+1}\log n}{18}\right)(4\sqrt{e\pi})^{n_{\ell+1}}.$$

Taking a union bound over the choice of $k_1,\ldots k_\ell$ (at most n^r), and the choice of $x^{(\ell+1)}\in U_{n_{\ell+1}},\ldots x^{(r)}\in U_{n_r}$ (at most $n^{(r-1)n_{\ell+1}}$), the probability that

$$f_\ell^2 > C_3 r f_{\ell+1}^2\log n$$

becomes at most

$$\exp\left(-\frac{C_3 r n_{\ell+1}\log n}{18}\right)(4\sqrt{e\pi})^{n_{\ell+1}} n^{r n_{\ell+1}}.$$

For large enough C_3 this is at most n^{-12r}. □

7.4 Finding planted cliques

We now turn to Theorem 9 and to the problem of finding a planted clique in a random graph. A random graph with a planted clique is constructed by taking a random graph and then adding every edge between vertices in some subset P to form the planted clique. We denote this graph as $G_{n,1/2}\cup K_p$. Letting A be the rth order subgraph parity tensor, we show that a vector $x\in B(0,1)$ that approximates the maximum of $A(\cdot)$ over the unit ball can be used to reveal the clique, using a modification of the algorithm proposed by Frieze and Kannan [21].

This implies an interesting connection between the tensor problem and the planted clique problem. For symmetric second order tensors (i.e. matrices), maximizing $A(\cdot)$ is equivalent to finding the top eigenvector and can be done in polynomial time. For

higher order tensors, maximizing $A(\cdot)$ is hard in general (see Appendix A); however, the complexity of maximizing this function is open if elements with repeated indices are zero. For random tensors, the hardness is also open. Given the reduction presented in this section, a hardness result for the planted clique problem would imply a similar hardness result for the tensor problem.

Given an x that approximates the maximum of $A(\cdot)$ over the unit ball, the algorithm for finding the planted clique is given in Alg. 7.4. The key ideas of using the top eigenvector of subgraph and of randomly choosing a set of vertices to "seed" the clique (steps 2a-2d) come from Frieze-Kannan [21]. The major difference in the algorithms is the use of the indicator decomposition. Frieze and Kannan sort the indices so that $x_1 \geq \ldots x_n$ and select one set S of the form $S = [j]$ where $\|A|_{S\times S}\|$ exceeds some threshold. They run steps (2a-2d) only on this set. By contrast Alg. 7.4 runs these steps on every $S = \sup(y^{(j)}(x))$ where $j = -\lceil r \log n \rceil, \ldots \lceil r \log n \rceil$.

The algorithm succeeds with high probability when a subset S is found such that $|S \cap P| \geq C\sqrt{|S| \log n}$, where C is an appropriate constant.

Lemma 49 (Frieze-Kannan). *There is a constant C_5 such that if $S \subseteq [n]$ satisfies $|S \cap P| \geq C_5\sqrt{|S| \log n}$, then with high probability steps 2a)-2d) of Alg. 7.4 find a set P' equal to P.*

To find such an subset S from a vector x, Frieze and Kannan require that $\sum_{i\in P} x_i \geq C \log n$. Using the indicator decomposition, as in the Alg 7.4, however, reduces this to $\sum_{i\in P} x_i \geq C\sqrt{\log n}$. Even more importantly, using the indicator decomposition means that only one element of the decomposition needs to point in the direction of the clique. The vector x could point in a very different direction and the algorithm would still succeed. We exploit this fact in our proof of Theorem 9. The relevant claim is the following.

Algorithm 5 An Algorithm for Recovering the Clique

Input:
1) Graph G.
2) Integer $p = |P|$.
3) Unit vector x.

Output: A clique of size p or FAILURE.

1. Calculate $y^{-\lceil r \log n \rceil}(x), \ldots, y^{\lceil r \log n \rceil}(x)$ as defined in the indicator decomposition.
2. For each such $y^{(j)}(x)$, let $S = \sup(y^{(j)}(x))$ and try the following:
   (a) Find v, the top eigenvector of the $1, -1$ adjacency matrix $A|_{S \times S}$.
   (b) Order the vertices (coordinates) such that $v_1 \geq \ldots \geq v_{|S|}$. (Assuming dot-prod is $\sqrt{1/2}$ below)
   (c) For $\ell = 1$ to $|S|$, repeat up to $n^{30} \log n$ times:
      i. Select $10 \log n$ vertices Q_1 at random from $[\ell]$.
      ii. Find Q_2, the set of common neighbors of Q_1 in G.
      iii. If the set of vertices with degree at least $7p/8$, say P' has cardinality p and forms a clique in G, then return P'.
   (d) Return FAILURE.

