TEXT-CLASSIFICATION METHODS AND THE MATHEMATICAL THEORY OF PRINCIPAL COMPONENTS

A Dissertation
Presented to
The Academic Faculty

By

Jiangning Chen

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

Georgia Institute of Technology

August 2019

TEXT-CLASSIFICATION METHODS AND THE MATHEMATICAL THEORY OF PRINCIPAL COMPONENTS

Approved by:

Dr. Matzinger, Advisor
School of Mathematics
Georgia Institute of Technology

Dr. Lounici, Advisor
School of Mathematics
Georgia Institute of Technology

Dr. Popescu
School of Mathematics
Georgia Institute of Technology

Dr. Bonetto
School of Mathematics
Georgia Institute of Technology

Dr. Huo
School of Industrial and Systems Engineering
Georgia Institute of Technology

Date Approved: April 8, 2019

Nous sommes des enfants, mais des enfants progressifs, pleins de force et de courage.

Évariste Galois

TO MY FAMILY

ACKNOWLEDGEMENTS

This thesis could not have been successfully completed without the invaluable assistance of many individuals and institutes. The following list of acknowledgements is, by no means, exhaustive.

I would like to first express my appreciation and gratitude to my advisors, Dr Karim Lounici and Dr Heinrich Matzinger, for their assistance, guidance and support. It is a great honor and great pleasure to work with them and I learned a lot from them in all these years. Their inspiring guidance helped me overcome the obstacles and difficulties in my Ph.D. research. Nothing can be achieved without their continuous support and encouragement during my Ph.D. study

Next, I would like to thank Dr. Popescu, Dr. Bonetto and Dr. Huo for serving as my committee members. I would like to extend my sincere gratitude to Dr. Ionel Popescu for his numerous guidance and suggestion on my research. I would also like to specially recognize Dr. Christian Houdre, Dr. Vladimir I. Koltchinskii, Dr. Mayya Zhilova and Dr. Haomin Zhou for their extraordinary teaching of probability and statistics related topics, which inspired me to do research in this field.

I am also very much indebted to the School of Mathematics at Georgia Institute of Technology for providing me continuous financial support opportunities throughout my Ph.D. studies. In particular, I would like to extend my gratitude to the English classes offered by Dr. Morag Burke, as well as the training and advice on teaching given by Ms. Klara Grodzinsky, these are all very valuable experience which I will never forget.

I've made a lot of friends during the Ph.D. study, they meant so much to me over years. I especially want to thank Dr. Yang Cao for introducing me to the opportunity at Georgia Tech, navigating me through the application and helping me settle down when I first came here. I have shared with great happiness and my friendship with all of them, including Tongzhou Chen, Juntao Duan, Qianli Hu, Changong Li, Ruilin Li, Zhibo Dai, Xiaowen

Sun, Xin Xing, Jiaqi Yang, Haoyan Zhai, Dr. Fan Zhou and Yirui Zhuang.

I would also like to thank all of my mathematics teachers during my early years. Many of them are worth-mentioning, including Guangmei Yin, Mei Yi, Peng Li, Yong Xu, Dr. Jihuai Shi and Dr. Guangtian Song. Particularly, I would like to extend my gratitude to my aunt Hua Jiang and uncle Tao Wang, they are all math teachers, without their inspiration and encouragement, I would never have pursued this path thus far.

Last but absolutely not least, I want to thank my parents, Jingyong Chen and Rong Jiang, for giving me the life, unconditional love and support. I would also like to say thanks to my wife, Hua Deng, for her constant love and support and for spending her life with me, I love her so much.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

This thesis studies three topics. First of all, in text classification, one may use Principal Components Analysis (PCA) as a dimension reduction technique, or with few topics even as unsupervised classification method. We investigate how useful it is for real life problems. The problem is that, often times the spectrum of the covariance matrix is wrongly estimated due to the ratio between sample space dimension over feature space dimension not being large enough. We show how to reconstruct the spectrum of the ground truth covariance matrix, given the spectrum of the estimated covariance for multivariate normal vectors. We then present an algorithm for reconstruction the spectrum in the case of sparse matrices related to text classification.

In the second part, we concentrate on schemes of PCA estimators. Consider the problem of finding the least eigenvalue and eigenvector of ground truth covariance matrix, a famous classical estimator are due to Krasulina. We state the convergence proof of Krasulina for the least eigenvalue and corresponding eigenvector, and then find their convergence rate.

In the last part, we consider the application problem, text classification, in the supervised view with traditional Naive-Bayes method. We find out an updated Naive-Bayes method with a new loss function, which loses the unbiased property of traditional Naive-Bayes method, but obtains a smaller variance of the estimator.

CHAPTER 1
SIZE OF THE SAMPLE NEEDED TO BE ABLE TO USE PRINCIPAL COMPONENT FOR DIMENSION REDUCTION

1.1 Introduction

Principal Components is often used in high dimensional statistics and machine learning to reduce dimension before applying another algorithm. Many times without the dimension reduction, over-fitting would happen. The principal components of a covariance matrix can be decomposed into two classes: the Principal Components which contain the structural information and the noise ones.

Consider a data matrix X of dimension $n \times p$, where the rows are i.i.d copies of a random vector $\vec{X}$. The largest eigenvectors (called Principal Components) of the covariance matrix $COV[\vec{X}]$ contain the structural information. So projecting the data onto the span of the leading eigenvectors usually operates a dimension reduction without loss of information.

The problem is, in reality, we are not given the population covariance $COV[\vec{X}]$, instead, we only know the estimated covariance, which is defined as the sample covariance

$$C\hat{O}V[\vec{X}] = \frac{X^t X}{n}.$$

Thus, instead of taking the eigenvectors with large eigenvalues of the population covariance, we take instead the eigenvectors with large eigevalues from the sample covariance. The covariance estimation error matrix E given by

$$E = C\hat{O}V[\vec{X}] - COV[\vec{X}],$$

which usually perturbs things a lot in the high dimensional where n and p are of same order.

Typically the eigenvectors (PCA) with leading eigenvalues get mixed up by E.

Since in many cases, we only want to reconstruct the span of the structural principal components and not reconstruct them individually, we only need the structural eigenvectors to not get mixed up with noise eigenvectors. Because in most cases, the principal components are not directly the partitions, but the linear combinations of them, see our examples about customer reviews in below.

Using a bound [1] of Koltchinskii and Lounici for the l_2-norm of E, we are able to show that by just increasing the sample size by a quantity $O(1)$, we are able to reconstruct without noise eigenvectors. We also need the eigenvalues of the eigenvectors, whose span we wish to reconstruct, to be separated from the noisy eigenvalues. We show that this is not the case for reconstructing the eigenvectors individually. Our proof involves a detailed look at the magnitude of the error E in different direction of the space, since in different regions, there will be vastly different orders. If one just uses the classical inequality from perturbation theory and applies the bound [1], one gets a bound which is often too large. In many theoretical models, they are in larger order. We show a numerical example, where the structural eigenvalues are only order $O(1)$ away from the noise eigenvalues. This is why our analysis is relevant.

Say you want to predict stars from customer reviews, that is you have a collection of customer's reviews where the customer also included a star ranking of product. You use this collection as training set. Trying to predict the number of stars given the review. Then, when you have some costumer's reviews which lacks the star-ranking you can predict it using your algorithm. Now, you can make each word in the texts as a feature and try any machine learning algorithm on it. Typically this will not work at all because of overfitting, thus, we need to reduce the dimension with Principal Components. Once your texts are hot-encoded you find the Principal Components of the data set and keep only a small number. Then you project the data on the principal components and use this as input for whatever machine learning algorithm you use. It is important to note that for this task it is not important to retrieve the eigenvectors, but only be able to retrieve the eigenspace generated by the leading eigenvectors. Thus, if the eigenvectors get mixed up among themselves, it does not matter.

Now, one of the problems is that you are getting the eigenvectors from the sample covariance matrix instead of the true covariance matrix. But the structural information is contained in the eigenvectors of the true covariance matrix. We know that when the sample size becomes large the sample covariance approaches the true covariance. The question is how large does the sample need to be, in order to be able to retrieve the subspace generated by the structural eigenvectors.

In this chapter, we are able to give the exact order up to a constant given the eigenvalues of the true covariance matrix for the case when the data is normal and under a few additional realistic assumptions.

Let us first give a simple example in financial stocks to clarify things. We know that the stocks may not be the best place of applications, but it is easy to understand

Let $\vec{X} = (X_1, X_2, X_3, \ldots, X_{2p})$ be the vector containing the daily returns of $2p$ different stocks on a given day. Assuming a two sector model and a general economy index M. Let S be the index of the first sector and T be an index of the second sector whilst M is the index of the general economy. We assume that the first p stocks depend only on S and M, so that for $i = 1, 2, \ldots, p$, we have

$$X_i = a_i S + c_i M + \epsilon_i.$$

And the stocks with indices from $p + 1$ to $2p$ depend only on T and M, so that for $i = p + 1, \ldots, 2p$

$$X_i = b_i T + c_i M + \epsilon_i.$$

Here the coefficients a_i, b_i and c_i are supposed to be constants. The term ϵ_i is a firm specific term. We assume that S, T, M and the ϵ_i's are uncorrelated. Let us also assume all the ϵ_i's have the same covariance σ^2.

Then the covariance matrix of the stocks is given by:

$$COV[\vec{X}] = \vec{a} \otimes \vec{a}^T + \vec{b} \otimes \vec{b}^T + \vec{c} \otimes \vec{c}^T + \sigma^2 I,$$

where

$$\vec{a} = (a_1, a_2, \ldots, a_p, 0, \ldots, 0)^T, \vec{b} = (0, 0, \ldots, 0, b_1, b_2, \ldots, b_p)^T, \vec{c} = (c_1, c_2, \ldots, \ldots, c_{2p})^T,$$

and I is the $2p \times 2p$ identity matrix.

If we assume that the general economy has no influence, that is: $c_1 = c_2 = \ldots = c_{2p} = 0$, we will have two eigenvectors: $\vec{\mu}_1 = \vec{a}, \vec{\mu}_2 = \vec{b}$ with leading eigenvalues:

$$\lambda_1 = \sum_i a_i^2 + \sigma^2, \lambda_2 = \sum_i b_i^2 + \sigma^2. \tag{1.1.1}$$

And all the other eigenvalues will equal to σ^2.

At this stage $\vec{a}$ and $\vec{b}$ are the leading principal components, because they have the two largest eigenvalues, while all other eigenvalues are of smaller order. Also note that these two vectors have non-zero entries only where the stock belongs to the corresponding sectors. So, these eigenvectors carry the **structural information** about which stocks belong to which sector.

Now, since we often work with standardized data, assuming also that S and T are standardized so that they have variance 1. Thus, the a_i's and b_i's are simply the correlation coefficients between stock i and the corresponding sector index. We also assume that the daily return of the stock is standardized. The coefficients are less than 1 in absolute value, in each sector they should also be bounded away from 0. Hence, for standardized data, since $VAR[X_i] = 1$ from 1.1.1 we find the order $\lambda_1, \lambda_2 = O(p)$ and $\sigma^2 < 1$.

This gives the general setting when you have a finite number of eigenvalues of order

$O(p)$, which carry the structural information, while the other eigenvalues are of order $O(1)$.

When the coefficients $c_i \neq 0$, we have:

$$COV[\vec{X}] = \vec{a} \otimes \vec{a} + \vec{b} \otimes \vec{b} + \vec{c} \otimes \vec{c} + \sigma^2 I.$$

Since the vectors $\vec{a}$, $\vec{b}$ and $\vec{c}$ are not necessarily orthogonal, these three vectors in general will not be eigenvectors. Instead, the three leading eigenvectors of the covariance matrix will be in the linear span of $< \vec{a}, \vec{b}, \vec{c} >$. So, in this case it would be of no use to reconstruct the principal components separately: we only need the span of the three largest eigenvectors, rather then having them separately. And anyhow, the vectors $\vec{a}, \vec{b}$ are not themselves' eigenvectors if $\vec{c}$ is not an eigenvector.

In general, the term ϵ_i will also not have all the same standard deviation. Put $\sigma_{\epsilon_i} := \sigma_i$, then:

$$COV[\vec{X}] = \vec{a} \otimes \vec{a} + \vec{b} \otimes \vec{b} + \vec{c} \otimes \vec{c} + \text{Diag}(\sigma_1^2, \sigma_2^2, \cdots, \sigma_{2p}^2),$$

where $\text{Diag}(\sigma_1^2, \sigma_2^2, \cdots, \sigma_{2p}^2)$ is a diagonal matrix with entries $\sigma_1^2, \sigma_2^2, \cdots, \sigma_{2p}^2$. In that case, there will be 3 large eigenvalues and all others will be of order 1. These others will be refered to as noise eigenvalues and the corresponding eigenvectors as noise eigenvector.

So the next question is how do we find out the span generated by the leading principal components. The answer is: we estimate the covariance matrix, and take the eigenvectors of the estimated covariance with largest eigenvectors for estimating the leading principal components.

For example, as the daily returns for n days for our $2p$ stocks, let $\vec{X}_i$ be the i-th day return vector:

$$\vec{X}_i = (X_{i1}, X_{i2}, \ldots, X_{i(2p)}),$$

where X_{ij} is the return of stock j on day i. Let X be the $n \times (2p)$ matrix obtained by stacking the $\vec{X}_i$, and assuming the rows to be i.i.d. normal. Since on the daily return basis, the expectation is a smaller order than standard deviation, we can assume it is 0 and thus we have estimated covariance:

$$C\hat{O}V[\vec{X}] := \frac{X^t \cdot X}{n}.$$

Note that when $n < 2p$, the estimated covariance above is defective, it has at least half the eigenvalues 0. Hence, half the eigenvalues of the estimate in that case would be wrong by a size $O(1)$. So using the estimated (also called sample) covariance matrix for finding the eigenvectors with largest eigenvalues can be problematic. Furthermore, often in theory, the eigenvalues corresponding to structural eigenvectors are of order $O(p)$. However, in real applications they are not very big for most of the times, and not that far from 1. The reason is as follows: take stocks for example. How many are there? Maybe 1000 but certainly not a million. So even though the leading eigenvalues theoretically grow linearly, often this does not help since we can not grow the data set too large.

From now on assume that the dimension of $\vec{X} = (X_1, X_2, \ldots, X_p)$ is p and not $2p$. Let n be the sample size. Then X is a $n \times p$ matrix with i.i.d multivariate normal rows with expectation 0 and each distributed like $\vec{X}$. Let λ_i be the i-the eigenvalues (in decreasing order) of $COV[\vec{X}]$ with corresponding unit eigenvector $\vec{\mu}_i$. That is to say, $\vec{\mu}_i$ is the i-the Principal Component of $COV[\vec{X}_i]$.

Let $\hat{\lambda}_i$ be the i-the eigenvalue (in decreasing order) of the estimated covariance matrix $C\hat{O}V[\vec{X}] = X^t X/n$ with corresponding unit eigenvector $\hat{\vec{\mu}}_i$ (Also in decreasing order). To simplify discussion, we assume that the noise eigenvalues are between 1 and 0. So, say λ_k is the first eigenvalue corresponding to noise, and $\lambda_k = 1$. Then we have the spectrum:

$$\lambda_1 > \lambda_2 > \lambda_3 > ... > \lambda_k > \lambda_{k+1} > ... > \lambda_p.$$

By the assumption above, the eigenvectors $\vec{\mu}_1, \vec{\mu}_2, \ldots, \vec{\mu}_{k-1}$ are structural ones.

Now for $i < k$, you compute $\hat{\vec{\mu}}_i$ from estimated covariance, is this a reliable estimation of $\vec{\mu}_i$? So $\hat{\lambda}_i$ is i-th eigenvalue of estimated covariance matrix $\frac{X^t X}{n}$, corresponding to eigenvector $\hat{\vec{\mu}}_i$. As we discussed before, we do not care that $\hat{\vec{\mu}}_i$ gets mixed up with the structural eigenvectors, since anyhow we just need to retrieve the linear span of the structural eigenvectors for dimension reduction. We only want to assure that we don't get a lot of the eigevectors $\vec{\mu}_j$ with $j \geq k$ in our estimate $\hat{\vec{\mu}}_i$. In other words, we want to keep the projection of $\hat{\vec{\mu}}_i$ onto the span $< \vec{\mu}_k, \vec{\mu}_{k+1}, \ldots, \vec{\mu}_p >$ smaller than a given constant $\epsilon > 0$.

What is the condition for this? First we assume that $\lambda_i \geq 2\lambda_k$, and the estimated eigenvalues $\hat{\lambda}_i$ will in general be larger than λ_i, by the fact large eigenvalues tend to be overestimated. Thus we also assume $\hat{\lambda}_i \geq 2\lambda_k$. With that assumption, we are able to use the result of Koltschinskii and Lounici to find that condition (1.1.5).

It is interestingly enough to increase the sample size by a $O(1)$ to be able to achieve the desired result, whereas we show that to retrieve the structural eigenvalues separately, this is not enough.

Let E denote the covariance estimation error matrix:

$$E := C\hat{O}V[\vec{X}] - COV[\vec{X}].$$

Then we can view the estimated covariance as the true covariance plus the perturbation E:

$$C\hat{O}V[\vec{X}] = COV[\vec{X}] + E.$$

Now, the true covariance $COV[\vec{X}]$ contains the structural information. But we are not given that ground truth covariance, instead we are given a perturbed version $COV[\vec{X}] + E$. Consider the coordinate system of the Principal components, let $Y_j := \vec{X}\vec{u}_j$ where $\vec{u}_j$ is the j-th eigenvector (Principal Component) of the covariance matrix. So, we work with

$$\vec{Y} = (Y_1, Y_2, \ldots, Y_p),$$

which has a diagonal covariance matrix since the y_j's are independent of each other. (It is known that, for normal vector, when we express the vector in basis of pca, we get independent coordinates). Furthermore, the covariance matrix is equal to

$$COV[\vec{Y}] = \text{Diag}(\sigma_j^2),$$

where $\sigma_j^2 := \sqrt{VAR[Y_j]}$, and assume $\sigma_1 > \sigma_2 > \ldots > \sigma_p$. With this notation, the eigenvalues can be written as $\lambda_j = \sigma_j^2$ for all $j = 1, 2, 3, \ldots, p$.

Note that $COV[\vec{Y}]$ and $COV[\vec{X}]$ have the same eigenvalues since one is obtained from the other by applying a unitary transformation. Assuming that $\vec{Y}^{(i)}$ is an i-th independent copy of $\vec{Y}$. In the new coordinate system, the estimated covariance is now equal to:

$$\hat{\Sigma} := C\hat{O}V[\vec{Y}] = \frac{1}{n} \sum_{i=1}^{n} \left[\vec{Y}^{(i)}\right]^{\mathrm{T}} \vec{Y}^{(i)}.$$

and the true covariance is:

$$\Sigma := \text{Diag}(\sigma_1^2, \ldots, \sigma_p^2).$$

Note that this does not change the spectral norm of the estimation error E, nor the eigenvalues of Σ or $\hat{\Sigma}$, nor any inner products between vectors, as Q is orthogonal. But the new coordinate system renders the analysis simpler, since all off-diagonal coefficients of Σ are zero and its eigenvectors are the canonical unit vectors $\vec{\mu}_i$, $(i = 1, \ldots, p)$.

We denote the spectral norm of the estimation error by $|E|$. It was long know that when $\sigma_i = 1$, $(i = 1, \ldots, p)$, then $|E|$ is typically of order a constant times $\sqrt{p}$, but in the case where the eigenvalues are not all of the same order of magnitude, the order of $|E|$ was unknown until the recent work of Koltchinskii and Lounici [1, 2], who proved that up to an unknown universal constant $C_1 > 0$, $|E|$ is typically bounded by

$$|E| \leq C_1 \times \frac{\max_{i \in \{1,2,\ldots,p\}} \sigma_i}{\sqrt{n}} \times \sqrt{\sigma_1^2 + \sigma_2^2 + \ldots + \sigma_p^2}, \tag{1.1.2}$$

with high probability.

They also proved that this is a tight bound. Since the approximation error in the i-th eigenvector of Σ can be bounded by $\frac{|E|}{\texttt{spectral gap}_\texttt{i}}$, this error is significantly smaller than $\epsilon \in (0,1)$ as long as n is large enough to guarantee that $\frac{|E|}{\texttt{spectral gap}_\texttt{i}} < \epsilon$ with high probability. Using (1.1.2), this yields the requirement

$$\sqrt{n} \geq \frac{C_1}{\epsilon} \times \frac{\max_{j=1,\ldots,p} \sigma_j}{\texttt{spectral gap}_\texttt{i}} \times \sqrt{\sigma_1^2 + \sigma_2^2 + \ldots + \sigma_p^2}. \tag{1.1.3}$$

In the present paper we show that this bound can be improved to

$$\sqrt{n} \geq \log(p)\frac{C}{\epsilon} \times \frac{\sigma_{i^*}}{\sqrt{\texttt{spectral gap}_{\texttt{i}^*}}} \times \sqrt{\sum_{j \neq i^*} \frac{\sigma_j^2}{|\sigma_j^2 - \sigma_{i^*}^2|}}, \tag{1.1.4}$$

which is to hold with high probability for some universal constant $C > 0$. Here i^* is the random index, which is the value for s so that $|\sigma_s^2 - \hat{\sigma}_i^2|$ gets minimized, and we added the term $\log(p)$ for later technical use. Note that 1.1.4 is satisfied when we have

$$\sqrt{n} \geq \frac{\log(p)C}{\epsilon} \frac{\sigma_{i^*}}{\texttt{spectral gap}_{\texttt{i}^*}} \sqrt{\sum_{j \neq i^*} \sigma_j^2}.$$

In other words, it is the index of the eigenvalue of the original covariance matrix which comes closest to the i-th estimated eigenvalue $\hat{\sigma}_i^2$. Actually, in formula 1.1.4, we could replace $\sigma_{i^*}^2$ by $\hat{\sigma}_i^2$, but later on, this would be more problematic. So here 1.1.4 is the condition to be added to retrieve eigenvector number i individually.

We numerate the estimated eigenvalues in descending order. Then typically, $\hat{\sigma}_i^2$ often does not vary a lot, and we could thus think of i^* as being very close to a constant. Note that for the (random) eigenvalues $\sigma_{i^*}^2$, the new bound (1.1.4) is of strictly smaller order than (1.1.3) due to the factor $\max_{j=1,2,\ldots,p} \sigma_j$ having been replaced by σ_{i^*} and to the square root in the spectral gap, the latter often being smaller than 1. In general, expression (1.1.4) is less than or equal to expression (1.1.3).

After having given the conditions to be able to retrieve eigenvector under 1.1.4, the main aim of the current chapter is to show what sample size is needed to ensure that a structural eigenvector does not get mixed up with the noise eigenvectors. We assume that the noise eigenvalues start at σ_k^2 and $\hat{\sigma}_i^2 > 2\sigma_k^2$, $(i > k)$. We find the bound:

$$\boxed{\sqrt{n} \geq \frac{C}{\epsilon} \frac{\sigma_{i^*}}{Gap_i} \sqrt{\sum_{j \neq i^*} \sigma_j^2}} \tag{1.1.5}$$

to guaranty this (see 1.3.24). Here $Gap_i := |\hat{\sigma}_i^2 - \sigma_k^2|$, hence Gap_i is not spectral gap, but the distance to the closest noise eigenvalue (So not the distance to the closest eigenvalue). The key point is that, on the right side of 1.1.5 we have $\hat{\sigma}_i$ instead of σ_1. With the classical bound is what we would get and it would be in most real life situations (Note: to prove this bound, we are slightly less precise than for the proof of 1.1.4, which is why we do not have the term $\log(p)$ in front).

To get the bound of the form (1.1.4), we need to avoid the trouble that the denominator might equal to 0, so we make the following mild condition on σ_i^2:

Condition 1.1.1. The spectrum $\{\sigma_1^2, \sigma_2^2, ..., \sigma_p^2\}$ of Σ satisfis: $\texttt{spectral gap}_\texttt{i} \neq \texttt{0}$, where $\texttt{spectral gap}_\texttt{i} := \min\{|\sigma_\texttt{j}^2 - \sigma_\texttt{i}^2| : \texttt{j} \neq \texttt{i}\}$.

Note that Condition1.1.1 does not necessarily mean that $\hat{\sigma}_i^2$ is the i-th largest eigenvalue of $\hat{\Sigma}$, and that it is not guaranteed that the condition is satisfied for all i. However, if all eigenvalues of Σ are non-coalescent, Condition 1.1.1 is asymptotically met for all i, and $\hat{\sigma}_i^2$ is asymptotically the i-th largest eigenvalue of $\hat{\Sigma}$. So let β be some constant number, for $\beta > 0.5$ we can reconstruct the eigenvalues of order $O(p^\beta)$ in sub-linear time, whilst for $\beta < 0.5$ we can not. Of course, often time in PCA as mentioned, we do not even want to reconstruct each eigenvector separately: rather we want to reconstruct a subspace of eigenvectors with large eigenvalues as a whole, for dimension reduction.

1.2 Numerical evaluation of text classification

Every thing in below, we test using real data. We test on real data but also re-simulate data to get synthetic data and test. The re-simulation is done because our real data sets are limited in size and we would like in many cases to see how large we need to take n to be able to have the structural eigenvectors not mixed up. So, we estimate the parameters from real data and then with that we re-simulate.

For text classification problem, let us first consider a simple example, which all documents $\vec{D}$ are samples generated by only 2 class. We treat each word as a independent feature with a fixed probability to appear in one document, then each document can be considered as a sample generated from a multinomial distribution. Now, assuming that first n_p documents are from class C_1, we have $d_i = p_i C_1 + \epsilon_i$,and assume the second n_q documents are from by class C_2, we have $d_i = q_i C_2 + \epsilon_i$, where ϵ_i is the random noise.

Then the covariance matrix of the documents is given by:

$$COV[\vec{D}] = \vec{p} \otimes \vec{p} + \vec{q} \otimes \vec{q} + D_\epsilon,$$

where

$$\vec{p} = (p_1, p_2, \ldots, p_{n_p}, 0, \ldots, 0)^T, \vec{q} = (0, 0, \ldots, 0, q_1, q_2, \ldots, q_{n_q})^T,$$

and D_ϵ is a diagonal matrix correspondence for the noise eigenvectors.

Now consider estimated covariance matrix $\frac{E[X^t X]}{n}$, where X is the matrix whose row i corresponding to the document i and has 0 and 1 entries depending on if the word appears.

We encode the documents with the way we mentioned above in Reuter's data [3], we pick the documents with sample size between 100 to 500, and only keep the highest correlation words for each topics (we finally keep 1000 words). After encoding, we get 8 classes and a big matrix with about 2000 rows and 1000 columns, each row is a sample, and each column is a feature, if the feature appears in that sample, we encode the corresponding

Table 1.1: Result of PCA clustering, each entry tells how many documents in that class are classified as corresponding Principal Component

Classes	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
'money-supply'	0	0	110	0	30	10	3	0
'coffee'	0	0	55	1	0	2	0	58
'sugar'	0	0	140	2	0	0	0	1
'trade'	7	82	171	95	1	2	0	1
'ship'	0	0	154	0	0	2	0	1
'crude'	0	0	315	0	1	3	2	83
'interest'	10	1	120	2	26	122	3	0
'money-fx'	90	0	110	3	28	10	62	2

entry 1, otherwise, we encode the corresponding entry 0.

Now we want to see if we can correctly classify these samples by this matrix.

PCA of XX^t is the traditional way to solve the problem: we first compute the covariance matrix of the samples, and operate the eigen-decomposition to the covariance matrix, then clustering samples by k-means algorithm to each of the eigenvector. This method works when the number of classes less than 3, when number of class greater than 3, it behaves terrible. See Table.1.1, we can see that most of the documents are classified in the direction of PC3, which means naive PCA does not work well in Reuter's dataset. As we mentioned above, there are two reasons why this method doesn't work: 1) there are errors between estimated covariance matrix and true covariance matrix; 2) the true distribution vector should lie in the span of principal components, not the same direction of principal component.

In order to make it works, We firstly tried to see how eigen-decompostion works in recovery vector $\vec{p}$. We can estimate vector $\vec{p}$ by adding all samples from same class, and normalize it. The following figure 1.2 shows how eigen-decompostion works.

We can see that when we take the number of principal components around 20, the projection of distribution onto principal component span get around 90%. Here the eigenvectors are obtained by first using our formula to compute eigenvalues, and then re-simulate with enough samples, and finally compute top 20 eigenvectors from the re-simulated ma-

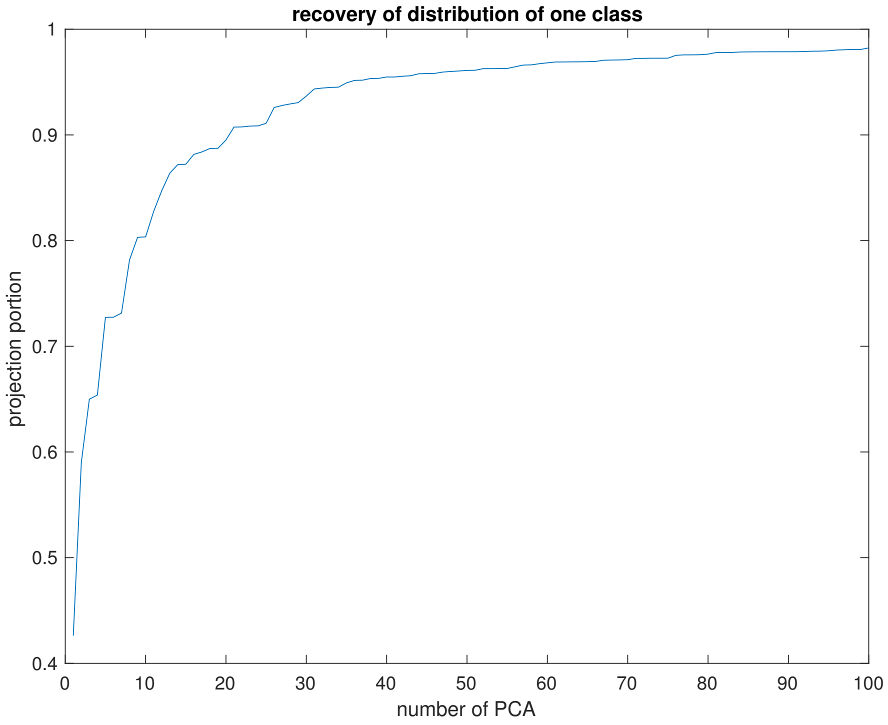


Figure 1.1: We take 100 principal components in estimated covariance, and project $\vec{p}$ onto the principal component plane. x-axis is the number of principal components we use, y-axis is the portion of projection.

trix.

