ABSTRACT

Title of dissertation: COMPUTATIONAL METHODS IN MACHINE LEARNING: TRANSPORT MODEL, HAAR WAVELET, DNA CLASSIFICATION, AND MRI

Franck Olivier Ndjakou Njeunje
Doctor of Philosophy, 2018

Dissertation directed by: Professor Wojciech Czaja
Department of Mathematics

Professor John J. Benedetto
Department of Mathematics

With the increasing amount of raw data generation produced every day, it has become pertinent to develop new techniques for data representation, analyses, and interpretation. Motivated by real-world applications, there is a trending interest in techniques such as dimensionality reduction, wavelet decomposition, and classification methods that allow for better understanding of data. This thesis details the development of a new non-linear dimension reduction technique based on transport model by advection. We provide a series of computational experiments, and practical applications in hyperspectral images to illustrate the strength of our algorithm. In wavelet decomposition, we construct a novel Haar approximation technique for functions f in the L^p-space, $0 < p < 1$, such that the approximants have support contained in the support of f. Furthermore, a classification algorithm to study tissue-specific deoxyribonucleic acids (DNA) is constructed using the support vector

machine. In magnetic resonance imaging, we provide an extension of the T_2-store-T_2 magnetic resonance relaxometry experiment used in the analysis of magnetization signal from 2 to N exchanging sites, where $N \geq 2$.

COMPUTATIONAL METHODS IN MACHINE LEARNING: TRANSPORT MODEL, HAAR WAVELET, DNA CLASSIFICATION, AND MRI.

by

Franck Olivier Ndjakou Njeunje

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
2018

Advisory Committee:
Professor Wojciech Czaja, Chair
Professor John J. Benedetto, Co-Chair
Professor Prakash Narayan
Professor Kasso Okoudjou
Professor Radu Balan

Dedication

To my parents Jean Njeunje and Sidonie Njeunje, thank you for your love and support.

Acknowledgments

First and foremost, I would like to thank my advisors Professor Wojciech Czaja and Professor John Benedetto for their invaluable support during the course this academic journey.

Wojciech has taught me to not settle until the final whistle is blown, while working with him I could always go one step forward and always make improvements. Thank you for always welcoming me into your office; your insights and direction have made my time at the university easier. Thanks to you I truly believe that I have gained great knowledge and learned new skills that would allow me to thrive in the next chapter of my life. I am grateful to have had you as my advisor.

John has always seen the big picture and seen it in a positive light. When I thought we hit a wall in our work together, he would always reassure me that all was well, and that this meant what we are doing is not easy, and would eventually lead to a better result. I have always appreciated and learned a lot from your lectures made complete by all your integrated stories. There is still so much more I have learned from your character, your preparation, your attention to detail, your time management, and most important, your humility. Even with all your years of experience, you find a way to let every one of your students shine. I am grateful to have had you as my advisor.

During the summer of 2016 I was fortunate to collaborate on a project with Dr. Ivan Ovcharenko and his research group at NCBI (National Center for Biotechnology Information) at the NIH (National Institutes of Health), and I am thankful for that

opportunity. Working with Ivan has allowed me to sharpen my skills in machine learning, I would like to thank him for the many hours spent in his office discussing and advancing the project. I would like to thank his research group, for all the insightful discussion we have had. A special thanks to Dr. Irina Hashmi for her help in mentoring me during my time at the lab and Dr. Wei Song for helping me afterwards.

During the summer of 2015 I was fortunate to collaborate on a project with Dr. Richard Spencer and his research group at NIA (National Institute on Aging) at the NIH (National Institutes of Health). I am grateful to Dr. Spencer for giving me this opportunity, and for his valuable time and input towards to completion of our project together.

I would like to also thank the rest of my committee members Professor Kasso Okoudjou, Professor Radu Balan, and my dean's representative, Professor Prakash Narayan. Thank you for your time and valuable input toward the final draft of my thesis, I know it is not an easy task but, I truly appreciate your help and support. Professor Howard Elman, thank you for all the lectures, discussions during office hours, and for reminding me of why I am here as a student.

Thank you to the Norbert Wiener Center, my office mates, and friends: Christiana Sabett, Yiran Li, Weilin Li, Matt Guay, Chae Clark, Matt Begue, Paul Koprowski, and Alex Cloninger, for all the discussions, stories, worries, laughter we shared together. Seeing you come in the office daily and putting in the work encouraged me to do the same, I am grateful for your company during all these years. A special thanks to Tempest Jamison; your support, and encouragement has been

a great help and has given me the extra push needed to bring this journey to an end. Alverda McCoy, you have been like a mother away from home, always reassuring me that before I knew it will all be done and this chapter will be behind me. Thank you for your time in the department, I am grateful that our paths have crossed. Thank you to the rest of the staff members for keeping everything running smoothly and making sure we, the students, have all the information we needed in a timely manner.

Many thanks to the AMSC program and its director, Professor Konstantina Trivisa, for accepting me into the program and giving me the opportunity to grow academically. Konstantina, I thank you for being the most friendly, happy, and welcoming director. Even before I joined the program, I could already tell it was going to be a great place to work, thanks to you.

I would also like to thank my master thesis advisors, Dr. Dinesh Ekanayake, Dr. Iraj Kalantari, Dr. Boris Petracovici, and Dr. Clifton Ealy at Western Illinois University for believing in me during my early years as a mathematician and encouraging me to further my education beyond what the school could offer.

Last but not least, I would like to thank my family - my brothers and sisters - we grew up together, look out for each other, and look up to each other. We never cease to show support and encouragement for one another in words and actions, and always challenge one another through our achievements. To my parents for being the bond that holds us together, and for your unconditional love and support, thank you.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

$\mathbb{R}$	Set of real numbers
$\mathbb{Z}$	Set of integers
$\mathbb{C}$	Set of complex numbers
$\mathbb{T}$	Torus group, identified with $[0, 1)$
$\times$	Product of two sets
$[a_{ij}]$	Matrix A with entries a_{ij}
DR	Dimension reduction
PCA	Principal components analysis
LE	Laplacian eigenmaps
DIF	Diffusion maps
ISO	Isomaps
SE	Schroedinger eigenmaps
TE	Transport eigenmaps
$\triangle$	Laplacian operator
∇	Gradient or divergence
div	Divergence
DT	Gateaux derivative of the operator T
ARI	Adjusted Rand index
OA	Overall accuracy
AA	Average accuracy
FS	F-score
κ	Cohen's Kappa coefficient
$\mathrm{tr}\,(\cdot)$	Trace of a matrice
$\mathrm{supp}\,(\cdot)$	Support of a function
$\|\cdot\|_{L^p}$	L^p norm
$\|\cdot\|$	Modulus, absolute value, or Lebesgue measure
MRA	Multi-resolution analysis
NIH	National Institutes of Health
NIA	National Institute on Aging
NCBI	National Center for Biotechnology Information
DNA	Deoxyribonucleic acid
SVM	Support vector machine
auROC	Receiver operating characteristics area under the curve
ROC	Receiver operating characteristics
AUC	Area under the curve
SVR	Singleton-vs-rest
MRI	Magnetic resonance imaging
NMR	Nuclear magnetic resonance

Chapter 1: Summary of results

The work in this thesis comes from a large array of data science topics including dimensionality reduction, wavelet decomposition, deoxyribonucleic acid (DNA) data classification, and magnetization signal analysis from magnetic resonance imaging (MRI). In Chapter 2, we developed a novel approach to non-linear dimensionality reduction as a generalization of the Laplacian eigenmaps algorithm by fusing together the spectral information with a subset of the ground truth. In Section 2.5, we provide a series of computational experiment to emphasize the differences and similarities of our algorithm, i.e., transport by advection, to similar algorithm such as Laplacian eigenmaps and Schroedinger eigenmaps. A controlled sample set experiment is done in Section 2.6 to show the strength of the transport algorithm in representing data set. In Section 2.7, we apply our algorithm to well-known hyperspectral images, and compare our results to other well-known dimension reduction algorithms.

Chapter 3 describes a wavelet decomposition based on the Haar function in the space $L^p(\mathbb{R}^d)$, where $0 < p < 1$ and $d = 1, 2$. Given a function f in $L^p(\mathbb{R}^d)$ with compact support, we were able to construct a Haar approximation of the function f from within, meaning the Haar wavelet approximants have support contained in the

support of f which was achieved in two steps. In Section 3.3, we consider functions f with symmetric support around the origin such that the endpoints are integer powers of 2, the formalism of this result is found in Theorems 3.3.3, and 3.3.15. In Section 3.4, we consider functions f with arbitrary compact supports, the formalism of this result is found in Theorems 3.4.4, and 3.4.10.

In Chapter 4, we were concerned with the design of an optimal binary classifier for deoxyribonucleic acid (DNA) data. The data set consists of two kind of DNA enhancers, singletons and multitons. Sections 4.2, 4.3, 4.4, and 4.5 describe the data set and provide insights leading the optimal classifier we chose. In Section 4.6, we apply our classifier on a sample data set from cardiac tissues, we also provide validation for the optimal characteristics of our classifier. Chapter 5 extents the analysis of the T_2-store-T_2 magnetic resonance relaxometry experiment to N exchanging sites. A solution to the system of coupled differential equations governing the relaxation of the magnetization at the sites is derived using ordinary differential equation techniques described in Section 5.3. Further results and analysis obtained from the aforementioned solution are given in Section 5.4.

Chapter 2: Transport operator on graph

2.1 Introduction

The work in this chapter is in collaboration with Professor Wojciech Czaja and Professor Pierre-Emmanuel Jabin.

Dimensionality reduction (DR) has been at the core of many data science applications for more than a century. The goal of DR is to reduce the dimension of a data set while preserving the most important information of that data set. One of the most famous DR algorithms is PCA (Principal Components Analysis) which can be traced back to the 1901 method performed by Pearson [78]. However, due to the linear nature of PCA, the method falls short in capturing the intrinsic structure of the data when a non-linear relationship governs the underlying structure within the data. Since then, the complex, non-linear, and growing amount of data have led scientists to come up with new techniques. A few well-known techniques are: kernel PCA [87], Isomap [98], (LLE) locally linear embedding [84], and (LE) Laplacian eigenmaps [5]. Today, the use of DR techniques vary based on applications from the classification of hyperspectral images [12, 14, 26, 93, 94, 102–104] to the prediction of stock market prices [105] and more [34, 41, 44].

The aforementioned non-linear DR methods lead to applications of linear op-

erators, e.g., the Laplacian. In this present study, we have developed a more general approach that constructs non-linear DR algorithms based on non-linear operators, such as appropriately chosen transport models. Recently, Gerber and Maggioni [38] sought to find the optimal transport method between two point sets based on an adaptive multiscale decomposition, which itself is derived from diffusion wavelets and diffusion maps. A different approach to the question of combining transport and efficient data representation is proposed by Benamou et al. [7,8], and by Nichols et al. [72]. In our work, we focus on the transport by advection, the active transportation of a distribution by a flow field [13, 48, 100], because of its well-studied properties as well as its partial similarity to Schroedinger Eigenmaps method of Czaja and Ehler [20]. The transport model has not been used in the literature as a tool for building a DR algorithm. Nevertheless, some related work can be found in the fields of water resource management and in bio-medical research [43], where Hansen and Shadden use DR to construct simplified transport models for cardiovascular flow.

The dynamic aspect of the transport by advection will additionally lead to applications in the field of clustering since it draws some parallels to the well-known diffusion maps algorithm [18]. Some recent work in this direction includes work by Yuan Li et al. [61], and Murphy and Maggioni [71]. At its core, our work will focus on exploring and exploiting the differences and similarities of this novel approach to the state-of-the-art dimensionality reduction and clustering algorithms found in the literature. The algorithm for our approach is given in Section 2.5. We provide an application of our algorithm for clustering and subsequent classification

of hyperspectral image data in Section 2.7.

2.2 Background

In many data science applications, it is known that high dimensional data intrinsically lie on a low dimensional manifold within the high dimensional space. Our method builds on the foundation of the Laplacian eigenmaps (LE) algorithm of Belkin and Niyogi [5] which is based on that preface. In this section, we provide a description of their algorithm for thorough explanation. Then, we review the Schroedinger eigenmaps (SE) algorithm and its connection to our method.

2.2.1 Laplacian eigenmaps

2.2.1.1 The optimal solution

Given a set of n points $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ in $\mathbb{R}^d$, the goal is to find an optimal embedding for these points in a lower m-dimensional space where $m \ll d$, while preserving local information. To achieve this goal, a weighted adjacency graph G, with weight w_{ij} and edges connecting nearby points, $\mathbf{x}_i$ and $\mathbf{x}_j$, is constructed. In case of $m = 1$, the problem is reduced to finding a map from the weighted graph to a line, so that points that are connected stay as close as possible in the mapped space. Let $\mathbf{y} = (y_1, y_2, \ldots, y_n)^T$ be such a map, whereas mathematically, this corresponds to choosing y_i to minimize the following objective

$$\sum_{i,j} (y_i - y_j)^2 w_{ij},$$

under the constraints

$$\mathbf{y}^T D\mathbf{y} = 1.$$

In the aforementioned optimization problem, the constraints serve to remove any arbitrary scaling factor in the embedding, and D is a diagonal matrix with entries $d_{ii} = \sum_i w_{ij}$. With the appropriate choice of weights w_{ij}, the objective function assigns a heavy penalty if the connected points $\mathbf{x}_i$ and $\mathbf{x}_j$ are mapped far apart. Therefore, minimizing the objective ensures that adjacent points remain close together after the mapping.

Note that for any $\mathbf{y}$, and given that the weight matrix $W = [w_{ij}]$ is symmetric, we have

$$\frac{1}{2}\sum_{i,j}(y_i - y_j)^2 w_{ij} = \mathbf{y}^T L\mathbf{y},$$

where, $L = D - W$, denoted as the graph Laplacian. The minimization problem is reduced to finding

$$\arg\min_{\mathbf{y}} \mathbf{y}^T L\mathbf{y} \quad \text{such that,} \quad \mathbf{y}^T D\mathbf{y} = 1.$$

Since L is positive semidefinite, the vector $\mathbf{y}$ that minimizes the objective function is given by the minimum eigenvalue solution to the generalized eigenvalue problem

$$L\mathbf{y} = \lambda D\mathbf{y}.$$

Note that to eliminate the trivial solution $\mathbf{1}$, with corresponding eigenvalue 0, additional constraints are added to the minimization, $\mathbf{y}^T D\mathbf{1} = 0$. Thus the optimal solution is now given by the eigenvector corresponding to the smallest non-zero eigenvalue. In general, if $m > 1$, the embedding is given by the $n \times m$ matrix

$\mathcal{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_m]$, where the i^{th} row corresponds to the embedded coordinates of the i^{th} points $\mathbf{x}_i$. In this case the objective to the minimization problem is written as

$$\sum_{i,j} \|\mathbf{y}^{(i)} - \mathbf{y}^{(j)}\|^2 w_{ij} = \operatorname{tr}(\mathcal{Y}^T L \mathcal{Y}), \tag{2.1}$$

where $\mathbf{y}^{(i)} = [\mathbf{y}_1(i), \ldots, \mathbf{y}_m(i)]^T$ is the m-dimensional representation of the i^{th} point $\mathbf{x}_i$. This reduce to finding

$$\arg\min_{\mathcal{Y}} \operatorname{tr}(\mathcal{Y}^T L \mathcal{Y}) \quad \text{such that,} \quad \mathcal{Y}^T D \mathcal{Y} = \mathbf{I}.$$

Thus the optimal solution is given by the eigenvectors corresponding to the m smallest non-zero eigenvalues to the generalized eigenvalue problem.

2.2.1.2 The adjacency graph

Given a set of n points $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ in $\mathbb{R}^d$, the adjacency graph is constructed by putting an edge between two points given that they are close enough in proximity. Proximity is defined in two ways, both based on the Euclidian distance. The first option is to connect two points if the Euclidian distance between them is less than a pre-defined fixed parameter ϵ. This method is known as the ϵ-neighborhoods, it is geometrically motivated and has the advantage of a naturally symmetric relationship. Nevertheless, this method often leads to disconnected graphs. An alternative method known as the k-nearest neighbors, constructs the graph by putting an edge between each point $\mathbf{x}_i$ and its k nearest neighbors according to the Euclidian distance. Despite the less geometrically intuitive aspect of the

k-nearest neighbors, it is a simpler method and allows us to have more control over the degree of connectivity within our graph. Moreover, it tends to lead to connected graphs, for these reasons, we will be adopting this method in our work.

2.2.1.3 The heat kernel and the weight matrix

After the construction of the adjacency graph, G, the weight matrix is constructed by adding weights on the connecting edges in G. A popular choice for the weights is based on the heat kernel, if two points $\mathbf{x}_i$ and $\mathbf{x}_j$ are connected,

$$w_{ij} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right);$$

otherwise, $w_{ij} = 0$. The parameter, σ, allows for control over the spread of information in the graph. A simpler alternative to the heat kernel consists of setting $w_{ij} = 1$ if the points $\mathbf{x}_i$ and $\mathbf{x}_j$ are connected, and setting $w_{ij} = 0$ if they are not. We note, however, that this is an extreme case of the heat kernel when the parameter $\sigma = \infty$, this simpler choice essentially gives equal importance to the all the connected neighbors. With the assumption that the points in X lie on a low dimensional manifold within the high dimensional space, Belkin and Niyogi justify the use of the heat kernel as a valid choice for assigning the weights by showing the relation between the graph Laplacian and the Laplace Beltrami operator on manifold, and later demonstrating that the Laplace Beltrami operator on differentiable functions on a manifold is closely related to the heat flow. For a complete discussion on the connection between the heat flow and the Laplace Beltrami operator on manifolds, refer to the original paper [5].

2.2.1.4 Algorithm

Given a set of n points $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ in $\mathbb{R}^d$, under the assumption that the points lie on a m-dimensional manifold where $m \ll d$, the goal is to find a set of n points $Y = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n\}$ in $\mathbb{R}^m$ where $\mathbf{y}_i$ represents $\mathbf{x}_i$ for all i from 1 to n. The LE algorithm we will be using in our work involves the following steps:

- **Step 1:** Construct the adjacency graph using the k-nearest neighbor (kNN) algorithm. This is done by putting an edge connecting nodes i and j given that $\mathbf{x}_i$ is among the k nearest neighbors of $\mathbf{x}_j$ according to the Euclidean metric.

- **Step 2:** Define a graph Laplacian, L, using the weight matrix, W. The weights in W are chosen using the heat kernel with parameter σ. If nodes i and j are connected,

$$w_{ij} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right);$$

  otherwise, $w_{ij} = 0$. The graph Laplacian is given by

$$L = D - W,$$

  where D is a diagonal matrix with entries $d_{ii} = \sum_i w_{ij}$.

- **Step 3:** Find the m-dimensional mapping by solving the generalized eigenvector problem,

$$L\mathbf{f} = \lambda D\mathbf{f}, \tag{2.2}$$

where $\mathbf{f}$ is a vector in $\mathbb{R}^n$ and λ is a real number. Let $\{\mathbf{f}_0, \mathbf{f}_1, \ldots, \mathbf{f}_{n-1}\}$ be the solution set to (2.2) written in ascending order according to their eigenvalues. The m-dimensional Euclidean space mapping is given by

$$\mathbf{x}_i \to [\mathbf{f}_1(i), \mathbf{f}_2(i), \ldots, \mathbf{f}_m(i)].$$

Due to the preservation of the local geometric structure of the manifold, this algorithm is not only useful for dimensionality reduction, but for clustering as well. One very popular example show casing the strength of the algorithm comes from the embedding of a data set laying on the "swiss roll" from 3-dimension to 2-dimension shown in Figure 2.1. The reason why this is such an important example is because well-known classical linear dimensionality reduction algorithms such as PCA (principal component analysis) or MDS (multidimensional scaling) have failed to represent the data accurately.

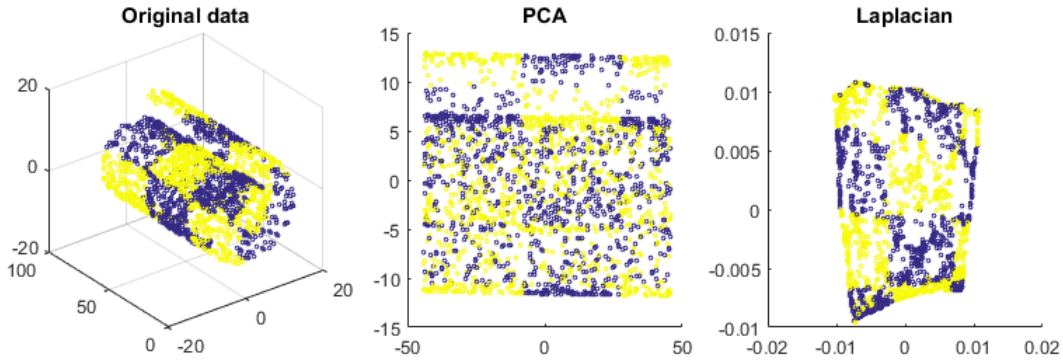

Figure 2.1: The leftmost plot represents a set of 2000 3-dimensional points sitting on a swiss roll; the middle plot represents the embedding in 2-dimension using principal component analysis (PCA); and the rightmost plot represents the same embedding using Laplacian eigenmaps (LE) with $k = 12$ (number of neighbors per node) and $\sigma = 1$.

2.2.2 Schroedinger eigenmaps

Czaja and Ehler [20] proposed the Schroedinger eigenmaps (SE) algorithm as a generalization to the LE algorithm. The goal in SE is similar to LE when no additional information is known about the ground truth of the data set. However, if partial knowledge about the data set is available, SE describes an efficient way to fuse this information into the LE algorithm to obtain better representation or more desirable results, addition work related to data fusion can be found in the following papers [11, 17, 27]. The extra information added into the LE algorithm is where the generalities comes from, in this way SE can be thought of as a semi-supervised DR algorithm. A physical motivation for SE is that partially labeled data sets lead to the notion of barrier potential which affect the diffusion processes on graph. By carefully choosing the location of the barrier potential, Czaja and Ehler are able to direct the diffusion process in order to identify the correct cluster containing the labels. There are various ways of encoding the extra information into the SE algorithm; in this study we consider barrier potentials defined as nonnegative diagonal matrix.

Similar to (2.1), the minimization problem for SE can be formulated as

$$\min_{\mathcal{Y}} \operatorname{tr}\left(\mathcal{Y}^T(L+\alpha V)\mathcal{Y}\right) \quad \text{such that,} \quad \mathcal{Y}^T D\mathcal{Y} = \mathbf{I}, \tag{2.3}$$

where V is a potential matrix and the parameter α is added emphasize the trade-off between the potential matrix and the Laplacian matrix. The minimization problem

(2.3) is equivalent to

$$\min_{\mathcal{Y}^T D \mathcal{Y} = \mathbf{I}} \frac{1}{2} \sum_{i,j} \|\mathbf{y}^{(i)} - \mathbf{y}^{(j)}\|^2 w_{ij} + \alpha \sum_i V(i) \|\mathbf{y}^{(i)}\|^2, \tag{2.4}$$

where V is the diagonal matrix with entries $V(1)$ through $V(n)$. As before, the first component of sum ensures that adjacent points remain close after the mapping. The second component of the sum add an extra level of clustering on the representation $\mathbf{y}^{(i)}$ which are associated with large value of $V(i)$. For instance, if V took the values 1 and 0 only, then the optimization problem (2.4) would yield a solution which forces increased clustering of the representations $\mathbf{y}^{(i)}$ of points associated with the value $V(i) = 1$. Note that even with the absence of extra information, we could use this potential to label points we which to represent together after the mapping.

Given a set of n points $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ in $\mathbb{R}^d$ and a function μ,

$$\mu : X \to \mathbb{R},$$

containing the extra information over the set of points X, the SE algorithm we will be using in our work involves the following steps:

- **Step 1:** Construct the adjacency graph using the k-nearest neighbor (kNN) algorithm. This is done by putting an edge connecting nodes i and j given that $\mathbf{x}_i$ is among the k nearest neighbors of $\mathbf{x}_j$ according to the Euclidean metric.

- **Step 2:** Define a graph Laplacian, L, using the weight matrix, W. The weights in W are chosen using the heat kernel with parameter σ. If nodes i

and j are connected,

$$w_{ij} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right);$$

otherwise, $w_{ij} = 0$. The graph Laplacian is given by

$$L = D - W,$$

where D is a diagonal matrix with entries $d_{ii} = \sum_i w_{ij}$.

- **Step 3:** Define the Schrodinger matrix, S, using the extra information, μ. The Schrodinger matrix is given by

$$S = L + \alpha V,$$

where α is a real number, and V is the potential matrix, a diagonal matrix given by

$$V = \begin{bmatrix} \mu_1 & & & \\ & \mu_2 & & \\ & & \ddots & \\ & & & \mu_n \end{bmatrix}, \tag{2.5}$$

where $\mu_i = \mu(\mathbf{x}_i)$ for all $i = 1, \ldots, n$.

- **Step 4:** Find the m-dimensional mapping by solving the generalized eigenvector problem,

$$S\mathbf{f} = \lambda D\mathbf{f}, \tag{2.6}$$

where $\mathbf{f}$ is a vector in $\mathbb{R}^n$ and λ is a real number. Let $\{\mathbf{f}_0, \mathbf{f}_1, \ldots, \mathbf{f}_{n-1}\}$ be the solution set to (2.6) written in ascending order according to their eigenvalues.

The m-dimensional Euclidean space mapping is given by

$$\mathbf{x}_i \to [\mathbf{f}_1(i), \mathbf{f}_2(i), \ldots, \mathbf{f}_m(i)].$$

2.2.3 Transport eigenmaps

The algorithm we are developing in this chapter, viz., transport eigenmaps (TE), has some similarities to SE in the sense that both algorithms use extra information about the data set to define a generalization of the LE algorithm. While SE uses potentials to encode to additional information, TE uses advection, the active transportation of a distribution by a flow field. In contrast to SE, TE comes from the non-linear operator which we describe in section 2.3.

2.3 The transport model

2.3.1 The continuous model

We consider a graph as a set of points $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ in $\mathbb{R}^d$, or equivalently as a set of indices i in $I = \{1, 2, \ldots, n\}$. We denote by A_i the set of adjacent indices to i (or equivalently the set of neighboring points to $\mathbf{x}_i$), we say j is adjacent to i if $j \in A_i$. We denote by $A = \{(i, j) : j \in A_i\}$ the set of edges of the graph. Let $\mathcal{P}$ denote the set of probability distributions μ from I to $\mathbb{R}^+$, μ is such that

$$\mu \in \mathcal{P} \Rightarrow \sum_i \mu_i = 1.$$

Let $\mathcal{E}$ denote the set of functions from A to $\mathbb{R}$, and $\mathcal{E}_a$ the set of functions v in $\mathcal{E}$ that are antisymmetric, that is,

$$v_{ij} = -v_{ji}.$$

Definition 2.3.1. Let $\mu \in \mathcal{P}$, and $v \in \mathcal{E}_a$ a velocity field that is itself a function of μ. We define the transport operator T acting on μ as follows:

$$T\mu = \triangle\mu - div(v\mu), \tag{2.7}$$

where $\triangle$ denotes the Laplacian defined as the divergence of the gradient acting on μ, and div denotes the divergence, a vector operator that produces a scalar field quantifying a vector field's source at each point.

The transport model we consider in equation (2.7) is also known as the transport by advection; it refers to the active transportation of a distribution by a flow field, v [8, 48]. Given an appropriately chosen flow field, we are able to direct the diffusion process in order to form desirable clusters.

2.3.2 The discrete model

In order to further study the transport model using computer tools, we propose the following discretization as well as matrix formulation.

Definition 2.3.2. Given a function μ on I, we define the gradient of μ as $\nabla\mu$ by

$$(\nabla\mu)_{ij} = w_{ij}(\mu_j - \mu_i).$$

We also define the Laplacian of μ as $\triangle\mu = div(\nabla\mu)$ by

$$(\triangle\mu)_i = \sum_{j \in A_i} w_{ij}(\mu_j - \mu_i),$$

where w_{ij} is a weight function from A to $\mathbb{R}^+$.

Note that the symbol ∇ for the gradient will be used at times to denote the divergence given no ambiguity arises; the gradient acts on scaler functions while the divergence acts on vector functions.

Definition 2.3.3. The centered discretization of $v\mu$ is given by:

$$(v\mu)^c_{ij} = v_{ij}\frac{\mu_i + \mu_j}{2}.$$

We consider a purely local type of flow by taking $v = \beta\nabla\mu$, where β is a real number. Using the central discretization, we obtain the following equation:

$$(T\mu)_i = \sum_{j\in A_i} w_{ij}(\mu_j - \mu_i) - \beta\sum_{j\in A_i} w_{ij}(\mu_j^2 - \mu_i^2), \quad \text{for each} \quad i \in I. \tag{2.8}$$

Equation (2.8) can be written in using matrices as follows:

$$T\mu = (D - W)\mu - \beta(ED_vE'^{+})\mu, \tag{2.9}$$

where W is the weight matrix with $W(\mathbf{x}_i, \mathbf{x}_j) = w_{ij}$, and D is a diagonal matrix with entries $d_{ii} = \sum_{j\in A_i} w_{ij}$. E is the incidence matrix of the corresponding graph, each column in E represents an edge on the graph; for each column, a 1 indicates that the edge is connected as outgoing edge, and a -1 indicates that the edge is connected as incoming edge. E'^{+} is the absolute value transpose of E, and D_v is a diagonal matrix with diagonal elements corresponding to the values of the vector v_{ij} for each edge (i, j) in A.

Our choice of discretization schemes in definition 2.3.2 and 2.3.3 is motivated by their well-defined analytic properties, however, it is important to note that although these discretization schemes are common in the literature, there are other

explorable schemes. In the next section, we show the congruency extent of our discrete and continuous formulations of the transport model.

2.3.3 Agreement between continuous and discrete model

We considered a couple of distribution functions, $\mu = f(x, y)$, defined on the unit circle, and $v = \nabla\mu$, i.e., $\beta = 1$, in order to visually show the agreement between the two models. The function μ is normalized so it integrates to 1, and is parametrized by $x = \cos(\theta)$ and $y = \sin(\theta)$ where $\theta \in [0, 2\pi)$. In the continuous case we have:

$$T\mu = \triangle\mu - div(\mu\nabla\mu) \quad \Rightarrow \quad \partial_\theta\mu = \partial_\theta^2\mu - \partial_\theta(\mu\partial_\theta\mu).$$

And in the discrete case we have:

$$(T\mu)_i = \sum_{j\in A_i} w_{ij}(\mu_j - \mu_i) - \sum_{j\in A_i} w_{ij}(\mu_j^2 - \mu_i^2), \quad \text{for each} \quad i \in I.$$

Example 2.3.4. In the two examples below we have three plots. From left to right, they respectively correspond to the plot of the distribution μ, the plot of the continuous model $T\mu$, and the plot of the discrete model $(T\mu)_i$ for $i \in I$, all as functions of the parameter θ.

- $f(x, y) = \frac{64(x^4y^4)}{3\pi}$

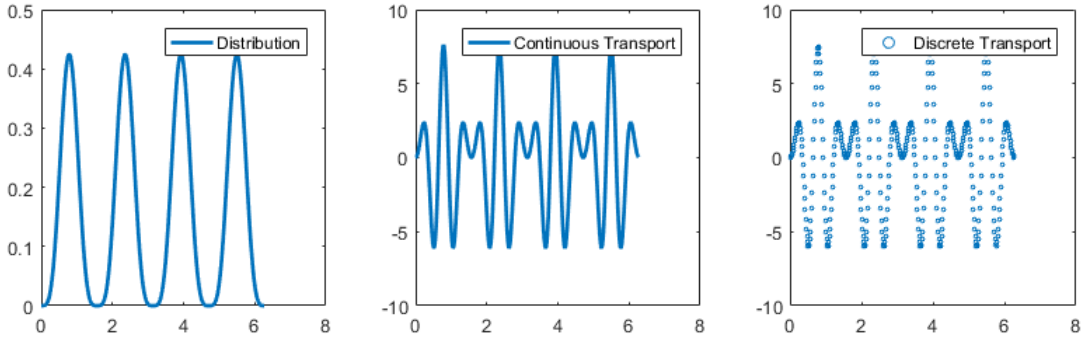

Figure 2.2: From left to right: the distribution μ, the continuous model, and the discrete model. $n = 400$ (number of data points), $k = 2$ (number of neighbors), $w_{ij} = 4000$ for $(i, j) \in A$.

- $f(x, y) = \frac{(x^3 y + 4y^2)}{2\pi}$

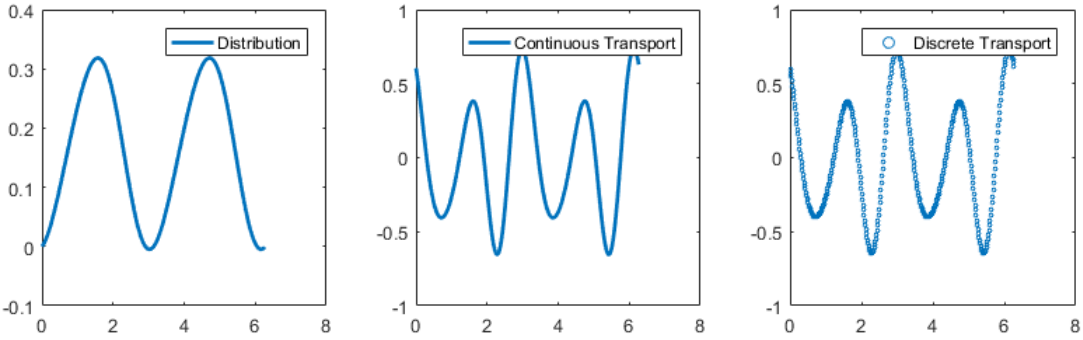

Figure 2.3: From left to right: the distribution μ, the continuous model, and the discrete model. $n = 400$ (number of data points), $k = 2$ (number of neighbors), $w_{ij} = 4000$ for $(i, j) \in A$.

2.4 Linearization of the transport model

In this section, we will linearize the transport model using its first derivative.

2.4.1 From the continuous model: Gateau Derivative

Consider two vector spaces U and V, where V is assumed to be a normed space, and denote by $[U, V]$ the set of all operators from U into V. Consider also an operator $T \in [U, V]$ with the domain $\Omega \subseteq U$; for vectors $\mu \in \Omega$ and $u \in U$, we assume that $\mu + tu \in \Omega$ for some $t \in R$. If the limit

$$DT(\mu)(u) = \lim_{t \to 0} \frac{T(\mu + tu) - T(\mu)}{t} \tag{2.10}$$

exists, then the vector $DT(\mu)(u) \in V$ is called the *Gateaux derivative* or *differential* of the operator T at the vector μ in the direction of the vector u [92]. And we say that T is Gateaux-differentiable at μ in the direction of u. If T is Gateaux-differentiable at every direction at μ, then we say that T is *Gateaux-differentiable at the vector* μ. If such is the case, then the operator $DT(\mu) : U \to V$ that assigns a vector $DT(\mu)(u) \in V$ to each vector $u \in U$ is called the *Gateaux derivative of* T *at the vector* μ. We sometimes call the operator $DT(\mu) \in [U, V]$, the *Gateaux derivative of* T. However, we shall usually not dwell too much upon the nuances of all these definitions and we just use the term "Gateaux derivative" provided no ambiguity arises. The limit in (2.10), in fact, means that for each number $\epsilon > 0$, there exists a number $\tau(\epsilon; u) > 0$ such that

$$\left\| \frac{T(\mu + tu) - T(\mu)}{t} - DT(\mu)(u) \right\| < \epsilon$$

whenever $t < \tau(\epsilon; u)$. If $\tau = \tau(\epsilon)$, then the limit is uniform, i.e., it is independent of the direction u.

