A quadratic programming approach to find faces in robust nonnegative matrix factorization

by

Sai Mali Ananthanarayanan

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Combinatorics and Optimization

Waterloo, Ontario, Canada, 2017

I hereby declare that I am the sole author of this report. This is a true copy of the report, including any required final revisions, as accepted by my examiners.

I understand that my report may be made electronically available to the public.

Abstract

Nonnegative matrix factorization (NMF) is a popular dimensionality reduction technique because it is easily interpretable and can discern useful features. For a given matrix $M \in \mathbb{R}^{n \times m}$ whose entries are nonnegative and an integer r smaller than both n and m, NMF is the problem of finding nonnegative matrices $A \in \mathbb{R}^{n \times r}$ and $W \in \mathbb{R}^{r \times m}$ such that $M = AW$. The matrix M could be noisy, in which case one seeks a robust algorithm that solves $M \approx AW$. The nonnegativity constraint in NMF has wide applications [19] in data science problems like document clustering [31, 41], facial feature extraction [25, 31], hyperspectral unmixing [32] etc.

Geometrically, the rows of M can be viewed as a set of points in $\mathbb{R}^m$. If we think of the rows of W as the vertices of an (unknown) W-simplex, then the data points lie in this W-simplex. Therefore, NMF asks us to deduce the vertices of the simplex given the data points.

NMF is a computationally hard problem [23] though certain assumptions like *separability* lead to polynomial time algorithms [1, 2, 35]. This assumes that all the vertices of the unknown simplex are already present as data points. In practice, this is not true in many settings [20]. Ge and Zou [17] assumed *subset separability* which uses higher dimensional structures and gave a polynomial time algorithm to find the NMF robustly. In this thesis, we effectively replace one of their key algorithms that finds faces. We show a quadratic programming based approach which is efficient and can be employed in practice. Under bounded noise, our algorithm finds the faces of the simplex which contain enough data points, thus helping in finding the NMF.

Acknowledgements

Thanks to my readers for their helpful comments on an earlier draft of the thesis and to Dr Rong Ge for answering my queries about his paper.

I would like to sincerely thank my advisor Dr Stephen Vavasis for his immense support and guidance over the last two years. He has been an inspiration, and I will miss his words of encouragement due to which I strived to learn more every week in a field new to me. Dr Henry Wolkowicz has been a tremendous mentor and he opened the gates to the optimization community to me. I owe my gratitude to Dr Joseph Cheriyan who led me to the C&O department and helped me settle in.

I thank the Associate Chairs of Graduate Studies Dr Jim Geelen and Dr Chaitanya Swamy, as well as Dr Alfred Menezes and the rest of the C&O department for the fantastic atmosphere and support.

"*Arpudhamaana nanbargal saerndhaa vetrigal kumiyumadaa*" in Tamil translates as "Victories will accumulate if you have great friends". They welcomed me with open hands and made my stay wonderful. Thanks to Vishnu, Nishad, Hemant, Anirudh, Sharat, Dhinakaran, Stefan, Jiale, Jimit, Priya, Karthik, Niranjan, Arvind, Abhinav, Cedric, Sanchit and everyone else. My parents and family are a source of unconstrained love 13423 km away.

To Varuna, who makes me infinitely happier everyday and whose support spurred me to work hard this year.

Dedication

This is dedicated to *Amma*, *Appa* and Seetha.

Table of Contents

List of Figures

Notation

Let $\mathbf{e}, \mathbf{e}_i, \mathbf{0}$ respectively denote the vector of all 1's, ith standard basis vector and vector of all zeros, all of of appropriate dimension. For $n \in \mathbb{N}$, We use $[n]$ as a shorthand for the set $\{1, 2, \ldots, n\}$. For any matrix $A \in \mathbb{R}^{n \times m}$, let A^i or $A(i, :)$ denote the ith row and $A(:, j)$ or A_j denote the jth column. For $i < j$, the submatrix of A with all rows, and columns between i and j is written by $A(:, i : j)$. Similarly, the submatrix of A consisting all columns, and rows between i and j is written by $A(i : j, :)$. The kth largest singular value of A (defined later) is denoted by $\sigma_k(A)$. We use bold letters for vectors, e.g., $\mathbf{p}$. For any subspace Q, let P_Q denote the projection matrix to Q.

Chapter 1

Introduction

Data observed in practice is often derived from multiple latent sources. One would like to infer the latent sources which are the mixture components of the data, along with the mixture distribution. A lot of linear dimensionality reduction (LDR) techniques are popular in data analysis and machine learning [19], one of which is nonnegative matrix factorization (NMF). Given a set of data vectors that are nonnegative, NMF extracts useful features and it is considered to be a dimensionality reduction technique. This technique has wide applications in data science [19, 21, 20].

Representing a set of n data points in $\mathbb{R}^m$ as a matrix $M \in \mathbb{R}^{n\times m}$, LDR usually entails computing a set of basis elements $\mathbf{w}_i \in \mathbb{R}^m, 1 \leq i \leq r < m$ whose linear combinations approximate the given data points. For each $1 \leq j \leq n$, the data point $M(j,:)$ is written as,

$$M(j,:) \approx \sum_{k=1}^{r} a_{j,k}\mathbf{w}_k \tag{1.1}$$

for some weights $\mathbf{a}_j = [a_{j,1}\ a_{j,2}\ \ldots\ a_{j,r}] \in \mathbb{R}^r$. Interpreting this as a LDR, the given m-dimensional data points are written out in a affine subspace of dimension r with the weight vectors $\mathbf{a}_j$'s providing the coordinates. Thus we are approximating a set of points in $\mathbb{R}^m$ using an affine subspace of lower dimension.

The problem of low rank matrix approximation is equivalent to this formulation, once we construct the weight matrix A such that the rows are the coordinates, $A(j,:) = \mathbf{a}_j$ for $1 \leq j \leq n$ and the basis matrix W such that the basis elements form the rows, $W(k,:) = \mathbf{w}_k$

for $1 \leq k \leq r$. Then the LDR (1.1) is equivalent to finding A and W so that,

$$M \approx AW \quad \text{for} \quad M \in \mathbb{R}^{n \times m}, A \in \mathbb{R}^{n \times r} \text{ and } W \in \mathbb{R}^{r \times m}. \tag{1.2}$$

Low rank matrix approximations are of interest since they help in extracting relevant information, especially in large sets of data (See [44],[20],[19] for further discussion). Some examples of application include data analysis [29], control [33], machine learning and data mining [15], graph theory [9] etc.

There are certain key issues we need to address in these models (1.1) and (1.2).

1. *Approximation measure*: There are many ways in literature to measure the error $M - AW$. Depending on the noise model, this could be the Frobenius norm $\|M - AW\|_F^2 = \sum_{i,j}(M-AW)_{ij}^2$ which leads to principal component analysis (PCA). Such a least squares error approach is popular due to the implicit assumption that the noise is Gaussian [19], and the approximation can be efficiently found using truncated singular value decomposition (SVD). It can also be shown that the resulting minimization problem in A and W has all local minima to be global [20].

2. *Assumptions on A and W*: One can make different assumptions based on the problem to be solved. Truncated SVD and PCA do not make any assumptions on A and W. The k-means problem requires finding a set of *centroids* in the same dimensional space as the data, so that the sum of the distances between each data point and the closest centroid is minimized. This is the same as constraining each row of A to be an element of the standard basis, so that the rows of W are the centroids. Problems like sparse PCA [12] ask for low rank matrix decompositions which assume that A and W are sparse. Independent Component Analysis [10] requires the columns of W to be independent. NMF is one such problem which imposes a constraint on A and W, namely the matrices are *componentwise nonnegative*. Thus, we require the nonnegative matrix M to be decomposed as $M \approx AW$ where $A \geq 0$ and $W \geq 0$.

Further discussions in detail can be found in [19],[20] and [22].

1.1 The NMF model

NMF has been studied since 1979 [3, 6, 7] and formally named in 1994 by Paatero and Tapper [37]. Subsequent works of Lee and Seung [31] as well as Donoho and Stodden [14] led to more interest in the problem. We give a formal definition of NMF.

Definition 1. *Suppose $M \in \mathbb{R}^{n\times m}$ such that $M \geq 0$. Suppose for a given $r < \min(n, m)$, we find matrices $A \in \mathbb{R}^{n\times r}$ such that $A \geq 0$ and $W \in \mathbb{R}^{r\times m}$ such that $W \geq 0$ satisfying*

$$M = AW.$$

Then (A, W) is called an exact *nonnegative matrix factorization of M. The integer r is called the* inner dimension *of the factorization and the smallest possible inner dimension is called the* nonnegative rank *of M, denoted as $rank_+(M)$.*

Finding (A, W) such that of $M = AW$ holds exactly is referred to in literature as *exact NMF* [20]. If equality is not expected, then NMF can be written as the following nonconvex optimization problem for a nonnegative matrix $M \in \mathbb{R}^{n\times m}$:

$$\text{NMF}: \quad \min_{A\in\mathbb{R}^{n\times r}, W\in\mathbb{R}^{r\times m}} \|M - AW\|_F^2 \quad \text{such that} \quad A \geq 0,\ W \geq 0. \tag{1.3}$$

There is an implicit assumption of a Gaussian noise model in the above formulation which does not apply in many practical settings. There are also other possible objective functions (Kullback-Leibler divergence for text mining [8], earth mover's distance for computer tasks [39] etc.) In the rest of this section, we explore some features of NMF and provide a brief survey of NMF algorithms relevant to our problem.

- Notice that $M = IM = MI$ is a trivial factorization (I is the identity matrix). This implies that the nonnegative rank satisfies $rank(M) \leq rank_+(M) \leq \min(n, m)$. The nonnegative rank is not less than the usual (real) rank of the matrix. Choosing an inner dimension to solve NMF is usually tricky. This is the problem of order model selection and there are certain techniques like looking at the singular value spectrum of M, trial and error (trying different values and choosing the one giving best results for the chosen problem) etc. that are used [19].

- The NMF problem is *ill-posed*, meaning there could be multiple solutions that factorize the given matrix M. If we have (A, W) as a solution, then so is $(AH, H^{-1}W)$ for any matrix H such that both components AH and $H^{-1}W$ are nonnegative. If H is the permutation of a positive diagonal matrix, then the new solution is simply a scaling and permutation of some of the rank-one factors. If H is not such a permutation, then it could lead to different interpretations of the solution, for example a different set of topics and classifications in text mining [19]. We address the issue of uniqueness in more detail in Chapter 2.

- NMF is considered to be easily interpretable and automatically extracts sparse factors. Looking at the first order optimality conditions of the minimization problem (1.3), one can see that the stationary points A and W contain zero entries. This makes the NMF problem more interpretable and sparse, i.e., it yields the 'true' latent factors A and W. Basis elements are similar to the given data due to the nonnegativity constraint and weights can be thought of as mixture component coefficients or 'activation' coefficients [20]. The weights being nonnegative allows us to think of an additive construction of the data points from the basis elements, leading to a parts-based and sparse representation of the data [31, 22].

- Another reason for the popularity of NMF arises from the wide applications of the nonnegativity constraint. These include topic recovery and document clustering in text mining [31, 41], facial feature extraction in image processing [25, 31], hyperspectral unmixing [32], computational biology [13], music analysis [16], collaborative filtering [34], community detection [47] etc.

- As simple as the formulation is, NMF is a computationally hard problem. Vavasis [45] proved that determining whether $rank(M) = rank_+(M)$ is NP-hard. There has been a surge of interest in polynomial time algorithms with proven error bounds. Arora et al., [2] showed for a subclass of NMF (under certain assumptions), there is an efficient algorithm ($O(mn)^{2^r r^2}$) which was later improved by Moitra [35] to $O(mn)^{r^2}$, which is polynomial in m, n for a fixed r. Further works can be found in [21] and [38]. There are heuristic algorithms with no guarantee on convergence [31] though they have been successful in many applications. Many of these run in $O(pnr)$ [19]. Using standard nonlinear optimization techniques to find locally optimal solutions is also a common approach but comes with no theoretical guarantee [22].

- There are practical NMF algorithms with strong theoretical guarantees [18, 22]. However, many of these techniques use the notion of *separability*, which is a strict condition that requires all rows of W to be already present in M. This was first introduced by Donoho and Stodden [14] and has been a popular assumption but not too common in practice. Recently, there is some work on NMF with little assumptions on the data except for the noise model e.g heavy noise model by Bhattacharya et al. [4]. Javadi and Montanari [28] use an archetypal analysis approach introduced by Cutler and Breiman [11].

1.2 Our contributions

We consider the work of Ge and Zou [17] where the notion of *subset separability* is introduced. This is a milder assumption than separability, but nonetheless their algorithm uses multiple convex programs to solve the problem. Such algorithms requiring linear and complex programs are hard to scale [19] due to complexity issues. Our contribution replaces the most expensive algorithm in their work, and we provide a quadratic programming (QP) based approach for the same. Our model works for bounded noise under similar assumptions to [17] for the problem.

Structure of the thesis: The organization of the rest of the thesis is as follows. In Chapter 2, we discuss preliminaries, define some of the concepts as well as provide lemmas used to derive our results. In Chapter 3, we explain the problem being solved and lay out the proposed QP model. In Chapter 4, we prove some theorems to show our model working in noiseless and noisy settings, as well as discuss the post processing to extract the required information out of the model. In Chapter 5, we perform some computational tests on simulated data which shows experimental proof of our model, and we present our conclusions.

Chapter 2

Preliminaries and Background

2.1 Algebraic preliminaries

2.1.1 Lagrangian Dual

Consider a quadratic programming problem in the unknown $\mathbf{y} \in \mathbb{R}^n$, of the form:

$$\begin{aligned} \text{Primal}: \min \quad & \mathbf{c}_0^T \mathbf{y} + \frac{1}{2} \mathbf{y}^T B \mathbf{y} \\ \text{subject to} \quad & Q\mathbf{y} \geq \mathbf{b}_0 \end{aligned} \tag{2.1}$$

where $\mathbf{c}_0 \in \mathbb{R}^n, B \in \mathbb{R}^{n \times n}, Q \in \mathbb{R}^{m \times n}$ and $\mathbf{b}_0 \in \mathbb{R}^m$. Assume B is a symmetric positive semidefinite matrix. We wish to write down the dual of the problem (2.1).

We use the standard minmax Lagrangian dual formulation, as follows. Let $\mathbf{d} \in \mathbb{R}^m$ be the dual variable. Then the Lagrangian formulation can be written as:

$$\begin{aligned} & \min_{\mathbf{y}} \max_{\mathbf{d} \leq \mathbf{0}} \left\{ \mathbf{c}_0^T \mathbf{y} + \frac{1}{2} \mathbf{y}^T B \mathbf{y} + (Q\mathbf{y} - \mathbf{b}_0)^T \mathbf{d} \right\} \\ & \geq \max_{\mathbf{d} \leq \mathbf{0}} \min_{\mathbf{y}} \left\{ \mathbf{c}_0^T \mathbf{y} + \frac{1}{2} \mathbf{y}^T B \mathbf{y} + (Q\mathbf{y} - \mathbf{b}_0)^T \mathbf{d} \right\} \quad (\because \text{ Minimax inequality}) \\ & = \max_{\mathbf{d} \geq \mathbf{0}} \min_{\mathbf{y}} \left\{ \mathbf{c}_0^T \mathbf{y} + \frac{1}{2} \mathbf{y}^T B \mathbf{y} - (Q\mathbf{y} - \mathbf{b}_0)^T \mathbf{d} \right\} \\ & = \max_{\mathbf{d} \geq \mathbf{0}} \min_{\mathbf{y}} \left\{ (\mathbf{c}_0 - Q^T \mathbf{d})^T \mathbf{y} + \frac{1}{2} \mathbf{y}^T B \mathbf{y} + \mathbf{b}_0^T \mathbf{d} \right\} \end{aligned} \tag{2.2}$$

The Minimax inequality in (2.2) is known to be tight ("strong duality") in the case of convex quadratic programming [5]. If $\mathbf{c}_0 - Q^T\mathbf{d}$ is not in the range of B, then the inner problem is unbounded. Otherwise, in the inner minimization problem in $\mathbf{y}$ the minimizer $\mathbf{y}^*$ satisfies the first-order optimality condition,

$$B\mathbf{y}^* = -(\mathbf{c}_0 - Q^T\mathbf{d}). \tag{2.3}$$

Subject to the constraints $\mathbf{d} \geq 0$ and $Q^T\mathbf{d} - B\mathbf{y}^* = \mathbf{c}_0$, the problem (2.2) is a maximization problem in the variables $\mathbf{d}$ and $\mathbf{y}^*$ with the objective function,

$$\begin{aligned}
&(\mathbf{c}_0 - Q^T\mathbf{d})^T\mathbf{y}^* + \frac{1}{2}(\mathbf{y}^*)^T B\mathbf{y}^* + \mathbf{b}_0^T\mathbf{d} \\
&= (\mathbf{c}_0 - Q^T\mathbf{d})^T\mathbf{y}^* - \frac{1}{2}(\mathbf{y}^*)^T(\mathbf{c}_0 - Q^T\mathbf{d}) + \mathbf{b}_0^T\mathbf{d} \quad (\because (2.3)) \\
&= \frac{1}{2}(\mathbf{c}_0 - Q^T\mathbf{d})^T\ \mathbf{y}^* + \mathbf{b}_0^T\mathbf{d} \\
&= -\frac{1}{2}(\mathbf{y}^*)^T B\ \mathbf{y}^* + \mathbf{b}_0^T\mathbf{d}. \quad (\because (2.3))
\end{aligned} \tag{2.4}$$

Let $\mathbf{z} = -\mathbf{y}^*$ be a dual variable. Then (2.3) becomes the constraint

$$B\mathbf{z} + Q^T\mathbf{d} = \mathbf{c}_0. \tag{2.5}$$

With the objective function (2.4) and the constraints above, the problem becomes,

$$= \max_{\substack{\mathbf{d}\geq 0 \\ B\mathbf{z}+Q^T\mathbf{d}=\mathbf{c}_0}} \left\{ -\frac{1}{2}\mathbf{z}^T B\mathbf{z} + \mathbf{b}_0^T\mathbf{d} \right\}.$$

Therefore for the dual variables $\mathbf{d} \in \mathbb{R}^m$ and $\mathbf{z} \in \mathbb{R}^n$, the dual problem to (2.1) is

$$\begin{aligned}
\text{Dual}: \max \quad & -\frac{1}{2}\mathbf{z}^T B\mathbf{z} + \mathbf{b}_0^T\mathbf{d} \\
\text{subject to} \quad & B\mathbf{z} + Q^T\mathbf{d} = \mathbf{c}_0 \\
& \mathbf{d} \geq \mathbf{0}.
\end{aligned} \tag{2.6}$$

Remark: How we deal with the dual variable $\mathbf{z}$ depends on B, the matrix corresponding to the quadratic term. If $\text{rank}(B) = 0$, then $\mathbf{z}$ drops out of (2.6) and this is now a Linear Program (LP). If B is of full rank, then $\mathbf{z}$ can be eliminated by solving the constraint (2.5) by inverting B. Finally if $0 < \text{rank}(B) < n$, then $\mathbf{z}$ can be eliminated by the introduction of new notation. This case is applicable to a QP we use in Chapter 4 and the details can be found there.

2.1.2 Singular Values and Norms

We prove a few lemmas about singular values and norms, that are used in our analysis. We start with the definition of singular value decomposition, given in Golub and Van Loan [24].