Lemma 50. *Let B' be a set of vectors $x \in B(0,1)$ such that*

$$|\sup(y^{(j)}(x)) \cap P| < C_5 \sqrt{|\sup(y^{(j)}(x))| \log n}$$

for every $j \in \{-\lceil r \log n \rceil, \ldots, \lceil r \log n \rceil\}$. Then, there is a constant C_1' such that with high probability

$$\sup_{x \in B'} A(x, \ldots, x) \leq {C_1'}^r r^{5r/2} \sqrt{n} \log^{3r/2} n.$$

Proof. By the same argument used in the discretization, we have that for any $x \in B'$

$$\begin{aligned} A(x, \ldots, x) &\leq (2\lceil r \log n \rceil)^r \max_{x^{(1)} \in Y^{(1)}(x), \ldots x^{(r)} \in Y^{(r)}(x)} A(x^{(1)}, \ldots, x^{(r)}) \\ &\leq (2\lceil r \log n \rceil)^r \max_{x^{(1)}, \ldots x^{(r)} \in U'} A(x^{(1)}, \ldots, x^{(r)}), \end{aligned} \tag{24}$$

where

$$U' = \{|S|^{-1/2} \chi^S : S \subseteq [n], |S \cap P| < C_5 \sqrt{|S| \log n}\}.$$

Consider an off-diagonal block $V_1 \times \ldots \times V_r$. For each $i \in 1 \ldots r$, let $P_i = V_i \cap P$ and let $R_i = V_i \setminus P$. Then, breaking the polynomial $A|_V(\cdot)$ up as a sum of 2^r terms, each corresponding to a choice of $S_1 \in \{P_1, R_1\}, \ldots, S_r \in \{P_r, R_r\}$ gives

$$\max_{x^{(1)},\ldots,x^{(r)} \in U'} A|_V(x^{(1)}, \ldots, x^{(r)}) \leq 2^r \max_{x^{(1)},\ldots,x^{(r)} \in U'} \sum_{S_1 \in \{P_1,R_1\},\ldots,S_r \in \{P_r,R_r\}} A|_{S_1 \times \ldots \times S_r}(x^{(1)}, \ldots, x^{(r)}). \quad (25)$$

By symmetry, without loss of generality we may consider the case where $S_i = R_i$ for $i = 1 \ldots r - \ell$ and $S_i = P_i$ for $i = r - \ell + 1 \ldots r$ for some ℓ. Let $\tilde{V} = R_1 \times \ldots \times R_{r-\ell} \times P_{r-\ell+1} \times \ldots \times P_r$. Then,

$$\max_{x^{(1)},\ldots,x^{(r)} \in U'} A|_{\tilde{V}}(x^{(1)}, \ldots, x^{(r)}) = \sum_{k_1 \in R_1} \cdots \sum_{k_{r-\ell} \in R_{r-\ell}} \prod_{i=1\ldots r-\ell} x^{(i)}_{k_i} \prod_{i,j:i,j \leq r-\ell} E_{k_i k_j} B^{(k_1,\ldots,k_{r-\ell})},$$

where (as defined (18))

$$B^{(k_1,\ldots,k_{r-\ell})}(x^{(r-\ell+1)}, \ldots, x^r) \sum_{k_{r-\ell+1} \in P_{r-\ell+1}} \cdots \sum_{k_r \in P_r} \prod_{i=r-\ell+1\ldots r} x^{(i)}_{k_i} \prod_{i,j:i,r-\ell+1<j} E_{k_i k_j}.$$

By the assumption that every $x^{(i)} \in U'$, this value is at most $(C_5 \log n)^{\ell/2}$. Thus,

$$\max_{x^{(1)},\ldots,x^{(r)} \in U'} A|_{\tilde{V}}(x^{(1)}, \ldots, x^{(r)}) \leq \sum_{k_1 \in R_1} \cdots \sum_{k_{r-\ell} \in R_{r-\ell}} \prod_{i=1\ldots r-\ell} x^{(i)}_{k_i} \prod_{i,j:i,j \leq r-\ell} E_{k_i k_j} (C_5 \log n)^{\ell/2}.$$

Note that every edge $E_{k_i k_j}$ above is random, so the polynomial may be bounded according to Lemma 46. Altogether,

$$\max_{x^{(1)},\ldots,x^{(r)} \in U'} A|_{\tilde{V}}(x^{(1)}, \ldots, x^{(r)}) \leq (\max\{C_5, C_3\} \log n)^{r/2}.$$

Combining (24),(25), and applying Lemma 41 completes the proof with C_1' chosen large enough. □

Proof of Theorem 9. The clique is found by finding a vector x such that $A(x, \ldots, x) \geq \alpha^r |P|^{r/2}$ and then running Algorithm 7.4 on this vector. Algorithm 7.4 clearly runs in polynomial time, so the theorem holds if the algorithm succeeds with high probability.

By Lemma 49 the algorithm does succeed with high probability when $x \notin B'$, i.e. when some $S \in \{\sup(y - \lceil r \log n \rceil(x), \ldots, \sup(y - \lceil r \log n \rceil(x)\}$ satisfies $|S \cap P| \geq C_5 \sqrt{|S| \log n}$.

