ABSTRACT

Title of dissertation: EXPLOITING DATA-DEPENDENT STRUCTURE FOR IMPROVING SENSOR ACQUISITION AND INTEGRATION

Alexander Cloninger, Doctor of Philosophy, 2014

Dissertation directed by: Professor Wojciech Czaja
Professor John J. Benedetto
Department of Mathematics

This thesis deals with two approaches to building efficient representations of data. The first is a study of compressive sensing and improved data acquisition. We outline the development of the theory, and proceed into its uses in matrix completion problems via convex optimization. The aim of this research is to prove that a general class of measurement operators, *bounded norm Parseval frames*, satisfy the necessary conditions for random subsampling and reconstruction. We then demonstrate an example of this theory in solving 2-dimensional Fredholm integrals with partial measurements. This has large ramifications in improved acquisition of nuclear magnetic resonance spectra, for which we give several examples.

The second part of this thesis studies the Laplacian Eigenmaps (LE) algorithm and its uses in data fusion. In particular, we build a natural approximate inversion algorithm for LE embeddings using L^1 regularization and MDS embedding techniques. We show how this inversion, combined with feature space rotation, leads to a novel form of data reconstruction and inpainting using a priori information. We

demonstrate this method on hyperspectral imagery and LIDAR.

We also aim to understand and characterize the embeddings the LE algorithm gives. To this end, we characterize the order in which eigenvectors of a disjoint graph emerge and the support of those eigenvectors. We then extend this characterization to weakly connected graphs with clusters of differing sizes, utilizing the theory of invariant subspace perturbations and proving some novel results.

EXPLOITING DATA-DEPENDENT STRUCTURE FOR IMPROVING SENSOR ACQUISITION AND INTEGRATION

by

Alexander Cloninger

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2014

Advisory Committee:
Professor Wojciech Czaja/Chair
Professor John J. Benedetto/Co-Chair
Professor Kasso Okoudjou
Professor Yuan Liao
Professor Rama Chellappa

Dedication

To my parents,

for giving me everything I have

and never letting me settle

Acknowledgments

I owe my gratitude to the many people that have made this thesis, and my mathematical career, possible. As the saying goes, it takes a village.

I have been luck enough to have an incredibly supportive network of advisers in the Norbert Wiener Center. First, I would like to thank my adviser Wojciech Czaja, for providing me with a countless number of insights, both mathematical and otherwise. His ability to move seamlessly between theory and application is one that I admire and attempt to emulate. During the last three years, Wojtek has challenged me when needed, yet remained accommodating and nonintrusive enough to allow me to forge my own path.

I am also immensely grateful to my co-adviser John Benedetto. Prior to meeting John, I was fairly certain I was going to leave grad school after getting my M.Sc. But the fascinating material in those first courses I took from him and the rapport we formed were the driving force in my decision to stay, and for that I am eternally grateful. Since that time, like the godfather of the Norbert Wiener Center, John has guided me through my schooling with knowledge and kindness, and perfectly prepared me for the next stage of my life.

I would also like to thank Kasso Okoudjou for his invaluable conversations and lessons about graph theory, as well as being another warm and supportive member of the Norbert Wiener Center family. I would like to thank Professor Rama Chellappa for introducing me to the academic world that exists outside of the math department, and for aiding me in laying the foundations of my academic career. I would also like

to thank Professor Yuan Liao for taking the time to be involved in the culmination of my time at University of Maryland.

Many thanks go out to Alverda McCoy for ensuring I didn't lose out on my dreams due to one of the many bureaucratic nightmares that tried to stand in the way, and to Professor Konstantina Trivisa for being the most positive person I've ever known. My time on the AMSC Student Council would have been a complete daze without the help and support both of you showed me.

No support network is complete without the friends and colleagues that keep you sane. My perpetual office mates (Travis, Tim, Bryant, Mikey, Jake, and Zsolt) contributed immensely to my intellectual growth and happiness, while serving as a necessary diversion from work. Thanks to all those NWC students that made hours in the lab enjoyable and productive (James, Chae, Matt, Matt, Paul, Clare, Ariel, Rongrong, Xuemei, and Ben), and thanks to all of the UMD Math department students that have made my time in the department entertaining.

To Kevin, Tyler, Joanna, Garrett, Carolina, Qian, and Karamatou, for making my life be about more than just work. You all have, at varying times, served as my emotional support and respite in ways I cannot begin to describe. I could not have gotten here without each and every one of you. And to the Ultimate group, for making me leave my computer from time to time and get outside.

Lastly, to the loving, intelligent, hardworking, and caring Kristin Larson. You have been my inspiration, my personal motivator, my sounding board, and my sensibility. You take all the worry out of my mind.

Table of Contents

List of Figures

Chapter 1: Introduction

Advancements in sensor construction, along with a reduction in production cost, has led to a deluge of data. Processing and interpreting this information has become a driving force in mathematics in recent years, and opened the door to many new fields of application of these mathematical tools. Most importantly, it has led to a shift in the way we approach data, from attempts to learn the data using *a priori* transforms (e.g., wavelets, sparse representations, filter banks), to transforms that are constructed with data-dependent structure (e.g., data-dependent graphs, manifold learning, dimension reduction).

Harmonic analysis, in its most general form, deals with the mathematics of efficient representations for signals and data. This thesis is the examination of how both classical and modern techniques in harmonic analysis can be utilized for learning data dependent structure of a system. In particular, this focus on learning structure splits into two branches: exploiting structure and sparsity to reduce sampling requirements prior to collection, and exploiting data structure to extract and fuse underlying parameters after collection has occurred.

The former branch of research ties in with the emerging field of compressive sensing, in which one exploits sparsity in the signal and incoherence in the mea-

surements to reduce the number of samples needed. The theory utilizes convex relaxation of combinatorial optimization schemes to reconstruct the signal, but is highly dependent on the structure of the measurement operator. Proving that a new class of measurement operators satisfies the necessary conditions for reconstruction is an active and critical area of research.

The latter branch of research is in graph and operator theoretic approaches to pattern recognition and machine learning. We focus on assembling multiple heterogeneous or homogeneous data sources into a common fused end product, for the purpose of improving knowledge compared to a single sensor. When data arrives from heterogeneous modalities and the readings are temporally or spatially separated, this fusion becomes highly non-trivial.

1.1 Background for Frames

This thesis' approach to data acquisition relies heavily on the theory of frame representations. Frames were originally introduced by Duffin and Schaeffer in [50]. They provide a natural generalization of orthonormal bases in Hilbert spaces. A general overview of the subject can be found in [11, 27, 34].

Definition 1.1.1. *For a Hilbert space* $\mathbb{H}$*, a set* $\{f_i\}_{i\in J} \subset \mathbb{H}$ *is a* frame if $\exists A, B > 0$ such that $\forall x \in \mathbb{H}$,

$$A\|x\|^2 \leq \sum_{i\in J} |\langle x, f_i\rangle|^2 \leq B\|x\|^2.$$

This definition serves as a generalization of orthonormal bases by relaxing

Parseval's relation

$$\|x\|^2 = \sum_i |\langle x, f_i \rangle|^2.$$

Here, instead of equality, we have a *lower frame bound* A and an *upper frame bound* B.

Definition 1.1.2. *Let $\{f_i\} \subset \mathbb{H}$ be a frame with bounds $0 < A \leq B$. Then*

1. *a* tight frame satisfies $A = B$, and
2. a *Parseval tight frame satisfies $A = B = 1$.*

Frames have the benefit of giving overcomplete representations of the function x, making them much more robust to errors and erasures than orthonormal bases [28, 34, 77]. This makes frames ideal for problems involving subsampled measurements, a property that we shall take advantage of in Section 2.3.

Definition 1.1.3. *Let $\{f_i\}_{i \in J}$ be a frame for $\mathbb{H}$. Then several properties associated with the frame are*

1. *the* analysis operator $A : \mathbb{H} \to \ell_2$ with $x \mapsto \{\langle x, f_i \rangle\}_{i \in J}$,
2. the *synthesis operator* $A^* : \ell_2 \to \mathbb{H}$ *with* $\{a_j\}_{j \in J} \mapsto \sum_{i \in J} a_i f_i$, *and*
3. *the* frame operator $S = A^* A : \mathbb{H} \to \mathbb{H}$ with $x \mapsto \sum_{i \in J} \langle x, f_i \rangle f_i$.

Since $\langle Sx, x \rangle = \sum_{i \in J} |\langle x, f_i \rangle|^2$, we know $A \cdot Id_{\mathbb{H}} \leq S \leq B \cdot Id_{\mathbb{H}}$, so we have a reconstruction formula

$$x = \sum_{i \in J} \langle x, S^{-1} f_i \rangle f_i = \sum_{i \in J} \langle x, f_i \rangle S^{-1} f_i.$$

Lemma 1.1.4. *Let $\{f_i\}_{i\in J} \subset \mathbb{H}$ be a tight frame with bound A. Then the frame operator S associated with the frame satisfies*

$$S = A \cdot Id_{\mathbb{H}}.$$

For the rest of this chapter, we shall focus specifically on tight frames and their associated properties. Specifically, we focus on Parseval tight frames whose frame elements have bounded norm, and our Hilbert space shall be the set of matrices $\mathbb{C}^{d\times d}$.

Definition 1.1.5. *A* bounded norm Parseval tight frame with incoherence μ *is a Parseval tight frame $\{\phi_j\}_{j\in J}$ on $\mathbb{C}^{d\times d}$ that also satisfies*

$$\|\phi_j\|^2 \leq \mu \frac{d}{|J|}, \quad \forall j \in J. \tag{1.1}$$

This definition generalizes the same type of bound for bases from [63]. Note that, in the case of $\{\phi_j\}_{j\in J}$ being a basis, $|J| = d^2$, reducing the bound in (1.1) to $\|\phi_j\|^2 \leq \mu/d$, c.f., [63].

This definition also generalizes the notion of *finite unit norm tight frames* from [12]. These are tight frames whose frame elements all have norm 1. Definition 1.1.5 relaxes the condition that every element has norm 1, and instead simply requires the energy of the largest norm frame element to be bounded.

1.2 Laplacian Eigenmaps

When dealing with the post-processing of collected data to learn inherent structure, we shall be utilizing the theory of dimension reduction, and specifically

Laplacian Eigenmaps (LE). The purpose of Laplacian Eigenmaps, as with all non linear dimensionality reduction techniques, is to create a mapping $\phi : \mathbb{R}^d \to \mathbb{R}^m$ where m is the inherent dimension of the underlying data. Intuitively, the embedding relies on understanding the Laplace-Beltrami operator on the manifold. The manifold is approximated by an adjacency graph on the data, and Laplace-Beltrami operator is approximated by a weighted Laplacian of the adjacency matrix.

Let $\Omega = \{x_1, ...x_n\} \subset \mathbb{R}^d$ be a set of training points, otherwise known as the *data space*. We have a positive, symmetric kernel $k : \Omega \times \Omega \to \mathbb{R}$ that encodes relationships between any two points in Ω. An example of such a kernel (and the most common for Laplacian Eigenmaps) is

$$k(x, y) = e^{-\frac{\|x-y\|_2^2}{2\sigma^2}}, \tag{1.2}$$

where σ is a parameter that controls the width of the Gaussian.

Using k as a similarity metric, each $x \in \Omega$ now has a neighborhood $\mathcal{N}(x) \subset \Omega$ of nearest neighbors to x. This neighborhood can be constructed by taking a fixed number of nearest neighbors of x, or by considering all points y such that $k(x, y) > 1 - \epsilon$.

Now we shall construct a graph $\Gamma \equiv (\Omega, w)$, where an edge between x_i and x_j has weight

$$K_{i,j} = \begin{cases} k(x_i, x_j) & : \quad x_j \in \mathcal{N}(x_i) \ \& \ i \neq j, \\ 0 & : \quad \text{otherwise.} \end{cases}$$

Note that we require $K = [K_{i,j}]_{i,j=1}^n$ to be symmetric in order to guarantee its eigenvectors are orthonormal, though whether K is symmetric depends on how the neighborhoods are generated. If K is not symmetric, simply define the weights

$\widetilde{K}_{i,j} = \max(K_{i,j}, K_{j,i})$. Also define the diagonal matrix D such that

$$D_{i,i} = \sum_{j=1}^{n} K_{i,j}.$$

We now define the *graph Laplacian* as

$$L = D - K.$$

Since K is symmetric, L is also symmetric. Also, L is semi-positive definite.

Finally, the m dimensional mapping for the LE embedding $\phi : \Omega \to \mathbb{R}^m$ comes from solving the normalized eigenvalue problem

$$D^{-\frac{1}{2}} L D^{-\frac{1}{2}} v = \lambda v, \tag{1.3}$$

for the m smallest non-zero eigenvalues. The m eigenvectors corresponding to those eigenvalues are used to form the embedding into $\mathbb{R}^m$. In other words, if $\{v_i\}_{i=1}^m$ are the eigenvectors associated with eigenvalues $\{\lambda_i\}_{i=1}^m$, then

$$\phi(x_i) = (v_1(i), ..., v_m(i)).$$

It is worth mentioning that $\{v_i\}_{i=1}^m$ are orthogonal, due to the fact that $D^{-\frac{1}{2}} L D^{-\frac{1}{2}}$ is symmetric. Also, ϕ preserves the local geometry of Ω, as it maps close points in Ω to close points in $\mathbb{R}^m$. This embedding into $\mathbb{R}^m$ shall be referred to as the *feature space*.

Laplacian Eigenmaps has been studied in a variety of contexts since its introduction by [9]. The original idea of representing smooth manifolds with eigenfunctions originated in [14], and has been extended by [74]. LE has been combined with the compressive sensing literature that will arise in Chapter 2 in [66]. Also,

Schrödinger eigenmaps generalized LE by adding a potential term to the graph, c.f., [40, 51].

1.3 Outline of Results

In Chapter 2, we introduce the ideas of compressive sensing and matrix completion. We then demonstrate that for bounded norm Parseval tight frames, which oversample the data, satisfy the necessary conditions to be a "nearly orthogonal" operator (specifically it satisfies the Restricted Isometry Property). We go on to show that this implies one can establish a minimum number of measurements necessary to recover a signal measured by such frames. This leads to improved signal acquisition from a limited number of measurements.

In Chapter 3, we introduce approximate inversion of Laplacian Eigenmaps. We develop a non-linear approach to the problem using tools from compressive sensing and optimization literature. This approximate inversion, otherwise known as a *pre-image*, is common in denoising literature. We shall use it in Chapter 6 as a novel approach to data fusion and reconstruction.

Chapter 4 deals with the theoretical underpinnings of Laplacian Eigenmaps and other graph based dimension reduction techniques. Under simple assumptions on data clusters, we prove results related to the order in which features within eigenvectors emerge. We also demonstrate that, in realistic scenarios, detection of anomalous (i.e., small) feature clusters becomes much more difficult than previously believed.

The focus of Chapter 5 is on an application of the frame operator results from Chapter 2 to nuclear magnetic resonance (NMR) spectroscopy. Here we demonstrate that NMR measurements (Laplace transform type measurements) can be described using bounded norm Parseval tight frames, and that reconstruction of a compressed data matrix can be achieved using matrix completion. This leads to a significant reduction of the number of measurements necessary to attain the desired resolution of the underlying solution, which correlates with a reduction in the amount of time necessary to acquire an NMR scan.

Finally, Chapter 6 uses the results from Chapter 3 to present a novel approach to data fusion and integration. After embedding each data source using Laplacian Eigenmaps, one can utilize the results of [35] to rotate both data sources into a common space. And after applying our pre-imaging algorithm to the new set of data, one is able to examine both data sets in a common, easy to visualize data space. We demonstrate the effectiveness of this algorithm on remote sensing data, including hyperspectral imagery (HSI) and LIDAR imagery.

Chapter 2: Improved Data Acquisition from Tight Frame Measurements

2.1 Sparsity, Compressibility, and Compressive Sensing

The Shannon-Nyquist sampling theorem states that continuous time band-limited signals can be exactly recovered from a set of uniformly spaced samples taken at a rate of twice the highest frequency present in the signal. Unfortunately, in many applications, the resulting Nyquist rate is far too large, making sample acquisition, or even sample storage, impossible.

To deal with the problem of data storage, we rely heavily on *compression*. The aim of compression is finding a concise representation of the data without much distortion. The most common version of compression is *transform coding*. in which one utilizes a family of bases or frames that are know a priori to generate an accurate *sparse* representation for signals in a class of interest.

Sparse representations refer to representing a signal f of length N with $K \ll N$ non-zero coefficients. In other words, let $\Phi = [\phi_1, ..., \phi_N]$ represent the transform being used for compression (eg. DCT, wavelet, shearlet, etc.), and $\mathcal{I}$ be the indices of the largest K elements of Φf. Then Φ generates an accurate sparse representation

of f if

$$\|f - \sum_{i \in \mathcal{I}} \langle f, \phi_i \rangle \phi_i\| < \epsilon.$$

In this case, f is also referred to as being *compressible*. This method of sparse approximation is commonly used in many schemes, including JPEG, MPEG, and MP3.

Dealing with the problem of data acquisition for large signals has exploded in popularity in the past ten years, following initial results from [23, 47, 87] regarding the question of magnetic resonance imaging. These results show that, if a signal is compressible under a certain class of transforms, then it was unnecessary to collect all N measurements, only to throw away all but K in the compression step.

More formally, let the object of interest be a signal $f \in \mathbb{C}^N$. If one selects a set $\Omega \subset \mathbb{Z}_N$ and an orthogonal matrix U, we define our measurement y to be

$$y = U_\Omega f,$$

where U_Ω is the $m \times n$ matrix consisting of the rows of U indexed by Ω. The goal is to recover f from this subsampled set of measurements.

Clearly, if $|\Omega| = N$, this problem would be trivial. The interesting case is when $|\Omega| = m \ll N$. This means that U_Ω is "fat" (has a high dimensional null space). In general, this is an unsolvable problem. However, it turns out that if f is sparse, it can be recovered exactly from y.

A simple approach would be to attempt to recover the sparse vector $f \in \mathbb{C}^N$ by solving the combinatorial optimization problem,

$$\min_{g \in \mathbb{C}^N} \|g\|_0, \quad U_\Omega g = U_\Omega f,$$

where $\|g\|_0$ is the number of nonzero terms in g. However, this problem is infeasible for even small N [23].

In order to avoid this level of computational complexity, one can instead attempt to recover f by solving

$$\min_{g \in \mathbb{C}^N} \|g\|_1, \quad U_\Omega g = U_\Omega f.$$

This was originally shown by Donoho in [48], but only for $U_\Omega = \mathcal{F}_\Omega$ (the discrete Fourier transform matrix). Also, if f is supported on a set $T \subset \mathbb{Z}_N$, [48] only guarantees reconstruction if

$$2|T|(N - |\Omega|) < N.$$

This result was extended in [23] for measurements taken in Fourier space. Given a signal $f \in \mathbb{C}^N$, and observed coefficients of $\hat{f} \in \mathbb{C}^N$ on some set $\Omega \subset \mathbb{Z}_N$, we analyze the problem of when it is possible to recover f from those observations by solving

$$\min_{g \in \mathbb{C}^N} \|g\|_{l_1}, \quad \hat{g}|_\Omega = \hat{f}|_\Omega, \tag{2.1}$$

where $\hat{f}|_\Omega$ is the restriction of $\hat{f}$ to the observed set Ω.

Theorem 2.1.1. *(Theorem 1.3, [23]) Let $f \in \mathbb{C}^N$ be some discrete signal with support set T, where T is unknown. Choose Ω of size $|\Omega| = m$ uniformly at random. For a given accuracy parameter M, if*

$$|T| \leq C_M (\log N)^{-1} |\Omega|, \tag{2.2}$$

then with probability at least $1 - \mathrm{O}\left(N^{-M}\right)$, the minimizer to problem (2.1) is unique and equal to f.

Theorem 2.1.1, while extremely important, is a very specific case of a much more general idea. Say we have any NxN orthogonal matrix U such that $U^*U = N \cdot I$. You observe $y = U_\Omega f$ on some set $\Omega \subset \mathbb{Z}_N$. You can attempt to recover some sparse $f \in \mathbb{C}^N$ by solving

$$\min_{g \in \mathbb{C}^N} \|g\|_1, \quad U_\Omega g = U_\Omega f. \tag{2.3}$$

As with Theorem 2.1.1, there are certain restrictions that will guarantee reconstruction with high probability. To make this concrete, we recall the following result from [22].

Theorem 2.1.2. *(Theorem 1.1, [22]) Fix some set $T \subset \mathbb{Z}_N$. Let U be an $N \times N$ orthogonal matrix with $\mu(U) = \max_{i,j} |U_{i,j}|$. Choose Ω of size $|\Omega| = m$, and a sign sequence z on T uniformly at random. If*

$$|\Omega| \geq C_0 |T| \mu^2(U) \log(N/\delta) \quad \textit{and} \quad |\Omega| \geq C_0' \log^2(N/\delta), \tag{2.4}$$

then with probability at least $1 - \delta$, every signal $f \in \mathbb{R}^N$ supported on T with $\imath^* sgn(f) = z$ can be recovered from $y = U_\Omega f$ by solving (2.3).

Remark: One application of this generalized U would be if U could be decomposed into the product of two matrices. In Theorem 2.1.1, we chose our *measurement basis* to be the Fourier domain, and we choose our *sparsity basis* to be the standard basis. Suppose instead $x \in \mathbb{C}^N$ and we observe m measurements of the form $y = \Phi x$ (call Φ our *measurement basis*). x may not be sparse in the standard basis, but maybe we can express $x = \Psi f$ for some sparse vector f (call Ψ our *sparsity basis*).

Then (2.3) can be used to solve this problem if

$$U = \Phi\Psi, \quad \Psi^*\Psi = I, \quad \Phi^*\Phi = N \cdot I.$$

Once we find some $g^\#$ which minimizes (2.3), the best estimate for x would be

$$x^\# = \Psi g^\#.$$

In the context of sparsity and measurement bases, $\mu(U)$ in Theorem 2.1.2 takes on added meaning. Since $U = \Phi\Psi$, we can see that

$$\mu(\Phi\Psi) = \max_{i,j} | \langle \phi_i, \psi_j \rangle |.$$

Clearly $\frac{1}{\sqrt{N}} \leq \mu(U) \leq 1$. If μ is close to $\frac{1}{\sqrt{N}}$, this means each of the measurement vectors (rows of Φ) are very "spread out" in the Ψ domain. This is another way of saying these bases are *mutually incoherent*, which reduces the number of measurements m required for Theorem 2.1.2.

Following these fundamental results, the study of reconstructing sparse signals from underdetermined measurements has expanded in a variety of directions, from sufficient conditions for measurements [26], to improved ideas of sparsity [1, 7], to phase reconstruction [24], to algorithms for L^1 optimization techniques [59]. A number of application areas have also benefited from compressive sensing, including single pixel imaging [8], radar [93], and background subtraction [29].

The direction we shall take this research in falls under the area of matrix completion. Specifically, we shall consider the problem of recovering a low rank matrix from some underdetermined set of measurements, c.f., [21, 25, 63].

2.2 Background for Matrix Completion

An $n \times n$ matrix X that is of rank r requires approximately nr parameters to be completely specified. If $r \ll n$, then X is seen as being compressible, as the number of parameters needed to specify it is much less than its n^2 entries. It is less clear how to recover X from a limited number of coefficients efficiently. But the results of [21] showed it is possible to recover X from, up to a constant, $nr \cdot \log(n)$ measurements by employing a simple optimization problem. Also, the types of measurements we utilize in this chapter, operator bases with bounded norm, originated from quantum state tomography [64].

Let $X \in \mathbb{R}^{s_1 \times s_2}$ be a rank r matrix of interest that is measured by

$$M_i = \langle \phi_i, X \rangle + e_i, \quad M_i \in \mathbb{R},$$

where $\{\phi_i\}_{i \in J}$ is a set measurement modalities with $|J| = N$, and e_i is a noise term such that $\|e\|_2 \leq \epsilon$. Let $\Omega = \{i_1, ..., i_m\} \subset \{1, ..., N\}$, $m < N$, be the set of measurements M_i that are observed. We define a masking operator as

$$\begin{aligned} \mathcal{R}_\Omega : \mathbb{R}^{s_1 \times s_2} &\to \mathbb{R}^m, \\ (\mathcal{R}_\Omega(X))_k = M_{i_k} &= \phi_{i_k}(X), \end{aligned} \tag{2.5}$$

and $M|_\Omega \in \mathbb{R}^m$ denotes the entire set of observed measurements.

The problem of matrix completion has been in the center of scientific interest and activity in recent years [16, 18, 20, 21, 24, 56, 63, 97]. The basic problem revolves around trying to recover a matrix $X \in \mathbb{R}^{s_1 \times s_2}$ from only a fraction of the N measurements taken. Without any additional assumptions, this is an ill-posed problem.

However, there have been a number of attempts to add natural assumptions to make this problem well posed. Other than assuming that X is low rank as we mentioned above, there are assumptions that X is positive definite [62, 81], or that X is a distance matrix [2], or that X is has a non-negative factorization [113]. A survey of some of these other methods can be found in [73].

Let our problem take the form

$$y = \mathcal{R}_\Omega(X) + z, \quad \|z\|_2 \leq \epsilon, \tag{2.6}$$

where z represents a noise vector that is typically white noise, though not necessarily.

For a general set of measurements $\{\phi_i\}$, when $m < N$ the problem of recovering X from y is clearly underdetermined. Even if $m > s_1 s_2$, there is no guarantee that $\text{span}\{\phi_i : i \in \Omega\} = \mathbb{R}^{s_1 \times s_2}$. And considering that the measurements are noisy, stable reconstruction becomes even more of an issue. This makes the problem of recovering X from $\mathcal{R}_\Omega(X)$, in a general setting, unattainable.

The naive way to proceed would be to exploit the fact that X is rank r, and solve the non-linear optimization problem

$$\begin{aligned} \min \quad & \text{rank}(Z) \\ \text{such that} \quad & \|\mathcal{R}_\Omega(Z) - y\|_2 \leq \epsilon. \end{aligned} \tag{2.7}$$

However, the objective function $\text{rank}(Z)$ makes the problem NP-hard. So instead we define the convex envelope of the rank function.

Definition 2.2.1. *Let $\sigma_i(X)$ be the i^{th} singular value of a rank r matrix X. Then the* nuclear norm *of X is*

$$\|X\|_* := \sum_{i=1}^{r} \sigma_i(X).$$

We now proceed by attempting to solve the convex relaxation of (2.7),

$$\begin{aligned} \min \quad & \|Z\|_* \\ \text{such that} \quad & \|\mathcal{R}_\Omega(Z) - y\|_2 \leq \epsilon. \end{aligned} \tag{2.8}$$

As with traditional compressive sensing, there exists a *restricted isometry property* (RIP) over the set matrices of rank r.

Definition 2.2.2. *A linear operator $\mathcal{R}_\Omega : \mathbb{R}^{s_1 \times s_2} \to \mathbb{R}^m$ satisfies the RIP of rank r with isometry constant δ_r if, for all rank r matrices X,*

$$(1 - \delta_r)\|X\|_F \leq \|\mathcal{R}_\Omega(X)\|_2 \leq (1 + \delta_r)\|X\|_F.$$

The RIP has been shown to be a sufficient condition to solve (2.8) [19, 56, 98]. These papers build on each other to establish the following theorem.

Theorem 2.2.3. *Let X be an arbitrary matrix in $\mathbb{C}^{s_1 \times s_2}$. Assume $\delta_{5r} < 1/10$. Then the $\widehat{X}$ obtained from solving (2.8) obeys*

$$\|\widehat{X} - X\|_F \leq C_0 \frac{\|X - X_r\|_*}{\sqrt{r}} + C_1 \epsilon, \tag{2.9}$$

where X_r is the best r rank approximation to X, and C_0, C_1 are small constants depending only on the isometry constant.

