

SPECTRAL ANALYSIS OF RICH NETWORK TOPOLOGY IN SOCIAL
NETWORKS

by

Leting Wu

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Computing and Information Systems

Charlotte

2013

Approved by:

Dr. Xintao Wu

Dr. Aidong Lu

Dr. Zbyszek W. Ras

Dr. Wensheng Wu

Dr. Zhiyi Zhang

©2013
Leting Wu
ALL RIGHTS RESERVED

ABSTRACT

LETING WU. Spectral analysis of rich network topology in social networks. (Under the direction of DR. XINTAO WU)

Social networks have received much attention these days. Researchers have developed different methods to study the structure and characteristics of the network topology. Our focus is on spectral analysis of the adjacency matrix of the underlying network. Recent work showed good properties in the adjacency spectral space but there are few theoretical and systematical studies to support their findings.

In this dissertation, we conduct an in-depth theoretical study to show the close relationship between algebraic spectral properties of the adjacency matrix and various patterns in a broad range of social networks such as friendship networks, alliance and war networks, and distrusted networks. In our framework, we apply matrix perturbation theory and approximate the eigenvectors of real graphs by those of the ideal cases. Our theoretical results show that the principal eigenvectors capture the structure of major communities and exhibit them as orthogonal lines/clusters rotated with certain angles from canonical axes. Our results also show that the minor eigenvectors with skew distributions in values capture weak or subtle signals hidden in local communities. We utilize our theoretical results to develop algorithms for several problems in social network analysis including community partition, anomaly detection and privacy preserving social network reconstruction. Empirical evaluations on various synthetic data and real-world social networks validate our theoretical findings and show the effectiveness of our algorithms.

In a nutshell, we theoretically study the patterns in the adjacency spectral space as well as conditions for their existence and explore the application of the spectral properties of the adjacency matrix in different tasks of social network analysis.

ACKNOWLEDGMENTS

In pursuing this PhD. degree, I have received many helps, suggestions and encouragements from many people. I hereby express my sincere gratitude to all of them.

First of all, I would like to thank for my dissertation committee members, Dr. Xintao Wu, Dr. Aidong Lu, Dr. Zbigniew W. Ras, Dr. Wensheng Wu and Dr. Zhiyi Zhang. I appreciate their time, efforts and advice on my research and dissertation. Especially, I would like to thank my advisor, Dr. Xintao Wu. He is my role model as an excellent researcher who always has clear thinking and rigorous logic. He gave many useful suggestions to help me improve in learning, thinking, writing and presenting. I appreciate his carefulness and patience very much. I also would like to thank for co-authors for their help in developing research ideas, presenting results and publishing papers, especially Dr. Xiaowei Ying, Yue Wang, Dr. Aidong Lu, Dr. Xianlin Hu from University of North Carolina at Charlotte and Dr. Zhi-Hua Zhou from Nanjing University of China.

It is a great pleasure to work in Data Privacy Lab. Thank for my labmates Dr. Kai Pan and Dr. Ling Guo who are great colleagues to work with. I would also like to thank for my friend Jing Xie who often shared me her experience and ideas in her PhD work.

Finally, I would like to thank for my parents for their love and support. This dissertation is dedicated to them.

This work was supported in part by U.S. National Science Foundation IIS-0546027, CNS-0831204 and CCF-1047621.

TABLE OF CONTENTS

LIST OF TABLES	viii
LIST OF FIGURES	ix
CHAPTER 1: INTRODUCTION	1
1.1 Spectral Analysis in Unsigned Graphs	3
1.2 Spectral Analysis in Signed Graphs	8
1.3 Anomalies and Signals	12
1.4 Privacy Perserving Graph Reconstruction	13
1.5 Paper Organization and Datasets	14
CHAPTER 2: SPECTRAL ANALYSIS OF COMMUNITY STRUCTURE IN UNSIGNED GRAPHS	20
2.1 Introduction	20
2.2 Notation	22
2.3 Spectral Perturbation	23
2.4 Spectral Analysis of Graph Topology	25
2.5 Adjacency Eigenspace based Clustering	34
2.6 Empirical Evaluation	36
2.7 Summary	41
CHAPTER 3: SPECTRAL ANALYSIS OF COMMUNITY STRUCTURE IN SIGNED GRAPHS	43
3.1 Introduction	43
3.2 Notation	45
3.3 Spectral Properties of k -Balanced Signed Graphs and Their Variants	46