So we take top 20 eigenvectors to get our PC space $\mathcal{S}_1$. And for each of the classes i, we take 3 samples, compute the average distribution $\vec{s_i}$, and project this sample onto our $\mathcal{S}$. We consider the distribution of class i as normalized vector: $\bar{s}_i = Proj_{\mathcal{S}}\vec{s_i}$. The following figure.1.2 shows the accuracy of text classification by using cosine similarity of $\bar{s}_i$ and the document, compared with using cosine similarity of the average distribution $\vec{s_i}$ and the document. We can see our method actually works better.

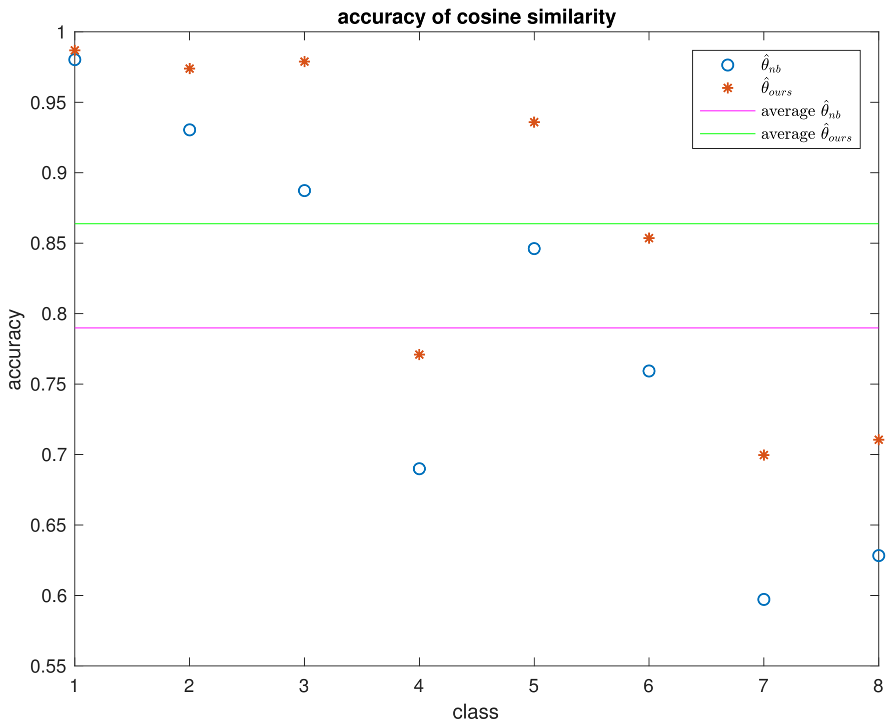


Figure 1.2: Accuracy of cosine similarity, using our estimator vs the average estimator. x-axis represents each class, y-xis shows the accuracy of that class. Two lines are average accuracy of two estimators.

1.3 Calculations

Next we do the calculation to prove our bounds for recovering a single eigenvector and also for recovering a span. For the span of eigenvectors we want to reconstruct only the span of those eigenvectors which contain structural information. So, we only have to prove that the eigenvectors of the sample covariance matrix with high index (we take $\hat{\sigma}_i^2$ to be an eigenvalue of the sample covariance matrix) which is larger than the noise-eigenvalues of the covariance matrix. For this we assume that at the index k, the noise eigenvalues start: that is $\sigma_k^2, \sigma_{k+1}^2, \ldots, \sigma_p^2$ are the eigenvalues corresponding to noise.

In what follows we have three subsections: the first one is for reconstructing a linear subspace of the structural eigenvector. Then we consider the finite dimensional case. Finally is about reconstructing a single eigenvector. In each there is slightly different notations about the eigenvector, let us summarize here:

1. For structural span reconstruction. We consider an eigenvalue of the sample covariance denoted by $\hat{\sigma}_i^2$ with $i > k$. We assume that it is larger than at least two times the largest noise-eigenvalues σ_k^2. The eigenvector corresponding to the eigenvalue $\hat{\sigma}_i^2$ is an eigenvector of the sample covariance matrix $\hat{\Sigma}$. We decompose that eigenvector into two orthogonal parts, so that the eigenvector corresponding to the eigenvalues $\hat{\sigma}^2$ can be written as the orthogonal sum: $\vec{u} + \Delta\vec{\mu}_i$. Here, $\vec{u}$ is not an eigenvector of the covariance matrix. Merely, $\vec{u}$ is contained in the structural part of the spectrum. That means that $\vec{u}$ is orthogonal to any noise-eigenvector of the covariance matrix, that is any eigenvector with index larger or equal to k. At the same time $\Delta\vec{\mu}_i$ is the part of the eigenvector which is the projection of the eigenvector onto the noise part of the spectrum. In other words, the projection onto the linear span of the eigenvectors of Σ having index larger equal to k. We also assume that the size of the eigenvector is 1. We simply assume the bound 1.1.2 on the norm of the covariance error matric $|E| = |\hat{\Sigma} - \Sigma|$ from Kolschinksii and Lounici [1] to hold with high probability in this part. We will just mention high probability, without quantifying it since any how we have a hard edge property. In the section 1.3.2, on getting a single eigenvector we are more precise and quantify the probabilities.

2. For reconstructing a single eigenvector $\vec{\mu}_i$ of the covariance matrix Σ with eigenvalue

λ_i. We have following setting:let $\vec{\mu}_i + \Delta\vec{\mu}_i$ be the corresponding eigenvector of the sample covariance matrix with corresponding eigenvalue $\hat{\sigma_i}^2$. Here $\Delta\vec{\mu}_i$ is taken orthogonal to $\vec{\mu}_i$. Then, we show a condition allowing to bound $\Delta\vec{\mu}_i$.

1.3.1 Bounding a sample eigenvector projection onto the noise PCA part

So here is the situation: We have a diagonal matrix Σ with entries

$$\sigma_1^2 > \sigma_2^2 > \ldots > \sigma_p^2$$

, and we perturb it with the matrix E. Now, assume that starting at k, the eigenvectors of Σ are noise, hence only the eigenvectors corresponding to the eigenvalues

$$\sigma_1^2, \sigma_2^2, \ldots, \sigma_{k-1}^2$$

are “structural eigenvalues”.

Let the perturbed matrix $\Sigma + E$ eigenvalue number i be denote by $\hat{\sigma}_i^2 = \lambda + \Delta\lambda$, where $i < k$. So in principal, with that index it should not be a noise eigenvalue, at least the corresponding eigenvector of Σ according to our assumption is structural.

The question is: does the same thing hold for the i-th eigenvector of the perturbed matrix? Let the i-th unit eigenvector of the perturbed matrix be denoted by $\hat{\vec{\mu}}_i$. Now, we decompose orthogonally into two pieces $\vec{u}$ and $\Delta\vec{\mu}_i$, so that

$$\hat{\vec{\mu}}_i = \vec{u} + \Delta\vec{\mu}_i$$

where $\Delta\vec{\mu}_i$ is the projection of $\hat{\vec{\mu}}_i$ onto the noise part of the spectrum that is onto the span $< \vec{\mu}_k, \vec{\mu}_k + 1, \ldots, \vec{\mu}_p >$. We also assume $\hat{\vec{\mu}}_i$ to have norm 1 and hence $|\vec{u}| \leq 1$.

Then $\vec{\mu}$ is the part in the structural part of the spectrum meaning that we can write

$$\vec{\mu} = (u_1, u_2, \ldots, u_{k-1}, 0, 0, 0, \ldots, 0),$$

for some coefficients $u_1, u_2, \ldots, u_{k-1}$. (Unlike what we do in singular eigenvalue reconstruction section, here, $\vec{u}$ is not necessarily an eigenvector of Σ). Now, we have the bound:

$$|\Delta\vec{\mu}_i| \leq \frac{|E|}{\mathtt{gap}_i}, \tag{1.3.1}$$

where

$$gap_i = |\hat{\sigma}_i^2 - \sigma_k^2|,$$

here $\mathtt{gap}_i$ does not represent the spectral gap to the next eigenvalue, but to the closest eigenvalues from a noise eigenvector. Now our inequality is a general inquality from perturbation theory. When we use the bound provided by Lounici and Koltchinskii to $|E|$, the inequality 1.3.1 yields

$$|\Delta\vec{\mu}_i| \leq C\frac{\sigma_1\sqrt{\sigma_1^2 + \sigma_2^2 + \ldots + \sigma_p^2}}{\sqrt{n} \cdot \mathtt{gap}_i}, \tag{1.3.2}$$

where C is a universal constant.

In the applied situations, we have in mind that the above inequality is not always optimal. Why? Typically σ_1 is of order $O(\sqrt{p})$, which is about $O(\sqrt{n})$ since in big data we often assume hand p of same order. Due to the normalization, we have $1 = \sigma_1^2 + \sigma_2^2 + \ldots + \sigma_p^2$. So the bound in inequality 1.3.2 is of order

$$O(\sigma_1/\mathtt{gap}_i). \tag{1.3.3}$$

In most real data application the first eigenvalue is gigantic, and then you have only a few eigenvalues which detach, but many eigenvalues which are structural are to be considered of order $O(1)$.

That means if you could increase things, then they would sometimes grow linearly in p, but often you can't see so even though theoretically these quantities would grow linearly with p, in reality we can not increase p. and hence have to think of some of the structural eigenvalues as being best modeled by $O(1)$. For example, in text classification. Take X to be the matrix documents by words. So, the i, j-the entry would be 1 if the j-th word appears in the i-th document and zero otherwise. Now, once you have all your vocabulary, you can not increase it. So if p is the number of words,you may not be able to increase. But say you analyze E-mails. You can probably increase their number. Often times in the applications we have in mind, we have values of 2 or 3 for eigenvalues which are important structurally.

In that case the ratio 1.3.3 is to be considerd of same order as σ_1 which typically would be order $O(\sqrt{p})$. In other words, in that case, we have no useful bound in 1.3.2. The goal now roughly speaking is to improve inequality given in 1.3.2 by having σ_1 being replaced by $\hat{\sigma}_i$, then we get the inequality:

$$|\Delta\vec{\mu}_i| \leq C\frac{\hat{\sigma}_i\sqrt{\sigma_1^2+\sigma_2^2+\ldots+\sigma_p^2}}{\sqrt{n}\cdot\mathtt{gap}_i}. \tag{1.3.4}$$

In that case, it will be possible to get $|\Delta\vec{u}_i|$ to be small with just having the ratio $\frac{n}{p}$ increase by a constant factor. (For this, note that $\mathtt{gap}_i$ and σ_i^2 have the same order. This is the case when we assume that $\hat{\sigma}_i^2$ is at least twice σ_k^2.)

Let us see how we can prove inequality 1.3.4. Let $\vec{u} + \Delta\vec{\mu}_i$ be an eigenvector of $\Sigma + E$ with eigenvalue $\hat{\sigma}_i^2$. Again Σ is the $p \times p$ covariance matrix whilst $\Sigma + E$ is the estimated covariance matrix $\hat{\Sigma}$. As before $\Delta\vec{\mu}_i$ is in the span of the eigenvectors of Σ which are "noise". That is, with index $\geq k$ which the eigenvectors are ordered according to decreasing eigenvalue. Furthermore $\vec{\mu}$ is the projection onto the orthogonal complement to the span generated by the "noise" eigenvalue. Hence, $\vec{\mu}$ has only the first $k-1$ entries which can be non zero. Since it is an eigenvector, we have:

$$(\Sigma + E)(\vec{u} + \Delta\vec{\mu}_i) = \hat{\sigma}_i^2 \cdot (\vec{u} + \Delta\vec{\mu}_i),$$

which yields:

$$(\Sigma - I \cdot \hat{\sigma}_i^2 + E)(\Delta\vec{\mu}_i) = -\vec{u}\Sigma + \hat{\sigma}_i^2 \cdot \vec{u} - E\vec{u}. \tag{1.3.5}$$

Now, when we apply the canonical orthogonal projection along the first $k-1$ coordinates, the terms $-\vec{u}\Sigma$ and $\hat{\sigma}_i^2 \cdot \vec{u}$ disappear. This is the same as taking the system of p equations given by 1.3.5 and leaving out the first $k-1$ equations. This yields (similarly to 1.3.35)

$$\Delta\vec{\mu}_i = D_i^{0.5} \cdot \left(\mathrm{I} - D_i^{0.5} E_i D_i^{0.5}\right)^{-1} \cdot D_i^{0.5} \cdot \begin{pmatrix} E_{k1} & E_{k2} & \dots & E_{k(k-1)} \\ E_{(k+1)1} & E_{(k+1)2} & \dots & E_{(k+1)(k-1)} \\ \vdots & \vdots & \ddots & \vdots \\ E_{p1} & E_{p2} & \dots & E_{p(k-1)} \end{pmatrix} \vec{u}, \tag{1.3.6}$$

where E_i designate the square $(p-k+1) \times (p-k+1)$ submatrix of E, which is obtained by deleting the first $k-1$ rows and the first $k-1$ columns:

$$E_i := E[k:p, k:p] = \begin{pmatrix} E_{kk} & E_{kk+1} & \dots & E_{kp} \\ E_{k+1,k} & E_{k+1,k+1} & \dots & E_{k+1,p} \\ \vdots & \vdots & \ddots & \vdots \\ E_{pk} & E_{pk+1} & \dots & E_{pp} \end{pmatrix}$$

Finally, we have D_i is the square matrix given by

$$D_i = \begin{pmatrix} \frac{1}{\lambda_k - \hat{\sigma}_i^2} & 0 & 0 & \dots & 0 \\ 0 & \frac{1}{\lambda_{k+1} - \hat{\sigma}_i^2} & 0 & \dots & 0 \\ 0 & 0 & \frac{1}{\lambda_{k+2} - \hat{\sigma}_i^2} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \frac{1}{\lambda_p - \hat{\sigma}_i^2} \end{pmatrix}.$$

Note that $D_i^{0.5}$ designates a square root of the matrix D_i, that is we replace each diagonal entry by its square root, even if the square root is a complex number.

Now, the first thing we want is to get

$$D_i^{0.5} E_i D_i^{0.5} \tag{1.3.7}$$

to be small. Let Σ_i be obtained from the diagonal covarianc matrix Σ by deleting the first $k-1$ rows and the first $k-1$ columns. Recall that $\vec{Y}$ is a vector of independent normal entries where the j-th entry has variance $\sigma_j^2 = \lambda_j$. Then we can write the random row vector Y as $\vec{Y} = \vec{N}\Sigma^{0.5}$, where $\vec{N}$ designates a row vector of length p with independent standard normal entries. If Y designates a matrix of dimension $n \times p$ with i.i.d. rows each having distribution like $\vec{Y}$, then our estimated covariance matrix is

$$\hat{\Sigma} = \frac{Y^t \cdot Y}{n} = \frac{\Sigma^{0.5} N \cdot N^t \Sigma^{0.5}}{n},$$

where N is a $n \times p$ matrix with standard normal entries.

So, for the covariance error matrix we get

$$E = \hat{\Sigma} - \Sigma = \frac{\Sigma^{0.5}(N \cdot N^t - I)\Sigma^{0.5}}{n}. \tag{1.3.8}$$

We can now apply formula 1.3.8 to the expression $D_i^{0.5} E_i D_i^{0.5}$ to get that:

$$D_i^{0.5} E_i D_i^{0.5} = \frac{D_i^{0.5} \Sigma_i^{0.5} (N \cdot N^t - I)_i \Sigma_i^{0.5} D_i^{0.5}}{n}, \tag{1.3.9}$$

where $\Sigma_i^{0.5}$ is obtained by deleting the first $k-1$ rows and columns from $\Sigma^{0.5}$, $(N \cdot N^t - I)_i$ is obtained from $(N \cdot N^t - I)$ by the same process. So the matrix on the right side of 1.3.9, is an estimated covariance matrix, but with coefficients σ_j being replaced by $\frac{\sigma_j}{\sqrt{\sigma_j^2 - \hat{\sigma}_i^2}}$, and j ranging over $k, k+1, \ldots, p$. Hence we can apply the formula of Koltchinskii and Klounici to find that, with high probability, the spectral norm:

$$|D_i^{0.5} E_i D_i^{0.5}| \leq \frac{C}{\sqrt{n}} \cdot \max_{j \geq k} \frac{\sigma_j}{\sqrt{|\sigma_j^2 - \hat{\sigma}_i^2|}} \cdot \sqrt{\Sigma_{j \geq k} \frac{\sigma_j^2}{|\sigma_j^2 - \hat{\sigma}_i^2|}}, \tag{1.3.10}$$

where again $C > 0$ is their universal constant. Now we assume that $\hat{\sigma}_i^2 \geq 2\sigma_k^2$, then

$$\max_{j \geq k} \frac{\sigma_j}{\sqrt{|\sigma_j^2 - \hat{\sigma}_i^2|}} \leq 1.$$

So if we want the right side of 1.3.10 to be less than a quantity ϵ, we just need the following inequality to hold:

$$\sqrt{n} \geq \frac{C}{\epsilon} \cdot \sqrt{\Sigma_{j \geq k} \frac{\sigma_j^2}{|\sigma_j^2 - \hat{\sigma}_i^2|}}. \tag{1.3.11}$$

Now we just have one more thing to bound in order to get $\Delta \vec{\mu}_i$ small according to 1.3.6. Note that we have:

$$D_i^{0.5} \cdot \begin{pmatrix} E_{k1} & E_{k2} & \ldots & E_{k(k-1)} \\ E_{(k+1)1} & E_{(k+1)2} & \ldots & E_{(k+1)(k-1)} \\ \vdots & \vdots & \ddots & \vdots \\ E_{p1} & E_{p2} & \ldots & E_{p(k-1)} \end{pmatrix} = \tag{1.3.12}$$

$$D_i^{0.5} \Sigma_i^{0.5} \left(\left(N^t N - I \right)_{[k:p, 1:(k-1)]} \right) \cdot diag(\sigma_1, \sigma_2, \ldots, \sigma_{k-1})) \tag{1.3.13}$$

where the restriction $(N^tN - I)_{[k:p,1:(k-1)]}$ is obtained from the matrix $N^tN - I$ by taking the first $k-1$ columns and the last k rows. Thus we have that 1.3.13 is the restriction of an estimated covariance matrix, and hence we could bound it using the Koltschinkii and Lounici formula. That estimated covariance matrix is equal to

$$diag(\vec{c}) \cdot \big(N^tN - I\big) \cdot diag(\vec{c}), \tag{1.3.14}$$

where $\vec{c}$ is the concatenation of the two vectors $(\sigma_1, \sigma_2, \ldots, \sigma_{k-1})$ and the vector obtained from taking the diagonal of $D_i^{0.5}\Sigma_i^{0.5}$. Also, we should mention that $diag(\vec{c})$ refers to the diagonal matrix, with $\vec{c}$ as diagonal. Now, you get 1.3.13 from the estimated covariance matrix 1.3.14 by deleting the first $k-1$ rows and the columns k to p. So a sum matrix has a smaller l_2 norm than the full matrix. Thus we can bound 1.3.14 using the koltschinskii and lounici formula and this gives us a bound on 1.3.13. That bound would be:

$$C \max_j c_j \frac{\sqrt{\sum_j c_j^2}}{\sqrt{n}}, \tag{1.3.15}$$

where $\vec{c} = (c_1, c_2, \ldots, c_p)$. Now, $\max_j c_j = \sigma_1$(Recall for this that typically $\mathtt{gap}_i > 1$, or at least that order of magnitude, which makes $\sigma_1 = c_1$ be the largest term of the vector $\vec{c}$). So in that bound we would have to use σ_1 instead of $\hat{\sigma}_i$, which would be needed to have our formula be useful in many cases (Assuming $\hat{\sigma}_i$ being of smaller order than σ_1).

Here is what we do: so far we had the first $k-1$ equations being the structural ones and we left them out and only used the other equations to obtain 1.3.6 from 1.3.5. This time we will leave out much less equations. For this, $k_1 > k_2$ are two integers so that: $k_1 := \max\{j|\sigma_j^2 > 2\hat{\sigma}_i^2\}$ and $k_2 := \min\{j|\sigma_j^2 j < \hat{\sigma}_i^2/2\}$.

Note that since we assume $\hat{\sigma}_i^2 > 2\sigma_k$, we have that $k_2 \leq k$. So this time from the system of equations 1.3.5 we keep the first k_1 and then those with index large than k_2, that is the last $p - k_2 - 1$. This time $\Delta\vec{\mu}_i$ is defined to be orthogonal projection of the eigenvector in 1.3.5 along the subspace generated by the subset of canonical vectors $\{\vec{e}_j | j \in [k_1, k_2]\}$, where

$\vec{e}_j$ refers to the k-th canonical vector. Similarly, vector $\vec{u}$ is now contained in subspace generated by the subset of canonical vectors $\{\vec{e}_j|\overrightarrow{j[}k_1, k_2]\}$.

We obtain then the same equation 1.3.35, but where E_i is the sub-matrix obtained from E by keeping the first k_1 rows and columns as well as those with index larger than k_2, as well as D_i and Σ_i.

Now, instead of having to bound 1.3.12, we will have to bound:

$$D_i^{0.5} \cdot \begin{pmatrix} E_{1k_1} & E_{1(k_1)+1} & \dots & E_{1(k_2-1)} \\ E_{2k_1} & E_{2(k_1+1)} & \dots & E_{2(k_2-1)} \\ \vdots & \vdots & \vdots & \vdots \\ E_{k_1k_1} & E_{k_1(k_1+1)} & \dots & E_{k_1(k_2-1)} \\ E_{k_2k_1} & E_{k_2(k_1)+1} & \dots & E_{k_2(k_2-1)} \\ E_{(k_2+1)k_1} & E_{(k_2+1)(k_1+1)} & \dots & E_{(k_2+1)(k_2-1)} \\ \vdots & \vdots & \vdots & \vdots \\ E_{pk_1} & E_{p(k_1+1)} & \dots & E_{p(k_2-1)} \end{pmatrix} = \tag{1.3.16}$$

$$D_i^{0.5}\Sigma_i^{0.5} \left(\left(N^tN - I\right)_{[1:k_1]U[k_2:p],[k_1:k_2]} \right) \cdot diag(\sigma_{k_1}, \sigma_{k_1+1}, \dots, \sigma_{k_2})) = \tag{1.3.17}$$

$$D_i^{0.5}\Sigma_i^{0.5} \left(\left(N^tN - I\right)_{[1:k_1]U[k_2:p],[k_1:k_2]} \right) \cdot diag(\frac{\sigma_{k_1}}{\hat{\sigma}_i}, \frac{\sigma_{k_1+1}}{\hat{\sigma}_i}, \dots, \frac{\sigma_{k_2}}{\hat{\sigma}_i}) \cdot \hat{\sigma}_i. \tag{1.3.18}$$

By our definition, we have

$$\frac{\sigma_j}{\hat{\sigma}_i} \leq 2 \tag{1.3.19}$$

for all $j \in (k_1, k_2)$, We can use this to bound expression 1.3.18. Expression 1.3.18 is a sub-matrix of an estimated covariance matrix times $\hat{\sigma}_i$, that estimated covariance matrix is similar to 1.3.14 with $\vec{c} = (c_1, c_2, \dots, c_p)$ different: $c_j := \frac{\sigma_j}{\sqrt{|\sigma_j^2 - \hat{\sigma}_i^2|}}$ for $j \notin [k_1, k_2]$, and $c_j := \frac{\sigma_j}{\hat{\sigma}_i}$ for $j \in [k_1, k_2]$.

Thus we get the upper bound 1.3.15. Note that this time $\max_j c_j \leq 2$, hence by plugging into the bound 1.3.15, we get that:

$$\begin{aligned}
D_i^{0.5} E_{[1:k_1]\cup[k_2:p],[k_1:k_2]} &\leq 2C\frac{\sqrt{\sum_j c_j^2}}{\sqrt{n}} \cdot \hat{\sigma}_i \\
&= 2C\frac{\hat{\sigma}_i}{\sqrt{n}}\sqrt{\sum_{j\in[k_1,k_2]} \frac{\sigma_j^2}{\hat{\sigma}_i^2} + \sum_{j\notin[k_1,k_2]} \frac{\sigma_j^2}{|\sigma_j^2 - \hat{\sigma}_i^2|}} \\
&\leq 2C\frac{\hat{\sigma}_i}{\sqrt{n}}\sqrt{\sum_{j\in[k_1,k_2]} 2 + \sum_{j\notin[k_1,k_2]} \frac{\sigma_j^2}{|\sigma_j^2 - \hat{\sigma}_i^2|}},
\end{aligned}$$

where the last inequality is obtained by 1.3.19.

Note that for $j \notin [k_1, k_2]$, we have $|\sigma_j^2 - \hat{\sigma}_i^2| \geq 0.5 \cdot \mathtt{gap}_i$, hence

$$\begin{aligned}
2C\frac{\hat{\sigma}_i}{\sqrt{n}}\sqrt{\sum_{j\in[k_1,k_2]} 2 + \sum_{j\notin[k_1,k_2]} \frac{\sigma_j^2}{|\sigma_j^2 - \hat{\sigma}_i^2|}} &\leq 2C\frac{\hat{\sigma}_i}{\sqrt{\mathtt{gap}_i}\sqrt{n}}\sqrt{\sum_{j\in[k_1,k_2]} 2\cdot \mathtt{gap}_i + \sum_{j\notin[k_1,k_2]} 2\sigma_j^2} \\
&\leq 16C\frac{\hat{\sigma}_i}{\sqrt{\mathtt{gap}_i}\sqrt{n}}\sqrt{\sum_{j\in[1,p]} \sigma_j^2},
\end{aligned}$$

where the very last inequality above is obtained by the fact that $4\mathtt{gap}_i \geq \sigma_j^2$ for all $j \in [k_1, k_2]$.

Thus we have:

$$\left| D_i^{0.5} \cdot \begin{pmatrix}
E_{1k_1} & E_{1(k_1)+1} & \dots & E_{1(k_2-1)} \\
E_{2k_1} & E_{2(k_1+1)} & \dots & E_{2(k_2-1)} \\
\vdots & \vdots & \vdots & \vdots \\
E_{k_1k_1} & E_{k_1(k_1+1)} & \dots & E_{k_1(k_2-1)} \\
E_{k_2k_1} & E_{k_2(k_1)+1} & \dots & E_{k_2(k_2-1)} \\
E_{(k_2+1)k_1} & E_{(k_2+1)(k_1+1)} & \dots & E_{(k_2+1)(k_2-1)} \\
\vdots & \vdots & \vdots & \vdots \\
E_{pk_1} & E_{p(k_1+1)} & \dots & E_{p(k_2-1)}
\end{pmatrix} \right| \leq 16C\frac{\hat{\sigma}_i}{\sqrt{\mathtt{gap}_i}\sqrt{n}}\sqrt{\sum_{j\in[1,p]} \sigma_j^2}, \tag{1.3.20}$$

where $\texttt{gap}_i = |\hat{\sigma}_i^2 - \sigma_k^2|$.

So, let us assume $\epsilon \in [0, 0.5]$. Then, we have

$$\frac{1}{1-\epsilon} = 1 + \epsilon + \epsilon^2 + \epsilon^3 + \ldots = 1 + \epsilon(1 + \epsilon + \epsilon^2 + \ldots) \leq 1 + 2\epsilon \leq 2. \tag{1.3.21}$$

Assume that

$$|D_i^{0.5} E_i D_i^{0.5}| \leq \epsilon, \tag{1.3.22}$$

where $\epsilon \in [0, 0.5]$. Then because of 1.3.21, we have

$$|I - D_i^{0.5} E_i D_i^{0.5}|^{-1} \leq 2. \tag{1.3.23}$$

From equation 1.3.6, and using 1.3.20, we get the following inequality:

$$|\Delta\vec{\mu}_i| \leq |D_i^{0.5}| \cdot \left| \left(\mathrm{I} - D_i^{0.5}\, E_i D_i^{0.5}\right)^{-1} \right| \, 16C \frac{\hat{\sigma}_i}{\sqrt{\texttt{gap}_i}\sqrt{n}} \sqrt{\sum_{j \in [1,p]} \sigma_j^2} \cdot |\vec{u}|.$$

The last inequality above together with the fact that by definition: $|D_i^{0.5}| \leq \frac{1}{\sqrt{\texttt{Gap}_i}}$, and assuming that 1.3.23 holds, implies that

$$\boxed{|\Delta\vec{\mu}_i| \leq 32C \cdot \frac{\hat{\sigma}_i}{\sqrt{n}} \frac{\sqrt{\sum_{j \in [1,p]} \sigma_j^2}}{\texttt{gap}_i}}. \tag{1.3.24}$$

We also used the fact that by definition $|\vec{u}| \leq 1$.

Now we have only one problem left: the bound 1.3.24 was obtained assuming 1.3.23. So, we need to see when 1.3.23 holds, or actually we need a bound of the type 1.3.22, for $\epsilon \in [0, 0.5]$.

To obtain this, first note that since by definition $\hat{\sigma}_i^2 > 2\sigma_k^2$, and since $\texttt{gap}_i = \hat{\sigma}_i^2 - \sigma_k^2$, we find $\frac{\hat{\sigma}_i^2}{\texttt{gap}_i} \leq 2$, applied to 1.3.24, leads to

$$|\Delta\vec{\mu}_i| \leq 32C \cdot \frac{\sqrt{2}}{\sqrt{n}} \frac{\sqrt{\sum_{j\in[1,p]} \sigma_j^2}}{\sqrt{\texttt{gap}_i}} \tag{1.3.25}$$

Next we rewrite inequality 1.3.10 considering that this time we have j in the integer set $J := [1, k_1] \cup [k_2, p]$. We obtain:

$$|D_i^{0.5} E_i D_i^{0.5}| \leq \frac{C}{\sqrt{n}} \cdot \max_{j\in J} \frac{\sigma_j}{\sqrt{|\sigma_j^2 - \hat{\sigma}_i^2|}} \cdot \sqrt{\Sigma_{j\in J} \frac{\sigma_j^2}{|\sigma_j^2 - \hat{\sigma}_i^2|}}, \tag{1.3.26}$$

which since in the current case the maximum on the right side of 1.3.26 is less than 2, we get

$$|D_i^{0.5} E_i D_i^{0.5}| \leq \frac{2C}{\sqrt{n}} \cdot \sqrt{\Sigma_{j\in J} \frac{\sigma_j^2}{|\sigma_j^2 - \hat{\sigma}_i^2|}} \leq \frac{2C}{\sqrt{n}} \cdot \frac{\sqrt{2\Sigma_j \sigma_j^2}}{\sqrt{\texttt{gap}_i}}. \tag{1.3.27}$$

For the very last inequality above, we used the fact that $2|\sigma_j^2 - \hat{\sigma}_i^2| \geq \texttt{gap}_i$ for all $j \in J$. Now, considering the bound on the right most side of 1.3.27, and note that it is bounded by the right side of 1.3.24. We assume $\hat{\sigma_i} > 1$, assume that $0.5 > \epsilon > 0$, if right side of 1.3.24 is less than ϵ, then by 1.3.27 we have

$$|D^{0.5} E_i D^{0.5}| \leq \epsilon \leq 0.5,$$

and hence we get 1.3.23 to hold, which implies that the bound 1.3.24 holds. To summarize: if the expreession on the right side of 1.3.24 is less or equal to 0.5, than inequality 1.3.24 holds. So, we don't have to worry that inequality 1.3.24 only holds when 1.3.23 holds. Because, when the bound on the right side of inequality 1.3.24 is small, then automatically inequality 1.3.24 is also true. Hence, inequality 1.3.24 holds as soon as the expression on the right side is less than 0.5.