This means that, if the measurement operator $\mathcal{R}_\Omega$ is RIP, then reconstruction via convex optimization behaves stably in the presence of noise.

2.3 Matrix Completion with Tight Frame Measurements

Whether a set of measurements $\{\phi_i\}$ satisfies RIP is a difficult question. As we said in Section 2.2, RIP is a sufficient condition for an operator to satisfy the

noise bounds of Theorem 2.2.3. Without this, there is no guarantee that solving (2.8) yields an accurate prediction of X. For this reason, the rest of this section shall focus on proving $\mathcal{R}_\Omega$ is an RIP operator for measurements which form a *bounded norm Parseval tight frame* from Definition 1.1.5. In other words, our measurement operator is

$$\begin{aligned} \mathcal{R}_\Omega &: \mathbb{R}^{s_1 \times s_2} \to \mathbb{R}^m, \\ (\mathcal{R}_\Omega(X))_k &= \langle \phi_{i_k}, X \rangle, \end{aligned} \tag{2.10}$$

where $\{\phi_i\}_{i \in J}$ form a bounded norm Parseval tight frame.

Our central theorem establishes bounds on the quality of reconstruction from (2.8) in the presence of noise, when $\{\phi_i\}$ form a bounded norm Parseval tight frame. The theorem and proof rely on a generalization of [86], which only assumes the measurements to be orthonormal basis elements.

It is interesting to note that, because our measurements are overcomplete ($|J| > s_1 s_2$), our system of equations is not necessarily underdetermined. However, 2.3.1 still gives guarantees on how the reconstruction scales with the noise, whether or not the system is underdetermined or overdetermined. This is a difference from most compressive sensing literature. Generally the goal is to show an underdetermined system still has a solution, which is stable. In our case we are showing that, regardless of whether or not the system is underdetermined, our reconstruction is stable in the presence of noise and the reconstruction error decreases monotonically with the number of measurements.

Theorem 2.3.1. *Let $\{\phi_j\}_{j \in J} \subset \mathbb{C}^{s_1 \times s_2}$ be a bounded norm Parseval tight frame, with incoherence parameter μ. Let $n = \max(s_1, s_2)$, and let the number of measurements*

m satisfy

$$m \geq C\mu rn \log^5 n \cdot \log |J|,$$

where C is a constant. Let the sampling operator $\mathcal{R}_\Omega$ *be defined for* $\Omega \subset J$*, with* $\Omega = \{i_1, ..., i_m\}$ *as*

$$\mathcal{R}_\Omega : \mathbb{C}^{s_1 \times s_2} \to \mathbb{C}^m,$$

$$(\mathcal{R}_\Omega(X))_j = \langle \phi_{i_j}, X \rangle, \ j = 1, ..., m.$$

Let measurements y satisfy (2.6). Then with probability greater than $1 - e^{-C\delta^2}$ *over the choice of* Ω*, the solution* $\widehat{X}$ *to (2.8) satisfies*

$$\|\widehat{X} - X\|_F \leq C_0 \frac{\|\widehat{X} - \widehat{X}_r\|_*}{\sqrt{r}} + C_1 p^{-1/2} \epsilon, \tag{2.11}$$

where $p = \frac{m}{|J|}$.

To prove this result, we need a key lemma, which establishes that our measurements satisfy RIP.

Lemma 2.3.2. *Let* $\{\phi_j\}_{j \in J} \subset \mathbb{C}^{s_1 \times s_2}$ *be a bounded norm Parseval tight frame, with incoherence parameter* μ*. Fix some* $0 < \delta < 1$*. Let* $n = \max(s_1, s_2)$*, and let the number of measurements m satisfy*

$$m \geq C\mu rn \log^5 n \cdot \log |J|, \tag{2.12}$$

where $C \propto 1/\delta^2$*. Let the sampling operator* $\mathcal{R}_\Omega$ *be defined for* $\Omega \subset J$*, with* $\Omega = \{i_1, ..., i_m\}$ *as*

$$\mathcal{R}_\Omega : \mathbb{C}^{s_1 \times s_2} \to \mathbb{C}^m,$$

$$(\mathcal{R}_\Omega(X))_j = \langle \phi_{i_j}, X \rangle, \ j = 1, ..., m.$$

Then with probability greater than $1 - e^{-C\delta^2}$ *over the choice of* Ω, $\sqrt{\frac{|J|}{m}}\mathcal{R}_\Omega$ *satisfies the RIP of rank* r *with isometry constant* δ.

The proof of this lemma is found in the appendix and follows [86], where the claim is proved for an orthonormal basis. The main point here is to generalize the measurements to a bounded norm Parseval tight frame (also mentioned in [92], however not considering when $m > n^2$).

Proof of Theorem 2.3.1. We assume that Lemma 2.3.2 is true. Lemma 2.3.2 states that $\sqrt{\frac{|J|}{m}}\mathcal{R}_\Omega$ satisfies the RIP. However, (2.8) is stated using only $\mathcal{R}_\Omega$ as the measurement operator.

This means we must include a scaling factor of $\sqrt{\frac{|J|}{m}}$ to understand the noise bound. Let $p = \frac{m}{|J|} = \frac{m}{N}$ be the percentage of elements observed. Then, to utilize RIP, we must try to solve the problem

$$\begin{aligned} &\min && \|Z\|_* \\ &\text{such that} && \|p^{-1/2}\mathcal{R}_\Omega(Z) - p^{-1/2}y\|_2 \leq p^{-1/2}\epsilon. \end{aligned} \tag{2.13}$$

While scaling by a constant does not affect the result of the minimization problem, it does help us better understand the error in our reconstruction.

Theorem 2.2.3 tells us that our reconstruction error is bounded by a constant multiple of the error bound. But (2.13) means we can rewrite the error bound as

$$\|\widehat{X} - X\|_F \leq C_0 \frac{\|\widehat{X} - \widehat{X}_r\|_*}{\sqrt{r}} + C_1 p^{-1/2}\epsilon,$$

thus attaining the desired inequality. □

Remark: Examination of the proof of Lemma 2.3.2 shows that the bound on m in (2.12) is actually not sharp. We recall from (2.15) in Section 2.4 that m is

actually bounded below by a factor of $\log m$. In (2.16) we simply overestimate this term with $\log |J|$ for simplicity. However, in reality the bound is

$$m \geq C\lambda\mu rn \log^5 n \cdot \log m.$$

Let $N = C\lambda\mu rn \log^5 n$. This would give the bound $m \geq e^{-W_{-1}(-1/N)}$, where W_{-1} is the lower branch of the Lambert W function [38]. Taking the first three terms of a series approximation of W_{-1} in terms of $\log(1/N)$ and $\log(\log(N))$ [37] gives us

$$\begin{aligned} m &\geq e^{-\log(1/N)} e^{\log(\log(N))} e^{-\frac{\log(\log(N))}{\log(1/N)}} \\ &= N \log(N) e^{-\frac{\log(\log(N))}{\log(1/N)}} \\ &= C\lambda\mu rn \log^5 n \cdot \log(C\lambda\mu rn \log^5 n) \cdot e^{\frac{\log(\log(C\lambda\mu rn \log^5 n))}{\log(C\lambda\mu rn \log^5 n)}}. \end{aligned} \tag{2.14}$$

Note that taking three terms is sufficient as each subsequent term is asymptotically small compared to the previous. The bound in (2.14) is clearly much more intricate than simply bounding by $m \geq C\lambda\mu rn \log^5 n \log |J|$, but for typical sizes of $|J|$, this results in m decreasing by less than 5% from its original size.

2.4 Proof of Lemma 2.3.2

Let us define

$$U = \{X \in \mathbb{C}^{s_1 \times s_2} | \|X\|_* \leq \sqrt{r}\|X\|_F\}.$$

Notice that the set of all rank r matrices in $\mathbb{C}^{s_1 \times s_2}$ is a subset of U by Hölder's inequality. For the proof, we need some notation.

$$U_2 = \{X \in \mathbb{C}^{s_1 \times s_2} | \|X\|_F \leq 1, \|X\|_* \leq \sqrt{r}\|X\|_F\},$$

$$\epsilon_r(\mathcal{A}) = \sup_{X \in U_2} |\langle X, (\mathcal{A}^*\mathcal{A} - \mathcal{I})X\rangle|.$$

RIP can be rewritten as

$$(1-\delta)^2\langle X, X\rangle \leq \langle X, \mathcal{A}^*\mathcal{A}X\rangle \leq (1+\delta)^2\langle X, X\rangle, \quad \forall X \in U,$$

which is implied by

$$|\langle X, (\mathcal{A}^*\mathcal{A} - \mathcal{I})X\rangle| \leq 2\delta - \delta^2, \quad \forall X \in U_2.$$

So we need to show $\epsilon_r(\mathcal{A}) \leq 2\delta - \delta^2 \equiv \epsilon$.

One can then define a norm on the set of all self-adjoint operators from $\mathbb{C}^{s_1 \times s_2}$ to $\mathbb{C}^{s_1 \times s_2}$ by

$$\|\mathcal{M}\|_{(r)} = \sup_{X \in U_2} |\langle X, \mathcal{M}X\rangle|.$$

The proof that this is a norm, and that the set of self-adjoint operators is a Banach space with respect to $\|\cdot\|_{(r)}$, is found in [86].

We can now write $\epsilon_r(\mathcal{A}) = \|\mathcal{A}^*\mathcal{A} - \mathcal{I}\|_{(r)}$. For our purposes, as with most compressive sensing proofs, we first bound the expected value $\mathbb{E}\epsilon_r(\mathcal{A})$, and then show that $\epsilon_r(\mathcal{A})$ is concentrated around its mean.

For our problem dealing with tight frame measurements, let $\mathcal{A}^*\mathcal{A} - \mathcal{I} = \sum_{i=1}^{m} \chi_i$, where $\chi_i = \frac{|J|}{m}\phi_i^*\phi_i - \frac{\mathcal{I}}{m}$. Also, let χ_i' be independent copies of the random variable χ_i. Finally, let ϵ_i be a random variable that takes values ± 1 with equal probability.

Then we have that

$$\begin{aligned}
\mathbb{E}_\Omega \epsilon_r(\mathcal{A}) &= \mathbb{E}_\Omega \left\| \sum \chi_i \right\|_{(r)} \\
&\leq \mathbb{E}_\Omega \left\| \sum \chi_i - \chi_i' \right\|_{(r)} \\
&= \mathbb{E}_\Omega \mathbb{E}_\epsilon \left\| \sum \epsilon_i (\chi_i - \chi_i') \right\|_{(r)} \\
&= \mathbb{E}_\Omega \mathbb{E}_\epsilon \left\| \sum \epsilon_i (\phi_i^* \phi_i - (\phi_i')^* \phi_i') \frac{|J|}{m} \right\|_{(r)} \\
&\leq 2\frac{n}{m} \mathbb{E}_\Omega \mathbb{E}_\epsilon \left\| \sum \epsilon_i \sqrt{\frac{|J|}{n}} \phi_i^* \phi_i \sqrt{\frac{|J|}{n}} \right\|_{(r)}.
\end{aligned}$$

Now we cite Lemma 3.1 in [86] which is general enough to remain unchanged in the case of tight frames.

Lemma 2.4.1. *Let $\{V_i\}_{i=1}^m \subset \mathbb{C}^{s_1 \times s_2}$ have uniformly bounded norm, $\|V_i\| \leq K$. Let $n = \max(s_1, s_2)$ and let $\{\epsilon_i\}_{i=1}^m$ be iid uniform ± 1 random variables. Then*

$$\mathbb{E}_\epsilon \left\| \sum_{i=1}^m \epsilon_i V_i^* V_i \right\|_{(r)} \leq C_1 \left\| \sum_{i=1}^m V_i^* V_i \right\|_{(r)}^{1/2}$$

where $C_1 = C_0 \sqrt{r} K \log^{5/2} n \log^{1/2} m$ and C_0 is a universal constant.

For our purposes, $V_i = \sqrt{\frac{|J|}{n}} \phi_i$. Then

$$\begin{aligned}
\mathbb{E}\epsilon_r(\mathcal{A}) &\leq 2C_1 \frac{n}{m} \mathbb{E}_\Omega \left\| \sum \sqrt{\frac{|J|}{n}} \phi_i^* \phi_i \sqrt{\frac{|J|}{n}} \right\|_{(r)}^{1/2} \\
&\leq 2C_1 \frac{n}{m} \left(\mathbb{E}_\Omega \left\| \sum \sqrt{\frac{|J|}{n}} \phi_i^* \phi_i \sqrt{\frac{|J|}{n}} \right\|_{(r)} \right)^{1/2} \\
&= 2C_1 \sqrt{\frac{n}{m}} \left(\mathbb{E} \|\mathcal{A}^* \mathcal{A}\| \right)^{1/2} \\
&\leq 2C_1 \sqrt{\frac{n}{m}} (\mathbb{E}\epsilon_r(\mathcal{A}) + 1)^{1/2}.
\end{aligned}$$

Here,

$$C_1 = C_0 \sqrt{r} \sqrt{\mu} \log^{5/2} n \cdot \log^{1/2} m. \tag{2.15}$$

If we take $E_0 = \mathbb{E}\epsilon_r(\mathcal{A})$ and $C = 2C_1\sqrt{\frac{n}{m}}$, then (2.15) gives us

$$E_0^2 - C^2E_0 - C^2 \leq 0.$$

Fix some $\lambda \geq 1$ and choose

$$\begin{aligned} m &\geq C\lambda\mu rn\log^5 n \cdot \log|J| \\ &\geq \lambda n(2C_1)^2. \end{aligned} \tag{2.16}$$

This makes $\mathbb{E}\epsilon_r(\mathcal{A}) \leq \frac{1}{\lambda} + \frac{1}{\sqrt{\lambda}}$.

The next step is to show $\epsilon_r(\mathcal{A})$ does not deviate far from $\mathbb{E}\epsilon_r(\mathcal{A})$. Let $\mathcal{A}^*\mathcal{A} - \mathcal{I} = \chi$ be a random variable and χ' be an independent copy of χ. We now notice

$$\Pr(\|\chi\|_{(r)} > 2\mathbb{E}\epsilon_r(\mathcal{A}) + u) \leq 2\Pr(\|\chi - \chi'\|_{(r)} > u).$$

Define $\mathcal{Y}_i = \chi_i - \chi_i'$, so that $\chi - \chi' = \mathcal{Y} = \sum_{i=1}^{m} \mathcal{Y}_i$. Clearly

$$\|\mathcal{Y}_i\|_{(r)} \leq 2\|\chi_i\|_{(r)} = 2\sup_{X\in U_2}\left|\frac{|J|}{m}|\langle\phi_i, X\rangle|^2 - \frac{1}{m}\|X\|_F^2\right| \leq 2\frac{nr\mu+1}{m} \leq \frac{1}{2\lambda C_0^2}.$$

We now use the following result by Ledoux and Talagrand in [83].

Theorem 2.4.2. *Let $\{\mathcal{Y}_i\}_{i=1}^m$ be independent symmetric random variables on some Banach space such that $\|\mathcal{Y}_i\| \leq R$. Let $\mathcal{Y} = \sum_{i=1}^{m} \mathcal{Y}_i$. Then for any integers $l \geq q$ and any $t > 0$*

$$\Pr(\|\mathcal{Y}\| \geq 8q\mathbb{E}\|\mathcal{Y}\| + 2Rl + t\mathbb{E}\|\mathcal{Y}\|) \leq (K/q)^l + 2e^{-t^2/256q},$$

where K is a universal constant.

Now for appropriate choices of q, l and t, and with an appropriate λ such that $\lambda \geq A/\epsilon^2$ for some constant A, we get that

$$\Pr(\|\chi\|_{(r)} \geq \epsilon) \leq e^{-C\epsilon^2\lambda},$$

where C is a constant. Thus, probability of failure is exponentially small in λ.

2.5 Conclusion

Theorem 2.3.1 builds upon the current matrix completion literature by generalizing the set of measurements allowed for reconstruction to a bounded norm Parseval tight frame. Because these frame measurements are overcomplete, (2.8) begins to blur the line between a traditional underdetermined compressive sensing optimization problem and a regularization problem for an overdetermined system. The number of measurements, m, is no longer bounded above by $s_1 s_2$, but instead can range to $|J|$. In some problems, $|J|$ may be only slightly larger than $s_1 s_2$, and $m < s_1 s_2$. But in other application areas, like Chapter 5, $|J| \gg s_1 s_2$ forces $m > s_1 s_2$.

Further work in this theory shall be to examine the behavior of the incoherence μ in a variety of contexts. It may be interesting to ask to what extend may it hinder the theory when $|J| \gg s_1 s_2$, or whether there are better bounds on m that don't involve μ as a parameter. These will be questions of further investigation.

Chapter 3: The Preimage Problem for Laplacian Eigenmaps

3.1 Introduction

Non-linear dimensionality reduction has become a very important field in the last decade due to the influx of big data. Broadly stated, dimension reduction techniques map data for the original *data space* into a *feature space* that is better suited for linear algorithms and classification. This is generally done by learning a kernel which encodes important information about the mapping based on the training data. A detailed explanation of kernel methods can be found in [84].

While the feature space is of primary importance in dimensionality reduction techniques, the problem of mapping the data from the feature space back into the input space is also crucial. This has become known as the *pre-image* problem [101]. Consider for example the problem of data fusion. When two different measurement modalities are present, it may be impossible to consider their fusion in the data space. However, it is possible to compare these modalities in a more appropriate, lower dimensional feature space [35]. The problem with this feature space fusion is that it may be important to pull back the data into the original space in order to better understand the results of the data fusion. This step is where the pre-image problem arises.

We start by noting that the pre-image may not necessarily exist [91]. The simplest example of this would be a line of data points

$$X = \left\{ x_i \in \mathbb{R}^d | x_{i+1} = x_i + z \text{ for fixed } z \in \mathbb{R}^d, i = 1, ..., n \right\}.$$

Any dimension reduction map $\phi : \mathbb{R}^d \to \mathbb{R}^2$ learned on X would reduce X to a straight line in $\mathbb{R}^2$. Now introduce a new point $\psi \notin \overline{\text{span}}(\phi(X))$. There does not exist an $x \in \mathbb{R}^d$ such that $\phi(x) = \psi$. Thus, the pre-image of ψ does not exist.

Despite this issue, it is still important to consider calculating an approximate pre-image. Several papers [3, 78, 79, 91, 101] have focused on this problem in the context of kernel PCA [102]. Unlike Laplacian Eigenmaps, kPCA forms a dense kernel matrix K measuring the distances between all input vectors. The algorithm then performs PCA on the double centered kernel matrix resulting in the principle components, with appropriate weights, as the desired feature vectors. The algorithm allows for the extension of PCA to non-linear manifolds by linearizing the manifold in higher dimensions. However, because K is dense for kernel PCA, computing the eigenvector decomposition is much more computationally expensive than finding the decomposition for the sparse Laplacian in LE [9].

We shall examine a pre-image algorithm for Laplacian Eigenmaps (LE), as defined in Section 1.2. Several papers [78, 109] use linear extrapolation to define pre-images for Laplacian Eigenmaps and diffusion maps. We shall focus on a non-linear pre-image algorithm and demonstrate its effectiveness in pulling new points added to the feature space back into the original data space.

3.2 Nyström Extension

Having defined the embedding ϕ on the training set Ω in Section 1.2, we aim to extend it to other points from the data manifold. Let $x \in \mathbb{R}^d \backslash \Omega$. The goal is to calculate $\phi(x)$ to embed x, but without having to recalculate the entire Laplacian Eigenmap embedding. The Nyström extension [13,57] is a linear time approximation of $\phi(x)$ where

$$\widehat{\phi}(x) = \sum_{i=1}^{n} K(x, x_i)\phi(x_i). \tag{3.1}$$

Note that this extends the entire mapping, but also requires us to extend the kernel K. It would be straightforward if K were defined over all Ω as a simple function defined in $\mathbb{R}^d$. However, in the case of Laplacian Eigenmaps, K depends inherently on the training data (due to the nearest neighbor condition), making the extension non-trivial.

In [13] the authors propose an extension for this and other dimension reduction maps. Let $\widetilde{k}$ be a nearest neighbor Gaussian kernel, like in (1.2). In other words, let

$$\widetilde{k}(x, x_i) = \begin{cases} k(x, x_j) & : \quad x_j \in \mathcal{N}(x), \\ 0 & : \quad \text{otherwise.} \end{cases}$$

Now we define the kernel extension to all of $\mathbb{R}^d$ to be

$$K(x, x_i) = \frac{\widetilde{k}(x, x_i)}{\sqrt{\sum_{j=1}^{n} \widetilde{k}(x, x_j) \sum_{j=1}^{n} \widetilde{k}(x_i, x_j)}}. \tag{3.2}$$

For notational ease, we shall use $K_x = [K(x, x_i)]_{i=1}^n$. We now write (3.1) in

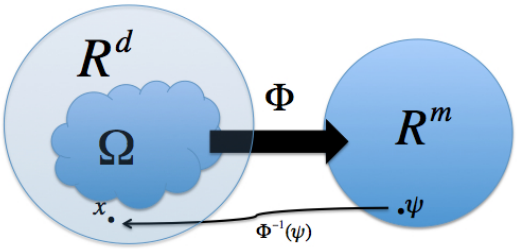


Figure 3.1: *Diagram of Laplacian Eigenmaps Mapping and Pre-image*

matrix form, namely

$$\widehat{\phi}(x) = \mathcal{E}^* K_x, \tag{3.3}$$

where $\mathcal{E} = [\phi_1, ..., \phi_m]$. Note that $\|K_x\|_0 = \#\mathcal{N}(x)$ due to the nearest neighbors criteria. This comes into play in Section 3.3.

3.2.1 The Pre-image Problem

The pre-image of $\psi \in \mathbb{R}^m$ is a point $x \in \mathbb{R}^d$ such that $\phi(x) = \psi$ (see Figure 3.1). Because x may not necessarily exist, this problem is ill-defined. So instead, we shall look for $x \in \mathbb{R}^d$ such that $\phi(x)$ is "as close as possible" to ψ by some definition of closeness. The most intuitive definition is

$$x = \arg\min_{x \in \mathbb{R}^d} \|\phi(x) - \psi\|_2.$$

This notion of minimizing distance is the common approach taken for kernel PCA [79,91,101]. However, this minimization is next to impossible without a closed form definition of $\phi : \mathbb{R}^d \to \mathbb{R}^m$, which we do not have.

For this reason, [3] makes the observation that, for kernel PCA, one could change the problem slightly to approximate $\phi(x)$ with the Nyström extension. Let

$\mathcal{E} \in \mathbb{R}^{m \times n}$ be the Nyström extension for a general dimension reduction scheme. We can create a new objective function by solving

$$x = \arg\min_{x \in \mathbb{R}^d} \|\mathcal{E} K_x - \psi\|_2. \tag{3.4}$$

From here, [3] solves (3.4) for the optimal K_x by calculating the Penrose-Moore pseudo-inverse. This gives us

$$\widehat{K}_x = \mathcal{E}^\dagger \psi. \tag{3.5}$$

Now in the special case of kernel PCA, we know that $(K_x)_i = e^{-\frac{\|x-y\|_2^2}{2\sigma^2}}$. This means (3.5) yields

$$\|x - x_i\|_2^2 = -2\sigma^2 \log((\widehat{K}_x)_i). \tag{3.6}$$

Finding x now reduces to a localization problem that is solved using MDS [3], and the distances between x and $\mathcal{N}(x)$. An explanation of this step is found in [60, 79].

In Laplacian Eigenmaps, however, the relationship between K_x and $\|x - x_i\|_2$ is not so immediate, as each $(K_x)_i$ is a function of $\|x - x_i\|_2$ for all $x_i \in \mathcal{N}(x)$. For this reason, one cannot immediately apply the methods of [3]. Section 3.3.1 solves this problem.

Since the original works by Mika, Schölkopf and Kwok, other attempts have been made to solve the pre-image problem by building upon their ideas. Notably in [114] the authors use a Laplacian, ridge, and weakly supervised penalty function, in conjunction with the optimization function, to improve the pre-image learning process. A non-iterative solution to the pre-image problem is proposed in [72] which improves the computational complexity of the original algorithms. In [55, 109] the

authors use diffusion maps as their embedding technique and seek to find a pre-image for the purposes of learning shape priors. They suggest that if a point lies outside of the convex hull of training points they could use its orthogonal projection onto the convex hull to acquire a valid pre-image. In [5, 6] the authors focus on the fact that there are labeled samples (the training points) available for the unknown pre-image map and with a kernel regression technique these samples can improve the pre-image learning process for minimal noise situations. Recently, in [107], the authors used diffusion kernels with linear extrapolation to learn to a pre-image map.

3.3 Laplacian Eigenmaps Pre-image

We now focus on the advancements that we have made to extend the pre-image problem to Laplacian Eigenmaps. In fact, we shall show in Section 3.3.2 that Laplacian Eigenmaps is actually the more intuitive dimension reduction map to use in a pre-image problem, due to the sparsity of K_x.

In Sections 3.3.2 and 3.3.3, we shall demonstrate the improvements our method provides by considering a simple experiment on the MNIST handwriting dataset, which contains 28×28 resolution images of handwritten digits. Section 3.4.2 uses this dataset for a complete analysis of our pre-image problem, but in Sections 3.3.2 and 3.3.3 we shall simply examine the accuracy of individual components of the algorithm. A training dataset was built using 200 randomly chosen samples for each digit (2000 samples total). Given a new image, we project this image into the feature space via the Nyström extension, and then, using a pre-image algorithm,

pull it back into the learned digit space. The feature space is 250 dimensions, less than a third the dimensionality of the original images.

3.3.1 Solving for Input Distances

The biggest challenge with simply applying the kernel PCA framework developed in [3] to Laplacian Eigenmaps comes in (3.6). Namely, Laplacian Eigenmaps does not have a simple, closed form solution that relates $(K_x)_i$ to $\|x - x_i\|_2$. Recall that for Laplacian Eigenmaps, $(K_x)_i$ behaves as in (3.2). The problem is that, due to the term $\sum_{j=1}^{n} \widetilde{k}(x, x_j)$ in the denominator, $(K_x)_i$ depends simultaneously on all $\|x - x_j\|_2$, $j \in \{1, ..., n\}$.

However, we recall that $\|K_x\|_0 = c \ll n$. Let $\{x_{i_1}, ..., x_{i_c}\} = \mathcal{N}(x)$, and define

$$e_j = k(x, x_{i_j}),$$

so that $e \in \mathbb{R}^c$ is a vector of the non-zero contributions to K_x, with

$$(K_x)_{i_j} = \frac{e_j}{\sqrt{\sum_l e_l \sum_{x_l \in \mathcal{N}(x_{i_j})} k(x_{i_j}, x_l)}}.$$

This means

$$\frac{e_j}{\sqrt{\sum_l e_l}} = (K_x)_{i_j} \sqrt{\sum_{x_l \in \mathcal{N}(x_{i_j})} k(x_{i_j}, x_l)} = a_j.$$

By squaring both sides, we obtain a non-linear system of c equations

$$e_j^2 - a_j^2 \sum_l e_l = 0, \quad 1 \leq j \leq c. \tag{3.7}$$

The system of equations in (3.7) can be solved with a variety of methods, which shall be discussed in a moment. But once the e_j are found, we solve for the

distances in the input space of the c nearest neighbors of x,

$$\|x - x_{i_j}\|_2^2 = -2\sigma^2 \log(e_j), \quad x_{i_j} \in \mathcal{N}(x). \tag{3.8}$$

From here, the MDS approach from [60, 79] is applied to calculate x, just as in the majority of algorithms for kernel PCA.