2.4.1.1 Laplacian derivative

The Gateaux derivative of the Laplacian operator produces the following result.

$$\begin{aligned} D\triangle(\mu)(u) &= \lim_{t\to 0}\frac{\triangle(\mu+tu)-\triangle(\mu)}{t} \\ &= \lim_{t\to 0}\frac{\triangle(\mu)+t\triangle(u)-\triangle(\mu)}{t} \\ D\triangle(\mu)(u) &= \triangle(u). \end{aligned}$$

2.4.1.2 Divergence derivative

The Gateaux derivative of the divergence operator produces the following result.

$$\begin{aligned} Ddiv(\mu)(u) &= \lim_{t\to 0}\frac{div((\mu+tu)\nabla(\mu+tu))-div(\mu\nabla\mu)}{t} \\ &= \lim_{t\to 0}\frac{(\mu+tu)\triangle(\mu+tu)+(\nabla(\mu+tu))^2-\mu\triangle\mu-(\nabla\mu)^2}{t} \\ &= \lim_{t\to 0}\mu\triangle u+u\triangle\mu+tu\triangle u+2\nabla\mu\nabla u+t(\triangle u)^2 \\ &= \mu\triangle u+u\triangle\mu+2\nabla\mu\nabla u \\ &= \nabla u\nabla\mu+u\triangle\mu+\nabla\mu\nabla u+\mu\triangle u \\ &= \nabla(u\nabla\mu)+\nabla(\mu\nabla u) \\ Ddiv(\mu)(u) &= div(u\nabla\mu)+div(\mu\nabla u). \end{aligned}$$

2.4.1.3 Transport derivative

Given that the Gateaux derivative is linear, we obtain the following derivative for the transport model:

$$DT(\mu)(u) = \triangle(u) - \beta(div(u\nabla\mu) + div(\mu\nabla u)). \tag{2.11}$$

2.4.2 From the discrete model

For verification purpose, we will compute the derivatives over the discrete model as well. Recall,

$$\begin{aligned}(T\mu)_i &= \sum_{j\in A_i} w_{ij}(\mu_j - \mu_i) - \beta \sum_{j\in A_i} w_{ij}(\mu_j^2 - \mu_i^2), \quad \text{for each} \quad i \in I \\ &= (F_l(\mu))_i - \beta(F_d(\mu))_i, \quad \text{for each} \quad i \in I.\end{aligned}$$

Where F_l and F_d are functions of the vector μ representing the Laplace and divergence component respectively.

2.4.2.1 Laplace derivative

Let us represent by F_l' the derivative of F_l with respect to μ, we have:

$$
\begin{aligned}
F_l(\mu) &= \begin{bmatrix} \sum_{j\in A_1} w_{1j}(\mu_j - \mu_1) \\ \sum_{j\in A_2} w_{2j}(\mu_j - \mu_2) \\ \vdots \\ \sum_{j\in A_n} w_{nj}(\mu_j - \mu_n) \end{bmatrix}, \\
F_l'(\mu) &= \begin{bmatrix} -\sum_{j\in A_1} w_{1j} & w_{12} & \dots & w_{1n} \\ w_{21} & -\sum_{j\in A_2} w_{2j} & \dots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & \dots & -\sum_{j\in A_n} w_{nj} \end{bmatrix} \equiv L,
\end{aligned}
$$

where L is the Laplacian matrix. This first part agrees with the Gateaux derivative since the derivative of the Laplacian component is again the Laplacian matrix.

2.4.2.2 Divergence derivative

Let us represent by F_d' the derivative of F_d with respect to μ, we have:

$$F_d(\mu) = \begin{bmatrix} \sum_{j\in A_1} w_{1j}(\mu_j^2 - \mu_1^2) \\ \sum_{j\in A_2} w_{2j}(\mu_j^2 - \mu_2^2) \\ \vdots \\ \sum_{j\in A_n} w_{nj}(\mu_j^2 - \mu_n^2) \end{bmatrix},$$

$$F_d'(\mu) = \begin{bmatrix} -2\sum_{j\in A_1} w_{1j}\mu_1 & 2w_{12}\mu_2 & \dots & 2w_{1n}\mu_n \\ 2w_{21}\mu_1 & -2\sum_{j\in A_2} w_{2j}\mu_2 & \dots & 2w_{2n}\mu_n \\ \vdots & \vdots & \ddots & \vdots \\ 2w_{n1}\mu_1 & 2w_{n2}\mu_2 & \dots & -2\sum_{j\in A_n} w_{nj}\mu_n \end{bmatrix} = 2C_\mu \circ L,$$

where the operation $\circ$ is the Hadamard product and the matrix C_μ is given by

$$C_\mu = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ \mu_1 & \mu_2 & \dots & \mu_n \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}.$$

Let us check whether this derivative agrees with the Gateaux derivative. Let $u \in U$ as describe in the definition of the Gateaux derivative, we have

$$
\begin{aligned}
(2C_\mu \circ L)u &= \begin{bmatrix} -2\sum_{j\in A_1} w_{1j}\mu_1 & 2w_{12}\mu_2 & \dots & 2w_{1n}\mu_n \\ 2w_{21}\mu_1 & -2\sum_{j\in A_2} w_{2j}\mu_2 & \dots & 2w_{2n}\mu_n \\ \vdots & \vdots & \ddots & \vdots \\ 2w_{n1}\mu_1 & 2w_{n2}\mu_2 & \dots & -2\sum_{j\in A_n} w_{nj}\mu_n \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} \\
&= \begin{bmatrix} -2\sum_{j\in A_1} w_{1j}\mu_1 u_1 + 2w_{12}\mu_2 u_2 + \ldots + 2w_{1n}\mu_n u_3 \\ \vdots \\ 2w_{n1}\mu_1 u_1 + 2w_{n2}\mu_2 u_2 + \ldots + -2\sum_{j\in A_n} w_{nj}\mu_n u_n \end{bmatrix} \\
(2C_\mu \circ L)u &= \begin{bmatrix} 2\sum_{j\in A_1} w_{1j}(\mu_j u_j - \mu_1 u_1) \\ \vdots \\ 2\sum_{j\in A_n} w_{nj}(\mu_j u_j - \mu_n u_n) \end{bmatrix}.
\end{aligned}
$$

Therefore we have,

$$((2C_\mu \circ L)u)_i = 2\sum_{j\in A_i} w_{ij}(\mu_j u_j - \mu_i u_i).$$

Note that from the Gateaux derivative of the divergence, using central discretization, we also have

$$
\begin{aligned}
[div(u\nabla\mu) + div(\mu\nabla u)]_i &= \sum_{j\in A_i}(u_i + u_j)w_{ij}(\mu_i - \mu_j) + \sum_{j\in A_i}(\mu_i + \mu_j)w_{ij}(u_i - u_j) \\
&= \sum_{j\in A_i} w_{ij}(u_i\mu_j - u_i\mu_i + u_j\mu_j - u_j\mu_i) + \\
&\qquad w_{ij}(\mu_i u_j - \mu_i u_i + \mu_j u_j - \mu_j u_i) \\
&= 2\sum_{j\in A_i} w_{ij}(\mu_j u_j - \mu_i u_i) \\
[div(u\nabla\mu) + div(\mu\nabla u)]_i &= ((2C_\mu \circ L)u)_i.
\end{aligned}
$$

We can see that both derivative agree.

2.4.3 Transport linearization

We finally then write the linearization, $\tilde{T}$, of the transport operator around any given distribution μ as

$$\tilde{T}(u) = [L - 2\beta C_\mu \circ L](u) \quad \forall u \in U. \tag{2.12}$$

2.5 Transport eigenmaps algorithm

Given a set of n points $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ in $\mathbb{R}^d$ and a function μ

$$\mu : X \to \mathbb{R}$$

over the set of points X, under the assumption that the points lie on a m-dimensional manifold where $m \ll d$, the goal is to find a set of n points $Y = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n\}$ in $\mathbb{R}^m$ where $\mathbf{y}_i$ represents $\mathbf{x}_i$ for all i from 1 to n. The transport eigenmaps algorithm involves the following steps:

- **Step 1:** Construct the adjacency graph using the k-nearest neighbor (kNN) algorithm. This is done by putting an edge connecting nodes i and j given that $\mathbf{x}_i$ is among the k nearest neighbors of $\mathbf{x}_j$ according to the Euclidean metric.

- **Step 2:** Define a graph Laplacian, L, using the weight matrix, W. The weights in W are chosen using the heat kernel with parameter σ. If nodes i and j are connected,

$$w_{ij} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right);$$

otherwise, $w_{ij} = 0$. The graph Laplacian is given by

$$L = D - W,$$

where D is a diagonal matrix with entries $d_{ii} = \sum_i w_{ij}$.

- **Step 3:** Define the linearized transport matrix, $\tilde{T}$, using the extra information, μ. The linearized transport matrix is given by

$$\tilde{T} = L - 2\beta C_\mu \circ L,$$

where β is a real number, the operation $\circ$ is the element-wise multiplication and

$$C_\mu = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ \mu_1 & \mu_2 & \dots & \mu_n \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}, \tag{2.13}$$

where $\mu_i = \mu(\mathbf{x}_i)$ for all $i = 1, \dots, n$.

- **Step 4:** Find the m-dimensional mapping by solving the generalized eigenvector problem,

$$\tilde{T}\mathbf{f} = \lambda D\mathbf{f}, \tag{2.14}$$

where $\mathbf{f}$ is a vector in $\mathbb{R}^n$ and λ is a real number. Let $\{\mathbf{f}_0, \mathbf{f}_1, \dots, \mathbf{f}_{n-1}\}$ be the solution set to (2.14) written in ascending order according to their eigenvalues. The m-dimensional Euclidean space mapping is given by

$$\mathbf{x}_i \to [\mathbf{f}_1(i), \mathbf{f}_2(i), \dots, \mathbf{f}_m(i)].$$

2.5.1 Eigenvectors: Laplacian and transport mapping

In this section, we look at the eigenvectors of the transport operator and see how they compare to Laplacian eigenmaps using an artificially generated set of points. In Figure 2.4, we consider a data set of 300 points, organized in three clusters each of size 100. The clusters are color coded to keep track of where each cluster is being mapped; the first cluster appears in blue, the second in green and the third in yellow. Note that since the action of the Laplacian operator on the graph does not depend on the density distribution over the set of points, the mapping using Laplacian eigenmaps is the same for each density function. However since the transport operator naturally depends on the density distribution on the set over points, the mapping using the transport operator varies accordingly. We observe that the clusters on which the advection is placed, i.e., which are assigned the greater density value, are more compactly represented in the mapped space than those assigned a lower density value.

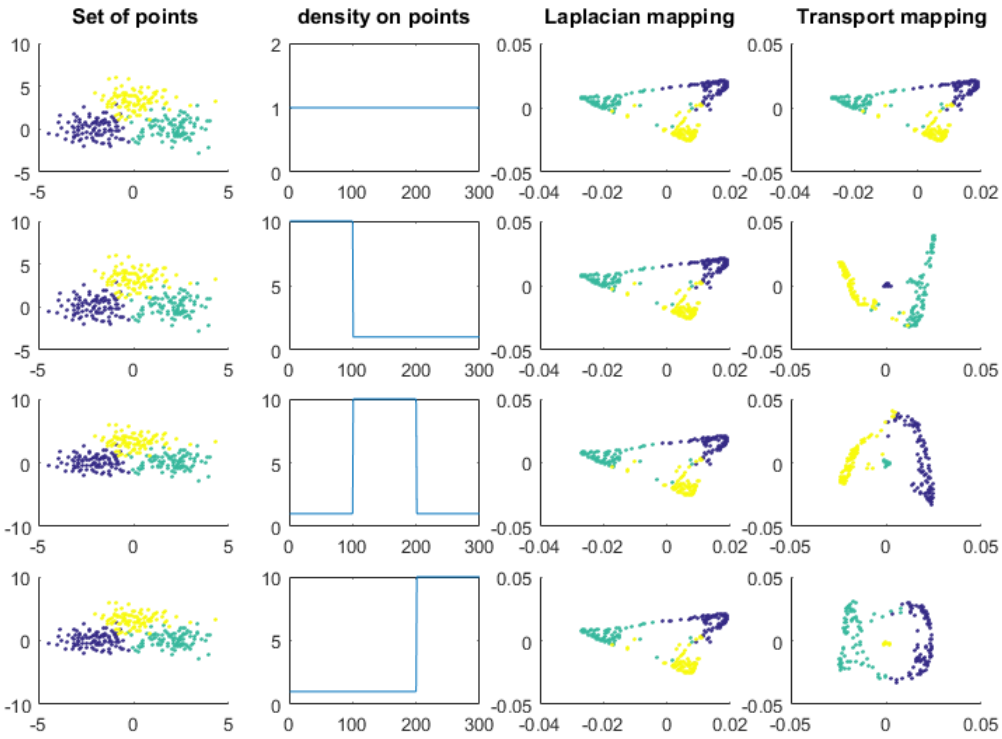

Figure 2.4: From left to right: the first column represents the data set of 300 points grouped in 3 clusters of 100 points each. The second column represents the density distribution $\beta\mu$; first, with $\beta = 1$, then the next three, with $\beta = 10$, on the later cases the advection is placed on the first (blue), second (green), and third (yellow) cluster respectively. The third column represents the mapping of the data set using the eigenvectors of the Laplacian operator on the graph. The fourth column represents the mapping of the data set using the eigenvectors of the transport operator on the graph.

2.5.2 Eigenvectors: Laplacian, transport, and Schroedinger mapping

For the purpose of explaining the results that follow, we use a metaphor in the example below (see Figure 2.5). Let us assume that the clusters, C_1 in blue, C_2 in green, and C_3 in yellow, represent continents and the points represent cities within

the continents.

In the case of the transport operator, the advection is placed on C_1, i.e., with $\beta = 10$, the function $\beta\mu$ takes the value $\beta = 10$ over the first cluster, C_1, and takes the value 1 over the other clusters, C_2 and C_3. In this case, the (inward) edges connecting any point (or city) in X to any point (or city) in C_1 are relatively strengthened (by a factor of $\beta = 10$ precisely). This translates to that the population over the continent C_1 is more likely to travel to cities within the continent C_1 than it is to travel to any other cities in X. This explains why in Figure 2.5 the representation given by the transport operator have cluster C_1 (in blue) completely separated from the other two and compacted within itself.

In the case of Schroedinger operator, the potential is placed on C_1, i.e., with $\alpha = 10$, the potential V is a diagonal matrix with entries $\alpha = 10$ for cluster C_1 (in our example, Figure 2.5, this corresponds to the first 100 diagonal entries) and 0 everywhere else. This translates to the population of each city within continent C_1 is more likely to stay (or travel) within their respective city than it is to travel to any other cities in X (including those within their continent). Nevertheless, the relative probability that the population of any given city within continent C_1 will travel to a different city (given that they are not staying in their current city at the next time step) remains unchanged. This explains why in Figure 2.5 the representation given by the Schroedinger operator have cluster C_1 (in blue) compacted within itself, but not completely separated from the other two clusters.

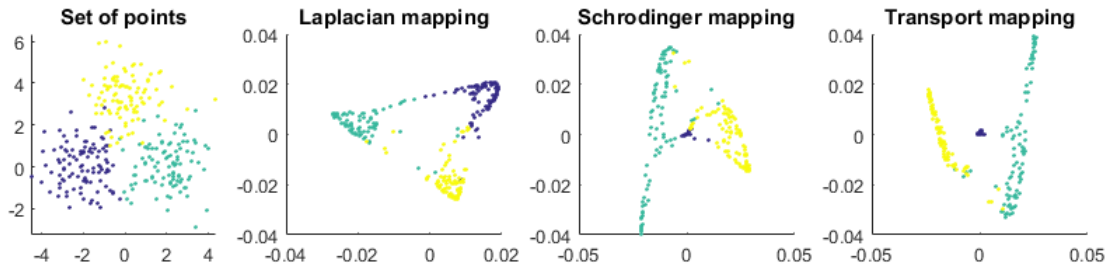

Figure 2.5: The first plot represents a set of points, 300 points grouped in 3 clusters of 100 points each, in the order blue, green, then yellow. The rests represents various mapping using the first and second eigenvectors with corresponding non-zero eigenvalue.

2.6 Controlled sample set experiments

2.6.1 The adjusted Rand index

The adjusted Rand index (ARI) is one of the most commonly used cluster validation indices and is recommended as the index of choice for measuring agreement between partitions [86]. An advantage of ARI over its predecessor, the Rand index, is its robustness against randomly generated clusters. Given a set X of n points and two partitions, e.g., clusterings, of these points, viz., $P = \{P_1, P_2, \ldots, P_r\}$ and $Q = \{Q_1, Q_2, \ldots, Q_s\}$, the adjusted Rand index is defined as

$$ARI = \frac{\sum_{ij} \binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}, \tag{2.15}$$

where $n_{ij} = |P_i \cap Q_j|$, $a_i = |P_i|$, and $b_j = |Q_j|$, for $i = 1, \ldots, r$ and $j = 1, \ldots, s$. The adjusted Rand index quantifies the similarities between two partitions while taking into account random chance assignments. A 0 indicates that the clusterings do not agree on any pair of points, and a 1 indicates that the clusterings are exactly

the same. From a mathematical standpoint, ARI is related to accuracy, but is applicable even when class labels are not used. In this light, ARI can be used as a cluster validation index for supervised and unsupervised clustering.

2.6.2 Setup

In this section, we demonstrate the strength of our algorithm using a large number of experiments. We will test the strength of our algorithm in its ability to faithfully represent the data. Each experiment consists of multiple arrangement of clusters increasing in difficulty in 2 or 3 dimension. In the 2-dimensional cases, we experiment with $3, 4, 5, 8$, and 9 clusters, while in the 3-dimensional cases we experiment with $4, 7, 8$, and 9 clusters, for a total of 9 cases. In each of the cases, we increase the difficulty by changing the parameters (position, spread or standard deviation, and added Gaussian noise) used to generate the data set, while maintaining connectivity under construction of the adjacency graph. Each individual cluster in the data set contains 100 samples. The number of neighbors used for the construction of the adjacency graph to ensure connectivity in all cases is set at $k = 50$, and we set the weight parameter to $\sigma = 1$ for simplicity. For TE, the parameter $\beta = 10$, and for SE, we introduced the parameter $\hat{\alpha}$ such that $\alpha = \hat{\alpha} \cdot \mathrm{tr}\,(L)/\mathrm{tr}\,(V)$, the SE's results are optimal for $\hat{\alpha} \geq 10^3$, we set $\hat{\alpha} = 10^3$ in all cases. The choice of our parameters is motivated by the set of experiments conducted in Section 2.7.2. The following are examples of configurations used in the experiment; in each case, the advection or potential is placed on the first cluster in blue:

- **The position:** We decrease the distance between the center of each clusters in the data set to zero, while keeping the standard deviation and added Gaussian noise constant (no added noise), see Figure 2.6.

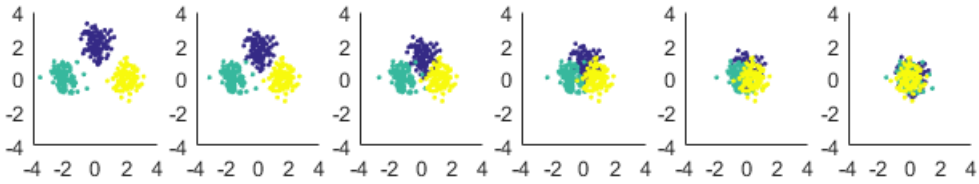

Figure 2.6: Example of data sets with various cluster's center position.

- **The spread:** We increase the standard deviation of each clusters in the data set, while keeping the position of the center of each cluster fixed and the added Gaussian noise constant (no added noise), see Figure 2.7.

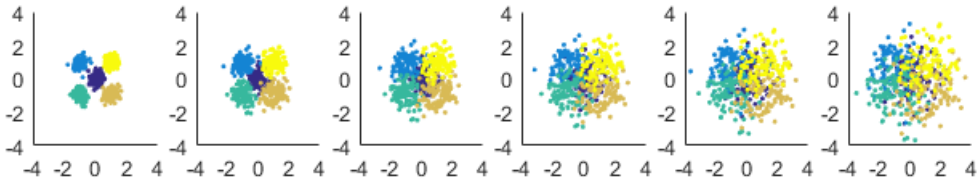

Figure 2.7: Example of data sets with various standard deviation.

- **The noise:** We increase the added Gaussian noise from none, while keeping the position of the center of each cluster fixed and the standard deviation constant, see Figure 2.8.

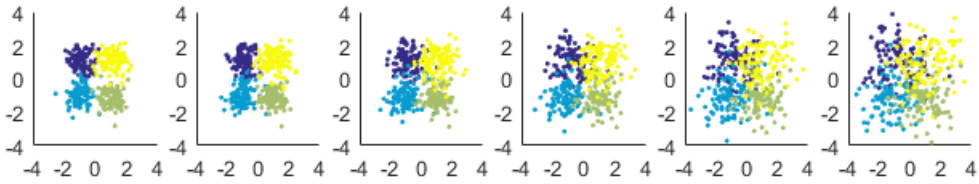

Figure 2.8: Example of data sets with various added Gaussian noise.

Given that each parameter is individually adjusted 6 times, that gives a total of 162 cases in both 2 and 3 dimensions. Since the data set for each of the 162 cases is produced at random by the software, each case is run a total of 20 times to reduce variance in the results, and the average is reported. For each individual run the following operations are performed:

- **Step 1:** Generate the data set, X, and corresponding labels.
- **Step 2:** Cluster the data set X using the k-means algorithm, and compute the agreement of the resulting clustering to the original labels using the adjusted Rand index (ARI_1).
- **Step 3:** Apply the dimension reduction algorithm on the data set X. While keeping the dimension of the output data set, Y, the same as the dimension of the data set X to ensure a fair comparison, cluster the data set Y using the k-means algorithm, and compute the agreement of the resulting clustering to the original labels using the adjusted Rand index ARI_2.
- **Step 4:** Compute and store the change in adjusted Rand index ($ARI_2 - ARI_1$).

2.6.3 Results

The following dimension reduction algorithms are used in the experiment: principal components analysis [78] (PCA), Laplacian eigenmaps [6] (LE), diffusion maps [18] (DIF), Isomap [98] (ISO), Schroedinger eigenmaps [14] (SE), transport eigenmaps (TE). Figure 2.9 shows the box plots for the change in ARI for each of the

methods used. We can see from these plots that the transport model outperforms the other methods in representing the data. On the one hand, we note that the strength of our algorithm is more evident on difficult cases, i.e.; when the number of clusters is high, the clusters are close to each other, the spread is high, or the noise level is high; see Figure 2.10. On the other hand, in simpler cases, the extra information provided by the density function, μ, does not provide a significant increase in performance; see Figure 2.11.

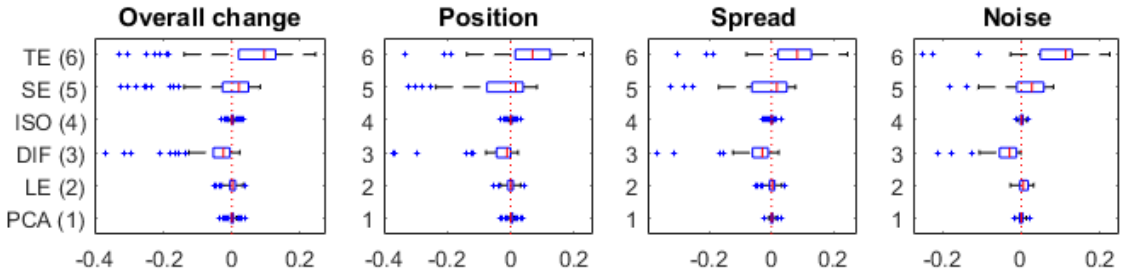

Figure 2.9: Box plot for the change of adjusted Rand index, all cases.

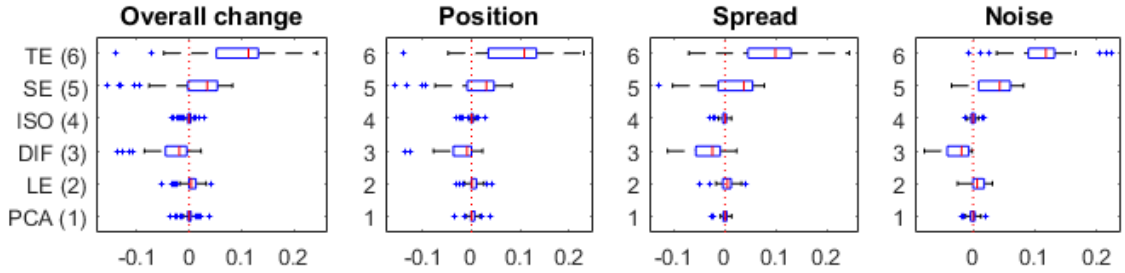

Figure 2.10: Box plot for the change of adjusted Rand index, complex cases.

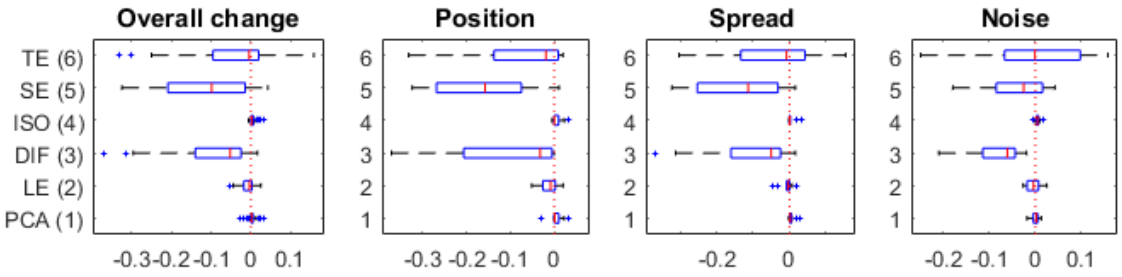

Figure 2.11: Box plot for the change of adjusted Rand index, simple cases.

2.7 Hyperspectral sample set experiments

2.7.1 Data

For our experiments, we take advantage of the hyperspectral data sets known as Indian Pines and Salinas. The Indian Pines image shown in Figure 2.12 was gathered by AVIRIS (Airborne Visible/Infrared Imaging Spectrometer) sensor over the Indian Pines test site in North-western Indiana. The Indian Pines image consists of 145×145 pixels image that contain 224 spectral bands in the wavelength range $0.4\ 10^{-6}$ to $2.5\ 10^{-6}$ meters. The ground truth available is designated into sixteen classes, see Table 2.1. The number of bands has been reduced to 200 by removing bands covering the region of water absorption: $[104-108]$, $[150-163]$, 220. Indian Pines data are available through Purdue's university MultiSpec site [1, 4].

The Salinas image shown in Figure 2.13 was gathered by AVIRIS sensor over Salinas Valley, California. The Salinas image consists of 512×217 pixels image that contain 224 spectral bands with approximately 3.7 meter high spatial resolution. The ground truth available is designated into sixteen classes, see Table 2.2. The

number of bands has been reduced to 204 by removing bands covering the region of water absorption: $[108-112]$, $[154-167]$, 224. Salinas data are publicly available online [1].

A small sub-scene of the Salinas image, denoted Salinas-A, is shown in Figure 2.14 and is usually used to speed up computational results. It comprises 86×83 pixels and includes six classes, see Table 2.3. We use a different cropped piece of the Salinas image, we denote Salinas-B, shown in Figure 2.15, leaving a $150 \times 100 \times 204$ data cube located within the same scene at [samples, lines] = [200-349, 40-139] and includes eight classes, see Table 2.4. We chose to work with this data because of the balanced spatial diversity found in the ground truth in contrast to Salinas-A, as well as the computational speed gained in contrast to the Salinas image.

We also considered grouping similar classes within the Indian Pines and Salinas-B data set to make new ground truths which we denoted Indian Pines-G and Salinas-B-G; see Table 2.5 and Table 2.6. This grouping is done because classification algorithms frequently misclassify samples of similar classes due to the similarities in their spectra information, e.g., samples in class 2–corn-notill frequently getting classified as belonging to class 3–corn-mintill and vice versa in the Indian Pines data set.

#	Class	Sample
0	Empty-space	10776
1	Alfalfa	46
2	Corn-notill	1428
3	Corn-mintill	830
4	Corn	237
5	Grass-pasture	483
6	Grass-trees	730
7	Grass-pasture-mowed	28
8	Hay-windrowed	478
9	Oats	20
10	Soybean-notill	972
11	Soybean-mintill	2455
12	Soybean-clean	593
13	Wheat	205
14	Woods	1265
15	Buildings-Grass-Trees-Drives	386
16	Stone-Steel-Towers	93

Table 2.1: Indian Pines classes.

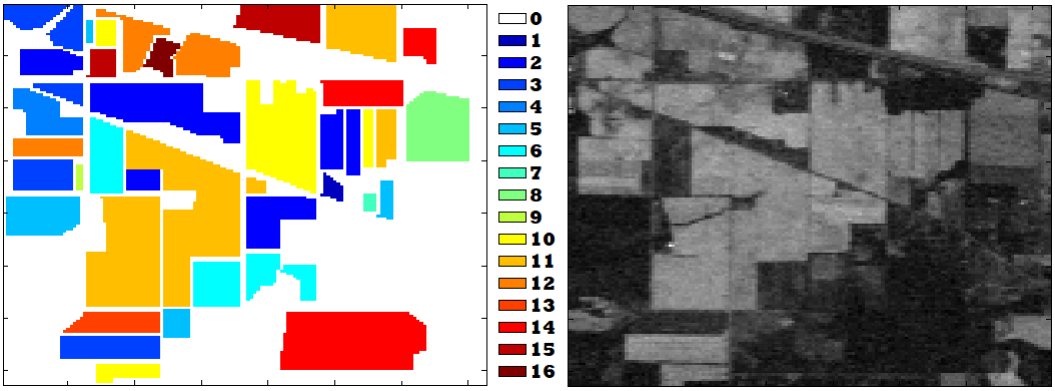

Figure 2.12: Ground truth of Indian Pines data set (left) and sample band of Indian Pines data set: 170 (right.)

#	Class	Sample
0	Empty-space	10776
1	Broccoli-green-weeds-1	2009
2	Broccoli-green-weeds-2	3726
3	Fallow	1976
4	Fallow-rough-plow	1394
5	Fallow-smooth	2678
6	Stubble	3959
7	Celery	3579
8	Grapes-untrained	11271
9	Soil-vineyard-develop	6203
10	Corn-senesced-green-weeds	3278
11	Lettuce-romaine-4wk	1068
12	Lettuce-romaine-5wk	1927
13	Lettuce-romaine-6wk	916
14	Lettuce-romaine-7wk	1070
15	Vineyard-untrained	7268
16	Vineyard-vertical-trellis	1807

Table 2.2: Salinas classes.

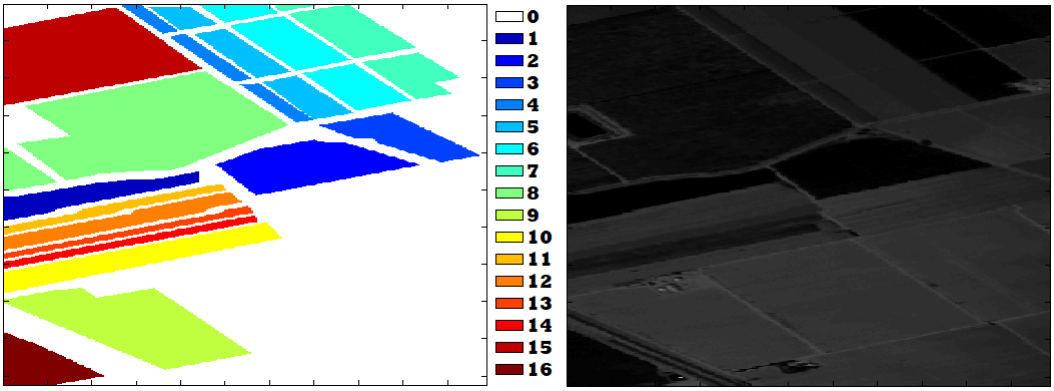

Figure 2.13: Ground truth of Salinas data set (left) and sample band of Salinas data set: 170 (right.)

#	Class	Sample
0	Empty-space	1790
1	Broccoli-green-weeds-1	391
10	Corn-senesced-green-weeds	1343
11	Lettuce-romaine-4wk	616
12	Lettuce-romaine-5wk	1525
13	Lettuce-romaine-6wk	674
14	Lettuce-romaine-7wk	799

Table 2.3: Salinas-A classes.

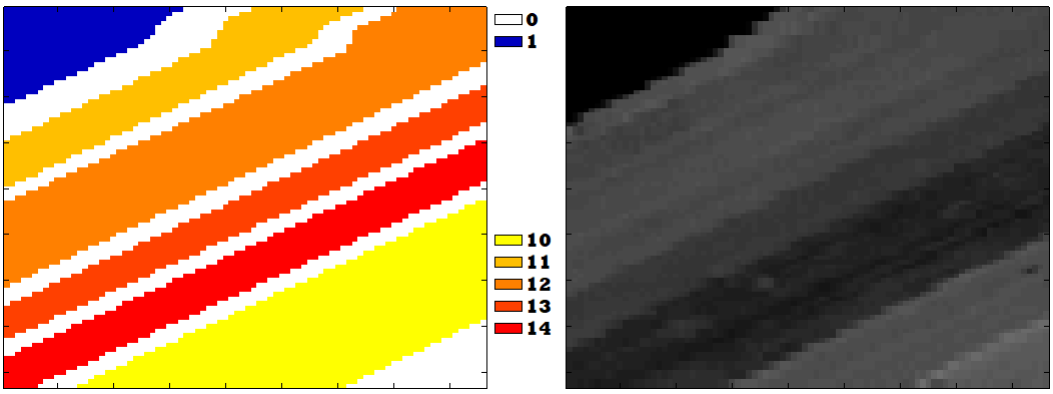

Figure 2.14: Ground truth of Salinas-A data set (left) and sample band of Salinas-A data set: 170 (right.)

#	Class	Sample
0	Empty-space	5826
1	Broccoli-green-weeds-1	914
2	Broccoli-green-weeds-2	1854
8	Grapes-untrained	1240
10	Corn-senesced-green-weeds	1959
11	Lettuce-romaine-4wk	655
12	Lettuce-romaine-5wk	1229
13	Lettuce-romaine-6wk	616
14	Lettuce-romaine-7wk	707

Table 2.4: Salinas-B classes.

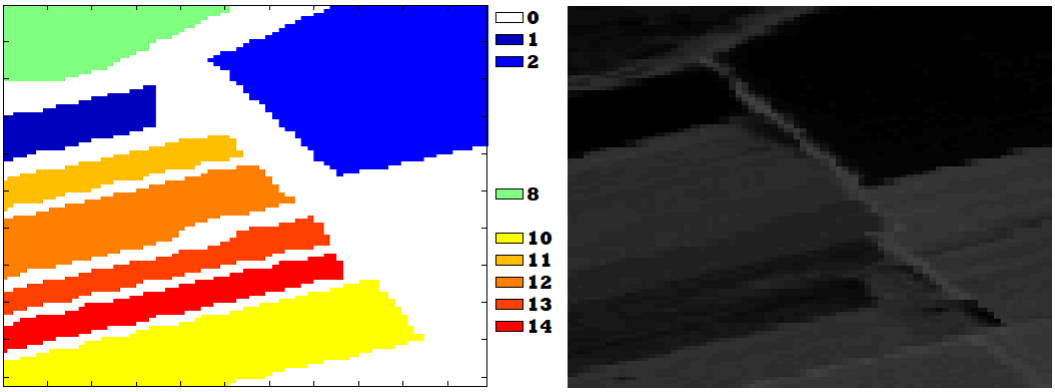

Figure 2.15: Ground truth of Salinas-B data set (left) and sample band of Salinas-B data set: 170 (right.)

#	Class	Sample
0	Empty-space	10776
1	Alfalfa	46
2	Corn	2495
5	Grass	1241
8	Hay-windrowed	478
9	Oats	20
10	Soybean	4020
13	Wheat	205
14	Woods	1265
15	Buildings-Grass-Trees-Drives	386
16	Stone-Steel-Towers	93

Table 2.5: Indian Pines-G classes, ground truth with corresponding grouped labels.

#	Class	Sample
0	Empty-space	5826
1	Broccoli-green-weeds-1	914
2	Broccoli-green-weeds-2	1854
8	Grapes-untrained	1240
10	Corn-senesced-green-weeds	1959
11	Lettuce-romaine	3207

Table 2.6: Salinas-B-G classes, ground truth with corresponding grouped labels.

