An Application of the Von Neumann Algorithm to Matrix Completion and other Data Problems

by

Abraham Varghese

(Under the direction of Ming-Jun Lai)

Abstract

The effectiveness and the convergence of the Von Neumann Algorithm are well established for the case of affine spaces and convex sets. We extend the convergence results to the case of more general sets, whose special case is the convex set. We prove the convergence of the algorithm when it is applied to data problems that include matrix completion, sparse vector recovery, and corrupted audio/Image recovery problems. We present the numerical evidence for the excellent performance of the algorithm in those settings. We also derive bounds (and exact formula in special cases) for the sparsity of a sparsest solution of Sparse Vector Recovery Problem. Such bounds have been unknown till now.

Index words: Von-Neumann Algorithm, Alternate Projection Algorithm, Matrix Completion, Sparse Vector Recovery, Sparsity Bounds

An Application of the Von Neumann Algorithm to Matrix Completion and other Data Problems

by

Abraham Varghese

B.Tech., The National Institute of Technology, Calicut, 2010

A Dissertation Submitted to the Graduate Faculty
of The University of Georgia in Partial Fulfillment
of the
Requirements for the Degree

Doctor of Philosophy

Athens, Georgia

2018

An Application of the Von Neumann Algorithm to Matrix Completion and other Data Problems

by

Abraham Varghese

Approved:

Major Professor: Ming-Jun Lai

Committee: Alexander Petukhov
Juan B. Gutierrez
Qing Zhang
Edward Azoff

Electronic Version Approved:

Suzanne Barbour
Dean of the Graduate School
The University of Georgia
May 2018

Acknowledgments

I would like to sincerely thank my adviser, Ming-Jun Lai, for taking me on as a student and for all of the guidance that he has provided over the past three years. I cannot imagine a more ideal mentor for someone with my particular strengths and weaknesses. He has guided me through every step of the process.

Big thank you to the members of my committee. Thank you Juan Gutierrez for giving me an opportunity to research in your lab. Special thanks to Edward Azoff for being my teaching mentor! Thank you for your genuine care and support, and knowing that your doors were always open was welcoming and reassuring to me.

I am so grateful to many wonderful teachers and mentors I have had over the years, especially Daniel Krashen, Dino Lorenzini and Lisa Townsley. I would like to thank my teachers before graduate school, especially Shanmuga Sundaram, Sanjay P.K., G. Abhilash and Shreedhar Inamdar for relentless nurturing of my love for math and science.

Papa and Mummy, I cannot thank you enough for your toil and sacrifices that have made me who I am. I could not have done it without you. I am deeply thankful to my wife and sweetheart Sarai for standing with me in this long journey and motivating me to never give up, even in the hardest times. My little Oliver, you have been a bundle of joy in these last days leading to graduation!

Above all, I would like to thank the Almighty and my saviour Jesus for His unending love and strength in finishing this thesis and far beyond.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The whole nature around us seems to be in an impassionate pursuit of two kinds. At times, it is a pursuit to maximize or minimize that which shall bring forth the highest plunder, whether it manifests as the least action principle of particle mechanics, Fermat's principle in the world of light, the maximization of utility (reward) or the minimization of surprise in economics, the maximum entropy principle of thermodynamics, Landauer's principle in information theory or as the pursuit of humanity for joy of the highest order. At other times, it is a pursuit to find the common ground between two worlds or two sets in general.[1]

A modest attempt to model these phenomena mathematically would lead us to classify the two kinds of pursuits into the following two problems:

1. Compute

$$\min_{\mathbf{x}\in A} f(\mathbf{x}), \tag{1.0.1}$$

where A is a closed set in $\mathbb{R}^n$ and f is a function on $\mathbb{R}^n$. Most often, we are more interested in the minimizer $\arg\min_{\mathbf{x}\in A} f(\mathbf{x})$ as compared to the minimum

[1]Readers who wish to skip the lengthy introduction can go to page 5 for a quick summary of the new results in this thesis.

value $\min_{\mathbf{x}\in A} f(\mathbf{x})$.

There is a vast literature on solving this centuries old problem. For example, when f is a Lipschitz differentiable convex function and A is a convex set, this minimization can be solved by using the projected gradient method. See, e.g. [5]. When f is a convex function, but not Lipschitz differentiable, we can use the proximal projected gradient descent method.

2. Given two sets A and B in $\mathbb{R}^n$, compute the set intersection

$$\text{Compute } A \cap B \text{ if } A \cap B \text{ is non-empty} \tag{1.0.2}$$

Furthermore, given a point $\mathbf{x} \in \mathbb{R}^n$, compute $P_{A\cap B}(\mathbf{x})$, where $P_{A\cap B}$ denotes ℓ_2-projection of $\mathbf{x}$ to the set $A \cap B$.

It is interesting to note that the Problem 1.0.1 above can be also cast as a special case of Problem 1.0.2. In fact, if we let $B = \{\mathbf{x}, \mathbf{x} = \arg\min_{\mathbf{y}\in\mathbb{R}^n} f(\mathbf{y})\}$ be the collection of minimizers, the first problem turns out to be a set intersection problem. Alternatively, we can also make an educated guess of the minimum value, say ϵ, and set B to be the level set $B = \{\mathbf{x}, f(\mathbf{x}) = \epsilon\}$. In fact, we would take that approach for tackling the matrix completion problem and the compressive sensing problem mentioned in the examples below.

The following is a list of some motivative examples, some of which we shall deal with in detail in the subsequent chapters.

Example 1.0.1 (Matrix Completion Problem). Consider $f(X) = \text{rank}(X)$, where $X = [x_{ij}]_{1\le i,j\le n}$ is a matrix of size $n \times n$. It is clear that $f(X)$ for $X \in \mathbb{R}^{n\times n}$ is not a continuous function. Let $\Omega \subseteq \{1, \cdots, n\} \times \{1, \cdots, n\}$ be a subset of the indices $\{1, \cdots, n\} \times \{1, \cdots, n\}$ and A be the collection of all matrices of $n \times n$ whose entries in Ω are the same as the given entries $M|_\Omega$, i.e. $A = \{X, X|_\Omega = M|_\Omega\}$ where $M|_\Omega$ is the matrix obtained from M by setting its entries in positions Ω^c to zero and the

entries in positions Ω are kept intact. Then the minimization in (1.0.1) will be the well-known Netflix problem.

Example 1.0.2 (Compressive Sensing Problem). Consider $f(\mathbf{x}) = \|\mathbf{x}\|_0$ and $A = \{\mathbf{x} \in \mathbb{R}^n, \|\Phi\mathbf{x} - \mathbf{b}\|^2 \leq \epsilon\}$, where Φ is a matrix of size $m \times n$ with $m << n$ and $\mathbf{b}$ is a given observation vector of size $m \times 1$. Then the minimization (1.0.1) will be the standard compressive sensing problem. See, e.g. [17]. In this case, f is not a continuous function while A is a convex set.

Example 1.0.3 (Nonnegative Compressive Sensing Problem). Consider $f(\mathbf{x}) = \|\Phi\mathbf{x} - \mathbf{b}\|^2$ for given matrix Φ of size $m \times n$ and vector $\mathbf{b}$ of size $m \times 1$ with $m << n$. Let $A = B \cap C$, where C is the collection of k-sparse vectors in $\mathbb{R}^n$ with $k < m$ and $B = \{\mathbf{x} \in \mathbb{R}^n_+, \|\mathbf{x}\|_1 \leq 1\}$. Then the corresponding minimization problem is the one studied in [31].

Example 1.0.4 (Portifolio Selection). Consider $f(\mathbf{x}) = \mathbf{x}^\top C\mathbf{x} - \lambda\nu^\top\mathbf{x}$ to be the combination of the expected return and the risk, where $\mathbf{x} \in \mathbb{R}^n$, $\lambda > 0$ is a parameter, the matrix $C \geq 0$ is a covariance matrix, and ν is the expected return vector of stocks $Y_i, i = 1, \cdots, n$. Let $B = \{\mathbf{x} \geq 0, \|\mathbf{x}\|_1 = 1\}$, a convex set and C be the collection of k-sparse vectors in $\mathbb{R}^n$. Then the minimization is an extension of the so-called Markowitz model proposed in 1952. The sparsity constraint arises because a stock trader can only pay attention to a small number k of different stocks.

Example 1.0.5 (Signal and Image Compression). Suppose that we are given a piece of music, a vector $\mathbf{y}$ of length n. We would like to compress it by subsampling of $\mathbf{y}$ to have $\mathbf{z}$ of size $m << n$. One is to recover $\mathbf{y}$ from $\mathbf{z}$. Assume that the music has a bandwidth $\leq \sigma$. Consider $f(\mathbf{x}) = \|\mathbf{x} - \mathbf{y}\|^2$. Let B be the collection of all vectors whose FFT is a σ-sparse vector. C is the collection of all vectors whose subsamples are $\mathbf{z}$. That is, $C = \{\mathbf{x} \in \mathbb{R}^n, \mathbf{x}|_\Omega = \mathbf{y}|_\Omega\}$. Then minimization provides a way to recover the signal from the intersection $A = B \cap C$. Similarly, we can formulate the image compression problem as the minimization Problem (1.0.1) using 2D FFT and

band-limited images.

Later in this dissertion, we will study various special cases of problem 1.0.2 using the alternating projection method, also popularly known as the Von Neumann Algorithm. We will prove convergence of the algorithm for different types of sets A and B.

Algorithm 1: Alternating Projection Algorithm (APA)

Let $\mathbf{y}_0 \in \mathbb{R}^n$ be an initial guess. For $i = 1, 2, \cdots$, we first project $\mathbf{y}_{i-1}$ to the set B, i.e.

$$\mathbf{x}_i = \mathcal{P}_B(\mathbf{y}_{i-1})$$

and then project $\mathbf{x}_i$ to the set A, i.e.

$$\mathbf{y}_i = \mathcal{P}_A(\mathbf{x}_i).$$

until a maximum number of iterations is achieved or $\|\mathbf{x}_i - \mathbf{y}_i\|$ is less than or equal to a given tolerance.

This algorithm has been rediscovered many times over the last many decades and has been used for the SIP (Set Intersection Problem) for a long time. One earliest possible reference is [66]. See also [4]. Certainly it can be easily extended to find the intersection of multiple subsets in $\mathbb{R}^n$ or of any Hilbert space. We use this algorithm to solve the minimization problem (1.0.1) by converting (1.0.1) into the Problem 1.0.2. We shall establish the convergence of 1 in several cases for closed sets A and B (not necessarily convex) and apply it to a few interesting application problems which include sparse solution of under-determined linear systems, matrix completion, and audio/image signal compression.

It is well known that when the sets A and B are affine linear spaces with closed sum, Algorithm 1 converges linearly independent of starting points (cf. [3] [66]). In the case when A and B are closed convex subsets in a finite dimensional Hilbert space, the algorithm converges and it converges to a point in the intersection if its non-empty. In an infinite dimensional Hilbert space, even when both A and B are

convex, the alternating projection method may not converge in norm. See [38] for a counterexample. There are some standard analyses in the literature (cf. e.g. [3] and [51]) to establish the convergence under some sufficient conditions. See [51] for ways to accelerate the algorithm.

Before we describe the contents of each chapter, we give a summary of *new* results we obtain in this dissertation:

(1) We prove the convergence, under certain conditions, of the alternating projection (AP) algorithm when applied to the well known matrix completion problem. We will also demonstrate the effectiveness of the algorithm by presenting numerical evidence.

(2) We first prove the local convergence of the AP algorithm when applied to sparse recovery problem. Then we introduce new quantities analogous to restricted isometry constant which when bounded ensure global convergence of the algorithm.

(3) We obtain bounds for the sparsity of a sparsest solution to classical sparse vector recovery problem. We are not aware of any non-trivial lower bounds in the literature prior to this dissertation.

(4) Convergence of the AP algorithm is well-studied in the case of convex sets but not so in the case of non-convex sets. We will introduce a special kind of non-convex sets, whose special case are convex sets, and prove the convergence of the algorithm in that regimen.

(5) We will devise a new scheme to reconstruct corrupted audio and image signals. We will also prove theoretical guarantees for success of the scheme.

We have arranged the chapters in the increasing order of complexity of sets A and B. More specifically,

(1) In Chapter 2, both A and B are affine subspaces of $\mathbb{R}^n$. In this chapter, we will present a *new* recovery scheme for corrupt audio and image signals in section 2.2. We will present theoretical guarantees and numerical evidence for the effectiveness of the signal recovery scheme we propose.

(2) In Chapter 3, both A and B contain certain subsets, called *faces*, which are epigraphs of a special class of functions that are not necessarily convex. Recall that the epigraph of a convex function is a convex set. Hence, this is a more general setting as compared to when both A and B are convex sets which is well studied in literature. We will prove the convergence of the algorithm for this case in section 3.2 and section 3.3. Also, we will show that most convex sets are a special case of the sets we consider in this chapter.

(3) In Chapter 4, A is an affine linear space and B is the collection of rank r matrices. Specifically, $A = \{X \in \mathbb{R}^{n\times n}, X|_\Omega = M_\Omega\}$, where Ω is a subset of indices $\{(i,j), 1 \leq i,j \leq n\}$ and $M|_\Omega$ is a given set of known entries of a $n \times n$ matrix M. This is popularly known as the Netflix problem. We will prove in this chapter the conditions for convergence of the algorithm and some related problems. We also will demonstrate the better performance of the algorithm as compared to state-of-the-art by presenting computational evidence.

(4) In Chapter 5, A is the affine linear space defined by $\{\mathbf{x} \mid \ \Phi\mathbf{x} = \mathbf{b}\}$ and B is the collection of bounded k-sparse vectors in $\mathbb{R}^n$. The classical compressive sensing problem given below comes under this category.

$$\min\{\|\mathbf{x}\|_0 : \quad \Phi\mathbf{x} = \mathbf{b}\}, \tag{1.0.3}$$

we need to minimize $f(\mathbf{x}) = \|\mathbf{x}\|_0$ and we can consider the minimizers of f are in the collection of all bounded k-sparse vectors. We shall first prove the local convergence of the algorithm for this problem followed by global convergence. We introduce a new quantity $\eta_s(A)$ associated with any matrix A; s is the sparsity. Conditioned on boundedness of η_{4s}, we will prove global convergence of the AP algorithm. We also obtain *novel* bounds on the sparsity of a sparsest solution.

Chapter 2

Alternating Projection Algorithm for Affine Linear spaces

In this chapter we will apply the theory of alternate projections on affine linear spaces to design a new scheme to compress or recover audio signals whose samples in random positions have been corrupted. We will present theoretical guarantees for the success of the scheme. We will conclude the chapter with numerical evidence for the effectiveness of the scheme.

Before we state the new scheme, for completeness sake, we will first present the proof of well known and classical result that the alternating projection algorithm, when applied to affine linear subspaces of a real or complex finite dimensional Euclidean space, converges linearly.

2.1 Convergence Analysis of APA for two affine Linear Spaces

Before we state the main theorem, we would like to define what 'angle' between two linear space is.

Definition 2.1.1. The *angle* between two affine linear spaces $\mathbf{a}+K$ and $\mathbf{b}+L$, where K and L are vector spaces, is defined to be the angle $\gamma := \gamma(K, L)$ between 0 and $\frac{\pi}{2}$ whose cosine is given by

$$\cos(\gamma) := \sup\left\{ \frac{\langle k, l\rangle}{\|k\| \, \|l\|} \ : \ k \in K \cap (K \cap L)^{\perp}, l \in L \cap (K \cap L)^{\perp} \right\}.$$

Lemma 2.1.2. *For any two distinct subspaces K and L of a finite-dimensional subspace of a Euclidean space,*

$$\cos(\gamma(K, L)) < 1$$

Proof. Using Cauchy-Schwartz inequality, it is clear that $\cos(\gamma(K, L)) \leq 1$. Now, assume, on the contrary, that $\cos(\gamma(K, L)) = 1$. It then implies that

$$\sup\left\{ \frac{\langle k, l\rangle}{\|k\| \, \|l\|} \ : \ k \in K \cap (K \cap L)^{\perp}, l \in L \cap (K \cap L)^{\perp} \right\} = 1 \tag{2.1.1}$$

Since $K \cap (K \cap L)^{\perp}$ and $L \cap (K \cap L)^{\perp}$ are subspaces of a finite dimensional Euclidean space, they are closed. Hence, the supremum in equation 2.1.1 is attained which implies that there exist $k \in K \cap (K \cap L)^{\perp}$ and $l \in L \cap (K \cap L)^{\perp}$ such that $\frac{\langle k, l\rangle}{\|k\|\|l\|} = 1$. It follows that $k, l \neq 0$ and $\langle k, l\rangle^2 = \|k\|^2 \, \|l\|^2$, which is the equality in a Cauchy-Schwartz inequality. Therefore, k and l are linearly dependent. Hence, we deduce that $k, l \in (K \cap L) \cap (K \cap L)^{\perp} = \{0\}$. So, $k = l = 0$ contradicting the fact that $k, l \neq 0$. □

We now state and prove one of well known classical result.

Theorem 2.1.3 (Aronszajn, 1950[1])**.** *Let A and B be distinct closed affine subspaces of a finite dimensional normed linear space, for example $\mathbb{R}^N$, say $A = \mathbf{a}+K$, $B = \mathbf{b}+L$ for vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^N$ and closed subspaces K, L. Then the Alternating Projection algorithm 1 converges linearly with rate $\cos(\gamma(K, L))$, independent of the starting point. More specifically, if $\mathbf{x}$ is the starting point, then the algorithm converges to*

$\mathcal{P}_{K\cap L}(\mathbf{x})$ *at a linear rate.*

Proof. We will do the proof for the case when the $A = K$ and $B = L$ are vector spaces and the general case follows from this case because the normed distances involved in the proof remains invariant under a translation. So, we may assume $\mathbf{a} = 0 = \mathbf{b}$. We will denote by $\mathcal{P}_K$ and $\mathcal{P}_L$ the projections onto spaces K and L respectively.

For notational convenience, set $M := K \cap L$. We begin by decomposing the vector $\mathbf{x}$ as follows

$$\mathbf{x} = \mathbf{x}_M + \mathbf{x}_{M^\perp}$$

where $\mathbf{x}_M \in M$ and $\mathbf{x}_{M^\perp} \in M^\perp$.

So,

$$\mathcal{P}_L\mathcal{P}_K(\mathbf{x}) = \mathcal{P}_L\mathcal{P}_K(\mathbf{x}_M) + \mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp}) = \mathbf{x}_M + \mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp})$$

Hence, by repeated application of the operator $\mathcal{P}_L\mathcal{P}_K$, we obtain for $n = 1, 2, \cdots$

$$(\mathcal{P}_L\mathcal{P}_K)^n(\mathbf{x}) = \mathbf{x}_M + (\mathcal{P}_L\mathcal{P}_K)^n(\mathbf{x}_{M^\perp}) \tag{2.1.2}$$

We proceed by noting that

$$\begin{aligned}
\|\mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp})\|^2 &= \langle \mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp}), \mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp})\rangle \\
&= \langle \mathcal{P}_K(\mathbf{x}_{M^\perp}) - \mathcal{P}_{L^\perp}\mathcal{P}_K(\mathbf{x}_{M^\perp}), \mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp})\rangle \\
&= \langle \mathcal{P}_K(\mathbf{x}_{M^\perp}), \mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp})\rangle \\
&\quad \text{using claim 1 and 2 given below} \\
&\leq \cos(\gamma(K, L)) \, \|\mathcal{P}_K(\mathbf{x}_{M^\perp})\| \, \|\mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp})\| \\
&\leq \cos(\gamma(K, L)) \, \|\mathbf{x}_{M^\perp}\| \, \|\mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp}\|
\end{aligned}$$

After cancelling $\|\mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp})\|$ from both sides of the above inequality, we obtain

$$\|\mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp})\| \leq \cos(\gamma(K,L))\,\|\mathbf{x}_{M^\perp}\|$$

By simple induction it follows that for $n = 1, 2, \cdots$

$$\|(\mathcal{P}_L\mathcal{P}_K)^n\,(\mathbf{x}_{M^\perp})\| \leq \cos(\gamma(K,L))^n\,\|\mathbf{x}_{M^\perp}\|$$

Using the fact that $\cos(\gamma(K,L)) < 1$ from lemma 2.1.2, it follows that

$$\|(\mathcal{P}_L\mathcal{P}_K)^n\,(\mathbf{x}_{M^\perp})\| \to 0$$

. Hence, from equation 2.1.2, we get $(\mathcal{P}_L\mathcal{P}_K)^n\,(\mathbf{x}) \to \mathbf{x}_M = \mathcal{P}_{K\cap L}(\mathbf{x})$. This completes the proof.

We used the following claims in above argument:

Claim 1: $\mathcal{P}_K(\mathbf{x}_{M^\perp}) \in K \cap M^\perp$

Proof:

Let $m \in M = K \cap L$ be an arbitrary element of $M = K \cap L$. Then

$$\begin{aligned}\langle \mathcal{P}_K(\mathbf{x}_{M^\perp}), m\rangle &= \langle \mathbf{x}_{M^\perp} - \mathcal{P}_{K^\perp}(\mathbf{x}_{M^\perp}), m\rangle \\ &= \langle \mathbf{x}_{M^\perp}, m\rangle - \langle \mathcal{P}_{K^\perp}(\mathbf{x}_{M^\perp}), m\rangle \\ &= 0 - 0 = 0\end{aligned}$$

The first summand in the second last step is zero because $\mathbf{x}_{M^\perp} \in M^\perp$ and $m \in M$. The second summand is zero because $\mathcal{P}_{K^\perp}(\mathbf{x}_{M^\perp}) \in K^\perp$ and $m \in K$.

Claim 2: $\mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp}) \in L \cap M^\perp$

Proof:

It is clear that $\mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp}) \in L$. To prove $\mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp}) \in M^\perp$, we proceed as follows: As above, let $m \in M = K \cap L$ be an arbitrary element of $M = K \cap L$. Then,

$$\begin{aligned}
\langle \mathcal{P}_L\mathcal{P}_K(\mathbf{x}_{M^\perp}), m\rangle &= \langle \mathcal{P}_K(\mathbf{x}_{M^\perp}) - \mathcal{P}_{L^\perp}\mathcal{P}_K(\mathbf{x}_{M^\perp}), m\rangle \\
&= \langle \mathcal{P}_K(\mathbf{x}_{M^\perp}), m\rangle - \langle \mathcal{P}_{L^\perp}\mathcal{P}_K(\mathbf{x}_{M^\perp}), m\rangle \\
&= \langle \mathcal{P}_K(\mathbf{x}_{M^\perp}), m\rangle - 0 \\
&= \langle \mathbf{x}_{M^\perp} - \mathcal{P}_{K^\perp}(\mathbf{x}_{M^\perp}), m\rangle \\
&= \langle \mathbf{x}_{M^\perp}, m\rangle - \langle \mathcal{P}_{K^\perp}(\mathbf{x}_{M^\perp}), m\rangle \\
&= 0 - 0 = 0
\end{aligned}$$

The term in third step is zero because $\mathcal{P}_{L^\perp}\mathcal{P}_K(\mathbf{x}_{M^\perp}) \in L^\perp$ and $m \in L$ □

2.2 Corrupted Audio Signal Reconstruction

In this section, we will apply the alternating projection technique to compress audio signals and images as well to recover them from corrupted samples. We will first prove the theoretical guarantee and motivation behind the compression/reconstruction method which is described in subsection 2.2.1.

For convenience, we begin with a single-channel audio signal which we will consider as a vector in a N-dimensional real Euclidean space. Let $\mathbf{x} = \{x_n\}_{n=0}^{N-1} \in \mathbb{R}^N$ be the given original audio signal which is a band-limited signal. We would like to compress it by choosing a subsample $\mathbf{z} = \mathbf{x}|_\Omega$ from $\mathbf{x}$ which will be enough to recover $\mathbf{x}$ by using Algorithm 1, where Ω is a subset of $\{0, 1, \cdots, N-1\}$. In this chapter, we will use $\hat{\mathbf{x}}$ to denote the discrete Fourier transform of $\mathbf{x}$ given by

$$\hat{\mathbf{x}}_j := \mathcal{F}(\mathbf{x})(j) = \sum_{k=0}^{N-1} x_k \exp\left(\frac{-2\pi jk}{N}\right), j = 0, \cdots, N.$$

Let σ be an integer such that $0 < \sigma < \frac{N}{2}$ and define N_σ to be the set of all $\sigma-$ band-limited real signals, i.e.

$$N_\sigma := \left\{ \mathbf{x} \in \mathbb{R}^N \mid \mathcal{F}(\mathbf{x})(j) = 0 \text{ for } \sigma < j < N - \sigma \right\}.$$

Fix $\Omega \subset \{0, 1, \cdots, N-1\}$. We can state our goal as follows: given only samples $x_n, n \in \Omega$ of a $\sigma-$band-limited audio signal $\mathbf{x}$, we would like to uniquely recover all the entries of the original $\mathbf{x} = \{x_n, n = 0, 1, \cdots, N-1\}$. We shall use 1 to do so. Indeed, set $A = N_\sigma$ and $B = \left\{ \mathbf{y} \in \mathbb{R}^N \mid y_i = x_i \text{ for } i \in \Omega \right\}$. Note that A is a linear space of dimension $2\sigma + 1$ while B is affine space of dimension $N - |\Omega|$, see lemma 2.2.1. Now, we will apply algorithm 1 between the sets A and B. We will prove (see theorem 2.2.4) that the algorithm will uniquely recover the audio signal if N satisfies certain conditions.

Let us start with the following lemmas.

Lemma 2.2.1. *$A = N_\sigma$ is a linear subspace of $\mathbb{R}^N$ which has real dimension $2\sigma + 1$ and $B = \left\{ \mathbf{y} \in \mathbb{R}^N \mid y_i = x_i \text{ for } i \in \Omega \right\}$ is an affine space of real dimension $N - |\Omega|$.*

Proof. The first fact that B is an affine space of dimension $N - |\Omega|$ is trivial. N_σ is a linear subspace because discrete Fourier transform is a linear operator. Using the formula for Inverse Fourier transform,

$$\begin{aligned} x_n &= \sum_{k=0}^{n-1} \hat{\mathbf{x}}_k \exp(\frac{2\pi k n}{N}) \\ &= \hat{\mathbf{x}}_0 + \sum_{k=1}^{\sigma} 2\Re(\hat{\mathbf{x}}_k \exp(\frac{2\pi k n}{N})) \text{ [since } \hat{\mathbf{x}}_k = 0 \text{ when } \sigma < k < n - \sigma] \\ &= \hat{\mathbf{x}}_0 + \sum_{k=1}^{\sigma} 2\Re(\hat{\mathbf{x}}_k) \cos(\frac{2\pi k n}{N}) - 2\Im(\hat{\mathbf{x}}_k) \sin(\frac{2\pi k n}{N}). \end{aligned}$$

Now the result follows by noting that $\hat{\mathbf{x}}_0, \Re(\hat{\mathbf{x}}_k)$ and $\Im(\hat{\mathbf{x}}_k)$, $k = 1, 2, \cdots, \sigma$ are independent real parameters. Here $\Re(\hat{\mathbf{x}}_k)$ and $\Im(\hat{\mathbf{x}}_k)$ denote the real and imaginary parts

of $\hat{\mathbf{x}}_k$ respectively. □

Lemma 2.2.2 ([61]). *Let $N = p$ be a prime number. As above, let $\hat{\mathbf{x}}$ denote the discrete Fourier transform of $\mathbf{x} \in \mathbb{R}^N$. Then, for any* non-zero $\mathbf{x} \in \mathbb{R}^N$

$$\text{Support}(\hat{\mathbf{x}}) + \text{Support}(\mathbf{x}) \geq p + 1.$$

where $\text{Support}(\mathbf{x})$ *is the number of non-zero components of* $\mathbf{x}$*. Conversely, if I_1 and I_2 are two non-empty subsets of $\{1, 2, \cdots, p\}$ such that $|I_1| + |I_2| \geq p + 1$, then there exists a vector* $\mathbf{x}$ *such that* $\text{Support}(\mathbf{x}) = I_1$ *and* $\text{Support}(\hat{\mathbf{x}}) = I_2$*.*

Proof. Refer to Theorem 1.1 in [61] for a proof. □

Lemma 2.2.3. *Assume $N = p$ is a prime number and $|\Omega| > 2\sigma$; then there is at most one vector $\mathbf{x} \in \mathbb{R}^N$ that is contained in both the sets B and N_σ.*

Proof. Assume $\mathbf{y}, \mathbf{z} \in B \cap N_\sigma$. Then $\mathbf{y} - \mathbf{z} \in N_\sigma$ and $y_i - z_i = 0$ for $i \in \Omega$. Hence $\text{Support}(\mathbf{y} - \mathbf{z}) \leq N - |\Omega|$ and $\text{Support}(\hat{\mathbf{y}} - \hat{\mathbf{z}}) \leq 2\sigma + 1$. Hence, $\text{Support}(\mathbf{y} - \mathbf{z}) + \text{Support}(\hat{\mathbf{y}} - \hat{\mathbf{z}}) \leq N - |\Omega| + 2\sigma + 1 \leq N = p$ which contradicts the statement of Lemma 2.2.2 if $\mathbf{y} - \mathbf{z} \neq 0$. Therefore $\mathbf{y} = \mathbf{z}$. □

We are now ready to establish the following

Theorem 2.2.4. *Assume $N = p$ is a prime number and $|\Omega| > 2\sigma$. Let $\mathbf{x} \in \mathbb{R}^N \cap N_\sigma$ be a σ-band-limited signal with only known samples in Ω positions. Then the Alternating Projection algorithm 1 applied to sets N_σ and B will recover the original signal $\mathbf{x} \in \mathbb{R}^N$ independently of the starting point.*

Proof. By Lemma 2.2.3, $\{\mathbf{x}\} = N_\sigma \cap B$. Since N_σ and B are affine spaces, the result follows from Theorem 2.1.3. □

Now, with above results as a motivation, we will present the scheme to recover an corrupted audio signal.

2.2.1 Corrupted Audio Signal Reconstruction Scheme

Based on the above discussion, we propose the following audio signal reconstruction scheme.

Algorithm 2: Corrupted Audio Signal Recovery Scheme

Data: $\mathbf{x}|_\Omega$ an audio signal whose uncorrupted samples are located in indices Ω, the bandlimit σ (Default value is $\sigma = \lfloor \frac{N}{4} \rfloor$)
Result: X_k, a close approximation of the original uncorrupted audio signal $\mathbf{x}$
Preprocessing: Partition the audio signal $\mathbf{x}$ into blocks of size N where N is a prime number with the same order of magnitude as the length of the signal
Initialize $X_k =$ First block of $\mathbf{x}|_\Omega$
repeat
  repeat
    Step 1: $Y_k = P_{N_\sigma}(X_k)$
    Step 2: $X_{k+1} = P_B(Y_k)$
  until $\|X_{k+1} - X_k\| < \epsilon$;
until *All blocks of* $\mathbf{x}\ |_\Omega$ *are processed*;

Remark 2.2.5. In the above scheme, we assumed the known samples $\mathbf{x}\ |_\Omega$ of the audio signal are clean samples without noise. We can modify the above scheme to make it more robust to noise by replacing the projection operator P_B with the projection operator P_{B_ϵ} where $B_\epsilon = \left\{\mathbf{y} \in \mathbb{R}^N \ \mid \ |y_i - x_i| \leq \epsilon_i \text{ for } i \in \Omega\right\}$. Explicitly, if

$$P_{B_\epsilon}(\mathbf{a}) = \mathbf{b}$$

then, for each i,

$$b_i = \begin{cases} a_i & \text{if } a_i \in [x_i - \epsilon_i, x_i + \epsilon_i] \\ x_i - \epsilon_i & \text{if } a_i < x_i - \epsilon_i \\ x_i + \epsilon_i & \text{if } a_i > x_i + \epsilon_i \end{cases}$$

. The $\epsilon = (\epsilon_1, \cdots, \epsilon_N)$ can be initialized according to the noise model we chose for the system. For example, the components ϵ_i of ϵ could be chosen from a Gaussian distribution.

2.2.2 Numerical Results

In this section, we present numerical evidence for the effectiveness of the audio recovery scheme we discussed in the previous section. We apply the scheme to fragments of three piece of music. *Mozart concerto for flute and harp, K299*, the *OST(Original Sound Track)* of the movie Lord of the Rings, and *The mission and How great thou art* by Piano guys serve as the sources of our audio signal fragments.

The implementation is in Matlab and all the computational results were obtained on a laptop computer with a 2.50 GHz CPU (4 cores with Matlabs multithreading option enabled) and 16 GB of memory. In our simulations, the set of observed entries Ω is sampled uniformly at random among all sets of cardinality $|\Omega|$.

We observed comparable levels of audio quality for the recovered signal as compared to the original signal. For example, *Mozart concerto for flute and harp, K299* sampled at 50% achieved a -0.03db ($= 20 \log\left(\frac{\|\mathbf{X}_k\|}{\|\mathbf{x}\|}\right)$) recovery. The difference in audio quality of the recovered signal and the original signal was virtually undetectable to human ears if the percentage of known samples was higher than 60%.