We now discuss solving (3.7). The simplest solution would be to use Newton's method to find the non-trivial zero of this system. However, only a noisy version of K_x is known, so a more stable algorithm is required. The system of equations in (3.7) is rewritten as

$$\min_{e \in \mathbb{R}^c} \|\mathcal{A}(e)\|_2,$$

where $\mathcal{A}_i(e) = e_i^2 - a_i^2 \sum_j e_j$, $i = 1, ..., c$. Now assume, without loss of generality, that $a_1 \geq a_2 \geq ... \geq a_c$. Then we add constraints to our minimization to force $1 \geq e_1 \geq ... \geq e_c \geq 0$, which is mathematically accurate, by defining a matrix B and a parameter α such that

$$B_{i,j} = \begin{cases} -1 + \alpha & : \quad j = i \\ 1 & : \quad j = i + 1 \\ 0 & : \quad \text{otherwise} \end{cases} .$$

Now (3.7) is solved with

$$\min_{e \in (0,1)^c} \|\mathcal{A}(e)\|_2 \quad \text{such that } B(e) < 0 \tag{3.9}$$

Table 3.1 demonstrates the advantages of using (3.9) over a standard root finding algorithm, e.g., Newton's method. In this experiment, the true $e_j = k(x, x_j)$ were

generated as uniform random variables with $e_j \in (\frac{2}{3}, 1)$, and $\#\mathcal{N}(x) = 25$. Gaussian noise, where the standard deviation is determined as a percentage of $\max(K_x)$, was added to K_x after it is calculated from e_j, $j \in \{1, ..., 25\}$. The error is taken to be $\|\widehat{e} - e\|_2 / \|e\|_2$. Clearly, (3.9) outperforms Newton's method in its stability in the presence of noise. For this reason, we shall use (3.9) in future pre-image calculations.

Noise Level	$\sigma = 2\%$ of $\max(K_x)$	$\sigma = 4\%$ of $\max(K_x)$	$\sigma = 6\%$ of $\max(K_x)$
Err. Newton's Method	0.0269	0.0575	0.0837
Err. Min. Problem (3.9)	0.0187	0.0347	0.0487

Table 3.1: *Average over 100 trials using various estimate methods. True* $e_i \in (\frac{2}{3}, 1)$ *are i.i.d. uniform random,* $i = \{1, ..., 25\}$. *Noise is added to* K_x, *and* $\widehat{e}$ *estimated via either Newton's method or (3.9). Err.=* $\|\widehat{e} - e\|_2 / \|e\|_2$.

3.3.2 Better Estimates of K_x

The pre-image problem is very susceptible to noise and errors. Almost all of this error comes in the estimation of K_x and the fact that (3.4) is akin to solving an underdetermined system. Thus, the least squares solution in (3.5) is far from the optimal K_x.

Instead, one can use the a priori information that $\|K_x\|_0 = c$. This sparsity is utilized by changing (3.4) to

$$\widehat{K}_x = \arg\min \|V^* K_x - \psi\|_2 \quad \text{such that } \|K_x\|_0 \leq c. \tag{3.10}$$

The constraint in (3.10) exactly mirrors the a priori information about K_x, and

could be solved using Orthogonal Matching Pursuit [94] or some other optimization algorithm with an L_0 constraint. However, the computational complexity of L_0 constraints grows quickly with n, which is the number of training points we begin with. This is a serious problem, as this removes the incentive of choosing a large n and training on as many points as possible. For this reason, we propose instead using an L_1 constraint,

$$\widehat{K}_x = \arg\min \|V^* K_x - \psi\|_2 \quad \text{such that } \|K_x\|_1 \leq \tau. \tag{3.11}$$

The L_1 constrained minimization problem is solved using any common quadratic programming algorithm with the same reconstruction guarantees and solved much faster than the L_0 problem [22].

The only issue that arises is lack of knowledge about the correct τ. In (3.10), c is understood immediately as the number of nearest neighbors in the graph ($\#\mathcal{N}(x)$). However, τ in (3.11) does not necessarily have immediate and intuitive bounds. In some applications, it may be possible to estimate τ a priori, but we assume for now that τ is unknown.

To analyze τ, we first consider the quantity $\alpha = \frac{\|\psi\|_2}{\|K_x\|_1}$, where K_x is the optimal kernel vector. Because $\psi = V^* K_x$, we know

$$\alpha = \left\| V^* \left(\frac{K_x}{\|K_x\|_1} \right) \right\|_2 = \left\| \sum_{x_i \in \mathcal{N}(x)} a_i V^*_{i,\cdot} \right\|_2,$$

where $V_{i,\cdot}$ is the i^{th} row of V and $a_i = \frac{(K_x)_i}{\|K_x\|_1}$. This means we need only estimate these weights the neighborhood $\mathcal{N}(x)$ and the associated weights a_i.

The easiest estimate of these weights is to begin with the pseudoinverse (3.5),

choose the c largest values $\{k_{i_1}, ..., k_{i_c}\}$, and choose

$$\widehat{a}_{i_j} = \frac{k_{i_j}}{\sum_j k_{i_j}}.$$

This gives us an estimate

$$\widehat{\alpha} = \left\| \sum_{j=1}^{c} \widehat{a}_{i_j} V^*_{i_j,\cdot} \right\|_2 .$$

Now that α has been estimated, we see that $\|K_x\|_1 = \frac{\|\psi\|_2}{\alpha}$. This means that, in order for K_x to be an admissible point for (3.11), we must have $\tau > \|K_x\|_1$. Thus, the estimate we choose for this parameter is

$$\tau = \frac{\|\psi\|_2}{\widehat{\alpha}} = \frac{\|\psi\|_2}{\left\| \sum_{j=1}^{c} \widehat{a}_{i_j} V^*_{i_j,\cdot} \right\|_2}. \tag{3.12}$$

The L_1 regularized step from (3.11) generates much more accurate estimates of the kernel vector K_x than previous methods. We demonstrate this using the MNIST dataset. Table 3.2 compares our L_1 regularized method with τ chosen via (3.12) to two methods that exist in literature and are based around the Penrose-Moore pseudoinverse. $\widehat{K}_x$ denotes the estimate from each method, and K_x is the true kernel vector for the new image. We note that one issue that occurs with existing methods is they misestimate the magnitude of the kernel vector, as one can see by examining $\|\widehat{K}_x\|_2 / \|K_x\|_2$.

Table 3.3 discusses our choice of τ in (3.12). Clearly, choosing an incorrect size of τ can significantly affect the reconstruction quality of $\widehat{K}_x$. Moreover, using (3.12) we get a dynamic estimate of τ for each new point. This is advantageous as compared to using a static value of τ as we do in the final column of the table.

$\tau = 5.276$ was chosen by taking the average value of τ from (3.12) across the 100 trials. While this static estimate works well, the dynamic choice of τ is clearly superior.

Method of Estimate	L_1 Min. in (3.11)	$\widehat{K}_x$ from [3]	$\widehat{K}_x$ from [79]
$\|\widehat{K}_x\|_2/\|K_x\|_2$	0.9838	6.0171	0.8387
$\|\widehat{K}_x - K_x\|_2/\|K_x\|_2$	0.0428	5.2063	0.5366

Table 3.2: *Average over 100 trials using various estimate methods. $\widehat{K}_x$ denotes the reconstructed kernel vector, K_x denotes the true kernel vector*

L_1 Bound in (3.11)	75% of τ from (3.12)	τ from (3.12)	125% of τ from (3.12)	Static $\tau = 5.276$
$\|\widehat{K}_x\|_2/\|K_x\|_2$	0.8075	0.9838	0.9752	0.9749
$\|\widehat{K}_x - K_x\|_2/\|K_x\|_2$	0.3063	0.0428	0.2153	0.0744

Table 3.3: *Average over 100 trials using various sizes of τ in L_1 minimization step. In first three cases, τ is chosen using (3.12). The static $\tau = 5.276$ was chosen as 5.276 is the average value of τ from (3.12) across the 100 trials.*

3.3.3 Noisy Pre-images

Until this point, we have assumed that the problem had no noise component. However, Section 3.4 details examples where this assumption is clearly violated. For that reason, we now consider a more complicated expression

$$\psi = \phi(x + \eta),$$

where $\eta \sim \mathcal{N}(0, \epsilon^2 \cdot I_{d\times d})$. Because ϕ is nonlinear, this increases the difficulty of finding x from ψ. Also, by adding noise to x, we are removing x from the range of the training data. And, as [3] points out, this causes $\|\psi\|$ to tend toward 0. For this reason, we consider the slight modification to our estimate of K_x when noise is present.

Let $K_{x+\eta}$ be the solution to the L_1 regularization problem in (3.11) when noise is present. In order to use the MDS step of (3.6), we must have an estimate for K_x instead. So the goal is to relate these two terms.

If we continue to take $k(x, y)$ to be Gaussian as in (1.2), then we see the i^{th} component must satisfy

$$\begin{aligned}
(K_{x+\eta})_i &= \frac{e^{-\frac{\|x-x_i+\eta\|^2}{2\sigma^2}}}{\sqrt{\sum_{x_j\in\mathcal{N}(x)} e^{-\frac{\|x-x_j+\eta\|^2}{2\sigma^2}} \sum_{x_j\in\mathcal{N}(x_i)} k(x_i, x_j)}} \\
&= \frac{e^{-\frac{\|x-x_i\|^2-2\langle x-x_i,\eta\rangle}{2\sigma^2}}}{\sqrt{\sum_{x_j\in\mathcal{N}(x)} e^{-\frac{\|x-x_j\|^2-2\langle x-x_j,\eta\rangle}{2\sigma^2}} \sum_{x_j\in\mathcal{N}(x_i)} k(x_i, x_j)}} \sqrt{e^{-\frac{\|\eta\|^2}{2\sigma^2}}} \\
&= \frac{e^{-\frac{\|x-x_i\|^2(1-X_i)}{2\sigma^2}}}{\sqrt{\sum_{x_j\in\mathcal{N}(x)} e^{-\frac{\|x-x_j\|^2(1-X_j)}{2\sigma^2}} \sum_{x_j\in\mathcal{N}(x_i)} k(x_i, x_j)}} \cdot e^{-\frac{\|\eta\|^2}{4\sigma^2}},
\end{aligned}$$

where $X_i := \frac{2\langle x-x_i,\eta\rangle}{\|x-x_i\|^2} \sim \mathcal{N}(0, \frac{4\epsilon^2}{\|x-x_i\|^2})$. Because $\mathrm{Var}(X_i) \ll 1$ for reasonable ϵ, we ignore these terms.

Now we address the right hand term of (3.13), the noise term $e^{-\frac{\|\eta\|^2}{4\sigma^2}}$. Clearly, $\|\eta\|^2 \sim \epsilon^2\chi^2(d)$. It is interesting to note that $e^{-\frac{\|\eta\|^2}{4\sigma^2}} \sim \ln\left(\Gamma(\frac{d}{2}, \frac{\epsilon^2}{4\sigma^2})\right)$, which is a log Gamma distribution. However, due to (3.6), we are only interested in $\log((K_x)_i)$.

This means we are able to simply focus on $\mathbb{E}\left(\frac{\|\eta\|^2}{4\sigma^2}\right)$, which is $\frac{\epsilon^2 d}{4\sigma^2}$. Thus, we take

$$\widehat{K}_x = K_{x+\eta} \cdot e^{\frac{\epsilon^2 d}{4\sigma^2}}. \tag{3.13}$$

This makes the distances

$$\|x - x_i\|_2^2 = -2\sigma^2 \left(\log((\widehat{K}_x)_i) + Z_i\right), \tag{3.14}$$

where $Z_i \sim \frac{\epsilon^2 \chi^2(d) - \epsilon^2 d}{4\sigma^2}$, which has mean 0 and variance $\frac{\epsilon^4 d}{8\sigma^4}$.

The size of ϵ can be estimated in any number of ways that are going to be application specific, either algorithmically or with a priori knowledge of the problem. As a simple example, the algorithm from [85] can be used to estimate ϵ in the case of RGB images.

We now examine the benefits of this step of the algorithm. In this experiment, the new image MNIST digit we project into the feature space has been corrupted by Gaussian noise with standard deviation $\epsilon = .2$. The noise level was estimated using [85]. Table 3.4 compares the kernel vector estimate $\widehat{K}_x$ for the new noisy image and the correct K_x calculated using the non-noisy version of that image, averaged over 100 trials. The errors for L_1 reconstruction without adjusting for noise and the estimate from [79] are just under 1 because these methods underestimate the magnitude of $\widehat{K}_x$, and error from using the algorithm in [3] is larger 1 because it overestimates the magnitude of $\widehat{K}_x$. The magnitude of $\widehat{K}_x$ is important in solving (3.9), which is why we report it here. Clearly, adjusting for noise in the kernel vector estimate is crucial.

To summarize, Algorithm 1 shows the complete process for calculating the pre-image of ψ. Complete tests of our method are located in Section 3.4.

Method of Estimate	Noise Adjusted L_1 in (3.13)	Not Noise Adjusted L_1	$\widehat{K}_x$ from [3]	$\widehat{K}_x$ from [79]
$\|\widehat{K}_x\|_2/\|K_x\|_2$	1.0239	0.2610	6.0171	0.2181
$\|\widehat{K}_x - K_x\|_2/\|K_x\|_2$	0.0752	0.7315	5.3563	0.7686

Table 3.4: *Average over 100 trials using various estimate methods. $\widehat{K}_x$ denotes the reconstructed kernel vector, K_x denotes the true kernel vector without the added noise*

Algorithm 1 Calculate Pre-image of $\psi \in \mathbb{R}^m$

Required: Training points $\Omega \in \mathbb{R}^d$

LE mapping $\phi : \Omega \to \mathbb{R}^m$

New point $\psi \in \mathbb{R}^m$

Result: $x \in \mathbb{R}^d$ that approximates $\phi^{-1}(\psi)$

1. Calculate $\widehat{K}_x$ using (3.11), setting τ using (3.12).
2. If there is noise present with variance ϵ^2, set $\widehat{K}_x \leftarrow \widehat{K}_x \cdot e^{\frac{\epsilon^2 d}{4\sigma^2}}$ as is done in (3.13).
3. Solve for e_i using (3.9).
4. Calculate for $\|x - x_i\|$ for all $x_i \in \mathcal{N}(x)$ using (3.8).
5. Find x using the $\|x - x_i\|$ calculated and the MDS algorithm from [60].

3.4 Examples

To test the feasibility of this method, we shall work with two experiments. Section 3.4.1 focuses on advantages of our method over other ways of defining the

pre-image for Laplacian Eigenmaps. Section 3.4.2 focuses on denoising, which is a common application in the pre-image literature [3, 79, 91, 101], and show the experimental advantages of our method.

3.4.1 Points Outside the Convex Hull of Training Data

Due to the ill-conditioned nature of the inverse problem for Laplacian Eigenmaps, one can define the approximate pre-image in different ways. In [78], the pre-image is defined in terms of linear extrapolation from training points. In other words, [78] defines the pre-image of $\psi \in \mathbb{R}^m$ to be

$$\phi^{-1}(\psi) = \sum_{x_i \in \Omega} w_i x_i, \tag{3.15}$$

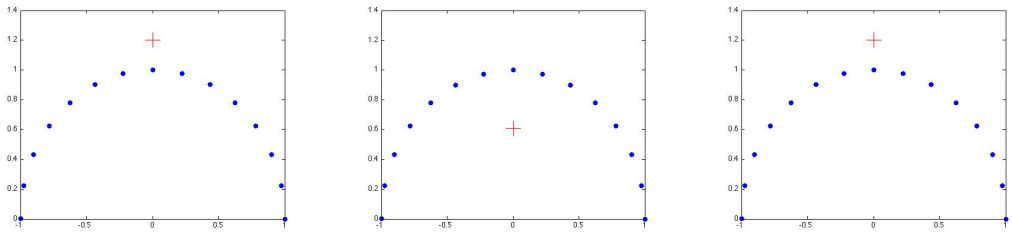
where

$$w_i = \frac{e^{-\frac{\|\psi - \phi(x_i)\|^2}{2\sigma_\psi^2}}}{\sum_{\phi(x_j) \in \mathcal{N}(\psi)} e^{-\frac{\|\psi - \phi(x_j)\|^2}{2\sigma_\psi^2}}}. \tag{3.16}$$

This effectively interpolates between points in $\Omega \subset \mathbb{R}^d$ based off the distances between ψ and the embedded training points in $\mathbb{R}^m$. However, because (3.15) is a weighted average of $x_i \in \Omega$ and $\sum w_i = 1$ from (3.16), $\phi^{-1}(\psi)$ must lie in the convex hull of Ω. This is not a limitation of our algorithm, as our reconstruction creates $\phi^{-1}(\psi)$ based off the MDS embedding technique of [60].

This issue is demonstrated in a simple experiment. Figure 3.2 shows a set of blue training points to be embedded using Laplacian Eigenmaps. The new point (red + marker) is then embedded via Nyström extension. The key is that the new point lies outside the convex hull of the training points. We then apply pre-

image algorithms to pull it back to the original space. The linear extrapolation method of [78] fails to properly reconstruct the extended point, whereas our pre-image algorithm properly recovers it based off the new point's distances from the training points.



(a) Original Points (b) Extrapolation [78] (c) LE Pre-image

Figure 3.2: *Pre-image of points outside convex hull: The training points (blue) are embedded via Laplacian Eigenmaps, and then the new point (red + marker) is embedded via Nyström extension. The new point is then pulled back into the original space.*

3.4.2 Digit Denoising

We shall compare the performance of the kernel PCA pre-image, a Laplacian Eigenmaps extension algorithm from [78], and our Laplacian Eigenmaps pre-image algorithm with L_1 regularization. We analyze our pre-image algorithm by comparing its performance using the 28×28 pixel MNIST handwriting dataset [82]. Section 3.3.3 gives a brief explanation of the approach, but we repeat it here for ease: A training dataset was built using 200 randomly chosen samples for each digit (2000 samples total). Given a new noisy image, we denoise by projecting this image into the feature space via the Nyström extension, and then using a pre-image algorithm

to pull it back into the digit space. For all algorithms, the feature space is 250 dimensions. The noise level ϵ was estimated using [85].

Table 3.5 shows the denoising results of the kernel PCA (kPCA) pre-image algorithm from [3], Laplacian Eigenmaps pre-image with the Penrose-Moore pseudoinverse, and Laplacian Eigenmaps with L_1 regularization. White Gaussian noise was added at several intensities, and the SNR is calculated over 10 samples of each digit.

	$\sigma^2 = .2$			$\sigma^2 = .4$			$\sigma^2 = .6$		
Digit	kPCA	Extrap	LE w/ L_1	kPCA	Intrpl	LE w/ L_1	kPCA	Intrpl	LE w/ L_1
0	4.28	3.17	5.38	4.08	2.79	5.09	4.20	3.24	4.75
1	5.37	4.77	4.85	5.02	4.55	4.64	5.13	4.28	4.45
2	4.27	2.55	5.12	3.92	2.50	4.76	3.68	2.76	4.54
3	4.17	3.15	4.73	4.02	2.97	4.30	3.94	2.73	4.32
4	3.66	2.51	4.65	3.66	2.24	4.32	3.37	2.34	3.72
5	3.54	2.55	4.63	3.48	2.50	4.39	3.35	2.53	3.99
6	4.20	2.59	5.41	3.98	2.70	5.07	3.99	2.50	4.85
7	4.33	3.08	4.85	4.35	3.73	4.50	4.04	3.73	3.90
8	3.68	2.55	4.44	3.33	2.40	4.16	3.55	2.24	4.10
9	3.97	3.36	4.72	3.76	3.18	4.25	3.67	3.07	4.01
Avg.	4.15	3.03	4.88	3.96	2.96	4.55	3.89	2.94	4.26

Table 3.5: *SNR for MNIST Dataset Comparing Pre-image Algorithms for kernel PCA, Laplacian Eigenmaps with linear extrapolation (Extrap) Scheme from [78], and Laplacian Eigenmaps with L_1 Regularization*

These results clearly show our LE pre-image algorithm with L_1 regularization performs just as well, if not slightly better, than the kernel PCA pre-image algorithm

of [3]. This is very promising for two reasons. First, as [84] points out, LE performs much better for data clustering than for dimensionality reduction. The fact that these LE reconstructions outperform kernel PCA with reduced dimensions is a very good sign. Second, LE has numerous computational advantages over kernel PCA. Namely, kernel PCA requires taking an eigenvalue decomposition of a full $n \times n$ matrix K, whereas LE only requires taking an eigenvalue decomposition of the sparse matrix $D^{-\frac{1}{2}}LD^{-\frac{1}{2}}$. For this reason, LE is more advantageous, especially in the region of large n (a large number of training points).

Also, these results show improvement over the linear extrapolation extension in [78]. This is explained by two observations. First, the estimate of $\widehat{K_x}$ in (3.11) and (3.13), followed by (3.9), give more accurate relationships between the embedded points than (3.16). Second, using the MDS embedding technique based off the distance estimates in (3.6), as detailed in [60, 79], give a more robust embedding technique than using a linear extrapolation formula (3.15).

Chapter 4: Emergence of Anomalous Features in Laplacian Eigenmaps

4.1 Introduction to Graph Theory in Dimension Reduction

Many nonlinear dimensionality reduction techniques, such as Laplacian Eigenmaps, Diffusion Maps [36], and Local Linear Embedding [99], center on building a graph on the data. This allows one to look at inter-data structure as a way to extract useful relationships. Successful analyses of these techniques focus on the assumption that the data lies on a smooth manifold and that the graph Laplacian approximates the Laplace-Beltrami operator on that manifold [10], or on the kernel applied to the data as a way to generate a better embedding [49, 103].

While these directions of research have led to numerous important results, they mostly ignore the important fact that the embeddings are completely dependent on the properties of the graph Laplacian. This chapter instead proposes to study these operators from the context of graph theory.

Graph theoretic analysis of dimension reduction allows the results to be independent of distance metric or local geometry of the data. An example of this is shown in Figure 4.1. These two data sets differ in geometry and even dimension.

However, the individual points relate to each other in a similar manner, and both data sets generate virtually identical graph Laplacians. Namely, both graphs consist of two nearly disjoint clusters.

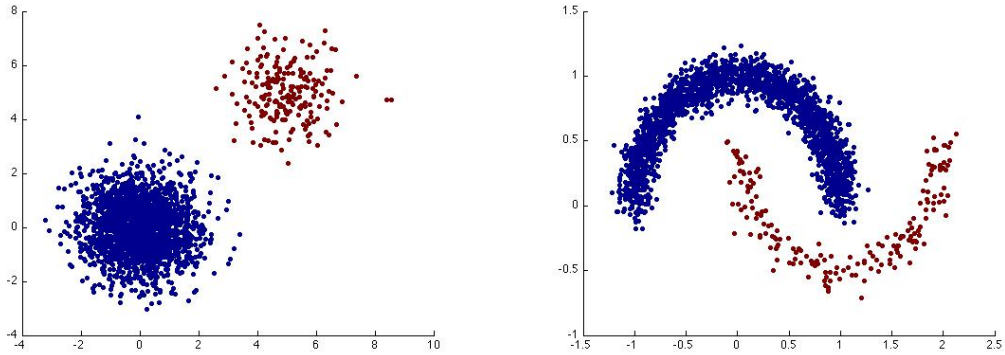
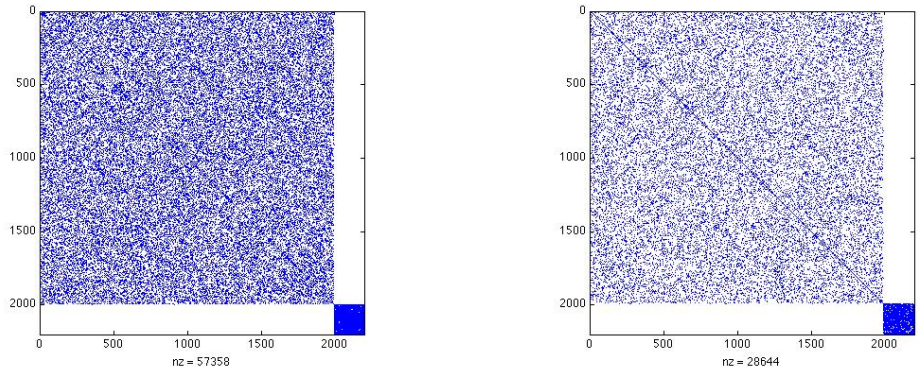


Figure 4.1: *Two data sets with differing geometries that generate virtually identical graph Laplacians with a Gaussian kernel.*

These clusters can also be described graph theoretically. Let us simplify the context slightly, and assume that the kernel $k(x, y)$ is an indicator function of whether x and y are nearest neighbors. We shall define a cluster on n points as being a randomly chosen k-regular graph with n nodes. This is because each row of the adjacency matrix has k non-zero entries due to the k nearest neighbors algorithm for choosing edges. Also, the edges in this cluster will be randomly chosen between points in the cluster, as there is no structural difference between the clustered points (only noise and slight variability). This means the degree of each clustered point will be k, and the distribution of those weights is independent of which point in the cluster is chosen.

Figure 4.2 shows this is a good approximation for the two datasets in Figure

4.1. We chose $k = 25$ nearest neighbors, and used a Gaussian kernel. For both graph Laplacians, the two clusters are almost completely disjoint. Also, we report the mean (μ) and standard deviation (σ) of the degree of the nodes. Clearly, all of the nodes have almost identical degree for both graphs, and are fairly close to being k-regular graphs.



(a) Gaussian Clusters, $\mu = 23.7$, $\sigma = 1.64$ (b) Moon Clusters, $\mu = 23.4$, $\sigma = 2.23$

Figure 4.2: *Non-zero terms in graph Laplacian generated by datasets in Figure 4.1. Note that the indices are pre-sorted into their respective clusters for easy visualization of the graph.*

For the rest of this chapter, we shall prove results about graphs with k-regular subgraph clusters. Also, we shall assume the kernel $k(x, y)$ is an indicator function of whether x and y are nearest neighbors. This shall allow us to approximate the behavior of Laplacian Eigenmaps by utilizing the vast literature that exists on regular graphs.

4.2 Eigenvector Distribution for Disjoint Clusters with Heterogeneous Sizes

For graph based non-linear dimensionality reduction techniques, such as LE, diffusion maps, and local linear embedding, the common assumption is that one only needs to keep the m smallest eigenvectors. However, the choice of m is commonly overlooked, other than assuming m must be at least as large as the intrinsic dimensionality of the data.

4.2.1 Example of Eigenvector Distribution

A general approach to choosing m is deciding on the intrinsic dimension of the data. However, Figure 4.3 demonstrates the choice of m is much more complicated. The data consists of two clusters in $\mathbb{R}^2$, with cluster C_1 containing 10,000 points, and cluster C_2 containing 1000 points. Laplacian eigenmaps is run on this example with a Gaussian kernel and 50 nearest neighbors. The images below show the 14 smallest eigenvectors with non-zero associated eigenvalues. Observe that, due solely to the differing sizes between the clusters, all but one of the eigenvectors has its entire energy concentrated in C_1.