2.7.2 Classification and validation metric

After the embedding, we use Matlab's 1-nearest neighbor algorithm to classify the data sets. We use 10% of the data from each class to train the classifier and the remaining number of data points, N_v, as the validation set. We took an average of ten runs the produce the confusion matrices (C), each using a disjoint set of data to train the classifier. We report the following validation metrics: the adjusted Rand index (ARI) between the predicted labels and the ground truth, the overall accuracy (OA), the average or weighted accuracy (AA), the average F-score (FS), and Cohen's kappa (κ) defined by

$$\kappa = \frac{N_v \sum_i (C_{i,i})^2 - \omega}{N_v^2 - \omega},$$

where $\omega = \sum_i C_{i,\cdot} C_{\cdot,i}$. Similar to ARI, the Cohen's kappa coefficient, κ, measures the degree of agreement between clusters, where a 0 indicates no agreement and a 1 indicates complete agreement between clusterings.

All the results presented in Section 2.7.3 were computed using $k = 12$ nearest neighbors and $\sigma = 1$ for the weight parameter. Additionally we chose the intrinsic dimension to be $m = 50$ and $m = 25$ for the Indian Pines and the Salinas-B data set respectively. The choice of our parameter came from two separate optimizations. The first was done by keeping the weight parameter fixed, $\sigma = 1$, while varying the number of neighbor k, depicted in Figure 2.16 (middle row), the second was done by keeping the number of neighbors fixed, $k = 12$, while varying the parameter σ shown in Figure 2.16 (bottom row). The results from the optimizations show that the choice of k and σ does not greatly affect the results; although, the higher the value for k the more connections is introduced between different classes, this leads to a higher number of misclassified samples and explains the slight decline in performance observed in Figure 2.16 (bottom). Therefore, the value for the weight parameter is fixed at $\sigma = 1$ for simplicity, while the value for the number of neighbors is fixed at $k = 12$ to ensure that we have connected graphs. During the optimizations described above, we also looked at the effect of the intrinsic dimension, m, on our validation metrics. The intrinsic dimension for the Indian Pines data set was chosen to reflect what was seen previously in literature and also based on our results in Figure 2.16 (top row) . Since the Salinas-B data set has never been analyzed before the intrinsic dimension was chosen solely based on our result in Figure 2.16 (top row).

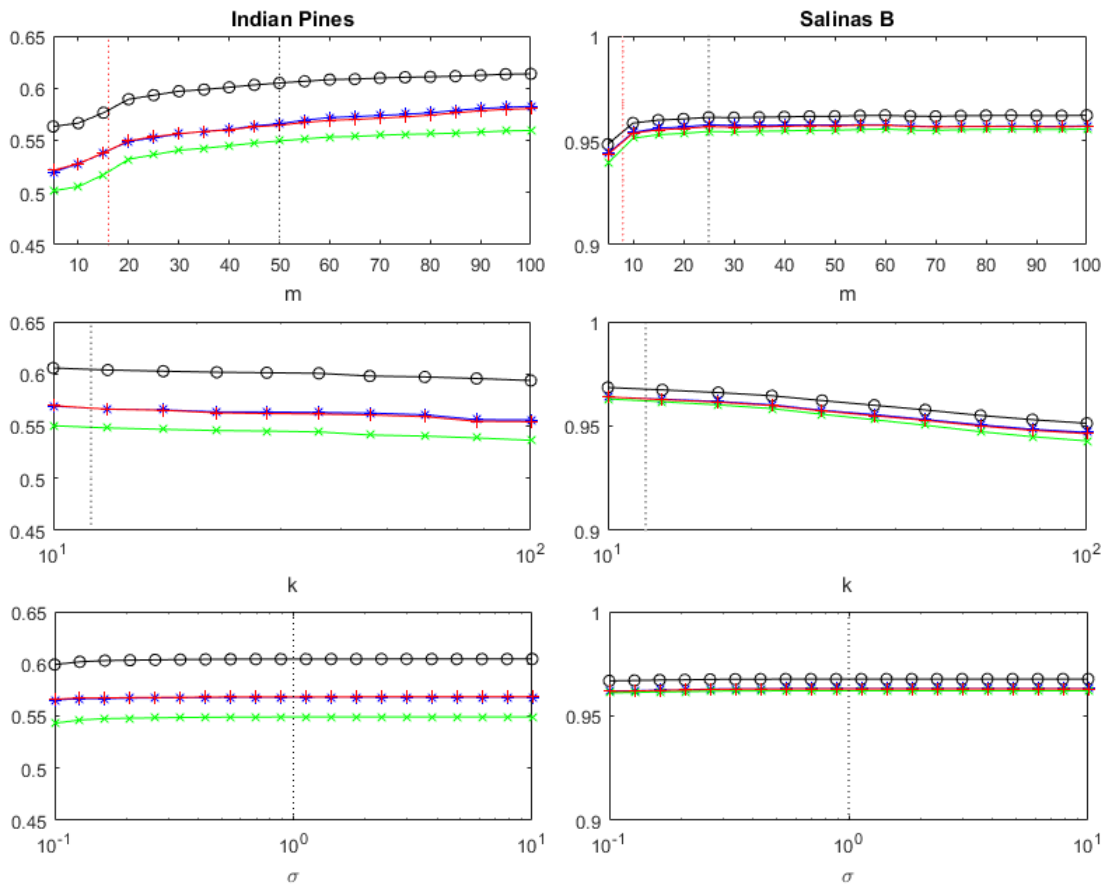

Figure 2.16: Optimal analysis of the parameters m (top), k (middle), and σ (bottom): overall accuracy (black circles), average accuracy (blue stars), F-score (red pluses), and Cohen's kappa coefficient (green x's). The red dashed vertical lines represent the number of classes and the black dashed vertical lines represent our choice for the parameter.

With k and σ fixed, the only free parameters remaining are α and β, introduced in the Schroedinger eigenmaps (SE) and the transport eigenmaps (TE) algorithm. For Schroedinger eigenmaps, we introduced the parameter $\hat{\alpha}$ defined by $\alpha = \hat{\alpha} \cdot \mathrm{tr}\,(L)/\mathrm{tr}\,(V)$ to balance the impact of the Laplacian matrix and the potential in the algorithm. We searched for the optimal value for $\hat{\alpha}$ for both Indian Pines and Salinas-B; see Figure 2.17. For transport eigenmaps, we searched directly for the

optimal value for β for both Indian Pines and Salinas-B, see Figure 2.18. All the results presented in Section 2.7.3 were computed using $\hat{\alpha} = 10^4$ for the Indian Pines data set and $\hat{\alpha} = 10^2$ for the Salinas-B data set for SE, and using $\beta = 10$ for both the Indian Pines and the Salinas-B data set for TE. In SE and TE, the extra information provided by the function μ comes from prior knowledge about the ground truth of some of the sample points in the data set. For instance, given that we have prior knowledge about class 11–soybean-mintill in the Indian Pines data set, we would place a potential for SE or an advection for TE on class 11–soybean-mintill using the function μ defined as follows:

$$\mu(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x} \in \text{Class 11–soybean-mintill,} \\ 0, & \text{elsewhere.} \end{cases}$$

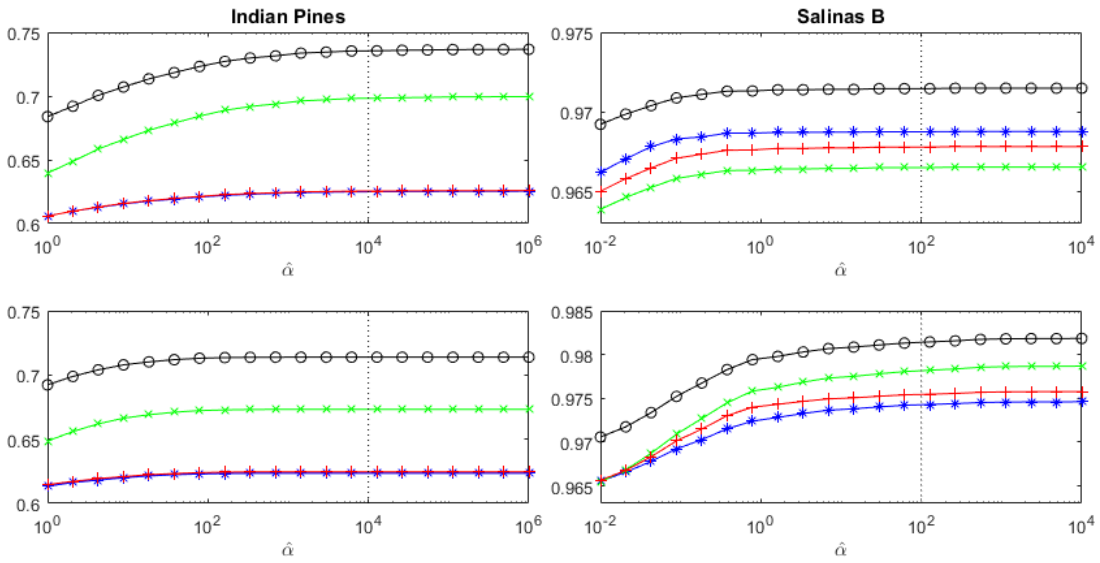

Figure 2.17: Optimal analysis of the parameter $\hat{\alpha}$. For the Indian Pines data set (left), the potential is placed on class 11–soybean (top) then on class 2–corn (bottom). For the Salinas-B data set (right), the potential is placed on class 11–lettuce (top) then on class 10–corn (bottom). The following performance measures are reported: overall accuracy (black circles), average accuracy (blue stars), F-score (red pluses), and Cohen's kappa coefficient (green x's). The black dashed vertical lines represent our choice for the parameter $\hat{\alpha}$.

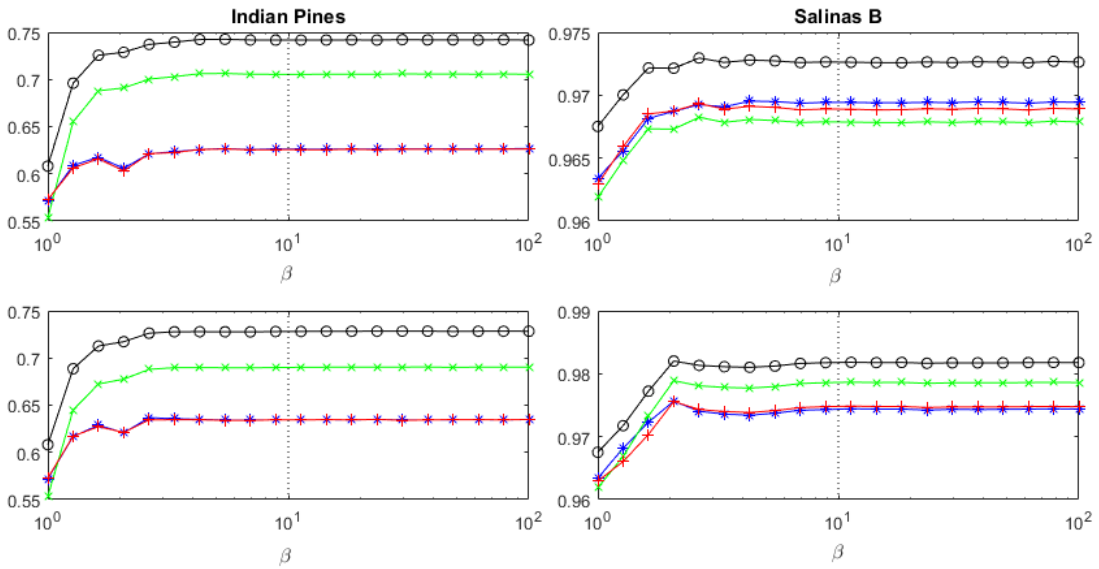

Figure 2.18: Optimal analysis of the parameter β. For the Indian Pines data set (left), the advection is placed on class 11–soybean (top) then on class 2–corn (bottom). For the Salinas-B data set (right), the advection is placed on class 11–lettuce (top) then on class 10–corn (bottom). The following performance measures are reported: overall accuracy (black circles), average accuracy (blue stars), F-score (red pluses), and Cohen's kappa coefficient (green x's). The black dashed vertical lines represent our choice for the parameter β.

2.7.3 Results

The following dimension reduction algorithms are used in the experiment: principal components analysis [78] (PCA), Laplacian eigenmaps [6] (LE), diffusion maps [18] (DIF), Isomap [98] (ISO), Schroedinger eigenmaps [14] (SE), transport eigenmaps (TE). Table 2.7 and Table 2.8 show the results for the Indian Pines and the Salinas-B images respectively. Table 2.9 and Table 2.10 show the results for the Indian Pines-G and the Salinas-B-G images respectively, after similar classes have been grouped together. The classification maps for each of the results are

also reported below each tables. Figure 2.19 and Figure 2.20 show the classification maps for the Indian Pines and the Salinas-B images respectively. Figure 2.21 and Figure 2.22 show the classification maps for the Indian Pines-G and the Salinas-B-G images respectively.

Similar to Section 2.6.3, we see that the transport model outperforms the other algorithms in representing the data, see the adjusted Rand index and the Cohen's kappa coefficient (in bold) in the aforementioned tables. A strong cluster agreement with the ground truth usually results in high overall accuracy, and that can be seen in tables as well. This quantitative performance is reflected in the quality of the aforementioned classification maps produced by the algorithm. The average accuracy and the F-score are validation metrics that serve as test scores to ensure that our results are not bias towards a few particular classes. A comparable average accuracy and the F-score across methods is an indication that the algorithms do not favor a few particular classes over the others.

IP	PCA	LE	DIF	ISO	SE-2	SE-11	TE-2	TE-11
ARI	0.4426	0.3694	0.4210	0.3929	0.5520	0.6955	**0.5735**	**0.7085**
OA	0.6761	0.6081	0.6556	0.6308	0.7138	0.7353	**0.7281**	**0.7418**
AA	0.6403	0.5719	0.6219	0.5979	0.6234	0.6249	0.6346	0.6257
FS	0.6471	0.5727	0.6211	0.5995	0.6245	0.6254	0.6347	0.6252
κ	0.6301	0.5532	0.6065	0.5785	0.6732	0.6981	**0.6900**	**0.7055**

Table 2.7: Classification results for Indian Pines (IP): the potential is placed on class 2–corn-notill, then on class 11–soybean-mintill for SE; and the advection is placed on class 2–corn-notill, then on class 11–soybean-mintill for TE

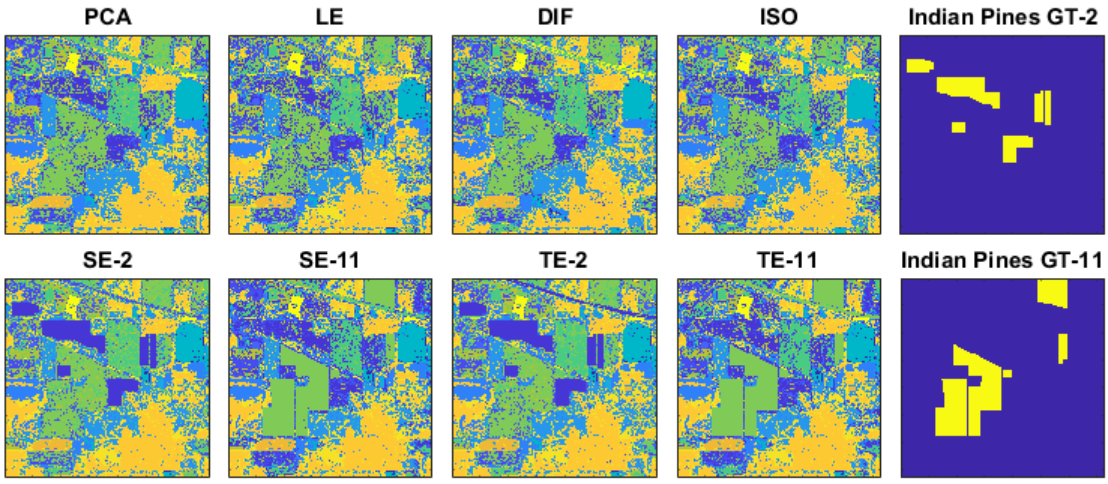

Figure 2.19: Classification map, the last column correspond to the ground truth (GT) data used in TE and SE algorithm.

SB	PCA	LE	DIF	ISO	SE-10	SE-14	TE-10	TE-14
ARI	0.9429	0.9323	0.9164	0.9439	0.9678	0.9439	**0.9708**	**0.9466**
OA	0.9729	0.9675	0.9603	0.9733	0.9814	0.9761	**0.9817**	**0.9781**
AA	0.9689	0.9633	0.9564	0.9699	0.9742	0.9777	0.9742	0.9802
FS	0.9692	0.9629	0.9556	0.9696	0.9754	0.9765	0.9747	0.9795
κ	0.9681	0.9618	0.9534	0.9686	0.9781	0.9720	**0.9785**	**0.9743**

Table 2.8: Classification results for Salinas-B (SB): the potential is placed on class 10–corn-senesced-green-weeds, then on class 14–lettuce-romaine-7wk for SE; and the advection is placed on class 10–corn-senesced-green-weeds, then on class 14–lettuce-romaine-7wk for TE

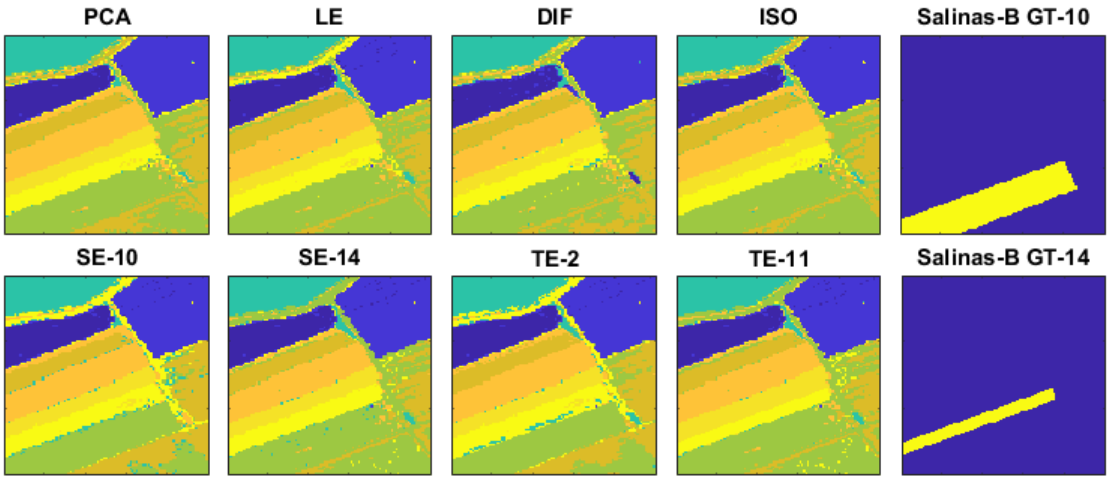

Figure 2.20: Classification map, the last column correspond to the ground truth (GT) data used in TE and SE algorithm.

IPG	PCA	LE	DIF	ISO	SE-2	SE-10	TE-2	TE-10
ARI	0.5329	0.4755	0.5102	0.4901	0.9148	0.8929	**0.9173**	**0.9311**
OA	0.7743	0.7285	0.7574	0.7418	**0.9180**	0.9088	0.9169	**0.9192**
AA	0.6986	0.6452	0.6883	0.6671	0.7128	0.7110	0.7101	0.7123
FS	0.7110	0.6472	0.6904	0.6738	0.7167	0.7156	0.7114	0.7148
κ	0.6995	0.6394	0.6769	0.6562	**0.8911**	0.8787	0.8896	**0.8926**

Table 2.9: Classification results for Indian Pines-G (IPG): the potential is placed on class 2–corn, then on class 10–soybean for SE; and the advection is placed on class 2–corn, then on class 10–soybean for TE

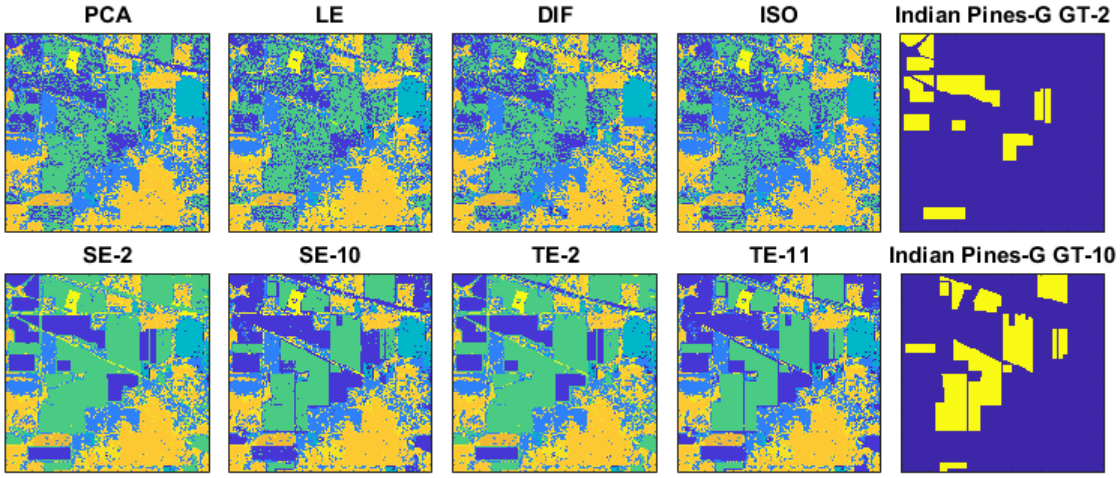

Figure 2.21: Classification map, the last column correspond to the ground truth (GT) data used in TE and SE algorithm.

SBG	PCA	LE	DIF	ISO	SE-10	SE-11	TE-10	TE-11
ARI	0.9459	0.9388	0.9153	0.9480	0.9779	0.9710	**0.9800**	**0.9767**
OA	0.9790	0.9755	0.9677	0.9795	0.9899	0.9858	**0.9910**	**0.9879**
AA	0.9769	0.9739	0.9668	0.9783	0.9876	0.9819	0.9884	0.9839
FS	0.9796	0.9754	0.9696	0.9797	0.9878	0.9828	0.9890	0.9849
κ	0.9725	0.9679	0.9576	0.9731	0.9868	0.9813	**0.9882**	**0.9842**

Table 2.10: Classification results for Salinas-B-G (SBG): the potential is placed on class 10–corn-senesced-green-weeds, then on class 11–lettuce for SE; and the advection is placed on class 10–corn-senesced-green-weeds, then on class 14–lettuce for TE

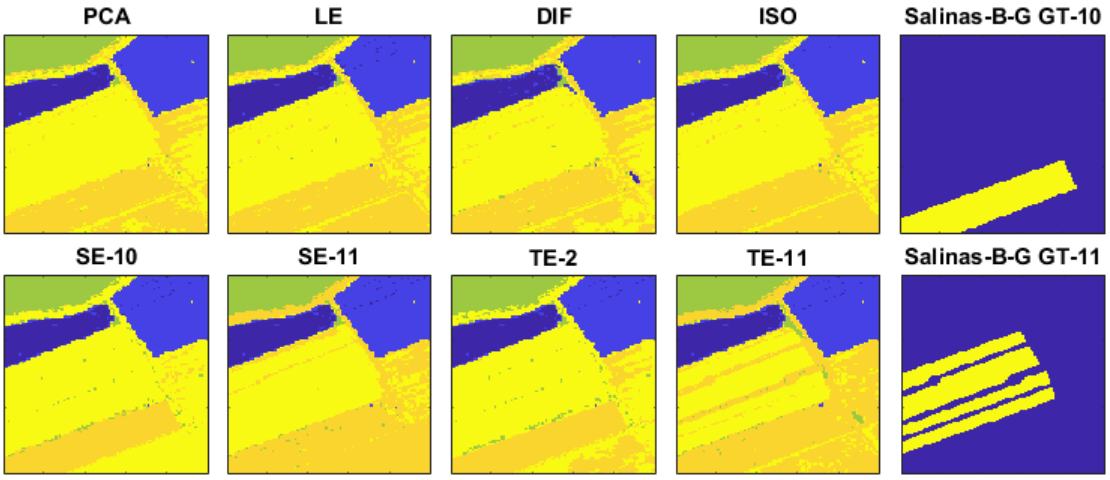

Figure 2.22: Classification map, the last column correspond to the ground truth (GT) data used in TE and SE algorithm.

We have done further experiments on Indian Pines-G and Salinas-G to see how the amount of information available from one particular class affects the performance measures for the transport and the Schroedinger algorithm. Figure 2.23 shows the

change in performance of TE from using 0% to using 100% of the ground truth with increments of 5% from a particular class. We observe that with no information provided, the performance measures of TE are the same as those of the LE. This result makes sense since without any extra information, TE and LE are exactly the same algorithm. However, as the amount of information given to TE increases, so do the performance measures. We repeat the latter experiment to include SE. Figure 2.24 shows the change in performance of TE and SE from using 0% to using 100% of the ground truth with increments of 5% from a particular class. Figure 2.25 shows the change in performance of TE and SE from using 80% to using 100% of the ground truth with increments of 1% from a particular class, this is done to have a closer look at the end behavior of the curves for TE and SE in Figure 2.24. We observe that although SE is ahead of TE given less than 100%, TE pulls ahead of SE when a more complete information is provided to both algorithms.

In lieu of these results, is it worth mentioning that in real-life applications, the extra information provided to the algorithms (TE and SE) does not come directly from the ground truth. Ideally, better and richer cluster information than the ground truth are produced using laboratory measurements and provided to the algorithms as the extra information. These laboratory measurements include various signals representing different materials in a wide range of conditions, e.g., lighting and weather. The use of the ground truth in our afformentioned results is simply due to the unavailability of those better and richer cluster information. Perhaps with a more complete set of laboratory measurements TE could surpass SE.

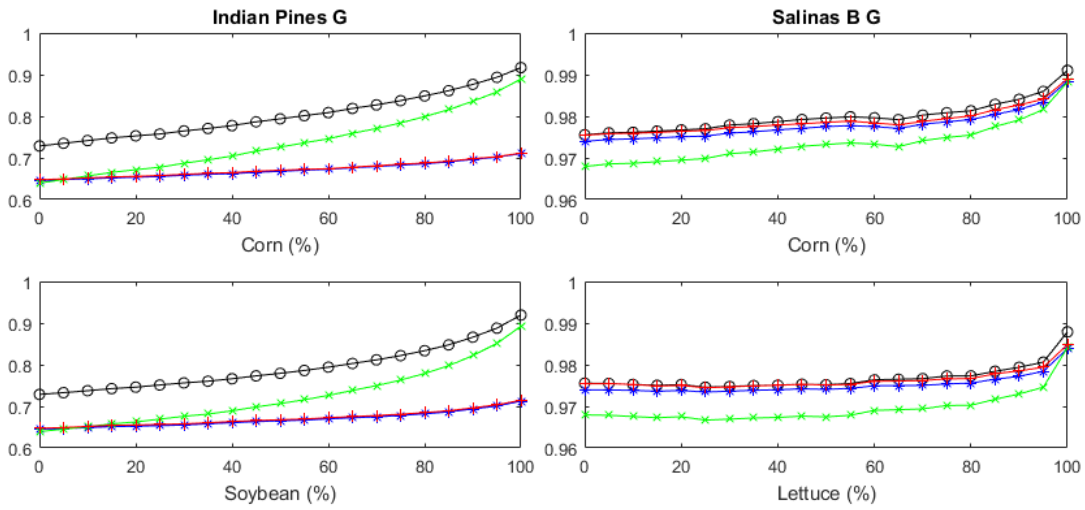

Figure 2.23: Classification performance measures for TE as a function of the amount of information provided. For Indian Pines-G (left) the advection is placed on class 2–corn (top), then on class 10–soybean (bottom). For Salinas-B-G (right) the advection is placed on class 10–corn-senesced-green-weeds (top), then on class 11–lettuce (bottom). The following performance measures are reported: overall accuracy (black circles), average accuracy (blue stars), F-score (red pluses), and Cohen's kappa coefficient (green x's).

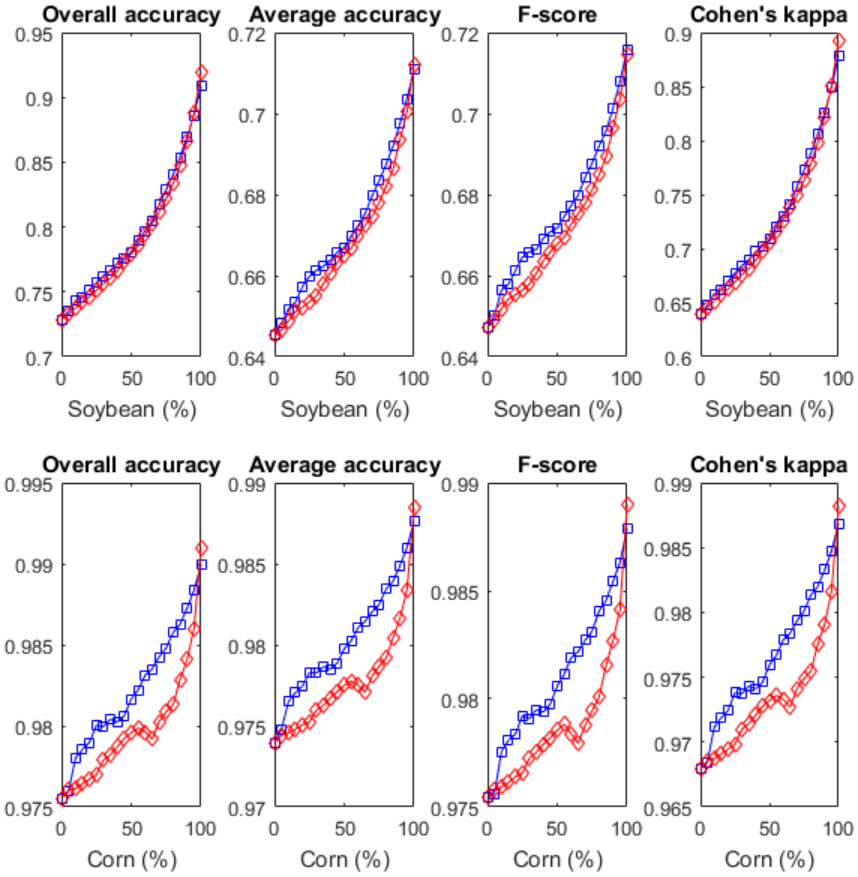

Figure 2.24: Classification performance measures for TE (red diamonds) and SE (blue squares) as a function of the amount of information provided, from 0% to 100% with increments of 5%. The Indian Pines-G data set (top row) is used with the advection and potential placed on class 10–soybean. The Salinas-B-G (bottom row) is used with the advection and potential placed on class 10–corn-senesced-green-weeds.

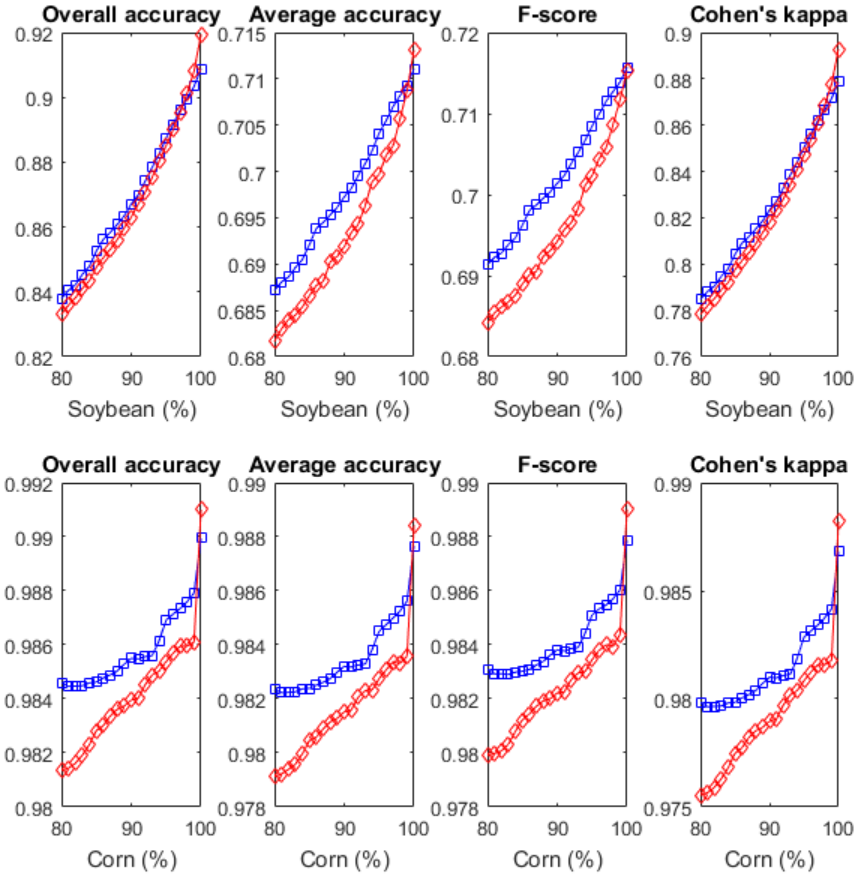

Figure 2.25: Classification performance measures for TE (red diamonds) and SE (blue squares) as a function of the amount of information provided, from 80% to 100% with increments of 1%. The Indian Pines-G data set (top row) is used with the advection and potential placed on class 10–soybean. The Salinas-B-G (bottom row) is used with the advection and potential placed on class 10–corn-senesced-green-weeds.

The last set of experiments compare the robustness of TE to other dimension reduction algorithms, viz., PCA, LE, and SE. For this experiment, we have added Gaussian noise to individual data points in the data set before it is processed by the dimension reduction algorithms. The added Gaussian noise has a mean of 0 and we selected 20 logarithmically spaced values for the standard deviation varying from

10^0 to 10^5 which covers the range for values taken by the individual data points in both set of data. In the case of TE and SE, the ground truths for class 10–soybean (Indian Pines-G) and class 11–lettuce (Salinas-B-G) are added to the algorithms. We gather from the experiments that SE and TE are more resilient to noise than PCA and LE; see Figure 2.26. Furthermore, although in general the decrease in performance happens almost at the same mark for all the algorithms, we notice that the performance measures associated with TE decline the slowest which makes TE the more most resilient algorithm.

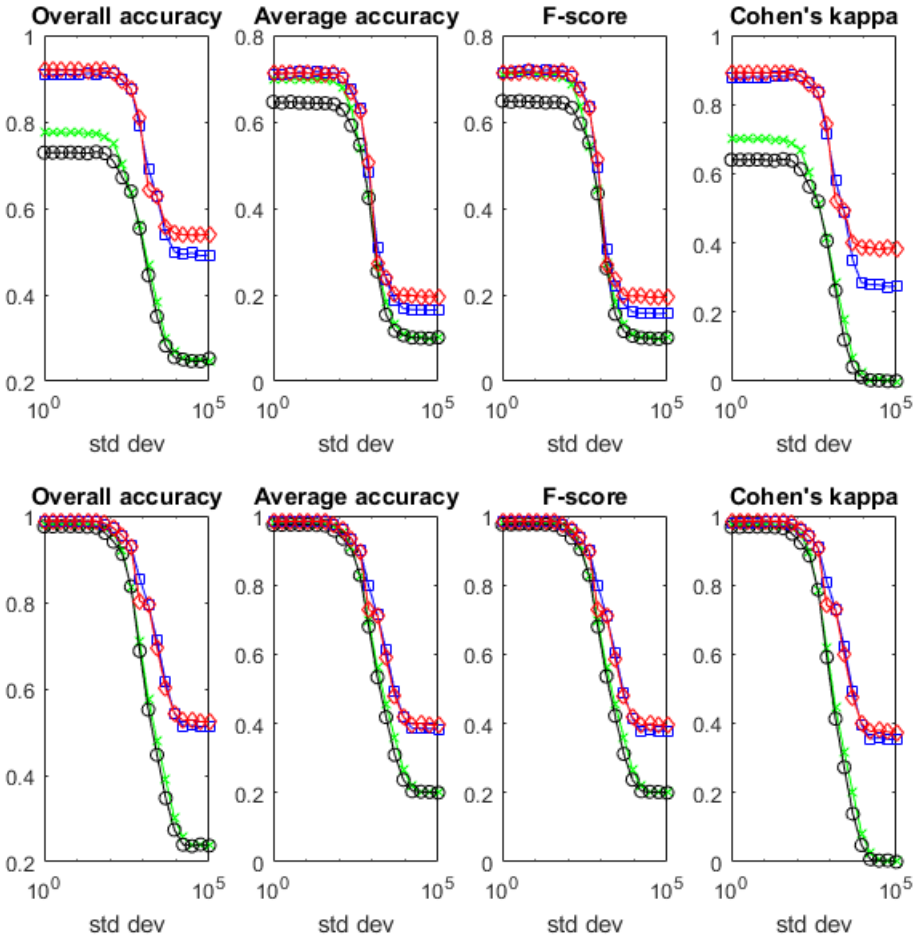

Figure 2.26: Classification performance measures for TE (red diamons), SE (blue boxes), PCA (green x's), and LE (black circles) as a function of noise. For Indian Pines-G (top row) the potential and advection are placed on class 10–soybean. For Salinas-B-G (bottom row) the potential and advection are placed on class 11–lettuce.