Definition 2 (Singular Value Decomposition)**.** *Suppose $A \in \mathbb{R}^{n\times m}$. Then we can factorize A as follows:*

$$A = U\Sigma V^T, \tag{2.7}$$

where $U \in \mathbb{R}^{n\times n}$ satisfies $U^TU = I$, $V \in \mathbb{R}^{m\times m}$ satisfies $V^TV = I$, and $\Sigma \in \mathbb{R}^{n\times m}$ is a diagonal matrix with the ith diagonal entry σ_i for some set of reals $\{\sigma_1, \ldots, \sigma_{\min(n,m)}\}$ satisfying

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_{\min(n,m)} \geq 0.$$

The elements of the set $\{\sigma_1, \ldots, \sigma_{\min(n,m)}\}$ are called the singular values *of A and the decomposition (2.7) is the* singular value decomposition *(SVD) of A.*

More details of SVD and its properties can be found in [24, 48, 5]. In particular, the largest singular value σ_1 is often denoted as $\sigma_{\max}$ and is equal to the 2-norm of the matrix A,

$$\sigma_{\max}(A) = \sup_{\substack{\mathbf{y}\in\mathbb{R}^n \\ \mathbf{y}\neq\mathbf{0}}} \frac{\|A\mathbf{y}\|}{\|\mathbf{y}\|} = \sup_{\substack{\mathbf{y}\in\mathbb{R}^n \\ \|\mathbf{y}\|=1, \mathbf{y}\neq\mathbf{0}}} \|A\mathbf{y}\| = \|A\|_2 . \tag{2.8}$$

Denote the smallest or minimum singular value of a matrix A by $\sigma_{\min}$. The smallest singular value is positive only for matrices of full rank. For a square matrix $A \in \mathbb{R}^{n\times n}$ of full rank, the *condition number* is defined as,

$$\text{cond}(A) = \|A\|_2 \left\|A^{-1}\right\|_2 = \frac{\sigma_{\max}(A)}{\sigma_{\min}(A)}. \tag{2.9}$$

We state the following theorem by Eckart and Young without proof. This is given, for example, as Theorem 2.5.3 in [24]:

Theorem 3. *Let $A \in \mathbb{R}^{u\times v}$ and let k be an integer satisfying $k < r =$ rank(A). Then,*

$$\min_{rank(B)\leq k} \|A - B\|_2 = \sigma_{k+1}$$

where σ_{k+1} is the $(k+1)$th largest singular value of A.

The above theorem gives a good characterization of any singular value of the matrix. The following lemmas are standard interlacing results and can be found, for example, in Golub and Van Loan [24].

Lemma 4. *Let $A \in \mathbb{R}^{u \times v}$. Then for any $1 \leq u' \leq u$ and $1 \leq v' \leq v$,*

$$\|A(1:u', 1:v')\|_2 \leq \|A\|_2 .$$

In other words, the 2-norm of a matrix is at least the 2-norm of a submatrix.

Proof. Consider $A(:, 1:v')$ first. Then,

$$\|A(:, 1:v')\|_2 = \sup\{\|A(:, 1:v')\mathbf{x}\| : \|\mathbf{x}\| = 1\}.$$

If $\mathbf{x}^*$ is the argument where the maximum is attained above then $\|\mathbf{x}^*\| = 1$ and,

$$\begin{aligned}
\|A(:, 1:v')\|_2 &= \|A(:, 1:v')\mathbf{x}^*\| \\
&= \left\| A \begin{bmatrix} \mathbf{x}^* \\ \\ \mathbf{0} \end{bmatrix} \right\| \\
&\leq \sup\{\|A\mathbf{x}\| : \|\mathbf{x}\| = 1\} \quad \left(\because \begin{bmatrix} \mathbf{x}^* \\ \\ \mathbf{0} \end{bmatrix} \text{ is a candidate for sup} \right) \\
&= \|A\|_2 \\
\implies \|A(:, 1:v')\|_2 &\leq \|A\|_2 .
\end{aligned}$$

Similarly, one can see that $\|A(1:u', 1:v')\|_2 \leq \|A(:, 1:v')\|_2$ and this completes the proof. □

We prove a lemma about the singular value of a submatrix.

Lemma 5. *If $A \in \mathbb{R}^{u \times v}$ and let $\sigma_i(B)$ denote the ith singular value of any matrix B. Then,*

$$\sigma_i(A(1:u', 1:v')) \leq \sigma_i(A)$$

for all $1 \leq u' \leq u, 1 \leq v' \leq v$ and $1 \leq i \leq \min(u, v)$. In other words, the ith singular value of any matrix is at least the ith singular value of any of its submatrices.

Proof. From Theorem 3, for any i,

$$\sigma_i(A) = \min\{\|A - B\|_2 : \text{ rank}(B) \leq i - 1\}. \tag{2.10}$$

Let B^* be the minimizer above, then $\text{rank}(B^*(1:u', 1:v')) \leq \text{ rank}(B^*) = i - 1$. Hence,

$$\begin{aligned}
\sigma_i(A(1:u', 1:v')) &\leq \left\| A(1:u', 1:v') - \underbrace{B^*(1:u', 1:v')}_{\text{Candidate for optimizer}} \right\| \\
&\leq \|A - B^*\| \quad (\because \quad \text{Lemma 4}) \\
&= \sigma_i(A) \\
\implies \sigma_i(A(1:u', 1:v')) &\leq \sigma_i(A)
\end{aligned}$$

as desired. □

Lemma 6. *Let $A \in \mathbb{R}^{u \times v}$ and $D \in \mathbb{R}^{w \times u}$. Then,*

$$\|DA\|_2 \leq (\max_i |D_{ii}|) \|A\|_2 .$$

Proof.

$$\begin{aligned}
\|DA\|_2 &= \sup\{\|DA\mathbf{x}\|_2 : \|\mathbf{x}\| = 1\} \\
&= \sqrt{D_{11}^2 \mathbf{y}_1^2 + \ldots + D_{vv}^2 \mathbf{y}_v^2} \quad (\text{For } \mathbf{y} = A\mathbf{x}) \\
&\leq (\max_i |D_{ii}|) \sqrt{\mathbf{y}_1^2 + \ldots + \mathbf{y}_v^2} \\
&\leq \sup\{(\max_i |D_{ii}|) \|A\mathbf{x}\|_2 : \|\mathbf{x}\| \leq 1\} \\
&= (\max_i |D_{ii}|) \|A\|_2 .
\end{aligned}$$

□

Note that in the second line of the proof, supremum is achieved by compactness.

2.1.3 Simplex

Definition 7 (Simplex)**.** *Given a set of affinely independent points* $\{\mathbf{x}_i\}_{i=0}^n \subset \mathbb{R}^m$*, the simplex* $\mathcal{U}$ *defined by these points is*

$$\mathcal{U} = \left\{ \sum_{i=0}^{n} \lambda_i \mathbf{x}_i : \sum_{i=0}^{n} \lambda_i = 1,\ \boldsymbol{\lambda} = [\lambda_0, \lambda_1, \ldots, \lambda_n] \geq \mathbf{0} \right\}.$$

The points $\{\mathbf{x}_i\}_{i=0}^n$ are the vertices of the simplex of dimension n and the simplex itself is just the convex hull of its vertices.

Definition 8 (Face of a simplex)**.** *A face* $S \subset [n] \cup \{0\}$ *of a simplex is the convex hull of the vertices* $\{\mathbf{x}_j : j \in S\}$*.*

Note that we may refer to both the indices of the subset of vertices as well as the convex hull of the subset by the same term *face*. This is clear from context. If we take the affine hull of all the vertices of a face, we get a unique affine subspace, whose dimension is the dimension of the face.

2.2 Geometric view of NMF

The NMF problem has a nice geometric interpretation that is often useful when developing algorithms. In the case of exact NMF, one can assume without loss of generality that the zero columns of M and W can be removed. Consider the diagonal matrices D_M whose jth diagonal entry is $\|M(j,:)\|_1$ and D_W whose jth diagonal entry is $\|W(j,:)\|_1$. From $M = AW$, we can scale all the matrices so that the new problem becomes,

$$\underbrace{D_M^{-1} M}_{M'} = \underbrace{D_M^{-1} A D_W}_{A'} \underbrace{D_W^{-1} W}_{W'}. \tag{2.11}$$

All rows of M', A' and W' now sum to 1. Therefore without loss of generality, we can always normalize the NMF. This leads to an interesting observation: All the rows of M are in the convex hull of the rows of W [20, 17]. If we think of the rows of W as the vertices of an (unknown) W-simplex, then the data points lie in this W-simplex. Given the data points, NMF asks us to deduce the vertices of the simplex. This is also related to the Nested Polytope Problem in computational geometry [20].

The NMF problem is ill-posed, and there could be multiple factorizations possible. Given a factorization $M = AW$ with nonnegative matrices A and W, one could perturb the vertices of W and maintain all the data points in M in the convex hull of the simplex. If we wish to find a unique factorization (up to scaling and permutation of the rows of W), then additional constraints need to be imposed. Sparsity [42] and minimum determinant [40] address the question of uniqueness. The aforementioned geometric viewpoint of NMF was provided by Thomas [43] and Chen [7]. There is some work on necessary conditions for uniqueness [36, 30]. Donoho and Stodden [14] introduced a condition called *separability*, which is a sufficient condition for uniqueness.

Definition 9 (Separable NMF)**.** *A NMF is separable if for each $\{W^i\}_{i=1}^r$, there exists a row M^{j_i} $(1 \leq j_i \leq n)$ of M such that $M^{j_i} = W^i$.*

Geometrically, this means that the vertices of the W-simplex that we are trying to find, are already present among the given data points. This is also equivalent to saying that the rows of the matrix A contain a permutation of the identity matrix. A more detailed analysis can be found in [27, 19, 20]. Though this seems like a simple enough geometric problem, its robustness to noise is much more difficult. There is plenty of work in NMF based on the separability condition or its variations [14, 2, 38, 1] where they try to set a noise bound that can be tolerated while still being able to recover the vertices, up to some error. In the presence of bounded noise, NMF can be solved in polynomial time with respect to mn, and r under the separability assumption [2, 1, 35]. Nevertheless, separability is not always found in practice and is a strict assumption. Some instances of practical occurences of separability are document classification, Raman spectral analysis and hyperspectral unmixing [20],

Our work is based on Ge and Zou [17]. They introduced the notion of *subset separability* that makes uses of higher dimensional structures of the simplex, not just the vertices. We provide the necessary background to discuss their algorithm for the subset separable case, which is defined in the next section.

2.3 Subset separability

Definition 10 (Filled Face)**.** *Given a factorization $M = AW$, a face of the W-simplex is said to be **filled** if there is at least one data point M^i in the relative interior of the face (or trivially when the face itself is a single vertex, that vertex is a data point).*

We can now supply the definition of *subset-separable* NMF, a milder assumption than separability [17].

Definition 11 (Subset-separable NMF)**.** *Given a factorization $M = AW$, the NMF is said to be subset separable if there exists a set of filled faces $\{S_i\}_{i=1}^k$ with $S_i \subset [r]$ such that for each $j \in [r]$, we have a subset $\{S_{j_i}\}_{i=1}^{k_j} \subset \{S_i\}_{i=1}^k$ whose intersection is exactly the vertex j.*

Geometrically, this means that every vertex is the intersection of some set of filled faces. Note that subset separability is equivalent to the property that for every $j_1 \neq j_2 \in [r]$, there exists a row i of A such that $A_{i,j_1} = 0$ and $A_{i,j_2} \neq 0$. A subset-separable NMF is a separable NMF (Definition 9) in the special case where the set of filled faces are exactly the vertices of the simplex.

As discussed previously, the NMF problem is ill-posed and aiming for a unique factorization is of interest. Among other features, finding a *minimal volume* solution holds an appeal. This means it is impossible to displace a single vertex such that the volume is lowered and still obtain a valid factorization (i.e., all data points still lying inside the simplex). Intuitively, this means there are some data points on the boundary of the simplex.

The following lemma due to Ge and Zou [17] shows that the filled faces being subset separable is a necessary (but not sufficient) condition for W to be volume minimizing.

Lemma 12. *Suppose $M = AW$ where W is of rank r and minimal volume. Then W is subset-separable.*

Proof. Suppose the factorization is not subset-separable. Then there exists $j_1 \neq j_2 \in [r]$, such that for every row i of A, either both A_{i,j_1} and A_{i,j_1} are both zero or both non-zero. Suppose we define new matrices A' and W' as follows, for some $\varepsilon \in [0, 1]$:

$$A_j^{'} = \begin{cases} \frac{1}{1-\varepsilon} A_j & j = j_1 \\ A_j - \frac{\varepsilon}{1-\varepsilon} A_{j_1} & j = j_2 \\ A_j & \text{otherwise} \end{cases} \tag{2.12}$$

$$(W')^j = \begin{cases} (1-\varepsilon) W^{j_1} + \varepsilon W^{j_2} & j = j_1 \\ W^j & \text{otherwise.} \end{cases} \tag{2.13}$$

Clearly, we found another NMF, $M = A'W' = AW$, since W is still nonnegative for the given range of ε. The support of A_{j_1} and A_{j_2} are the same, therefore there is a value of ε

that leaves A'_{j_2} nonnegative. Therefore, we have a valid NMF of M with only one vertex of the simplex changed. The ratio of the volume of W' to the volume of W is $1-\varepsilon < 1$, which is a contradiction to the minimal volume assumption. □

The *filled faces* defined above lie on the boundary of the convex hull of the data points. A general class of filled faces, called *properly filled faces* are computationally efficient to find [17] and we define them as follows.

Definition 13 ((N, H, γ) Properly Filled Faces)**.** *Let N be a positive integer, $H > 0$ and $\gamma > 0$ be real numbers. A set of faces $S_1, \ldots, S_k \in [r]$ of W is (N, H, γ) properly filled if:*

1. *(Center point.) For any filled face $|S_i| > 1$, there is a row i^* of A with support S_i. This row i^* is in the convex hull of the other rows of A. Moreover, there exists a convex combination $M^{i^*} = \sum_{i \in [n] \setminus i^*} \alpha_i M^i$, such that $M^{i^*} = \sum_{i \in [n] \setminus i^*} \alpha_i M^i (M^i)^T$ has rank $|S_i|$ with smallest singular value not less than γ. M^* is called the **center of this face**.*
2. *For any set $|S_i| > 1$, at least N rows of A have support S_i.*
3. *(General Positions Property.) For any affine subspace Q of dimension $t \in (1, r)$, the existence of at least N rows of M in an ε neighborhood of Q implies the existence of a non-singleton set S_i with corresponding affine subspace Q_i with the following property. Let $Q = \mathbf{v}_0 + L$ where L is a linear subspace, and $Q_i = \mathbf{v}_1 + L_i$ where L_i is a linear subspace such that $\mathbf{v}_0$ and $\mathbf{v}_1$ satisfy $\|\mathbf{v}_0 - \mathbf{v}_1\| \leq H\varepsilon$. Additionally for an orthonormal basis V_{L_i}, $\|P_{L^\perp} V_{L_i}\| \leq H\varepsilon$ holds.*

The first condition means that while expressing the center point as a convex combination of the other points, only those points making a nonzero contribution in the convex combination lie in the same face as the center point. The second condition means that properly filled faces are unlike other subspaces, i.e., they are faces of the true solution since they contain many points. The third condition intuitively means that every subspace contains many points close to a properly filled face. Points that are not in the lower dimensional faces $S_1, \ldots, S_k$ are in general positions, so that a random subspace and a properly filled face can be distinguished.

2.4 Face-Intersect algorithm

In their paper, Ge and Zou [17] produce what they call the *Face-Intersect* algorithm to compute the NMF under the subset separability assumption. If $M = AW$ is subset sepa-

rable by properly filled faces $S_1, \ldots, S_k$, then their algorithm computes A and W in a time polynomial in n, m and r. However in practical settings, data points often contain noise, in this case every row of M has an added noise component (say with bound ε). They show that if $M = AW$ is subset separable by (N, H, γ) properly filled faces, and W has all rows norm bounded by 1 and rth singular value of W not too small, then the Face-Intersect algorithm solves the NMF robustly in polynomial time in n, m and r.

The gist of the Face-Intersect algorithm is as follows:

1. Find all subspaces corresponding to the properly filled faces $S_1, \ldots, S_k$ (non-singletons).
2. Systematically take the intersection of subspaces to obtain the set of intersection vertices.
3. Now the remaining vertices are the singleton sets. Run an algorithm to find these anchors (singleton points).
4. This gives us W and using M, compute A.

We elaborate on each step in the following sections.

2.4.1 Finding non-singleton properly filled faces

The problem of finding filled faces occurs in the literature (subspace clustering [46], subspace recovery [26]) but the authors of [46] and [26] make strong assumptions on subspace independence, among other conditions. These assumptions do not fit well with our problem. Moreover, our problem contains other useful information not used in these methods. For example, the filled faces are on the boundary of the convex hull of data, a fact we can utilize to our advantage.

As stated earlier, the first condition of Definition 13 means that if we write the center point as a convex combination of the other points, only those points making a positive contribution in the convex combination lie in the same face as the center point. Then to get the subspace, one only needs to take the affine hull of these points. If we have noisy data points with ε small compared to γ (which is smaller than the least singular value of the matrix in the condition), then one can find this *nice* convex combination that gives the data points using an iterative procedure [17]. Thus, the algorithm finds a properly filled face given the center point.

Algorithm 1 Finding one properly filled face given the center point

1: **Input**: Set of points $\{\mathbf{x}_i\}_{i=1}^h$ in $\mathbb{R}^m$ and a center point $\mathbf{x}_0 \in \mathbb{R}^m$.
2: **Output**: Subspace L corresponding to a properly filled face that contains $\mathbf{x}_0$.
3: Initialize $L = \emptyset$ and $Ldimchange > 0$.
4: **while** $Ldimchange > 0$ **do**
▶ Only run if dimension of subspace increases.
5: Solve for the weights $\mathbf{w}$ as unknown and B_L being any orthonormal basis of L:

$$\begin{aligned} \mathbf{w}^* = \text{argmax} \quad & tr(P_{L^\perp} \sum_{i=1}^{h} w_i \mathbf{x}_i \mathbf{x}_i^T P_{L^\perp}) \\ \text{subject to} \quad & w_i \geq 0 \ \ \forall\, i \in [h] \\ & \sum_{i=1}^{h} w_i = 1 \\ & \left\| \mathbf{x}_0 - \sum_{i=1}^{h} w_i \mathbf{x}_i \right\| \leq 2\varepsilon \\ & diag\Big(B_L^T \big(\sum_{i=1}^{h} w_i \mathbf{x}_i \mathbf{x}_i^T \big) B_L \Big) \geq \gamma/2 \end{aligned} \tag{2.14}$$

6: Let $d = dim(L)$. Set L' to be the span of all singular vectors of $\sum_{i=1}^h w_i^* \mathbf{x}_i \mathbf{x}_i^T$ whose singular values are bigger than $\gamma/2d$.
7: Set $Ldimchange = dim(L') - dim(L)$ and $L = L'$.
8: **end while**

Notice that the objective function of (2.14) is the component of $\sum_{i=1}^n w_i \mathbf{x}_i \mathbf{x}_i^T$ outside the subspace L. The constraints ensure we maintain the center point as a convex combination of the other points while maintaining large singular values for the current subspace. The authors [17] show that Algorithm 1 converges quickly, with the dimension of L increasing until we terminate when it does not increase anymore. It arrives at a *nice* convex combination from which we can extract the subspace containing the properly filled face. They only make the following assumptions:

1. The noise in the data is bounded by ε.
2. The unknown W-simplex does not have singular values too small.

3. The center point $\mathbf{x}_0$ belongs to a properly filled face created by d vertices of the W-simplex.

The algorithm could generate false positives, which are subspaces that do not correspond to any properly filled faces. However, these subspaces do not contain enough data points, because according to the *general positions* property (Condition 3 in Definition 13), any subspace with enough number of points is close to a properly filled face. One can use this to weed out these false positives.