This can serve as a problem for a number of reasons. For one, there are no intercluster features in the data. However, 13 of the first 14 eigenvectors are picking up erroneous features in C_1. This can lead to issues when the embedded points $\phi(\Omega)$ are inputs to a clustering algorithm such as k-means [89] or support

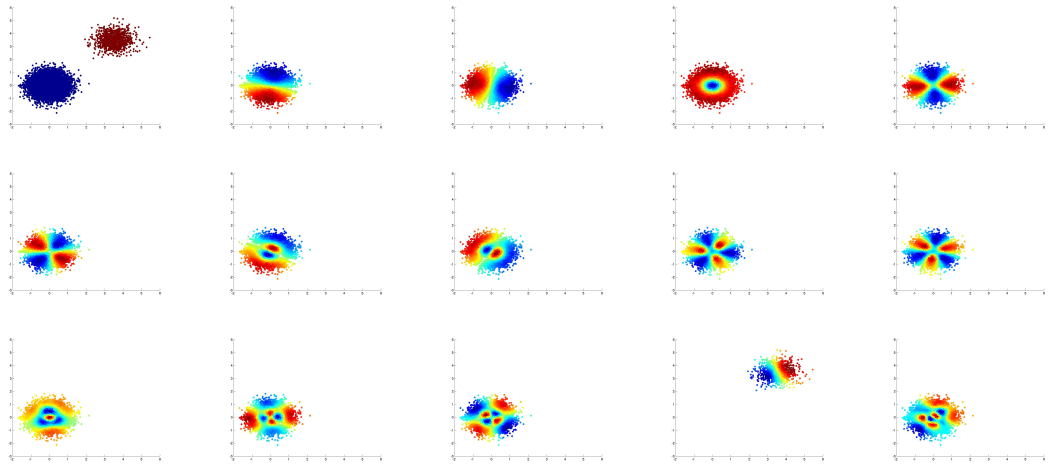


Figure 4.3: *Top left image shows the two original clusters. Then moving left to right, top to bottom are the intensities of each eigenvector of the graph Laplacian. Notice that the first appearance of the smaller cluster does not occur until the* 13^{th} *eigenvector.*

vector machines [39]. These erroneous features are given undue weight in clustering, leading to errors in classification.

Second, and most importantly, despite C_2 constituting significant data, almost all the energy in $\phi(\Omega)$ is concentrated in C_1. Again, this poses problems for clustering and classification algorithms. To see this, fix $i_0 \in \{1, ..., m\}$ such that $supp(\phi^{i_0}) \subset C_1$. This means $\phi^{i_0}(x) = 0$ for $x \in C_2$. However, $\exists x, y \in C_1$ such that $\phi^{i_0}(x) < 0$ and $\phi^{i_0}(y) > 0$. Thus, a separating line on ϕ^{i_0} would be unable to differentiate C_1 from C_2. And since most of the energy of $\phi(\Omega)$ lies in C_1, most $i \in \{1, ..., m\}$ satisfy $supp(\phi^i) \subset C_1$.

4.2.2 Rigorous Derivation of Eigenvector Distribution

The logic behind the phenomenon of Laplacian eigenvector localization is based in the distribution of eigenvalues of the Laplacian. Specifically, it depends on the eigenvalue distribution of regular graphs. The first significant progress in this problem came 30 years ago in a paper by McKay [90].

Theorem 4.2.1. *(Theorem 1.1, [90]) Let $X_1, X_2, ...$ be a sequence of random graphs with corresponding adjacency matrices $A_1, A_2, ...$, each with degree $k \geq 2$. Let $n(X_i)$ be the number of nodes for graph X_i, and $c_k(X_i)$ be the number of cycles of length k. Let the family $\{X_i\}$ satisfy $n(X_i) \to \infty$ and $c_k(X_i)/n(X_i) \to 0$ as $i \to \infty$. Then the empirical spectral distribution of the scaled adjacency matrix $\frac{1}{\sqrt{k-1}}A_n$, $F_{n,d}(x) = |\{i : \lambda_i(\frac{1}{\sqrt{k-1}}A_n) < x\}|/n$ converges to the semicircle law*

$$f_d(x) = \frac{1}{2\pi}\sqrt{4 - x^2}, \quad -2 < x < 2. \tag{4.1}$$

Following this result, the necessity to avoid cycles was removed in exchange for proving results about *random regular graphs*. Also, it raised the question of whether such convergence results could be made for n.

Definition 4.2.2. *The family of regular graphs $\mathcal{G}_{n,k}$ is the set of all graphs $G = (V, E)$ with n nodes and $\forall x \in V$, $deg(x) \equiv \sum_{\{x,y\} \in E} w_{x,y} = k$.*

Theorem 4.2.3. *(Theorem 2, [52]) Fix $\delta > 0$ and let $k = (\log(n))^\gamma$, and let $\eta = \frac{1}{2}(\exp(k^{-\alpha}) - \exp(-k^{-\alpha}))$ for $0 < \alpha < \min(1, 1/\gamma)$. Then there exists an N large enough such that $\forall n > N$, for $G \in \mathcal{G}_{n,k}$ chosen randomly with adjacency matrix A,*

for any interval $\mathcal{I} \subset \mathbb{R}$ *such that* $|\mathcal{I}| \geq \max\{2\eta, \eta/(-\delta \log \delta)\}$,

$$|\mathcal{N}_{\mathcal{I}} - n \int_{\mathcal{I}} f_d(x)dx| < n\delta|\mathcal{I}|$$

with probability at least $1 - o(1/n)$. *Here,* $\mathcal{N}_I$ *is the number of eigenvalues of* $\frac{1}{\sqrt{k-1}}A$ *in the interval* $\mathcal{I}$ *and* f_d *is the semicircle law in (4.1).*

Using this result, we begin to address the phenomenon that occurs in Figure 4.3.

Theorem 4.2.4. *Let* $\Gamma = (\Omega, E)$ *be an undirected graph. Suppose* Ω *can be split into disjoint two clusters* C_1 *and* C_2 *such that, for the subgraph* G_1 *generated by* C_1 *and the subgraph* G_2 *generated by* C_2, $G_1 \in \mathcal{G}_{n,k}$ *and* $G_2 \in \mathcal{G}_{\frac{n}{D},k}$. *Furthermore, assume* $\nexists\{x, y\} \in E$ *such that* $x \in C_1$ *and* $y \in C_2$.

Fix δ, k, α, *and* η *as in Theorem 4.2.3. Choose any interval* $\mathcal{I} \subset [0, 2]$ *such that* $|\mathcal{I}| \geq \frac{\sqrt{k-1}}{k} \max\{2\eta, \eta/(-\delta \log \delta)\}$.

Let L *denote the graph Laplacian, and* $\sigma_1, ..., \sigma_m$ *denote the* m *eigenvalues of* L *that lie in* $\mathcal{I}$. *Then there exists an orthonormal basis* $\{v_1, ..., v_m\}$ *of associated eigenvectors such that, if* $\mathcal{N}_{\mathcal{I}}^1 = |\{i : \text{supp}(v_i) \subset C_1\}|$ *and* $\mathcal{N}_{\mathcal{I}}^2 = |\{i : \text{supp}(v_i) \subset C_2\}|$, *then* $\mathcal{N}_{\mathcal{I}}^1 + \mathcal{N}_{\mathcal{I}}^2 = m$ *and there exists some* N *such that* $\forall n > N$ *numbers of points,*

$$|\mathcal{N}_{\mathcal{I}}^1 - D\mathcal{N}_{\mathcal{I}}^2| \leq 2\delta n \frac{k}{\sqrt{k-1}}|\mathcal{I}|$$

with probability at least $1 - o(1/n)$ *over the choice of subgraphs* G_1 *and* G_2.

Moreover, m *satisfies*

$$|m - (n + \frac{n}{D}) \int_{\mathcal{I}} f_d(x)dx| < \delta(n + \frac{n}{D}) \frac{k}{\sqrt{k-1}}|\mathcal{I}|,$$

again with probability at least $1 - o(1/n)$.

4.3 Proof of Theorem 4.2.4

To prove this theorem, we shall use a couple of basic lemmas.

Lemma 4.3.1. *Let* $\Gamma = (\Omega, E)$ *be a graph that can be separated into two disjoint components* G_1 *and* G_2 *of size* $|G_1| = n$ *and* $|G_2| = m$. *Let* A *be the adjacency matrix for* Γ, *and* A_1 *and* A_2 *be the adjacency matrices for* G_1 *and* G_2, *respectively. Then*

1. *the spectrum of* A *satisfies* $\sigma(A) = \sigma(A_1) \cup \sigma(A_2)$, *and*
2. *if the eigenpairs of* A_1 *and* A_2 *are* $\{(\lambda_1^i, v_1^i)\}_{i=1}^n$ *and* $\{(\lambda_2^j, v_2^j)\}_{j=1}^m$ *respectively, then* $\{(\lambda_1^i, [(v_1^i)^\intercal, 0]^\intercal)\}_{i=1}^n$ *and* $\{(\lambda_2^j, [0, (v_2^j)^\intercal]^\intercal)\}_{j=1}^m$ *are corresponding eigenpairs of* A.

Proof. Let G_1 have n nodes, and G_2 have m nodes. By assumption

$$A = \begin{pmatrix} A_1 & 0 \\ 0 & A_2 \end{pmatrix}.$$

Notice that v_i is an eigenvector of A_1 if and only if $[v_i, 0]^\intercal$ is an eigenvector of A, and both share the same eigenvalue λ_i. This accounts for n eigenvalues of A. And since the same holds for A_2 and $[0, v_i]^\intercal$, we have accounted for another m eigenvalues of A. But since A is an $(n+m) \times (n+m)$ matrix, these are the only possible eigenvalues of A. Also, $\{[(v_1^i)^\intercal, 0]^\intercal\}_{i=1}^n$ and $\{[0, (v_2^j)^\intercal]^\intercal\}_{j=1}^m$ clearly form an eigenbasis for A. □

Lemma 4.3.2. *Let* $\Gamma = (\Omega, E)$ *be a* k*-regular graph with adjacency matrix* A. *Then*

1. *the normalized Laplacian satisfies* $L = I - \frac{1}{k}A$, *and*

2. $\sigma(L) = \{1 - \frac{\sqrt{k-1}}{k}\lambda : \lambda \in \sigma(\frac{1}{\sqrt{k-1}}A)\}$, *and*

3. $\frac{1}{\sqrt{k-1}}Av = \lambda v \iff Lv = (1 - \frac{\sqrt{k-1}}{k}\lambda)v$.

Proof. 1. By definition, $L = I - D^{-1/2}AD^{-1/2}$, where D is a diagonal matrix such that $D_{i,i} = \sum_j w_{i,j}$. However, for a k-regular graph, each node satisfies $\deg(x_i) = \sum_j w_{i,j} = k$. Thus $L = I - (kI)^{-1/2}A(kI)^{-1/2} = I - \frac{1}{k}A$.

2. Let $\lambda \in \sigma(\frac{1}{\sqrt{k-1}}A)$, meaning $\det(\lambda I - \frac{1}{\sqrt{k-1}}A) = 0$. Then

$$\begin{aligned}
&\det(\lambda I - \frac{1}{\sqrt{k-1}}A) = 0 \\
&\implies \det(\frac{\sqrt{k-1}}{k}\lambda I - \frac{1}{k}A) = 0 \\
&\implies \det((\frac{\sqrt{k-1}}{k}\lambda - 1)I - (\frac{1}{k}A - I)) = 0 \\
&\implies \det(-\widetilde{\lambda}I + L) = 0,
\end{aligned}$$

   where $\widetilde{\lambda} = 1 - \frac{\sqrt{k-1}}{k}\lambda$, and $\widetilde{\lambda}$ is an eigenvalue of L.

3. By definition,

$$\begin{aligned}
Lv &= (1 - \frac{\sqrt{k-1}}{k}\lambda)v \\
&\iff (I - \frac{1}{k}A)v = v - \frac{\sqrt{k-1}}{k}\lambda v \\
&\iff Av = \sqrt{k-1}\lambda v \\
&\iff \frac{1}{\sqrt{k-1}}Av = \lambda v.
\end{aligned}$$

□

Proof of Theorem 4.2.4. The proof of this theorem relies heavily on Theorem 4.2.3. Note, k can be kept constant for both G_1 and G_2 by setting $k = \log(n)^{\gamma_1} = \log(n/D)^{\gamma_2}$. Choose $\alpha < \min\{1, 1/\gamma_1, 1/\gamma_2\}$, and set η accordingly. Now for both G_1 and G_2, the parameters are set and constant across the clusters.

Let A be the adjacency matrix of Γ. By Lemma 4.3.1, $\sigma(A) = \sigma(A_1) \cup \sigma(A_2)$. Also, by Lemma 4.3.2, $\sigma(L) = \left(1 - \frac{\sqrt{k-1}}{k}\sigma(\frac{1}{\sqrt{k-1}}A_1)\right) \cup \left(1 - \frac{\sqrt{k-1}}{k}\sigma(\frac{1}{\sqrt{k-1}}A_2)\right)$.

Let $\mathcal{I} \subset [0, 2]$, and for notation let $\mathcal{I}_A = \frac{k}{\sqrt{k-1}}(1 - \mathcal{I})$ be the corresponding interval for $\sigma(\frac{1}{\sqrt{k-1}}A)$. Now let $\lambda_i \in \mathcal{I}_A$ and v_i be the associated eigenvector. Then by Lemma 4.3.1, if $\lambda_i \in \sigma(\frac{1}{\sqrt{k-1}}A_1)$ then $\text{supp}(v_i) \subset C_1$ and $i \in \mathcal{N}^1_{\mathcal{I}_A}$. And if $\lambda_i \in \sigma(\frac{1}{\sqrt{k-1}}A_2)$ then $\text{supp}(v_i) \subset C_2$ and $i \in \mathcal{N}^2_{\mathcal{I}_A}$.

By Lemma 4.3.2, $\frac{1}{\sqrt{k-1}}Av_i = \lambda_i v_i \iff Lv_i = (1 - \frac{\sqrt{k-1}}{k}\lambda_i)v_i$, so we know $(1 - \frac{\sqrt{k-1}}{k}\lambda_i)$ is an eigenvalue of L with the corresponding eigenvector v_i remaining unchanged. This means $\mathcal{N}^1_{\mathcal{I}} = \mathcal{N}^1_{\mathcal{I}_A}$ and $\mathcal{N}^2_{\mathcal{I}} = \mathcal{N}^2_{\mathcal{I}_A}$.

Since $i \in \mathcal{N}^1_{\mathcal{I}_A} \iff \lambda_i \in \sigma(\frac{1}{\sqrt{k-1}}A_1)$ (and same for A_2), Theorem 4.2.3 guarantees us that

$$|\mathcal{N}^1_{\mathcal{I}_A} - n\int_{\mathcal{I}_A} f_d(x)dx| < n\delta|\mathcal{I}_A|,$$
$$|\mathcal{N}^2_{\mathcal{I}_A} - \frac{n}{D}\int_{\mathcal{I}_A} f_d(x)dx| < \frac{n}{D}\delta|\mathcal{I}_A|.$$

This means

$$\begin{aligned}
|\mathcal{N}_{\mathcal{I}}^1 - D\mathcal{N}_{\mathcal{I}}^2| &= |\mathcal{N}_{\mathcal{I}_A}^1 - D\mathcal{N}_{\mathcal{I}_A}^2 + n\int_{\mathcal{I}_A} f_d(x)dx - n\int_{\mathcal{I}_A} f_d(x)dx| \\
&\leq |\mathcal{N}_{\mathcal{I}_A}^1 - n\int_{\mathcal{I}_A} f_d(x)dx| + |D\mathcal{N}_{\mathcal{I}_A}^2 - n\int_{\mathcal{I}_A} f_d(x)dx| \\
&\leq 2n\delta|\mathcal{I}_A| \\
&= 2n\delta\frac{k}{\sqrt{k-1}}|\mathcal{I}|.
\end{aligned}$$

As for a bound on m, we know $m = \mathcal{N}_{\mathcal{I}}^1 + \mathcal{N}_{\mathcal{I}}^2$ since $\{i : \text{supp}(v_i) \subset C_1\} \cap \{i : \text{supp}(v_i) \subset C_2\} = \emptyset$. This means

$$\begin{aligned}
|m - (n + \frac{n}{D})\int_{\mathcal{I}} f_d(x)dx| &= |\mathcal{N}_{\mathcal{I}}^1 + \mathcal{N}_{\mathcal{I}}^2 - (n + \frac{n}{D})\int_{\mathcal{I}} f_d(x)dx| \\
&\leq |\mathcal{N}_{\mathcal{I}}^1 - n\int_{\mathcal{I}} f_d(x)dx| + |\mathcal{N}_{\mathcal{I}}^2 - \frac{n}{D}\int_{\mathcal{I}} f_d(x)dx| \\
&\leq \delta(n + \frac{n}{D})\frac{k}{\sqrt{k-1}}|\mathcal{I}|.
\end{aligned}$$

□

4.4 Weakly Connected Clusters with Heterogeneous Sizes

In applications of dimension reduction techniques, it is unlikely that clusters are disjoint. However, Theorem 4.2.4 serves as a first step in the direction of a theory of eigenvector localization for Laplacian Eigenmaps.

The next question that arises concerns the behavior weakly connected clusters with heterogeneous sizes (ie. when there exist a small number of edges between the clusters). This characterizes a larger and more realistic class of data analysis problems.

Definition 4.4.1. *A graph with* weakly connected clusters of order t *is a connected graph with adjacency matrix*

$$A = \begin{pmatrix} A_1 & B_{1,2} \\ B_{1,2}^{\intercal} & A_2 \end{pmatrix},$$

where $B_{1,2}$ has t non-zero entries, and A_1 and A_2 are adjacency matrices of k-regular graphs.

This definition is equivalent to characterizing a graph with two clusters, and t edges linking the two clusters. We now characterize the eigenvalues and eigenvectors of a graph with weakly connected clusters as a problem of matrix perturbation. Consider two graphs H and G, where H is a disjoint regular graph from the assumptions of Theorem 4.2.4, and G is a graph with weakly connected clusters of order t. In other words, the adjacency matrix $A_H = \begin{pmatrix} A_1 & 0 \\ 0 & A_2 \end{pmatrix}$ and $A_G = \begin{pmatrix} A_1 & B_{1,2} \\ B_{1,2}^{\intercal} & A_2 \end{pmatrix}$.

Then $A_G = A_H + B$, where B is a block 2×2, $2t$ sparse adjacency matrix that only has terms on the block off-diagonal. Clearly, one can see A_G as a perturbed version of A_H, and the eigenvalues and eigenvectors of A_H are completely characterized by Theorem 4.2.4. This makes perturbation theory a valid approach to showing the eigenvalues and eigenvectors of A_G (and the graph Laplacian L_G) do not deviate much from the known quantities of A_H.

4.4.1 Eigenvalue Distribution

First, we shall consider the eigenvalue distribution of this new perturbed matrix.

Theorem 4.4.2. *(Theorem 2.4, [32]) Let G be a graph with weakly connected clusters of order t and H be the graph of the two disjoint clusters. If*

$$\lambda_1 \leq ... \leq \lambda_n, \text{ and}$$

$$\theta_1 \leq ... \leq \theta_n$$

are the eigenvalues of the normalized Laplacians L_G and L_H respectively, then

$$\theta_{i-t} \leq \lambda_i \leq \theta_{i+t}, \qquad 1 \leq i \leq n,$$

with the convention that $\theta_{-t} = ... = \theta_0 = 0$ and $\theta_{n+1} = ... = \theta_{n+t} = 2$.

Theorem 4.4.2 guarantees the order of the eigenvalues shall not deviate much from Theorem 4.2.4. Specifically, it leads to the following lemma.

Lemma 4.4.3. *Let $\Gamma = (\Omega, E)$ be a graph with weakly connected clusters of order t, such that one cluster is of size n and the other cluster is of size $\frac{n}{D}$. Fix δ, k, α, η, and $\mathcal{I}$ as in Theorem 4.2.4.*

Let L denote the graph Laplacian, and $\sigma_1, ..., \sigma_m$ denote the m eigenvalues of L that lie in $\mathcal{I}$. Then m satisfies

$$|m - (n + \frac{n}{D}) \int_{\mathcal{I}} f_d(x)dx| < \delta(n + \frac{n}{D}) \frac{k}{\sqrt{k-1}} |\mathcal{I}| + 2t,$$

again with probability at least $1 - o(1/n)$.

Proof. Let G be a graph with weakly connected clusters of order t and H be the graph of the two disjoint clusters, with Laplacians L_G and L_H respectively. Let

$$\lambda_{min} = \min\{\lambda \in \sigma(L_H) : \lambda \in \mathcal{I}\}, \qquad \lambda_{max} = \max\{\lambda \in \sigma(L_H) : \lambda \in \mathcal{I}\},$$

$$\widetilde{\lambda}_{min} = \min\{\widetilde{\lambda} \in \sigma(L_G) : \widetilde{\lambda} \in \mathcal{I}\}, \qquad \widetilde{\lambda}_{max} = \max\{\widetilde{\lambda} \in \sigma(L_G) : \widetilde{\lambda} \in \mathcal{I}\}.$$

By Theorem 4.4.2,

$$N_{min} = |\{\widetilde{\lambda} \in \sigma(L_G) : \widetilde{\lambda}_{min} \leq \widetilde{\lambda} \leq \lambda_{min} \text{ or } \lambda_{min} \leq \widetilde{\lambda} \leq \widetilde{\lambda}_{min}\}| \leq t,$$

$$N_{max} = |\{\widetilde{\lambda} \in \sigma(L_G) : \lambda_{max} \leq \widetilde{\lambda} \leq \widetilde{\lambda}_{max} \text{ or } \widetilde{\lambda}_{max} \leq \widetilde{\lambda} \leq \lambda_{max}\}| \leq t.$$

Then if m_G (resp. m_H) is the number of eigenvalues of L_G (resp. L_H) in $\mathcal{I}$,

$$\begin{aligned}
|m_G - (n + \frac{n}{D}) \int_{\mathcal{I}} f_d(x)dx| &= |m_G - m_H + m_H - (n + \frac{n}{D}) \int_{\mathcal{I}} f_d(x)dx| \\
&\leq |m_G - m_H| + |m_H - (n + \frac{n}{D}) \int_{\mathcal{I}} f_d(x)dx| \\
&\leq N_{min} + N_{max} + |m_H - (n + \frac{n}{D}) \int_{\mathcal{I}} f_d(x)dx| \\
&\leq 2t + \delta(n + \frac{n}{D}) \frac{k}{\sqrt{k-1}} |\mathcal{I}|.
\end{aligned}$$

□

4.4.2 Eigenvector Distribution

Now, we shall consider the eigenvector distribution of a graph with weakly connected clusters by considering it as a matrix perturbation problem. Davis and Kahan [41, 42] were the first to give general theorems relating to the invariant subspaces of two Hermitian matrices. These results were extended by Stewart [105] via an iterative process for generating the invariant subspaces. For a detailed account of matrix perturbation results, see [106].

These theories are center around the distribution of eigenvalues and eigenvalue gaps.

Definition 4.4.4. *The* eigenvalue separation *of two $n \times n$ matrices A and B with spectrum $\sigma(A) = \{\lambda_1^A, ..., \lambda_n^A\}$ and $\sigma(B) = \{\lambda_1^B, ..., \lambda_n^B\}$. Then the separation is*

defined as

$$\text{sep}(A, B) = \min_{i,j} |\lambda_i^A - \lambda_j^B|.$$

Theorem 4.4.5. *(Theorem 4.11, [105]) Let $A, E \in \mathbb{C}^{n\times n}$. Let $X = [X_1, X_2]$ be a unitary matrix with $X_1 \in \mathbb{C}^{n\times l}$, and suppose $\mathcal{R}(X_1)$ is an invariant subspace of A. Let*

$$X^*AX = \begin{pmatrix} A_{1,1} & A_{1,2} \\ 0 & A_{2,2} \end{pmatrix}, \quad X^*EX = \begin{pmatrix} E_{1,1} & E_{1,2} \\ E_{2,1} & E_{2,2} \end{pmatrix}.$$

Let $\delta = \text{sep}(A_{1,1}, A_{2,2}) - \|E_{1,1}\| - \|E_{2,2}\|$. Then if

$$\frac{\|E_{2,1}\|(\|A_{1,2}\| + \|E_{1,2}\|)}{\delta^2} \leq \frac{1}{4}, \tag{4.2}$$

there is a matrix P satisfying

$$\|P\| \leq 2\frac{\|E_{2,1}\|}{\delta}$$

such that

$$\widetilde{X_1} = (X_1 + X_2P)(I + P^*P)^{-1//2} \tag{4.3}$$

is an invariant subspace of $A + E$.

Theorem 4.4.5 gives a sufficient condition for guaranteeing an eigenspace X_1 remains relatively preserved under perturbation. Under the condition that A is a graph with weakly connected clusters, and $\text{sep}(A_{1,1}, A_{2,2}) \neq 0$, Theorem 4.4.5 gives us bounds the individual eigenvectors under perturbation.

This type of theorem is an approach to showing the eigenvectors of a graph with weakly connected clusters remains localized. It implies the fact that greater

eigenvalue separation leads to better eigenvector localization. However, the conditions that need to be satisfied are too strict for our problem.

To demonstrate this disparity between theory and example, consider the problem in Figure 4.4. In this dataset, there are 7 edges connecting C_1 and C_2. $|C_1| = 1989$ and $|C_2| = 211$, meaning $|C_1| = D \cdot |C_2|$ where $D = 9.4$. We shall examine the smallest 10% of eigenvalues and their associated eigenvectors, as these are the vectors that are chosen for the Laplacian Eigenmaps algorithm.

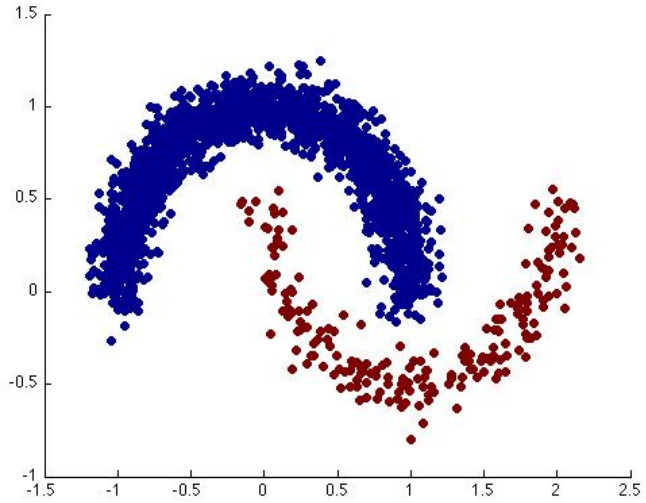
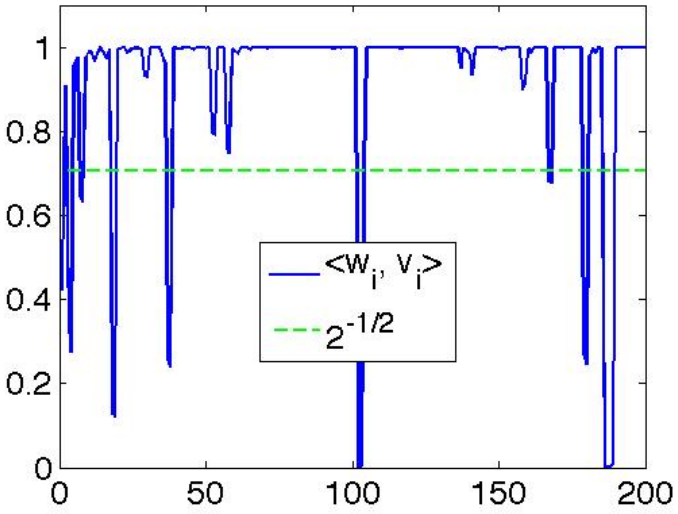


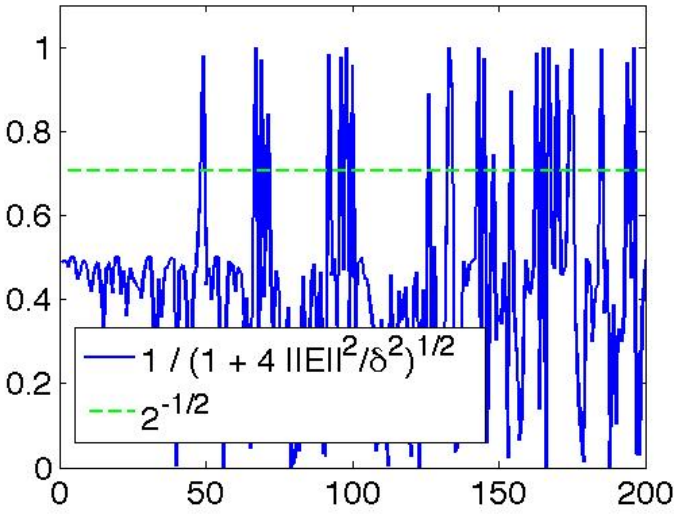
Figure 4.4: *Two Moons Weakly Connected.* $|C_1| = 1989$, $|C_2| = 211$

Let L_G be the Laplacian of the graph with weakly connected clusters, and L_H be the Laplacian of the graph with disjoint clusters. Let $\{v_1, ..., v_{200}\}$ be the eigenvectors of L_G and $\{w_1, ..., w_{200}\}$ be the eigenvectors of L_H. Figure 4.5 plots $\langle v_i, w_i \rangle$ for $i \in \{1, ..., 200\}$. Clearly, there is a large discrepancy between theory and practice. Theorem 4.4.5 only predicts 26 eigenvectors satisfy the assumptions of the spectral gap necessary to guarantee (4.3) holds for $\|P\| < 1$ (which would imply

$\langle v_i, w_i \rangle > \frac{\sqrt{2}}{2}$). However, 180 of the eigenvectors actually satisfy $\langle v_i, w_i \rangle > \frac{\sqrt{2}}{2}$.