	vii
3.4 Signed Graphs with Dominated Positive Inner-community Edges	60
3.5 Partite-dominated Signed Graphs	66
3.6 Unified Adjacency Eigenspace based Clustering	75
3.7 Empirical Evaluation	77
3.8 Summary	84
CHAPTER 4: WEAK ANOMALIES AND SIGNAL DETECTION	85
4.1 Introduction	85
4.2 Preliminary	87
4.3 Embedded Signal Detection	89
4.4 Empirical Evaluation	106
4.5 Summary	109
CHAPTER 5: PRIVACY PRESERVING GRAPH RECONSTRUCTION	111
5.1 Introduction	111
5.2 Preliminary	115
5.3 Low Rank Approximation on Graph Data	118
5.4 Reconstruction from Randomized Graphs	126
5.5 Empirical Evaluation	131
5.6 Reconstruction Accuracy on Low Rank Graphs	136
5.7 Summary	140
CHAPTER 6: CONCLUSIONS AND FUTURE WORK	141
REFERENCES	149

LIST OF TABLES

TABLE 2.1:	Statistics of the spectra for some networks. δ values for both the Laplacian and the normal matrices (shown in bold) and $\ \Delta\mathcal{L}\ _2$ for the Laplacian matrix and $\ \Delta\mathcal{N}\ _2$ for the normal matrix (shown in italic) violate conditions in Lemma 2.1.	38
TABLE 2.2:	Statistics of networks and partition quality of <i>AdjCluster</i> (“ k ” is the number of communities, “ <i>DBI</i> ” is the Davies-Bouldin Index, “Angle” is the average angle between centroids, and “ Q ” is the modularity.)	39
TABLE 2.3:	Accuracy (%) of clustering results (“Lap” denotes the geometric Laplacian clustering, “NCut” denotes the normalized cut, “HE” denotes the modularity based clustering, and SpokEn denotes EigenSpoke.)	40
TABLE 3.1:	Statistics of networks and partition quality of <i>UniAdjCluster</i>	81
TABLE 3.2:	Accuracy (%) of clustering results	83
TABLE 4.1:	Notations	89
TABLE 4.2:	Kurtosis and L_1 -norm of the first four eigenvectors of the observed graphs – $G(10000, 0.01)$ embedded with different signals	105
TABLE 4.3:	Eigenvalues, L_1 -norm and kurtosis of the graph of 3-community background with 3 signals	106
TABLE 5.1:	Notations	115
TABLE 5.2:	The reconstructed features for three data sets ($k = 0.4m$)	133
TABLE 5.3:	Reconstruction quality for <i>polblogs</i> network at different noise levels	134
TABLE 5.4:	Normalized Frobenius distance of reconstruction for the synthetic graphs from <i>polblogs</i> ($k = 0.4m$)	138
TABLE 5.5:	Feature reconstruction quality of the synthetic graphs from <i>polblogs</i> ($k = 0.4m$)	140

LIST OF FIGURES

FIGURE 2.1:	The political books network [Krebs, 2006]: nodes represent books about US politics sold by the Amazon.com while edges represent frequent co-purchasing of books by the same buyers on Amazon. Each node is labeled as “liberal” (blue), “neutral” (white), or “conservative” (red) by Mark Newman	22
FIGURE 2.2:	Illustration example: A graph with two communities	28
FIGURE 2.3:	Illustration examples of line orthogonality property	37
FIGURE 3.1:	<i>Syn-2Bal</i> : rotation and deviation with inter-community edges ($p = 0.05$)	50
FIGURE 3.2:	<i>Syn-2Bal</i> with different types and sizes of inter-community edges	53
FIGURE 3.3:	Spectral coordinates of unbalanced graphs perturbed from <i>Syn-2Bal</i>	55
FIGURE 3.4:	The spectral coordinates of the 3-balanced graph <i>Syn-3Bal</i> . (b)-(d): add negative inter-community edges; (e)-(f): add positive inter-community edges	59
FIGURE 3.5:	The spectral coordinates of a unbalanced synthetic graph generated via flipping signs of inner- and inter-community edges of <i>Syn-3Bal</i> with $p = 0.1$ or 1	59
FIGURE 3.6:	The Laplacian spectral space of signed graphs	60
FIGURE 3.7:	Separation of communities for signed graphs with dominated positive inner-community edges	78
FIGURE 3.8:	k -partite with different density	80
FIGURE 3.9:	Spectral space of <i>Syn-6</i> spanned by the first five eigenvectors with largest eigenvalues in magnitude	80
FIGURE 3.10:	Map of the 5 communities in <i>COW</i>	82
FIGURE 4.1:	Scatter-plot of the first four eigenvectors of $G(10000, 0.01)$	92