2.8 Conclusion

In this chapter, we proposed a novel approach to semi-supervised non-linear dimensionality reduction based on Laplacian eigenmaps. Although our algorithm is similar to Schroedinger eigenmaps, it is derived directly from a truly non-linear

transport model. We provide a set of experiments based on artificially generated data sets and on publicly available hyperspectral data sets to show that our algorithm exhibits superior/competitive performance to a variety of algorithms for reducing the dimension of the data provided to a standard classification algorithm. In this work, we do not discuss the time complexity aspect of our method compared to other methods. We also note that our algorithm can be improved by choosing alternative flow fields and/or using different linearization techniques. We will consider exploring these avenues in future further directions.

Chapter 3: Haar approximation from within for $L^p(\mathbb{R}^d)$, $0 < p < 1$

3.1 Introduction

The work in this chapter is in collaboration with Professor John Benedetto.

In 1909, Alfred Haar developed the Haar theory for $L^2(\mathbb{T})$ [39], i.e., the space of square integrable functions on $\mathbb{R}/\mathbb{Z}$. Yves Meyer and Stéphane Mallat introduced multiresolution analysis (MRA) on $\mathbb{R}^d$, e.g., see [63] (1989). This is a multiscale process inspired by Haar's ideas as well as more current work and models from speech [29, 57, 88] and image processing [65]. Mallat demonstrated that from any multiscale approximation, one can associate a generating function $\psi(t)$ known as the wavelet function such that the system $\{\psi_{m,n} : (m,n) \in \mathbb{Z} \times \mathbb{Z}\} = \{2^{m/2}\psi(2^m t - n) : (m,n) \in \mathbb{Z} \times \mathbb{Z}\}$ is an orthonormal basis for $L^2(\mathbb{R})$. MRA wavelet theory is now a highly developed area of mathematics [21, 64, 68], and with the relevance of the L^0 norm, e.g., in compressed sensing, it is natural to develop wavelet theory for L^p, $0 < p < 1$. A notable feature of the L^2 theory is the leakage of the approximants outside of the support of the given function. Our fundamental observation developed in this chapter is that if f is compactly supported, then the Haar wavelet approximants have support contained in the support of f when $0 < p < 1$. We quantify this assertion in Theorems 3.3.3, 3.3.15, 3.4.4, and 3.4.10; and illustrate these results in

our simulations in examples 3.4.5, 3.4.6, 3.4.11, and 3.4.12.

3.2 Haar function

The *Haar function* [39] named $\psi : \mathbb{R} \to \mathbb{C}$ is defined as

$$\psi(t) = \begin{cases} 1, & \text{if } t \in [0, 1/2), \\ -1, & \text{if } t \in [1/2, 1), \\ 0, & \text{otherwise,} \end{cases}$$

i.e., $\psi = \mathbb{1}_{[0,1/2)} - \mathbb{1}_{[1/2,1)}$, where $\mathbb{1}_X$ is the characteristic function of the set $X \subseteq \mathbb{R}$. Our main results in this paper concern Haar representation or wavelet system in the space $L^p(\mathbb{R}^d)$, where $0 < p < 1$ and $d = 1, 2$.

Definition 3.2.1. Given $f \in L^p(\mathbb{R}^d)$, $0 < p < 1$. We define the *pseudo-norm* of f as $\|f\|_{L^p(\mathbb{R}^d)}$ where

$$\|f\|_{L^p(\mathbb{R}^d)} = \left(\int_{\mathbb{R}^d} |f(t)|^p \, dt \right)^{1/p}.$$

Note that for $p \geq 1$, the function $\|\cdot\|_{L^p(\mathbb{R}^d)} : L^p(\mathbb{R}^d) \to [0, \infty)$ denotes the norm. For the remainder of the chapter, we refer to the pseudo-norm as the "norm," and will sometimes write $\|\cdot\|_{L^p(\mathbb{R}^d)}$ as $\|\cdot\|_p$ for simplicity. For $p < 1$, although we do not have an actual norm on the $L^p(\mathbb{R}^d)$ space, it is well known that we can define a metric if we omit the p^{th} root [22], [19].

Definition 3.2.2. Given $f, g \in L^p(\mathbb{R}^d)$, $0 < p < 1$. We define the metric function $\rho : L^p(\mathbb{R}^d) \times L^p(\mathbb{R}^d) \to [0, \infty)$ as

$$\rho(f, g) = \|f - g\|^p_{L^p(\mathbb{R}^d)} = \int_{\mathbb{R}^d} |f(t) - g(t)|^p \, dt.$$

Definition 3.2.3. Given the generating Haar function ψ and $p > 0$. The Haar dyadic wavelet system for ψ as an element of $L^p(\mathbb{R})$ is the sequence $\{\psi_{m,n} : (m,n) \in \mathbb{Z} \times \mathbb{Z}\}$, where each $\psi_{m,n}$ is defined as

$$\psi_{m,n}(t) = 2^{m/p}\psi(2^m t - n) \quad \text{on } \mathbb{R}.$$

We note that this system is fundamental in wavelet theory [15, 16, 21, 23, 52, 53, 62, 63, 68]. We denote the support of $\psi_{m,n}$ as $\overline{I_{m,n}}$, with $I_{m,n} = \left[\frac{n}{2^m}, \frac{n+1}{2^m}\right)$, where $\overline{X}$ is the closure of the set $X \subseteq \mathbb{R}$, and note that

$$\|\psi_{m,n}(t)\|_{L^p(\mathbb{R})} = \left(\int |\psi_{m,n}(t)|^p \, dt\right)^{1/p} = \left((2^{m/p})^p |I_{m,n}|\right)^{1/p} = \left(2^m 2^{-m}\right)^{1/p} = 1.$$

In $L^p(\mathbb{R}^2)$, there are three generating Haar functions making up the dyadic wavelet system. They are formed by tensor products of univariate wavelet ψ and unit function $\mathbb{1}_{[0,1)}$; viz., vertical (v), horizontal (h), and diagonal (d) Haar functions. They are defined as

$$\begin{aligned}
\psi^{(v)} &= (\mathbb{1}_{[0,1/2)} - \mathbb{1}_{[1/2,1)}) \times \mathbb{1}_{[0,1)}, \\
\psi^{(h)} &= \mathbb{1}_{[0,1)} \times (\mathbb{1}_{[0,1/2)} - \mathbb{1}_{[1/2,1)}), \\
\psi^{(d)} &= (\mathbb{1}_{[0,1/2)} - \mathbb{1}_{[1/2,1)}) \times (\mathbb{1}_{[0,1/2)} - \mathbb{1}_{[1/2,1)}).
\end{aligned}$$

Note that $\operatorname{supp}\psi^{(v)} = \operatorname{supp}\psi^{(h)} = \operatorname{supp}\psi^{(d)} = [0,1] \times [0,1] = [0,1]^2$. Definition 3.2.3 is extended $L^p(\mathbb{R}^2)$ in Definition 3.2.4.

Definition 3.2.4. Given the generating Haar functions $\psi^{(v)}$, $\psi^{(h)}$, and $\psi^{(d)}$, and $p > 0$. The Haar dyadic wavelet system for $\psi^{(v)}$, $\psi^{(h)}$, and $\psi^{(d)}$ as elements of $L^p(\mathbb{R}^2)$ is the sequence $\{\psi^{(v)}_{m,n}, \psi^{(h)}_{m,n}, \psi^{(d)}_{m,n} : (m,n) \in \mathbb{Z} \times \mathbb{Z}^2\}$, where each $\psi^{(v)}_{m,n}$, $\psi^{(h)}_{m,n}$,

and $\psi^{(d)}_{m,n}$ is defined as

$$\begin{aligned}
\psi^{(v)}_{m,n}(t) &= 2^{2m/p}\psi^{(v)}(2^m t - n), \\
\psi^{(h)}_{m,n}(t) &= 2^{2m/p}\psi^{(h)}(2^m t - n), \\
\psi^{(d)}_{m,n}(t) &= 2^{2m/p}\psi^{(d)}(2^m t - n) \quad \text{on } \mathbb{R}^2.
\end{aligned}$$

The support of $\psi^{(v)}_{m,n}$, $\psi^{(h)}_{m,n}$, and $\psi^{(d)}_{m,n}$ is $\overline{I_{m,n}}$, where $I_{m,n} = \left[\frac{n_1}{2^m}, \frac{n_1+1}{2^m}\right) \times \left[\frac{n_2}{2^m}, \frac{n_2+1}{2^m}\right)$.

3.3 Haar approximation

3.3.1 1-dimensional

Theorem 3.3.1. *Given $N > 0$. Let $f \in L^p(\mathbb{R})$, where $0 < p < 1$, and suppose $\operatorname{supp} f \subseteq [-2^N, 2^N]$. Assume that there exists M such that for $n \in \{-2^{M+N}, \ldots, 2^{M+N} - 1\}$, f is constant on $I_{M,n}$. Then, for all $\epsilon > 0$, there is a sequence of sums,*

$$f_{M,N,k} = \sum_{(i,j)\in S_{M,N,k}} a_{i,j}\psi_{i,j}, \quad a_{i,j} \in \mathbb{C},$$

indexed by $k \geq 1$, where $S_{M,N,k} \subseteq \mathbb{Z} \times \mathbb{Z}$ and $\operatorname{card} S_{M,N,k} < \infty$; and these sums have the following properties:

$$\text{if } (i,j) \in S_{M,N,k} \text{ then } \operatorname{supp}\psi_{i,j} \subseteq \operatorname{supp} f,$$

and

$$\exists K = K(\epsilon) \text{ such that } \forall k > K,\ \|f - f_{M,N,k}\|_p < \epsilon/2.$$

Proof. Let us define $f_0 = f$ so that f_0 is constant on $I_{M,n}$. We define f_{-1} as the *pairwise average* of the function f_0. By that we mean for all $t \in I_{M-1,n}$, where

$n \in \{-2^{M+N-1}, \ldots, 2^{M+N-1} - 1\}$,

$$f_{-1}(t) = \frac{1}{2}\left(f_0\left(\frac{2n}{2^M}\right) + f_0\left(\frac{2n+1}{2^M}\right)\right),$$

i.e., f_{-1} is constant on $I_{M-1,n}$. We define f_{-2} as the pairwise average of the function f_{-1}, i.e., f_{-2} is constant on $I_{M-2,n}$, and so on. We also define the error function $e_{-1} = f_0 - f_{-1}$, $e_{-2} = f_{-1} - f_{-2}$, and so on. First, we notice that

$$\begin{aligned}
e_{-1}\left(\frac{2n}{2^M}\right) &= f_0\left(\frac{2n}{2^M}\right) - f_{-1}\left(\frac{2n}{2^M}\right) \\
&= f_0\left(\frac{2n}{2^M}\right) - \frac{1}{2}\left(f_0\left(\frac{2n}{2^M}\right) + f_0\left(\frac{2n+1}{2^M}\right)\right) \\
&= \frac{1}{2}\left(f_0\left(\frac{2n}{2^M}\right) - f_0\left(\frac{2n+1}{2^M}\right)\right) \\
&= -f_0\left(\frac{2n+1}{2^M}\right) + \frac{1}{2}\left(f_0\left(\frac{2n}{2^M}\right) + f_0\left(\frac{2n+1}{2^M}\right)\right) \\
&= -f_0\left(\frac{2n+1}{2^M}\right) + f_{-1}\left(\frac{2n+1}{2^M}\right) \\
e_{-1}\left(\frac{2n}{2^M}\right) &= -e_{-1}\left(\frac{2n+1}{2^M}\right).
\end{aligned}$$

Since

$$\psi_{m,n}(t) = \begin{cases} 2^{m/p}, & \text{if } t \in I_{m+1,2n} \\ -2^{m/p}, & \text{if } t \in I_{m+1,2n+1}, \end{cases}$$

for $t \in I_{M-1,n}$, where $n \in \{-2^{N+M-1}, \ldots, 2^{N+M-1} - 1\}$, we have

$$\begin{aligned}
e_{-1}(t)\psi_{M-1,n}(t) &= \begin{cases} 2^{(M-1)/p}\, e_{-1}(\frac{2n}{2^M}), & \text{if } t \in I_{M,2n} \\ -2^{(M-1)/p}\, e_{-1}(\frac{2n+1}{2^M}), & \text{if } t \in I_{M,2n+1} \end{cases} \\
&= \begin{cases} 2^{(M-1)/p}\, e_{-1}(\frac{2n}{2^M}), & \text{if } t \in I_{M,2n} \\ 2^{(M-1)/p}\, e_{-1}(\frac{2n}{2^M}), & \text{if } t \in I_{M,2n+1}. \end{cases}
\end{aligned}$$

Therefore, for $t \in [-2^N, 2^N)$, we have

$$e_{-1}(t) = \sum_{n=-2^{N+M-1}}^{2^{N+M-1}-1} 2^{-(M-1)/p} e_{-1}(\frac{2n}{2^M}) \psi_{M-1,n}(t).$$

We now construct the sum to approximate f_0.

$$\begin{aligned}
f_0 &= f_{-1} + e_{-1} \\
&= f_{-1} + \sum_{n=-2^{N+M-1}}^{2^{N+M-1}-1} a_{M-1,n}\psi_{M-1,n} \\
&= f_{-2} + e_{-2} + \sum_{n=-2^{N+M-1}}^{2^{N+M-1}-1} a_{M-1,n}\psi_{M-1,n} \\
&= f_{-2} + \sum_{n=-2^{N+M-2}}^{2^{N+M-2}-1} a_{M-2,n}\psi_{M-2,n} + \sum_{n=-2^{N+M-1}}^{2^{N+M-1}-1} a_{M-1,n}\psi_{M-1,n} \\
&= f_{-2} + \sum_{m=1}^{2} \sum_{n=-2^{N+M-m}}^{2^{N+M-m}-1} a_{M-m,n}\psi_{M-m,n} \\
&\vdots \\
&= f_{+,-} + \sum_{m=1}^{M+N} \sum_{n=-2^{N+M-m}}^{2^{N+M-m}-1} a_{M-m,n}\psi_{M-m,n} \\
f_0 &= f_{+,-} + s_{M,N}.
\end{aligned}$$

With

$$a_{M-m,n} = 2^{-(M-m)/p}\, e_{-m}\left(\frac{2n}{2^{M-m+1}}\right) \quad \text{for all } m \text{ and } n,$$

and

$$f_{+,-} = f_+ + f_-,$$

where

$$\begin{aligned}
f_+ &= f^\alpha = \alpha\, \mathbb{1}_{[0,2^N)}, \quad \text{and} \quad \alpha = \frac{1}{2^{M+N}} \sum_{n=0}^{2^{M+N}-1} f_0\left(\frac{n}{2^M}\right); \\
f_- &= f^\beta = \beta\, \mathbb{1}_{[-2^N,0)}, \quad \text{and} \quad \beta = \frac{1}{2^{M+N}} \sum_{n=-2^{M+N}}^{-1} f_0\left(\frac{n}{2^M}\right).
\end{aligned}$$

We construct sums s^α, and s^β to approximate f^α, and f^β, respectively. For s^α, we proceed as follows. Define

$$\begin{aligned} f_1^\alpha = s_1^\alpha &= \alpha 2^{N/p}\psi_{-N,0} = \alpha\psi(2^{-N}t), \\ f_2^\alpha &= 2\alpha 2^{(N-1)/p}\psi_{-N+1,1} = 2\alpha\psi(2^{-N+1}t-1), \\ s_2^\alpha &= s_1^\alpha + f_2^\alpha, \\ f^\alpha - s_2^\alpha &= 4\alpha\,\mathbb{1}_{[2^N-2^N/2^2,2^N)}. \end{aligned}$$

Next, define

$$\begin{aligned} f_3^\alpha &= 4\alpha 2^{(N-2)/p}\psi_{-N+2,3} = 4\alpha\psi(2^{-N+2}t-3), \\ s_3^\alpha &= s_2^\alpha + f_3^\alpha, \\ f^\alpha - s_3^\alpha &= 8\alpha\,\mathbb{1}_{[2^N-2^N/2^3,2^N)}. \end{aligned}$$

In general, we have

$$\begin{aligned} f_k^\alpha &= 2^{k-1}\alpha 2^{(N-k+1)/p}\psi_{-N+k-1,2^{k-1}-1} = 2^{k-1}\alpha\psi(2^{-N+k-1}t-(2^{k-1}-1)), \\ s_k^\alpha &= s_{k-1}^\alpha + f_k^\alpha, \\ f^\alpha - s_k^\alpha &= 2^k\alpha\,\mathbb{1}_{[2^N-2^N/2^k,2^N)}. \end{aligned}$$

Consequently, for all $p>0$,

$$\|f^\alpha - s_k^\alpha\|_{L^p(\mathbb{R})} = ((2^k\alpha)^p 2^{N-k})^{1/p} = \alpha 2^{N/p}2^{k(1-1/p)},$$

and for all $0<p<1$,

$$\lim_{k\to\infty}\|f^\alpha - s_k^\alpha\|_{L^p(\mathbb{R})} = 0.$$

Thus for $\epsilon > 0$ as in the statement, we take $\epsilon_0 = (\epsilon/2)(1/2^{1/p})$. Then there exists $K > 0$ such that

$$\forall k > K,\ \|f^\alpha - s_k^\alpha\|_{L^p(\mathbb{R})} < \epsilon_0,$$

therefore,

$$\rho(f^\alpha, s_k^\alpha) = (2^k\alpha)^p 2^{N-k} < \frac{1}{2}\left(\frac{\epsilon}{2}\right)^p.$$

Similarly, for s^β, we have

$$\begin{aligned}
f_1^\beta = s_1^\beta &= -\beta 2^{N/p}\psi_{-N,-1} = -\beta\psi(2^{-N}t + 1), \\
f_k^\beta &= -2^{k-1}\beta 2^{(N-k+1)/p}\psi_{-N+k-1,-2^{k-1}} = -2^{k-1}\beta\psi(2^{-N+k-1}t + 2^{k-1}), \\
s_k^\beta &= s_{k-1}^\beta + f_k^\beta, \\
f^\beta - s_k^\beta &= 2^k\beta\,\mathbb{1}_{[-2^N, -2^N + 2^N/2^k)}.
\end{aligned}$$

Consequently, for all $p > 0$,

$$\|f^\beta - s_k^\beta\|_{L^p(\mathbb{R})} = ((2^k\beta)^p 2^{N-k})^{1/p} = \beta 2^{N/p} 2^{k(1-1/p)},$$

and for all $0 < p < 1$,

$$\lim_{k\to\infty} \|f^\beta - s_k^\beta\|_{L^p(\mathbb{R})} = 0.$$

Thus for $\epsilon > 0$ as in the statement, we take $\epsilon_0 = (\epsilon/2)(1/2^{1/p})$. Then there exists $K > 0$ such that

$$\forall k > K,\ \|f^\beta - s_k^\beta\|_{L^p(\mathbb{R})} < \epsilon_0,$$

therefore,

$$\rho(f^\beta, s_k^\beta) = (2^k\beta)^p 2^{N-k} < \frac{1}{2}\left(\frac{\epsilon}{2}\right)^p.$$

Let $c_m = \max\{\alpha, \beta\}$, we need

$$(2^k\alpha)^p 2^{N-k} + (2^k\beta)^p 2^{N-k} < \left(\frac{\epsilon}{2}\right)^p,$$

this means

$$k > \frac{\log(\epsilon) - \log(2^{1+1/p} c_m 2^{N/p})}{(1-1/p)\log(2)}.$$

Take

$$K = \left\lceil \frac{\log(\epsilon) - \log(2^{1+1/p} c_m 2^{N/p})}{(1-1/p)\log(2)} \right\rceil,$$

and

$$f_{M,N,k} = s_k^\alpha + s_k^\beta + s_{M,N};$$

therefore, for all $k > K$ and for all $0 < p < 1$ we have

$$\begin{aligned} \|f - f_{M,N,k}\|_p^p &= \rho(f, f_{M,N,k}) \\ &\leq \rho(f^\alpha, s_k^\alpha) + \rho(f^\beta, s_k^\beta) \\ &< \frac{1}{2}\left(\frac{\epsilon}{2}\right)^p + \frac{1}{2}\left(\frac{\epsilon}{2}\right)^p \\ &< \left(\frac{\epsilon}{2}\right)^p, \end{aligned}$$

thus,

$$\|f - f_{M,N,k}\|_p^p < \epsilon/2.$$

□

Theorem 3.3.2. *Given $N > 0$. Let $f \in L^p(\mathbb{R})$, where $0 < p < 1$, and suppose f is a continuous function on $\mathbb{R}$, with supp $f \subseteq [-2^N, 2^N]$. Then,*

$$\forall \epsilon > 0,\ \exists M = M(\epsilon) > 0 \text{ and } f_M, \quad \text{such that}$$

$$\forall n \in \{-2^{M+N}, \ldots, 2^{M+N}-1\},\ f_M \text{ is constant on } I_{M,n},$$

and

$$\|f - f_M\|_p < \epsilon/2.$$

Proof. Define f_M as

$$f_M(t) = \sum_{n=-2^{M+N}}^{2^{M+N}-1} f(t_n)\, \mathbb{1}_{I_{M,n}}, \quad \text{where } t_n \in I_{M,n}.$$

The sequence $\{f_M\},\ M = 0, 1, 2, \ldots$ is uniformly convergent to f on $[-2^N, 2^N)$. Thus, for $\epsilon > 0$ as in the statement, we take $\epsilon_0 = (\epsilon/2)/(2^{(N+1)/p})$. Then, there is $M_0 > 0$ such that,

$$\forall M > M_0 \text{ and } \forall t \in [-2^N, 2^N), \quad |f(t) - f_M(t)| < \epsilon_0.$$

Hence, for $M > M_0$ we have

$$\begin{aligned}
\|f - f_M\|_p &= \left(\int |f(t) - f_M(t)|^p \, dt \right)^{1/p} \\
&\leq \left(\int_{-2^N}^{2^N} (\sup_{t \in [-2^N, 2^N)} |f(t) - f_M(t)|)^p \, dt \right)^{1/p} \\
&< \left(\int_{-2^N}^{2^N} (\epsilon_0)^p \, dt \right)^{1/p} \\
&= (2^{N+1} \epsilon_0^p)^{1/p} \\
&= 2^{(N+1)/p} \frac{\epsilon/2}{2^{(N+1)/p}} \\
&= \epsilon/2.
\end{aligned}$$

□

Theorem 3.3.3. *Given $N > 0$. Let $f \in L^p(\mathbb{R})$, where $0 < p < 1$ suppose f is a continuous function on $\mathbb{R}$, with supp $f \subseteq [-2^N, 2^N]$. Then, for all $\epsilon > 0$, there is an*

$M = M(\epsilon)$, and there is a sequence of sums,

$$f_{M,N,k} = \sum_{(i,j)\in S_{M,N,k}} a_{i,j}\psi_{i,j}, \quad a_{i,j} \in \mathbb{C},$$

indexed by $k \geq 1$, where $S_{M,N,k} \subseteq \mathbb{Z} \times \mathbb{Z}$ and $\operatorname{card} S_{M,N,k} < \infty$; and these sums have the following properties:

$$\text{if } (i,j) \in S_{M,N,k} \text{ then } \operatorname{supp}\psi_{i,j} \subseteq \operatorname{supp} f,$$

and

$$\exists K = K(\epsilon) \text{ such that } \forall k > K,\ \|f - f_{M,N,k}\|_p < \epsilon.$$

Proof. Given $\epsilon > 0$. From Theorem 3.3.2 we know that there is an M and an f_M such that

$$\forall n \in \{-2^{M+N}, \ldots, 2^{M+N} - 1\},\ f_M \text{ is constant on } I_{M,n},$$

and

$$\|f - f_M\|_p < \frac{\epsilon}{2^{1/p}}.$$

Therefore,

$$\rho(f, f_M) < \epsilon^p/2. \tag{3.1}$$

Additionally, from Theorem 3.3.1, we know that there is a $K = K(\epsilon)$ and a sequence of sums,

$$f_{M,N,k} = \sum_{(i,j)\in S_{M,N,k}} a_{i,j}\psi_{i,j}, \text{ where each } \quad a_{i,j} \in \mathbb{C}$$

such that

$$\forall k > K,\ \|f_M - f_{M,N,k}\|_p < \frac{\epsilon}{2^{1/p}}.$$

Therefore,

$$\forall k > K,\ \rho(f_M, f_{M,N,k}) < \epsilon^p/2. \tag{3.2}$$

From (3.1) and (3.2) we have

$$\rho(f, f_{M,N,k}) \leq \rho(f, f_M) + \rho(f_M, f_{M,N,k}) = \epsilon^p,$$

therefore,

$$\forall k > K,\ \|f - f_{M,N,k}\|_p < \epsilon.$$

□

Remark 3.3.4. Major work in wavelet theory for $0 < p < 1$ is due to Oswald [74], Filippov [30, 31, 33], and Laugesen [59]. They use the terminology, "representation systems", as well as wavelet systems, and they have provided a significant and deep analysis of such systems, e.g., including modulus of continuity criteria. Although Theorem 3.3.3 is a special case of one of their results, i.e., Theorem 1 [33], Filippov and Oswald do not quantify the weights of the approximants nor study their supports. In fact, Theorem 3.3.3, with its emphasis on support, was proved independently by Benedetto about 1990 using standard wavelet techniques that had been recently developed for L^2 by Meyer [68], Daubechies [21], and Mallat [65].

Researchers have made important contributions to the study of representation systems [24, 30, 31, 50, 75, 76, 99] since they were introduced in 1968 by Talaljan as a generalization of the notion of basis [95]. These contributions range from approximation by polynomials [25, 85, 90, 91] and by series of exponentials [56], to more general integer translates [3]. Further work has been done to study approximation

properties of representation systems for complete orthogonal systems [95, 96], and to provide conditions under which subsystems or incomplete systems would satisfy the properties of representation systems [30–33, 97].

Remark 3.3.5. There are several mathematical motivations for having the support of wavelet approximants being contained in the support of the given function. In measure theory, if μ is a complex regular Borel measure on $\mathbb{R}$, then μ is an element of the dual space of $C_0(\mathbb{R})$, which is the space of continuous functions vanishing at infinity, taken with the sup norm (by the Riesz representation theorem). Any such μ can be approximated in the weak $*$-topology by a net of discrete measures, where the supports of the discrete measures are contained in $\operatorname{supp}\mu$, see, e.g., [10, 42]. Another motivation is from the topic of spectral synthesis where synthesizable pseudo-measures μ are defined by the property that they can be approximated in the weak $*$-topology by discrete measures whose supports are contained in $\operatorname{supp}\mu$, see, e.g., [9].

Example 3.3.6. Figure 3.1, shows a graphical representation of the averages' and errors' constructions used in the proof of Theorem 3.3.1.

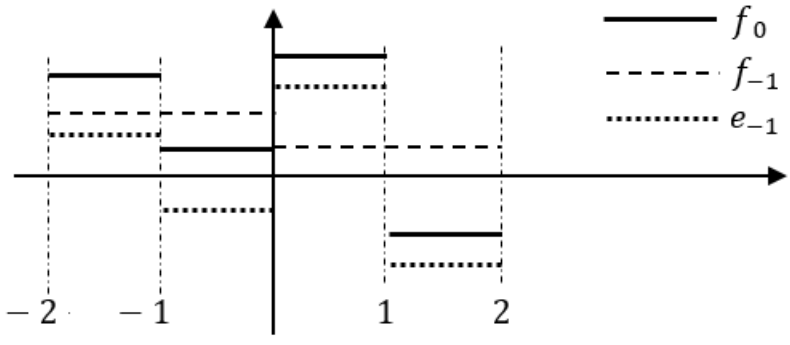

Figure 3.1: Construction of f_{-1} and e_{-1} from f_0 with $N = 1$ and $M = 0$.

Example 3.3.7. This numerical example leads to the constants α and β in the proof of Theorem 3.3.1. For illustration purpose the function $f_0 = f$ is defined in $[-2, 2)$, with $N = 1$ and $M = 1$. The function f_0 can be viewed as the following vector:

$$\begin{aligned}
f_0 &= [\ 2\ 4\ 0\ -2\ -4\ 0\ 4\ 8\ ] \\
f_{-1} &= [\ 3\ 3\ -1\ -1\ -2\ -2\ 6\ 6\ ] \\
e_{-1} = f_0 - f_{-1} &= [\ -1\ 1\ 1\ -1\ -2\ 2\ -2\ 2\ ] \\
f_{-2} &= [\ 1\ 1\ 1\ 1\ 2\ 2\ 2\ 2\ ] \\
e_{-2} = f_{-1} - f_{-2} &= [\ 2\ 2\ -2\ -2\ -4\ -4\ 4\ 4\ ]
\end{aligned}$$

Since $m = 1 \ldots M + N$, this is as far as we go and obtain $\alpha = 2$, and $\beta = 1$. In this example, $f_{+,-} = f_{-2}$.

Example 3.3.8. Figure 3.2 shows a graphical representation of the construction of s^α used in the proof of Theorem 3.3.1.

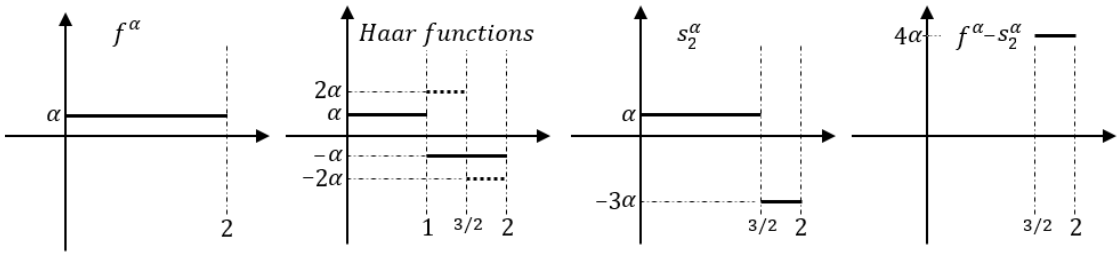

Figure 3.2: Construction of s_2^α using a sum of Haar functions. From left to right, the first plot represents the function f^α, the second plot represents the functions f_1^α and f_2^α in solid and dotted lines respectively, the third plot represents the sum $s_2^\alpha = f_1^\alpha + f_2^\alpha$, and the fourth plot represents the approximation error $f^\alpha - s_2^\alpha$.

Example 3.3.9. Figure 3.3 shows an example of the sum approximating the com-

pactly supported normalized *sinc function*:

$$f(t) = \frac{\sin(\pi t)}{\pi t} \mathbb{1}_{[-2,2)}.$$

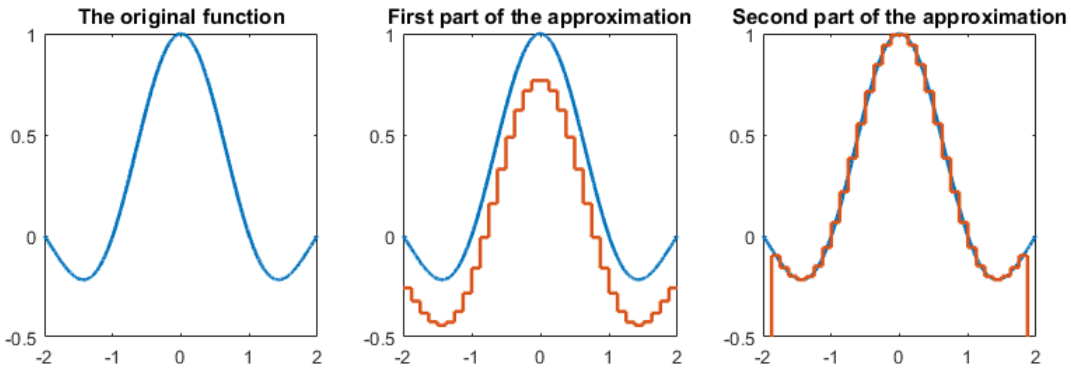

Figure 3.3: Approximation of the continuous compactly supported sinc function with $N = 1$ and $M = 3$, represented in blue in all three plot. From left to right, in the second plot the red graph (step function) corresponds to the double sum $s_{M,N} = f_0 - f_{+,-}$, and in the third plot the red graph (step function) corresponds to the final approximation $f_{3,1,3}$ of f.

Example 3.3.10. Figure 3.4 shows an example of the sum approximating the compactly supported *Gaussian function*:

$$f(t) = e^{-\pi t^2} \mathbb{1}_{[-2,2)}.$$

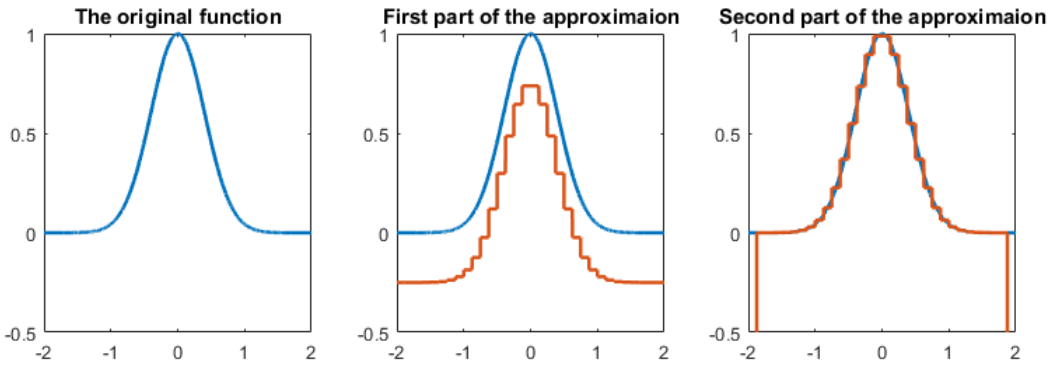

Figure 3.4: Approximation of the continuous compactly supported Gaussian function with $N = 1$ and $M = 3$, represented in blue in all three plot. From left to right, in the second plot the red graph (step function) corresponds to the double sum $s_{M,N} = f_0 - f_{+,-}$, and in the third plot the red graph (step function) corresponds to the final approximation $f_{3,1,3}$ of f.

Example 3.3.11. Figure 3.5 shows an example of the sum approximating the compactly supported *Poisson function*:

$$f(t) = e^{-2\pi|t|}\mathbb{1}_{[-2,2)}.$$

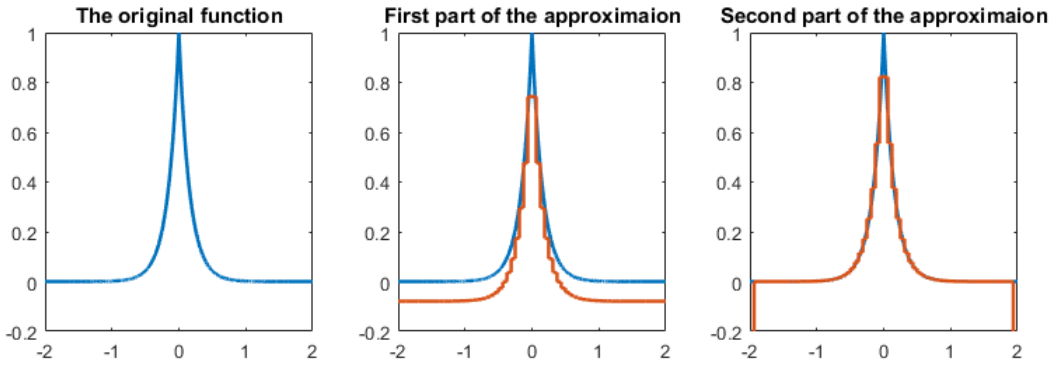

Figure 3.5: Approximation of the continuous compactly supported Poisson function with $N = 1$ and $M = 4$, represented in blue in all three plot. From left to right, in the second plot the red graph (step function) corresponds to the double sum $s_{M,N} = f_0 - f_{+,-}$, and in the third plot the red graph (step function) corresponds to the final approximation $f_{4,1,4}$ of f.

