

SAMPLING AND FILTERING OF SIGNALS ON GRAPHS
WITH APPLICATIONS TO ACTIVE LEARNING AND IMAGE
PROCESSING

by

Akshay Gadde

A Dissertation Presented to the
FACULTY OF THE USC GRADUATE SCHOOL
UNIVERSITY OF SOUTHERN CALIFORNIA

In Partial Fulfillment of the
Requirements for the Degree
DOCTOR OF PHILOSOPHY
(ELECTRICAL ENGINEERING)

August 2017

Copyright 2017

Akshay Gadde

Dedication

To my family.

Acknowledgments

My doctoral endeavor would not have been possible without the help of my teachers, family and friends. It is my pleasure to thank everyone of them. First and foremost, I would like to express my profound gratitude towards my adviser, Prof. Antonio Ortega, for his guidance and support. Prof. Ortega has helped me develop from a curious student into a mature researcher by guiding me through every stage of the research process, from problem formulation to clear exposition of ideas. At the same time, he gave me the opportunity and encouragement to be independent and provided me with moral support whenever I was feeling stuck or lost in the process.

It is my pleasant duty to thank Prof. Salman Avestimehr who taught me to look at any problem from a theoretical point of view. My interactions with him have been highly rewarding. I am very grateful to Prof. Yan Liu, Prof. Richard Leahy and Prof. Keith Jenkins for serving on my qualifying exam and/or dissertation committee. Their comments have certainly helped improve this thesis. I am indebted to all my teachers at USC. I learned a lot from their courses. Their careful explanation of advanced topics forms the basis of all my scientific investigations. I must also thank my collaborators in research, Dr. Sunil Narang, Aamir Anis, Dr. Eyal En Gad, Eduardo Sanou and Jessie Chao. My discussions with them have been immensely useful and without their input this thesis would be incomplete.

Interaction with my friends in the department and outside has provided me with emotional relief much needed in a stressful research atmosphere. For that, I would like to thank Praveen Sharma, Ruchir Travadi, Aamir Anis, Hilmi Egilmez, Eduardo Pavez and Eric Bencomo.

Throughout my stay at USC, I have enjoyed the emotional support provided by my parents, my sister and my brother whose pride in my work has kept my

spirits buoyant throughout. They have been a constant source of inspiration and encouragement which has sustained me in every way. I cannot thank them enough.

Finally, I am highly indebted to USC for providing all facilities for research and a congenial, exciting environment.

Contents

Dedication	ii
Acknowledgments	iii
List of Tables	viii
List of Figures	ix
Abstract	xi
1 Introduction	1
1.1 Related Work	2
1.2 Research Questions and Contributions	4
1.2.1 Sampling Theory of Graph Signals	4
1.2.2 Graph Construction from Data	7
1.3 Outline	8
2 Graph Fourier Transform and Filtering	10
2.1 Notation	10
2.2 Notions of Frequency for Graph Signals	11
2.3 Examples of Variation Operators	12
2.3.1 Variation on Undirected Graphs	12
2.3.2 Variation on Directed Graphs	13
2.4 Spectral Filtering	17
2.4.1 Polynomial Approximation for Fast Filtering	17
2.5 Summary	18
3 Sampling Set Selection and Reconstruction of Graph Signals	19
3.1 Related Work	20
3.2 Sampling Theory for Graph Signals	21
3.2.1 Uniqueness of Reconstruction	22
3.2.2 Issue of Stability and Choice of Sampling set	23
3.3 Sampling Set Selection Using Graph Spectral Proxies	26

3.3.1	Cutoff Frequency	26
3.3.2	Best Sampling Set of Given Size	28
3.3.3	Finding the Best Sampling Set	30
3.4	GFT-free Bandlimited Reconstruction	30
3.5	Complexity	34
3.6	Experiments	35
3.6.1	Sampling Set Selection	35
3.6.2	Efficient Bandlimited Reconstruction	36
3.7	Conclusion	37
4	Active Semi-supervised Learning Using Sampling Theory	39
4.1	Introduction	39
4.2	Graph Sampling Based Active Semi-Supervised Learning	41
4.2.1	Proposed method	42
4.2.2	Graph Theoretic Interpretation	44
4.2.3	Prediction Error and Number of Labels	45
4.3	Related Work	46
4.4	Experiments	48
4.4.1	Handwritten digits classification	49
4.4.2	Text classification	50
4.4.3	Spoken letters classification	51
4.4.4	Effect of parameter k	52
4.5	Summary	52
5	Probabilistic Interpretation of Sampling Theory and Extension to Adaptive Sampling	54
5.1	Probabilistic Interpretation of Sampling Theory	57
5.1.1	GRF Prior and Observation Likelihood	57
5.1.2	Bandlimited Reconstruction as MAP Inference	59
5.1.3	Active Learning as Bayesian Experiment Design	60
5.1.4	Optimality of Non-Adaptive Set Selection in the GRF Model	63
5.2	Bayesian Active Learning Using p -Laplacian Based Prior	64
5.2.1	Signal Prior Based on p -Laplacian	64
5.2.2	Bayesian Inference Using p -Laplacian Based Prior	65
5.2.3	Sequential Sampling Using the Posterior Covariance	67
5.3	Experiments	68
5.4	Conclusion	69
6	Efficient Graph Construction from Data for Image Processing and Learning	71
6.1	Laplacian Based Smoothness and GMRF	76
6.2	Proposed Graph Construction Method	76

6.2.1	Graph Estimation by Sparse Non-negative Regression	77
6.2.2	Simplification and Fast Computation	78
6.2.3	Computational Complexity	80
6.2.4	Interpretation and Discussion	80
6.3	Experiments	81
6.3.1	Image Denoising with Sparse Inverse BF Graph	81
6.3.2	Clustering	82
6.3.3	Semi-supervised Learning	83
6.4	Summary	85
7	Conclusions and Future Work	87
7.1	Main Contributions	87
7.2	Future Work	88
	Reference List	90
A	Properties of Spectral Proxies	99

List of Tables

2.1	Different choices of the variation operator \mathbf{L} for defining GFT bases.	15
5.1	Different optimality criteria for active learning on graphs	62
6.1	PSNR (in dB) of images denoised using Wiener filters given by the GFTs computed using 7×7 BF graph, 4-NN heuristic graph and proposed NNK graph.	83
6.2	Average number of non-zero entries in the adjacency matrices of different graphs.	84
6.3	Clustering accuracy with k -NN and NNK (proposed) graph for different values of k and σ	85

List of Figures

2.1	Variation in the eigenvectors of the symmetric normalized Laplacian of a graph. As λ increases, one can see that variation in corresponding graph signal (i.e., eigenvector) also increases.	11
3.1	Iterative reconstruction using POCS	32
3.2	Spectral response of an approximate polynomial filter of degree 10. $\omega = 1, \alpha = 8$	33
3.3	Reconstruction MSE vs. number of samples. The large reconstruction errors for $ \mathcal{S} < 50$ arise due to non-uniqueness of bandlimited reconstruction and hence, are less meaningful.	36
3.4	Relative error of reconstruction vs. number of POCS iterations using exact low pass filter and polynomial filters of different degrees.	37
4.1	Cumulative distribution of energy in the GFT coefficients (with GFT defined using symmetric normalized Laplacian) of one of the class membership functions pertaining to the three real-world dataset experiments considered in Section 4.4. Note that most of the energy is concentrated in the low-pass region.	42
4.2	Toy example comparing the nodes selected using different active learning methods	49
4.3	Comparison of active semi-supervised learning methods on real datasets. Plots show the average classification accuracies for different percentages of labeled data.	51
4.4	Effect of k on classification accuracy of the proposed method. Plots show the average classification accuracy for different percentages of labeled data.	52

5.1	Plots show the average classification accuracies with different number of observed labels selected using 1-Laplacian based adaptive active learning (red) and the 2-Laplacian based active learning method of Chapter 4 with $k = 1$ (black) and $k = 4$ (blue).	69
6.1	Spectral clustering with k -NN and proposed graph, $k = 10, \sigma = 0.1$	84
6.2	SSL results with k -NN and NNK (proposed) graph for different values of k and σ	85

Abstract

Graph signals provide a natural representation for data in many applications such as social networks, web information analysis, sensor networks and machine learning. Traditional data such as images and videos can also be represented as signals on graphs. A frequency domain representation for graph signals can be obtained using the eigenvectors and eigenvalues of operators that measure the variation in signals taking into account the underlying connectivity in the graph. Based on this, we develop a sampling theory for graph signals that answers the following questions: 1. When can we uniquely and stably reconstruct a bandlimited graph signal from its samples on a subset of the nodes? 2. What is the best subset of nodes for sampling a signal so that the resulting bandlimited reconstruction is most stable? 3. How to compute a bandlimited reconstruction efficiently from a subset of samples? The algorithms developed for sampling set selection and reconstruction do not require explicit eigenvalue decomposition of the variation operator and admit efficient, localized implementation. Using graph sampling theory, we propose effective graph based active semi-supervised learning techniques. We also give a probabilistic interpretation of graph sampling. Based on this interpretation, we generalize the framework of sampling on graphs using Bayesian methods to give an adaptive sampling method in which the future choice of nodes to be sampled depends on the samples observed in the past.

Additionally, we study the problem of constructing a sparse graph efficiently from given data and a kernel function that measures pairwise similarity between data points. The proposed graph construction method leads to graph based learning and clustering algorithms that outperform the conventional k -nearest neighbor methods. We also use the proposed graph construction method to provide an efficient alternative to the well-known bilateral filter by representing an image as a sparse graph in which the nodes correspond to the pixels in the image.

Chapter 1

Introduction

A graph is a collection of nodes (or vertices) that are connected together by edges (or links). The edges can be directed and weighted, where the weight on each edge represents the affinity between the vertices that it connects as dictated by the problem at hand. Graphs provide a natural representation for data in many application domains, such as social networks, web information analysis, sensor networks and machine learning [79, 72]. They can also be used to represent conventional data, such as images and videos [29, 62]. A graph signal is a function defined over the nodes of a graph. Analyzing graph signals taking into account the underlying connectivity information is very important in all of these application domains. For example, in social networks it would be of interest to see how personal attributes influence formation of communities, in sensor networks analyzing the correlations between measurements on different sensors may give insights for designing efficient data gathering algorithms, in learning or ranking problems the goal is to predict the unknown labels of nodes from a few training node labels based on the similarity information given by the graph.

Traditional signal processing techniques such as sampling, filtering and wavelet transforms have proven to be very effective tools for analysis, approximation, denoising and interpolation of signals in regular Euclidean spaces. Graph signal processing (GSP) aims to extend these tools to signals on graphs [79, 72]. The problem of generalizing signal processing techniques to graph signals is non-trivial since they lie in irregular non-Euclidean spaces. Thus, analogs of even the simplest signal processing operations such as shifting, downsampling, dilation and filtering are not easily apparent for graph signals. A key challenge in graph signal processing is to design localized algorithms (in which the output at each vertex depends

on its local neighborhood) that scale well with the large sizes of graphs that arise in real world applications.

The main goal of this thesis is to develop a sampling theory for graph signals. Specifically, we focus on the following questions:

- When can we recover a graph signal from a subset of its samples?
- How to choose a good subset of nodes for sampling?
- How can we efficiently reconstruct the signal from the observed samples?

Application of the proposed theory to the problem of active semi-supervised learning is explored in detail. We also provide a probabilistic interpretation of graph sampling. Using this interpretation, we generalize the sampling framework to adaptive sampling, in which the future choice of nodes to be sampled depends on the signal samples observed in the past. A part of the thesis deals with the problem of constructing a sparse graph efficiently from given data and a kernel function that measures pairwise similarity between data points. The proposed graph construction method leads to graph based learning and clustering algorithms that outperform the conventional k -nearest neighbor based methods. We also use the proposed graph construction method to provide an efficient alternative to the well-known bilateral filter for image processing. A detailed description of the research questions studied in this thesis is provided in Section 1.2.

1.1 Related Work

The Fourier transform provides a frequency domain representation for traditional signals, which allows us to characterize their smoothness. Similarly, in order to analyze graph signals, it is useful to have a way to represent them in a graph-dependent basis. Such a basis can be defined using the eigenvectors and eigenvalues of matrices that allow us to measure the variation in a graph signal, taking into account the connectivity information given by the graph. These matrices include the adjacency matrix [73], the Laplacian matrix and its normalized versions [42, 17]. The representation of a graph signal in the basis given by the eigenvectors of the above matrices is called its Graph Fourier Transform (GFT). The GFT enables us to formalize natural smoothness assumptions on the graph signals and to define spectral domain filtering on graphs. Based on the GFT, several wavelet filterbank designs

have been proposed for graph signals, which offer trade-off between vertex and frequency domain localization [42, 60, 61]. These filter banks have two components namely, filtering and downsampling. The filters used in these designs are in the form of the polynomials of the graph Laplacian. Their frequency response is localized while admitting localized implementation in the vertex domain. The downsampling operation involves dropping the samples of the filter output on a subset of the nodes. An important question in this context is to decide which samples to drop (or preserve). Methods based on maximum graph cuts and spanning trees have been proposed to choose the sampling set for filterbank design [59, 63]. The problem of unique reconstruction from the given sampling subset is studied in [67].

Graph based methods have also been proposed in the machine learning literature. In many learning problems, unlabeled data is abundant but labels are scarce and expensive to obtain, often requiring human expertise or elaborate experiments. Active semi-supervised learning (SSL) is an effective way to minimize the cost of labeling [95, 76]. As opposed to supervised learning, which only uses the labeled data to train a classifier, semi-supervised learning techniques learn from both the labeled data and the inherent clustering present in the unlabeled data to improve label prediction. An active learning approach allows the learning algorithm to choose which data points to label, i.e., those that are most helpful in predicting the rest of the labels. A graph based approach to active SSL begins by constructing a graph in which the nodes correspond to data points and weighted edges capture the similarity between them. The *cluster assumption* [12] says that similar nodes are more likely to have same labels, i.e., a graph signal given by the labels is smooth. Different ways of characterizing this smoothness lead to different SSL methods [5, 94, 81, 4]. A graph based approach to active learning chooses those nodes for labeling that minimize the expected classification error of the graph based SSL methods [96, 46] and are well connected to the unlabeled nodes [40].

While many datasets such as social, web and sensor networks are inherently graph structured, in most applications the graph is constructed from data, where each node represents a vector in \mathbb{R}^d , in order to use graph based approaches. The most commonly used graph construction heuristics in clustering and SSL are the k -nearest neighbor and ϵ -neighborhood methods, which connect each node to a few of its nearest neighbors. Other methods have been proposed for graph construction in various contexts such as dimensionality reduction [69] and clustering [25]. Aspects

of GSP can also be viewed through the lens of the Gaussian Markov random field model. In this model, the graph Laplacian corresponds to the inverse covariance of a Gaussian distribution and the problem of graph construction boils down to the problem of inverse covariance estimation using multiple observations drawn from the GMRF [66, 55, 87].

Graph signal filtering and sampling have also found applications in adaptive, edge-aware image processing [29, 56, 26] and compression [62, 11]. In these applications, an image is represented as a signal on a graph in which nodes correspond to pixels and links between the nodes capture similarity between them. Many modern image filtering techniques such as bilateral filtering [83], non-local means filtering [8], kernel regression [82] etc. can be thought of as filters on an image-dependent graph with different link weights. Graph wavelets can be applied to image-dependent graphs to get effective image compression methods [62, 11].

1.2 Research Questions and Contributions

1.2.1 Sampling Theory of Graph Signals

Sampling theory is of immense importance in traditional signal processing, providing a link between analog and discrete time domains and also serving as a major component in many discrete time signal processing systems. Fundamentally, it deals with the problem of recovering a signal from a subset of its samples. It provides the conditions under which the signal has a *unique* and *stable* reconstruction from the given samples. Conversely, it gives the minimum sampling density required in order to get a unique and stable reconstruction for a signal that meets the modeling assumptions. Typically, the signal model is characterized by bandlimitedness in the Fourier domain. For example, the classical Nyquist-Shannon sampling theorem states that a signal in $L^2(\mathbb{R})$ with bandwidth f can be uniquely reconstructed by its (uniformly spaced) samples if the sampling rate is higher than $2f$. Analogous results have been obtained for both regular and irregular sampling of discrete time signals bandlimited in the DFT domain [36].

Sampling theory of graph signals similarly deals with the problem of recovering a graph signal from its samples on a subset of nodes of the graph. A graph signal is said to be ω -bandlimited if its GFT is supported on frequencies in $[0, \omega]$. Bandwidth of a graph signal is defined as the maximum graph frequency for which

it has a non-zero GFT coefficient. Bandwidth can be thought of as a measure of smoothness of a graph signal, i.e., a graph signal with small bandwidth is smooth on the graph. Based on these definitions, we answer the following questions:

1. *When can we uniquely recover an ω -bandlimited graph signal from its samples on a given subset \mathcal{S} of the nodes?* We give the necessary and sufficient condition under which this is possible. Using this condition, we provide a bound on the maximum bandwidth $\omega_c(\mathcal{S})$ that a signal can have (i.e., the cutoff frequency) so that it can be uniquely reconstructed from the given sampling set [57, 1, 2]. The derived bound is GFT-free, i.e., it does not need computation of the GFT basis and therefore, can be computed efficiently.
2. *What is the optimal sampling set for stable bandlimited reconstruction?* In practice, observed samples are noisy. Also, the signals are not necessarily exactly bandlimited. A poor choice of sampling set can result in a very ill-conditioned reconstruction operator that amplifies the sample perturbations caused by noise and model mismatch and thus can lead to large reconstruction errors. Hence, selecting a sampling set that gives stable reconstructions is vital. We show that an optimal sampling set, which minimizes the worst case reconstruction error, is the one that maximizes the bound on the cutoff frequency for unique recovery [1, 2]. We give a greedy algorithm to find an approximately optimal sampling set.
3. *How to find a bandlimited reconstruction efficiently from the given samples?* A naive method of computing the bandlimited reconstruction from given samples involves solving a least squares problem in terms of the GFT basis. However, graphs that model real world data have a large number of nodes. Storage and processing of such graphs demands decentralized memory and computation. Computing their GFT basis is not practical. To circumvent this issue, we develop an efficient GFT-free algorithm for approximate bandlimited reconstruction. The proposed reconstruction is localized, i.e., output at each node depends only on its local neighborhood and thus allows distributed implementation.

Probabilistic Interpretation of Sampling Theory

The smoothness assumption on graph signals can also be formalized by assuming that they can be modeled with a Gaussian Random Field (GRF), in which the covariance depends on the graph structure. Under this model, likelihood of observing a signal increases as its smoothness on the graph increases. We show that bandlimited reconstruction is equivalent to MAP inference on this GRF. The probabilistic model allows us to apply the framework of Bayesian experiment design for sampling set selection. This formulation provides a unified view of various sampling set selection methods proposed in the literature [46, 54] in the context of graph based active learning. Specifically, we can show that an optimal sampling set that maximizes the bound on the cutoff frequency minimizes a function of the predictive covariance [32].

Adaptive Sampling Using Bayesian Methods

The probabilistic view of graph sampling theory allows us to consider its extension to adaptive sampling. The problem of finding the optimal sampling set as posed in the context of sampling theory entails selecting a subset of nodes *before* observing the signal. The choice depends only on the graph and is unaffected by observed signal samples. However, in many applications graph signal samples are observed sequentially one at a time or in small batches. An important question in this setting is if and how we can adapt the sampling strategy using the observed samples in order to improve the future choice of sampling set. Such an adaptation can also involve modifying the graph based on the observed samples.

In order to get an adaptive sampling method, we propose a different prior for graph signals using the concept of p -Laplacian, which is more suited for discrete valued signals that arise in classification problems. Because of the non-Gaussianity of the proposed prior, the posterior predictive covariance depends on the observed labels. Applying the framework of Bayesian experiment design to this model gives us an adaptive sampling scheme, in which the choice of future samples depends on the labels observed in the past.

Application: Active Semi-supervised Learning

Many important practical problems can be formulated as graph signal sampling problems, including active semi-supervised learning [30], sensor placement for environment monitoring [52], design of lifting transforms for image/video compression [11] etc. This thesis, in particular, explores its application to graph based active semi-supervised learning in detail.

Based on the cluster assumption, the graph signal formed by the node labels is likely to be smooth and approximately bandlimited. Therefore, selecting the optimal sampling set for bandlimited reconstruction is a good active learning criterion and the iterative bandlimited reconstruction method gives an efficient label prediction algorithm [30]. Our proposed adaptive sampling method is useful when data points can be labeled sequentially or in small batches. We show that the proposed sampling and reconstruction methods outperform many state of the art active SSL methods.

1.2.2 Graph Construction from Data

Graph based approaches for learning and clustering begin by constructing a graph from given data in the form of vectors in \mathbb{R}^d and a pairwise similarity kernel. Constructing a good graph is important for graph based methods to be effective. A constructed graph should have the following desirable properties:

- Signals of interest should be smooth with respect to the graph since this is the underlying assumption in most graph based methods;
- Graph construction should be robust to data noise;
- Graph should be sparse for graph based methods to be efficient;
- Graph construction should have low computational complexity and memory requirement so that it is scalable to large datasets.

We propose an efficient method for graph construction from data, in which each node represents a vector in \mathbb{R}^d . The complexity of our proposed method is of the same order as that of the k -nearest neighbor method. Motivated by the cluster assumption in semi-supervised learning, we interpret the similarity between

each pair of vectors, given by a positive definite kernel function, as the covariance between the signal values on the corresponding nodes. We then find a graph whose Laplacian approximates the inverse of the kernel similarity matrix by representing each vector as a linear combination of other vectors using ℓ_1 regularized non-negative kernel regression. The proposed method produces a sparse graph that is robust to data noise and choice of kernel parameters.

Application: Efficient Alternative to the Bilateral Filter

In addition to graph based clustering and semi-supervised learning, we also consider the application of our proposed graph construction method for adaptive image filtering. We focus on the bilateral filter (BF), which is widely used for edge-preserving smoothing [83]. The weights of the BF are given by a positive definite similarity kernel that depends on the geometric as well as photometric distance between the pixels. The BF can be interpreted as a simple, 1-hop filter on a dense graph, whose adjacency matrix equals the kernel matrix. Since the kernel matrix is dense, computational complexity of the BF is high. In order to provide an efficient alternative to the BF, we use our proposed graph construction method to construct a much sparser image adaptive graph, whose Laplacian approximates the inverse of the BF kernel matrix and has similar eigen-structure. We can then define multi-hop, Laplacian polynomial filters on the proposed sparse graph that offer similar performance as the BF with lower complexity. Such a sparse image adaptive graph can also be useful in other applications such as image interpolation and compression.

1.3 Outline

In this chapter, we described and motivated the research questions studied in this thesis. In Chapter 2, we define the notions of frequency and bandlimitedness for graph signals, which allow us to formalize the smoothness assumption on them. Based on the frequency definition, spectral filtering of graph signals is also defined. In Chapter 3, the conditions for obtaining a unique and stable bandlimited reconstruction using samples on a subset of the nodes are derived. Using these conditions, a greedy algorithm for selecting a good sampling set, which leads to a stable bandlimited reconstruction, is proposed. We also give an efficient and localized

algorithm for computing an approximate bandlimited reconstruction using a given subset of samples. In Chapter 4, we apply the proposed methods of sampling set selection and bandlimited reconstruction to the problem of active semi-supervised learning. In Chapter 5, a probabilistic interpretation for these methods is given by defining a Gaussian random field model for the signals based on the graph. Based on this interpretation, we propose a different prior for graph signals using the concept of p -Laplacian, which is better suited for classification problems. An adaptive active learning method is developed using the proposed prior. In Chapter 6, we describe an efficient method for constructing a graph from data, which can then be used for clustering and semi-supervised learning. The proposed method also provides an efficient alternative to the bilateral filter. We conclude in Chapter 7 with a discussion of possible directions for future work.

Chapter 2

Graph Fourier Transform and Filtering

In this chapter, we introduce spectral domain representations for graph signals. Such a representation allows us to characterize the smoothness of signals with respect to the graph. It leads naturally to the concepts of filtering and bandlimit-
edness, which are essential for formulating a sampling theory. The ideas presented in this chapter are based on the work in [42, 2, 73]. The rest of the chapter is organized as follows. In Section 2.1, we describe the notation used throughout this thesis. Section 2.2 defines the graph Fourier transform (GFT) using eigenvalues and eigenvectors of operators used for computing the variation in a graph signal. Examples of these variation operators are given in Section 2.3 for both undirected and directed graphs. In Section 2.4, we define spectral filtering of graph signals and present a fast method to implement these filters.

2.1 Notation

A graph $G = (\mathcal{V}, \mathcal{E})$ is a collection of nodes indexed by the set $\mathcal{V} = \{1, \dots, N\}$ and connected by edges (or links) $\mathcal{E} = \{(i, j, w_{ij})\}$, where (i, j, w_{ij}) denotes an edge of weight $w_{ij} \in \mathbb{R}^+$ pointing from node i to node j . We restrict the edge weights to be non-negative since this ensures that the symmetric graph variation operators (defined in Section 2.3) are positive semi-definite. The assumption is useful in many applications of interest. The adjacency matrix \mathbf{W} of the graph is an $N \times N$ matrix with $\mathbf{W}(i, j) = w_{ij}$. A graph signal is a function $f : \mathcal{V} \rightarrow \mathbb{R}$ defined on the vertices of the graph (i.e., a scalar value assigned to each vertex). It can be

represented as a vector $\mathbf{f} \in \mathbb{R}^N$, where f_i is the function value on the i^{th} vertex. For any $\mathbf{f} \in \mathbb{R}^N$ and a set $\mathcal{S} \subseteq \{1, \dots, N\}$, we use $\mathbf{f}_{\mathcal{S}}$ to denote a sub-vector of \mathbf{f} consisting of components indexed by \mathcal{S} . Similarly, for $\mathbf{A} \in \mathbb{R}^{N \times N}$, $\mathbf{A}_{\mathcal{S}_1 \mathcal{S}_2}$ is used to denote the sub-matrix of \mathbf{A} with rows indexed by \mathcal{S}_1 and columns indexed by \mathcal{S}_2 . For simplicity, we denote $\mathbf{A}_{\mathcal{S} \mathcal{S}}$ by $\mathbf{A}_{\mathcal{S}}$. The complement of \mathcal{S} in \mathcal{V} is denoted by $\mathcal{S}^c = \mathcal{V} \setminus \mathcal{S}$. Further, we define $L_2(\mathcal{S})$ to be the space of all graph signals that are zero everywhere except possibly on the subset of nodes \mathcal{S} , i.e.,

$$L_2(\mathcal{S}) = \{\mathbf{f} \in \mathbb{R}^N \mid \mathbf{f}_{\mathcal{S}^c} = \mathbf{0}\}. \quad (2.1)$$

2.2 Notions of Frequency for Graph Signals

In order to formulate a sampling theory for graph signals, we need a notion of frequency that enables us to characterize the level of smoothness of the graph signal with respect to the graph. In practice this is achieved by defining analogs of operators such as shift or variation from traditional signal processing, which allow one to transform a signal or measure its properties while taking into account the underlying connectivity over the graph [2]. Let \mathbf{L} be such an operator in the form of an $N \times N$ matrix. A variation operator creates a notion of smoothness for graph signals through its spectrum. Specifically, assume that \mathbf{L} has eigenvalues $|\lambda_1| \leq \dots \leq |\lambda_N|$ and corresponding eigenvectors $\{\mathbf{u}^1, \dots, \mathbf{u}^N\}$. Then, these eigenvectors provide a Fourier-like basis for graph signals with the frequencies given by the corresponding eigenvalues. For each \mathbf{L} , one can also define a variation functional $\text{Var}(\mathbf{L}, \mathbf{f})$ that measures the variation in any signal \mathbf{f} with respect to \mathbf{L} . If $|\lambda_i| \leq |\lambda_j|$, then $\text{Var}(\mathbf{L}, \mathbf{u}^i) \leq \text{Var}(\mathbf{L}, \mathbf{u}^j)$ (see Figure 2.1).

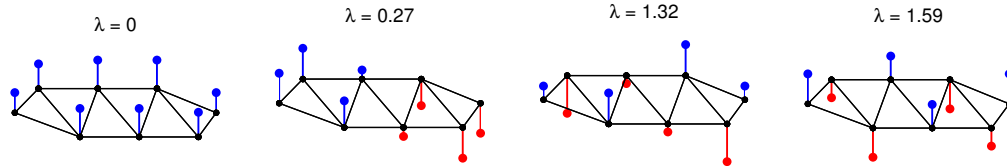


Figure 2.1: Variation in the eigenvectors of the symmetric normalized Laplacian of a graph. As λ increases, one can see that variation in corresponding graph signal (i.e., eigenvector) also increases.

The graph Fourier transform (GFT) $\tilde{\mathbf{f}}$ of a signal \mathbf{f} is given by its representation in the above basis, $\tilde{\mathbf{f}} = \mathbf{U}^{-1}\mathbf{f}$, where $\mathbf{U} = [\mathbf{u}^1 \dots \mathbf{u}^N]$. Note that a GFT

can be defined using different variation operators. Examples of possible variation operators are reviewed in Section 2.3. If the variation operator \mathbf{L} is symmetric then its eigenvectors are orthogonal leading to an orthogonal GFT. In some cases, \mathbf{L} may not be diagonalizable. In such cases, one can resort to the Jordan normal form [73] and use generalized eigenvectors.