The results of the experiments are shown in tables 2.1, 2.2 and 2.3. In the tables, we have used the following abbreviations:

- RMS error is the Root Mean Square Error given by $\|X_k - \mathbf{x}\|_F / \sqrt{N}$
- Relative Error is the ratio of Root Mean Square error and Root Mean Square Amplitude of the audio signal given by $\|X_k - \mathbf{x}\|_F / \|\mathbf{x}\|_F$

2.3 Corrupted Image Reconstruction Scheme

In this section, we will apply the analogue of the above scheme of corrupted audio signal reconstruction to recover corrupted images. Assume that the original image is a matrix $\mathbf{X} \in \mathbb{R}^{N \times N}$. We denote the 2-dimensional discrete Fourier transform (DFT)

Table 2.1: Recovery Rates for Mozart Concerto for flute and harp, K.299

Known Samples (%)	RMS Error	Relative Error	Time (seconds)
90	9.0e-06	5.13e-04	260
80	1.12e-05	6.38e-04	269
70	6.8e-05	3.87e-03	259
60	3.01e-04	1.70e-02	260
50	9.64e-04	5.49e-02	263
40	2.59e-03	1.47e-01	252
30	5.62e-03	3.204e-01	252
20	9.722e-03	5.54e-01	263
10	1.39e-02	7.93e-01	257

Table 2.2: Recovery Rates for Breaking of the fellowship, Lord of the Rings

Known Samples (%)	RMS Error	Relative Error	Time (seconds)
90	2.42e-05	1.911e-04	123
80	6.28e-05	4.96e-04	210
70	4.99e-04	3.94e-03	222
60	2.26e-03	1.78e-02	218
50	7.28e-03	5.75e-02	207
40	1.91e-02	1.510e-01	206
30	4.08e-02	3.226e-01	207
20	7.05e-02	5.575e-01	214

Table 2.3: Recovery Rates for The Mission and How great thou art, Piano Guys

Known Samples (%)	RMS Error	Relative Error	Time (seconds)
90	1.082e-04	4.76e-04	261
80	2.406e-04	1.05e-03	262
70	7.166e-04	3.15e-03	262
60	3.82e-03	1.68e-02	249
50	1.28e-03	5.64e-02	246
40	3.35e-02	1.47e-01	278
30	7.22e-02	3.17e-01	275
20	1.26e-01	5.55e-01	277
10	1.80e-01	7.92e-01	241

of a matrix $\mathbf{Y}$ by $\hat{\mathbf{Y}}$. Then the sets A and B in this setting would be given by

$$A := \left\{ \mathbf{Y} \in \mathbb{R}^{N \times N} \mid \hat{\mathbf{Y}}_{ij} = 0 \text{ if } \sigma < i < N - \sigma \text{ or } \sigma < j < N - \sigma \right\}$$

and

$$B := \left\{ \mathbf{Y} \in \mathbb{R}^{N \times N} \mid Y_{ij} = X_{ij} \text{ for } (i,j) \in \Omega \right\}$$

We will apply the APA algorithm to the above sets to recover the original image from the known samples $\mathbf{X}_\Omega$. Since all images used in the experiments have size 512×512 which is small in comparison with that of an audio signal, images were not partitioned into blocks unlike in the case of audio signal recovery scheme we discussed in the previous section.

The experiments are conducted in Matlab and all the computational results were obtained on a laptop computer with a 2.50 GHz CPU (4 cores with Matlabs multithreading option enabled) and 16 GB of memory. In our simulations, the set of observed entries Ω is sampled uniformly at random among all sets of cardinality $|\Omega|$.

The results of the experiments are shown in figures 2.1, 2.2 and 2.3. Our results show that recovery was not spectacular. Since the support of image's Fourier spectrum is large in comparison with its size, it results in the dimension of spaces A and B being larger, hence their intersection might not be unique. We could circumvent this issue by truncating the DFT to a smaller size by setting larger frequency components to zero. But this would result in the degradation of the visual quality.

Remark 2.3.1. Since most real world images can be considered as approximately low ranked matrices, we will consider an alternate way to recover missing samples of an image using the matrix completion technique in Chapter 4. There we will replace the set A with a set of low rank matrices and apply APA.

Figure 2.1: Top row: The original image; Rest of the rows: left column correspond to image with missing entries and right column images are reconstructed using APA described in section 2.3 from the corresponding images on left

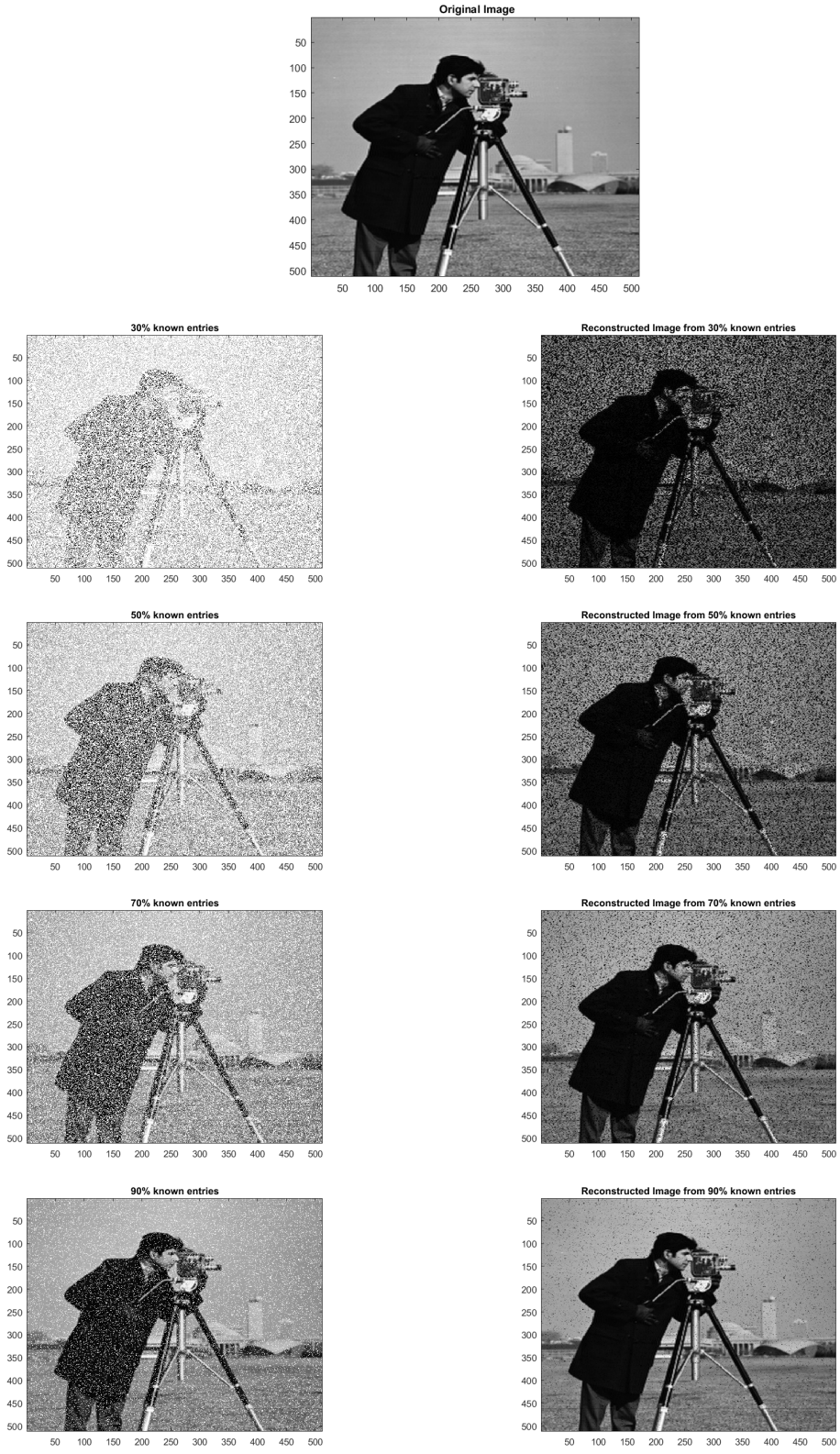

Figure 2.2: Top row: The original image; Rest of the rows: left column correspond to image with entries missing and right column images are reconstructed using APA described in section 2.3 from the corresponding images on left

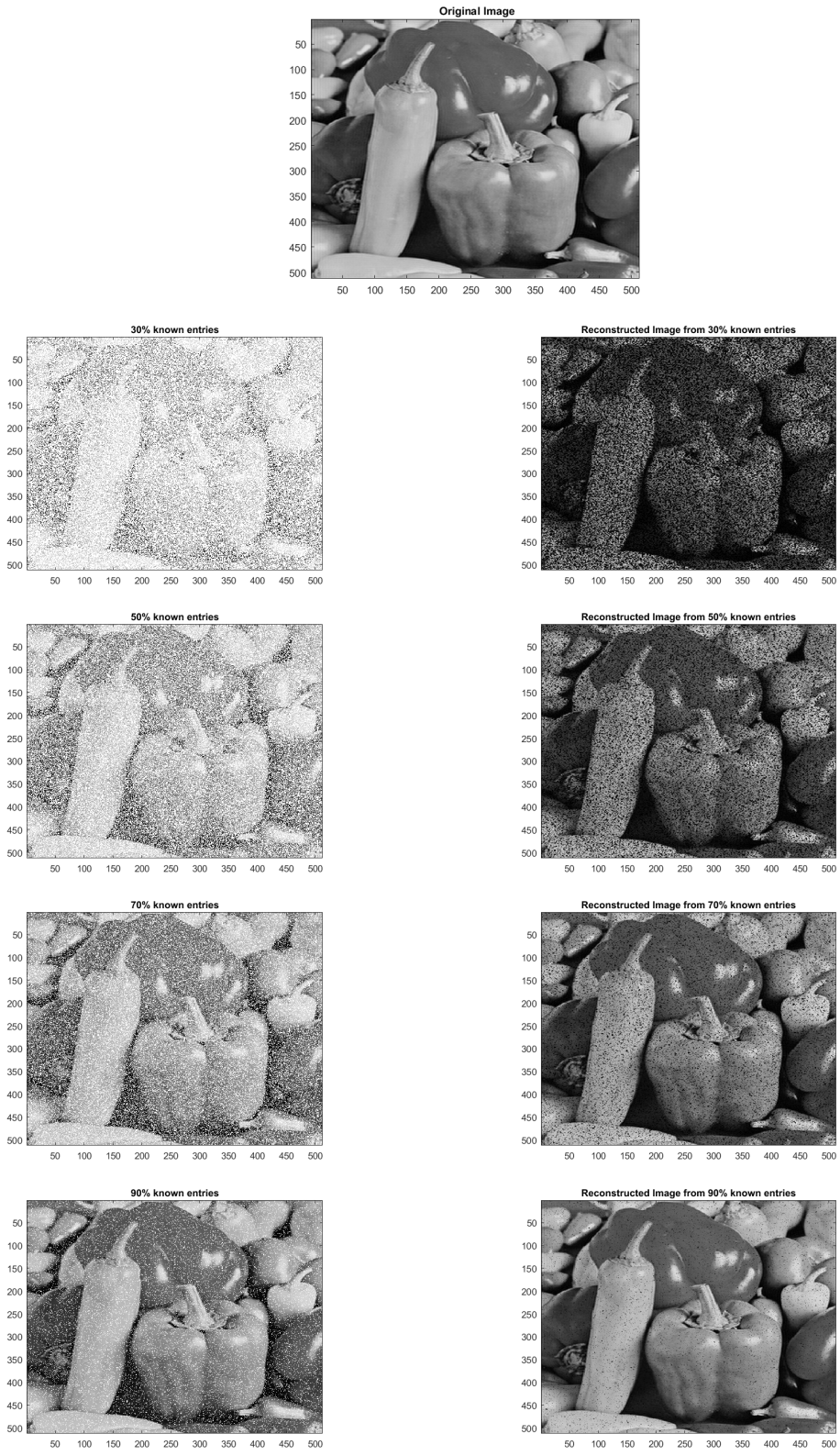

Figure 2.3: Top row: The original image; Rest of the rows: left column correspond to image with entries missing and right column images are reconstructed using APA described in section 2.3 from the corresponding images on left

Chapter 3

Alternating Projection Algorithm for Convex and Non-convex Sets

In the second section of this chapter, motivated by results in its first section, we extend the convergence of the AP algorithm to more general sets which are partially epigraphs of a certain class of functions we will define in that section. In the first section, we will first review some of the basics of the theory of alternate projection algorithms on convex sets. Most of the results presented in the first section can be found in [34].

3.1 APA for Convex Sets

Throughout this section, we will assume that the two sets whose intersection points we are seeking are denoted by Q_1 and Q_2. We will assume that Q_1 and Q_2 are closed and convex subsets of the real Euclidean subspace $\mathbb{R}^n$ which have a nonempty intersection $Q := Q_1 \cap Q_2$. Note that all the results in this section readily extends to the case of complex Euclidean space.

Throughout this section, we will use the cyclic notation $\alpha(n) = \begin{cases} 1 & \text{if } n \text{ is even} \\ 2 & \text{if } n \text{ is odd} \end{cases}$ and we will use $\mathcal{P}_{\alpha(n)}$ to denote the projection onto set $Q_{\alpha(n)}$. For example, $\mathcal{P}_{\alpha(1)}$ and $\mathcal{P}_{\alpha(2)}$ denote the projection onto the sets Q_1 and Q_2 respectively.

With the above notation, the alternate projection algorithm can be concisely expressed as

$$x_{n+1} = \mathcal{P}_{\alpha(n)}(x_n)$$

where x_n is the n^{th} iterate and the initial point x_0 is arbitrary.

Before we prove convergence of the alternate projection algorithm for the convex sets, we shall review some basic results about the projection operator, which shall be needed to the prove the convergence result.

3.1.1 Preliminaries

Definition 3.1.1. The projection $\mathcal{P}_{\mathbf{S}}(\mathbf{x})$ of a point $\mathbf{x} \in \mathbb{R}^n$ to a set $\mathbf{S}$ is defined as a point in $\mathbf{S}$ which is closest to the point $\mathbf{x}$. In general, projection is not single-valued. But it is single-valued in the case when set $\mathbf{S}$ is convex. More formally,

$$\mathcal{P}_{\mathbf{S}}(\mathbf{x}) = \arg\min_{\mathbf{s}\in\mathbf{S}} \|\mathbf{s} - \mathbf{x}\|$$

We will review some basic properties of the projection operator to a closed convex set:

Lemma 3.1.2. *If $\mathcal{P}_A(x)$ denotes the projection of a point $x \in \mathbb{R}^n$ onto a closed convex set $A \subset \mathbb{R}^n$. Then*

$$\langle x - \mathcal{P}_A(x), y - \mathcal{P}_A(x)\rangle \leq 0 \text{ for all } y \in A \tag{3.1.1}$$

where $\langle \cdot, \cdot \rangle$ denotes the standard inner product on $\mathbb{R}^n$.

Proof. Since the set A is convex, for $0 \leq \lambda < 1$, we have $\lambda y + (1-\lambda)\mathcal{P}_A(x) \in A$. So, from the definition of projection, it follows that

$$\begin{aligned}
&\|x - \mathcal{P}_A(x)\|^2 \leq \|x - \lambda y - (1-\lambda)\mathcal{P}_A(x)\|^2 \\
&= \|x - \mathcal{P}_A(x)\|^2 + \lambda^2 \|y - \mathcal{P}_A(x)\|^2 - 2\lambda\langle x - \mathcal{P}_A(x), y - \mathcal{P}_A(x)\rangle \text{ for all } \lambda \in (0,1)
\end{aligned}$$

Hence,

$$\lambda \|y - \mathcal{P}_A(x)\|^2 \geq 2\langle x - \mathcal{P}_A(x), y - \mathcal{P}_A(x)\rangle \text{ for all } \lambda \in (0,1)$$

Taking the limit $\lambda \to 0$, we get the required result. □

Next we prove that the projection operator onto a convex set is a non-expansive operator.

Lemma 3.1.3. *The projection operator $\mathcal{P}_A$ to a closed convex set A is non-expansive. In other words,*

$$\|\mathcal{P}_A(x) - \mathcal{P}_A(y)\| \leq \|x - y\|$$

for all $x, y \in A$

Proof. Using lemma 3.1.2 twice, we get

$$\langle x - \mathcal{P}_A(x), \mathcal{P}_A(y) - \mathcal{P}_A(x)\rangle \leq 0$$

and

$$\langle y - \mathcal{P}_A(y), \mathcal{P}_A(x) - \mathcal{P}_A(y)\rangle \leq 0$$

Adding the above inequalities, we obtain

$$\begin{aligned}
\langle \mathcal{P}_A(x) - x + y - \mathcal{P}_A(y), \mathcal{P}_A(x) - \mathcal{P}_A(y)\rangle = \langle \mathcal{P}_A(x) - \mathcal{P}_A(y), \mathcal{P}_A(x) - \mathcal{P}_A(y)\rangle + \\
\langle y - x, \mathcal{P}_A(x) - \mathcal{P}_A(y)\rangle \leq 0
\end{aligned}$$

That is,

$$\|\mathcal{P}_A(x) - \mathcal{P}_A(y)\|^2 + \langle y - x, \mathcal{P}_A(x) - \mathcal{P}_A(y)\rangle \leq 0$$

Rewriting the above equation and applying Cauchy-Schwarz inequality, we obtain

$$\|\mathcal{P}_A(x) - \mathcal{P}_A(y)\|^2 \leq \langle x - y, \mathcal{P}_A(x) - \mathcal{P}_A(y)\rangle \leq \|\mathcal{P}_A(x) - \mathcal{P}_A(y)\| \, \|x - y\|$$

After cancelling the factor $\|\mathcal{P}_A(x) - \mathcal{P}_A(y)\|$ from both sides, the result follows. □

3.1.2 Convergence of APA for Closed Convex Sets

In this section, we will prove the following

Theorem 3.1.4 ([34])**.** *The alternate projection algorithm applied to the closed convex sets Q_1 and Q_2 which has a nonempty intersection Q converges to a point in the intersection Q.*

We will follow the proof in Gubin et al. [34] after necessary notational simplification. Before we state the proof, some lemmas have to be established.

Lemma 3.1.5. *For any $x \in Q = Q_1 \cap Q_2$,*

$$\|x_{n+1} - x\| \leq \|x_n - x\| \tag{3.1.2}$$

Proof. We have

$$\begin{aligned}
\|x_{n+1} - x\|^2 = \left\|\mathcal{P}_{\alpha(n)}(x_n) - x_n\right\|^2 &= \|x_n - x\|^2 + \left\|\mathcal{P}_{\alpha(n)}(x_n) - x_n\right\|^2 \\
&+ 2\langle x_n - x, \mathcal{P}_{\alpha(n)}(x_n) - x_n\rangle = \|x_n - x\|^2 - \left\|\mathcal{P}_{\alpha(n)}(x_n) - x_n\right\|^2 \\
&+ 2\langle \mathcal{P}_{\alpha(n)}(x_n) - x_n, \mathcal{P}_{\alpha(n)}(x_n) - x\rangle \\
&\leq \|x_n - x\|^2 - \left\|\mathcal{P}_{\alpha(n)}(x_n) - x_n\right\|^2 \leq \|x_n - x\|^2
\end{aligned}$$

□

Lemma 3.1.6.

$$\lim_{n\to\infty} \max(d(x_n, Q_1), d(x_n, Q_2)) = 0 \tag{3.1.3}$$

where $d(x, A) = \inf\{\|x - y\| \mid y \in A\}$ *denotes the distance of the point* x *to set* A.

Proof. By the definition of the projection, we have $d(x_n, Q) = \left\|\mathcal{P}_{\alpha(n)}(x_n) - x_n\right\|$. Hence, applying the lemma 3.1.5, we get

$$\|\mathcal{P}_Q(x_n) - x_n\| \geq \|x_{n+1} - \mathcal{P}_Q(x_n)\| \geq \|x_{n+1} - \mathcal{P}_Q(x_{n+1})\|$$

Hence, $\|\mathcal{P}_Q(x_n) - x_n\|$ is a monotonically decreasing sequence bounded below by 0. So it converges to a limit say $\lim_{n\to\infty} \|\mathcal{P}_Q(x_n) - x_n\| =: l$ Also,

$$\begin{aligned}
\|\mathcal{P}_Q(x_n) - x_n\|^2 - \|\mathcal{P}_Q(x_{n+1}) - x_{n+1}\|^2 &\geq \|\mathcal{P}_Q(x_n) - x_n\|^2 - \|\mathcal{P}_Q(x_n) - x_{n+1}\|^2 \\
&= \|\mathcal{P}_Q(x_n) - x_n\|^2 - \left\|\mathcal{P}_{\alpha(n)}(x_n) - \mathcal{P}_Q(x_n)\right\|^2 \\
&= \|\mathcal{P}_Q(x_n) - x_n\|^2 - \|\mathcal{P}_Q(x_n) - x_n\|^2 \\
&\quad - \left\|\mathcal{P}_{\alpha(n)}(x_n) - x_n\right\|^2 - 2\langle x_n - \mathcal{P}_Q(x_n), \mathcal{P}_{\alpha(n)}(x_n) - x_n\rangle \\
= \left\|\mathcal{P}_{\alpha(n)}(x_n) - x_n\right\|^2 + 2\langle \mathcal{P}_Q(x_n) - \mathcal{P}_{\alpha(n)}(x_n), \mathcal{P}_{\alpha(n)}(x_n) - x_n\rangle &\geq \left\|\mathcal{P}_{\alpha(n)}(x_n) - x_n\right\|^2
\end{aligned}$$

where the last inequality in the last step results from applying lemma 3.1.2. Now taking limit $n \to \infty$, we obtain $\left\|x_n - \mathcal{P}_{\alpha(n)}(x_n)\right\| \to 0$

Now, let $\epsilon > 0$. Choose N such that $\left\|x_n - \mathcal{P}_{\alpha(n)}(x_n)\right\| \leq \epsilon/4$ for all $n \geq N$. Then for $i = 1, 2$, we can find $k < 2$ such that $\alpha(n + k) = i$. So, we have for $n \geq N$

$$\begin{aligned}
d(x_n, Q_i) = \left\|x_n - \mathcal{P}_{\alpha(i)}(x_n)\right\| \leq \|x_n - x_{n+k}\| + \left\|x_{n+k} - \mathcal{P}_{\alpha(n+k)(x_{n+k})}\right\| \leq \\
\|x_n - x_{n+1}\| + \cdots + \|x_{n+k-1} - x_{n+k}\| + \left\|x_{n+k} - \mathcal{P}_{\alpha(n+k)}(x_{n+k})\right\| \leq \frac{k\epsilon}{2} + \frac{\epsilon}{4} < \epsilon
\end{aligned}$$

Since ϵ is arbitrary, the result follows. □

Lemma 3.1.7.

$$\lim_{n\to\infty} d(x_n, Q) = \lim_{n\to\infty} \|x_n - \mathcal{P}_Q(x_n)\| = 0 \tag{3.1.4}$$

Proof. Assume, on the contrary, that there exist a subsequence x_{n_k} such that

$$\lim_{n\to\infty} d(x_{n_k}, Q) \geq l > 0$$

. Since the sequence x_{n_k} is bounded, it has a subsequence, which for notational simplicity will also be denoted by x_{n_k}, which converges to some point $x^\star$. Since $d(x_{n_k}, Q_i) \to 0$ for every i, and Q_i is closed, $x^\star \in Q_i$ for $i = 1, 2$. Therefore $x^\star \in Q$ which contradicts the assumption that $d(x_{n_k}, Q) \geq l > 0$ □

3.1.3 Proof of the Main Theorem 3.1.4

Proof. Let $B(x, r)$ denote the closed ball, centered at x, of radius r. Set

$$B_m = \bigcap_{i=0}^{m} B\left(\mathcal{P}_Q(x_i), d(x_i, Q)\right)$$

. Since repeated application of the lemma 3.1.5 gives us $\|x_m - \mathcal{P}_Q(x_n)\| \leq d(x_n, Q)$ for all $n \leq m$, we can conclude that the sets B_m are non-empty. Clearly $B_{m+1} \subset B_m$ for all m. Hence, the sets B_m forms a nested sequence of closed, convex and non-empty sets. Therefore $\bigcap_{i=0}^{\infty} B_i \neq \emptyset$. So let $x^\star \in \bigcap_{i=0}^{\infty} B_i$. So, we have

$$\|x_n - x^\star\| \leq \|x_n - \mathcal{P}_Q(x_n)\| + \|\mathcal{P}_Q(x_n) - x^\star\| \leq 2d(x_n, Q)$$

Since, $d(x_n, Q) \to 0$ using lemma 3.1.7, it follows that $x_n \to x^\star$ □

Remark 3.1.8. Note that unlike in the case of affine linear spaces discussed in the previous chapter, we cannot guarantee that x_n converges to $\mathcal{P}_Q(x_0)$. The following

example in the Figure 3.1 readily illustrates that. In fact, Dykstra [25] invented a more sophisticated algorithm which guarantees that x_n does converge in norm to $\mathcal{P}_Q(x_0)$.

Dykstra's algorithm can described as follows:

Algorithm 3: Dykstra's Alternating Projection Algorithm

Set $q_{-1} = q_0 = 0$. Generate the sequences (x_n) and (q_n) by

$$x_n := \mathcal{P}_{\alpha(n)}(x_{n-1} + q_{n-2})$$

and

$$q_n := x_{n-1} + q_{n-2} - x_n$$

until the maximum number of iterations is achieved or $\|x_n - x_{n-1}\|$ is less than or equal to a given tolerance.

Boyle and Dykstra [9] proved that the sequence x_n always converges to $\mathcal{P}_Q(x_0)$. See also Deutsch and Hundal [21] and Perkins [52] for convergence analysis of Dykstra's algorithm in more general setting.

It is a well known fact that a function is convex if and only if its epigraph is a convex set. We shall include a proof of it in the next section. So, it naturally motivates us to study more general sets, which are not necessarily convex, whose 'face' is a graph of a general class of functions. The study and analysis of APA on such sets will form the theme of the next section.

3.2 APA for General Sets

3.2.1 Preliminaries

In this section, we extend the results of previous section to sets that are not convex. We study the sets that can be described as the epigraphs of a certain class of functions. We will be restricting ourselves to a 'nice' class of functions. We will then show that convex sets are a special case of the sets we consider here. Before we delve into the

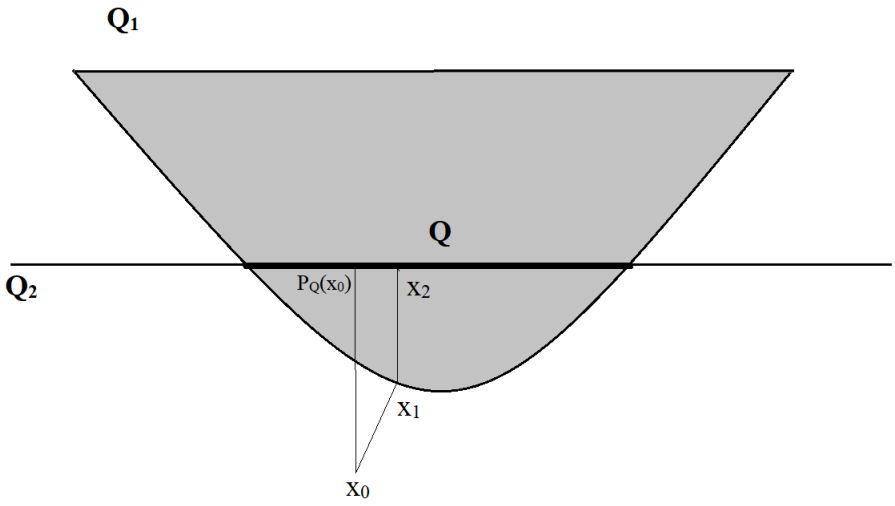

Figure 3.1: Illustration that shows that the iterates x_n of APA algorithm do not necessarily converge to $\mathcal{P}_Q(x_0)$ in the general case of convex sets. The thick shaded lines denotes the intersection Q.

details, some definitions are in order:

Definition 3.2.1. We will define the *Face of set A with respect to set B* denoted by $\text{Face}_B(A)$, as

$$\text{Face}_B(A) := \{a \in A \ \mid \ \exists b \in B \text{ s.t. } a = \arg\min_{x \in A} \|x - b\|\}$$

In the cases, when the projection operator P_A, where P_A denotes the projection onto set A, is single-valued, $\text{Face}_B(A) = P_A(B)$. See figure 3.2 for an example.

Definition 3.2.2. An *epigraph* $\text{epi}(f)$ of a function $f : \mathbb{R}^n \to \mathbb{R}$ is defined by

$$\text{epi}(f) = \left\{(\mathbf{x}, t) \in \mathbb{R}^{n+1} : \quad \mathbf{x} \in \mathbb{R}^n, \mathrm{t} \geq f(\mathbf{x})\right\}$$

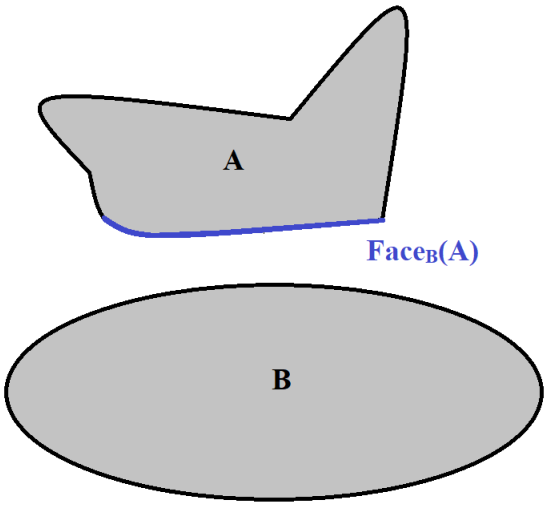

Figure 3.2: Illustration of *Face of set A with respect to set B*. Face$_B(A)$ is shaded blue.

Lemma 3.2.3. *A function is convex if and only if its epigraph is a convex set.*

Proof. Assume that f is a convex function. Then, let $(\mathbf{x}_1, t_1), \ldots, (\mathbf{x}_k, t_k) \in \text{epi}(f)$. So, $t_i \geq f(x_i)$ for all $i = 1, 2, \cdots, k$. Now, for any $\lambda_1, \ldots, \lambda_k \in [0, 1]$ such that $\sum \lambda_i = 1$, the point $(\mathbf{x}, t) := \lambda_i \sum (x_i, t_i) = (\sum \lambda_i x_i, \sum \lambda_i t_i)$ has the property that

$$t = \sum \lambda_i t_i \geq \sum \lambda_i f(x_i) \geq f\left(\sum \lambda_i x_i\right) = f(\mathbf{x}).$$

where the last inequality is due to convexity of f. This implies that $(\mathbf{x}, t) = \lambda_i \sum (x_i, t_i) \in \text{epi}(f)$. Hence, epi$(f)$ is a convex set.

Now, to prove the converse, assume epi(f) is a convex set. Let $\mathbf{x}, \mathbf{y} \in \text{Domain}(f)$. Then, clearly by definition, both the points $(\mathbf{x}, f(\mathbf{x}))$ and $(\mathbf{y}, f(\mathbf{y}))$ lie in epi(f). Since epi(f) is a convex set, it follows that $(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}, \lambda f(\mathbf{x}) + (1-\lambda) f(\mathbf{y})) = \lambda(\mathbf{x}, f(\mathbf{x})) + (1-\lambda)(\mathbf{y}, f(\mathbf{y})) \in \text{epi}(f)$ whenever $\lambda \in [0, 1]$. Hence, by definition of

epi(f), we deduce that $f(\lambda\mathbf{x}+(1-\lambda)\mathbf{y}) \leq \lambda f(\mathbf{x})+(1-\lambda)f(\mathbf{y})$ for $\lambda \in [0,1]$. Therefore, f is a convex function. □

Our first goal is to define that special class of functions we mentioned (but did not define) earlier by generalizing the strict convex functions a bit further. For convenience, we assume that functions in the class we define below has the unique minimum at $\mathbf{x} = \mathbf{0}$.