Example 3.3.12. Figure 3.6 shows an example of the sum approximating the compactly supported *Fejér function*:

$$f(t) = \left(\frac{\sin(\pi t)}{\pi t}\right)^2 \mathbb{1}_{[-2,2)}.$$

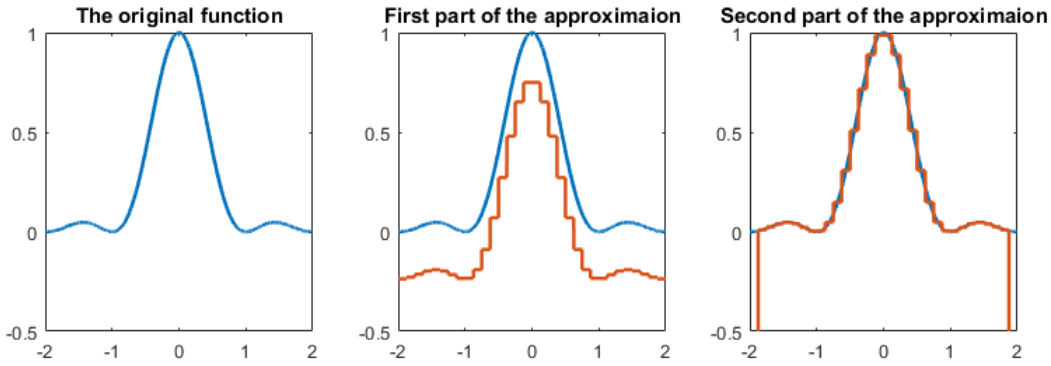

Figure 3.6: Approximation of the continuous compactly supported Fejér function with $N = 1$ and $M = 3$, represented in blue in all three plot. From left to right, in the second plot the red graph (step function) corresponds to the double sum $s_{M,N} = f_0 - f_{+,-}$, and in the third plot the red graph (step function) corresponds to the final approximation $f_{3,1,3}$ of f.

3.3.2 2-dimensional

Theorem 3.3.13. *Given $N > 0$. Let $f \in L^p(\mathbb{R}^2)$, where $0 < p < 1$, and suppose $\operatorname{supp} f \subseteq [-2^N, 2^N] \times [-2^N, 2^N]$. Assume that there exists M such that for $n = (n_1, n_2) \in \{-2^{M+N}, \ldots, 2^{M+N} - 1\} \times \{-2^{M+N}, \ldots, 2^{M+N} - 1\}$, f is constant on $S_{M,n}$. Then for all $\epsilon > 0$, there is a sequence of sums,*

$$f_{M,N,k} = \sum_{(i,j)\in S_{M,N,k}} a_{i,j}^{(v)}\psi_{i,j}^{(v)} + a_{i,j}^{(h)}\psi_{i,j}^{(h)} + a_{i,j}^{(d)}\psi_{i,j}^{(d)}, \quad a_{i,j}^{(v)}, a_{i,j}^{(h)}, a_{i,j}^{(d)} \in \mathbb{C},$$

indexed by $k \geq 1$*, where* $S_{M,N,k} \subseteq \mathbb{Z} \times \mathbb{Z}^2$ *and* $\operatorname{card} S_{M,N,k} < \infty$*; and these sums have the following properties:*

$$\text{if } (i,j) \in S_{M,N,k} \text{ then } \operatorname{supp} \psi_{i,j}^{(v)}, \psi_{i,j}^{(h)}, \psi_{i,j}^{(d)} \subseteq \operatorname{supp} f,$$

and

$$\forall k > K,\ \|f - f_{M,N,k}\|_p < \epsilon/2.$$

Proof. Let us define $f_0 = f$ so that f_0 is constant on $S_{M,n}$. We define f_{-1} as the 2×2-*block average* of the function f_0. By that we mean for all $t = (t_1, t_2) \in S_{M-1,n}$, where $n = (n_1, n_2) \in \{-2^{M+N-1}, \ldots, 2^{M+N-1} - 1\} \times \{-2^{M+N-1}, \ldots, 2^{M+N-1} - 1\}$,

$$f_{-1}(t) = \frac{1}{4}\left(f_0\left(\frac{2n}{2^M}\right) + f_0\left(\frac{2n_1}{2^M}, \frac{2n_2+1}{2^M}\right) + f_0\left(\frac{2n_1+1}{2^M}, \frac{2n_2}{2^M}\right) + f_0\left(\frac{2n+1}{2^M}\right)\right),$$

i.e., f_{-1} is constant on $S_{M-1,n}$. We define f_{-2} as the 2×2-block average of the function f_{-1}, i.e., f_{-2} is constant on $S_{M-2,n}$, and so on. Let us also define the error function $e_{-1} = f_0 - f_{-1}$, i.e., e_{-1} is constant on $S_{M,n}$; and $e_{-2} = f_{-1} - f_{-2}$, i.e., e_{-2} is constant on $S_{M-1,n}$; and so on. With a method similar to the one used in the proof of Theorem 3.3.1 we approximate f_0 as

$$f_0 = f_{+,-} + s_{M,N},$$

where

$$s_{M,N} = \sum_{m=1}^{M+N} \sum_{n_1} \sum_{n_2} a_{M-m,n}^{(v)} \psi_{M-m,n}^{(v)} + a_{M-m,n}^{(h)} \psi_{M-m,n}^{(h)} + a_{M-m,n}^{(d)} \psi_{M-m,n}^{(d)},$$

and

$$\begin{aligned}
a^{(v)}_{M-m,n} &= 2^{-2(M-m)/p}\, e^{(v)}_{-m}\left(\frac{2n_1}{2^{M-m+1}}, \frac{2n_2}{2^{M-m+1}}\right), \\
a^{(h)}_{M-m,n} &= 2^{-2(M-m)/p}\, e^{(h)}_{-m}\left(\frac{2n_1}{2^{M-m+1}}, \frac{2n_2}{2^{M-m+1}}\right), \\
a^{(d)}_{M-m,n} &= 2^{-2(M-m)/p}\, e^{(d)}_{-m}\left(\frac{2n_1}{2^{M-m+1}}, \frac{2n_2}{2^{M-m+1}}\right),
\end{aligned}$$

for all m and $n = (n_1, n_2)$. The functions $e^{(v)}_{-m}$, $e^{(h)}_{-m}$ and $e^{(d)}_{-m}$ are respectively the *vertical*, *horizontal*, and *diagonal blockwise average* of e_{-m}. Note that the function e_{-m} is constant on $S_{M-m+1,n}$ and so would the functions $e^{(v)}_{-m}$, $e^{(h)}_{-m}$ and $e^{(d)}_{-m}$. The latter averages are defined as follow, for all $t \in S_{M-m,n}$, where $n \in \{-2^{M+N-m}, \ldots, 2^{M+N-m}-1\} \times \{-2^{M+N-m}, \ldots, 2^{M+N-m}-1\}$, and $m \in \{1, \ldots, M+N\}$,

$$\begin{aligned}
e^{(v)}_{-m}\left(\frac{2n}{2^M}\right) &= e^{(v)}_{-m}\left(\frac{2n_1}{2^M}, \frac{2n_2+1}{2^M}\right) \\
&= \frac{1}{2}\left(e_{-m}\left(\frac{2n}{2^M}\right) + e_{-m}\left(\frac{2n_1}{2^M}, \frac{2n_2+1}{2^M}\right)\right), \\
e^{(v)}_{-m}\left(\frac{2n_1+1}{2^M}, \frac{2n_2}{2^M}\right) &= e^{(v)}_{-m}\left(\frac{2n+1}{2^M}\right) \\
&= \frac{1}{2}\left(e_{-m}\left(\frac{2n_1+1}{2^M}, \frac{2n_2}{2^M}\right) + e_{-m}\left(\frac{2n+1}{2^M}\right)\right);
\end{aligned}$$

$$\begin{aligned}
e^{(h)}_{-m}\left(\frac{2n}{2^M}\right) &= e^{(h)}_{-m}\left(\frac{2n_1+1}{2^M}, \frac{2n_2}{2^M}\right) \\
&= \frac{1}{2}\left(e_{-m}\left(\frac{2n}{2^M}\right) + e_{-m}\left(\frac{2n_1+1}{2^M}, \frac{2n_2}{2^M}\right)\right), \\
e^{(h)}_{-m}\left(\frac{2n_1}{2^M}, \frac{2n_2+1}{2^M}\right) &= e^{(h)}_{-m}\left(\frac{2n+1}{2^M}\right) \\
&= \frac{1}{2}\left(e_{-m}\left(\frac{2n_1}{2^M}, \frac{2n_2+1}{2^M}\right) + e_{-m}\left(\frac{2n+1}{2^M}\right)\right); \quad \text{and}
\end{aligned}$$

$$
\begin{aligned}
e^{(d)}_{-m}\left(\frac{2n}{2^M}\right) &= e^{(d)}_{-m}\left(\frac{2n_1+1}{2^M},\frac{2n_2+1}{2^M}\right)\\
&= \frac{1}{2}\left(e_{-m}\left(\frac{2n}{2^M}\right)+e_{-m}\left(\frac{2n_1+1}{2^M},\frac{2n_2+1}{2^M}\right)\right),\\
e^{(d)}_{-m}\left(\frac{2n_1}{2^M},\frac{2n_2+1}{2^M}\right) &= e^{(d)}_{-m}\left(\frac{2n_1+1}{2^M},\frac{2n_2}{2^M}\right)\\
&= \frac{1}{2}\left(e_{-m}\left(\frac{2n_1}{2^M},\frac{2n_2+1}{2^M}\right)+e_{-m}\left(\frac{2n_1+1}{2^M},\frac{2n_2}{2^M}\right)\right).
\end{aligned}
$$

The function $f_{+,-}$ is defined as

$$f_{+,-} = f_{++} + f_{-+} + f_{--} + f_{+-},$$

where

$$
\begin{aligned}
f_{++} &= f^\alpha = \alpha\,\mathbb{1}_{[0,2^N)\times[0,2^N)}, \quad \alpha = \frac{1}{2^{2(M+N)}}\sum_{n_1=0}^{2^{M+N}-1}\sum_{n_2=0}^{2^{M+N}-1} f_0\left(\frac{n_1}{2^M},\frac{n_2}{2^M}\right);\\
f_{-+} &= f^\beta = \beta\,\mathbb{1}_{[-2^N,0)\times[0,2^N)}, \quad \beta = \frac{1}{2^{2(M+N)}}\sum_{n_1=-2^{M+N}}^{-1}\sum_{n_2=0}^{2^{M+N}-1} f_0\left(\frac{n_1}{2^M},\frac{n_2}{2^M}\right);\\
f_{--} &= f^\gamma = \gamma\,\mathbb{1}_{[-2^N,0)\times[-2^N,0)}, \quad \gamma = \frac{1}{2^{2(M+N)}}\sum_{n_1=-2^{M+N}}^{-1}\sum_{n_2=-2^{M+N}}^{-1} f_0\left(\frac{n_1}{2^M},\frac{n_2}{2^M}\right);
\end{aligned}
$$

and

$$f_{+-} = f^\delta = \delta\,\mathbb{1}_{[0,2^N)\times[-2^N,0)}, \quad \delta = \frac{1}{2^{2(M+N)}}\sum_{n_1=0}^{2^{M+N}-1}\sum_{n_2=-2^{M+N}}^{-1} f_0\left(\frac{n_1}{2^M},\frac{n_2}{2^M}\right).$$

We construct sums s^α, s^β, s^γ, and s^δ to approximate f^α, f^β, f^γ, and f^δ, respectively. These sums can be constructed using either one of the functions $\psi^{(v)}$, $\psi^{(h)}$, or $\psi^{(d)}$ alone, squeezing the error to the left and right, top and bottom, or first and second diagonal of the support respectively. In this proof we use the vertical

function $\psi^{(v)}$. For s^α we proceed as follows. Define

$$\begin{aligned} f_1^\alpha = s_1^\alpha &= \alpha 2^{2N/p}\psi^{(v)}_{-N,(0,0)} = \alpha\psi^{(v)}(2^{-N}t), \\ f_2^\alpha &= 2\alpha 2^{2(N-1)/p}(\psi^{(v)}_{-N+1,(1,0)} + \psi^{(v)}_{-N+1,(1,1)}), \\ s_2^\alpha &= s_1^\alpha + f_2^\alpha, \\ f^\alpha - s_2^\alpha &= 4\alpha\, \mathbb{1}_{[2^N - 2^N/2^2, 2^N)\times[0,2^N)}. \end{aligned}$$

Next, define

$$\begin{aligned} f_3^\alpha &= 4\alpha 2^{2(N-2)/p}\sum_{i=0}^{3}\psi^{(v)}_{-N+2,(3,i)}, \\ s_3^\alpha &= s_2^\alpha + f_3^\alpha, \\ f^\alpha - s_3^\alpha &= 8\alpha\, \mathbb{1}_{[2^N - 2^N/2^3, 2^N)\times[0,2^N)}. \end{aligned}$$

In general, we have

$$\begin{aligned} f_k^\alpha &= 2^{k-1}\alpha 2^{2(N-k+1)/p}\sum_{i=0}^{2^{k-1}-1}\psi^{(v)}_{-N+k-1,(2^{k-1}-1,i)}, \\ s_k^\alpha &= s_{k-1}^\alpha + f_k^\alpha, \\ f^\alpha - s_k^\alpha &= 2^k\alpha\, \mathbb{1}_{[2^N - 2^N/2^k, 2^N)\times[0,2^N)}. \end{aligned}$$

Consequently, for all $p > 0$,

$$\|f^\alpha - s_k^\alpha\|_{L^p(\mathbb{R}^2)} = ((2^k\alpha)^p 2^{N-k}2^N)^{1/p} = \alpha 2^{2N/p}2^{k(1-1/p)},$$

and for all $0 < p < 1$,

$$\lim_{k\to\infty}\|f^\alpha - s_k^\alpha\|_{L^p(\mathbb{R}^2)} = 0.$$

Similarly, for s^β, we have

$$\begin{aligned} f_1^\beta = s_1^\beta &= -\beta 2^{2N/p} \psi^{(v)}_{-N,(-1,0)}, \\ f_k^\beta &= -2^{k-1}\beta 2^{2(N-k+1)/p} \sum_{i=0}^{2^{k-1}-1} \psi^{(v)}_{-N+k-1,(-2^{k-1},i)}, \\ s_k^\beta &= s_{k-1}^\beta + f_k^\beta, \\ f^\beta - s_k^\beta &= 2^k \beta \, \mathbb{1}_{[-2^N,-2^N+2^N/2^k)\times[0,2^N)}. \end{aligned}$$

Consequently, for all $p > 0$,

$$\|f^\beta - s_k^\beta\|_{L^p(\mathbb{R}^2)} = ((2^k\beta)^p 2^{N-k} 2^N)^{1/p} = \beta 2^{2N/p} 2^{k(1-1/p)},$$

and for all $0 < p < 1$,

$$\lim_{k\to\infty} \|f^\beta - s_k^\beta\|_{L^p(\mathbb{R}^2)} = 0.$$

Similarly, for s^γ, we have

$$\begin{aligned} f_1^\gamma = s_1^\gamma &= -\gamma 2^{2N/p} \psi^{(v)}_{-N,(-1,-1)}, \\ f_k^\gamma &= -2^{k-1}\gamma 2^{2(N-k+1)/p} \sum_{i=-2^{k-1}}^{-1} \psi^{(v)}_{-N+k-1,(-2^{k-1},i)}, \\ s_k^\gamma &= s_{k-1}^\gamma + f_k^\gamma, \\ f^\gamma - s_k^\gamma &= 2^k \gamma \, \mathbb{1}_{[-2^N,-2^N+2^N/2^k)\times[-2^N,0)}. \end{aligned}$$

Consequently, for all $p > 0$,

$$\|f^\gamma - s_k^\gamma\|_{L^p(\mathbb{R}^2)} = ((2^k\gamma)^p 2^{N-k} 2^N)^{1/p} = \gamma 2^{2N/p} 2^{k(1-1/p)},$$

and for all $0 < p < 1$,

$$\lim_{k\to\infty} \|f^\gamma - s_k^\gamma\|_{L^p(\mathbb{R}^2)} = 0.$$

Finally, for s^δ, we have

$$\begin{aligned} f_1^\delta = s_1^\delta &= \delta 2^{2N/p} \psi^{(v)}_{-N,(0,-1)}, \\ f_k^\delta &= 2^{k-1}\delta 2^{2(N-k+1)/p} \sum_{i=-2^{k-1}}^{-1} \psi^{(v)}_{-N+k-1,(2^{k-1}-1,i)}, \\ s_k^\delta &= s_{k-1}^\delta + f_k^\delta, \\ f^\delta - s_k^\delta &= 2^k \delta \, \mathbb{1}_{[2^N - 2^N/2^k, 2^N)\times[-2^N,0)}. \end{aligned}$$

Consequently, for all $p > 0$,

$$\|f^\delta - s_k^\delta\|_{L^p(\mathbb{R}^2)} = ((2^k\delta)^p 2^{N-k} 2^N)^{1/p} = \delta 2^{2N/p} 2^{k(1-1/p)},$$

and for all $0 < p < 1$,

$$\lim_{k\to\infty} \|f^\delta - s_k^\delta\|_{L^p(\mathbb{R}^2)} = 0.$$

Let $c_m = \max\{\alpha, \beta, \gamma, \delta\}$, we need

$$(2^k\alpha)^p 2^{N-k} 2^N + (2^k\beta)^p 2^{N-k} 2^N + (2^k\gamma)^p 2^{N-k} 2^N + (2^k\delta)^p 2^{N-k} 2^N < \left(\frac{\epsilon}{2}\right)^p,$$

this means

$$k > \frac{\log(\epsilon) - \log(2^{2+1/p} c_m 2^{2N/p})}{(1-1/p)\log(2)}.$$

Take

$$K = \left\lceil \frac{\log(\epsilon) - \log(2^{2+1/p} c_m 2^{2N/p})}{(1-1/p)\log(2)} \right\rceil,$$

and

$$f_{M,N,k} = s_k^\alpha + s_k^\beta + s_k^\gamma + s_k^\delta + s_{M,N},$$

therefore, for all $k > K$ and for all $0 < p < 1$ we have

$$\|f - f_{M,N,k}\|_p < \epsilon/2.$$

□

Theorem 3.3.14. *Given $N > 0$. Let $f \in L^p(\mathbb{R}^2)$, where $0 < p < 1$, and suppose f is a continuous function on $\mathbb{R}^2$, with supp $f \subseteq [-2^N, 2^N] \times [-2^N, 2^N]$. Then,*

$$\forall \epsilon > 0,\ \exists M = M(\epsilon) > 0 \text{ and } f_M, \quad \text{such that}$$

$$\forall n \in \{-2^{M+N}, \ldots, 2^{M+N} - 1\} \times \{-2^{M+N}, \ldots, 2^{M+N} - 1\},\ f_M \text{ is constant on } S_{M,n},$$

and

$$\|f - f_M\|_p < \epsilon/2.$$

Proof. Define f_M as

$$f_M(t) = \sum_{n_1=-2^{M+N}}^{2^{M+N}-1} \sum_{n_2=-2^{M+N}}^{2^{M+N}-1} f(t_n)\, \mathbb{1}_{S_{M,n}}, \quad \text{where } t_n \in S_{M,n}.$$

The sequence $\{f_M\}$, $M = 0, 1, 2, \ldots$ is uniformly convergent to f on $[-2^N, 2^N) \times [-2^N, 2^N)$. Thus, for $\epsilon > 0$ as in the statement, we take $\epsilon_0 = (\epsilon/2)/(2^{2(N+1)/p})$. Then, there is $M_0 > 0$ such that,

$$\forall M > M_0 \text{ and } \forall t \in [-2^N, 2^N) \times [-2^N, 2^N), \quad |f(t) - f_M(t)| < \epsilon_0.$$

Hence, for $M > M_0$ we have

$$\begin{aligned}
\|f - f_M\|_p &= \left(\int |f(t) - f_M(t)|^p \, dt \right)^{1/p} \\
&\leq \left(\int_{-2^N}^{2^N} \int_{-2^N}^{2^N} (\sup_{t \in [-2^N, 2^N) \times [-2^N, 2^N)} |f(t) - f_M(t)|)^p \, dt \right)^{1/p} \\
&< \left(\int_{-2^N}^{2^N} \int_{-2^N}^{2^N} (\epsilon_0)^p \, dt \right)^{1/p} \\
&= (2^{2(N+1)} \epsilon_0^p)^{1/p} \\
&= 2^{2(N+1)/p} \frac{\epsilon/2}{2^{2(N+1)/p}} \\
&= \epsilon/2.
\end{aligned}$$

□

Theorem 3.3.15. *Given $N > 0$. Let $f \in L^p(\mathbb{R}^2)$, where $0 < p < 1$, and suppose f is a continuous function on $\mathbb{R}^2$, with supp $f \subseteq [-2^N, 2^N] \times [-2^N, 2^N]$. Then for all $\epsilon > 0$, there is an $M = M(\epsilon)$, and there is a sequence of sums,*

$$f_{M,N,k} = \sum_{(i,j)\in S_{M,N,k}} a_{i,j}^{(v)}\psi_{i,j}^{(v)} + a_{i,j}^{(h)}\psi_{i,j}^{(h)} + a_{i,j}^{(d)}\psi_{i,j}^{(d)}, \quad a_{i,j}^{(v)}, a_{i,j}^{(h)}, a_{i,j}^{(d)} \in \mathbb{C},$$

indexed by $k \geq 1$, where $S_{M,N,k} \subseteq \mathbb{Z} \times \mathbb{Z}^2$ and card $S_{M,N,k} < \infty$; and these sums have the following properties:

$$\text{if } (i,j) \in S_{M,N,k} \text{ then } \textit{supp}\, \psi_{i,j}^{(v)}, \psi_{i,j}^{(h)}, \psi_{i,j}^{(d)} \subseteq \textit{supp}\, f,$$

and

$$\exists K = K(\epsilon) \text{ such that } \forall k > K,\ \|f - f_{M,N,K}\|_p < \epsilon.$$

Proof. Given $\epsilon > 0$. From Theorem 3.3.14 we know that there is an M and an f_M such that

$$\forall n \in \{-2^{M+N}, \ldots, 2^{M+N} - 1\} \times \{-2^{M+N}, \ldots, 2^{M+N} - 1\},\ f_M \text{ is constant on } S_{M,n},$$

and

$$\|f - f_M\|_p < \frac{\epsilon}{2^{1/p}}.$$

Therefore,

$$\rho(f, f_M) < \epsilon^p/2. \tag{3.3}$$

Additionally, from Theorem 3.3.13, we know that there is a $K = K(\epsilon)$ and a sequence of sums,

$$f_{M,N,k} = \sum_{(i,j)\in S_{M,N,k}} a_{i,j}^{(v)}\psi_{i,j}^{(v)} + a_{i,j}^{(h)}\psi_{i,j}^{(h)} + a_{i,j}^{(d)}\psi_{i,j}^{(d)}, \text{ where each } \quad a_{i,j}^{(v)}, a_{i,j}^{(h)}, a_{i,j}^{(d)} \in \mathbb{C},$$

such that

$$\forall k > K,\ \|f - f_{M,N,k}\|_p < \frac{\epsilon}{2^{1/p}}.$$

Therefore,

$$\forall k > K,\ \rho(f_M, f_{M,N,k}) < \epsilon^p/2. \tag{3.4}$$

From (3.3) and (3.4) we have

$$\rho(f, f_{M,N,k}) \leq \rho(f, f_M) + \rho(f_M, f_{M,N,k}) = \epsilon^p,$$

therefore,

$$\forall k > K,\ \|f - f_{M,N,k}\|_p < \epsilon.$$

□

Example 3.3.16. This numerical example leads to the constants α, β, γ, and δ as well as the functions $e^{(v)}_{-m}$, $e^{(h)}_{-m}$, and $e^{(d)}_{-m}$ in the proof of Theorem 3.3.13. For illustration purpose the function $f_0 = f$ is defined in $[-2, 2) \times [-2, 2)$, with $N = 1$ and $M = 0$. The function f_0 can be viewed as the following matrix:

$$f_0 = \begin{bmatrix} 0 & -2 & 3 & 2 \\ -2 & 8 & 5 & 2 \\ -3 & 4 & 5 & 2 \\ 7 & 0 & -1 & 2 \end{bmatrix},\ f_{-1} = \begin{bmatrix} 1 & 1 & 3 & 3 \\ 1 & 1 & 3 & 3 \\ 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \end{bmatrix},\ e_{-1} = \begin{bmatrix} -1 & -3 & 0 & -1 \\ -3 & 7 & 2 & -1 \\ -5 & 2 & 3 & 0 \\ 5 & -2 & -3 & 0 \end{bmatrix}.$$

Since $m = 1 \ldots M + N$, this is as far as we go and obtain $\alpha = 3$, $\beta = 1$, $\gamma = 2$, and

$\delta = 2$. Since in our example $f_{+,-} = f_{-1}$, for $m = 1$ we also have

$$e_{-1}^{(h)} = \begin{bmatrix} -2 & -2 & -1/2 & -1/2 \\ 2 & 2 & 1/2 & 1/2 \\ -3/2 & -3/2 & 3/2 & 3/2 \\ 3/2 & 3/2 & -3/2 & -3/2 \end{bmatrix},$$

$$e_{-1}^{(v)} = \begin{bmatrix} -2 & 2 & 1 & -1 \\ -2 & 2 & 1 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad e_{-1}^{(d)} = \begin{bmatrix} 3 & -3 & -1/2 & 1/2 \\ -3 & 3 & 1/2 & -1/2 \\ -7/2 & 7/2 & 3/2 & -3/2 \\ 7/2 & -7/2 & -3/2 & 3/2 \end{bmatrix}.$$

Example 3.3.17. Figure 3.7 shows a graphical representation of the construction of s^α used in the proof of Theorem 3.3.13.

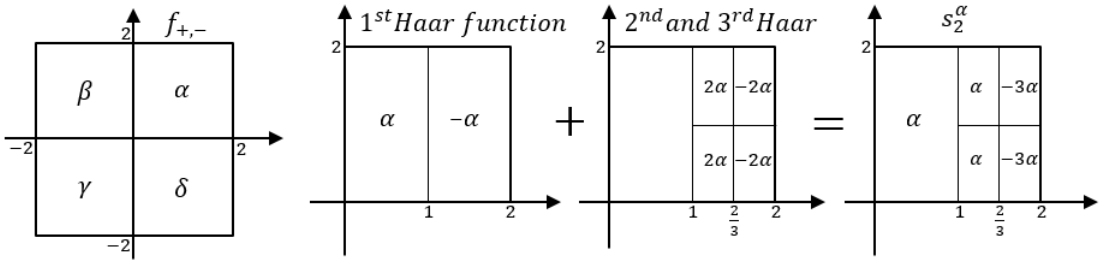

Figure 3.7: Construction of s_2^α using a sum of Haar functions. These are projection of 3-d plot onto 2-d planes, the numbers α, β, γ, δ, and their multiples represent the height of the function on the third dimension. From left to right, the first plot represents the function $f_{+,-}$, the second plot represents the function f_1^α, the third plot represents the function f_2^α (made up of two Haar functions added together), and the fourth plot represents the sum $s_2^\alpha = f_1^\alpha + f_2^\alpha$.

Example 3.3.18. Figure 3.8 shows an example of the sum approximating the com-

pactly supported normalized *sinc2 function*:

$$f(t_1, t_2) = \frac{\sin(\pi t_1)}{\pi t_1} \frac{\sin(\pi t_2)}{\pi t_2} \mathbb{1}_{[-2,2)\times[-2,2)}.$$

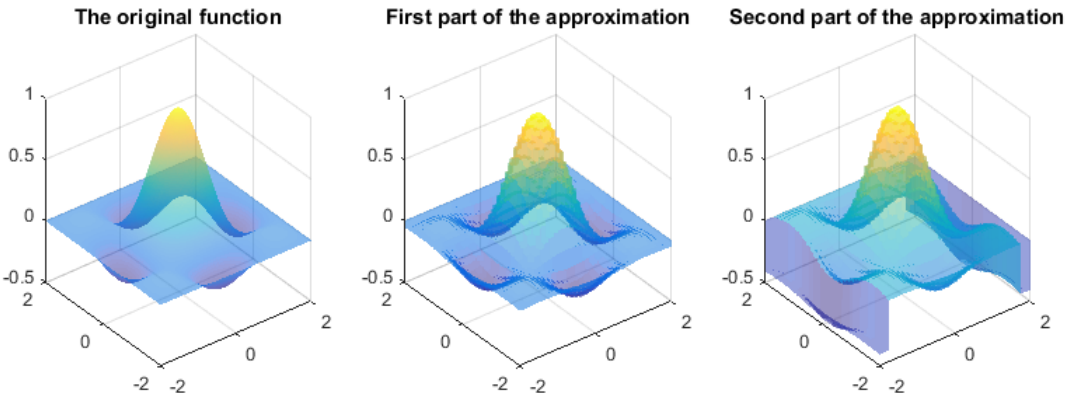

Figure 3.8: Approximation of the compactly supported sinc2 function with $N = 1$ and $M = 3$. From left to right, the first plot corresponds to the sinc2 function, the second plot corresponds to the triple sum representing $s_{M,N} = f_0 - f_{+,-}$, and the third plot corresponds to the final approximation $f_{3,1,3}$ of f.

Example 3.3.19. Figure 3.9 shows an example of the sum approximating the compactly supported *Gaussian2 function*:

$$f(t_1, t_2) = e^{-\pi(t_1^2+t_2^2)} \mathbb{1}_{[-2,2)\times[-2,2)}.$$

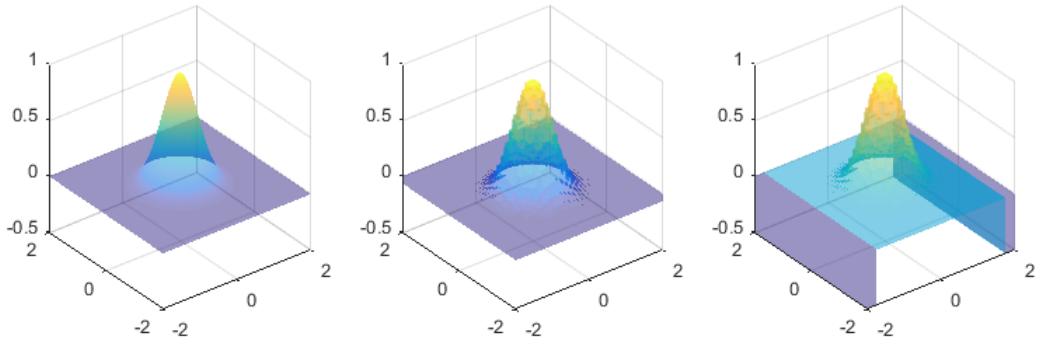

Figure 3.9: Approximation of the compactly supported Gaussian2 function with $N = 1$ and $M = 3$. From left to right, the first plot corresponds to the Gaussian2 function, the second plot corresponds to the triple sum representing $s_{M,N} = f_0 - f_{+,-}$, and the third plot corresponds to the final approximation $f_{3,1,3}$ of f.

Example 3.3.20. Figure 3.10 give an example of the sum approximating the compactly supported *Poisson2 function*:

$$f(t_1, t_2) = e^{-2\pi(|t_1|+|t_2|)} \mathbb{1}_{[-1,1)\times[-1,1)}.$$

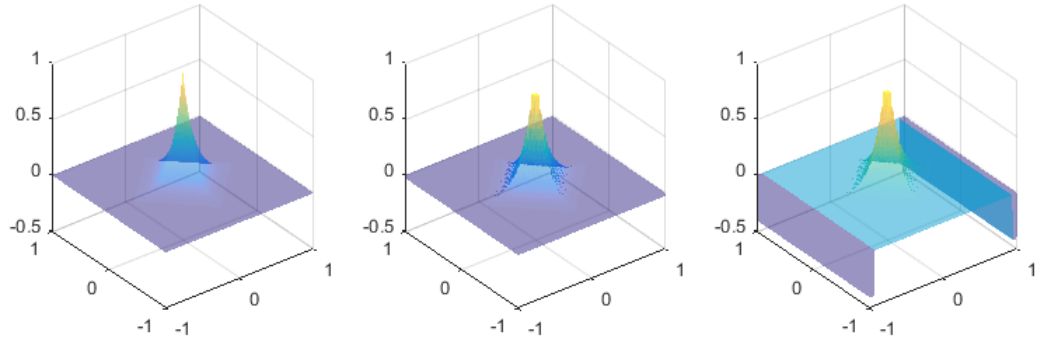

Figure 3.10: Approximation of the compactly supported Poisson2 function with $N = 0$ and $M = 5$. From left to right, the first plot corresponds to the Poisson2 function, the second plot corresponds to the triple sum representing $s_{M,N} = f_0 - f_{+,-}$, and the third plot corresponds to the final approximation $f_{5,0,5}$ of f.

Example 3.3.21. Figure 3.11 shows an example of the sum approximating the compactly supported *Fejér2 function*:

$$f(t_1, t_2) = \left(\frac{\sin(\pi t_1)}{\pi t_1} \frac{\sin(\pi t_2)}{\pi t_2} \right)^2 \mathbb{1}_{[-2,2)\times[-2,2)}.$$

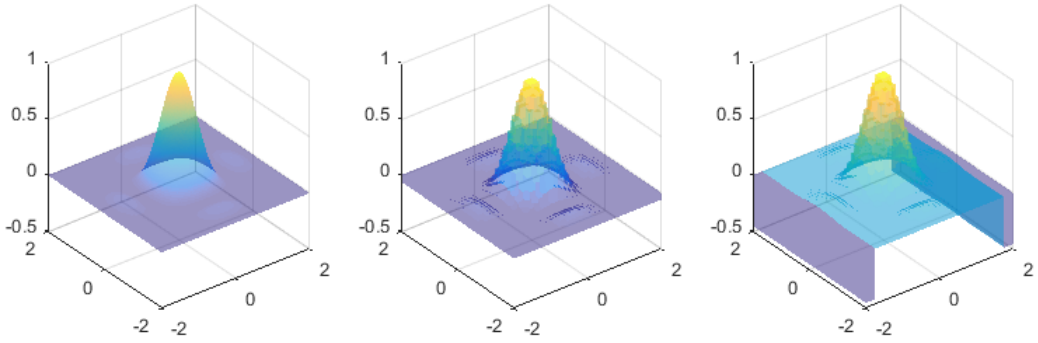

Figure 3.11: Approximation of the compactly supported Fejér2 function with $N = 1$ and $M = 3$. From left to right, the first plot corresponds to the Fejér2 function, the second plot corresponds to the triple sum representing $s_{M,N} = f_0 - f_{+,-}$, and the third plot corresponds to the final approximation $f_{3,1,3}$ of f.

3.4 Adaptive Haar approximation

Definition 3.4.1. Let $f \in L^p(\mathbb{R}, p > 0$ such that $\operatorname{supp} f = [A, B]$ with $A, B \in \mathbb{R}$. Given the Haar function ψ, the Adaptive Haar dyadic wavelet system for ψ as an element of $L^p(\mathbb{R})$ is the sequence $\{\widetilde{\psi}_{m,n} : (m, n) \in \mathbb{Z} \times \mathbb{Z}\}$, where each $\widetilde{\psi}_{m,n}$ is defined as

$$\widetilde{\psi}_{m,n}(t; s, c) = (2^m/p)^{1/p} \psi(2^m(t - s)/c - n) \quad \text{on } \mathbb{R}.$$

Where $s = (A + B)/2$ and $c = (B - A)/2$.