Our contribution: This thesis focuses on a replacement for Algorithm 1. For the sake of completeness, we present the rest of Ge and Zou's method [17] to show how Algorithm 1 is used to compute an NMF.

Finding the properly filled non-singleton faces is accomplished with the following algorithm:

Algorithm 2 Finding all properly filled faces

1: **Input**: Noisy data points $M \in \mathbb{R}^{n \times m}$ with a subset-separable factorization and which contains (N, H, γ) properly filled faces.
2: **Output**: All non-singleton properly filled faces.
3: **for** $i = 1, 2, \ldots, n$ **do**
4: Set the center point $\mathbf{x}_0 = M^i$ and $\mathbf{x}_1, \ldots, \mathbf{x}_h$ as the other data points.
5: Run Algorithm 1 that finds a subspace L.
6: Check if $\dim(L) < r$ and there are at least N points within a small distance, then add L to the collection of subspaces.
7: **end for**
8: Suppose we find two subspaces L_1, L_2 of different dimension (say $\dim(L_1) < \dim(L_2)$) in the collection and the component of L_1 along the orthogonal complement of L_2 is small, then L_2 is a false positive and can be removed. This is by the general positions property in Definition 13.
9: Among the remaining subspaces in the collection, there could be some close to each other and we merge them.

2.4.2 Obtaining vertices by taking subspace intersections

Suppose we have a noisy NMF which is subset-separable and contains (N, H, γ) properly filled faces, Algorithm 2 gives us noisy subspaces L_i which are close to the true subspaces.

Each subspace is non-singleton and is the affine hull of some subset S_i of the vertices of the W-simplex. Recall the existence of filled faces $S_1, \ldots, S_k \subset [r]$ in Definition 11 of subset-separability. Assume the first h are non-singleton. Then our goal is to find the intersection vertices, each of which is a unique intersection of the elements of some subset of $\{S_1, \ldots, S_h\}$ [17].

Problem of set intersections: Given some sets $S_1, \ldots, S_h \subset [r]$, we wish to find an unknown subset of vertices $P \subset [r]$ with the property that for each vertex $i \in P$, there are some $\{S_{i_k}\}$ satisfying $i = \cap_k S_{i_k}$.

It is inefficient to simply take all possible combinations of the sets and we could also get vertices that are not intersection vertices. Moreover, we only have access to subspaces L_i that correspond to each set S_i. However, taking the intersection of subspaces is the same as taking the intersections of the sets, i.e., intersection of subspaces leads to the affine hull of intersection of the corresponding faces. Similarly, affine hull of subspaces are akin to union of sets. The dimension of the subspace L_i is one less than the size of the set S_i. The following algorithm by Ge and Zou [17] obtains all the intersection vertices. As we observed above, working with sets or subspaces makes little difference since we can perform the same operations.

Algorithm 3 Finding intersection vertices from subsets

Input: Sets $S_1, \ldots, S_h$ which are not singleton.
Output: $P \subset [r]$ ▶ The set of all intersection vertices of the W-simplex.
Set $R = \emptyset$. ▶ Set containing vertices we have already found.
Set $S = \emptyset$. ▶ Result of systematically taking set intersections.
for $i = 1, 2, \ldots, r$ **do**
    $S = [r]$.
    for $j = 1, \ldots, h$ **do**
        if $|S \cap S_j| < |S|$ and $S \cap S_j \not\subset R$ **then**
            ▶ Second condition ensures we avoid finding vertices already in R.
            $S = S \cap S_j$.
        end if
    end for
    $R = R \cup S$.
        ▶ If we have $(S \setminus R) \cap P = \emptyset$, add S to R and remove the vertices not in P.
    Add S to P if $|S| = 1$. ▶ Found an element of P.
end for

Observe that R increases in size by at least one in every iteration and hence the algorithm terminates in r iterations. The smallest singular value of W should not be too small, else this proof may not work. Under this assumption, we can find all the vertices in P. Only the singleton sets are left.

2.4.3 Finding singleton sets

We only need to find vertices in $[r] \setminus P$, which exactly correspond to the singleton sets ($|S_i| = 1$). If $P = \emptyset$, this is similar to the separability assumption discussed earlier. The singleton vertices appear in the data points and an algorithm is adapted from Gillis and Vavasis [22].

Algorithm 4 Finding the singleton sets remaining

Input: Noisy data M, intersection vertices found so far $W^1, \ldots, W^{|P|}$
Output: $W^{|P|+1}, \ldots, W^r$ ▶ Remaining vertices of the simplex.
for $i = |P| + 1$ to r **do**
    Let $L = \text{affinehull}\{W^1, \ldots, W^{i-1}\}$
    Set $W^i = M^j$, where $j = \text{ argmax } \|P_{L^\perp} M^j\|$.
end for

Thus we have found all the vertices of the W-simplex. This completes the steps of the Face-Intersect algorithm.

2.4.4 Remarks about the Face-intersect algorithm

For a subset-separable NMF $M = AW$ with $M \in \mathbb{R}^{n \times m}$ and inner dimension r, let the number of filled faces in the subset separability assumption be k (i.e., the filled faces are $S_1, \ldots, S_k$). The total running time of Face-intersect algorithm is $O(nmr + nd.OPT + kr^4 + nr^3)$, where OPT is the running time of the convex optimization problem in Algorithm 1 [17]. The authors have reported the convergence of Face-Intersect after roughly k calls to OPT.

Clearly, the subset-separability assumption is milder than separability, upon which many previous works are based. Still we are required to solve many convex programs due to Algorithm 1. Such algorithms are typically hard to scale [19] due to their high complexity.

The Face-Intersect algorithm exploits collinearity properties to identify the data and makes assumptions on 'genericity' [28]. Using the higher dimensional structure (faces) of the simplex is a marked improvement over using just vertices in the case of separability.

Our contribution makes use of assumptions similar to Ge and Zou [17] and we replace one of their key algorithms (Algorithm 1) with a quadratic programming problem. Running the convex program in Algorithm 1 is the most expensive part of the Face-intersect algorithm and thus our contribution is an improvement on solving the NMF robustly.

Chapter 3

QP formulation

Let us state the assumptions in our replacement for Algorithm 1:

1. **Input**: Set of points $\{\mathbf{x}_i\}_{i=1}^n$ in $\mathbb{R}^m$ which are the rows of M. The data points have been translated so that $\mathbf{x}_1 = \mathbf{0}$. This is the point believed to be at the center i.e., $\mathbf{x}_0$ in Algorithm 1.
2. The data points have been scaled in a way to be described later.
3. **Output**: Linear subspace L corresponding to a filled face of the W-simplex. The steps to recover the subspace L are described in Chapter 4.

To obtain L, we consider the following quadratic programming (QP) problem:

$$\begin{aligned}
\text{Primal}: \min \quad & \sum_{i=1}^{n} s_i + c\sum_{i=1}^{n} t_i + c^2 \sum_{i=1}^{n} u_i + \frac{\lambda}{2}\|\mathbf{p}\|^2 \\
\text{subject to} \quad & \mathbf{p}^T\mathbf{x}_i + s_i + t_i + u_i \geq 1 \quad \forall\ i \in [n] \\
& s_i \geq 0 \quad \forall\ i \in [n] \\
& s_i \leq 1 - \delta \quad \forall\ i \in [n] \\
& t_i \geq 0 \quad \forall\ i \in [n] \\
& t_i \leq \delta \quad \forall\ i \in [n] \\
& u_i \geq 0 \quad \forall\ i \in [n].
\end{aligned} \tag{3.1}$$

The values of the parameters $c \gg 1,\ \lambda > 0$ and $\delta \in (0, 1)$ are chosen appropriately later in the thesis. Note that $\mathbf{p} \in \mathbb{R}^m, \{s_i\}_{i=1}^n, \{t_i\}_{i=1}^n$ and $\{u_i\}_{i=1}^n$ are the unknowns in this QP. We

prove that every optimizer to the QP has the same value of the vector $\mathbf{p}$, in the following lemma.

Lemma 14. *Let* $(\mathbf{p}, \mathbf{s}, \mathbf{t}, \mathbf{u})$ *and* $(\mathbf{p}', \mathbf{s}', \mathbf{t}', \mathbf{u}')$ *be two optimizers to the QP (3.1). Then* $\mathbf{p} = \mathbf{p}'$ *holds.*

Proof. Suppose not. Let $\mathbf{p} \neq \mathbf{p}'$. Consider the tuple $(\mathbf{p}^{\text{mid}}, \mathbf{s}^{\text{mid}}, \mathbf{t}^{\text{mid}}, \mathbf{u}^{\text{mid}}) = (\frac{\mathbf{p}+\mathbf{p}'}{2}, \frac{\mathbf{s}+\mathbf{s}'}{2}, \frac{\mathbf{t}+\mathbf{t}'}{2}, \frac{\mathbf{u}+\mathbf{u}'}{2})$. Because the linear constraints of the QP are convex, $(\mathbf{p}^{\text{mid}}, \mathbf{s}^{\text{mid}}, \mathbf{t}^{\text{mid}}, \mathbf{u}^{\text{mid}})$ is clearly a feasible point. Since both $(\mathbf{p}, \mathbf{s}, \mathbf{t}, \mathbf{u})$ and $(\mathbf{p}', \mathbf{s}', \mathbf{t}', \mathbf{u}')$ are optimizers, they have the same objective value.

$$
\begin{aligned}
\text{Optimal value} &= \sum_{i=1}^{n} s_i + c\sum_{i=1}^{n} t_i + c^2 \sum_{i=1}^{n} u_i + \frac{\lambda}{2}\left\|\mathbf{p}\right\|^2 \\
&= \sum_{i=1}^{n} s_i^{'} + c\sum_{i=1}^{n} t_i^{'} + c^2 \sum_{i=1}^{n} u_i^{'} + \frac{\lambda}{2}\left\|\mathbf{p}'\right\|^2 \\
&= \sum_{i=1}^{n} \frac{s_i + s_i^{'}}{2} + c\sum_{i=1}^{n} \frac{t_i + t_i^{'}}{2} + c^2 \sum_{i=1}^{n} \frac{u_i + u_i^{'}}{2} + \frac{\lambda}{2}.\left(\frac{\left\|\mathbf{p}\right\|^2 + \left\|\mathbf{p}'\right\|^2}{2}\right) \\
&= \sum_{i=1}^{n} s_i^{\text{mid}} + c\sum_{i=1}^{n} t_i^{\text{mid}} + c^2 \sum_{i=1}^{n} u_i^{\text{mid}} + \frac{\lambda}{2}.\left(\frac{\left\|\mathbf{p}\right\|^2 + \left\|\mathbf{p}'\right\|^2}{2}\right) \\
&= \sum_{i=1}^{n} s_i^{\text{mid}} + c\sum_{i=1}^{n} t_i^{\text{mid}} + c^2 \sum_{i=1}^{n} u_i^{\text{mid}} + \frac{\lambda}{2}.\left(\left\|\frac{\mathbf{p}+\mathbf{p}'}{2}\right\|^2 + \left\|\frac{\mathbf{p}-\mathbf{p}'}{2}\right\|^2\right) \\
&> \sum_{i=1}^{n} s_i^{\text{mid}} + c\sum_{i=1}^{n} t_i^{\text{mid}} + c^2 \sum_{i=1}^{n} u_i^{\text{mid}} + \frac{\lambda}{2}.\left\|\frac{\mathbf{p}+\mathbf{p}'}{2}\right\|^2 \quad (\because \mathbf{p} \neq \mathbf{p}') \\
&= \sum_{i=1}^{n} s_i^{\text{mid}} + c\sum_{i=1}^{n} t_i^{\text{mid}} + c^2 \sum_{i=1}^{n} u_i^{\text{mid}} + \frac{\lambda}{2}.\left\|\mathbf{p}^{\text{mid}}\right\|^2 \\
&= \text{Objective value at the feasible point } (\mathbf{p}^{\text{mid}}, \mathbf{s}^{\text{mid}}, \mathbf{t}^{\text{mid}}, \mathbf{u}^{\text{mid}}).
\end{aligned}
$$

Thus we have found a feasible point which has lower objective than the optimizer, which is a contradiction. Therefore $\mathbf{p} = \mathbf{p}'$ is true. □

We can get the desired linear subspace from the minimizer $\mathbf{p}^*$ of the above QP. If $\mathbf{p}$ is specified, the problem is separable in the remaining variables. The other unknowns $\{s_i\}_{i=1}^n$, $\{t_i\}_{i=1}^n$ and $\{u_i\}_{i=1}^n$ are separable by i and the optimal choices are as follows:

- If $(\mathbf{p}^*)^T\mathbf{x}_i \geq 1$, then $s_i = t_i = u_i = 0$.
- If $1 - \delta \leq (\mathbf{p}^*)^T\mathbf{x}_i < 1$, then $s_i = 1 - (\mathbf{p}^*)^T\mathbf{x}_i$, $t_i = 0$ and $u_i = 0$.
- If $0 \leq (\mathbf{p}^*)^T\mathbf{x}_i \leq 1 - \delta$, then $s_i = 1 - \delta$, $t_i = \delta - (\mathbf{p}^*)^T\mathbf{x}_i$ and $u_i = 0$.
- If $(\mathbf{p}^*)^T\mathbf{x}_i \leq 0$, then $s_i = 1 - \delta, t_i = \delta$ and $u_i = -(\mathbf{p}^*)^T\mathbf{x}_i$.

Intuitively, $\mathbf{p}$ should make a large positive inner product with each data point $\mathbf{x}_i$. Because of the s_i, t_i and u_i terms in the objective, the ideal case occurs when $\mathbf{p}^T\mathbf{x}_i = 1$ for every i since this forces each s_i, t_i and u_i to zero. It is still good if $\mathbf{p}^T\mathbf{x}_i > 1$, which is the same as the previous case except for a higher objective due to the norm term. Having $\mathbf{p}^T\mathbf{x}_i < 0$ for any i is bad, since this makes u_i positive and this is penalized by a huge weight (c^2).

As a consequence, the optimal $\mathbf{p}$ should almost be orthogonal to all $\mathbf{x}_i$'s in L. Otherwise if $\mathbf{p}$ makes a positive inner product with some $\mathbf{x}_i$, then the well-centering assumption forces $\mathbf{p}$ to make a negative inner product with another $\mathbf{x}_j$. This makes $u_j > 0$ and the objective value is high due to the penalty term on u_j.

Define the optimal objective value given $\mathbf{p}$, as follows:

$$\phi(\mathbf{p}) = \sum_{i=1}^{n} s_i + c\sum_{i=1}^{n} t_i + c^2 \sum_{i=1}^{n} u_i + \frac{\lambda}{2}\|\mathbf{p}\|^2 \tag{3.2}$$

where s_i, t_i and u_i are chosen as per the steps above. We observe without proving, that this is a piecewise quadratic convex function.

3.1 Assumptions on the data points

The noiseless data points are $\{\mathbf{x}_i\}_{i=1}^n \in \mathbb{R}^m$. Suppose we have a subspace $L \subset \mathbb{R}^m$ of dimension k. Say that some of the data points lie exactly on L. Say l of the noiseless data points $\{\mathbf{x}_i : i \in S_L \subset [n]\}$ lie on L. Without loss of generality, let us relabel the data points such that $S_L = [l]$.

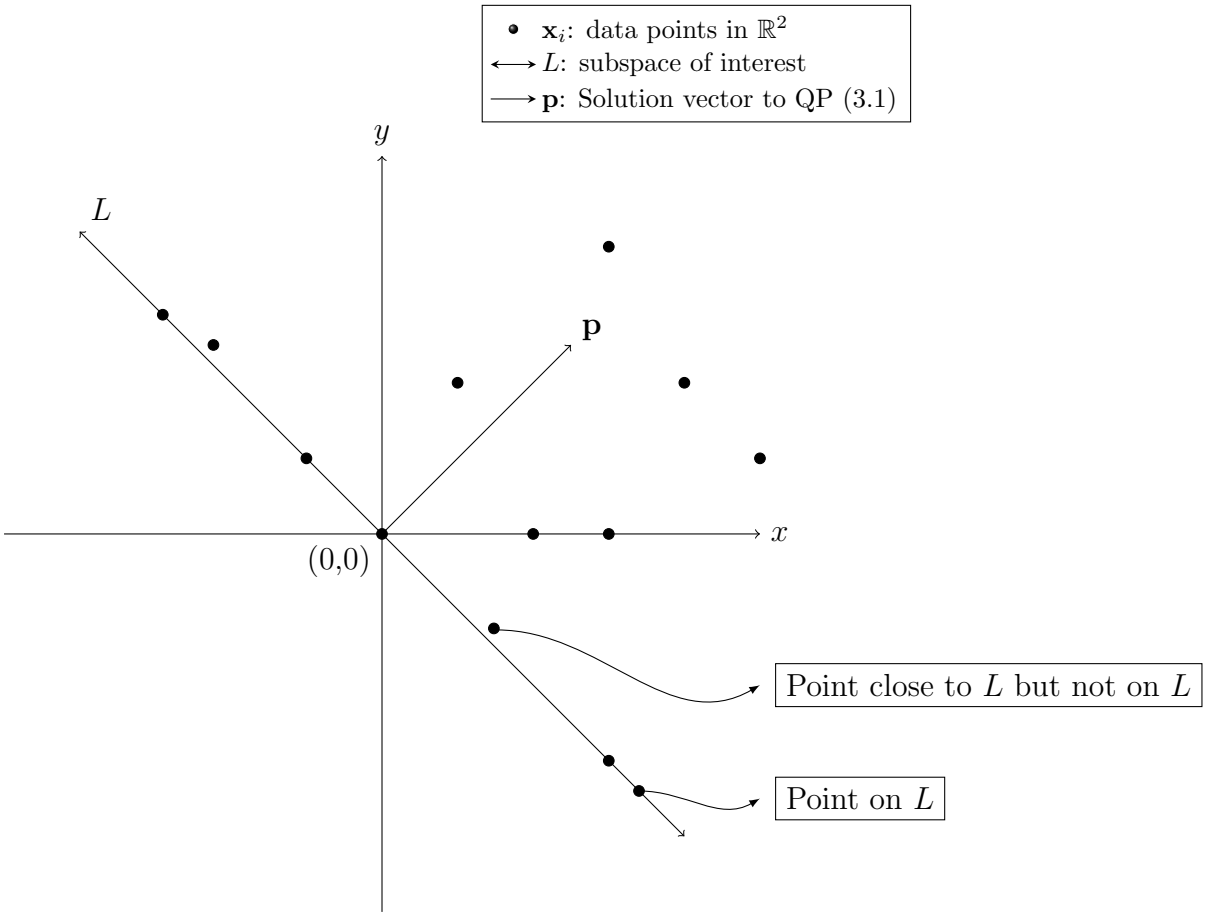

Figure 3.1: Example of subspace L in $\mathbb{R}^2$. Dimension of L here is $k = 1$. There are many points almost collinear to L and the remaining points lie on one side of L. The optimum vector $\mathbf{p} \in \mathbb{R}^2$ to QP (3.1) is shown. Note that $\mathbf{p}$ is almost orthogonal to L and points inward to the half-space containing the data.