1.3.2 Finite dimension case: reconstruct a single eigenvector

Take σ_i^2 to satisfy Condition 1.1.1 and write $\lambda_i = \sigma_i^2$ and $\lambda_i + \Delta\lambda_i = \hat{\sigma}_i^2$, as well as $\vec{\mu}_i$ and $\vec{\mu}_i + \Delta\vec{\mu}_i$ for eigenvectors of Σ and $\hat{\Sigma}$ that are associated with these eigenvalues. Furthermore, we take $\vec{\mu}_i$ to be a unit vector, and $\Delta\vec{\mu}_i$ to be orthogonal to $\vec{\mu}_i$, so that $\vec{\mu}_i + \Delta\vec{\mu}_i$ is not a unit vector. We denote the associated unit vector by

$$\hat{\vec{\mu}}_i = \frac{\vec{\mu}_i + \Delta\vec{\mu}_i}{\|\vec{\mu}_i + \Delta\vec{\mu}_i\|},$$

but find it easier to work with $\vec{\mu}_i + \Delta\vec{\mu}_i$, because the i-th component of the latter is zero. We may thus think of $\Delta\lambda_i$ and $\Delta\vec{\mu}_i$ as the perturbations to the eigenvalue λ_i and eigenvector $\vec{\mu}_i$ caused by adding the estimation error E to the ground truth covariance $\Sigma = \mathrm{COV}[\vec{X}]$.

Anderson [4] showed that for fixed p and n going to infinity,

$$\Delta\vec{\mu}_i \approx \frac{\sigma_i}{\sqrt{n}}\, Z_i, \tag{1.3.28}$$

where $Z_i = [Z_{1i}, \dots, Z_{pi}]$ is a random vector of size p with coefficients

$$Z_{ji} = \begin{cases} \dfrac{\sigma_j}{\sigma_j^2 - \sigma_i^2}\, N_{ji}, & (j \neq i), \\[2ex] 0, & (j = i), \end{cases}$$

and where the random variables N_{ji} converge (jointly, in distribution) to independent standard Gaussians when $n \to \infty$. Thus, to guarantee that $\|\Delta\vec{\mu}_i\| < \epsilon$ with high confidence, we need $\sigma_i \|Z_i\|/\sqrt{n} < \epsilon$ with high probability. Assuming the variables N_{ij} to be close to i.i.d. standard Gaussians and the approximation (1.3.28) to hold, one finds

$$\|\Delta\vec{\mu}_i\| \approx \frac{\sigma_i}{\sqrt{n}} \times \sqrt{\sum_{j\neq i} \frac{\sigma_j^2}{(\sigma_j^2 - \sigma_i^2)^2}},$$

which would imply that $\|\Delta\mu_i\| < \epsilon$ for

$$\sqrt{n} > \frac{C}{\epsilon} \times \sigma_i \times \sqrt{\sum_{j \neq i} \frac{\sigma_j^2}{(\sigma_j^2 - \sigma_i^2)^2}},$$

with C depending on the required confidence level. This bound would be of smaller order than (1.1.4), but unfortunately, the more stringent condition (1.1.4) is necessary for the approximation (1.3.28) to hold for n large enough independently of p.

Let us gain a quick oversight of how (1.3.28) arises in the finite-dimensional case, and how the argument has to be amended in the infinite-dimensional case: We have

$$\Sigma\, \vec{\mu}_i = \lambda_i \vec{\mu}_i, \tag{1.3.29}$$

$$\hat{\Sigma}\, [\vec{\mu}_i + \Delta\vec{\mu}_i] = (\lambda_i + \Delta\lambda_i)[\vec{\mu}_i + \Delta\vec{\mu}_i]. \tag{1.3.30}$$

Subtracting (1.3.29) from (1.3.30) and using $\hat{\Sigma} = \Sigma + E$ yields

$$[\Sigma - (\lambda_i + \Delta\lambda_i)\, \mathrm{I}_p]\, \Delta\vec{\mu}_i + E\Delta\vec{\mu}_i = -E\vec{\mu}_i + \Delta\lambda_i\vec{\mu}_i \tag{1.3.31}$$

where I_p is the $p \times p$ identity matrix. Now, in the finite-dimensional case where p is fixed and n tends to infinity, E, $\Delta\vec{\mu}_i$ and $\Delta\lambda_i$ are all of order $1/\sqrt{n}$, hence the terms $\Delta\lambda_i\Delta\vec{\mu}_i$ and $E\Delta\vec{\mu}_i$ are of the smaller order $1/n$ and can be neglected in the asymptotics, so as to arrive at the approximation

$$[\Sigma - \lambda_i\, \mathrm{I}_p]\, \Delta\vec{\mu}_i \approx -E\vec{\mu}_i + \Delta\lambda_i\vec{\mu}_i \tag{1.3.32}$$

Using the facts that $\Sigma = \mathrm{Diag}(\lambda_j)$, $\vec{\mu}_i$ is the i-th unit vector and that $\vec{\mu}_i$ and $\Delta\vec{\mu}_i$ are mutually orthogonal by construction, the i-th equation of system (1.3.32) yields

$$\Delta\lambda_i \approx E_{ii},$$

and dividing the j-th equation of the system (1.3.32) by $\lambda_j - \lambda_i$ $(j \neq i)$ yields

$$\Delta\vec{\mu}_i \approx -\begin{pmatrix} \dfrac{E_{1i}}{\sigma_1^2 - \sigma_i^2} \\ \dfrac{E_{2i}}{\sigma_2^2 - \sigma_i^2} \\ \vdots \\ \dfrac{E_{(i-1)i}}{\sigma_{i-1}^2 - \sigma_i} \\ 0 \\ \dfrac{E_{(i+1)i}}{\sigma_{i+1}^2 - \sigma_i^2} \\ \vdots \\ \dfrac{E_{pi}}{\sigma_p^2 - \sigma_i^2} \end{pmatrix} = \frac{-\sigma_i}{\sqrt{n}} \begin{pmatrix} \dfrac{\sigma_1}{\sigma_1^1 - \sigma_i^2} N_{1i} \\ \dfrac{\sigma_2}{\sigma_2^2 - \sigma_i^2} N_{2i} \\ \vdots \\ \dfrac{\sigma_{i-1}}{\sigma_{i-1}^2 - \sigma_i} N_{(i-1)i} \\ 0 \\ \dfrac{\sigma_{i+1}}{\sigma_{i+1}^2 - \sigma_i^2} N_{(i+1)i} \\ \vdots \\ \dfrac{\sigma_p}{\sigma_p^2 - \sigma_i^2} N_{pi} \end{pmatrix}, \qquad (1.3.33)$$

where

$$N_{st} := \frac{\sqrt{n}}{\sigma_s \sigma_t} E_{st} \qquad (1.3.34)$$

for all $s, t \in 1, \ldots, p$ with $s \neq t$. The random variables $N_{1i}, N_{2i}, \ldots, N_{pi}$ typically converge in joint distribution to i.i.d. standard Gaussians.

1.3.3 Infinite dimension case: reconstruct a single eigenvector

In contrast, in the infinite-dimensional case the terms $\Delta\lambda_i \Delta\vec{\mu}_i$ and $E\Delta\vec{\mu}_i$ can no longer be asymptotically disregarded, as p is also allowed to tend to infinity at up to a linear rate in n. Let P_i denote the orthogonal projection into the orthogonal complement $\vec{\mu}_i^{\perp}$ of $\vec{\mu}_i$, and define the $(p-1)\times(p-1)$ diagonal matrix D_i as follows:

$$D_i = \begin{pmatrix} \frac{1}{\lambda_1-(\lambda_i+\Delta\lambda_i)} & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \frac{1}{\lambda_2-(\lambda_i+\Delta\lambda_i)} & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & \frac{1}{\lambda_{i-1}-(\lambda_i+\Delta\lambda_i)} & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & \frac{1}{\lambda_{i+1}-(\lambda_i+\Delta\lambda_i)} & \dots & 0 \\ \vdots & \vdots & \dots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 & \dots & \frac{1}{\lambda_p-(\lambda_i+\Delta\lambda_i)} \end{pmatrix}.$$

After removing the i-the equation from (1.3.31), multiply by D_i and solving for $\Delta\vec{\mu}_i$ so as to get:

$$\Delta\vec{\mu}_i = (\mathrm{I}_{p-1} - D_i\,E_i)^{-1}\,D_i \begin{pmatrix} E_{1i} \\ E_{2i} \\ \vdots \\ E_{(i-1)i} \\ E_{(i+1)i} \\ \vdots \\ E_{pi} \end{pmatrix} = -\frac{\sigma_i}{\sqrt{n}}(\mathrm{I}_{p-1} - D_i\,E_i)^{-1} \begin{pmatrix} \frac{\sigma_1}{\sigma_1^1-\sigma_i^2-\Delta\lambda_i}N_{1i} \\ \frac{\sigma_2}{\sigma_2^2-\sigma_i^2-\Delta\lambda_i}N_{2i} \\ \vdots \\ \frac{\sigma_{i-1}}{\sigma_{i-1}^2-\sigma_i^2-\Delta\lambda_i}N_{(i-1)i} \\ \frac{\sigma_{i+1}}{\sigma_{i+1}^2-\sigma_i^2-\Delta\lambda_i}N_{(i+1)i} \\ \vdots \\ \frac{\sigma_p}{\sigma_p^2-\sigma_i^2-\Delta\lambda_i}N_{pi} \end{pmatrix} \tag{1.3.35}$$

where E_i is the $(p-1)\times(p-1)$ matrix obtained from E by deleting the i-th row and column. Here, we commit a small language abuse, sincer in 1.3.35 the vector $\Delta\vec{\mu}_i$ is taken to be $(p-1)$-dimensional. So, if we wanted to be very precise, we should replace in 1.3.35 $\Delta\vec{\mu}_i$ by $P_i(\Delta\vec{\mu}_i)$, where P_i is the orthogonal projection along the i-th canonical basis-vectors. In other words, the vector $\Delta\vec{\mu}_i$ in 1.3.35 is obtained from the previous $\Delta\vec{\mu}_i$ by simply removing the i-entry (which is 0 anyhow). Also, N_{st} is defined as in (1.3.34).

Comparing (1.3.35) with formula (1.3.33) from the finite-dimensional case, we note the following:

Firstly (1.3.35) is an exact formula, whilst (1.3.33) is an approximation.

Secondly, instead of the term σ_i^2 in the finite dimensional formula, it is the term $\sigma_i^2 + \Delta\lambda_i$ in the denominators on the r.h.s. of (1.3.35). If we take a fixed distribution for the eigenvalues σ_j^2, and let n and p go to infinity at the same time, then the difference $\Delta\lambda_{\epsilon n}$ converges to a non-zero value for any $\epsilon \in (0, 1)$.

We are going to replace $\sigma_i^2 + \Delta\lambda_i$ by the eigenvalue among the σ_j^2's which comes the closest, that is $\sigma_{i^*}^2$. The lemma below shows that any upper bound we have for $\Delta\vec{\mu}_i$ based on formula 1.3.35, when we replace $\lambda_i^2 + \Delta\lambda_i$ by $\sigma_{i^*}^2$, we need to multiply the bound by at most a factor 2.

Lemma 1.3.1. *For all $j \neq i^*$, it is true that*

$$\frac{1}{\left|\sigma_j^2 - \sigma_i^2 - \Delta\lambda_i\right|} \leq \frac{2}{\left|\sigma_j^2 - \sigma_{i^*}^2\right|}, \tag{1.3.36}$$

where i^* is the random index j for which $|\sigma_j^2 - \sigma_i^2 - \Delta\lambda_i|$ gets minimized. So, in other words, it is the index of the σ_j^2 which comes closest to $\sigma_i^2 - \Delta\lambda_i$.

Proof. By definition of i^* we have that

$$|\sigma_{i^*}^2 - (\lambda_i + \Delta\lambda_i)| \leq |\sigma_j^2 - (\lambda_i + \Delta\lambda_i)|, \quad \forall j \neq i,$$

and

$$\frac{1}{\left|\sigma_j^2 - \sigma_i^2 - \Delta\lambda_i\right|} = \frac{\left|\sigma_j^2 - (\lambda_i + \Delta\lambda_i) - [\sigma_{i^*}^2 - (\lambda_i + \Delta\lambda_i]\right|}{\left|\sigma_j^2 - (\lambda_i + \Delta\lambda_{i^*})\right|} \times \frac{1}{\left|\sigma_j^2 - \sigma_i^2\right|} \leq \frac{2}{\left|\sigma_j^2 - \sigma_{i^*}^2\right|},$$

as claimed. □

Thirdly, and most significantly, the term D_iE appears in the r.h.s. of (1.3.35). Let E_i be the matrix obtained by deleting the i-th row and column of E. If it is possible to prove that

$|D_iE_i| \ll 1$, then by the Neumann Series Formula,

$$(\mathrm{I}_{p-1} - D_iE_i)^{-1} = \mathrm{I}_{p-1} + D_iE_i + (D_iE_i)^2 + (D_iE_i)^3 + \dots, \tag{1.3.37}$$

we can argue along the lines of the finite-dimensional case. However, instead of bounding $|D_iE_i|$, we will bound

$$\Lambda_i := \left|\, |D_i|^{0.5} E_i |D_i|^{0.5} \,\right|, \tag{1.3.38}$$

where $|D_i|$ denotes the matrix obtained by replacing the coefficients of D_i by their absolute values. Note that there exists a diagonal matrix J with diagonal coefficients ± 1, depending on the sign of the corresponding coefficient of D_i, such that

$$(D_iE_i)^k = J|D_i|^{0.5} \cdot \left(|D_i|^{0.5}E_i|D_i|^{0.5}J\right)^k \cdot |D_i|^{-0.5}J.$$

This implies that if $\Lambda_i \leq \epsilon \in (0,1)$, then the Neumann series (1.3.37) converges and

$$\left|D_iE_i + (D_iE_i)^2 + (D_iE_i)^3 + \dots\right| \leq |D_i^{0.5}| \cdot \frac{\epsilon}{1-\epsilon} \cdot |D_i^{-0.5}|. \tag{1.3.39}$$

Now, with the convergence of this Von Neumann series, we get that equation 1.3.35 can be rewritten as

$$\Delta\vec{\mu}_i = -\frac{\sigma_i}{\sqrt{n}}|D_i|^{0.5}J\left(I_{p-1} + \sum_{k=1}^{\infty}\left(|D_i|^{0.5}E_i|D_i|^{0.5}\right)^k J\right)\vec{W}_i, \tag{1.3.40}$$

where :

$$\vec{W}_i =$$

$$\Big(\frac{\sigma_1}{\sqrt{|\sigma_1^1 - \hat{\sigma}_i^2|}}N_{1i}, \dots, \frac{\sigma_{i-1}}{\sqrt{|\sigma_{i-1}^2 - \hat{\sigma}_i^2|}}N_{(i-1)i}, \frac{\sigma_{i+1}}{\sqrt{|\sigma_{i+1}^2 - \hat{\sigma}_i^2|}}N_{(i+1)i}, \dots, \frac{\sigma_p}{\sqrt{|\sigma_p^2 - \hat{\sigma}_i^2|}}N_{pi}\Big),$$

and $\hat{\sigma}_i^2 = \sigma_i + \Delta\lambda_i$.

Now, if $\Lambda_i \leq \epsilon < 1$, Equation 1.3.40 implies:

$$|\Delta\vec{\mu}_i| \leq \frac{\sigma_{i^*}}{\sqrt{n}} \frac{1}{\sqrt{\mathtt{spectral\ gap_{i^*}}}} \left(1 + \frac{\epsilon}{1-\epsilon}\right) \cdot |\vec{W}_i|. \tag{1.3.41}$$

where we used the following inequality:

$$|D_i| \leq \frac{\sqrt{2}}{\mathtt{spectral\ gap}_{i^*}}.$$

In order to have this last inequality above hold, we replace the expression $\sigma_i^2 + \Delta\lambda_i$ by the closest σ_j^2 in D_i and incur at most a factor 2 according to Lemma 1.3.1. We do the same replacement in the wector $\vec{W}_i$ and obtain a vector $\vec{V}_i$ (see 1.4.1). In Lemma 1.4.1 below, we obtain a bound for the Euclidian norm of $\vec{V}_i$, which for $\vec{W}_i$ translates into the following likely bound:

$$|\vec{W}_i| \leq C \ln(p) \sqrt{\sum_{j \neq i^*} \frac{\sigma_j^2}{|\sigma_j^2 - \sigma_{i^*}^2|}} \tag{1.3.42}$$

We can now replace $|W_{i^*}|$ in 1.3.41 by the right side of inequality 1.3.42, with condition 1.1.4, we obtain the following:

$$\boxed{|\Delta\vec{\mu}_i| \leq \epsilon \cdot \left(1 + \frac{\epsilon}{1-\epsilon}\right).} \tag{1.3.43}$$

Our main result on single eigenvector reconstruction, is that the bound 1.3.43 follows with high probability from inequality 1.1.4. This main result is stated precisely below in Theorem 1.5.3.

We have explained somewhat informally so far, how inequality 1.3.43 follows from two things: the bound on Λ_i and the bound on $|\vec{V}_i|$. Below, in Lemma 1.4.2 we will show this one more time, but more in detail. The high probability of the bound used for the norm $|\vec{V}_i|$ is proven in Lemma 1.4.1 and 1.5.1.

One more thing needed: to bound Λ_i with high probability. This is done in Lemma 1.5.2 below. For the proof of that lemma, we employ the already mentionned bound on the spectral norm of the error in covariance matrix estimation developed Koltschinksii and Klounici's recent papers [1]. Their formula is applicable to (1.3.38), as $|D_i|^{0.5}E_i|D_i|^{0.5}$ has an interpretation as covariance estimation error matrix for a multivariate Gaussian random vector with zero mean, independent coefficients, and whose j-th coefficient has standard deviation

$$\frac{\sigma_j}{\sqrt{\left|\sigma_j^2 - \sigma_i^2 - \Delta\lambda_i\right|}}, \quad (j \neq i^*). \tag{1.3.44}$$

The bound on Λ_i is thus given by (1.1.2) and with σ_j replaced by expression (1.3.44), and the requirement that this bound be smaller than $\epsilon \in (0,1)$ yields (1.1.4). To see why this is so, hold i fixed and let j vary. Then, the expression 1.3.44 decreases in value as σ_j^2 goes away from $\sigma_i^2 - \Delta\lambda_i$. This implies that the maximum of expression 1.3.44 (for fixed i) is found in the σ_j^2 closest to $\sigma_i^2 - \Delta\lambda_i$ either to the right or to the left. So, that maximum is then of order $\frac{\sigma_{i^*}}{\sqrt{\mathtt{spectral\ gap_{i^*}}}}$. Which leads to formula 1.1.4.

1.3.4 Which eigenvectors should we compare?

Note that our formula 1.3.35 has been derived for comparing the i-th eigenvector of the original covariance matrix to the i-th eigenvector of the estimated covariance matrix. We had mentionned that we would numerate the eigenvalues in decreasing order. However, we have not used this. In other words, formula 1.3.35 holds for any numeration of the eigenvalues and their corresponding eigenvectors. This is to say, that suprisingly enough, Formula 1.3.35 can be used, for comparing any pair of eigenvectors where one is from the original covariance and the second is from the estimated covariance. Thus formula 1.3.35 can be written out for comparing the i_1-th eigenvector of the original covariance matrix, with the i_2-th eigenvector of the estimated one for any pair $(i_1, i_i) \in \{1, 2, \ldots, p\}^2$. In this sense we can write $\Delta\vec{\mu}_{i_1,i_2}$, for the difference between the i_1-th eigenvector of the original covariance matrix, and the i_2-th eigenvector of the estimated covariance. Note that, with that notation, we have to replace all the i's in formula 1.3.35 by i_1, except in one place:

the eigenvalue of the eigenvector of the estimated covariance matrix to which we wish to compare the original eigenvector, should be i_2.

This means that $\hat{\sigma}_i^2 = \hat{\lambda}_i = \sigma_i^2 + \Delta\lambda_i$ has to be replace by $\hat{\sigma}_{i_2}^2 = \hat{\lambda}_{i_2}$. And this is the only place, where in formula 1.3.35 will appear! It is quite surprising that formula 1.3.35 mainly depends on i_1 and only in one place does the index i_2 appear! But, then again, formula 1.3.35 will not work unless we take the eigenvalue of the original covariance matrix, which comes closest to the eigenvalue of the estimated covariance matrix. Because, otherwise there is potentially a enormous term in the sum:

$$\sum_{j \neq i_1} \frac{1}{\sigma_j^2 - \hat{\lambda}_{i_2}} = \sum_{j \neq i_1} \frac{1}{\sigma_j^2 - \hat{\sigma}_{i_2}^2}. \tag{1.3.45}$$

that is the term for the index j for which σ_j^2 comes closest to $\hat{\lambda}_{i_2}$. By taking i_1 to be the index j for which σ_j^2 comes closest to $\hat{\sigma}_{i_2}$ the largest term in the sum 1.3.45 gets kicked out. This is what we are going to do. That is we take $i = i_2$ where i is a non-random given integer. Then for i_1 we take i^*, that is the index, by taking the index i_1 so that for the compared eigenvalues comes closest to the estimated eigenvalue, (assuming that the values of σ_j^2 are close to equidistant around $\sigma_{i_1}^2$), the smallest term in the sum 1.3.45, will be of linear order in $O(\frac{1}{\texttt{spectral gap}_{i_1}})$ and not otherwise uncontrollably large.

Without this choice of i_1, formula 1.3.35 is not useful, because $D_i E_i$ will not have a small norm. So, in what follows, i will refer to the index of the estimated eigenvalue $\hat{\sigma}_i^2$ which we consider. Then i_1 is the index of the eigenvalue of the orginal covariance matrix, which comes closest to $\hat{\sigma}_i^2$, that is

$$i_1 = i^*.$$

This means that with this new notation, our original $\Delta\vec{\mu}_i$ is equal to:

$$\Delta\vec{\mu}_i := \Delta\vec{\mu}_{i^*,i}.$$

In formula 1.3.35, the index i has to be replaced everywhere else by i^*, except for $\sigma_i^2 + \Delta\lambda_i = \hat{\sigma}_i^2$, where we keep $\hat{\sigma}_i^2$ So, the new formula 1.3.35 can be written as:

$$\Delta\vec{\mu}_i = -\frac{\sigma_{i^*}}{\sqrt{n}}(\mathrm{I}_{p-1} - D_{i^*}\, E_{i^*})^{-1} \begin{pmatrix} \frac{\sigma_1}{\sigma_1^1 - \hat{\sigma}_i^2} N_{1i^*} \\ \frac{\sigma_2}{\sigma_2^2 - \hat{\sigma}_i^2} N_{2i^*} \\ \vdots \\ \frac{\sigma_{i^*-1}}{\sigma_{i^*-1}^2 - \hat{\sigma}_i^2} N_{(i^*-1)i^*} \\ \frac{\sigma_{i^*+1}}{\sigma_{i^*+1}^2 - \hat{\sigma}_i^2} N_{(i^*+1)i^*} \\ \vdots \\ \frac{\sigma_p}{\sigma_p^2 - \hat{\sigma}_i^2} N_{pi^*} \end{pmatrix} \tag{1.3.46}$$

Again, we take $\Delta\vec{\mu}_i$ to be $(p-1)$-dimensional. Later in the formula above, we replace $\hat{\sigma}_i^2$ by $\sigma_{i^*}^2$ and incur whilst doing so at most a factor 2 in the norm, as explained in Lemma 1.3.1.

1.3.5 How our bound can be used by practitioners.

Our bound 1.1.4 is given as a probabilistic bound. Indeed in it you have the random index i^*. In real life data, due to the concentration of measure, i^* has a fluctuation which is of smaller order than its average size. By this, we mean that typically $E[i^*]$ is of larger order than $\sqrt{VAR[i^*]}$. So if the map: $i \mapsto \texttt{spectral gap}_i$ is quite regular (meaning that if you change i only microscopically, then the order of magnitude of $\texttt{spectral gap}_i$ remains about the same), then the bound given in 1.1.4 is also to be considered largely non-random, despite the i^* in it being random. This is to say that the bound's expected value is of larger order than its fluctuation, given regularity of the spectral gap function. Note that the formula on the right side of 1.1.4 only depends on the ground truth eigenvalues (that is the σ_j^2's) and on i^*. The σ_j^2's are not known at prior. But, there are recovery algorithms for the true spectrum for big data, which work much faster than the time it would take to reconstruct all the eigenvectors.

In such a situation, we don't want to simulate the data by using the ”reconstructed true spectrum” to determine approximately the value of i^*. Instead, here is what we do:

- We propose that practitioners use our bound 1.1.4 as non-random bound by simply replacing the random index i^* by i, where i is the index of the eigenvector they wish to reconstruct.

- In our simulation, when we use the bound 1.1.4 with i^* replaced by i, we always get the correct order of magnitude for the sample size needed to reconstruct the i-th eigenvector as can be seen in table 1.2 below. (We took the spectral gap: $\texttt{spectral gap}_i$ to be quite regular as a function of i. Otherwise, this might not work.)

See the result of these simulation below in table 1.2.

For these simulations we took a data set of 800 stocks and 2000 days of daily returns. We recover the spectrum of ”ground-truth” covariance matrix using our algorithm. (One can check that one gets close to the true ground covariance spectrum by re-simulating using that new-found spectrum, and checking that it produces the same spectrum in the sample covariance from the one observed in the original sample covariance from data. The reconstructed “ground-truth” spectrum used to re-simulate the data will be very regular. That is to say that microscopically it seems to tend to behave like a local renewal process. Of course, we can not be sure about the regularity of the “real ground truth” spectrum. Indeed, if we have to spectrum which are very identical macroscopically, but microscopically they are different, then we might not be able to tell. Indeed, both spectral will generate approximately the same observable spectrum in the sample covariance matrix. We believe that the true ground truth spectrum is somewhat regular however. the reason is that, in real life there are always some noises which tend to smoother things out.)

Having recovered the “ground-truth” covariance's spectrum, we use it to simulate data and check how big the sample size n needs to be in order to be able to recover the i-th eigenvalue. For using the bound 1.1.4 in practice, we leave out the logarithmic term, replace i^* by i, and put the constant C equal to 1. The new formula is then:

$$n_{\texttt{our}} = \frac{\sigma_i^2}{\texttt{spectral gap}_\texttt{i}} \sum_{j \neq i} \frac{\sigma_j^2}{|\sigma_j^2 - \sigma_i^2|}. \tag{1.3.47}$$

Then, we also calculate the bound from 1.1.3:

$$n_{1.2} = \frac{\sigma_1^2}{(\texttt{spectral gap}_\texttt{i})^2} \cdot \sum_{j \neq i} \sigma_j^2.$$

In what follows, $\vec{\mu}_i$ is the i-th unit eigenvector of the true covariance matrix and $\hat{\vec{\mu}}_i$ is the i-th unite eigenvector of the estimated covariance matrix. The eigenvector are numbered in decreasing order of their corresponding eigenvalues. Then how close these two are to each other can be seen in the value of the dot product: $\hat{\vec{\mu}}_i \cdot \vec{\mu}_i$. If that product is close to one in absolute value, then our estimate of the i-th eigenvector is good.

In the next table below (Table.1.2), in every row, $\hat{\vec{\mu}}_i \cdot \vec{\mu}_i$ is simulated for three different sample sizes, which are: $0.5 n_{\texttt{our}}$, $n_{\texttt{our}}$ and $2 n_{\texttt{our}}$. We see that in each case (depending on p and i), the value $n_{\texttt{our}}$ is indeed the right order of magnitude for where the estimated i-th unit eigenvector starts getting close to the i-th eigenvector. In our simulation, $n_{\texttt{our}}$ has been verified that it always gives the right order of magnitude of where the estimated and the ground truth i-th eigenvectors get close. We can see that in each case, the bound $n_{1.2}$ is completely off, sometime by an order of a million! In other words, for $\epsilon < 1$, requesting:

$$\frac{|E|}{\texttt{spectral gap}_\texttt{i}} < \epsilon, \tag{1.3.48}$$

replacing $|E|$ by the formula 1.1.2 of Koltchinskii and Lounici, gives a bound on n which is very much not tight, except for first eigenvector which in general is irrelevant! That is: the sample size needed to reconstruct a given eigenvector of the ground truth covariance matrix, is well approximated according to our finding by $n_{\texttt{our}}$, but not by $n_{1.2}$.

The next question is: why does the bound $n_{\texttt{our}}$ work so well with real data even though

Table 1.2: Simulation of 800 stocks data set of daily returns

p	i	$n_{\texttt{our bound}}$	$n_{\texttt{bound 1.2}}$	$\vec{\mu}_i \cdot \hat{\vec{\mu}}_i(0.5n_{our})$	$\vec{\mu}_i \cdot \hat{\vec{\mu}}_i(n_{our})$	$\vec{\mu}_i \cdot \hat{\vec{\mu}}_i(2n_{our})$
800	10	2458	$2740 \cdot 10^6$	0.56	0.78	0.93
800	5	651	$1.1 \cdot 10^6$	0.79	0.9	0.95
800	30	185000	$3.8 \cdot 10^{12}$			
200	15	131000	$147 \cdot 10^6$	0.5	0.6	0.71

we replace the random i^* by i, whilst our theoretical proof is for i^*?

We have following three reasons:

- Regularity of σ_i^2 paired with lower order of the fluctuation of i^*. By this, we mean that, if we vary i in the neighborhood $E[i^*] - const\sqrt{VAR[i^*]}, E[i^*] + const\sqrt{VAR[i^*]}$, then the order of $\texttt{spectral gap}_i$ remains about the same, for a integer constant $const > 2$, which is not too small. This insures that, in practice, the bound 1.1.4 can be treated as non-random, despite it containing the random i^*.