The support of $\widetilde{\psi}_{m,n}$ is $\overline{\widetilde{I}_{m,n}}$, with

$$\widetilde{I}_{m,n} = \left[\frac{cn}{2^m} + s, \frac{c(n+1)}{2^m} + s\right),$$

and

$$\|\widetilde{\psi}_{m,n}(t;s,c)\|_{L^p(\mathbb{R})} = \left(\int |\widetilde{\psi}_{m,n}(t;s,c)|^p \, dt\right)^{1/p} = \left((2^m/c)|\widetilde{I}_{m,n}|\right)^{1/p} = 1.$$

Theorem 3.4.2. *Let $f \in L^p(\mathbb{R})$, where $0 < p < 1$, and suppose $\operatorname{supp} f \subseteq [A,B]$ where $A, B \in \mathbb{R}$. Assume that there exists M such that for $n \in \{-2^M, \ldots, 2^M - 1\}$, f is constant on $\widetilde{I}_{M,n}$. Then, for all $\epsilon > 0$, there is a sequence of sums,*

$$f_{M,k} = \sum_{(i,j)\in S_{M,N,k}} \tilde{a}_{i,j}\widetilde{\psi}_{i,j}, \quad \tilde{a}_{i,j} \in \mathbb{C},$$

indexed by $k \geq 1$, where $S_{M,N,k} \subseteq \mathbb{Z} \times \mathbb{Z}$ and $\operatorname{card} S_{M,N,k} < \infty$ with the following properties:

$$\text{if } (i,j) \in S_{M,N,k} \text{ then } \operatorname{supp}\widetilde{\psi}_{i,j} \subseteq \operatorname{supp} f,$$

and

$$\exists K = K(\epsilon) \text{ such that } \forall k > K, \ \|f - f_{M,k}\|_p < \epsilon/2.$$

Proof. The proof follows directly from the proof of Theorem 3.3.1. With $N = 0$, let $s = (A+B)/2$ and $c = (B-A)/2$ as in Definition 3.4.1, we proceed as follows. As in Proof 3.3.1 we have:

$$\begin{aligned} f_0 &= f_{+,-} + \sum_{m=1}^{M} \sum_{n=-2^{M-m}}^{2^{M-m}-1} \tilde{a}_{M-m,n}\widetilde{\psi}_{M-m,n}, \\ f_0 &= f_{+,-} + s_M. \end{aligned}$$

.

With

$$\tilde{a}_{M-m,n} = c^{1/p} 2^{-(M-m)/p}\, e_{-m}\left(\frac{2n}{2^{M-m+1}}\right) \quad \text{for all } m \text{ and } n,$$

and

$$f_{+,-} = f_+ + f_-,$$

where

$$\begin{aligned} f_+ &= f^\alpha = \alpha\, \mathbb{1}_{[s,B)}, \quad \text{and} \quad \alpha = \frac{1}{2^M} \sum_{n=0}^{2^M-1} f_0\left(\frac{n}{2^M}\right); \\ f_- &= f^\beta = \beta\, \mathbb{1}_{[A,s)}, \quad \text{and} \quad \beta = \frac{1}{2^M} \sum_{n=-2^M}^{-1} f_0\left(\frac{n}{2^M}\right). \end{aligned}$$

We construct sums s^α, and s^β to approximate f^α, and f^β, respectively. For s^α, we proceed as follows. Define

$$\begin{aligned} f_1^\alpha = s_1^\alpha &= \alpha c^{1/p} \widetilde{\psi}_{0,0}, \\ f_k^\alpha &= 2^{k-1} \alpha c^{1/p} 2^{(-k+1)/p} \widetilde{\psi}_{k-1,2^{k-1}-1}, \\ s_k^\alpha &= s_{k-1}^\alpha + f_k^\alpha, \\ f^\alpha - s_k^\alpha &= 2^k \alpha\, \mathbb{1}_{[B-c/2^k,B)}. \end{aligned}$$

Consequently, for all $p > 0$,

$$\|f^\alpha - s_k^\alpha\|_{L^p(\mathbb{R})} = ((2^k\alpha)^p c 2^{-k})^{1/p} = \alpha c^{1/p} 2^{k(1-1/p)},$$

and for all $0 < p < 1$,

$$\lim_{k\to\infty} \|f^\alpha - s_k^\alpha\|_{L^p(\mathbb{R})} = 0.$$

Similarly, for s^β, we have

$$\begin{aligned} f_1^\beta = s_1^\beta &= -\beta c^{1/p} \widetilde{\psi}_{0,-1}, \\ f_k^\beta &= -2^{k-1} \beta c^{1/p} 2^{(-k+1)/p} \widetilde{\psi}_{k-1,-2^{k-1}}, \\ s_k^\beta &= s_{k-1}^\beta + f_k^\beta, \\ f^\beta - s_k^\beta &= 2^k \beta \, \mathbb{1}_{[A, A+c/2^k)}. \end{aligned}$$

Consequently, for all $p > 0$,

$$\|f^\beta - s_k^\beta\|_{L^p(\mathbb{R})} = ((2^k\beta)^p c 2^{-k})^{1/p} = \beta c^{1/p} 2^{k(1-1/p)},$$

and for all $0 < p < 1$,

$$\lim_{k\to\infty} \|f^\beta - s_k^\beta\|_{L^p(\mathbb{R})} = 0.$$

Let $c_m = \max\{\alpha, \beta\}$, we need

$$(2^k\alpha)^p c 2^{-k} + (2^k\beta)^p c 2^{-k} < \left(\frac{\epsilon}{2}\right)^p,$$

this means

$$k > \frac{\log(\epsilon) - \log(2^{1+1/p} c_m c^{1/p})}{(1-1/p)\log(2)}.$$

Let us take

$$K = \left\lceil \frac{\log(\epsilon) - \log(2^{1+1/p} c_m c^{1/p})}{(1-1/p)\log(2)} \right\rceil,$$

and

$$f_{M,k} = s_k^\alpha + s_k^\beta + s_M;$$

therefore, for all $k > K$ and for all $0 < p < 1$ we have

$$\|f - f_{M,k}\|_p < \epsilon/2.$$

□

Theorem 3.4.3. *Let $f \in L^p(\mathbb{R})$, where $0 < p < 1$, and suppose f is a continuous function on $\mathbb{R}$, with supp $f \subseteq [A, B]$. Then,*

$$\forall \epsilon > 0,\ \exists M = M(\epsilon) > 0 \text{ and } f_M, \quad \text{such that}$$

$$\forall n \in \{-2^M, \ldots, 2^M - 1\},\ f_M \text{ is constant on } \widetilde{I}_{M,n},$$

and

$$\|f - f_M\|_p < \epsilon/2.$$

Proof. Directly from the proof of Theorem 3.3.2, with $N = 0$, we define f_M as

$$f_M(t) = \sum_{n=-2^M}^{2^M-1} f(t_n)\, \mathbb{1}_{\widetilde{I}_{M,n}}, \quad \text{where } t_n \in \widetilde{I}_{M,n}.$$

Since f is defined on $[A, B)$, as in Proof 3.3.2, we take $\epsilon_0 = (\epsilon/2)/(2c^{1/p})$, and for $M > M_0$ we have

$$\|f - f_M\|_p < \epsilon/2.$$

□

Theorem 3.4.4. *Let $f \in L^p(\mathbb{R})$, where $0 < p < 1$ suppose f is a continuous function on $\mathbb{R}$, with supp $f \subseteq [A, B]$. Then, for all $\epsilon > 0$, there is an $M = M(\epsilon)$, and there is a sequence of sums,*

$$f_{M,k} = \sum_{(i,j) \in S_{M,k}} \tilde{a}_{i,j} \widetilde{\psi}_{i,j}, \quad \tilde{a}_{i,j} \in \mathbb{C},$$

indexed by $k \geq 1$, where $S_{M,k} \subseteq \mathbb{Z} \times \mathbb{Z}$ and card $S_{M,k} < \infty$, with the following properties:

$$\text{if } (i, j) \in S_{M,k} \text{ then } \textit{supp}\, \widetilde{\psi}_{i,j} \subseteq \textit{supp}\, f,$$

and

$$\exists K = K(\epsilon) \text{ such that } \forall k > K,\ \|f - f_{M,k}\|_p < \epsilon.$$

Proof. Directly following from the proof of Theorem 3.3.3, with $N = 0$, we use Theorem 3.4.3 and Theorem 3.4.2, and follow the step as in Proof 3.3.3. □

Example 3.4.5. Figure 3.12 shows an example of the sum approximating the compactly supported normalized *sinc function*:

$$f(t) = \frac{\sin(\pi t)}{\pi t} \mathbb{1}_{[0.5,2)}.$$

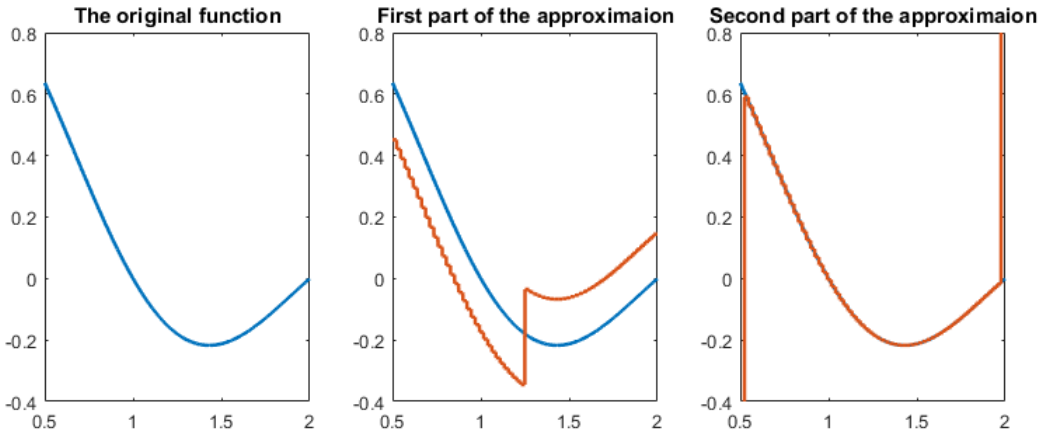

Figure 3.12: Approximation of the continuous compactly supported sinc function with support $[.5, 2]$ and $M = 5$, represented in blue in all three plot. From left to right, in the second plot the red graph (step function) corresponds to the double sum $s_M = f_0 - f_{+,-}$, and in the third plot the red graph (step function) corresponds to the final approximation of f.

Example 3.4.6. Figure 3.13 shows an example of the sum approximating the compactly supported normalized *sinc function* added to a linear function:

$$f(t) = t + \frac{\sin(\pi t)}{\pi t} \mathbb{1}_{[-2,4)}.$$

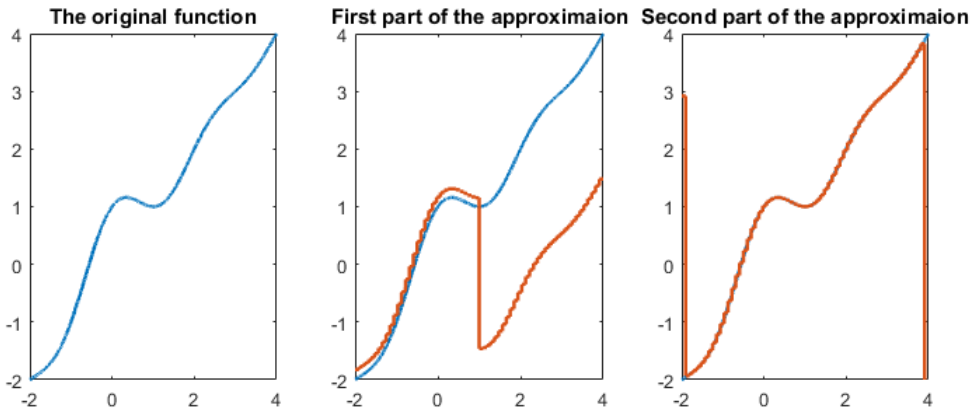

Figure 3.13: Approximation of the continuous compactly supported function $f(t)$ with support $[-2, 4]$ and $M = 5$, represented in blue in all three plot. From left to right, in the second plot the red graph (step function) corresponds to the double sum $s_M = f_0 - f_{+,-}$, and in the third plot the red graph (step function) corresponds to the final approximation of f.

Definition 3.4.7. Let $f \in L^p(\mathbb{R}^2,\ p > 0$ such that $\operatorname{supp} f = [A, B] \times [C, D]$ and $A, B, C, D \in \mathbb{R}$. Given the Haar functions $\psi^{(v)}$, $\psi^{(h)}$, and $\psi^{(d)}$, the adaptive Haar dyadic wavelet system for $\psi^{(v)}$, $\psi^{(h)}$, and $\psi^{(d)}$ as elements of $L^p(\mathbb{R}^2)$ is the sequence $\{\widetilde{\psi}^{(v)}_{m,n}, \widetilde{\psi}^{(h)}_{m,n}, \widetilde{\psi}^{(d)}_{m,n} : (m, n) \in \mathbb{Z} \times \mathbb{Z}^2\}$, where each $\widetilde{\psi}^{(v)}_{m,n}$, $\widetilde{\psi}^{(h)}_{m,n}$, and $\widetilde{\psi}^{(d)}_{m,n}$ are defined as

$$\begin{aligned}
\widetilde{\psi}^{(v)}_{m,n}(t; s, c) &= \left(\frac{2^{2m}}{c_1 c_2}\right)^{1/p} \psi^{(v)}\left(\frac{2^m(t - s_1)}{c_1} - n_1, \frac{2^m(t - s_2)}{c_2} - n_2\right), \\
\widetilde{\psi}^{(h)}_{m,n}(t; s, c) &= \left(\frac{2^{2m}}{c_1 c_2}\right)^{1/p} \psi^{(h)}\left(\frac{2^m(t - s_1)}{c_1} - n_1, \frac{2^m(t - s_2)}{c_2} - n_2\right), \\
\widetilde{\psi}^{(d)}_{m,n}(t; s, c) &= \left(\frac{2^{2m}}{c_1 c_2}\right)^{1/p} \psi^{(d)}\left(\frac{2^m(t - s_1)}{c_1} - n_1, \frac{2^m(t - s_2)}{c_2} - n_2\right).
\end{aligned}$$

Where $s = (s_1, s_2) = ((A + B)/2, (C + D)/2)$ and $c = (c_1, c_2) = ((B - A)/2, (D - C)/2)$.

The support of $\widetilde{\psi}^{(v)}_{m,n}$, $\widetilde{\psi}^{(h)}_{m,n}$, and $\widetilde{\psi}^{(d)}_{m,n}$ is $\overline{\widetilde{I}_{m,n}}$, with

$$\widetilde{I}_{m,n} = \left[\frac{c_1 n_1}{2^m} + s_1, \frac{c_1(n_1 + 1)}{2^m} + s_1\right) \times \left[\frac{c_2 n_2}{2^m} + s_2, \frac{c_2(n_2 + 1)}{2^m} + s_2\right),$$

and

$$\|\widetilde{\psi}^{(v)}_{m,n}(t;s,c)\|_{L^p(\mathbb{R}^2)} = \|\widetilde{\psi}^{(h)}_{m,n}(t;s,c)\|_{L^p(\mathbb{R}^2)} = \|\widetilde{\psi}^{(d)}_{m,n}(t;s,c)\|_{L^p(\mathbb{R}^2)} = 1.$$

Theorem 3.4.8. *Let $f \in L^p(\mathbb{R}^2)$, where $0 < p < 1$, and suppose $\operatorname{supp} f \subseteq [A,B] \times [C,D]$. Assume that there exists M such that for $n = (n_1, n_2) \in \{-2^M, \ldots, 2^M - 1\} \times \{-2^M, \ldots, 2^M - 1\}$, f is constant on $\widetilde{I}_{M,n}$. Then for all $\epsilon > 0$, there is a sequence of sums,*

$$f_{M,k} = \sum_{(i,j)\in S_{M,k}} \tilde{a}^{(v)}_{i,j}\widetilde{\psi}^{(v)}_{i,j} + \tilde{a}^{(h)}_{i,j}\widetilde{\psi}^{(h)}_{i,j} + \tilde{a}^{(d)}_{i,j}\widetilde{\psi}^{(d)}_{i,j}, \quad \tilde{a}^{(v)}_{i,j}, \tilde{a}^{(h)}_{i,j}, \tilde{a}^{(d)}_{i,j} \in \mathbb{C},$$

indexed by $k \geq 1$, where $S_{M,k} \subseteq \mathbb{Z} \times \mathbb{Z}^2$ and $\operatorname{card} S_{M,k} < \infty$; and these sums have the following properties:

$$\text{if } (i,j) \in S_{M,k} \text{ then } \operatorname{supp} \widetilde{\psi}^{(v)}_{i,j}, \widetilde{\psi}^{(h)}_{i,j}, \widetilde{\psi}^{(d)}_{i,j} \subseteq \operatorname{supp} f,$$

and

$$\exists K = K(\epsilon) \text{ such that } \forall k > K,\ \|f - f_{M,k}\|_p < \epsilon/2.$$

Proof. The proof follows directly from the proof of Theorem 3.3.13. With $N = 0$, let s and c as in Definition 3.4.7, we proceed as follows. As in Proof 3.3.13 we have:

$$f_0 = f_{+,-} + s_M,$$

where

$$s_M = \sum_{m=1}^{M} \sum_{n_1} \sum_{n_2} \tilde{a}^{(v)}_{M-m,n}\widetilde{\psi}^{(v)}_{M-m,n} + \tilde{a}^{(h)}_{M-m,n}\widetilde{\psi}^{(h)}_{M-m,n} + \tilde{a}^{(d)}_{M-m,n}\widetilde{\psi}^{(d)}_{M-m,n},$$

with

$$\begin{aligned}
\tilde{a}^{(v)}_{M-m,n} &= (c_1c_2)^{1/p}2^{-2(M-m)/p}\, e^{(v)}_{-m}\left(\frac{2n_1}{2^{M-m+1}}, \frac{2n_2}{2^{M-m+1}}\right),\\
\tilde{a}^{(h)}_{M-m,n} &= (c_1c_2)^{1/p}2^{-2(M-m)/p}\, e^{(h)}_{-m}\left(\frac{2n_1}{2^{M-m+1}}, \frac{2n_2}{2^{M-m+1}}\right),\\
\tilde{a}^{(d)}_{M-m,n} &= (c_1c_2)^{1/p}2^{-2(M-m)/p}\, e^{(d)}_{-m}\left(\frac{2n_1}{2^{M-m+1}}, \frac{2n_2}{2^{M-m+1}}\right),
\end{aligned}$$

for all m and $n = (n_1, n_2)$; and the function $f_{+,-}$ is defined as

$$f_{+,-} = f_{++} + f_{-+} + f_{--} + f_{+-},$$

where

$$\begin{aligned}
f_{++} &= f^{\alpha} = \alpha\, \mathbb{1}_{[s,B)\times[s,D)},\\
f_{-+} &= f^{\beta} = \beta\, \mathbb{1}_{[A,s)\times[s,D)},\\
f_{--} &= f^{\gamma} = \gamma\, \mathbb{1}_{[A,s)\times[C,s)},\\
f_{+-} &= f^{\delta} = \delta\, \mathbb{1}_{[s,B)\times[C,s)}.
\end{aligned}$$

We construct sums s^{α}, s^{β}, s^{γ}, and s^{δ} to approximate f^{α}, f^{β}, f^{γ}, and f^{δ}, respectively, using the vertical function $\psi^{(v)}$. For s^{α} we proceed as follows. Define

$$\begin{aligned}
f^{\alpha}_1 = s^{\alpha}_1 &= \alpha(c_1c_2)^{1/p}\widetilde{\psi}^{(v)}_{0,(0,0)},\\
f^{\alpha}_k &= 2^{k-1}\alpha(c_1c_2)^{1/p}2^{2(-k+1)/p}\sum_{i=0}^{2^{k-1}-1}\widetilde{\psi}^{(v)}_{k-1,(2^{k-1}-1,i)},\\
s^{\alpha}_k &= s^{\alpha}_{k-1} + f^{\alpha}_k,\\
f^{\alpha} - s^{\alpha}_k &= 2^k\alpha\, \mathbb{1}_{[B-c_1/2^k,B)\times[s_2,D)}.
\end{aligned}$$

Consequently, for all $p > 0$,

$$\|f^{\alpha} - s^{\alpha}_k\|_{L^p(\mathbb{R}^2)} = ((2^k\alpha)^p c_1 2^{-k} c_2)^{1/p} = \alpha(c_1c_2)^{1/p}2^{k(1-1/p)},$$

and for all $0 < p < 1$,

$$\lim_{k\to\infty} \|f^\alpha - s_k^\alpha\|_{L^p(\mathbb{R}^2)} = 0.$$

Similarly, for s^β, we have

$$\begin{aligned} f_1^\beta = s_1^\beta &= -\beta(c_1c_2)^{1/p}\widetilde{\psi}^{(v)}_{0,(-1,0)}, \\ f_k^\beta &= -2^{k-1}\beta(c_1c_2)^{1/p}2^{2(-k+1)/p}\sum_{i=0}^{2^{k-1}-1}\widetilde{\psi}^{(v)}_{k-1,(-2^{k-1},i)}, \\ s_k^\beta &= s_{k-1}^\beta + f_k^\beta, \\ f^\beta - s_k^\beta &= 2^k\beta\,\mathbb{1}_{[A,A+c_1/2^k)\times[s_2,D)}. \end{aligned}$$

Consequently, for all $p > 0$,

$$\|f^\beta - s_k^\beta\|_{L^p(\mathbb{R}^2)} = ((2^k\beta)^p c_1 2^{-k} c_2)^{1/p} = \beta(c_1c_2)^{1/p}2^{k(1-1/p)},$$

and for all $0 < p < 1$,

$$\lim_{k\to\infty} \|f^\beta - s_k^\beta\|_{L^p(\mathbb{R}^2)} = 0.$$

Similarly, for s^γ, we have

$$\begin{aligned} f_1^\gamma = s_1^\gamma &= -\gamma(c_1c_2)^{1/p}\widetilde{\psi}^{(v)}_{0,(-1,-1)}, \\ f_k^\gamma &= -2^{k-1}\gamma(c_1c_2)^{1/p}2^{2(-k+1)/p}\sum_{i=-2^{k-1}}^{-1}\widetilde{\psi}^{(v)}_{k-1,(-2^{k-1},i)}, \\ s_k^\gamma &= s_{k-1}^\gamma + f_k^\gamma, \\ f^\gamma - s_k^\gamma &= 2^k\gamma\,\mathbb{1}_{[A,A+c_1/2^k)\times[C,s_2)}. \end{aligned}$$

Consequently, for all $p > 0$,

$$\|f^\gamma - s_k^\gamma\|_{L^p(\mathbb{R}^2)} = ((2^k\gamma)^p (c_1c_2)^{1/p} 2^{-k})^{1/p} = \gamma(c_1c_2)^{1/p}2^{k(1-1/p)},$$

and for all $0 < p < 1$,

$$\lim_{k\to\infty} \|f^\gamma - s_k^\gamma\|_{L^p(\mathbb{R}^2)} = 0.$$

Finally, for s^δ, we have

$$\begin{aligned} f_1^\delta = s_1^\delta &= \delta(c_1c_2)^{1/p}\widetilde{\psi}^{(v)}_{0,(0,-1)}, \\ f_k^\delta &= 2^{k-1}\delta(c_1c_2)^{1/p}2^{2(-k+1)/p}\sum_{i=-2^{k-1}}^{-1}\widetilde{\psi}^{(v)}_{k-1,(2^{k-1}-1,i)}, \\ s_k^\delta &= s_{k-1}^\delta + f_k^\delta, \\ f^\delta - s_k^\delta &= 2^k\delta\,\mathbb{1}_{[B-c_1/2^k,B)\times[C,s_2)}. \end{aligned}$$

Consequently, for all $p > 0$,

$$\|f^\delta - s_k^\delta\|_{L^p(\mathbb{R}^2)} = ((2^k\delta)^p(c_1c_2)^{1/p}2^{-k})^{1/p} = \delta(c_1c_2)^{1/p}2^{k(1-1/p)},$$

and for all $0 < p < 1$,

$$\lim_{k\to\infty} \|f^\delta - s_k^\delta\|_{L^p(\mathbb{R}^2)} = 0.$$

Let $c_m = \max\{\alpha, \beta, \gamma, \delta\}$, we need

$$(2^k\alpha)^p c_1 2^{-k} c_2 + (2^k\beta)^p c_1 2^{-k} c_2 + (2^k\gamma)^p c_1 2^{-k} c_2 + (2^k\delta)^p c_1 2^{-k} c_2 < \left(\frac{\epsilon}{2}\right)^p,$$

this means

$$k > \frac{\log(\epsilon) - \log(2^{2+1/p}c_m(c_1c_2)^{1/p})}{(1-1/p)\log(2)}.$$

Let us take

$$K = \left\lceil \frac{\log(\epsilon) - \log(2^{2+1/p}c_m(c_1c_2)^{1/p})}{(1-1/p)\log(2)} \right\rceil,$$

and

$$f_{M,k} = s_k^\alpha + s_k^\beta + s_k^\gamma + s_k^\delta + s_M,$$

therefore, for all $k > K$ and for all $0 < p < 1$ we have

$$\|f - f_{M,k}\|_p < \epsilon/2.$$

□

Theorem 3.4.9. *Let $f \in L^p(\mathbb{R}^2)$, where $0 < p < 1$, and suppose f is a continuous function on $\mathbb{R}^2$, with supp $f \subseteq [A, B] \times [C, D]$. Then,*

$$\forall \epsilon > 0,\ \exists M = M(\epsilon) > 0 \text{ and } f_M, \quad \text{such that}$$

$$\forall n \in \{-2^M, \ldots, 2^M - 1\} \times \{-2^M, \ldots, 2^M - 1\},\ f_M \text{ is constant on } \widetilde{I}_{M,n},$$

and

$$\|f - f_M\|_p < \epsilon/2.$$

Proof. Directly following from the proof of Theorem 3.3.14, with $N = 0$, we define f_M as

$$f_M(t) = \sum_{n_1=-2^M}^{2^M-1} \sum_{n_2=-2^M}^{2^M-1} f(t_n)\, \mathbb{1}_{\widetilde{I}_{M,n}}, \quad \text{where } t_n \in \widetilde{I}_{M,n}.$$

Since f is defined on $[A, B)\times[C, D)$, as in Proof 3.3.14, we take $\epsilon_0 = (\epsilon/2)/(4(c_1c_2)^{1/p})$, and for $M > M_0$ we have

$$\|f - f_M\|_p < \epsilon/2.$$

□

Theorem 3.4.10. *Let $f \in L^p(\mathbb{R}^2)$, where $0 < p < 1$, and suppose f is a continuous function on $\mathbb{R}^2$, with supp $f \subseteq [A, B] \times [C, D]$. Then for all $\epsilon > 0$, there is an $M = M(\epsilon)$, and there is a sequence of sums,*

$$f_{M,k} = \sum_{(i,j)\in S_{M,k}} \tilde{a}^{(v)}_{i,j} \widetilde{\psi}^{(v)}_{i,j} + \tilde{a}^{(h)}_{i,j} \widetilde{\psi}^{(h)}_{i,j} + \tilde{a}^{(d)}_{i,j} \widetilde{\psi}^{(d)}_{i,j}, \quad \tilde{a}^{(v)}_{i,j}, \tilde{a}^{(h)}_{i,j}, \tilde{a}^{(d)}_{i,j} \in \mathbb{C},$$

indexed by $k \geq 1$, *where* $S_{M,k} \subseteq \mathbb{Z} \times \mathbb{Z}^2$ *and* $\operatorname{card} S_{M,k} < \infty$; *and these sums have the following properties:*

$$\text{if } (i,j) \in S_{M,k} \text{ then } \operatorname{supp} \widetilde{\psi}_{i,j}^{(v)}, \widetilde{\psi}_{i,j}^{(h)}, \widetilde{\psi}_{i,j}^{(d)} \subseteq \operatorname{supp} f,$$

and

$$\exists K = K(\epsilon) \text{ such that } \forall k > K, \ \|f - f_{M,K}\|_p < \epsilon.$$

Proof. Directly following from the proof of Theorem 3.3.15, with $N = 0$, we use Theorem 3.4.9 and Theorem 3.4.8, and follow the step as in Proof 3.3.15. □

Example 3.4.11. Figure 3.14 shows an example of the sum approximating the compactly supported normalized *sinc2 function*:

$$f(t_1, t_2) = \frac{\sin(\pi t_1)}{\pi t_1} \frac{\sin(\pi t_2)}{\pi t_2} \mathbb{1}_{[-2,0)\times[-1,1)}.$$

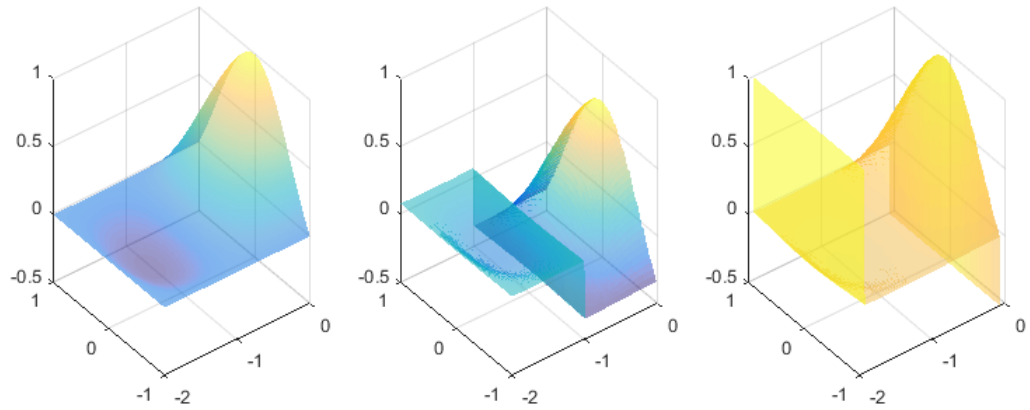

Figure 3.14: Approximation of the compactly supported sinc2 function with support $[-2, 0] \times [-1, 1]$ and $M = 5$. From left to right, the first plot corresponds to the sinc2 function, the second plot corresponds to the triple sum representing $s_M = f_0 - f_{+,-}$, and the third plot corresponds to the final approximation of f.

Example 3.4.12. Figure 3.15 shows an example of the sum approximating the compactly supported mixture of *Gaussian function*:

$$f(t_1, t_2) = e^{-\pi(t_1^2+t_2^2)} + e^{-\pi(t_1^2+(t_2+2)^2)} \mathbb{1}_{[-2,2)\times[-3,1)}.$$

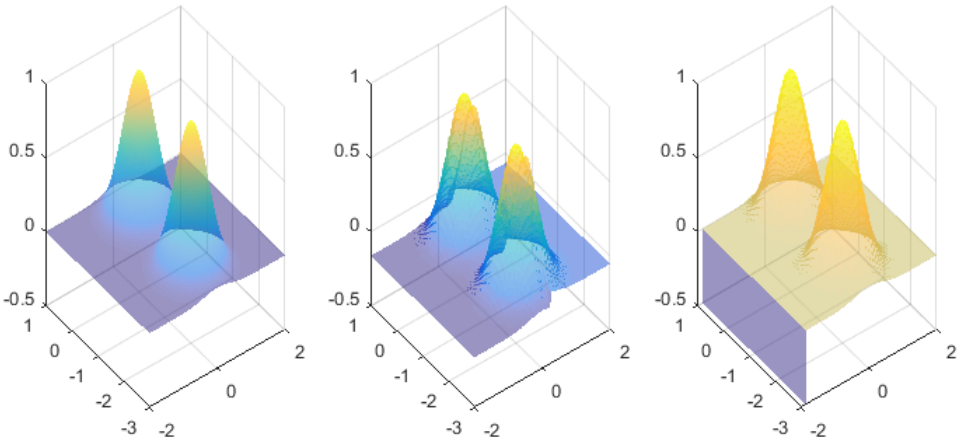

Figure 3.15: Approximation of the compactly supported function $f(t_1, t_2)$ with $M = 5$. From left to right, the first plot corresponds to the mixture of Gaussian function, the second plot corresponds to the triple sum representing $s_M = f_0 - f_{+,-}$, and the third plot corresponds to the final approximation.

3.5 Conclusion

In this chapter, we demonstrated the construction of a Haar approximation from within for compactly supported function $f \in L^p(\mathbb{R}^d)$, $0 < p < 1$ and $d = 1, 2$. In particular, we were able to show that we can construct approximants with support contained in the support of f. This important feature has not been shown to be possible in the space of L^2 functions. Our main results are given in Theorems 3.3.3, 3.3.15, 3.4.4, and 3.4.10.

Chapter 4: On the classification of multiton enhancers

4.1 Introduction

The work in this chapter is in collaboration with Dr. Ivan Ovcharenko and his research group at the National Center for Biotechnology Information (NCBI) within the National Library of Medicine (NLM), which is a branch of the National Institutes of Health (NIH).

The project goal is to identify the signature patterns of enhancers which are particular deoxyribonucleic acid (DNA) segments containing multiple transcription factor binding sites that increase or enhance the likelihood of gene expression. The research group extracted the distinct DNA sequence encryption of singleton enhancers which are enhancers without any other tissue-specific enhancers in the same locus; these are different from redundant enhancers and multiple enhancers of the same locus. Redundant enhancers are a source of evolutionary novelty [45] and have been celebrated among scientists for its breakthrough in preventing the loss of genes' function due to mutation [35, 73, 79]. The extracted encryptions allow for the development of an accurate sequence classifier and can identify a subset of redundant enhancers with similar sequence signatures to singleton enhancers called singleton-like multiton enhancers. Enhancers from DNA sequences have been pre-

dicted using methods such as support vector machine (SVM) and deep learning algorithms demonstrated in experiments by D. Lee et al., B. Zhu et al., and B. Alipanahi et al. [2,60,106]. We ascertained that these singleton-like multiton enhancers had properties similar to that of singleton enhancers and would act as primary activators of gene transcription by establishing connections with distal enhancers. To analyze the differences and similarities between singleton-like multiton enhancers and singleton enhancers, we had to determine a way to distinguish the singleton-like multiton enhancers given a set of redundant enhancers. The following sections describe our methodology and results.

4.2 Dataset and strategy

The data available has several lists of enhancers (both singleton and redundant) that were extracted for and organized by various tissues such as the heart, liver, brain, etc. These lists of enhancers were extracted and described by Dr. Irina Hashmi; these lists may be obtained with the permission of Dr. Ivan Ovcharenko and his research group at NCBI. With access to the entire human reference genome 19 (hg19), made available from the public database Roadmap MNEMONICS bed files [58], we were able to extract random DNA sequences to treat as a control. Control sets were pulled from the entire hg19 due to the lack of prior examples of singleton-like multiton enhancers to use as the training set for the classification.

For a given tissue, the plan was to build a classifier by training it over a combination set of tissue-specific singleton enhancers (positive set) and a set of

randomly selected aforementioned DNA sequences (negative set). The resulting classifier was then applied to the corresponding set of redundant enhancers with the expectation that the set will be split into two groups due to their significant difference in genomic sequence. The two groups would be those that identify as singleton enhancers (singleton-like multiton enhancers) and the rest (non-singleton-like multiton enhancers).

To determine the classifiers performance in identifying singleton enhancers we paid attention to two measures: the receiver operating characteristics area under the curve (auROC) also known as the receiver operating characteristic (ROC) [67, 107] and the accuracy of the classification in both specificity (true negative rate) and sensitivity (true positive rate). There are three important choices to make when classifying data: control set, feature extraction method, classification method. Moving forward, we will present our considerations and final choices.

4.3 Control

Given there are no prior results or examples of singleton-like enhancers, our chosen control set used to build the singleton-vs-rest (SVR) classifier determined the final results, interpretations, and conclusion. With an inaccurate control set, our findings could lack objectivity; therefore, it was ideal that a good balance between bias and variance within the selected DNA sequences.

4.3.1 Random sequences

From a mathematical standpoint, a control set would consist of randomly selected DNA sequences from the entire hg19 with a few fair requirements. First, the control set should have the same number of DNA sequences as the corresponding set of singleton enhancers. Second, the randomly selected sequences should have the same length as the sequences of singleton enhancers used as the positive set. The issue with this type of control set is that the resulting SVR does not quite fit into the purpose of this project. With the goal to use the SVR to partition a given set of redundant enhancers between singleton-like multiton enhancers and the non-singleton-like multiton enhancers, both the negative class and positive class should have properties attributed to enhancers. These considerations helped determine our second candidate for the control set.