We claim $x \notin B'$ with high probability. Otherwise, for some $x \in B'$,

$$A(x, \ldots, x) \geq \alpha^r p^{r/2} > C_0^r r^{5r/2} \sqrt{n} \log^{3r/2} n.$$

This is a low probability event by Lemma 50 if $C_0 \geq C_1'$. □

APPENDIX A

HARDNESS OF TENSOR POLYNOMIAL MAXIMIZATION

A.1 Introduction

In general, it is hard to find the maximum of non-concave functions, even over convex sets. An interesting exception to this rule is maximizing the quadratic forms of symmetric matrices over the unit ball, i.e. finding $x \in B(0,1)$ such that

$$x^T A x \geq \max_{x \in B(0,1)} x^T A x - \epsilon$$

where A is a symmetric matrix and ϵ is some small value (potentially exponentially small in the size of the problem). The top eigenvector A is such a maximizing vector, and it can be approximated in polynomial time and quite efficiently in practice. The simplest algorithm is to multiply a random vector x with a high power of A, say A^p, and return $A^p x / \|A^p x\|$.

A natural extension of this problem is to maximize the polynomial of some higher order symmetric tensor instead of the polynomial of a symmetric matrix. That is, for a symmetric tensor A of order r, find $x^* \in B(0,1)$ such that $A(x^*) = \max_{x \in B(0,1)} A(x) - \epsilon$, where

$$A(x) = \sum_{i_1, \ldots, i_r} A_{i_1 \ldots i_r} x_{i_1} \ldots x_{i_r}.$$

Here we show that there is no polynomial time algorithm to solve this problem for tensors of order at least 4, unless $P = NP$. To prove this, we provide a reduction from max-cut.

Max-cut was one of the original twenty-one NP-complete problems identified by Karp [30]. For a graph $G = ([n], E)$, the maximum-cut is a partition of [n] into two

complementary sets, $P, \bar{P}$ such that $|P \times \bar{P} \cap E|$ is maximized over all partitions. Håstad has shown that it is NP-hard to approximate the maximum cut to within a factor of 16/17. Goemans and Williamson have given an approximation algorithm achieves a 0.878 factor approximation [24]. Assuming the unique games conjecture [33] and $BPP \neq NP$ this is the best possible polynomial time approximation [34]

Theorem 11. *Let $\alpha \in (1/2, 1]$ be a fixed constant and let $\alpha' > \alpha$. For any graph G with n vertices, a cut of size α times the maximum cut can be found in time $O(n)+T(n, r, \alpha')$, where $T(n, r, \alpha')$ is the time necessary to find $x \in B(0, 1)$ such that*

$$A(x) \geq \alpha'^{\lfloor r/4 \rfloor} \max_{x \in B(0,1)} A(x),$$

where A is an rth order tensor polynomial for $r \geq 4$.

Corollary 51. *For any $\alpha' > 16/17$, it is NP-hard to find $x \in B(0, 1)$ such that*

$$A(x) \geq \alpha'^{\lfloor r/4 \rfloor} \max_{x \in B(0,1)} A(x),$$

where A is an rth order tensor polynomial for $r \geq 4$.

A.2 Reduction

The max-cut problem can naturally be thought of as maximizing the function

$$\sum_{(i,j) \in E} |x_i - x_j|,$$

where x is constrained to the $-1, 1$ lattice. Turning this into a 4th order tensor polynomial is easy, as we can simply replace $|x_i - x_j|$ with $(x_i - x_j)^4$ and preserve the maxima. More challenging, however, is ensuring that these are the maxima, not just over the lattice, but also over the ball. Let $P, \bar{P}$ be a maximum cut, which cuts M edges and let x be the vector where $x_i = 1/\sqrt{n}$ if $i \in P$ and $x_i = -1/\sqrt{n}$ if $i \in \bar{P}$. Then

$$\sum_{(i,j) \in E} (x_i - x_j)^4 = \frac{16M}{n^2}.$$

On the other hand, if $x = e_i$, then

$$\sum_{(i,j)\in E} (x_i - x_j)^4 = \text{degree}(i),$$

which could be much larger.

To cope with this difficulty, we add a penalty function to our objective, choosing to maximize

$$A(x) = \sum_{(i,j)\in E} (x_i - x_j)^4 - C \sum_{i,j\in[n]} (x_i^2 - x_j^2)^2. \tag{26}$$

For large enough, C this forces the maxima of $A(x)$ to be close to the $-1/\sqrt{n}, 1/\sqrt{n}$ lattice as desired. Rounding to the nearest lattice point gives the solution as described in Alg 6.