Definition 3.2.4. We will call a continuously differentiable function $f : \mathbb{R}^n \to \mathbb{R}^+ \cup \{0\}$ belonging to class $\mathcal{N}$ or in short a class $\mathcal{N}$ function, if it satisfies the following conditions:

1. $f(\mathbf{x}) \geq 0$ with equality if and only if $\mathbf{x} = \mathbf{0}$
2. $\nabla f(\mathbf{x})^T \mathbf{x} \geq 0$
3. $\nabla f(\mathbf{x}) = \mathbf{0}$ if and only if $\mathbf{x} = \mathbf{0}$

Here are some examples of class $\mathcal{N}$ functions:

- $f(x,y) = \log(1 + x^2 + y^2)$. This is a non-convex class $\mathcal{N}$ function. The graph of the function is shown in figure 3.3.
- $f(x,y) = x^2 + y^2 + \sin(\frac{x^2}{2}) + \sin(\frac{y^2}{2})$. This is also a non-convex class $\mathcal{N}$ function. The graph of the function is shown in figure 3.4. It is not obvious that the function vanishes only at the origin but we can prove it as follows: First note that we can rewrite the function using trignometric identities as $f(x,y) = 2\sin(\frac{x^2+y^2}{4})\cos(\frac{x^2-y^2}{4}) + x^2 + y^2$. Now, set $a = \frac{x^2+y^2}{4}$ and $b = \frac{y^2}{2}$. Then the function can be rewritten as $f(a,b) = 2\sin(a)\cos(a-b) + 4a$. A straightforward computation gives us $f(0,b) = 0$ for any b and $\frac{\partial f}{\partial a} = 2\cos(2a-b) + 4 = 2(\cos(2a-b)+2) > 0$. That is, fixing b, f is an increasing function in variable

a. Hence, for any b, $f(a,b) > f(0,b) = 0$. Therefore f vanishes only when $a = 0$ equivalently when $x = y = 0$. The details of computation to check properties 2 and 3 is straightforward and is left to the reader.

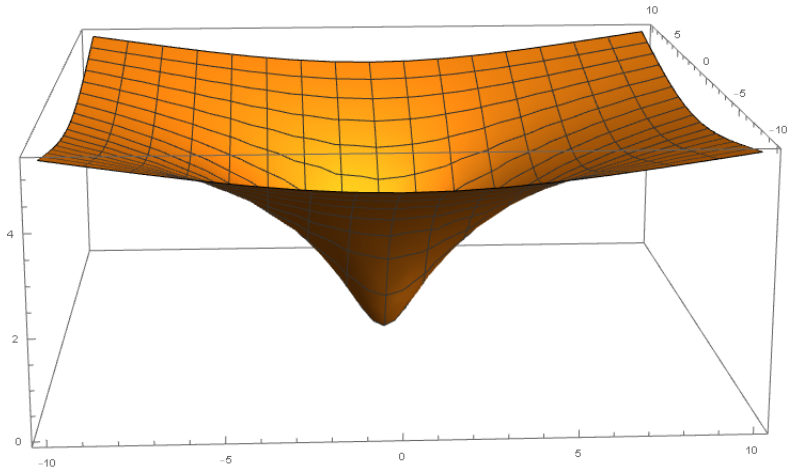

Figure 3.3: Graph of the class $\mathcal{N}$ function $f(x,y) = \log(1 + x^2 + y^2)$

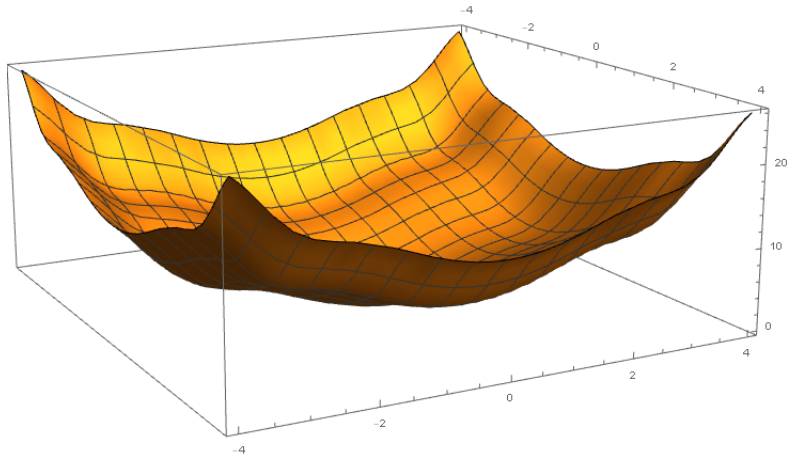

Figure 3.4: Graph of the class $\mathcal{N}$ function $f(x,y) = x^2 + y^2 + \sin(\frac{x^2}{2}) + \sin(\frac{y^2}{2})$

Now we are ready to establish our main results:

3.2.2 Convergence of APA for General Sets

In this section, we will prove the following main theorem (**theorem 3.2.7**):

Theorem: *Let A and B be two sets in $\mathbb{R}^{n+1}$ with a unique intersection point. Assume there exists a hyperplane H that strictly separates the sets A and B except at their unique intersection point. Also, assume that the* $\text{Face}_B(A)$ *and* $\text{Face}_A(B)$ *are given by translates of graph of functions $f|_V$ and $g|_V$ respectively, where $f|_V$ and $g|_V$ are respectively the restriction of some class $\mathcal{N}$ functions f and g to a neighbourhood V of the intersection point. (Refer to figure 3.5 for an illustration) Then, regardless of the starting point, alternate projection converges to their unique intersection point.*

Before we prove the main theorem, we will prove some special cases of the main theorem which will used in the proof of the main theorem.

Theorem 3.2.5. *Let f be a class $\mathcal{N}$ function and let set A be the epigraph of the function f in $\mathbb{R}^{n+1}$. $A := \{\mathbf{x} \in \mathbb{R}^{n+1} \mid x_{n+1} \geq f(x_1, x_2, \cdots, x_n)\}$. Let H be a hyperplane, which we assume, without loss of generality, to be the hyperplane defined as $H := \{\mathbf{x} \in \mathbb{R}^{n+1} \mid x_{n+1} = 0\}$. Then, regardless of the starting point, the alternating projection algorithm converges to the origin, the unique intersection point of sets A and H.*

Proof. Note that for any starting point $\mathbf{x}^0$, $\mathbf{x}^1 = \mathcal{P}_A\mathcal{P}_H(\mathbf{x}^0)$ lies in A and all the subsequent iterates also lies in A. So, without loss of generality, we can assume all the iterates $\mathbf{x}^k$ lies in A.

Now, let $\mathbf{x}^k = (x_1^k, x_2^k, \cdots, x_{n+1}^k) \in A$ be the k^{th} iterate in the alternate projection

algorithm. Begin by noting that $\mathcal{P}_H(x_1^k, x_2^k, \cdots, x_{n+1}^k) = (x_1^k, x_2^k, \cdots, x_n^k, 0)$. Now,

$$\begin{aligned}
\mathbf{x}^{k+1} &= (x_1^{k+1}, x_2^{k+1}, \cdots, x_{n+1}^{k+1}) \\
&= \mathcal{P}_A \mathcal{P}_H(x_1^k, x_2^k, \cdots, x_{n+1}^k) \\
&= \mathcal{P}_A(x_1^k, x_2^k, \cdots, x_n^k, 0) \\
&= \arg\min_{a \in A} (a_1 - x_1^k)^2 + \cdots + (a_n - x_n^k)^2 + (f(a_1, a_2, \cdots, a_n) - 0)^2
\end{aligned}$$

Hence

$$(x_1^{k+1}, x_2^{k+1}, \cdots, x_n^{k+1}) = \arg\min_{a \in A} (a_1 - x_1^k)^2 + \cdots + (a_n - x_n^k)^2 + (f(a_1, a_2, \cdots, a_n) - 0)^2$$

for $i = 1, \cdots, n$

and

$$x_{n+1}^{k+1} = f(x_1^{k+1}, x_2^{k+1}, \cdots, x_n^{k+1})$$

So, taking the gradient of $(a_1 - x_1^k)^2 + \cdots + (a_n - x_n^k)^2 + (f(a_1, a_2, \cdots, a_n) - 0)^2$ and setting it to zero, we deduce that

For each $i = 1, 2, \cdots, n$

$$(x_i^{k+1} - x_i^k) + f(x_1^{k+1}, x_2^{k+1}, \cdots, x_n^{k+1}) \frac{\partial f}{\partial x_i}(x_1^{k+1}, x_2^{k+1}, \cdots, x_n^{k+1}) = 0$$

Therefore,

$$x_i^{k+1} + f(x_1^{k+1}, x_2^{k+1}, \cdots, x_n^{k+1}) \frac{\partial f}{\partial x_i}(x_1^{k+1}, x_2^{k+1}, \cdots, x_n^{k+1}) = x_i^k \tag{3.2.1}$$

Multiplying both sides of the equation by x_i^{k+1} and taking summation, we obtain

$$\sum_{i=1}^{n} (x_i^{k+1})^2 + \sum_{i=1}^{n} x_i^{k+1} f(x_1^{k+1}, x_2^{k+1}, \cdots, x_n^{k+1}) \frac{\partial f}{\partial x_i}(x_1^{k+1}, x_2^{k+1}, \cdots, x_n^{k+1}) = \sum_{i=1}^{n} x_i^k x_i^{k+1}$$

which can be concisely rewritten as

$$\left\|\mathbf{x}^{k+1}\right\|^2 + f(\mathbf{x}^{k+1}) \nabla f(\mathbf{x}^{k+1})^T \mathbf{x}^{k+1} = \langle \mathbf{x}^{k+1}, \mathbf{x}^k \rangle$$

where $\mathbf{x}^k = (x_1^k, \cdots, x_n^k)$

Now, since f is in class $\mathcal{N}$, $f(\mathbf{x}^{k+1}) \geq 0$ and $\nabla f(\mathbf{x}^{k+1})^T \mathbf{x}^{k+1} \geq 0$. Therefore we obtain

$$\left\|\mathbf{x}^{k+1}\right\|^2 \leq \langle \mathbf{x}^{k+1}, \mathbf{x}^k \rangle$$

Using Cauchy-Schwartz inequality on the right side of above equation yields

$$\left\|\mathbf{x}^{k+1}\right\|^2 \leq \left\|\mathbf{x}^{k+1}\right\| \left\|\mathbf{x}^k\right\|$$

Hence, we get $\left\|\mathbf{x}^{k+1}\right\| \leq \left\|\mathbf{x}^k\right\|$. Therefore the sequence $\{\mathbf{x}^k\}_{k=1}^{\infty}$ converges.

So, assume

$$x_i^k \to L_i$$

Then by equation 3.2.1, for each $i = 1, \cdots, n$ we have

$$f(L_1, L_2, \cdots, L_n) \frac{\partial f}{\partial x_i}(L_1, L_2, \cdots, L_n) = 0$$

So, either $f(L_1, L_2, \cdots, L_n) = 0$

OR

$f(L_1, L_2, \cdots, L_n) \neq 0$ and $\frac{\partial f}{\partial x_i}(L_1, L_2, \cdots, L_n) = 0$ for all $i = 1, \cdots, n$. The second case cannot occur because $\frac{\partial f}{\partial x_i}(L_1, L_2, \cdots, L_n) = 0$ for all $i = 1, \cdots, n$ means that $\nabla f(L_1, L_2, \cdots, L_n) = 0$. By hypothesis, ∇f can vanish if and only if $(L_1, L_2, \cdots, L_n) = \mathbf{0}$ which in turn implies that $f(L_1, L_2, \cdots, L_n) = 0$, a contradiction. Hence, we can conclude that the first case is true, namely $f(L_1, L_2, \cdots, L_n) = 0$ which in turn leads to $L_1 = L_2 = \cdots = L_n = 0$. Hence, $L_{n+1} = f(L_1, L_2, \cdots, L_n) = f(0, 0, \cdots, 0) = 0$. In short, $x^k \to 0$. Therefore we conclude that, regardless of the starting point, the alternating projection algorithm iterates converge to the origin which, by hypothesis, is the unique intersection point of the sets A and H.

□

Theorem 3.2.6. *Let f and g be two class $\mathcal{N}$ functions. Let set A be the epigraph of the function f in $\mathbb{R}^{n+1}$. In other words, $A := \{\mathbf{x} \in \mathbb{R}^{n+1} \mid x_{n+1} \geq f(x_1, x_2, \cdots, x_n)\}$. Similarly let B be the set defined as $B := \{\mathbf{x} \in \mathbb{R}^{n+1} \mid x_{n+1} \leq -g(x_1, x_2, \cdots, x_n)\}$. Then, regardless of the starting point, the alternating projection algorithm converges to origin, the unique intersection point of sets A and B.*

Proof. For any starting point $\mathbf{x}^0$, $\mathbf{x}^1 = \mathcal{P}_A\mathcal{P}_B(\mathbf{x}^0)$ lies in A and all the subsequent iterates also lies in A. So, without loss of generality, we can assume all the iterates $\mathbf{x}^k$ lie in A and similarly assume that all iterates $\mathbf{y}^k$ lie in B.

Now, recall the alternating projection algorithm:

$$\mathbf{y}^k := \mathcal{P}_B(\mathbf{x}^k) \text{ and } \mathbf{x}^{k+1} := \mathcal{P}_A(\mathbf{y}^k) = \mathcal{P}_A\mathcal{P}_B(\mathbf{x}^k)$$

Since $\mathbf{y}^k = \arg\min_{\mathbf{b} \in B} \left\|\mathbf{b} - \mathbf{x}^k\right\|$ and $\mathbf{x}^{k+1} = \arg\min_{\mathbf{a} \in A} \left\|\mathbf{a} - \mathbf{y}^k\right\|$, it is easy to observe that $y^k_{n+1} = -g(y^k_1, y^k_2, \cdots, y^k_n)$ and $x^{k+1}_{n+1} = f(x^{k+1}_1, x^{k+1}_2, \cdots, x^{k+1}_n)$. Using the

argument as in theorem above we deduce that

$$y_i^k + \left(f(x_1^k, \cdots, x_n^k) + g(y_1^k, \cdots, y_n^k)\right) \frac{\partial g}{\partial x_i}(y_1^k, \cdots, y_n^k) = x_i^k \tag{3.2.2}$$

and

$$x_i^{k+1} + \left(f(x_1^{k+1}, \cdots, x_n^{k+1}) + g(y_1^k, \cdots, y_n^k)\right) \frac{\partial f}{\partial x_i}(x_1^{k+1}, \cdots, x_n^{k+1}) = y_i^k \tag{3.2.3}$$

for $i = 1, \cdots, n$

Now, using the fact that f and g are class $\mathcal{N}$ functions, a similar argument as in above theorem yields us the following:

$$\left\|\mathbf{y}^k\right\|^2 + (f(\mathbf{x}^k) + g(\mathbf{y}^k))\nabla g(\mathbf{y}^k)^T \mathbf{y}^k = \langle \mathbf{y}^k, \mathbf{x}^k \rangle \tag{3.2.4}$$

and

$$\left\|\mathbf{x}^{k+1}\right\|^2 + (f(\mathbf{x}^{k+1}) + g(\mathbf{y}^k))\nabla f(\mathbf{x}^{k+1})^T \mathbf{x}^{k+1} = \langle \mathbf{x}^{k+1}, \mathbf{y}^k \rangle \tag{3.2.5}$$

Since f and g are class $\mathcal{N}$ functions, f and g are non-negative functions and $\nabla g(\mathbf{y}^k)^T \mathbf{y}^k, \nabla f(\mathbf{x}^{k+1})^T \mathbf{x}^{k+1} \geq 0$. So, by using Cauchy Schwartz on the right hand side, we obtain

$$\left\|\mathbf{y}^k\right\| \leq \left\|\mathbf{x}^k\right\|$$

and

$$\left\|\mathbf{x}^{k+1}\right\| \leq \left\|\mathbf{y}^k\right\|$$

Therefore, for all $k = 1, \cdots$, $\left\|\mathbf{x}^{k+1}\right\| \leq \left\|\mathbf{y}^k\right\|$ and $\left\|\mathbf{y}^{k+1}\right\| \leq \left\|\mathbf{x}^k\right\|$. so the sequence

$\{\mathbf{x}^k\}_{k=1}^{\infty}$ converges. So, assume

$$\mathbf{x}^k \to \mathbf{L}_1 \text{ and } \mathbf{y}^k \to \mathbf{L}_2$$

Then by equation 3.2.4 and 3.2.5, we get

$$\|\mathbf{L}_2\|^2 + (f(\mathbf{L}_1) + g(\mathbf{L}_2))\, \nabla g(\mathbf{L}_2)^T \mathbf{L}_2 = \langle \mathbf{L}_2, \mathbf{L}_1 \rangle \tag{3.2.6}$$

and

$$\|\mathbf{L}_1\|^2 + (f(\mathbf{L}_1) + g(\mathbf{L}_2))\, \nabla f(\mathbf{L}_1)^T \mathbf{L}_1 = \langle \mathbf{L}_2, \mathbf{L}_1 \rangle \tag{3.2.7}$$

adding the equations 3.2.6 and 3.2.7, we obtain

$$\|\mathbf{L}_1 - \mathbf{L}_2\|^2 + (f(\mathbf{L}_1) + g(\mathbf{L}_2)) \left(\nabla g(\mathbf{L}_2)^T \mathbf{L}_2 + \nabla f(\mathbf{L}_1)^T \mathbf{L}_1\right) = 0$$

. Noting that the left-hand side is sum of two non-negative terms, we obtain $\mathbf{L}_1 = \mathbf{L}_2$. Now, using a similar argument as in the above theorem allows us to conclude that $\mathbf{L}_1 = \mathbf{L}_2 = 0$ which implies $x_{n+1}^k \to f(\mathbf{L}_1) = f(\mathbf{0}) = 0$. In other words, $\mathbf{x}^k = (x_1^k, \cdots, x_n^k) \to \mathbf{0}$.

Hence, regardless of the starting point, the alternating projection algorithm iterates converge to the origin which, by hypothesis, is the unique intersection point of the sets A and B. □

Now, from the above theorem, we can deduce the following

Theorem 3.2.7. *Let A and B be two sets in $\mathbb{R}^{n+1}$ with a unique intersection point. Assume there exists a hyperplane H that strictly separates the sets A and B except at their unique intersection point. Also, assume that* $\text{Face}_A(B)$ *and* $\text{Face}_B(A)$ *are given by the graph of functions $f|_V$ and $g|_V$ respectively, where $f|_V$ and $g|_V$ are respectively*

the restriction of some class $\mathcal{N}$ functions f and g to a neighbourhood V of the intersection point. (Refer to figure 3.5 for an illustration). Then, regardless of the starting point, the Alternate Projection algorithm converges to their unique intersection point.

Proof. After a translation and change of coordinates if necessary, without loss of generality assume that the hyperplane H can be described by the equation $\{x_{n+1} = 0\}$ and the unique intersection point is the origin $(0, 0, \cdots, 0)$. Then the result follows from theorem 3.2.6. □

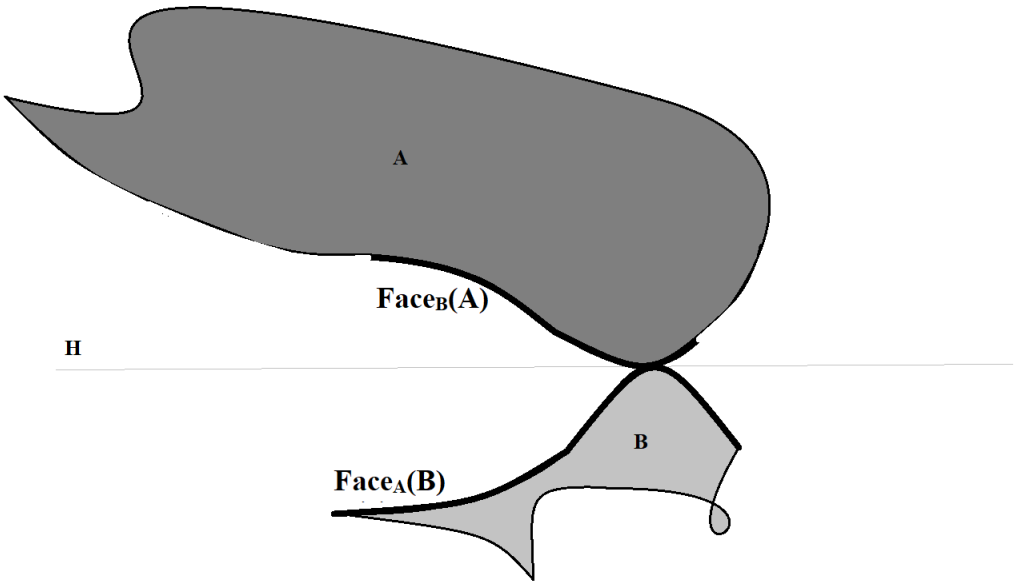

Figure 3.5: A and B are examples of sets where $\text{Face}_B(A)$ and $\text{Face}_A(B)$ are graphs of class $\mathcal{N}$ functions.

3.3 Convex Sets: A Special Case

In this section, we will show that if two sets A and B have the following properties:

1. A and B are closed and convex sets.

2. A and B have smooth boundaries ∂A and ∂B.

3. A and B intersect at a unique point $\mathbf{q}$.

4. A and B can be separated by a hyperplane H, whose equation is say

$$H : \langle \mathbf{v}, \cdot \rangle = c$$

where $\mathbf{v}$ is the unit normal vector of H pointing towards A and c is some constant. That is,

$$\langle \mathbf{v}, \mathbf{a} \rangle \geq c, \qquad \langle \mathbf{v}, \mathbf{b} \rangle \leq c \tag{3.3.1}$$

for all $\mathbf{a} \in A$ and $\mathbf{b} \in B$. Additionally $A \cap H = B \cap H = \{\mathbf{q}\}$.

then, they become a special case of the general sets we considered in theorem 3.2.7.

We will begin by proving that a convex function is a class $\mathcal{N}$ function.

Lemma 3.3.1. *A non-negative continuously differentiable convex function $f : \mathbb{R}^n \to \mathbb{R}^+ \cup \{0\}$ with a unique minimizer is a class $\mathcal{N}$ function after a suitable translation. In particular, strictly convex smooth functions are class $\mathcal{N}$ functions upto a rigid transformation.*

Proof. It is well known that if f is a strictly convex function, f has a unique minimizer. Assume now that f has a unique minimizer. So, let $\arg\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) = \mathbf{x}^\star$. Also, suppose that $m = f(\mathbf{x}^\star) = \min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$. Then we claim that the translate $g(\mathbf{x}) := f(\mathbf{x} + \mathbf{x}^\star) - m$ is a class $\mathcal{N}$ function. In fact, it is clear that $g(\mathbf{0}) = \mathbf{0}$ and $g(\mathbf{x}) \geq 0$. It is also clearly evident that $\nabla g(\mathbf{x}) = \nabla f(\mathbf{x} + \mathbf{x}^\star)$. Now, since f is strictly convex and continuously

differentiable, we deduce that ∇f vanishes only at the minimizer $\mathbf{x}^\star$. Hence, $\nabla g(\mathbf{x}) = \mathbf{0}$ if and only if $\mathbf{x} = \mathbf{0}$. It is now left to prove that $\nabla g(\mathbf{x})^T\mathbf{x} \geq 0$.

In fact, using a variant of Kachurovskii's theorem, stated below as lemma 3.3.2, we obtain $\nabla g(\mathbf{x})^T(\mathbf{y} - \mathbf{x}) < g(\mathbf{y}) - g(\mathbf{x})$. In particular, setting $\mathbf{y} = 0$, we obtain $\nabla g(\mathbf{x})^T\mathbf{x} > g(\mathbf{x}) > 0$.

□

Lemma 3.3.2 (Kachurovskii [41],1960)**.** *Let K be a convex subset of a Banach space V and let $g : K \to \mathbb{R} \cup \{+\infty\}$ be an extended real-valued function that is Frchet differentiable with derivative $dg(\mathbf{x}) : V \to \mathbb{R}$ at each point $\mathbf{x}$ in K. (In fact, $dg(\mathbf{x})$ is an element of the continuous dual space $V^\star$.) Then, the following are equivalent:*

- *g is a convex function;*
- $\big(\mathrm{d}g(\mathbf{x}) - \mathrm{d}g(\mathbf{y})\big)(\mathbf{x} - \mathbf{y}) \geq 0.$
- $\mathrm{d}g(\mathbf{x})(\mathbf{y} - \mathbf{x}) \leq g(\mathbf{y}) - g(\mathbf{x})$

Now, we have to show that $\mathrm{Face}_A(B)$ and $\mathrm{Face}_B(A)$ are graphs of class $\mathcal{N}$ functions.

Remark 3.3.3. *From now on until the end of the chapter, we will only prove statements with respect to sets A and H. Please be aware that corresponding statements and definitions with respect to sets B and H also holds true.*

In this section we will prove that $\mathrm{Face}_A(B)$ is the graph of a convex function in a suitable sense. That, along with lemma 3.3.1, will complete our argument that when the sets A and B are convex sets, we are in the setting of theorem 3.2.7.

We will begin by constructing functions the convex functions f and g. Define the functions $f : \mathrm{Face}_A(H) \to \mathbb{R}$ and $g : \mathrm{Face}_B(H) \to \mathbb{R}$ as follows: Let $\mathbf{a}^\star$ be the point closest to $\mathbf{h}$ among all points $\mathbf{a} \in A$ that has the property $\mathcal{P}_H(\mathbf{a}) = \mathbf{h}$. Then,

$f(\mathbf{h}) = \|\mathbf{a}^\star - \mathbf{h}\|$. In other words, $f(\mathbf{h})$ is the solution to the optimization problem

$$\min_{\mathbf{a}\in A \text{ s.t. } \mathcal{P}_H(\mathbf{a})=\mathbf{h}} \|\mathbf{a}-\mathbf{h}\|\,, \tag{3.3.2}$$

That is,

$$f(\mathbf{h}) := \min_{\mathbf{a}\in A \text{ s.t. } \mathcal{P}_H(\mathbf{a})=\mathbf{h}} \|\mathbf{a}-\mathbf{h}\|$$

for all $\mathbf{h} \in \text{Face}_A(H)$

Similarly define

$$g(\mathbf{h}) := \min_{\mathbf{b}\in B \text{ s.t. } \mathcal{P}_H(\mathbf{b})=\mathbf{h}} \|\mathbf{b}-\mathbf{h}\|$$

for all $\mathbf{h} \in \text{Face}_B(H)$. See figure 3.6.

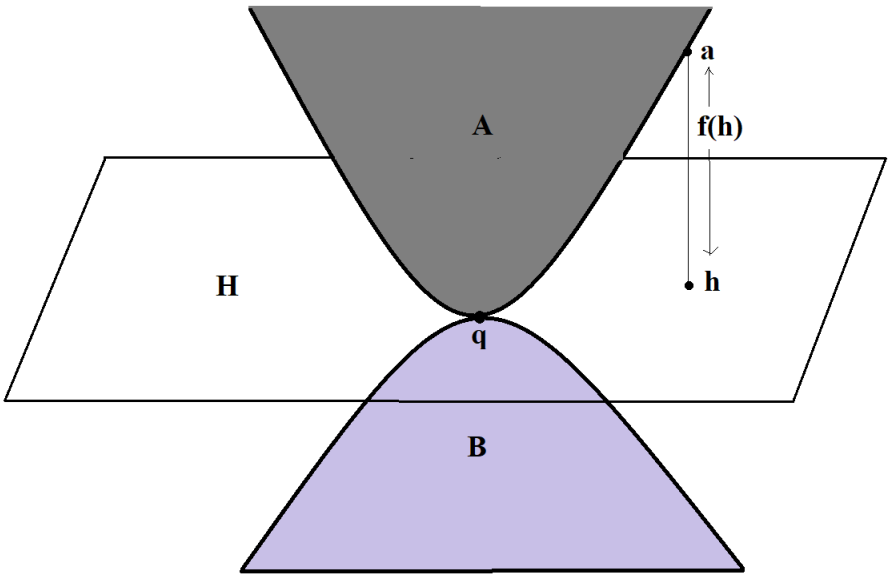

Figure 3.6: H is a hyperplane separating convex sets A and B; Point $\mathbf{h}$ is the projection of point $\mathbf{a}$ onto H and $f(\mathbf{h}) = \|\mathbf{a}-\mathbf{h}\|$

Remark 3.3.4. We have few remarks in order:

(a) It is easy to see that the Problem 3.3.2 has an unique minimizer. Indeed, set of minimizers of the Problem 3.3.2 forms a closed convex set. In fact, the set $\mathcal{P}_H^{-1}(\mathbf{h})$ of points $\mathbf{x} \in \mathbb{R}^{n+1}$ such that $\mathcal{P}_H(\mathbf{x}) = \mathbf{h}$ forms a closed convex cone, see Phelps [53] and Phelps [54] (This holds true for actually *any* set H in an Euclidean space). Since A is a closed convex set and the intersection of two closed convex sets is a closed convex set, the set $\mathcal{P}_H^{-1}(\mathbf{h}) \cap A$ of points $\mathbf{a} \in A$ such that $\mathcal{P}_H(\mathbf{a}) = \mathbf{h}$ forms a closed convex set. Now, the uniqueness follows from the fact that the distance of the points in a closed convex set to any fixed point has an unique minimizer. In particular, the distance of points in $\mathcal{P}_H^{-1}(\mathbf{h}) \cap A$ to $\mathbf{h}$ has an unique minimizer.

From above discussion, it is clear that we can define a single-valued function

$$\phi_f : \mathrm{Face}_A(H) \to A$$

defined by

$$\phi_f(\mathbf{h}) = \underset{\mathbf{a} \in A \text{ s.t. } \mathcal{P}_H(\mathbf{a}) = \mathbf{h}}{\arg\min} \|\mathbf{a} - \mathbf{h}\|$$

Similarly define ϕ_g as

$$\phi_g(\mathbf{h}) = \underset{\mathbf{b} \in B \text{ s.t. } \mathcal{P}_H(\mathbf{b}) = \mathbf{h}}{\arg\min} \|\mathbf{b} - \mathbf{h}\|$$

(b) The points $\phi_f(\mathbf{h})$ are determined by $\mathbf{h}$ and $f(\mathbf{h})$. Indeed, $\phi_f(\mathbf{h}) - \mathbf{h} = \phi_f(\mathbf{h}) - \mathcal{P}_H(\phi_f(\mathbf{h})) = \beta\mathbf{v}$ for some scalar β, where $\mathbf{v}$ is the unit normal to H (see equation 3.3.1). From equation 3.3.1, $\langle \phi_f(\mathbf{h}), \mathbf{v} \rangle \geq 0$ which implies that $\langle \mathbf{h} + \beta\mathbf{v}, \mathbf{v} \rangle = \langle \mathbf{h}, \mathbf{v} \rangle + \beta\langle \mathbf{v}, \mathbf{v} \rangle = c + \beta \|\mathbf{v}\|^2 = c + \beta \|\mathbf{v}\|^2 = c + \beta \geq c$. Hence $\beta \geq 0$. Now, $f(\mathbf{h}) = \|\phi_f(\mathbf{h}) - \mathbf{h}\| = |\beta| \|\mathbf{v}\| = |\beta| = \beta$. Therefore, $\phi_f(\mathbf{h}) = \mathbf{h} + f(\mathbf{h})\mathbf{v}$. Said differently, if, after a translation, the ambient space is viewed

as span$\{H, \mathbf{v}\}$, then $\phi_f(\mathbf{h})$ would have coordinates $(\mathbf{h}, f(\mathbf{h}))$. In short, $\phi_f(\mathbf{h})$ lies on the graph of f in the new coordinate system. Hence, $\phi_f(\text{Face}_A(H))$ lies in the graph, in the sense described above, of f .

Lemma 3.3.5.

$$\text{Face}_B(A) \subseteq \phi_f(\text{Face}_A(H))$$

where the map ϕ_f is as defined in part (a) of remark 3.3.4. Similarly,

$$\text{Face}_A(B) \subseteq \phi_g(\text{Face}_B(H))$$

. see figure 3.7 for an illustration.

Proof. We begin by noting that, since A, B and H are closed convex sets, the projection operators associated with them are single-valued. Hence $\text{Face}_B(A) = \mathcal{P}_A(B)$ and $\text{Face}_A(H) = \mathcal{P}_H(A)$. So, our goal is to prove that $\mathcal{P}_A(B) \subseteq \phi_f(\mathcal{P}_H(A))$. So, let $\tilde{\mathbf{b}} \in B$ be an element of B. Let $\tilde{\mathbf{a}} := \mathcal{P}_A(\tilde{\mathbf{b}})$ and $\tilde{\mathbf{h}} := \mathcal{P}_H(\tilde{\mathbf{a}})$. We have to show that $\tilde{\mathbf{a}} = \phi_f(\mathbf{h})$ for some $\mathbf{h} \in H$. we claim the following:

Claim: $\tilde{\mathbf{a}} = \phi_f(\tilde{\mathbf{h}})$

Proof: By definition of ϕ_f, we have to show that

$$\tilde{\mathbf{a}} = \underset{\mathbf{a} \in A \text{ s.t. } \mathcal{P}_H(\mathbf{a}) = \tilde{\mathbf{h}}}{\arg\min} \left\| \mathbf{a} - \tilde{\mathbf{h}} \right\|$$

By definition of $\tilde{\mathbf{a}}$, $\tilde{\mathbf{a}} \in A$ and $\mathcal{P}_H(\tilde{\mathbf{a}}) = \tilde{\mathbf{h}}$. Hence, $\tilde{\mathbf{a}}$ is in the feasible set of the optimization problem. So, we are left to prove that it is in fact the minimizer.