- Assuming that the σ_i^2's are in decreasing order $\sigma_1^2 > \sigma_2^2 > \ldots > \sigma_p^2$. The map

$$i \mapsto n_{\texttt{our}} = \frac{\sigma_i^2}{\texttt{spectral gap}_\texttt{i}} \sum_{j \neq i} \frac{\sigma_j^2}{|\sigma_j^2 - \sigma_i^2|} \tag{1.3.49}$$

  should be increasing in i. According to our experience, this condition is almost always satisfied with real life data provided the regularity of the spectral gap as a function of the index i. What properties of the spectrum make this condition be satisfied? Roughly speaking we can say that it should be satisfied when there is regularity and enough convexity of the spectrum $i \mapsto \sigma_i^2$. To have a heuristic of why this holds, note that $n_{\texttt{our}}$ can be roughly approximated as

$$i \mapsto \frac{\sigma_i^4}{(\texttt{spectral gap}_i)^2}. \tag{1.3.50}$$

  (For this we assume regularity, so that the σ_j^2's are close to being on a ladder for j close to i. Like if they would be generated locally by a renewal process). So if

$i \mapsto \sigma_i^2$ is convex enough, 1.3.50 is going to be increasing, hence 1.3.49 should also be increasing. (However when $i \mapsto \sigma_i^2$ is strictly linear, then 1.3.50 becomes decreasing and not increasing!)

- The larger eigenvalues (the only ones we want to reconstruct) get over-estimated (due to min, max property) meaning that typically $\sigma_i^2 \leq \hat{\sigma}_i^2$, for those indexes i which we want to reconstruct the corresponding eigenvector. Again, this condition is usually met in real-life data according to our experience. But it is easy, to construct counterexamples like step functions, where every step is part over and under estimated at the same time.

These three reasons above imply that a sample size larger than $n'_{\texttt{our}}$, where

$$n'_{\texttt{our}} = \frac{\log^2(p) \cdot C^2}{\epsilon^2} \cdot n_{\texttt{our}}$$

is enough to reconstruct with high probability correctly the i-th eigenvector. Here as usual $\epsilon \ll 1$ is a constant less than 1.

Let us explain a little informally why $n'_{\texttt{our}}$ is enough a sample size to reconstruct the i-th eigenvector correctly provided our three conditions hold:

Let p, i_1 and $n > n_{our}(p, i_1)$ be three non-random integers. Here i_1 will designate the index of the eigenvector which we wish to reconstruct.

Now, we assume that we are dealing with the higher part of the spectrum where the eigenvalues get over-estimated. (These are the only eigenvectors we are interested to reconstruct). This means that $i^* \geq i_1$ with high probability (Here $i^* = i^*(n, p, i_1)$).

Assuming that $i^*(n, p, i_1) \leq i_1$, then $n'_{\texttt{our}}(p, i^*) < n'_{our}(p, i_1)$, and hence $n \geq n'_{\texttt{our}}(p, i^*)$, thus the map $i \mapsto n'_{\texttt{our}}$ is increasing. According to the main result of this article that i_1-th eigenvector of the sample covariance matrix (when we take sample size n) is close to an eigenvector from the ground truth covariance matrix with index $i \geq i_1$. This argument can be repeated for any $i_2 \geq i_1$.

Indeed, since by definition, $i \mapsto \hat{\sigma}_i^2$ is decreasing, we find that $i^*(n,p,i_2) > i^*(n,p,i_1)$, hence we get $n'_{\texttt{our}}(p,i_2) < n'_{\texttt{our}}(p,i_1)$ since the function $n_{\texttt{our}}$ is assumed increasing in i. So, $n \geq n'_{\texttt{our}}(p,i_2)$. According to our main result implies that the i_2-th eigenvector of the sample covariance matrix (estimated with sample size n) is close to an eigenvector of the ground truth covariance matrix with index $i \geq i_2$. This argument can also be made for any index $i_2 \geq i_1$.

So all the eigenvectors of the sample covariance matrix (estimated with n samples), with index greater or equal to i_1 are close to an eigenvector of the true covariance matrix with an index less or equal to i_1. The eigenvector of the estimated covariance matrix are orthogonal to each other. So two of them can not be close to the same eigenvector of the true covariance matrix at the same time. Thus the only way this is possible is if each eigenvector of the sample covariance matrix with index $i \geq i_1$ is close to i-th eigenvector of the ground truth covariance matrix.

This finishes explaining why it follows from the main result of this paper, that if $n > n'_{our}(p,i_1)$, then all eigenvectors of the sample covariance matrix with index $i \geq i_1$, are closed to their respective eigenvectors of the ground-truth covariance matrix with high probability.

The above argument is an outline of a rigorous proof and not a heuristic argument. In reality, in our opinion, we do not need the $\log^2(p)$ factor which is present in the bound $n'_{\texttt{our}}$. This factor is only there to allow an easier formal proof. The problem is that the formula of Koltschinskii and Lounici 1.1.2 has been proven only for non-random σ_i^2's. But, the main part of our proof is to bound the spectral norm of

$$|D_{i^*}(0)|^{0.5} E_{i^*} |D_{i^*}(0)|^{0.5} \tag{1.3.51}$$

using the formula 1.1.2. It turns out that the matrix 1.3.51 is the error matrix of estimating a covariance where the ground truth eigenvalues σ_j^2 are replaced by

$$\frac{\sigma_j^2}{|\sigma_j^2 - \sigma_{i^*}^2|} \tag{1.3.52}$$

for $j \neq i^*$. Except that the formula 1.1.2 has been proven for non-random eigenvalues of the ground truth. Whilst expression 1.3.52 is random through i^*. To avoid this problem we replace i^* by i in 1.3.52 and in $|D_i^*(0)|^{0.5}|$ and then go on bounding

$$||D_i(0)|^{0.5} E_i |D_i(0)|^{0.5}| \tag{1.3.53}$$

with the help of formula 1.1.2 for every $i \in \{1, 2, \ldots, p\}$. In order to bound 1.3.53 for every $i \in \{1, 2, \ldots, p\}$, we need a smaller probability, and this is where the factor $\log(p)$ in the bound 1.1.4 comes from. But in practice, if we would think of i^* as non-random we would not need the $\log(p)$ factor.

The next question is why is our bound $n_{\mathtt{our}}$ not just a lower bound, but the right order of magnitude for the sample size needed to reconstruct the i-th eigenvector?

In our opinion the reason is as following: we can rewrite equation 1.3.35 as

$$\Delta\vec{\mu}_i = |D_i|^{0.5} J \cdot \left(I_{p-1} - |D_i|^{0.5} E_i |D_i|^{0.5}\right)^{-1} \cdot |D_i|^{0.5} J \begin{pmatrix} E_{1i} \\ E_{2i} \\ \vdots \\ E_{(i-1)i} \\ E_{(i+1)i} \\ \vdots \\ E_{pi} \end{pmatrix} .$$

Note that the expression on the right side of 1.1.2, according to Koltchinskii and Lounici, is not just a bound, but the actual order of magnitude of $|E|$. So, we can apply this to the matrix

$$|D_i|^{0.5}E_i|D_i|^{0.5}, \tag{1.3.54}$$

replacing σ_j^2 in formula 1.1.2 by $\frac{\sigma_j^2}{|\sigma_j^2-\sigma_i^2|}$, for $j \neq i$. This yields that, if n is below $n_{\texttt{our}}$ by a big enough constant factor, then with high probability, 1.3.54 has a norm quite above 1. Since the spectrum of 1.3.54 is going to be dense, there will be some eigenvalues of 1.3.54 which are close to 1. So, the identity minus 1.3.54 must have eigenvalues close to 0, which leads

$$\left(I_{p-1} - |D_i|^{0.5}E_i|D_i|^{0.5}\right)^{-1}$$

to have a very large spectral norm. This then ensures that expression 1.3.53 is not small and hence we can not reconstruct the i-th eigenvector. To make this a formal argument would of course require more precise calculations.

To explain what the potentially tremendous applications of our formula 1.3.47 for the order of the sample size, what are needed for reconstructionof the i-th eigenvector is:

1. When a practitioner ask you: ”What is the meaning of this principal component (eigen vector of covariance matrix) that I have computed from this large data set?” You can calculate the sample size needed for getting this eigenvector back. If he/she has used a lesser sample size, then you can answer: “no meaning since the eigenvector is messed up with the noise eigenvectors.”

2. For big data one can now calculate how to chose n and p to calculate the i-th eigenvector, so as to incur last calculation time. Indeed, when we increase p, the dimension of the matrix increases, potentially leads to more computation time. But the spectral gap is also increasing, which leads to less computation time. Thus, finding the ideal p and n can be done by our formula 1.3.47 for n_{our}.

1.4 Detailed evalutation of the perturbated eigenvectors

We define the vector:

$$\vec{V}_i = \left(\frac{\sigma_i}{\sqrt{|\sigma_1^2 - \sigma_i^2|}} N_{1i} \quad \ldots \quad \frac{\sigma_{i-1}}{\sqrt{|\sigma_{i-1}^2 - \sigma_i^2|}} N_{(i-1),i} \quad \frac{\sigma_{i+1}}{\sqrt{|\sigma_{i+1}^2 - \sigma_i^2|}} N_{(i+1),i} \quad \ldots \quad \frac{\sigma_p}{\sqrt{|\sigma_p^2 - \sigma_i^2|}} N_{pi} \right)^{\mathrm{T}}. \tag{1.4.1}$$

Lemma 1.4.1. *Assume that inequality 1.1.4 holds when we replace i^* by i. Then, there exists a universal constant $C > 0$ not depending on n or p so that:*

$$\mathrm{P}\left(\|\vec{V}_i\| \leq C \cdot \ln(p) \sqrt{\sum_{j \neq i} \frac{\sigma_j^2}{|\sigma_j^2 - \sigma_i^2|}} \right) \geq 1 - p^{-\ln(p)}.$$

Proof. Note that

$$\frac{N_{j,i}}{\sqrt{n}} = \frac{E_{ij}}{\sigma_i \sigma_j} \tag{1.4.2}$$

where we recall that E_{ij} is the ij-th entry of the matrix E. Again, E is the error matrix in estimating the covariance, hence

$$E = \hat{\Sigma} - \Sigma = \sum_{i=1}^{n} \left[\frac{\vec{Y}^{(i)}}{\sqrt{n}} \right]^T \cdot \frac{\vec{Y}^{(i)}}{\sqrt{n}} - \Sigma, \tag{1.4.3}$$

where

$$\Sigma = Diag(\sigma_1^2, \sigma_2^2, \ldots, \sigma_p^2).$$

Note that $\frac{E_{ij}}{\sigma_i \sigma_j}$ is the i, j-th entry of the matrix $\Sigma^{-0.5} E \Sigma^{-0.5}$, where

$$\Sigma^{-0.5} = Diag(\sigma_1^{-1}, \sigma_2^{-1}, \ldots, \sigma_p^{-1}).$$

Now, with the help of 1.4.3, we find

$$\Sigma^{-0.5} E \Sigma^{-0.5} = \frac{1}{n} \sum_{i=1}^{n} \left[\vec{Y}^{(i)} \Sigma^{-0.5} \right]^T \cdot \vec{Y}^{(i)} \Sigma^{-0.5} - I, \tag{1.4.4}$$

where I is the $p \times p$ identity matrix. Now, $\vec{Y}(i)\Sigma^{-0.5}$ is a vector with i.i.d. standard normal entries. Let $\Sigma_{\nu}^{0.5} = Diag(\nu_1, \nu_2, \ldots, \nu_p)$ and $\Sigma_{\nu} = Diag(\nu_1^2, \nu_2^2, \ldots, \nu_p^2)$ where, $\nu_1, \nu_2, \ldots, \nu_p$ is a sequence of positive numbers. Then, $\vec{Y}(i)\Sigma^{-0.5}\Sigma_{\nu}^{0.5}$ is a normal vector with independent entries, where the j-th entry has standard deviation ν_j. Furthermore, the random vector $\vec{Y}(i)\Sigma^{-0.5}\Sigma_{\nu}^{0.5}$ has covariance matrix Σ_{ν}. From 1.4.4, we find

$$\Sigma_{\nu}^{0.5} \Sigma^{-0.5} E \Sigma^{-0.5} \Sigma_{\nu}^{0.5} = \frac{1}{n} \sum_{i=1}^{n} \left[\vec{Y}^{(i)} \Sigma^{-0.5} \Sigma_{\nu}^{0.5} \right]^T \cdot \vec{Y}^{(i)} \Sigma^{-0.5} \Sigma_{\nu}^{0.5} - \Sigma_{\nu}. \tag{1.4.5}$$

We see that on the right side of equation 1.4.5, we have the error matrix when estimating a covariance matrix of the random vectors $\vec{Y}^{(i)}\Sigma^{-0.5}\Sigma_{\nu}^{0.5}$. These are vectors with independent normal entries where the j-th entry has standard deviation ν_j. So, we can apply the formula (2.4) of Theorem 2 of Koltschinskii and Klounici [2] for the spectral norm of that matrix. The formula given by Koltschinskii and Lounici is that

$$||\hat{\Sigma} - \Sigma||_{\infty} \leq C ||\Sigma||_{\infty} \max \left(\sqrt{\frac{r(\Sigma)}{n}}, \frac{r(\Sigma)}{n}, \sqrt{\frac{t}{n}}, \frac{t}{n} \right), \tag{1.4.6}$$

holds with probability at least e^{-t}, where

$$r(\Sigma) := \frac{\sum_{j=1}^{p} \sigma_j^2}{\max_j \sigma_j^2}$$

is the *effective rank* of Σ. We are going to apply inequality 1.4.6 to the covariance matrix Σ_{ν}. So, in 1.4.6 we replace Σ by Σ_{ν}.

Now, for $j \neq i$ take $\nu_j := \frac{\sigma_j}{\sqrt{|\sigma_j^2 - \sigma_i^2|}}$ and $\nu_i := \max_{j \neq i} \nu_j$. Note that $\max_{j \neq i} \nu_j$ is approximately $\frac{\sigma_i}{|\texttt{spectral gap}_i|}$, because ν_j decreases in both directions when j goes away from i. So then condition 1.1.4 implies that

$$\sqrt{\frac{r(\Sigma_\nu)}{n}} \leq C_3 \frac{1}{\log^2(p) \cdot ||\Sigma_\nu||_\infty}, \tag{1.4.7}$$

where $C_3 > 0$ is a universal constant. Assuming that $||\Sigma_\nu||_\infty = \max_j \nu_j \geq O(1)$. So $\frac{r(\Sigma_\nu)}{n} \leq O(\frac{1}{\ln(p)})$, and we hence assume that $\frac{r(\Sigma_\nu)}{n} < 1$, which then implies that

$$\sqrt{\frac{r(\Sigma_\nu)}{n}} > \frac{r(\Sigma_\nu)}{n}. \tag{1.4.8}$$

Let us put $t = \log^2(p) r(\Sigma_\nu)$. By inequality 1.4.7, we find:

$$\sqrt{\frac{t}{n}} = \log(p)\sqrt{\frac{r(\Sigma_\nu)}{n}} \leq O(\frac{1}{\ln(p)}). \tag{1.4.9}$$

If we assume that $\sqrt{t/n} < 1$, then $\sqrt{t/n} > t/n$. This together with 1.4.8 in 1.4.6 yields the next inequality:

$$||\hat{\Sigma}_\nu - \Sigma_\nu||_\infty \leq C||\Sigma_\nu||_\infty \sqrt{\frac{t}{n}} = C \cdot \ln(p) \cdot ||\Sigma_\nu||_\infty \sqrt{\frac{r(\Sigma_\nu)}{n}}, \tag{1.4.10}$$

which must hold with probability at least

$$1 - e^{-t} = 1 - e^{-\ln^2(p) \cdot r(\Sigma_\nu)} \geq 1 - e^{-\ln^2(p)} = 1 - p^{-\ln(p)},$$

where we used that $r(\Sigma_\nu) \geq 1$ by definition. Here $\hat{\Sigma}_\nu$ designates the estimated covariance matrix when the true covariance matrix is Σ_ν instead of Σ. For this we keep the same sample size. So, the estimation error in covariance matrix when the true covariance is Σ_ν can be written as

$$\hat{\Sigma}_\nu - \Sigma_\nu = \Sigma_\nu^{0.5} \Sigma^{-0.5} E \Sigma^{-0.5} \Sigma_\nu^{0.5}.$$

Thus we can rewrite 1.4.10 as:

$$P\left(\|\Sigma_\nu^{0.5}\Sigma^{-0.5}E\Sigma^{-0.5}\Sigma_\nu^{0.5}\| \ \leq \ \ln(p)\frac{C}{\sqrt{n}}\cdot(\max_j \nu_j)\cdot\sqrt{\sum_{j=1}^{p}\nu_j^2}\right) \ \geq \ 1-p^{-\ln(p)}, \tag{1.4.11}$$

where the norm is the spectral norm of the matrix. Now, the spectral norm of a matrix is larger equal than the Euclidian norm of any column. so, take the i-th column for example. You find then that 1.4.11 implies that

$$P\left(\nu_i\sqrt{\sum_j\left(\nu_j\frac{E_{ij}}{\sigma_i\sigma_j}\right)^2} \leq \ln(p)\frac{C}{\sqrt{n}}\cdot(\max_j \nu_j)\cdot\sqrt{\sum_{j=1}^{p}\nu_j^2}\right) \ \geq \ 1-e^{-p},$$

with the help of 1.4.2, we get:

$$P\left(\nu_i\sqrt{\sum_j(\nu_j N_{ij})^2} \ \leq \ \ln(p)C\cdot(\max_j \nu_j)\cdot\sqrt{\sum_{j=1}^{p}\nu_j^2}\right) \geq 1-e^{-p}. \tag{1.4.12}$$

Now, recall that for $j \neq i$ we have taken $\nu_j := \frac{\sigma_j}{\sqrt{|\sigma_j^2-\sigma_i^2|}}$ and $\nu_i := \max_{j\neq i}\nu_j$. Then, inside the probability ν_i and $\max_j \nu_j$ cancel each other out. Thus:

$$P\left(\sqrt{\sum_{j\neq i}(\nu_j N_{ij})^2} \ \leq \ \ln(p)C\cdot\sqrt{\sum_{j=1}^{p}\nu_j^2}\right) \geq 1-p^{-\ln(p)}$$

Plugging into the last inequality above for ν_j, we find:

$$P\left(\sqrt{\sum_{j\neq i}\left(\frac{\sigma_j N_{ij}}{\sqrt{|\sigma_j^2-\sigma_i^2|}}\right)^2} \ \leq \ \ln(p)2C\cdot\sqrt{\sum_{j\neq i}^{p}\frac{\sigma_j^2}{|\sigma_j^2-\sigma_i^2|}}\right) \geq 1-p^{-\ln(p)},$$

which can also be written as:

$$P\left(|\vec{V}_i| \leq \ln(p)2C\cdot\sqrt{\sum_{j\neq i}^{p}\frac{\sigma_j^2}{|\sigma_j^2-\sigma_i^2|}}\right) \geq 1-p^{-\ln(p)},$$

which finishes our proof.

□

So, what we want to do is to show that $|\Delta\vec{\mu}_i|$ is small, given condition 1.1.4. So far, we have explained a little informally, why when condition 1.1.4 holds, then thanks to equation 1.3.46 we get $|\Delta\vec{\mu}_i|$ to be small with high probability.

Next we are going to go through the argument one more time in a slightly more formal manner: first we introduce two events $A^{n,p}$ and $B^{n,p}$. Then, in Lemma 1.4.2 we show that the events $A^{n,p}$ and $B^{n,p}$ jointly imply that $|\Delta\vec{\mu}_i|$ is small given condition 1.1.4. From there we need then only the high probability of the events $A^{n,p}$ and $B^{n,p}$ to guaranty that $|\Delta\vec{\mu}_i|$ is small with high probability. This is then the content of Theorem 1.5.3. The high probability of $A^{n,p}$ and $B^{n,p}$ follows quite directly from our Lemma 1.4.1.

We are now ready to put all of this formally. For this we define formal events:

- Let $A^{n,p}$ be the event that the random vector related to the expression on the right side of 1.3.46 is bounded as follows:

$$\left|\begin{pmatrix} \frac{\sigma_1}{\sqrt{|\sigma_1^2-\sigma_{i^*}^2|}}N_{1i^*} \\ \frac{\sigma_2}{\sqrt{|\sigma_2^2-\sigma_{i^*}^2|}}N_{2i^*} \\ \vdots \\ \frac{\sigma_{i^*-1}}{\sqrt{|\sigma_{i^*-1}^2-\sigma_{i^*}^2|}}N_{(i^*-1)i^*} \\ \frac{\sigma_{i^*+1}}{\sqrt{|\sigma_{i^*+1}^2-\sigma_{i^*}^2|}}N_{(i^*+1)i^*} \\ \vdots \\ \frac{\sigma_p}{\sqrt{|\sigma_p^2-\sigma_{i^*}^2|}}N_{pi^*} \end{pmatrix}\right| \leq C\cdot\log(p)\sqrt{\sum_{j\neq i}\frac{\sigma_j^2}{|\sigma_j^2-\sigma_{i^*}^2|}}, \tag{1.4.13}$$

  where $C>0$ is the constant from Lemma 1.4.1.

  Note that the above expression implies that the vector in the expression on the right side of 1.3.46 satisfies the same bound but with an additional factor 2 according to

Lemma 1.3.1. In other words, the above inequality 1.4.13 also holds, if we replace everywhere $\sigma_{i^*}^2$ by $\lambda_i + \Delta\lambda_i$ and multiply the bound by a factor 2.

- Let $B^{n,p}$ be the event that the spectral norm of the random Wishart matrix

$$|D_{i^*}(0)|^{0.5} E_{i^*} |D_{i^*}(0)|^{0.5} \tag{1.4.14}$$

is bounded according to our formula in Lemma 1.4.1. Hence, B^{np} is the event that inequality 1.4.10 (as well as the inside part of 1.4.11) holds for 1.4.14, note that 1.4.14 is an estimated covariance matrix. That is, 1.4.10 holds for Σ_ν and $\hat{\Sigma}_\nu$, which is a vector with j-th entry equal to

$$\frac{\sigma_j}{\sqrt{|\sigma_j^2 - \frac{\sigma_{i^*+1}^2 + \sigma_{i^*-1}^2}{2}|}}.$$

Note that $A^{n,p}$ and $B^{n,p}$ are both depending on the parameter i used to chose the eigenvalue of the estimated covariance matrix. We do not include it into the notation of our events to not make notations too cumbersome. Our main combinatorial lemma is given next. It shows that given that the events $A^{n,p}$ and $B^{n,p}$, and that n satisfies condition 1.1.4, then $|\Delta\vec{\mu}_i|$ is going to be small.

Lemma 1.4.2. *Let $i \in \{1, 2, \ldots, p\}$. Assuming that $A^{n,p}$ and $B^{n,p}$ both hold. Let $\epsilon \in (0,1)$. Assuming also that σ_{i^*} over the spectral gap i is bigger than $2/\sqrt{3}$. Assume that the sample size n is sufficiently large so that it satisfies:*

$$2C\log^2(p)\frac{\sigma_{i^*}\cdot 2\sqrt{2}}{\epsilon\cdot\sqrt{\texttt{spectral gap}_{\texttt{i}^*}}}\sqrt{\sum_{i\neq j}\frac{\sigma_j^2}{|\sigma_j^2 - \sigma_{i^*}^2|}} \leq \sqrt{n} \tag{1.4.15}$$

where $C > 1$ *is the constant from our Lemma 1.4.1. Then, we have*

$$\frac{\sigma_{i^*}}{\sqrt{n}} \left\| \begin{pmatrix} \frac{\sigma_1}{\sigma_1^2 - \sigma_i^2 - \Delta\lambda_i} N_{1i^*} \\ \frac{\sigma_2}{\sigma_2^2 - \sigma_1^2 - \Delta\lambda_i} N_{2i^*} \\ \vdots \\ \frac{\sigma_{i^*-1}}{\sigma_{i^*-1}^2 - \sigma_i^2 - \Delta\lambda_i} N_{(i^*-1)i^*} \\ \frac{\sigma_{i^*+1}}{\sigma_{i^*+1}^2 - \sigma_i^2 - \Delta\lambda_i} N_{(i^*+1)i^*} \\ \vdots \\ \frac{\sigma_p}{\sigma_p^2 - \sigma_i^2 - \Delta\lambda_i} N_{pi^*} \end{pmatrix} \right\| \leq \epsilon \tag{1.4.16}$$

and

$$\left| \left(\sum_{k \geq 1} (D_{i^*} \cdot E_{i^*})^k \right) \cdot \frac{\sigma_{i^*}}{\sqrt{n}} \begin{pmatrix} \frac{\sigma_1}{\sigma_1^2 - \sigma_i^2 - \Delta\lambda_i} N_{1i^*} \\ \frac{\sigma_2}{\sigma_2^2 - \sigma_1^2 - \Delta\lambda_i} N_{2i^*} \\ \vdots \\ \frac{\sigma_{i^*-1}}{\sigma_{i^*-1}^2 - \sigma_i^2 - \Delta\lambda_i} N_{(i^*-1)i^*} \\ \frac{\sigma_{i^*+1}}{\sigma_{i^*+1}^2 - \sigma_i^2 - \Delta\lambda_i} N_{(i^*+1)i^*} \\ \vdots \\ \frac{\sigma_p}{\sigma_p^2 - \sigma_i^2 - \Delta\lambda_i} N_{pi^*} \end{pmatrix} \right| \leq \frac{\epsilon^2}{1 - \epsilon}. \tag{1.4.17}$$

Hence

$$|\Delta\vec{\mu}_i| \leq \epsilon + \frac{\epsilon^2}{1 - \epsilon}. \tag{1.4.18}$$

Proof.

- In order to bound the left side of 1.4.16, apply the main inequality 1.4.15 and event A^{np}. This way we find that the expression on the left side of inequality 1.4.16 is bounded by

$$C \log^2(p) \cdot \frac{2\sigma_{i^*}}{\sqrt{n} \cdot \sqrt{\texttt{spectral gap}_{i^*}}} \sqrt{\sum_{j \neq i^*} \frac{\sigma_j^2}{|\sigma_j^2 - \sigma_{i^*}^2|}}$$

which according to inequality 1.4.15 is less than ϵ. So, this finishes proving inequality 1.4.16.

- Let $\vec{N}$ denote the random vector given by

$$\vec{N} := (\sigma_1 N_{1i^*}, \sigma_2 N_{2i^*}, \dots, \sigma_{i^*-1} N_{(i^*-1)i^*}, \sigma_{(i^*+1)i^*} N_{(i^*+1)1}, \dots, \sigma_p N_{pi^*}),$$

where N_{ji} is defined in 1.3.34. Then, we have that the right side of inequality 1.4.17 can be written as:

$$\left\| |D_{i^*}|^{0.5} J \sum_{k\geq 1} \left(|D_{i^*}|^{0.5} E_{i^*} |D_{i^*}|^{0.5} J \right)^k \quad |D_{i^*}|^{0.5} \frac{\sigma_{i^*}}{\sqrt{n}} \vec{N} \right\| = \tag{1.4.19}$$

$$\left\| |D_{i^*}|^{0.5} J \sum_{k\geq 1} \left(\frac{|D_{i^*}|^{0.5}}{|D_{i^*}(0)|^{0.5}} |D_{i^*}(0)|^{0.5} E_{i^*} |D_{i^*}(0)|^{0.5} \frac{|D_{i^*}|^{0.5}}{|D_{i^*}(0)|^{0.5}} J \right)^k \quad |D_{i^*}|^{0.5} \frac{\sigma_{i^*}}{\sqrt{n}} \vec{N} \right\|, \tag{1.4.20}$$

where $\frac{|D_{i^*}|^{0.5}}{|D_{i^*}(0)|^{0.5}}$ designates the $(p-1)\times(p-1)$-diagonal matrix having as j-th entry the j-th entry of $|D_{i^*}|^{0.5}$ divided by the j-th entry of $|D_j(0)|^{0.5}$.

Now, by the event $B^{n,p}$ and using 1.3.36, we get:

$$|||D_{i^*}(0)|^{0.5} E_{i^*} |D_{i^*}(0)|^{0.5}| \leq \log(p) \cdot \frac{2C}{\sqrt{n}} \left(\frac{\sigma_{i^*}}{\sqrt{\texttt{spectral gap}_{\texttt{i}^*}}} \right) \cdot \sqrt{\sum_{j\neq i^*} \frac{\sigma_j^2}{|\sigma_j^2 - \sigma_{i^*}^2|}}. \tag{1.4.21}$$

Now due to our condition 1.4.15, we get that the right side of 1.4.21 is less than 0.5ϵ leading to

$$|||D_{i^*}(0)|^{0.5} E_{i^*} |D_{i^*}(0)|^{0.5}| \leq 0.5\epsilon. \tag{1.4.22}$$

As mentioned, we chose the eigenvector of the estimated covariance matrix first and then the eigenvector of the true covariance matrix with closest eigenvalue. This implies that, the largest entry of $\frac{|D_{i^*}|}{|D_{i^*}(0)|}$ is at most 2. (Here $\frac{|D_{i^*}|}{|D_{i^*}(0)|}$ designate the diagonal entry whose j-th entry is the absolute value of the j-th entry of D_{i^*} divided by the absolute value of the j-th entry of $D_{i^*}(0)$). Hence, the spectral norm of the matrix $\frac{|D_{i^*}|^{0.5}}{|D_{i^*}(0)|^{0.5}}$ is less or equal to $\sqrt{2}$.

Now using this with the bound 1.4.22, we finally find out that expression 1.4.20 is less or equal to

$$\left||D_{i^*}|^{0.5}\right| \cdot \sum_{k\geq 1} (\epsilon)^k \quad \cdot \frac{\sigma_{i^*}}{\sqrt{n}} \cdot \left||D_{i^*}|^{0.5}\vec{N}\right| \leq \frac{1}{\sqrt{\texttt{spectral gap}_{\texttt{i}^*}}} \cdot \frac{\epsilon}{1-\epsilon} \cdot \frac{\sigma_{i^*}}{\sqrt{n}} \cdot \left||D_{i^*}|^{0.5}\vec{N}\right|. \tag{1.4.23}$$

When the event $A^{n,p}$ holds,

$$||D_{i^*}|^{0.5}\vec{N}| \leq 2C\log(p)\sqrt{\sum_{j\neq i^*} \frac{\sigma_j^2}{|\sigma_j^2 - \sigma_{i^*}^2|}}.$$

Applying this last inequality to 1.4.23, we find out that expression 1.4.20 is less or equal to

$$\frac{C\log(p)}{\sqrt{\texttt{spectral gap}_{\texttt{i}^*}}} \cdot \frac{\epsilon}{1-\epsilon} \cdot \frac{2\sigma_{i^*}}{\sqrt{n}} \cdot \sqrt{\sum_{j\neq i^*} \frac{\sigma_j^2}{|\sigma_j^2 - \sigma_{i^*}^2|}}$$

which due to condition 1.4.15 is bounded from above by: $\frac{\epsilon^2}{1-\epsilon}$. Thus the left side of 1.4.17 is less or equal to $\frac{\epsilon^2}{1-\epsilon}$.