To give the objective function a higher order, we raise the original A to the $\lfloor r/4 \rfloor$th power and add a dummy variable x_{n+1} to account for the remainder. The tensor polynomial becomes

$$A^{(r)}(x) = x_{n+1}^{(r \bmod 4)} \left(\sum_{(i,j)\in E} (x_i - x_j)^4 - C \sum_{i,j\in[n]} (x_i^2 - x_j^2)^2 \right)^{\lfloor r/4 \rfloor}.$$

Algorithm 6 Max-cut

Input: Graph G.
Output: A partition of the vertices $P, \bar{P}$.

1. Find x such that $A^{(r)}(x) \geq \alpha'^{\lfloor r/4 \rfloor} \max_{x\in B(0,1)} A^{(r)}(x)$, where

$$A^{(r)}(x) = x_{n+1}^{(r \bmod 4)} \left(\sum_{(i,j)\in E} (x_i - x_j)^4 - C \sum_{i,j\in[n]} (x_i^2 - x_j^2)^2 \right)^{\lfloor r/4 \rfloor}.$$

2. Set $P = \{i \in [n] : x_i > 0\}$, $\bar{P} = [n] \setminus P$.

A.3 Analysis

We begin by giving the reduction for $r = 4$.

Lemma 52. *Let α and α' be fixed constants such that $1 \geq \alpha' > \alpha \geq 1/2$ and let A be defined according to (26). For large enough C, if $x \in B(0,1)$ satisfies*

$$A(x) \geq \max\left\{\alpha' \max_{x \in B(0,1)} A(x), 16/n^2\right\},$$

then partitioning the vertices according to $\text{sign}(x_i)$ *yields a cut of size α times the maximum cut.*

Proof. Suppose not. Let $P, \bar{P}$ be a maximum cut and let z be the unit vector where $z_i = 1/\sqrt{n}$ if $i \in P$ and $z_i = -1/\sqrt{n}$ otherwise. Similarly, let y be the unit vector where $y_i = \text{sign}(x_i)/\sqrt{n}$. If x does not yield an α approximation to the max-cut, then $A(z) > A(y)/\alpha$.

Therefore, we argue

$$\begin{aligned} A(x) &\geq \alpha' \max_{x \in B(0,1)} A(x) \\ &\geq \alpha' A(z) \\ &> \frac{\alpha'}{\alpha} A(y) \\ &= \frac{\alpha'}{\alpha} A(x) - \frac{\alpha'}{\alpha}(A(x) - A(y)) \\ &\geq A(x) + (\frac{\alpha'}{\alpha} - 1)A(x) - \frac{\alpha'}{\alpha}(A(x) - A(y)). \end{aligned} \tag{27}$$

Letting $\epsilon = x - y$,

$$\begin{aligned} A(x) - A(y) &\leq \sum_{i,j \in E} (y_i + \epsilon_i - y_j - \epsilon_j)^4 - (y_i - y_j)^4 \\ &\leq \sum_{i,j \in E} 4(\epsilon_i - \epsilon_j)(y_i - y_j)^3 + 6(\epsilon_i - \epsilon_j)^2(y_i - y_j)^2 \\ &\quad +4(\epsilon_i - \epsilon_j)^3(y_i - y_j) + (\epsilon_i - \epsilon_j)^4 \\ &\leq 30n^2 \max_i |\epsilon_i|. \end{aligned}$$

By the following claim, ϵ_i is controlled by the parameter C.

Claim 53. *If x is a unit vector such that $A(x) > 0$, then*

$$\max_i |\epsilon_i| \leq \sqrt{\frac{n^3}{C}}.$$

For large enough C, therefore

$$\max_i |\epsilon_i| \leq \left(1 - \frac{\alpha}{\alpha'}\right) \frac{1}{2n^4}.$$

Thus,

$$\begin{aligned} A(x) - A(y) &< \left(1 - \frac{\alpha}{\alpha'}\right) \frac{16}{n^2} \\ &\leq \left(1 - \frac{\alpha}{\alpha'}\right) A(x). \end{aligned} \tag{28}$$

Combining (27) and (28) shows $A(x) > A(x)$, yielding a contradiction and proving the lemma. □

Proof of Claim 53. Recall that y is the vector such that $y_i = \text{sign}(x_i)/\sqrt{n}$) and that $\epsilon = x - y$. Note that for every i such that $x_i > 0$, $\epsilon_i > -1/\sqrt{n}$ and for every i such that $x_i < 0$, $\epsilon_i < 1/\sqrt{n}$. Thus, $|2y + \epsilon| > 1/\sqrt{n}$. It follows that

$$\max_i \frac{\epsilon_i^2}{n} \leq \max_i \epsilon_i^2 (2y_i + \epsilon_i)^2 = \max_i ((y_i + \epsilon_i)^2 - 1/n)^2 \leq \sum_{i,j} ({x_i}^2 - {x_j}^2)^2.$$

Now, because $A(x) > 0$,

$$C \sum_{i,j} ({x_i}^2 - {x_j}^2)^2 < \sum_{(i,j) \in E} (x_i - x_j)^4 < n^2.$$

Combining the above inequalities shows that

$$C \max_i \frac{\epsilon^2}{n} < n^2,$$

proving the lemma. □

A.3.1 Higher Order Tensors

Clearly, when r is a multiple of 4, then $A^{(r)}(x) = (A^{(4)}(x))^r/4$ Hence, an $\alpha'^{\lfloor r/4 \rfloor}$ approximation to $A^{(r)}$ yields an α' approximation to $A^{(4)}$.