So, let us assume, on the contrary, that there exist $\mathbf{a} \in A$ such that $\mathcal{P}(\mathbf{a}) = \tilde{\mathbf{h}}$ and $\left\| \mathbf{a} - \tilde{\mathbf{h}} \right\| < \left\| \tilde{\mathbf{a}} - \tilde{\mathbf{h}} \right\|$. Since $\mathcal{P}_H(\mathbf{a}) = \tilde{\mathbf{h}}$ and $\mathcal{P}_H(\tilde{\mathbf{a}}) = \tilde{\mathbf{h}}$, we can write $\mathbf{a} = \tilde{\mathbf{h}} + \beta\mathbf{v}$ and $\tilde{\mathbf{a}} = \tilde{\mathbf{h}} + \tilde{\beta}\mathbf{v}$ for some non-negative constants β and $\tilde{\beta}$; it is non-negative because

$\mathbf{a}, \tilde{\mathbf{a}} \in A$ and inequality 3.3.1. Using the assumption $\left\|\mathbf{a} - \tilde{\mathbf{h}}\right\| < \left\|\tilde{\mathbf{a}} - \tilde{\mathbf{h}}\right\|$, we get

$$0 < \beta < \tilde{\beta}$$

.

Also, convexity of the set A and lemma 3.1.2 gives us

$$\langle \tilde{\mathbf{b}} - \tilde{\mathbf{a}}, \mathbf{a} - \tilde{\mathbf{a}} \leq 0 \rangle$$

which implies

$$\langle \tilde{\mathbf{b}} - \tilde{\mathbf{a}}, (\beta - \tilde{\beta})\mathbf{v} \leq 0 \rangle$$

Using the above inequality, namely $0 < \beta < \tilde{\beta}$, we get

$$\langle \tilde{\mathbf{b}} - \tilde{\mathbf{a}}, \mathbf{v} \leq 0 \rangle \tag{3.3.3}$$

Now,

$$\begin{aligned}
\left\|\tilde{\mathbf{b}} - \tilde{\mathbf{a}}\right\|^2 - \left\|\tilde{\mathbf{b}} - \mathbf{a}\right\|^2 &= \|\tilde{\mathbf{a}}\|^2 - \|\mathbf{a}\|^2 - 2\langle \tilde{\mathbf{a}}, \tilde{\mathbf{b}} \rangle + 2\langle \mathbf{a}, \tilde{\mathbf{b}} \rangle \\
&= \left(\left\|\tilde{\mathbf{h}}\right\|^2 + \tilde{\beta}^2 + 2\tilde{\beta}\langle \tilde{\mathbf{h}}, \mathbf{v} \rangle \right) - \left(\left\|\tilde{\mathbf{h}}\right\|^2 + \beta^2 + 2\beta\langle \tilde{\mathbf{h}}, \mathbf{v} \rangle \right) \\
&\quad + 2\langle \mathbf{a} - \tilde{\mathbf{a}}, \tilde{\mathbf{b}} \rangle \\
&= \left(\left\|\tilde{\mathbf{h}}\right\|^2 + \tilde{\beta}^2 + 2\tilde{\beta}c \right) - \left(\left\|\tilde{\mathbf{h}}\right\|^2 + \beta^2 + 2\beta c \right) + 2\langle (\beta - \tilde{\beta})\mathbf{v}, \tilde{\mathbf{b}} \rangle \\
&= \left(\tilde{\beta}^2 - \beta^2 \right) + 2(\beta - \tilde{\beta})(\langle \mathbf{v}, \tilde{\mathbf{b}} \rangle - c) \\
&= \left(\tilde{\beta}^2 - \beta^2 \right) + 2(\tilde{\beta} - \beta)(c - \langle \mathbf{v}, \tilde{\mathbf{b}} \rangle) \\
&> 0
\end{aligned} \tag{3.3.4}$$

where in the last step we used the fact that $\tilde{\mathbf{b}} \in B$ and $\langle \mathbf{v}, \mathbf{b} \rangle \leq c$ for all $\mathbf{b} \in B$.

Now recall that $\mathcal{P}_A(\tilde{\mathbf{b}}) = \tilde{\mathbf{a}}$. Therefore, we must have

$$\left\|\tilde{\mathbf{b}} - \tilde{\mathbf{a}}\right\| \leq \left\|\tilde{\mathbf{b}} - \hat{\mathbf{a}}\right\|$$

for any $\hat{\mathbf{a}} \in A$. In particular,

$$\left\|\tilde{\mathbf{b}} - \tilde{\mathbf{a}}\right\| \leq \left\|\tilde{\mathbf{b}} - \mathbf{a}\right\|$$

This is a contradiction to the inequality 3.3.4, thereby completing the proof. □

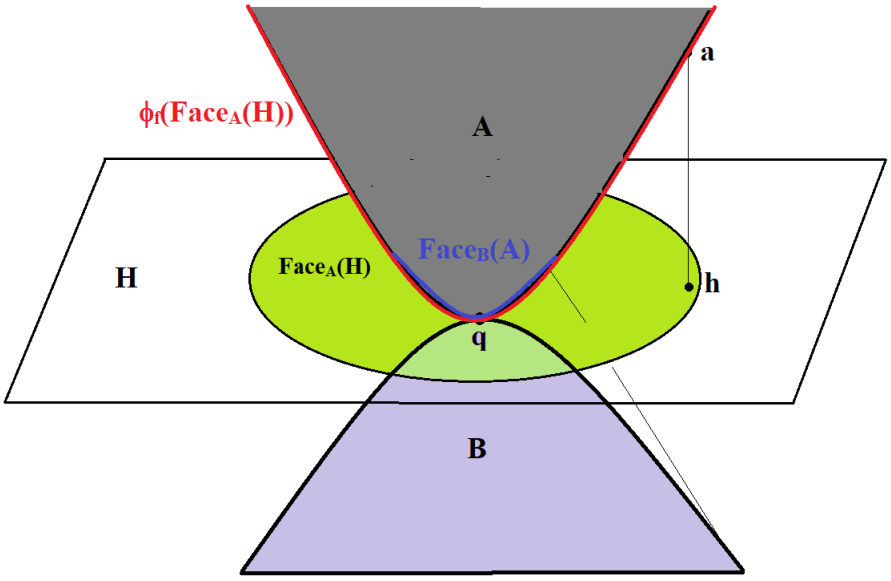

Figure 3.7: An illustration showing $\text{Face}_B(A) \subseteq \phi_f(\text{Face}_A(H))$

We are now in a position to prove that the 'graph' of f contains $\text{Face}_B(A)$.

Proposition 3.3.6. *The graph of f contains* $\text{Face}_B(A)$ *where the graph of f is in the sense that is described in part (b) of remark 3.3.4*

Proof. From discussion in part (b) of remark 3.3.4, it follows that graph of f is $\phi_f(\text{Face}_A(H))$. The result now follows from proposition 3.3.5. □

Lemma 3.3.7. *Let S be any set in the Euclidean space. Then, for any $\mathbf{x}$ such that $\|\mathbf{x} - \mathcal{P}_S(\mathbf{x})\| = \operatorname{dist}(\mathbf{x}, S) > 0$, we have*

$$\mathbf{s} := \mathcal{P}_S(\mathbf{x}) \in \partial S$$

where ∂S denotes the boundary of set S defined by $\partial S := S \backslash \operatorname{int}(S)$. In particular,

$$\operatorname{Face}_B(A) \subseteq \phi_f(\operatorname{Face}_A(H)) \subseteq \partial A$$

Proof. Assume, on the contrary, that $\mathcal{P}_S(\mathbf{x}) \in \operatorname{int}(S)$. Then, there exist $\epsilon > 0$ such that $\operatorname{Ball}(\mathbf{s}, \epsilon) \in S$. We can choose ϵ small enough so that $0 < \epsilon < 2\|\mathbf{x} - \mathbf{s}\|$. Now, let $\hat{\mathbf{s}} := \mathbf{s} + \frac{\epsilon}{2}\frac{\mathbf{x}-\mathbf{s}}{\|\mathbf{x}-\mathbf{s}\|}$. Then,

$$\begin{aligned}
\|\mathbf{x} - \hat{\mathbf{s}}\| &= \left\| \mathbf{x} - \mathbf{s} + \frac{\epsilon}{2}\frac{\mathbf{x}-\mathbf{s}}{\|\mathbf{x}-\mathbf{s}\|} \right\| \\
&= \left\| \left(1 - \frac{\epsilon}{2\|\mathbf{x}-\mathbf{s}\|}\right)(\mathbf{x}-\mathbf{s}) \right\| \\
&= \left(1 - \frac{\epsilon}{2\|\mathbf{x}-\mathbf{s}\|}\right)\|\mathbf{x}-\mathbf{s}\| \\
&\left(1 - \frac{\epsilon}{2\|\mathbf{x}-\mathbf{s}\|}\right)\|\mathbf{x} - \mathcal{P}_S(\mathbf{x})\| \qquad\qquad < \|\mathbf{x} - \mathcal{P}_S(\mathbf{x})\|
\end{aligned}$$

which is a contradiction.

For proving the second part of the lemma, let $C_{\mathbf{h}} := \mathcal{P}_H^{-1}(\mathbf{h}) \cap A$. Then, from part (a) of remark 3.3.4, we infer that

$$\mathcal{P}_{C_{\mathbf{h}}}(\mathbf{h}) = \phi_f(\mathbf{h})$$

for all $\mathbf{h} \in H$. Therefore,

$$\phi_f(\mathbf{h}) \in \partial C_{\mathbf{h}} = \partial(\mathcal{P}_H^{-1}(\mathbf{h}) \cap A) \subseteq \partial A$$

except possibly at the intersection point. Note that we used the fact that A is closed to So, $\phi_f(\text{Face}_A(H)) \subseteq \partial A$. The other inclusion was proved in lemma 3.3.5. $\square$

We will now proceed to prove that f and g are convex functions. But before we do that we need a lemma. Although the projection to an affine linear space is not a linear operation in general, it has the following nice property:

Lemma 3.3.8. *Let $H = \mathbf{p} + K$ be a affine linear space in a Hilbert space, where K is a vector subspace and $\mathbf{p}$ is a point in the Hilbert space. Then, for any $\mathbf{x}_1, \mathbf{x}_2$ and scalar λ,*

$$\mathcal{P}_H(\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2) = \lambda \mathcal{P}_H(\mathbf{x}_1) + (1-\lambda)\mathcal{P}_H(\mathbf{x}_2)$$

Proof.

$$\begin{aligned}
\mathcal{P}_H(\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2) &= \mathbf{p} + \mathcal{P}_K(\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2 - \mathbf{p}) \\
&= \mathbf{p} + \mathcal{P}_K(\lambda(\mathbf{x}_1 - \mathbf{p}) + (1-\lambda)(\mathbf{x}_2 - \mathbf{p})) \\
&= \mathbf{p} + \lambda \mathcal{P}_K(\mathbf{x}_1 - \mathbf{p}) + (1-\lambda)\mathcal{P}_K(\mathbf{x}_2 - \mathbf{p}) \\
&= \lambda(\mathbf{p} + \mathcal{P}_K(\mathbf{x}_1 - \mathbf{p})) + (1-\lambda)(\mathbf{p} + \mathcal{P}_K(\mathbf{x}_2 - \mathbf{p})) \\
&= \lambda \mathcal{P}_H(\mathbf{x}_1) + (1-\lambda)\mathcal{P}_H(\mathbf{x}_2)
\end{aligned}$$

Note that we used the linearity of the projection operator to a linear space in the third step above. $\square$

Lemma 3.3.9. *The functions f and g defined above are smooth convex functions*

Proof. We will prove that f is convex. The proof of g being convex is similar.

Let $\mathbf{h}_1, \mathbf{h}_2 \in \text{Face}_A(H)$. Firstly, recall that

$$f(\mathbf{h}) = \min_{\mathbf{a} \in A \text{ s.t. } \mathcal{P}_H(\mathbf{a}) = \mathbf{h}} \|\mathbf{a} - \mathbf{h}\|,$$

So, let

$$\mathbf{a}_i = \underset{\mathbf{a}\in A \text{ s.t. } \mathcal{P}_H(\mathbf{a})=\mathbf{h}_i}{\arg\min} \|\mathbf{a}-\mathbf{h}_i\|$$

for each $i = 1, 2$

Then, $f(\mathbf{h}_1) = \|\mathbf{a}_1 - \mathbf{h}_1\|, f(\mathbf{h}_2) = \|\mathbf{a}_1 - \mathbf{h}_2\|, \mathcal{P}_H(\mathbf{a}_1) = \mathbf{h}_1$ and $\mathcal{P}_H(\mathbf{a}_2) = \mathbf{h}_2$.

Since a hyperplane is an affine linear space, we can use lemma 3.3.8 and get

$$\mathcal{P}_H(\lambda\mathbf{a}_1 + (1-\lambda)\mathbf{a}_2) = \lambda\mathcal{P}_H(\mathbf{a}_1) + (1-\lambda)\mathcal{P}_H(\mathbf{a}_2) = \lambda\mathbf{h}_1 + (1-\lambda)\mathbf{h}_2 \tag{3.3.5}$$

We also have

$$\begin{aligned}\|\lambda\mathbf{a}_1 + (1-\lambda)\mathbf{a}_2 - \mathcal{P}_H(\lambda\mathbf{a}_1 + (1-\lambda)\mathbf{a}_2)\| &= \|\lambda\mathbf{a}_1 + (1-\lambda)\mathbf{a}_2 - (\lambda\mathbf{h}_1 + (1-\lambda)\mathbf{h}_2)\| \\ &= \|\lambda(\mathbf{a}_1 - \mathbf{h}_1) + (1-\lambda)(\mathbf{a}_2 - \mathbf{h}_2)\| \\ &\leq \lambda\|\mathbf{a}_1 - \mathbf{h}_1\| + (1-\lambda)\|\mathbf{a}_2 - \mathbf{h}_2\| \\ &= \lambda f(\mathbf{h}_1) + (1-\lambda)f(\mathbf{h}_2) \end{aligned} \tag{3.3.6}$$

By definition,

$$f(\lambda\mathbf{h}_1 + (1-\lambda)\mathbf{h}_2) = \min_{\mathbf{a}\in A \text{ s.t. } P_H(\mathbf{a})=\lambda\mathbf{h}_1+(1-\lambda)\mathbf{h}_2} \|\mathbf{a} - (\lambda\mathbf{h}_1 + (1-\lambda)\mathbf{h}_2)\|$$

So, from equations 3.3.5, 3.3.6 and the fact that $\lambda\mathbf{a}_1 + (1-\lambda)\mathbf{a}_2 \in A$ since A is a convex set, we deduce that

$$f(\lambda\mathbf{h}_1 + (1-\lambda)\mathbf{h}_2) \leq \lambda f(\mathbf{h}_1) + (1-\lambda)f(\mathbf{h}_2)$$

Now, we will prove that f is a smooth function. Since a function is smooth if and only if its graph is a smooth set, using proposition 3.3.6 we obtain that f is smooth if ∂A is smooth which indeed is the case by assumption in the beginning of this section.

Therefore, f is a smooth function. A similar argument shows that g is a smooth function. $\square$

Chapter 4

Alternating Projection Algorithm for Matrix Completion

4.1 Introduction to Matrix Completion

The last two decades have witnessed a resurgence of research in sparse solutions of under-determined linear systems, matrix completion and recovery. The matrix completion problem was inspired by the Netflix problem (cf. Netflix [50]) and was pioneered by Candès, Recht, 2010 [12] and Candès and Tao, 2010 [14]. The problem can be explained as follows. One would like to recover a matrix $M \in \mathbb{R}^{m,n}$ from a given set of entries $M_{ij}, (i,j) \in \Omega \subset \{1, \cdots, m\} \times \{1, \cdots, n\}$ by filling in the missing entries such that the resulting matrix has the lowest possible rank. In other words, we solve the following rank minimization problem:

$$\min_{X \in \mathbb{R}^{m \times n}} \quad \text{rank } (X): \quad \text{such that} \quad \mathcal{A}_\Omega(X) = \mathcal{A}_\Omega(M), \tag{4.1.1}$$

where $\mathcal{A}_\Omega(X) = \mathcal{A}_\Omega(M)$ means the entries of the matrix X are the same entries of matrix M for indices $(i,j) \in \Omega$. In general, $\mathcal{A}_\Omega$ could be any sampling operator, like for example a Gaussian sampling operator $\mathcal{A}_\Omega(X) = \langle G, X \rangle$ where G is a matrix with

entries from a normalized Gaussian distribution. Clearly, if we are only given a few entries, say one entry of matrix M of size 2×2, we are not able to recover M even after assuming that the rank of M is 1. There are necessary conditions on how many entries one must know in order to be able to recover M. Information theoretic lower bound can be found in Candès and Tao [14].

There are many approaches to recover such a matrix developed in the last ten years. One popular approach is to find a matrix with minimal summation of its singular values. That is,

$$\min_{X \in \mathbb{R}^{m \times n}} \{\|X\|_*, \qquad \mathcal{A}_\Omega(X) = \mathcal{A}_\Omega(M)\}, \tag{4.1.2}$$

where $\|X\|_* = \sum_{i=1}^{k} \sigma_i(X)$ is the nuclear norm of X with $k = \min\{m, n\}$ and $\sigma_i(X)$ are singular values of matrix X. It is known that $f(X) = \|X\|_*$ is a convex function of X. So, the above problem (4.1.2) is a convex minimization problem. By adding $\frac{1}{\lambda}\|X\|_F$ to the minimizing functional in (4.1.2), the resulting minimization problem can be solved by using Uzawa type algorithms in [Cai, Candès, Shen, 2010[10]] or solved by using its dual formulation, e.g. in [Lai and Yin, 2013[44]. The minimization in (4.1.2) can also reformulated as a fixed point iteration and Nestrov's acceleration technique can be used. See [Ma, Goldfarb, Chen, 2011[46]] and [Toh and Yun, 2010[63]]. This constrained minimization (4.1.2) is usually converted into an unconstrained minimization using the Lagrange multiplier method or the augmented Lagrange minimization method. The alternating direction method of multiplier (ADMM) can be used to complete a matrix. See [Tao and Yuan, 2011[60]], and [Yang and Yuan, 2012[73]], Many researchers have studied the matrix completion via variants of the constrained convex minimization approach.

Certainly, rank completion is also studied by using other approaches. See [Jain, Meka and Dhillon, 2010 [39]] for the singular value projection method and [Wen, Yin,

Zhang, 2012[71], [Tanner and Wei, 2016[59]] for alternating least squares, the SOR approaches, steepest descent minimization approaches. See [Lai, Xu, Yin, 2013[45]] for ℓ_q minimization approach for $q \in (0, 1)$. In addition, a greedy approach, e.g. orthogonal matching pursuit (OMP) and iterative hard thresholding approach can be used as well. See [Wang, Lai, Lu, and Ye, 2015 [69]] and [Tanner and Wei, 2013[58]]. Iteratively reweighted nuclear norm minimization, Riemannian conjugated gradient method, and alternating projection algorithm in [Mohan and Fazel, 2012[48]], [Vandereycken, 2013[65]], [Cai, Wang, and Wei, 2016[11]], [Wei, Cai, Chan, and Leung, 2016[70]], [Jiang, Zhong, Liu, and Song, 2017[40], and etc.. Among all these algorithms, the computational algorithm proposed in [69] seems the most efficient one in completing an incomplete matrix. However, the absolute accuracy of the completed matrices is still a question. It is a popular practice to use the relative Frobenius norm errors, i.e., $\|M - M_k\|_F / \|M\|_F$ where M_k is the kth iteration from a matrix completion algorithm, to measure the accuracy for a given matrix M of size $m \times n$. When the size of M is very large and so is $\|M\|_F$ and the missing rate $1 - |\Omega|/(mn) << 1$ small or $|\Omega| \approx mn$, the relative Frobenous norm error will be very small anyway and hence will not be a good measure of errors, where Ω is the set of the indices of known entries. Instead, we would use an absolute error measurement such as the maximum norm of all entries of the residual matrix $M - M_k$ to check the accuracy of the completed matrices. With absolute error measurements, many of the existing algorithms mentioned above fail to produce an accurate recovery. Certainly, the main possible reason may be attributed to the relaxation gap stemming from the relaxation of the rank minimization problem. It remains an open problem to find sufficient conditions which ensure the uniqueness of the minimization. However, some sufficient conditions are unrealistic, e.g. only one entry is missing. Designing efficient, scalable and commercially useful matrix completion algorithms is an active area of research. A

numerical study of the Alternating Projection algorithm as applied to matrix completion is found in reference Jiang et al. [40]. We were aware of the good numerical performance of the algorithm prior to the publication of the reference. However, we would like to analyze why the algorithm is convergent, under what kind of conditions the algorithm is convergent, and under what situation the convergence is linear.

The rest of this chapter is organized as follows. In the next section, we study the convergence of the AP algorithm. The section is divided into three subsections. We first study the case when the guess rank r_g is the same as the rank of the matrix to be completed. Next we study the remaining case when r_g is not the same as the rank of the matrix whose known entries are given. In the section 4.3, we will demonstrate the excellent performance of the AP algorithm when starting from an initial matrix obtained using the OR1MP algorithm in Wang et al. [69]. An application to image processing will also be shown to demonstrate the effectiveness of the AP algorithm. Finally, we will conclude this chapter with some remarks on the existence of matrix completion.

4.2 APA for Matrix Completion

For convenience, we will work with square matrices. All our results would hold true for rectangular matrices as well. Let $\mathcal{M}_r$ be the manifold in $\mathbb{R}^{n^2}$ consisting of $n \times n$ matrices of rank r and denote by $P_{\mathcal{M}_r}$ the projection operator onto the manifold $\mathcal{M}_r$. Fix a rank r matrix M of size $n \times n$ (see remark 4.2.1). Next consider the affine space $\mathcal{A}_\Omega$ defined as follows:

$$\mathcal{A}_\Omega := \{X \mid \mathcal{P}_\Omega(X - M) = 0\}.$$

The affine space $\mathcal{A}_\Omega$ consists of matrices which has exactly same entries as M with indices in Ω. Although it is a convex set, $\mathcal{A}_\Omega$ is not a bounded set. Starting with an initial guess $X_0 = \mathcal{P}_\Omega(M)$ or a good initial guess (see our numerical experiments near

the end of this section), the Alternating Projection (AP) Algorithm can be simply stated as follows:

Algorithm 4: Alternating Projection Algorithm for Matrix Completion

Data: Rank r of the solution M, the tolerance ϵ whose default value is 1e-6
Result: X_k, a close approximation of M
Initialize $X_0 = \mathcal{P}_\Omega(M)$ or any other good guess;
repeat
  Step 1: $Y_k = P_{\mathcal{M}_r}(X_k)$
  Step 2: $X_{k+1} = P_{\mathcal{A}_\Omega}(Y_k)$
until $\|X_{k+1} - X_k\| < \epsilon$;

In Algorithm 4 above, the computation of the projection $P_{\mathcal{M}_r}$ can be realized easily by using the singular value decomposition and Eckart-Young-Mirsky theorem (Eckart and Young [26]). $P_{\mathcal{A}_\Omega}$ is the projection onto $\mathcal{A}_\Omega$. The computation $P_{\mathcal{A}_\Omega}(Y_k)$ is obtained simply by setting the matrix entries of Y_k in positions Ω equal to the corresponding entries in M. Therefore, the algorithm is simple and easy without any computationally intensive minimization steps.

Remark 4.2.1. (a) An important issue is the distribution of $\Omega \subset \{(i,j), i,j = 1, \cdots, n\}$. Clearly, if a column of M is completely missing, one is not able to recover this column no matter what rank r of M is and how large $m = |\Omega|$ is. If we let $\mathbf{x} \in \mathbb{R}^{n^2-m}$ be the unknown entries of M, the determinant of the sub-matrix of any $r+1$ rows and $r+1$ columns of M will be zero which forms a polynomial equation with coefficients formed from known entries $M|_\Omega$. We have $n^2 - m$ unknowns while $\binom{n}{r+1}^2$ submatrices from M which will result in $\binom{n}{r+1}^2$ polynomial equations. Since we have $n^2 - m < n^2 - 2nr + r^2 = (n-r)^2$ unknowns and $\binom{n}{r+1}^2$ equations, the system of polynomial equations is overdetermined. We have to assume that the system is consistent, i.e. the system has a solution. Otherwise, the overdetermined system has no solution, i.e. the matrix M can not be completed. Hence, for the rest of this chapter, let us assume that the overdetermined system of polynomial equations have a solution, i.e.

M can be completed. In fact, If one randomly chose values for the entries in a fixed location set Ω of a matrix, one will most always not be able to complete the matrix to a rank r matrix. See Theorem 4.4.3. Hence, we shall discuss the convergence of the AP algorithm under the assumption that the given entries are from a rank r matrix to begin with.

As discussed in the remark 4.2.1, for the rest of this section, we shall assume that the given entries are from a matrix of rank r. However, in general, we do not know the rank r of M in advance. Thus, we have to make a guess of r. Let r_g be a guessed rank. As we know, any reasonable choice of r_g must satisfy $m > 2nr_g - r_g^2$, we still have either $r_g < r$, $r_g = r$ or $r_g > r$. When $r_g = \text{rank}(M)$, we can show that Algorithm 4 converge to M_{r_g} linearly. Otherwise, when $r_g \neq \text{rank}(M)$, Algorithm 4 converges to a matrix which has rank at the most r_g under some conditions and may not be the desired matrix M. Thus, this section is divided into two parts. We shall discuss the two cases in the two subsections.

4.2.1 Convergence of Algorithm 4 When $r_g = \textbf{Rank}(M)$

We start with some preliminary results.

Lemma 4.2.2. *Let L be a linear subspace of $\mathbb{R}^n$. Suppose P_L denote the orthogonal projection onto L. Then, for any $x \in \mathbb{R}^n$*

$$\|x\| = \|P_L(x)\| \text{ if and only if } x \in L$$

Equivalently,

$$\|P_L(x)\| < \|x\| \text{ if and only if } x \notin L$$

Proof. The 'if' part is clear. So, let us prove the 'only if' part.

Let $l_1, l_2, \cdots l_k$ be a orthonormal basis of L. Extend it to a orthonormal basis

$l_1, l_2, \cdots l_n$ of $\mathbb{R}^n$. Then,

$$x = \sum_{i=1}^{n} \langle x, l_i \rangle l_i$$

and

$$\|x\|^2 = \sum_{i=1}^{n} \langle x, l_i \rangle^2 = \|P_L(x)\|^2 + \sum_{i=k+1}^{n} \langle x, l_i \rangle^2$$

Now it follows that if $\|x\| = \|P_L(x)\|$, then $\sum_{i=k+1}^{n} \langle x, l_i \rangle^2 = 0$, which implies $\langle x, l_i \rangle = 0$ for all $i \geq k+1$. Therefore, $x = \sum_{i=1}^{k} \langle x, l_i \rangle l_i \in L$. □

Lemma 4.2.3. *Let L_1 and L_2 be two linear subspaces of $\mathbb{R}^n$. Suppose P_{L_1} and P_{L_2} denote the orthogonal projection onto L_1 and L_2 respectively. Then, $L_1 \cap L_2 = \{0\}$ if and only if*

$$\|P_{L_2} P_{L_1}\| < 1. \tag{4.2.1}$$

Proof. Assume $L_1 \cap L_2 = \{0\}$. Let $x \neq 0 \in \mathbb{R}^n$. Then if $P_{L_1}(x) = 0$, then $P_{L_2} P_{L_1}(x) = 0 < \|x\|$. Otherwise, $P_{L_1}(x) \neq 0$. Since $L_1 \cap L_2 = \{0\}$, $P_{L_1}(x) \notin L_2$. Therefore, using Lemma 4.2.2, we get

$$\|P_{L_2} P_{L_1}(x)\| < \|P_{L_1}(x)\| \leq \|P_{L_1}\| \, \|x\| \leq \|x\| .$$

Hence, we have

$$\|P_{L_2} P_{L_1}(x)\| < \|x\|$$

for all non-zero $x \neq 0 \in \mathbb{R}^n$. So,

$$\|P_{L_2} P_{L_1}\| < 1.$$

To prove the other direction, assume $\|P_{L_2} P_{L_1}\| < 1$. Assume, on the contrary, that $L_1 \cap L_2 \neq \{0\}$. Let $x \neq 0 \in L_1 \cap L_2$ be a nonzero vector in the intersection. Then $P_{L_2} P_{L_1}(x) = P_{L_2}(x) = x$ which implies that $\|P_{L_2} P_{L_1}(x)\| = \|x\|$, contradicting the assumption. □

Lemma 4.2.4. *Let $M \in \mathcal{M}_r$. Then the projection operator $P_{\mathcal{M}_r}$ is well defined (single-valued) in a neighborhood of M and is differentiable with gradient*

$$\nabla P_{\mathcal{M}_r}(M) = \mathrm{P}_{T_{\mathcal{M}_r}(M)}, \tag{4.2.2}$$

where $T_{\mathcal{M}}(M)$ is the tangent space of $\mathcal{M}$ at M and $\mathrm{P}_{T_{\mathcal{M}}(M)}$ is the projection operator onto the tangent space.

Proof. Since the projection $P_{\mathcal{M}_r}$ of a matrix X is obtained by hard thresholding the least $n-r$ singular values, we see that the projection is unique if $\sigma_r(M) \neq \sigma_{r+1}(M) \geq 0$. Now consider the neighborhood V of M given by

$$V := \left\{ X \in \mathbb{R}^{n\times n} \mid \|X - M\|_F < \frac{\sigma_r(M)}{4} \right\}.$$

Then, by Weyl's [72] or more generally Mirsky's [47] perturbation bounds on singular values, we have

$$|\sigma_r(X) - \sigma_r(M)| \leq \|X - M\|_F < \frac{\sigma_r(M)}{4}$$

and

$$|\sigma_{r+1}(X) - \sigma_{r+1}(M)| \leq \|X - M\|_F < \frac{\sigma_r(M)}{4}.$$

Hence, noting $\sigma_{r+1}(M) = 0$, we observe that

$$\sigma_{r+1}(X) < \frac{\sigma_r(M)}{4} < \frac{3\sigma_r(M)}{4} < \sigma_r(X).$$

In particular,

$$\sigma_r(X) \neq \sigma_{r+1}(X).$$

Therefore, $P_{\mathcal{M}_r}$ is single valued in the neighborhood V.

For second part of the result, we refer to Theorem 25 in Feppon and Lermusiaux [27] which is stated below. We have changed the notations for ease of reading. In particular, note that although the X has rank greater than r in Feppon and Lermusiaux [27], its easy to see that their proof goes through when X has rank greater than or equal to r. Intuitively, it is easy to see that the gradient vector of the projection $P_{\mathcal{M}_r}$ of smooth manifold $\mathcal{M}_r$ at M will be the projection onto the tangent plane $T_{\mathcal{M}_r}$ at M in general. □

The following result was used in the proof above.

Theorem 4.2.5 (Theorem 25 in Feppon and Lermusiaux [27]). *Consider $X \in \mathbb{R}^{n\times m}$ with rank greater than r and denote $X = \sum_{i=1}^{r+k} \sigma_i u_i v_i^\top$ be its SVD decomposition, where the singular values are ordered decreasingly: $\sigma_1 \geq \sigma_2 \geq \cdots \sigma_{r+k}$. Suppose that the orthogonal projection $P_{\mathcal{M}_r}(X)$ of X onto $\mathcal{M}_r$ is uniquely defined, that is $\sigma_r(X) > \sigma_{r+1}(X)$. Then $P_{\mathcal{M}_r}$, the SVD truncation operator of order r, is differentiable at X and the differential in a direction Y is given by the formula*

$$\begin{aligned}\nabla_Y P_{\mathcal{M}_r}(X) =& \mathrm{P}_{T_{\mathcal{M}_r}(P_{\mathcal{M}_r}(X))}(Y) \\ &+ \sum_{\substack{1\leq i\leq r\\ 1\leq j\leq k}} \left[\frac{\sigma_{r+j}}{\sigma_i - \sigma_{r+j}}\langle Y, \Phi^+_{i,r+j}\rangle \Phi^+_{i,r+j} - \frac{\sigma_{r+j}}{\sigma_i + \sigma_{r+j}}\langle Y, \Phi^-_{i,r+j}\rangle \Phi^-_{i,r+j}\right] \quad (4.2.3)\end{aligned}$$

where

$$\Phi^{\pm}_{i,r+j} = \frac{1}{\sqrt{2}}(u_{r+j}v_i^\top \pm u_i v_{r+j}^\top)$$

are the principal directions corresponding to the principal curvature of the manifold of rank-r matrices.

We are now ready to establish the convergence of Algorithm 4 under a sufficient condition.