Assume the data points are translated such that one of the data points is the origin (let $\mathbf{x}_1 = \mathbf{0}$). We have two assumptions for *well-centering*. The first assumption is that $\mathbf{0}$ lies in the convex hull of $\{\mathbf{x}_2, \ldots, \mathbf{x}_l\}$.

$$\mathbf{0} = \sum_{i=2}^{l} \alpha_i \mathbf{x}_i \tag{3.3}$$

where the weights α_i are nonnegative and sum to 1. The second assumption for well-centering is also a condition on the data. Let $\mathbf{y}_i = \alpha_i \mathbf{x}_i$ for all $i \in [l]$. Suppose $Y =$

$[\mathbf{y}_1, \ldots, \mathbf{y}_l]^T$. We have a singular value condition on the data: the kth singular value of Y (denoted by σ_Y) cannot be too small. This condition is similar to Condition 1 in Definition 13 in Chapter 2. If σ_Y is the kth largest singular value of Y,

$$\sigma_Y = \sigma_k(Y) = \inf_{\substack{\|\mathbf{x}\|=1 \\ \mathbf{x}\in L}} \left\|Y^T\mathbf{x}\right\|. \tag{3.4}$$

Figure 3.1 presents an example of data points in 2-dimensional space which satisfy the above assumptions. The subspace L is of dimension $k = 1$. Note that origin is a data point and the points in L are well-centered.

Recall the notion of W-simplex in Chapter 2. Let the vertices of the W-simplex be denoted by $\mathbf{v}_1, \ldots, \mathbf{v}_{m+1}$. We use the following definition of quality of the shape of the simplex, common in literature: the shape quality (given in (2.9)) is

$$\kappa_{cond} = \ \text{cond}(\bar{V}) = \left\|\bar{V}\right\| \left\|\bar{V}^{-1}\right\| \tag{3.5}$$

where $\bar{V} \in \mathbb{R}^{m\times m}$ is given by $\bar{V} = [\mathbf{v}_2, \ldots, \mathbf{v}_{m+1}] - \mathbf{v}_1\mathbf{e}^T$. Note that the definition (3.5) is translation and rotation invariant, but depends on the choice of $\mathbf{v}_1$ (up to a constant factor). Also assume scaling of the data, such that

$$\left\|\bar{V}\right\| = 1 \tag{3.6}$$

From the scaling of the data $\left\|\bar{V}\right\| = 1$, we know that the data points have an upper bound: For every $j \in [n]$ let $\mathbf{x}_j = \bar{V}\boldsymbol{\theta}_j$ where $\boldsymbol{\theta}_j^T\mathbf{e} = 1$ and $\boldsymbol{\theta}_j \geq \mathbf{0}$. Then $\|\mathbf{x}_j\| \leq \left\|\bar{V}\right\| \|\boldsymbol{\theta}_j\| \leq 1$. Therefore,

$$\|\mathbf{x}_j\| \leq 1 \quad \forall j \in [n]. \tag{3.7}$$

Suppose the data points are noisy, which is a realistic assumption. The noisy data points can be described in terms of the noiseless data points with added noise:

$$\hat{\mathbf{x}}_i = \mathbf{x}_i + \boldsymbol{\varepsilon}_i \ \text{ for } \ i \in [n] \tag{3.8}$$

where $\boldsymbol{\varepsilon}_i$'s are bounded random noise (say $||\boldsymbol{\varepsilon}_i|| < \varepsilon$ for some $\varepsilon > 0$).

3.2 Lemma

We prove a lemma used in Chapter 4.

Lemma 15. *Let the vertices of the W-simplex be $\mathbf{v}_1, \ldots, \mathbf{v}_{m+1}$ and let $L = affinehull\{\mathbf{v}_1, \ldots, \mathbf{v}_{k+1}\}$ be a linear subspace of dimension k (and hence contains the origin). Then,*

1. *There exists $\hat{\mathbf{p}} \in \mathbb{R}^m$ satisfying $\hat{\mathbf{p}}^T\mathbf{x} \geq \kappa$ dist$(\mathbf{x}, L)$ for any $\mathbf{x} \in conv\{\mathbf{v}_1, \ldots, \mathbf{v}_{m+1}\}$ and for a constant $\kappa > 0$ that depends on the data.*
2. *$\hat{\mathbf{p}}^T\mathbf{x}_i \leq 1$ for all $i \in [n]$.*
3. *$\|\hat{\mathbf{p}}\| \leq 1$.*

Proof. Let $V_1 = [\mathbf{v}_1, \ldots, \mathbf{v}_{k+1}], V_2 = [\mathbf{v}_{k+2}, \ldots, \mathbf{v}_{m+1}]$ and $V = [V_1, V_2]$. Let the square matrix $C \in \mathbb{R}^{(m+1)\times(m+1)}$ be defined as follows.

$$C = \begin{bmatrix} V_1 & V_2 \\ \mathbf{e}_{k+1}^T & \mathbf{e}_{m-k}^T \end{bmatrix} \tag{3.9}$$

Recall the affine independence of the vertices of the W-simplex in Section 2.1.3, and it is trivial to see that this is equivalent to stating that C is non-singular.

Say $\hat{\mathbf{p}} = PV_2\mathbf{f}$, where P is the projector onto $L^\perp$ and the vector $\mathbf{f} \in \mathbb{R}^{m-k}$ is yet to be determined. For $\mathbf{x} \in \text{conv}\{\mathbf{v}_1, \ldots, \mathbf{v}_{m+1}\}$, we wish to show for a constant $\kappa > 0$,

$$\hat{\mathbf{p}}^T\mathbf{x} \geq \kappa \text{ dist}(\mathbf{x}, L). \tag{3.10}$$

Let $\mathbf{x} = \mathbf{x}_L + \mathbf{x}'$, where $\mathbf{x}_L \in \text{cone}\{\mathbf{v}_1, \ldots, \mathbf{v}_{k+1}\}$ and $\mathbf{x}' \in \text{cone}\{\mathbf{v}_{k+2}, \ldots, \mathbf{v}_{m+1}\}$. Observe that both $\hat{\mathbf{p}}^T\mathbf{x}'$ and dist$(\mathbf{x}', L)$ are invariant if we add $\mathbf{x}_L$ to $\mathbf{x}'$ since $\mathbf{x}_L \in L \subset \hat{\mathbf{p}}^\perp$. Therefore to show (3.10) is true, it suffices to prove for some constant $\kappa > 0$ and for every $\mathbf{x}' \in \text{cone}\{\mathbf{v}_{k+2}, \ldots, \mathbf{v}_{m+1}\}$,

$$\hat{\mathbf{p}}^T\mathbf{x}' \geq \kappa \text{ dist}(\mathbf{x}', L) \tag{3.11}$$

Since $\mathbf{x}' \in \text{cone}\{\mathbf{v}_{k+2}, \ldots, \mathbf{v}_{m+1}\}$ and $\mathbf{x} \in \text{conv}\{\mathbf{v}_1, \ldots, \mathbf{v}_{m+1}\}$, we can write $\mathbf{x}' = V_2\boldsymbol{\lambda}$ for some nonnegative vector of coefficients $\boldsymbol{\lambda} \in \mathbb{R}^{m-k}$ satisfying $\boldsymbol{\lambda}^T\mathbf{e}_{m-k} \leq 1$. Observe,

$$\begin{aligned} \text{dist}(\mathbf{x}', L) = \|P\mathbf{x}'\| &\leq \|\mathbf{x}'\| \\ &= \|V_2\boldsymbol{\lambda}\| \\ &\leq \|V_2\| \, \|\boldsymbol{\lambda}\| \\ &\leq \left\|\bar{V}\right\| \|\boldsymbol{\lambda}\| \\ \implies \text{dist}(\mathbf{x}', L) &\leq \|\boldsymbol{\lambda}\| \,. \quad (\because (3.6)) \end{aligned} \tag{3.12}$$

Our next step is to look for a lower bound on $\hat{\mathbf{p}}^T\mathbf{x}'$. Observe,

$$\hat{\mathbf{p}}^T\mathbf{x}' = (PV_2\mathbf{f})^T(V_2\boldsymbol{\lambda}) = \mathbf{f}^T V_2^T P V_2 \boldsymbol{\lambda}. \tag{3.13}$$

We claim that $PV_2 \in \mathbb{R}^{m\times(m-k)}$ has full rank. Suppose not, then there exists a nonzero vector $\mathbf{y} \in \mathbb{R}^{m-k}$ such that $PV_2\mathbf{y} = \mathbf{0}$. If $\mathbf{e}_{m-k}^T\mathbf{y} \neq 0$, then we can rescale $\mathbf{y}$ such that $\mathbf{e}_{m-k}^T\mathbf{y} = 1$. Let $\mathbf{z} = V_2\mathbf{y}$. Then $P\mathbf{z} = \mathbf{0}$ implies $\mathbf{z} \in L$, since P is the projector onto $L^\perp$. But L is the affine hull of the columns of V_1. There exists $\mathbf{d} \in \mathbb{R}^{k+1}$ satisfying $V_1\mathbf{d} = \mathbf{z}$ and $\mathbf{e}_{k+1}^T\mathbf{d} = 1$. Using (3.9),

$$\begin{aligned} C\begin{bmatrix}\mathbf{d}\\ \mathbf{y}\end{bmatrix} &= \begin{bmatrix} V_1 & V_2 \\ \mathbf{e}_{k+1}^T & \mathbf{e}_{m-k}^T\end{bmatrix}\begin{bmatrix}\mathbf{d}\\ -\mathbf{y}\end{bmatrix} \\ &= \begin{bmatrix} V_1\mathbf{d} - V_2\mathbf{y} \\ \mathbf{e}_{k+1}^T\mathbf{d} - \mathbf{e}_{m-k}^T\mathbf{y}\end{bmatrix} \\ &= \begin{bmatrix}\mathbf{z}-\mathbf{z}\\ 1-1\end{bmatrix} \\ \implies C\begin{bmatrix}\mathbf{d}\\ \mathbf{y}\end{bmatrix} &= \mathbf{0}, \end{aligned} \tag{3.14}$$

which contradicts the non-singularity of C. We assumed above that $\mathbf{e}_{m-k}^T\mathbf{y} \neq 0$. In the other case, let $\mathbf{e}_{m-k}^T\mathbf{y} = 0$. Construct $\mathbf{z}$ and $\mathbf{d}$ as above. Since 0 is in the affine hull of V_1, let $\mathbf{d}' \in \mathbb{R}^{k+1}$ be a vector such that $V_1\mathbf{d}' = \mathbf{0}$ and $\mathbf{e}_{k+1}^T\mathbf{d}' = 1$. Then $V_1(\mathbf{d}-\mathbf{d}') = \mathbf{z}$ and $\mathbf{e}_{k+1}^T(\mathbf{d}-\mathbf{d}') = 0$. Using the same argument to arrive at (3.14), we can see that

$$C\begin{bmatrix}\mathbf{d}-\mathbf{d}'\\ \mathbf{y}\end{bmatrix} = \mathbf{0}$$

which again contradicts the non-singularity of C. Thus $PV_2 \in \mathbb{R}^{m\times(m-k)}$ has rank $m-k$.

Now,

$$\begin{aligned}
\operatorname{rank}(V_2^T P V_2) &= \operatorname{rank}(V_2^T P^2 V_2) \ \text{(Since } P \text{ is a projection matrix)} \\
&= \operatorname{rank}\big((V_2^T P^T).(P V_2)\big) \\
&= \operatorname{rank}\big((P V_2)^T P V_2\big) \\
&= m-k \ \ (\because \operatorname{rank}(B) = \operatorname{rank}(B^T B), \text{ for any matrix } B) \\
\implies \operatorname{rank}(V_2^T P V_2) &= m-k.
\end{aligned}$$

Thus the matrix $V_2^T P V_2 \in \mathbb{R}^{(m-k)\times(m-k)}$ is of full rank and hence is invertible. If we pick $\mathbf{f} = \alpha(V_2^T P V_2)^{-1}\mathbf{e}$, where $\mathbf{e} \in \mathbb{R}^m$ is the vector of all ones and $\alpha \in \mathbb{R}$ is yet to be determined, equation (3.13) becomes,

$$\begin{aligned}
\hat{\mathbf{p}}^T \mathbf{x}' &= \alpha((V_2^T P V_2)^{-1}\mathbf{e})^T V_2^T P V_2 \boldsymbol{\lambda} \\
&= \alpha \mathbf{e}^T \boldsymbol{\lambda} \\
&= \alpha \left\|\boldsymbol{\lambda}\right\|_1 \quad (\because \boldsymbol{\lambda} \geq \mathbf{0}) \\
&\geq \alpha \left\|\boldsymbol{\lambda}\right\|_2 \\
\implies \hat{\mathbf{p}}^T \mathbf{x}' &\geq \alpha \left\|\boldsymbol{\lambda}\right\|_2 . \qquad (3.15)
\end{aligned}$$

From equations (3.12) and (3.15) we have the following conclusion,

$$\hat{\mathbf{p}}^T \mathbf{x}' \geq \alpha \ \operatorname{dist}(\mathbf{x}', L) \qquad (3.16)$$

and hence equation (3.11) is true for $\kappa = \alpha$.

To show $\hat{\mathbf{p}}^T \mathbf{x}_i \leq 1$ for all i, we first show an upper bound on $\hat{\mathbf{p}}$. Let us look for a bound on $(V_2^T P V_2)^{-1}$ in terms of the data. Let Z be an orthonormal basis of $L = \operatorname{Range}(V_1)$. Then we can write $V_1 = Z\Lambda$ for some $\Lambda \in \mathbb{R}^{k\times(k+1)}$. Extend Z to an orthogonal matrix $Q = [Z, Y]$, where Y is an orthonormal basis of $L^\perp$. Then $P = YY^T$ and hence $V_2^T P V_2 = V_2^T Y Y^T V_2 = (Y^T V_2)^T Y^T V_2$. Therefore, the task simplifies to finding a lower bound on the smallest singular value of $Y^T V_2 \in \mathbb{R}^{(m-k)\times(m-k)}$.

Because of the scaling of the data (3.6), the shape quality can be related to $\left\|\bar{V}^{-1}\right\|$,

i.e., $\left\|\bar{V}^{-1}\right\| = \kappa_{\text{cond}}$. If we denote $V_1' = V_1(2:k+1)$, observe

$$\begin{aligned} Q^T\bar{V} &= [Z,Y]^T\bar{V} \\ &= [Z,Y]^T[\underbrace{V_1' - \mathbf{v_1}\mathbf{e}^T}_{\in\mathbb{R}^{m\times k}}, \underbrace{V_2 - \mathbf{v_1}\mathbf{e}^T}_{\in\mathbb{R}^{m\times(m-k)}}] \\ &= \begin{bmatrix} Z^T(V_1' - \mathbf{v_1}\mathbf{e}^T) & Z^T(V_2 - \mathbf{v_1}\mathbf{e}^T) \\ Y^T(V_1' - \mathbf{v_1}\mathbf{e}^T) & Y^T(V_2 - \mathbf{v_1}\mathbf{e}^T) \end{bmatrix}. \end{aligned}$$

Since Y is a basis of $L^\perp$ and both $\mathbf{v}_1$ and the columns of V_1' are in L, it follows that $Y^T(V_1' - \mathbf{v_1}\mathbf{e}^T) = Y^T\mathbf{v_1}\mathbf{e}^T = 0$. Therefore,

$$Q^T\bar{V} = \begin{bmatrix} Z^T(V_1' - \mathbf{v_1}\mathbf{e}^T) & Z^T(V_2 - \mathbf{v_1}\mathbf{e}^T) \\ 0 & Y^TV_2 \end{bmatrix}$$

$$\implies \left\|\bar{V}^{-1}\right\| = \left\|(Q^T\bar{V})^{-1}\right\| = \left\| \begin{bmatrix} [Z^T(V_1' - \mathbf{v_1}\mathbf{e}^T)]^{-1} & U \\ 0 & (Y^TV_2)^{-1} \end{bmatrix} \right\| \quad (U \text{ is some matrix})$$

$$\implies \left\|\bar{V}^{-1}\right\| \geq \left\|(Y^TV_2)^{-1}\right\| = \frac{1}{\sigma_{m-k}(Y^TV_2)} \tag{3.17}$$

where $\sigma_{m-k}(Y^TV_2)$ is the smallest singular value of Y^TV_2. From the shape quality (3.5), we can see that $\sigma_{m-k}(Y^TV_2) \geq \frac{1}{\kappa_{\text{cond}}}$. We find an upper bound on $\hat{\mathbf{p}}$, as follows.

$$\begin{aligned} \|\hat{\mathbf{p}}\| &= \left\|\alpha PV_2(V_2^TPV_2)^{-1}\mathbf{e}\right\| \\ &\leq \alpha \|V_2\| \left\|(V_2^TPV_2)^{-1}\right\| \sqrt{n} \\ &= \alpha\sqrt{n} \|V_2\| \lambda_{\max}\{(V_2^TPV_2)^{-1}\} \\ &\leq \alpha\sqrt{n} \|V_2\| \kappa_{\text{cond}}^2 \quad (\text{From (3.17)}) \\ &\leq \alpha\kappa_{\text{cond}}^2\sqrt{n} \quad (\text{From (3.6)}) \\ \implies \|\hat{\mathbf{p}}\| &\leq \hat{\kappa} = \alpha\kappa_{\text{cond}}^2\sqrt{n}. \end{aligned}$$

If we choose $\alpha = \frac{1}{\kappa_{\text{cond}}^2\sqrt{n}}$, then we have,

$$\|\hat{\mathbf{p}}\| \leq 1. \tag{3.18}$$

By the upper bound on the datapoints (3.7),

$$\hat{\mathbf{p}}^T \mathbf{x}_j \leq \|\hat{\mathbf{p}}\| \leq 1. \tag{3.19}$$

Therefore, (3.16), (3.18 and (3.19) show that:

1. There exists $\hat{\mathbf{p}} \in \mathbb{R}^m$ satisfying $\hat{\mathbf{p}}^T \mathbf{x} \geq \kappa \operatorname{dist}(\mathbf{x}, L)$ for any $\mathbf{x} \in \operatorname{conv}\{\mathbf{v}_1, \ldots, \mathbf{v}_{m+1}\}$, where $\kappa = \frac{1}{\kappa_{\text{cond}}^2 \sqrt{n}}$ is a constant that depends on the data.
2. $\hat{\mathbf{p}}^T \mathbf{x}_i \leq 1$ for all $i \in [n]$.
3. $\|\hat{\mathbf{p}}\| \leq 1$.