Now consider the case where $r = 4t + k$ for $k \in \{1, 2, 3\}$. Let x be a unit vector in $\mathbb{R}^{n+1}$ that achieves the α'^t factor approximation, and let $\tilde{x}$ be the restriction of this vector to the first n vertices, normalized to unit sphere. Then,

$$\begin{aligned}\left(\frac{k}{n}\right)^{k/2}\left(1-\frac{k}{n}\right)^{(n-k)/2}\max_{x\in B(0,1)} A^{(4t)}(x) &= \max_{x\in B(0,1)} A^{(4t+k)}(x)\\ &\leq \alpha'^{4t} A(x)\\ &\leq \alpha'^{4t}\left(\frac{k}{n}\right)^{k/2}\left(1-\frac{k}{n}\right)^{(n-k)/2} A^{(4t)}(\tilde{(x)})\end{aligned}$$

Thus, a α'^t factor approximation for $A^{(4t+k)}$ also yields a α'^t factor approximation for $A^{(4t)}$, via a restriction to the first n coordinates. Therefore, $\tilde{x}$ reveals an α approximation to the max-cut.

APPENDIX B

RECOVERING A CLIQUE

Here, we give a Frieze and Kannan's proof of Lemma 49 for the reader's convenience. First, we show that the top eigenvector of $A|_{S\times S}$ is close to the indicator vector for $S \cap P$.

Claim 54. *There is a constant C such that for every $S \subseteq [n]$ where $|S \cap P| \geq C\sqrt{|S| \log n}$, the top eigenvector v of the matrix $A|_{S\times S}$ satisfies*

$$\sum_{i \in S \cap P} v_i > \sqrt{|S \cap P|/2}$$

Proof. The adjacency matrix A can be written as the sum of $\chi^P \chi^{P^T}$ and a matrix R representing the randomly chosen edges. Let $u = \chi^{S\cap P}/\sqrt{|S \cap P|}$ Suppose that v is the top eigenvector of $A|_{S\times S}$ and let $c = u \cdot v$. Then

$$\begin{aligned}
|S \cap P|^{1/2} &= A(u,u) \\
&\leq A|_{S\times S}(v,v) \\
&= c^2 A|_{S\times S}(u,u) \\
&\quad +2c\sqrt{1-c^2}A|_{S\times S}(u, v-cu) + (1-c^2)A|_{S\times S}(v-cu, v-cu) \\
&\leq c^2|S \cap P|^{1/2} + 3\|R|_{S\times S}\|.
\end{aligned}$$

Hence

$$c^2 \geq 1 - 3\frac{\|R|_{S\times S}\|}{C\sqrt{|S| \log n}}.$$

By taking a union bound over the subsets S of a fixed size, it follows from well-known results on the norms of symmetric matrices ([23, 49], also Lemma 42) that with high probability

$$\|R|_{S\times S}\| = O(\sqrt{|S| \log n})$$

for every $S \subseteq [n]$. Therefore, the theorem holds for a large enough constant C. $\square$

Next, we show that the clique is dense in the first $8|S \cap P|$ coordinates (ordered according to the top eigenvector v).

Claim 55. *Suppose* $v_1 \geq \ldots \geq v_n$ *and* $\sum_{i \in S \cap P} v_i > \sqrt{|S \cap P|/2}$. *Then for* $\ell = 8|S \cap P|$

$$|[\ell] \cap P| \geq \frac{|S \cap P|}{8}.$$

Proof of Claim 55. For any integer ℓ,

$$\begin{aligned}
\sqrt{\ell} &\geq \sum_{i \leq \ell} v_i \\
&\geq \frac{\ell}{|S \cap P|} \sum_{i > \ell, i \in P} v_i \\
&= \frac{\ell}{|S \cap P|} \left(\sum_{i \in P} v_i - \sum_{i \leq \ell, i \in P} v_i \right) \\
&\geq \frac{\ell}{|S \cap P|} \left(\sqrt{|S \cap P|/2} - \sqrt{|[\ell] \cap P|} \right).
\end{aligned}$$

Thus,

$$\sqrt{|[\ell] \cap P|} \geq \sqrt{|S \cap P|/2} - \frac{|S \cap P|}{\sqrt{\ell}}.$$