- We are now ready to bound $|\Delta\vec{\mu}_i|$. We use 1.4.22 together with the fact that the largest entry of $\frac{|D_{i^*}|}{|D_{i^*}(0)|}$ is at most 2, gets us

$$\left(|D_{i^*}|^{0.5} E_{i^*} |D_{i^*}|^{0.5}\right) \leq \epsilon$$

thus we have convergence of the following series:

$$(I - D_{i^*}E_{i^*})^{-1} = I + D_{i^*}E_{i^*} + (D_{i^*}E_{i^*})^2 + (D_{i^*}E_{i^*})^3 + \ldots.$$

When we apply the last equation above to 1.3.46, we obtain:

$$\Delta\vec{\mu}_i =$$

$$\frac{\sigma_{i^*}}{\sqrt{n}}\begin{pmatrix} \frac{\sigma_1}{\sigma_1^2-\sigma_i^2-\Delta\lambda_i}N_{1i^*} \\ \frac{\sigma_2}{\sigma_2^2-\sigma_1^2-\Delta\lambda_i}N_{2i^*} \\ \vdots \\ \frac{\sigma_{i^*-1}}{\sigma_{i^*-1}^2-\sigma_i^2-\Delta\lambda_i}N_{(i^*-1)i^*} \\ \frac{\sigma_{i^*+1}}{\sigma_{i^*+1}^2-\sigma_i^2-\Delta\lambda_i}N_{(i^*+1)i^*} \\ \vdots \\ \frac{\sigma_p}{\sigma_p^2-\sigma_i^2-\Delta\lambda_i}N_{pi^*} \end{pmatrix} + \left(\sum_{k\geq 1}(D_i\cdot E_{i^*})^k\right)\cdot\frac{\sigma_{i^*}}{\sqrt{n}}\begin{pmatrix} \frac{\sigma_1}{\sigma_1^2-\sigma_i^2-\Delta\lambda_i}N_{1i^*} \\ \frac{\sigma_2}{\sigma_2^2-\sigma_1^2-\Delta\lambda_i}N_{2i^*} \\ \vdots \\ \frac{\sigma_{i^*-1}}{\sigma_{i^*-1}^2-\sigma_i^2-\Delta\lambda_i}N_{(i^*-1)i^*} \\ \frac{\sigma_{i^*+1}}{\sigma_{i^*+1}^2-\sigma_i^2-\Delta\lambda_i}N_{(i^*+1)i^*} \\ \vdots \\ \frac{\sigma_p}{\sigma_p^2-\sigma_i^2-\Delta\lambda_i}N_{pi^*} \end{pmatrix}.$$

Applying inequality 1.4.16 to the first term in the last sum above and inequality 1.4.17, yields finally

$$|\Delta\vec{\mu}_i| \leq \epsilon + \frac{\epsilon^2}{1-\epsilon}$$

which finishes to prove 1.4.18

□

1.5 High probability of the events and main theorem

In the last lemma above, we have shown that the events $A^{n,p}$ and $B^{n,p}$ when condition 1.4.15, make $|\Delta\vec{\mu}_i|$ small. Now it just remains to show that $A^{n,p}$ and $B^{n,p}$ have high probability when that condition holds. That is what we will do next. We will denote by $A^{n,p,c}$ and $B^{n,p,c}$ the complement of $A^{n,p}$ and $B^{n,p}$ respectively.

Lemma 1.5.1. *Let n, p be two natural numbers. Let $i \in \{1, 2, \ldots, p\}$. Then, if condition 1.4.15 is satisfied, the event $A^{n,p}$ has high probability. More precisely, we have that*

$$P(A^{n,p,c} \cap C^{n,p,i}) \leq p^{-(\ln(p)-1)},$$

where $C^{n,p,i}$ is the event that inequality 1.4.15 holds.

Proof. Let s be an non-random integer in $[1, p]$. Let $A^{n,p,s}$ designate the event that $A^{n,p}$ holds for i^* being replaced by s, and when condition 1.4.15 is met when we replace in it i^* by s. This means that when condition 1.4.15 does not hold, with i^* replaced by s, then the event $A^{n,p,s}$ always holds. Thus $A^{n,p,s}$ is an intersection of two events.

First the event that inequality 1.4.13 holds, for i^* replaced by s. Second that inequality 1.4.15 does not hold, for i^* replaced by s. Then, according to Lemma 1.4.1, we find

$$P(A^{n,p,s}) \geq 1 - p^{\log(p)}. \tag{1.5.1}$$

But we have

$$\cap_{s=1}^{p} A^{n,p,s} \subset A^{n,p} \cup C^{n,p,i,c},$$

hence

$$P(A^{n,p,c} \cap C^{n,p,i,c}) \leq \sum_{s=1}^{p} (1 - P(A^{n,p,s}),$$

which together with 1.5.1 leads to our desired bound:

$$P(A^{n,p,c} \cap C^{n,p,i}) \leq p \cdot p^{-\log(p)}.$$

□

Lemma 1.5.2. *Let n, p be two natural numbers. Let $i \in \{1, 2, \ldots, p\}$. Then, if condition 1.4.15 is satisfied, the event $B^{n,p}$ has high probability. More precisely, we have that*

$$P(B^{n,p,c} \cap C^{n,p,i}) \leq p^{-(\ln(p)-1)},$$

where $C^{n,p,i}$ is the event that inequality 1.4.15 holds.

Proof. Similar to the proof for the event $A^{n,p}$. □

We are now ready for our Main Theorem which bounds the difference $\Delta\vec{\mu}_i$ between original and estimated eigenvector of the covariance matrix. Let us quickly remind one more time how we define $\Delta\vec{\mu}_i$:

We take the i-th eigenvalue of the estimated covariance matrix and the corresponding eigenvector. (Assuming singular eigenvalues. In principle one can take any numeration one wants of the eigenvalues of the estimated covariance matrix.) Then, we take $\vec{\mu}_{i^*}$ to be the eigenvector of the original covariance matrix, whose eigenvalue comes the closest. Finally, we take the eigenvector of the original covariance matrix to be unitary and $\Delta\vec{\mu}_i$ to be orthogonal to it. (In this manner, the eigenvector of the estimated matrix considered will typically not be exactly unitary).

With this setting, we get now our main Theorem:

Theorem 1.5.3. *Let n, p be two natural numbers and let $i \in [1, p]$. Let i^* designated the (random) index of the eigenvalue of the original covariance matrix, which is the closest one to the i-th eigenvalue of the estimated covariance matrix. When the sample size n satisfies*

$$2C\frac{\log(p)\cdot\sigma_{i^*}\cdot 2\sqrt{2}}{\epsilon\cdot\sqrt{\texttt{spectral gap}_{\texttt{i}^*}}}\sqrt{\sum_{i\neq j}\frac{\sigma_j^2}{|\sigma_j^2-\sigma_{i^*}^2|}}\leq\sqrt{n}, \tag{1.5.2}$$

the difference $\Delta\vec{\mu}_i$ between the i-th eigenvectors of the estimated covariance and the i^* eigenvector of the original covariance can be bounded with high probability. More precisely:

$$P(\left\{|\Delta\vec{\mu}_i|\geq\epsilon+\frac{\epsilon^2}{1-\epsilon}\right\}\cap C^{n,p,i})\ \leq 2p^{-(p-1)}.$$

Proof. Just apply Lemma 1.5.1 and 1.5.2 to lemma 1.4.2 □

CHAPTER 2

CONVERGENCE OF KRASULINA ESTIMATOR

2.1 Introduction

Principal component analysis (PCA) is one of the most widely used dimension reduction techniques in data analysis. Suppose $X_1, X_2, ..., X_n$ are vectors drawn i.i.d. from a distribution with mean zero and covariance Σ, where $\Sigma \in \mathbb{R}^{d\times d}$ is unknown. Let $A_n = X_n X_n^T$, then $E[A_n] = \Sigma$. We are interested in finding eigenvalues of matrix Σ and the corresponding eigenvectors if identifiable. In this Chapter, we are going to talk about the famous eigenvalue and eigenvector estimators recently, and focus on the convergence proof of Krasulina estimator[5].

2.1.1 Offline setting

This problem has been intensively studied especially in the offline setting where all the observations are available at once, see [6, 7, 8, 9, 10, 11, 12]. For instance, [8] derived the sharp minimax rate of estimation of the eigenvectors for the following Frobenius risk $E[\|\Theta\Theta^T - \hat{\Theta}\hat{\Theta}^T\|_F^2]$, where $\Theta = [\theta_1, \theta_2, ..., \theta_r]$ is the matrix of eigenvectors and $\hat{\Theta}$ is the corresponding estimator.

Recently, [1, 2, 13] derived subtle results about the behavior of the standard PCA method in an infinite-dimensional setting. They showed that:

$$E\|\hat{\Sigma} - \Sigma\| \asymp \|\Sigma\|(\sqrt{\frac{r(\Sigma)}{n}} \bigvee \frac{r(\Sigma)}{n}),$$

where

$$r(\Sigma) = \frac{(E\|X\|)^2}{\|\Sigma\|}.$$

Moreover, under the assumption that $r(\Sigma) \lesssim n$, they proved that, for all $t \geq 1$, with probability at least $1e^t$,

$$|\|\hat{\Sigma} - \Sigma\| - E\|\hat{\Sigma} - \Sigma\|| \lesssim \|\Sigma\|(\sqrt{\frac{t}{n}} \bigvee \frac{t}{n}).$$

2.1.2 Online setting

In the high dimensional setting and for massive data sets, the computational complexity of PCA may become an issue. Indeed, for data in $\mathbb{R}^d$, the default method needs storage space in the order of $O(d^2)$. Therefore, it is interesting to develop online incremental schemes that only take one data point at a time to update estimators of eigenvectors and eigenvalues. The least storage consuming methods only need $O(d)$ space to compute one eigenvector.

Assume matrix Σ has the standard decomposition:

$$\Sigma = \sum_{j=1}^{d} \lambda_j \theta_j \otimes \theta_j, \tag{2.1.1}$$

where eigenvalues λ_j's satisfy: $\lambda_1 < \lambda_2 \leq \lambda_3 \leq ... < \lambda_d$ and θ_j are the corresponding eigenvectors. We assume here that $\lambda_1 < \lambda_2$ so that θ_1 is identifiable up to sign. To compute the smallest eigenvalue and corresponding eigenvector, Krasulina[5] suggested the following stochastic gradient scheme. At time $n+1$, the estimate of the smallest eigenvector V_{n+1} is updated as follows:

$$V_{n+1} = V_n - \gamma_{n+1}\xi_{n+1}, \tag{2.1.2}$$

where $\{\gamma_n\}$ is the learning rate, typically, $\{\gamma_n\}$ is chosen such that

$$\sum \gamma_n = \infty, \sum \gamma_n^2 < \infty. \tag{2.1.3}$$

For example, $\gamma_n = \frac{c}{n}$ where c is an absolute constant. And

$$\begin{aligned}\xi_{n+1} &=< X_{n+1}, V_n > \cdot X_{n+1} - \frac{< X_{n+1}, V_n >^2}{\|V_n\|^2} \cdot V_n \\ &= A_{n+1} \cdot V_n - \frac{< A_{n+1} V_n, V_n >}{\|V_n\|^2} \cdot V_n.\end{aligned}$$

There has been a lot of effort to compute the spectrum decomposition. Oja and Karhunen[14] suggested a method which is closely related to Krasulina's, they use the update for the leading eigenvector as follows:

$$V_{n+1} = \frac{V_n + \gamma_{n+1} < X_{n+1}, V_n > X_{n+1}}{\|V_n + \gamma_{n+1} < X_{n+1}, V_n > X_{n+1}\|}. \tag{2.1.4}$$

[5, 14] proved that these estimators converge almost surely under the assumption (2.1.1), (2.1.3) and $E[\|X_n\|^k] < \infty$ for some suitable k.

There are many other incremental estimators whose convergence has not been established yet. [15] introduces a candid covariance-free incremental PCA algorithm with assumption (2.1.1), they suggest the estimator:

$$V_{n+1} = \frac{n-1-l}{n} V_{n-1} + \frac{1+l}{n} X_n X_n^T \frac{V_{n-1}}{\|V_{n-1}\|}, \tag{2.1.5}$$

where l is called the amnesic parameter. With the presence of l, larger weight is given to new samples and the effect of old samples will fade out gradually. Typically, l ranges from 2 to 4. They also addressed the estimation of additional eigenvectors by first subtracting from the data its projection on the estimated eigenvectors, then applying (2.1.5). [16] considers PCA problem as stochastic optimization problem, it considers an unknown source distribution over $\mathbb{R}^d$, and would like to find the k-dimensional subspace maximizing the variance of the distribution inside the subspace. They solve the problem by stochastic gradient descent, and suggests the updates:

$$V_{n+1} = \mathcal{P}_{orth}(V_n + \eta_n X_n X_n^T V_n),$$

where $\mathcal{P}_{orth}(V)$ performs a projection with respect to the spectral norm of VV^T onto the set of $d \times d$ matrices with k eigenvalues equal to 1 and the rest 0, η_n is the step size.

There also exist many results which analyze incremental PCA from the statistical perspective. They mainly show the asymptotic consistency of estimators under certain conditions. For example, [17] suggests a Block-Stochastic Power Method with the assumption:

$$X_n = AZ_n + E_n, \tag{2.1.6}$$

where A is a fixed matrix, Z_n is a multivariate normal random variable, i.e. $Z_n \sim N(0, I)$, and E_n is the noise vector, also sampled from multivariate normal random variable, i.e. $E_n \sim N(0, \sigma^2 I)$. For a fixed block size B, they update the estimator as:

$$V_{n+1} = \frac{\frac{1}{B}\sum_{t\in(n-B,n]} < V_n, X_t > X_t}{\|\frac{1}{B}\sum_{t\in(n-B,n]} < V_n, X_t > X_t\|}. \tag{2.1.7}$$

It proves that under (2.1.6), for any $\epsilon > 0$, estimator (2.1.7) satisfies

$$\mathbb{P}(\|V_n - \theta_1\| \leq \epsilon) = 0.99,$$

given $n = O(\frac{\log(d/\epsilon)}{\log((\sigma^2+0.75)(\sigma^2+0.5))})$ and block size $B = O(\frac{(1+3(\sigma+\sigma^2)\sqrt{d})^2}{\epsilon^2})$. [18] finds an upper bound in probability $1-\delta$ of alignment loss function $1 - \frac{<V_n,\theta_1>^2}{\|V_n\|^2}$ for Oja's estimator (2.1.4) with assumption:

$$\|A_i - \Sigma\| \leq A \text{ and } \|E[(A_i - \Sigma)^2]\| \leq B,$$

for the following choice of step size $\gamma_n = O(\frac{\alpha}{g_1(\beta+n)})$, where $\alpha > \frac{1}{2}$, $g_1 = \lambda_1 - \lambda_2$,

$\beta = O(\frac{A\alpha}{g_1} \bigvee \frac{(B+\lambda_1^2)\alpha^2}{g_1^2+\delta})$, and $n > \beta$.

As for non-asymptotic result, [19] derives sub-optimal bound on the alignment loss

$$L(V_n, \theta_1) := E\left[1 - \frac{< V_n, \theta_1 >^2}{\|V_n\|^2}\right],$$

for the following choice of the learning rate: $\gamma_n = \frac{1}{g_1 n}$, where $g_1 = \lambda_1 - \lambda_2$, provided that $n \gtrsim_p \frac{d^2}{g_1^2}$.

[20] introduces Mini-batch Power Method, for batch size B:

$$V_{n+1} = \frac{\frac{1}{B}\sum_{t\in(n-B,n]} < V_n, X_t > X_t - \beta V_{n-1}}{\|\frac{1}{B}\sum_{t\in(n-B,n]} < V_n, X_t > X_t - \beta V_{n-1}\|}, \tag{2.1.8}$$

where β is the Momentum parameter. When

$$\beta \in [\frac{\lambda_2^2}{4}, \frac{\lambda_1^2}{4}), \ \|V_0\| = 1, \text{ and } | < V_0, \theta_1 > | \geq \frac{1}{2},$$

for any $\delta \in (0,1)$ and $\epsilon \in (0,1)$, we have $\mathbb{P}\left(L(V_n, \theta_1) \leq \epsilon\right) \geq 1 - 2\delta$., with assumption: $n = \frac{\sqrt{\beta}}{\sqrt{\lambda_1^2 - 4\beta}} \log(\frac{32}{\delta\epsilon})$ and $\|E[(A_i - \Sigma)^2]\| \leq \frac{(\lambda_1^2 - 4\beta)\delta\epsilon}{256\sqrt{d}n}$.

Krasulina states the convergence of the smallest eigenvalue and eigenvector estimators, but did not provide convergence rate. In this Chapter, we find the rate of convergence for both eigenvalue and eigenvector estimators of Krasulina (2.1.2) under a relatively mild assumption. Our analysis reveals a slower rate of convergence of eigenvalue estimator $\hat{\lambda_1} = \frac{<A_n V_n, V_n>}{\|V_n\|^2}$ and corresponding eigenvector estimator $\hat{\theta_1} = \frac{V_n}{\|V_n\|}$ as compared to the offline setting for Krasulina's scheme.

Notations: for any vector $x \in \mathbb{R}^d$, we denote by $\|x\|$ the $l^2 - norm$ of x. For the sake of simplicity, for any matrix A, $\|A\|$ will refer to the operator norm of A, specifically, $\|A\| = \sup_{u,v} \frac{<Au,v>}{\|u\|\|v\|}$. For series $\{x\}_n, \{y\}_n$, $x_n \asymp_p y_n$ is defined as: $\forall \epsilon > 0$, there exists a finite $M > 0$ and a finite $N > 0$, such that $P(\frac{1}{M} < |\frac{y_n}{x_n}| < M) < 1 - \epsilon$, $\forall n > N$. $y_n \lesssim_p x_n$ is defined as: $\forall \epsilon > 0$, there exists a finite $M > 0$ and a finite $N > 0$, such that

$P(|\frac{y_n}{x_n}| < M) < 1 - \epsilon$.

2.2 Main Results

We now state our main result:

Theorem 2.2.1. *Assume* $\lambda_1 < \lambda_2$, (2.1.3) *and* $E\|A_n\|^2 < \infty$*, Set* $g = \lambda_2 - \lambda_1$*. Then the Krasulina estimator* (2.1.2) *satisfies as* $n \to \infty$*,*

$$|\hat{\lambda_1} - \lambda_1| \asymp_p \frac{\|\Sigma\|}{\sqrt{n}} \cdot (\sqrt{E[\|A_n\|^2]} \bigvee \|\Sigma\|)$$

and

$$L(V_n, \theta_1) \asymp_p \frac{\|\Sigma\|}{g\sqrt{n}} \cdot (\sqrt{E[\|A_n\|^2]} \bigvee \|\Sigma\|).$$

In Particular, if we require the X_k's to be normal random vectors, then

$$\|A_n\| = \|X_n\|^2 \overset{d}{=} \sum_{j=1}^{d} \lambda_j Z_j^2,$$

where $Z_j \overset{i.i.d.}{\sim} N(0,1)$. Consequently, we get

$$E[\|A_n\|^2] = E[\sum_{j=1}^{d} \lambda_j^2 Z_j^4 + 2\sum_{i\neq j} \lambda_i \lambda_j Z_i^2 Z_j^2] = 2tr(\Sigma^2) + tr(\Sigma)^2 \lesssim_p tr(\Sigma)^2.$$

Thus we have following corollary:

Corollary 2.2.2. *Let the Assumptions of Theorem 2.2.1 be satisfied. Assume in addition that* $\{X_k\}$ *are i.i.d. zero mean normal random vectors with covariance matrix* Σ*. We have for the Krasulina scheme* (2.1.2) *as* $n \to \infty$ *that*

$$|\hat{\lambda_1} - \lambda_1| \asymp_p \frac{\|\Sigma\| tr(\Sigma)}{\sqrt{n}},$$

and

$$L(V_n, \theta_1) \asymp_p \frac{\|\Sigma\| tr(\Sigma)}{g\sqrt{n}}.$$

2.3 Proof of the Theorem

We first state a basic result in probability that will be used throughout the paper.

Lemma 2.3.1. *Let $\{Y_n\}_n$ be a sequence of real-valued random variable. We assume that for all $n \geq 1$, Y_n is zero mean and square integrable. Define $S_n = \sum_{k=1}^{n} Y_k$. If $\sum_{n\geq 1} E[Y_n^2] < \infty$, then $\{S_n\}_n$ converges to a real-valued random variable in probability.*

Proof. By definition, $S_n = \sum_{k=1}^{n} Y_k$, since Y_n is square integrable:

$$E[|S_{n+r} - S_n|^2] = E[(\sum_{i=n+1}^{n+r} Y_i)^2] = \sum_{i=n+1}^{n+r} E[Y_i^2] + \sum_{n+1\leq i<j\leq n+r} 2E[Y_i \cdot Y_j]. \qquad (2.3.1)$$

Since Y_n is zero mean, then for $i < j$:

$$E[Y_i \cdot Y_j] = E[E[Y_i \cdot Y_j | \mathcal{F}_i]] = E[Y_i \cdot E[Y_j | \mathcal{F}_i]] = 0,$$

plug it into (2.3.1), we obtain:

$$E[|S_{n+r} - S_n|^2] = \sum_{i=n+1}^{n+r} E[Y_i^2] \leq \sum_{i>n} E[Y_i^2],$$

this is the remainder term of a convergence series, thus $\{S_n\}_n$ is Cauchy, so $\{S_n\}_n$ converges to a real-valued random variable in $\mathcal{L}^2$. By Kolmogorov inequality, Lemma 2.3.1 follows. □

Now, we start by bounding the asymptotic expectation of $\|V_n\|^2$:

Lemma 2.3.2. $\lim_{n\to\infty} E\|V_n\|^2 < \infty$.

Proof. First, we prove that V_n and ξ_{n+1} are orthogonal for any $n \geq 1$.

Let $W_n = X_{n+1} - \frac{<X_{n+1},V_n>}{\|V_n\|^2} \cdot V_n$, we have:

$$\begin{aligned} \xi_{n+1} &= < X_{n+1}, V_n > \cdot X_{n+1} - \frac{< X_{n+1}, V_n >^2}{\|V_n\|^2} \cdot V_n \\ &= < X_{n+1}, V_n > (X_{n+1} - \frac{< X_{n+1}, V_n >}{\|V_n\|^2} \cdot V_n) \\ &= < X_{n+1}, V_n > \cdot W_n. \end{aligned}$$

We note that $< W_n, V_n >= 0$, so

$$\|\xi_{n+1}\| =< X_{n+1}, V_n > \cdot \|W_n\| \leq< X_{n+1}, V_n > \cdot \|X_{n+1}\| \leq \|X_{n+1}\|^2 \|V_n\|,$$

thus:

$$E[\|\xi_{n+1}\| | \mathcal{F}_n] \leq E[\|X_{n+1}\|^2] \cdot \|V_n\| = tr(\Sigma)\|V_n\|. \tag{2.3.2}$$

Now since $\xi_n \perp V_{n-1}$, we have

$$\|V_n\|^2 = \|V_{n-1} - \gamma_n \xi_n\|^2 = \|V_{n-1}\|^2 + \gamma_n^2 \|\xi_n\|^2,$$

thus:

$$\begin{aligned} E[\|V_n\|^2 | \mathcal{F}_{n-1}] &= \|V_{n-1}\|^2 + \gamma_n^2 E[\|\xi_n\|^2 | \mathcal{F}_{n-1}] \\ &\leq \|V_{n-1}\|^2 + \gamma_n^2 tr(\Sigma)^2 \|V_{n-1}\|^2 \\ &= (1 + \gamma_n^2 tr(\Sigma)^2) \|V_{n-1}\|^2 \end{aligned}$$

Thus:

$$\begin{aligned} E\|V_n\|^2 &\leq (1 + \gamma_n^2 tr(\Sigma)^2) E\|V_{n-1}\|^2 \\ &\leq ... \leq \prod_{i=2}^{n} (1 + \gamma_i^2 tr(\Sigma)^2) \cdot E\|V_1\|^2 \end{aligned}$$

By assumption (2.1.3), we have $\sum_{i=1}^{\infty}\gamma_i^2 tr(\Sigma)^2 < \infty$, thus: $\prod_{i=1}^{n-1}(1+\gamma_i^2 tr(\Sigma)^2) < \infty$, thus $\lim_{n\to\infty} E\|V_n\|^2 < \infty$. □

Next, let $\mu(V_n) = \frac{<\Sigma V_n, V_n>}{\|V_n\|^2}$, and $a_1^{(n)} =< V_n, \theta_1 >$. We first prove the convergence in probability of the sequence of V_n and $a_1^{(n)}$. Specifically, $\mu(V_n)$ converges to λ_1, and V_n converges to a vector which is alined with θ_1. To prove that, we can recursively properly apply the inequality, to show the Cauchy property of sequence $\mu(V_n)$ and $a_1^{(n)}$.

Lemma 2.3.3. *$\mu(V_n) = \frac{<\Sigma V_n, V_n>}{\|V_n\|^2}$ converges a.s. to μ as $n \to \infty$.*

Proof.

$$\begin{aligned}
\mu(V_{n+1}) &= \frac{< \Sigma V_n - \gamma_{n+1}\cdot\Sigma\xi_{n+1}, V_n - \gamma_{n+1}\xi_{n+1} >}{\|V_n - \gamma_{n+1}\xi_{n+1}\|^2} \\
&= \frac{< \Sigma V_n, V_n > +\gamma_{n+1}^2 < \Sigma\xi_{n+1}, \xi_{n+1} > -2\gamma_{n+1} < \xi_{n+1}, \Sigma V_n >}{\|V_n\|^2 + \gamma_{n+1}^2\|\xi_{n+1}\|^2} \\
&= \frac{1}{1+\gamma_{n+1}^2\frac{\|\xi_{n+1}\|^2}{\|V_n\|^2}}(\mu(V_n) - 2\gamma_{n+1}\frac{< \xi_{n+1}, \Sigma V_n >}{\|V_n\|^2} \\
&\quad +\gamma_{n+1}^2\frac{< \Sigma\xi_{n+1}, \xi_{n+1} >}{\|V_n\|^2})
\end{aligned}$$

Since:

$$\begin{aligned}
< \xi_{n+1}, \Sigma V_n > &= < A_{n+1}V_n, \Sigma V_n > - \frac{< A_{n+1}V_n, V_n >< \Sigma V_n, V_n >}{\|V_n\|^2} \\
&= \|\Sigma V_n\|^2 - \frac{< \Sigma V_n, V_n >^2}{\|V_n\|^2} + < A_{n+1}V_n, \Sigma V_n > - \|\Sigma V_n\|^2 \\
&\quad - \frac{< A_{n+1}V_n, V_n >< \Sigma V_n, V_n >}{\|V_n\|^2} + \frac{< \Sigma V_n, V_n >^2}{\|V_n\|^2} \\
&= (< (A_{n+1}-\Sigma)V_n, \Sigma V_n > - \frac{< (A_{n+1}-\Sigma)V_n, V_n >}{\|V_n\|^2} \\
&\quad \cdot < \Sigma V_n, V_n >) + (\|\Sigma V_n\|^2 - \frac{< \Sigma V_n, V_n >^2}{\|V_n\|^2})
\end{aligned}$$

Let

$$f(V_n) = \frac{\|\Sigma V_n\|^2}{\|V_n\|^2} - \frac{< \Sigma V_n, V_n >^2}{\|V_n\|^4}, \tag{2.3.3}$$

$$Z_n = \frac{< (A_{n+1} - \Sigma)V_n, \Sigma V_n >}{\|V_n\|^2} - \frac{< (A_{n+1} - \Sigma)V_n, V_n >}{\|V_n\|^4} \cdot < \Sigma V_n, V_n >, \qquad (2.3.4)$$

thus: $\frac{<\xi_{n+1}, \Sigma V_n>}{\|V_n\|^2} = f(V_n) + Z_n$.

so $\mu(V_{n+1}) = \frac{1}{1+\gamma_{n+1}^2 \frac{\|\xi_{n+1}\|^2}{\|V_n\|^2}} (\mu(V_n) - 2\gamma_{n+1} f(V_n) - 2\gamma_{n+1} Z_n + \gamma_{n+1}^2 \frac{<\Sigma\xi_{n+1}, \xi_{n+1}>}{\|V_n\|^2})$.

Let

$$a_n = \gamma_{n+1} Z_n, b_n = \gamma_{n+1}^2 \frac{< \Sigma\xi_{n+1}, \xi_{n+1} >}{\|V_n\|^2}, c_n = \frac{1}{1 + \gamma_{n+1}^2 \frac{\|\xi_{n+1}\|^2}{\|V_n\|^2}}, \qquad (2.3.5)$$

thus:

$$\mu(V_{n+1}) = c_n \cdot (\mu(V_n) - 2\gamma_{n+1} f(V_n) - 2a_n + b_n).$$

Now we have:

$$\mu(V_{n+1}) - c_n \cdot \mu(V_n) = -2\gamma_{n+1} c_n f(V_n) - 2a_n c_n + b_n c_n. \qquad (2.3.6)$$

For series $\{a_n\}$, since Z_n is centered and $E[Z_n^2]$ is bounded, by lemma 2.3.1:

$$\sum_{i>k} Var(a_i) \asymp_p \sum_{i>k} \gamma_i^2 < \infty,$$

thus $\sum_{n=1}^{\infty} a_n < \infty$.

For series $\{b_n\}$, by (2.3.2):

$$E[\|\xi_n\| | \mathcal{F}_{n-1}] \leq tr(\Sigma) \|V_n\|,$$

thus

$$E[b_n | \mathcal{F}_n] = \gamma_{n+1}^2 E[\frac{< \Sigma\xi_{n+1}, \xi_{n+1} >}{\|V_n\|^2} | \mathcal{F}_n] \leq \gamma_{n+1}^2 \|\Sigma\| tr(\Sigma)^2.$$

By (2.3.2), we have $\sum_{n=1}^{\infty} b_n < \infty$.

For series $\{c_n\}$, $\prod c_n = \prod \frac{1}{1+\gamma_{n+1}^2 \frac{\|\xi_{n+1}\|^2}{\|V_n\|^2}}$ converges when $\prod 1 + \gamma_{n+1}^2 \frac{\|\xi_{n+1}\|^2}{\|V_n\|^2}$ converges. $\prod 1 + \gamma_{n+1}^2 \frac{\|\xi_{n+1}\|^2}{\|V_n\|^2}$ has the same convergence properties as $\sum \gamma_{n+1}^2 \frac{\|\xi_{n+1}\|^2}{\|V_n\|^2}$. By (2.3.2),

$$E[\frac{\|\xi_{n+1}\|^2}{\|V_n\|^2}|\mathcal{F}_n] \leq tr(\Sigma)^2,$$

we have $\prod_{n=1}^{\infty} c_n < \infty$.