Theorem 4.2.6. *Assume $T_{\mathcal{A}_\Omega}(M) \cap T_{\mathcal{M}_r}(M) = \{0\}$. Then Algorithm 4 converges to M locally at a linear rate, i.e. there exists a neighborhood V around M such that if $X_0 \in V$, then there exists a positive constant $c < 1$ such that*

$$\|X_k - M\| < c^k \|X_0 - M\|, \tag{4.2.4}$$

where X_k is the kth iteration from Algorithm 1.

Proof. For notational convenience, let

$$f(X) := P_{\mathcal{A}_\Omega}(P_{\mathcal{M}_r}(X)).$$

Note that $\mathcal{A}_\Omega$ is an affine space, the gradient $\nabla P_{\mathcal{A}_\Omega}$ of the projection $P_{\mathcal{A}_\Omega}$ is the projection onto the tangent space of the affine space $\mathcal{A}_\Omega$. By Lemma 4.2.4 and chain rule, we have

$$(\nabla f)(X) = P_{T_{\mathcal{A}_\Omega}(M)}(\mathrm{P}_{T_{\mathcal{M}_r}(M)}(X)).$$

as $T_{\mathcal{A}_\Omega}(M) = T_{\mathcal{A}_\Omega}(X)$ for all X.

Now from the definition of differentiability of f at M, we have

$$\lim_{X \to M} \frac{\|f(X) - f(M) - \nabla f(M) \cdot (X - M)\|}{\|X - M\|} = 0.$$

Hence, there exist an open ball V, say a ball $V = B_{r_0}(M)$ centered at M of radius r_0 around M such that, for all $X \in V$

$$\frac{\|f(X) - f(M) - \nabla f(M) \cdot (X - M)\|}{\|X - M\|} < \epsilon,$$

where $\epsilon = \frac{1 - \|\nabla f\|}{2} > 0$.

Using our hypothesis and Lemma 4.2.3, we have

$$\|\nabla f(M)\| = \left\| P_{T_{\mathcal{A}_\Omega}(M)} \, \mathrm{P}_{T_{\mathcal{M}_r}(M)} \right\| < 1$$

. Therefore, for all $X \in V$, we use $M = f(M)$ to have

$$\begin{aligned}
\|f(X) - M\| &= \|f(X) - f(M)\| \\
&\leq \|f(X) - f(M) - \nabla f \cdot (X - M)\| + \|\nabla f(M) \cdot (X - M)\| \\
&< \epsilon \|X - M\| + \|\nabla f(M)\| \, \|(X - M)\| \\
&= (\epsilon + \|\nabla f(M)\|) \, \|X - M\| \\
&\leq \frac{1 + \|\nabla f(M)\|}{2} \|X - M\| .
\end{aligned}$$

where $\frac{1+\|\nabla f(M)\|}{2} < 1$ since $\|\nabla f(M)\| < 1$ as discussed above.

Setting $c = \frac{1+\|\nabla f(M)\|}{2} < 1$, we can rewrite the above inequality as follows:

$$\|f(X) - M\| < c \, \|X - M\| \text{ for all } X \in V. \tag{4.2.5}$$

Hence, if $X_k \in V = B_{r_0}(M)$, we use $X_{k+1} = f(X_k)$ to have

$$\|X_{k+1} - M\| = \|f(X_k) - M\| < c \, \|X_k - M\| \leq r_0$$

which implies $X_{k+1} \in V = B_{r_0}(M)$. So, if the initial guess $X_0 \in V$, we have, by induction,

$$X_k \in V \text{ for all } k$$

and

$$\|X_k - M\| \leq c^k \, \|X_0 - M\| .$$

We have thus completed the proof. □

We will now derive certain equivalent conditions for hypothesis $T_{\mathcal{A}_\Omega}(M) \cap T_{\mathcal{M}_r}(M) = \{0\}$ of the above theorem. Let us recall the following fact well known in the literature. For the sake of completeness, we include a proof.

Lemma 4.2.7. *The tangent space $T_{\mathcal{M}_r}(M)$ has an explicit description as follows:*

$$T_{\mathcal{M}_r}(M) = \left\{ XM + MY \mid X \in \mathbb{R}^{n\times n} \text{ and } Y \in \mathbb{R}^{n\times n} \right\}. \tag{4.2.6}$$

Proof. First recall that the tangent space $T_{\mathcal{M}_r}(M)$ to a manifold $\mathcal{M}_r$ at a point M is the linear space spanned by all the tangent vectors at 0 to smooth curves $\gamma : \mathbb{R} \to \mathcal{M}_r$ such that $\gamma(0) = M$.

Now let $M \in \mathcal{M}_r$ be a $n \times n$ matrix of rank r. We can write $M = X_0 Y_0^\top$ where $X_0, Y_0 \in \mathbb{R}^{n\times r}$ and both X_0 and Y_0 have full column rank. This is possible because M has exactly rank r.

Let $\gamma(t) = X(t)Y(t)^\top$ be a smooth curve such that $X(0) = X_0$ and $Y(0) = Y_0$. Hence, $\gamma(0) = X_0 Y_0^\top = M$. Since X_0 and Y_0 have full column rank, X_0 and Y_0 have a $r \times r$ minor that does not vanish. Since nonvanishing of a minor is an open condition, there exist an open neighbourhood of M to which if we restrict the curve γ, we can assume $X(t)$ and $Y(t)$ have full column rank. In other words, we can assume, without loss of generality, that $X(t)^\top X(t)$ and $Y(t)^\top Y(t)$ are invertible $r \times r$ matrices for all t.

Using product rule, we obtain

$$\begin{aligned}
\dot{\gamma}(0) &= \dot{X}(0)Y(0)^\top + X(0)\dot{Y}(0)^\top \\
&= \dot{X}(0)Y_0^\top + X_0\dot{Y}(0)^\top \\
&= \dot{X}(0)(X_0^\top X_0)^{-1}(X_0^\top X_0)Y_0^\top + X_0(Y_0^\top Y)(Y_0^\top Y_0)^{-1}\dot{Y}(0)^\top \\
&= \left(\dot{X}(0)(X_0^\top X_0)^{-1}X_0^\top\right)(X_0Y_0^\top) + (X_0Y_0^\top)\left(Y_0(Y_0^\top Y)^{-1}\dot{Y}(0)^\top\right) \\
&= \left(\dot{X}(0)(X_0^\top X_0)^{-1}X_0^\top\right)M + M\left(Y_0(Y_0^\top Y_0)^{-1}\dot{Y}(0)^\top\right) \\
&\in \left\{XM + MY \mid X \in \mathbb{R}^{n\times n} \text{ and } Y \in \mathbb{R}^{n\times n}\right\}.
\end{aligned}$$

Now, to prove the reverse inclusion, let

$$AM + MB \in \left\{XM + MY \mid X \in \mathbb{R}^{n\times n} \text{ and } Y \in \mathbb{R}^{n\times n}\right\}$$

. Consider the smooth curve $\gamma(t) = X(t)Y(t)^\top$ defined by

$$X(t) = t(AX_0) + X_0$$

and

$$Y(t) = t\left((Y_0B)^\top\right) + Y_0.$$

An easy computation shows that $\gamma(0) = M$ and $\dot{\gamma}(0) = AM + MB$. Hence, we get the equality

$$T_{\mathcal{M}_r}(M) = \left\{XM + MY \mid X \in \mathbb{R}^{n\times n} \text{ and } Y \in \mathbb{R}^{n\times n}\right\}$$

This completes the proof. □

One can consider $T_{\mathcal{M}_r}(M)$ as a linear space in $\mathbb{R}^{n^2}$ by rewriting it as

$$T_{\mathcal{M}_r}(M) \cong \text{Range}(T_M) = \left\{ T_M \cdot \begin{bmatrix} (X^1)^\top \\ \vdots \\ (X^n)^\top \\ Y_1 \\ \vdots \\ Y_n \end{bmatrix} \mid X \in \mathbb{R}^{n\times n} \text{ and } Y \in \mathbb{R}^{n\times n} \right\}$$

where T_M is a block matrix of size $n^2 \times 2n^2$ consisting of $2n^3$ blocks of size $1 \times n$, X^i and X_j denotes the i^{th} row and j^{th} column of a matrix X respectively.

Explicitly, T_M would take the form

$$T_M = \left[\begin{array}{c|c|c|c|c|c|c|c|c|c|c|c|c|c|c} M_1^\top & 0 & \cdots & 0 & \cdots & \cdots & \cdots & 0 & M^1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ \hline \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \hline 0 & 0 & \cdots & 0 & M_j^\top & 0 & \cdots & 0 & 0 & \cdots & 0 & M^i & 0 & \cdots & 0 \\ \hline \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{array}\right]$$

where the each row corresponds to each index in $\{1, 2, \cdots, n\} \times \{1, 2, \cdots, n\}$.

Let T_M^Ω and $T_M^{\Omega^c}$ denote the matrix obtained from T_M by choosing the rows corresponding to Ω and Ω^c, respectively.

Example 4.2.8. Suppose $M = \begin{bmatrix} 1 & 4 \\ 2 & 8 \end{bmatrix}$ and $\Omega = \{(1,2),(2,1)\}$. Then

$$T_M^\Omega = \left[\begin{array}{cc|cc|cc|cc} 4 & 8 & 0 & 0 & 0 & 0 & 1 & 4 \\ \hline 0 & 0 & 1 & 2 & 2 & 8 & 0 & 0 \end{array}\right]$$

$$T_M^{\Omega^c} = \left[\begin{array}{cc|cc|cc|cc} 1 & 2 & 0 & 0 & 1 & 4 & 0 & 0 \\ \hline 0 & 0 & 4 & 8 & 0 & 0 & 2 & 8 \end{array}\right]$$

and

$$T_M = \left[\begin{array}{cc|cc|cc|cc} 4 & 8 & 0 & 0 & 0 & 0 & 1 & 4 \\ \hline 0 & 0 & 1 & 2 & 2 & 8 & 0 & 0 \\ \hline 1 & 2 & 0 & 0 & 1 & 4 & 0 & 0 \\ \hline 0 & 0 & 4 & 8 & 0 & 0 & 2 & 8 \end{array}\right]$$

Example 4.2.9. Suppose $M = \begin{bmatrix} -3 & -1 & -4 \\ 9 & 3 & 12 \\ 6 & 2 & 8 \end{bmatrix}$ and $\Omega = \{(1,1),(1,3),(2,2),(3,1)\}$,

Then

$$T_M^{\Omega} = \left[\begin{array}{ccc|ccc|ccc|ccc|ccc|ccc} -3 & 9 & 6 & 0 & 0 & 0 & 0 & 0 & 0 & -3 & -1 & -4 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline -4 & 12 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3 & -1 & -4 \\ \hline 0 & 0 & 0 & -1 & 3 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 9 & 3 & 12 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & -3 & 9 & 6 & 6 & 2 & 8 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right]$$

$$T_M^{\Omega^c} = \left[\begin{array}{ccc|ccc|ccc|ccc|ccc|ccc} -1 & 3 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3 & -1 & -4 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & -3 & 9 & 6 & 0 & 0 & 0 & 9 & 3 & 12 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & -4 & 12 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9 & 3 & 12 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & -1 & 3 & 2 & 0 & 0 & 0 & 6 & 2 & 8 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & -4 & 12 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 6 & 2 & 8 \end{array}\right]$$

and

$$T_M = \left[\begin{array}{ccc|ccc|ccc|ccc|ccc|ccc} -3 & 9 & 6 & 0 & 0 & 0 & 0 & 0 & 0 & -3 & -1 & -4 & 0 & 0 & 0 & 0 & 0 & 0 \\ -4 & 12 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3 & -1 & -4 \\ \hline 0 & 0 & 0 & -1 & 3 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 9 & 3 & 12 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & -3 & 9 & 6 & 6 & 2 & 8 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline -1 & 3 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3 & -1 & -4 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & -3 & 9 & 6 & 0 & 0 & 0 & 9 & 3 & 12 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & -4 & 12 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9 & 3 & 12 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & -1 & 3 & 2 & 0 & 0 & 0 & 6 & 2 & 8 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & -4 & 12 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 6 & 2 & 8 \end{array}\right]$$

Next we need

Lemma 4.2.10. *The tangent space $T_{\mathcal{A}_\Omega}(M)$ at M can be given explicitly as follows.*

$$T_{\mathcal{A}_\Omega}(M) = \left\{X \in \mathbb{R}^{n\times n} \mid P_\Omega(X) = 0\right\}. \tag{4.2.7}$$

Proof. Recall that

$$\mathcal{A}_M := \{X \mid P_\Omega(X - M) = 0\}.$$

Since $P_\Omega(X - M) = P_\Omega(X) - P_\Omega(M) = P_\Omega(X) - P_\Omega(P_\Omega(M)) = P_\Omega(X - P_\Omega(M))$, we get that the set $\mathcal{A}_\Omega$ is a translation of the linear space $\{X \in \mathbb{R}^{n\times n} \mid P_\Omega(X) = 0\}$ by $P_\Omega(M)$, i.e.

$$\mathcal{A}_\Omega = \left\{X \in \mathbb{R}^{n\times n} \mid P_\Omega(X) = 0\right\} + P_\Omega(M)$$

Hence we have that the tangent space of $\mathcal{A}_\Omega$ at M is equal to the tangent space of the vector space $\{X \in \mathbb{R}^{n\times n} \mid P_\Omega(X) = 0\}$ at $M - P_\Omega(M)$. But the tangent space of a vector space at any point is the vector space itself. Hence the result follows. □

With the above preparation, we have following proposition:

Proposition 4.2.11. *The following statements are equivalent:*

1. $T_{\mathcal{A}_\Omega}(M) \cap T_{\mathcal{M}_r}(M) = \{0\}$

2. $\text{Rowspace}\left(T_M^{\Omega^c}\right) \subseteq \text{Rowspace}\left(T_M^{\Omega}\right)$

3. $\text{Rank}\left(T_M^{\Omega}\right) = 2nr - r^2$, *where* $r = \text{Rank}(M)$

4. *The matrix* $V^\Omega(M)$ *of size* $|\Omega| \times |\Omega|$ *defined by*

$$V^{\Omega}_{(i_1,j_1),(i_2,j_2)}(M) = \begin{cases} 0 & i_1 \neq i_2 \text{ and } j_1 \neq j_2 \\ \langle M_{j_1}, M_{j_2} \rangle & i_1 = i_2 \text{ and } j_1 \neq j_2 \\ \langle M^{i_1}, M^{i_2} \rangle & i_1 \neq i_2 \text{ and } j_1 = j_2 \\ \|M^{i_1}\|^2 + \|M_{j_1}\|^2 & i_1 = i_2 \text{ and } j_1 = j_2 \end{cases} \tag{4.2.8}$$

has rank $2nr - r^2$, *where* M_j *stands for the* j*th column and* M^i *for the* i*th row of* M.

Proof. (1) $\iff$ (2) Note that the elements of $T_{\mathcal{A}_\Omega}(M) \cap T_{\mathcal{M}_r}(M)$ consists of matrices of the form $XM + MY$ such that the elements in positions Ω is zero by Lemmas 4.2.7 and 4.2.10. Hence, observing that $T_{\mathcal{M}_r}(M)$ can be considered as the range of T_M and that the rows of T_M correspond to each index in $\{1, 2, \cdots, n\} \times \{1, 2, \cdots, n\}$, we can conclude that $T_{\mathcal{A}_\Omega}(M) \cap T_{\mathcal{M}_r}(M) = \{0\}$ if and only if

$$\text{NullSpace}\left(T_M^{\Omega}\right) \subseteq \text{NullSpace}\left(T_M^{\Omega^c}\right)$$

which is equivalent to

$$\text{NullSpace}\left(T_M^{\Omega}\right)^{\perp} \supseteq \text{NullSpace}\left(T_M^{\Omega^c}\right)^{\perp}$$

The result follows by noting that

$$\text{Rowspace}\left(T_M^{\Omega^c}\right) = \text{NullSpace}\left(T_M^{\Omega^c}\right)^{\perp}$$

and

$$\text{Rowspace}\left(T_M^{\Omega}\right) = \text{NullSpace}\left(T_M^{\Omega}\right)^{\perp}.$$

(2) $\iff$ (3) We begin by recalling that dimension of a tangent space is equal to dimension of the manifold. So, $\dim(T_{\mathcal{M}_r}(M)) = 2nr - r^2$. Now

$$2nr - r^2 = \dim(T_{\mathcal{M}}(M)) = \dim(Range(T_M)) = \text{Rank}(T_M) = \text{Rank}\left(\text{Rowspace}\left(T_M\right)\right).$$

Now the equivalence (2) $\iff$ (3) follows by recalling that T_M^{Ω} and $T_M^{\Omega^c}$ were obtained from T_M by choosing the rows corresponding to Ω and Ω^c, respectively

(3) $\iff$ (4) The equivalence follows from fact that $V^{\Omega}(M) = T_M^{\Omega}\left(T_M^{\Omega}\right)^{\top}$. Hence $\text{Rank}(V^{\Omega}(M)) = \text{Rank}(T_M^{\Omega})$. □

Remark 4.2.12. Note that the size of the matrix T_M^{Ω} is $|\Omega| \times 2n^2$ which is considerably larger than the size of the matrix $V^{\Omega}(M)$ which has size $|\Omega| \times |\Omega|$. Therefore, since rank computation is a memory intensive process, it is much efficient to check the statement (4) of above proposition as compared to statement (3).
In general, the rank of $V^{\Omega}(M)$ is less than or equal to $2nr - r^2$. The equality occurs when the tangent spaces intersect trivially.

The following example is an illustration of the linear convergence of the error when the condition $T_{\mathcal{A}_{\Omega}}(M) \cap T_{\mathcal{M}_r}(M) = \{0\}$ is satisfied.

Example 4.2.13. We find a 15×15 matrix M of rank 2 which has 28% of entries missing. A straightforward computation shows that $\text{Rank}(V^{\Omega}(M)) = 2nr - r^2$. Hence, M satisfies the condition $T_{\mathcal{A}_\Omega}(M) \cap T_{\mathcal{M}_r}(M) = \{0\}$. Hence, by Theorems 4.2.11 and 4.2.6, we know that Algorithm 4 will converge in a linear fashion.

$$M =$$

$$\begin{bmatrix}
0.3474 & 0.0897 & 0.3971 & 0.4644 & 0.4168 & 0.7576 & 0.8206 & 0.8161 & 0.3279 & 0.3851 & 0.0825 & 0.4742 & 0.7684 & 0.6113 & 0.3832 \\
0.1502 & 0.0414 & 0.2196 & 0.2450 & 0.2731 & 0.4415 & 0.4293 & 0.4358 & 0.1859 & 0.1574 & 0.0493 & 0.2386 & 0.4502 & 0.3087 & 0.1999 \\
0.3853 & 0.1079 & 0.5912 & 0.6542 & 0.7544 & 1.1986 & 1.1445 & 1.1660 & 0.5024 & 0.3985 & 0.1343 & 0.6315 & 1.2231 & 0.8176 & 0.5325 \\
0.2174 & 0.0577 & 0.2760 & 0.3160 & 0.3141 & 0.5394 & 0.5562 & 0.5582 & 0.2305 & 0.2358 & 0.0594 & 0.3160 & 0.5484 & 0.4080 & 0.2594 \\
0.2124 & 0.0493 & 0.1453 & 0.1940 & 0.0662 & 0.2317 & 0.3503 & 0.3303 & 0.1109 & 0.2539 & 0.0228 & 0.2216 & 0.2302 & 0.2835 & 0.1647 \\
0.1026 & 0.0238 & 0.0701 & 0.0936 & 0.0318 & 0.1117 & 0.1691 & 0.1594 & 0.0535 & 0.1227 & 0.0110 & 0.1070 & 0.1110 & 0.1368 & 0.0795 \\
0.2429 & 0.0600 & 0.2290 & 0.2798 & 0.1972 & 0.4141 & 0.4982 & 0.4864 & 0.1846 & 0.2785 & 0.0439 & 0.2974 & 0.4176 & 0.3823 & 0.2332 \\
0.3848 & 0.0895 & 0.2658 & 0.3538 & 0.1248 & 0.4257 & 0.6385 & 0.6028 & 0.2032 & 0.4595 & 0.0421 & 0.4033 & 0.4232 & 0.5159 & 0.3002 \\
0.3698 & 0.1015 & 0.5311 & 0.5943 & 0.6536 & 1.0640 & 1.0419 & 1.0562 & 0.4488 & 0.3894 & 0.1186 & 0.5806 & 1.0845 & 0.7511 & 0.4852 \\
0.3631 & 0.0880 & 0.3138 & 0.3919 & 0.2395 & 0.5511 & 0.7003 & 0.6776 & 0.2496 & 0.4217 & 0.0575 & 0.4246 & 0.5540 & 0.5451 & 0.3282 \\
0.2081 & 0.0480 & 0.1369 & 0.1850 & 0.0542 & 0.2139 & 0.3347 & 0.3141 & 0.1036 & 0.2498 & 0.0208 & 0.2133 & 0.2120 & 0.2726 & 0.1575 \\
0.5203 & 0.1334 & 0.5792 & 0.6812 & 0.5942 & 1.0977 & 1.2049 & 1.1953 & 0.4769 & 0.5797 & 0.1192 & 0.6992 & 1.1126 & 0.9011 & 0.5627 \\
0.4871 & 0.1231 & 0.5111 & 0.6090 & 0.4961 & 0.9538 & 1.0797 & 1.0652 & 0.4178 & 0.5487 & 0.1028 & 0.6328 & 0.9651 & 0.8148 & 0.5047 \\
0.0287 & 0.0122 & 0.1183 & 0.1173 & 0.2001 & 0.2658 & 0.2007 & 0.2154 & 0.1057 & 0.0156 & 0.0311 & 0.0991 & 0.2738 & 0.1297 & 0.0927 \\
0.2602 & 0.0617 & 0.1997 & 0.2577 & 0.1230 & 0.3353 & 0.4628 & 0.4422 & 0.1558 & 0.3070 & 0.0341 & 0.2867 & 0.3353 & 0.3673 & 0.2173
\end{bmatrix}$$

and

$$M_\Omega =$$

$$\begin{bmatrix}
0 & 0.0897 & 0.3971 & 0 & 0.4168 & 0.7576 & 0.8206 & 0 & 0.3279 & 0.3851 & 0.0825 & 0 & 0.7684 & 0.6113 & 0.3832 \\
0.1502 & 0 & 0 & 0.2450 & 0 & 0.4415 & 0.4293 & 0.4358 & 0 & 0.1574 & 0 & 0.2386 & 0.4502 & 0 & 0.1999 \\
0.3853 & 0.1079 & 0.5912 & 0.6542 & 0.7544 & 1.1986 & 0 & 1.1660 & 0.5024 & 0 & 0.1343 & 0 & 1.2231 & 0.8176 & 0.5325 \\
0.2174 & 0.0577 & 0.2760 & 0 & 0.3141 & 0 & 0 & 0.5582 & 0.2305 & 0 & 0.0594 & 0.3160 & 0.5484 & 0.4080 & 0 \\
0 & 0 & 0.1453 & 0.1940 & 0.0662 & 0.2317 & 0.3503 & 0 & 0.1109 & 0 & 0 & 0.2216 & 0 & 0 & 0.1647 \\
0.1026 & 0.0238 & 0.0701 & 0.0936 & 0.0318 & 0.1117 & 0.1691 & 0.1594 & 0.0535 & 0.1227 & 0.0110 & 0 & 0.1110 & 0.1368 & 0.0795 \\
0.2429 & 0.0600 & 0.2290 & 0.2798 & 0 & 0 & 0.4982 & 0 & 0.1846 & 0 & 0 & 0.2974 & 0.4176 & 0.3823 & 0.2332 \\
0 & 0 & 0 & 0.3538 & 0.1248 & 0 & 0 & 0 & 0 & 0.4595 & 0 & 0 & 0 & 0.5159 & 0.3002 \\
0 & 0.1015 & 0.5311 & 0.5943 & 0.6536 & 1.0640 & 1.0419 & 1.0562 & 0.4488 & 0.3894 & 0 & 0.5806 & 1.0845 & 0 & 0 \\
0.3631 & 0.0880 & 0.3138 & 0 & 0.2395 & 0.5511 & 0.7003 & 0.6776 & 0.2496 & 0 & 0.0575 & 0.4246 & 0.5540 & 0.5451 & 0 \\
0.2081 & 0.0480 & 0.1369 & 0.1850 & 0.0542 & 0.2139 & 0 & 0 & 0.1036 & 0.2498 & 0.0208 & 0.2133 & 0 & 0.2726 & 0.1575 \\
0 & 0.1334 & 0.5792 & 0 & 0.5942 & 1.0977 & 1.2049 & 1.1953 & 0 & 0 & 0.1192 & 0.6992 & 0 & 0.9011 & 0.5627 \\
0.4871 & 0 & 0.5111 & 0.6090 & 0 & 0.9538 & 1.0797 & 0 & 0 & 0.5487 & 0.1028 & 0.6328 & 0.9651 & 0.8148 & 0.5047 \\
0.0287 & 0.0122 & 0.1183 & 0.1173 & 0.2001 & 0.2658 & 0.2007 & 0.2154 & 0 & 0.0156 & 0 & 0.0991 & 0.2738 & 0.1297 & 0.0927 \\
0.2602 & 0.0617 & 0.1997 & 0.2577 & 0.1230 & 0.3353 & 0.4628 & 0 & 0.1558 & 0.3070 & 0.0341 & 0.2867 & 0.3353 & 0.3673 & 0.2173
\end{bmatrix}$$

where 0 stands for the unknown entries.

Notice from the graph in Figure 4.1 that as the iterations progress, the X_k would eventually land in a neighborhood of M where the convergence become linear.

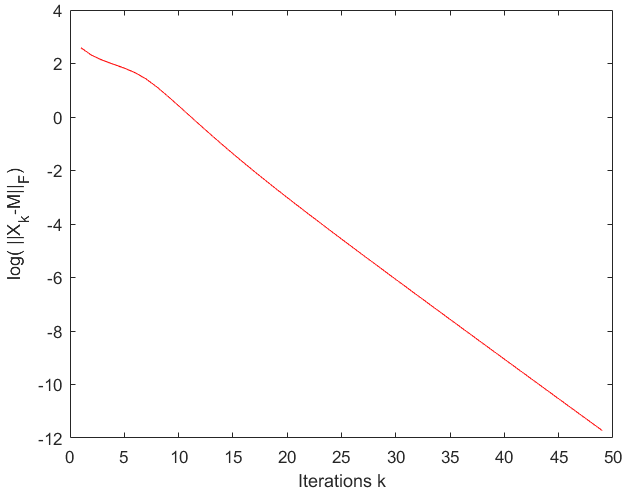

Figure 4.1: Linear Convergence of the Iterations from Algorithm 4

The construction of T_M^Ω enables us to choose Ω such that T_M^Ω is of full rank. We end with this subsection with the following

Corollary 4.2.14. *Given M with rank r, for any integer m such that $2nr - r^2 \leq m \leq n^2$, there exists a subset Ω with $m = |\Omega|$ such that V^Ω is of full rank, equivalently $T_{\mathcal{A}_\Omega}(M) \cap T_{\mathcal{M}_r}(M) = \{0\}$ and Algorithm 4 can find M in a linear fashion for a good initial guess.*

Proof. We mainly choose Ω such that the corresponding rows of T_M which form T_M^Ω of rank $2nr - r^2$. Then Theorems 4.2.11 and 4.2.6 can be applied. □

4.2.2 Convergence of Algorithm 4 When $r_g \neq$ Rank(M)

In this subsection, we show that the algorithm does converge under certain reasonable assumption irrespective of whether our guessed rank r_g is same as the rank r of matrix M or not. We begin with two trivial results.

Lemma 4.2.15. *Let Y_k and X_{k+1} be the matrices we obtain in the step 1 and step 2 of the k^{th} iteration of Algorithm 4. Then*

$$X_{k+1} = \begin{cases} (Y_k)_{i,j} & \text{if } (i,j) \notin \Omega \\ M_{i,j} & \text{Otherwise.} \end{cases}$$

That is, X_{k+1} is the orthogonal projection of Y_k onto $\mathcal{A}_\Omega$.

Lemma 4.2.16. *Let $X_{k+1} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ be the standard singular value decomposition with $\mathbf{\Sigma} = \text{diag}\{\sigma_1, \cdots, \sigma_n\}$. Then*

$$Y_{k+1} = \mathbf{U}\tilde{\mathbf{\Sigma}}\mathbf{V}^\top,$$

where $\tilde{\mathbf{\Sigma}} = diag\{\sigma_1, \cdots, \sigma_{r_g}, 0, \cdots, 0\}$.

Also Y_{k+1} is the orthogonal projection of X_{k+1} onto $\mathcal{M}_{r_g}$.

$\mathcal{M}_{r_g}$, the collection of $n \times n$ real (complex) matrices of rank r_g, forms a quasi-affine real (complex) variety and is a manifold of real (complex) dimension $r_g(2n - r_g)$.

It is well known that Y_k, obtained from X_k by SVD truncation, is the orthogonal projection of X_k onto $\mathcal{M}_{r_g}$. Hence, $X_k - Y_k$ must be orthogonal to the tangent space of $\mathcal{M}_{r_g}$ at Y_k. Recall from earlier section that tangent space of $\mathcal{M}_{r_g}$ at the point X is given by

$$T_{\mathcal{M}_{r_g}}(X) = \left\{AX + XB, A \in \mathbb{R}^{m\times m}, B \in \mathbb{R}^{n\times n}\right\}$$

Lemma 4.2.17. *Y_k satisfies:*

$$\langle AY_k + Y_kB, X_k - Y_k\rangle = 0 \text{ for all } k, A \in \mathbb{R}^{n\times n} \text{ and } B \in \mathbb{R}^{n\times n}.$$

Proof. Let $X_k = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ and $Y_k = \mathbf{U}\tilde{\mathbf{\Sigma}}\mathbf{V}^\top$, where $\Sigma = \text{diag}\{\sigma_1, \cdots, \sigma_n\}$ and $\tilde{\Sigma} = \text{diag}\{\sigma_1, \cdots, \sigma_{r_g}, 0, \cdots, 0\}$ be the singular value decompositions of X_k and Y_k respec-

tively.

$$\begin{aligned}
\langle AY_k + Y_kB, X_k - Y_k\rangle &= \text{Trace}\left((X_k - Y_k)^\top (AY_k + Y_kB)\right) \\
&= \text{Trace}\left(\mathbf{V}(\mathbf{\Sigma} - \tilde{\mathbf{\Sigma}})\mathbf{U}^\top AY_k\right) + \text{Trace}\left(\mathbf{V}(\mathbf{\Sigma} - \tilde{\mathbf{\Sigma}})\mathbf{U}^\top Y_kB\right) \\
&= \text{Trace}\left(\mathbf{V}(\mathbf{\Sigma} - \tilde{\mathbf{\Sigma}})\mathbf{U}^\top A\mathbf{U}\tilde{\mathbf{\Sigma}}\mathbf{V}^\top\right) + \text{Trace}\left(\mathbf{V}(\mathbf{\Sigma} - \tilde{\mathbf{\Sigma}})\mathbf{U}^\top\mathbf{U}\tilde{\mathbf{\Sigma}}\mathbf{V}^\top B\right) \\
&= \text{Trace}\left(\mathbf{V}^\top\mathbf{V}(\mathbf{\Sigma} - \tilde{\mathbf{\Sigma}})\mathbf{U}^\top A\mathbf{U}\tilde{\mathbf{\Sigma}}\right) + \text{Trace}\left(\mathbf{V}(\mathbf{\Sigma} - \tilde{\mathbf{\Sigma}})\tilde{\mathbf{\Sigma}}\mathbf{V}^\top B\right) \\
&= \text{Trace}\left(\tilde{\mathbf{\Sigma}}(\mathbf{\Sigma} - \tilde{\mathbf{\Sigma}})\mathbf{U}^\top A\mathbf{U}\right) + \text{Trace}\left(\mathbf{V}(\mathbf{\Sigma} - \tilde{\mathbf{\Sigma}})\tilde{\mathbf{\Sigma}}\mathbf{V}^\top B\right) \\
&= 0.
\end{aligned}$$

The last step uses the fact that $\tilde{\Sigma}(\Sigma - \tilde{\Sigma}) = (\Sigma - \tilde{\Sigma})\tilde{\Sigma} = 0$. □

From the definitions, it follows that

$$\|X_k - Y_k\| \geq \|X_{k+1} - Y_k\| \geq \|X_{k+1} - Y_{k+1}\| \text{ for all } k. \tag{4.2.9}$$

From equation (4.2.9), we observe that $\|X_k - Y_k\|$ is a non-increasing sequence bounded below by 0, it thus converges to its infimum. Thus, we have

Lemma 4.2.18. *The sequence*

$$\|X_k - Y_k\|$$

converges.