Taking $\ell = 8|S \cap P|$ (optimal), we have

$$\sqrt{|[\ell] \cap P|} \geq \frac{1}{2\sqrt{2}} \sqrt{|S \cap P|}.$$

$\square$

Given this density, it is possible to pick $10 \log n$ vertices from the clique and use this as a seed to find the rest of the clique. When $\ell = 8|S \cap P|$, in each iteration there is at least a

$$8^{-10 \log n} = n^{-30}$$

chance that $Q_1 \subseteq P$. With high probability, no set of $10 \log n$ vertices in P has more than $2 \log n$ common neighbors outside of P in G. The contrary probability is

$$\binom{|P|}{10 \log n} \binom{n}{2 \log n} 2^{-20 \log^2 n} = o(1).$$

Letting Q_2 be the common neighbors of Q_1 in G, it follows that $Q_2 \supseteq P$ and $|Q_2 \backslash P| \leq 2 \log n$. Now, with high probability no common neighbor has degree more than $3|P|/4$ in P, because

$$n \binom{|P|}{10 \log n} \binom{n}{2 \log n} \exp(-|P|/24) = o(1).$$

for $|P| > 312 \log^2 n$.

Thus, with high probability no vertex outside of P will have degree greater than $7|P|/8$ in the subgraph induced by Q_2.

APPENDIX C

ROBUST PCA CODE

```
%-------------------------------------------------------------
%File: main.m
%-------------------------------------------------------------
function main()

%Multiplier for radius
beta = 3.0;
%Upper bound on fraction of bad examples
epsilon = 1/6;
%Number of dimensions
n = 100;

[A,id] = get_adverse_data(n, beta);

%display mask
dm = rand(1, size(A,2)) < 0.1;

figure(1);
plot(A(1, dm & ~(id == 4)),A(2, dm & ~(id == 4)),'b*');
hold on;
plot(A(1, dm & (id == 4)),A(2, dm & (id == 4)),'r+');
axis equal;
title('Projection to Intermean Subspace');
hold off;

figure(2);
[B,P] = my_pca(A,2);
plot(B(1, dm & ~(id == 4)), B(2, dm & ~(id == 4)),'b*');
hold on;
plot(B(1, dm & (id == 4)), B(2, dm &(id == 4)),'r+');
axis equal;
title('Projection to PCA Subspace');
hold off;

figure(3);
[B,P] = rpca(A,2,epsilon,beta);
plot(B(1, dm & ~(id == 4)), B(2, dm & ~(id == 4)),'b*')
hold on
plot(B(1, dm & (id == 4)), B(2, dm & (id == 4)),'r+');
axis equal;
title('Projection to Robust PCA Subspace');
hold off;

%-------------------------------------------------------------
%File: get_adverse_data.m
%-------------------------------------------------------------
function [A,id] = get_adverse_data(n,beta)
%Input:
% n: the number of dimensions
% beta: factor to be used by denoising
%Output:
```

```
% A: a matrix whose cols are the data points
% id: a integer vector indicating which points belong to
% which component.

k = 3;
mgood = 1000;
%mgood = 100;
mbad =  mgood/2;
%Component Means and Variances
U = [0,-sqrt(3)/2, sqrt(3)/2; 1, -0.5, -0.5; zeros(n-2,k)];
s = [.1,.1,.1];
s_max = max(s);

%Good data
[G,id] = sample_sphere_mix([mgood,mgood, mgood], U, s );

%Approximate radius of the good data.
r = sqrt(max(sum((G - repmat(G(:,1),1,3*mgood)).^2,1)));

%Bad data
s = 2;
t = floor(mbad/2/s);
B = [zeros(2,2*s*t); repmat([eye(s),-eye(s)],1,t) * beta/2 * r];
B = [B;  zeros(n-2-s,size(B,2))];

%All data
A = [G,B];
id = [id ,int32((k+1)*ones(1,size(B,2)))];

end

function [A,id] = sample_sphere_mix(m, U, s)
% m: a vector of length k indicating the number of samples
% from each component.
% U: a matrix whose columns are the means of the components.
% s: a vector of length k indicating the variance of each
% component.

n = size(U,1);
k = size(U,2);

A = [];
id = int32([]);

for i=1:k
  A = [A, s(i) * randn(n,m(i)) + repmat(U(:,i),1,m(i))];
  id = [id, repmat(i,1,m(i))];
end
end


%-------------------------------------------------------------
%File: rpca.m
%-------------------------------------------------------------
function [B,P] = rpca(A,k,epsilon,beta)
%Input:
% A: a matrix whose cols are the data points.
% k: the number of components to be returned
% epsilon: an upper bound on the fraction of bad data
% beta: multipier for denoising
%Output:
```

```
% B: Projection of cols of A onto top k components
% P: a matrix whose rows are the top k components

n = size(A,1);

A_c = A;
P_c = eye(n);
while n > 2
  mask = reject_outliers(A_c,epsilon, beta);
  n = ceil(n/2);
  [A_c,P] = my_pca(A_c(:,mask),n);
  P_c = P * P_c;
end
B = P_c * A;
P = P_c;

%--------------------------------------------------------------
%File: my_pca.m
%--------------------------------------------------------------
function [B,P] = my_pca(A,k)
%Input:
% A: a matrix whose cols are the data points.
% k: the number of principal components to be returned
%Output:
% B: Projection of cols of A onto top k principal components
% P: a matrix whose rows are the top k principal components

m = size(A,2);
c = sum(A,2);

[U,S,V] = svd(A*A' - c * c'/m);

P = U(:,1:k)';

B =  P * A;

%--------------------------------------------------------------
%File: reject_outliers.m
%--------------------------------------------------------------
function mask = reject_outliers(A,delta, beta)
%Input:
% A: a matrix whose cols are the data points.
% delta: an upper bound on the fraction of noise points.
% beta: a multiplier to determine the acceptance radius.
%Output:
% mask: a boolean vector indicating 1 for non-noise and 0
% for noise

%Downsampling to speed things up
mask = rand(1,size(A,2)) < 10/(delta * size(A,2));
B = A(:,mask);

%Number of samples
m = size(B,2);

%How many distances should we distrust?
r = floor(m * delta);

%Computing rth furthest points
d = zeros(m,1);
```

```
idx = zeros(m,1);
for i=1:m
    z = sum((B - repmat(B(:,i),1,m)).^2,1);
    [x,y] = sort(z,'descend');
    d(i) = x(r+1);
    idx(i) = y(r+1);
end
[x,y] = sort(d,'descend');

t = x(r+1);
p = B(:,idx(y(r+1)));

mask = sum((A - repmat(p,1,size(A,2))).^2) < beta * t;
```