A signal \mathbf{f} is said to be ω -bandlimited if $\tilde{\mathbf{f}}_i = 0$ for all i with $|\lambda_i| > \omega$. In other words, GFT of an ω -bandlimited \mathbf{f} is supported on frequencies in $[0, \omega]$. If $\{\lambda_1, \lambda_2, \dots, \lambda_r\}$ are the eigenvalues of \mathbf{L} less than or equal to ω in magnitude, then any ω -bandlimited signal can be written as a linear combination of the corresponding eigenvectors:

$$\mathbf{f} = \sum_{i=1}^r \tilde{\mathbf{f}}_i \mathbf{u}^i = \mathbf{U}_{\mathcal{V}\mathcal{R}} \tilde{\mathbf{f}}_{\mathcal{R}}, \quad (2.2)$$

where $\mathcal{R} = \{1, \dots, r\}$. The space of ω -bandlimited signals is called Paley-Wiener space and is denoted by $PW_\omega(G)$ [67]. Note that $PW_\omega(G) = \text{range}(\mathbf{U}_{\mathcal{V}\mathcal{R}})$ (i.e., the span of columns of $\mathbf{U}_{\mathcal{V}\mathcal{R}}$). Bandwidth of a signal \mathbf{f} is defined as the largest among absolute values of eigenvalues corresponding to non-zero GFT coefficients of \mathbf{f} , i.e.,

$$\omega(\mathbf{f}) \triangleq \max_i \{|\lambda_i| \mid \tilde{\mathbf{f}}_i \neq 0\}. \quad (2.3)$$

A key ingredient in our theory is an approximation of the bandwidth of a signal using powers of the variation operator \mathbf{L} , as explained in Section 3.3. This approximation holds for all of the variation operators defined in the next section on both undirected and directed graphs. Therefore, the proposed theory remains valid for GFTs based on any of these operators.

2.3 Examples of Variation Operators

2.3.1 Variation on Undirected Graphs

In undirected graphs, the most commonly used variation operator is the combinatorial Laplacian [16] given by:

$$\mathbf{L} = \mathbf{D} - \mathbf{W}, \quad (2.4)$$

where \mathbf{D} is the diagonal degree matrix $\text{diag}\{d_1, \dots, d_N\}$ with $d_i = \sum_j w_{ij}$. Since, $w_{ij} = w_{ji}$ for undirected graphs, this matrix is symmetric. As a result, it has

real non-negative eigenvalues $\lambda_i \geq 0$ and an orthogonal set of eigenvectors. The variation functional associated with this operator is known as the *graph Laplacian quadratic form* [79] given by:

$$\text{Var}_{QF}(\mathbf{f}) = \mathbf{f}^\top \mathbf{L} \mathbf{f} = \frac{1}{2} \sum_{i,j} w_{ij} (f_i - f_j)^2. \quad (2.5)$$

One can normalize the combinatorial Laplacian to obtain the symmetric normalized Laplacian and the (asymmetric) random walk Laplacian given as

$$\mathbf{L}_{sym} = \mathbf{D}^{-1/2} \mathbf{L} \mathbf{D}^{-1/2}, \quad \mathbf{L}_{rw} = \mathbf{D}^{-1} \mathbf{L}. \quad (2.6)$$

Both \mathbf{L}_{sym} and \mathbf{L}_{rw} have non-negative eigenvalues. However, the eigenvectors of \mathbf{L}_{rw} are not orthogonal as \mathbf{L}_{rw} is not symmetric. The eigenvectors of \mathbf{L}_{sym} , on the other hand, are orthogonal. The variation functional associated with \mathbf{L}_{sym} has a nice interpretation as it normalizes the signal values on the nodes by the degree:

$$\text{Var}_{QF_{sym}}(\mathbf{f}) = \mathbf{f}^\top \mathbf{L}_{sym} \mathbf{f} = \frac{1}{2} \sum_{i,j} w_{ij} \left(\frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}} \right)^2. \quad (2.7)$$

2.3.2 Variation on Directed Graphs

Note that variation operators defined for directed graphs can also be used for undirected graphs since each undirected edge can be thought of as two oppositely pointing directed edges.

Variation using the adjacency matrix This approach involves considering the adjacency matrix as a *shift operator* over the graph (see [73] for details). For any signal $\mathbf{f} \in \mathbb{R}^N$, the signal $\mathbf{W}\mathbf{f}$ is considered as a shifted version of \mathbf{f} over the graph, analogous to the shift operation defined in digital signal processing. Using this analogy, [73] defines *total variation* of a signal \mathbf{f} on the graph as

$$\text{Var}_{TV}^p(\mathbf{f}) = \left\| \mathbf{f} - \frac{1}{|\mu_{\max}|} \mathbf{W}\mathbf{f} \right\|_p, \quad (2.8)$$

where $p = 1, 2$ and μ_{\max} denotes the eigenvalue of \mathbf{W} with the largest magnitude. It can be shown that for two eigenvalues $|\mu_i| < |\mu_j|$ of \mathbf{W} , the corresponding

eigenvectors \mathbf{v}_i and \mathbf{v}_j satisfy $\text{Var}_{TV}^p(\mathbf{v}^i) < \text{Var}_{TV}^p(\mathbf{v}^j)$. In order to be consistent with our convention, one can define the variation operator as $\mathbf{L} = \mathbf{I} - \mathbf{W}/|\mu_{\max}|$ which has the same eigenvectors as \mathbf{W} with eigenvalues $\lambda_i = 1 - \mu_i/|\mu_{\max}|$. This allows us to have the same ordering for the graph frequencies and the variations in the basis vectors. Note that for directed graphs, where \mathbf{W} is not symmetric, the GFT basis vectors will not be orthogonal. Further, for some adjacency matrices, there may not exist a complete set of linearly independent eigenvectors. In such cases, one can use generalized eigenvectors in the Jordan normal form of \mathbf{W} as stated before [73].

Variation using the hub-authority model This notion of variation is based on the hub-authority model [49] for specific directed graphs such as a hyperlinked environment (e.g., the web). This model distinguishes between two types of nodes. Hub nodes, subset \mathcal{H} , are nodes that point to other nodes, whereas authority nodes, \mathcal{A} , are the nodes to which other nodes point. Note that a node can be both a hub and an authority simultaneously. In a directed network, we need to define two kinds of degrees for each node $i \in \mathcal{V}$, namely the in-degree $p_i = \sum_j w_{ji}$ and the out-degree $q_i = \sum_j w_{ij}$. The co-linkage between two authorities $i, j \in \mathcal{A}$ or two hubs $i, j \in \mathcal{H}$ is defined as:

$$c_{ij} = \sum_{h \in \mathcal{H}} \frac{w_{hi}w_{hj}}{q_h} \quad \text{and} \quad c_{ij} = \sum_{a \in \mathcal{A}} \frac{w_{ia}w_{ja}}{p_a}, \quad (2.9)$$

respectively, and can be thought of as a cumulative link weight between two authorities (or hubs). Based on this, one can define a variation functional for a signal \mathbf{f} on the authority nodes [91] as:

$$\text{Var}_{\mathcal{A}}(\mathbf{f}) = \frac{1}{2} \sum_{i,j \in \mathcal{A}} c_{ij} \left(\frac{f_i}{\sqrt{p_i}} - \frac{f_j}{\sqrt{p_j}} \right)^2. \quad (2.10)$$

In order to write the above functional in matrix form, define $\mathbf{T} = \mathbf{D}_q^{-1/2} \mathbf{W} \mathbf{D}_p^{-1/2}$, where $\mathbf{D}_p^{-1/2}$ and $\mathbf{D}_q^{-1/2}$ are diagonal matrices with

$$(\mathbf{D}_p^{-1/2})_{ii} = \begin{cases} \frac{1}{\sqrt{p_i}} & \text{if } p_i \neq 0 \\ 0 & \text{otherwise,} \end{cases} \quad (\mathbf{D}_q^{-1/2})_{ii} = \begin{cases} \frac{1}{\sqrt{q_i}} & \text{if } q_i \neq 0 \\ 0 & \text{otherwise.} \end{cases}$$

Table 2.1: Different choices of the variation operator \mathbf{L} for defining GFT bases.

Operator	Expression	Graph type	Variation functional	Properties
Combinatorial	$\mathbf{L} = \mathbf{D} - \mathbf{W}$	Undirected	$\mathbf{f}^\top \mathbf{L} \mathbf{f} = \frac{1}{2} \sum_{i,j} w_{ij} (\mathbf{f}_i - \mathbf{f}_j)^2$	Symmetric, $\lambda_i \geq 0$, \mathbf{U} orthogonal
Symmetric normalized	$\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{W} \mathbf{D}^{-1/2}$	Undirected	$\mathbf{f}^\top \mathbf{L} \mathbf{f} = \frac{1}{2} \sum_{i,j} w_{ij} \left(\frac{\mathbf{f}_i}{\sqrt{d_i}} - \frac{\mathbf{f}_j}{\sqrt{d_j}} \right)^2$	Symm. , $\lambda_i \in [0, 2]$, \mathbf{U} orthogonal
Random walk (undirected)	$\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1} \mathbf{W}$	Undirected	$\ \mathbf{L} \mathbf{f}\ $	Asymmetric, $\lambda_i \geq 0$, \mathbf{U} non-orthogonal
Adjacency-based	$\mathbf{L} = \mathbf{I} - \frac{1}{\mu_{\max}} \mathbf{W}$, μ_{\max} : maximum eigenvalue of \mathbf{W}	Undirected/ Directed	$\ \mathbf{L} \mathbf{f}\ _p, p = 1, 2$	Asym., non-orthog. \mathbf{U} for directed graphs, $\text{Re}\{\lambda_i\} \geq 0$
Hub-authority	$\mathbf{L} = \gamma(\mathbf{I} - \mathbf{T}^\top \mathbf{T}) + (1 - \gamma)(\mathbf{I} - \mathbf{T} \mathbf{T}^\top)$, $\mathbf{T} = \mathbf{D}_p^{-1/2} \mathbf{W} \mathbf{D}_q^{-1/2}$, $\mathbf{D}_p = \text{diag}\{p_i\}$, $p_i = \sum_j w_{ji}$, $\mathbf{D}_q = \text{diag}\{q_i\}$, $q_i = \sum_j w_{ij}$	Directed	$\mathbf{f}^\top \mathbf{L} \mathbf{f}$, see text	Symmetric, $\lambda_i \geq 0$, \mathbf{U} orthogonal
Random walk (directed)	$\mathbf{L} = \frac{1}{2} (\mathbf{\Pi}^{1/2} \mathbf{P} \mathbf{\Pi}^{-1/2} + \mathbf{\Pi}^{-1/2} \mathbf{P}^\top \mathbf{\Pi}^{1/2})$, $\mathbf{P}_{ij} = w_{ij} / \sum_j w_{ij}$, $\mathbf{\Pi} = \text{diag}\{\pi_i\}$	Directed	$\mathbf{f}^\top \mathbf{L} \mathbf{f} = \frac{1}{2} \sum_{i,j} \pi_i \mathbf{P}_{ij} \left(\frac{\mathbf{f}_i}{\sqrt{\pi_i}} - \frac{\mathbf{f}_j}{\sqrt{\pi_j}} \right)^2$	Symmetric, $\lambda_i \geq 0$, \mathbf{U} orthogonal

It is possible to show that $\text{Var}_{\mathcal{A}}(\mathbf{f}) = \mathbf{f}^\top \mathbf{L}_{\mathcal{A}} \mathbf{f}$, where $\mathbf{L}_{\mathcal{A}} = \mathbf{I} - \mathbf{T}^\top \mathbf{T}$ [91]. A variation functional for a signal \mathbf{f} on the hub nodes can be defined in the same way as (2.10) and can be written in a matrix form as $\text{Var}_{\mathcal{H}}(\mathbf{f}) = \mathbf{f}^\top \mathbf{L}_{\mathcal{H}} \mathbf{f}$, where $\mathbf{L}_{\mathcal{H}} = \mathbf{I} - \mathbf{T} \mathbf{T}^\top$. A convex combination $\text{Var}_{\gamma}(\mathbf{f}) = \gamma \text{Var}_{\mathcal{A}}(\mathbf{f}) + (1 - \gamma) \text{Var}_{\mathcal{H}}(\mathbf{f})$, with $\gamma \in [0, 1]$, can be used to define a variation functional for \mathbf{f} on the whole vertex set \mathcal{V} . Note that the corresponding variation operator $\mathbf{L}_{\gamma} = \gamma \mathbf{L}_{\mathcal{A}} + (1 - \gamma) \mathbf{L}_{\mathcal{H}}$ is symmetric and positive semi-definite. Hence, eigenvectors and eigenvalues of \mathbf{L}_{γ} can be used to define an orthogonal GFT similar to the undirected case, where the variation in the eigenvector increases as the corresponding eigenvalue increases.

Variation using the random walk model Every directed graph has an associated random walk with a probability transition matrix \mathbf{P} given by:

$$\mathbf{P}_{ij} = \frac{w_{ij}}{\sum_j w_{ij}}. \quad (2.11)$$

By the Perron-Frobenius theorem, if \mathbf{P} is irreducible then it has a stationary distribution $\boldsymbol{\pi}$ which satisfies $\boldsymbol{\pi} \mathbf{P} = \boldsymbol{\pi}$ [45]. One can then define the following variation functional for signals on directed graphs [17, 92]:

$$\text{Var}_{rw}(\mathbf{f}) = \frac{1}{2} \sum_{i,j} \pi_i \mathbf{P}_{ij} \left(\frac{f_i}{\sqrt{\pi_i}} - \frac{f_j}{\sqrt{\pi_j}} \right)^2. \quad (2.12)$$

Note that if the graph is undirected, the above expression reduces to (2.7) since, in that case, $\pi_i = d_i / \sum_j d_j$. Intuitively, $\pi_i \mathbf{P}_{ij}$ can be thought of as the probability of transition from node i to j in the steady state. We expect it to be large if i is similar to j . Thus, a big difference in signal values on nodes similar to each other contributes more to the variation. A justification for the above functional in terms of generalization of normalized cut to directed graphs is given in [17, 92]. Let $\boldsymbol{\Pi} = \text{diag}\{\pi_1, \dots, \pi_n\}$. Then $\text{Var}_{rw}(\mathbf{f})$ can be written as $\mathbf{f}^\top \mathbf{L} \mathbf{f}$, where

$$\mathbf{L} = \mathbf{I} - \frac{1}{2} \left(\boldsymbol{\Pi}^{1/2} \mathbf{P} \boldsymbol{\Pi}^{-1/2} + \boldsymbol{\Pi}^{-1/2} \mathbf{P}^\top \boldsymbol{\Pi}^{1/2} \right). \quad (2.13)$$

It is easy to see that the above \mathbf{L} is a symmetric positive semi-definite matrix. Therefore, its eigenvectors can be used to define an orthonormal GFT, where the variation in the eigenvector increases as the corresponding eigenvalue increases.

Table 2.1 summarizes different choices of GFT bases based on the above variation operators. Once an appropriate definition of GFT is chosen depending on the application, we can define the concept of filtering in that frequency domain.

2.4 Spectral Filtering

Suppose that the GFT is defined using a variation operator $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$, where $\mathbf{U} = [\mathbf{u}^1, \dots, \mathbf{u}^N]$ is a matrix whose columns are the eigenvectors of \mathbf{L} and $\mathbf{\Lambda} = \text{diag}\{\lambda_1, \dots, \lambda_N\}$ is a diagonal matrix of corresponding eigenvalues. A graph filter is an operator of the form $\mathbf{P} = \mathbf{U}h(\mathbf{\Lambda})\mathbf{U}^{-1}$, where $h : \mathbb{R} \rightarrow \mathbb{R}$ is its spectral response. It acts on an input signal \mathbf{f} by modulating its GFT coefficients $\tilde{\mathbf{f}}$ to obtain the output GFT coefficients $\tilde{y}_l = h(\lambda_l)\tilde{x}_l$, $l = 1, \dots, N$. Taking the inverse GFT of $\tilde{\mathbf{y}}$ gives the output \mathbf{y} ,

$$\mathbf{y} = \mathbf{U}h(\mathbf{\Lambda})\mathbf{U}^{-1}\mathbf{x}. \quad (2.14)$$

A naive application of a graph filter using (2.14) requires explicit computation of eigenvalues and eigenvectors of \mathbf{L} which has a computational complexity of $O(N^3)$ and space complexity of $O(N^2)$. This may not be practical for graphs with very large number of nodes N , which commonly arise in applications.

Fortunately these graphs are often sparse, i.e., the number of edges in the graph is of the same order as the number of nodes, $|\mathcal{E}| = O(N)$. Therefore, the number of non-zero entries in \mathbf{L} is also $O(N)$. In such cases there exists a much more efficient way for approximate application of a graph filter [42].

2.4.1 Polynomial Approximation for Fast Filtering

In order to circumvent eigen-decomposition of \mathbf{L} for filtering, we approximate the spectral response $h(\lambda)$ of the filter \mathbf{P} by a polynomial $h^{\text{poly}}(\lambda) = \sum_{j=1}^k a_j \lambda^j$. This allows us to represent the filter as

$$\mathbf{P} \approx \mathbf{P}^{\text{poly}} = \sum_{j=1}^k a_j \mathbf{L}^j. \quad (2.15)$$

Computing $\mathbf{P}^{\text{poly}}\mathbf{x}$ involves accessing \mathbf{L} only through matrix-vector multiplications. If \mathbf{L} is sparse, then each of these multiplications can be implemented with

a complexity that is linear in N . Furthermore, it can be shown that a degree k polynomial filter is k -hop localized [60], i.e., for any two nodes u, v if v is not in the k -hop neighborhood of u then $\mathbf{P}^{\text{poly}}(u, v) = 0$.

We approximate $h(\lambda)$ with the truncated Chebyshev polynomials in the interval $[0, \lambda_N]$ as proposed in [42]. Chebyshev polynomials are a good proxy for the mini-max polynomials that minimize a bound on $\|\mathbf{P} - \mathbf{P}^{\text{poly}}\|$. If the spectral response to be approximated is not a continuous function of λ , then it is better to approximate it with a smooth, continuous function first, in order to ensure a good polynomial approximation.

2.5 Summary

In this chapter, we formally defined graph signals and their spectral representation. We introduced variation operators which allow us to measure the variation in a signal with respect to the connectivity information given by the graph. We gave several examples of variation operators for both undirected and directed graphs. We defined the graph Fourier transform (GFT) using eigenvalues and eigenvectors of these operators. Based on the GFT, we defined the filtering operation for graph signals and described an efficient way to implement it. The framework of sampling theory developed in subsequent chapters is built on the basics discussed in this chapter.

Chapter 3

Sampling Set Selection and Reconstruction of Graph Signals

In this chapter, we begin our study of sampling theory in earnest. As explained in Chapter 2, the smoothness assumption on a graph signal is formalized in terms of bandlimitedness in the GFT domain. Based on this signal model, we consider the following questions:

1. Given a subset of nodes \mathcal{S} ($\subseteq \mathcal{V}$) to be sampled, what is the maximum bandwidth that a signal \mathbf{f} can have so that it can be uniquely and stably reconstructed from its samples $\mathbf{f}_{\mathcal{S}}$?
2. Given the signal bandwidth, what is the best subset of nodes to be sampled for a unique and stable reconstruction?

Stability is an important issue in the choice of sampling set. In practice, signals are only approximately bandlimited and/or samples are noisy. Therefore, selecting a good sampling set, which makes the resulting reconstructions robust against noise and model mismatch, is very important. We also consider the problem of finding a bandlimited reconstruction of a graph signal using observed samples efficiently.

Most recent approaches for formulating a sampling theory for graph signals involve computing a portion of the graph Fourier basis. However, as discussed in Chapter 2, when the graph of interest is large, computing and storing multiple eigenvectors of its variation operator, \mathbf{L} , increases the numerical complexity and memory requirement significantly. Therefore, it is desirable to have GFT-free

Work in this chapter was published in [1, 2, 58].

methods for sampling set selection and reconstruction that access \mathbf{L} only through matrix-vector multiplication. To achieve this, we define *graph spectral proxies* based on powers of the variation operator in order to approximate the bandwidth of graph signals. These proxies can be computed using localized operations in a distributed fashion with minimal storage cost, thus forming the key ingredient of our approach. Using these proxies, we give an approximate bound on the maximum bandwidth of graph signals (cutoff frequency) that guarantees unique reconstruction with the given samples. We show that this bound also gives us a measure of reconstruction stability for a given sampling set. We formulate the problem of optimizing the sampling set by greedily adding nodes to the sampling set in order to maximize the bound. of given size. We also provide an efficient, iterative bandlimited reconstruction algorithm using polynomial filters defined in Chapter 2. Thus, our formulation, despite being spectrally motivated, is GFT-free.

The rest of this chapter is organized as follows: Section 3.1 reviews some of the prior work on sampling set selection for graph signal reconstruction. In Section 3.2, we consider the problems of uniqueness and stability of bandlimited reconstruction and sampling set selection, assuming that the GFT basis is known explicitly. Section 3.3 addresses these problems using graph spectral proxies. The problem of GFT-free bandlimited reconstruction using observed samples is considered in Section 3.4. In Section 3.5, we discuss the time and space complexity of the proposed algorithms. The effectiveness of our approach is demonstrated through numerical experiments in Section 3.6. We conclude this chapter in Section 3.7.

3.1 Related Work

Sampling theory for graph signals was first studied in [67], where a sufficient condition for unique recovery of bandlimited signals is stated for a given sampling set. The necessary and sufficient condition for uniqueness of bandlimited reconstruction with given sampling set are also given in [77, 14] in a form that assumes that the GFT basis is explicitly known. Previous methods for sampling set selection in graphs can be classified into two types, spectral-domain methods and vertex-domain methods, which are summarized below.

Spectral-domain approaches

Spectral-domain approaches use set selection criteria that are motivated by the bandlimited signal model. For example, the work of [77] requires computation and processing of the first r eigenvectors of the graph Laplacian to construct a sampling set that guarantees unique (but not necessarily stable) reconstruction for a signal spanned by those eigenvectors. Similarly, a greedy algorithm for selecting stable sampling sets for a given bandlimited space is proposed in [14]. It considers a spectral-domain criterion, using minimum singular values of submatrices of the graph Fourier transform matrix, to minimize the effect of sample noise in the worst case. It is also possible to generalize this approach using ideas from the theory of optimal experiment design [47] to select sampling sets that minimize different measures of reconstruction error when the samples are noisy. Greedy algorithms can then be used to find sets which are approximately optimal with respect to these criteria.

Vertex-domain approaches

Alternative approaches to sampling set selection do not consider graph spectral information and instead rely on vertex-domain characteristics. Examples include [59] and [63], which select sampling sets based on maximum graph cuts and spanning trees, respectively. However, these methods are better suited for designing downsampling operators required in bipartite graph multiresolution transforms [60, 61]. Specifically, they do not consider the issue of optimality of sampling sets in terms of quality of bandlimited reconstruction. Further, it can be shown that the maximum graph-cut based sampling set selection criterion is closely related to a special case of our proposed approach.

3.2 Sampling Theory for Graph Signals

In this section, we address the issue of uniqueness and stability of bandlimited graph signal reconstruction and discuss different optimality criteria for sampling set selection assuming that the GFT basis is known explicitly. The uniqueness conditions in this section are equivalent to the ones in [77, 14, 24]. However, the specific form in which we present these conditions lets us give a GFT-free definition of cutoff frequency. This, together with the spectral proxies defined

later in Section 3.3, allows us to circumvent the explicit computation of the graph Fourier basis to ensure uniqueness and find a good sampling set.

The results in this section are useful when the graphs under consideration are small and thus computing the spectrum of their variation operators is computationally feasible. They also serve as a guideline for tackling the aforementioned questions when the graphs are large and computation and storage of the graph Fourier basis is impractical.

3.2.1 Uniqueness of Reconstruction

In order to give a necessary and sufficient condition for unique identifiability of any bandlimited signal $\mathbf{f} \in PW_\omega(G)$ (defined in Section 2.2) from its samples $\mathbf{f}_\mathcal{S}$ on the sampling set \mathcal{S} , we first state the concept of uniqueness set [67].

Definition 1 (Uniqueness set). *A subset of nodes \mathcal{S} is a uniqueness set for the space $PW_\omega(G)$ iff $\mathbf{x}_\mathcal{S} = \mathbf{y}_\mathcal{S}$ implies $\mathbf{x} = \mathbf{y}$ for all $\mathbf{x}, \mathbf{y} \in PW_\omega(G)$.*

Unique identifiability requires that no two bandlimited signals have the same samples on the sampling set as ensured by the following theorem [1].

Theorem 1 (Unique sampling). *\mathcal{S} is a uniqueness set for $PW_\omega(G)$ if and only if $PW_\omega(G) \cap L_2(\mathcal{S}^c) = \{\mathbf{0}\}$.*

Proof. Given $PW_\omega(G) \cap L_2(\mathcal{S}^c) = \{\mathbf{0}\}$, assume that \mathcal{S} is not a uniqueness set. Then, there exist $\mathbf{f}, \mathbf{g} \in PW_\omega(G), \mathbf{f} \neq \mathbf{g}$ such that $\mathbf{f}_\mathcal{S} = \mathbf{g}_\mathcal{S}$. Hence, we have $\mathbf{f} - \mathbf{g} \in L_2(\mathcal{S}^c), \mathbf{f} - \mathbf{g} \neq \mathbf{0}$. Also, $\mathbf{f} - \mathbf{g} \in PW_\omega(G)$ due to closure. But this is a contradiction as $PW_\omega(G) \cap L_2(\mathcal{S}^c) = \{\mathbf{0}\}$. Therefore, \mathcal{S} must be a uniqueness set.

Conversely, we are given that \mathcal{S} is a uniqueness set. Let ϕ be any signal in $PW_\omega(G) \cap L_2(\mathcal{S}^c)$. Then, for any $\mathbf{f} \in PW_\omega(G)$, we have $\mathbf{g} = \mathbf{f} + \phi \in PW_\omega(G)$ and $\mathbf{f}(\mathcal{S}) = \mathbf{g}(\mathcal{S})$. But since \mathcal{S} is a uniqueness set, one must have $\mathbf{f} = \mathbf{g}$, which implies $\phi = \mathbf{0}$. Therefore, $PW_\omega(G) \cap L_2(\mathcal{S}^c) = \{\mathbf{0}\}$. \square

Let \mathbf{S} be a matrix whose columns are indicator functions for nodes in \mathcal{S} . Note that $\mathbf{S}^\top : \mathbb{R}^N \rightarrow \mathbb{R}^{|\mathcal{S}|}$ is the sampling operator with $\mathbf{S}^\top \mathbf{f} = \mathbf{f}_\mathcal{S}$. Theorem 1 essentially states that no signal in $PW_\omega(G)$ is in the null space $\mathcal{N}(\mathbf{S}^\top)$ of the sampling operator. Let $\lambda_1 \leq \dots \leq \lambda_N$ be the graph frequencies corresponding to the inverse GFT matrix $\mathbf{U} = [\mathbf{u}^1, \dots, \mathbf{u}^N]$. Define $\mathcal{R} = \{1, \dots, r\}$, where λ_r is the largest graph frequency less than ω . Any $\mathbf{f} \in PW_\omega(G)$ can be written as

$\mathbf{f} = \mathbf{U}_{\mathcal{V}\mathcal{R}}\mathbf{c}$. Thus, for unique sampling of any signal in $PW_\omega(G)$ on \mathcal{S} , we need $\mathbf{S}^\top \mathbf{U}_{\mathcal{V}\mathcal{R}}\mathbf{c} = \mathbf{U}_{\mathcal{S}\mathcal{R}}\mathbf{c} \neq \mathbf{0} \ \forall \mathbf{c} \neq \mathbf{0}$. This observation leads to the following corollary (which is also given in [13]).

Corollary 1. *\mathcal{S} is a uniqueness set for $PW_\omega(G)$ iff $\mathbf{U}_{\mathcal{S}\mathcal{R}}$ has full column rank.*

If $\mathbf{U}_{\mathcal{S}\mathcal{R}}$ has a full column rank, then a unique reconstruction $\hat{\mathbf{f}} \in PW_\omega(G)$ can be obtained by finding the unique least squares solution to $\mathbf{f}_\mathcal{S} = \mathbf{U}_{\mathcal{S}\mathcal{R}}\mathbf{c}$:

$$\hat{\mathbf{f}} = \mathbf{U}_{\mathcal{V}\mathcal{R}}\mathbf{U}_{\mathcal{S}\mathcal{R}}^+\mathbf{f}_\mathcal{S}, \quad (3.1)$$

where $\mathbf{U}_{\mathcal{S}\mathcal{R}}^+ = (\mathbf{U}_{\mathcal{S}\mathcal{R}}^\top \mathbf{U}_{\mathcal{S}\mathcal{R}})^{-1} \mathbf{U}_{\mathcal{S}\mathcal{R}}^\top$ is the Moore-Penrose pseudo-inverse of $\mathbf{U}_{\mathcal{S}\mathcal{R}}$. The above reconstruction formula is also known as consistent reconstruction [24] since it keeps the observed samples unchanged¹, i.e., $\hat{\mathbf{f}}_\mathcal{S} = \mathbf{f}_\mathcal{S}$. Moreover, it is easy to see that if the original signal $\mathbf{f} \in PW_\omega(G)$, then $\hat{\mathbf{f}} = \mathbf{f}$.

3.2.2 Issue of Stability and Choice of Sampling set

Note that selecting a sampling set \mathcal{S} for $PW_\omega(G)$ amounts to selecting a set of rows of $\mathbf{U}_{\mathcal{V}\mathcal{R}}$. It is always possible to find a sampling set of size $r = \dim(PW_\omega(G))$ which uniquely samples the signals in $PW_\omega(G)$ as proven below.

Proposition 1. *For any $PW_\omega(G)$, there always exists a uniqueness set \mathcal{S} of size $|\mathcal{S}| = r$, where $r = \dim(PW_\omega(G))$.*

Proof. Since $\{\mathbf{u}^i\}_{i=1}^r$ are linearly independent, the matrix $\mathbf{U}_{\mathcal{V}\mathcal{R}}$ has full column rank equal to r . Further, since the row rank of a matrix equals its column rank, we can always find a linearly independent set \mathcal{S} of r rows such that $\mathbf{U}_{\mathcal{S}\mathcal{R}}$ has full rank that equals r , thus proving our claim. \square

In most cases picking r nodes randomly gives a full rank $\mathbf{U}_{\mathcal{S}\mathcal{R}}$. However, all sampling sets of given size are not equally good. A bad choice of \mathcal{S} can give an ill-conditioned $\mathbf{U}_{\mathcal{S}\mathcal{R}}$ which in turn leads to an unstable reconstruction $\hat{\mathbf{f}}$. Stability of reconstruction is important when the true signal \mathbf{f} is only approximately bandlimited (which is the case for most signals in practice) or when the samples are

¹Existence of a sample consistent reconstruction in $PW_\omega(G)$ requires that $PW_\omega(G) \oplus L_2(\mathcal{S}^c) = \mathbb{R}^N$ [24].

noisy. The reconstruction error in this case depends on the noise and the model mismatch as well as the choice of sampling set. The best sampling set achieves the smallest reconstruction error.