This completes the proof. □

Chapter 4

Subspace recovery

We highlight the choice of constants used in this chapter. The constants that depend entirely on the data are n (number of data points), σ_Y (well centering condition (3.4)), and κ (constant in Lemma 15). For λ yet to be chosen, let us pick the following constants:

Constant	c	δ	η	η'
Value	$\frac{8n^{3/2}}{\sigma_Y}$	$\frac{\sigma_Y}{8n^{3/2}}$	$\frac{\sigma_Y}{2n\kappa}\sqrt{\lambda}$	$\frac{\sigma_Y}{64n^2}\sqrt{\lambda}$

(4.1)

4.1 Problem statement

Recall the quadratic program (3.1):

$$\begin{aligned} \text{Primal}: \min \quad & \sum_{i=1}^{n} s_i + c\sum_{i=1}^{n} t_i + c^2 \sum_{i=1}^{n} u_i + \frac{\lambda}{2}\|\mathbf{p}\|^2 \\ \text{subject to} \quad & \mathbf{p}^T\mathbf{x}_i + s_i + t_i + u_i \geq 1 \ \ \forall\ i \in [n] \\ & s_i \geq 0 \ \ \forall\ i \in [n] \\ & s_i \leq 1-\delta \ \ \forall\ i \in [n] \\ & t_i \geq 0 \ \ \forall\ i \in [n] \\ & t_i \leq \delta \ \ \forall\ i \in [n] \\ & u_i \geq 0 \ \ \forall\ i \in [n]. \end{aligned} \tag{4.2}$$

Our goal in formulating the above QP (4.2) is to recover the subspace L. We simplify the problem (4.2) by defining the following quantities:

$$\mathbf{y} = \begin{bmatrix} \mathbf{p} & \mathbf{s} & \mathbf{t} & \mathbf{u} \end{bmatrix}^T \in \mathbb{R}^{m+3n}$$

$$\mathbf{c}_0 = \begin{bmatrix} \mathbf{0}_m & \mathbf{e}_n & c\mathbf{e}_n & c^2\mathbf{e}_n \end{bmatrix}^T \in \mathbb{R}^{m+3n}$$

$$B = \begin{bmatrix} \lambda I_m & 0_{m\times n} & 0_{m\times n} & 0_{m\times n} \\ 0_{n\times m} & 0_{n\times n} & 0_{n\times n} & 0_{n\times n} \\ 0_{n\times m} & 0_{n\times n} & 0_{n\times n} & 0_{n\times n} \\ 0_{n\times m} & 0_{n\times n} & 0_{n\times n} & 0_{n\times n} \end{bmatrix} \in \mathbb{R}^{(m+3n)\times(m+3n)}$$

$$Q = \left[\begin{array}{c|c|c|c} \begin{matrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_n^T \end{matrix} & I_n & I_n & I_n \\ \hline 0_{n\times m} & I_n & 0_{n\times n} & 0_{n\times n} \\ \hline 0_{n\times m} & -I_n & 0_{n\times n} & 0_{n\times n} \\ \hline 0_{n\times m} & 0_{n\times n} & I_n & 0_{n\times n} \\ \hline 0_{n\times m} & 0_{n\times n} & -I_n & 0_{n\times n} \\ \hline 0_{n\times m} & 0_{n\times n} & 0_{n\times n} & I_n \end{array}\right] \in \mathbb{R}^{6n\times(m+3n)}$$

$$\mathbf{b}_0 = \begin{bmatrix} \mathbf{e}_n & \mathbf{0}_n & -(1-\delta)\mathbf{e}_n & \mathbf{0}_n & -\delta\mathbf{e}_n & \mathbf{0}_n \end{bmatrix}^T \in \mathbb{R}^{6n}$$

then we can write the problem (4.2) as a quadratic programming problem:

$$\begin{aligned} \text{Primal}: \min \quad & \mathbf{c}_0^T \mathbf{y} + \frac{1}{2} \mathbf{y}^T B \mathbf{y} \\ \text{subject to} \quad & Q\mathbf{y} \geq \mathbf{b}_0. \end{aligned} \tag{4.3}$$

Using the Lagrangian dual formulation in Chapter 2 Subsection 2.1.1, the dual of (4.3) for the dual variables $\mathbf{d} \in \mathbb{R}^{6n}$ and $\mathbf{z} \in \mathbb{R}^{m+3n}$ is

$$\begin{aligned} \text{Dual}: \max \quad & -\frac{1}{2} \mathbf{z}^T B \mathbf{z} + \mathbf{b}_0^T \mathbf{d} \\ \text{subject to} \quad & B\mathbf{z} + Q^T \mathbf{d} = \mathbf{c}_0 \\ & \mathbf{d} \geq \mathbf{0}. \end{aligned} \tag{4.4}$$

Let the dual variables corresponding to each of the six sets of constraints be $\{\mathbf{q}, \mathbf{d}^4, \mathbf{r}, \mathbf{d}^4, \mathbf{w}, \mathbf{d}^6\} \subset \mathbb{R}^n$ respectively. Then

$$\mathbf{d} = \begin{bmatrix} \mathbf{q} & \mathbf{d}^2 & \mathbf{r} & \mathbf{d}^4 & \mathbf{w} & \mathbf{d}^6 \end{bmatrix}^T.$$

Primal constraint	Corresponding dual variable
$\mathbf{p}^T \mathbf{x}_i + s_i + t_i + u_i \geq 1$	$\mathbf{q}$
$s_i \geq 0$	$\mathbf{d}^2$
$s_i \leq 1 - \delta$	$\mathbf{r}$
$t_i \geq 0$	$\mathbf{d}^4$
$t_i \leq \delta$	$\mathbf{w}$
$u_i \geq 0$	$\mathbf{d}^6$

From the above table, we get the complementary slackness conditions. For all $1 \leq i \leq n$,

$$(\mathbf{p}^T\mathbf{x}_i + s_i + t_i + u_i - 1) \times q_i = 0 \tag{4.5}$$
$$s_i \times d^2(i) = 0 \tag{4.6}$$
$$(s_i - 1 + \delta) \times r_i = 0 \tag{4.7}$$
$$t_i \times d^4(i) = 0 \tag{4.8}$$
$$(t_i - \delta) \times w_i = 0 \tag{4.9}$$
$$u_i \times d^6(i) = 0. \tag{4.10}$$

Let $\mathbf{z} = \begin{bmatrix} \mathbf{z}_p & \mathbf{z}_s & \mathbf{z}_t & \mathbf{z}_u \end{bmatrix}^T \in \mathbb{R}^{m+3n}$. Then we have,

$$B\mathbf{z} = \begin{bmatrix} \lambda\mathbf{z}_p \\ \mathbf{0}_n \\ \mathbf{0}_n \\ \mathbf{0}_n \end{bmatrix} \quad \text{and,} \quad Q^T\mathbf{d} = \left[\begin{array}{ccc|c|c|c|c|c} \mathbf{x}_1 & \dots & \mathbf{x}_n & 0_{m\times n} & 0_{m\times n} & 0_{m\times n} & 0_{m\times n} & 0_{m\times n} \\ \hline & I_n & & I_n & -I_n & 0_{n\times n} & 0_{n\times n} & 0_{n\times n} \\ \hline & I_n & & 0_{n\times n} & 0_{n\times n} & I_n & -I_n & 0_{n\times n} \\ \hline & I_n & & 0_{n\times n} & 0_{n\times n} & 0_{n\times n} & 0_{n\times n} & I_n \end{array}\right] \begin{bmatrix} \mathbf{q} \\ \mathbf{d}^2 \\ \mathbf{r} \\ \mathbf{d}^4 \\ \mathbf{w} \\ \mathbf{d}^6 \end{bmatrix}.$$

The first constraint of (4.4) becomes,

$$\begin{bmatrix} \lambda\mathbf{z}_p \\ \mathbf{0}_n \\ \mathbf{0}_n \\ \mathbf{0}_n \end{bmatrix} + \begin{bmatrix} \sum_{i=1}^n q(i)\mathbf{x}_i \\ \mathbf{q} + \mathbf{d}^2 - \mathbf{r} \\ \mathbf{q} + \mathbf{d}^4 - \mathbf{w} \\ \mathbf{q} + \mathbf{d}^6 \end{bmatrix} = \begin{bmatrix} \mathbf{0}_n \\ \mathbf{e}_n \\ c\mathbf{e}_n \\ c^2\mathbf{e}_n \end{bmatrix}. \tag{4.11}$$

The following constraints arise from (4.11):

$$\sum_{i=1}^{n} q(i)\mathbf{x}_i = -\lambda \mathbf{z}_p \tag{4.12}$$

$$\mathbf{q} + \mathbf{d}^2 - \mathbf{r} = \mathbf{e}_n \tag{4.13}$$

$$\mathbf{q} + \mathbf{d}^4 - \mathbf{w} = c\mathbf{e}_n \tag{4.14}$$

$$\mathbf{q} + \mathbf{d}^6 = c^2\mathbf{e}_n \tag{4.15}$$

$$\mathbf{d} \geq \mathbf{0}. \tag{4.16}$$

The objective function of (4.4) is now

$$\begin{aligned}
-\frac{1}{2}\mathbf{z}^T B\mathbf{z} + \mathbf{b}_0^T\mathbf{d} &= -\frac{\lambda}{2}\left\|\mathbf{z}_p\right\|^2 + \mathbf{q}^T\mathbf{e}_n - (1-\delta)\mathbf{r}^T\mathbf{e}_n - \delta\mathbf{w}^T\mathbf{e}_n \\
&= \frac{-1}{2\lambda}\left\|\sum_{i=1}^{n} q_i\mathbf{x}_i\right\|^2 + (1-\delta)(\mathbf{q}-\mathbf{r})^T\mathbf{e}_n + \delta(\mathbf{q}-\mathbf{w})^T\mathbf{e}_n \quad \text{(Using (4.12))} \\
&= \frac{-1}{2\lambda}\left\|\sum_{i=1}^{n} q_i\mathbf{x}_i\right\|^2 + \sum_{i=1}^{n}\big[(1-\delta)(q_i - r_i) + \delta(q_i - w_i)\big].
\end{aligned}$$

Dropping the dual variables $\{\mathbf{d}^2, \mathbf{d}^2, \mathbf{d}^2\}$ from the dual constraints (4.13)-(4.16) and including the nonnegativity constraint $\mathbf{d} \geq \mathbf{0}$ for the remaining variables $\{\mathbf{q}, \mathbf{r}, \mathbf{w}\}$, the dual problem is of the following form.

$$\begin{aligned}
\text{Dual}: \max \quad & \frac{-1}{2\lambda}\left\|\sum_{i=1}^{n} q_i\mathbf{x}_i\right\|^2 + \sum_{i=1}^{n}\big[(1-\delta)(q_i - r_i) + \delta(q_i - w_i)\big] \\
\text{subject to} \quad & q_i - r_i \leq 1 \quad \forall\ i \in [n] \\
& q_i - w_i \leq c \quad \forall\ i \in [n] \\
& q_i \leq c^2 \quad \forall\ i \in [n] \\
& \mathbf{q} \geq \mathbf{0}, \mathbf{r} \geq \mathbf{0}, \mathbf{w} \geq \mathbf{0}.
\end{aligned} \tag{4.17}$$

For the dual feasible point $\mathbf{q} = \mathbf{r} = \mathbf{w} = \mathbf{0}$ we can see that the dual objective has value 0. Therefore for any dual optimizer $(\mathbf{q}^*, \mathbf{r}^*, \mathbf{w}^*)$, the objective value is at least 0.

$$
\begin{aligned}
0 &\leq \frac{-1}{2\lambda} \left\| \sum_{i=1}^{n} q_i^* \mathbf{x}_i \right\|^2 + \sum_{i=1}^{n} \left[(1-\delta)(q_i^* - r_i^*) + \delta(q_i^* - w_i^*) \right] \\
\implies \frac{1}{2\lambda} \left\| \sum_{i=1}^{n} q_i^* \mathbf{x}_i \right\|^2 &\leq \sum_{i=1}^{n} \left[(1-\delta)(q_i^* - r_i^*) + \delta(q_i^* - w_i^*) \right] \\
\implies \frac{1}{2\lambda} \left\| \sum_{i=1}^{n} q_i^* \mathbf{x}_i \right\|^2 &\leq \sum_{i=1}^{n} \left[(1-\delta) + \delta c \right] \quad (\because (4.17)) \\
\implies \left\| \sum_{i=1}^{n} q_i^* \mathbf{x}_i \right\| &\leq \sqrt{2\lambda}\sqrt{n + n(1-\delta)} \quad (\because \delta c = 1 \text{ by } (4.1)) \\
\left\| \sum_{i=1}^{n} q_i^* \mathbf{x}_i \right\| &\leq 2\sqrt{\lambda n}. \qquad (4.18)
\end{aligned}
$$

4.2 Noiseless Data

We prove two theorems which help us detect L using the above formulation. For η and δ in (4.1),

Theorem 16. *If* $dist(\mathbf{x}_{\hat{i}}, L) \geq \eta$ *for some* $\hat{i} \in [n]$*, then* $\mathbf{p}^T \mathbf{x}_{\hat{i}} \geq \delta$ *for the optimal* $\mathbf{p}$ *to the QP (4.2).*

Proof. Suppose not. Let $\mathbf{p}^T \mathbf{x}_{\hat{i}} < \delta$. Then from the primal constraints, $s_{\hat{i}} = 1 - \delta$ and $t_{\hat{i}} > 0$. Since we are looking at the optimizer, the complementary slackness conditions (4.7) and (4.8) combined with the dual constraints (4.17) yields $q_{\hat{i}}^* - r_{\hat{i}}^* = 1$ and $q_{\hat{i}}^* - w_{\hat{i}}^* = c$. If $X = [\mathbf{x}_1 \ldots, \mathbf{x}_n] \in \mathbb{R}^{m \times n}$ then (4.18) is:

$$\|X\mathbf{q}^*\| \leq 2\sqrt{\lambda n}. \qquad (4.19)$$

Consider the $\hat{\mathbf{p}}$ from Lemma 15, then $\|\hat{\mathbf{p}}\| \leq 1$ and,

$$
\begin{aligned}
\hat{\mathbf{p}}^T X\mathbf{q}^* &\leq \|\hat{\mathbf{p}}\| \, \|X\mathbf{q}^*\| \\
\implies \sum_{i=1}^{n} (\hat{\mathbf{p}}^T \mathbf{x}_i q_i^*) &\leq 2\sqrt{\lambda n}. \quad (\because (4.19)) \qquad (4.20)
\end{aligned}
$$

From Lemma 15, we can say that $\hat{\mathbf{p}}^T\mathbf{x}_i \geq 0$ for all i. Using Lemma 15 and the fact that $q_{\hat{i}} = w_{\hat{i}} + c \geq c \geq 0$ ($\because w_{\hat{i}} \geq 0$),

$$\sum_{i=1}^{n}(\hat{\mathbf{p}}^T\mathbf{x}_i)q_i \geq \hat{\mathbf{p}}^T\mathbf{x}_{\hat{i}}q_{\hat{i}} \geq c\kappa \text{ dist}(\mathbf{x}_{\hat{i}}, L) \geq c\kappa\eta = 4\sqrt{\lambda n}. \ \ (\because (4.1)) \tag{4.21}$$

Using this in equation (4.20), we get

$$4\sqrt{\lambda n} \leq 2\sqrt{\lambda n}. \tag{4.22}$$

which is a contradiction and hence the assumption that $\mathbf{p}^T\mathbf{x}_{\hat{i}} < \delta$ is false. This completes the proof. □

In the next theorem, we first show that the component along L of the optimal $\mathbf{p}$ to (4.2) is small.

Theorem 17. *The magnitude of the optimal* $\mathbf{p}$ *to the QP (4.2) has an upper bound of* $\sqrt{\frac{4n}{\lambda}}$. *Furthermore, the component of* $\mathbf{p}$ *along* L *is of magnitude at most* $\frac{\sigma_Y}{32n^{3/2}}$.

Proof. For the optimal $\mathbf{p}$ to the QP (4.2), the objective value is at least $\frac{\lambda}{2}\|\mathbf{p}\|^2$ (since other terms are nonnegative). At the feasible point where $\mathbf{p} = \mathbf{0}$, we have $s_i = 1 - \delta,\ t_i = \delta$ and $u_i = 0$ for all $i \in [n]$. The objective value at this point is not lower than the objective at the optimizer. Therefore,

$$\begin{aligned} \frac{\lambda}{2}\|\mathbf{p}\|^2 &\leq n(1-\delta) + nc\delta \\ &\leq n + nc\delta \ \ (\because n\delta > 0) \\ &= 2n \ \ (\because \delta c = 1 \text{ from } (4.1)) \\ \implies \|\mathbf{p}\| &\leq \sqrt{\frac{4n}{\lambda}}. \end{aligned} \tag{4.23}$$

Let the optimal $\mathbf{p}$ be written as $\mathbf{p} = \underbrace{\mathbf{p}_L}_{\in L} + \underbrace{\mathbf{p}_{L^\perp}}_{\in L^\perp}$. Then for all $i \in [l]$ we have $\mathbf{x}_i \in L$ and hence $\mathbf{p}_{L^\perp}^T\mathbf{x}_i = 0$. Therefore,

$$\mathbf{p}^T\mathbf{x}_i = \mathbf{p}_L^T\mathbf{x}_i \ \ \forall\ i \in [l]. \tag{4.24}$$

We use the well centering condition in L (3.3). Since $\mathbf{0}$ is in the convex hull of $\mathbf{x}_2, \ldots, \mathbf{x}_l$, there exists nonnegative coefficients $\{\alpha_i\}_{i=2}^l$ such that $\sum_{i=2}^l \alpha_i = 1$ and $\sum_{i=2}^l \alpha_i \mathbf{x}_i = \mathbf{0}$.

$$\begin{aligned}
&\sum_{i=2}^{l} \alpha_i \mathbf{x}_i = \mathbf{0} \\
\implies &\mathbf{p}^T \sum_{i=2}^{l} \alpha_i \mathbf{x}_i = 0 \\
\implies &\sum_{i=2}^{l} \alpha_i \mathbf{p}^T \mathbf{x}_i = 0 \\
\implies &\sum_{i=2}^{l} \alpha_i \mathbf{p}_L^T \mathbf{x}_i = 0 \quad (\because (4.24)) \\
&\text{Let } \mathbf{y}_i = \alpha_i \mathbf{x}_i \quad \forall\, i \in \{2, \ldots, l\} \\
\implies &\sum_{i=2}^{l} \mathbf{p}_L^T \mathbf{y}_i = 0.
\end{aligned}$$

Let

$$\mu \triangleq \min_{i \in \{2,\ldots,l\}} \mathbf{p}_L^T \mathbf{x}_i = \mathbf{p}_L^T \mathbf{x}_j \quad \text{for some } j \in \{2, \ldots, l\}. \tag{4.25}$$

Clearly, $\mu \leq 0$. Then for all $g \in \{2, \ldots, l\}$,

$$\begin{aligned}
&\mathbf{p}_L^T \mathbf{y}_g = -\sum_{\substack{i=2 \\ i \neq g}}^{l} \mathbf{p}_L^T \mathbf{y}_i \\
&\alpha_g \mu \leq \mathbf{p}_L^T \mathbf{y}_g \leq -\Big(\sum_{\substack{i=2 \\ i \neq g}}^{l} \alpha_i \Big) \mu \leq -\mu \quad (\because (4.25)) \\
\implies &\mathbf{p}_L^T \mathbf{y}_g \in [\alpha_g \mu, -\mu] \\
\implies &|\mathbf{p}_L^T \mathbf{y}_g| \leq -\mu = |\mu|.
\end{aligned} \tag{4.26}$$

Let $Y = [\mathbf{y}_1, \ldots, \mathbf{y}_l]^T$. From (3.4), we have this singular value condition on the data,

$$\begin{aligned}
\sigma_Y &= \inf_{\substack{\|\mathbf{x}\|=1 \\ \mathbf{x}\in L}} \left\|Y^T\mathbf{x}\right\| \\
&\leq \left\|Y^T \frac{\mathbf{p}_L}{\|\mathbf{p}_L\|}\right\| \quad (\because \mathbf{p}_L \in L) \\
&= \sqrt{\sum_{i=1}^{l} \left(\mathbf{y}_i^T \frac{\mathbf{p}_L}{\|\mathbf{p}_L\|}\right)^2} \\
&\leq \sqrt{\sum_{i=1}^{l} \frac{\mu^2}{\|\mathbf{p}_L\|^2}} \quad (\because (4.26)) \\
&= \frac{|\mu|\sqrt{l}}{\|\mathbf{p}_L\|} \\
&\leq \frac{|\mu|\sqrt{n}}{\|\mathbf{p}_L\|} \\
\implies \|\mathbf{p}_L\| &\leq \frac{|\mu|\sqrt{n}}{\sigma_Y}. \qquad (4.27)
\end{aligned}$$