4.3.2 Guanine/Cytosine (GC) content

The GC content is the ratio of the number of G and C over the total number of G, C, Adenine (A), and Thymine (T) in a DNA sequence. The GC content of enhancers is usually lower than the average in the genome [40, 54] except for transcribed enhancers. After considering this difference, the control set is selected randomly with the same aforementioned requirements with the addition of the same GC content of the sequences in both the positive class and negative class. A limitation of this type of control is that it does not allow for much variation between the two sets, and the results appeared nearly randomized. Also, it is important to

note that keeping the GC content the same, keeps the AC content the same as well as the repeat content (the number of missing nucleotides in a DNA sequence over the length of the sequence).

4.3.3 Repeat content

For this choice of control set, we kept the length of the randomly selected sequences the same as those in the positive class. Furthermore, we kept the repeat content the same, while relaxing the condition on the GC content (or equivalently the AC content) which reduces the bias while increasing variance within the control set.

4.4 Feature extraction

Given a biological tissue, the positive class (the corresponding set of singletons) and the negative class (the corresponding set of controls) was selected and the next phase was to decide which set of features to extract which will allow for the transformation of a given DNA sequence into a numerical vector that a computer can understand and process for the best classification results.

The following four feature extraction techniques are thoroughly explained in the KeBABS software manual [77]. As a brief summary of terms:

- **Spectrum**: The feature space consists of all sub-sequence of length k (consecutive k nucleotides known as k-mers). During the extraction we considered reverse complements to be identical to their forward counterpart.

- **Mismatch**: With parameters k and m the feature space still consists of all the k-mers; with m representing the maximum number of mismatches allowed for each k-mer.

- **Gappy pair**: With parameter k and m the feature space consists of a pair of k-mers separated by a maximum of m arbitrary nucleotides in the sequence.

- **Motif**: The feature space consists of a pre-defined list of these features.

After experimenting with these four methods, the Spectrum and Motif list of features performed better on average than the Mismatch and the Gappy pair list of features on the data set. Mismatch and Gappy pair have an advantage over the other two methods when the DNA sequences in the data set have been genetically transformed. On one hand, with the Mismatch, some of the nucleotides were genetically changed and on the other hand, with the Gappy pair some nucleotides were introduced into the original sequences. The Motif feature space is suitable for our purpose because it constitutes a pre-defined list of features, called transcription factors, built within a biological framework. Therefore, the Motif feature space provided the most meaningful conclusions based on the classification results. The Spectrum feature space is a natural choice and gives the best classification results over the other methods with $k = 1, 2, 5, 6$ or 7. With low values of k, the interpretation of the classification results were close to meaningless as the list of features was not yet significant. Fortunately, with a high value for k (i.e. $k = 6$) the list of features is rich enough and can be used to deduce meaningful interpretations. Using the Spectrum feature space, an algorithm exists to map a list of k-mers with the

value of k high enough to corresponding lists of transcription factors. The latter choice gave the ideal balance between classification accuracy and interpretability.

4.5 Classification and techniques

In machine learning, it is imperative to use an appropriate classification method. This section provides insights on our method of choice.

4.5.1 Multi-layer vs single-layer model

For a given biological tissue, the average number of singleton enhancers (positive class) available is in the order of hundreds; and, using k-mers to extract the feature vector (with $k = 6$) results in numerical vectors of dimension equal to $m = 2048$. In general, multi-layer methods such as neural networks and deep learning require a very large amount of training data (in the order of ten thousands in this case) to produce significant results [51, 80]. In this case, it was not advantageous to use multi-layer methods; therefore, single-layer methods were chosen.

4.5.2 Classification algorithm

After experimenting with a few single-layer classification methods, e.g., classification trees, k-nearest neighbors, SVM, and regularized least square regression (Lasso), we found that SVM and Lasso produced the best classification results which became our methods of focus. After a series of tests, the classification algorithm of choice became SVM, for two reasons: SVM performs better than Lasso on average

and Lasso is a regularized method equipped with a parameter search algorithm thus making it time consuming compared to SVM when constructing new models.

4.5.3 Regularization methods

After selecting a classification algorithm for our dataset in which there is an average of hundreds of data samples in each dataset and dimension exceeding the thousand, the next step is to improve the results by exploring some regularization methods. Two regularization methods were examined: feature reduction (feature selection) and feature scaling. There is a clear distinction between the two methods as feature reduction assumes only a few features are significantly contributing to the classification results and attempts to find those features. Examples of feature reduction algorithm include: principal component analysis (PCA), sequential feature selection, and enrichment (most expressed features). On the other hand, feature scaling assumes that all the features are important, although some are more highly expressed. The goal is to scale all the features accordingly so that no feature is at a disadvantage thus given each feature a chance to influence the outcome. The best improvement of the results used feature scaling combined with SVM which also sped up convergence.

4.6 Cardiac tissue results

In this section we describe select results that were obtained using the data corresponding to cardiac tissue (E105) in the heart. The positive class corresponds

to the set of singleton enhancers and the negative class (control) consists of randomly selected DNA sequences with controlled (or matching) repeat content. Since the control set is selected at random, the results shown here are the average over 20 repetitions or various set of controls. Table 4.1 and Table 4.2 show the same type of results using Motifs and Spectrum (6-mers) respectively for feature extraction.

E105 (Heart)	AUC tr	AUC ts	Acc	Spe	Sen
SVM (Motifs)	1	0.8103	72.78 %	72.24 %	73.33 %
PCA(20) + SVM (Motifs)	0.8647	0.8634	65.22 %	70.61 %	59.83 %
Scale + SVM (Motifs)	0.8785	0.8729	60.05 %	84.86 %	35.23 %

Table 4.1: Classification results with Motifs as the feature extraction method. The first row shows results using SVM without regularization. The second row shows results using SVM with feature reduction, PCA is used to reduce the dimension from $m = 2048$ to 20. The third row shows results using SVM with feature scaling. The following abbreviations are used in the table: tr (training), ts (testing), Acc (accuracy), Spe (specificity), Sen (sensitivity).

E105 (Heart)	AUC tr	AUC ts	Acc	Spe	Sen
SVM (6-mers)	1	0.8438	76.97 %	78.10 %	75.85 %
PCA(20) + SVM (6-mers)	0.8981	0.8786	80.51 %	85.58 %	75.44 %
Scale + SVM (6-mers)	0.9025	0.8934	80.41 %	79.12 %	81.70 %

Table 4.2: Classification results with 6-mers as the feature extraction method. The first row shows results using SVM without regularization. The second row shows results using SVM with feature reduction, PCA is used to reduce the dimension from $m = 2048$ to 20. The third row shows results using SVM with feature scaling. The following abbreviations are used in the table: tr (training), ts (testing), Acc (accuracy), Spe (specificity), Sen (sensitivity).

Within both tables of data, the first rows show the results obtained when applying SVM directly, the second rows pre- process the data using PCA with 20 principal components before pushing the data through SVM, and the third rows pre-process the data using scaling before pushing the data through SVM.

The need to pre-process the data comes from our first row of results (the first rows in Table 4.1 and Table 4.2). When looking at the ROCs area under the curve for the training and the testing data, it is clear that the SVM algorithm is overfitting the data as there is a big gap between the two numbers. In terms of feature extraction methods, two observations can be made: first, although the overfitting issue is resolved, accuracy suffers when using Motifs (shown in Table 4.1) to extract features compare to Spectrum (shown in Table 4.2). Second, the difference between specificity and sensitivity when using Motifs indicate that one of the classes

is highly favored over the other. Therefore, Spectrum was used as the feature extraction method of choice.

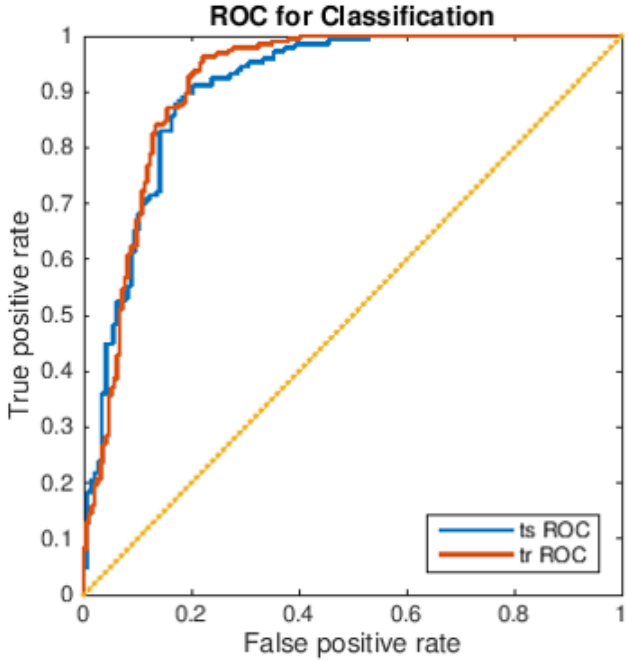

Figure 4.1: ROC for training and testing for the heart tissue, using scaling then SVM (6-mers).

4.7 Conclusion

In this chapter, we constructed an optimal classifier to identify the signature patterns of enhancers in order to analyze the differences and similarities between singleton-like multiton enhancer and singleton enhancers. After applying all the following factors: extracting the features using 6-mers, pre-processing the data using scaling, and using SVM as the classification algorithm of choice, Figure 4.1 shows how close the ROC for the testing data (in blue) compared to the ROC for the training data (in red). These results are an indication of a reached optimal classification.

Chapter 5: Analysis of the T_2-store-T_2 magnetic resonance relaxometry experiment with N exchanging sites

5.1 Introduction

The work in this chapter is in collaboration with Dr. Richard G. Spencer and his research group, the Magnetic Resonance Imaging and Spectroscopy Section at the National Institute on Aging (NIA), a branch of the National Institutes of Health (NIH).

Magnetic resonance imaging (MRI) is a tool that has been increasingly used as a sensitive, noninvasive imaging method for diagnosing anatomy and pathology, including osteoarthritis. In nuclear magnetic resonance (NMR) relaxation experiments, the relaxation time, T_1, is often referred to as spin-lattice relaxation time, and it describes the regrowth of net sample magnetization along the axis defined by the main magnetic field; this regrowth occurs between radio-frequency pulses and moves towards the thermal equilibrium value given by Curies law. The relaxation time, T_2, is often referred to as spin-spin relaxation time or transverse relaxation time, and it defines the rate of magnetization loss perpendicular to the main magnetic field as a function of time following an excitation radio-frequency pulse [28,36].

This decay occurs due to spin dephasing, which results from both random motion and small susceptibility differences within the material under investigation. T_2 is often used in biomedical magnetic resonance as a marker of anatomical pathology, such as early degeneration in articular cartilage [70, 82].

Methods have been developed to improve the analysis of relaxation times including multi-exponential transverse relaxation analysis [49, 81, 83], multivariate analysis [81], and support vector regression [49]. Methods for measuring relaxation time have also been derived using the effect of exchange on saturation factors [46,47] and magnetizations using the one pulse sequence [89]. These methods have been further optimized through prior knowledge of the parameter range [55]. Additionally, Galban and Spencer [37] used progressive saturation to measure relaxation time and chemical exchange rates in multiple sites.

Two-dimensional relaxometry experiments define correlations between relaxation times. A T_2-store-T_2 experiment consists of a period of T_2 sensitization, followed by a period of longitudinal magnetization storage, and a final period of T_2-sensitive signal acquisition. The first detailed 2-compartment analysis of a T_2-store-T_2 experiment was used to analyze exchange of water between two compartments [69]. This experiment can be used to demonstrate the existence of exchange between water compartments, as well as the rate of exchange. Note that in this context, compartments and sites refer to different water environments that may spatially coincide, but exhibit distinct T_1 or T_2 relaxation behavior. This may be due to loose binding with macromolecules. Washburn and Callaghan [101] considered up to four sites when observing the movement of water over time between pores

or sites, differing in sizes in Castlegate sandstone.

The goal of our study was to numerically implement equations (3)-(14) developed by Monteilhet et al. [69] by developing the system of coupled differential equations driving the relaxation and exchange of the magnetization in the sites. After verifying our implementation, we developed an extension of the results to N sites, where $N \geq 2$. All simulations were performed in *MATLAB*.

5.2 2 sites experiment and notation

The T_2-store-T_2 experiment consists of three time-intervals. The first interval is a pulse train consisting of a 90-degree excitation pulse followed by a train of echo-producing 180-degree pulses. At the end of the sequence, a 90-degree flip-up pulse is applied to store the magnetization along the z-axis for the duration of the second interval. During this storage interval, magnetization is exchanged between water sites; the effects of exchange are greater for longer storage times. Thereafter, the longitudinal magnetization is re-converted to transverse magnetization with an additional 90-degree pulse. During the third period, a second pulse train consisting of n echo pulses following this excitation pulse is applied, with the series of echo intensities recorded. The following notation is used in our analysis:

- $M^{(i)}$ represents the magnetization at site i.
- $M_{eq}^{(i)}$ represents the equilibrium magnetization at site i.
- k_{ij} represents the exchange rate from site i to site j.

- $T_1^{(i)}$ and $T_2^{(i)}$ represent the relaxation times at site i.
- $R^{(i)} = 1/T^{(i)}$ represents the relaxation rate at site i.
- t_1, t_2, and t_3 represent the duration of the first T_2 period, storage period, and second T_2 period respectively.
- N represents the number of sites.

Given two sites, "1" and "2", the system of coupled differential equations governing the relaxation of the magnetization at the sites are well known [66], viz.,

$$\begin{cases} \frac{dM^{(1)}}{dt} = -k_{12}M^{(1)} + k_{21}M^{(2)} + R^{(1)}(M_{eq}^{(1)} - M^{(1)}), \\ \frac{dM^{(2)}}{dt} = -k_{21}M^{(2)} + k_{12}M^{(1)} + R^{(2)}(M_{eq}^{(2)} - M^{(2)}), \end{cases} \tag{5.1}$$

and the detailed balance equation for this system is given by:

$$k_{12}M_{eq}^{(1)} = k_{21}M_{eq}^{(2)}. \tag{5.2}$$

An implementation of the detailed solution to system (5.1) provided in Monteilhet's paper [69], viz., equations (3)-(14), was done using the following parameters:

- $M_{eq}^{(1)} = M_{eq}^{(2)} = 0.5$.
- $T_2^{(1)} = 0.25\ ms$, $T_2^{(2)} = 2.5\ ms$, and $T_1^{(1)} = T_1^{(2)} = 1\ s$.
- $t_1 = t_3 = logspace(-1, 1, 128)\ ms$, i.e., 128 logarithmically spaced values from 10^{-1} to 10^{1}, and $t_2 = 0.2, 1.0$ and $5.0\ ms$.
- $k = k_{12} = k_{21}$ with $k = 0.2, 1.0$ and $5.0\ ms^{-1}$.

The implementation of the solution to system (5.1) provided by Monteilhet yields the results shown here in Figure 5.1.

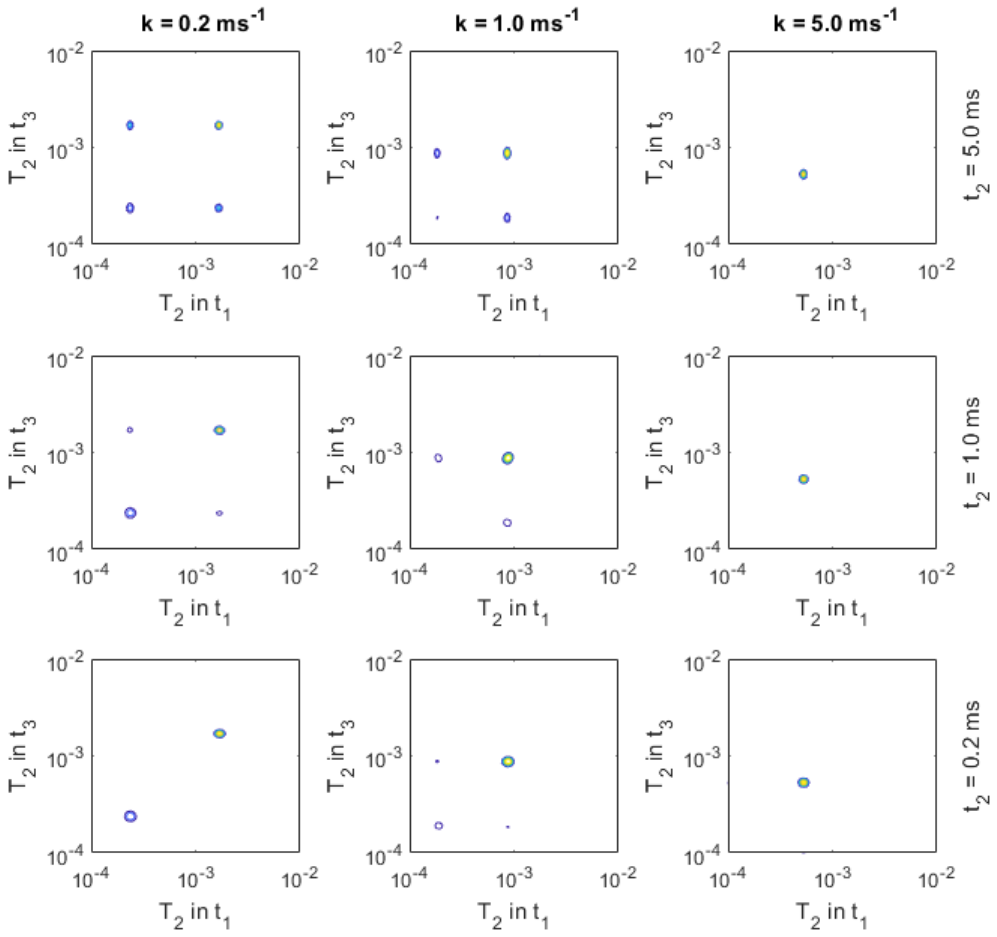

Figure 5.1: Calculated two sites T_2-store-T_2 spectra with increasing exchange rate $k = 0.2, 1.0$ and $5.0\ ms^{-1}$ (left to right) and increasing storing time $t_2 = 0.2, 1.0$ and $5.0\ ms$ (bottom to top.)

The plots in Figure 5.1 represent the observed two-dimensional magnetization signal at the end of the experiment, and the axes correspond to relaxation times. The peaks are artificially broadened using a blurring technique to make them visible. A contour plot of the signal is shown in each plot for various storing times t_2

and exchange rates k. The diagonal peaks represent the T_2 relaxation time of the underlying components at the sites. The off-diagonal peaks arise due to ongoing exchange between these components; therefore, their positions are defined by the two diagonal peaks corresponding to the sites involved in the exchange. Note that as the storage time increases so does the amplitude of the off-diagonal peaks, which implies a greater exchange between the sites. The rate at which the exchange happens depends on the parameter k; therefore, we ascertain that for slower exchange rate, k, molecules are not entirely transferred from one site to the other, however, for faster exchange rate we observe a complete transfer of molecules from one site to the other. In either case, the observed T_2 value resolve at an intermediate value between the two sites, this is more noticeable the faster the exchange happens, i.e., for larger values of k. This happens because the relaxation time T_2 characterizes of the molecules found in each site; therefore, at the end of the exchange the sites are occupied by a mixture of molecules given rise to the observed hybrid relaxation time. It is worth mentioning that in practical applications, the observed magnetization signal is processed using the inverse Laplace transform to produce the results we have here, which are then used to determine the value of unknown parameters such as k. In the absence of storage time ($t_2 = 0$) or with small enough storage time ($t_2 \approx 0$) there would not be any visible off-diagonal peaks. A storage time allowing for the transfer from one site to another to take place is essential to the experiment, since useful information is derived from the exchange. In section 5.3, we provided a general solution to system (5.1), i.e., for $N \geq 2$ sites, using techniques of ordinary differential equation.

5.3 Extension to N sites

This section details our steps to extend system (5.1) to a system with an arbitrary number of sites, N. In this case, the system of equations can be given as:

$$\frac{dM^{(i)}}{dt} = -\sum_{j\neq i}^{N} k_{ij}M^{(i)} + \sum_{j\neq i}^{N} k_{ji}M^{(j)} + R^{(i)}(M_{eq}^{(i)} - M^{(i)}), \quad \forall i = 1\dots N, \tag{5.3}$$

and the detailed balance equations are given by:

$$\sum_{j\neq i}^{N} k_{ij}M_{eq}^{(i)} - \sum_{j\neq i}^{N} k_{ji}M_{eq}^{(j)} = 0, \quad \forall i = 1\cdots N. \tag{5.4}$$

Next, we used the detailed balance equations (5.4) to turn the system of equations (5.3) from a non-homogeneous system to a homogeneous system,

$$\frac{d(M^{(i)} - M_{eq}^{(i)})}{dt} = (-R^{(i)} - \sum_{j\neq i}^{N} k_{ij})(M^{(i)} - M_{eq}^{(i)}) + \sum_{j\neq i}^{N}(k_{ji}(M^{(j)} - M_{eq}^{(j)})), \tag{5.5}$$

which were rewritten using matrices and vectors:

$$\frac{d\mathbf{X}}{t} = \mathbf{A}\mathbf{X}, \tag{5.6}$$

where $\mathbf{X} = \mathbf{M} - \mathbf{M}_{eq}$ is an $N \times 1$ column vector with

$$\mathbf{M} = \begin{bmatrix} M^{(1)} \\ M^{(2)} \\ \vdots \\ M^{(N)} \end{bmatrix} \quad \text{and} \quad \mathbf{M}_{eq} = \begin{bmatrix} M_{eq}^{(1)} \\ M_{eq}^{(2)} \\ \vdots \\ M_{eq}^{(N)} \end{bmatrix};$$

and where $\mathbf{A}$ is an $N \times N$ matrix such that,

$$\mathbf{A} = \begin{bmatrix} -R^{(1)} - \sum_{j\neq 1}^{N} k_{1j} & k_{21} & k_{31} & \cdots & k_{N1} \\ k_{12} & -R^{(2)} - \sum_{j\neq 2}^{N} k_{2j} & k_{32} & \cdots & k_{N2} \\ k_{13} & k_{23} & \ddots & \vdots & k_{N3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ k_{1N} & k_{2N} & k_{3N} & \cdots & -R^{(N)} - \sum_{j=1}^{N-1} k_{Nj} \end{bmatrix}.$$

The solution to the differential equation (5.6) is given as:

$$\begin{aligned} \mathbf{X} &= \exp(\mathbf{A}t)\mathbf{X}(t=0) \\ \mathbf{M} &= \exp(\mathbf{A}t)(\mathbf{M}(t=0) - \mathbf{M}_{eq}) + \mathbf{M}_{eq}, \end{aligned}$$

with $\mathbf{M}(t=0)$ as the initial condition to the system.

This solution is used to derive the overall solution to the T_2-store-T_2 experiment. We start from the beginning, the first period (first T_2 period) right after the magnetization has been flipped to the transverse plane. $\mathbf{M}_{eq} = \mathbf{0}$ and $\mathbf{M}(t=0) = \mathbf{M}_0$ to yield the solution:

$$\mathbf{M}_1(t) = \exp(\mathbf{A}_2 t)\mathbf{M}_0,$$

where $\mathbf{A}_2$ is the matrix $\mathbf{A}$ evaluated using the T_2 relaxation time. Let t_1 be the duration of the first period so that at the end the magnitude of the magnetization recorded can be used as the initial state for the storing period. Therefore, at the beginning of the second period (storing period), $\mathbf{M}_{eq} = \mathbf{M}_0$ and $\mathbf{M}(t=0) = \mathbf{M}_1(t_1)$ to yield the solution:

$$\mathbf{M}_2(t) = \exp(\mathbf{A}_1 t)(\mathbf{M}_1(t_1) - \mathbf{M}_0) + \mathbf{M}_0,$$

where $\mathbf{A}_1$ is the matrix $\mathbf{A}$ evaluated using the T_1 relaxation time. Let t_2 be the duration of the second period so that at the end the magnitude of the magnetization recorded can be used as an initial state for the third period (second T_2 period). At the beginning of the third period, $\mathbf{M}_{eq} = \mathbf{0}$ and $\mathbf{M}(t = 0) = \mathbf{M}_2(t_2)$ to yield the solution:

$$\mathbf{M}_3(t) = \exp(\mathbf{A}_2 t)\mathbf{M}_2(t_2).$$

With t_3 as the duration of the third period the observation at the end of the T_2-store-T_2 is then found to be $\mathbf{M}_3(t_3) = \mathbf{M}(t_1, t_2, t_3)$ and was written as:

$$\mathbf{M}(t_1, t_2, t_3) = \exp(\mathbf{A}_2 t_3)(\exp(\mathbf{A}_1 t_2)((\exp(\mathbf{A}_2 t_1)\mathbf{M}_0) - \mathbf{M}_0) + \mathbf{M}_0). \tag{5.7}$$

A numerical implementation of our solution, i.e., equation (5.7), was done using the same set of parameters as in section 5.2 for validation. The implementation of our solution to system (5.1) yields the results shown in Figure 5.2, these results are congruent with the results shown previously in Figure 5.1, using the solution provided by Monteilhet, which validates our solution. We use our solution in section 5.4 to provide additional results for $N \geq 2$.

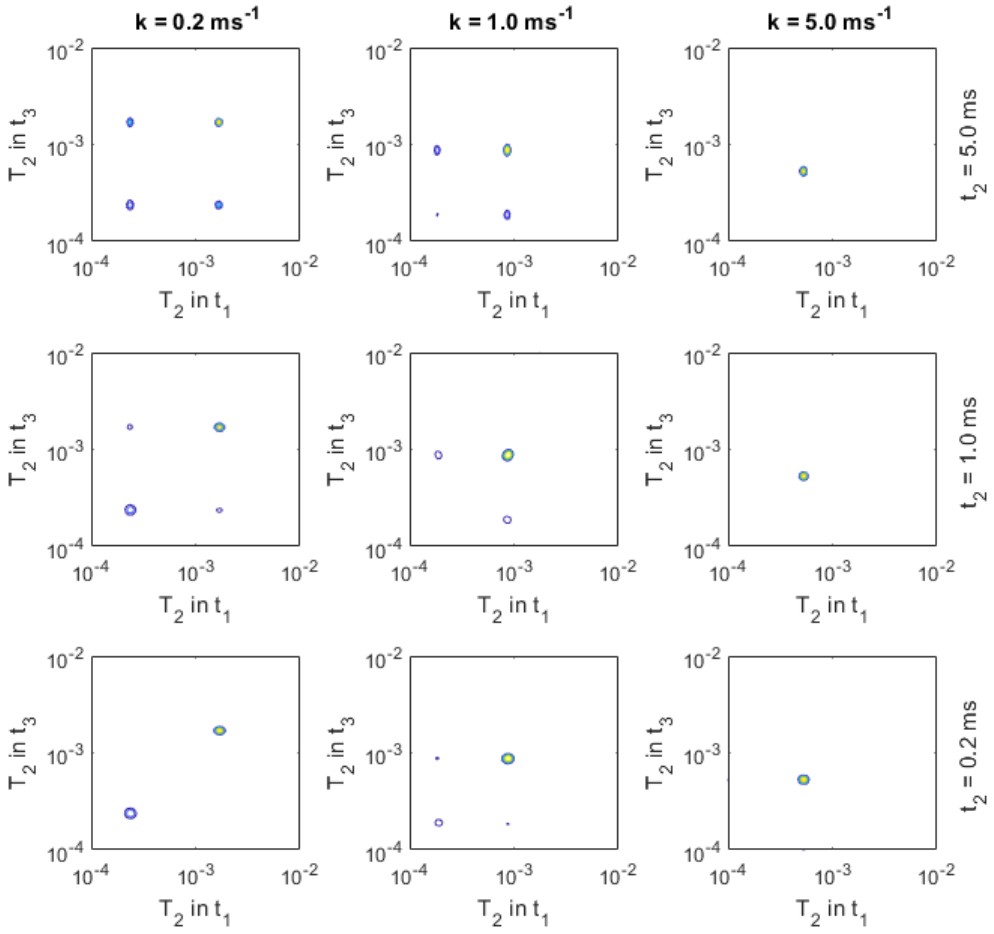

Figure 5.2: Calculated two sites T_2-store-T_2 spectra with increasing exchange rate $k = 0.2, 1.0$ and $5.0\ ms^{-1}$ (left to right) and increasing storing time $t_2 = 0.2, 1.0$ and $5.0\ ms$ (bottom to top.)

5.4 Additional results with $N \geq 2$

Additional results were obtained using the following parameters:

- $N = 2$, 3, 4, and 5, the number of sites is explicitly stated for each experiment.
- $M_{eq}^{(i)} = 1/N$, for $i = 1 \cdots N$.
- The values of t_1, t_3, t_2, $T_1^{(i)}$, and $T_2^{(i)}$ are explicitly stated for each experiment.

- $k = k_{ij}$, for $i \neq j$ between 1 and N, the values of k are explicitly stated for each experiment.

The observations made in this section are similar to those made in Section 5.2, i.e., the longer the storage time allowed in the experiment the more exchange between sites is observed. This is evident in all the figures below; see Figure 5.3 and Figure 5.4. Notice the observed relaxation times T_2 progressively adjust with faster exchange rates to account for the presence of new molecules in the sites. Figure 5.5 and Figure 5.6 provide an example experiment with four sites, for the various storing time, $t_2 = 0.2, 1.0$, and $5.0\ ms$, and various exchange rates, $k = 0.01, 0.05, 0.1, 0.5$, and $1.0\ ms^{-1}$. Following the increasing exchange rates from left to right we can see how the exchange progresses, starting with the four sites. Then, we progressively observe exchanges taking place among sites until all the molecules have transfer to the few remaining sites (notice the shift in T_2 value here as well). A similar experiment is provided for five sites; see Figure 5.7 and Figure 5.8.

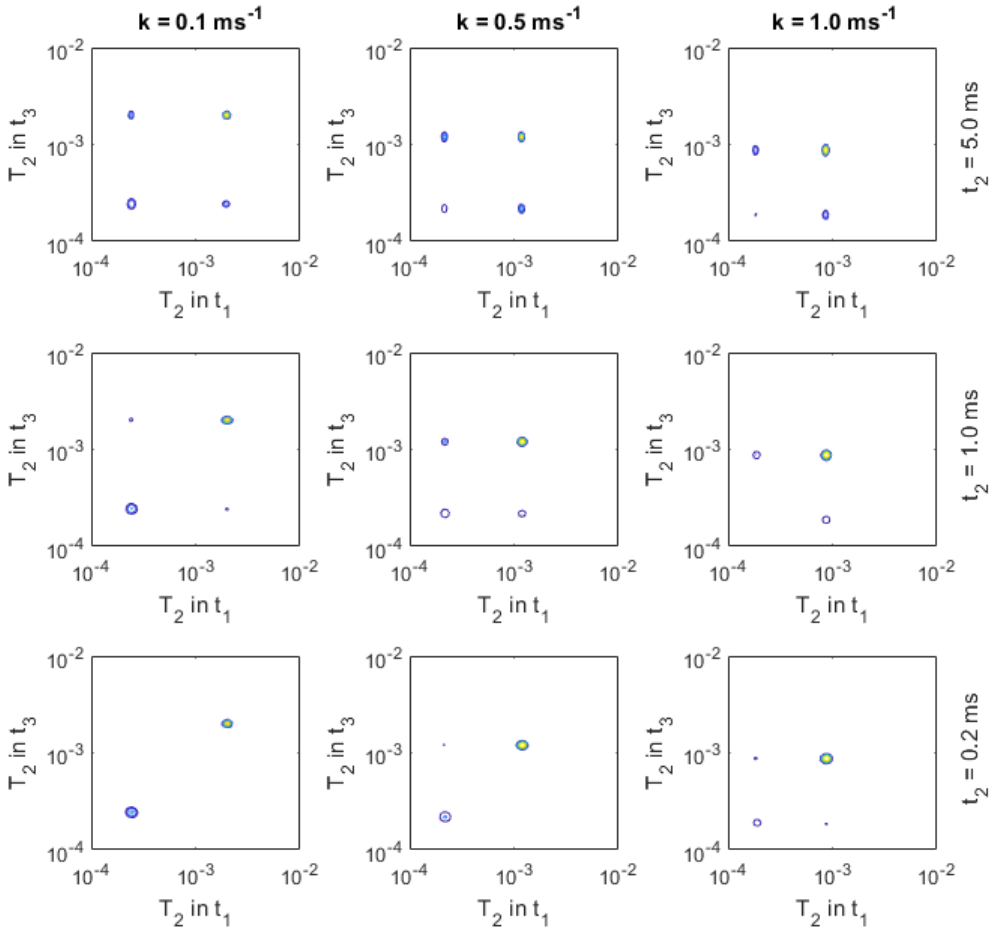

Figure 5.3: Calculated two sites T_2-store-T_2 spectra with increasing exchange rate $k = 0.1, 0.5$ and $1.0\ ms^{-1}$ (left to right) and increasing storing time $t_2 = 0.2, 1.0$ and $5.0\ ms$ (bottom to top.) For this experiment, $T_2^{(1)} = 0.25\ ms$, $T_2^{(2)} = 2.5\ ms$, and $T_1^{(i)} = 1\ s$, for $i = 1, 2$, and $t_1 = t_3 = logspace(-1, 1, 128)\ ms$.

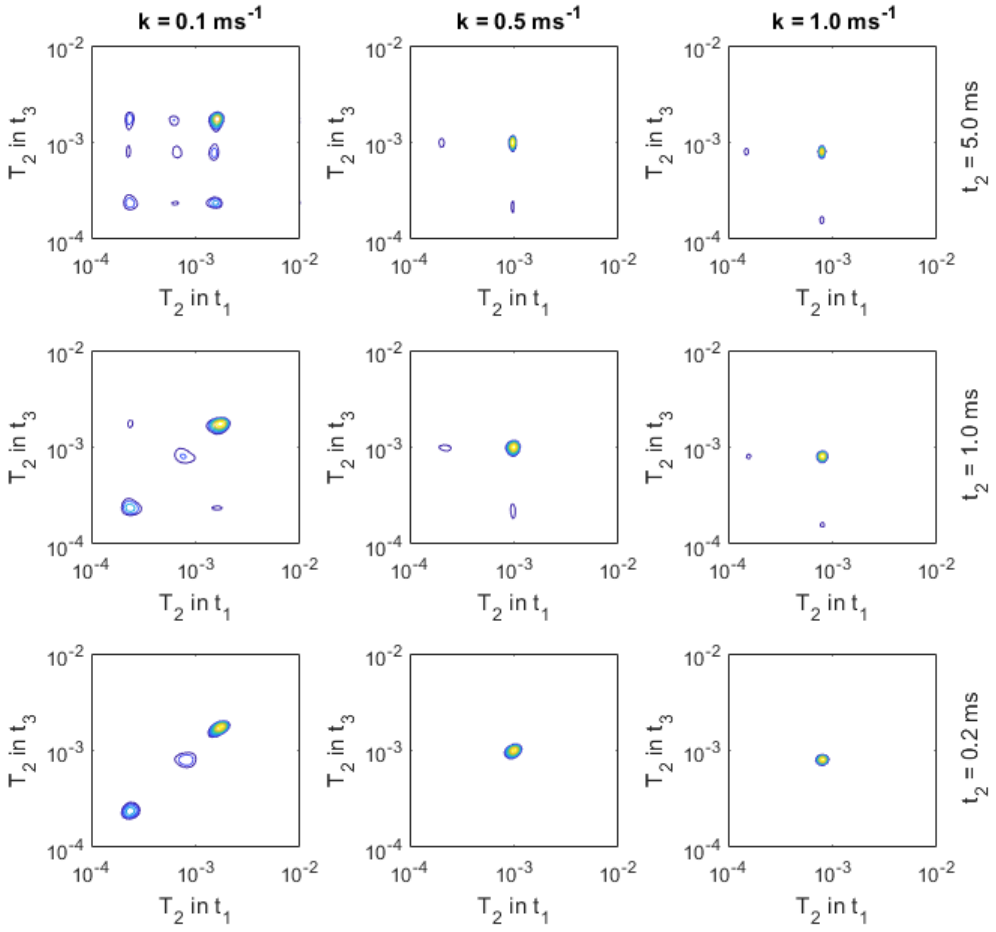

Figure 5.4: Calculated three sites T_2-store-T_2 spectra with increasing exchange rate $k = 0.1, 0.5$ and $1.0\ ms^{-1}$ (left to right) and increasing storing time $t_2 = 0.2, 1.0$ and $5.0\ ms$ (bottom to top.) For this experiment, $T_2^{(1)} = 0.25\ ms$, $T_2^{(2)} = 1.0\ ms$, $T_2^{(3)} = 2.5\ ms$, and $T_1^{(i)} = 1\ s$, for $i = 1, 2, 3$, and $t_1 = t_3 = logspace(-1, 1, 128)\ ms$.