Effect of noise

We first consider the case when the observed samples are noisy. Let $\mathbf{f} \in PW_\omega(G)$ be the true signal and $\mathbf{n} \in \mathbb{R}^{|S|}$ be the noise introduced during sampling. The observed samples are then given by $\mathbf{y}_S = \mathbf{f}_S + \mathbf{n}$. Using (3.1), we get the following reconstruction

$$\hat{\mathbf{f}} = \mathbf{U}_{\mathcal{V}\mathcal{R}} \mathbf{U}_{\mathcal{S}\mathcal{R}}^+ \mathbf{f}_S + \mathbf{U}_{\mathcal{V}\mathcal{R}} \mathbf{U}_{\mathcal{S}\mathcal{R}}^+ \mathbf{n}. \quad (3.2)$$

Since $\mathbf{f} \in PW_\omega(G)$, $\mathbf{U}_{\mathcal{V}\mathcal{R}} \mathbf{U}_{\mathcal{S}\mathcal{R}}^+ \mathbf{f}_S = \mathbf{f}$. The reconstruction error equals $\mathbf{e} = \hat{\mathbf{f}} - \mathbf{f} = \mathbf{U}_{\mathcal{V}\mathcal{R}} \mathbf{U}_{\mathcal{S}\mathcal{R}}^+ \mathbf{n}$. If we assume that the entries of \mathbf{n} are i.i.d. with zero mean and unit variance, then the covariance matrix of the reconstruction error is given by

$$\mathbf{E} = \mathbb{E}[\mathbf{e}\mathbf{e}^\top] = \mathbf{U}_{\mathcal{V}\mathcal{R}} (\mathbf{U}_{\mathcal{S}\mathcal{R}}^\top \mathbf{U}_{\mathcal{S}\mathcal{R}})^{-1} \mathbf{U}_{\mathcal{V}\mathcal{R}}^\top. \quad (3.3)$$

Different costs can be defined to measure the reconstruction error as a function of the error covariance matrix. These cost functions are based on optimal design of experiments [6]. If we define the optimal sampling set \mathcal{S}^{opt} of size m , as the set that minimizes the mean squared error, then

$$\mathcal{S}^{\text{A-opt}} = \arg \min_{|\mathcal{S}|=m} \text{Tr}[\mathbf{E}] = \text{Tr}[(\mathbf{U}_{\mathcal{S}\mathcal{R}}^\top \mathbf{U}_{\mathcal{S}\mathcal{R}})^{-1}]. \quad (3.4)$$

This is analogous to the so-called *A*-optimal design. Similarly, minimizing the maximum eigenvalue of the error covariance matrix leads to the *E*-optimal design. The optimal sampling set with this criterion is given by

$$\mathcal{S}^{\text{E-opt}} = \arg \min_{|\mathcal{S}|=m} \lambda_{\max}(\mathbf{E}) = \arg \max_{|\mathcal{S}|=m} \sigma_{\min}(\mathbf{U}_{\mathcal{S}\mathcal{R}}), \quad (3.5)$$

where $\sigma_{\min}(\cdot)$ denotes the smallest singular value of a matrix. This can be thought of as a sampling set that minimizes the worst case reconstruction error. Note that the above criterion is equivalent to the one proposed in [14]. Both *A*- and *E*-optimality criteria lead to combinatorial problems, but it is possible to develop greedy approximate solutions to these problems.

So far we assumed that the true signal $\mathbf{f} \in PW_\omega(G)$ and hence, $\mathbf{U}_{\mathcal{VR}} \mathbf{U}_{\mathcal{SR}}^+ \mathbf{f}_\mathcal{S} = \mathbf{f}$. However, in most applications, the signals are only approximately bandlimited. The reconstruction error in such a case is analyzed next.

Effect of model mismatch

Let $\mathbf{P} = \mathbf{U}_{\mathcal{VR}} \mathbf{U}_{\mathcal{VR}}^\top$ be the projector for $PW_\omega(G)$ and $\mathbf{Q} = \mathbf{S} \mathbf{S}^\top$ be the projector for $L_2(\mathcal{S})$. Assume that the true signal is given by $\mathbf{f} = \mathbf{f}^* + \Delta \mathbf{f}$, where $\mathbf{f}^* = \mathbf{P} \mathbf{f}$ is the bandlimited component of the signal and $\Delta \mathbf{f} = \mathbf{P}^\perp \mathbf{f}$ captures the “high-pass component” (i.e., the model mismatch). If we use (3.1) for reconstructing \mathbf{f} , then a tight upper bound on the reconstruction error [24] is given by

$$\|\mathbf{f} - \hat{\mathbf{f}}\| \leq \frac{1}{\sigma_{\min}(\mathbf{U}_{\mathcal{SR}})} \|\Delta \mathbf{f}\|. \quad (3.6)$$

We define an optimal sampling set \mathcal{S}^{opt} of size m for $PW_\omega(G)$ as the set that minimizes the worst case reconstruction error. Thus, to find this set we need to solve a similar problem as (3.5).

The quantity $\sigma_{\min}(\mathbf{U}_{\mathcal{SR}})$ has a nice geometric interpretation. $\sigma_{\min}(\mathbf{U}_{\mathcal{SR}})$ equals the cosine of the maximum angle between subspaces $PW_\omega(G)$ and $L_2(\mathcal{S})$, which is defined as

$$\cos(\theta_{\max}) = \inf_{\mathbf{f} \in PW_\omega(G), \|\mathbf{f}\|=1} \|\mathbf{Q} \mathbf{f}\|. \quad (3.7)$$

$\sigma_{\min}(\mathbf{U}_{\mathcal{SR}}) = \cos(\theta_{\max}) > 0$ when the uniqueness condition in Theorem 1 is satisfied and the error is bounded. Intuitively, the above equation says that for the worst case error to be minimum, the sampling and reconstruction subspaces should be as aligned as possible.

As stated before, the problem of finding \mathcal{S} that maximizes $\sigma_{\min}(\mathbf{U}_{\mathcal{SR}})$ is combinatorial. It is possible to define a greedy algorithm to get an approximate solution [14]. A simple greedy heuristic to approximate \mathcal{S}^{opt} is to perform column-wise Gaussian elimination over $\mathbf{U}_{\mathcal{VR}}$ with partial row pivoting. The indices of the pivot rows in that case form a good estimate of \mathcal{S}^{opt} in practice.

The methods described above require computation of many eigenvectors of the variation operator \mathbf{L} . We circumvent this issue in the next section, by presenting GFT-free techniques that allow us to express the condition for unique bandlimited

reconstruction and methods for sampling set selection via simple operations using the variation operator.

3.3 Sampling Set Selection Using Graph Spectral Proxies

Our proposed approach for sampling set selection is obtained by defining graph spectral proxies based on powers of \mathbf{L} . These spectral proxies do not require eigen-decomposition of \mathbf{L} and still allow us to define a measure of quality of sampling sets. As we will show, a sampling set optimal with respect to these spectral proxies ensures a small reconstruction error bound. The following discussion holds for any choice of the variation operator \mathbf{L} in Table 2.1.

3.3.1 Cutoff Frequency

In order to obtain a measure of quality for a sampling set \mathcal{S} , we first find the cutoff frequency associated with it, which can be defined as the largest frequency ω such that \mathcal{S} is a uniqueness set for $PW_\omega(G)$. It follows from Theorem 1 that, for \mathcal{S} to be a uniqueness set of $PW_\omega(G)$, ω needs to be less than the minimum possible bandwidth that a signal in $L_2(\mathcal{S}^c)$ can have. This would ensure that no signal from $L_2(\mathcal{S}^c)$ can be a part of $PW_\omega(G)$. Thus, the cutoff frequency $\omega_c(\mathcal{S})$ for a sampling set \mathcal{S} can be expressed as:

$$\omega_c(\mathcal{S}) \triangleq \min_{\phi \in L_2(\mathcal{S}^c), \phi \neq \mathbf{0}} \omega(\phi). \quad (3.8)$$

To use the equation above, we first need a tool to approximately compute the bandwidth $\omega(\phi)$ of any given signal ϕ without computing the graph Fourier coefficients explicitly. Our proposed method for bandwidth estimation is based on the following definition:

Definition 2 (Graph Spectral Proxies). *For any signal $\mathbf{f} \neq \mathbf{0}$, we define its k^{th} spectral proxy $\omega_k(\mathbf{f})$ with $k \in \mathbb{Z}^+$ as*

$$\omega_k(\mathbf{f}) \triangleq \left(\frac{\|\mathbf{L}^k \mathbf{f}\|}{\|\mathbf{f}\|} \right)^{1/k}. \quad (3.9)$$

For an operator \mathbf{L} with real eigenvalues and eigenvectors, $\omega_k(\mathbf{f})$ can be shown to increase monotonically with k :

$$\forall \mathbf{f}, k_1 < k_2 \Rightarrow \omega_{k_1}(\mathbf{f}) \leq \omega_{k_2}(\mathbf{f}). \quad (3.10)$$

These quantities are bounded from above, as a result, $\lim_{k \rightarrow \infty} \omega_k(\mathbf{f})$ exists for all \mathbf{f} . Consequently, it is easy to prove that if $\omega(\mathbf{f})$ denotes the bandwidth of a signal \mathbf{f} , then

$$\forall k > 0, \omega_k(\mathbf{f}) \leq \lim_{j \rightarrow \infty} \omega_j(\mathbf{f}) = \omega(\mathbf{f}). \quad (3.11)$$

Note that (3.11) also holds for an asymmetric \mathbf{L} that has complex eigenvalues and eigenvectors. The proofs of (3.10) and (3.11) are provided in Appendix A. These properties give us an important insight: as we increase the value of k , the spectral proxies tend to have a value close to the actual bandwidth of the signal, i.e., they essentially indicate the frequency localization of the signal energy. Therefore, using $\omega_k(\phi)$ as a proxy for $\omega(\phi)$ (i.e., bandwidth of ϕ) is justified and leads us to define the *cut-off frequency estimate of order k* as

$$\Omega_k(\mathcal{S}) \triangleq \min_{\phi \in L_2(\mathcal{S}^c)} \omega_k(\phi) = \min_{\phi \in L_2(\mathcal{S}^c)} \left(\frac{\|\mathbf{L}^k \phi\|}{\|\phi\|} \right)^{1/k}. \quad (3.12)$$

Using the definitions of $\Omega_k(\mathcal{S})$ and $\omega_c(\mathcal{S})$ along with (3.10) and (3.11), we conclude that for any $k_1 < k_2$:

$$\omega_c(\mathcal{S}) \geq \lim_{k \rightarrow \infty} \Omega_k(\mathcal{S}) \geq \Omega_{k_2}(\mathcal{S}) \geq \Omega_{k_1}(\mathcal{S}). \quad (3.13)$$

Using (3.13) and (3.8), we now state the following proposition:

Proposition 2. *For any k , \mathcal{S} is a uniqueness set for $PW_\omega(G)$ if, $\omega < \Omega_k(\mathcal{S})$. $\Omega_k(\mathcal{S})$ can be computed from (3.12) as*

$$\Omega_k(\mathcal{S}) = \left[\min_{\psi} \frac{\psi^t ((\mathbf{L}^\top)^k \mathbf{L}^k)_{\mathcal{S}^c} \psi}{\psi^t \psi} \right]^{1/2k} = (\sigma_{1,k})^{1/2k}, \quad (3.14)$$

where $\sigma_{1,k}$ denotes the smallest eigenvalue of the reduced matrix $((\mathbf{L}^\top)^k \mathbf{L}^k)_{\mathcal{S}^c}$. Further, if $\psi_{1,k}$ is the corresponding eigenvector, and ϕ_k^* minimizes $\omega_k(\phi)$ in (3.12) (i.e., it approximates the smoothest possible signal in $L_2(\mathcal{S}^c)$), then

$$\phi_k^*(\mathcal{S}^c) = \psi_{1,k}, \quad \phi_k^*(\mathcal{S}) = \mathbf{0}. \quad (3.15)$$

We note from (3.13) that in order to get a better estimate of the true cut-off frequency, one simply needs a higher k . Therefore, there is a trade-off between accuracy of the estimate on the one hand, and increased complexity and reduced numerical stability on the other (that arise by taking higher powers of \mathbf{L}). The benefit of using a higher value of k is experimentally demonstrated in Section 3.6 (see also Sections 4.2.2 and 4.4.4). The issue of increase in the complexity due to higher k is discussed in detail in [2].

3.3.2 Best Sampling Set of Given Size

As shown in Proposition 2, $\Omega_k(\mathcal{S})$ is an estimate of the smallest bandwidth that a signal in $L_2(\mathcal{S}^c)$ can have and any signal in $PW_\omega(G)$ is uniquely sampled on \mathcal{S} if $\omega < \Omega_k(\mathcal{S})$. Intuitively, we would like the projection of $L_2(\mathcal{S}^c)$ along $PW_\omega(G)$ to be as small as possible. Based on this intuition, we propose the following optimality criterion for selecting the best sampling set of size m :

$$\mathcal{S}^{\text{opt}} = \arg \max_{|\mathcal{S}|=m} \Omega_k(\mathcal{S}). \quad (3.16)$$

In order to motivate the criterion above, we relate $\Omega_k(\mathcal{S})$ to $\sigma_{\min}(\mathbf{U}_{\mathcal{SR}})$ from (3.5) and (3.6). Let \mathbf{P} denote the projector for $PW_\omega(G)$. Then it can be shown that [50]:

$$\sigma_{\min}(\mathbf{U}_{\mathcal{SR}}) = \inf_{\mathbf{f} \in L_2(\mathcal{S}^c), \|\mathbf{f}\|=1} \|\mathbf{f} - \mathbf{P}\mathbf{f}\| = \sqrt{\sum_{i: \omega < \lambda_i} |\tilde{f}_i^*|^2} \quad (3.17)$$

where $\mathbf{f}^* \in L_2(\mathcal{S}^c)$ is the minimizer of the left hand side and \tilde{f}_i^* denotes its i -th GFT coefficient. Note that $\omega(\mathbf{f}^*) \geq \Omega_k(\mathcal{S})$ since $\Omega_k(\mathcal{S})$ is the smallest bandwidth that any signal in $L_2(\mathcal{S}^c)$ can have. Therefore,

$$\sigma_{\min}(\mathbf{U}_{\mathcal{SR}}) \geq \sqrt{\sum_{i: \omega < \lambda_i \leq \Omega_k(\mathcal{S})} |\tilde{f}_i^*|^2}. \quad (3.18)$$

The equation above shows that maximizing $\Omega_k(\mathcal{S})$ increases the lower bound $\sigma_{\min}(\mathbf{U}_{\mathcal{SR}})$. This is desirable since it leads to reduction in the upper bound on the reconstruction error (see (3.5) and (3.6)).

We now show that $\Omega_k(\mathcal{S})$ also arises in the bound on the reconstruction error when the reconstruction is obtained by variational energy minimization:

$$\hat{\mathbf{f}}_m = \arg \min_{\mathbf{y} \in \mathbb{R}^N} \|\mathbf{L}^m \mathbf{y}\| \text{ subject to } \mathbf{y}_{\mathcal{S}} = \mathbf{f}_{\mathcal{S}}. \quad (3.19)$$

It was shown in [68] that if $\mathbf{f} \in PW_{\omega}(G)$, then the reconstruction error $\|\hat{\mathbf{f}}_m - \mathbf{f}\|/\|\mathbf{f}\|$, for a given m , is upper-bounded by $2(\omega/\Omega_1(\mathcal{S}))^m$. This bound is suboptimal and can be improved by replacing $\Omega_1(\mathcal{S})$ with $\Omega_k(\mathcal{S})$ (which, from (3.13), is at least as large as $\Omega_1(\mathcal{S})$) for any $k \leq m$, as shown in the following theorem:

Theorem 2. *Let $\hat{\mathbf{f}}_m$ be the solution to (3.19) for a signal $\mathbf{f} \in PW_{\omega}(G)$. Then, for any $k \leq m$,*

$$\|\hat{\mathbf{f}}_m - \mathbf{f}\| \leq 2 \left(\frac{\omega}{\Omega_k(\mathcal{S})} \right)^m \|\mathbf{f}\|. \quad (3.20)$$

Proof. Note that $(\hat{\mathbf{f}}_m - \mathbf{f}) \in L_2(\mathcal{S}^c)$. Therefore, from (3.12)

$$\begin{aligned} \|\hat{\mathbf{f}}_m - \mathbf{f}\| &\leq \frac{1}{(\Omega_m(\mathcal{S}))^m} \|\mathbf{L}^m(\hat{\mathbf{f}}_m - \mathbf{f})\| \\ &\leq \frac{1}{(\Omega_m(\mathcal{S}))^m} (\|\mathbf{L}^m \hat{\mathbf{f}}_m\| + \|\mathbf{L}^m \mathbf{f}\|) \end{aligned} \quad (3.21)$$

$$\leq \frac{2}{(\Omega_m(\mathcal{S}))^m} \|\mathbf{L}^m \mathbf{f}\| \quad (3.22)$$

$$\leq 2 \left(\frac{\omega_m(\mathbf{f})}{\Omega_m(\mathcal{S})} \right)^m \|\mathbf{f}\| \quad (3.23)$$

$$\leq 2 \left(\frac{\omega}{\Omega_k(\mathcal{S})} \right)^m \|\mathbf{f}\|.$$

(3.21) follows from triangle inequality. (3.22) holds because $\hat{\mathbf{f}}_m$ minimizes $\|\mathbf{L}^m \hat{\mathbf{f}}_m\|$ over all sample consistent signals. (3.23) follows from the definition of $\omega_m(\mathbf{f})$ and the last step follows from (3.11) and (3.13). \square

Note that for the error bound in (3.20) to go to zero as $m \rightarrow \infty$, ω must be less than $\Omega_k(\mathcal{S})$. Thus, increasing $\Omega_k(\mathcal{S})$ allows us to reconstruct signals in a larger bandlimited space using the variational method. Moreover, for a fixed m and k ,

a higher value of $\Omega_k(\mathcal{S})$ leads to a lower reconstruction error bound. The optimal sampling set $\mathcal{S}_k^{\text{opt}}$ in (3.16) essentially minimizes this error bound.

3.3.3 Finding the Best Sampling Set

The problem posed in (3.16) is combinatorial because we need to compute $\Omega_k(\mathcal{S})$ for every possible subset \mathcal{S} of size m . We use a greedy algorithm to find an approximately optimal set. At each iteration, this algorithm samples one node from the unsampled set that leads to the largest increase in the cutoff frequency estimate $\Omega_k(\mathcal{S})$ (see Algorithm 1). The greedy selection of nodes can be further accelerated using a gradient descent based method described in [1]. Intuitively, the proposed greedy method selects a node farthest from the previously sampled nodes at each iteration so that unsampled nodes are well connected to the sampled nodes (see Section 4.2.2).

One can show that the cutoff frequency estimate $\Omega_k(\mathcal{S})$ associated with a sampling set can only increase (or remain unchanged) when a node is added to it. This is stated more formally in the following proposition.

Proposition 3. *Let \mathcal{S}_1 and \mathcal{S}_2 be two subsets of nodes of G with $\mathcal{S}_1 \subseteq \mathcal{S}_2$. Then $\Omega_k(\mathcal{S}_1) \leq \Omega_k(\mathcal{S}_2)$.*

This turns out to be a straightforward consequence of the eigenvalue interlacing property for symmetric matrices.

Theorem 3 (Eigenvalue interlacing [41]). *Let \mathbf{B} be a symmetric $n \times n$ matrix. Let $\mathcal{R} = \{1, 2, \dots, r\}$, for $1 \leq r \leq n - 1$ and $\mathbf{B}_r = \mathbf{B}_{\mathcal{R}}$. Let $\lambda_k(\mathbf{B}_r)$ be the k -th largest eigenvalue of \mathbf{B}_r . Then the following interlacing property holds:*

$$\lambda_{r+1}(\mathbf{B}_{r+1}) \leq \lambda_r(\mathbf{B}_r) \leq \lambda_r(\mathbf{B}_{r+1}) \leq \dots \leq \lambda_2(\mathbf{B}_{r+1}) \leq \lambda_1(\mathbf{B}_r) \leq \lambda_1(\mathbf{B}_{r+1}).$$

The above theorem implies that if $\mathcal{S}_1 \subseteq \mathcal{S}_2$, then $\mathcal{S}_2^c \subseteq \mathcal{S}_1^c$ and thus, $\lambda_{\min} \left[\left((\mathbf{L}^\top)^k \mathbf{L}^k \right)_{\mathcal{S}_1^c} \right] \leq \lambda_{\min} \left[\left((\mathbf{L}^\top)^k \mathbf{L}^k \right)_{\mathcal{S}_2^c} \right]$.

3.4 GFT-free Bandlimited Reconstruction

Exact bandlimited reconstruction with observed graph signal samples can be obtained using (3.1) if the GFT basis is known. In this section, we present an

Algorithm 1 Greedy heuristic for estimating \mathcal{S}^{opt}

Input: $G = \{\mathcal{V}, E\}$, \mathbf{L} , number of samples m , some $k \in \mathbb{Z}^+$

Initialize: $\mathcal{S} = \{\emptyset\}$.

1: **while** $|\mathcal{S}| < m$ **do**

2: $v \leftarrow \arg \max_{i \in \mathcal{S}^c} \Omega_k(\mathcal{S}_{+i})$, where $\mathcal{S}_{+i} = \mathcal{S} \cup \{i\}$.

3: $\mathcal{S} \leftarrow \mathcal{S} \cup \{v\}$.

4: **end while**

5: $\mathcal{S}^{\text{est}} \leftarrow \mathcal{S}$.

efficient GFT-free method for computing an approximate bandlimited reconstruction. Our formulation begins with an iterative algorithm for exact bandlimited reconstruction that uses an ideal low pass filter in the GFT domain (i.e., a projector of $PW_\omega(G)$). Although the ideal low pass filtering operation requires knowing the GFT basis, it is amenable to approximation using polynomial filters described in Section 2.4. This leads to an iterative GFT-free method for approximate bandlimited reconstruction.

Iterative Algorithm for Exact Bandlimited Reconstruction

The proposed iterative method is based on the Papoulis-Gerchberg algorithm [65, 35, 74] in classical signal processing, which is used to reconstruct a band-limited signal from irregular samples. It is a special case of the projection onto convex sets (POCS) [85] method used to find a point in the intersection of two closed convex sets. The convex sets of interest in the present context are:

$$C_1 = \{\mathbf{x} : \mathbf{S}^\top \mathbf{x} = \mathbf{S}^\top \mathbf{f}\} \quad (3.24)$$

$$C_2 = PW_\omega(G). \quad (3.25)$$

A sample consistent bandlimited reconstruction $\hat{\mathbf{f}}$ lies in the intersection of C_1 and C_2 . The proposed algorithm aims to find this reconstruction by projecting an initial guess alternately on C_1 and C_2 (see Figure 3.1).

The projector for C_1 is the low pass graph filter $\mathbf{P} : \mathbb{R}^N \rightarrow PW_\omega(G)$. It can be written as $\mathbf{P} = \mathbf{U}h(\mathbf{\Lambda})\mathbf{U}^{-1}$, where \mathbf{U} is the inverse GFT matrix, $\mathbf{\Lambda}$ is the diagonal

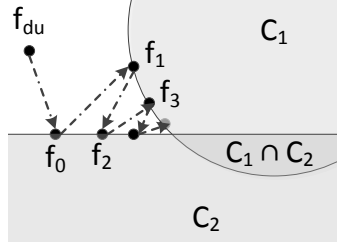


Figure 3.1: Iterative reconstruction using POCS

matrix of corresponding graph frequencies and the spectral response $h : \mathbb{R} \rightarrow \mathbb{R}$ is given by

$$h(\lambda) = \begin{cases} 1 & \text{if } \lambda < \omega \\ 0 & \text{if } \lambda \geq \omega. \end{cases} \quad (3.26)$$

The projector $\mathbf{Q} : \mathbb{R}^N \rightarrow C_2$ for C_2 simply substitutes the samples of any signal \mathbf{x} on \mathcal{S} by $\mathbf{f}_{\mathcal{S}}$. It can be written as

$$\mathbf{Q}\mathbf{x} = \mathbf{x} + \mathbf{S}\mathbf{S}^\top(\mathbf{f}_{du} - \mathbf{x}), \quad (3.27)$$

where $\mathbf{f}_{du} = \mathbf{S}\mathbf{S}^\top\mathbf{f}$ is the graph signal obtained by inserting zeros at the unsampled nodes. With this notation the proposed iterative algorithm can be written as

$$\begin{aligned} \mathbf{f}_0 &= \mathbf{P}\mathbf{f}_{du} \\ \mathbf{f}_{k+1} &= \mathbf{P}\mathbf{Q}\mathbf{f}_k. \end{aligned} \quad (3.28)$$

At each iteration the algorithm resets the signal samples on \mathcal{S} to the actual given samples and then projects the signal onto the low-pass space $PW_\omega(G)$.

Convergence

Let us define an operator $\mathbf{T} = \mathbf{P}\mathbf{Q}$. It has been shown in [85] that an iterative algorithm of the form $\mathbf{x}_{k+1} = \mathbf{T}\mathbf{x}_k$ converges to a fixed point of \mathbf{T} if

1. \mathbf{T} is non-expansive, i.e., $\|\mathbf{T}\mathbf{x} - \mathbf{T}\mathbf{y}\| \leq \|\mathbf{x} - \mathbf{y}\|$
2. \mathbf{T} is asymptotically regular, i.e., $\|\mathbf{T}\mathbf{x}_{k+1} - \mathbf{T}\mathbf{x}_k\| \rightarrow 0$ as $k \rightarrow \infty$.

\mathbf{P} is a bandlimiting operator and hence, is non-expansive. \mathbf{Q} is non expansive because $\|\mathbf{Q}\mathbf{x} - \mathbf{Q}\mathbf{y}\| = \|(\mathbf{I} - \mathbf{S}\mathbf{S}^\top)(\mathbf{x} - \mathbf{y})\| \leq \|\mathbf{x} - \mathbf{y}\|$. Since both \mathbf{P} and \mathbf{Q} are

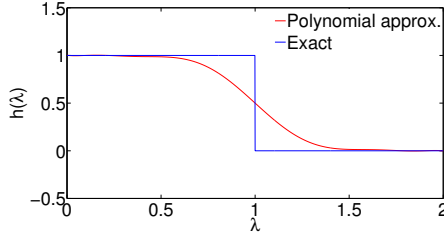


Figure 3.2: Spectral response of an approximate polynomial filter of degree 10. $\omega = 1, \alpha = 8$.

non-expansive, \mathbf{T} is also non-expansive. Asymptotic regularity of \mathbf{T} can also be proved as shown in [74]. Note that if $\hat{\mathbf{f}}$ is a fixed point of \mathbf{T} then $\hat{\mathbf{f}} \in C_1 \cap C_2$. Therefore, the proposed algorithm converges to a sample consistent bandlimited reconstruction (assuming it exists).

Approximate Iterative Reconstruction with Polynomial Filter

Exact computation of \mathbf{P} requires knowing the GFT. In order to circumvent the GFT computation, we approximate \mathbf{P} with a Chebyshev polynomial in \mathbf{L} as explained in Section 2.4. This gives us an approximate and very efficient GFT-free bandlimited reconstruction algorithm². Since a degree k polynomial graph filter is k -hop localized, the proposed algorithm also allows for distributed implementation.

Note that $h(\lambda)$ in (3.26) is not a continuous function of λ . In order to ensure a good Chebyshev polynomial approximation, we can replace $h(\lambda)$ with a smooth, continuous sigmoid-like function (see Figure 3.2)

$$h'(\lambda) = \frac{1}{(1 + \exp(\alpha(\lambda - \omega)))}. \quad (3.29)$$

Due to these approximations in the filter, the reconstructed signal obtained by our method is not exactly bandlimited. However, in many applications (e.g., see Chapter 4) we do not expect the signals to be exactly bandlimited anyway. Thus, using a filter with slowly decaying spectral response may end up improving the result in such applications.

²Further acceleration of the proposed method is possible using the conjugate gradient approach. See [31] for details

3.5 Complexity

We now discuss the time and space complexity of our algorithms. The most complex step in the greedy procedure for maximizing $\Omega_k(\mathcal{S})$ is computing the smallest eigen-pair of $(\mathbf{L}^k)_{\mathcal{S}^c}$ (see [2] for details). This can be accomplished using an iterative Rayleigh-quotient minimization based algorithm. Specifically, the locally-optimal pre-conditioned conjugate gradient (LOPCG) method [51] is suitable for this approach. Note that $(\mathbf{L}^k)_{\mathcal{S}^c}$ can be written as $\mathbf{I}_{\mathcal{S}^c, \mathcal{V}} \mathbf{L} \mathbf{L} \dots \mathbf{L} \mathbf{I}_{\mathcal{V}, \mathcal{S}^c}$, hence the eigenvalue computation can be broken into atomic matrix-vector products: $\mathbf{L}\mathbf{x}$. Typically, the graphs encountered in learning applications are sparse, leading to efficient implementations of $\mathbf{L}\mathbf{x}$. If $|\mathbf{L}|$ denotes the number of non-zero elements in \mathbf{L} , then the complexity of the matrix-vector product is $O(|\mathbf{L}|)$. The complexity of each eigen-pair computation for $(\mathbf{L}^k)_{\mathcal{S}^c}$ is then $O(k|\mathbf{L}|r)$, where r is a constant equal to the average number of iterations required for the LOPCG algorithm (r depends on the spectral properties of \mathbf{L} and is independent of its size $|\mathcal{V}|$). The complexity of the label selection algorithm then becomes $O(k|\mathbf{L}|mr)$, where m is the number of labels requested.