(a) Vector Angles



(b) Vector Angles Predicted by Theorem 4.4.5

Figure 4.5: *Actual Vector Angles* $\langle v_i, w_i \rangle$ *for the first* 200 *eigenvectors of data from Figure 4.4 versus Predicted spectral gap from Theorem 4.4.5.*

A more enlightening depiction of this discrepancy can be seen in Figure 4.6.

This is another plot of the vector angles (same as Figure 4.5(a)), except now the indices for which $\text{supp}(w_i) \subset C_2$ are marked with a vertical line. Recall from Theorem 4.2.4, this occurs on average once out of every D indices.

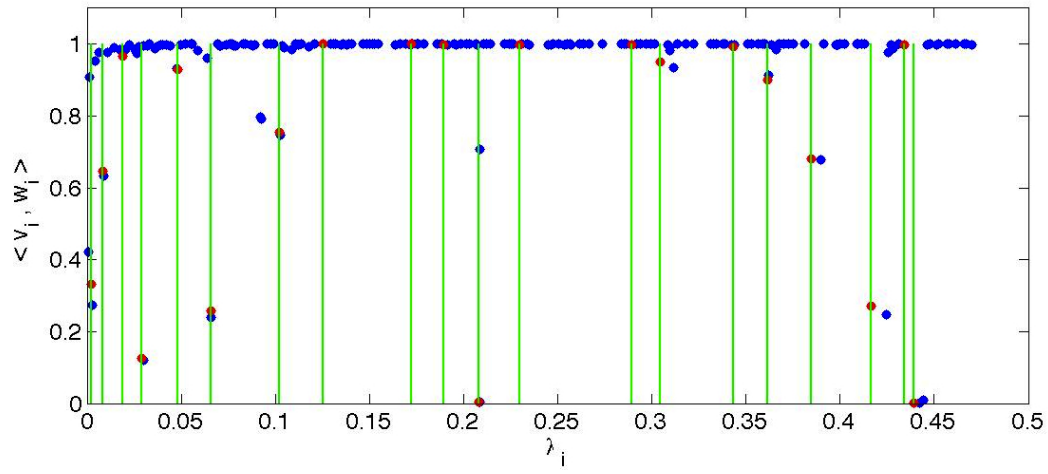


Figure 4.6: *Vector Angles for first* 200 *eigenvectors of data from Figure 4.4, with green vertical lines denoting eigenvalues for which* $\lambda_i \in \{\lambda_i : \text{supp}(w_i) \subset C_2\}$. *Blue dot:* $\{\langle w_i, v_i\rangle : \text{supp}(w_i) \subset C_1\}$, *Red dot:* $\{\langle w_i, v_i\rangle : \text{supp}(w_i) \subset C_2\}$.

Notice that the only deviations $\langle v_i, w_i\rangle$ make from the neighborhood of 1 occur on or incredibly near the indices for which $\text{supp}(w_i) \subset C_2$. This suggests why Theorem 4.4.5 is not sufficient for the current setting. Theorem 4.4.5 gives a condition for which $\|P\| < 1$. However, it does not speak to *which* eigenvectors from X_2 contribute to $\widetilde{X_1}$ in (4.3), regardless of whether (4.2) is violated.

Figure 4.6 suggests that the eigenvectors from X_2 that contribute to $\widetilde{X_1}$ are exactly those that are nearest in eigenvalue. This is why only points near a vertical line for $\lambda_i \in \{\lambda_i : \text{supp}(w_i) \subset C_2\}$ have vector angles far from 1. This leads to the following conjecture:

Conjecture 4.4.6. *Let $A, E \in \mathbb{C}^{n\times n}$. Let $X = [X_1, X_2]$ be a unitary matrix with $X_1 = [X_1^1, ..., X_1^l] \in \mathbb{C}^{n\times l}$. Suppose $\mathcal{R}(X_1)$ is an invariant subspace of A associated with eigenvalue λ, and the columns $X_2 = [X_2^1, ..., X_2^{n-l}]$ correspond to eigenvalues $[\lambda_1, ..., \lambda_{n-l}]$. Let*

$$X^*AX = \begin{pmatrix} A_{1,1} & A_{1,2} \\ 0 & A_{2,2} \end{pmatrix}, \quad X^*EX = \begin{pmatrix} E_{1,1} & E_{1,2} \\ E_{2,1} & E_{2,2} \end{pmatrix}.$$

Let $\epsilon > 0$ satisfy some minimum size condition depending on $\|E\|$. Then if $\exists C_1 \subsetneq \{1, ..., n\}$ such that $supp(X_1^i) \subset C_1$ for $i = 1, ..., l$ and

$$\lambda_i \in (\lambda - \epsilon, \lambda + \epsilon) \implies \text{supp}(X_2^i) \subset C_1,$$

there is a matrix P such that $\widetilde{X_1} = (X_1 + X_2P)(I + P^*P)^{-1//2}$ is an invariant subspace of $A + E$, and $supp(\widetilde{X_1^i}) \subset C_1$ for $i \in \{1, ..., l\}$.

It may be the case that Conjecture 4.4.6 must be refined to only refer to symmetric diagonally dominant matrices, of which graph Laplacians are an example.

4.5 Partial Result in the Direction of Conjecture 4.4.6

For this section, we shall utilize the singular value decomposition (SVD) instead of the eigendecomposition. To avoid ambiguity, we shall define the singular values of a matrix A to be $\sigma_1 \geq ... \geq \sigma_n \geq 0$. Also, we shall define the SVD $A = U\Sigma V^*$ by calculating the right singular vectors V and the singular values Σ from the eigendecomposition of A^*A and the right singular values U via $U = A * V * \Sigma^\dagger$.

Lemma 4.5.1. *Let A be an $n\times n$ symmetric real matrix. Let the SVD of $A = U\Sigma V^*$ and the eigendecomposition of $A = XDX^{-1}$. Then $V = X$.*

Proof. Because A is symmetric, its eigenvectors form an orthonormal basis for $\mathbb{R}^n$. This means $X^{-1} = X^*$ since $XX^* = X^*X = Id_{n\times n}$.

The right singular vectors of A are calculated via the eigendecomposition of A^*A. However, $A^*A = (XDX^*)^*(XDX^*) = X(DD^*)X^*$. This means the eigenvectors of A^*A, and thus the right singular vectors of A, are X. Thus, $V = X$. □

Let λ be an eigenvalue of A. Our results shall refer to the singular vectors (and thus eigenvectors) of $\widetilde{A} = A - \lambda I$. The reason for this is that, for the SVD of $\widetilde{A}$, we now the singular value $\sigma_n = 0$ as λ is an eigenvalue of A. This means, to speak of the eigenvector of A corresponding to λ, we may simply refer to the singular vector v_n corresponding to $\sigma_n = 0$. This result is along the lines of Theorem 3 in [100].

Theorem 4.5.2. *Let A by an $n \times n$ matrix with SVD $A = U\Sigma V^*$. Partition $V = [V_1, V_2, v_n]$ where $v_n \in \mathbb{R}^n$, $V_2 \in \mathbb{R}^{n\times s}$. Moreover, assume $\exists C \subsetneq \{1, ..., n\}$ such that supp$(v_i) \subset C$ for $i \in \{n - s, ..., n\}$. Let $x \in \mathbb{R}^n$ such that $\|x\|_2 = 1$. Then*

$$\sum_{i\in C^c} |x_i|^2 \leq \frac{\|Ax\|_2^2 - \|Av_n\|_2^2}{\sigma_{n-s-1}^2(A) - \sigma_n^2(A)}.$$

Proof. Let $x = V_1c_1 + V_2c_2 + v_nc_3$ where $c_1 \in \mathbb{R}^{n-s-1}$, $c_2 \in \mathbb{R}^s$, and $c_3 \in \mathbb{R}$. We shall bound the quantity $\|c_1\|$ to arrive at our desired result. Partition U and Σ in the same way as V. Recalling that U and V have orthogonal columns,

$$\begin{aligned}
\|Ax\|_2^2 &= \|U_1\Sigma_1V_1^*x + U_2\Sigma_1V_2^*x + u_n\sigma_nv_n^*x\|_2^2 \\
&= \|\Sigma_1c_1\|_2^2 + \|\Sigma_2c_2\|_2^2 + \|\sigma_nc_3\|_2^2 \\
&\geq \sigma_{n-s-1}^2\|c_1\|^2 + \sigma_{n-1}^2\|c_2\|^2 + \sigma_n^2|c_3|^2,
\end{aligned}$$

where the inequality comes from the fact that $\sigma_1 \geq ... \geq \sigma_n \geq 0$. Now note that

$$|c_3|^2 = 1 - \|c_1\|_2^2 - \|c_2\|_2^2,$$

since $x^*x = c_1^*c_1 + c_2^*c_2 + c_3^*c_3 = 1$. This means

$$\|Ax\|_2^2 \geq (\sigma_{n-s-1}^2 - \sigma_n^2)\|c_1\|_2^2 + (\sigma_{n-1}^2 - \sigma_n^2)\|c_2\|_2^2 + \sigma_n^2.$$

Noting that $\|Av_n\|_2^2 = \sigma_n^2$ and $(\sigma_{n-1}^2 - \sigma_n^2)\|c_2\|_2^2 > 0$, we clearly see

$$\begin{aligned} \|Ax\|_2^2 - \|Av_n\|_2^2 &\geq (\sigma_{n-s-1}^2 - \sigma_n^2)\|c_1\|_2^2 \\ \implies \|c_1\|_2^2 &\leq \frac{\|Ax\|_2^2 - \|Av_n\|_2^2}{\sigma_{n-s-1}^2 - \sigma_n^2}. \end{aligned}$$

Now recall that $(V_2)_{i,j} = (v_n)_i = 0$ for all $i \notin C$ and $j \in \{1, ..., s\}$. This means

$$\begin{aligned} \sum_{i \in C^c} |x_i|^2 &\leq \sum_{i=1}^{n} \sum_{j=1}^{n-s-1} |(V_1)_{i,j} c_j|^2 \\ &= \sum_{j=1}^{n-s-1} |c_j|^2 \sum_{i=1}^{n} |(V_1)_{i,j}|^2 \\ &= \sum_{j=1}^{n-s-1} |c_j|^2 \\ &\leq \frac{\|Ax\|_2^2 - \|Av_n\|_2^2}{\sigma_{n-s-1}^2 - \sigma_n^2}. \end{aligned}$$

□

Theorem 4.5.2 demonstrates that, when there exists a series of singular vectors concentrated on a subset of the points C, then the angle between those singular vectors to a new vector is inversely proportional to the square of the singular value. This, along with Lemma 4.5.1, lead to following immediate corollary.

Corollary 4.5.3. *Let A be a symmetric $n \times n$ matrix with eigendecomposition $A = V\Sigma V^*$. Let (λ_i, v_i) be an eigenpair of A. Partition $V = [V_1, V_2, v_i, V_3, V_4]$ where $V_2, V_3 \in \mathbb{R}^{n\times s}$. Moreover, assume $\exists C \subsetneq \{1, ..., n\}$ such that supp$(v_i) \subset C$ and supp$(v_j) \subset C$ where v_j is a column of V_2, V_3. Let $(\widetilde{\lambda}, x)$ an eigenvector of the perturbed matrix $A + E$, where $x = [x_1, ..., x_n]$. Then*

$$\sum_{j\in C^c} |x_j|^2 \leq \frac{\|(\widetilde{\lambda} - \lambda_i)x - Ex\|_2^2}{\min(\lambda_i - \lambda_{i-s}, \lambda_{i+s} - \lambda_i)^2}.$$

Proof. We applying Theorem 4.5.2 to the matrix $A - \lambda_i I$. Lemma 4.5.1 allows us to shift the problem from singular vectors to eigenvectors. The only other change of note lies in the denominator. The denominator is

$$\|(A - \lambda_i I)x\|_2^2 - \|(A - \lambda_i I)v_i\|_2^2.$$

We notice the right hand term is 0 as (λ_i, v_i) is an eigenpair. For the left hand term, we notice

$$\|(A - \lambda_i I)x\|_2^2 = \|(A + E)x - Ex - \lambda_i x\|_2^2.$$

Since $(A + E)x = \widetilde{\lambda}x$, the desired result follows immediately. □

Using Corollary 4.5.3, we attempt to predict the number of eigenvectors from Figure 4.4 that remain concentrated in the appropriate cluster. Recall that 180 of the first 200 eigenvectors satisfied $\langle v_i, w_i\rangle > \frac{\sqrt{2}}{2}$, which is a similar condition to predicting that $\sum_{j\in C^c} |x_j|^2 < .5$. Corollary 4.5.3 predicts that 130 of the eigenvectors will satisfy $\sum_{j\in C^c} |x_j|^2 < .5$ and remain concentrated in their respective clusters. While this is less than the 180 that actually remain localized, the prediction of 130 is far better than the prediction of 26 that occurs using Theorem 4.4.5. More importantly,

of those 130 predicted eigenvectors, 127 are concentrated on the larger cluster C_1. Only 3 are concentrated on the smaller cluster C_2.

4.6 Interpretations and Conclusions

Let us assume there is a positive result to Conjecture 4.4.6. Then the results of Theorem 4.2.4, along with Figure 4.6 and Conjecture 4.4.6, suggest a negative result for differentiating small clusters C_2 from a larger background cluster C_1 using Laplacian Eigenmaps. These results would suggest that small clusters are forced to 0 for most eigenvectors of the graph Laplacian. This makes classification, and especially determining inter-cluster differences, very difficult.

On top of that, Conjecture 4.4.6 suggests that even if eigenvalue λ_i has an eigenvector w_i supported on C_2 for the disjoint graph, its corresponding eigenvector for the graph with weakly connected clusters v_i may not remain supported on C_2. This is because, while $\text{supp}(v_i) \subset C_2$, $\text{supp}(v_{i-1})$ and $\text{supp}(v_{i+1})$ are most likely concentrated on C_1 due to Theorem 4.2.4.

Chapter 5: Solving 2D Fredholm Integral from Incomplete Measurements for Improved Acquisition of NMR Spectra

5.1 Introduction

5.1.1 2D Fredholm Integral

We present a method of solving the 2-dimensional Fredholm integral of the first kind from a limited number of measurements. This is particularly useful in the field of nuclear magnetic resonance (NMR), in which making a sufficient number of measurements takes several hours. Our work is an extension of the algorithm in [111] based on the new idea of matrix completion, cf. [21, 63, 98].

A two-dimensional Fredholm integral of the first kind is written as

$$g(x,y) = \int\int k_1(x,s)k_2(y,t)f(s,t)dsdt,$$

where k_1 and k_2 are continuous Hilbert-Schmidt kernel functions and $f, g \in L^2(\mathbb{R}^2)$, cf. [61]. Two dimensional Fourier, Laplace, and Hankel transforms are all common examples of Fredholm integral equations. Applications of these transformations arise in any number of fields, including methods for solving PDEs [65], image deblurring [15, 75], and moment generating functions [76]. This chapter specifically focuses on

Laplace type transforms, where the kernel singular values decay quickly to zero.

To present the main idea of the problem, the data M is measured over sampling times τ_1 and τ_2, and is related to the object of interest $\mathcal{F}(x, y)$ by a 2-D Fredholm integral of the first kind with a tensor product kernel,

$$M(\tau_1, \tau_2) = \int\int k_1(x, \tau_1) k_2(y, \tau_2) \mathcal{F}(x, y) dx dy + \epsilon(\tau_1, \tau_2),$$

where $\epsilon(\tau_1, \tau_2)$ is assumed to be Gaussian, white noise. In most applications, including NMR, the kernels k_1 and k_2 are explicit functions that are known to be smooth and continuous a priori. Solving a Fredholm integral with smooth kernels is an ill-conditioned problem, since the kernel's singular values decay quickly to zero [69]. This makes the problem particularly interesting, as small variations in the data can lead to large fluctuations in the solution.

For our purposes, $\mathcal{F}(x, y)$ represents the joint probability density function of the variable x and y. Specifically in NMR, x and y can be the measurements of the two combination of the longitudinal relaxation time T1, transverse relaxation time T2, diffusion D and other dynamic properties. Knowledge of the correlation of these properties of a sample is used to identify its microstructure properties and dynamics [17].

This chapter focuses on the discretized version of the 2D Fredholm integral,

$$M = K_1 F K_2' + E, \tag{5.1}$$

where our data is the matrix $M \in \mathbb{R}^{N_1 \times N_2}$, matrices $K_1 \in \mathbb{R}^{N_1 \times N_x}$ and $K_2 \in \mathbb{R}^{N_2 \times N_y}$ are discretized versions of the smooth kernels k_1 and k_2, and the matrix $F \in \mathbb{R}^{N_x \times N_y}$ is the discretized version of the probability density function $\mathcal{F}(x, y)$ we are interested

in recovering. We also assume that each element of the Gaussian noise matrix E is zero mean and constant variance. And since we have assumed that $\mathcal{F}(x, y)$ is a joint pdf, each element of F is non-negative.

5.1.2 Existing Algorithm in [111]

Venkataramanan, Song, and Hürlimann [111] laid out an efficient strategy for solving this problem given complete knowledge of the data matrix M. The approach centers around finding an intelligent way to solve the Tikhonov regularization problem,

$$\widehat{F} = \arg\min_{F \geq 0} \|M - K_1 F K_2'\|_F^2 + \alpha \|F\|_F^2, \tag{5.2}$$

where $\|\cdot\|_F$ is the Frobenius norm.

There are three steps to the algorithm in [111] for solving (5.2).

1. *Compress the Data:* Let the SVD of K_i be

$$K_i = U_i S_i V_i', \quad i \in \{1, 2\}.$$

Because K_1 and K_2 are sampled from smooth functions k_1 and k_2, the singular values decay quickly to 0. Let s_1 be the number of non-zero singular values of K_1 and s_2 number of non-zero singular values of K_2. Then $U_i \in \mathbb{R}^{N_i \times s_i}$ and $S_i \in \mathbb{R}^{s_i \times s_i}$ for $i = 1, 2$, as well as $V_1 \in \mathbb{R}^{N_x \times s_1}$ and $V_2 \in \mathbb{R}^{N_y \times s_2}$.

The data matrix M can be projected onto the column space of K_1 and the row space of K_2 by $U_1 U_1' M U_2 U_2'$. We denote this as $\widetilde{M} = U_1' M U_2$. The Tikhonov

regularization problem (5.2) is now rewritten as

$$\widehat{F} = \arg\min_{F\geq 0} \|U_1 \widetilde{M} U_2' - U_1 U_1' K_1 F K_2' U_2 U_2'\|_F^2 \tag{5.3}$$

$$+ \|M\|_F^2 - \|U_1 \widetilde{M} U_2'\|_F^2 + \alpha \|F\|_F^2$$

$$= \arg\min_{F\geq 0} \|\widetilde{M} - (S_1 V_1') F (S_2 V_2')'\|_F^2 + \alpha \|F\|_F^2, \tag{5.4}$$

where (5.4) comes from U_1 and U_2 having orthogonal columns, and the second and third terms in (5.3) being independent of F. The key note here is that $\widetilde{M} \in \mathbb{R}^{s_1 \times s_2}$, which significantly reduces the complexity of the computations.

2. *Optimization:* For a given value of α, (5.4) has a unique solution due to the second term being quadratic. We shall detail the method of finding this solution in Section 5.3.

3. *Choosing α:* Once (5.4) has been solved for a specific α, an update for α is chosen based on the characteristics of the solution in Step 2. Repeat between Steps 2 and 3 until convergence. Again, this is detailed in Section 5.3.

5.1.3 Subsampling NMR Measurements

The approach in [111] assumes complete knowledge of the data matrix M. However, in applications with NMR, there is a cost associated with collecting all the elements of M, which is time. With the microstructure-related information contained in the multidimensional diffusion-relaxation correlation spectra of the biological sample [43, 46, 54, 70, 95, 110] and high-resolution spatial information that magnetic resonance imaging (MRI) technique can provide, there is a need to combine

the multidimensional correlation spectra NMR with 2D/3D MRI for pre-clinical and clinical applications [44]. Without any acceleration, however, it could take several days to acquire this data.

In practice, the potential pulse sequences for the combined multidimensional diffusion-relaxation MRI would be single spin echo (90°-180°-acquisition and spatial localization) with saturation, inversion recovery, driven-equilibrium preparation to measure T1-T2 correlation and diffusion weighting preparation for D-T2 measurements. With these MRI pulse sequences, a single point in the two dimensional T1-T2 or D-T2 space is acquired for each "shot", and the total time for the sampling of the T1-T2 or D-T2 space is determined directly by the number of measurements required to recover F from (5.2). Together with rapid MRI acquisition techniques, which can include, e.g., parallel imaging [96], echo planar imaging (EPI) [53], gradient-recalled echo [71], sparse sampling with compressed sensing [87], along with a vastly reduced number of sample points in M, could reduce the total experiment time sufficiently to make this promising technique practicable for pre-clinical and clinical *in vivo* studies.

Notice that, despite collecting all $N_1 \times N_2$ data points in M, Step 1 of the algorithm immediately throws away a large amount of that information, reducing the number of data points to a matrix of size $s_1 \times s_2$. $\widetilde{M}$ is effectively a *compressed* version of the original M, containing the same information in a smaller number of entries. But this raises the question of why all of M must be collected when a large amount of information is immediately thrown away, since we are only interested in $\widetilde{M}$.

The goal for the rest of this chapter is to use the compressive sensing results from Chapter 2 to reduce the number of measurements required to determine $\widetilde{M}$. This allows for a stable reconstruction of F from a limited number of measurements, and the possibility of reducing the time required for the acquisition of the 2D multidimensional correlated spectra.

5.2 Relation to Parseval Tight Frame Compressive Sensing

For the NMR problem, let us say

$$\begin{aligned} M = &\quad K_1 F K_2' + E \\ = &\quad U_1 \widetilde{M}_0 U_2' + E, \end{aligned} \tag{5.5}$$

where $U_i \in \mathbb{R}^{N_i \times s_i}$, $\widetilde{M} \in \mathbb{R}^{s_1 \times s_2}$, and $E \in \mathbb{R}^{N_1 \times N_2}$. This means that

$$\widetilde{M}_0 = S_1 V_1' F V_2 S_2. \tag{5.6}$$

To subsample the data matrix M, we shall observe it on random entries. Let $\Omega \subset \{1, ...N_1\} \times \{1, ...N_2\}$ be the set of indices where we observe M. For $|\Omega| = m$, let the indices be ordered as $\Omega = \{(i_k, j_k)\}_{k=1}^m$. Then we define the masking operator $\mathcal{A}_\Omega$ as

$$\mathcal{A}_\Omega : \mathbb{R}^{N_1 \times N_2} \to \mathbb{R}^m$$

$$(\mathcal{A}_\Omega(X))_k = X_{i_k, j_k}$$

Recall that the goal is to recover $\widetilde{M}_0$. This means that our actual sampling operator

is

$$\mathcal{R}_\Omega : \mathbb{R}^{s_1 \times s_2} \to \mathbb{R}^m$$

$$\mathcal{R}_\Omega(X) = \mathcal{A}_\Omega(U_1 X U_2')$$

Now the problem of speeding up NMR can be written as an attempt to recover $\widetilde{M_0}$ from measurements

$$y = \mathcal{R}_\Omega(\widetilde{M_0}) + e, \quad \|e\|_2 \leq \epsilon. \tag{5.7}$$

Note that [111] is assuming $\Omega = \{1, ...N_1\} \times \{1, ...N_2\}$, making the sampling operator $\mathcal{R}_\Omega(\widetilde{M_0}) = U_1 \widetilde{M_0} U_2'$.

Then in the notation of this NMR problem, our recovery step takes the form

$$\begin{array}{ll} \min & \|Z\|_* \\ \text{such that} & \|\mathcal{R}_\Omega(Z) - y\|_2 \leq \epsilon. \end{array} \tag{5.8}$$

Now notice that in our problem, ignoring noise, each observation can be written as

$$\begin{aligned} M_{j,k} &= (u_1^j)\widetilde{M_0}(u_2^k)' \\ &= \langle (u_1^j)'(u_2^k), \widetilde{M_0} \rangle, \end{aligned}$$

where u_1^j (resp. u_2^j) is the j^{th} row of U_1 (resp. U_2). Noting that U_1 and U_2 are left orthogonal (ie. $U_i' U_i = Id_{s_i}$), one can immediately show that $\{(u_1^j)'(u_2^k)\}_{(j,k) \in \mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2}}$ forms a Parseval tight frame for $\mathbb{R}^{s_1 \times s_2}$. Also, because K_1 and K_2 are discretized versions of smooth continuous function, $\{(u_1^i)'(u_2^j)\}$ are a bounded norm frame for a reasonable constant μ (further discussion of μ in Section 5.4.2). Thus, $\mathcal{R}_\Omega$ is

generated by randomly selecting measurements from a bounded norm Parseval tight frame.

This means reconstruction of $\widetilde{M_0}$ from $\mathcal{R}_\Omega(\widetilde{M_0})$ meets the assumptions of Theorem 2.3.1 from Chapter 2. Thus, if the number of measurements is

$$m \geq C\mu r n \log^5 n \cdot \log(N_1 N_2),$$

where $n = \max(s_1, s_2)$, we are guaranteed that the solution to (5.8), which we shall denote $\widetilde{M}$, satisfies

$$\|\widetilde{M} - \widetilde{M_0}\|_F \leq C_0 \frac{\|\widetilde{M_0} - \widetilde{M_{0,r}}\|_*}{\sqrt{r}} + C_1 p^{-1/2} \epsilon,$$

where $p = \frac{m}{N_1 N_2}$.

5.3 Inverse 2D Fredholm Integral Algorithm with Nuclear Norm Minimization

The algorithm for solving for F in (5.1) from partial data consists of three steps. An overview of the original algorithm in [111] is in Section 2.1. Our modification and the specifics of each step are detailed below.