REFERENCES

[1] Achlioptas, D. and McSherry, F., "On spectral learning of mixtures of distributions," in *Proc. of COLT*, 2005.

[2] Alon, N., Krivelevich, M., and Sudakov, B., "Finding a large hidden clique in a random graph," *Random Structures and Algorithms*, vol. 13, pp. 457–466, 1998.

[3] Arora, S. and Kannan, R., "Learning mixtures of arbitrary gaussians," *Annals of Applied Probability*, vol. 15, no. 1A, pp. 69–92, 2005.

[4] Berry, M., Dumais, S., and O'Brien, G., "Using linear algebra for intelligent information retrieval," *SIAM Review*, vol. 37, no. 4, pp. 573–595, 1995.

[5] Brubaker, S. C., "Robust pca and clustering on noisy mixtures," in *Proc. of SODA*, 2009.

[6] Brubaker, S. C. and Vempala, S., "Isotropic pca and affine-invariant clustering," in *Building Bridges Between Mathematics and Computer Science* (Grötschel, M. and Katona, G., eds.), vol. 19 of *Bolyai Society Mathematical Studies*, 2008.

[7] Brubaker, S. C. and Vempala, S., "Random tensors and planted cliques," in *Proc. of RANDOM*, 2009.

[8] Chaudhuri, K. and Rao, S., "Beyond gaussians: Spectral methods for learning mixtures of heavy-tailed distributions," in *Proc. of COLT*, 2008.

[9] Chaudhuri, K. and Rao, S., "Learning mixtures of product distributions using correlations and independence," in *Proc. of COLT*, 2008.

[10] Dasgupta, A., Hopcroft, J., Kleinberg, J., and Sandler, M., "On learning mixtures of heavy-tailed distributions," in *Proc. of FOCS*, 2005.

[11] DasGupta, S., "Learning mixtures of gaussians," in *Proc. of FOCS*, 1999.

[12] DasGupta, S. and Schulman, L., "A two-round variant of em for gaussian mixtures," in *Proc. of UAI*, 2000.

[13] Dempster, A., Laird, N., and Rubin, D., "Maximum likelihood from incomplete data via the em algorithm," *JRSS B*, vol. 39, pp. 1–38, 1977.

[14] Duda, R. O., Hart, P., and Stork, D., *Pattern Classification*. John Wiley & Sons, 2001.

[15] Dumais, S., Furnas, G., Landauer, T., and Deerwester, S., "Using latent semantic analysis to improve information retrieval," in *Proc. of CHI*, pp. 281–285, 1988.

[16] Feige, U. and Krauthgamer, R., "Finding and certifying a large hidden clique in a semirandom graph," *Random Structures and Algorithms*, vol. 16, no. 2, pp. 195–208, 2000.

[17] Feldman, J. and O'Donnell, R., "Learning mixtures of product distributions over discrete domains," *SIAM Journal on Computing*, vol. 37, no. 5, pp. 1536–1564, 2008.

[18] Feldman, J., Servedio, R. A., and O'Donnell, R., "Pac learning axis-aligned mixtures of gaussians with no separation assumption," in *Proc. of COLT*, pp. 20–34, 2006.