In the iterative reconstruction algorithm, since we use polynomial graph filters, once again the atomic step is the matrix-vector product $\mathbf{L}\mathbf{x}$. The complexity of this algorithm can be given as $O(|\mathbf{L}|pq)$, where p is the order of the polynomial used to design the filter and q is the average number of iterations required for convergence. Again, both these parameters are independent of $|\mathcal{V}|$. Thus, the overall complexity of our algorithm is $O(|\mathbf{L}|(kmr + pq))$. In addition, our algorithm has major advantages in terms of space complexity: since the atomic operation at each step is the matrix-vector product $\mathbf{L}\mathbf{x}$, we only need to store \mathbf{L} and a constant number of vectors. Moreover, the structure of the Laplacian matrix allows one to perform the aforementioned operations in a distributed fashion. This makes it well-suited for large-scale implementations using software packages such as GraphLab [53].

3.6 Experiments

3.6.1 Sampling Set Selection

In this experiment, we numerically evaluate the performance of the proposed sampling set selection. The experiment involves comparing the reconstruction errors of different sampling set selection algorithms in conjunction with exact consistent bandlimited reconstruction obtained using (3.1). We compare our approach with the following methods:

M1: This method [14] uses a greedy algorithm to approximate the \mathcal{S} that maximizes $\sigma_{\min}(\mathbf{U}_{\mathcal{SR}})$. Consistent bandlimited reconstruction (3.1) is then used to estimate the unknown samples.

M2: At each iteration i , this method [77] finds the representation of \mathbf{u}_i as $\sum_{j < i} \beta_j \mathbf{u}_j + \sum_{u \notin \mathcal{S}} \alpha_u \mathbf{1}_u$, where $\mathbf{1}_u$ is the delta function on u . The node v with maximum $|\alpha_v|$ is sampled. Reconstruction is done using (3.1).

Both the above methods assume that a portion of the frequency basis is known and the signal to be recovered is exactly bandlimited. As a baseline, we also compare all sampling set selection methods against uniform random sampling.

We consider an undirected Erdős-Renyi random graph (unweighted) with 1000 nodes and connection probability 0.01. The graph signal to be sampled is exactly bandlimited with $r = \dim PW_{\omega}(G) = 50$ and non-zero GFT coefficients are generated from $\mathcal{N}(1, 0.5^2)$. The samples are noisy with additive iid Gaussian noise such that the SNR equals 20dB. We generate 50 graph signals using the above model, use the sampling sets obtained from all the methods to perform reconstruction and plot the mean of the mean squared error (MSE) for different sizes of sampling sets. For our algorithm, we set the value of k to 2, 8 and 14. The result is illustrated in Figure 3.3. Note that when the size of the sampling set is less than $r = 50$, the reconstruction error is very high. This is expected, because the uniqueness condition is not satisfied by the sampling set. Beyond $|\mathcal{S}| = r$, we observe that our sampling method leads to smaller reconstruction error in most cases. This indicates that our method is robust to noise and model mismatch. Uniform random sampling performs very badly as expected, due to lack of stability considerations. We also observe that using higher values of k in our method leads

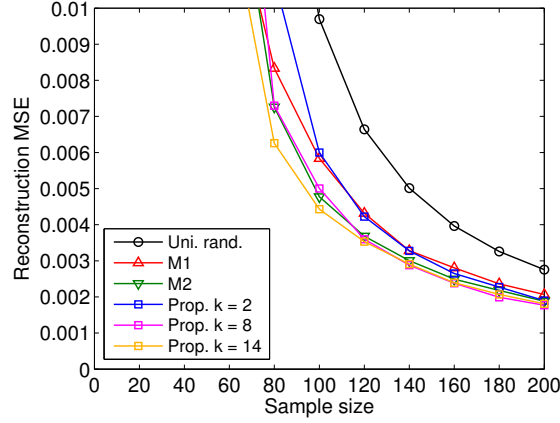


Figure 3.3: Reconstruction MSE vs. number of samples. The large reconstruction errors for $|\mathcal{S}| < 50$ arise due to non-uniqueness of bandlimited reconstruction and hence, are less meaningful.

to better sampling sets. This is because using a higher k allows us to optimize a better estimate of the cutoff frequency.

3.6.2 Efficient Bandlimited Reconstruction

This experiment demonstrates the effectiveness of our proposed bandlimited reconstruction method. We consider the same Erdős-Renyi graph that was used in the previous example. We randomly select 500 nodes in the graph for sampling. The graph signal to be reconstructed is bandlimited with $r = \dim PW_\omega(G) = 100$ and random GFT coefficients drawn from $\mathcal{N}(1, 0.5^2)$. r is chosen such that λ_r is less than the cutoff frequency associated with the sampling set. We obtain reconstructions with the signal samples observed on the subset of the nodes selected using the following methods:

1. POCS using the exact low pass filter.
2. POCS using a polynomial filter of degree 10 that approximates the low pass filter with spectral response (3.29) with $\alpha = 100$.
3. POCS using a similar polynomial filter with degree 40.

Figure 3.4 shows the relative error of reconstruction $\|\mathbf{f} - \hat{\mathbf{f}}_i\|/\|\mathbf{f}\|$ obtained with different number of POCS iterations i , averaged over 10 trials. We observe that our iterative POCS method converges to an exact bandlimited reconstruction if the

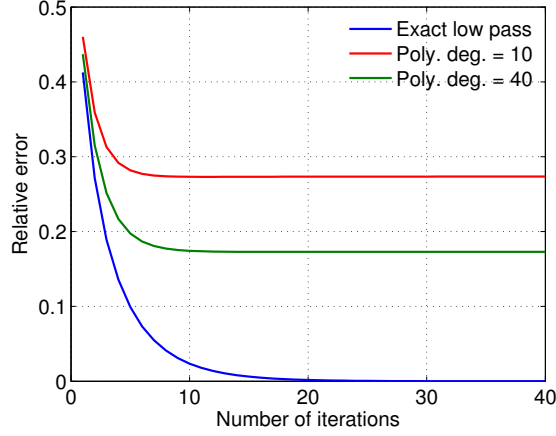


Figure 3.4: Relative error of reconstruction vs. number of POCS iterations using exact low pass filter and polynomial filters of different degrees.

exact low pass filter is used. The GFT-free POCS method using polynomial low pass filters also converges to a fixed point in very few iterations. Although the final reconstruction is not exactly bandlimited, the reconstruction error is small. Using a polynomial filter with higher degree that better approximates the exact low pass filter leads to a smaller reconstruction error at the cost of increased complexity.

3.7 Conclusion

We studied the problem of selecting an optimal sampling set for reconstruction of bandlimited graph signals. The starting point of our framework is the notion of the Graph Fourier Transform (GFT), which is defined via an appropriate variation operator. Our goal is to find good sampling sets for reconstructing signals that are bandlimited in the above frequency domain. We showed that when the samples are noisy or the true signal is only approximately bandlimited, the reconstruction error depends not only on the model mismatch but also on the choice of sampling set. We proposed a measure of quality for the sampling sets, namely the cutoff frequency, that can be computed without finding the GFT basis explicitly. A sampling set that maximizes the cutoff frequency is shown to minimize the reconstruction error. We also proposed a greedy algorithm which finds an approximately optimal set. The proposed algorithm can be efficiently implemented in a distributed and parallel fashion. Together with the proposed localized, bandlimited reconstruction

algorithm, it gives an effective method for sampling and reconstruction of smooth graph signals on large graphs.

The present work opens up some new questions for further research. The problem of finding a sampling set with maximum cutoff frequency is combinatorial. The proposed greedy algorithm gives only an approximate solution to this problem. It would be useful to find a polynomial time algorithm with theoretical guarantees on the quality of approximation.

The proposed set selection method is not adaptive, i.e., the choice of sampling locations does not depend on previously observed samples. This can be a limitation in applications that require batch sampling. In such cases, it would be desirable to have an adaptive sampling set selection scheme which takes into account the previously observed samples to refine the choice of nodes to be sampled in the future. This problem is considered in Chapter 5.

Chapter 4

Active Semi-supervised Learning Using Sampling Theory

4.1 Introduction

In many real-life machine learning tasks, labeled data is scarce whereas unlabeled data is easily available. Active semi-supervised learning is an effective approach for such scenarios. A semi-supervised learning technique must not only learn from the labeled data but also from the inherent clustering present in the unlabeled data [95]. Further, when the labeling is expensive, it is better to let the learner choose the data points to be labeled so that it can pick the most informative and representative labels. Thus, in an active learning scenario, the goal is to achieve the maximum gain in terms of learning ability for a given, and small, number of label queries. In this chapter, we present a novel approach to active semi-supervised learning based on the framework sampling theory of graph signals developed in Chapter 3.

Active learning has been studied in different problem scenarios such as online stream-based sampling, adaptive sampling etc. (see [76] for a review). We focus on the problem of pool-based batch-mode active semi-supervised learning, where there is a large static collection of unlabeled data from which a very small percentage of data points have to be selected in order to be labeled. Batch operation (i.e., selecting a *set* of data points to be labeled) is more realistic in scenarios such as crowd-sourcing where it would not be practical to submit for labeling one data

Work in this chapter was published in part in [30].

point at a time. Therefore, we focus on the problem of optimizing batches of any size without using any label information, which would be the case when selecting the first batch of data points to be labeled. The problem of adaptive sampling, in which the choice of data points to be labeled in the future depends on the labels observed in the past, is considered in Chapter 5.

Applying a graph perspective to semi-supervised learning is not new. In a graph-based formulation, the data points are represented by nodes of a graph and the edges capture the similarity between the nodes they connect. For example, the weight on an edge might be a function of the distance between the two points in the feature space chosen for the classification task. The membership function of a given class can be thought of as a “graph signal”, which has a scalar value at each of the nodes (e.g., 1 or 0 depending on whether or not the data point belongs to the class). Since features have been chosen to be meaningful for the classification task, it is reasonable to expect that nodes that are close together in the feature space are likely to have the same label. Conversely, nodes that are far away in the feature space are less likely to have the same label. Thus, we expect the membership function to be *smooth* on the graph, i.e., moving from a node to its neighbors in the graph is unlikely to lead to changes in the membership. With this in mind, the semi-supervised learning problem can be viewed as a problem of interpolating a smooth graph signal from samples observed on a subset of the nodes. This view has led to many effective techniques such as MinCut [5], Gaussian random fields and harmonic functions [94], local and global consistency [91], manifold regularization [4] and spectral graph kernels [81].

Active learning has also benefited from this graph based-view. Many active learning approaches use a graph to quantify the quality of sampling sets [37, 39, 40]. One methodology is to try and pick a subset of nodes that captures the underlying low-dimensional manifold represented by the graph. Another is to pick the nodes to be labeled in such a way that unlabeled nodes are strongly connected to them. Some methods select those samples which lead to minimization of generalization error bound. We discuss some of these methods in Section 4.3.

Many of the semi-supervised learning methods mentioned above are *global*, in the sense that they require inversion or eigen-decomposition of large matrices associated with the underlying graph. This poses a problem in scalable and distributed implementation of these algorithms. Most graph-based active learning methods suffer from the same problem. Another issue with these methods is that

they do not give conditions under which the graph signal can be uniquely and perfectly interpolated from its samples on the chosen subset.

In this chapter, we show that theoretical results on graph signal sampling and interpolation in Chapter 3 provide a rigorous and unified framework to select points to be labeled and subsequently perform semi-supervised learning. The sampling set selection algorithm presented in Section 3.3 selects a subset of nodes for sampling in order to ensure stable recovery of smooth graph signals. We use this algorithm for choosing the best nodes for labeling (i.e., active learning), since the label signal forms a smooth graph signal. The algorithm also has a compelling graph theoretic interpretation. We also give an effective and efficient semi-supervised label prediction method based on the iterative bandlimited reconstruction algorithm in Section 3.4. Both our algorithms are well-suited for large-scale distributed implementation. We show that our method outperforms several state of the art methods by testing on multiple real datasets.

The rest of the chapter is organized as follows. In Section 4.2, we apply the framework of sampling theory to derive the proposed active semi-supervised learning approach. Section 4.3 summarizes the related prior work. Experiments are presented in Section 4.4. We conclude in Section 4.5 with a brief summary and some remarks.

4.2 Graph Sampling Based Active Semi-Supervised Learning

As noted earlier, if the edges of the graph represent similarity between the nodes, then a graph signal defined using the membership functions of a particular class tends to be smooth. This is illustrated experimentally in Figure 4.1, where the GFT is defined using the symmetric normalized graph Laplacian. In Section 3.3, we showed how to estimate the reconstruction cut-off frequency for a set of vertices. In practice, class membership signals are not strictly bandlimited (see Figure 4.1). Thus, we will be approximating a non-bandlimited signal with one that is bandlimited to the cutoff frequency of the chosen vertex set. We observe in our experiments in Section 4.4 that the easiest task in terms of prediction performance is the one in which the label signal has the highest percentage of energy in the low frequencies. The key idea in our work is that, even though we cannot recover the

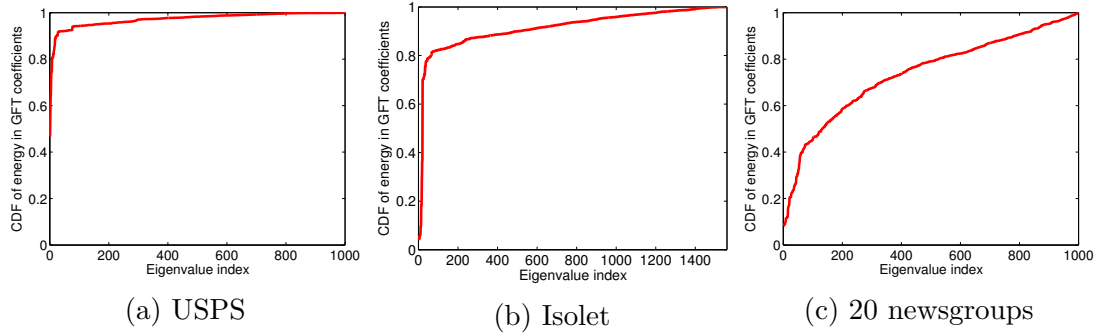


Figure 4.1: Cumulative distribution of energy in the GFT coefficients (with GFT defined using symmetric normalized Laplacian) of one of the class membership functions pertaining to the three real-world dataset experiments considered in Section 4.4. Note that most of the energy is concentrated in the low-pass region.

“true” membership signal exactly from its samples, an active learning approach *should aim at selecting the sampling set with maximum cutoff frequency*. This is because a sampling set with maximum cutoff frequency leads to bandlimited reconstructions with small reconstruction error bound as shown in Section 3.3¹. Also $PW_\omega(G) \subset PW_{\omega'}(G)$ if $\omega \leq \omega'$ and thus, for any signal, its best approximation with a signal from $PW_{\omega'}(G)$ can be no worse (in terms of l_2 error) than its best approximation with a signal from $PW_\omega(G)$.

In this setting, predicting the labels of the unknown data points using the labeled data amounts to reconstructing a bandlimited graph signal from its values on the sampling set. Thus, based on the above reasoning the active learning strategy should be to find a set \mathcal{S} with size equal to a given target number of data points to be labeled, so that the cut-off frequency of \mathcal{S} is maximized.

4.2.1 Proposed method

We now present the details of our method. We target a multi-class active semi-supervised learning problem with C classes. The true membership function for class j is denoted as $\mathbf{f}^j : \mathcal{V} \rightarrow \{0, 1\}$, where $\mathbf{f}^j(i) = 1$ indicates that node i belongs to class j . These membership functions are taken to be the graph signals for our setting. The predicted membership functions for each class take real values and

¹See also Section 5.1 in which we show that a sampling set that maximizes the cutoff frequency leads to the smallest prediction covariance if signals are assumed to follow a Gaussian random field based probabilistic smoothness model.

are denoted as $\hat{\mathbf{f}}^j : \mathcal{V} \rightarrow \mathbb{R}$. The predicted label of node i is given by $\arg \max_j \hat{\mathbf{f}}^j(i)$. We denote the labeled set as \mathcal{S} and the unlabeled set as $\mathcal{S}^c = \mathcal{V} \setminus \mathcal{S}$. Then, our solution to the active semi-supervised learning task can be formally summarized as follows:

1. Given a size m and parameter k , we define the optimal labeled set \mathcal{S}^{opt} as follows:

$$\mathcal{S}^{\text{opt}} = \arg \max_{\mathcal{S}: |\mathcal{S}|=m} \Omega_k(\mathcal{S}) \quad (4.1)$$

We find an approximate solution \mathcal{S}^* to problem above in a greedy fashion by adding a node to \mathcal{S} that leads to maximum the increase in $\Omega_k(\mathcal{S})$ at each step (see Section 3.3).

2. Next, we query the labels of nodes selected in \mathcal{S}^* .
3. Finally, we determine the predicted membership functions $\hat{\mathbf{f}}^j$ for each class from $\mathbf{f}^j(\mathcal{S}^*), j = 1, \dots, C$ using the POCS iterative method described in Section 3.4, where $\mathcal{S} = \mathcal{S}^*$ and $\omega = \Omega_k(\mathcal{S}^*)$ are used in (3.24) and (3.25) to construct the convex sets.

Remarks

In our experiments, we use the gradient descent based approach in [1] for accelerating the greedy node selection method (Algorithm 1) in Section 3.3. This procedure is summarized with Algorithm 2.

Algorithm 2 Greedy heuristic for finding \mathcal{S}^*

Input: $G = \{\mathcal{V}, E\}$, \mathbf{L} , target size m , parameter $k \in \mathbb{Z}^+$.

Initialize: $\mathcal{S} = \{\emptyset\}$.

- 1: **while** $|\mathcal{S}| \leq m$ **do**
 - 2: For \mathcal{S} , compute the smoothest signal $\phi_k^* \in L_2(\mathcal{S}^c)$ using (3.14) and (3.15).
 - 3: $v \leftarrow \arg \max_i [(\phi_k^*(i))^2]$.
 - 4: $\mathcal{S} \leftarrow \mathcal{S} \cup v$.
 - 5: **end while**
 - 6: $\mathcal{S}^* \leftarrow \mathcal{S}$.
-

In POCS reconstruction, we use polynomial filters of degree 10 by first approximating the spectral response of the ideal bandlimiting filter by a sigmoid function (3.29) with $\omega = \Omega_k(\mathcal{S}^*)$ and $\alpha = 8$.

4.2.2 Graph Theoretic Interpretation

In this section, we will provide an intuitive interpretation for our node selection algorithm in terms of connected-ness among the nodes. To simplify the exposition, we consider the maximization problem (4.1) for $k = 1$:

$$\Omega_1(\mathcal{S}) = \inf_{\substack{\mathbf{x}(\mathcal{S})=\mathbf{0} \\ \|\mathbf{x}\|=1}} \mathbf{x}^\top \mathbf{L} \mathbf{x} \quad (4.2)$$

This expression appears more commonly as part of discrete Dirichlet eigenvalue problems on graphs. Specifically, it is equal to the Dirichlet energy of the subset \mathcal{S}^c [16, 64]. The sampling set selection problem seeks to identify the subset \mathcal{S} that maximizes this objective function. To give an intuitive interpretation of our goal, we expand the objective function for any \mathbf{x} with constraint $\mathbf{x}(\mathcal{S}) = \mathbf{0}$ as follows:

$$\begin{aligned} \mathbf{x}^\top \mathbf{L} \mathbf{x} &= \sum_{i \sim j} w_{ij} \left(\frac{x_i}{\sqrt{d_i}} - \frac{x_j}{\sqrt{d_j}} \right)^2 \\ &= \sum_{\substack{i \sim j \\ i \in \mathcal{S}, j \in \mathcal{S}^c}} w_{ij} \left(\frac{x_j^2}{d_j} \right) + \sum_{\substack{i \sim j \\ i, j \in \mathcal{S}^c}} w_{ij} \left(\frac{x_i}{\sqrt{d_i}} - \frac{x_j}{\sqrt{d_j}} \right)^2. \end{aligned} \quad (4.3)$$

The minimizer in the equation above is the first Dirichlet eigenvector which is guaranteed to have strictly positive values on \mathcal{S}^c [64]. Therefore, the contribution of the second term is expected to be negligible compared to that of the first one. Thus, we get

$$\mathbf{x}^\top \mathbf{L} \mathbf{x} \approx \sum_{j \in \mathcal{S}^c} \left(\frac{p_j}{d_j} \right) x_j^2, \quad (4.4)$$

where, $p_j = \sum_{i \in \mathcal{S}} w_{ij}$ is defined as the “partial out-degree” of node $j \in \mathcal{S}^c$, i.e., it is the sum of weights of edges crossing over to the set \mathcal{S} . Therefore, given a current selected \mathcal{S} , the greedy algorithm selects the next node, to be added to \mathcal{S} , that maximizes the increase in

$$\Omega_1(\mathcal{S}) \approx \inf_{\|\mathbf{x}\|=1} \sum_{j \in \mathcal{S}^c} \left(\frac{p_j}{d_j} \right) x_j^2. \quad (4.5)$$

Due to the constraint $\|\mathbf{x}\| = 1$, the expression being minimized is essentially an infimum over a convex combination of the fractional out-degrees and its value is largely determined by nodes $j \in \mathcal{S}^c$ for which p_j/d_j is small. In other words, we must worry about those nodes that have a low ratio of partial degree to the actual degree. Thus, in the simplest case, our selection algorithm tries to remove those nodes from the unlabeled set that are weakly connected to nodes in the labeled set. This makes intuitive sense as, in the end, most prediction algorithms involve propagation of labels from the labeled to the unlabeled nodes. If an unlabeled node is strongly connected to various numerous points, its label can be assigned with greater confidence.

Maximizing $\Omega_k(\mathcal{S})$ with $k > 1$, which involves taking a higher power k in $\mathbf{x}^\top \mathbf{L}^k \mathbf{x}$, takes into account multi-hop paths while ensuring better connectedness between \mathcal{S} and \mathcal{S}^c . This effect is especially important in sparsely connected graphs and the benefit of increasing k becomes less noticeable when the graphs are dense [2].

4.2.3 Prediction Error and Number of Labels

As discussed in Section 3.2, given the samples $\mathbf{f}(\mathcal{S})$ of the true graph signal on a subset of nodes $\mathcal{S} \subset \mathcal{V}$, its estimate on \mathcal{S}^c is obtained by solving the following problem:

$$\hat{\mathbf{f}}(\mathcal{S}^c) = \mathbf{U}_{\mathcal{S}^c, \mathcal{R}} \boldsymbol{\alpha}^* \text{ where, } \boldsymbol{\alpha}^* = \arg \min_{\boldsymbol{\alpha}} \|\mathbf{U}_{\mathcal{S}, \mathcal{R}} \boldsymbol{\alpha} - \mathbf{f}(\mathcal{S})\| \quad (4.6)$$

Here, \mathcal{R} is the index set of eigenvectors with eigenvalues less than the cut-off $\omega_c(\mathcal{S})$. If the true signal $\mathbf{f} \in PW_{\omega_c(\mathcal{S})}(G)$, then the prediction is perfect. However, this is not the case in most problems. The prediction error $\|\mathbf{f} - \hat{\mathbf{f}}\|$ roughly equals the portion of energy of the true signal in $[\omega_c(\mathcal{S}), \lambda_N]$ frequency band. By choosing the sampling set \mathcal{S} that maximizes $\omega_c(\mathcal{S})$, we try to capture most of the signal energy and thus, reduce the prediction error.

An important question in the context of active learning is determining the minimum number of labels required so that the prediction error $\|\mathbf{f} - \hat{\mathbf{f}}\|$ is less than some given tolerance δ . To find this we first characterize the smoothness $\gamma(\mathbf{f})$ of a signal \mathbf{f} as

$$\gamma(\mathbf{f}) = \min \theta \text{ s.t. } \|\mathbf{f} - \mathcal{P}_\theta \mathbf{f}\| \leq \delta$$

The following theorem gives a lower bound on the minimum of number of labels required in terms of $\gamma(\mathbf{f})$.

Theorem 4. *If $\hat{\mathbf{f}}$ is obtained by solving (4.6), then the minimum number of labels ℓ required to satisfy $\|\mathbf{f} - \hat{\mathbf{f}}\| \leq \delta$ is greater than p , where p is the number of eigenvalues of \mathbf{L} less than the smoothness, $\gamma(\mathbf{f})$, of signal \mathbf{f} .*

Proof. In order for (4.6) to have a unique solution, $\mathbf{U}_{\mathcal{S}, \mathcal{R}}$ needs to have full column rank, which implies that $\ell = |\mathcal{S}| \geq |\mathcal{R}|$. Now, for $\|\mathbf{f} - \hat{\mathbf{f}}\| \leq \delta$ to hold the bandwidth of $\hat{\mathbf{f}}$ has to be at least $\gamma(\mathbf{f})$, or in other words, $|\mathcal{R}| \geq p$. This gives us the desired result as $\ell \geq |\mathcal{R}| \geq p$. \square

4.3 Related Work

Different frameworks have been proposed for pool-based batch-mode active semi-supervised learning including optimal experiment design [89, 86], generalization error bound minimization [37, 38] and submodular optimization [39, 40, 44]. We now point out connections between some of the graph based approaches in the above categories and our graph signal sampling theory based framework.

The notion of frequency given by the GFT is closely related to Laplacian eigenmap, a well known non-linear dimensionality reduction technique [3]. Laplacian eigenmap represents data points lying on a low dimensional manifold embedded in a high-dimensional space by points in an r dimensional Euclidean space with coordinates of point i given by $(\mathbf{u}_i^1, \dots, \mathbf{u}_i^r)$ (i.e., the values that the first r GFT basis vectors take on node i). By selecting nodes that maximize the bandwidth of the space of recoverable signals, we try to capture as many dimensions of the manifold structure of the data with as few samples as possible. A related active learning method proposed by Zhang et al. [89] is based on local linear embedding (LLE), a different technique for approximating low-dimensional manifold structure of data [69]. The approach in [89] uses optimal experiment design to choose the most representative data points from the manifold, using which one can recover the whole data set by local linear reconstruction.

Gu and Han [37] propose a method based on minimizing the generalization error bound for learning with local and global consistency (LLGC) [90]. Their

formulation boils down to choosing a subset \mathcal{S} that minimizes $\text{Tr}((\mu \mathbf{L}_{\mathcal{S}} + \mathbf{I})^{-2})$. To relate this formulation to our proposed method, note that

$$\text{Tr}((\mu \mathbf{L}_{\mathcal{S}} + \mathbf{I})^{-2}) = \sum_i \frac{1}{(\zeta_i + 1)^2} \leq \frac{|\mathcal{S}|}{(\zeta_1 + 1)^2}$$

where, $\zeta_1 \leq \dots \leq \zeta_{|\mathcal{S}|}$ denote the eigenvalues of $\mathbf{L}_{\mathcal{S}}$. Loosely speaking, minimizing the above objective function is equivalent to maximizing the smallest eigenvalue ζ_1 of $\mathbf{L}_{\mathcal{S}}$. Using an argument similar to the one in Section 4.2.2, we can show that this method essentially tries to ensure that the labeled set is well-connected to the unlabeled set. Our method, on the other hand, ensures that the unlabeled set is well-connected to the labeled set while taking into account multi-hop paths by allowing higher degrees of \mathbf{L}^k beyond $k = 1$.

Submodular functions have been used for active semi-supervised learning on graphs by Guillory and Bilmes [40, 39]. In this work, the subset of nodes $S \subset \mathcal{V}$ is chosen to maximize

$$\Psi(S) = \min_{T \subseteq \mathcal{V} \setminus S: T \neq \emptyset} \frac{\Gamma(T)}{|T|}, \quad (4.7)$$

where $\Gamma(T)$ denotes the cut function $\sum_{i \in T, j \notin T} w_{ij}$. Intuitively, maximizing $\Psi(S)$ ensures that no subset of unlabeled nodes is weakly connected to the labeled set S . This agrees with the graph theoretic interpretation of our method given in Section 4.2.2. [40] also provides a bound on the prediction error in terms $\Psi(S)$ and a smoothness function $\Phi(\mathbf{f}) = \sum_{i,j} w_{ij} |f_i - f_j|$. This bound gives a theoretical justification for semi-supervised learning using min-cuts [5]. It also motivates a graph partitioning-based active learning heuristic [39] which says that to select ℓ nodes to label, the graph should be partitioned into ℓ clusters and one node should be picked at random from each cluster. Graph partitioning can be done using spectral clustering methods [84] that use the first r eigenvectors of the graph Laplacian to represent each node in an Euclidean space as in Laplacian eigenmap. Selecting one node from each of the ℓ clusters obtained using spectral clustering amounts to selecting ℓ rows from $\mathbf{U}_{\mathcal{VR}}$.

4.4 Experiments

We compare our method against four active semi-supervised learning approaches mentioned in the previous section, namely, LLR [89], LLGC error bound minimization [37], METIS graph partitioning based heuristic [39] and Ψ -max [40]. The details of implementation of each method are as follows:

1. The LLR approach [89] allows any prediction method once the samples to be queried are chosen. We use the Laplacian regularized least squares (LapRLS) [4] method for prediction (used in [89]).
2. In our implementation of the LLGC bound method [37], we fix the parameter μ to 0.01. Since this approach is based on minimizing the generalization error bound for LLGC, we use LLGC for prediction with the queried samples.²
3. The normalized cut based active learning heuristic of Guillory and Bilmes [39] is implemented using the METIS graph partitioning package [48]. This algorithm chooses a random node to label from each partition, so we average the error rates over a 100 trials.

The parameter k in our method is fixed to 8 for these experiments. Its effect on classification accuracy is studied in Section 4.4.4. In addition to the above methods, we also compare with the random sampling strategy. In the random sampling case, we use LapRLS to predict the unknown labels from the randomly queried samples and report the average error rates over 30 trials.