And by Cauchy-Schwartz inequality:

$$f(V_n) = \frac{\|\Sigma V_n\|^2}{\|V_n\|^2} - \frac{< \Sigma V_n, V_n >^2}{\|V_n\|^4} \geq 0. \tag{2.3.7}$$

Now, if $\liminf \mu(V_n) < \limsup \mu(V_n)$, choose a, b such that $\liminf \mu(V_n) < a < b < \limsup \mu(V_n)$, find m_1, n_1 large enough, such that $\mu(V_{n_1}) < a, \mu(V_{m_1}) > b$, and for all $n_1 < j < m_1$, we have $a \leq \mu(V_j) \leq b$. Thus:

$$\mu(V_{m_1}) - \mu(V_{n_1}) \prod_{i=n_1}^{m_1-1} c_i > b - a.$$

On the other hand:

$$\begin{aligned} \mu(V_{m_1}) - \mu(V_{n_1}) \prod_{i=n_1}^{m_1-1} c_i &= \sum_{j=n_1}^{m_1-1} [(-2\gamma_{j+1} \cdot f(V_j) - 2a_j + b_j) \cdot \prod_{i=j}^{m_1-1} c_j] \quad (2.3.8) \\ &\leq \sum_{j=n_1}^{m_1-1} [(-2a_j + b_j) \cdot \prod_{i=j}^{m_1-1} c_j] \\ &\to 0 \text{ as } n_1, m_1 \to \infty, \end{aligned}$$

which is a contradiction, thus $\mu(V_n) \to \mu$ with probability 1. □

Lemma 2.3.4. *$a_1^{(n)} =< V_n, \theta_1 >$, where θ_1 is the eigenvector of λ_1, $a_1^{(n)}$ converges to some value a_1 with probability 1 as $n \to \infty$.*

Proof. Since $V_{n+1} = V_n - \gamma_{n+1}\xi_{n+1}$, $\xi_{n+1} = A_{n+1}V_n - \frac{<A_{n+1}V_n, V_n>}{\|V_n\|^2} V_n$, by definition of $a_1^{(n)} =< V_n, \theta_1 >$ and $\mu(V_n) = \frac{<\Sigma V_n, V_n>}{\|V_n\|^2}$, also by the nature: $< \Sigma V_n, \theta_1 >=<$

$V_n, \Sigma\theta_1 >=< V_n, \lambda_1\theta_1 >= \lambda_1 a_1^{(n)}$, we have:

$$\begin{aligned}
a_1^{(n+1)} &= < V_{n+1}, \theta_1 > = < V_n - \gamma_{n+1}\xi_{n+1}, \theta_1 > \\
&= < V_n, \theta_1 > -\gamma_{n+1} < A_{n+1}V_n - \frac{< A_{n+1}V_n, V_n >}{\|V_n\|^2} V_n, \theta_1 > \\
&= a_1^{(n)} + \gamma_{n+1} < \frac{< \Sigma V_n, V_n >}{\|V_n\|^2} V_n + \frac{< (A_{n+1} - \Sigma)V_n, V_n >}{\|V_n\|^2} V_n - \Sigma V_n \\
&\quad +(\Sigma - A_{n+1})V_n, \theta_1 > \\
&= a_1^{(n)} + \gamma_{n+1}(\mu(V_n) - \lambda_1)a_1^{(n)} + \gamma_{n+1}Z'_n \\
&= a_1^{(n)}(1 + \gamma_{n+1}(\mu(V_n) - \lambda_1)) + \gamma_{n+1}Z'_n,
\end{aligned}$$

where $Z'_n =< (\Sigma - A_{n+1})V_n, \theta_1 > + \frac{<(A_{n+1}-\Sigma)V_n, V_n>}{\|V_n\|^2} a_1^{(n)}$.

Since $E[\|V_n\|^2] = E[\|V_{n-1}\|^2] + \gamma_n^2 E[\|\xi_n\|^2] \leq E[\|V_{n_1}\|^2] + \gamma_n^2\|\Sigma\|^2 E\|V_{n-1}\|^2 \leq \prod_{n=1}^{\infty}(1+\gamma_n^2\|\Sigma\|^2) \leq \infty$, Z'_n is centered and $E[Z_n'^2]$ is bounded, by lemma 2.3.1, $\sum_{n=1}^{\infty} \gamma_n Z'_n < \infty$.

Now, if $\liminf a_1^{(n)} < \limsup a_1^{(n)}$, choose a, b such that $\liminf a_1^{(n)} < a < b < \limsup a_1^{(n)}$, find m_1, n_1, such that: $m_1 \geq n_1 \geq N$, $a_1^{(m_1)} < a$, $a_1^{(n_1)} > b$, for $j \in (n_1, m_1)$, $a \leq a_1^{(j)} \leq b$. Since λ_1 is the smallest eigenvalue, $\mu(V_k) \geq \lambda_1$.

Thus:

$$a_1^{(m_1)} - a_1^{(n_1)} \prod_{k=n_1}^{m_1} (1 + \gamma_{k+1}(\mu(V_k) - \lambda_1)) \leq a_1^{(m_1)} - a_1^{(n_1)} < a - b \leq 0.$$

On the other hand:

$$\begin{aligned} & a_1^{(m_1)} - a_1^{(n_1)} \prod_{k=n_1}^{m_1} (1 + \gamma_{k+1}(\mu(V_k) - \lambda_1)) \\ = & \sum_{j=n_1}^{m_1-1} \gamma_j Z'_j \prod_{i=j}^{m-1} (1 + \gamma_{j+1}(\mu(V_j) - \lambda_1)) \\ \geq & \sum_{j=n_1}^{m_1-1} \gamma_j Z'_j \end{aligned} \tag{2.3.9}$$

Since $\sum_{j=1}^{\infty} \gamma_j Z'_j < \infty$, let $n_1 \to \infty$, we can let $\sum_{j=n_1}^{m_1-1} \gamma_j Z'_j$ as closed to 0 as we want, which is a contradiction.

Thus $a_1^{(n)} \to a_1$ with probability 1. □

Now we get the idea that $\mu(V_n)$ and $a_1^{(n)}$ are both convergence with probability 1, and by the proof above, all coefficients in (2.3.6) are convergence with probability 1, so does the part $\gamma_{n+1} c_n f(V_n)$. By find the convergence rate for each of these parts, we can find the convergence rate for $\mu(V_n)$.

Lemma 2.3.5. *(1) $\mu(V_n) \to \lambda_1$ as $n \to \infty$ with probability 1, and (2) the convergence rate of $\frac{<A_n V_n, V_n>}{\|V_n\|^2}$ to λ_1 is in the order of $O(\frac{\|\Sigma\|}{\sqrt{n}} \cdot (\sqrt{E[\|A_n\|^2]}) \bigvee \|\Sigma\|)$.*

Proof. (1)

$$\begin{aligned} a_1^{(n+1)} &= \, < V_{n+1}, \theta_1 > = \, < V_{n+1}, \theta_1 > = \, < V_n - \gamma_{n+1}\xi_{n+1}, \theta_1 > \\ &= \, < V_n, \theta_1 > - \gamma_{n+1} < A_{n+1}V_n - \frac{< A_{n+1}V_n, V_n >}{\|V_n\|^2}, \theta_1 > \\ &= a_1^{(n)} + \gamma_{n+1} \frac{< A_{n+1}V_n, V_n >}{\|V_n\|^2} a_1^{(n)} - \gamma_{n+1} < A_{n+1}V_n, \theta_1 > \\ &= a_1^{(n)} + \gamma_{n+1} \frac{< \Sigma V_n, V_n >}{\|V_n\|^2} a_1^{(n)} - \gamma_{n+1} < V_n, \Sigma\theta_1 > \\ &\quad + \gamma_{n+1} \frac{< A_{n+1}V_n, V_n >}{\|V_n\|^2} a_1^{(n)} - \gamma_{n+1} < A_{n+1}V_n, \theta_1 > \\ &\quad - \gamma_{n+1} \frac{< \Sigma V_n, V_n >}{\|V_n\|^2} a_1^{(n)} + \gamma_{n+1} < V_n, \Sigma\theta_1 > \\ &= a_1^{(n)}(1 + \gamma_{n+1}(\mu(V_n) - \lambda_1)) + \gamma_{n+1} Z'_n, \end{aligned}$$

where $Z'_n =< (\Sigma - A_{n+1})V_n, \theta_1 > + \frac{<(A_{n+1}-\Sigma)V_n, V_n>}{\|V_n\|^2} a_1^{(n)}$, which is centered and bounded, then by Jensen's inequality:

$$\begin{aligned} E|a_1^{(n+1)}| &\geq E|a_1^{(n)}|(1 + \gamma_{n+1}(\frac{E[\mu(V_n)|a_1^{(n)}|]}{E|a_1^{(n)}|} - \lambda_1)) \\ &\geq \prod_{k=1}^{n}(1 + \gamma_{k+1}(\frac{E[\mu(V_k)|a_1^{(k)}|]}{E|a_1^{(k)}|} - \lambda_1))E|a_1^{(1)}| \end{aligned}$$

By Lemma 2.3.4, $\{a_1^{(n)}\}$ convergence, then

$$\prod_{k=1}^{\infty}(1 + \gamma_{k+1}(\frac{E[\mu(V_k)|a_1^{(k)}|]}{E|a_1^{(k)}|} - \lambda_1)) < \infty,$$

thus:

$$\sum_{k=1}^{\infty}\gamma_{k+1}(\frac{E[\mu(V_k)|a_1^{(k)}|]}{E|a_1^{(k)}|} - \lambda_1) < \infty.$$

By (2.1.3), $\lim_{k\to\infty} \frac{E[\mu(V_k)|a_1^{(k)}|]}{E|a_1^{(k)}|} - \lambda_1 = 0$.

By dominant convergence theorem: $\lim_{k\to\infty} a_1^{(k)} = a_1$, $\lim_{k\to\infty} \mu(V_k) = \mu$. Thus: $\frac{\mu a_1}{a_1} = \lambda_1$, therefore, $\mu = \lambda_1$.

(2)

$$\begin{aligned} \lambda_1 - \frac{< A_n V_n, V_n >}{\|V_n\|^2} &= (\lambda_1 - \mu(V_n)) + (\mu(V_n) - \frac{< A_n V_n, V_n >}{\|V_n\|^2}) \\ &= (\lambda_1 - \mu(V_n)) + (\frac{< (\Sigma - A_n)V_n, V_n >}{\|V_n\|^2}) \end{aligned}$$

Since $E[\frac{<(\Sigma-A_n)V_n,V_n>}{\|V_n\|^2}] = 0$, we only need to consider $|\lambda_1 - \mu(V_n)|$. From (2.3.6) we have:

$$\mu(V_{n+1}) - c_n \cdot \mu(V_n) = -2\gamma_{n+1}c_n f(V_n) - 2a_n c_n + b_n c_n = (-2\gamma_{n+1}f(V_n) - 2a_n + b_n)c_n,$$

where a_j, b_j and c_j are defined the same as (2.3.5). The same way as we get (2.3.8), keep increase V_{n+1} to V_m recursively, we have:

$$\mu(V_m) - \mu(V_n) \prod_{i=n}^{m-1} c_i = \sum_{j=n}^{m-1} (b_j - 2\gamma_{j+1} f(V_j) - 2a_j) \prod_{i=j}^{m-1} c_i.$$

Now, by (2.3.2): $E[\|\xi_n\| | \mathcal{F}_{n-1}] \leq tr(\Sigma)\|V_n\|$.

For b_j part,

$$\begin{aligned} \sum_{j=n}^{\infty} E[b_j|\mathcal{F}_j] &= \sum_{j=n}^{\infty} \gamma_{j+1}^2 E[\frac{< \Sigma\xi_{j+1}, \xi_{j+1} >}{\|V_j\|^2} |\mathcal{F}_j] \leq \sum_{j=n}^{\infty} \gamma_{j+1}^2 \frac{\|\Sigma\| E[\|\xi_{j+1}\|^2|\mathcal{F}_j]}{\|V_j\|^2} \\ &\leq \sum_{j=n}^{\infty} \gamma_{j+1}^2 \frac{\|\Sigma\| tr(\Sigma)^2 \|V_j\|^2}{\|V_j\|^2} = \sum_{j=n}^{\infty} \gamma_{j+1}^2 \cdot c, \end{aligned}$$

thus its rate of convergence is $O(\frac{1}{n})$

For a_j part, $\sum_{j=n}^{\infty} a_j = \sum_{j=n}^{\infty} \gamma_{j+1} Z_j$, Z_j is centered and $E[Z_j^2]$ is bounded, by lemma 2.3.1, $E[|S - S_n|^2] \leq \sum_{i>n} E[a_i^2]$, whose rate of convergence is $O(\frac{1}{n})$, thus $\sum_{j=n}^{\infty} a_j$ has the rate of convergence $O(\frac{1}{\sqrt{n}})$.

For c_j part, by proof of the lemma 2.3.3, $\prod_{i=n}^{\infty} c_i$ has the same convergence properties as $\sum_{i=n}^{\infty} \gamma_{i+1}^2 \frac{\|\xi_{i+1}\|^2}{\|V_i\|^2}$. By (2.3.2):

$$E[\frac{\|\xi_{i+1}\|^2}{\|V_i\|^2} |\mathcal{F}_i] \leq E[\frac{tr(\Sigma)^2 \|V_i\|^2}{\|V_i\|^2}] = tr(\Sigma)^2,$$

thus $\prod_{i=n}^{\infty} c_i$ has the rate of convergence $O(\frac{1}{n})$.

For $f(V_j)$ part, by assumption 2, rewrite $V_n = \sum_{i=1}^{d} a_i^{(n)} \theta_i$, where d is the dimension. From (2.3.8), we have: $\sum_{n=1}^{\infty} \gamma_{n+1} f(V_n) \prod_{k=1}^{n-1} (1 + \gamma_{k+1}^2 \frac{\|\xi_{k+1}\|^2}{\|V_k\|^2})^{-1} < \infty$ with probability 1. Since we have $\gamma_n \asymp_p \frac{1}{n}$ and $f(V_n) \geq 0 \; \forall n$, if $\liminf_{n\to\infty} f(V_n) = c$, then $\sum_{n=1}^{\infty} \gamma_{n+1} f(V_n) \prod_{k=1}^{n-1} (1 + \gamma_{k+1}^2 \frac{\|\xi_{k+1}\|^2}{\|V_k\|^2})^{-1} = \infty$, thus $c = 0$.

Now, by nature of eigenvector and eigenvalue, as well as assumption 2: $\theta_i^2 = 1, \theta_i\theta_j = 0$ for $i \neq j$, and $\|V_n\|^2 = \sum_{i=1}^{d} (a_i^{(n)})^2$.

Thus:

$$\begin{aligned} f(V_n) &= \frac{\|\Sigma V_n\|^2}{\|V_n\|^2} - \frac{< \Sigma V_n, V_n >^2}{\|V_n\|^4} \\ &= \frac{(\sum_{i=1}^d a_i^{(n)} \lambda_i \theta_i)^2}{\|V_n\|^2} - \mu(V_n)^2 \\ &= \frac{\sum_{i=1}^d (a_i^{(n)})^2 (\lambda_i^2 - \mu(V_n)^2)}{\|V_n\|^2}, \end{aligned} \tag{2.3.10}$$

which leads to the result: $f(V_j) \to 0$ with the same rate of $\mu(V_n) \to \lambda_1$.

Thus, $\frac{<A_n V_n, V_n>}{\|V_n\|^2}$ converges to λ_1 the same rate as a_j part, has the rate of convergence $O(\frac{1}{\sqrt{n}})$. More precisely, by proof of the Lemma 2.3.1, $E[|S_{n+r} - S_n|^2] \leq \sum_{i>n} E[X_i^2]$ if $\{X_n\}_n$ is 0 mean. Then for $a_j = \gamma_{j+1} Z_j$, we have

$$E[|S - S_n|^2] \leq \sum_{i>n} E[a_i^2] \lesssim_p \sum_{i>n} \frac{1}{i^2} E[Z_i^2].$$

Now for Z_n, by (2.3.4), we have:

$$\begin{aligned} \|Z_n\| &= \|\frac{< (A_{n+1} - \Sigma) V_n, \Sigma V_n >}{\|V_n\|^2} - \frac{< (A_{n+1} - \Sigma) V_n, V_n >}{\|V_n\|^4} \cdot < \Sigma V_n, V_n > \| \\ &\leq \|\frac{< (A_{n+1} - \Sigma) V_n, \Sigma V_n >}{\|V_n\|^2}\| + \|\frac{< (A_{n+1} - \Sigma) V_n, V_n >}{\|V_n\|^4} \cdot < \Sigma V_n, V_n > \| \\ &\leq \|\frac{< (A_{n+1} - \Sigma) V_n, V_n >}{\|V_n\|^2}\| \cdot \|\Sigma\| + \|\frac{< (A_{n+1} - \Sigma) V_n, V_n >}{\|V_n\|^2} \\ &\quad \cdot \frac{< \Sigma V_n, V_n >}{\|V_n\|^2}\| \\ &\lesssim_p \|\frac{< (A_{n+1} - \Sigma) V_n, V_n >}{\|V_n\|^2}\| \cdot \|\Sigma\| \\ &\leq \|A_{n+1} - \Sigma\| \|\Sigma\| \\ &\leq (\|A_{n+1}\| + \|\Sigma\|) \|\Sigma\|. \end{aligned}$$

Thus:

$$\begin{aligned} E[Z_n^2] &\leq \|\Sigma\|^2 E[\|A_{n+1}\|^2 + \|\Sigma\|^2 + 2\|A_{n+1}\|\|\Sigma\|] \\ &\lesssim_p \|\Sigma\|^2 E[\|A_{n+1}\|^2 + \|\Sigma\|^2] \\ &\asymp_p \|\Sigma\|^2 \cdot (E[\|A_n\|^2] \bigvee \|\Sigma\|^2). \end{aligned}$$

So $E[|S - S_n|^2]$ has rate of convergence $O(\frac{1}{n} \cdot \|\Sigma\|^2 \cdot (E[\|A_n\|^2] \bigvee \|\Sigma\|^2))$, thus $\sum_{j=n}^{\infty} a_j$ has rate of convergence $O(\frac{\|\Sigma\|}{\sqrt{n}} \cdot (\sqrt{E[\|A_n\|^2]} \bigvee \|\Sigma\|))$.

□

Lemma 2.3.6. *(1) $V_n \to a_1^{(n)}\theta_1$ with probability 1 and (2) $\frac{<V_n, \theta_1>^2}{\|V_n\|^2}$ approach to 1 in the order of $\frac{d\|\Sigma\|}{g\sqrt{n}} \cdot (\sqrt{E[\|A_n\|^2]} \bigvee \|\Sigma\|)$ with probability 1.*

Proof. (1) We already proved that $f(V_n) \to 0$ and $\mu(V_n) \to \lambda_1$ in lemma 2.3.5, thus $\lambda_i - \mu(V_n) > 0$ for $i \neq 1$ when n large enough. By (2.3.10),

$$0 = \lim_{n\to\infty} f(V_n) = \lim_{n\to\infty} \frac{\sum_{i=1}^{d} (a_i^{(n)})^2 (\lambda_i^2 - \mu(V_n)^2)}{\|V_n\|^2},$$

$a_i^{(n)} = 0$ when $i \neq 1$, thus $V_n \to a_1^{(n)}\theta_1$ with probability 1.

(2) By previous argument, we have:

$$\begin{aligned} f(V_n) &= \frac{\sum_{i=1}^{d} (a_i^{(n)})^2 (\lambda_i^2 - \mu(V_n)^2)}{\|V_n\|^2} \\ &= \frac{(a_1^{(n)})^2 (\lambda_1^2 - \mu(V_n)^2)}{\|V_n\|^2} + \frac{\sum_{i=2}^{d} (a_i^{(n)})^2 (\lambda_i^2 - \mu(V_n)^2)}{\|V_n\|^2}, \end{aligned}$$

convergence with the same rate of $\mu(V_n) \to \lambda_1$, we have $\frac{\sum_{i=2}^{\infty} (a_i^{(n)})^2 (\lambda_i^2 - \mu(V_n)^2)}{\|V_n\|^2} \to 0$ at least the same rate as $\frac{(a_1^{(n)})^2 (\lambda_1^2 - \mu(V_n)^2)}{\|V_n\|^2} \to 0$.

By part (1), $\mu(V_n)$ has rate of convergence $O(\frac{\|\Sigma\|}{\sqrt{n}} \cdot (\sqrt{E[\|A_n\|^2]}) \bigvee \|\Sigma\|)$, we have

$$\frac{\sum_{i=2}^{\infty} (a_i^{(n)})^2 (\lambda_i^2 - \lambda_1^2)}{\|V_n\|^2} \asymp_p \frac{\|\Sigma\|}{\sqrt{n}} \cdot (\sqrt{E[\|A_n\|^2]} \bigvee \|\Sigma\|) \cdot \frac{(a_1^{(n)})^2 \lambda_1}{\|V_n\|^2},$$

let $g = |\lambda_1 - \lambda_2|$, thus:

$$\begin{aligned}\sum_{i=2}^{\infty}(a_i^{(n)})^2 &\asymp_p \frac{\|\Sigma\|}{\sqrt{n}} \cdot (\sqrt{E[\|A_n\|^2]} \bigvee \|\Sigma\|) \cdot \frac{(a_1^{(n)})^2\lambda_1}{|(\lambda_i - \lambda_1)(\lambda_i + \lambda_1)|} \\ &\lesssim_p \frac{\|\Sigma\|\|V_n\|^2}{g\sqrt{n}} \cdot (\sqrt{E[\|A_n\|^2]} \bigvee \|\Sigma\|)\end{aligned}$$

Now by assumption 2, $\|V_n\|^2 = \sum_{i=1}^{d}(a_i^{(n)})^2$, thus:

$$\|V_n\|^2 - (a_1^{(n)})^2 = \sum_{i=2}^{\infty}(a_i^{(n)})^2 \lesssim_p \frac{\|\Sigma\|\|V_n\|^2}{g\sqrt{n}} \cdot (\sqrt{E[\|A_n\|^2]} \bigvee \|\Sigma\|).$$

Above all:

$$1 - \frac{< V_n, \theta_1 >^2}{\|V_n\|^2} \lesssim_p \frac{\|\Sigma\|}{g\sqrt{n}} \cdot (\sqrt{E[\|A_n\|^2]} \bigvee \|\Sigma\|).$$

□

2.4 Experiment

The dataset $X \in \mathbb{R}^{10^6 \times 10}$ was just generated through its singular value decomposition. Specifically, we fix a 10×10 diagonal matrix $\Sigma = diag\{1, 0.9, \cdots, 0.9\}$ and generate random orthogonal projection matrix $U \in \mathbb{R}^{10^6 \times 10}$ and random orthogonal matrix $V \in \mathbb{R}^{10 \times 10}$. And the dataset $X = \sqrt{n}U\Sigma V^T$, which guarantees that the matrix $A = \frac{1}{n}X^TX$ has eigen-gap 0.1. See Figure.1.

2.5 Conclusion

We derived the asymptotic rate of convergence for the estimation of the smallest eigenvalue and corresponding eigenvector of the Krasulina scheme. There are several important questions related to Online PCA.

1. The Krasulina scheme only requires $O(d)$ storage space complexity against $O(d^2)$ for standard PCA in the offline setting, however, we paid a price in the rate of con-

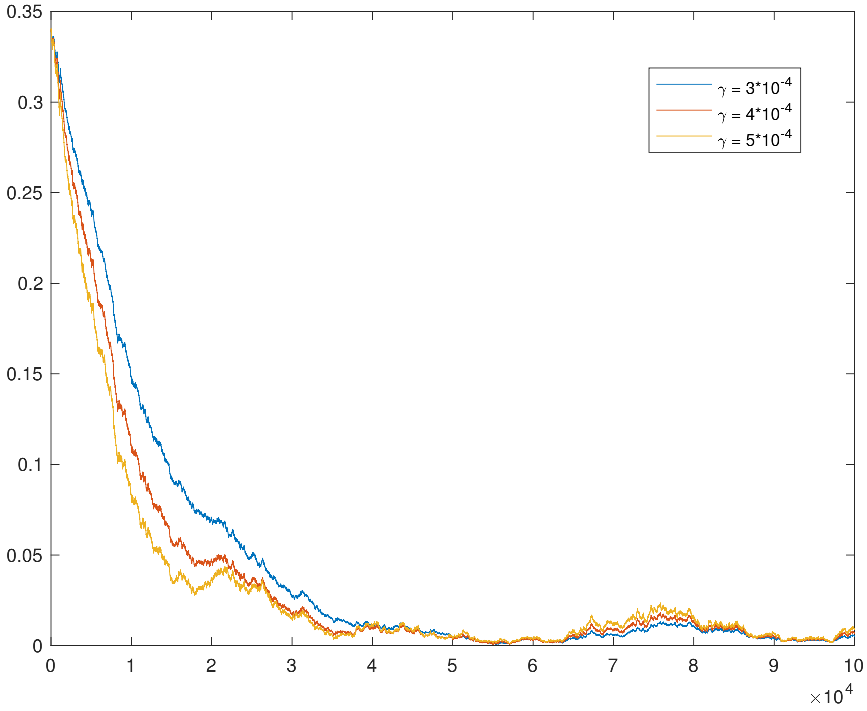


Figure 2.1: Convergence of Krasulina Scheme

Table 2.1: Comparison of different schemes for PCA.

Scheme	Computation complexity	Space complexity	Convergence rate	Setting
Standard PCA	$O(d^2n)$	$O(nd^2)$	$O(\|\Sigma\| \cdot (\sqrt{\frac{r(\Sigma)}{n}} \bigvee \frac{r(\Sigma)}{n}))$	Offline
Sparse PCA [8]	$O(d^2n)$	$O(nd^2)$	$O(\frac{k_q^*}{n\lambda}(d + \log \frac{d}{k_q^*}))$	Offline
Krasulina	$O(dn)$	$O(d)$	$O(\frac{\|\Sigma\|}{\sqrt{n}} \cdot (\sqrt{E[\|A_n\|^2]} \bigvee \|\Sigma\|))$	Online

vergence that is significantly slower than offline setting. See Table.2.1, it compares the different schemes. The computational complexity is correspondence to the complexity of one eigenvalue and eigenvector. the convergence rates are given for the operator norm. For the sparse PCA scheme of [8], k_q^* denotes the sparsity level of the eigenvectors.

An interesting question would be whether the Krasulina scheme can achieve the offline rate of convergence.

The simulation study seems to confirm the slow convergence rate of Krasulina's scheme. It would be interesting to build an acceleration for this scheme. This problem has been investigated by [20] where negative numerical results were provided for usual acceleration schemes. Therefore this question remains largely open.

2. Note that the proof argument in the original paper [5] only gives the consistency of the smallest eigenvalue and corresponding eigenvector for the Krasulina scheme. As we built upon this argument in this Chapter, we only provide the rate of convergence for the smallest eigenvalue and corresponding eigenvector. The reason for this limitation can be traced back to (2.3.9). The fact that λ_1 is the smallest eigenvalue is key to prove that the sequence $a_1^{(n)} = \langle V_n, \theta_1 \rangle$ is Cauchy and thus converging. Tackling other eigenvalues will require a new argument.

3. Finally, it would be of interest to derive rates of convergence for other online PCA schemes including Oja and naive PCA.

CHAPTER 3
UPDATED TEXT CLASSIFICATION METHODS

3.1 Introduction

Text classification problem has long been an interesting research field, the aim of text classification is to develop algorithm to find the categories of given documents. Text classification has many applications in natural language processing (NLP), such as spam filtering, email routing, and sentimental analysis. Despite intensive work,remains an open problem today.

This problem has been studied for many aspects, including: supervised classification problem, if we are given the labeled training data; unsupervised clustering problem, if we only have documents without labeling; as well as feature selection.

For supervised problem, if we assume that all the categories are independent multinomial distributions, and each document is a sample generated by that distribution, a straight forward idea is to using some linear models to distinguish them, such as support vector machine (SVM)[21, 22], which is used to find the ”maximum-margin hyperplane” that divides the documents with different labels. The algorithm is defined so that the distance between the hyperplane and the nearest sample d_i from either group is maximized. The hyperplane can be written as the set of documents vector $\vec{d}$ satisfying:

$$\vec{w} \cdot \vec{d} - b = 0,$$

where $\vec{w}$ is the normal vector to the hyperplane. Under the same assumption, another effective classifier, using scores based on the probability of given documents conditioned on the categories, is called naive Bayesian classifier[23, 24]. This classifier learns from training data to estimate the distribution of each categories, then we can compute the conditional probability of each documents d_i given the class label C_i by applying Bayes rule, then the predicting of the classes is done by choosing the highest posterior probability. The algorithm to get the label for a given document d is given by:

$$label(d) = \underset{j}{\operatorname{argmax}} P(C_j)P(d|C_j).$$

When we understand the documents as sequence of words, to understand the order of the words, given the data set large enough, we can using deep learning models such as Recurrent Neural Network (RNN)[25, 26].

For unsupervised problem. We have traditional method SVD (Singular Value Decomposition)[27] for the dimension reduction and clustering. There also exist some algorithms based on EM algorithm, such as pLSA (Probabilistic latent semantic analysis)[28], which consider the probability of each co-occurrence as a mixture of conditionally independent multinomial distributions:

$$\begin{aligned} P(w,d) &= \sum_C P(C)P(d|C)P(w|C) \\ &= P(d)\sum_C P(C|d)P(w|C), \end{aligned}$$

where w and d are observed words and documents, and C been the words' topic. As we mentioned, the parameters are learned by EM algorithm. Using the same idea, but assuming that the topic distribution has sparse Dirichlet prior, we have algorithm LDA (Latent Dirichlet allocation)[29]. The sparse Dirichlet priors encode the intuition that documents cover only a small set of topics and that topics use only a small set of words frequently. In practice, this results in a better disambiguation of words and a more precise assignment of documents to topics.

There are also many results in feature engineering, such as tf-idf[30], n-gram, or inproved tf-idf with other feature selection[31].

In many circumstances, the process of labeling is distributed among less-than-expert assessors. Therefore, their labeling for hundreds of pictures, texts, or messages a day is error-prone. The concept of partial labeling seeks to remedy the labor: instead of offering one or some exact labels, the annotators can offer a set of possible candidate solutions for one sample, thus providing a 'buffer' against potential mistakes. Other partial labeling settings involve repeated labeling to filter out noises, or assessing the quality of the labelers to enhance performances of the models.

As the data size in companies such as FANG(Facebook, Amazon, Netflix, Google) constantly reaches the magnitude of Petabyte, the demand for quick, yet still precise labeling is ever growing. Viewing some practices, the partial labeling frames that we know of have certain limitations. For example, in a real-world situation concerning NLP, if the task is to determine the class/classes of one article, an annotator with a bachelor major in American literature might find it difficult to determine if an article with words dotted with 'viscosity', 'gradient', and 'Laplacian' etc., belongs to computer science, math, physics, chemistry, or none of the classes above. As a result, the annotator might struggle within some limited amount of time amid a large pool of label classes and is likely to make imprecise choices even in a lenient, positive-oriented partial labeling environment. Another issue is the cost. Repeated labeling and keeping track of the performance of each labeler (assuming the sources where the labels are obtained are steady) may be pricey, and the anonymity of the labelers can raise another barrier wall to several partial labeling approaches.