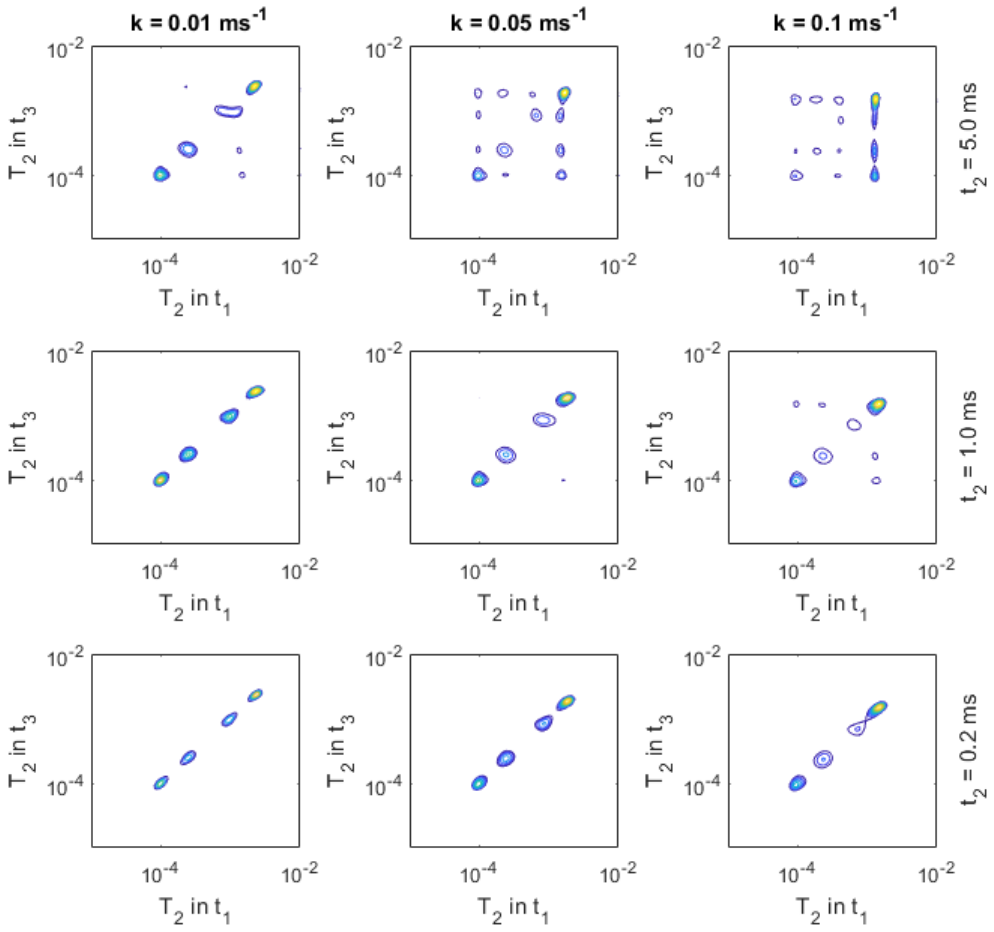

Figure 5.5: Calculated four sites T_2-store-T_2 spectra with increasing exchange rate $k = 0.01, 0.05$ and $0.1\ ms^{-1}$ (left to right) and increasing storing time $t_2 = 0.2, 1.0$ and $5.0\ ms$ (bottom to top.) For this experiment, $T_2^{(1)} = 0.1\ ms$, $T_2^{(2)} = 0.25\ ms$, $T_2^{(3)} = 1\ ms$, $T_2^{(4)} = 2.5\ ms$, $T_1^{(i)} = 1\ s$, for $i = 1, 2, 3, 4$, and $t_1 = t_3 = logspace(-2, 1, 128)\ ms$.

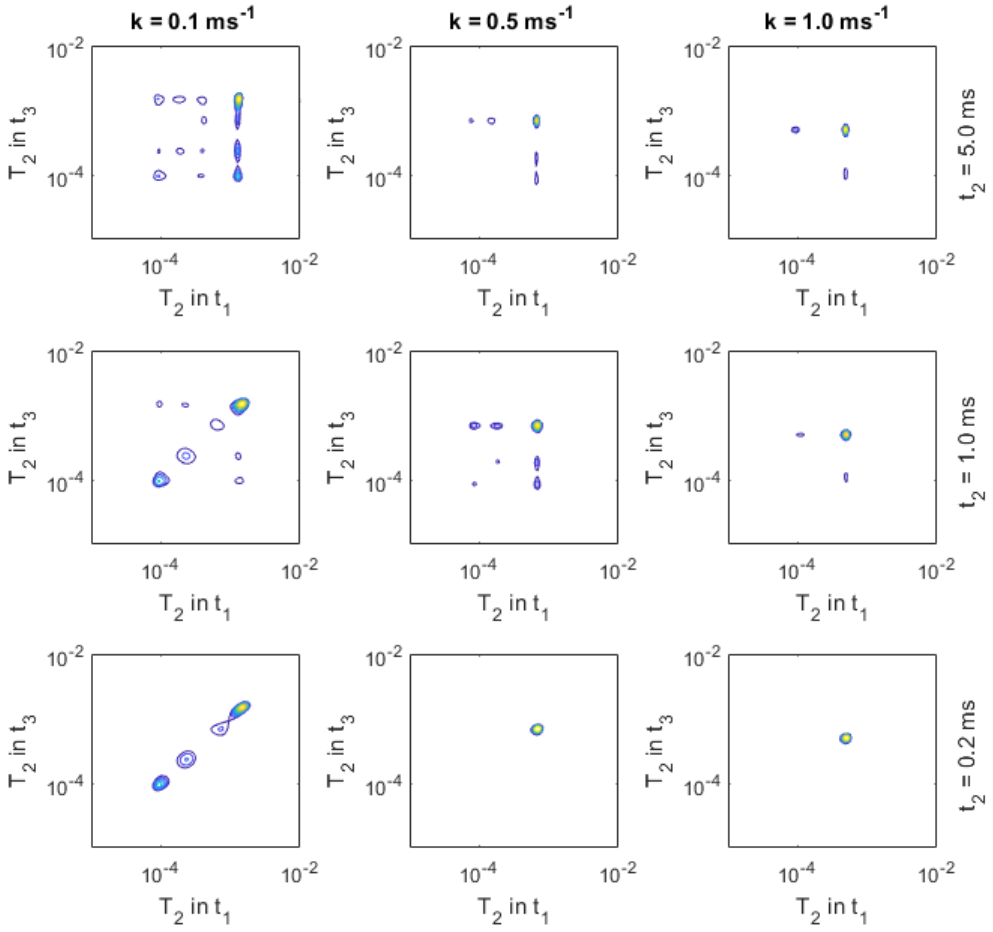

Figure 5.6: Calculated four sites T_2-store-T_2 spectra with increasing exchange rate $k = 0.1, 0.5$ and $1.0\ ms^{-1}$ (left to right) and increasing storing time $t_2 = 0.2, 1.0$ and $5.0\ ms$ (bottom to top.) For this experiment, $T_2^{(1)} = 0.1\ ms$, $T_2^{(2)} = 0.25\ ms$, $T_2^{(3)} = 1\ ms$, $T_2^{(4)} = 2.5\ ms$, $T_1^{(i)} = 1\ s$, for $i = 1, 2, 3, 4$, and $t_1 = t_3 = logspace(-2, 1, 128)\ ms$.

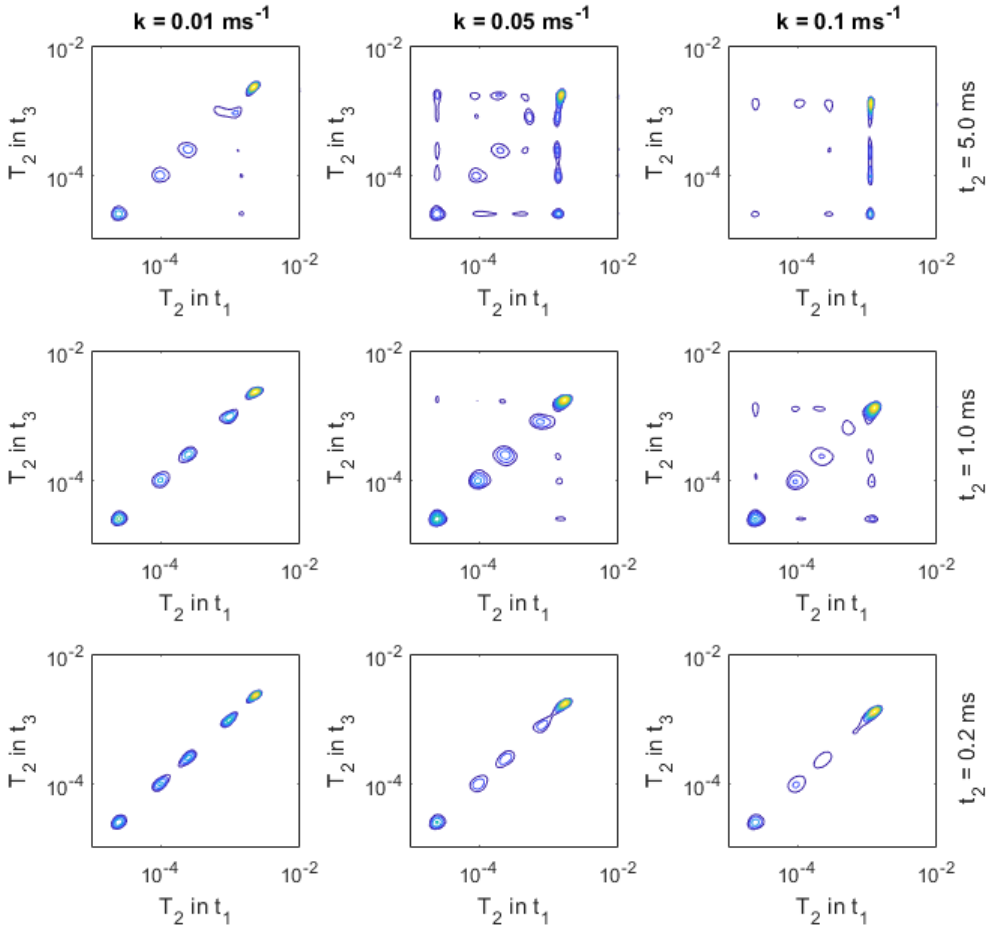

Figure 5.7: Calculated five sites T_2-store-T_2 spectra with increasing exchange rate $k = 0.01, 0.05$ and $0.1\ ms^{-1}$ (left to right) and increasing storing time $t_2 = 0.2, 1.0$ and $5.0\ ms$ (bottom to top.) For this experiment, $T_2^{(1)} = 0.025\ ms$, $T_2^{(2)} = 0.1\ ms$, $T_2^{(3)} = 0.25\ ms$, $T_2^{(4)} = 1\ ms$, $T_2^{(5)} = 2.5\ ms$, $T_1^{(i)} = 1\ s$, for $i = 1, 2, 3, 4, 5$, and $t_1 = t_3 = logspace(-2, 1, 128)\ ms$.

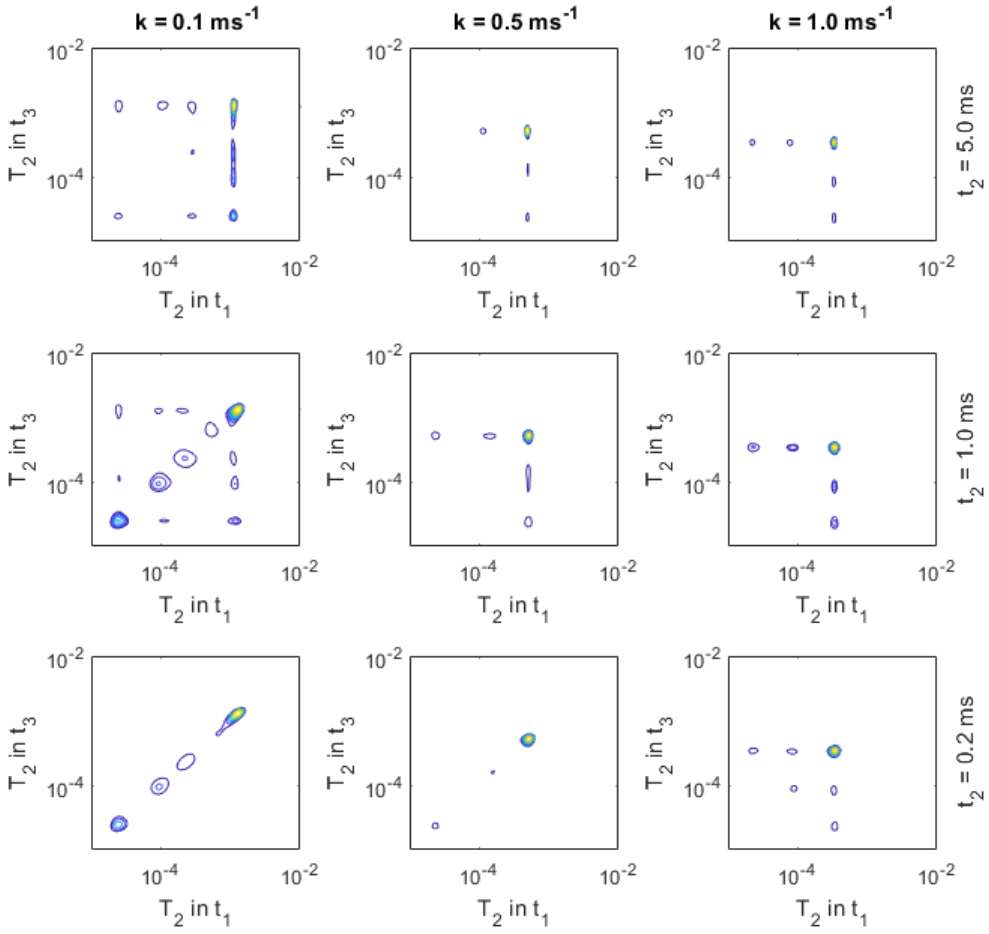

Figure 5.8: Calculated five sites T_2-store-T_2 spectra with increasing exchange rate $k = 0.1, 0.5$ and $1.0\ ms^{-1}$ (left to right) and increasing storing time $t_2 = 0.2, 1.0$ and $5.0\ ms$ (bottom to top.) For this experiment, $T_2^{(1)} = 0.025\ ms$, $T_2^{(2)} = 0.1\ ms$, $T_2^{(3)} = 0.25\ ms$, $T_2^{(4)} = 1\ ms$, $T_2^{(5)} = 2.5\ ms$, $T_1^{(i)} = 1\ s$, for $i = 1, 2, 3, 4, 5$, and $t_1 = t_3 = logspace(-2, 1, 128)\ ms$.

5.5 Conclution

In this chapter, we provided a theoretical expansion of the analysis of the T_2-store-T_2 magnetic resonance relaxometry experiment with N exchanging sites. We have shown that our developed the solution using ordinary differential equation

matches the previous solution provided by Monteilhet et al. [69] for $N = 2$ exchanging sites. Furthermore, we provided additional results for $N \geq 2$, and showed that our mathematical results agree with physical and experimental observations, i.e., longer storage times allow for more exchange to take place between the sites.

Bibliography

[1] *Hyperspectral remote sensing scenes*, `http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes`, Accessed: 2018-04-04.

[2] B. Alipanahi, A. Delong, M. T. Weirauch, and B. J. Frey, *Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning*, Nature Biotechnology **33** (2015), no. 8, 831.

[3] A. Atzmon and A. Olevski, *Completeness of integer translates in function spaces on R*, journal of Approximation Theory **87** (1996), no. 3, 291–327.

[4] M. F. Baumgardner, L. L. Biehl, and D. A. Landgrebe, *220 band AVIRIS hyperspectral image data set: June 12, 1992 Indian Pine test site 3*, Sep 2015.

[5] M. Belkin and P. Niyogi, *Laplacian eigenmaps and spectral techniques for embedding and clustering*, Advances in Neural Information Processing Systems, 2002, pp. 585–591.

[6] ———, *Laplacian eigenmaps for dimensionality reduction and data representation*, Neural Computation **15** (2003), no. 6, 1373–1396.

[7] J-D. Benamou and G. Carlier, *Augmented Lagrangian methods for transport optimization, mean field games and degenerate elliptic equations*, Journal of Optimization Theory and Applications **167** (2015), no. 1, 1–26.

[8] J-D. Benamou, B. D. Froese, and A. M. Oberman, *Numerical solution of the optimal transportation problem using the Monge–Ampère equation*, Journal of Computational Physics **260** (2014), 107–126.

[9] J. J. Benedetto, *Spectral synthesis*, vol. 66, Academic Press, 1976.

[10] J. J. Benedetto and W. Czaja, *Integration and modern analysis*, Springer Science & Business Media, 2010.

[11] J. J. Benedetto, W. Czaja, J. Dobrosotskaya, T. Doster, K. Duke, and D. Gillis, *Semi-supervised learning of heterogeneous data in remote sensing imagery*, Independent Component Analyses, Compressive Sampling, Wavelets, Neural Net, Biosystems, and Nanoengineering X, vol. 8401, International Society for Optics and Photonics, 2012, p. 840104.

[12] J. J. Benedetto, W. Czaja, J. C. Flake, and M. Hirn, *Frame based kernel methods for automatic classification in hyperspectral data*, Geoscience and Remote Sensing Symposium, 2009 IEEE International, IGARSS 2009, vol. 4, IEEE, 2009, pp. IV–697.

[13] T. Bennett, *Transport by advection and diffusion*, Wiley Global Education, 2012.

[14] N. D. Cahill, W. Czaja, and D. W. Messinger, *Schroedinger eigenmaps with nondiagonal potentials for spatial-spectral clustering of hyperspectral imagery*, Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XX, vol. 9088, International Society for Optics and Photonics, 2014, p. 908804.

[15] C. K. Chui and X. Shi, *Bessel sequences and affine frames*, Applied and Computational Harmonic Analysis **1** (1993), no. 1, 29–49.

[16] ———, *On L^p-boundedness of affine frame operators*, Indagationes Mathematicae **4** (1993), no. 4, 431–438.

[17] A. Cloninger, W. Czaja, and T. Doster, *The pre-image problem for Laplacian eigenmaps utilizing L^1 regularization with applications to data fusion*, Inverse Problems **33** (2017), no. 7, 074006.

[18] R. R. Coifman and S. Lafon, *Diffusion maps*, Applied and Computational Harmonic Analysis **21** (2006), no. 1, 5–30.

[19] K. Conrad, *L^p spaces for $0 < p < 1$*, `http://www.math.uconn.edu/~kconrad/blurbs/analysis/lpspace.pdf`, Accessed: 2018-06-09.

[20] W. Czaja and M. Ehler, *Schroedinger eigenmaps for the analysis of biomedical data*, IEEE Transactions on Pattern Analysis and Machine Intelligence **35** (2013), no. 5, 1274–1280.

[21] I. Daubechies, *Ten lectures on wavelets*, vol. 61, Siam, 1992.

[22] M. M. Day et al., *The spaces L^p with $0 < p < 1$*, Bulletin of the American Mathematical Society **46** (1940), no. 10, 816–823.

[23] C. De Boor, R. A. DeVore, and A. Ron, *Approximation from shift-invariant subspaces of $L^2(R^d)$*, Transactions of the American Mathematical Society **341** (1994), no. 2, 787–806.

[24] R. A. DeVore, B. Jawerth, and V. Popov, *Compression of wavelet decompositions*, American Journal of Mathematics **114** (1992), no. 4, 737–785.

[25] R. A. DeVore, D. Leviatan, and X. M. Yu, *Polynomial approximation in L^p $(0 < p < 1)$*, Constructive Approximation **8** (1992), no. 2, 187–201.

[26] L. P. Dorado-Munoz and D. W. Messinger, *Initial study of Schroedinger eigenmaps for spectral target detection*, Optical Engineering **55** (2016), no. 8, 083101.

[27] T. J. Doster, *Harmonic analysis inspired data fusion for applications in remote sensing*, Ph.D. thesis, University of Maryland, College Park, 2014.

[28] R. R. Ernst, G. Bodenhausen, A. Wokaun, et al., *Principles of nuclear magnetic resonance in one and two dimensions*, vol. 14, Clarendon Press Oxford, 1987.

[29] D. Esteban and C. Galand, *Application of quadrature mirror filters to split band voice coding schemes*, Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'77., vol. 2, IEEE, 1977, pp. 191–195.

[30] V. I. Filippov, *On subsystems of the Haar system in the space $(E)\phi$ with $\lim_{t\to\infty} \phi(t) = 0$*, Zametki **51**, 97–106.

[31] ———, *Subsystems of the Faber–Schauder system in function space*, Izvestiya Vysshikh Uchebnykh Zavedenii. Matematika (1991), no. 2, 78–85.

[32] ———, *On the completeness and other properties of some function systems in L^p, $0 < p < \infty$*, Journal of Approximation Theory **94** (1998), no. 1, 42–53.

[33] V. I. Filippov and P. Oswald, *Representation in L^p by series of translates and dilates of one function*, Journal of Approximation Theory **82** (1995), no. 1, 15–29.

[34] J. C. Flake, *The multiplicative Zak transform, dimension reduction, and wavelet analysis of LIDAR data*, University of Maryland, College Park, 2010.

[35] N. Frankel, G. K. Davis, D. Vargas, S. Wang, F. Payre, and D. L. Stern, *Phenotypic robustness conferred by apparently redundant transcriptional enhancers*, Nature **466** (2010), no. 7305, 490.

[36] H. Friebolin and J. K. Becconsall, *Basic one-and two-dimensional NMR spectroscopy*, VCH Weinheim, 1993.

[37] C. J. Galbán and R. G. Spencer, *Measurement of spin-lattice relaxation times and chemical exchange rates in multiple-site systems using progressive saturation*, Magnetic Resonance in Medicine **58** (2007), no. 1, 8–18.

[38] S. Gerber and M. Maggioni, *Multiscale strategies for computing optimal transport*, J. Mach. Learn. Res. **18** (2017), no. 1, 2440–2471.

[39] A. Haar, *Zur theorie der orthogonalen funktionensysteme*, Mathematische Annalen **69** (1910), no. 3, 331–371.

[40] W. Haerty and C. P. Ponting, *Unexpected selection to retain high GC content and splicing enhancers within exons of multiexonic lncRNA loci*, RNA **21** (2015), no. 3, 320–332.

[41] A. Halevy, *Extensions of Laplacian eigenmaps for manifold learning*, Ph.D. thesis, 2011.

[42] P. R. Halmos et al., *N. bourbaki, intégration*, Bulletin of the American Mathematical Society **59** (1953), no. 3, 249–255.

[43] K. B. Hansen and S. C. Shadden, *A reduced-dimensional model for near-wall transport in cardiovascular flows*, Biomechanics and Modeling in Mechanobiology **15** (2016), no. 3, 713–722.

[44] M. J. Hirn, *Enumeration of harmonic frames and frame based dimension reduction*, University of Maryland, College Park, 2009.

[45] J-W. Hong, D. A. Hendrix, and M. S. Levine, *Shadow enhancers as a source of evolutionary novelty*, Science (New York, NY) **321** (2008), no. 5894, 1314.

[46] A. Horská, J. Horsky, and R. G. Spencer, *Measurement of spin-lattice relaxation times in systems undergoing chemical exchange*, Journal of Magnetic Resonance, Series A **110** (1994), no. 1, 82–89.

[47] A. Horská and R. G. Spencer, *Measurement of spin-lattice relaxation times and kinetic rate constants in rat muscle using progressive partial saturation and steady-state saturation transfer*, Magnetic Resonance in Medicine **36** (1996), no. 2, 232–240.

[48] W. Hundsdorfer and J. G. Verwer, *Numerical solution of time-dependent advection-diffusion-reaction equations*, vol. 33, Springer Science & Business Media, 2013.

[49] O. N. Irrechukwu, D. A. Reiter, P-C. Lin, R. A. Roque, K. W. Fishbein, and R. G. Spencer, *Characterization of engineered cartilage constructs using multiexponential T_2 relaxation analysis and support vector regression*, Tissue Engineering Part C: Methods **18** (2012), no. 6, 433–443.

[50] V. I. Ivanov, *Representation of measurable functions by multiple trigonometric series*, Trudy Matematicheskogo Instituta imeni VA Steklova **164** (1983), 100–123.

[51] A. K. Jain and B. Chandrasekaran, *39 Dimensionality and sample size considerations in pattern recognition practice*, Handbook of Statistics **2** (1982), 835–855.

[52] R-Q. Jia and J. J. Lei, *Approximation by multiinteger translates of functions having global support*, Journal of Approximation Theory **72** (1993), no. 1, 2–23.

[53] R-Q. Jia and C. A. Micchelli, *Using the refinement equations for the construction of pre-wavelets II: powers of two*, Curves and Surfaces, Elsevier, 1991, pp. 209–246.

[54] K. R. Kalari, M. Casavant, T. B. Bair, H. L. Keen, J. M. Comeron, T. L. Casavant, and T. E. Scheetz, *First exons and introns–a survey of GC content and gene structure in the human genome*, In Silico Biology **6** (2006), no. 3, 237–242.

[55] H. Katki, G. H. Weiss, J. E. Kiefer, H. Taitelbaum, and R. G. Spencer, *Optimization of magnetization transfer experiments to measure first-order rate constants and spin-lattice relaxation times*, NMR in Biomedicine **9** (1996), no. 3, 135–139.

[56] J. F. Korobeĭnik, *Representing systems*, Mathematics of the USSR-Izvestiya **12** (1978), no. 2, 309.

[57] R. Kronland-Martinet, J. Morlet, and A. Grossmann, *Analysis of sound patterns through wavelet transforms*, International Journal of Pattern Recognition and Artificial Intelligence **1** (1987), no. 02, 273–302.

[58] A. Kundaje, W. Meuleman, J. Ernst, M. Bilenky, A. Yen, A. Heravi-Moussavi, P. Kheradpour, Z. Zhang, J. Wang, M. J Ziller, et al., *Integrative analysis of 111 reference human epigenomes*, Nature **518** (2015), no. 7539, 317.

[59] R. S. Laugesen, *Affine synthesis onto L^p when $0 < p \leq 1$*, Journal of Fourier Analysis and Applications **14** (2008), no. 2, 235–266.

[60] D. Lee, R. Karchin, and M. A. Beer, *Discriminative prediction of mammalian enhancers from DNA sequence*, Genome Research **21** (2011), no. 12, 2167–2180.

[61] Y. Li, Y. Liu, and C. Zhang, *Discriminant diffusion maps based k-nearest-neighbour for batch process fault detection*, The Canadian Journal of Chemical Engineering (2018).

[62] W. R. Madych, *Some elementary properties of multiresolution analyses of $L^2(R^n)$*, Wavelets, Elsevier, 1992, pp. 259–294.

[63] S. Mallat, *Multiresolution approximations and wavelet orthonormal bases of* $L^2(R)$, Transactions of the American Mathematical Society **315** (1989), no. 1, 69–87.

[64] ———, *A wavelet tour of signal processing*, Academic Press, 1999.

[65] S. G. Mallat, *A theory for multiresolution signal decomposition: the wavelet representation*, IEEE Transactions on Pattern Analysis and Machine Intelligence **11** (1989), no. 7, 674–693.

[66] H. M. McConnell, *Reaction rates by nuclear magnetic resonance*, The Journal of Chemical Physics **28** (1958), no. 3, 430–431.

[67] C. E. Metz, *Basic principles of ROC analysis*, Seminars in Nuclear Medicine, vol. 8, Elsevier, 1978, pp. 283–298.

[68] Y. Meyer, *Ondelettes et opérateurs: I: ondelettes; II: opérateurs de Calderón-Zygmund*, Hermann, 1990.

[69] L. Monteilhet, J-P. Korb, J. Mitchell, and P. J. McDonald, *Observation of exchange of micropore water in cement pastes by two-dimensional* T_2 - T_2 *nuclear magnetic resonance relaxometry*, Physical Review E **74** (2006), no. 6, 061404.

[70] T. J. Mosher and B. J. Dardzinski, *Cartilage MRI* T_2 *relaxation time mapping: overview and applications*, Seminars in Musculoskeletal Radiology, vol. 8, 2004, pp. 355–368.

[71] J. Murphy and M. Maggioni, *Unsupervised geometric learning of hyperspectral images*, arXiv preprint arXiv:1704.07961 (2017).

[72] J. M. Nichols, A. T. Watnik, T. Doster, S. Park, A. Kanaev, L. Cattell, and G. K. Rohde, *An optimal transport model for imaging in atmospheric turbulence*, arXiv preprint arXiv:1705.01050 (2017).

[73] M. Osterwalder, I. Barozzi, V. Tissières, Y. Fukuda-Yuzawa, B. J. Mannion, S. Y. Afzal, E. A. Lee, Y. Zhu, I. Plajzer-Frick, C. S. Pickle, et al., *Enhancer redundancy provides phenotypic robustness in mammalian development*, Nature (2018).

[74] P. Oswald, L^p*-approximation durch Reihen nach dem Haar-orthogonalsystem und dem FaberSchauder-System*, Journal of Approximation Theory **33** (1981), no. 1, 1–27.

[75] ———, *Fourier series and the conjugate function in the classes* $\phi(L)$, Analysis Mathematica **8** (1982), no. 4, 287–303.

[76] ———, *On some convergence properties of Haar-Fourier series in the classes* $\phi(L)$, Acta Mathematica Hungarica **42** (1983), no. 3-4, 279–293.

[77] J. Palme, S. Hochreiter, and U. Bodenhofer, *KeBABS: an R package for kernel-based analysis of biological sequences*, Bioinformatics **31** (2015), no. 15, 2574–2576.

[78] K. Pearson, *On lines and planes of closest fit to systems of point in space*, Philosophical Magazine **2** (1901), no. 11, 559–572.

[79] M. W. Perry, A. N. Boettiger, J. P. Bothma, and M. Levine, *Shadow enhancers foster robustness of Drosophila gastrulation*, Current Biology **20** (2010), no. 17, 1562–1567.

[80] S. J. Raudys, A. K. Jain, et al., *Small sample size effects in statistical pattern recognition: recommendations for practitioners*, IEEE Transactions on Pattern Analysis and Machine Intelligence **13** (1991), no. 3, 252–264.

[81] D. A. Reiter, O. Irrechukwu, P-C. Lin, S. Moghadam, S. V. Thaer, N. Pleshko, and R. G. Spencer, *Improved MR-based characterization of engineered cartilage using multiexponential T_2 relaxation and multivariate analysis*, NMR in Biomedicine **25** (2012), no. 3, 476–488.

[82] D. A. Reiter, R. L. Magin, W. Li, J. J. Trujillo, M. P. Velasco, and R. G. Spencer, *Anomalous T_2 relaxation in normal and degraded cartilage*, Magnetic Resonance in Medicine **76** (2016), no. 3, 953–962.

[83] D. A. Reiter, R. A. Roque, P-C. Lin, S. B. Doty, N. Pleshko, and R. G. Spencer, *Improved specificity of cartilage matrix evaluation using multiexponential transverse relaxation analysis applied to pathomimetically degraded cartilage*, NMR in Biomedicine **24** (2011), no. 10, 1286–1294.

[84] S. T. Roweis and L. K. Saul, *Nonlinear dimensionality reduction by locally linear embedding*, Science **290** (2000), no. 5500, 2323–2326.

[85] K. V. Runovskii, *On approximation by families of linear polynomial operators in L^p-spaces,* $0 < p < 1$, Matematicheskii Sbornik **185** (1994), no. 8, 81–102.

[86] J. M. Santos and M. Embrechts, *On the use of the adjusted Rand index as a metric for evaluating supervised classification*, International Conference on Artificial Neural Networks, Springer, 2009, pp. 175–184.

[87] B. Schölkopf, A. Smola, and K-R. Müller, *Kernel principal component analysis*, International Conference on Artificial Neural Networks, Springer, 1997, pp. 583–588.

[88] M. Smith and T. Barnwell, *Exact reconstruction techniques for tree-structured subband coders*, IEEE Transactions on Acoustics, Speech, and Signal Processing **34** (1986), no. 3, 434–441.

[89] R. G. Spencer and K. W. Fishbein, *Measurement of spin-lattice relaxation times and concentrations in systems with chemical exchange using the one-pulse sequence: breakdown of the Ernst model for partial saturation in nuclear magnetic resonance spectroscopy*, Journal of Magnetic Resonance **142** (2000), no. 1, 120–135.

[90] È. A. Storozhenko, *On the approximation by algebraic polynomials of functions of class L^p, $0 < p < 1$*, Izvestiya Rossiiskoi Akademii Nauk. Seriya Matematicheskaya **41** (1977), no. 3, 652–662.

[91] È. A. Storozhenko, V. G. Krotov, and P. Oswald, *Direct and converse theorems of Jackson type in L^p spaces, $0 < p < 1$*, Matematicheskii Sbornik **140** (1975), no. 3, 395–415.

[92] E. S. Şuhubi, *Differentiation of operators*, Functional Analysis, Springer, 2003, pp. 613–676.

[93] W. Sun, A. Halevy, J. J. Benedetto, W. Czaja, W. Li, C. Liu, B. Shi, and R. Wang, *Nonlinear dimensionality reduction via the ENH-LTSA method for hyperspectral image classification*, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **7** (2014), no. 2, 375–388.

[94] W. Sun, A. Halevy, J. J. Benedetto, W. Czaja, C. Liu, H. Wu, B. Shi, and W. Li, *UL-Isomap based nonlinear dimensionality reduction for hyperspectral imagery classification*, ISPRS Journal of Photogrammetry and Remote Sensing **89** (2014), 25–36.

[95] A. A. Talaljan, *On systems of functions whose series represent arbitrary measurable functions*, Sbornik: Mathematics **5** (1968), no. 1, 39–51.

[96] ———, *Representation of functions of the space $L^p[0,1]$, $0 < p < 1$, by orthogonal series*, Acta Math **21** (1970), no. 1/2, 1–9.

[97] ———, *On approximation properties of certain incomplete systems*, Sbornik: Mathematics **43** (1982), no. 4, 443–471.

[98] J. B. Tenenbaum, V. De Silva, and J. C. Langford, *A global geometric framework for nonlinear dimensionality reduction*, Science **290** (2000), no. 5500, 2319–2323.

[99] P. L. Ul'yanov, *Representation of functions by series and classes $\phi(L)$*, Russian Mathematical Surveys **27** (1972), no. 2, 1.

[100] C. Vreugdenhil and B. Koren, *Numerical methods for advection- diffusion problems*, Notes on Numerical Fluid Mechanics (1993).

[101] K. E. Washburn and P. T. Callaghan, *Tracking pore to pore exchange using relaxation exchange spectroscopy*, Physical Review Letters **97** (2006), no. 17, 175502.

[102] D. P. Widemann, *Dimensionality reduction for hyperspectral data*, University of Maryland, College Park, 2008.

[103] X. Zhang, S. E. Chew, Z. Xu, and N. D. Cahill, *SLIC superpixels for efficient graph-based dimensionality reduction of hyperspectral imagery*, Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XXI, vol. 9472, International Society for Optics and Photonics, 2015, p. 947209.

[104] W. Zhao and S. Du, *Spectral–spatial feature extraction for hyperspectral image classification: a dimension reduction and deep learning approach*, IEEE Transactions on Geoscience and Remote Sensing **54** (2016), no. 8, 4544–4554.

[105] X. Zhong and D. Enke, *Forecasting daily stock market return using dimensionality reduction*, Expert Systems with Applications **67** (2017), 126–139.

[106] B. Zhu, W. Zhang, T. Zhang, B. Liu, and J. Jiang, *Genome-wide prediction and validation of intergenic enhancers in Arabidopsis using open chromatin signatures*, The Plant Cell **27** (2015), no. 9, 2415–2426.

[107] M. H. Zweig and G. Campbell, *Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine.*, Clinical Chemistry **39** (1993), no. 4, 561–577.