To intuitively demonstrate the effectiveness of our method, we first test it on the two circles toy data shown in Figure 4.2. The data is comprised of 200 nodes from which we would like to select 8 nodes to query. We construct a weighted sparse graph by connecting each node to its 10 nearest neighbors while ensuring that the connections are symmetric. The edge weights are computed with the Gaussian kernel $\exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right)$ (except in the case of Ψ -max where the graph is unweighted). It can be seen from Figure 4.2 that all the methods choose 4 points from each of the two circles. Additionally, the proposed approach selects evenly spaced data points within one circle, while at the same time maximizing the spacing between the selected data points in different circles. This ensures that

²In our experiments, we observed that the greedy algorithm given in [37] did not converge to a good solution. So we use Monte-Carlo simulations to minimize the objective function.

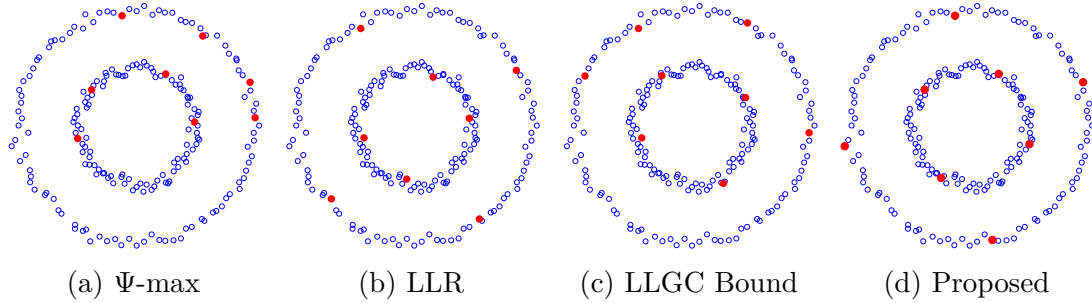


Figure 4.2: Toy example comparing the nodes selected using different active learning methods

the unlabeled nodes are as well-connected to the labeled nodes as possible, which in turn leads to effective label propagation.

We tested our method in three application scenarios: Handwritten digit recognition, text classification and spoken letters recognition. In these experiments, we do not compare with Ψ -max since the method has computational complexity of $O(N^6)$ and, to the best of our knowledge, is not scalable. Next, we provide the details of each experiment. Both the datasets and the graph construction procedures used are typical of what has been used in the literature.

4.4.1 Handwritten digits classification

In this experiment, we used our proposed active semi-supervised learning algorithm to perform a classification task on the USPS handwritten digits dataset³. This dataset consists of 1100 16×16 pixel images for each of the digits 0 to 9. We used 100 randomly selected samples for each digit class to create one instance of our dataset. Thus each instance consists of 1000 feature vectors ($100 \text{ samples/class} \times 10 \text{ digit classes}$) of dimension 256.

We construct a symmetric, weighted K -nearest neighbor graph with $K = 10$ and Gaussian kernel weights, $w_{ij} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right)$, where \mathbf{x}_i is the 256-dimensional feature vector composed of pixel intensity values for each image. The parameter σ is chosen to be 1/3-rd of the average distance to the 10-th nearest neighbor for all data points. Using the graph constructed, we select the points to label and report prediction error after reconstruction using our semi-supervised

³<http://www.cs.nyu.edu/~roweis/data.html>

learning algorithm. We repeat the classification over 10 such instances of the dataset and report the average classification error. The results are illustrated in Figure 4.3(a). We observe that our proposed method outperforms the others and gives very good classification accuracy even with very few labeled samples.

4.4.2 Text classification

For our text classification example, we use the 20 newsgroups dataset⁴. It contains around 20,000 documents, partitioned in 20 different newsgroups. For our experiment, we consider 10 groups of documents, namely, {comp.graphics, comp.os.ms-windows.misc, comp.sys.ibm.pc.hardware, comp.sys.mac.hardware, rec.autos, rec.motorcycles, sci.crypt, sci.electronics, sci.med, sci.space}, and randomly choose 100 data points from each group. We generate 10 such instances of 1000 data points each and report the average errors. We clean the dataset by removing the words that appear in fewer than 20 documents and then select only the 3000 most frequent ones from the remaining words. To form the feature vectors representing the documents, we use the term frequency-inverse document frequency (tf-idf) statistic of these words. The tf-idf statistic captures the relative importance of a word in a document in a corpus:

$$\text{tf-idf} = (1 + \log(\text{tf})) \times \log\left(\frac{N}{\text{idf}}\right) \quad (4.8)$$

where, tf is the frequency of a word in a document, idf is the number of documents in which the word appears and N is the total number of documents. Thus, we get 1000 feature vectors in 3000 dimensional space. To form the graph of documents, we compute the pairwise cosine similarity between their feature vectors. Each node is connected to the 10 nodes that are most similar to it and the resultant graph is then symmetrized. The classification results in Figure 4.3(b) show that our method performs very well compared to others. However, the absolute error rates are not very good. This is due to the high similarity between different newsgroups which makes the problem inherently difficult.

⁴<http://qwone.com/~jason/20Newsgroups/>

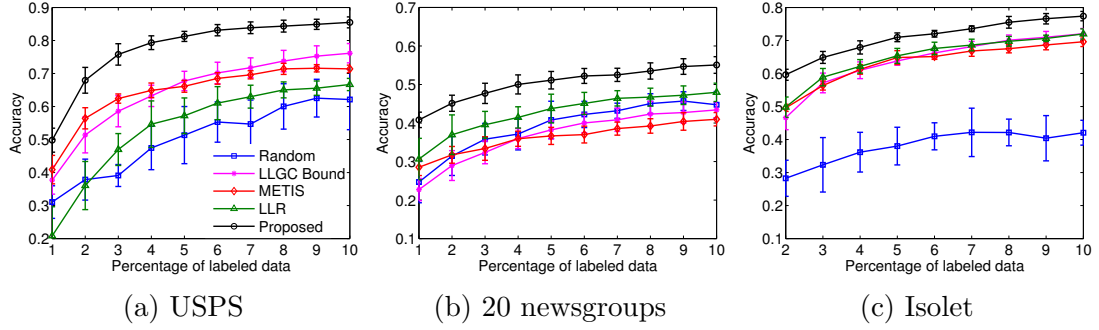


Figure 4.3: Comparison of active semi-supervised learning methods on real datasets. Plots show the average classification accuracies for different percentages of labeled data.

4.4.3 Spoken letters classification

For the spoken letters classification example, we considered the Isolet dataset⁵. It consists of letters of the English alphabet spoken in isolation twice by 150 different subjects. The speakers are grouped into 5 sets of 30 speakers each, with the groups referred to as isolet1 through isolet5. Each alphabet utterance has been pre-processed beforehand to create a 617-dimensional feature vector.

For this experiment, we considered the task of active semi-supervised classification of utterances into the 26 alphabet categories. To form an instance of the dataset, 60 utterances are randomly selected out of 300 for each alphabet. Thus, each instance consists of $60 \times 26 = 1560$ data points of dimension 617. As in the hand-written digits classification problem, the graph is constructed using Gaussian kernel weights between nodes, with σ taken as $1/3$ -rd of the average distance to the K -th nearest neighbor for each data point. We select $K = 10$ for our experiment. Sparsification of the graph is carried out approximately using K -nearest neighbor criterion. With the constructed graph, we perform active semi-supervised learning using all the methods. The experiment is repeated over 10 instances of the dataset and average prediction error is reported in Figure 4.3(c). Note that we start with 2% labeled points to ensure that each method gets a fair chance of selecting at least one point to label from each of the 26 classes. We observe that our method outperforms the others.

⁵<http://archive.ics.uci.edu/ml/datasets/ISOLET>

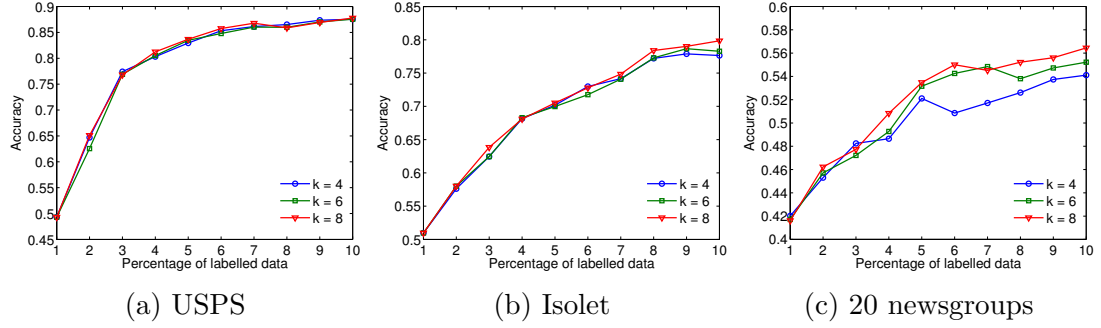


Figure 4.4: Effect of k on classification accuracy of the proposed method. Plots show the average classification accuracy for different percentages of labeled data.

4.4.4 Effect of parameter k

To study the effect of parameter k in the proposed method on classification accuracy we repeat the above experiments for different values of k . Figure 4.4 shows the results. For the USPS and Isolet datasets, the classification accuracies remain largely unchanged for different values of k . For the 20 Newsgroups dataset, a slight improvement in classification accuracies is observed for higher values of k . This result agrees with the distribution of GFT coefficients of the class membership functions in each dataset shown in Figure 4.1. In USPS and Isolet datasets, most of the energy of the graph signal (i.e., the class membership functions) is contained in the first few frequencies. Thus, increasing the value of k , so that a better estimate of cut-off frequency is maximized during the choice of sampling set, is not necessary. In other words, maximizing a loose estimate of the cut-off frequency is sufficient. However, the membership functions in the 20 Newsgroups dataset have a significant fraction of their energy spread over high frequencies as shown in Figure 4.1. Due to this, maximizing a tighter estimate of the cut-off frequency, which leads to a sampling set that allows for stable reconstruction of signals with larger bandwidths, results in higher accuracies.

4.5 Summary

In this chapter, we introduced a novel framework for batch mode active semi-supervised learning based on sampling theory for graph signals. The proposed active learning framework aims to select the subset nodes which maximizes the cutoff frequency and thus, gives most stable reconstructions. This interpretation

leads to a very efficient greedy algorithm. We provided intuition about how the method tries to choose the nodes which are most representative of the data. We also presented an efficient semi-supervised learning method based on bandlimited interpolation. We showed, through experiments on real data, that our two algorithms, in conjunction, perform very well compared to state of the art methods. In the future, we would like to provide tighter bounds on the number of labels required for desired prediction accuracy. It would be useful to consider an extension of the proposed framework to a partially batch setting so that we can incorporate the label information from previous batches to improve the future choice of sampling set. An approach to achieve this is introduced in Chapter 5.

Chapter 5

Probabilistic Interpretation of Sampling Theory and Extension to Adaptive Sampling

In this chapter, we develop a probabilistic interpretation for the graph sampling theoretic methods for active semi-supervised learning from Chapter 4. Our interpretation is based on defining a graph-based probabilistic prior model for signals in order to characterize their smoothness. Using this interpretation, we propose adaptive sampling and label prediction methods that are better suited to the discrete valued graph signals encountered in classification problems.

In the sampling based prediction techniques of Chapter 4, label prediction is considered as a graph signal reconstruction problem. The characterization of a subset of nodes given by the sampling theory, namely the associated cutoff frequency, is used as a criterion to choose the optimal set nodes to be labeled for active learning. As presented in the last chapter, the sampling theoretic methods for active and semi-supervised learning are purely deterministic. Their probabilistic interpretation is desired for the following reasons:

1. It leads to a principled Bayesian way to refine the model parameters (which are given by the underlying graph) as more data is observed.

This chapter is partially based on our work in [32].

2. It makes the relationship between the sampling theoretic approach and previously proposed semi-supervised [94] and active learning [46, 54] methods more apparent.

A probabilistic approach for active semi-supervised learning starts by defining a graph based prior distribution for signals and a likelihood model for observations. The problem of label prediction then boils down to finding the posterior mean or MAP estimate. Active learning can be formulated as Bayesian experiment design [10], which is the problem of selecting the best subset from the set of available measurements of a vector \mathbf{x} so that the error (given by a suitable loss function) in the reconstruction of \mathbf{x} using those measurements is minimized.

We begin by defining a Gaussian random field (GRF) prior for graph signals with a covariance matrix that depends on the graph. We show that, when conditions of the graph signal sampling theorem are satisfied, bandlimited reconstruction of a graph signal from a subset of its samples is equivalent to finding the MAP estimate of the unobserved samples, given the observations, with a low rank approximation of the above GRF. We then show that a sampling set of given size with the largest associated cut-off frequency, which is optimal from a sampling theoretic point of view, minimizes the largest eigenvalue of prediction covariance. Various other graph based active learning methods proposed in the literature can be viewed as minimizing some function of the prediction covariance under the GRF model assumption. We show that if the labels are assumed to follow a GRF model then the expected prediction error, defined as a function of the prediction covariance, is minimized by a non-adaptive sampling strategy and using the previously observed samples in future sample selection does not help in reducing it further.

A non-adaptive sampling strategy can be useful if all the labels in the budget need to be queried simultaneously. But in many applications, labels can be queried one at a time or in batches. If the prior model for the labels is not accurate then non-adaptive sampling will not be very effective. Therefore, we suggest a way to introduce adaptation in the sampling process so that the choice of future samples depends not only on the graph but also on the samples observed in the past. This can be useful because of the following reasons:

1. The graph provides a probabilistic model for the smoothness assumption on the node labels. However, the graph itself is constructed using the feature vectors associated with the nodes. Using the partially observed signal (i.e.,

observed labels) can allow us to refine the signal model by changing the graph.

2. Using the observed labels, we can predict the labels for the rest of the nodes to reveal a rough decision boundary. Nodes whose predicted labels are most ambiguous are closer to the boundary. Sampling near this boundary can allow us to converge to the correct labels faster.

In order to develop an adaptive sampling method, we first propose a new prior for graph signals using the concept of p -Laplacian, which is better suited for discrete valued labels in classification problems. Since the proposed prior is not Gaussian, the posterior covariance of the signal depends on the observed labels. We use variational Bayesian inference techniques to find the posterior covariance approximately. The nodes to be sampled are then selected such that a suitable function of this posterior covariance is minimized. Since the posterior covariance depends on the observed labels, sampling set selection is adaptive. Because the proposed prior offers a more realistic model for the discrete valued graph signals in classification problems, adaptive sampling and prediction methods based on this prior lead to better classification accuracy.

The rest of this chapter is organized as follows. In Section 5.1, we apply the framework of Bayesian inference and experiment design to the GRF model to give a probabilistic interpretation of sampling theory of graph signals. This also provides a unified view of various active learning methods on graphs. We also explain why a non-adaptive sampling strategy is sufficient to select a sampling set that minimizes the prediction error (defined as a functional of the prediction covariance) under this GRF model. In Section 5.2, we introduce a new prior model for graph signals based on the concept of p -Laplacian. We use it to propose an adaptive sampling set selection method, in which the choice of future samples depends on the signal samples observed in the past. Numerical experiments are presented in Section 5.3, which show that adaptive sampling can give better classification accuracy than non-adaptive sampling with the same number of observed labels.

5.1 Probabilistic Interpretation of Sampling Theory

5.1.1 GRF Prior and Observation Likelihood

The smoothness assumption on the vector of node labels \mathbf{f} can be formalized by imposing a Gaussian random field (GRF) prior on them:

$$p(\mathbf{f}) \propto \exp\left(-\frac{1}{2}\mathbf{f}^\top \mathbf{L} \mathbf{f}\right), \quad (5.1)$$

Under this model, vectors of labels with high variation $\mathbf{f}^\top \mathbf{L} \mathbf{f}$ are less likely¹. Let \mathbf{K} denote the covariance matrix of the GRF. Then, from the above equation $\mathbf{K} = \mathbf{L}^{-1}$. Most of the variation operators, \mathbf{L} , introduced in Section 2.3 are singular. In this case, \mathbf{L} can be replaced with $\mathbf{L} + \delta \mathbf{I}$ in order to get a bounded \mathbf{K} . $1/\delta$ can be interpreted as the variance of the GFT coefficient of \mathbf{f} corresponding zero frequency.

Let \mathcal{S} be a subset of the nodes in the graph and $\mathcal{S}^c = \mathcal{V} \setminus \mathcal{S}$. If \mathbf{f} has a distribution given by (5.1) then the conditional distribution of $\mathbf{f}_{\mathcal{S}^c}$ given $\mathbf{f}_{\mathcal{S}}$ is also Gaussian with mean and covariance given by

$$\boldsymbol{\mu}_{\mathcal{S}^c|\mathcal{S}} = \mathbf{K}_{\mathcal{S}^c\mathcal{S}}(\mathbf{K}_{\mathcal{S}})^{-1}\mathbf{f}_{\mathcal{S}} \quad \text{and} \quad (5.2)$$

$$\mathbf{K}_{\mathcal{S}^c|\mathcal{S}} = \mathbf{K}_{\mathcal{S}^c} - \mathbf{K}_{\mathcal{S}^c\mathcal{S}}(\mathbf{K}_{\mathcal{S}})^{-1}\mathbf{K}_{\mathcal{S}\mathcal{S}^c} \quad (5.3)$$

respectively. In order to write the above expressions in the form of sub-matrices of \mathbf{L} , we can use the block matrix inversion formula to get:

$$\begin{bmatrix} \mathbf{K}_{\mathcal{S}^c} & \mathbf{K}_{\mathcal{S}^c\mathcal{S}} \\ \mathbf{K}_{\mathcal{S}\mathcal{S}^c} & \mathbf{K}_{\mathcal{S}} \end{bmatrix} = \begin{bmatrix} \mathbf{M}_{\mathbf{L}_{\mathcal{S}}}^{-1} & -(\mathbf{L}_{\mathcal{S}^c})^{-1}\mathbf{L}_{\mathcal{S}^c\mathcal{S}}\mathbf{M}_{\mathbf{L}_{\mathcal{S}^c}}^{-1} \\ -(\mathbf{L}_{\mathcal{S}})^{-1}\mathbf{L}_{\mathcal{S}\mathcal{S}^c}\mathbf{M}_{\mathbf{L}_{\mathcal{S}}}^{-1} & \mathbf{M}_{\mathbf{L}_{\mathcal{S}^c}}^{-1} \end{bmatrix},$$

$$\begin{aligned} \text{where } \mathbf{M}_{\mathbf{L}_{\mathcal{S}}} &= \mathbf{L}_{\mathcal{S}^c} - \mathbf{L}_{\mathcal{S}^c\mathcal{S}}(\mathbf{L}_{\mathcal{S}})^{-1}\mathbf{L}_{\mathcal{S}\mathcal{S}^c}, \\ \mathbf{M}_{\mathbf{L}_{\mathcal{S}^c}} &= \mathbf{L}_{\mathcal{S}} - \mathbf{L}_{\mathcal{S}\mathcal{S}^c}(\mathbf{L}_{\mathcal{S}^c})^{-1}\mathbf{L}_{\mathcal{S}^c\mathcal{S}} \end{aligned} \quad (5.4)$$

¹The prior distribution (5.1) can be generalized in order to put a stronger smoothness assumption on the labels by replacing \mathbf{L} with a high pass function $h(\mathbf{L})$ of \mathbf{L} . For example, using $h(\mathbf{L}) = \mathbf{L}^k$ with $k > 1$ leads to the cutoff maximization method for active learning (from Chapter 4) under this model. Theoretical justification for this choice is provided in [93, 32].

are the Schur complements of $\mathbf{L}_{\mathcal{S}}$ and $\mathbf{L}_{\mathcal{S}^c}$ respectively. Note that the Schur complement of $\mathbf{L}_{\mathcal{S}^c}$ equals the Laplacian of the Kron reduction corresponding to the nodes in \mathcal{S} . Kron reduction is widely used in applications such as electrical circuit analysis, finite element analysis [22], multiscale graph transform design [78] etc. for reconnecting a subset of the nodes by taking into account the original connections in the whole graph. (5.4) shows that Kron reduction corresponding to \mathcal{S} preserves the covariance of $\mathbf{f}_{\mathcal{S}}$ if \mathbf{f} follows the GRF model (5.1). Using (5.4) we can write the conditional mean and covariance as a function of \mathbf{L} as

$$\boldsymbol{\mu}_{\mathcal{S}^c|\mathcal{S}} = -(\mathbf{L}_{\mathcal{S}^c})^{-1}\mathbf{L}_{\mathcal{S}^c\mathcal{S}}\mathbf{f}_{\mathcal{S}}, \quad (5.5)$$

$$\mathbf{K}_{\mathcal{S}^c|\mathcal{S}} = (\mathbf{L}_{\mathcal{S}^c})^{-1}. \quad (5.6)$$

We observe the labels of a subset of the nodes \mathcal{S} through the following measurement model

$$\mathbf{b} = \mathbf{D}_{\mathcal{S}}\mathbf{f} + \boldsymbol{\epsilon}, \quad (5.7)$$

where $\boldsymbol{\epsilon}$ is zero-mean, white Gaussian noise with variance σ^2 and the sampling matrix $\mathbf{D}_{\mathcal{S}}$ is obtained by taking the rows of an $N \times N$ identity matrix corresponding to the subset \mathcal{S} . The conditional likelihood is then given by

$$p(\mathbf{b}|\mathbf{f}) = \mathcal{N}(\mathbf{b}|\mathbf{D}_{\mathcal{S}}\mathbf{f}, \sigma^2\mathbf{I}), \quad (5.8)$$

where the notation $\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes a Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. The posterior distribution $p(\mathbf{f}|\mathbf{b})$ is also Gaussian with covariance given by

$$\text{cov}(\mathbf{f}|\mathbf{b}) = \left(\mathbf{L} + \frac{1}{\sigma^2}\mathbf{D}_{\mathcal{S}}^{\top}\mathbf{D}_{\mathcal{S}} \right)^{-1}. \quad (5.9)$$

Using the block matrix inversion formula, we can write the covariance of the labels to be predicted $\mathbf{f}_{\mathcal{S}^c}$ as follows

$$\text{cov}(\mathbf{f}_{\mathcal{S}^c}|\mathbf{b}) = \left(\mathbf{L}_{\mathcal{S}^c} - \sigma^2\mathbf{L}_{\mathcal{S}^c\mathcal{S}}(\sigma^2\mathbf{L}_{\mathcal{S}} + \mathbf{I})^{-1}\mathbf{L}_{\mathcal{S}\mathcal{S}^c} \right)^{-1}. \quad (5.10)$$

Note that as the noise $\sigma^2 \rightarrow 0$, the above expression reduces to $\mathbf{L}_{\mathcal{S}^c}^{-1}$ as in (5.6).

5.1.2 Bandlimited Reconstruction as MAP Inference

Let λ_r be the largest eigenvalue of \mathbf{L} that is less than ω . We define $\hat{\mathbf{K}}$ to be a low rank approximation of \mathbf{L}^{-1} that contains only the spectral components corresponding to $\{\lambda_1, \dots, \lambda_r\}$, i.e.,

$$\hat{\mathbf{K}} = \sum_{i=1}^r \frac{1}{\lambda_i} \mathbf{u}^i \mathbf{u}^{i\top} = \mathbf{U}_{\mathcal{VR}} \boldsymbol{\Sigma}_{\mathcal{R}} \mathbf{U}_{\mathcal{VR}}^\top. \quad (5.11)$$

As in Section 5.1.1, if \mathbf{L} is singular with $\lambda_1 = 0$, we assume that it is replaced by $\mathbf{L} + \delta \mathbf{I}$ in order to get bounded $\hat{\mathbf{K}}$. Now, consider the problem of reconstructing a random signal generated using a GRF with covariance $\hat{\mathbf{K}}$, from its samples on \mathcal{S} . The following theorem shows that, if conditions of the sampling theorem are satisfied, then the error of bandlimited reconstruction is zero.

Theorem 5. *Let \mathbf{f} be a random graph signal generated using the GRF with covariance $\hat{\mathbf{K}}$ given by (5.11). Let $\hat{\mathbf{f}}_{\mathcal{S}^c}$ be the bandlimited reconstruction of $\mathbf{f}_{\mathcal{S}^c}$ obtained from its samples on \mathcal{S} , where \mathcal{S} is a uniqueness set for $PW_\omega(G)$. Then, $\mathbf{f}_{\mathcal{S}^c} = \hat{\mathbf{f}}_{\mathcal{S}^c}$.*

Before proving the above theorem, we show, in the lemma below, that bandlimited reconstruction is equivalent to MAP inference on the GRF with covariance $\hat{\mathbf{K}}$.

Lemma 1. *Let $\mathcal{S} \subseteq \mathcal{V}$ be a uniqueness set for $PW_\omega(G)$. Then the MAP estimate of $\mathbf{f}_{\mathcal{S}^c}$ given $\mathbf{f}_{\mathcal{S}}$ in a GRF with covariance matrix $\hat{\mathbf{K}}$ is equal to the bandlimited reconstruction given by (3.1).*

Proof. Under a permutation which groups together nodes in \mathcal{S}^c and \mathcal{S} , we can write $\hat{\mathbf{K}}$ as the following block matrix

$$\begin{bmatrix} \hat{\mathbf{K}}_{\mathcal{S}^c} & \hat{\mathbf{K}}_{\mathcal{S}^c \mathcal{S}} \\ \hat{\mathbf{K}}_{\mathcal{S} \mathcal{S}^c} & \hat{\mathbf{K}}_{\mathcal{S}} \end{bmatrix} = \begin{bmatrix} \mathbf{U}_{\mathcal{S}^c \mathcal{R}} \boldsymbol{\Sigma}_{\mathcal{R}} \mathbf{U}_{\mathcal{S}^c \mathcal{R}}^\top & \mathbf{U}_{\mathcal{S}^c \mathcal{R}} \boldsymbol{\Sigma}_{\mathcal{R}} \mathbf{U}_{\mathcal{S} \mathcal{R}}^\top \\ \mathbf{U}_{\mathcal{S} \mathcal{R}} \boldsymbol{\Sigma}_{\mathcal{R}} \mathbf{U}_{\mathcal{S}^c \mathcal{R}}^\top & \mathbf{U}_{\mathcal{S} \mathcal{R}} \boldsymbol{\Sigma}_{\mathcal{R}} \mathbf{U}_{\mathcal{S} \mathcal{R}}^\top \end{bmatrix} \quad (5.12)$$

Therefore, we can write the MAP estimate obtained with covariance $\hat{\mathbf{K}}$ as,

$$\hat{\boldsymbol{\mu}}_{\mathcal{S}^c | \mathcal{S}} = \mathbf{U}_{\mathcal{S}^c \mathcal{R}} \boldsymbol{\Sigma}_{\mathcal{R}} \mathbf{U}_{\mathcal{S} \mathcal{R}}^\top (\mathbf{U}_{\mathcal{S} \mathcal{R}} \boldsymbol{\Sigma}_{\mathcal{R}} \mathbf{U}_{\mathcal{S} \mathcal{R}}^\top)^+ \mathbf{f}_{\mathcal{S}}. \quad (5.13)$$

Because $\omega < \omega(\mathcal{S})$, we have that $\mathbf{U}_{\mathcal{SR}}$ has full column rank and equivalently, $\mathbf{U}_{\mathcal{SR}}^\top$ has full row rank. Therefore, we can write $(\mathbf{U}_{\mathcal{SR}}\boldsymbol{\Sigma}_{\mathcal{R}}\mathbf{U}_{\mathcal{SR}}^\top)^+ = (\mathbf{U}_{\mathcal{SR}}^\top)^+\boldsymbol{\Sigma}_{\mathcal{R}}^+\mathbf{U}_{\mathcal{SR}}^+$ and $\mathbf{U}_{\mathcal{SR}}^+ = (\mathbf{U}_{\mathcal{SR}}^\top\mathbf{U}_{\mathcal{SR}})^{-1}\mathbf{U}_{\mathcal{SR}}^\top$. Simplifying (5.13) using these equalities leads to

$$\hat{\mathbf{f}}_{\mathcal{S}^c} = \mathbf{U}_{\mathcal{S}^c\mathcal{R}}(\mathbf{U}_{\mathcal{SR}}^\top\mathbf{U}_{\mathcal{SR}})^{-1}\mathbf{U}_{\mathcal{SR}}^\top\mathbf{f}_{\mathcal{S}},$$

which is equal to the least squares solution given in (3.1). \square

Proof of Theorem 5. From Lemma 1, $\hat{\mathbf{f}}_{\mathcal{S}^c} = \hat{\boldsymbol{\mu}}_{\mathcal{S}^c|\mathcal{S}}$. Therefore,

$$\mathbb{E}(\|\mathbf{f}_{\mathcal{S}^c} - \hat{\mathbf{f}}_{\mathcal{S}^c}\|^2) = \text{Tr}(\mathbb{E}(\mathbf{f}_{\mathcal{S}^c} - \hat{\boldsymbol{\mu}}_{\mathcal{S}^c|\mathcal{S}})(\mathbf{f}_{\mathcal{S}^c} - \hat{\boldsymbol{\mu}}_{\mathcal{S}^c|\mathcal{S}})^\top) = \text{Tr}(\hat{\mathbf{K}}_{\mathcal{S}^c|\mathcal{S}}).$$

Now, $\hat{\mathbf{K}}_{\mathcal{S}^c|\mathcal{S}} = \hat{\mathbf{K}}_{\mathcal{S}^c} - \hat{\mathbf{K}}_{\mathcal{S}^c\mathcal{S}}(\hat{\mathbf{K}}_{\mathcal{S}})^+\hat{\mathbf{K}}_{\mathcal{S}\mathcal{S}^c}$. Using the block form of $\hat{\mathbf{K}}$ in (5.12), and the fact that $\mathbf{U}_{\mathcal{SR}}$ has full column rank, it is easy to show that $\hat{\mathbf{K}}_{\mathcal{S}^c|\mathcal{S}} = \mathbf{0}$, which implies $\mathbb{E}(\|\mathbf{f}_{\mathcal{S}^c} - \hat{\mathbf{f}}_{\mathcal{S}^c}\|^2) = 0$. But since $\|\mathbf{f}_{\mathcal{S}^c} - \hat{\mathbf{f}}_{\mathcal{S}^c}\| \geq 0$, we get $\|\mathbf{f}_{\mathcal{S}^c} - \hat{\mathbf{f}}_{\mathcal{S}^c}\| = 0$ which in turn implies $\mathbf{f}_{\mathcal{S}^c} = \hat{\mathbf{f}}_{\mathcal{S}^c}$. \square

5.1.3 Active Learning as Bayesian Experiment Design

The goal of active learning is to select a subset of labels \mathcal{S} to be observed so that the unobserved labels $\mathbf{f}_{\mathcal{S}^c}$ can be estimated with least uncertainty. Let $u(\mathbf{f}_{\mathcal{S}^c}, \mathbf{b})$ be a loss function that quantifies the error in the prediction of $\mathbf{f}_{\mathcal{S}^c}$ given the observations \mathbf{b} . Taking the expectation of the loss function with respect to the joint density of $(\mathbf{f}_{\mathcal{S}^c}, \mathbf{b})$, we get

$$u(\mathcal{S}) = \mathbb{E}_{\mathbf{b}}\mathbb{E}_{\mathbf{f}_{\mathcal{S}^c}|\mathbf{b}}(u(\mathbf{f}_{\mathcal{S}^c}, \mathbf{b})) = \int p(\mathbf{b}) \int p(\mathbf{f}_{\mathcal{S}^c}|\mathbf{b})u(\mathbf{f}_{\mathcal{S}^c}, \mathbf{b})d\mathbf{f}_{\mathcal{S}^c}d\mathbf{b}. \quad (5.14)$$

The goal of an active learning algorithm is to choose \mathcal{S} so that $u(\mathcal{S})$ is minimized over all possible choices of given size m , i.e.,

$$\mathcal{S}^{\text{opt}} = \arg \min_{|\mathcal{S}|=m} u(\mathcal{S}). \quad (5.15)$$

For example, if we estimate $\mathbf{f}_{\mathcal{S}^c}$ using the conditional mean $\mathbb{E}(\mathbf{f}_{\mathcal{S}^c}|\mathbf{b})$ and consider the ℓ_2 error $\|\mathbf{f}_{\mathcal{S}^c} - \mathbb{E}(\mathbf{f}_{\mathcal{S}^c}|\mathbf{b})\|_2^2$ to be the loss function, then $u(\mathcal{S}) = \text{Tr}(\text{cov}(\mathbf{f}_{\mathcal{S}^c}|\mathbf{b}))$. As $\sigma^2 \rightarrow 0$, $u(\mathcal{S}) = \text{Tr}(\mathbf{K}_{\mathcal{S}^c|\mathcal{S}})$.