In order to get a good upper bound on $\|\mathbf{p}_L\|$, we need an upper bound on $|\mu|$. We know that for some j, $\mu = \mathbf{p}_L^T\mathbf{x}_j \leq 0$. From our discussions before (3.2), $s_j = 1-\delta, t_j = \delta, u_j = |\mu|$. Thus, the objective function value of the optimizer is at least $c^2|\mu|$. This should be lower than the objective value at any another feasible point. Consider the case where $\mathbf{p} = \mathbf{0}$ in (3.2). The objective value here is exactly $n(1-\delta) + nc\delta = n(2-\delta)$ ($\because$ (4.1)). We have,

$$\begin{aligned}
c^2|\mu| &\leq n(2-\delta) \\
\implies \frac{64n^3}{\sigma_Y^2}|\mu| &\leq 2n \quad (\because (4.1)) \\
\implies |\mu| &\leq \frac{\sigma_Y^2}{32n^2}. \qquad (4.28)
\end{aligned}$$

Using (4.28) in (4.27),

$$\|\mathbf{p}_L\| \leq \frac{\sigma_Y}{32n^{3/2}}. \qquad (4.29)$$

□

This leads to a converse to Theorem 17 except for a different value of η. For η' given in (4.1),

Theorem 18. *If $\mathbf{p}^T\mathbf{x}_{\hat{i}} \geq \delta$ for the optimal $\mathbf{p}$ to the QP (4.2) and for some $\hat{i} \in [n]$, then dist$(\mathbf{x}_{\hat{i}}, L) \geq \eta'$.*

Proof. Pick the optimal $\mathbf{p}$ to the QP (4.2). Suppose $\mathbf{p}^T\mathbf{x}_{\hat{i}} \geq \delta$. For any $1 \leq \hat{i} \leq n$,

$$\begin{aligned}\frac{\sigma_Y}{8n^{3/2}} = \delta \leq \mathbf{p}^T\mathbf{x}_{\hat{i}} &= \mathbf{p}_L^T\mathbf{x}_{\hat{i}} + \mathbf{p}_{L^\perp}^T\mathbf{x}_{\hat{i}} \\ &\leq \|\mathbf{p}_L\| \,.\, \|\mathbf{x}_{\hat{i}}\| + \mathbf{p}_{L^\perp}^T\mathbf{x}_{\hat{i}} \\ &\leq \frac{\sigma_Y}{32n^{3/2}} + \mathbf{p}_{L^\perp}^T\mathbf{x}_{\hat{i}}. \quad (\because (3.7), \text{Theorem 17})\end{aligned}$$

Suppose we split $\mathbf{x}_{\hat{i}} = \mathbf{x}_{\hat{i},L} + \mathbf{x}_{\hat{i},L^\perp}$ where $\mathbf{x}_{\hat{i},L} \in L$ and $\mathbf{x}_{\hat{i},L^\perp} \in L^\perp$. Then $\mathbf{p}_{L^\perp} \perp \mathbf{x}_{\hat{i},L}$. Therefore, $\mathbf{p}_{L^\perp}^T\mathbf{x}_{\hat{i}} = \mathbf{p}_{L^\perp}^T\mathbf{x}_{\hat{i},L^\perp}$ and hence,

$$\begin{aligned}\frac{\sigma_Y}{8n^{3/2}} - \frac{\sigma_Y}{32n^{3/2}} \leq \mathbf{p}_{L^\perp}^T\mathbf{x}_{\hat{i},L^\perp} &\leq \|\mathbf{p}_{L^\perp}\| \,.\, \left\|\mathbf{x}_{\hat{i},L^\perp}\right\| \leq \sqrt{\frac{4n}{\lambda}} \,.\, \left\|\mathbf{x}_{\hat{i},L^\perp}\right\| \quad (\because \text{Theorem 17}) \\ \implies \frac{\frac{3\sigma_Y}{32n^{3/2}}}{\sqrt{\frac{4n}{\lambda}}} &\leq \left\|\mathbf{x}_{\hat{i},L^\perp}\right\| = \text{ dist}(\mathbf{x}_{\hat{i}}, L) \\ \implies \eta' = \frac{\sigma_Y}{64n^2}\sqrt{\lambda} \leq \frac{3\sigma_Y\sqrt{\lambda}}{64n^2} &\leq \text{ dist}(\mathbf{x}_{\hat{i}}, L).\end{aligned}$$

□

4.3 Noisy case

Let us now consider the data points with noise. For each $i \in [n]$, let the noisy data points be $\hat{\mathbf{x}}_i = \mathbf{x}_i + \boldsymbol{\varepsilon}_i$, where the noise component is bounded, $\|\boldsymbol{\varepsilon}_i\| < \varepsilon$ for some $\varepsilon > 0$. Then the new primal and dual problems are written for the noisy data points:

$$\begin{aligned}\text{Primal}: \min \quad & \sum_{i=1}^n s_i + c\sum_{i=1}^n t_i + c^2\sum_{i=1}^n u_i + \frac{\lambda}{2}\|\mathbf{p}\|^2 \\ \text{subject to} \quad & \mathbf{p}^T\hat{\mathbf{x}}_i + s_i + t_i + u_i \geq 1 \ \ \forall\ i \in [n] \\ & s_i \geq 0 \ \ \forall\ i \in [n] \\ & s_i \leq 1 - \delta \ \ \forall\ i \in [n] \\ & t_i \geq 0 \ \ \forall\ i \in [n] \\ & t_i \leq \delta \ \ \forall\ i \in [n] \\ & u_i \geq 0 \ \ \forall\ i \in [n].\end{aligned} \tag{4.30}$$

$$
\begin{aligned}
\text{Dual}: \max \quad & \frac{-1}{2\lambda}\left\|\sum_{i=1}^{n} q_i \hat{\mathbf{x}}_i\right\|^2 + \sum_{i=1}^{n}\left[(1-\delta)(q_i - r_i) + \delta(q_i - w_i)\right] \\
\text{subject to} \quad & q_i - r_i \leq 1 \quad \forall\ i \in [n] \\
& q_i - w_i \leq c \quad \forall\ i \in [n] \\
& q_i \leq c^2 \quad \forall\ i \in [n] \\
& \mathbf{q} \geq \mathbf{0}, \mathbf{r} \geq \mathbf{0}, \mathbf{w} \geq \mathbf{0}.
\end{aligned} \tag{4.31}
$$

We prove similar results to Theorem 16 and Theorem 18. For η and δ in (4.1),

Theorem 19. *If* $dist(\hat{\mathbf{x}}_{\hat{i}}, L) \geq \eta$ *for some* $\hat{i} \in [n]$ *and the noise upper bound satisfies* $\varepsilon < \frac{\sqrt{\lambda}}{\left(\frac{4n\kappa}{\sigma_Y} + \frac{64n^{3.5}}{\sigma_Y^2}\right)}$, *then* $\mathbf{p}^T \hat{\mathbf{x}}_{\hat{i}} \geq \delta$ *for the optimal* $\mathbf{p}$ *to the QP (4.30).*

Proof. Suppose not. Let $\mathbf{p}^T \hat{\mathbf{x}}_{\hat{i}} < \delta$. Then from the primal constraints, $s_{\hat{i}} = 1 - \delta$ and $t_{\hat{i}} > 0$. Since we are looking at the optimizer, the complementary slackness conditions (4.7) and (4.8) combined with the dual constraints (4.31) yields $q_{\hat{i}} - r_{\hat{i}} = 1$ and $q_{\hat{i}} - w_{\hat{i}} = c$. If $\hat{X} = [\hat{\mathbf{x}}_1 \ldots, \hat{\mathbf{x}}_n] \in \mathbb{R}^{m \times n}$ then (4.18) is:

$$
\begin{aligned}
\left\|\hat{X}\mathbf{q}\right\| &\leq \|X\mathbf{q}\| + \left\|\sum_{i=1}^{n} \boldsymbol{\varepsilon}_i q_i\right\| \\
&\leq 2\sqrt{\lambda n} + nc^2\varepsilon. \quad (\because \hat{q}_i \leq c^2 \text{ by (4.31)})
\end{aligned} \tag{4.32}
$$

Consider the $\hat{\mathbf{p}}$ from Lemma 15:

$$
\begin{aligned}
\hat{\mathbf{p}}^T \hat{X}\mathbf{q} &\leq \|\hat{\mathbf{p}}\| \left\|\hat{X}\mathbf{q}\right\| \\
\implies \sum_{i=1}^{n} (\hat{\mathbf{p}}^T \hat{\mathbf{x}}_i q_i) &\leq 2\sqrt{\lambda n} + nc^2\varepsilon \quad (\because (4.32)) \\
\implies \sum_{i=1}^{n} (\hat{\mathbf{p}}^T \mathbf{x}_i q_i - \|\hat{\mathbf{p}}\| \|\varepsilon\| q_i) \leq \sum_{i=1}^{n} (\hat{\mathbf{p}}^T \hat{\mathbf{x}}_i q_i) &\leq 2\sqrt{\lambda n} + nc^2\varepsilon \\
\implies \sum_{i=1}^{n} (\hat{\mathbf{p}}^T \mathbf{x}_i q_i) &\leq 2\sqrt{\lambda n} + 2nc^2\varepsilon. \quad (\because \hat{q}_i \leq c^2)
\end{aligned} \tag{4.33}
$$

From Lemma 15, we can say that $\hat{\mathbf{p}}^T\mathbf{x}_i \geq 0$ for all i. Using Lemma 15 and the fact that $q_{\hat{i}} = w_{\hat{i}} + c \geq c > 0 \;\; (\because w_{\hat{i}} \geq 0)$,

$$\sum_{i=1}^{n}(\hat{\mathbf{p}}^T\mathbf{x}_i)q_i \geq \hat{\mathbf{p}}^T\mathbf{x}_{\hat{i}}q_{\hat{i}} \geq c\kappa \text{ dist}(\hat{\mathbf{x}}_{\hat{i}} - \boldsymbol{\varepsilon}_{\hat{i}}, L) \geq c\kappa\eta - c\kappa\varepsilon = 4\sqrt{\lambda n} - c\kappa\varepsilon. \;\; (\because (4.1)) \;\; (4.34)$$

Using this in equation (4.33), we get

$$\begin{aligned} & 4\sqrt{\lambda n} - c\kappa\varepsilon \leq 2\sqrt{\lambda n} + 2nc^2\varepsilon \\ & \implies 2\sqrt{\lambda n} \leq (c\kappa + 2nc^2)\varepsilon = \left(\frac{8n^{3/2}\kappa}{\sigma_Y} + \frac{128n^4}{\sigma_Y^2}\right)\varepsilon \;\; (\because (4.1)) \\ \implies & \frac{\sqrt{\lambda}}{\left(\frac{4n\kappa}{\sigma_Y} + \frac{64n^{3.5}}{\sigma_Y^2}\right)} \leq \varepsilon. \end{aligned} \tag{4.35}$$

Since we have the noise bound

$$\varepsilon < \frac{\sqrt{\lambda}}{\left(\frac{4n\kappa}{\sigma_Y} + \frac{64n^{3.5}}{\sigma_Y^2}\right)} \tag{4.36}$$

(4.35) is violated. This leads to a contradiction and hence the assumption that $\mathbf{p}^T\hat{\mathbf{x}}_{\hat{i}} < \delta$ is false. This completes the proof. □

We prove a theorem analogous to Theorem 17. Even for the optimal $\mathbf{p}$ to the noisy problem (4.30), the component along L of the optimal $\mathbf{p}$ is small.

Theorem 20. *The magnitude of the optimal* $\mathbf{p}$ *to the QP (4.30) has an upper bound of* $\sqrt{\frac{4n}{\lambda}}$. *Furthermore, the component of* $\mathbf{p}$ *along* L *is of magnitude at most* $\frac{\sigma_Y}{32n^{3/2}} + \varepsilon\frac{2n}{\sigma_Y\sqrt{\lambda}}$.

Proof. For the optimal $\mathbf{p}$ to the QP (4.30), the objective value is at least $\frac{\lambda}{2}\|\mathbf{p}\|^2$ (since other terms are nonnegative). At the feasible point where $\mathbf{p} = \mathbf{0}$, we have $s_i = 1-\delta$, $t_i = \delta$ and $u_i = 0$ for all $i \in [n]$. The objective value at this point is not lower than the objective at the optimizer. Therefore,

$$\begin{aligned} \frac{\lambda}{2}\|\mathbf{p}\|^2 &\leq n(1-\delta) + nc\delta \\ &\leq n + nc\delta \;\; (\because n\delta > 0) \\ &= 2n \;\; (\because \delta c = 1 \text{ from } (4.1)) \\ \implies \|\mathbf{p}\| &\leq \sqrt{\frac{4n}{\lambda}}. \end{aligned} \tag{4.37}$$

For each $i \in [n]$, we claim the following lower bound:

$$\mathbf{p}^T \hat{\mathbf{x}}_i \geq -\frac{2n}{c^2} = -\frac{\sigma_Y^2}{32n^2}. \tag{4.38}$$

To prove this, assume that for some $j \in [n]$ (4.38) does not hold. Then $\mathbf{p}^T \hat{\mathbf{x}}_i < -\frac{2n}{c^2}$. The first constraint of the primal (4.30) implies $s_j = 1 - \delta,\ t_j = \delta$ and $u_j = -\mathbf{p}^T \hat{\mathbf{x}}_j > \frac{2n}{c^2}$. Since all terms in the objective are nonnegative, the objective at the optimizer is at least $c^2 u_j$. Using the value of the objective at the feasible point $\mathbf{p} = 0$ as an upper bound,

$$2n < c^2 u_j \leq n(1-\delta) + nc\delta < 2n,$$

which is a contradiction and hence (4.38) holds.

Let the optimal $\mathbf{p}$ be written as $\mathbf{p} = \underbrace{\mathbf{p}_L}_{\in L} + \underbrace{\mathbf{p}_{L^\perp}}_{\in L^\perp}$. Then for all $i \in [l]$ we have $\mathbf{x}_i \in L$ and hence $\mathbf{p}_{L^\perp}^T \mathbf{x}_i = 0$. Therefore,

$$\mathbf{p}^T \mathbf{x}_i = \mathbf{p}_L^T \mathbf{x}_i \ \ \forall\ i \in [l]. \tag{4.39}$$

We use the well centering condition in L (3.3). Since $\mathbf{0}$ is in the convex hull of $\mathbf{x}_2, \ldots, \mathbf{x}_l$, there exists nonnegative coefficients $\{\alpha_i\}_{i=2}^l$ such that $\sum_{i=2}^l \alpha_i = 1$ and $\sum_{i=2}^l \alpha_i \mathbf{x}_i = \mathbf{0}$.

$$\begin{aligned}
&\sum_{i=2}^{l} \alpha_i \mathbf{x}_i = \mathbf{0} \\
\implies &\mathbf{p}^T \sum_{i=2}^{l} \alpha_i \mathbf{x}_i = 0 \\
\implies &\sum_{i=2}^{l} \alpha_i \mathbf{p}^T \mathbf{x}_i = 0 \\
\implies &\sum_{i=2}^{l} \alpha_i \mathbf{p}_L^T \mathbf{x}_i = 0 \ \ (\because (4.39)) \\
&\text{Let } \mathbf{y}_i = \alpha_i \mathbf{x}_i \ \ \forall\ i \in \{2, \ldots, l\} \\
\implies &\sum_{i=2}^{l} \mathbf{p}_L^T \mathbf{y}_i = 0.
\end{aligned}$$

Let

$$\mu \triangleq \min_{i \in \{2,\ldots,l\}} \mathbf{p}_L^T \mathbf{x}_i = \mathbf{p}_L^T \mathbf{x}_j \ \text{ for some } j \in \{2, \ldots, l\}. \tag{4.40}$$

Clearly, $\mu \leq 0$. Then for all $g \in \{2, \ldots, l\}$,

$$\begin{aligned}
\mathbf{p}_L^T \mathbf{y}_g &= -\sum_{\substack{i=2 \\ i \neq g}}^{l} \mathbf{p}_L^T \mathbf{y}_i \\
\alpha_g \mu \leq \mathbf{p}_L^T \mathbf{y}_g &\leq -\Big(\sum_{\substack{i=2 \\ i \neq g}}^{l} \alpha_i \Big) \mu \leq -\mu \quad (\because (4.25)) \\
\implies \mathbf{p}_L^T \mathbf{y}_g &\in [\alpha_g \mu, -\mu] \\
\implies |\mathbf{p}_L^T \mathbf{y}_g| &\leq -\mu = |\mu|. \qquad (4.41)
\end{aligned}$$

Let $Y = [\mathbf{y}_1, \ldots, \mathbf{y}_l]^T$. From (3.4), we have this singular value condition on the data,

$$\begin{aligned}
\sigma_Y &= \inf_{\substack{\|\mathbf{x}\|=1 \\ \mathbf{x} \in L}} \left\| Y^T \mathbf{x} \right\| \\
&\leq \left\| Y^T \frac{\mathbf{p}_L}{\|\mathbf{p}_L\|} \right\| \quad (\because \mathbf{p}_L \in L) \\
&= \sqrt{\sum_{i=1}^{l} \left(\mathbf{y}_i^T \frac{\mathbf{p}_L}{\|\mathbf{p}_L\|} \right)^2} \\
&\leq \sqrt{\sum_{i=1}^{l} \frac{\mu^2}{\|\mathbf{p}_L\|^2}} \quad (\because (4.26)) \\
&= \frac{|\mu| \sqrt{l}}{\|\mathbf{p}_L\|} \\
&\leq \frac{|\mu| \sqrt{n}}{\|\mathbf{p}_L\|} \\
\implies \|\mathbf{p}_L\| &\leq \frac{|\mu| \sqrt{n}}{\sigma_Y}. \qquad (4.42)
\end{aligned}$$

In order to get a good upper bound on $\|\mathbf{p}_L\|$, we need an upper bound on $|\mu|$. We know

that for some j, $\mu = \mathbf{p}_L^T \mathbf{x}_j \leq 0$. Using (4.38),

$$\begin{aligned}
\mu &= \mathbf{p}_L^T \mathbf{x}_j \\
&= \mathbf{p}^T \hat{\mathbf{x}}_j - \mathbf{p}^T \boldsymbol{\varepsilon}_j \quad (\because (4.39)) \\
&\geq -\frac{\sigma_Y^2}{32n^2} - \varepsilon\sqrt{\frac{4n}{\lambda}} \quad (\because (4.38), (4.37) \text{ and Cauchy-Schwarz inequality}) \\
\implies |\mu| &\leq \frac{\sigma_Y^2}{32n^2} + \varepsilon\sqrt{\frac{4n}{\lambda}}. \qquad (4.43)
\end{aligned}$$

Using (4.43) in (4.42),

$$\|\mathbf{p}_L\| \leq \frac{\sigma_Y}{32n^{3/2}} + \varepsilon\frac{2n}{\sigma_Y\sqrt{\lambda}}. \qquad (4.44)$$

□

Similar to Theorem 18, the following theorem is a converse of Theorem 19, except for a change from η to η'. For η' given in (4.1),

Theorem 21. *Suppose $\mathbf{p}^T \hat{\mathbf{x}}_{\hat{i}} \geq \delta$ for the optimal $\mathbf{p}$ to the QP (4.30) and for some $\hat{i} \in [n]$. If the noise upper bound satisfies $\varepsilon \leq \min\left(\frac{\frac{\sigma_Y}{32n^{3/2}}}{\left(\frac{\sigma_Y}{16n^{3/2}} + \sqrt{\frac{4n}{\lambda}}\right)}, \frac{\sigma_Y^2}{64n^{5/2}}\sqrt{\lambda}\right)$, then $dist(\hat{\mathbf{x}}_{\hat{i}}, L) \geq \eta'$.*

Proof. Suppose $\mathbf{p}^T \hat{\mathbf{x}}_{\hat{i}} \geq \delta$. For any $1 \leq \hat{i} \leq n$,

$$\begin{aligned}
\frac{\sigma_Y}{8n^{3/2}} = \delta &\leq \mathbf{p}^T \hat{\mathbf{x}}_{\hat{i}} \quad (\because (4.1)) \\
&= \mathbf{p}_L^T \hat{\mathbf{x}}_{\hat{i}} + \mathbf{p}_{L^\perp}^T \hat{\mathbf{x}}_{\hat{i}} \\
&\leq ||\mathbf{p}_L||.||\hat{\mathbf{x}}_{\hat{i}}|| + \mathbf{p}_{L^\perp}^T \hat{\mathbf{x}}_{\hat{i}} \\
&\leq \left(\frac{\sigma_Y}{32n^{3/2}} + \varepsilon\frac{2n}{\sigma_Y\sqrt{\lambda}}\right)(1+\varepsilon) + \mathbf{p}_{L^\perp}^T \mathbf{x}_{\hat{i}} + \mathbf{p}_{L^\perp}^T \boldsymbol{\varepsilon}_{\hat{i}} \quad (\because 3.7, \text{Theorem 20}) \\
&\leq \frac{\sigma_Y}{16n^{3/2}}(1+\varepsilon) + \mathbf{p}_{L^\perp}^T \mathbf{x}_{\hat{i}} + \mathbf{p}_{L^\perp}^T \boldsymbol{\varepsilon}_{\hat{i}}. \quad \left(\because \ \varepsilon \leq \frac{\sigma_Y^2}{64n^{5/2}}\sqrt{\lambda}\right)
\end{aligned}$$

Suppose we split $\mathbf{x}_{\hat{i}} = \mathbf{x}_{\hat{i},L} + \mathbf{x}_{\hat{i},L^\perp}$ where $\mathbf{x}_{\hat{i},L} \in L$ and $\mathbf{x}_{\hat{i},L^\perp} \in L^\perp$. Then $\mathbf{p}_{L^\perp} \perp \mathbf{x}_{\hat{i},L}$.