1. *Construct $\widetilde{M}$ from Given Measurements:* Let $y = \mathcal{R}_\Omega(\widetilde{M_0}) + e$ be the set of observed measurements, as in (5.7). While Section 2.3 guarantees (5.8) has a stable solution from a limited number of measurements, we can also solve the relaxed Lagrangian form

$$\min_X \mu \|X\|_* + \frac{1}{2} \|\mathcal{R}_\Omega(X) - y\|_2^2. \tag{5.9}$$

To solve (5.9), we use the *Singular Value Thresholding* algorithm from [16,88]. To do this, we need some notation. Let the matrix derivative of the L_2 norm term be written as

$$\begin{aligned} g(X) &= \mathcal{R}_\Omega^*(\mathcal{R}_\Omega(X) - y) \\ &= U_1'(\mathcal{A}_\Omega^*(\mathcal{A}_\Omega(U_1 X U_2') - y))U_2. \end{aligned}$$

We also need the singular value thresholding operator $\mathcal{S}_\nu$ that reduces each singular value of some matrix X by ν. In other words, if the SVD of $X = U\Sigma V'$, then

$$\mathcal{S}_\nu(X) = U\widetilde{\Sigma}V', \quad \widetilde{\Sigma}_{i,j} = \begin{cases} \max(\Sigma_{i,i} - \nu, 0) & i = j, \\ 0 & otherwise. \end{cases}$$

Using this notation, the algorithm can then be written as a simple, two step iterative process. Choose a $\tau > 0$. Then, for any initial condition, solve the iterative process

$$\begin{cases} Y^k = X^k - \tau g(X^k) \\ X^{k+1} = \mathcal{S}_{\tau\mu}(Y^k) \end{cases}. \tag{5.10}$$

The choices of τ and μ are detailed in [88], along with adaptations of this method that speed up convergence. However, this method is guaranteed to converge to the correct solution.

This means that, given partial observations y, the iteration scheme in (5.10) converges to a matrix $\widetilde{M}$, which is a good approximation of $\widetilde{M} + 0$. Once $\widetilde{M}$ has been generated, we recover F by solving

$$\arg\min_{F \geq 0} \|\widetilde{M} - (S_1 V_1')F(S_2 V_2')'\|_F^2 + \alpha\|F\|_F^2. \tag{5.11}$$

2. *Optimization:* For a given value of α, (5.11) has a unique solution due to the second term being quadratic. This constrained optimization problem is then mapped onto an unconstrained optimization problem for estimating a vector c.

   Let f be the vectorized version of F and m be a vectorized version of $\widetilde{M}$. Then we define the vector c from f implicitly by

$$f = \max(0, K'c), \quad \text{where } K = (S_1 V_1') \otimes (S_2 V_2').$$

   Here, $\otimes$ denotes the Kronecker product of two matrices. This definition of c comes from the constraint that $F \geq 0$ in (5.11), which can now be reformed as the unconstrained minimization problem

$$\min \left(\frac{1}{2} c'[G(c) + \alpha I]c - c'm \right), \tag{5.12}$$

   where

$$G(c) = K \begin{bmatrix} H(K'_{1,\cdot}c) & 0 & \dots & 0 \\ 0 & H(K'_{2,\cdot}c) & \dots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & H(K'_{N_x \times N_y,\cdot}c) \end{bmatrix} K',$$

   and $H(x)$ is the Heavyside function. Also, $K_{i,\cdot}$ denotes the i^{th} row of K. The optimization problem (5.12) is solved using a simple gradient descent algorithm.

3. *Choosing α:* There are several methods for choosing the optimal value of α.

- *BRD Method:* Once an iteration of Step 2 has been completed, it is shown in [111] that a better value of α can be calculated by

$$\alpha_{opt} = \frac{\sqrt{s_1 s_2}}{\|c\|}.$$

If one iterates between Step 2 and the BRD method, the value of α converges to an optimal value. This method is very fast, however it can have convergence issues in the presence of large amounts of noise, as well as on real data [104].

- *S-Curve:* Let F_α be the value returned from Step 2 for a fixed α. The choice of α should be large enough that F_α is not being overfitted and unstable to noise, yet small enough that F_α actually matches reality. This is done by examining the "fit-error"

$$\chi(\alpha) = \|M - K_1 F_\alpha K_2'\|_F.$$

This is effectively calculating the standard deviation of the resulting reconstruction. Plotting $\chi(\alpha)$ for various values of α generates an S-curve, as in Figure 5.1. The interesting value of α occurs at the bottom "heel" of the curve (i.e., $\frac{d \log \chi(\alpha)}{d \log \alpha} \approx .1$). This is because, at α_{heel}, the fit error is no longer demonstrating overfitting as it is to the left of α_{heel}, yet is still matching the original data, unlike to the right of α_{heel}. This method is slower than the BRD method, however it is usually more stable in the presence of noise.

For the rest of this chapter, we use the S-curve method of choosing α.

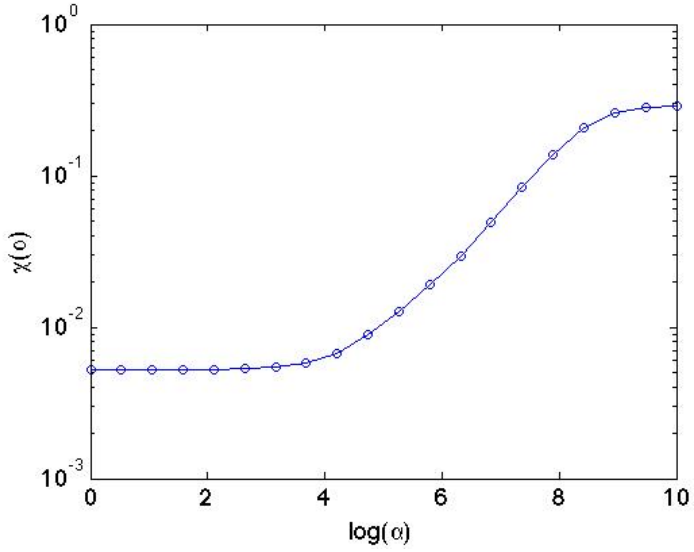


Figure 5.1: *Plot of the fit error for various* α

5.4 Numerical Considerations

Chapter 2 gives theoretical guarantees about error of estimating $\widetilde{M_0}$ with the recovered $\widetilde{M}$. We shall address several issues related to practical applications in this Section. We shall let $\widetilde{M_0}$ be the original compressed data matrix we are hoping to recover, and let $\widetilde{M}$ be the approximation obtained by solving (5.8) for the sampling operator $\mathcal{R}_\Omega$. We consider the guarantee given in (2.9) term by term.

For the rest of this chapter, we take the kernels K_1 and K_2 to be Laplace type kernels with quickly decaying singular values. For our purposes, we shall use the kernels $k_1(\tau_1, x) = 1 - e^{-\tau_1/x}$ and $k_2(\tau_2, y) = e^{-\tau_2/y}$ to represent the general data structure of most multi-exponential NMR spectroscopy measurements. The same kernels shall be used in Section 5.5 for simulations and experiments. Also, τ_1 is logarithmically sampled between 0.0005 and 4 and τ_2 is linearly sampled between

0.0002 and 0.4, as these are typical values in practice. Also for this section, F is taken to be a two peak distribution, namely Model 3 from Section 5.5.

When needed, we set $s_1 = s_2 = 20$. This choice is determined by the discrete Picard Condition (DPC) [68]. For ill-conditioned kernel problems $Kf = g$, with $\{u_i\}$ denoting left singular vectors of K and $\{\sigma_i\}$ the corresponding singular values, the DPC guarantees the best reconstruction of f is given by keeping all $\sigma_i \neq 0$ such that $\frac{|u_i^* g|}{\sigma_i}$ on average decays to zero as σ_i decrease. For our kernels with tensor product structure in (5.1), Figure 5.2 shows the relevant singular values and vectors to keep. The $s_1 = s_2 = 20$ rectangle provides a close estimate for what fits inside this curve, implying that at a minimum we could set $s_1 = s_2 = 20$ to satisfy DPC. DPC provides a stronger condition than simply keeping the largest singular values, or attempting to preserve some large percentage of the energy [67].

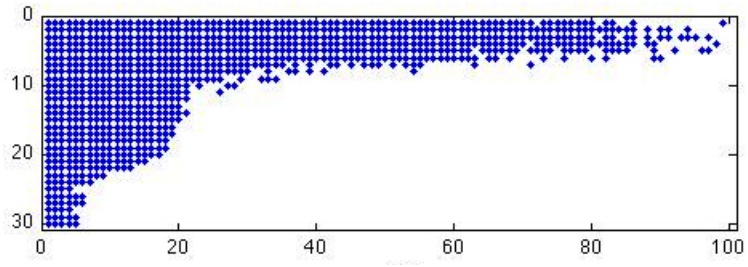


Figure 5.2: *Points denote which singular values of K_1 (rows of plot) and K_2 (columns of plot) to keep in order to satisfy the discrete Picard condition for stable inversion.*

5.4.1 Noise Bound in Practice

Theorem 2.2.3 hinges on the assumption that $\delta_{5r} < 1/10$, where δ_r is the isometry constant for rank r. This puts a constraint on the maximum size of r.

Let us denote that maximal rank by r_0. If we knew a priori that $\widetilde{M}_0$ was at most rank r_0, then this term of $\frac{\|\widetilde{M}_0 - \widetilde{M}_{0,r}\|_*}{\sqrt{r}}$ would have zero contribution, as $\widetilde{M}_0 = \widetilde{M}_{0,r}$. However, because of (5.6), $\widetilde{M}_0$ could theoretically be full rank, since S_1 and S_2 are decaying but not necessarily 0.

This problem is rectified by utilizing the knowledge that K_1 and K_2 have rapidly decaying singular values. Figure 5.3 shows just how rapidly the singular values decay, for a typical choice of kernels and discretization points. This means $\widetilde{M}_0$ from (5.6) must have even more rapidly decaying singular values, as $V_1' F V_2$ is multiplied by both S_1 and S_2. Figure 5.4 shows that the singular values of $\widetilde{M}_0$ drop to zero almost immediately for a typical compressed data matrix.

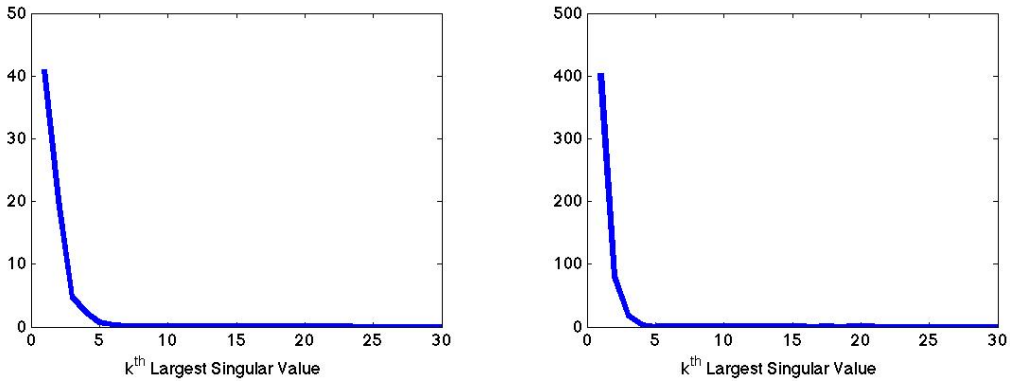


Figure 5.3: *Plots of the singular value decay of the kernels. Left: K_1, Right: K_2*

This means that even for small r_0, $\frac{\|\widetilde{M}_0 - \widetilde{M}_{0,r}\|_*}{\sqrt{r}} \leq \left\| \sum_{i=r_0+1}^{\min(s_1,s_2)} \sigma_i(\widetilde{M}_0) \right\|$ is very close to zero, as the tail singular values of $\widetilde{M}_0$ are almost exactly zero.

Figure 5.5 shows how the relative error decays for larger percentages of measurement, and how that curve matches the predicted curve of $p^{-1/2}\|e\|_2$. One can see from this curve that the rank r error does not play any significant role in the

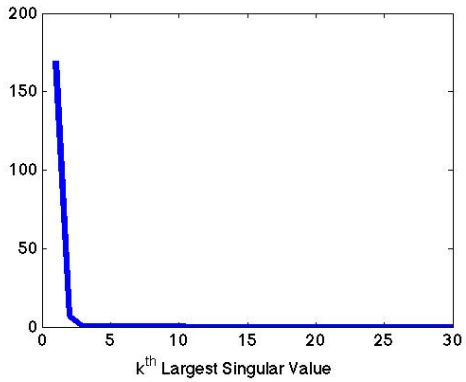


Figure 5.4: *Plot of the singular value decay for data matrix M*

reconstruction error.

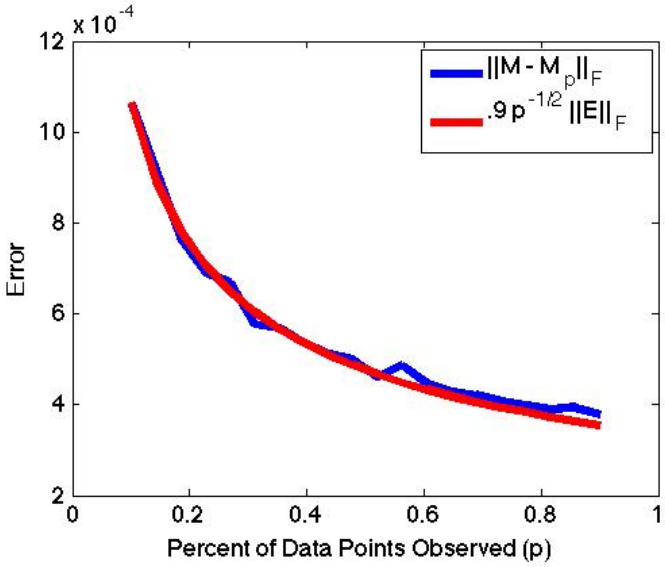


Figure 5.5: *Plot of the error in reconstruction*

5.4.2 Incoherence

The incoherence parameter μ to bound the number of measurements in (2.12) plays a vital role in determining m in practice. It determines whether the mea-

surements $\{u_i' v_j\}$ are viable for reconstruction from significantly reduced m, even though they form a Parseval tight frame.

To demonstrate that μ does not make reconstruction prohibitive, we demonstrate on a typical example of K_1 and K_2, as described at the beginning of this section.

Figure 5.6 shows the $\|\phi_j\|^2 \frac{|J|}{n}$ for each measurement $\{u_i' v_j\}$ from the above description, making $\mu = \max \|\phi_j\|^2 \frac{|J|}{n} = 89.9$. While this bound on μ is not ideal, as it makes $m > n^2$, there are two important notes to consider. First, as was mentioned in Section 2.3, Theorem 2.2.3 guarantees strong error bounds regardless of the system being underdetermined. Second, as is shown in Section 5.4.3, the estimate $\widetilde{M}$ is still significantly better than a simple least squares minimization, which in theory applies as the system isn't underdetermined.

Also note from Figure 5.6 the fact that mean($\|\phi_j\|^2 \frac{|J|}{n}$) and median($\|\phi_j\|^2 \frac{|J|}{n}$) differ greatly from max($\|\phi_j\|^2 \frac{|J|}{n}$). This implies that, while a small number of the entries are somewhat problematic and coherent with the elementary basis, the vast majority of terms are perfectly incoherent. This implies that Theorem 2.2.3 is a non-optimal lower bound on m. Future work shall be to examine the possibility of bounding m below with an average or median coherence, or considering a reweighted nuclear norm sampling similar to [33]. Another possibility is to examine the idea of asymptotic incoherence [1].

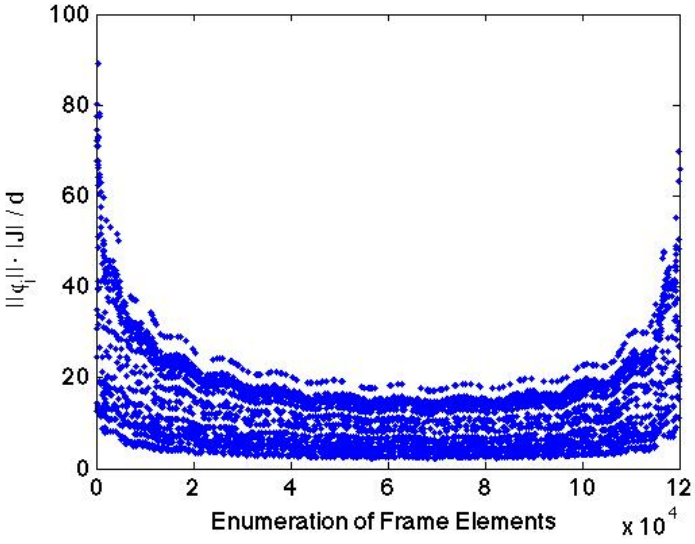


Figure 5.6: *Plot of* $\|u_i' v_j\| \frac{|J|}{n}$ *for each measurement element from the NMR problem.*

5.4.3 Least Squares Comparison

One could also attempt to solve for $\widetilde{M_0}$ using a least squares algorithm on the observed measurements via the Moore-Penrose pseudoinverse. However, as we shall show, due to noise and ill-conditioning, this is not a viable alternative to the nuclear norm minimization algorithm employed throughout this chapter. As an example, we shall again use K_1 and K_2 as described in the beginning of this section. The noise shall range over various signal-to-noise ratio.

We will consider a noisy estimate $\widetilde{M}$ of the compressed matrix $\widetilde{M_0}$, generated either through the pseudoinverse, nuclear norm minimization, or simply projecting a full set of measurements M via $U_1' M U_2$. Figure 5.7 shows the relative error of each of these recoveries, defining error to be

$$\frac{\|\widetilde{M_0} - \widetilde{M}\|_F}{\|\widetilde{M_0}\|_F}.$$

Clearly, nuclear norm minimization, even for a small fraction of measurements kept, mirrors the full measurement compression almost perfectly, as was shown in Figure 5.5. However, the least squares minimization error is drastically higher. Even at 20% measurements kept, the difference in error between least squares reconstruction and the full measurement projection error is 4 times higher than the difference between nuclear norm reconstruction and the full measurement projection error.

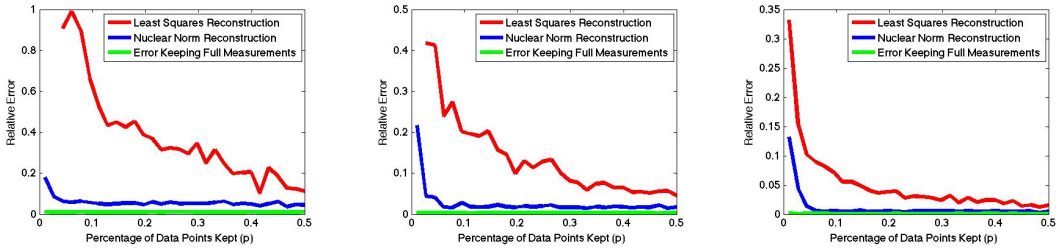


Figure 5.7: *Relative error of least squares approximation compared to nuclear norm minimization versus percentage of measurements kept. Left: SNR=15dB, Center: SNR=25dB, Right: SNR=35dB.*

5.5 Simulation Results

In simulation, we shall use the kernels $k_1(\tau_1, x) = 1 - e^{-\tau_1/x}$ and $k_2(\tau_2, y) = e^{-\tau_2/y}$ and sample τ_1 logarithmically and τ_2 linearly, as was done in Section 5.4. Our simulations revolve around inverting subsampled simulated data to recover the density function $F(x, y)$. We shall test three models of $F(x, y)$. In model 1, $F(x, y)$ is a small variance Gaussian. In model 2, $F(x, y)$ is a positively correlated density function. In model 3, $F(x, y)$ is a two peak density, one peak being a small circular Gaussian and the other being a ridge with positive correlation.

The data is generated for a model of $F(x, y)$ by discretizing F and computing

$$M = K_1 F K_2' + E,$$

where E is Gaussian noise. That data is then randomly subsampled by only keeping λ fraction of the entries.

Each true model density $F(x, y)$ is sampled logarithmically in x and y. τ_1 is logarithmically sampled $N_1 = 30$ times, and τ_2 is linearly sampled $N_2 = 4000$ times. Each model is examined for various SNR and values of λ, and α is chosen using the S-curve approach for each trial.

Let us also define the signal-to-noise ratio (SNR) for our data to be

$$\text{SNR} = 10 \log_{10} \frac{\|M\|^2}{\|E\|^2} \text{dB}.$$

Note that [111] has extensively examined Steps 2 and 3 of this algorithm, including the effects of α and SNR on the reconstruction of F. Our examination focuses on the differences between the F generated from full knowledge of the data and the F generated from subsampled data. For this reason, F_{full} refers to the correlation spectra generated from full knowledge of the data using the algorithm from [111]. F_λ refers to the correlation spectra generated from only λ fraction of the measurements using our algorithm.

5.5.1 Model 1

In this model, $F(x, y)$ is a small variance Gaussian. This is the simplest example of a correlation spectra, given that the dimensions are uncorrelated. $F(x, y)$

is centered at $(x, y) = (.1, .1)$ and have standard deviation .02. The maximum signal amplitude is normalized to 1. This model of $F(x, y)$ is a base case for any algorithm. In other words, any legitimate algorithm to invert the 2D Fredholm integral must at a minimum be successful in this case.

Figure 5.8 shows the quality of reconstruction of a simple spectra with an SNR of 30dB. Figure 5.9 shows the same spectra, but with an SNR of 15dB. Almost nothing is lost in either reconstruction, implying that both the original algorithm and our compressive sensing algorithm are very robust to noise for this simple spectra.

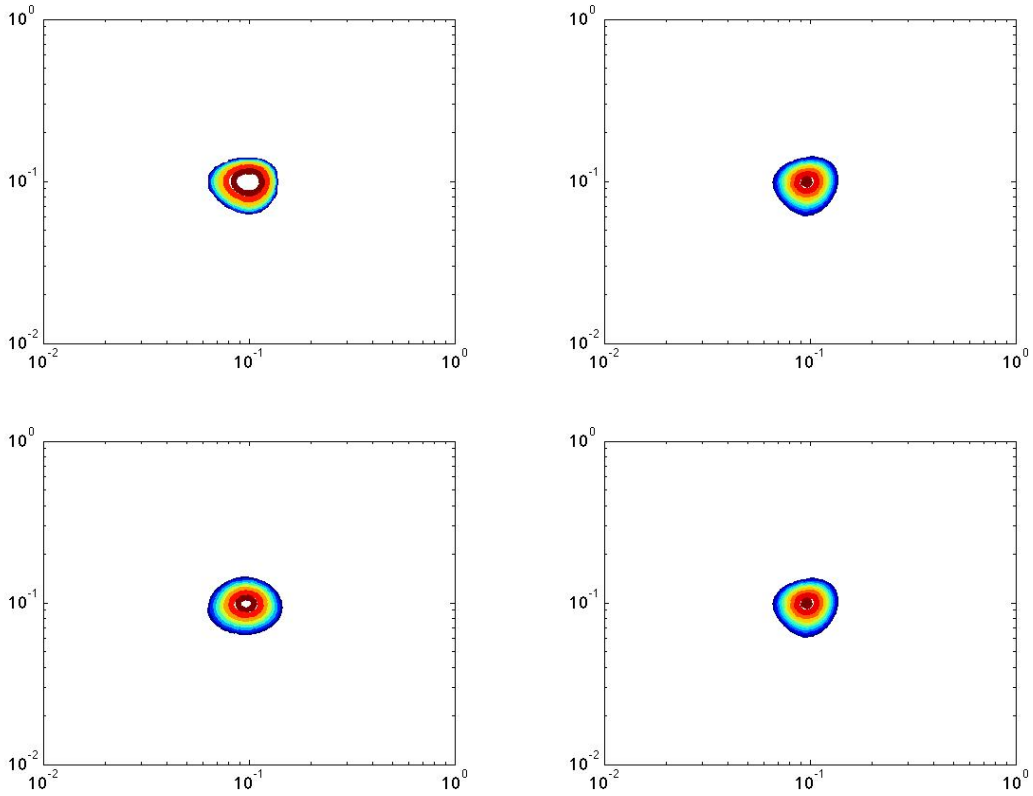


Figure 5.8: *Model 1 with SNR of 30dB. (Top-Left) True spectra, (Top-Right) F_{full}, (Bottom-Left) Reconstruction from 30% Measurements, (Bottom-Right) Reconstruction from 10% Measurements*

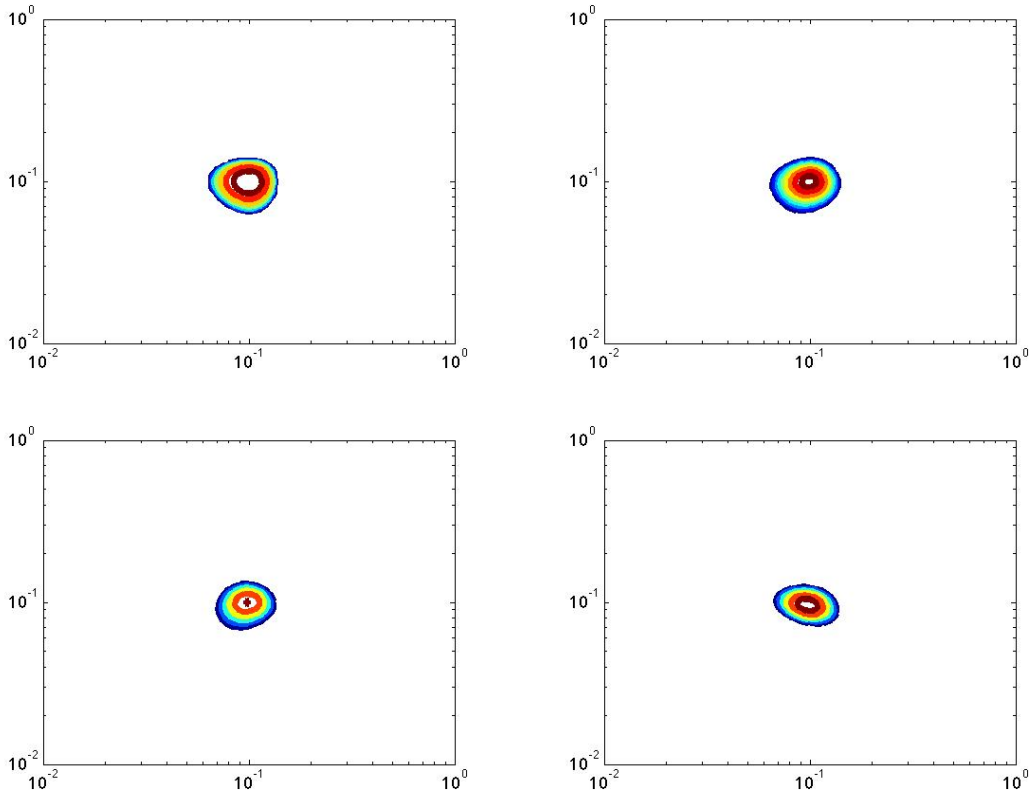


Figure 5.9: *Model 1 with SNR of 15dB. (Top-Left) True spectra, (Top-Right) F_{full}, (Bottom-Left) Reconstruction from 30% Measurements, (Bottom-Right) Reconstruction from 10% Measurements*

5.5.2 Model 2

In this model, $F(x, y)$ is a positively correlated density function. The spectra has a positive correlation, thus creating a ridge through the space. $F(x, y)$ is centered at $(x, y) = (.1, .1)$, with the variance in $x+y$ direction being 7 times greater than the variance in the $x - y$ direction. The maximum signal amplitude is normalized to 1. This is an example of a spectra where it is essential to consider the two dimensional image. A projection onto one dimensions would yield an incomplete understanding

of the spectra, as neither projection would convey that the ridge is very thin. This is a more practical test of our inversion algorithm.

Figure 5.10 shows the quality of reconstruction of a correlated spectra with an SNR of 30dB. Figure 5.11 shows the same spectra, but with an SNR of 20dB. There is slight degradation in the 10% reconstruction, but the reconstructed spectra is still incredibly close to F_{full}. Overall, both of these figures show the quality of our compressive sensing reconstruction relative to using the full data.