[19] Francis, J., "The qr transformation: a unitary analogue to the lr transformation," *Computer Journal*, vol. 4, no. 1-2, pp. 265–272,332–345, 1961.

[20] Freund, Y. and Mansour, Y., "Estimating a mixture of two product distributions," in *Proc. of COLT*, pp. 53–62, 1999.

[21] Frieze, A. and Kannan, R., "A new approach to the planted clique problem," in *Proc. of FST & TCS*, 2008.

[22] Fukunaga, K., *Introduction to Statistical Pattern Recognition*. Academic Press, 1990.

[23] Füredi, Z. and Komlós, J., "The eigenvalues of random symmetric matrices," *Combinatorica*, vol. 1, no. 3, pp. 233–241, 1981.

[24] Goemans, M. and Willamson, D., "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming," *Journal of the ACM*, vol. 42, no. 6, pp. 1115–1145, 1995.

[25] Golub, G. and van der Vorst, H., "Eigenvalue computation in the 20th century," *Journal of Computational and Applied Mathematics*, vol. 123, pp. 35–65, 2000.

[26] H. Moon, P. P., "Computational and performance aspects of pca-based face recognition algorithms," *Perception*, vol. 30, pp. 303–321, 2001.

[27] Hawkins, T., "Cauchy and the spectral theory of matrices," *Historia Mathematica*, vol. 2, pp. 1–20, 1975.

[28] Jerrum, M., "Large cliques elude the metropolis process," *Random Structures and Algorithms*, vol. 3, no. 4, pp. 347–360, 1992.

[29] Kannan, R., Salmasian, H., and Vempala, S., "The spectral method for general mixture models," *SIAM Journal on Computing*, vol. 38, no. 3, pp. 1141–1156, 2008.

[30] Karp, R., "Reducibility among combinatorial problems," in *Complexity of Computer Computation*, pp. 85–103, Plenum Press, 1972.

[31] Karp, R., "The probabilistic analysis of some combinatorial search algorithms," in *Algorithms and Complexity: New Directions and Recent Results*, pp. 1–19, Academic Press, 1976.

[32] Kearns, M. and Li, M., "Learning in the presence of malicious errors," *SIAM Journal on Computing*, vol. 22, no. 4, pp. 807–837, 1993.

[33] Khot, S., "On the power of unique 2-prover 1-round games," in *Proc. of IEEE Conference on Computational Complexity*, 2002.

[34] Khot, S., Kindler, G., Mossel, E., and O'Donnell, R., "Optimal inapproximability results for max-cut and other 2-variable csps?," *SIAM Journal on Computing*, vol. 37, no. 1, pp. 319–357, 2007.

[35] Kline, M., *Mathematical thought from ancient to modern times*. Oxford University Press, 1971.

[36] Kublanovskaya, V., "On some algorithms for the solution of the complete eigenvalue problem," *USSR Computational Mathematics and Mathematical Physics*, vol. 3, pp. 637–657, 1961.

[37] Kucera, L., "Expected complexity of graph partitioning problems," *Discrete Applied Mathematics*, vol. 57, pp. 193–212, 1995.

[38] Lovász, L. and Vempala, S., "The geometry of logconcave functions and sampling algorithms," *Random Structures and Algorithms*, vol. 30, no. 3, pp. 307–358, 2007.

[39] MacQueen, J. B., "Some methods for classification and analysis of multivariate observations," in *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 281–297, 1967.

[40] McSherry, F., "Spectral partitioning of random graphs," in *FOCS*, pp. 529–537, 2001.

[41] Papadimitriou, C., Raghavan, P., Tamaki, H., and Vempala, S., "Latent semantic indexing: A probabilistic analysis," in *Proc. of PODS*, 1998.

[42] Pearson, K., "On lines and planes of closest fit to systems of points in space," *Philosophical Magazine*, vol. 2, no. 6, pp. 559–572, 1901.

[43] Rudelson, M., "Random vectors in the isotropic position," *Journal of Functional Analysis*, vol. 164, pp. 60–72, 1999.

[44] Rudelson, M. and Vershynin, R., "Sampling from large matrices: An approach through geometric functional analysis," *Journal of the ACM*, vol. 54, no. 4, 2007.

[45] Stewart, G. and guang Sun, J., *Matrix Perturbation Theory*. Academic Press, Inc., 1990.

[46] Turk, M. and Pentland, A., "Faces recognition using eigenfaces," in *Proc. of CVPR*, pp. 586–591, 1991.

[47] Valiant, L. G., "Learning disjunction of conjunctions," in *Proc. of IJCAI*, pp. 560–566, 1985.

[48] Vempala, S. and Wang, G., "A spectral algorithm for learning mixtures of distributions," *Journal of Computer and System Sciences*, vol. 68, no. 4, pp. 841–860, 2004.

[49] Vu, V. H., "Spectral norm of random matrices," in *Proc. of STOC*, pp. 423–430, 2005.