So let

$$L = \lim_k \|X_k - Y_k\|^2 . \tag{4.2.10}$$

Next we have

Lemma 4.2.19.

$$\|X_{k+1} - X_k\|^2 + \|X_{k+1} - Y_k\|^2 = \|X_k - Y_k\|^2 \tag{4.2.11}$$

Proof. The result (4.2.11) follows from Lemmas 4.2.15 and 4.2.16. In fact we have used the fact $\langle X_{k+1} - X_k, X_{k+1} - Y_k \rangle = 0$ to have (4.2.11). □

Lemma 4.2.20. *The series*

$$\sum_{k=1}^{\infty} \|X_{k+1} - X_k\|^2$$

converges. In particular

$$\|X_{k+1} - X_k\| \to 0.$$

Proof. We use (4.2.11) and (4.2.9) to get

$$\|X_k - Y_k\|^2 \geq \|X_{k+1} - X_k\|^2 + \|X_{k+1} - Y_{k+1}\|^2$$

summing both sides from $k = 1$ to n we get

$$\sum_{k=1}^{n} \|X_k - Y_k\|^2 \geq \sum_{k=1}^{n} \|X_{k+1} - X_k\|^2 + \sum_{k=1}^{n} \|X_{k+1} - Y_{k+1}\|^2 .$$

From which it follows that

$$\|X_1 - Y_1\|^2 \geq \|X_n - Y_n\|^2 + \sum_{k=1}^{n} \|X_{k+1} - X_k\|^2 \geq \sum_{k=1}^{n} \|X_{k+1} - X_k\|^2$$

Thus the partial sums of the $\sum_{k=1}^{\infty} \|X_{k+1} - X_k\|^2$ forms an non-decreasing sequence bounded from above. The result follows immediately. □

Lemma 4.2.21. *The series*

$$\sum_{k=1}^{\infty} \|(X_k - Y_k)_{\Omega^c}\|^2$$

converges. In particular

$$\|(X_k - Y_k)_{\Omega^c}\| \to 0.$$

Proof.

$$\begin{aligned}\|X_{k+1} - X_k\|^2 &= \|(X_{k+1} - X_k)_\Omega\|^2 + \|(X_{k+1} - X_k)_{\Omega^c}\|^2 \\ &= \|(X_{k+1})_\Omega - (X_k)_\Omega\|^2 + \|(X_{k+1})_{\Omega^c} - (X_k)_{\Omega^c}\|^2 \\ &= \|M_\Omega - M_\Omega\|^2 + \|(X_{k+1})_{\Omega^c} - (X_k)_{\Omega^c}\|^2 \\ &= \|(X_{k+1})_{\Omega^c} - (X_k)_{\Omega^c}\|^2 .\end{aligned}$$

Now noting that $(X_{k+1})_{\Omega^c} = (Y_k)_{\Omega^c}$ the above equation simplifies

$$\|X_{k+1} - X_k\|^2 = \|(Y_k)_{\Omega^c} - (X_k)_{\Omega^c}\|^2$$

Summing both sides and using Lemma 4.2.20, the result follows. □

With the above preparation, we are finally ready to establish the main convergence result in this subsection.

Theorem 4.2.22. *There exist a subsequence of $(Y_k)_\Omega$ that converges, say without loss of generality, $(Y_k)_\Omega \to y^\star$. Assume that there are only finitely many rank-r matrices Y such that $P_\Omega(Y) = y^\star$. Then there exist subsequences X_{k_j} and Y_{k_j} which converge, say $Y^\star$ and $X^\star$ such that*

$$X_{k_j} \to X^\star \text{ and } Y_{k_j} \to Y^\star.$$

Furthermore, we have $X^\star|_{\Omega^c} = Y^\star|_{\Omega^c}$ and

$$X^\star \in \mathcal{A}_\Omega \text{ and } \operatorname{rank}(Y^\star) \leq r_g. \tag{4.2.12}$$

Proof. By Lemma 4.2.18, $\|X_k - Y_k\| \to \sqrt{L}$, we see that the sequence $\|M_\Omega - (Y_k)_\Omega\| = \|(X_k)_\Omega - (Y_k)_\Omega\| \leq 2\sqrt{L}$ for all $k \geq 1$ without loss of generality. It follows that $\|(Y_k)_\Omega\|, k \geq 1$ are a bounded sequence and hence, $\|(Y_k)_\Omega\| \leq C_1 < \infty$ for a positive

constant C_1 and $(Y_k)_\Omega \to y^*$ without loss of generality

Under the assumption that there are finitely many $Y \in \overline{\mathcal{M}_{r_g}}$ such that $P_\Omega(Y) = y^*$, we next claim that $Y_k, k \geq 1$ are bounded. Indeed, for any matrix $Y \in \overline{\mathcal{M}_{r_g}}$, the set of matrices with rank $\leq r_g$, if we write the entries in Y_{Ω^c} as variables, say $\mathbf{x} \in \mathbb{R}^{n^2-m}$ while the entries $Y|_\Omega$ are known, the determinant of any $(r+1) \times (r+1)$ minor of Y will be zero and is a polynomial function of variables $\mathbf{x}$ with coefficients based on the known entries $Y|_\Omega$. Thus, vanishing of all $(r+1) \times (r+1)$ minors would form a set of $(\binom{n}{r_g+1})^2$ polynomial equations with variables $\mathbf{x}$ and coefficients from entries in $Y|_\Omega$. By our assumption, this set of polynomial equations have *finitely* many solutions when the coeffficients of the system is derived from the Ω entries of $y^\star$. Since the zeros of these polynomial equations are continuously dependent on the coefficients of polynomial functions, we see that there are finitely many solutions to the polynomial system when coefficients are derived from $(Y_k)_\Omega$ that are sufficiently close to $y^\star$. We can bound the zeros by using the coefficients. More precisely, these polynomial equations can be reduced to a triangular system (cf. Chen and Moreno Maza [18]), that is, writing $\mathbf{x} = (x_1, \cdots, x_{n^2-m})$ for a fixed order of these unknown entries,

$$\begin{cases} f_1(x_1) = 0, \\ f_2(x_1, x_2) = 0, \\ \cdots\cdots, \\ f_{n^2-m}(x_1, \cdots, x_{n^2-m}) = 0 \end{cases} \tag{4.2.13}$$

for a set of polynomial functions $f_1, \cdots, f_{n^2-m}$ by using one of the computational methods discussed in Aubry and Maza [2]. Certainly, for each $k \geq 1$, these f_i are dependent on k in the sense that the coefficients of f_i are dependent on the values $Y_k|_\Omega$. Then we can use any standard bound of the zeros of univariate polynomials to find a bound of these variables $\mathbf{x}$ iteratively from the reduced system above. Indeed,

the bound on x_1 of this system is obtained by $\max\{1, |a_i|, i = 1, \cdots, r+1\}$ with coefficients a_i of the first univariate equation $f_1 = 0$ which are dependent on $Y_k|_\Omega$. Since $Y_k|_\Omega$ is bounded by C_1, we see x_1 is bounded in terms of C_1. Then x_2 can be bounded from the second equation which is now univariate if assuming x_1 is known. x_2 can be bounded in terms of the coefficients of f_2 and the bound on x_1. And so on. In summary, all the entries of Y_k with indices in Ω^c can be bounded in terms of the entries in $Y_k|_\Omega$. In other words, $\|Y_k\| \leq C_2 < \infty$ with a positive constant C_2 for all $k \geq 1$ which is dependent on C_1 above.

It now follows that there exists a subsequence Y_{k_j} which converges to $Y^\star$. Next by (4.2.10), X_k are bounded because of Y_k are bounded and hence, $X_k, , k \geq 1$ have a convergent subsequence and $X_{k_j} \to X^\star$ when $k_j \to \infty$ without loss of generality. By Lemma 4.2.21, we have $(Y^\star)_{\Omega^c} = (X^\star)_{\Omega^c}$. Finally, it is easy to see (5.2.9) which follows from the facts that set $\mathcal{A}_\Omega$ and set $\overline{\mathcal{M}_{r_g}}$ are closed sets. These complete the proof. □

Although we do not know how to check if there are only finitely many matrices $Y \in \overline{\mathcal{M}_r}$ satisfying $(Y)_\Omega = \mathbf{x}$, we can see if the norms of Y_k are bounded or not from the algorithm. If they are bounded, the conclusions of Theorem 4.2.22 hold. In general, $X^\star \neq Y^\star$ as r_g is not equal to $\text{rank}(M)$. For example, when $r_g < \text{rank}(M)$, Y^* will not be equal to M and hence, $Y^\star$ does not satisfy $(Y^\star)_\Omega = M_\Omega$ in general. Of course X^* satisfies the interpolation conditions $(X^*)_\Omega = M_\Omega$, but $\text{rank}(X^\star)$ may be bigger than r_g. That is, informally speaking, when $r_g < \text{rank}(M)$, the chance of $X^\star = M$ is bigger than the chance $Y^* = M$. On the other hand, when $r_g > \text{rank}(M)$, there are more possibilities of matrices with rank $= r_g$ satisfying the interpolatory conditions. Anyway, if $X^\star - Y^\star \neq 0$, the guess r_g is not correct and we need to increase r_g.

Finally, even though $X^* \neq Y^*$ in general, they satisfy the following nice property.

Proposition 4.2.23. *Let $X^\star$ and $Y^\star$ be matrices in (5.2.9) Then,*

$$Y^\star(X^\star)^\top = Y^\star(Y^\star)^\top \text{ and } (Y^\star)^\top X^\star = (Y^\star)^\top Y^\star.$$

Proof. Using Lemma 4.2.17, we obtain

$$\langle AY^\star + Y^\star B, X^\star - Y^\star \rangle = 0$$

for all $A, B \in \mathbb{R}^{n\times n}$ which implies

$$\langle A^\top, Y^\star(X^\star - Y^\star)^\top \rangle + \langle B, (Y^\star)^\top(X^\star - Y^\star) \rangle = 0$$

for all $A, B \in \mathbb{R}^{n\times n}$. Hence,

$$Y^\star(X^\star - Y^\star)^\top = 0 \text{ and } (Y^\star)^\top(X^\star - Y^\star) = 0.$$

Rearranging the above equations, we obtain the required result. □

4.3 Numerical Results

In this section, we first present some results based on the simple initial guess $X_0 = P_\Omega(M)$. The robustness of Algorithm 4 was demonstrated in Jiang et al. [40]. We shall not repeat the similar numerical experimental results. We mainly present numerical results based on a good strategy to choose quality initial guesses which lead even better performance of Algorithm 4. That is, we recall an efficient computational algorithm called OR1MP for matrix completion in Wang et al. [69]. We use the OR1MP algorithm to get a completed matrix which serves as an initial guess X_0. Our numerical experimental results show that this new initial guess gives more accurate

completion. We measure the error matrices by using the maximum norm of all entries of the matrices. One can see that the maximum norm error is very small and hence, the recovered matrix is very accurate. We shall also use Algorithm 4 to recover images from their partial pixel values and demonstrate that Algorithm 4 is able to recover the images better visually. Thus, this section is divided into two subsections.

Our implementation is in Matlab and all the computational results were obtained on a laptop computer with a 2.50 GHz CPU (4 cores with Matlabs multithreading option enabled) and 16 GB of memory. In our simulations, we generate $n \times n$ matrices of rank r by uniformly sampling r pairs of $n \times 1$ matrices (u_i, v_i) and the rank r matrix is $\sum u_i v_i^T$. The set of observed entries Ω is sampled uniformly at random among all sets of cardinality $|\Omega|$.

4.3.1 Numerical Results: Initial Matrices from the OR1MP Algorithm

In all the experiments in this subsection, we used the initial matrix X_0 from the OR1MP algorithm in Wang et al. [69] based on the $P_\Omega(M)$ using a few iterations, that is, $X_0 = \text{OR1MP}(P_\Omega(M))$.

Example 4.3.1. In this example, we show the maximum missing rate that Algorithm 4 can recover a matrix when its rank is fixed. Together we show the computational times. Abbreviations used in Tables in this example are as follows:

M.R. = Missing Rate, the fraction of missing entries = $\frac{m}{n^2}$,

O.R. = oversampling ratio = $\frac{m}{2nr-r^2}$,

M.C.E. = Maximum Component Error = $\max_{i,j}|(X_{recovered})_{i,j} - M_{i,j}|$,

A.R.E. = Average Relative Error = $\|P_\Omega(Y_k) - P_\Omega(M)\|_F \,/\, \|P_\Omega(M)\|_F$,

Table 4.1: Numerical results based on 100×100 matrices averaged over 20 runs

Rank	M.R.	O.R.	M.C.E	A.R.E	Time
2	0.80	5	9.5202e-04	3.7217e-05	0.4171
5	0.61	4	6.0350e-04	1.0894e-05	0.2648
10	0.43	3	4.4343e-04	4.4977e-06	0.2778
20	0.28	2	6.0317e-04	2.0486e-06	0.8492
35	0.25	1.3	1.2698e-06	0.0015	2.8798
50	0.025	1.3	0.0013	7.2350e-07	1.2605

Table 4.2: Numerical results based on 250×250 matrices averaged over 20 runs

Rank	M.R.	O.R.	M.C.E	A.R.E	Time
10	0.76	3	6.6930e-04	3.0595e-06	1.2283
20	0.53	3	2.2215e-04	1.0460e-06	1.3495
50	0.28	2	2.0560e-04	3.3083e-07	2.2624
75	0.18	1.6	2.6951e-04	2.0955e-07	4.5208
100	0.168	1.3	3.9345e-04	1.6690e-07	14.3622
125	0.025	1.3	6.2102e-04	1.1374e-07	8.8464

Table 4.3: Numerical results based on 500×500 matrices averaged over 10 runs

Rank	M.R.	O.R.	M.C.E	A.R.E	Time
25	0.70	3	2.8565e-04	5.5169e-07	4.5253
50	0.62	2	1.6818e-04	2.4458e-07	10.7270
100	0.28	2	8.6199e-05	8.0210e-08	11.3425
150	0.23	1.5	1.2031e-04	5.6053e-08	35.3097
200	0.04	1.5	1.5896e-04	3.2623e-08	24.1821
250	0.0250	1.3	3.3090e-04	2.8449e-08	46.9267

Table 4.4: Numerical results based on 1000×1000 matrices averaged over 10 runs

Rank	M.R.	O.R.	M.C.E	A.R.E	Time
50	0.70	3	6.8718e-05	1.3722e-07	30.1813
100	0.52	2.5	3.7074e-05	5.2213e-08	50.0631
200	0.10	2.5	2.6120e-05	1.2338e-08	42.7043
300	0.05	1.85	5.1339e-05	1.0448e-08	83.9782
400	0.04	1.5	7.1099e-05	8.0391e-09	186.2271
500	0.0025	1.33	2.4592e-04	6.5708e-09	226.3912

Example 4.3.2. Next we provide another tables to show that our algorithm is very effective in recovering the original matrix. We let the missing rate $= 0.1, 0.2, \cdots, 0.9$ and find the largest rank our algorithm can complete within maximum norm error $< 1e-3$, that is, every entry of the completed matrix is accurate to the first three digits. That is, for a fixed missing rate δ, we randomly find the known indices set Ω with $|\Omega|/(n^2) = 1-\delta$ and then we randomly generate a matrix M of size $n \times n$ with rank $r \geq 1$. We use M_Ω, Ω, and r to recover M (the stopping criterion is $1e-5$ of the consecutive iterations), check if the completed matrix $\widehat{M}$ approximates M in the maximum norm within $\epsilon = 1e-3$, and repeat the computation in 10 times. If all 10 computations are able to accurately recover M, we advance r by $r+1$ and repeat the above procedures until the accurate recovery is less than 10 times for a fixed r. In this way, we can find the largest rank for a fixed missing rate. As we use two initial guesses, we summarize the computational results in Table 4.5.

missing rates	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	
largest ranks	30	16	19	14	9	7	5	2	1	OR1MP
largest ranks	13	14	13	10	9	7	3	2	1	M_Ω

Table 4.5: maximum ranks based on matrices of size 100×100 with initial values from OR1MP (second row) and from the initial matrix M_Ω (third row)

From Table 4.5, we can see that using OR1MP algorithm to generate an initial guess for Algorithm 4 is much better when the rates of missing entries are small. When the rate of missing entries are large, the performance is similar. If this table is compared with the ones in Wei et al. [70], we remind the reader that we use a much tougher criterion $\epsilon = 1e-3$ in the maximum norm to find the maximum rank than the relative Frobenius norm error used in Wei et al. [70].

If we use the standard relative Frobenius norm error, we have largest ranks that Algorithm 4 can recover 100% times listed in Table 4.6 with two different initial guesses. We can see that the performance increases greatly when using a completed matrix from OR1MP algorithm.

missing rates	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	
largest ranks	132	106	83	68	41	40	25	14	4	OR1MP
largest ranks	33	29	25	24	19	15	11	8	4	M_Ω

Table 4.6: maximum ranks based on matrices of size 200×200 with initial values from OR1MC (second row) and from the initial matrix M_Ω (third row)

4.3.2 Comparative Performance Analysis with other Algorithms

In this section, we would compare our algorithm with two among the many popular matrix completion algorithms, namely Singular Value Thresholding (SVT) algorithm and Iteratively Reweighted Least Squares Minimization (IRLSM). The algorithms are described in appendix A and B

This section contains some numerical results of comparison with SVT, IRLSM with our APA for matrix completion. Let $e_{APA} = M - M_{APA}, e_{IRLSM} = M - M_{IRLSM}$ and $e_{SVT} = M - M_{SVT}$. They are the average computed based on 100 repeated experiments.

missing%	times	$\|\|e_{APA}\|\|_\infty$	times	$\|\|e_{IRLSM}\|\|_\infty$	times	$\|\|e_{SVT}\|\|_\infty$
60	1.20e+00	4.91e-08	1.39e+00	3.84e-02	2.19	0.3613
50	4.85e-01	3.63e-08	8.47e-01	8.68e-07	2.19	0.2515
40	2.60e-01	2.82e-08	5.86e-01	7.07e-07	2.17	0.1721
30	1.52e-01	2.43e-08	4.82e-01	7.47e-07	2.13	0.1291
20	9.78e-02	1.78e-08	4.32e-01	8.28e-07	1.29	0.0878
10	6.59e-02	9.45e-09	4.08e-01	8.84e-07	1.12	0.0592

Table 4.7: Size of Matrix is 100×100, rank = 10 over 100 repeated experiments

missing %	times	$\|\|e_{APA}\|\|_\infty$	times	$\|\|e_{IRLSM}\|\|_\infty$	times	$\|\|e_{SVT}\|\|_\infty$
60	7.32e+0	1.939e-01	3.61e-01	8.806e-01	2.59	0.8064
50	1.63e+0	6.275e-08	2.24e+00	6.210e-03	2.27	0.4727
40	6.97e-01	4.414e-08	1.15e+00	1.414e-06	2.31	0.3184
30	3.68e-01	3.485e-08	7.79e-01	1.238e-06	2.32	0.2306
20	2.26e-01	2.708e-08	6.75e-01	1.305e-06	2.35	0.1705
10	1.37e-01	1.547e-08	6.05e-01	1.511e-06	1.54	0.1106

Table 4.8: Size of Matrix is 100×100, rank $= 15$ over 100 repeated experiments

missing%	times	$\|\|e_{APA}\|\|_\infty$	times	$\|\|e_{IRLSM}\|\|_\infty$	times	$\|\|e_{SVT}\|\|_\infty$
60	4.16e+01	4.232e+00	5.27e-01	2.445e+00	4.16e+00	1.2944
50	6.71e+00	1.270e-07	8.81e-01	3.730e-01	2.45e+00	0.8157
40	1.79e+00	6.986e-08	2.68e+00	3.288e-06	2.35e+00	0.5205
30	8.30e-01	5.130e-08	1.49e+00	2.013e-06	2.40e+00	0.3730
20	4.47e-01	3.952e-08	1.07e+00	1.828e-06	2.42e+00	0.2720
10	2.60e-01	2.275e-08	9.35e-01	1.808e-06	2.55e+00	0.1849

Table 4.9: Size of Matrix is 100×100, rank $= 20$ over 100 repeated experiments

missing%	times	$\|\|e_{APA}\|\|_\infty$	times	$\|\|e_{IRLSM}\|\|_\infty$	times	$\|\|e_{SVT}\|\|_\infty$
60	6.00e+01	3.322e+00	7.77e-01	3.231e+00	6.06e+00	1.7071
50	2.39e+01	3.536e-01	7.28e-01	1.610e+00	2.39e+00	1.1471
40	5.21e+00	1.244e-07	4.29e+00	6.544e-02	2.34e+00	0.8052
30	1.93e+00	7.600e-08	3.05e+00	3.611e-06	2.51e+00	0.5559
20	8.78e-01	5.649e-08	1.77e+00	2.526e-06	2.51e+00	0.4075
10	4.59e-01	3.467e-08	1.35e+00	2.452e-06	2.51e+00	0.2869

Table 4.10: Size of Matrix is 100×100, rank $= 25$ over 100 repeated experiments

missing rate	times	$\|\|e_{APA}\|\|_\infty$	times	$\|\|e_{IRLSM}\|\|_\infty$	times	$\|\|e_{SVT}\|\|_\infty$
60	1.67e+01	2.341e-08	3.84e+02	8.450e-06	3.07e+01	0.3177
50	1.02e+01	1.973e-08	3.05e+02	8.666e-06	3.89e+01	0.2210
40	6.71e+00	1.705e-08	2.80e+02	1.072e-05	4.57e+01	0.1589
30	4.79e+00	1.294e-08	2.81e+02	1.311e-05	5.31e+01	0.1189
20	3.39e+00	7.924e-09	2.57e+02	1.470e-05	5.89e+01	0.0830
10	3.09e+00	3.638e-09	2.54e+02	1.654e-05	6.44e+01	0.0548

Table 4.11: Size of Matrix is 1000×1000, rank $= 50$ over 100 repeated experiments

4.3.3 Image Recovery from Partial Pixel Values

We shall use Algorithm 4 to recover images from partial pixel values.

Example 4.3.3. Let us use the standard images knee, penny and thank as testing matrices of pixel values. The image knee is of size 691×691. The image penny is a matrix of size 128×128 and the image thank is of size 300×300. For image knee, we use a missing rate 0.85 to generate M_Ω and use rank=25 to find an approximation of the image knee by using the well-known matrix completion OR1MP algorithm in Wang et al. [69], then we feed the approximation as an initial guess to Algorithm 4 to get a better approximation. Also we use the same known entries M_Ω as an initial guess in our Algorithm 4 to find an approximation of the image directly. All these images are shown in Figure 4.2. We do the same for the images penny and thank. See Figures 4.3 and 4.4. Visually, we can see that starting from an initial guess obtained from the OR1MP algorithm, our Algorithm 4 produces a much better approximation to the image. For image penny, we are able to see the face of Lincoln and the word as well as number 1984 are much cleaner although the root-mean square error (RMSE) may not be better. Many images have been experimented with similar performance.

4.4 Remarks on Existence of Matrix Completion

Recall $\mathcal{M}_r$ is the set of all matrices of size $n \times n$ with rank r and $\overline{\mathcal{M}_r}$ is the set of all matrices with rank $\leq r$. It is clear that $\overline{\mathcal{M}_r}$ is the closure of $\mathcal{M}_r$ in the Zariski sense (cf. Zariski [74]). It is easy to see that dimension $\mathcal{M}_r$ is $2nr - r^2$ (cf. Proposition 12.2 in Harris [36] for a proof). Then the dimension of $\overline{\mathcal{M}_r}$ is also $2nr - r^2$. Also, it is clear that $\overline{\mathcal{M}_r}$ is an algebraic variety. In fact, $\overline{\mathcal{M}_r}$ is an irreducible variety.

Lemma 4.4.1. *$\overline{\mathcal{M}_r}$ is an irreducible variety..*

Proof. Denote by $GL(n)$ the set of invertible $n \times n$ matrices. Consider the action of $GL(n) \times GL(n)$ on $M_n(R)$ given by: $(G_1, G_2) \cdot M \mapsto G_1 M G_2^{-1}$, for all $G_1, G_2 \in$

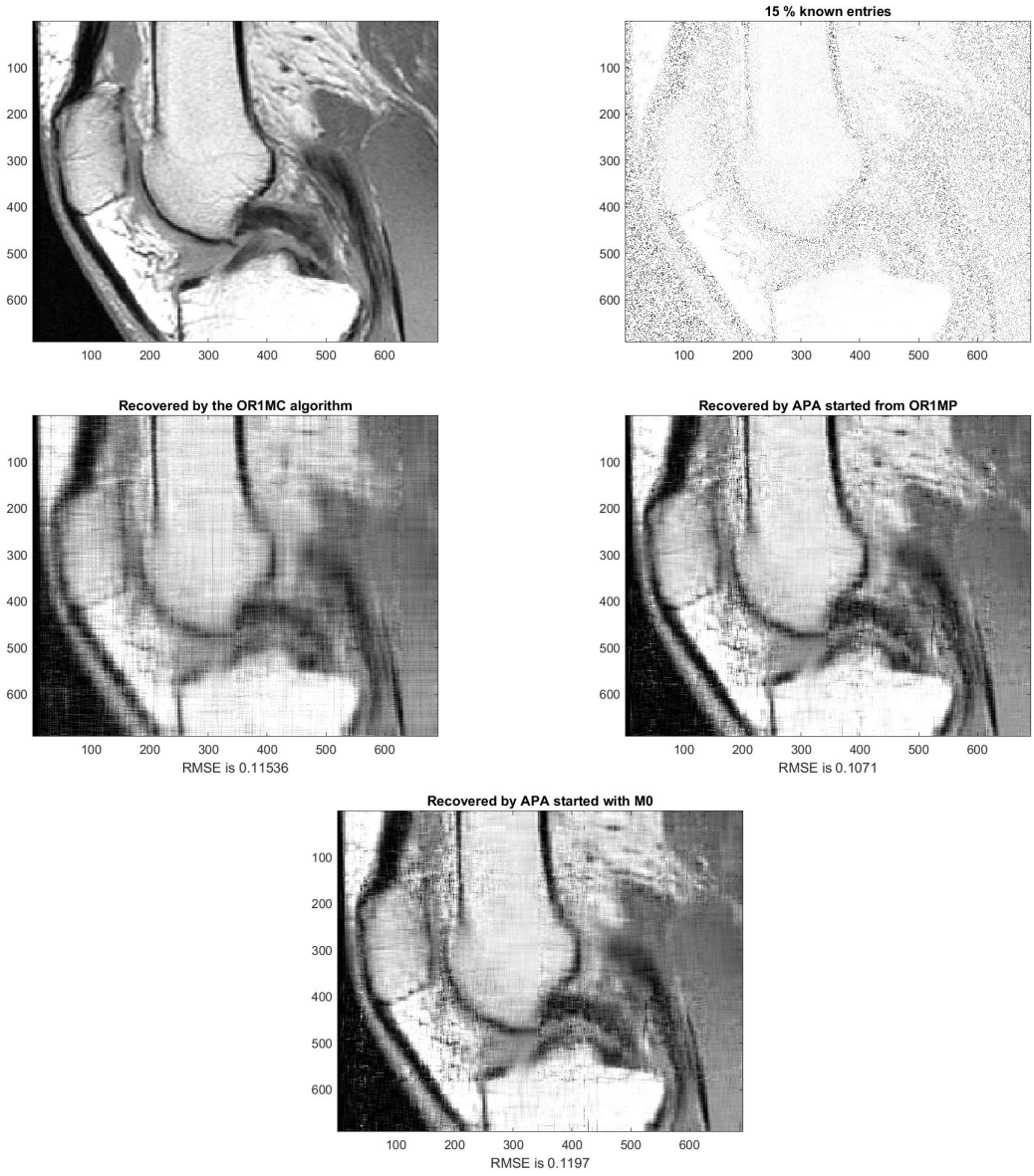

Figure 4.2: Top row: The original image and the image of 15% known entries; Rest of the rows: Outputs from Algorithm OR1MP, Algorithm 4 with initial guess from the Algorithm OR1MP and Algorthm 4 from the 15% known entries based on rank 25.

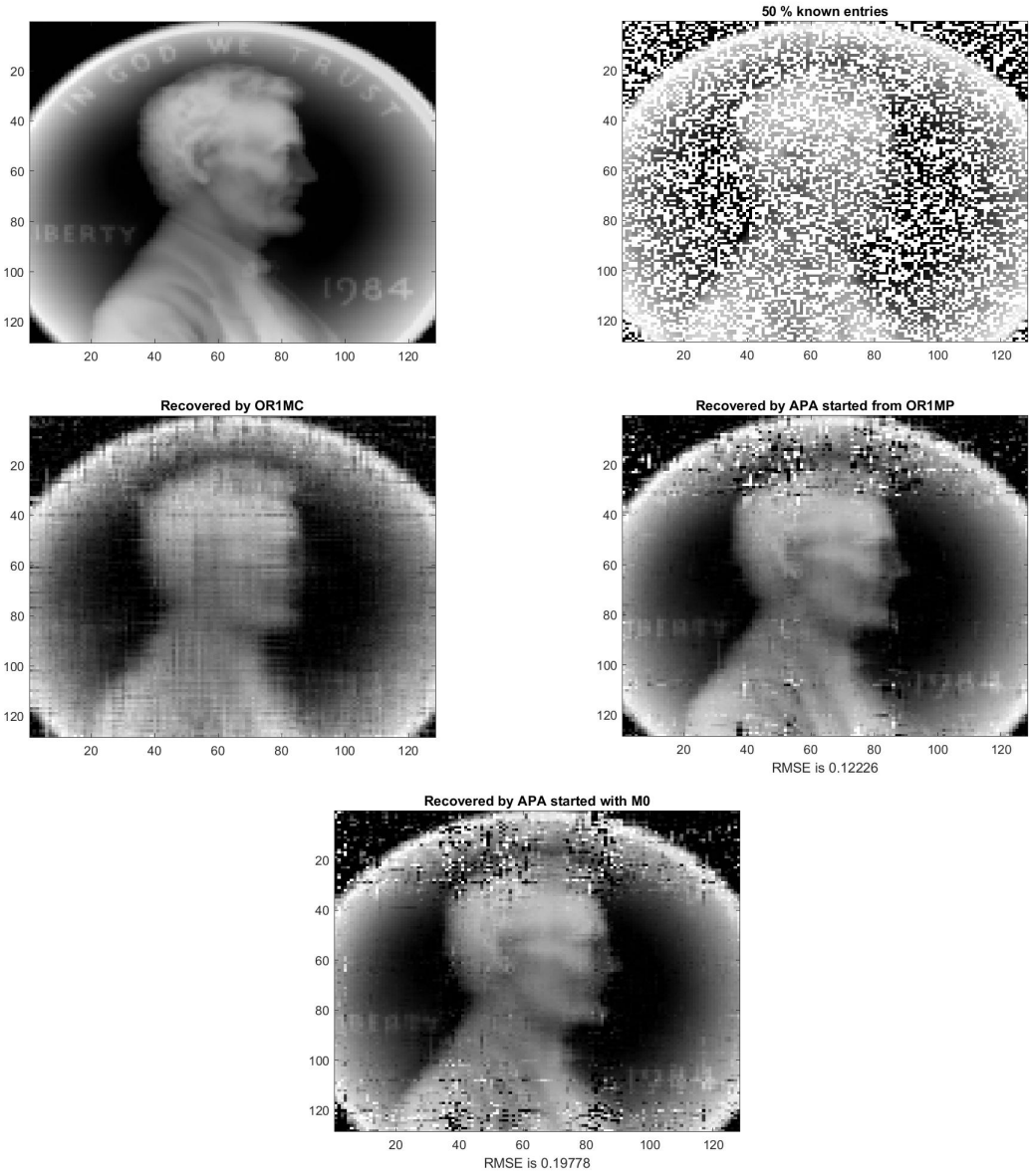

Figure 4.3: Top row: The original image and the image of 50% known entries; Rest of the rows: Outputs from Algorithm OR1MP, Algorithm 4 with initial guess from the Algorithm OR1MP and Algorthm 4 from the 50% known entries based on rank 25.

Figure 4.4: Top row: The original image and the image of 50% known entries; Rest of the rows: Outputs from Algorithm OR1MP, Algorithm 4 with initial guess from the Algorithm OR1MP and Algorthm 4 from the 50% known entries based on rank 25.

$GL(n)$. Fix a rank r matrix M. Then the variety $\mathcal{M}_r$ is the orbit of M. Hence, we have a surjective morphism, a regular algebraic map described by polynomials, from $GL(n) \times GL(n)$ onto $\mathcal{M}_r$. Since $GL(n) \times GL(n)$ is an irreducible variety, so is $\mathcal{M}_r$. Hence, the closure $\overline{\mathcal{M}_{r_g}}$ of the irreducible set $\mathcal{M}_{r_g}$ is also irreducible *c.f* (cf. Example I.1.4 in Hartshorne [37]). □

Consider the map

$$\Phi_\Omega : \overline{\mathcal{M}_r} \to \mathbb{C}^m$$

given by projecting any matrix $X \in \overline{\mathcal{M}_r}$ to its entries in position Ω which form a vector in $\mathbb{R}^m$. Thus, $\Phi_\Omega(\overline{\mathcal{M}_r})$ are exactly the set of all $r-$feasible vectors in $\mathbb{C}^m$. As the projection Φ_Ω is 'nice' (a polynomial map) unlike a Peano curve mapping $[0,1] \to [0,1]^2$), we expect that $\dim(\Phi_\Omega(\overline{\mathcal{M}_r}))$ is less than or equal to $\dim(\overline{\mathcal{M}_r})$ which is less than the dimension of $\mathbb{C}^m$. Thus, $\Phi_\Omega(\overline{\mathcal{M}_r})$ is not able to occupy the whole space $\mathbb{C}^m$. The Lebesgue measure of $\Phi_\Omega(\overline{\mathcal{M}_r})$ is zero and hence, randomly choosing a vector $\mathbf{x} \in \mathbb{C}^m$ will not be in $\Phi_\Omega(\overline{\mathcal{M}_r})$ most likely. Certainly, these intuitions should be made more precise. Recall the following result from Theorem 1.25 in Sec 6.3 of Shafarevich and Hirsch [57].