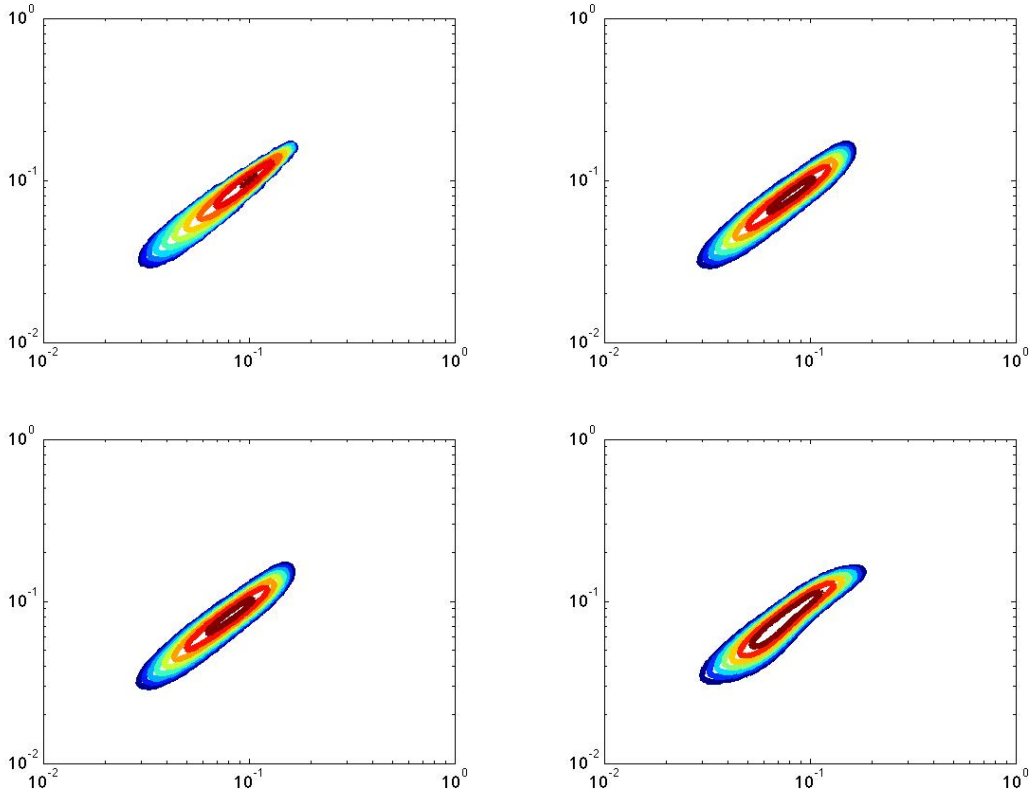


Figure 5.10: *Model 2 with SNR of 30dB. (Top-Left) True spectra, (Top-Right) F_{full}, (Bottom-Left) Reconstruction from 30% Measurements, (Bottom-Right) Reconstruction from 10% Measurements*

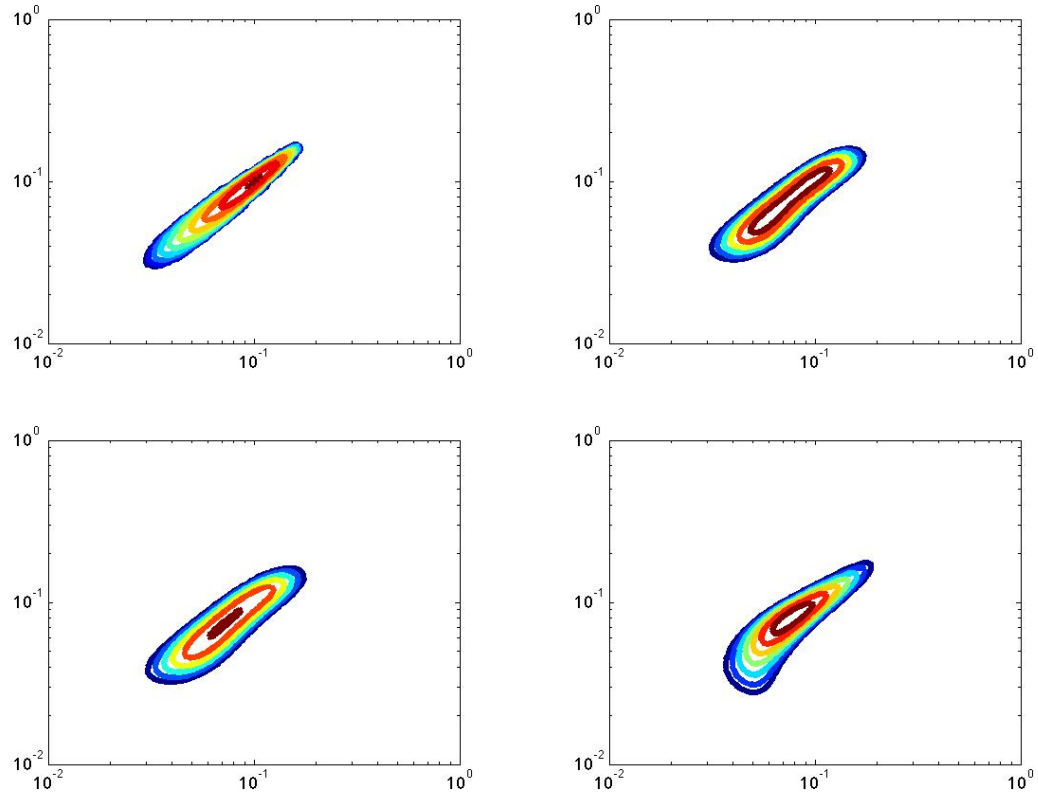


Figure 5.11: *Model 2 with SNR of 20dB. (Top-Left) True spectra, (Top-Right)* F_{full}*, (Bottom-Left) Reconstruction from 30% Measurements, (Bottom-Right) Reconstruction from 10% Measurements*

5.5.3 Model 3

In this model, $F(x, y)$ is a two peak density, with one peak being a small circular Gaussian and the other being a ridge with positive correlation. The ridge is centered at $(x, y) = (.1, .1)$, with the variance in $x + y$ direction being 7 times greater than the variance in the $x - y$ direction. The circular part is centered at $(x, y) = (.05, .4)$. The maximum signal amplitude is normalized to 1. This is an example of a common, complicated spectra that occurs during experimentation.

Figure 5.12 shows the quality of reconstruction of a two peak spectra with an SNR of 35dB. In this instance, there is some degradation from F_{full} to any of the reconstructed data sets. Once again, there is slight degradation in the 10% model, but the compressive sensing reconstructions are still very close matches to F_{full}.

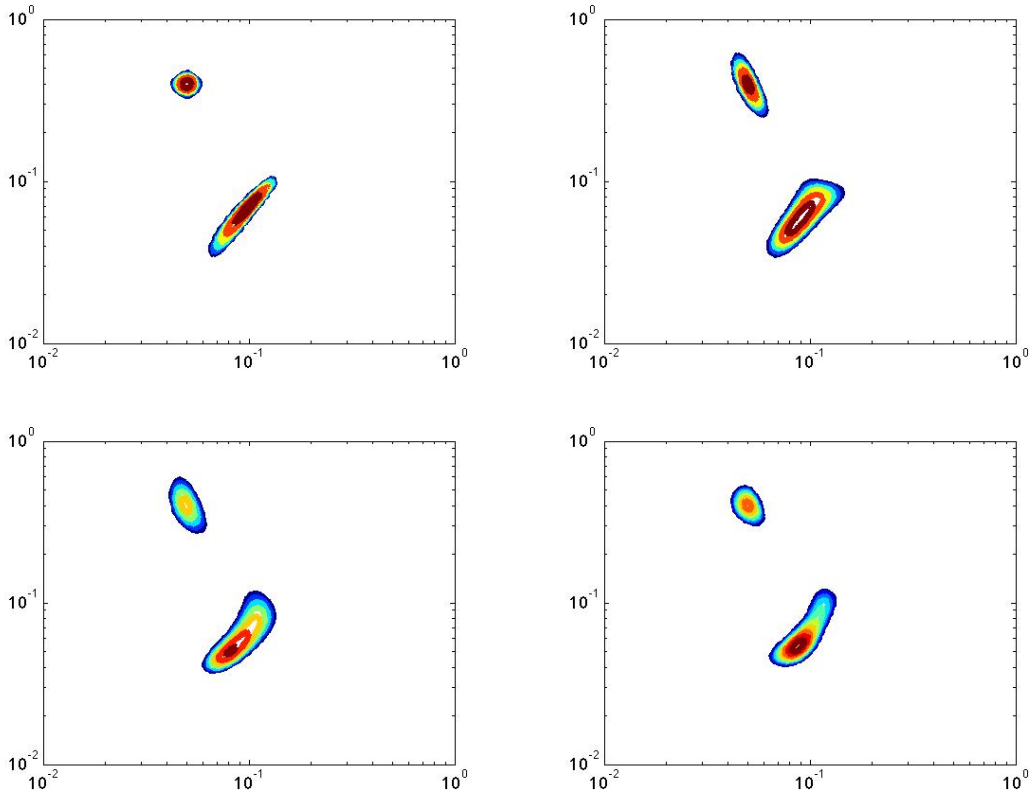


Figure 5.12: *Model 3 with SNR of 30dB. (Top-Left) True spectra, (Top-Right) F_{full}, (Bottom-Left) Reconstruction from 30% Measurements, (Bottom-Right) Reconstruction from 10% Measurements*

5.6 Conclusion

In this chapter, we introduce a matrix completion framework for solving two-dimensional Fredholm integrals. This method allows us to invert the discretized transformation via Tikhonov regularization using far fewer measurements than previous algorithms. We proved that the nuclear norm minimization reconstruction of the measurements is stable and computationally efficient, and demonstrated that resulting estimate of $\mathcal{F}(x, y)$ is consistent with using the full set of measurements. This allows us in application to reduce the measurements conducted by a factor of 5 or more.

While the theoretical framework of this paper applies to 2D NMR spectroscopy, the approach is easily generalized to larger dimensional measurements. This allows for accelerated acquisition of 3D correlation maps [4] that would otherwise take days to collect. This shall be a subject of forthcoming work.

Chapter 6: Data Fusion and Reconstruction with Preimages

6.1 Data Fusion Algorithm

Constructing a novel data fusion framework has been a major research focus in recent years. When the data comes from homogeneous sensors, the so-called multi-sensor problem, a number of algorithms exist. However, less is known about fusion of heterogeneous sensors. In [35], the authors develop a diffusion maps approach to merging heterogeneous sensors in a common, fused space. However, that fusion space is the feature space $\mathbb{R}^m$, not the original image space $\mathbb{R}^d$. In other words, the fusion space has no physical meaning.

This means there are basic questions that cannot be addressed in their framework. For example, [35] considers the idea of viewing a scene with two different hyperspectral cameras. This is an interesting problem when the cameras measure at different frequencies, which means there is no immediate way to fuse these measurements. The authors address the problem of measuring changes to the scene with Camera A having taken a picture at some time and Camera B having taken a picture at some later time, and they arrive at a novel method of analyzing these changes.

But what if Camera A has some section of its pixels occluded? Is there any

way to use the information from Camera B to fill in those missing pixels? In the framework of [35], this problem cannot be considered. However, we are set to propose a novel algorithm to answer questions of this type. This algorithm is completely based off of spectral similarity; there is no spatial component to this approach, as there is in [112].

It is important to note that, while [35] creates this framework for diffusion maps, the same argument applies for Laplacian Eigenmaps when solving the normalized eigenvalue problem in (1.3). We shall demonstrate this in Section 6.2. The algorithm goes as follows:

For notation, let pixel x_B be measured by Camera B, but not by Camera A. We shall use this information to recover an estimate of x_A, the occluded pixel from Camera A.

1. Let X be the set of pixels that are common to both Camera A and Camera B. Let (X, k_α), $\alpha \in \{A, B\}$ be the set of measurements of the common scene by each camera. First, we embed each (X, k_α) with Laplacian Eigenmaps into its own feature space Γ_α with the mapping $\Phi_\alpha : X \to \Gamma_\alpha$. For notation, $m_\alpha \in \mathbb{N}$ is the feature space dimension (so $\Gamma_\alpha \subset \mathbb{R}^{m_\alpha}$), and

$$\Phi_\alpha(x) = \left(\phi_\alpha^{(i)}(x)\right)_{i=1}^{m_\alpha},$$

where $\{\phi_\alpha^{(i)}\}$ are the eigenvectors from solving (1.3).

Use the Nyström extension to generate $\widehat{\Phi_B}(x_B)$.

2. We must now embed $\Gamma_\alpha, \alpha \in \mathcal{I}$ into one common space in order to use $\widehat{\Phi_B}(x_B)$ to gain information about x_A. For this reason, we define a rotation operator

$\mathcal{O}_{B\to A} : \Gamma_B \to \Gamma_A$ by

$$\mathcal{O}_{B\to A}x = \left(\sum_{j=1}^{m_B} x_j \langle \phi_A^{(i)}, \phi_B^{(j)} \rangle\right)_{i=1}^{m_A}.$$

One thing to note is that $\{\phi_A^{(i)}\}$ form an orthogonal basis for Γ_A, so this is similar to a change of basis operation between two different spaces.

Calculate the rotation of $\widehat{\Phi_B}(x_B)$ into Γ_A with $\mathcal{O}_{B\to A}\widehat{\Phi_B}(x_B)$

3. Let $\psi = \mathcal{O}_{B\to A}\widehat{\Phi_B}(x_B)$ and use our LE pre-image algorithm to recover x_A.

6.2 Feature Space Rotation for Laplacian Eigenmaps

For this section, let (X, μ) be a measure space with $\mu(X) = 1$. We assume we have two kernels $k_A : X \times X \to [0, 1]$ and $k_B : X \times X \to [0, 1]$. For the rest of the section, if we denote an operator $(\cdot)_\alpha$, that means $\alpha \in \{A, B\}$. These definitions follow from the diffusion maps equivalent in [35].

Define $m_\alpha(x) = \int_X k_\alpha(x, z)d\mu(z)$. Then we define a similarity function

$$a_\alpha(x, y) = \frac{k_\alpha(x, y)}{\sqrt{m_\alpha(x) \cdot m_\alpha(y)}},$$

which are elements of the matrix $A_\alpha = [a_\alpha(x, y)]_{x,y\in X}$. From here, we define the normalized Laplacian $L_\alpha = I - A_\alpha$. If $L_\alpha \in L^2(X \times X, \mu \otimes \mu)$, then it has a discrete set of eigenfunctions $\{\phi_\alpha^{(i)}\}$. The LE embedding (i.e., the m smallest eigenvectors of L_α) is denoted by $\Phi_\alpha(x) = [\phi_\alpha^{(1)}(x), ..., \phi_\alpha^{(m)}(x)]$.

If there were only one kernel (i.e., $L_\alpha = L$), then we could define a metric on

the data, the LE distance, to be

$$D(x,y)^2 = \int_X \left(\sum_{i=1}^{m} \phi^{(i)}(x)\phi^{(i)}(u) - \sum_{i=1}^{m} \phi^{(j)}(y)\phi^{(j)}(u) \right)^2 d\mu(u),$$

and immediately see that $D(x,y)^2 = \|\Phi(x) - \Phi(y)\|_2^2$.

In the case of two kernels and two Laplacians, L_A and L_B, we define the LE distance similarly between a point x_A and y_B as

$$D(x_A,y_B)^2 = \int_X \left(\sum_{i=1}^{m} \phi_A^{(i)}(x)\phi_A^{(i)}(u) - \sum_{i=1}^{m} \phi_B^{(j)}(y)\phi_B^{(j)}(u) \right)^2 d\mu(u). \tag{6.1}$$

However, $D(x_A,y_B)^2 \neq \|\Phi_A(x) - \Phi_B(y)\|_2^2$. Expanding (6.1) gives

$$\begin{aligned} D(x_A,y_B)^2 = &\sum_{i=1}^{m} \phi_A^{(i)}(x)^2 + \sum_{i=1}^{m} \phi_B^{(i)}(y)^2 \\ &- 2\sum_{i,j=1}^{m} \phi_A^{(i)}(x)\phi_B^{(j)}(y) \int_X \phi_A^{(i)}(u)\phi_B^{(j)}(u)d\mu(u). \end{aligned} \tag{6.2}$$

Due to the final term in (6.2), we define the rotation operator

$$\mathcal{O}_{B\to A}x = \left(\sum_{j=1}^{m} x_j \langle \phi_A^{(i)}, \phi_B^{(j)} \rangle \right)_{i=1}^{m}, \qquad x \in \Phi_B(X).$$

Finally, we see that

$$D(x_A,y_B) = \|\Phi_A(x) - \mathcal{O}_{B\to A}\Phi_B(y)\|_2.$$

6.3 Data Reconstruction for Hyperspectral Imagery

This section shows an application of this method on the well known AVIRIS Indian Pines hyperspectral image data set [80]. Figure 6.1 shows two of the more than two hundred frequency bands collected in the Indian Pines hyperspectral data set; an image that consists of 16 labeled classes of various types of vegetation. This

image contains a number of different spectrum, ranging from grass to soybeans to roads. For our experiment, we started with bands in the region of 900nm - 1400nm, because this is the most important spectral region for vegetation [108]. We then randomly selected twenty of these bands to be the spectrum for Camera A, and another twenty bands disjoint from the first to be the spectrum for Camera B. We then occluded a random pixel from each class (because some of the classes have a small number of samples, occluding too many pixels from one class could prove problematic). Experiments were run keeping only 25 dimensions for each LE embedding, and using 25 nearest neighbors.

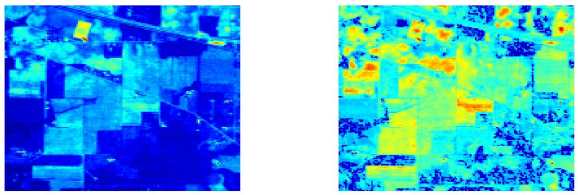


Figure 6.1: *Random Bands of Indian Pines Hyperspectral Image*

Figure 6.2 shows several of the classes that were reconstructed using our algorithm. Figure 6.3 shows those same pixels, but from Camera B. It is clear that the pixels are significantly changed by our algorithm, and that the reconstructions match very closely with the original pixels from Camera A. Table 6.1 shows the reconstruction errors for every class in the Indian Pines image. Every one of the classes is reconstructed fairly accurately, despite the fact that some of the classes have as few as 20 samples in the image.

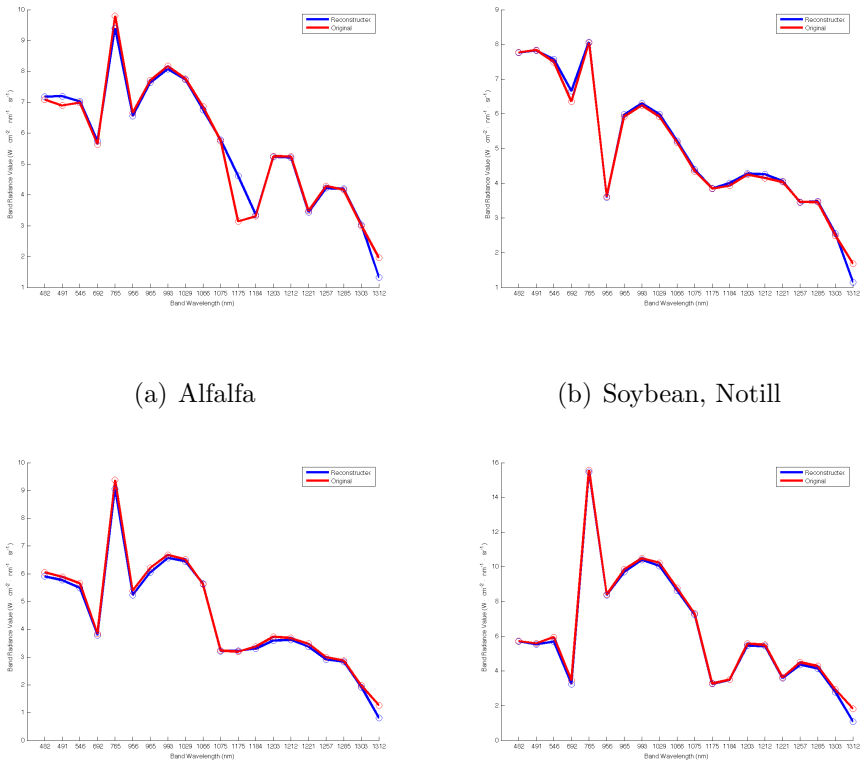
As one more example of power of this reconstruction, we ran an experiment

Class	L_1 **Reg. Rel. Error**
Alfalfa	0.0448
Corn-notill	0.0548
Corn-mintill	0.0709
Corn	0.0505
Grass-pasture	0.0596
Grass-trees	0.0544
Grass-pasture-mowed	0.0519
Hay-windrowed	0.0445
Oats	0.0590
Soybean-notill	0.0117
Soybean-mintill	0.0697
Soybean-clean	0.0194
Wheat	0.0177
Woods	0.0138
Buildings-Grass-Trees-Drives	0.0892
Stone-Steel-Towers	0.0801

Table 6.1: *Reconstruction Error for Random Pixel from Each Class. Error* $= \frac{\|\widehat{x_A} - x_A\|_2}{\|x_A\|_2}$

in which an entire block of the pixels in Camera A is occluded. Figure 6.4 shows an example from one of the bands of Camera A (it is a zoomed in on the bottom left corner of the image). If we let $\mathcal{I}$ be the set of all occluded pixels x, then our relative error is

$$\frac{1}{|\mathcal{I}|} \sum_{x \in \mathcal{I}} \frac{\|\widehat{x_A} - x_A\|_2}{\|x_A\|_2} = 0.0885.$$



(a) Alfalfa

(b) Soybean, Notill

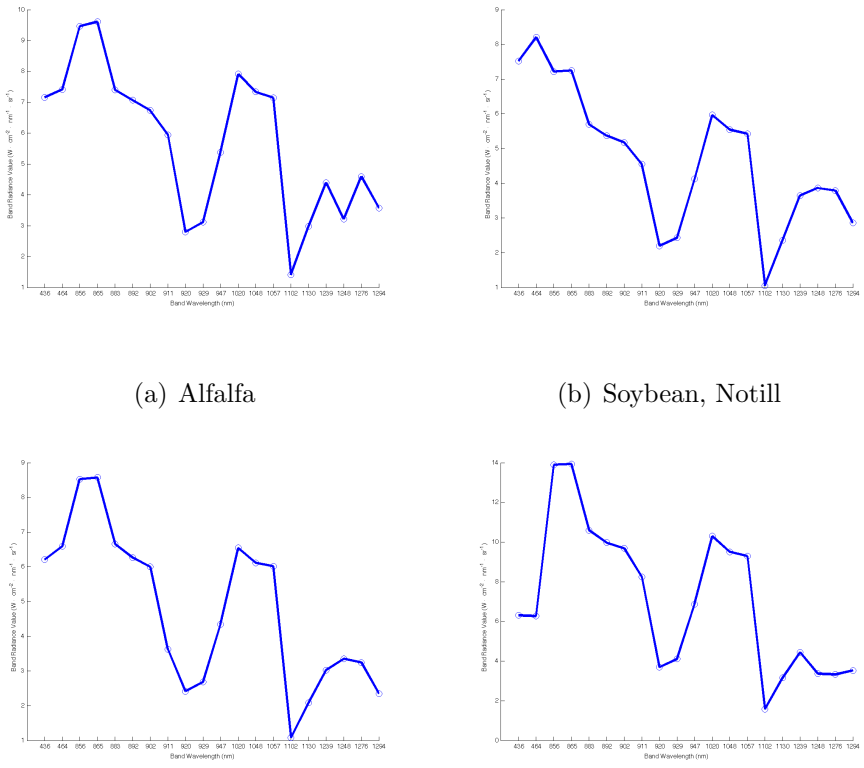
(c) Wheat

(d) Woods

Figure 6.2: *Reconstructed Pixels of Camera A from Four Classes of Indian Pines HSI*

6.4 LIDAR Reconstruction from HSI Measurements

This section deals with the data fusion algorithm of Section 6.1 for HS and LIDAR imagery, which are drastically heterogeneous modalities. LIDAR is a technology that measures height of objects below, which can be vital information in urban environments or forest canopy surveying. We shall run two experiments of this type: one on an artificial HSI LIDAR dataset, and a second on the MUUFL Gulfport Campus dataset [58]. In both cases, the HS and LIDAR images have pixel registration.



(a) Alfalfa

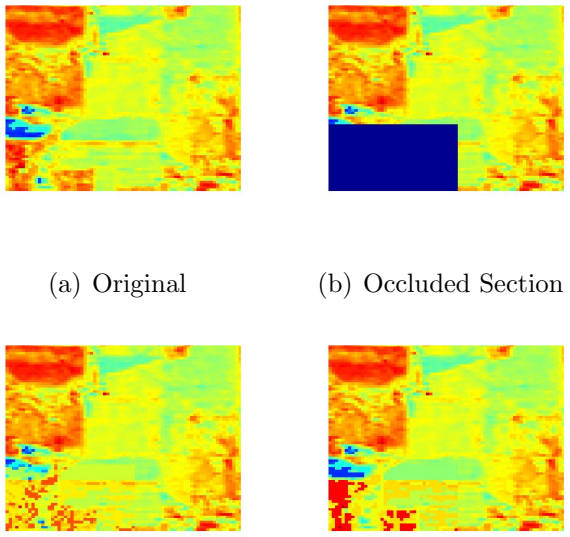
(b) Soybean, Notill

(c) Wheat

(d) Woods

Figure 6.3: *Same Pixels from Figure 6.2 from Camera B*

It is important to note that, unlike cameras with different HSI bands, LIDAR and HSI contain asymmetric information. Namely, if there are multiple classes of materials at the same elevation (e.g. roads and grass), LIDAR has no way of distinguishing between the classes. However, there is clear separation between these classes for HSI. This implies that certain types of fusion and reconstruction are unattainable. For this reason, we shall consider using HSI (camera B) to reconstruct missing information from LIDAR (camera A).



(a) Original (b) Occluded Section

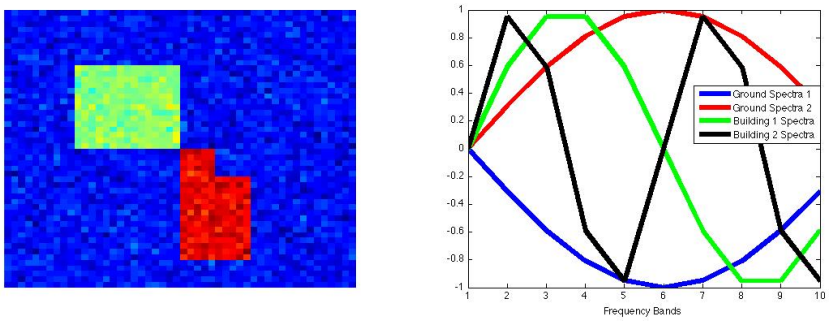
(c) Interpolation from [78] (d) L_1 Regularized Pre-image

Figure 6.4: *Bottom Left Corner of 966nm Wavelength Band of Camera A*

6.4.1 Artificial HSI LIDAR Dataset

The artificial dataset consists of ground and two buildings of differing heights (heights 0, 1, and 2). Each building is given a distinct HSI signature, whereas the ground contains multiple signatures. See Figure 6.5 for details. Each HSI pixel is corrupted by high frequency, low amplitude sinusoidal noise. The LIDAR heights are corrupted by white Gaussian noise with standard deviation 10% the height of the shorter building.