The sampling set \mathcal{S}^{opt} , which maximizes the approximate cutoff frequency $\lambda_{\min}(\mathbf{L}_{\mathcal{S}^c})$ (as proposed in Chapter 4), minimizes the largest eigenvalue of the prediction covariance, i.e.,

$$\arg \max_{|\mathcal{S}|=m} \lambda_{\min}(\mathbf{L}_{\mathcal{S}^c}) = \arg \min_{|\mathcal{S}|=m} \lambda_{\max}(\mathbf{K}_{\mathcal{S}^c|\mathcal{S}}). \quad (5.16)$$

This follows from the fact that for the GRF model (5.1), we have $\mathbf{K}_{\mathcal{S}^c|\mathcal{S}} = (\mathbf{L}_{\mathcal{S}^c})^{-1}$. Maximizing $\lambda_{\min}^k(\mathbf{L}_{\mathcal{S}^c})$ with $k > 1$ is equivalent to minimizing the largest eigenvalue of the prediction covariance when the prior is $p(\mathbf{f}) \propto \exp(-\mathbf{f}^\top \mathbf{L}^k \mathbf{f})$.

Different choices of loss functions u lead to different active learning criteria, which are summarized in Table 5.1 (assuming $\sigma^2 \rightarrow 0$). The relationship $\mathbf{K}_{\mathcal{S}^c|\mathcal{S}} = (\mathbf{L}_{\mathcal{S}^c})^{-1}$, allows us to give a graph theoretic motivation for minimizing some of the choices of $u(\mathcal{S})$ as a way of finding a good sampling set. For example, it was shown in Chapter 4 that minimizing $\lambda_{\max}[\mathbf{K}_{\mathcal{S}^c|\mathcal{S}}]$ or equivalently, maximizing $\lambda_{\min}[\mathbf{L}_{\mathcal{S}^c}]$ ensures that unsampled nodes \mathcal{S}^c are strongly connected to sampled nodes \mathcal{S} .

Algorithm for finding \mathcal{S}^{opt}

For any of the choices for $u(\mathcal{S})$ described in Table 5.1, the problem (5.15) of finding an optimal set \mathcal{S}^{opt} is NP hard. It is possible find an approximate solution using a greedy sequential sampling algorithm. Such an algorithm adds a node v from \mathcal{S}^c to \mathcal{S} at each iteration that causes $u(\mathcal{S})$ to decrease maximally. More specifically, let \mathcal{S}_i be the subset of nodes of size i sampled so far with measurement \mathbf{b}_i . The goal in sequential sampling is to select a node $v \in \mathcal{S}_i^c$ to get a new measurement b_v . The new sampling set is $\mathcal{S}_{i+1} = \mathcal{S}_i \cup \{v\}$. If $u(\mathcal{S})$ is a functional of $\mathbf{L}_{\mathcal{S}^c}^{-1}$ as in Table 5.1, then it is easy to find $u(\mathcal{S}_{i+1})$ from $u(\mathcal{S}_i)$ since $\mathbf{L}_{\mathcal{S}_{i+1}^c}^{-1}$ can be updated easily from $\mathbf{L}_{\mathcal{S}_i^c}^{-1}$ using the block matrix inverse and Sherman-Morrison formula [54]. Letting $\Sigma^i = \mathbf{L}_{\mathcal{S}_i^c}^{-1}$ and $\Sigma^{i+1} = \mathbf{L}_{\mathcal{S}_{i+1}^c}^{-1}$, then

$$\begin{pmatrix} \Sigma^{i+1} & \mathbf{0} \\ \mathbf{0} & 0 \end{pmatrix} = \Sigma^i - \frac{1}{\Sigma_{vv}^i} \Sigma_v^i \Sigma_v^{i\top}, \quad (5.17)$$

where Σ_v^i denotes the column of Σ^i corresponding to node v . For the E -optimality criterion, which tries to minimize $\lambda_{\max}(\mathbf{L}_{\mathcal{S}^c}^{-1})$ (or equivalently, maximize $\lambda_{\min}(\mathbf{L}_{\mathcal{S}^c})$), we can use the approximate greedy method in Chapter 4 (see also [2]).

Table 5.1: Different optimality criteria for active learning on graphs

Name	$u(\mathbf{f}_{S^c}, \mathbf{f}_S, \mathcal{S})$	$u(\mathcal{S})$	$u(\mathcal{S})$ in terms of \mathbf{L}
V -(or A -) optimality [46]	$\ \mathbf{f}_{S^c} - \boldsymbol{\mu}_{S^c \mathcal{S}}\ ^2$	$\text{Tr}(\mathbf{K}_{S^c \mathcal{S}})$	$\text{Tr}((\mathbf{L}_{S^c})^{-1})$
E -optimality [32]	$(\mathbf{c}^\top (\mathbf{f}_{S^c} - \boldsymbol{\mu}_{S^c \mathcal{S}}))^2$	$\max_{\ \mathbf{c}\ =1} u(\mathcal{S}, \mathbf{c}) = \lambda_{\max}[\mathbf{K}_{S^c \mathcal{S}}]$, where $u(\mathcal{S}, \mathbf{c}) = \mathbf{c}^\top \mathbf{K}_{S^c \mathcal{S}} \mathbf{c}$	$(\lambda_{\min}[\mathbf{L}_{S^c}])^{-1}$
Σ -optimality [34]	$(\mathbf{1}^\top (\mathbf{f}_{S^c} - \boldsymbol{\mu}_{S^c \mathcal{S}}))^2$	$\sum_{ij} (\mathbf{K}_{S^c \mathcal{S}})_{ij}$	$\sum_{ij} ((\mathbf{L}_{S^c})^{-1})_{ij}$
Minimum entropy or D -optimality [19]	$-\log p(\mathbf{f}_{S^c} \mathbf{f}_S)$	$\log \det(\mathbf{K}_{S^c \mathcal{S}})$	$\det((\mathbf{L}_{S^c})^{-1})$
Maximum mutual information [52]	$-\log \left(\frac{p(\mathbf{f}_{S^c} \mathbf{f}_S)}{p(\mathbf{f}_{S^c})} \right)$	$\log \left(\frac{\det(\mathbf{K}_{S^c \mathcal{S}})}{\det(\mathbf{K}_{S^c})} \right)$	$\log \left(\frac{\det((\mathbf{L}_{S^c})^{-1})}{\det((\mathbf{L}_{S^c} - \mathbf{L}_{S^c\mathcal{S}}(\mathbf{L}_S)^{-1} \mathbf{L}_{SS^c})^{-1})} \right)$

Let $\mathcal{S}^{\text{approx}}$ be the solution given by the greedy algorithm. In general, it is difficult to quantify how close $u(\mathcal{S}^{\text{approx}})$ is to $u(\mathcal{S}^{\text{opt}})$. But in the special case when $u(\mathcal{S})$ is a submodular function of \mathcal{S} , it is possible to bound $u(\mathcal{S}^{\text{approx}})$ within a constant factor of $u(\mathcal{S}^{\text{opt}})$ [52, 34]. In practice, the greedy algorithm gives a good solution even when the objective function is not submodular [30].

Choice of $u(\mathcal{S})$

It is not clear if one optimality criterion is superior to others. The choice of $u(\mathcal{S})$ depends on the final objective for which the selected samples are to be used. For example, if the goal is to predict real valued labels with minimum mean squared error then $u(\mathcal{S}) = \text{Tr}(\mathbf{K}_{\mathcal{S}^c|\mathcal{S}})$ (i.e., A -optimality) is a reasonable choice. Σ -optimality criterion, which minimizes $u(\mathcal{S}) = \sum_{ij} (\mathbf{K}_{\mathcal{S}^c|\mathcal{S}})_{ij}$, ensures good prediction of the value of fraction of data points in one class and thus, is more suitable for surveying problems [34]. E -optimality can be thought of as a minimax generalization of Σ -optimality (however, it does not appear to directly correspond to any function $u(\mathbf{f}_{\mathcal{S}^c}, \mathbf{b})$). In practice, sampling based on E -optimality criterion leads to superior classification performance compared to other criteria [32]. Information theoretic criteria (i.e., entropy and mutual information) aim at selecting nodes that are most helpful for refining the signal model.

5.1.4 Optimality of Non-Adaptive Set Selection in the GRF Model

With the GRF model, the posterior distribution $p(\mathbf{f}_{\mathcal{S}^c}|\mathbf{b})$ is also Gaussian with covariance $\text{cov}(\mathbf{f}_{\mathcal{S}^c}|\mathbf{b}) \approx \mathbf{L}_{\mathcal{S}^c}^{-1}$ that does not depend on the observed samples \mathbf{b} . For different choices of $u(\mathbf{f}_{\mathcal{S}^c}, \mathbf{b})$, which are quadratic functions of $(\mathbf{f}_{\mathcal{S}^c} - \mathbf{E}(\mathbf{f}_{\mathcal{S}^c}|\mathbf{b}))$ as in Table 5.1, $\mathbf{E}_{\mathbf{f}_{\mathcal{S}^c}|\mathbf{b}} u(\mathbf{f}_{\mathcal{S}^c}, \mathbf{b})$ is a function only of the conditional covariance $\text{cov}(\mathbf{f}_{\mathcal{S}^c}|\mathbf{b})$ and does not depend on the observations \mathbf{b} . Hence, active learning algorithms that use such loss functions and are based on the GRF model cannot adapt to the observed samples.

Note that the graph signals in classification problems are discrete valued. Although the GRF model is useful in this case, it does not take into account the discreteness of signals and does not allow for adaptation in sampling process. We address these issues in the next section by proposing a different prior for graph signals, which is more suited to discrete valued labels. The proposed prior is based

on the concept of p -Laplacian and is non-Gaussian. The posterior covariance of the signal depends on the observed signal samples. Therefore, any active learning scheme based on this model, which minimizes a posterior covariance based objective as in Table 5.1, is adaptive.

5.2 Bayesian Active Learning Using p -Laplacian Based Prior

5.2.1 Signal Prior Based on p -Laplacian

The standard graph Laplacian \mathbf{L} induces the quadratic form (2.5) for a graph signal $f : \mathcal{V} \rightarrow \mathbb{R}$ that measures the variation in the signal with respect to the graph. The p -Laplacian [9] \mathbf{L}_p is an operator that generalizes (2.5) as follows,

$$\langle \mathbf{f}, \mathbf{L}_p \mathbf{f} \rangle = \sum_{i \sim j} w_{ij} |\mathbf{f}_i - \mathbf{f}_j|^p \quad \text{for } p \geq 1. \quad (5.18)$$

We only consider the p -Laplacian with $p = 1$. Based on this notion of signal variation, we propose the following prior model for graph signals:

$$p(\mathbf{f}) \propto \exp(-\tau \sum_{i \sim j} w_{ij} |\mathbf{f}_i - \mathbf{f}_j|), \quad \text{where } \tau > 0. \quad (5.19)$$

Under this model, vectors of labels \mathbf{f} with high variation $\sum_{i \sim j} w_{ij} |\mathbf{f}_i - \mathbf{f}_j|$ are less likely.

In order to motivate this prior model for the discrete valued signals in classification problems, we represent the variation form (5.18) with $p = 1$ using the incidence matrix for the graph, which is defined as follows:

Definition 3. *Incidence matrix \mathbf{M} of a directed graph $G = (\mathcal{V}, \mathcal{E}, \mathbf{W})$ is a $|\mathcal{E}| \times |\mathcal{V}|$ matrix. Each row \mathbf{M}_e of \mathbf{M} corresponds to an edge e . If e goes from node i to node j and has weight w_{ij} , then*

$$\begin{aligned} \mathbf{M}_{ei} &= w_{ij} \\ \mathbf{M}_{ej} &= -w_{ij} \\ \mathbf{M}_{ek} &= 0 \text{ for all } k \neq i, j. \end{aligned}$$

For undirected graphs, edge orientations can be chosen arbitrarily.²

With this definition, 1-Laplacian based signal variation can be written as $\sum_{i \sim j} w_{ij} |\mathbf{f}_i - \mathbf{f}_j| = \|\mathbf{M}\mathbf{f}\|_{\ell_1}$. The signal prior (5.19) can then be written as a product of Laplace priors on $(\mathbf{M}\mathbf{f})_e$

$$p(\mathbf{f}) \propto \prod_e \exp(-\tau |(\mathbf{M}\mathbf{f})_e|). \quad (5.20)$$

A Laplace prior of the form (5.20) is commonly used to enforce sparsity with computationally tractable algorithms in sparse signal reconstruction methods [75]. The above prior promotes solutions \mathbf{f} such that $\mathbf{M}\mathbf{f}$ is sparse, i.e., $\mathbf{M}\mathbf{f}$ has a few “dominant” entries and the rest of the entries of $\mathbf{M}\mathbf{f}$ are zero. Such sparsity is desired when \mathbf{f} is discrete valued as is the case in classification problems. For example, when $\mathbf{f}_i \in \{+1, -1\}$, then $(\mathbf{M}\mathbf{f})_e = 0$ if $\mathbf{f}_i = \mathbf{f}_j$ and $(\mathbf{M}\mathbf{f})_e = 2w_{ij}$ if $\mathbf{f}_i \neq \mathbf{f}_j$. Because of the cluster assumption, we expect that, in a sparse, distance-based neighborhood graph, there are a much smaller number of edges connecting two oppositely labeled nodes than the edges connecting two similarly labeled nodes. Therefore, $\mathbf{M}\mathbf{f}$ is expected to be sparse.

Another motivation for the 1-Laplacian based prior can be given by considering its application in spectral clustering [9]. It is well known that the second eigenvector of the 2-Laplacian minimizes a relaxation of the ratio cut by dropping the constraint that the cut indicator vector be binary [84]. The second eigenvector of the p -Laplacian with $p < 2$ allows for minimization of a better relaxation of the ratio cut. The histogram of the resulting solution shows two distinct peaks corresponding to two clusters [9] as $p \rightarrow 1$ (and therefore, gives a sparse $\mathbf{M}\mathbf{f}$).

5.2.2 Bayesian Inference Using p -Laplacian Based Prior

We consider the same measurement model as in (5.7) with conditional likelihood $p(\mathbf{b}|\mathbf{f}) = \mathcal{N}(\mathbf{b}|\mathbf{D}_S\mathbf{f}, \sigma^2\mathbf{I})$. Using Bayes’ theorem we can write the posterior distribution over the signals as

$$p(\mathbf{f}|\mathbf{b}) = \frac{1}{p(\mathbf{b})} \mathcal{N}(\mathbf{b}|\mathbf{D}_S\mathbf{f}, \sigma^2\mathbf{I}) \prod_e \exp(-\tau |(\mathbf{M}\mathbf{f})_e|), \text{ where} \quad (5.21)$$

²An edge e in an undirected graph connecting nodes i and j can be assumed to be a directed edge that goes either from node i to node j or from node j to node i . This is because we only need to consider the absolute values of the pairwise differences $|\mathbf{f}_i - \mathbf{f}_j|$.

$$p(\mathbf{b}) = \int \mathcal{N}(\mathbf{b}|\mathbf{D}_S \mathbf{f}, \sigma^2 \mathbf{I}) \prod_e \exp(-\tau |(\mathbf{M}\mathbf{f})_e|) d\mathbf{f}. \quad (5.22)$$

If the goal is only to get a point estimate for the signal, one way is to find the mode of the posterior (i.e., MAP estimation) as follows:³

$$\hat{\mathbf{f}} = \arg \min_{\mathbf{f}} \frac{1}{\sigma^2} \|\mathbf{b} - \mathbf{D}_S \mathbf{f}\|_2^2 + \tau \|\mathbf{M}\mathbf{f}\|_1. \quad (5.23)$$

However, for Bayesian active learning, we need to find the posterior covariance of \mathbf{f}_{S^c} . Note that the posterior is non-Gaussian. Finding its mean and covariance is analytically intractable since it requires integration over a high dimensional non-Gaussian distribution (5.21).

An efficient variational method for approximate Bayesian inference in sparse linear models has been proposed in [75]. We use it to approximate the posterior mean and covariance of \mathbf{f} . This method approximates the posterior distribution by a Gaussian $q(\mathbf{f}|\mathbf{b})$. The main idea behind the Gaussian approximation is the fact that the Laplace distribution admits a tight lower bound in the form of a Gaussian function of width γ_e , i.e.,

$$\exp(-\tau |s_e|) = \max_{\gamma_e \geq 0} \exp\left(-\frac{s_e^2}{2\gamma_e} - \frac{\tau^2 \gamma_e}{2}\right), \text{ where } s_e = (\mathbf{M}\mathbf{f})_e. \quad (5.24)$$

Plugging this into the expression for $p(\mathbf{b})$, we get

$$p(\mathbf{b}) \geq \max_{\boldsymbol{\gamma} \geq \mathbf{0}} \exp\left(-\frac{\tau^2}{2} \|\boldsymbol{\gamma}\|_1\right) \int \mathcal{N}(\mathbf{b}|\mathbf{D}_S \mathbf{f}, \sigma^2 \mathbf{I}) \exp\left(-\frac{1}{2} \mathbf{f}^\top \mathbf{M}^\top \boldsymbol{\Gamma}^{-1} \mathbf{M} \mathbf{f}\right) d\mathbf{f}, \quad (5.25)$$

where $\boldsymbol{\gamma}$ is a vector of γ_e 's from (5.24) and $\boldsymbol{\Gamma} = \text{diag}(\boldsymbol{\gamma})$. Note that for a fixed $\boldsymbol{\gamma}$, the above integral is easy to evaluate since it takes a Gaussian form in \mathbf{f} . The posterior also takes a Gaussian form with

$$\text{cov}_q(\mathbf{f}|\mathbf{b}) := \mathbf{C} := \mathbf{A}^{-1} = \left(\frac{1}{\sigma^2} \mathbf{D}_S^\top \mathbf{D}_S + \mathbf{M}^\top \boldsymbol{\Gamma}^{-1} \mathbf{M}\right)^{-1}. \quad (5.26)$$

³This is analogous to the total variation regularization framework used in image processing [70].

The optimal value of γ is found by maximizing the lower bound on $p(\mathbf{b})$ given in (5.25). Using the fact that

$$\int \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) d\mathbf{x} = \sqrt{2\pi \det(\boldsymbol{\Sigma})} \max_{\mathbf{x}} \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}),$$

we can write the maximization problem (5.25) as

$$\min_{\gamma \geq 0} \log \det(\mathbf{A}) + \tau^2 \|\gamma\|_1 + \min_{\mathbf{f}} \frac{1}{\sigma^2} \|\mathbf{b} - \mathbf{D}_S \mathbf{f}\|_2^2 + \mathbf{f}^\top \mathbf{M}^\top \boldsymbol{\Gamma}^{-1} \mathbf{M} \mathbf{f}. \quad (5.27)$$

We use the method in [75] to solve the above problem by iterative alternate minimization over \mathbf{f} and γ , i.e., minimizing over \mathbf{f} while keeping γ fixed and then minimizing over γ while fixing \mathbf{f} to the previously obtained value. Since the objective is convex in \mathbf{f} and γ , it can be minimized efficiently. Once the optimal value of γ is found, the posterior covariance \mathbf{C} can be obtained by (5.26).

5.2.3 Sequential Sampling Using the Posterior Covariance

Let \mathcal{S}_i be the subset of nodes of size i sampled so far with measurement \mathbf{b}_i . Let \mathbf{C}^i be the covariance of the Gaussian approximation of the posterior $q(\mathbf{f}|\mathbf{b}_i)$. In order to choose the best node $v \in \mathcal{S}_i^c$ to get a new measurement b_v , we need to find the posterior covariance $\mathbf{C}^{i+1} = \text{cov}_q(\mathbf{f}|\mathbf{b}_i, b_v)$. Using the Gaussian approximation $q(\mathbf{f}|\mathbf{b}_i)$, the posterior covariance \mathbf{C}^{i+1} can be easily computed as follows,

$$\mathbf{C}^{i+1} = \left(\mathbf{C}^{i-1} + \frac{1}{\sigma^2} \mathbf{D}_v^\top \mathbf{D}_v \right)^{-1} = \mathbf{C}^i - \frac{1}{\sigma^2 + \mathbf{C}_{vv}^i} \mathbf{C}_v^i \mathbf{C}_v^{i\top}, \quad (5.28)$$

where \mathbf{D}_v is the row of an $N \times N$ identity matrix corresponding to node v and \mathbf{C}_v^i is the column of \mathbf{C}_i corresponding to node v . The second equality follows from the Sherman-Morrison formula.

The best node $v \in \mathcal{S}_i^c$ is then given by

$$v^* = \arg \max_{v \in \mathcal{S}_i^c} \psi \left(\mathbf{C}_{\mathcal{S}_{i+1}^c}^{i+1} \right), \quad \mathcal{S}_{i+1} = \mathcal{S}_i \cup \{v\}, \quad (5.29)$$

where $\psi(\cdot)$ can be chosen based on any of the criteria listed in Table 5.1. For sequential sampling using the E -optimality criterion, which seeks to minimize $\lambda_{\max}(\text{cov}_q(\mathbf{f}|\mathbf{b}_i, b_v))$, we can use the approximate greedy method in [2]. The method

in [2] computes the eigenvector of $\mathbf{L}_{\mathcal{S}^c}$ with the *smallest* eigenvalue at each iteration and samples the node where this eigenvector has the maximum absolute value. Note that $\mathbf{L}_{\mathcal{S}^c}$ is the inverse of the posterior covariance in the GRF model. With the Bayesian inference method explained in Section 5.2.2, we get the posterior covariance (as opposed to inverse covariance). Therefore, we need to compute its eigenvector with the *largest* eigenvalue and sample the node where this eigenvector has the maximum absolute value. This is because the eigenvector of the inverse covariance with the smallest eigenvalue equals the eigenvector of the covariance with the largest eigenvalue.

Once a new observation b_v is made, the posterior covariance is updated using the method explained in Section 5.2.2. Since the posterior covariance depends on the past observations, the sampling strategy based on the 1-Laplacian is adaptive.

5.3 Experiments

We compare the 1-Laplacian based adaptive active learning method with 2-Laplacian based active learning described in Chapter 4. We restrict our attention to binary classification problems in which each label $f_i \in \{+1, -1\}$. For fair comparison, we use the E -optimality criterion in the 1-Laplacian based sampling strategy, since the active learning method in Chapter 4 can also be interpreted to be an E -optimal strategy with prior distribution $p(\mathbf{f}) \propto \exp(-\mathbf{f}^\top \mathbf{L}^k \mathbf{f}/2)$ [32].

We apply the active learning methods for classification in two real world datasets. In the first example, we use a subset of the 20 Newsgroups dataset described in Chapter 4, containing 2 classes of documents, namely, {comp.sys.ibm.pc.hardware, comp.sys.mac.hardware}. We randomly choose 500 data points from each class to generate 10 instances of 1000 data points each. The feature vectors describing each document and the similarity graphs are computed as in Chapter 4. In the second example, we consider a subset of the Isolet spoken letters dataset described in Chapter 4 containing 2 classes corresponding to letters ‘f’ and ‘s’. Again, we randomly choose 200 data points from each class to generate 10 instances of 400 data points each and construct similarity graphs as described in Chapter 4.

In each instance of the dataset, with the graph constructed, we select the points to label using the 1-Laplacian based adaptive active learning method and the 2-Laplacian based active learning method of Chapter 4 with $k = 1$ and $k = 4$. The

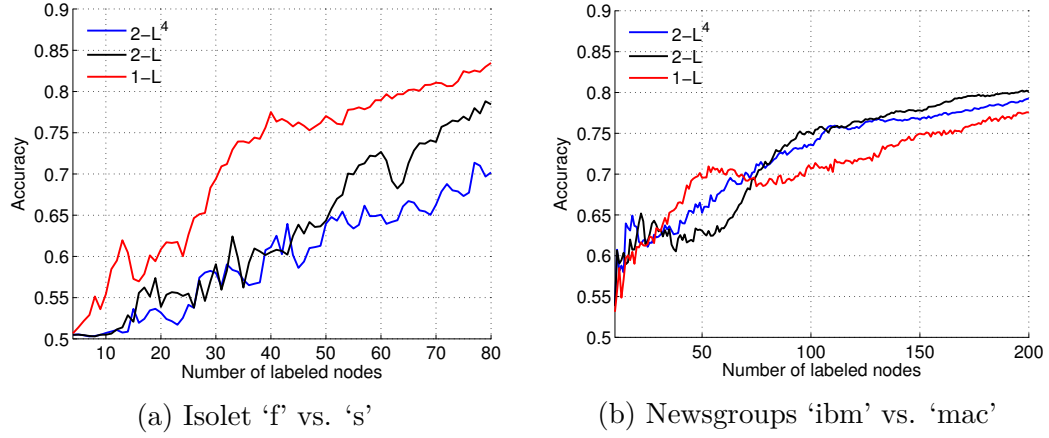


Figure 5.1: Plots show the average classification accuracies with different number of observed labels selected using 1-Laplacian based adaptive active learning (red) and the 2-Laplacian based active learning method of Chapter 4 with $k = 1$ (black) and $k = 4$ (blue).

rest of the labels are then predicted using the observed labels. We sample upto 20% of the data points in each of the 10 instances of datasets and report average prediction accuracies obtained with a given number of samples. The results are shown in Figure 5.1. We observe that the 1-Laplacian based adaptive sampling method gives much better accuracy than the 2-Laplacian based non-adaptive sampling method for the same number of observed labels in the Isolet dataset. For Newsgroups dataset, the 1-Laplacian based sampling method performs better in the early stages and the performance remains comparable as the number of observed labels increases.

5.4 Conclusion

In this chapter, we formulated the problem of active learning on a graph using the framework Bayesian experiment design. This formulation provides a unified view of different graph based active learning methods proposed in the literature. We showed that if the prior on the graph signal is assumed to be Gaussian and the prediction error is defined to be a functional of the prediction covariance, then a non-adaptive sampling strategy, (in which the choice of future samples does not depend on the labels observed in the past) is optimal.

We then proposed a new prior for graph signals using the concept of p -Laplacian. This prior is more suited for discrete valued signals which arise in classification problems (since the labels are discrete valued). We used the approximate Bayesian inference method in [75] to find the posterior covariance of the unobserved labels given the observed labels. This covariance matrix is then used to select the future nodes to be sampled. Due to the non-Gaussianity of the prior, the posterior covariance depends on the observed labels, leading to an adaptive sampling strategy. Experiments show that such an adaptive sampling method can give better accuracy than a non-adaptive sampling strategy based on the Gaussian prior with the same number of samples.

In future, we would like to understand if it is possible to develop a fast heuristic for computing the modified graph Laplacian corresponding to the inverse of the posterior covariance using the Laplacian of the original graph and the observed labels. This would enable us to avoid the expensive variational Bayesian inference step for computing the posterior covariance and make the proposed adaptive active learning method more scalable.

Chapter 6

Efficient Graph Construction from Data for Image Processing and Learning

In this chapter, we consider the problem of graph construction from data. Graphs provide a useful model for data in many applications. The nodes in the graph represent data points in some domain and the edges capture connectivity or similarity between the data points. The graph structure can be inherent to the data as is the case in applications such as social, transportation or communication networks, where the edges capture friendships or communication links. In many other applications, graph is *constructed from the data* (where each data point is represented by a vector in \mathbb{R}^d) in order to discover some underlying structure in it. Examples of such application include clustering [84], semi-supervised learning [94, 30], collaborative filtering [27, 58], outlier detection [7], compression [61] etc. Conventional data such as images and videos also benefit from a graph representation [62, 29]. Constructing a good graph is very important for graph based techniques to be effective in the aforementioned applications. In Chapters 4 and 5, we proposed algorithms for graph-based active semi-supervised learning. Although our proposed algorithms assume that the labels form a smooth signal on the graph, we did not address the problem of constructing a graph from data such that the smoothness assumption remains valid. In our experiments, we used the k -nearest neighbor method to construct the graph.