Therefore, $\mathbf{p}_{L^\perp}^T \mathbf{x}_{\hat{i}} = \mathbf{p}_{L^\perp}^T \mathbf{x}_{\hat{i},L^\perp}$ and hence,

$$\begin{aligned}
\frac{\sigma_Y}{8n^{3/2}} - \frac{\sigma_Y}{16n^{3/2}}(1+\varepsilon) &\leq \mathbf{p}_{L^\perp}^T \mathbf{x}_{\hat{i},L^\perp} + \mathbf{p}_{L^\perp}^T \boldsymbol{\varepsilon}_{\hat{i}} \qquad (4.45)\\
&\leq \|\mathbf{p}_{L^\perp}\| .(\left\|\mathbf{x}_{\hat{i},L^\perp}\right\| + \varepsilon)\\
&\leq \sqrt{\frac{4n}{\lambda}}.(\left\|\mathbf{x}_{\hat{i},L^\perp}\right\| + \varepsilon) \quad (\because \text{Theorem 17})\\
\implies \frac{\frac{\sigma_Y}{16n^{3/2}} - \varepsilon\frac{\sigma_Y}{16n^{3/2}}}{\sqrt{\frac{4n}{\lambda}}} - \varepsilon &\leq \left\|\mathbf{x}_{\hat{i},L^\perp}\right\| = \ \text{dist}(\mathbf{x}_{\hat{i}}, L)\\
\implies \frac{\frac{\sigma_Y}{16n^{3/2}} - \varepsilon\frac{\sigma_Y}{16n^{3/2}} - \varepsilon\sqrt{\frac{4n}{\lambda}}}{\sqrt{\frac{4n}{\lambda}}} &\leq \ \text{dist}(\mathbf{x}_{\hat{i}}, L). \qquad (4.46)
\end{aligned}$$

Since we have the noise upper bound,

$$\varepsilon < \frac{\frac{\sigma_Y}{32n^{3/2}}}{\left(\frac{\sigma_Y}{16n^{3/2}} + \sqrt{\frac{4n}{\lambda}}\right)}$$

(4.46) reduces to the desired inequality:

$$\eta' = \frac{\sigma_Y}{64n^2}\sqrt{\lambda} \leq \ \text{dist}(\mathbf{x}_{\hat{i}}, L).$$

□

In order for both Theorem 19 and Theorem 21 to work, a good choice of ε would be

$$\begin{aligned}
\varepsilon &< \min\left(\frac{\sqrt{\lambda}}{\left(\frac{4n\kappa}{\sigma_Y} + \frac{64n^{3.5}}{\sigma_Y^2}\right)}, \frac{\frac{\sigma_Y}{32n^{3/2}}}{\left(\frac{\sigma_Y}{16n^{3/2}} + \sqrt{\frac{4n}{\lambda}}\right)}, \frac{\sigma_Y^2}{64n^{5/2}}\sqrt{\lambda}\right)\\
&= \min\left(\frac{\sigma_Y^2\sqrt{\lambda}}{4n(\kappa\sigma_Y + 16n^{2.5})}, \frac{\sigma_Y\sqrt{\lambda}}{64n^2 + 2\sqrt{\lambda}\sigma_Y}, \frac{\sigma_Y^2}{64n^{5/2}}\sqrt{\lambda}\right)\\
&= \min\left(\frac{\sigma_Y^2\sqrt{\lambda}}{4n(\kappa\sigma_Y + 16n^{2.5})}, \frac{\sigma_Y\sqrt{\lambda}}{64n^2 + 2\sqrt{\lambda}\sigma_Y}\right) \quad \left(\because \ \frac{\sigma_Y^2\sqrt{\lambda}}{4n(\kappa\sigma_Y + 16n^{2.5})} < \frac{\sigma_Y^2}{64n^{5/2}}\sqrt{\lambda}\right)\\
\implies \varepsilon &< \min\left(\frac{\sigma_Y^2\sqrt{\lambda}}{4n(\kappa\sigma_Y + 16n^{2.5})}, \frac{\sigma_Y\sqrt{\lambda}}{64n^2 + 2\sqrt{\lambda}\sigma_Y}\right) \qquad (4.47)
\end{aligned}$$

4.4 Post processing

Suppose we have the noisy data points and we solve the QP (4.30) efficiently. Now we obtain the optimal $\mathbf{p}$ to the QP and partition the data points according to whether $\mathbf{p}^T\hat{\mathbf{x}}_i \leq \delta$ or $\mathbf{p}^T\hat{\mathbf{x}}_i > \delta$. The Theorems 19 and 21 tell us how close each of the partitions are to L. If ε is small enough so that

$$\varepsilon < \eta' \tag{4.48}$$

then from Theorem 21 we can see for all $i \in [l]$,

$$\text{dist}(\hat{\mathbf{x}}_i, L) < \eta' \implies \mathbf{p}^T\hat{\mathbf{x}}_i < \delta.$$

Let $N = \{\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_l, \hat{\mathbf{x}}_{l+1}, \ldots, \hat{\mathbf{x}}_q\}$ be the set of all noisy data points such that $\mathbf{p}^T\hat{\mathbf{x}}_i < \delta$. If $\dim(L) = k$, then we show that the kth singular value of $[\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_q]$ is large. Recall that $\sum_{i=1}^{l} \alpha_i = 1$ and $\alpha_i \geq 0$ in the well centering condition (3.3).

$$\begin{aligned}
\sigma_k([\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_q]) &\geq \sigma_k([\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_l]) \quad (\because \text{ Lemma 5}) \\
&= \inf_{\text{rank}(B) \leq k-1} \|[\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_l] - B\|_2 \quad (\because \text{ Theorem 3}) \\
&\geq \inf_{\text{rank}(B) \leq k-1} \|[\alpha_1\hat{\mathbf{x}}_1, \ldots, \alpha_l\hat{\mathbf{x}}_l] - [\alpha_1 B(:,1) \ldots \alpha_l B(:,l)]\|_2 \quad (\because \text{ Lemma 6}) \\
&= \inf_{\text{rank}(B) \leq k-1} \Bigg(\|[\alpha_1\mathbf{x}_1, \ldots, \alpha_l\mathbf{x}_l] - [\alpha_1 B(:,1) \ldots \alpha_l B(:,l)]\|_2 \\
&\qquad\qquad - \|[\alpha_1\mathbf{x}_1, \ldots, \alpha_l\mathbf{x}_l] - [\alpha_1\hat{\mathbf{x}}_1, \ldots, \alpha_l\hat{\mathbf{x}}_l]\|_2 \Bigg) \\
&\geq \sigma_Y - \|[\alpha_1\boldsymbol{\varepsilon}_1, \ldots, \alpha_l\boldsymbol{\varepsilon}_l]\|_2 \quad (\because \text{ Definition of } \sigma_Y) \\
&\geq \sigma_Y - (\max_i |\alpha_i|) \, \|[\boldsymbol{\varepsilon}_1, \ldots, \boldsymbol{\varepsilon}_l]\|_2 \quad (\because \text{ Lemma 6}) \\
\implies \sigma_k([\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_q]) &\geq \sigma_Y - \varepsilon.
\end{aligned} \tag{4.49}$$

Since $\mathbf{p}^T\hat{\mathbf{x}}_i < \delta$ for all points in N, by Theorem 19 we have $\text{dist}(\hat{\mathbf{x}}_i, L) < \eta$. Let $B \in \mathbb{R}^{m \times k}$ be a basis of L. Then each point in N can be written as $\hat{\mathbf{x}}_i = \underbrace{B\mathbf{f}_i}_{\in L} + \underbrace{\mathbf{n}_i}_{\in L^\perp}$. We show that

the $(k+1)$th singular value of $[\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_q]$ is small.

$$\begin{aligned}
\sigma_{k+1}([\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_q]) &= \sigma_{k+1}([B\mathbf{f}_1 + \mathbf{n}_1, \ldots, B\mathbf{f}_q + \mathbf{n}_q]) \\
&= \sigma_{k+1}([B\mathbf{f}_1, \ldots, B\mathbf{f}_q] + [\mathbf{n}_1, \ldots, \mathbf{n}_q]) \\
&\leq \sigma_{k+1}(\underbrace{[B\mathbf{f}_1, \ldots, B\mathbf{f}_q]}_{rank \quad k}) + \|[\mathbf{n}_1, \ldots, \mathbf{n}_q]\| \\
&\leq \eta \quad (\because \|\mathbf{n}_i\| \leq \eta, \quad \forall\ 1 \leq i \leq q) \\
\sigma_{k+1}([\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_q]) &\leq \frac{\sigma_Y}{2n\kappa}\sqrt{\lambda}. \qquad (4.50)
\end{aligned}$$

Note that the noise bound (4.47) combined with (4.48) yields,

$$\varepsilon < \min\left(\frac{\sigma_Y}{64n^2}\sqrt{\lambda}, \frac{\sigma_Y^2\sqrt{\lambda}}{4n(\kappa\sigma_Y + 16n^{2.5})}, \frac{\sigma_Y\sqrt{\lambda}}{64n^2 + 2\sqrt{\lambda}\sigma_Y}\right).$$

But $\frac{\sigma_Y\sqrt{\lambda}}{64n^2+2\sqrt{\lambda}\sigma_Y} < \frac{\sigma_Y}{64n^2}\sqrt{\lambda}$ and hence,

$$\varepsilon < \min\left(\frac{\sigma_Y^2\sqrt{\lambda}}{4n(\kappa\sigma_Y + 16n^{2.5})}, \frac{\sigma_Y\sqrt{\lambda}}{64n^2 + 2\sqrt{\lambda}\sigma_Y}\right). \qquad (4.51)$$

We make our choice of λ now,

$$\lambda = \frac{1024n^4\kappa^2}{(32n+\kappa)^2}. \qquad (4.52)$$

This ensures that

$$\begin{aligned}
\frac{\sigma_Y}{2n\kappa}\sqrt{\lambda} &= \frac{\sigma_Y}{2n\kappa} \times \frac{32n^2\kappa}{(32n+\kappa)} \\
&< \frac{\sigma_Y}{2n\kappa} \times \frac{64n^2\kappa}{(32n+\kappa)} \\
&= \frac{32n\sigma_Y}{(32n+\kappa)} \\
&= \sigma_Y - \frac{\kappa\sigma_Y}{(32n+\kappa)} \\
&= \sigma_Y - \frac{\sigma_Y}{32n^2}\sqrt{\lambda} \quad (\because (4.52)) \\
&< \sigma_Y - \frac{\sigma_Y}{64n^2}\sqrt{\lambda} \\
&< \sigma_Y - \varepsilon. \quad (\because (4.51))
\end{aligned}$$

Combine this with equations (4.49) and (4.50) and use the value of λ in (4.52)

$$\sigma_{k+1}([\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_q]) \leq \frac{16n\sigma_Y}{(32n+\kappa)} < \sigma_Y - \varepsilon \leq \sigma_k([\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_q]). \tag{4.53}$$

We have shown that for the matrix $[\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_q]$, the kth singular value is large and the $(k+1)$th singular value is small. It is clear that by choosing a smaller value of λ, one can increase this gap arbitrarily, at the cost of placing a stricter upper bound on ε.

We update the constants (since we have picked λ) as follows:

Constant	c	δ	η	η'
Value	$\frac{8n^{3/2}}{\sigma_Y}$	$\frac{\sigma_Y}{8n^{3/2}}$	$\frac{16n\sigma_Y}{(32n+\kappa)}$	$\frac{\kappa\sigma_Y}{2(32n+\kappa)}$

(4.54)

Substituting (4.52) in (4.51), the noise upper bound should satisfy the following inequality for the above method to work:

$$\varepsilon < \min\left(\frac{8\sigma_Y^2 n\kappa}{(32n+\kappa)(\kappa\sigma_Y + 16n^{2.5})}, \frac{\sigma_Y\kappa}{64n+2\kappa+2\kappa\sigma_Y}\right). \tag{4.55}$$

4.5 Summary

We summarize our algorithm, which is a replacement for Algorithm 1.

Algorithm 5 Finding one properly filled face under well-centering assumptions

1: **Input**: Set of noisy points $\{\hat{\mathbf{x}}_i\}_{i=1}^n$ in $\mathbb{R}^m$. The magnitude of the noise has an upper bound satisfying (4.55). It is known that a properly filled face L of dimension k contains some data points obeying the well-centering assumptions. The points are scaled and translated as described in Section 3.1.

2: **Output**: We compute k, the dimension of L and recover the subspace L approximately.

3: For parameters described in Section 3.1, solve the QP (3.1) and obtain any optimizer. Let $\mathbf{p}^*$ be the optimal value of the variable $\mathbf{p}$.

4: Let $N = [\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_q]$ be a matrix comprised of all data points which obey $(\mathbf{p}^*)^T \hat{\mathbf{x}}_i < \delta$ (The parameter δ is given in Section 3.1).

5: The number of singular values of $[\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_q]$ greater than $\frac{16n\sigma_Y}{(32n+\kappa)}$ (due to (4.53)) gives k, the dimension of L.

6: In the noiseless setting, the singular vectors corresponding to the largest k singular values give an exact basis for L as $\delta \to 0$. When there is bounded noise in the data, the singular vectors corresponding to the largest k singular values give an approximate basis for L.

If there is no noise in the data, we can exactly obtain L using Step 6. With bounded noise, we can still obtain an approximation to L (we obtain all data points within distance η to L). The quality of the approximation depends on the noise.

Note that Algorithm 1 of Ge and Zou [17] is a quadratically constrained quadratic program (QCQP) because of the constraint $\left\| \mathbf{x}_0 - \sum_{i=1}^h w_i \mathbf{x}_i \right\| \leq 2\varepsilon$ and needs to be solved iteratively for each face. Algorithm 5 is a quadratic program (QP) which is less expensive to solve than a QCQP. Our algorithm is only a single pass through a QP.

Chapter 5

Experiments and Conclusions

5.1 Experiments

We tested Algorithm 5 for simulated data and observed that the algorithm performs well even under presence of noise. The data was simulated by sampling from a Dirichlet distribution, a continuous multivariable distribution which is parameterized by a vector of positive reals $\boldsymbol{\alpha}$ of the same dimension of the data. Sampling from the Dirichlet distribution is done using a Gamma distribution sampler. We sample from the Gamma distribution as many times as the dimension of the data, using the corresponding α_i as the Gamma parameter. The resulting vector is then normalized, so that the components sum to 1. For noisy data, bounded noise was sampled from an uniform distribution and added to the data.

In our test case, we generate a set of data points lying on L as follows. Given k, the dimension of L we choose a value of 100 for $(k+1)$ components of $\boldsymbol{\alpha}$ and 0 for the other components. The Dirichlet distribution yields a number of data points (say n_1) which contain exactly $(k+1)$ non zero components lying inside the unit simplex. In the noisy case, we choose a noise upper bond ε and generate a $(m \times n_1)$ matrix of values sampled from the uniform distribution. We scale this by ε to keep the noise bound and add the noise to the data.

Next, we generate n_2 data points corresponding to those not in L using the Dirichlet distribution. For this purpose, we set an arbitrary number $(< m)$ of components of $\boldsymbol{\alpha}$ to be 100 and the rest to be 1. This gives us points which are nonzero in every component but these do not necessarily lie in L. The total number of data points is $n = n_1 + n_2$.

An appropriate data point was chosen to be the origin. In the noisy case, since even points in the first set have lots of nonzero components, we picked a point which had exactly $(k+1)$ components above some threshold. We start the threshold at some small value (say 10^{-4}) and increase it by a factor of 10 until we find a point satisfying the above condition. All data points were translated so that this point becomes the origin, to satisfy the well-centering assumption. We used MATLAB 2016b Version 9.1 for implementing the algorithm. The built-in quadratic program solver `quadprog` in MATLAB uses an interior point method to solve the quadratic problem in Algorithm 5.

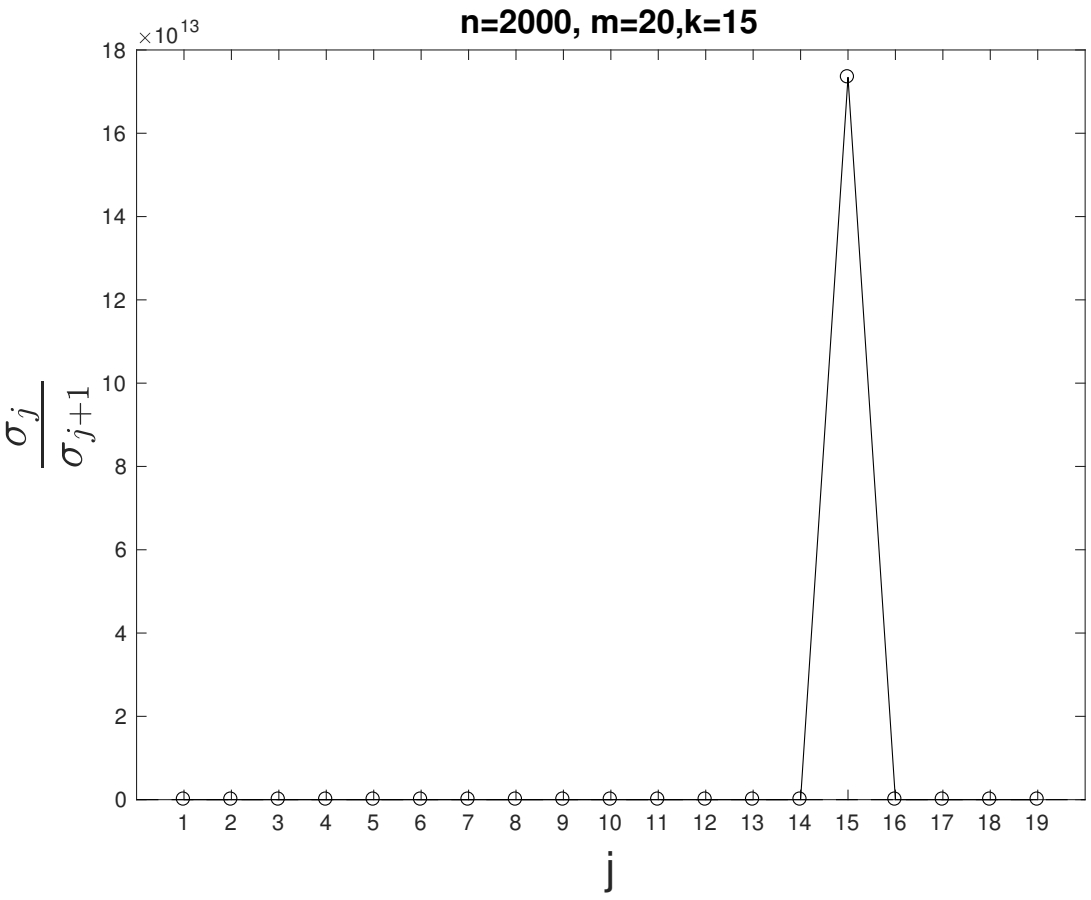

Figure 5.1: Plot of ratio of consecutive singular values. Data without noise for 2000 data points in $\mathbb{R}^{20}$. The subspace L has dimension $k = 15$.