FIGURE 4.2: <i>ER</i> signal $G(100, 0.3)$	92
FIGURE 4.3: Scatter-plot of \mathbf{x}_1 and \mathbf{x}_2	92
FIGURE 4.4: Histogram of \mathbf{x}_1 to show the difference between the signal entries and background entries	96
FIGURE 4.5: Histogram of \mathbf{x}_2 to show the difference between the signal entries and background entries	97
FIGURE 4.6: 3-Community background with 3 signals	107
FIGURE 4.7: AstroPh network	108
FIGURE 4.8: Comparison of Kurtosis and L_1 -norm	108
FIGURE 4.9: Add three signals to AstroPh	109
FIGURE 5.1: The process of graph modification and reconstruction	114
FIGURE 5.2: Synthetic random graph with two disconnected communities	122
FIGURE 5.3: Synthetic random quasi-bipartite graph	124
FIGURE 5.4: Synthetic random graph with two clear but connected communities	125
FIGURE 5.5: Original, randomized and reconstructed features for polblog network, r varies from 1 to 200, $k = 0.4m$	131
FIGURE 5.6: $d(\hat{A}, A)$ for the three networks, as r varies ($k = 0.4m$)	135
FIGURE 5.7: The normalized Frobenius distance of the synthetic graphs	139

CHAPTER 1: INTRODUCTION

With the fast development of internet and large databases, online social networks are growing rapidly. Launched in February 2004, the famous online social network Facebook¹ has over one billion of active users. Other social networks such as Twitter² or Sina Weibo³ have hundreds of millions of registered users. The analysis on the social networks gains increasing attention in various application domains such as marketing, psychological or epidemiological researches, and homeland security. However, it is not an easy task. The complexity of network data itself already brings large challenges to researchers. Other related issues such as privacy concern make the situation more complicated. In this dissertation, we focus on the relationships in the networks and the rich information they carry. We will show that spectral analysis based on the adjacency matrix offer solutions for various graph analysis tasks such as graph partition, signal detection and privacy preserving graph publishing.

Social networks are usually modeled as graphs with nodes representing individuals and edges representing the relationship between two individuals. An unsigned graph usually contain one type of relationship. One typical example is the friendship network on Facebook. Researches have developed various methods to capture the structure and characteristics of the networks from different perspectives[Costa et al.,

¹<http://www.facebook.com/>

²<https://twitter.com>

³<http://www.weibo.com/>

2007; Newman, 2003; Strogatz, 2001]. They discovered various properties of social networks such as power-law degree distributions, small-world phenomenon and the community structure. The community structure was pointed out as one important property[Girvan and Newman, 2002]: nodes tend to join in tightly knit communities while there are only loose connections in-between communities. In other words, two nodes from the same community are more likely connected than those from different communities. Much work then focused on this property. In [Newman, 2006], the authors introduced modularity measure to quantify the strength of communities by the difference from the real network to a null model. Several graph partition algorithms were later developed based on maximizing modularity[Clauset et al., 2004; Shiga et al., 2007].