This chapter is partially based on our work in [29] and [33].

In graph based formulation of learning and clustering, we are given data in the form of N vectors in \mathbb{R}^d and a pairwise similarity kernel. The kernel matrix is a dense $N \times N$ matrix. Using graph based algorithms on the dense graph represented by the kernel matrix is computationally inefficient. Motivated by this, we consider the following question: *Is it possible to obtain from a dense kernel graph a sparse graph representation that has similar eigenstructure?* Two of the most widely used graph sparsification heuristics are the k -nearest neighbor (k -NN) method and the ϵ -neighborhood method. The k -NN method connects each node only to k of its most similar neighbors based on the kernel function value for each pair of data vectors. The ϵ -neighborhood method connects a node i to another node j if j falls within the ball of radius ϵ centered at i . Although these heuristic methods provide good results in many applications (see for example Chapter 4), they have no clear theoretical justification and are very sensitive to data noise.

We explore the potential for improvement in the performance of graph based techniques by constructing “good” graphs in a principled way. We can formulate the following “wish list” for a good graph representation for data:

1. **Sparsity:** The graph should be sparse, i.e., the number of edges should be of the same order as the number of nodes. This allows for near linear time implementation of different graph-based techniques, making them scalable to large data sets.
2. **Smoothness:** The signals of interest (defined on the nodes of the graph) should be smooth with respect to the graph Fourier transform (GFT). This ensures that two nodes connected by an edge share similar signal values, whereas nodes with very different signal values are not connected. Smoothness is a key assumption for successful application of GSP techniques.
3. **Complexity:** Complexity of graph construction algorithm should be small for it to be scalable to large data sets.

One of the applications of the sparse graph construction problem is to provide an efficient alternative to the well-known bilateral filter (BF) [83], which is widely used for edge-aware image filtering. The BF can be interpreted as a simple one-hop low pass filter on a graph [29] in which the nodes represent pixels and the edge weights capture the similarity between them as given by a positive definite similarity kernel that depends on the geometric as well as photometric distance

between the pixels¹. In the graph associated with the $k \times k$ BF, each node is connected to k^2 neighbors with weights given by the BF kernel function. For large values of k ($k = 5, 7, 9$ are commonly used), such a graph is very dense. Computational complexity of a single application of BF is roughly $O(mnk^2)$, where $m \times n$ is the image size. Using such dense BF graph to apply additional GSP tools (such as graph wavelets [62] or graph based regularization [29, 26]) for adaptive image processing can be even more computationally complex. Therefore, constructing a sparse image dependent graph using the BF kernel matrix is useful for efficient edge-aware image filtering.

Prior Work

The problem of graph construction from data has been studied in the past in various contexts such as graphical model estimation, non-linear dimensionality reduction, subspace clustering etc. In graphical model estimation, the nodes correspond to random variables in a Gaussian Markov random field (GMRF) and the observations are in the form of multiple realizations of these random variables. The goal is to estimate a sparse graph which represents the inverse covariance (or precision) matrix of the GMRF [55, 87, 28] using these observations. In some cases, it is desirable to get a precision matrix in the form of a graph Laplacian in order to use GSP methods on the estimated graph. Examples of such (generalized) Laplacian estimation methods include [80, 23, 21].

The problem setting considered in this chapter is different. Our goal is construct a graph in which each node represents a feature vector of a data point in \mathbb{R}^d . This problem has been considered before in the context of non-linear dimensionality reduction [69] and subspace clustering [25]. The local linear embedding (LLE) method in [69] constructs a graph based on representation of each data point as a linear combination of its neighbors. However, the method is not robust against data noise [15]. In subspace clustering, the data is assumed to be drawn from a union of subspaces. The goal is then to construct a graph such that two nodes are connected only if they belong to the same subspace [25]. This is achieved by representing each data point as a sparse linear combination of the rest. The sparsity regularizer used in the linear reconstructions makes this method robust

¹The discussion can be applied to other adaptive image filtering methods such as non-local means [8] and kernel regression [56] by changing the pairwise pixel similarity kernel

against data noise. However, when the data comes from an underlying non-linear manifold instead of a subspace (i.e., a linear manifold) such a model may not be appropriate.

Most of the above methods have a high computational complexity of $\Omega(N^2)$, where N is the number of nodes in the graph. Therefore, they are not feasible for use in large data sets.

Contributions

Given N data points represented by their feature vectors in \mathbb{R}^d and a positive definite pairwise similarity kernel, we propose an efficient method for constructing a graph that can be used in applications such as image filtering, clustering and semi-supervised learning. We interpret the kernel similarity between two feature vectors as the covariance between the signal samples on the two nodes. The motivation for this interpretation is the *cluster assumption* of semi-supervised learning [12], which states that any two similar data points are expected to have similar labels. We propose to estimate a sparse graph by approximating the inverse of the kernel matrix in the form of a generalized Laplacian (GL) (see Section 6.1). Our method can be thought of as a sparse estimator of the inverse covariance (or precision) matrix of a GMRF with sample covariance given by the kernel matrix. The zero entries of a GMRF precision matrix capture conditional independence relationships between the nodes. If the kernel similarity between data points decays rapidly as the distance between them increases, then it is reasonable to expect each node to be conditionally independent of others, given the nodes most similar to it. Therefore, we expect the inverse of the kernel similarity matrix to be well-approximated by a sparse matrix.

Eigenvectors of the proposed graph approximate the eigenvectors to the kernel matrix (because a matrix and its inverse have the same eigenvectors). Graph signals of interest (which are labels in learning applications and pixel intensities in image filtering) have a low frequency representation in the GFT of the graph represented by the kernel matrix because of the cluster assumption. Therefore, they are also low pass on the proposed graph.

Our proposed method estimates the graph that represents the inverse of the kernel matrix by performing minimum mean squared error (MMSE) regression at

each node with the feature vectors of the other nodes using the kernel inner product. Since the regressions are *kernelized*, the proposed method can be used even when the data is drawn from a non-linear manifold. The regression coefficients are forced to be non-negative in order to get a Laplacian. We also add an ℓ_1 regularization term in each regression problem in order to get a sparse Laplacian. The regularization term also provides robustness against noise in the data. Our method is similar in spirit to the sparse inverse covariance estimation method proposed in [87]. However, we restrict our estimate of the inverse of the kernel matrix to be in the form a graph Laplacian as in [23]. Because of the non-negativity constraint, the ℓ_1 regularized least squares regression problems are reduced to quadratic programs (QPs). Using the KKT conditions, we show that the superset of the support of solution of each regression problem can be obtained by thresholding the entries of the kernel matrix. Moreover, by choosing the regularization parameter carefully, the size of each QP can be made a constant (independent of N). The resulting algorithm has the same asymptotic computational complexity as the k -NN graph construction method.

We apply the proposed graph construction method to perform image filtering, spectral clustering and semi-supervised learning. Our results show that the proposed method has a superior performance compared to the k -NN method while being more robust to the choice of k and kernel parameters.

The rest of this chapter is organized as follows. In Section 6.1, we provide a brief review of the GMRF model for signals and a method for estimating a GL inverse of a positive definite matrix. Our proposed method for sparse graph estimation by inverting the kernel matrix is described in Section 6.2. Experimental results for image filtering, spectral clustering and semi-supervised learning are presented in Section 6.3. Section 6.4 concludes the chapter.

Notation

We use \mathbf{x}_{-i} to denote the subvector of \mathbf{x} with its i -th component removed and $\mathbf{M}_{-i,-j}$ to denote the submatrix of \mathbf{M} with its i -th row and j -th column removed. Notations such as $\mathbf{M}_{i,-j}$ (i.e., i -th row of \mathbf{M} with j -th entry removed) are defined similarly.

6.1 Laplacian Based Smoothness and GMRF

The GMRF model for signals [32, 88] (see also Chapter 5) is given by:

$$p(\mathbf{y}) \propto \exp \left(- \sum_{i,j} \mathbf{Q}_{ij} (\mathbf{y}_i - \mathbf{y}_j)^2 - \sum_i \mathbf{Q}_{ii} \mathbf{y}_i^2 \right). \quad (6.1)$$

The term in the exponent can be rewritten as $-\mathbf{y}^\top \mathbf{Q} \mathbf{y}$, where \mathbf{Q} is the inverse covariance (or precision) matrix. If \mathbf{Q} is a symmetric positive semi-definite matrix of the form $\alpha \mathbf{I} - \mathbf{N}$, where \mathbf{I} is the identity matrix and $\mathbf{N}_{ij} \geq 0 \ \forall i, j$, then it is called a generalized Laplacian (GL). Note that if $\mathbf{Q}_{ii} = \sum_j -\mathbf{Q}_{ij}$, then \mathbf{Q} is a Laplacian, while in general \mathbf{Q} can be interpreted as a Laplacian with self loops. If $\mathbf{y}^\top \mathbf{Q} \mathbf{y}$ is small, then \mathbf{y} will have high likelihood with respect to the GMRF. Therefore, if the GMRF is represented as a graph with Laplacian \mathbf{Q} , then a graph signal with high likelihood will be smooth on that graph.

For \mathbf{y} drawn with the distribution of (6.1), the conditional correlation between \mathbf{y}_i and \mathbf{y}_j given the rest of the variables $\text{corr}(\mathbf{y}_i, \mathbf{y}_j | \mathbf{y}_{k \neq i,j}) = -\mathbf{Q}_{ij} / \sqrt{\mathbf{Q}_{ii} \mathbf{Q}_{jj}}$. $\mathbf{y}_i, \mathbf{y}_j$ are conditionally independent iff $\mathbf{Q}_{ij} = 0$. Therefore, \mathbf{Q} is expected to be sparse [71]. \mathbf{Q} is also sparse in cases of interest, where a sparse graph is used to impose a Gaussian smoothness prior on signals (see Chapter 5).

An algorithm to estimate the inverse of a positive definite matrix \mathbf{K} in the form of a GL \mathbf{Q} has been proposed in [66]. It solves the following problem:

$$\min_{\mathbf{Q} \succeq \mathbf{0}; \mathbf{Q}_{ij} \leq 0, i \neq j} -\log \det(\mathbf{Q}) + \text{Tr}(\mathbf{K} \mathbf{Q}). \quad (6.2)$$

If \mathbf{K} is a sample covariance matrix, then the above problem can be thought of as a maximum likelihood estimation problem of a GMRF under GL constraints. Our approach for estimating a sparse graph from data is based on finding an approximate solution to a problem similar to (6.2) with \mathbf{K} given by the kernel matrix.

6.2 Proposed Graph Construction Method

We consider a set of data points $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N)$, where $\mathbf{x}_i \in \mathbb{R}^d$. Each data point \mathbf{x}_i has a label y_i associated with it. The labels can be binary, categorical or

real valued. Let $k: \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}^+$ be a symmetric positive definite kernel function which gives the similarity between two data points; $(\mathbf{x}_i, \mathbf{x}_j) \mapsto k(\mathbf{x}_i, \mathbf{x}_j)$. Examples of commonly used kernel functions are given below:

$$\begin{aligned} \text{Gaussian kernel: } & k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2), \\ \text{cosine kernel: } & k(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^\top \mathbf{x}_j / \|\mathbf{x}_i\| \|\mathbf{x}_j\|, \\ \text{polynomial kernel: } & k(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^\top \mathbf{x}_j + c)^d. \end{aligned} \tag{6.3}$$

$\mathbf{K} \in \mathbb{R}^{N \times N}$ denotes the kernel (or Gram) matrix with $\mathbf{K}_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$. We assume that $\mathbf{y} = (y_1, \dots, y_N)^\top$ follows a Gaussian Markov random field (GMRF) model with zero mean and the kernel matrix \mathbf{K} as an estimate of its covariance. Note that the Gram matrix is symmetric positive definite [43] and hence, satisfies the requirements of being a covariance matrix.

Our goal is to construct a sparse graph with N nodes, where each node i corresponds to a data point \mathbf{x}_i , such that the graph signal formed by the labels \mathbf{y} will be smooth with respect to that graph. Specifically, we propose to construct a graph Laplacian which is approximately equal to the inverse of \mathbf{K} .

6.2.1 Graph Estimation by Sparse Non-negative Regression

We assume that \mathbf{y} follows a multivariate Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{K})$. Then the conditional distribution of y_i given \mathbf{y}_{-i} equals

$$y_i | \mathbf{y}_{-i} \sim \mathcal{N}(\mathbf{K}_{i,-i} \mathbf{K}_{-i,-i}^{-1} \mathbf{y}_{-i}, \mathbf{K}_{ii} - \mathbf{K}_{i,-i} \mathbf{K}_{-i,-i}^{-1} \mathbf{K}_{-i,i}).$$

This can be thought of as the following regression equation [87]:

$$y_i = \mathbf{y}_{-i}^\top \boldsymbol{\theta} + \epsilon, \tag{6.4}$$

where $\boldsymbol{\theta} = \mathbf{K}_{-i,-i}^{-1} \mathbf{K}_{-i,i} \in \mathbb{R}^{N-1}$ is the coefficient vector and $\epsilon = \mathcal{N}(0, \mathbf{K}_{ii} - \mathbf{K}_{i,-i} \mathbf{K}_{-i,-i}^{-1} \mathbf{K}_{-i,i})$ is independent \mathbf{y}_{-i} . Using the block matrix inversion formula, it can be shown that [87] if $\boldsymbol{\Omega} = \mathbf{K}^{-1}$, then

$$\begin{aligned}\boldsymbol{\Omega}_{ii} &= (\mathbf{K}_{ii} - \mathbf{K}_{i,-i} \mathbf{K}_{-i,-i}^{-1} \mathbf{K}_{-i,i})^{-1} = (\text{var}(\epsilon))^{-1} \\ \boldsymbol{\Omega}_{i,-i} &= -\boldsymbol{\Omega}_{ii} \mathbf{K}_{i,-i} \mathbf{K}_{-i,-i}^{-1} = -(\text{var}(\epsilon))^{-1} \boldsymbol{\theta}.\end{aligned}\tag{6.5}$$

This shows that entries of the precision matrix can be estimated by the linear regression of y_i with \mathbf{y}_{-i} . Since we would like the precision matrix to be sparse and in the form of a Laplacian, we impose additional constraints on $\boldsymbol{\theta}$. Specifically, $\boldsymbol{\theta}$ should be sparse and each of its entry should be non-negative. To impose the sparsity constraint, we penalize the ℓ_1 norm of $\boldsymbol{\theta}$ to get the following Lasso-like problem with non-negativity constraints:

$$\begin{aligned}\underset{\boldsymbol{\theta}}{\text{minimize}} \quad & \frac{1}{2} \mathbb{E} [(y_i - \mathbf{y}_{-i}^\top \boldsymbol{\theta})^2] + \eta \|\boldsymbol{\theta}\|_1 \\ \text{subject to} \quad & \boldsymbol{\theta} \geq \mathbf{0},\end{aligned}\tag{6.6}$$

where $\boldsymbol{\theta} \geq \mathbf{0}$ is interpreted element-wise and η is the regularization parameter. Since $\boldsymbol{\theta} \geq \mathbf{0}$, $\|\boldsymbol{\theta}\|_1 = \mathbf{1}^\top \boldsymbol{\theta}$, where $\mathbf{1}$ denotes a vector of ones. Note that

$$\mathbb{E} [(y_i - \mathbf{y}_{-i}^\top \boldsymbol{\theta})^2] = \mathbf{K}_{ii} - 2\mathbf{K}_{-i,i}^\top \boldsymbol{\theta} + \boldsymbol{\theta}^\top \mathbf{K}_{-i,-i} \boldsymbol{\theta}.$$

Thus, (6.6) can be expressed as a quadratic program (QP):

$$\begin{aligned}\underset{\boldsymbol{\theta}}{\text{minimize}} \quad & \frac{1}{2} \boldsymbol{\theta}^\top \mathbf{K}_{-i,-i} \boldsymbol{\theta} - \mathbf{K}_{-i,i}^\top \boldsymbol{\theta} + \eta \mathbf{1}^\top \boldsymbol{\theta} \\ \text{subject to} \quad & \boldsymbol{\theta} \geq \mathbf{0}.\end{aligned}\tag{6.7}$$

To estimate the Laplacian, we solve (6.7) at each node i .

6.2.2 Simplification and Fast Computation

There exists an efficient iterative algorithm [18] for solving QPs of the form (6.7) at a quadratic convergence rate. The algorithm performs Cholesky factorization (or alternatively, preconditioned conjugate gradient descent) at each iteration, which has a numerical complexity of $O(N^3)$. Therefore, solving (6.7) directly is not

practical for large problems. Fortunately, the problem size can be greatly reduced since it is possible to obtain the support of $\boldsymbol{\theta}$ merely by thresholding the entries of $\mathbf{K}_{-i,i}$ as shown below.

Proposition 4. *Let $\boldsymbol{\theta}^*$ be a solution of (6.7). If $\eta > \mathbf{K}_{ij}$, then the corresponding entry of $\boldsymbol{\theta}^*$ equals zero, i.e., $\theta_l^* = 0$, where $l = j$ if $j \leq (i - 1)$ and $l = (j - 1)$ if $j \geq (i + 1)$.*

Proof. The Karush-Kuhn-Tucker (KKT) [6] optimality conditions for (6.7) are

$$\mathbf{K}_{-i,-i}\boldsymbol{\theta} + \eta\mathbf{1} - \mathbf{K}_{-i,i} - \boldsymbol{\lambda} = \mathbf{0} \quad (6.8)$$

$$\lambda_l \theta_l = 0, \quad l = 1, \dots, N - 1 \quad (6.9)$$

$$\boldsymbol{\theta} \geq \mathbf{0} \quad (6.10)$$

$$\boldsymbol{\lambda} \geq \mathbf{0}. \quad (6.11)$$

Since $\mathbf{K}_{-i,-i}\boldsymbol{\theta} \geq \mathbf{0}$, we must have $\eta\mathbf{1} - \mathbf{K}_{-i,i} - \boldsymbol{\lambda} \leq \mathbf{0}$ because of (6.8). If $\eta - \mathbf{K}_{ij} > 0$, then $\lambda_l > 0$. From (6.9), $\lambda_l > 0$ only if $\theta_l = 0$. Therefore, $\eta > \mathbf{K}_{ij} \Rightarrow \theta_l = 0$. \square

Let \mathcal{S} be the subset of indices j such that $\mathbf{K}_{ij} \geq \eta$. Then Proposition 4 implies that $\boldsymbol{\theta}_{\mathcal{S}^c}^* = \mathbf{0}$. Therefore, (6.7) can be simplified as

$$\begin{aligned} & \underset{\boldsymbol{\theta}}{\text{minimize}} \quad \frac{1}{2}\boldsymbol{\theta}^\top \mathbf{G}\boldsymbol{\theta} + (\eta\mathbf{1} - \mathbf{g})^\top \boldsymbol{\theta} \\ & \text{subject to} \quad \boldsymbol{\theta} \geq \mathbf{0}, \end{aligned} \quad (6.12)$$

where \mathbf{G} is the submatrix of $\mathbf{K}_{-i,-i}$ corresponding to the rows and columns in \mathcal{S} and \mathbf{g} is the corresponding subvector of $\mathbf{K}_{-i,i}$. If η is chosen such that at each node i , $\mathbf{K}_{ij} > \eta$ for only the top k values in $\mathbf{K}_{-i,i}$ (i.e., k nearest neighbors of i), then the sizes of \mathbf{G} and \mathbf{g} in (6.12) are $k \times k$ and $k \times 1$, respectively. Thus, the numerical complexity of each iterative step for solving (6.12) is reduced to $O(k^3)$ (which is independent of the total number of nodes N).

We solve (6.12) for every node. In order to obtain the edge weights for node i , the entries of the solution $\boldsymbol{\theta}^*$ of (6.12) need to be scaled by $(\text{var}(\epsilon))^{-1}$ as shown in (6.5). An estimate of $\text{var}(\epsilon)$ can be obtained as follows:

$$\text{var}(\epsilon) = \mathbb{E} \left[(y_i - \mathbf{y}_{-i}^\top \boldsymbol{\theta})^2 \right] = \mathbf{K}_{ii} - 2\mathbf{g}^\top \boldsymbol{\theta}^* + \boldsymbol{\theta}^{*\top} \mathbf{G} \boldsymbol{\theta}^*. \quad (6.13)$$

Thus, the row of the adjacency matrix \mathbf{W} corresponding to node i is given by $\mathbf{W}_{i,\mathcal{S}} = (\text{var}(\epsilon))^{-1}\boldsymbol{\theta}^*$ and $\mathbf{W}_{i,\mathcal{S}^c} = \mathbf{0}$. Note that the resulting adjacency matrix may not be symmetric. We symmetrize it by selecting $\mathbf{W}' := (\mathbf{W} + \mathbf{W}^\top)/2$.

6.2.3 Computational Complexity

The proposed method consists of two steps. The first step is to build a k -NN graph. Brute-force implementation of k -NN graph construction is $O(kN^2)$. Fortunately, there are efficient algorithms [20] to construct a k -NN graph in roughly $O(N^{1.14})$ time. Once the k -NN graph is constructed, we need to solve a QP of size k at each node (assuming thresholds η are chosen accordingly for each node). As mentioned before, there exists an efficient iterative algorithm for solving QPs [18]. Each iterative step of this algorithm has a complexity of k^3 . The algorithm typically requires less than 20 iterations to converge. The overall complexity of the second step of the proposed method is $O(Nk^3)$.

Thus, the proposed method can be implemented to run very efficiently with computational complexity which is nearly linear in the number of nodes N . Moreover, it is intrinsically suitable for distributed implementation since both the k -NN graph construction algorithm [20] and the QPs at each node utilize data only in the local neighborhood of each node.

6.2.4 Interpretation and Discussion

In order to gain some intuition about how the proposed method sets the graph weights, we consider a small example with three nodes, i, j and k . In order to find the weights \mathbf{W}_{ij} and \mathbf{W}_{ik} , we have to solve (6.12) with

$$\mathbf{G} = \begin{pmatrix} 1 & \mathbf{K}_{jk} \\ \mathbf{K}_{jk} & 1 \end{pmatrix} \text{ and } \mathbf{g} = \begin{pmatrix} \mathbf{K}_{ij} \\ \mathbf{K}_{ik} \end{pmatrix}.$$

Here we assume that $k(\mathbf{x}, \mathbf{x}) = 1$ for all \mathbf{x} , which is the case for most commonly used kernels such as the Gaussian kernel, the cosine kernel etc. For simplicity, assume that both $\boldsymbol{\theta}_j^*, \boldsymbol{\theta}_k^* > 0$ and $\eta = 0$. Hence,

$$\begin{pmatrix} \mathbf{W}_{ij} \\ \mathbf{W}_{ik} \end{pmatrix} = c \cdot \mathbf{G}^{-1} \mathbf{g} = c \begin{pmatrix} \mathbf{K}_{ij} - \mathbf{K}_{jk} \mathbf{K}_{ik} \\ \mathbf{K}_{ik} - \mathbf{K}_{jk} \mathbf{K}_{ij} \end{pmatrix},$$

where c is a constant independent of the index j, k . To compare the relative strength of connections, we compute

$$\mathbf{W}_{ij} - \mathbf{W}_{ik} = c(\mathbf{K}_{ij} - \mathbf{K}_{ik})(1 + \mathbf{K}_{jk}). \quad (6.14)$$

If $\mathbf{K}_{ij} > \mathbf{K}_{ik}$, then (6.14) shows that *in the resulting graph, strong similarity (\mathbf{K}_{ij}) between nodes i and j leads to a relatively even stronger connection between them whereas weak similarity (\mathbf{K}_{ik}) leads to a further weakened connection.*

6.3 Experiments

The proposed method of estimating a graph by local sparse non-negative kernel regressions (NNK) can be used as a post-processing step once a k -NN graph is constructed from the data. We test its performance in three applications, namely, image filtering, clustering and semi-supervised learning, where a symmetric, weighted k -NN graph is commonly used.

6.3.1 Image Denoising with Sparse Inverse BF Graph

The BF can be interpreted as a filter on a dense image dependent graph with edge weights given by the BF kernel

$$\mathbf{K}_{ij} = \exp\left(-\frac{\|\mathbf{p}_i - \mathbf{p}_j\|^2}{2\sigma_d^2}\right) \exp\left(-\frac{(\mathbf{f}_i - \mathbf{f}_j)^2}{2\sigma_r^2}\right), \quad (6.15)$$

where \mathbf{p}_i denotes the position of pixel i and \mathbf{f}_i is the pixel intensity. We use the proposed NNK method to construct a sparse graph that approximates the inverse of the BF kernel matrix.

We compare the performance of the NNC graph in image denoising with 7×7 BF graph and a sparse graph obtained using the heuristic similar to k -NN (with $k = 4$) in [33]. The NNC graph is obtained by selecting η at each node such that the set of non-zero entries in its adjacency matrix forms a subset of the set of non-zero entries in the adjacency matrix of the 4-NN like graph. We consider the problem of denoising four 16×16 blocks from the “Lena” image corrupted by i.i.d. Gaussian noise. We use the Wiener filters given by the GFTs computed using the different graphs. Wiener filter is the optimal filter for minimizing the MSE. Its spectral response [56] is given by $h^w(\lambda_i) = \tilde{\mathbf{f}}_i^2 / (\tilde{\mathbf{f}}_i^2 + \sigma_n^2)$, where σ_n^2 is the noise variance².

The PSNR values of the results for different images at various noise levels are shown in Table 6.1. The table shows that the proposed NNC graph gives better or comparable results to the 4-NN like graph. Both of these graphs (which approximate the inverse of the BF kernel matrix) perform better than the Wiener filter given by the dense 7×7 BF graph, while being much sparser (see Table 6.2).

6.3.2 Clustering

In this example, we consider the problem of two way clustering of the two moons dataset shown in Figure 6.1(a) using spectral clustering [84]. Each data point $\mathbf{x}_i \in \mathbb{R}^2$. Spectral clustering begins by constructing a graph from the data and uses the first m ($m = 2$ in this example) eigenvectors of the normalized graph Laplacian to find the clusters. We use the Gaussian kernel (6.3) to measure the similarity between any two points $\mathbf{x}_i, \mathbf{x}_j$ and construct symmetric, weighted k -NN graph.

Table 6.3 shows the accuracy of spectral clustering with k -NN and proposed graphs for different values of k and kernel width σ . From Table 6.3, we see that the proposed graph construction method consistently outperforms k -NN and its performance is more robust to the choice of k and σ than that of k -NN.

²Comparison between the 7×7 BF and a polynomial filter on the sparse inverse BF kernel graph is provided in [33]. It shows that the polynomial filter on the sparse inverse BF kernel graph gives superior denoising performance than the 7×7 BF with lower computational complexity

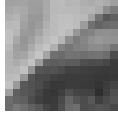


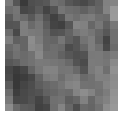
Block	Noisy	7×7 BF	4-NN	NNK
	15	26.09	27.11	27.15
	20	29.65	31.00	30.63
	25	33.98	34.45	34.50
	30	38.81	38.67	38.98
	15	28.07	28.92	28.20
	20	32.49	31.79	32.74
	25	36.05	35.91	36.75
	30	40.47	39.77	40.14
	15	39.88	39.58	36.32
	20	42.26	41.73	38.22
	25	43.85	43.19	41.71
	30	45.51	45.52	44.41
	15	25.82	26.59	26.64
	20	29.57	30.11	29.76
	25	33.64	34.15	33.95
	30	38.57	38.59	38.68

Table 6.1: PSNR (in dB) of images denoised using Wiener filters given by the GFTs computed using 7×7 BF graph, 4-NN heuristic graph and proposed NNK graph.

6.3.3 Semi-supervised Learning

In this example, we apply the proposed method for semi-supervised classification of the USPS handwritten digits data. This data consists of 1100 16×16 images each of digits 0 to 9. We use 100 randomly selected samples for each digit class to create one instance of our dataset. Thus, each instance consists of 1000 feature vectors (100 samples/class \times 10 digit classes) of dimension 256.

As in the previous example, we use the Gaussian kernel to measure the similarity between each pair of points $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^{256}$. We observe the labels of a randomly

Graph	7×7 BF	4-NN	NNK
avg. nnz	8649	1168	1021

Table 6.2: Average number of non-zero entries in the adjacency matrices of different graphs.

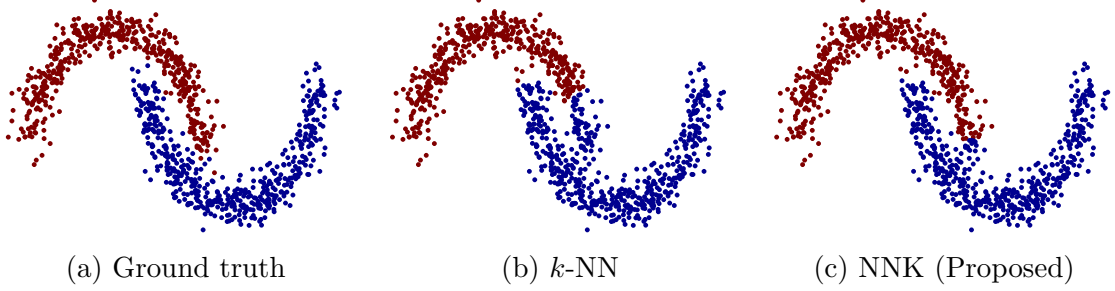


Figure 6.1: Spectral clustering with k -NN and proposed graph, $k = 10, \sigma = 0.1$

selected subset of the nodes in the graph and predict the labels of the rest of the nodes using the method in [94]. The experiment is performed using both the symmetric, weighted k -NN graph and the proposed NNK graph, which is obtained by selecting η at each node such that the set of non-zero entries in its adjacency matrix forms a subset of the set of non-zero entries in the adjacency matrix of the k -NN graph. The experiment is repeated over 10 instances of the dataset. Figure 6.2 shows the plots of average classification error vs. the number of observed labels with both types of graphs constructed using different values of k and Gaussian kernel width σ .

From Figure 6.2, we can make the following observations. Choice of σ is important in k -NN graph, especially when k is large. A very small σ can change similarity values significantly when distances vary only a little. On the other hand, when σ is very large even a large, change in distance does not significantly change the corresponding similarity value. Our proposed method gives better results than the k -NN graph construction method and is much more robust to the choice of k and σ .