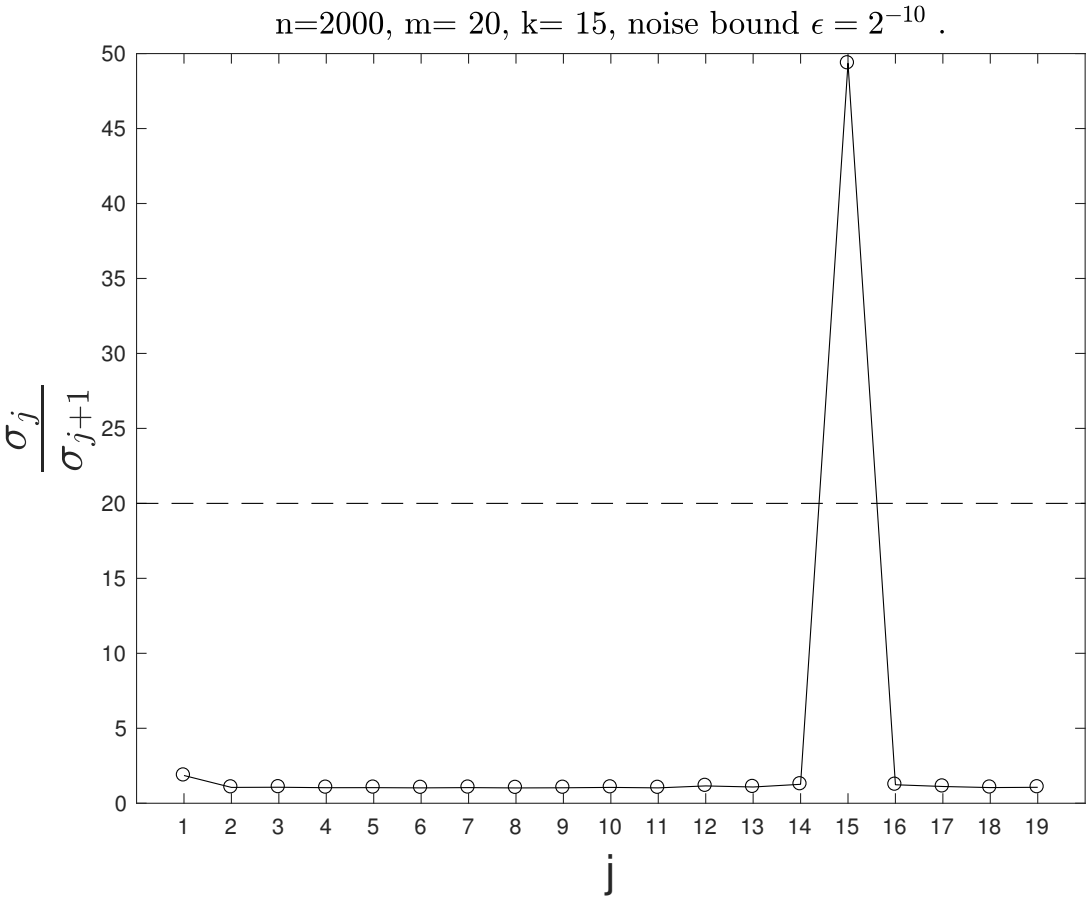

Figure 5.2: Plot of ratio of consecutive singular values. Bounded noise ($\varepsilon = 2^{-10} \approx 10^{-3}$) is added to the data used in Figure 5.1. The subspace L has dimension $k = 15$.

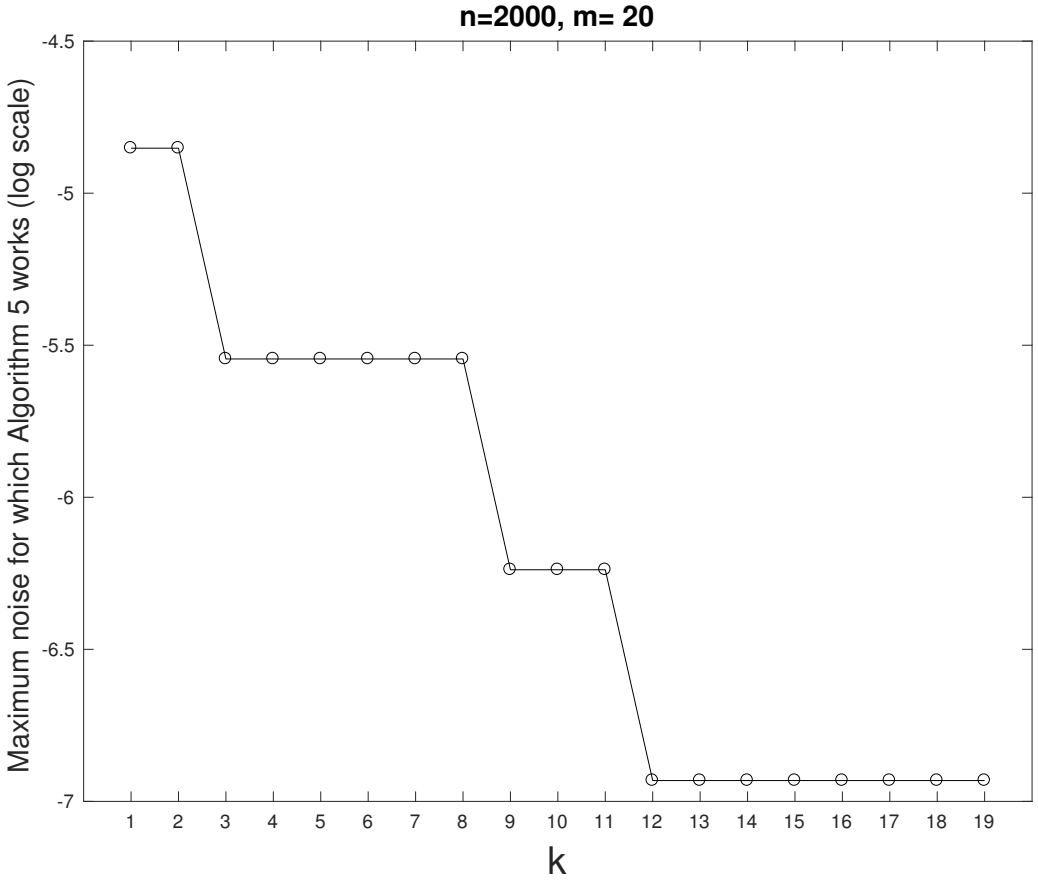

Figure 5.3: Log plot of maximum noise tolerated by Algorithm 5 in order to get a clear threshold of 20 in the ratio of singular values. The maximum noise tolerated for each subspace dimension $k \in [1, 19]$ is shown.

Figure 5.1 is an example of a data without noise, with 2000 points in $\mathbb{R}^{20}$. For the dimension $k = 15$ of the subspace L, we can see a big jump in the ratio of consecutive singular values of the matrix in Step 4 of Algorithm 5. The jump occurs at the index 15, i.e., σ_{15}/σ_{16} is large. We seek a jump above some threshold (say 40) in order to guess the dimension. There is a jump above the threshold for the 15th ratio in Figure 5.1.

Figure 5.2 is the same data as above, but bounded noise is added. The upper bound of the noise is $\varepsilon = 2^{-10} \approx 10^{-3}$. with 2000 points in $\mathbb{R}^{20}$. For the dimension $k = 15$ of the subspace L, we can see a big jump in the ratio of consecutive singular values of the matrix in Step 4 of Algorithm 5. The jump occurs at the index 15, i.e., σ_{15}/σ_{16} is large. We seek a jump above a lower threshold than the noiseless case (say 20) in order to guess the dimension. Notice the jump above the threshold for the 15th ratio in Figure 5.2.

The ratio test was introduced because we do not know σ_Y in practice, so we cannot use Step 5 of Algorithm 5 in a practical setting. Figures 5.1 and 5.2 show that the ratio test is sensible in practice.

Figure 5.3 shows the maximum noise for which Algorithm 5 identifies the dimension of L with the threshold 20 for jump in the ratio of consecutive singular values. We obtain the maximum noise tolerated for dimensions of L from $1, \ldots, m-1$. The plot is in the log scale and we observe that as the dimension k increases, there is a decrease in the maximum noise for which the Algorithm 5. This shows that it is harder to find high-dimensional filled faces, i.e., we can only find them for a noise level lower than low-dimensional faces. This is expected since for random points in a high-dimensional face, it is less likely that one of the points is well-centered.

5.2 Conclusions

We considered the work of Ge and Zou [17] who introduced the notion of subset-separability, which is a milder assumption than separability. Moreover, this is a necessary but not sufficient condition for the W-simplex to be volume minimizing (and hence unique). The most expensive step in their *Face-Intersect* algorithm is the problem of finding a properly filled face given the center point. Our replacement, Algorithm 5 makes assumptions about well-centering of the data (as does Algorithm 1) and produces an efficient quadratic program for finding a filled face, even in the presence of bounded noise. For a linearly constrained quadratic program, the number of iterations of a good interior point method is approximately $O(\sqrt{n}\log(\frac{1}{\varepsilon}))$ with each iteration $O(v^2 n)$ where ε is the error in computed value compared to actual value, n is the number of polyhedral constraints and v is the number of variables [49]. The noisy NMF problem can be solved using Algorithm 5 and the rest of the procedure due to Ge and Zou [17] which was explained in Chapter 2.

Theoretically, one possible improvement to our algorithm could involve simplifying the

QP formulation. The 2-norm in the objective is to control the magnitude of the variable $\mathbf{p}$. One could possibly use the 1-norm or the infinity norm and make the QP into a linear program. Note that this results in little change in the time complexity, which seems to depend only on the number of polyhedral constraints. Also the test in Step 5 of Algorithm 5 is complicated in practice, because we do not usually know σ_Y from the data.

In this thesis, our algorithm was shown in action for simulated data in both noiseless and noisy settings. Practically, our algorithm could be incorporated into the NMF procedure and tested in experiments using real data to solve a NMF problem.

References

[1] Sanjeev Arora, Rong Ge, Yonatan Halpern, David Mimno, Ankur Moitra, David Sontag, Yichen Wu, and Michael Zhu. A practical algorithm for topic modeling with provable guarantees. In *International Conference on Machine Learning*, pages 280–288, 2013.

[2] Sanjeev Arora, Rong Ge, Ravindran Kannan, and Ankur Moitra. Computing a nonnegative matrix factorization–provably. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 145–162. ACM, 2012.

[3] Abraham Berman and Robert J Plemmons. Nonnegative matrices in the mathematical sciences. Reprinted in 1994 by *Society for Industrial and Applied Mathematics*, 1979.

[4] Chiranjib Bhattacharya, Navin Goyal, Ravindran Kannan, and Jagdeep Pani. Nonnegative matrix factorization under heavy noise. In *International Conference on Machine Learning*, pages 1426–1434, 2016.

[5] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[6] Stephen L Campbell and George D Poole. Computing nonnegative rank factorizations. *Linear Algebra and its Applications*, 35:175–182, 1981.

[7] Ji-Cheng Chen. The nonnegative rank factorizations of nonnegative matrices. *Linear algebra and its applications*, 62:207–217, 1984.

[8] Eric C Chi and Tamara G Kolda. On tensors, sparsity, and nonnegative factorizations. *SIAM Journal on Matrix Analysis and Applications*, 33(4):1272–1299, 2012.

[9] Fan RK Chung. *Spectral graph theory*. American Mathematical Soc., 1997. 92.

[10] Pierre Comon. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314, 1994.

[11] Adele Cutler and Leo Breiman. Archetypal analysis. *Technometrics*, 36(4):338–347, 1994.

[12] Alexandre d'Aspremont, Laurent E Ghaoui, Michael I Jordan, and Gert R Lanckriet. A direct formulation for sparse PCA using semidefinite programming. In *Advances in neural information processing systems*, pages 41–48, 2005.

[13] Karthik Devarajan. Nonnegative matrix factorization: an analytical and interpretive tool in computational biology. *PLoS computational biology*, 4(7):e1000029, 2008.

[14] David Donoho and Victoria Stodden. When does non-negative matrix factorization give a correct decomposition into parts? In *Advances in neural information processing systems*, pages 1141–1148, 2004.

[15] Lars Eldén. *Matrix methods in data mining and pattern recognition*. SIAM, 2007.

[16] Cédric Févotte, Nancy Bertin, and Jean-Louis Durrieu. Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis. *Neural computation*, 21(3):793–830, 2009.

[17] Rong Ge and James Zou. Intersecting faces: Non-negative matrix factorization with new guarantees. In David Blei and Francis Bach, editors, *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 2295–2303. JMLR Workshop and Conference Proceedings, 2015.

[18] Nicolas Gillis. Robustness analysis of hottopixx, a linear programming model for factoring nonnegative matrices. *SIAM Journal on Matrix Analysis and Applications*, 34(3):1189–1212, 2013.

[19] Nicolas Gillis. The why and how of nonnegative matrix factorization. *Regularization, Optimization, Kernels, and Support Vector Machines*, 12(257), 2014.

[20] Nicolas Gillis. Introduction to nonnegative matrix factorization. *arXiv preprint arXiv:1703.00663*, 2017.

[21] Nicolas Gillis and Robert Luce. Robust near-separable nonnegative matrix factorization using linear optimization. *Journal of Machine Learning Research*, 15(1):1249–1280, 2014.

[22] Nicolas Gillis and Stephen A Vavasis. Fast and robust recursive algorithms for separable nonnegative matrix factorization. *IEEE transactions on pattern analysis and machine intelligence*, 36(4):698–714, 2014.

[23] Nicolas Gillis and Stephen A Vavasis. On the complexity of robust PCA and l_1-norm low-rank matrix approximation. *arXiv preprint arXiv:1509.09236*, 2015.

[24] Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU Press, 4th edition, 2012.

[25] David Guillamet and Jordi Vitria. Non-negative matrix factorization for face recognition. *CCIA*, 2:336–344, 2002.

[26] Moritz Hardt and Ankur Moitra. Algorithms and hardness for robust subspace recovery. In *Conference on Learning Theory*, pages 354–375, 2013.

[27] Kejun Huang, Nicholas D Sidiropoulos, and Ananthram Swami. Non-negative matrix factorization revisited: Uniqueness and algorithm for symmetric decomposition. *IEEE Transactions on Signal Processing*, 62(1):211–224, 2014.

[28] Hamid Javadi and Andrea Montanari. Non-negative matrix factorization via archetypal analysis. *arXiv preprint arXiv:1705.02994*, 2017.

[29] Ian T Jolliffe. Principal component analysis and factor analysis. In *Principal component analysis*, pages 115–128. Springer, 1986.

[30] Hans Laurberg, Mads Græsbøll Christensen, Mark D Plumbley, Lars Kai Hansen, and Søren Holdt Jensen. Theorems on positive data: On the uniqueness of NMF. *Computational intelligence and neuroscience*, 2008, 2008.

[31] Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788, 1999.

[32] Wing-Kin Ma, José M Bioucas-Dias, Tsung-Han Chan, Nicolas Gillis, Paul Gader, Antonio J Plaza, ArulMurugan Ambikapathi, and Chong-Yung Chi. A signal processing perspective on hyperspectral unmixing: Insights from remote sensing. *IEEE Signal Processing Magazine*, 31(1):67–81, 2014.

[33] Ivan Markovsky. *Low rank approximation: algorithms, implementation, applications*. Springer Science & Business Media, 2011.

[34] Prem Melville and Vikas Sindhwani. Recommender systems. In *Encyclopedia of machine learning*, pages 829–838. Springer, 2011.

[35] Ankur Moitra. An almost optimal algorithm for computing nonnegative rank. *SIAM Journal on Computing*, 45(1):156–173, 2016.

[36] Saïd Moussaoui, David Brie, and Jérôme Idier. Non-negative source separation: range of admissible solutions and conditions for the uniqueness of the solution. In *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*, volume 5, pages v–289. IEEE, 2005.

[37] Pentti Paatero and Unto Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126, 1994.

[38] Ben Recht, Christopher Re, Joel Tropp, and Victor Bittorf. Factoring nonnegative matrices with linear programs. In *Advances in Neural Information Processing Systems*, pages 1214–1222, 2012.

[39] Roman Sandler and Michael Lindenbaum. Nonnegative matrix factorization with earth mover's distance metric for image analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1590–1602, 2011.

[40] Reinhard Schachtner, Gerhard Pöppel, and Elmar Wolfgang Lang. Towards unique solutions of non-negative matrix factorization problems by a determinant criterion. *Digital Signal Processing*, 21(4):528–534, 2011.

[41] Farial Shahnaz, Michael W Berry, V Paul Pauca, and Robert J Plemmons. Document clustering using nonnegative matrix factorization. *Information Processing & Management*, 42(2):373–386, 2006.

[42] Fabian J Theis, Kurt Stadlthanner, and Toshihisa Tanaka. First results on uniqueness of sparse non-negative matrix factorization. In *Signal Processing Conference, 2005 13th European*, pages 1–4. IEEE, 2005.

[43] LB Thomas. Rank factorization of nonnegative matrices (A. Berman). *SIAM Review*, 16(3):393, 1974.

[44] Madeleine Udell, Corinne Horn, Reza Zadeh, Stephen Boyd, et al. Generalized low rank models. *Foundations and Trends® in Machine Learning*, 9(1):1–118, 2016.

[45] Stephen A Vavasis. On the complexity of nonnegative matrix factorization. *SIAM Journal on Optimization*, 20(3):1364–1377, 2009.

[46] René Vidal. Subspace clustering. *IEEE Signal Processing Magazine*, 28(2):52–68, 2011.

[47] Fei Wang, Tao Li, Xin Wang, Shenghuo Zhu, and Chris Ding. Community discovery using nonnegative matrix factorization. *Data Mining and Knowledge Discovery*, 22(3):493–521, 2011.

[48] Stephen J Wright and Jorge Nocedal. Numerical optimization. *Springer Science*, 35(67-68):7, 1999.

[49] Yinyu Ye. Interior point algorithms-theory and analysis, 1998.