Lemma 4.4.2. *Let $f : X \to Y$ be a regular map between irreducible varieties. Suppose that f is surjective: $f(X) = Y$, and that $\dim(X) = n$, $\dim(Y) = m$. Then $m \leq n$, and*

1. *$\dim(F) \geq n - m$ for any $y \in Y$ and for any component F of the fibre $f^{-1}(y)$;*
2. *there exists a nonempty open subset $U \subset Y$ such that $\dim(f^{-1}(y)) = n - m$ for $y \in U$.*

We are now ready to prove

Theorem 4.4.3. *If one chooses randomly the entries of a matrix in the positions Ω, probability of completing the matrix to a rank r matrix with given known entries is 0.*

Proof. We mainly use Lemma 4.4.2. Let $X = \overline{\mathcal{M}_r}$ which is an irreducible variety by Lemma 4.4.1. Let $Y = \Phi_\Omega(\overline{\mathcal{M}_r})$ which is also an irreducible variety as it is a continuous image of the irreducible variety $\overline{\mathcal{M}_r}$. Clearly, Φ_Ω is a regular map, we have $\dim \Phi_\Omega(\overline{\mathcal{M}_r}) \leq \dim(\overline{\mathcal{M}_r}) = 2nr - r^2 < m$. Thus, $\Phi_\Omega(\overline{\mathcal{M}_r})$ is a proper lower dimensional closed subset in $\mathbb{C}^m$. For almost all points in $\mathbb{C}^m$, they do not belong to $\Phi_\Omega(\overline{\mathcal{M}_r})$. In other words, for almost all points $\mathbf{x} \in \mathbb{C}^m$, there is no matrix $X \in \overline{\mathcal{M}_r}$ such that $\Phi_\Omega(X) = \mathbf{x}$. □

Next define the subset $\chi_\Omega \subset \overline{\mathcal{M}_r}$ by

$$\chi_\Omega = \left\{ X \in \overline{\mathcal{M}_r} \mid \Phi_\Omega^{-1}(\Phi_\Omega(X)) \text{ is zero dimensional} \right\}.$$

As we are working over Noetherian fields like $\mathbb{R}$ or $\mathbb{C}$, it is worthwhile to keep in mind that all zero dimensional varieties over such fields will have only finitely many points. Next, we recall the following result from Proposition 11.12 in Harris [36].

Lemma 4.4.4. *Let X be a quasi-projective variety and $\pi : X \to \mathbb{P}^m$ a regular map; let Y be closure of the image. For any $p \in X$, let $X_p = \pi^{-1}\pi(p)) \subseteq X$ be the fiber of π through p, and let $\mu(p) = \dim_p(X_p)$ be the local dimension of X_p at p. Then $\mu(p)$ is an upper-semicontinuous function of p, in the Zariski topology on X - that is, for any m the locus of points $p \in X$ such that $\dim_p(X_p) > m$ is closed in X. Moreover, if $X_0 \subseteq X$ is any irreducible component, $Y_0 \subseteq Y$ the closure of its image and μ the minimum value of $\mu(p)$ on X_0, then*

$$dim(X_0) = dim(Y_0) + \mu. \tag{4.4.1}$$

As we saw that $\dim(\Phi_\Omega(\overline{\mathcal{M}_r}) \leq \dim(\overline{\mathcal{M}_r})$, we can be more precise about these dimensions as shown in the following

Lemma 4.4.5. *Assume* $m > \dim(\overline{\mathcal{M}_r})$. *Then* χ_Ω *is open subset of* $\overline{\mathcal{M}_r}$ *and* $\dim(\overline{\mathcal{M}_r}) = \dim(\overline{\Phi_\Omega(\overline{\mathcal{M}_r})}) = \dim(\Phi_\Omega(\overline{\mathcal{M}_r}))$ *if and only if* $\chi_\Omega \neq \emptyset$.

Proof. Assume $\dim(\overline{\mathcal{M}_r}) = \dim(\overline{\Phi_\Omega(\overline{\mathcal{M}_r})}) = \dim(\Phi_\Omega(\overline{\mathcal{M}_r}))$. Then using Lemma 4.4.2, there exists a nonempty open subset $U \subset \Phi_\Omega(\overline{\mathcal{M}_r})$ such that $\dim(\Phi_\Omega^{-1}(y)) = 0$ for all $y \in U$. This implies that $\Phi_\Omega^{-1}(y) \in \chi_\Omega$. Hence $\chi_\Omega \neq \emptyset$.

We now prove the converse. Assume $\chi_\Omega \neq \emptyset$. We will apply Lemma 4.4.4 above by setting $X = \overline{\mathcal{M}_{r_g}}$, $Y = \Phi_\Omega(\overline{\mathcal{M}_{r_g}})$ and $\pi = \Phi_\Omega$. Couple of things to note here are that it does not matter whether we take the closure in $\mathbb{P}^m$ or in $\mathbb{C}^m$ since $\mathbb{C}^m$ is an open set in $\mathbb{P}^m$ and the Zariski topology of the affine space $\mathbb{C}^m$ is induced from the Zariski topology of $\mathbb{P}^m$. $\overline{\mathcal{M}_{r_g}}$ is an affine variety. Therefore, it is a quasi-projective variety.

By our assumption, χ_Ω is not empty. It follows that there is a point $p \in Y$ such that $\pi^{-1}(p)$ is zero dimensional. Since zero is the least dimension possible, we have $\mu = 0$. Hence, using (4.4.1) above, we have $\dim(\overline{\mathcal{M}_r}) = \dim(\overline{\Phi_\Omega(\overline{\mathcal{M}_r})})$. But dimension does not change upon taking closure. So, $\dim(\Phi_\Omega(\overline{\mathcal{M}_r})) = \dim(\overline{\Phi_\Omega(\overline{\mathcal{M}_r})})$. Also, using Lemma 4.4.6, $\chi_\Omega = \{x \in X : \dim(\phi^{-1}\phi(x)) < 1\}$ is an open subset of $\overline{\mathcal{M}_r}$. □

In the proof above, the following result was used. See I.8. Corollary 3 in Mumford [49].

Lemma 4.4.6. *Let* $\phi : X \to Y$ *be a morphism of affine varieties. Let* $\phi^{-1}\phi(x) = Z_1 \cup \cdots \cup Z_j$ *be the irreducible components of* $\phi^{-1}\phi(x)$. *Let* $e(x)$ *be the maximum of the dimensions of the* $Z_i, i = 1, \cdots, j$. *Let* $S_n(\phi) := \{x \in X : e(x) \geq n\}$. *Then, for any* $n \geq 1$, $S_n(\phi)$ *is a Zariski closed subset of* X. *Equivalently* $\{x \in X : \dim(\phi^{-1}\phi(x)) < n\}$ *is an open subset of* X.

Finally, we need the following

Definition 4.4.7. The *degree* of an affine or projective variety of dimension k is the number of intersection points of the variety with k hyperplanes in general position.

For example, the degree of the algebraic variety $\overline{\mathcal{M}_r}$ is known. See Example 14.4.11 in Fulton [32], stated below for convenience.

Lemma 4.4.8. *Degree of the algebraic variety* $\overline{\mathcal{M}_r}$ *is*

$$\prod_{i=0}^{n-r-1} \frac{\binom{n+i}{r}}{\binom{r+i}{r}}$$

We are now ready to prove another main result in this section.

Theorem 4.4.9. *Assume that there exist a finite r-feasible vector* $\mathbf{x} \in \mathbb{C}^m$ *over the given* Ω*. Then, with probability 1, any randomly chosen r-feasible vector* $\mathbf{y}$ *is finite r-feasible. In other words, if one randomly chooses a feasible vector* $\mathbf{y}$ *in the positions* Ω*, then, with probability 1, the matrix can be completed into a rank-r matrix only in* finitely many ways*. In additional, the number of ways to complete will be less than or equal to* $\prod_{i=0}^{n-r-1} \frac{\binom{n+i}{r}}{\binom{r+i}{r}}$.

Proof. We begin by noting that, both $\overline{\mathcal{M}_r}$ and $\Phi_\Omega(\overline{\mathcal{M}_r})$ are irreducible varieties. So, the closure $\overline{\Phi_\Omega(\overline{\mathcal{M}_r})}$ is also an irreducible variety. By the assumption and using Lemma 4.4.5, $\dim(\overline{\mathcal{M}_r}) = \dim(\overline{\Phi_\Omega(\overline{\mathcal{M}_r})})$. Hence, applying Lemma 4.4.2, there exist a nonempty open subset $U \subset \overline{\Phi_\Omega(\overline{\mathcal{M}_r})}$ such that $\Phi_\Omega^{-1}(y)$ is zero-dimensional for all $y \in U$. In other words, If we choose the m entries in positions Ω of a matrix from the open set U, then there are finitely many ways to complete the matrix. The result now follows by recalling that a Zariski open set in an irreducible variety is a dense set whose complement has Lebesque measure zero.

When we fix m entries of a matrix M, the set of matrices of rank r which has those entries in the positions Ω are exactly the intersection points of the variety $\overline{\mathcal{M}_r}$ with m hyperplanes, namely the hyperplanes defined by equations of form $M_{ij} = constant$.

Since $m > \dim(\overline{\mathcal{M}_r}) = 2nr - r^2$, the number of intersection points would be lesser than degree of $\overline{\mathcal{M}_r}$ generically. Now using the exact formula for the degree from lemma 4.4.8, the result follows. $\square$

Regarding Theorem 4.2.22, we have the following open problem: given $\mathbf{x} \in \mathcal{C}^m$, how to check if there are only finitely many matrices $Y \in \overline{\mathcal{M}_r}$ satisfying $(Y)_\Omega = \mathbf{x}$.

Chapter 5

Alternating Projection Algorithm for the Sparse Recovery Problem

5.1 Introduction to Sparse Recovery Problem

The matrix completion problem we considered in the last chapter is closely related to the following classical problem in the area of compressed sensing known as sparse vector recovery problem:

$$\begin{aligned} \underset{X}{\text{minimize}} \quad & \|X\|_0 \\ \text{subject to} \quad & A\mathbf{x} = \mathbf{b}, \end{aligned} \tag{5.1.1}$$

where $A \in \mathbb{R}^{n \times N}, \mathbf{x} \in \mathbb{R}^N, \mathbf{b} \in \mathbb{R}^n, n << N$ and $\|\mathbf{x}\|_0$ is the ℓ_0 quasi-norm of a vector $\mathbf{x}$. Recall that the ℓ_0 quasi-norm of a vector is the number of non-zero components of the vector.

Sparse solutions of underdetermined linear systems have been studied for last twenty years starting from [Chen, Donoho, Saunders, 1998[19]] and [Tibshirani, 1996[62]]

and then became a major subject of research as a part of compressive sensing study since 2006 due to [Donoho, 2006[23]], [Candés, 2006[17]], [Candés and Tao, 2005[13]], and [Candés, Romberg, and Tao, 2006[15]]. Many numerical algorithms have been developed since then. Several algorithms are based on classic convex minimization approach (cf. e.g. Hale et al. [35], Beck and Teboulle [6], Lai and Yin [44], and etc.). Several algorithms are based on iteratively reweighted ℓ_1 minimization or ℓ_2 minimizations (cf. [Candés, Watkin, and Boyd, 2008[16]], [Daubechies, DeVore, 2010,[20]] and [Lai, Xu, and Yin, 2013[45]]). Several researchers started the ℓ_q minimization for $q \in (0, 1)$, e.g. in [Foucart and Lai, 2009[30]] and [Lai and Wang, 2011[43]]. Various other algorithms are based on greedy or orthogonal matching pursuit (cf. e.g. [DeVore and Temlyakov, 1996[22]], [Tropp, 2004[64]], and [Kozlov and Petukhov, 2010[42]]). some algorithms are also based on the hard thresholding technique such as in [Blumensath and Davies, 2009[7]], [Blumensath and Davies, 2010[8]], [Foucart, 2011[29]] and etc. Various other numerical methods were also proposed. See, e.g. [Dohono, Maleki, and Montanari, 2009[24]], [Rangan, 2011[55]], [Gong, Zhang, Lu, Huang, and Ye,2013[33]], [Wang and Ye, 2014[68]] and etc.

To the best of our knowledge, the method in Kozlov and Petukhov [42] is the most effective in finding sparse solutions. Thus, we shall apply the alternating projection method to the sparse recovery problem and establish some sufficient conditions for local and global convergence of the algorithm. We will conclude the chapter by deriving upper and lower bounds for sparsity of a minimizer of the problem 5.1.1.

5.1.1 Notation

Let $\mathcal{L}_s(\mathbb{R}^N)$ denote the collection of all $s-$sparse vectors in $\mathbb{R}^N$,

$$\mathcal{L}_s(\mathbb{R}^N) := \left\{ x \in \mathbb{R}^N \mid \quad \|x\|_0 = s \right\}$$

and $\mathcal{P}_{\mathcal{L}_s}$ and $\mathcal{P}_{\mathcal{A}}$ denote the projection onto the set $\mathcal{L}_s(\mathbb{R}^N)$ and the affine space $\mathcal{A} := \{\mathbf{x} : A\mathbf{x} = \mathbf{b}\}$, respectively. It is easy to know $\mathcal{A} = \text{Null}(A) + \mathbf{x}_0$, where $\mathbf{x}_0 \in \mathbb{R}^N$ satisfies $A\mathbf{x}_0 = \mathbf{b}$. Note that the projection $\mathcal{P}_{\mathcal{L}_s}(\mathbf{x}_k)$ can be computed easily by setting the smallest $n - s$ components of the vector $\mathbf{x}_k$ to zero.

5.2 APA for Sparse Recovery Problem

Our algorithm can be stated as follows:

Algorithm 5: Alternating Projection Algorithm for ℓ_0 Minimization

Data: Sparsity s of the solution $\mathbf{x}_\star$, the tolerance ϵ whose default value is 1e-6

Result: $\mathbf{x}_k$ a close approximation of $\mathbf{x}_\star$

Initialize $\mathbf{x}_0$ to a random vector in the affine space $\mathcal{A}$;

repeat

Step 1: $\mathbf{y}_k = \mathcal{P}_{\mathcal{L}_s}(\mathbf{x}_k)$

Step 2: $\mathbf{x}_{k+1} = \mathcal{P}_{\mathcal{A}}(\mathbf{y}_k)$;

until *The smallest* $n_2 - s$ *components of* $\mathbf{x}_{k+1}$ *have magnitude less than* ϵ;

We will now discuss the convergence of Algorithm 5. Then we shall present its numerical performance in the next section.

5.2.1 Convergence of the APA Algorithm for Sparse Recovery Problem

Local Convergence

In this section, we will prove the local convergence of the algorithm. Before we begin, we need some elementary results.

Lemma 5.2.1. *Let $\mathcal{L}_s(\mathbb{R}^n)$ be the collection defined as follows.*

$$\mathcal{L}_s(\mathbb{R}^n) = \bigsqcup_{\mathcal{I}} \{x \in \mathbb{R}^n \mid x_j = 0 \quad \forall j \in \mathcal{I}^c\},$$

where the index set $\mathcal{I}$ ranges over all the subsets of $\{1, 2, \cdots, n_1\}$ which has cardinality s. Here, $\bigsqcup_{\mathcal{I}}$ stands for the disjoint union over $\mathcal{I}$. Then $\mathcal{L}_s(\mathbb{R}^n)$ consists of a disjoint union of affine spaces.

Proof. It is easy to see that the statement is correct. □

Lemma 5.2.2. *The set of vectors in $\mathbb{R}^N$ for which $\mathcal{P}_{\mathcal{L}_s}(x)$ is single-valued, is given by the open set*

$$V_s = \left\{x \in \mathbb{R}^{n_2} \mid |x_{i_1}| \geq |x_{i_2}| \geq \cdots |x_{i_{n_2}}|, |x_{i_{s+1}}| \neq |x_{i_s}|\right\}$$

consisting of vectors which has the property that if one arrange the components in decreasing order of magnitude, then s^{th} and $(s+1)^{th}$ terms are distinct.

Proof. We first start by noting that the projection $\mathcal{P}_{\mathcal{L}_s}(\mathbf{x})$ is obtained by setting the smallest $N-s$ components in magnitude of the vector x to zero. Hence, the projection is single-valued if the $N-s$ smallest components of $\mathbf{x}$ are in unique positions(indices). Hence we must have that the $(N-s)^{th}$ and $(N-s+1)^{th}$ components of x must be distinct. Now we will show that the set V_s is an open set. Let $\mathbf{x} = (x_1, x_2, \cdots, x_N) \in \mathbb{R}^N$ with $|x_{i_1}| \geq |x_{i_2}| \geq \cdots |x_{i_N}|, |x_{i_{s+1}}| \neq |x_{i_s}|$. Let

$$\epsilon := \frac{||x_{i_{s+1}}| - |x_{i_s}||}{4}$$

Consider an open ball $B_\epsilon(\mathbf{x})$ centered at $\mathbf{x}$ of radius ϵ. We have, for all $\mathbf{y} \in B_\epsilon(\mathbf{x})$ and $j \in \{1, 2, \cdots, N\}$,

$$||y_j| - |x_j|| \leq |y_j - x_j| \leq \|\mathbf{y} - \mathbf{x}\| < \epsilon$$

Therefore, we have

$$|y_{i_{j+1}}| \leq |x_{i_{j+1}}| + ||y_{i_{j+1}}| - |x_{i_{j+1}}|| < |x_{i_{s+1}}| + \epsilon < \frac{||x_{i_{s+1}}| + |x_{i_s}||}{2}$$

for $j \geq s$. Similarly,

$$|y_{i_j}| \geq |x_{i_j}| - ||x_{i_j}| - |y_{i_j}|| > |x_{i_s}| - \epsilon > \frac{||x_{i_{s+1}}| + |x_{i_s}||}{2}$$

for $j \leq s$. Hence, we deduce that, for all $\mathbf{y} \in B_\epsilon(\mathbf{x})$ and $j \in \{1, 2, \cdots, s\}$, $|y_{i_j}| \geq |y_{i_s}|$, which implies that $\mathbf{y} \in V_s$ and, therefore, $B_\epsilon(\mathbf{x}) \subset V_s$. □

Next let us recall the following well-known theorem, whose proof we reviewed in Chapter 2.

Theorem 5.2.3 (Von Neumann [67])**.** *If L_1 and L_2 are two closed subspaces of a Hilbert space X, then the sequence of operators*

$$\mathcal{P}_{L_1}, \mathcal{P}_{L_2}\mathcal{P}_{L_1}, \mathcal{P}_{L_1}\mathcal{P}_{L_2}\mathcal{P}_{L_1}, \mathcal{P}_{L_2}\mathcal{P}_{L_1}\mathcal{P}_{L_2}\mathcal{P}_{L_1}, \cdots$$

converge to $\mathcal{P}_{L_1 \cap L_2}$. In other words,

$$\lim_{k\to\infty} (\mathcal{P}_{L_2}\mathcal{P}_{L_1})^k(x) = \mathcal{P}_{L_1 \cap L_2}(x)$$

for all $x \in X$.

With the above preparation, we are ready to prove local the convergence of the theorem.

Theorem 5.2.4. *If $\mathbf{x}_\star$ is an isolated point of $\mathcal{L}_s(\mathbb{R}^N) \cap \mathcal{A}$. Then, Algorithm 5 will locally converge to $\mathbf{x}_\star$ linearly.*

Proof. Let $\mathcal{I} = \text{Supp}(\mathbf{x}_\star)$ be the support of $\mathbf{x}_\star$ and $s = \|\mathbf{x}_\star\|_0$. Consider an open set V_s of vectors which has the property that their $n - s$ smallest components are are in

unique positions(indices). In fact, V_s can be concretely described as

$$V_s = \left\{ \mathbf{x} \in \mathbb{R}^N \mid \quad |x_{i_1}| \geq |x_{i_2}| \geq \cdots |x_{i_{n_2}}|, |x_{i_s}| \neq |x_{i_{s+1}}| \right\}.$$

Clearly $\mathbf{x}_\star \in V_s$. Let $B(r)$ be an open ball centered at $\mathbf{x}_\star$ and of radius r completely contained inside V_s. Since $B(r) \subseteq V_s$, for any $\mathbf{x} \in B(r)$, the projection $\mathcal{P}_{\mathcal{L}_s}(\mathbf{x})$ is uniquely defined. Since affine spaces in a finite dimensional Euclidean space are closed, one can shrink the ball $B(r)$, if necessary, such that the restriction $\mathcal{L}_s(\mathbb{R}^{n_2})|_{B(r)}$ of the set of $s-$sparse vectors to the open set $B(r)$ is an affine space. Then under the assumption of the hypothesis in this theorem, the result follows from Theorem 5.2.3. □

A sufficient condition that guarantees that $\mathbf{x}_\star$ is an isolated point is that the tangent spaces of $\mathcal{L}_s(\mathbb{R}^N)$ and $\mathcal{A}$ intersect trivially.

Lemma 5.2.5. *Assume A has the following property:*

$$\mathcal{L}_s(\mathbb{R}^N) \cap \mathrm{Null}(A) = \{0\}, \tag{5.2.1}$$

where $\mathrm{Null}(A)$ *is the null space of A. Furthermore, assume that $\mathbf{x}_\star \in \mathcal{L}_s(\mathbb{R}^N) \cap \mathcal{A}$. Then $\mathbf{x}_\star$ is an isolated point of $\mathcal{L}_s(\mathbb{R}^N) \cap \mathcal{A}$.*

Proof. Assume, on the contrary, that $\mathbf{x}_\star$ is not an isolated point of the set $\mathcal{L}_s(\mathbb{R}^{n_2}) \cap \mathcal{A}$. Then, since A and $\mathcal{L}_s(\mathbb{R}^N)$ are locally affine spaces, there exist a linear space L of dimension greater than or equal to 1 such that $L + \mathbf{x}_\star \subseteq \mathcal{L}_s(\mathbb{R}^N) \cap \mathcal{A}$. Since each of the intersecting spaces are affine spaces locally, L must lie also in the intersection of their tangent spaces. Hence,

$$L \subseteq T_{\mathcal{L}_s(\mathbb{R}^N)}(\mathbf{x}_\star) \cap \mathrm{Null}(A)$$

where $T_{\mathcal{L}_s(\mathbb{R}^N)}(\mathbf{x}_\star)$ is the tangent space to $\mathcal{L}_s(\mathbb{R}^N)$ at the point $\mathbf{x}_\star$. Now, since $\mathcal{L}_s(\mathbb{R}^N)$

is an union of linear spaces, let us assume $\mathbf{x}_\star \in L_0 \subseteq \mathcal{L}_s(\mathbb{R}^N)$ lies in a linear space L_0 contained in $\mathcal{L}_s(\mathbb{R}^N)$. Therefore, we have

$$L \subseteq T_{L_0}(\mathbf{x}_\star) \cap \text{Null}(A) = L_0 \cap \text{Null}(A) \subseteq \mathcal{L}_s(\mathbb{R}^N) \cap \text{Null}(A) = \{0\}$$

which leads to the contradiction as L is of dimension greater than or equal to 1. Note that, in order to derive the equality in the last equation, we have used the fact that the tangent space of a linear space is the linear space itself. □

The discussion above leads to the following result

Theorem 5.2.6. *Under the assumption (5.2.1) in Lemma 5.2.5, Algorithm 5 will locally converge linearly.*

Proof. We simply combine Lemma 5.2.5 and Theorem 5.2.4 together to have this result. □

Global Convergence

Now we will investigate conditions under which the above algorithm will converge globally. We shall assume, without loss of generality, that A has full row rank. We say A satisfies the $\eta-$condition if A satisfies the following property: there exists a positive number $\eta_s < 1$ such that

$$\max_{M_{s\times s} \subset A^\top (AA^\top)^{-1} A - I_{n\times n}} \sigma_1(M_{s\times s}) \leq \eta_s, \tag{5.2.2}$$

where $\sigma_1(M)$ is the largest singular value of matrix M. We will abuse the notation a bit and let

$$\eta_s(A) := \max_{M_{s\times s} \subset A^\top (AA^\top)^{-1} A - I_{n\times n}} \sigma_1(M_{s\times s})$$

Note that the $\eta-$condition is different from the classic restricted isometry constant

δ_s which is recalled as follows. The RIC is the smallest $\delta > 0$ such that

$$(1-\delta)\|\mathbf{x}\|_2^2 \leq \|A\mathbf{x}\|_2^2 \leq (1+\delta)\|\mathbf{x}\|_2^2 \tag{5.2.3}$$

for all s-sparse vectors $\mathbf{x} \in \mathbb{R}^n$. It is equivalent to

$$\delta_s = \max_{\substack{S\subset\{1,\cdots,n\}\\ \#(S)\leq s}} \|A_S^\top A_S - \mathbf{Id}\|_{2\to 2} \tag{5.2.4}$$

according to [31]. Similarly, we can rewrite η_s in the following way. For any index sets S_1, S_2 with $\#(S_1) \leq s$ and $\#(S_2) \leq s$ and for any vector $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, we see

$$\begin{aligned}(A_{S_1}\mathbf{x}_{S_1})^\top (AA^\top)^{-1}(A_{S_2}\mathbf{y}_{S_2}) - \mathbf{x}_{S_1}^\top \mathbf{y}_{S_2} &= \langle (A_{S_1}^\top (AA^\top)^{-1} A_{S_2} - \mathbf{Id})\mathbf{x}_{S_1}, \mathbf{y}_{S_2}\rangle \\ &\leq \mathbf{x}_{S_1}^\top (M_{s\times s} - \mathbf{Id})\mathbf{y}_{S_2} \leq \|\mathbf{x}_{S_1}\|_2 \sigma_1(M_{s\times s})\|\mathbf{y}_{S_2}\|_2\end{aligned}$$

or

$$\max_{\mathbf{x}_{S_1}\neq 0, \mathbf{y}_{S_2}\neq 0} \frac{\langle (A_{S_1}^\top (AA^\top)^{-1} A_{S_2} - \mathbf{Id})\mathbf{x}_{S_1}, \mathbf{y}_{S_2}\rangle}{\|\mathbf{x}_{S_1}\|_2 \|\mathbf{y}_{S_2}\|} \leq \sigma_1(M_{s\times s}) \leq \eta_s. \tag{5.2.5}$$

Therefore, we obtain the following characterization for $\eta_s(A)$

Theorem 5.2.7. *We have*

$$\eta_s(A) = \max_{\substack{S_1,S_2\subset\{1,\cdots,n\}\\ \#(S_1)\leq s, \#(S_2)\leq s}} \|A_{S_1}^\top (AA^\top)^{-1} A_{S_2} - (\mathbf{Id})_{S_1,S_2}\|_{2\to 2}. \tag{5.2.6}$$

Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ be two vectors with sparsity $\|\mathbf{x}\|_0 \leq s$ and $\|\mathbf{y}\|_0 \leq s$. If $\sup(\mathbf{x}) \cap$

$\sup(\mathbf{y}) = \emptyset$, then

$$\begin{aligned}|\langle (AA^\top)^{-1}A_{S_2}\mathbf{y}, A_{S_1}\mathbf{x}\rangle| =& |\langle (AA^\top)^{-1}A_{S_2}\mathbf{y}, A_{S_1}\mathbf{x}\rangle - \langle (\mathbf{Id}_{S_1,S_2}\mathbf{y}, \mathbf{x}\rangle| \\ =& |\langle A_{S_1}^\top (AA^\top)^{-1}A_{S_2} - \mathbf{Id}_{S_1,S_2})\mathbf{y}, \mathbf{x}\rangle| \\ \leq& \|A_{S_1}^\top (AA^\top)^{-1}A_{S_2} - (\mathbf{Id})_{S_1,S_2}\|_{2\to 2}\|\mathbf{x}\|_2\|\mathbf{y}\|_2 \\ \leq& \eta_s\|\mathbf{x}\|_2\|\mathbf{y}\|_2. \end{aligned} \tag{5.2.7}$$

Let us recall a concept called the (s,t)-restricted orthogonality condition $\theta_{s,t} = \theta_{s,t}(M)$ of a matrix $M \in \mathbb{R}^{m\times n}$ is the smallest $\theta \geq 0$ such that

$$|\langle M\mathbf{u}, M\mathbf{v}\rangle| \leq \theta\|\mathbf{u}\|_2\|\mathbf{v}\|_2 \tag{5.2.8}$$

(cf. Chapter 6 in [31]). That is, according to [31],

$$\theta_{s,t} = \max\{\|M_T^\top M_S\|_{2\to 2}, S\cap T = \emptyset, \#(S) \leq s, \#(T) \leq s\}. \tag{5.2.9}$$

Returning to the $\eta-$condition, we assume that A is of full row rank.

Theorem 5.2.8. *Define by $M = (AA^\top)^{-1/2}A$ a normalized sensing matrix. Then*

$$\eta_s(A) = \theta_{s,s}(M). \tag{5.2.10}$$

Proof. Indeed, we have

$$\begin{aligned}\|M_T^\top M_S\|_{2\to 2} &= \max\{\frac{|\langle M_T\mathbf{u}_T, M_S\mathbf{v}_S\rangle|}{\|\mathbf{u}_T\|_2\|\mathbf{v}_S\|_2},\ s\text{-sparse vectors } \mathbf{u}_T, \mathbf{v}_S\} \\ &= \max\{\frac{|\langle A_S^\top (AA^\top)^{-1}A_T\mathbf{u}_T, \mathbf{v}_S\rangle|}{\|\mathbf{u}_T\|_2\|\mathbf{v}_S\|_2},\ s\text{-sparse vectors } \mathbf{u}_T, \mathbf{v}_S\} \\ &\leq \|A_S^\top (AA^\top)^{-1}A_T - (\mathbf{Id})_{S,T}\|_{2\to 2} \leq \eta_s. \end{aligned} \tag{5.2.11}$$

It thus follows $\theta_{s,s} \leq \eta_s$. Similarly, we have the other direction. This completes the

proof of (5.2.10). □

As many properties of $\theta_{s,s}$ are known, the relation in (5.2.10) helps understand η_s better. For example, $\theta_{s,s} \leq \delta_{2s}$ and $\delta_{2s} \leq \delta_s + \theta_{s,s}$.

We are now ready to establish the following convergence result.

Theorem 5.2.9. *Suppose that A is of full row rank and satisfies the $\eta-$condition with $\eta_{4s}(A) < 1/2$. Then Algorithm 5 will globally converge linearly.*

Proof. Let $\mathbf{x}^* \in \mathbb{R}^n$ be a solution of (5.1.1) with $\mathbf{x}^* \in A \cap B$. Starting with an initial guess $\mathbf{y}_0$, we find

$$\mathbf{x}_{k+1} = \mathcal{P}_A(\mathbf{y}_k) = A^\top (AA^\top)^{-1}\mathbf{b} - (A^\top (AA^\top)^{-1}A - I_{n\times n})\mathbf{y}_k.$$

Also, we have

$$\|\mathbf{y}_{k+1} - \mathbf{x}_{k+1}\|^2 = \|\mathbf{y}_{k+1} - \mathbf{x}^*\|^2 + \|\mathbf{x}^* - \mathbf{x}_{k+1}\|^2 + 2\langle \mathbf{y}_{k+1} - \mathbf{x}^*, \mathbf{x}^* - \mathbf{x}_{k+1}\rangle$$

and

$$\|\mathbf{y}_{k+1} - \mathbf{x}_{k+1}\|^2 = \|\mathcal{P}_B(\mathbf{x}_{k+1}) - \mathbf{x}_{k+1}\|^2 \leq \|\mathbf{x}^* - \mathbf{x}_{k+1}\|^2.$$

Combining the right-hand sides of the above two equations together leads to

$$\begin{aligned}
\|\mathbf{y}_{k+1} - \mathbf{x}^*\|^2 &\leq 2\langle \mathbf{y}_{k+1} - \mathbf{x}^*, \mathbf{x}_{k+1} - \mathbf{x}^*\rangle \\
&= 2\langle \mathbf{y}_{k+1} - \mathbf{x}^*, A^\top (AA^\top)^{-1}\mathbf{b} - (A^\top (AA^\top)^{-1}A - I_{n\times n})\mathbf{y}_k - \mathbf{x}^*\rangle \\
&= 2\langle \mathbf{y}_{k+1} - \mathbf{x}^*, (A^\top (AA^\top)^{-1}A - I_{n\times n})(\mathbf{x}^* - \mathbf{y}_k)\rangle \\
&\leq 2\|\mathbf{y}_{k+1} - \mathbf{x}^*\|\eta_{4s}(A)\|\mathbf{y}_k - \mathbf{x}^*\|.
\end{aligned}$$

In other words, we have

$$\|\mathbf{y}_{k+1} - \mathbf{x}^*\| \leq 2\eta_{4s}(A)\|\mathbf{y}_k - \mathbf{x}^*\|$$

for all $k \geq 1$. Since, by hypothesis, $2\eta_{4s}(A) < 1$, we have

$$\|\mathbf{y}_{k+1} - \mathbf{x}^*\| < \|\mathbf{y}_k - \mathbf{x}^*\|$$

. This completes the proof. □

5.3 Numerical Results

We have used Algorithm 5 to compute sparse solutions and compare the performance of several existing algorithms. Mainly, we compare with the iteratively reweighted ℓ_1 minimization (CWB for short) in Candes et al. [16], the L^1 greedy algorithm (KP) proposed in Kozlov and Petukhov [42], the FISTA in Beck and Teboulle [6], the hard iterative pursuit (HTP) in Foucart [29], and generalized approximate message passing algorithm (GAMP) in Donoho et al. [24], Rangan [55]. LV stands for our Algorithm 5.