	$k = 5$		$k = 10$		$k = 40$	
	k -NN	NNK	k -NN	NNK	k -NN	NNK
$\sigma = 0.1$	84.9	94.0	93.2	98.2	94.4	96.2
$\sigma = 1$	82.1	93.3	88.8	95.0	88.4	89.8
$\sigma = 4$	82.1	92.9	88.6	94.1	88.0	88.4

Table 6.3: Clustering accuracy with k -NN and NNK (proposed) graph for different values of k and σ

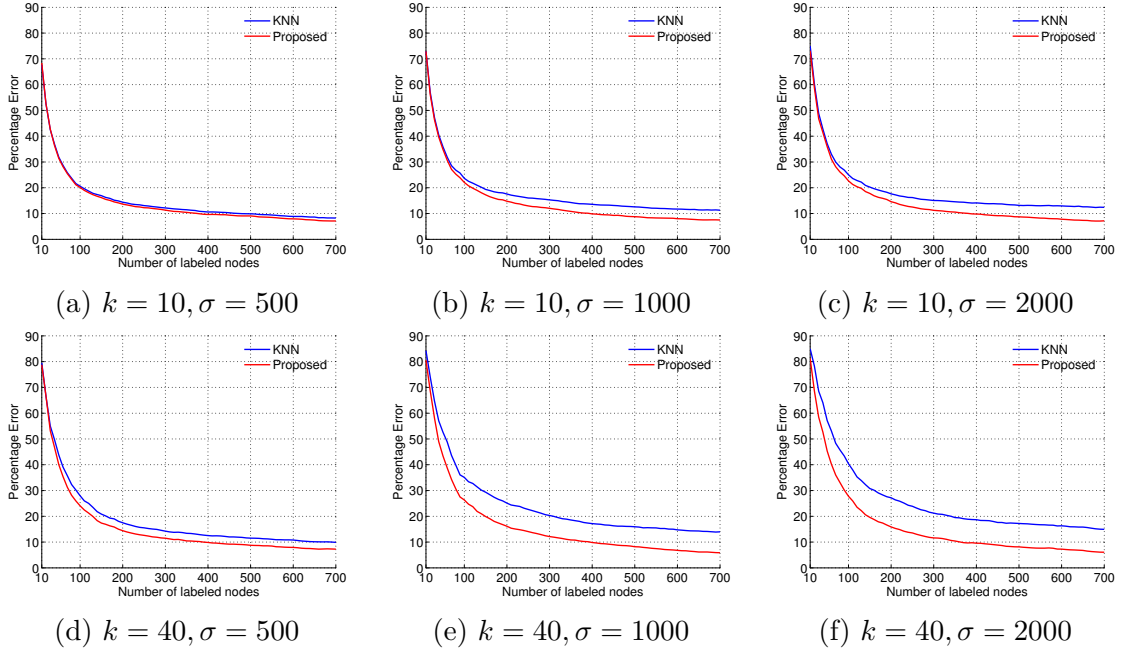


Figure 6.2: SSL results with k -NN and NNK (proposed) graph for different values of k and σ .

6.4 Summary

We proposed an efficient method for constructing a graph, given N data points in \mathbb{R}^d and a positive definite kernel function, which computes the similarity between each pair of data points. The proposed graph construction methods provides a computationally efficient alternative to the dense bilateral filter used for adaptive image filtering. It can also be used in graph based clustering and learning

algorithms. Our graph construction method approximates the inverse of the kernel matrix by representing each data point as a combination of other data points using ℓ_1 regularized non-negative kernel least squares regression. It has nearly the same computational complexity as the k -NN graph construction method and is expected to be robust to data noise. Numerical experiments show that the proposed method performs better in applications compared to the k -NN method and is more robust to the choice of k and kernel parameters.

Chapter 7

Conclusions and Future Work

7.1 Main Contributions

In this thesis, we proposed efficient techniques for sampling and reconstruction of signals defined on nodes of a graph. The proposed graph sampling theory is based on the notion of graph Fourier transform (GFT) defined using eigenvalues and eigenvectors of operators that measure the variation in a graph signal taking into account the edge connectivity. We gave necessary and sufficient conditions under which a graph signal bandlimited in the GFT domain can be uniquely and stably reconstructed from its samples on a subset of the nodes. Using this condition, we proposed an efficient algorithm for selecting a good sampling set that allows for stable bandlimited reconstructions that are robust against sampling noise and model mismatch. We also provided an efficient algorithm for obtaining an approximate bandlimited reconstruction with the observed samples using graph filters in the form of polynomials in the variation operator. The main advantage of our approach over other methods proposed in the past is that, although it is motivated by the GFT, our approach does not require computation of the GFT basis vectors. The proposed methods access the variation operator only via matrix-vector multiplication and thus allow distributed and localized implementation. They can also be applied to directed graphs with appropriate variation operators.

We gave a probabilistic interpretation of graph sampling theory by posing the sampling set selection problem as an experiment design problem for minimizing predictive covariance assuming graph based smoothness prior for signals. Based on this interpretation, we extended our sampling framework to adaptive sampling, where the future choice of nodes to be sampled depends on the samples observed

in the past. This extension used the concept of 1-Laplacian to define a graph signal prior better suited for discrete valued signals. We considered an application of the proposed sampling and reconstruction methods to graph based active semi-supervised learning in detail. Our approach gives better classification accuracy with a given number of labels than other state of the art methods in different real world data sets.

Finally, we proposed an efficient method for constructing a sparse graph from given data in the form of vectors in \mathbb{R}^d and a pairwise similarity kernel, which is the first step in graph based learning and clustering approaches. Our proposed method has roughly the same computational complexity as the k -NN graph construction method. In addition to graph based clustering and semi-supervised learning, we applied the our method to provide an efficient alternative to the bilateral filter, a well-known tool for edge-aware image filtering. The proposed graph leads to superior performance in spectral clustering, semi-supervised learning and image denoising compared to the k -NN method.

7.2 Future Work

There are several important questions that we would like to consider in future. The greedy sampling set selection algorithm proposed in Chapter 3 gives only an approximate solution to the combinatorial cutoff frequency maximization problem. It would be useful to find a polynomial time algorithm with theoretical guarantees on the quality of approximation.

In our adaptive sampling method from Chapter 5, the graph is modified based on the observed samples using an approximate Bayesian inference method. We would like to find a fast heuristic for computing the modified graph in order to make the proposed adaptive sampling method more scalable. Such a heuristic can also allow us to construct better graphs from data, which would lead to improved semi-supervised learning performance.

The adaptive sampling framework can be generalized even further by considering different graph signal priors, observation likelihood models and error metrics. An interesting theoretical problem is to provide bounds on the sampling complexity (i.e., the number of samples required to achieve desired reconstruction accuracy) for different graph signal models.

Since our methods for graph signal sampling and reconstruction access the graph variation operator only via matrix-vector multiplications, these can be implemented using a distributed graph processing framework such as GraphLab [53] in order to make them applicable to large datasets.

In Chapter 6, we constructed an edge-aware, sparse graph representation for images by approximating the inverse of the BF kernel matrix and considered its application in image denoising. In future, we would like to consider image dependent graphs obtained using similarity kernels corresponding to different edge-aware image filters such as non-local means [8] and locally adaptive kernel repression [82]. Application of these graphs in other image processing tasks such as super-resolution, in-painting and compression would be also useful.

Reference List

- [1] A. Anis, A. Gadde, and A. Ortega, “Towards a sampling theorem for signals on arbitrary graphs,” in *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on*, 2014.
- [2] —, “Efficient sampling set selection for bandlimited graph signals using graph spectral proxies,” *IEEE Transactions on Signal Processing*, 2016, arXiv:1510.00297 [cs.IT].
- [3] M. Belkin and P. Niyogi, “Semi-supervised learning on Riemannian manifolds,” *Machine learning*, vol. 56, no. 1-3, pp. 209–239, 2004.
- [4] M. Belkin, P. Niyogi, and V. Sindhwani, “Manifold regularization: A geometric framework for learning from labeled and unlabeled examples,” *The Journal of Machine Learning Research*, vol. 7, pp. 2399–2434, 2006.
- [5] A. Blum and S. Chawla, “Learning from labeled and unlabeled data using graph mincuts,” in *Proceedings of the 18th International Conference on Machine Learning*, 2001, pp. 19–26.
- [6] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2009.
- [7] M. Brito, E. Chavez, A. Quiroz, and J. Yukich, “Connectivity of the mutual k -nearest neighbor graph in clustering and outlier detection,” *Statistics & Probability Letters*, vol. 35, no. 1, pp. 33–42, 1997.
- [8] A. Buades, B. Coll, and J.-M. Morel, “A review of image denoising algorithms, with a new one,” *Multiscale Modeling & Simulation*, vol. 4, no. 2, pp. 490–530, 2005.
- [9] T. Bühler and M. Hein, “Spectral clustering based on the graph p -laplacian,” in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 81–88.
- [10] K. Chaloner and I. Verdinelli, “Bayesian experimental design: A review,” *Statistical Science*, pp. 273–304, 1995.

- [11] Y.-H. Chao, A. Ortega, W. Hu, and G. Cheung, “Edge-adaptive depth map coding with lifting transform on graphs,” in *Picture Coding Symposium*, 2015.
- [12] O. Chapelle, B. Schölkopf, A. Zien *et al.*, *Semi-supervised learning*. MIT Press, Cambridge, 2006.
- [13] S. Chen, A. Sandryhaila, and J. Kovacevic, “Sampling theory for graph signals,” in *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on*, April 2015.
- [14] S. Chen, R. Varma, A. Sandryhaila, and J. Kovacevic, “Discrete signal processing on graphs: Sampling theory,” *Signal Processing, IEEE Transactions on*, 2015.
- [15] B. Cheng, J. Yang, S. Yan, Y. Fu, and T. S. Huang, “Learning with ℓ^1 -graph for image analysis,” *IEEE transactions on image processing*, vol. 19, no. 4, pp. 858–866, 2010.
- [16] F. R. K. Chung, *Spectral graph theory*. CBMS Regional Conference Series in Mathematics, AMS, 1997, vol. 92.
- [17] F. Chung, “Laplacians and the Cheeger inequality for directed graphs,” *Annals of Combinatorics*, vol. 9, no. 1, pp. 1–19, 2005.
- [18] T. F. Coleman and Y. Li, “A reflective Newton method for minimizing a quadratic function subject to bounds on some of the variables,” *SIAM Journal on Optimization*, vol. 6, no. 4, pp. 1040–1058, 1996.
- [19] C. Currin, T. Mitchell, M. Morris, and D. Ylvisaker, “Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments,” *Journal of the American Statistical Association*, vol. 86, no. 416, pp. 953–963, 1991.
- [20] W. Dong, C. Moses, and K. Li, “Efficient k-nearest neighbor graph construction for generic similarity measures,” in *Proceedings of the 20th international conference on World wide web*. ACM, 2011, pp. 577–586.
- [21] X. Dong, D. Thanou, P. Frossard, and P. Vandergheynst, “Learning laplacian matrix in smooth graph signal representations,” *IEEE Transactions on Signal Processing*, vol. 64, no. 23, pp. 6160–6173, 2016.
- [22] F. Dorfler and F. Bullo, “Kron reduction of graphs with applications to electrical networks,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 60, no. 1, pp. 150–163, 2013.

- [23] H. Egilmez, E. Pavez, and A. Ortega, “Graph learning from data under structural and laplacian constraints,” 2016, arXiv:1611.05181 [cs.LG].
- [24] Y. C. Eldar, “Sampling with arbitrary sampling and reconstruction spaces and oblique dual frame vectors,” *Journal of Fourier Analysis and Applications*, vol. 9, no. 1, pp. 77–96, 2003.
- [25] E. Elhamifar and R. Vidal, “Sparse subspace clustering: Algorithm, theory, and applications,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 11, pp. 2765–2781, 2013.
- [26] A. Elmoataz, O. Lezoray, and S. Bougleux, “Nonlocal discrete regularization on weighted graphs: a framework for image and manifold processing,” *IEEE Transactions on Image Processing*, vol. 17, no. 7, pp. 1047–1060, 2008.
- [27] F. Fouss, A. Pirotte, J.-M. Renders, and M. Saerens, “Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation,” *IEEE Transactions on knowledge and data engineering*, vol. 19, no. 3, 2007.
- [28] J. Friedman, T. Hastie, and R. Tibshirani, “Sparse inverse covariance estimation with the graphical lasso,” *Biostatistics*, vol. 9, no. 3, pp. 432–441, 2008.
- [29] A. Gadde, S. Narang, and A. Ortega, “Bilateral filter: Graph spectral interpretation and extensions,” in *Image Processing (ICIP), 2013 20th IEEE International Conference on*, Sept 2013, pp. 1222–1226.
- [30] A. Gadde, A. Anis, and A. Ortega, “Active semi-supervised learning using sampling theory for graph signals,” in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 492–501.
- [31] A. Gadde, A. Knyazev, D. Tian, and H. Mansour, “Guided signal reconstruction with application to image magnification,” in *Signal and Information Processing (GlobalSIP), IEEE Global Conference on*, 2015, arXiv:1509.02465 [cs.IT].
- [32] A. Gadde and A. Ortega, “A probabilistic interpretation of sampling theory of graph signals,” in *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on*, April 2015, arXiv:1503.06629 [cs.LG].
- [33] A. Gadde, M. Xu, and A. Ortega, “Sparse inverse bilateral filter for image processing,” in *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on*, 2017.

- [34] R. Garnett, Y. Krishnamurthy, X. Xiong, R. Mann, and J. Schneider, “Bayesian optimal active search and surveying,” in *International Conference on Machine Learning (ICML)*, 2012, pp. 1239–1246.
- [35] R. Gerchberg, “Super-resolution through error energy reduction,” *Journal of Modern Optics*, vol. 21, no. 9, pp. 709–720, 1974.
- [36] K. Gröchenig, “A discrete theory of irregular sampling,” *Linear Algebra and its applications*, vol. 193, pp. 129–150, 1993.
- [37] Q. Gu and J. Han, “Towards active learning on graphs: An error bound minimization approach,” in *Proceedings of 12th IEEE International Conference on Data Mining*, 2012, pp. 882–887.
- [38] Q. Gu, T. Zhang, C. Ding, and J. Han, “Selective labeling via error bound minimization,” in *Advances in Neural Information Processing Systems 25*, 2012, pp. 332–340.
- [39] A. Guillory and J. Bilmes, “Label selection on graphs,” in *Advances in Neural Information Processing Systems 22*, 2009, pp. 691–699.
- [40] —, “Active semi-supervised learning using submodular functions,” in *Proceedings of 27th Conference on Uncertainty in Artificial Intelligence*, 2011, pp. 274–282.
- [41] W. H. Haemers, “Interlacing eigenvalues and graphs,” *Linear Algebra and its applications*, vol. 226, pp. 593–616, 1995.
- [42] D. Hammond, P. Vandergheynst, and R. Gribonval, “Wavelets on graphs via spectral graph theory,” *Applied and Computational Harmonic Analysis*, vol. 30, no. 2, pp. 129 – 150, 2011.
- [43] T. Hofmann, B. Schölkopf, and A. J. Smola, “Kernel methods in machine learning,” *The Annals of Statistics*, pp. 1171–1220, 2008.
- [44] S. C. Hoi, R. Jin, J. Zhu, and M. R. Lyu, “Batch mode active learning and its application to medical image classification,” in *Proceedings of the 23rd International conference on Machine learning*, 2006, pp. 417–424.
- [45] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge University Press, 2012.
- [46] M. Ji and J. Han, “A variance minimization criterion to active learning on graphs,” in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, vol. 22, 2012, pp. 556–564.

- [47] S. Joshi and S. Boyd, “Sensor selection via convex optimization,” *Signal Processing, IEEE Transactions on*, vol. 57, no. 2, pp. 451–462, 2009.
- [48] G. Karypis and V. Kumar, “A fast and high quality multilevel scheme for partitioning irregular graphs,” *SIAM Journal on Scientific Computing*, vol. 20, no. 1, 1998.
- [49] J. M. Kleinberg, “Authoritative sources in a hyperlinked environment,” *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999.
- [50] A. Knyazev, A. Jujunashvili, and M. Argentati, “Angles between infinite dimensional subspaces with applications to the Rayleigh–Ritz and alternating projectors methods,” *Journal of Functional Analysis*, vol. 259, no. 6, pp. 1323–1345, 2010.
- [51] A. V. Knyazev, “Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method,” *SIAM journal on scientific computing*, vol. 23, no. 2, pp. 517–541, 2001.
- [52] A. Krause, A. Singh, and C. Guestrin, “Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies,” *The Journal of Machine Learning Research*, vol. 9, pp. 235–284, 2008.
- [53] Y. Low, D. Bickson, J. Gonzalez, C. Guestrin, A. Kyrola, and J. M. Hellerstein, “Distributed Graphlab: A framework for machine learning and data mining in the cloud,” *Proc. VLDB Endow.*, vol. 5, no. 8, pp. 716–727, Apr. 2012.
- [54] Y. Ma, R. Garnett, and J. Schneider, “ Σ -optimality for active learning on Gaussian random fields,” in *Advances in Neural Information Processing Systems*, 2013, pp. 2751–2759.
- [55] N. Meinshausen and P. Bühlmann, “High-dimensional graphs and variable selection with the lasso,” *The Annals of Statistics*, vol. 34, no. 3, pp. 1436–1462, 06 2006.
- [56] P. Milanfar, “A tour of modern image filtering: New insights and methods, both practical and theoretical,” *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 106–128, Jan. 2013.
- [57] S. K. Narang, A. Gadde, and A. Ortega, “Signal processing techniques for interpolation in graph structured data,” in *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on*, 2013, pp. 5445–5449.

- [58] S. K. Narang, A. Gadde, E. Sanou, and A. Ortega, “Localized iterative methods for interpolation in graph structured data,” in *Signal and Information Processing (GlobalSIP), IEEE Global Conference on*, 2013.
- [59] S. Narang and A. Ortega, “Local two-channel critically sampled filter-banks on graphs,” in *Image Processing (ICIP), 2010 17th IEEE International Conference on*, Sept 2010, pp. 333–336.
- [60] —, “Perfect reconstruction two-channel wavelet filter banks for graph structured data,” *IEEE Transactions on Signal Processing*, vol. 60, no. 6, pp. 2786–2799, June 2012.
- [61] —, “Compact support biorthogonal wavelet filterbanks for arbitrary undirected graphs,” *Signal Processing, IEEE Transactions on*, vol. 61, no. 19, pp. 4673–4685, Oct 2013.
- [62] S. K. Narang, Y. H. Chao, and A. Ortega, “Graph-wavelet filterbanks for edge-aware image processing,” in *Statistical Signal Processing Workshop (SSP), IEEE*, 2012, pp. 141–144.
- [63] H. Nguyen and M. Do, “Downsampling of signals on graphs via maximum spanning trees,” *Signal Processing, IEEE Transactions on*, vol. 63, no. 1, pp. 182–191, Jan 2015.
- [64] B. Osting, C. D. White, and E. Oudet, “Minimal Dirichlet energy partitions for graphs,” Aug. 2013, arXiv:1308.4915 [math.OC].
- [65] A. Papoulis, “A new algorithm in spectral analysis and band-limited extrapolation,” *Circuits and Systems, IEEE Transactions on*, vol. 22, no. 9, pp. 735–742, 1975.
- [66] E. Pavez and A. Ortega, “Generalized Laplacian precision matrix estimation for graph signal processing,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016, pp. 6350–6354.
- [67] I. Pesenson, “Sampling in Paley-Wiener spaces on combinatorial graphs,” *Transactions of the American Mathematical Society*, vol. 360, no. 10, pp. 5603–5627, 2008.
- [68] —, “Variational splines and Paley–Wiener spaces on combinatorial graphs,” *Constructive Approximation*, 2009.
- [69] S. T. Roweis and L. K. Saul, “Nonlinear dimensionality reduction by locally linear embedding,” *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

- [70] L. I. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” *Physica D: Nonlinear Phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.
- [71] H. Rue and L. Held, *Gaussian Markov random fields: Theory and applications*. CRC Press, 2005.
- [72] A. Sandryhaila and J. Moura, “Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure,” *Signal Processing Magazine, IEEE*, vol. 31, no. 5, pp. 80–90, Sept 2014.
- [73] —, “Discrete signal processing on graphs: Frequency analysis,” *Signal Processing, IEEE Transactions on*, vol. 62, no. 12, pp. 3042–3054, June 2014.
- [74] K. Sauer and J. Allebach, “Iterative reconstruction of bandlimited images from nonuniformly spaced samples,” *Circuits and Systems, IEEE Transactions on*, vol. 34, no. 12, pp. 1497–1506, 1987.
- [75] M. W. Seeger and H. Nickisch, “Large scale Bayesian inference and experimental design for sparse linear models,” *SIAM Journal on Imaging Sciences*, vol. 4, no. 1, pp. 166–199, 2011.
- [76] B. Settles, “Active learning literature survey,” University of Wisconsin–Madison, Computer Sciences Technical Report 1648, 2010.
- [77] H. Shomorony and A. Avestimehr, “Sampling large data on graphs,” in *Signal and Information Processing (GlobalSIP), 2014 IEEE Global Conference on*, Dec 2014, pp. 933–936.
- [78] D. I. Shuman, M. J. Faraji, and P. Vandergheynst, “A multiscale pyramid transform for graph signals,” *IEEE Transactions on Signal Processing*, vol. 64, no. 8, pp. 2119–2134.
- [79] D. Shuman, S. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, “The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains,” *Signal Processing Magazine, IEEE*, vol. 30, no. 3, pp. 83–98, May 2013.
- [80] M. Slawski and M. Hein, “Estimation of positive definite M-matrices and structure learning for attractive Gaussian Markov random fields,” *Linear Algebra and its Applications*, vol. 473, pp. 145–179, 2015.
- [81] A. J. Smola and R. Kondor, “Kernels and regularization on graphs,” in *Learning theory and kernel machines*. Springer, 2003, pp. 144–158.

- [82] H. Takeda, S. Farsiu, and P. Milanfar, “Kernel regression for image processing and reconstruction,” *Image Processing, IEEE Transactions on*, vol. 16, no. 2, pp. 349–366, 2007.
- [83] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images,” in *Computer Vision, 1998. Sixth International Conference on*. IEEE, 1998, pp. 839–846.
- [84] U. Von Luxburg, “A tutorial on spectral clustering,” *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.
- [85] D. Youla and H. Webb, “Image restoration by the method of convex projections : Part I - Theory,” *Medical Imaging, IEEE Transactions on*, vol. 1, no. 2, pp. 81–94, 1982.
- [86] K. Yu, J. Bi, and V. Tresp, “Active learning via transductive experimental design,” in *Proceedings of the 23rd International conference on Machine learning*, 2006, pp. 1081–1088.
- [87] M. Yuan, “High dimensional inverse covariance matrix estimation via linear programming,” *Journal of Machine Learning Research*, vol. 11, no. Aug, pp. 2261–2286, 2010.
- [88] C. Zhang, D. Florencio, and P. A. Chou, “Graph signal processing: A probabilistic framework,” Tech. Rep., April 2015. [Online]. Available: <https://www.microsoft.com/en-us/research/publication/graph-signal-processing-a-probabilistic-framework/>
- [89] L. Zhang, C. Chen, J. Bu, D. Cai, X. He, and T. S. Huang, “Active learning based on locally linear reconstruction,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 10, pp. 2026–2038, 2011.
- [90] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, “Learning with local and global consistency,” in *Advances in Neural Information Processing Systems 16*, 2004.
- [91] D. Zhou, T. Hofmann, and B. Schölkopf, “Semi-supervised learning on directed graphs,” in *Advances in Neural Information Processing Systems 17*, L. K. Saul, Y. Weiss, and L. Bottou, Eds. Cambridge, MA: MIT Press, 2004, pp. 1633–1640.
- [92] D. Zhou, J. Huang, and B. Schölkopf, “Learning from labeled and unlabeled data on a directed graph,” in *Proceedings of the 22nd International conference on Machine learning*, 2005, pp. 1036–1043.

- [93] X. Zhou and M. Belkin, “Semi-supervised learning by higher order regularization,” in *International Conference on Artificial Intelligence and Statistics*, 2011, pp. 892–900.
- [94] X. Zhu, Z. Ghahramani, and J. Lafferty, “Semi-supervised learning using Gaussian fields and harmonic functions,” in *International Conference on Machine Learning (ICML)*, vol. 3, 2003, pp. 912–919.
- [95] X. Zhu, “Semi-supervised learning literature survey,” Computer Sciences, University of Wisconsin-Madison, Tech. Rep. 1530, 2008.
- [96] X. Zhu, J. Lafferty, and Z. Ghahramani, “Combining active learning and semi-supervised learning using Gaussian fields and harmonic functions,” in *ICML 2003 workshop on the continuum from labeled to unlabeled data in machine learning and data mining*, 2003, pp. 58–65.

Appendix A

Properties of Spectral Proxies

In this section, we prove the monotonicity and convergence properties of $\omega_k(\mathbf{f})$.

Proposition 5. *If \mathbf{L} has real eigenvalues and eigenvectors, then for any $k_1 < k_2$, we have $\omega_{k_1}(\mathbf{f}) \leq \omega_{k_2}(\mathbf{f})$, $\forall \mathbf{f}$.*

Proof. We first expand $\omega_{k_1}(\mathbf{f})$ as follows:

$$\begin{aligned} (\omega_{k_1}(\mathbf{f}))^{2k_1} &= \left(\frac{\|\mathbf{L}^{k_1} \mathbf{f}\|}{\|\mathbf{f}\|} \right)^2 \\ &= \frac{\sum_{i,j} (\lambda_i \lambda_j)^{k_1} \tilde{\mathbf{f}}_i \tilde{\mathbf{f}}_j \mathbf{u}_i^\top \mathbf{u}_j}{\sum_{i,j} \tilde{\mathbf{f}}_i \tilde{\mathbf{f}}_j \mathbf{u}_i^\top \mathbf{u}_j} \end{aligned} \quad (\text{A.1})$$

$$= \sum_{i,j} (\lambda_i \lambda_j)^{k_1} c_{ij} \quad (\text{A.2})$$

where $c_{ij} = \tilde{\mathbf{f}}_i \tilde{\mathbf{f}}_j \mathbf{u}_i^\top \mathbf{u}_j / \sum_{i,j} \tilde{\mathbf{f}}_i \tilde{\mathbf{f}}_j \mathbf{u}_i^\top \mathbf{u}_j$. Now, consider the function $f(x) = x^{k_2/k_1}$. Note that since $k_1 < k_2$, $f(x)$ is a convex function. Further, since $\sum_{i,j} c_{ij} = 1$, we can use Jensen's inequality in the above equation to get

$$\left(\sum_{i,j} (\lambda_i \lambda_j)^{k_1} c_{ij} \right)^{k_2/k_1} \leq \sum_{i,j} \left((\lambda_i \lambda_j)^{k_1} \right)^{k_2/k_1} c_{ij} \quad (\text{A.3})$$

$$\begin{aligned} \Rightarrow \left(\sum_{i,j} (\lambda_i \lambda_j)^{k_1} c_{ij} \right)^{1/2k_1} &\leq \left(\sum_{i,j} (\lambda_i \lambda_j)^{k_2} c_{ij} \right)^{1/2k_2} \\ \Rightarrow \omega_{k_1}(\mathbf{f}) &\leq \omega_{k_2}(\mathbf{f}) \end{aligned} \quad (\text{A.4})$$

If \mathbf{L} has real entries, but complex eigenvalues and eigenvectors, then these occur in conjugate pairs, hence, the above summation is real. However, in that case, $\omega_k(\mathbf{f})$

is not guaranteed to increase in a monotonous fashion, since c_{ij} 's are not real and Jensen's inequality breaks down. \square

Proposition 6. *Let $\omega(\mathbf{f})$ be the bandwidth of any signal \mathbf{f} . Then, the following holds:*

$$\omega(\mathbf{f}) = \lim_{k \rightarrow \infty} \omega_k(\mathbf{f}) = \lim_{k \rightarrow \infty} \left(\frac{\|\mathbf{L}^k \mathbf{f}\|}{\|\mathbf{f}\|} \right)^{1/k} \quad (\text{A.5})$$

Proof. We first consider the case when \mathbf{L} has real eigenvalues and eigenvectors. Let $\omega(\mathbf{f}) = \lambda_p$, then we have:

$$\omega_k(\mathbf{f}) = \left(\frac{\sum_{i,j=1}^p (\lambda_i \lambda_j)^k \tilde{\mathbf{f}}_i \tilde{\mathbf{f}}_j \mathbf{u}_i^\top \mathbf{u}_j}{\sum_{i,j=1}^p \tilde{\mathbf{f}}_i \tilde{\mathbf{f}}_j \mathbf{u}_i^\top \mathbf{u}_j} \right)^{1/2k} \quad (\text{A.6})$$

$$= \lambda_p \left(c_{pp} + \sum_{(i,j) \neq (p,p)} \left(\frac{\lambda_i}{\lambda_p} \frac{\lambda_j}{\lambda_p} \right)^k c_{ij} \right)^{1/2k} \quad (\text{A.7})$$

where $c_{ij} = \tilde{\mathbf{f}}_i \tilde{\mathbf{f}}_j \mathbf{u}_i^\top \mathbf{u}_j / \sum_{i,j} \tilde{\mathbf{f}}_i \tilde{\mathbf{f}}_j \mathbf{u}_i^\top \mathbf{u}_j$. Taking limits, it is easy to observe that the term in parentheses evaluates to 1. Hence, we have

$$\lim_{k \rightarrow \infty} \omega_k(\mathbf{f}) = \lambda_p = \omega(\mathbf{f}) \quad (\text{A.8})$$

Now, if \mathbf{L} has complex eigenvalues and eigenvectors, then these have to occur in conjugate pairs since \mathbf{L} has real entries. Hence, for this case, we do a similar expansion as above and take $|\lambda_p|$ out of the expression. Then, the limit of the remaining term is once again equal to 1. \square