Taking into consideration the real world scenarios, we present a new method to tackle the problem of how to gather at a large scale partially correct information from diverse annotators, while remaining efficient and budget-friendly. Still taking the above text classification as the example. Although that same annotator might not easily distinguish which categories the above-mentioned article belongs to, he/she can tell in a few seconds the article is not related to cuisines, TV-entertainments, or parenting. In our partial label formulation, the safe choices, crossed-off categories labeled by annotators, can still be of benefit. Furthermore, when contradictory labels are marked on one training sample and the identities of the labelers unknown, our introduced self-correcting estimator can select, and learn from the categories where the labels agree.

In this Chapter, we still assume that documents are generated according to a multinomial event model[32]. The first section, We defined a new method to estimate centroid based on the symmetric KL-divergence between the distribution of documents and their class centroids, which works better than original average estimated centroid in naive Bayes method, then in the second section, we define a new method based on traditional Naive Bayes estimator, and in the last section, we applied our new method in partial labeled problem.

3.2 Centroid estimation based on symmetric KL divergence

Notations: In this section, document belong to class j with index i is represented as a vector $d_i^j = (x_{i_1}, x_{i_2}, ..., x_{i_{|V|}})$ of word counts where V is the vocabulary, and each $x_{i_t} \in \{0, 1, 2, ...\}$ indicates how often w_t occurs in d_i. c_i denotes the centroid of the class C_i, since we use the assumption that documents are generated according to a multinomial event model, $c_i = (c_{i_1}, c_{i_2}, ...c_{i_{|V|}})$ satisfies: $\sum_{j=1}^{|V|} c_j = 1$.

3.2.1 Our Model

Let $p = (p_1, p_2, ..., p_n)$, $q = (q_1, q_2, ...q_n)$ be two multinomial distributions, the KL-divergence is defined as:

$$KL(p, q) = \sum_{i=1}^{n} p_i \log \frac{p_i}{q_i}.$$

KL-divergence measures how much one probability distribution is different from another, it is strongly connected with naive bayes classifier. Given class prior probabilities $p(C_j)$ and assuming independence of the words, normalize of document vector of d, the most likely class for a document $d = (d_1, d_2, ..., d_{|V|})$ satisfying $\sum_{i=1}^{|V|} d_i = 1$ is computed as:

$$\begin{aligned} label(d) &= \underset{j}{\operatorname{argmax}} P(C_j)P(d|C_j) \\ &= \underset{j}{\operatorname{argmax}} P(C_j)\prod_{i=1}^{|V|}(c_{j_i})^{d_i} \\ &= \underset{j}{\operatorname{argmax}} \log P(C_j) + \sum_{i=1}^{|V|} d_i \log c_{j_i} \\ &= \underset{j}{\operatorname{argmax}} \log P(C_j) - \sum_{i=1}^{|V|} d_i \log \frac{d_i}{c_{j_i}} \\ &= \underset{j}{\operatorname{argmin}} - \log P(C_j) + KL(d, c_j). \end{aligned} \tag{3.2.1}$$

To make it symmetric of p and q, we add in another term related to $q \log p$ as regularizer to get symmetric KL-divergence:

$$SKL(p,q) = \sum_{i=1}^{n}(p_i - q_i)\log\frac{p_i}{q_i}.$$

To compare several measures of difference of two distributions, let $p = (x, 1-x)$, $q = (0.01, 0.99)$, Figure.3.1 shows how the difference of two vectors change under different measures. We can see that for p and q far from each other, the difference of SKL decay faster, and for closer distributions, it decreases slower than linear speed. So SKL should be a good choice to distinguish distributions.

In the labeled training set, for each classes, we use SKL to find the centroid, whose sum of symmetric KL-divergence to all documents in that class reaches minimum, more specifically, the centoid is defined as following:

$$c_i = \underset{q}{\operatorname{argmin}} \sum_{p \in C_i} SKL(p,q). \tag{3.2.2}$$

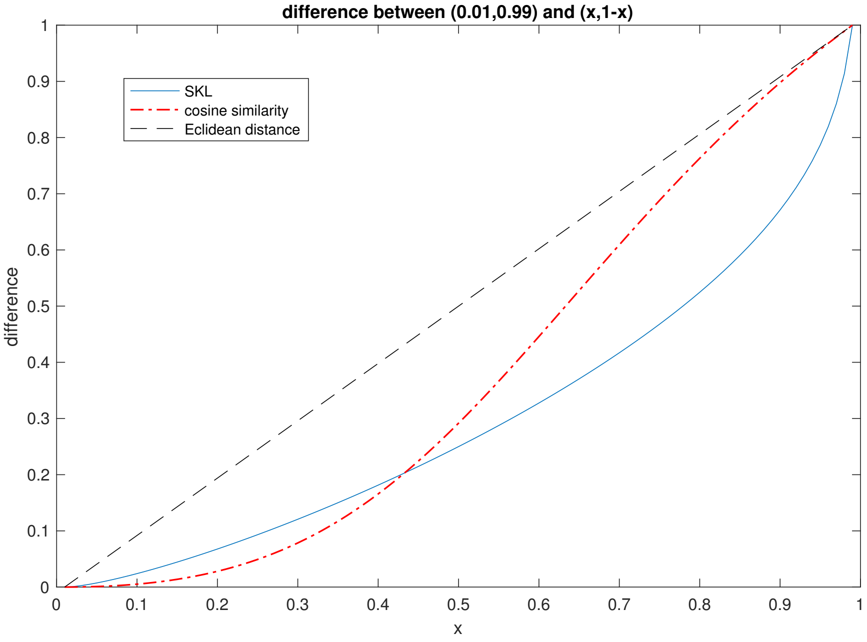


Figure 3.1: how difference changes between $p = (x, 1 - x)$ and $q = (0.01, 0.99)$ in SKL, cosine similarity and Eclidean distance.

Let $f(q) = \sum_{p \in C_i} SKL(p, q)$, since:

$$\begin{aligned} f(q) &= \sum_{j=1}^{|C_j|} \sum_{i=1}^{|V|} (p_i^j \log \frac{p_i^j}{q_i} + q_i \log \frac{q_i}{p_i^j}) \\ &= \sum_{j=1}^{|C_j|} \sum_{i=1}^{|V|} p_i^j \log p_i^j - p_i^j \log q_i + q_i \log q_i \\ &\quad -q_i \log p_i^j. \end{aligned}$$

Take partial derivative to q_i we obtain:

$$\frac{\partial f}{\partial q_i} = (\sum_{j=1}^{|C_j|} -\frac{p_i^j}{q_i} + \log q_i + 1 - \log p_i^j).$$

Thus:

$$\begin{cases} \dfrac{\partial^2 f}{\partial q_i^2} = \displaystyle\sum_{j=1}^{|C_j|} (\frac{p_i^j}{q_i^2} + \frac{1}{q_i}) \\ \dfrac{\partial^2 f}{\partial q_i q_k} = 0 \end{cases}$$

We can see that this is a convex problem. So we can obtain the global minimizer from minimization problem 3.2.2. After we get the estimation of centroid, we apply that in orginal naive bayes method 3.2.1, under this estimator, we expected it works better than original estimator of centroid.

3.2.2 Minimization problem

To solve 3.2.2 on the discrete probability manifold, the Wasserstein is used to get the gradient system. To this ends, suppose the graph structure $G = (V, E)$ is given where V are nodes set containing all the words involved and E defines the edge set which links the graph to be a connected graph. And in the examples below, the simplest histogram structure is used, that is, all the words are linked one by one in some order in a line. Also denote $n = |V|$ be the number of nodes on the graph.

Now consider a energy function $\mathcal{F}(\rho)$, let

$$F_i(\rho) = \frac{\partial}{\partial \rho_i} \mathcal{F}(\rho)$$

define the orientation O on G to be that for $(i, j) \in E$, the direction is from i to j if $F_i > F_j$ and that is arbitrary if $F_i = F_j$, denoting as $(i \to j) \in O$. Then the construction of the gradient of a potential function Φ based on the orientation is

$$\nabla_G \Phi = (\Phi_i - \Phi_j)_{(i \to j) \in O}, \quad (\phi_i)_{i=1}^n \in \mathbb{R}^n$$

Then, an inner product can be written as

$$(\nabla_G \Phi, \nabla_G \Phi)_\rho = \frac{1}{2} \sum_{(i \to j) \in O} g_{ij}(\rho)(\Phi_i - \Phi_j)^2$$

where

$$g_{ij}(\rho) = \begin{cases} \rho_i & \text{if } (i \to j) \in O \\ \rho_j & \text{if } (j \to i) \in O \end{cases}$$

and the gradient flow under this metric is known as discrete 2-Wasserstein gradient flow since the discrete 2-Wasserstein distance is defined as

$$W_2(\rho^0, \rho^1) = \inf_{\rho \in \mathcal{C}}$$
$$\left\{ \left(\int_0^1 (\nabla_G \Phi, \nabla_G \Phi)_\rho \right)^{\frac{1}{2}} : \frac{\partial \rho}{\partial t} + \nabla_G \cdot (\rho \nabla_G \Phi) = 0 \right\}$$

Now consider the energy function to be

$$\mathcal{F}(\rho) = \sum_{p \in C_i} SKL(p, \rho)$$

and the gradient flow can be written as

$$\dot{\rho}_i + \sum_{j \in N(i)} g_{ij}(\rho)(F_i(\rho) - F_j(\rho)) = 0$$

Solving this ODE obtains the solution for problem 3.2.2.

3.2.3 Experiment

We applied our method on seven topics of single labeled documents in Reuters-21578, we find the accuracy of naive bayes using our centroid estimator increasing faster than original method, see Figure.3.2, and when training size is large enough, our method achieves substantial improvements over the traditional method.

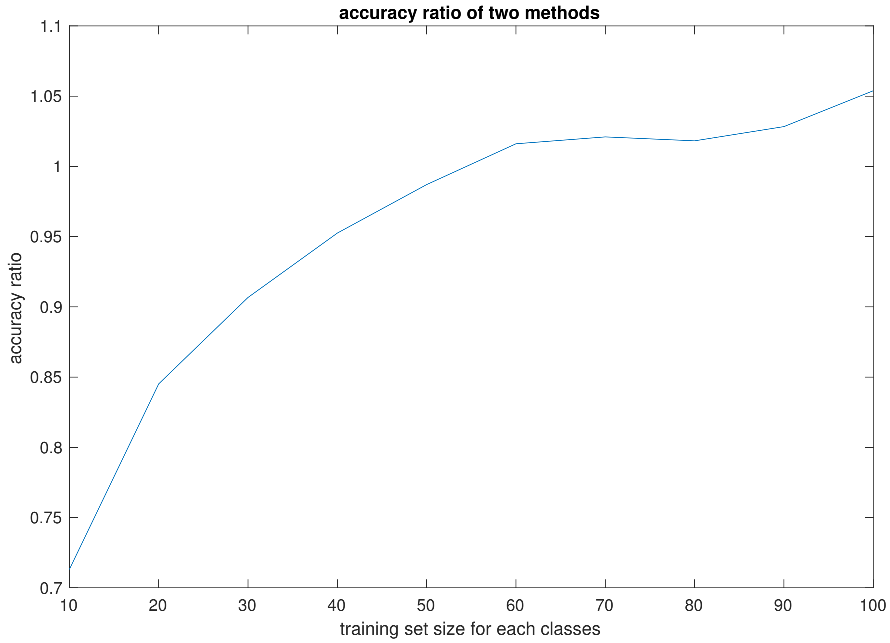


Figure 3.2: Average accuracy ratio under seven topics.

Table 3.1: average SKL to other classes

coffee	sugar	trade	ship	crude	interst	money-fx
9.0348	8.9305	6.2703	9.1293	7.3662	7.4778	6.9361

For each single class, the behave of our method versus traditional naive bayes estimator can be find in Figure.3.3. We can a clear increasing trend for topics as training size becoming larger.

Table.3.1 shows the average SKL to other classes, from Figure.3.3 we can see that class ’trade’ is the only one doesn’t have trend of increasing, that might because it is very closed to other classes, and SKL cannot distinguish it well based on our observation in Figure3.1.

3.2.4 Open problems

1. In this section, we find better estimator for centroid using naive bayes, can we find similar result for other estimators?

2. Can this centroid estimator be extended to be used in unsupervised learning problem?

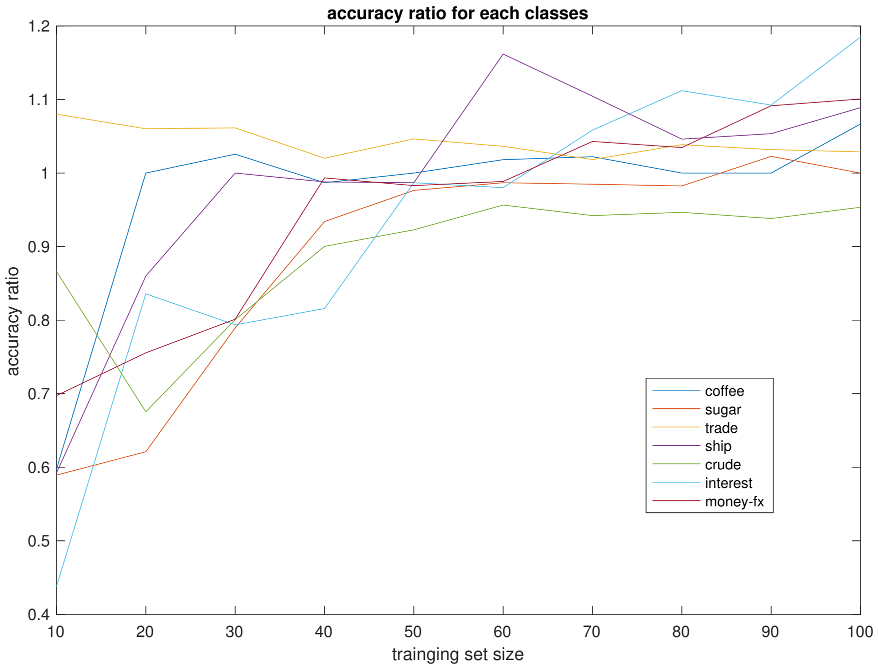


Figure 3.3: Accuracy ratio for seven topics.

3. When we solve the minimization problem, we have a graph structure for each feature. We are using a connecting graph now, can we use the partially connected graph to demonstrate correlation of words?

3.3 Updated naive bayes estimator for text classification problem

3.3.1 General Setting

Consider a classification problem with sample $x \in S$ and class set C, where

$$C = \{C_1, C_2, ..., C_k\}.$$

We are interested in finding our estimator:

$$\hat{y} = f(x;\theta) = (f_1(x;\theta), f_2(x;\theta), ..., f_k(x;\theta))$$

for y, where $\theta = \{\theta_1, \theta_2, ..., \theta_m\}$ is the parameter, and $f_i(x; \theta)$ is the likelihood function of sample x in class C_i. Define: $y = (y_1, y_2, ..., y_k)$, if x is in class C_i, then $y_i = 1$. Notice that if this is a single label problem, then we have: $\sum_{i=1}^{k} y_i = 1$.

3.3.2 Naive Bayes classifier in text classification problem

For Naive Bayes model. Let class C_i with centroid $\theta_i = (\theta_{i_1}, \theta_{i_2}, ..., \theta_{i_v})$, where v is the total number of the words and θ_i satisfies: $\sum_{j=1}^{v} \theta_{i_j} = 1$. Assuming independence of the words, the most likely class for a document $d = (x_1, x_2, ..., x_v)$ is computed as:

$$\begin{aligned} label(d) &= \operatorname*{argmax}_i P(C_i) P(d|C_i) \\ &= \operatorname*{argmax}_i P(C_i) \prod_{j=1}^{v} (\theta_{i_j})^{x_j} \\ &= \operatorname*{argmax}_i \log P(C_i) + \sum_{j=1}^{v} x_j \log \theta_{i_j}. \end{aligned} \tag{3.3.1}$$

So we have:

$$\log f_i(d, \theta) = \log P(C_i) + \sum_{j=1}^{v} x_j \log \theta_{i_j}.$$

For a class C_i, we have the standard likelihood function:

$$L(\theta) = \prod_{x \in C_i} \prod_{j=1}^{v} \theta_{i_j}^{x_j}, \tag{3.3.2}$$

Take logarithm for both side, we obtain the log-likelihood function:

$$\log L(\theta) = \sum_{x \in C_i} \sum_{j=1}^{v} x_j \log \theta_{i_j}. \tag{3.3.3}$$

We would like to solve optimization problem:

$$\begin{aligned} \max \quad & L(\theta) & \quad (3.3.4) \\ \text{subject to :} \quad & \sum_{j=1}^{v} \theta_{i_j} = 1 & \\ & \theta_{i_j} \geq 0. & \quad (3.3.5) \end{aligned}$$

The problem (3.3.4) can be solve explicitly with (3.3.3) by Lagrange Multiplier, for class C_i, we have $\theta_i = \{\theta_{i_1}, \theta_{i_2}, ..., \theta_{i_v}\}$, where:

$$\hat{\theta}_{i_j} = \frac{\sum_{d \in C_i} x_j}{\sum_{d \in C_i} \sum_{j=1}^{v} x_j}. \quad (3.3.6)$$

For estimator $\hat{\theta}$, we have following theorem.

Theorem 3.3.1. *Assume we have normalized length of each document, that is:* $\sum_{j=1}^{v} x_j = m$ *for all* d*, the estimator* (3.3.6) *satisfies following properties:*

1. $\hat{\theta}_{i_j}$ *is unbiased.*

2. $E[|\hat{\theta}_{i_j} - \theta_{i_j}|^2] = \frac{\theta_{i_j}(1-\theta_{i_j})}{|C_i|m}$.

Proof. With assumption $\sum_{j=1}^{v} x_j = m$, we can rewrite (3.3.6) as:

$$\hat{\theta}_{i_j} = \frac{\sum_{d \in C_i} x_j}{\sum_{d \in C_i} m} = \frac{\sum_{d \in C_i} x_j}{|C_i|m}.$$

Since $d = (x_1, x_2, ..., x_v)$ is multinomial distribution, with d in class C_i, we have: $E[x_j] = m \cdot \theta_{i_j}$, and $E[x_j^2] = m\theta_{i_j}(1 - \theta_{i_j} + m\theta_{i_j})$.

1.
$$\hat{\theta}_{i_j} = E[\frac{\sum_{d \in C_i} x_j}{|C_i|m}] = \frac{\sum_{d \in C_i} E[x_j]}{|C_i|m} = \frac{\sum_{d \in C_i} m \cdot \theta_{i_j}}{|C_i|m} = \theta_{i_j}.$$

Thus $\hat{\theta}_{i_j}$ is unbiased.

2. By (1), we have:

$$E[|\hat{\theta}_{i_j} - \theta_{i_j}|^2] = E[\hat{\theta}_{i_j}^2] - 2\theta_{i_j} E[\hat{\theta}_{i_j}] + \theta_{i_j}^2 = E[\hat{\theta}_{i_j}^2] - \theta_{i_j}^2.$$

Then

$$\hat{\theta}_{i_j}^2 = \frac{(\sum_{d \in C_i} x_j)^2}{|C_i|^2 m^2} = \frac{\sum_{d \in C_i} x_j^2 + \sum_{d_1, d_2 \in C_i} 2x_j^{d_1} x_j^{d_2}}{|C_i|^2 m^2}, \tag{3.3.7}$$

where $d_i = (x_1^{d_i}, x_2^{d_i}, ..., x_v^{d_i})$ for $i = 1, 2$. Since:

$$E[\sum_{d \in C_i} x_j^2] = \frac{|C_i| m \theta_{i_j} (1 - \theta_{i_j} + m\theta_{i_j})}{|C_i|^2 m^2} = \frac{\theta_{i_j}(1 - \theta_{i_j} + m\theta_{i_j})}{|C_i| m},$$

and

$$E[\sum_{d_1, d_2 \in C_i} 2x_j^{d_1} x_j^{d_2}] = \frac{|C_i|(|C_i| - 1) m^2 \theta_{i_j}^2}{|C_i|^2 m^2} = \frac{(|C_i| - 1)\theta_{i_j}^2}{|C_i|}.$$

Plugging them into (3.3.7) obtains:

$$E[\hat{\theta}_{i_j}^2] = \frac{\theta_{i_j}(1 - \theta_{i_j})}{|C_i| m} + \theta_{i_j}^2,$$

thus: $E[|\hat{\theta}_{i_j} - \theta_{i_j}|^2] = \frac{\theta_{i_j}(1-\theta_{i_j})}{|C_i| m}$.

□

3.3.3 Main Result

From Theorem.3.3.1, we can see that traditional Naive Bayes estimator $\hat{\theta}$ is an unbiased estimator with variance $O(\frac{\theta_{i_j}(1-\theta_{i_j})}{|C_i| m})$. Now we are trying to get our estimators, and prove that it can perform better than traditional Naive Bayes estimator.

Define:

$$\begin{aligned} L_1(\theta) &= \prod_{x\in S}\prod_{i=1}^{k} f_i(x;\theta)^{y_i(x)+t} \\ &= \prod_{x\in S}(\prod_{j=1}^{v}\theta_{i_j}^{x_j})^{y_i(x)+t}. \end{aligned} \tag{3.3.8}$$

Take logarithm for both side, we obtain the log-likelihood function:

$$\log L_1(\theta) = \sum_{x\in S}(y_i(x)+t)\sum_{j=1}^{v} x_j \log\theta_{i_j}. \tag{3.3.9}$$

We would like to solve optimization problem:

$$\max \quad \log L_1(\theta) \tag{3.3.10}$$

$$\begin{aligned} \text{subject to :} \quad & \sum_{j=1}^{v}\theta_{i_j} = 1 \\ & \theta_{i_j} \geq 0. \end{aligned} \tag{3.3.11}$$

Let:

$$G_i = 1 - \sum_{j=1}^{v}\theta_{i_j},$$

by Lagrange multiplier, we have:

$$\begin{cases} \dfrac{\partial\log(L_1)}{\partial\theta_{i_j}} + \lambda_i\dfrac{\partial G_i}{\partial\theta_{i_j}} = 0\,\forall\,1\leq i\leq k \text{ and } \forall\,1\leq j\leq v \\ \displaystyle\sum_{j=1}^{v}\theta_{i_j} = 1,\ \forall\,1\leq i\leq k \end{cases}$$

plug in, we obtain:

$$\begin{cases} \displaystyle\sum_{x\in S}\frac{(y_i(x)+t)x_j}{\theta_{i_j}} - \lambda_i = 0,\ \forall\,1\leq i\leq k \text{ and } \forall\,1\leq j\leq v \\ \displaystyle\sum_{j=1}^{v}\theta_{i_j} = 1,\ \forall\,1\leq i\leq k \end{cases} \tag{3.3.12}$$

Solve (3.3.12), we got the solution of optimization problem (3.3.10):

$$\hat{\theta}_{i_j}^{L_1} = \frac{\sum_{x \in S}(y_i(x) + t)x_j}{\sum_{j=1}^{v} \sum_{x \in S}(y_i(x) + t)x_j}. \tag{3.3.13}$$

Assume for each classes, we have prior distribution $p_1, p_2, ..., p_k$, and assume we have normalized length of each document, that is: $\sum_{j=1}^{v} x_j = m$. Then:

$$\begin{aligned} E[\hat{\theta}_{i_j}^{L_1}] &= \frac{\sum_{x \in S}(y_i(x) + t)E[x_j]}{\sum_{x \in S}(y_i(x) + t)m} \\ &= \frac{\sum_{x \in S} tE[x_j] + \sum_{x \in C_i} E[x_j]}{\sum_{x \in S} tm + \sum_{x \in C_i} m} \\ &= \frac{t|S| \sum_{l=1}^{k} p_l \theta_{l_j} + \theta_{i_j}|C_i|}{t|S| + |C_i|}. \end{aligned}$$

Thus:

$$E[|\hat{\theta}_{i_j}^{L_1} - \theta_{i_j}|] = \frac{t|S|| \sum_{l=1}^{k} p_l \theta_{l_j} - \theta_{i_j}|}{t|S| + |C_i|}. \tag{3.3.14}$$

On the other hand,

$$\begin{aligned} E[(\hat{\theta}_{i_j}^{L_1})^2] &= \frac{(\sum_{x \in S}(y_i(x) + t)E[x_j])^2}{(\sum_{x \in S}(y_i(x) + t)m)^2} \\ &= \frac{\sum_{x \in S}(2t+1)y_i(x)E[x_j^2] + \sum_{x \in S} t^2 E[x_j^2]}{\sum_{x \in S}(2t+1)y_i(x)m^2 + \sum_{x \in S} t^2 m^2} \\ &= \frac{(2t+1)|C_i|\theta_{i_j}(1 - \theta_{i_j} + m\theta_{i_j}) + t^2|S| \sum_{l=1}^{k} p_l \theta_{l_j}(1 - \theta_{l_j} + m\theta_{l_j})}{(2t+1)m|C_i| + t^2|S|m}. \end{aligned}$$

Thus:

$$\begin{aligned} E[|\hat{\theta}_{i_j}^{L_1} - \theta_{i_j}|^2] &= E[(\hat{\theta}_{i_j}^{L_1})^2] - 2E[\hat{\theta}_{i_j}^{L_1}]\theta_{i_j} + \theta_{i_j}^2 \\ &= \frac{(2t+1)|C_i|\theta_{i_j}(1 - \theta_{i_j} + m\theta_{i_j}) + t^2|S| \sum_{l=1}^{k} p_l \theta_{l_j}(1 - \theta_{l_j} + m\theta_{l_j})}{(2t+1)m|C_i| + t^2|S|m} \\ &\quad -2\frac{t|S| \sum_{l=1}^{k} p_l \theta_{l_j} + \theta_{i_j}|C_i|}{t|S| + |C_i|}\theta_{i_j} + \theta_{i_j}^2 \rightarrow \frac{1}{|S|m} \, as \, t \rightarrow \infty. \end{aligned}$$

We can see that $E[|\hat{\theta}_{i_j}^{L_1} - \theta_{i_j}|^2]$ is in $O(\frac{1}{|S|})$ when t is large, which means it convergent faster than standard Naive Bayes $O(\frac{1}{|C_i|})$, however, since $E[|\hat{\theta}_{i_j}^{L_1} - \theta_{i_j}|] \neq 0$, it is not an unbiased estimator. To determine parameter t, assume the cost for unbiased estimator is c_1, the cost for convergence speed is c_2, then the parameter t can be solve by the following optimization problem:

$$t = \arg\min_t c_1 E[|\hat{\theta}_{i_j}^{L_1} - \theta_{i_j}|] + c_2 E[|\hat{\theta}_{i_j}^{L_1} - \theta_{i_j}|^2].$$

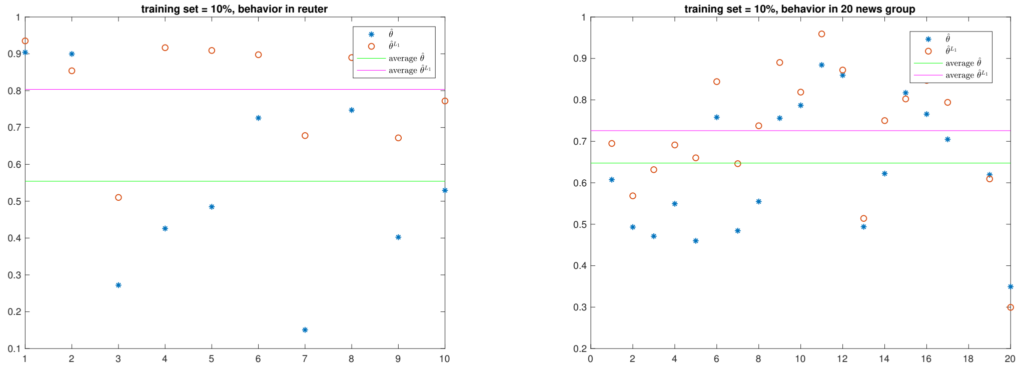
3.3.4 Experiment

We applied our method on top 10 topics of single labeled documents in Reuters-21578 data[3], and 20 news group data[33]. we compare the result of traditional Naive Bayes estimator (3.3.6): $\hat{\theta}_{i_j}$, and our estimator (3.3.13): $\hat{\theta}_{i_j}^{L_1}$. In the simulation, t is chosen to be 1 in all the following figures.

First of all, we run both the algorithms on these two sample sets. We know that when sample size becomes large enough, both estimators actually convergence into something else, but when training set small, our estimator should converge faster. Thus we first take the training size relatively small. See Figure.3.4(a) and Figure.3.4(b). According from the experiment, we can see our method is more accurate for most of the classes, and more accurate in average.

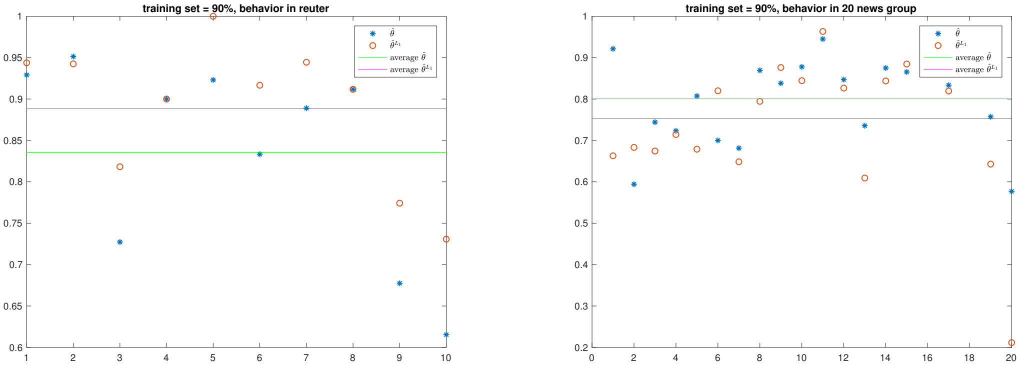
Then we test our estimator $\hat{\theta}^{L_1}$ with larger dataset. In our analysis before, we know that as dataset becomes large enough, our estimator converges to something else, so we expect a better result in traditional Naive Bayes estimator. See Figure.3.5(a) and Figure.3.5(b). According from the experiment, we can see for 20 news group, Naive Bayes already becomes better than our method, but our method is still more accurate than Naive Bayes in Reuter's data. This might because we have a huge unbalance dataset in Reuter's data, 90% of the training set is still not large enough for many classes.