Our implementation is in Matlab and all the computational results were obtained on a laptop computer with a 2.50 GHz CPU (4 cores with Matlabs multithreading option enabled) and 16 GB of memory. In our simulations, we generate $n \times n$ matrices of rank r by uniformly sampling r pairs of $n \times 1$ matrices (u_i, v_i) and the rank r matrix is $\sum u_i v_i^T$. The set of observed entries Ω is sampled uniformly at random among all sets of cardinality $|\Omega|$.

We present the frequency of recovery of Gaussian random matrices of size 128×256 with sparsity varied from 10 to 70 over 500 repeated runs with a tolerance $1e-3$ in maximum norm. In Figure 5.1(left figure), we show the performance of various algorithms. Next we repeat the same experiments based on uniform random matrices

of size 128×256. The performance of frequency of recovery from various algorithm is shown in Figure 5.1 (right).

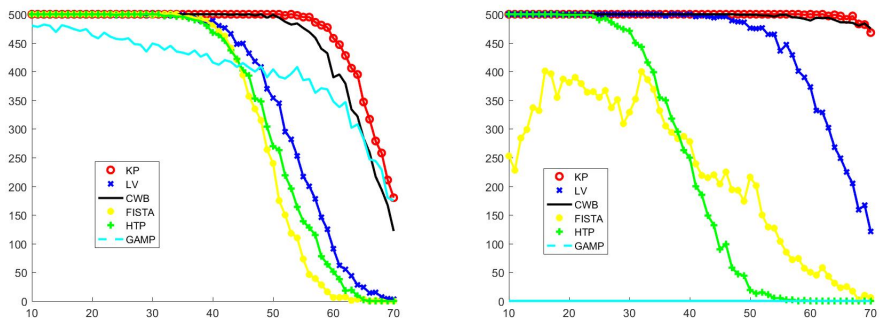

Figure 5.1: Frequency of Sparse Recovery by Various Algorithms from Gaussian random matrices (left) and from uniform random matrices (right)

5.4 Bounds on the Sparsity of a Sparsest Solution

In this section, we will prove bounds on the sparsity of a sparsest solution of the linear system $A\mathbf{x} = \mathbf{b}$, where A is a $m \times n$ matrix. We will assume $\mathbf{x}^\star$ is a sparsest solution of the system. Formally,

$$\mathbf{x}^\star = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_0 \text{ s.t. } A\mathbf{x} = \mathbf{b}$$

We will assume $\mathbf{b} \neq 0$ for the case $\mathbf{b} = 0$ is not interesting. After a scaling if necessary, we will also assume, without loss of generality, that $\|\mathbf{b}\| = 1$. Before we state the bounds, we need some definitions:

Definition 5.4.1. The *spark* of a matrix A, denoted by $\text{spark}(A)$, is the smallest number k such that there exists a set of k columns in A which are linearly dependent. That is,

$$\text{spark}(A) = \min_{\mathbf{x} \neq 0} \|\mathbf{x}\|_0 \text{ s.t. } A\mathbf{x} = 0$$

where $\|\mathbf{x}\|_0$ denotes the number of non-zero components of the vector $\mathbf{x}$.

Now we define a new matrix $\tilde{A}$ as follows: for each $i = 1, 2, \cdots, n$, the i^{th} column $\tilde{A}_i$ of the matrix $\tilde{A}$ is obtained from i^{th} column A_i of the matrix A by orthogonalizing it with respect to the vector $\mathbf{b}$. In other words,

$$\tilde{A}_i = A_i - \langle A_i, \mathbf{b} \rangle \mathbf{b}$$

and

$$\tilde{A} = [\tilde{A}_1 \mid \tilde{A}_2 \mid \cdots \mid \tilde{A}_n]$$

More concisely, after normalizing the system $A\mathbf{x} = \mathbf{b}$ so that $\|\mathbf{b}\| = 1$, $\tilde{A}$ can be described as

$$\tilde{A} = (I - \mathbf{b}\mathbf{b}^T)A$$

where I is an $m \times m$ identity matrix.

Now we are ready to prove the first bound

Theorem 5.4.2.

$$\{\mathbf{x} \mid A\mathbf{x} = \mathbf{b}\} \subseteq \{\mathbf{x} \mid \tilde{A}\mathbf{x} = 0\}$$

In particular,

$$\text{spark}(\tilde{A}) \leq \|\mathbf{x}^\star\|_0$$

Proof. Now note that for any scalars $\alpha_1, \cdots, \alpha_n$, we have

$$\begin{aligned} \alpha_1 \tilde{A}_1 + \cdots + \alpha_n \tilde{A}_n &= \alpha_1 (A_1 - \langle A_1, \mathbf{b} \rangle \mathbf{b}) + \cdots + \alpha_n (A_n - \langle A_n, \mathbf{b} \rangle \mathbf{b}) \\ &= \alpha_1 A_1 + \cdots + \alpha_n A_n - \langle \alpha_1 A_1 + \cdots + \alpha_n A_n, \mathbf{b} \rangle \mathbf{b} \end{aligned} \tag{5.4.1}$$

So, if $\mathbf{x} = (x_1, \cdots, x_n) \in \{\mathbf{x} \mid A\mathbf{x} = \mathbf{b}\}$, then

$$\begin{aligned}\tilde{A}\mathbf{x} = x_1\tilde{A}_1 + \cdots + x_n\tilde{A}_n &= x_1(A_1 - \langle A_1, \mathbf{b}\rangle\mathbf{b}) + \cdots + x_n(A_n - \langle A_n, \mathbf{b}\rangle\mathbf{b}) \\ &= x_1A_1 + \cdots + x_nA_n - \langle x_1A_1 + \cdots + x_nA_n, \mathbf{b}\rangle\mathbf{b} \\ &= \mathbf{b} - \langle \mathbf{b}, \mathbf{b}\rangle\mathbf{b} = 0\end{aligned}$$

The last step follows from the assumption that $\|\mathbf{b}\| = 1$.

For proving the second part of the theorem, note that $\{\mathbf{x} \mid A\mathbf{x} = \mathbf{b}\} \subseteq \{\mathbf{x} \mid \tilde{A}\mathbf{x} = 0\}$. So, we have

$$\text{spark}(\tilde{A}) = \min_{\{\mathbf{x}|\tilde{A}\mathbf{x}=0\}} \|\mathbf{x}\|_0 \leq \min_{\mathbf{x}\in\{\mathbf{x}|A\mathbf{x}=\mathbf{b}\}} \|\mathbf{x}\|_0 = \|\mathbf{x}^\star\|_0$$

□

Now we will need the following proposition to prove our next two main results

Proposition 5.4.3. *If* $\mathbf{x}$ *is a vector such that* $\tilde{A}\mathbf{x} = 0$*, then exactly one of the following holds true:*

- $A\mathbf{x} = 0$
- *There exist a non-zero scalar* λ *such that* $A(\frac{\mathbf{x}}{\lambda}) = \mathbf{b}$*.*

Proof. Assume $\mathbf{x} = (x_1, \cdots, x_n)$ is a vector such that $\tilde{A}\mathbf{x} = 0$, then using equation 5.4.1, we get

$$\begin{aligned}0 = \tilde{A}\mathbf{x} &= x_1\tilde{A}_1 + \cdots + x_n\tilde{A}_n \\ &= x_1A_1 + \cdots + x_nA_n - \langle x_1A_1 + \cdots + x_nA_n, \mathbf{b}\rangle\mathbf{b} \qquad (5.4.2)\end{aligned}$$

Now we have two cases:

Case 1: $\langle x_1 A_1 + \cdots + x_n A_n, \mathbf{b} \rangle = 0$

In this case, using equation 5.4.2, we get

$$x_1 A_1 + \cdots + x_n A_n = 0$$

so, $A\mathbf{x} = 0$

Case 2: $\langle x_1 A_1 + \cdots + x_n A_n, \mathbf{b} \rangle \neq 0$

For this case set $\lambda := \langle x_1 A_1 + \cdots + x_n A_n, \mathbf{b} \rangle$. Then, we get $A(\frac{\mathbf{x}}{\lambda}) = \mathbf{b}$. It is clear that the two cases cannot hold together. Indeed if $A\mathbf{x} = 0$ and $A(\frac{\mathbf{x}}{\lambda}) = \mathbf{b}$, then $\mathbf{b} = 0$ which contradicts our assumption that $\mathbf{b} \neq 0$. Thus the proof is complete. □

We are ready to prove our next two results,

Theorem 5.4.4.

$$\text{spark}(\tilde{A}) \leq \text{spark}(A)$$

Proof. Using equation 5.4.1, we can easily observe that $\{\mathbf{x} \mid A\mathbf{x} = 0\} \subseteq \{\mathbf{x} \mid \tilde{A}\mathbf{x} = 0\}$. Hence,

$$\text{spark}(\tilde{A}) := \min_{\{\mathbf{x} | \tilde{A}\mathbf{x} = 0\}} \|\mathbf{x}\|_0 \leq \min_{\mathbf{x} \in \{\mathbf{x} | A\mathbf{x} = 0\}} \|\mathbf{x}\|_0 =: \text{spark}(A)$$

□

Theorem 5.4.5. *Either of the following holds true always:*

(i) $\text{spark}(A) = \text{spark}(\tilde{A}) \leq \|\mathbf{x}^\star\|_0$

(ii) $\|\mathbf{x}^\star\|_0 = \text{spark}(\tilde{A}) \leq \text{spark}(A)$

Proof. Let $\tilde{\mathbf{x}} = \underset{\mathbf{x} \text{ s.t. } \tilde{A}\mathbf{x} = 0}{\arg\min} \|\mathbf{x}\|_0$. Therefore, $\|\tilde{\mathbf{x}}\|_0 = \text{spark}(\tilde{A})$. Now, using proposition 5.4.3, either $A\tilde{\mathbf{x}} = 0$ or there exist a non-zero scalar λ such that $A(\frac{\tilde{\mathbf{x}}}{\lambda}) = \mathbf{b}$.

case 1: $A\tilde{\mathbf{x}} = 0$

In this case, from the definition of spark, we get $\text{spark}(A) \leq \|\tilde{\mathbf{x}}\|_0 = \text{spark}(\tilde{A})$. Now the statement in part (i) of the theorem follows by using theorems 5.4.2 and 5.4.4.

case 2:

there exist a non-zero scalar λ such that $A(\frac{\tilde{\mathbf{x}}}{\lambda}) = \mathbf{b}$. Since $\mathbf{x}^\star$ is a sparsest solution to the system $A\mathbf{x} = \mathbf{b}$, we have

$$\|\mathbf{x}^\star\|_0 \leq \left\|\frac{\tilde{\mathbf{x}}}{\lambda}\right\|_0 = \|\tilde{\mathbf{x}}\|_0 = \text{spark}(\tilde{A})$$

. Combining the above equation with theorem 5.4.2, we obtain $\|\mathbf{x}^\star\|_0 = \text{spark}(\tilde{A})$. Now, using 5.4.4, we obtain the statement in part (ii).

□

Lastly, we wish to derive another lower bound for the sparsity. So, define the matrix A_{aug} as

$$A_{\text{aug}} := [A \mid \mathbf{b}]$$

. That is, A_{aug} is the augmented matrix obtained by concatenating the vector $\mathbf{b}$ as a new column to matrix A.

Then, we have the following lower bound on sparsity

Theorem 5.4.6.

$$\text{spark}(A_{aug}) - 1 \leq \|\mathbf{x}^\star\|_0$$

Proof. Since $A\mathbf{x} = \mathbf{b}$, we have $A_{\text{aug}} \begin{bmatrix} \mathbf{x}^\star \\ -1 \end{bmatrix} = 0$. Hence,

$$\|\mathbf{x}^\star\|_0 + 1 = \left\| \begin{bmatrix} \mathbf{x}^\star \\ -1 \end{bmatrix} \right\|_0 \geq \text{spark}(A_{\text{aug}})$$

□

Remark 5.4.7. From the above theorems we can observe two things:

1. We always have a lower bound for the sparsity of a sparsest solution, namely

$$\max\{\text{spark}(\tilde{A}), \text{spark}(A_{\text{aug}}) - 1\} \leq \|\mathbf{x}^\star\|_0$$

   .

2. If $\text{spark}(\tilde{A}) \neq \text{spark}(A)$ (hence $\text{spark}(\tilde{A}) < \text{spark}(A)$), then we know exactly the sparsity of a sparsest solution. In fact, when $\text{spark}(\tilde{A}) < \text{spark}(A)$, $\|\mathbf{x}^\star\| = \text{spark}(\tilde{A})$ and is bounded from above by $\text{spark}(A)$.

Bibliography

[1] N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68:337– 403, 1950.

[2] P. Aubry and M. M. Maza. Triangular sets for solving polynomial systems: a comparative implementation of four methods. *Journal of Symbolic Computation*, 28(1):125 – 154, 1999. ISSN 0747-7171. doi: https://doi.org/10.1006/jsco.1999.0270. URL `http://www.sciencedirect.com/science/article/pii/S0747717199902705`.

[3] H. Bauschke and J. Borwein. On the convergence of von neumann's alternating projection algorithm for two sets. *Set-Valued Analysis*, 1(2):185–212, 1993.

[4] H. H. Bauschke and J. M. Borwein. On projection algorithms for solving convex feasibility problems. *SIAM review*, 38(3):367–426, 1996.

[5] A. Beck. *Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with MATLAB*, volume 19. SIAM, 2014.

[6] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009. doi: 10.1137/080716542. URL `https://doi.org/10.1137/080716542`.

[7] T. Blumensath and M. E. Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27(3):265 – 274, 2009.

ISSN 1063-5203. doi: https://doi.org/10.1016/j.acha.2009.04.002. URL http://www.sciencedirect.com/science/article/pii/S1063520309000384.

[8] T. Blumensath and M. E. Davies. Normalized iterative hard thresholding: Guaranteed stability and performance. *IEEE Journal of Selected Topics in Signal Processing*, 4(2):298–309, April 2010. ISSN 1932-4553. doi: 10.1109/JSTSP.2010.2042411.

[9] J. P. Boyle and R. L. Dykstra. A method for finding projections onto the intersection of convex sets in hilbert spaces. *Advances in Order Restricted Statistical Inference,Lecture Notes in Statistics*, 37:28– 47, 1985.

[10] J.-F. Cai, E. J. Cands, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010. doi: 10.1137/080738970. URL https://doi.org/10.1137/080738970.

[11] J.-F. Cai, T. Wang, and K. Wei. Fast and provable algorithms for spectrally sparse signal reconstruction via low-rank hankel matrix completion. *Applied and Computational Harmonic Analysis*, 2017. ISSN 1063-5203. doi: https://doi.org/10.1016/j.acha.2017.04.004. URL http://www.sciencedirect.com/science/article/pii/S1063520317300295.

[12] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717, 2009.

[13] E. J. Candes and T. Tao. Decoding by linear programming. *IEEE transactions on information theory*, 51(12):4203–4215, 2005.

[14] E. J. Candès and T. Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5):2053–2080, 2010.

[15] E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509, 2006.

[16] E. J. Candes, M. B. Wakin, and S. P. Boyd. Enhancing sparsity by reweighted 1 minimization. *Journal of Fourier analysis and applications*, 14(5):877–905, 2008.

[17] E. J. Candès et al. Compressive sampling. In *Proceedings of the international congress of mathematicians*, volume 3, pages 1433–1452. Madrid, Spain, 2006.

[18] C. Chen and M. Moreno Maza. Algorithms for computing triangular decompositions of polynomial systems. In *Proceedings of the 36th international symposium on Symbolic and algebraic computation*, pages 83–90. ACM, 2011.

[19] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM review*, 43(1):129–159, 2001.

[20] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk. Iteratively reweighted least squares minimization for sparse recovery. *Communications on Pure and Applied Mathematics*, 63(1):1–38, 2010.

[21] F. Deutsch and H. Hundal. The rate of convergence of dykstra's cyclic projections algorithm: The polyhedral case. *Numerical Functional Analysis and Optimization*, 15(5-6):537–565, 1994.

[22] R. A. DeVore and V. N. Temlyakov. Some remarks on greedy algorithms. *Advances in computational Mathematics*, 5(1):173–187, 1996.

[23] D. L. Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

[24] D. L. Donoho, A. Maleki, and A. Montanari. Message-passing algorithms for

compressed sensing. *Proceedings of the National Academy of Sciences*, 106(45): 18914–18919, 2009.

[25] R. L. Dykstra. An algorithm for restricted least squares regression. *Journal of the American Statistical Association*, 78(384):837–842, 1983.

[26] C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

[27] F. Feppon and P. F. Lermusiaux. A geometric approach to dynamical model-order reduction. *arXiv preprint arXiv:1705.08521*, 2017.

[28] M. Fornasier, H. Rauhut, and R. Ward. Low-rank matrix recovery via iteratively reweighted least squares minimization. *SIAM Journal on Optimization*, 21(4): 1614–1640, 2011.

[29] S. Foucart. Hard thresholding pursuit: an algorithm for compressive sensing. *SIAM Journal on Numerical Analysis*, 49(6):2543–2563, 2011.

[30] S. Foucart and M.-J. Lai. Sparsest solutions of underdetermined linear systems via q-minimization for 0¡ q 1. *Applied and Computational Harmonic Analysis*, 26(3):395–407, 2009.

[31] S. Foucart and H. Rauhut. *A mathematical introduction to compressive sensing*, volume 1. Birkhäuser Basel, 2013.

[32] W. Fulton. *Intersection theory*, volume 2. Springer Science & Business Media, 2013.

[33] P. Gong, C. Zhang, Z. Lu, J. Huang, and J. Ye. A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems. In *International Conference on Machine Learning*, pages 37–45, 2013.

[34] L. Gubin, B. Polyak, and E. Raik. The method of projections for finding the common point of convex sets. *USSR Computational Mathematics and Mathematical Physics*, 7(6):1–24, 1967.

[35] E. T. Hale, W. Yin, and Y. Zhang. Fixed-point continuation for \ell_1-minimization: Methodology and convergence. *SIAM Journal on Optimization*, 19(3):1107–1130, 2008.

[36] J. Harris. *Algebraic geometry: a first course*, volume 133. Springer Science & Business Media, 2013.

[37] R. Hartshorne. *Algebraic geometry*, volume 52. Springer Science & Business Media, 2013.

[38] H. S. Hundal. An alternating projection that does not converge in norm. *Nonlinear Analysis: Theory, Methods & Applications*, 57(1):35–61, 2004.

[39] P. Jain, R. Meka, and I. S. Dhillon. Guaranteed rank minimization via singular value projection. In *Advances in Neural Information Processing Systems*, pages 937–945, 2010.

[40] X. Jiang, Z. Zhong, X. Liu, and H. C. So. Robust matrix completion via alternating projection. *IEEE Signal Processing Letters*, 24(5):579–583, 2017.

[41] R. Kachurovskii. Monotone operators and convex functionals. *Uspekhi Matematicheskikh Nauk*, 15(4):213–215, 1960.

[42] I. Kozlov and A. Petukhov. Sparse solutions of underdetermined linear systems. In *Handbook of Geomathematics*, pages 1243–1259. Springer, 2010.

[43] M.-J. Lai and J. Wang. An unconstrained \ell_q minimization with 0q\leq1 for sparse solution of underdetermined linear systems. *SIAM Journal on Optimization*, 21(1):82–101, 2011.

[44] M.-J. Lai and W. Yin. Augmented \ell_1 and nuclear-norm models with a globally linearly convergent algorithm. *SIAM Journal on Imaging Sciences*, 6(2):1059–1091, 2013.

[45] M.-J. Lai, Y. Xu, and W. Yin. Improved iteratively reweighted least squares for unconstrained smoothed \ell_q minimization. *SIAM Journal on Numerical Analysis*, 51(2):927–957, 2013.

[46] S. Ma, D. Goldfarb, and L. Chen. Fixed point and bregman iterative methods for matrix rank minimization. *Mathematical Programming*, 128(1):321–353, 2011.

[47] L. Mirsky. Symmetric gauge functions and unitarily invariant norms. *The quarterly journal of mathematics*, 11(1):50–59, 1960.

[48] K. Mohan and M. Fazel. Iterative reweighted algorithms for matrix rank minimization. *Journal of Machine Learning Research*, 13(Nov):3441–3473, 2012.

[49] D. Mumford. *The red book of varieties and schemes: includes the Michigan lectures (1974) on curves and their Jacobians*, volume 1358. Springer Science & Business Media, 1999.

[50] Netflix. The netflix prize, 2006. URL `http://www.netflixprize.com/`.

[51] C. J. Pang. Accelerating the alternating projection algorithm for the case of affine subspaces using supporting hyperplanes. *Linear Algebra and its Applications*, 469:419–439, 2015.

[52] C. Perkins. A convergence analysis of dykstra's algorithm for polyhedral sets. *SIAM Journal on Numerical Analysis*, 40(2):792–804, 2002.

[53] R. Phelps. Convex sets and nearest points. *Proceedings of the American Mathematical Society*, 8(4):790–797, 1957.

[54] R. Phelps. Convex sets and nearest points. ii. *Proceedings of the American Mathematical Society*, 9(6):867–873, 1958.

[55] S. Rangan. Generalized approximate message passing for estimation with random linear mixing. In *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*, pages 2168–2172. IEEE, 2011.

[56] B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3): 471–501, 2010.

[57] I. R. Shafarevich and K. A. Hirsch. *Basic algebraic geometry*, volume 2. Springer, 1994.

[58] J. Tanner and K. Wei. Normalized iterative hard thresholding for matrix completion. *SIAM Journal on Scientific Computing*, 35(5):S104–S125, 2013.

[59] J. Tanner and K. Wei. Low rank matrix completion by alternating steepest descent methods. *Applied and Computational Harmonic Analysis*, 40(2):417–429, 2016.

[60] M. Tao and X. Yuan. Recovering low-rank and sparse components of matrices from incomplete and noisy observations. *SIAM Journal on Optimization*, 21(1): 57–81, 2011.

[61] T. Tao. An uncertainty principle for cyclic groups of prime order. *arXiv preprint math/0308286*, 2003.

[62] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[63] K.-C. Toh and S. Yun. An accelerated proximal gradient algorithm for nuclear

norm regularized linear least squares problems. *Pacific Journal of Optimization*, 6(615-640):15, 2010.

[64] J. A. Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Transactions on Information theory*, 50(10):2231–2242, 2004.

[65] B. Vandereycken. Low-rank matrix completion by riemannian optimization. *SIAM Journal on Optimization*, 23(2):1214–1236, 2013.

[66] J. Von Neumann. *Functional operators*, volume 2. Princeton University Press, 1955.

[67] J. Von Neumann. *Functional Operators (AM-22), Volume 2: The Geometry of Orthogonal Spaces.(AM-22)*, volume 2. Princeton University Press, 2016.

[68] J. Wang, J. Zhou, P. Wonka, and J. Ye. Lasso screening rules via dual polytope projection. In *Advances in Neural Information Processing Systems*, pages 1070–1078, 2013.

[69] Z. Wang, M.-J. Lai, Z. Lu, W. Fan, H. Davulcu, and J. Ye. Orthogonal rank-one matrix pursuit for low rank matrix completion. *SIAM Journal on Scientific Computing*, 37(1):A488–A514, 2015.

[70] K. Wei, J.-F. Cai, T. F. Chan, and S. Leung. Guarantees of riemannian optimization for low rank matrix completion. *arXiv preprint arXiv:1603.06610*, 2016.

[71] Z. Wen, W. Yin, and Y. Zhang. Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm. *Mathematical Programming Computation*, pages 1–29, 2012.

[72] H. Weyl. Das asymptotische verteilungsgesetz der eigenwerte linearer partieller differentialgleichungen (mit einer anwendung auf die theorie der hohlraumstrahlung). *Mathematische Annalen*, 71(4):441–479, 1912.

[73] J. Yang and X. Yuan. Linearized augmented lagrangian and alternating direction methods for nuclear norm minimization. *Mathematics of computation*, 82(281): 301–329, 2013.

[74] O. Zariski. On the purity of the branch locus of algebraic functions. *Proceedings of the National Academy of Sciences*, 44(8):791–796, 1958.

Appendices

Appendix A

Iteratively Reweighted Least Squares Minimisation

We will describe the IRLSM (Iteratively Reweighted Least Squares Minimisation) algorithm for matrix completion devised by M. Fornasier, H. Rauhut and R. Ward in 2011, [28]

Definition A.0.1. Define the *ϵ-stabilization* $\mathbf{X}_\epsilon$ of a matrix $\mathbf{X} \in \mathbb{R}^{n\times n}$ as

$$\mathbf{X}_\epsilon = \mathbf{U}\Sigma_\epsilon\mathbf{V}^\star$$

where $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^\star$ is the *singular value decomposition* of matrix $\mathbf{X}$, $\Sigma = \text{diag}(\sigma_1, \cdots, \sigma_n)$ and $\Sigma_\epsilon = \text{diag}(\max\{\sigma_j, \epsilon\})$. The singular values below the threshold ϵ is increased to the threshold.

The authors have proved the convergence of the algorithm conditional on the sampling operator satisfying certain restricted isometric property. Before we state their result (theorem A.0.3), we need the following definition:

Definition A.0.2 (Recht et al. [56])**.** Let $\mathscr{S} : \mathbb{R}^{n\times p} \to \mathbb{C}^m$ be a linear map. For every integer k, with $1 \leq k \leq n$, define the $k-$restricted isometry constant $\delta_k = \delta_k(\mathscr{S}) > 0$

Algorithm 6: IRLSM algorithm for low-rank matrix recovery:

Data: $\mathscr{S}(\mathbf{M})$ obtained from sampling a rank r matrix $\mathbf{M}$ using the sampling operator $\mathscr{S}$.

Result: $\mathbf{X}^l$, a close approximation of the original rank r matrix $\mathbf{M}$

Initialize $\mathbf{W}^0 = I \in \mathbb{R}^{n\times n}$. Set $\epsilon_0 := 1, K > r$ and $\gamma > 0$

repeat

Step 1: $\mathbf{X}^l := \arg\min_{\mathscr{S}(\mathbf{X})=\mathscr{S}(\mathbf{M})} \left\|(\mathbf{W}^{l-1})^{1/2}\mathbf{X}\right\|_F^2$ where $\mathscr{S}$ is the sampling operator

Step 2: $\epsilon_l := \min\{\epsilon_{l-1}, \gamma\sigma_{K+1}(\mathbf{X}^l)\}$

Step 3: Compute $\mathbf{U}^l \in \mathbb{R}^{n\times n}$ and $\Sigma^l = \mathrm{diag}(\sigma_1^l, \cdots, \sigma_n^l) \in \mathbb{R}^{n\times n}$ for which

$$\mathbf{X}^l(\mathbf{X}^l)^\star = \mathbf{U}^l(\Sigma^l)^2(\mathbf{U}^l)^\star$$

Step 4: $\mathbf{W}_l = \mathbf{U}^l(\Sigma_{\epsilon l}^l)^{-1}(\mathbf{U}^l)^\star$ where Σ_ϵ denotes the ϵ-stabilization of the matrix, see definition A.0.1.

Step 5: The algorithm stop if $\epsilon = 0$; in this case, define $\mathbf{X}^j := \mathbf{X}^l$ for $j > l$. In general, the algorithm generates an infinite sequence $(\mathbf{X}^l)_{l\in\mathbf{N}}$

until $\left\|\mathbf{X}^{l+1} - \mathbf{X}^l\right\| < \epsilon$;

to be the smallest number such that

$$(1-\delta_k)\left\|\mathbf{X}\right\|_F^2 \leq \left\|\mathscr{S}(\mathbf{X})\right\|_{l_2^m}^2 \leq (1+\delta_k)\left\|\mathbf{X}\right\|_F^2$$

holds for all rank$-k$ matrices $\mathbf{X}$.

Theorem A.0.3 (Fornasier et al. [28])**.** *The IRLSM algorithm with parameters $\gamma = 1/n$ and $K \in \mathbf{N}$. Let $\mathscr{S} : \mathbb{R}^{n\times p} \to \mathbb{C}^m$ be a surjective map with restricted isometry constants (see definition A.0.2) δ_{3K}, δ_{4K} satisfying $\eta = \frac{\sqrt{2}\delta_{4K}}{1-\delta_{3K}} < 1 - \frac{2}{K-2}$. Then, if there exist a rank r matrix $\mathbf{X}$ satisfying $\mathscr{S} = \mathscr{M}$, where $\mathscr{M} \in \mathbb{C}^m$, with $r < K - \frac{2\eta}{1-\eta}$, the sequence $(\mathbf{X}^l)_{l\in\mathbf{N}}$ converges to $\mathbf{X}$.*

Appendix B

Singular Value Thresholding Algorithm for Matrix Completion

In this section, we will describe the SVT (Singular Value Thresholding) algorithm for matrix completion devised by J. Cai, E. Candès and Z. Shen in 2010, [10].

The authors have proved that the SVT algorithm converges to a solution of the relaxed-version of the matrix completion problem described below:

$$\min_{\mathbf{X}\in\mathbb{R}^{n_1\times n_2}} \quad \tau\|\mathbf{X}\|_\star + \frac{1}{2}\|\mathbf{X}\|_F^2 \quad \text{subject to} \quad \mathcal{P}_\Omega(\mathbf{X}) = \mathcal{P}_\Omega(\mathbf{M}), \tag{B.0.1}$$

A definition is in order before we state their algorithm.

Definition B.0.1. Define the shrinkage operation $\text{Shrink}(\mathbf{X}, \tau)$ of a matrix $\mathbf{X} \in \mathbb{R}^{n\times n}$ as

$$\text{Shrink}(\mathbf{X}, \tau) = \mathbf{U}\mathcal{D}_\tau(\Sigma)\mathbf{V}^\star$$

where $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^\star$ is the *singular value decomposition* of matrix $\mathbf{X}$, $\Sigma = \text{diag}(\sigma_1, \cdots, \sigma_n)$ and $\mathcal{D}_\tau(\Sigma) = \text{diag}(\{(\sigma_j - \tau)_+\})$. t_+ is the positive part of t, that is $t_+ = \max(0, t)$. The shrinkage operation effectively applies an soft-threshold to the singular values of

the matrix $\mathbf{X}$.

Algorithm 7: SVT algorithm for low-rank matrix recovery:

Data: $\mathcal{P}_\Omega(\mathbf{M})$ obtained from sampling a rank r matrix $\mathbf{M}$ at positions Ω
Result: $\mathbf{X}^k$, a close approximation of the original rank r matrix $\mathbf{M}$
Initialize $\mathbf{Y}^0 = 0 \in \mathbb{R}^{n_1 \times n_2}$. Set $\tau > 0$ and a sequence of step sizes $\{\delta_k\}_{k \geq 1}$
repeat
  Step 1: $\mathbf{X}^k := \text{Shrink}(\mathbf{Y}^{k-1}, \tau)$
  Step 2: $\mathbf{Y}^k = \mathbf{Y}^{k-1} + \delta_k \mathcal{P}_\Omega(M - \mathbf{X}^k)$
until a stopping criterion is reached. For example, $\left\|\mathbf{X}^{k+1} - \mathbf{X}^k\right\| < \epsilon$;

Theorem B.0.2 (Cai et al. [10])**.** *If* $0 < \inf \delta_k \leq \sup \delta_k < 2$*, then the iterates* $\{\mathbf{X}^k\}$ *obtained using SVT algorithm converges to the unique solution of the optimization problem B.0.1.*

Remark B.0.3. Please note that δ_k in the theorem B.0.2 are step sizes and not to be confused with the restricted isometry constants described in the Appendix A.