For the LE embeddings, 250 dimensions were kept for HSI and 50 dimensions for LIDAR. This number of dimensions is necessary for the feature space rotation. This is evidenced by Figure 6.6. The top row shows the HSI spectra of each data



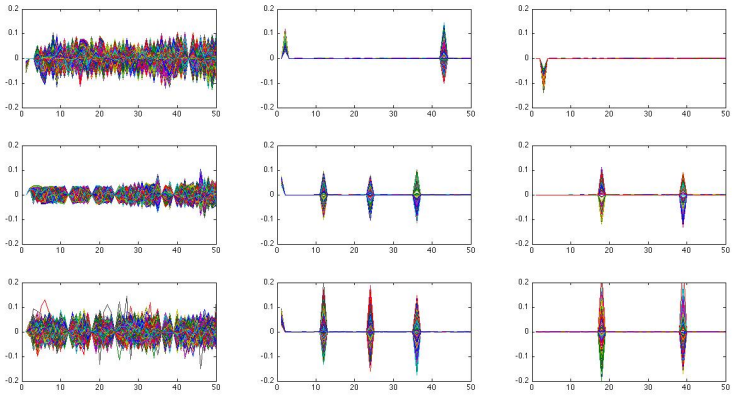
(a) LIDAR (b) Pure HSI Spectra

Figure 6.5: *LIDAR and Pure HSI Bands for Artificial Data Fusion Experiment*

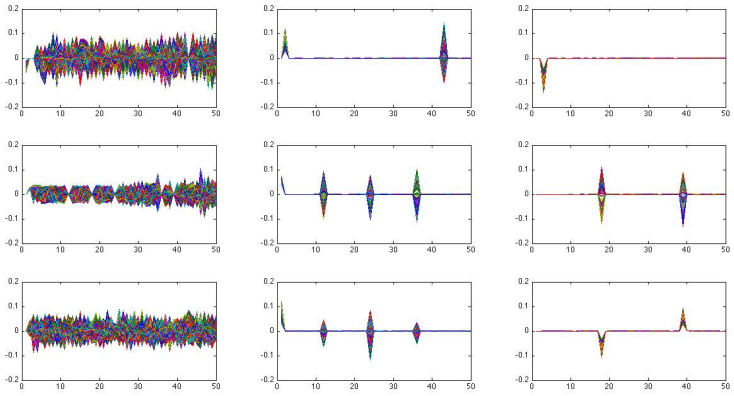
point, the middle row shows the LIDAR spectra of each data point, and the bottom row shows the HSI spectra after having been rotated into the LIDAR feature space. The left column is spectra corresponding to the ground class, the middle column is spectra corresponding to the building 1 class, and the right column is the spectra corresponding to the building 2 class.

In an ideal setting, the middle and bottom columns should be identical, implying the rotation of HSI spectra worked perfectly. For 250 HSI dimensions, these images match fairly well. However, as the number of HSI dimensions decrease, these images begin to diverge. These differences make inpainting via pre-imaging algorithms difficult, as the input is flawed.

Figure 6.7 shows the quality of reconstruction of the missing LIDAR pixels. Note that the reconstructed pixels are actually of a higher SNR than the original measurements, implying that our algorithm has actually denoised these pixels. In fact, if we let $\mathcal{I}$ be the set of all occluded pixels x, and x_A is the true LIDAR height,



(a) Spectra with 250 HSI dimensions and 50 LIDAR dimensions

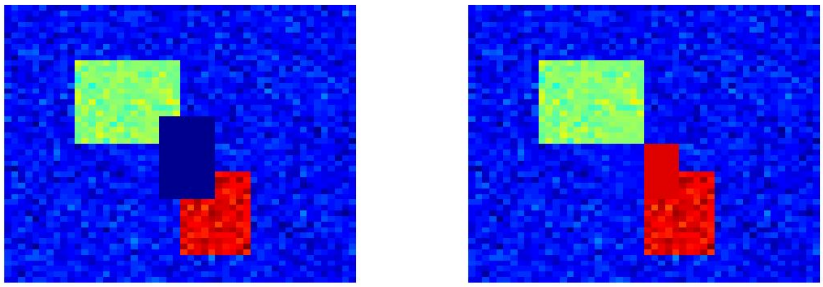


(b) Spectra with 50 HSI dimensions and 50 LIDAR dimensions

Figure 6.6: *Spectra for different classes of artificial HSI and LIDAR. Top row: HSI spectra, Middle row: LIDAR spectra, Bottom row: HSI rotated into LIDAR space; Left column: ground level class, Middle column: level 1 building class, Right column: level 2 building class*

then our relative error is

$$\frac{1}{|\mathcal{I}|}\sum_{x\in\mathcal{I}}\|\widehat{x_A}-x_A\|_2 = 0.0373.$$

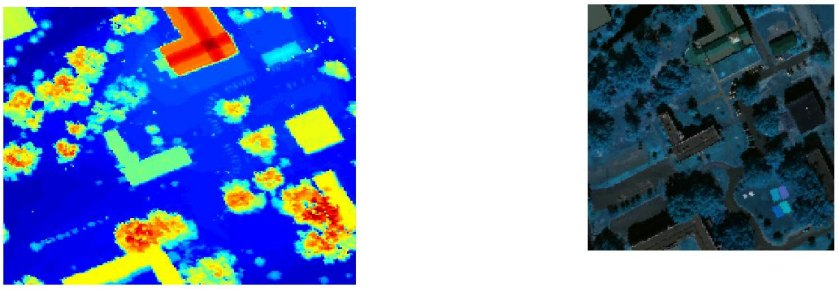


(a) Occluded LIDAR Pixels (b) Reconstructed LIDAR Pixels

Figure 6.7: *Missing LIDAR and Reconstructed LIDAR for Artificial Data Fusion Experiment*

6.4.2 MUUFL Gulfport HSI LIDAR Dataset

The MUUFL Gulfport dataset consists of coregistered LIDAR and HS images of University of Southern Mississippi Gulfport campus. See Figure 6.8 for details. Each HSI pixel contains 72 frequency bands.

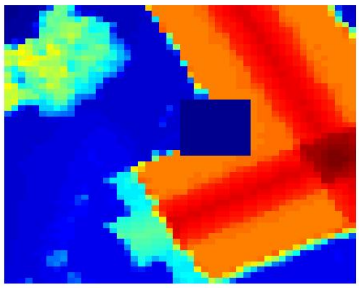


(a) LIDAR (b) Pseudocolor Image

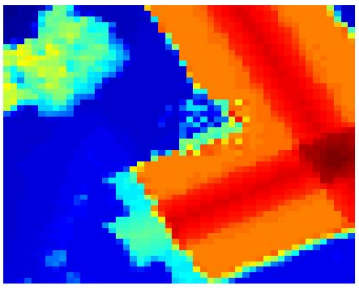
Figure 6.8: *LIDAR and Pseudocolor Image Made From Three HSI bands*

Figure 6.9 shows the quality of reconstruction of the missing LIDAR pixels.

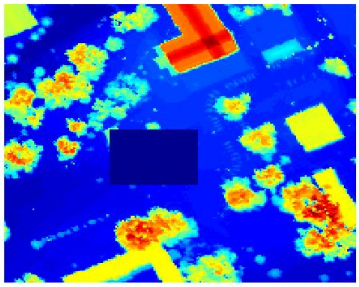
We occluded pixels in two separate sections of the image. In both cases, the reconstructions reflect the geometry of the missing buildings.



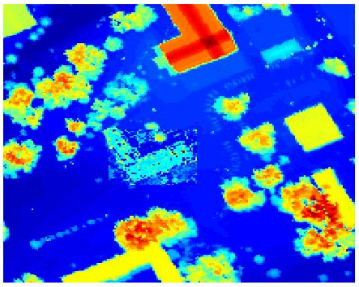
(a) Occluded LIDAR Pixels



(b) Reconstructed LIDAR Pixels



(c) Occluded LIDAR Pixels



(d) Reconstructed LIDAR Pixels

Figure 6.9: *Missing LIDAR and Reconstructed LIDAR for Gulfport Data Fusion Experiment*

6.4.3 Advantages of Pre-image Inpainting

There are two important takeaways from this LIDAR reconstruction. First, by using our method which is completely based on the spectral properties of an individual pixel, we are able to reconstruct concave sections of the image with the same precision as the convex sections. This is one advantage of our method over

spatial inpainting methods, such as total variation regularization [30] or wavelet and shearlet inpainting [31, 45].

The second take away is the viability of this approach on a large scale. Once enough training data has been collected to learn the correspondences between HSI and LIDAR pixels, one would be able to generate LIDAR images from entire areas where only HSI collection has occurred. Because there is no spatial component, the reconstructed LIDAR pixels need not be in the center of the image. It may even be geographically separated from the training data, provided the material breakdown and general elevation schemes are similar to the training data.

Figure 6.10 demonstrates the second point. The left two thirds of the image were taken as training data. The right third of the image was only measured using HSI bands, and that entire LIDAR region was reconstructed from our pre-image algorithm applied to these HSI bands, along with knowledge of the training data.

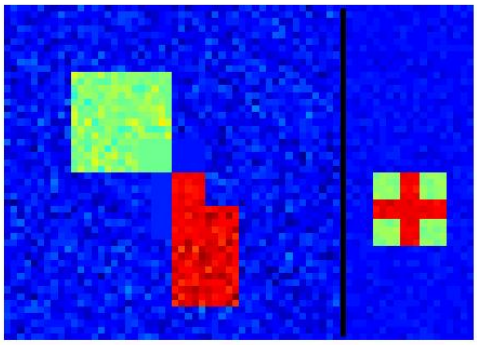


Figure 6.10: *Right Third of LIDAR Image Reconstructed Entirely from HSI Observation in that Region. Note: Black separating line added for visualization purposes*

Bibliography

[1] B. Adcock, A. Hansen, C. Poon, and B. Roman. Breaking the coherence barrier: asymptotic incoherence and asymptotic sparsity in compressed sensing. *arXiv preprint arXiv:1302.0561*, pages 1–44, 2013.

[2] A. Alfakih, A. Khandani, and H. Wolkowicz. Solving Euclidean distance matrix completion problems via semidefinite programming. *Comput. Optim. Appl.*, 12:13–30, 1998.

[3] P. Arias, G. Randall, and G. Sapiro. Connecting the out-of-sample and pre-image problems in kernel methods. *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2007.

[4] C. Arns, K. Washburn, and P. Callaghan. Multidimensional NMR Inverse Laplace Spectroscopy in Petrophysics. *Petrophysics*, 48(5):380–392, 2007.

[5] G. Bakır, A. Zien, and K. Tsuda. Learning to find graph pre-images. In *Pattern Recognition*, pages 253–261. Springer, 2004.

[6] G. H. Bakır, J. Weston, and B. Schölkopf. Learning to find pre-images. *Advances in neural information processing systems*, 16(7):449–456, 2004.

[7] R. Baraniuk, V. Cevher, M. Duarte, and C. Hegde. Model-based compressive sensing. *IEEE Trans. on Info. Theory*, 56(4):1982–2001, 2010.

[8] R. G. Baraniuk. Single-pixel imaging via compressive sampling. *IEEE Signal Processing Magazine*, 2008.

[9] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *IEEE Transactions on Neural Computation*, 2003.

[10] M. Belkin and P. Niyogi. Convergence of laplacian eigenmaps. In *NIPS*, pages 129–136, 2006.

[11] J. J. Benedetto. Irregular sampling and frames. *wavelets: A Tutorial in Theory and Applications*, 2:445–507, 1992.

[12] J. J. Benedetto and M. Fickus. Finite normalized tight frames. *Advances in Computational Mathematics*, 18(2-4):357–385, 2003.

[13] Y. Bengio, J. F. Paiement, and P. Vincent. Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering. In *Advances in Neural Information Processing Systems*, pages 177–184. MIT Press, 2003.

[14] P. Bérard, G. Besson, and S. Gallot. Embedding Riemannian manifolds by their heat kernel. *Geometric & Functional Analysis GAFA*, 4(4), 1994.

[15] S. Bonettini, R. Zanella, and L. Zanni. A scaled gradient projection method for constrained image deblurring. *Inverse Problems*, 25:1–23, 2009.

[16] J. Cai, E. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM J. Optim.*, 20:1956–1982, 2010.

[17] P. Callaghan, C. Arns, P. Galvosas, M. Hunter, Y. Qiao, and K. Washburn. Recent Fourier and Laplace perspectives for multidimensional NMR in porous media. *Magn. Reson. Imaging*, 25:441–444, 2007.

[18] E. Candès, Y. Eldar, T. Strohmer, and V. Voroninski. Phase retrieval via matrix completion. *SIAM J. Imaging Sci.*, 6:199–225, 2013.

[19] E. Candès and Y. Plan. Tight oracle bounds for low-rank matrix recovery from a minimal number of random measurements. *IEEE Trans. Inform. Theory*, 57:2342–2359, 2009.

[20] E. Candès and Y. Plan. Matrix completion with noise. *Proc. IEEE*, 98:925–936, 2010.

[21] E. Candès and B. Recht. Exact matrix completion via convex optimization. *Found. Comput. Math.*, 9:717–772, 2008.

[22] E. Candès and J. Romberg. Sparsity and incoherence in compressive sampling. *Inverse Problems*, 23:969–985, 2007.

[23] E. Candès, J. Romberg, and T. Tao. Robust Uncertainty Principles : Exact Signal Reconstruction from Highly Incomplete Frequency Information. *IEEE Trans. Inform. Theory*, 52:489 – 509, 2006.

[24] E. Candès, T. Strohmer, and V. Voroninski. Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming. *Comm. Pure Appl. Math.*, 66:1241–1274, 2012.

[25] E. Candès and T. Tao. The Power of Convex Relaxation : Near-Optimal Matrix Completion. *IEEE Trans. Inform. Theory*, 56:2053–2080, 2010.

[26] E. J. Candes. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathematique*, 346(9-10):589–592, 2008.

[27] P. G. Casazza and G. Kutyniok. *Finite frames*. Birkhäuser, 2012.

[28] P. G. Casazza, M.Fickus, J. Kovačević, M. T. Leon, and J. C. Tremain. A Physical Interpretation of Tight Frames. *in Harmonic Analysis and Applications, C. Heil, ed., Appl. Numer. Harmon. Anal.*, pages 51–76, 2006.

[29] V. Cevher, A. Sankaranarayanan, M. F. Duarte, D. Reddy, R. G. Baraniuk, and R. Chellappa. Compressive sensing for background subtraction. In *Computer Vision–ECCV 2008*, pages 155–168. Springer, 2008.

[30] T. F. Chan and J. Shen. Nontexture inpainting by curvature-driven diffusions. *Journal of Visual Communication and Image Representation*, 12(4):436–449, 2001.

[31] T. F. Chan, J. Shen, and H.-M. Zhou. Total variation wavelet inpainting. *Journal of Mathematical Imaging and Vision*, 25(1):107–125, 2006.

[32] G. Chen, G. Davis, F. Hall, Z. Li, K. Patel, and M. Stewart. An interlacing result on normalized laplacians. *SIAM J. Discret. Math.*, 18(2):353–361, Feb. 2005.

[33] Y. Chen, S. Bhojanapalli, R. Ward, and S. Sanghavi. Coherent Matrix Completion. *arXiv preprint arXiv: ...*, pages 1–26, 2013.

[34] O. Christensen. *An Introduction to Frames and Riesz Bases*. Birkh auser, Boston, MA, 2003.

[35] R. R. Coifman and M. J. Hirn. Diffusion maps for changing data. *Applied and Computational Harmonic Analysis*, 2013.

[36] R. R. Coifman and S. Lafon. Diffusion maps. *Applied and computational harmonic analysis*, 21(1):5–30, 2006.

[37] L. Comtet. *Advanced Combinatorics*. Reidel, Dordrecht, Holland, 1974.

[38] R. Corless, G. Gonnet, D. Hare, D. Jeffrey, and D. Knuth. On the Lambert W Function. *Adv. Comput. Math.*, 5:329–359, 1996.

[39] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[40] W. Czaja and M. Ehler. Schroedinger eigenmaps for the analysis of bio-medical data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013.

[41] C. Davis. The Rotation of Eigenvectors by a Perturbation. *Journal of Mathematical Analysis and Applications*, pages 159–173, 1963.

[42] C. Davis and W. Kahan. The rotation of eigenvectors by a perturbation III. *SIAM Journal on Numerical Analysis*, 7(l), 1970.

[43] S. Deoni, B. Rutt, T. Arun, C. Pierpaoli, and D. Jones. Gleaning multicomponent t1 and t2 information from steady-state imaging data. *Magn. Reson. Med.*, 60:1372–1387, 2008.

[44] S. Deoni, B. Rutt, and T. Peters. Rapid combined t1 and t2 mapping using gradient recalled acquisition in the steady state. *Magn. Reson. Med.*, 49:515–526, 2003.

[45] J. A. Dobrosotskaya and A. L. Bertozzi. A wavelet-laplace variational technique for image deconvolution and inpainting. *IEEE Transactions on Image Processing*, 17(5):657–663, 2008.

[46] M. Does, C. Beaulieu, P. Allen, and R. Snyder. Multi-component t1 relaxation and magnetisation transfer in peripheral nerve. *Magn. Reson. Imaging*, 16:1033–1041, 1998.

[47] D. Donoho. Compressed sensing. *IEEE Trans. Inform. Theory*, 52:1289 – 1306, 2006.

[48] D. Donoho and P. Stark. Uncertainty principles and signal recovery. *SIAM Journal on Applied Mathematics*, pages 906–931, 1989.

[49] C. Dsilva, R. Talmon, N. Rabin, R. Coifman, and I. Kevrekidis. Nonlinear Intrinsic Variables and State Reconstruction in Multiscale Simulations. *arXiv preprint arXiv: ...*, pages 1–27, 2013.

[50] R. J. Duffin and A. C. Schaeffer. A class of nonharmonic fourier series. *Transactions of the American Mathematical Society*, pages 341–366, 1952.

[51] K. W. Duke. *A Study of the Relationship Between Spectrum and Geometry Through Fourier Frames and Laplacian Eigenmaps*. PhD thesis, University of Maryland, 2012.

[52] I. Dumitriu and S. Pal. Sparse regular random graphs: Spectral density and eigenvectors. *The Annals of Probability*, 40(5):2197–2235, Sept. 2012.

[53] R. Edelman, P. Wielopolski, and F. Schmitt. Echo-planar mr imaging. *Radiology*, 192:600–612, 1994.

[54] A. E. English, K. P. Whittall, M. L. G. Joy, and R. M. Henkelman. Quantitative two-dimensional time correlation relaxometry. *Magnetic Resonance in Medicine*, 22(2):425–434, 1991.

[55] P. Etyngier, F. Segonne, and R. Keriven. Shape priors using manifold learning techniques. In *International Conference on Computer Vision*, pages 1–8. IEEE, 2007.

[56] M. Fazel, E. Candes, B. Recht, and P. Parrilo. Compressed sensing and robust recovery of low rank matrices. *Proc. of Asilomar Conference*, 2008.

[57] C. Fowlkes, S. Belongie, and J. Malik. Efficient spatiotemporal grouping using the Nyström method. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages I–231–I–238, 2001.

[58] P. Gader, A. Zare, R. Close, J. Aitken, and G. Tuell. MUUFL Gulfport Hyperspectral and LiDAR Airborne Data Set. Technical report, University of Florida, Gainesville, FL, 10 2013.

[59] T. Goldstein and S. Osher. The split bregman method for l1-regularized problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009.

[60] J. C. Gower. Adding a point to vector diagrams in multivariate analysis. *Biometrika*, 55(3):pp. 582–585, 1968.

[61] C. Groetsch. *The theory of Tikhonov regularization for Fredholm equations of the first kind*. Pitman, 1984.

[62] R. Grone, C. Johnson, E. Sà, and H. Wolkowicz. Positive definite completions of partial hermitian matrices. *Linear Algebra Appl.*, 58:109–124, 1984.

[63] D. Gross. Recovering low-rank matrices from few coefficients in any basis. *IEEE Trans. Inform. Theory*, 57:1548 – 1566, 2011.

[64] D. Gross, Y. Liu, S. Flammia, S. Becker, and J. Eisert. Quantum state tomography via compressed sensing. *Phys. Rev. Lett.*, 105, 2010.

[65] R. Haberman. *Applied Partial Differential Equations with Fourier Series and Boundary Value Problems*. Pearson, London, England, 2004.

[66] A. Halevy. *Extensions of laplacian eigenmaps for manifold learning*. PhD thesis, University of Maryland, 2011.

[67] P. Hansen. The discrete picard condition for discrete ill-posed problems. *BIT Numerical Mathematics*, 30(4):658–672, 1990.

[68] P. Hansen. *Rank-deficient and discrete ill-posed problems: numerical aspects of linear inversion*, volume 4. Siam, 1998.

[69] R. Hanson. A numerical method for solving Fredholm integral equations of the first kind using singular values. *SIAM J. Numer. Anal.*, 8:616–622, 1971.

[70] R. Harrison, M. Bronskill, and R. Henkelman. Magnetization transfer and t2 relaxation components in tissue. *Magn. Reson. Med.*, 33(4):490–496, 1995.

[71] R. Hashemi, W. Bradley, and C. Lisanti. *MRI the Basics*. Lippincott Williams & Wilkins, Philadelphia, 2004.

[72] P. Honeine and C. Richard. Solving the pre-image problem in kernel machines: A direct method. In *IEEE International Workshop on Machine Learning for Signal Processing*, pages 1–6. IEEE, 2009.

[73] C. Johnson. Matrix completion problems: a survey. In *Proc. Sympos. Appl. Math.*, volume 40, pages 171–198, 1990.

[74] P. Jones, M. Maggioni, and R. Schul. Universal Local Parametrizations via Heat Kernels and Eigenfunctions of the Laplacian. *Ann. Acad. Scient. Fen*, 35(203):1–45, 2010.

[75] J. Kamm and J. Nagy. Kronecker product and SVD approximations in image restoration. *Linear Algebra Appl.*, 284:177–192, 1998.

[76] L. Koralov and Y. Sinai. *Theory of Probability and Random Processes*. Springer, Berlin, Germany, 2007.

[77] J. Kovacevic, P. Dragotti, and V. Goyal. Filter Bank Frame Expansions with Erasures. *IEEE Trans. Inform. Theory*, 48:1439 – 1450, 2002.

[78] D. Kushnir, A. Haddad, and R. R. Coifman. Anisotropic diffusion on submanifolds with application to earth structure classification. *Applied and Computational Harmonic Analysis*, 2012.

[79] J. T. Kwok and I. W. Tsang. The pre-image problem in kernel methods. *IEEE Transactions on Neural Networks*, 15(6):1517–25, Nov 2004.

[80] D. Langrebe. Indiana's Indian Pines 1992 data set. Technical report, Purdue, 1992.

[81] M. Laurent. The real positive semidefinite completion problem for series-parallel graphs. *Linear Algebra Appl.*, 252:347–366, 1997.

[82] Y. LeCun and C. Cortes. The MNIST database of handwritten digits, 1998.

[83] M. Ledoux and M. Talagrand. *Probability in Banach Spaces*. Springer, Berlin, Germany, 1991.

[84] J. A. Lee and M. Verleysen. *Nonlinear Dimensionality Reduction*. Springer, 2007.

[85] X. Liu, M. Tanaka, and M. Okutomi. Noise level estimation using weak textured patches of a single noisy image. In *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pages 665–668, Sept 2012.

[86] Y. Liu. Universal low-rank matrix recovery from Pauli measurements. *Adv. Neural Inf. Process. Syst.*, 24:1638–1646, 2011.

[87] M. Lustig, D. Donoho, and J. Pauly. Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magn. Reson. Med.*, 58:1182–1195, 2007.

[88] S. Ma, D. Goldfarb, and L. Chen. Fixed point and Bregman iterative methods for matrix rank minimization. *Math. Program. Series A*, 128:321–353, 2011.

[89] J. MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. California, USA, 1967.

[90] B. McKay. The Expected Eigenvalue Distribution of a Large Regular Graph. *Linear Algebra and its Applications*, 10017:203–216, 1981.

[91] S. Mika, B. Schölkopf, A. J. Smola, K.-R. Müller, M. Scholz, and G. Rätsch. Kernel pca and de-noising in feature spaces. In *NIPS*, volume 11, pages 536–542, 1998.

[92] M. Ohliger, V. Nesme, D. Gross, and J. Eisert. Continuous-variable quantum compressed sensing. *Available on arXiv:1111.0853v3*, 2012.

[93] V. B. Patel, G. Easley, D. Healy, and R. Chellappa. Compressed synthetic aperture radar. *Selected Topics in Signal Processing, IEEE Journal of*, 4(2):244–254, Apr. 2010.

[94] Y. Pati, R. Rezaiifar, and P. S. Krishnaprasad. Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. In *Proceedings of the 27th Annual Asilomar Conference on Signals, Systems, and Computers*, pages 40–44, 1993.

[95] S. Peled, D. Cory, S. Raymond, D. Kirschner, and F. Jolesz. Water diffusion, t(2), and compartmentation in frog sciatic nerve. *Magn Reson Med.*, 42:911–918, 1999.

[96] K. Pruessmann, M. Weiger, M. Scheidegger, and P. Boesiger. Sense: sensitivity encoding for fast mri. *Magn. Reson. Med.*, 42:952–962, 1999.

[97] B. Recht. A simpler approach to matrix completion. *J. Mach. Learn. Res.*, 12:3413–3430, 2011.

[98] B. Recht, M. Fazel, and P. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Rev.*, 52:471–501, 2010.

[99] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[100] A. Ruhe. Perturbation bounds for means of eigenvalues and invariant subspaces. *BIT Numerical Mathematics*, 10:343–354, 1970.

[101] B. Schölkopf, S. Mika, C. J. C. Burges, P. Knirsch, K.-R. Müller, G. Rätsch, and A. J. Smola. Input space versus feature space in kernel-based methods. *IEEE Transactions on Neural Networks*, 10(5):1000–1017, 1999.

[102] B. Schölkopf, A. Smola, and K. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *IEEE Transactions on Neural Computation*, 10(5):1299–1319, 1998.

[103] A. Singer and R. R. Coifman. Non-linear independent component analysis with diffusion maps. *Applied and Computational Harmonic Analysis*, 25(2):226–239, 2008.

[104] Y. Song, L. Venkataramanan, M. D. Hürlimann, M. Flaum, P. Frulla, and C. Straley. T(1)–T(2) correlation spectra obtained using a fast two-dimensional Laplace inversion. *J. Magn. Reson.*, 154:261–268, 2002.

[105] G. Stewart. Error and perturbation bounds for subspaces associated with certain eigenvalue problems. *SIAM review*, 15(4):727–764, 1973.

[106] G. W. Stewart and J. G. Sun. *Matrix perturbation theory*. Academic press, 1990.

[107] R. Talmon, D. Kushnir, R. Coifman, I. Cohen, and S. Gannot. Parametrization of linear systems using diffusion kernels. *IEEE Transactions on Signal Processing*, 60(3):1159–1173, 2012.

[108] P. S. Thenkabail, E. A. Enclona, M. S. Ashton, and B. V. D. Meer. Accuracy assessments of hyperspectral waveband performance for vegetation analysis applications. *Remote Sensing of Environment*, 91, 2004.

[109] N. Thorstensen, F. Segonne, and R. Keriven. Pre-image as karcher mean using diffusion maps: Application to shape and image denoising. In *Scale Space and Variational Methods in Computer Vision*, pages 721–732. Springer, 2009.

[110] A. Travis and M. Does. Selective excitation of myelin water using inversion-recovery-based preparations. *Magn. Reson. Med.*, 54:743–747, 2005.

[111] L. Venkataramanan, Y. Song, and M. D. Hürlimann. Solving Fredholm Integrals of the First Kind With Tensor Product Structure in 2 and 2 . 5 Dimensions. *IEEE Trans. Signal Proces.*, 50:1017 – 1026, 2002.

[112] Z. Xing, M. Zhou, A. Castrodad, G. Sapiro, and L. Carin. Dictionary learning for noisy and incomplete hyperspectral images. *SIAM Journal on Imaging Sciences*, 5:33–56, 2012.

[113] S. Zhang, W. Wang, J. Ford, and F. Makedon. Learning from Incomplete Ratings Using Non-negative Matrix Factorization. *in Proc. SIAM Conf. Data Mining*, pages 549–553, 2006.

[114] W. S. Zheng, J. Lai, and P. C. Yuen. Penalized preimage learning in kernel principal component analysis. *IEEE Transactions on Neural Networks*, 21(4):551–570, 2010.