Finally, We try to apply same training set and test the accuracy just on training set, we find traditional Naive Bayes estimator actually achieve better result, that means it might



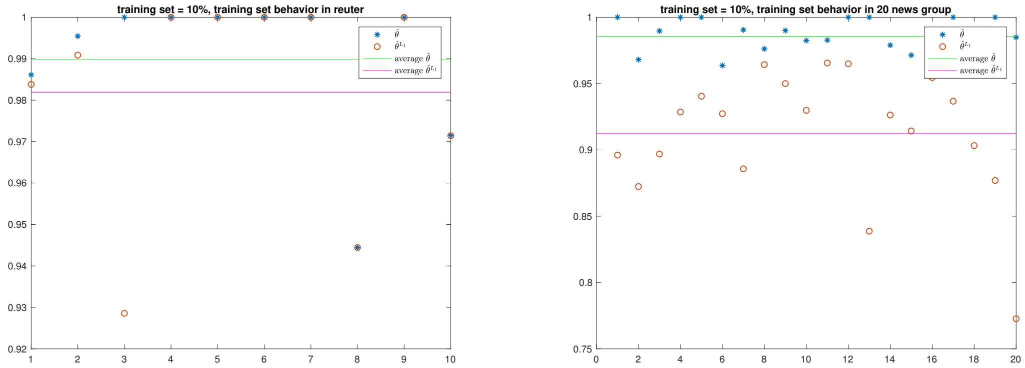
(a) (b)

Figure 3.4: We take 10 largest groups in Reuter-21578 dataset (a) and 20 news group dataset (b), and take 10% of the data as training set. The y-axis is the accuracy, and the x-axis is the class index.



(a) (b)

Figure 3.5: We take 10 largest groups in Reuter-21578 dataset (a) and 20 news group dataset (b), and take 90% of the data as training set. The y-axis is the accuracy, and the x-axis is the class index.



(a) (b)

Figure 3.6: We take 10 largest groups in Reuter-21578 dataset (a), and 20 news group dataset (b), and take 10% of the data as training set. We test the result on training set. The y-axis is the accuracy, and the x-axis is the class index.

have more over-fitting problems. This might be the reason why our method works better when dataset is not too large: adding the parameter t help us bring some uncertainly in training process, which help avoid over-fitting. See Figure.3.6(a) and Figure.3.6(b).

3.4 A cost-reducing partial labeling estimator in text classification problem

In this section, we are going to introduce partial labeling problem, and illustrate how to apply our method to solve it.

3.4.1 Related work

The text classification problem is seeking a way to best distinguish different types of documents[34, 35]. Being a traditional natural language processing problem, one needs to make full use of the words and sentences, converting them into various input features, and applying different models to process training and testing. A common way to convert words into features is to encoding them based on the term frequency and inverse document frequency, as well as the sequence of the words. There are many results about this, for example, tf-idf[30] encodes term t in document d of corpus D as:

$$tfidf(t,d,D) = tf(t,d) \cdot idf(t,D),$$

where $tf(t,d)$ is defined as term frequency, it can be computed as $tf(t,d) = \frac{|t:t \in d|}{|d|}$, and $idf(t,D)$ is defined as inverse document frequency, it can be computed as

$$idf(t,D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}.$$

We also have n-gram techniques, which first combines n nearest words together as a single term, and then encodes it with tf-idf. Recently, instead of using tf-idf, [31] defines a new feature selection score for text classification based on the KL-divergence between the distribution of words in training documents and their classes.

A popular model to achieve our aim is to use Naive Bayes model[23, 24], the label for a given document d is given by:

$$label(d) = \underset{j}{\operatorname{argmax}} P(C_j)P(d|C_j),$$

where C_j is the j-th class. For example, we can treat each class as a multinomial distribution, and the corresponding documents are samples generated by the distribution. With this assumption, we desire to find the centroid for every class, by either using the maximum likelihood function or defining other different objective functions[36] in both supervised and unsupervised learning version[28]. Although the assumption of this method is not exact in this task, Naive Bayes achieves high accuracy in practical problems.

There are also other approaches to this problem, one of which is simply finding linear boundaries of classes with support vector machine[22, 21]. Recurrent Neural Network (RNN)[26, 25] combined with word embedding is also a widely used model for this problem.

In real life, one may have different type of labels[37], in which circumstance, semi-supervised learning or partial-label problems need to be considered [38]. There are several methods to encode the partial label information into the learning framework. For the partial label data set, one can define a new loss combining all information of the possible labels, for example, in [39], the authors modify the traditional L^2 loss:

$$L(w) = \frac{1}{n+m}\left[\sum_{i=1}^{n} l(x_i, y_i, w) + \sum_{i=1}^{m} l(x_i, Y_i, w)\right],$$

where Y_i is the possible label set for x_i and $l(x_i, Y_i, w)$ is a non-negative loss function, and in [38], they defined convex loss for partial labels as:

$$L_\Psi(g(x), y) = \Psi(\frac{1}{|y|}\sum_{a\in y} g_a(x)) + \sum_{a\notin y} \Psi(-g_a(x)),$$

where Ψ is a convex function, y is a singleton, and $g_a(x)$ is a score function for label a as input x. A modification of the likelihood function is as well an approach to this problem and [40] gives the following optimization problem using Naive Bayes method

$$\theta^* = \arg\max_\theta \sum_i \sum_{y_i \in S_i} p(y|x_i, \theta)$$

where S_i is the possible labels for x_i.

Meanwhile, the similarity of features among data could be considered to give a confidence of each potential labels for a certain data. In [41], K nearest neighbor (KNN) is adopted to construct a graph structure with the information of features while Rocchio and Rocchio with clustering are used in [37].

3.4.2 General Setting

Consider a classification problem with sample $x \in S$ and class set C, where

$$C = \{C_1, C_2, ..., C_k\}.$$

We are interested in finding our estimator:

$$\hat{y} = f(x;\theta) = (f_1(x;\theta), f_2(x;\theta), ..., f_k(x;\theta))$$

for y, where $\theta = \{\theta_1, \theta_2, ..., \theta_m\}$ is the parameter, and $f_i(x;\theta)$ is the likelihood function of sample x in class C_i. Now assuming that in training set, we have two types of dataset S_1 and S_2, such that $S = S_1 \cup S_2$:

1. dataset S_1: we know exactly that sample x is in a class, and not in other classes. In this case, define: $y = (y_1, y_2, ..., y_k)$, if x is in class C_i, then $y_i = 1$. Notice that if this is a single label problem, then we have: $\sum_{i=1}^{k} y_i = 1$.

2. dataset S_2: we only have the information that sample x is not in a class, then $y_i = 0$. In this case, define: $z = (z_1, z_2, ..., z_k)$, if x is not in class C_i, we have $z_i = 1$.

To build the model, we define the following likelihood ratio function and likelihood function:

$$L_1(\theta) = \prod_{x\in S_1}\prod_{i=1}^{k} f_i(x;\theta)^{y_i} \prod_{x\in S_2}\prod_{i=1}^{k} f_i(x;\theta)^{\frac{1-z_i}{k-\sum_{j\neq i} z_j}}. \quad (3.4.1)$$

$$L_2(\theta) = \prod_{x\in S} \frac{\prod_{i=1}^{k} f_i(x;\theta)^{y_i(x)+t}}{\prod_{i=1}^{k} f_i(x;\theta)^{z_i(x)}} = \prod_{x\in S}\prod_{i=1}^{k} f_i(x;\theta)^{y_i(x)-z_i(x)+t}. \quad (3.4.2)$$

The t in L_2 satisfy $t > 1$, which is a parameter to avoid non-convexity.

The intuition of L_1 is to consider the sample labeled $z_i = 1$ has equal probability to be labeled in the other classes, each of the classes will have probability $\frac{1-z_i}{k-\sum_{j\neq i} z_j}$. And the intuition of L_2 is to consider this in a likelihood ratio way, the $z_i = 1$ labeled sample will have negative affection for class C_i, so we put it in the denominator. With $t > 1$, all the terms in denominator will finally be canceled out, so that even $f_i(x;\theta) = 0$ for some sample $x \in S$ will not cause trouble. Another intuition for L_2 is that, it can be self-correct the repeated data, which has been labeled incorrectly.

Take logarithm for both side, we obtain the following functions:

$$\log(L_1(\theta)) = \sum_{x \in S_1} \sum_{i=1}^{k} y_i(x) \log f_i(x, \theta) + \sum_{x \in S_2} \sum_{i=1}^{k} \frac{1 - z_i}{k - \sum_{j \neq i} z_j} \log f_i(x, \theta), \qquad (3.4.3)$$

and

$$\log(L_2(\theta)) = \sum_{x \in S} \sum_{i=1}^{k} (y_i(x) + t - z_i(x)) \log f_i(x, \theta). \qquad (3.4.4)$$

We would like to find our estimator $\hat{\theta}$ such that (3.4.4) or (3.4.3) reaches maximum.

3.4.3 Main Result

From Theorem.3.3.1, we can see that traditional Naive Bayes estimator $\hat{\theta}$ is an unbiased estimator with variance $O(\frac{\theta_{i_j}(1-\theta_{i_j})}{|C_i|m})$. Now we are trying to solve our estimators, and prove they can use the data in dataset S_2, and perform better than traditional Naive Bayes estimator.

Text classification with L_1 setting (3.4.1)

In order to use data both in S_1 and S_2, we would like to solve (3.3.4) with $L(\theta) = L_1(\theta)$, where L_1 is defined as (3.4.1), let:

$$G_i = 1 - \sum_{j=1}^{v} \theta_{i_j},$$

by Lagrange multiplier, we have:

$$\begin{cases} \dfrac{\partial \log(L_1)}{\partial \theta_{i_j}} + \lambda_i \dfrac{\partial G_i}{\partial \theta_{i_j}} = 0 \; \forall \; 1 \le i \le k \text{ and } \forall \; 1 \le j \le v \\ \displaystyle\sum_{j=1}^{v} \theta_{i_j} = 1, \; \forall \; 1 \le i \le k \end{cases}$$

Plug in, we obtain:

$$\begin{cases} \displaystyle\sum_{x \in S_1} \frac{y_i(x) x_j}{\theta_{i_j}} + \sum_{x \in S_2} \frac{1 - z_i(x)}{k - \sum_{l \ne i} z_l(x)} \cdot \frac{x_j}{\theta_{i_j}} - \lambda_i = 0, \; \forall \; 1 \le i \le k \text{ and } \forall \; 1 \le j \le v \\ \displaystyle\sum_{j=1}^{v} \theta_{i_j} = 1, \; \forall \; 1 \le i \le k \end{cases} \tag{3.4.5}$$

Solve (3.4.5), we got the solution of optimization problem (3.3.4):

$$\hat{\theta}_{i_j}^{L_1} = \frac{\sum_{x \in S_1} y_i(x) x_j + \sum_{x \in S_2} \frac{1 - z_i(x)}{k - \sum_{l \ne i} z_l(x)} x_j}{\sum_{x \in S_1} y_i(x) \sum_{j=1}^{v} x_j + \sum_{x \in S_2} \frac{1 - z_i(x)}{k - \sum_{l \ne i} z_l(x)} \sum_{j=1}^{v} x_j}. \tag{3.4.6}$$

Text classification with L_2 setting (3.4.2)

Another way to use both S_1 and S_2 dataset is to solve (3.3.4) with $L(\theta) = L_2(\theta)$, where L_2 is defined as (3.4.2), let:

$$G_i = 1 - \sum_{j=1}^{v} \theta_{i_j},$$

by Lagrange multiplier, we have:

$$\begin{cases} \dfrac{\partial \log(L_2)}{\partial \theta_{i_j}} + \lambda_i \dfrac{\partial G_i}{\partial \theta_{i_j}} = 0 \; \forall \; 1 \le i \le k \text{ and } \forall \; 1 \le j \le v \\ \displaystyle\sum_{j=1}^{v} \theta_{i_j} = 1, \; \forall \; 1 \le i \le k \end{cases}$$

Plug in, we obtain:

$$\begin{cases} \sum_{x\in S}(y_i(x)+t-z_i(x))\dfrac{x_j}{\theta_{i_j}} - \lambda_i = 0 \; \forall\, 1 \leq i \leq k \text{ and } \forall\, 1 \leq j \leq v \\ \sum_{j=1}^{v} \theta_{i_j} = 1, \; \forall\, 1 \leq i \leq k \end{cases} \tag{3.4.7}$$

Solve (3.4.7), we got the solution of optimization problem (3.3.4):

$$\hat{\theta}_{i_j}^{L_2} = \frac{\sum_{x\in S}(y_i(x)+t-z_i(x))x_j}{\sum_{j=1}^{v}\sum_{x\in S}(y_i(x)+t-z_i(x))x_j}. \tag{3.4.8}$$

Notice that the parameter t here is used to avoid non-convexity, when $0 \leq t < 1$, the optimization problem (3.3.4) has the optimizer located on the boundary of θ, which cannot be solved explicitly.

Improvement of Naive Bayes estimator with only S_1 dataset

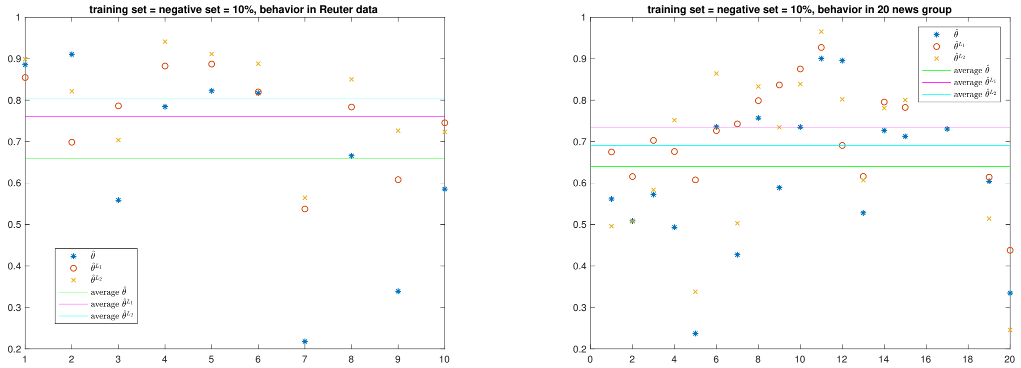
Now assume that we don't have dataset S_2, but only have dataset $S = S_1$, can we still do better than traditional Naive Bayes estimator $\hat{\theta}$? To improve the estimator, we can try to use L_1 or L_2 setting. With $z(x) = 1 - y(x)$, we can define function z on S_1 dataset.

With simple computation, we have the estimator of L_1 is the same as $\hat{\theta}_{i_j}$. as for the estimator for L_2, we have:

$$\hat{\theta}_{i_j}^{*} = \frac{\sum_{x\in S}(2y_i(x)+t-1)x_j}{\sum_{j=1}^{v}\sum_{x\in S}(2y_i(x)+t-1)x_j}, \tag{3.4.9}$$

3.4.4 Experiment

We applied our method on top 10 topics of single labeled documents in Reuters-21578 data[3], and 20 news group data[33]. we compare the result of traditional Naive Bayes estimator $\hat{\theta}_{i_j}$ and our estimator $\hat{\theta}_{i_j}^{L_1}$, $\hat{\theta}_{i_j}^{L_2}$, as well as $\hat{\theta}_{i_j}^{*}$. t is chosen to be 2 in all the following figures. The data in S_2 is generated randomly by not belong to a class, for example, if we



(a) (b)

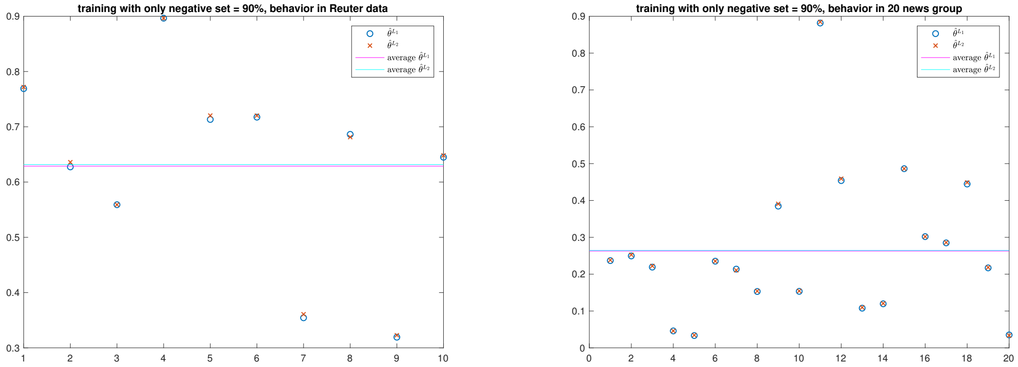
Figure 3.7: We take 10 largest groups in Reuter-21578 dataset (a) and 20 news group dataset (b), and take 20% of the data as training set, among which $|S_1| = |S_2|$. The y-axis is the accuracy, and the x-axis is the class index.

know a document d is in class 1 among 10 classes in Reuter's data, to put d in S_2, we randomly pick one class from 2 to 10, and mark d not in that class.

First of all, we run all the algorithms on these two sample sets. We know that when sample size becomes large enough, our estimators actually convergence into something else, but when sample size small enough, our estimator should converge faster. Thus we take the training size relatively small. See Figure.3.7(a) and Figure.3.7(b). According from the experiment, we can see our methods are more accurate for most of the classes, and more accurate in average.

Then we consider a more extreme case. If we have a dataset with $|S_1| = 0$, that is to say, we have no positive labeled data. In this setting, traditional Naive Bayes will not work, but what will we get from our estimators? See Figure.3.8(a) and Figure.3.8(b). We can see we can still get some information from negative labeled data. The accuracy is not as good as Figure.3.7(b) and Figure.3.7(a), that is because for each of the sample, negative label is only a part of information of positive label.

At last, we test our estimator $\hat{\theta}^{L_2}$ with only S_1 dataset, see Figure.3.9(a) and Figure.3.9(b). We can see our method achieve better result than traditional Naive Bayes estimator. We try to apply same training set and test the accuracy just on training set, we find

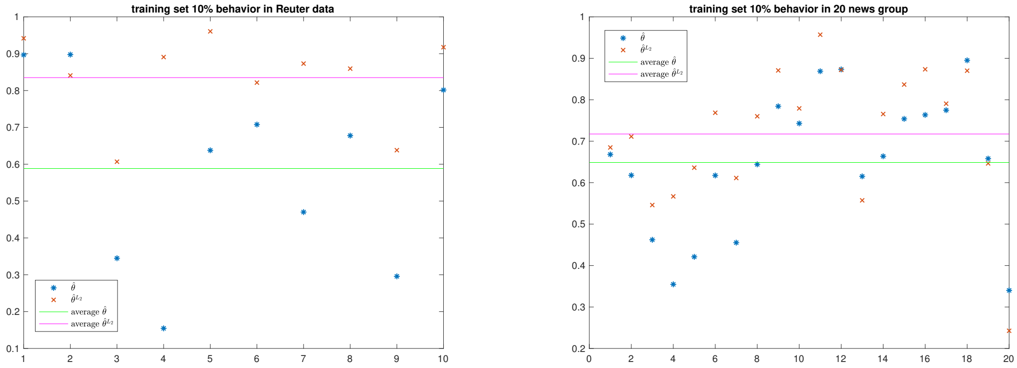


(a)

(b)

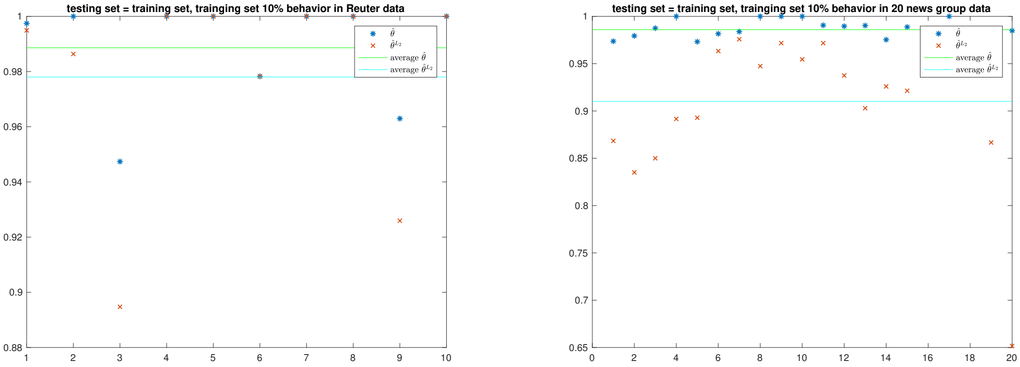
Figure 3.8: We take 10 largest groups in Reuter-21578 dataset (a), and 20 news group dataset (b), and take 90% of the data as S_2 training set. The y-axis is the accuracy, and the x-axis is the class index.

traditional Naive Bayes estimator actually achieve better result, that means it might have more over-fitting problems, see Figure.3.10(a) and Figure.3.10(b).



(a) (b)

Figure 3.9: We take 10 largest groups in Reuter-21578 dataset (a), and 20 news group dataset (b), and take 10% of the data as S_1 training set. The y-axis is the accuracy, and the x-axis is the class index.



(a) (b)

Figure 3.10: We take 10 largest groups in Reuter-21578 dataset(a), and 20 news group dataset (b), and take 10% of the data as S_1 training set. We test the result on training set. The y-axis is the accuracy, and the x-axis is the class index.

REFERENCES

[1] V. Koltchinskii and K. Lounici, “Concentration Inequalities and Moment Bounds for Sample Covariance Operators,” *arXiv:1405.2468 [math]*, May 2014, arXiv: 1405.2468.

[2] V. Koltchinskii, K. Lounici, *et al.*, “Asymptotics and concentration bounds for bilinear forms of spectral projectors of sample covariance,” in *Annales de l’Institut Henri Poincaré, Probabilités et Statistiques*, Institut Henri Poincaré, vol. 52, 2016, pp. 1976–2013.

[3] D. D. Lewis, *Reuters-21578*.

[4] T. W. Anderson, “An introduction to multivariate statistical analysis,” Wiley New York, Tech. Rep., 1962.

[5] T. P. Krasulina, “The method of stochastic approximation for the determination of the least eigenvalue of a symmetrical matrix,” *USSR Computational Mathematics and Mathematical Physics*, vol. 9, no. 6, pp. 189–195, 1969.

[6] R. Arora, A. Cotter, and N. Srebro, “Stochastic optimization of pca with capped msg,” in *Advances in Neural Information Processing Systems*, 2013, pp. 1815–1823.

[7] G. Blanchard, O. Bousquet, and L. Zwald, “Statistical properties of kernel principal component analysis,” *Machine Learning*, vol. 66, no. 2, pp. 259–294, 2007.

[8] T. T. Cai, Z. Ma, and Y. Wu, “Sparse PCA: Optimal rates and adaptive estimation,” *The Annals of Statistics*, vol. 41, no. 6, pp. 3074–3110, Dec. 2013, arXiv: 1211.1309.

[9] S. T. Roweis, “Em algorithms for pca and spca,” in *Advances in neural information processing systems*, 1998, pp. 626–632.

[10] V. Q. Vu and J. Lei, “Minimax rates of estimation for sparse PCA in high dimensions,” in *International Conference on Artificial Intelligence and Statistics*, 2012, pp. 1278–1286.

[11] M. K. Warmuth and D. Kuzmin, “Randomized pca algorithms with regret bounds that are logarithmic in the dimension,” in *Advances in neural information processing systems*, 2007, pp. 1481–1488.

[12] L. Zwald and G. Blanchard, “On the convergence of eigenspaces in kernel principal component analysis,” in *Advances in neural information processing systems*, 2006, pp. 1649–1656.

[13] V. Koltchinskii, K. Lounici, *et al.*, “Normal approximation and concentration of spectral projectors of sample covariance,” *The Annals of Statistics*, vol. 45, no. 1, pp. 121–157, 2017.

[14] E. Oja and J. Karhunen, “On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix,” *Journal of mathematical analysis and applications*, vol. 106, no. 1, pp. 69–84, 1985.

[15] J. Weng, Y. Zhang, and W.-S. Hwang, “Candid covariance-free incremental principal component analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 1034–1040, 2003.

[16] R. Arora, A. Cotter, K. Livescu, and N. Srebro, “Stochastic optimization for PCA and PLS,” in *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Oct. 2012, pp. 861–868.

[17] I. Mitliagkas, C. Caramanis, and P. Jain, “Memory limited, streaming pca,” in *Advances in Neural Information Processing Systems*, 2013, pp. 2886–2894.

[18] P. Jain, C. Jin, S. M. Kakade, P. Netrapalli, and A. Sidford, “Streaming pca: Matching matrix bernstein and near-optimal finite sample guarantees for ojas algorithm,” in *Conference on Learning Theory*, 2016, pp. 1147–1164.

[19] A. Balsubramani, S. Dasgupta, and Y. Freund, “The fast convergence of incremental pca,” in *Advances in Neural Information Processing Systems*, 2013, pp. 3174–3182.

[20] C. De Sa, B. He, I. Mitliagkas, C. R, and P. Xu, “Accelerated stochastic power iteration,” *arXiv preprint arXiv:1707.02670*, 2017.

[21] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[22] T. Joachims, “Text categorization with support vector machines: Learning with many relevant features,” in *European conference on machine learning*, Springer, 1998, pp. 137–142.

[23] N. Friedman, D. Geiger, and M. Goldszmidt, “Bayesian network classifiers,” *Machine learning*, vol. 29, no. 2-3, pp. 131–163, 1997.

[24] P. Langley, W. Iba, K. Thompson, *et al.*, “An analysis of bayesian classifiers,” in *Aaai*, vol. 90, 1992, pp. 223–228.

[25] D. Tang, B. Qin, and T. Liu, “Document modeling with gated recurrent neural network for sentiment classification,” in *Proceedings of the 2015 conference on empirical methods in natural language processing*, 2015, pp. 1422–1432.

[26] P. Liu, X. Qiu, and X. Huang, "Recurrent neural network for text classification with multi-task learning," *arXiv preprint arXiv:1605.05101*, 2016.

[27] R. Albright, "Taming text with the svd," *SAS Institute Inc*, 2004.

[28] T. Hofmann, "Probabilistic latent semantic analysis," in *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, Morgan Kaufmann Publishers Inc., 1999, pp. 289–296.

[29] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[30] J. Ramos *et al.*, "Using tf-idf to determine word relevance in document queries," in *Proceedings of the first instructional conference on machine learning*, vol. 242, 2003, pp. 133–142.

[31] K.-M. Schneider, "A new feature selection score for multinomial naive bayes text classification based on kl-divergence," in *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, Association for Computational Linguistics, 2004, p. 24.

[32] A. McCallum, K. Nigam, *et al.*, "A comparison of event models for naive bayes text classification," in *AAAI-98 workshop on learning for text categorization*, Citeseer, vol. 752, 1998, pp. 41–48.

[33] K. Lang, *20 newsgroups data set.*

[34] S. Dumais, J. Platt, D. Heckerman, and M. Sahami, "Inductive learning algorithms and representations for text categorization," in *Proceedings of the seventh international conference on Information and knowledge management*, ACM, 1998, pp. 148–155.

[35] L. S. Larkey, "Automatic essay grading using text categorization techniques," in *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, 1998, pp. 90–95.

[36] J. Chen, H. Matzinger, H. Zhai, and M. Zhou, "Centroid estimation based on symmetric kl divergence for multinomial text classification problem," *arXiv preprint arXiv:1808.10261*, 2018.

[37] X. Li and B. Liu, "Learning to classify texts using positive and unlabeled data," in *IJCAI*, vol. 3, 2003, pp. 587–592.

[38] T. Cour, B. Sapp, and B. Taskar, "Learning from partial labels," *Journal of Machine Learning Research*, vol. 12, no. May, pp. 1501–1536, 2011.

[39] N. Nguyen and R. Caruana, "Classification with partial labels," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2008, pp. 551–559.

[40] R. Jin and Z. Ghahramani, "Learning with multiple labels," in *Advances in neural information processing systems*, 2003, pp. 921–928.

[41] M.-L. Zhang, B.-B. Zhou, and X.-Y. Liu, "Partial label learning via feature-aware disambiguation," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, 2016, pp. 1335–1344.

[42] O. Shamir, "A stochastic pca and svd algorithm with an exponential convergence rate," in *International Conference on Machine Learning*, 2015, pp. 144–152.

[43] V. S. Sheng, F. Provost, and P. G. Ipeirotis, "Get another label? improving data quality and data mining using multiple, noisy labelers," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2008, pp. 614–622.

[44] M.-L. Zhang and Z.-H. Zhou, "Ml-knn: A lazy learning approach to multi-label learning," *Pattern recognition*, vol. 40, no. 7, pp. 2038–2048, 2007.

[45] ——, "A review on multi-label learning algorithms," *IEEE transactions on knowledge and data engineering*, vol. 26, no. 8, pp. 1819–1837, 2014.

[46] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown, "Learning multi-label scene classification," *Pattern recognition*, vol. 37, no. 9, pp. 1757–1771, 2004.

[47] A. McCallum, "Multi-label text classification with a mixture model trained by em," in *AAAI workshop on Text Learning*, 1999, pp. 1–7.

[48] I. Rish *et al.*, "An empirical study of the naive bayes classifier," in *IJCAI 2001 workshop on empirical methods in artificial intelligence*, IBM New York, vol. 3, 2001, pp. 41–46.

[49] Jeff, *Dimension reduction with pca*.

[50] O. Alter, P. O. Brown, and D. Botstein, "Singular value decomposition for genome-wide expression data processing and modeling," *Proceedings of the National Academy of Sciences*, vol. 97, no. 18, pp. 10 101–10 106, 2000.

[51] O. Alter and G. H. Golub, "Singular value decomposition of genome-scale mrna lengths distribution reveals asymmetry in rna gel electrophoresis band broadening," *Proceedings of the National Academy of Sciences*, vol. 103, no. 32, pp. 11 828–11 833, 2006.

[52] N. M. Bertagnolli, J. A. Drake, J. M. Tennessen, and O. Alter, “Svd identifies transcript length distribution functions from dna microarray data and reveals evolutionary forces globally affecting gbm metabolism,” *PloS one*, vol. 8, no. 11, e78913, 2013.

[53] C. Davis and W. M. Kahan, “The rotation of eigenvectors by a perturbation. iii,” *SIAM Journal on Numerical Analysis*, vol. 7, no. 1, pp. 1–46, 1970.

[54] L. Omberg, J. R. Meyerson, K. Kobayashi, L. S. Drury, J. F. Diffley, and O. Alter, “Global effects of dna replication and dna replication origin activity on eukaryotic gene expression,” *Molecular systems biology*, vol. 5, no. 1, p. 312, 2009.

[55] Y. Yu, T. Wang, and R. J. Samworth, “A useful variant of the davis–kahan theorem for statisticians,” *Biometrika*, vol. 102, no. 2, pp. 315–323, 2014.