Meanwhile, the relationship between two nodes could be inherently negative to express distrust or dislike among people, e.g., the distrusted relationship on Epinion⁴. Unlike the positive relationship such as friendship to “pull” nodes together, the negative relationship “pushes” nodes away from each other. We call the graphs with both positive and negative relationships as signed graphs. Originally introduced in anthropology and sociology, signed graphs were used to model friendship and enmity. The authors in [Davis, 1967; Inohara, 2002] showed that the stability of sentiments is equivalent to k -balanced(clusterable): nodes in the same communities have positive connections and any two nodes from different communities have negative connections. General signed graphs are often unbalanced and have more complicated and unstable structures. In order to analyze signed graphs, researchers extended some of

⁴<http://www.epinions.com/>

existing measures and algorithms for unsigned graphs signed graphs. For example, the authors in [Traag and Bruggeman, 2009] extended the definition of modularity on signed graphs and developed a graph partition algorithm to maximize the signed modularity.

Instead of giant components as communities, some other papers focused on much smaller subgraphs of the networks. The subgraph can be formed by a set of most influential nodes[Kempe et al., 2003] or attackers[Backstrom et al., 2007]. Because of the small size of the subgraphs, many traditional topology-based detection methods (e.g., [Eberle and Holder, 2007; Noble and Cook, 2003]), which explore the graph topology directly, often fail to locate anomalies and signals. The detection of subtle subgraphs is difficult especially when the whole graph is very large and has multiple large communities.

1.1 Spectral Analysis in Unsigned Graphs

In this dissertation, we specially focus on one approach, spectral graph analysis, to analyze the social networks. Spectral analysis of the adjacency matrix and the variants(e.g., Laplacian matrix and the normal matrix) uses the intimate relationship between the algebraic spectral properties of the matrices and the combinatorial characteristics of the graphs to study the structure of social networks and detect hidden patterns. For an unsigned graph, the adjacency matrix A has its entry a_{ij} equals 1 if there is some relationship between nodes i and j and equals 0 otherwise. In the following, we first revisit the well studied Laplacian and the normal matrix and then turn to our focus: the adjacency matrix.

The Laplacian matrix is defined as $\mathcal{L} = D - A$ where $D = \text{diag}(d_1, \dots, d_n)$ is the

diagonal matrix with the degree of the i -th node, d_i , on the i -th diagonal entry. The normal matrix is defined as $\mathcal{N} = D^{-1}A$. Sometimes the normal matrix may also refer to its symmetric variant $D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$. The two matrices have the same eigenvalues, and their eigenvector entries are different by a factor of the node degrees of corresponding entries.

There is a large body existing literature on examining the eigenvectors of the Laplacian matrix or the normal matrix with various applications such as spectral clustering [Chan et al., 1993; Ding et al., 2001; Hagen and Kahng, 1992; Huang et al., 2008; Ng et al., 2001; Pothen et al., 1990; Shi and Malik, 2000] and graph visualization [Belkin and Niyogi, 2002]. Two most important milestones of spectral clustering are Ratio Cut [Hagen and Kahng, 1992] and Normalized Cut [Shi and Malik, 2000]. Each of the two papers introduced a graph partition algorithm that is directly associated with the eigenvalue problems of Laplacian matrix or the normal matrix. In Ratio Cut, the authors built an objective function to partition a graph with a minimal number of edges to cut. An indicator (x_{i1}, \dots, x_{ik}) is assigned to node i in community u with only $x_{iu} = 1$ and 0 otherwise. The total number of edges among communities is equal to $w = \frac{1}{2} \sum_{i=1}^k \sum_{u,v=1}^n a_{uv}(x_{ui} - x_{vi})^2$. Rewrite w into matrix form: $w = X^T(D - A)X$. By applying Lagrange multipliers, we can easily show that the eigenvectors associated to the leading k non-trivial eigenvalues (the smallest comes first) of the Laplacian matrix is the solution to the optimization problem. However, Ratio Cut often gives a skew partition. Instead of cutting between the major components of the graph, it usually finds a much smaller subgraph to separate from the major components. Normalized Cut later was introduced [Shi and Malik, 2000] to reweigh the edges to

cut between two communities by the sum of node degrees of each community. The objective function can be transformed to the eigenvalue problem of the normal matrix similarly. Notice the graph partition problem is NP-hard. The eigenvalue problems of the matrices only offer heuristic solutions of the graph partition problem: the entries of the indicator vector are relaxed to real numbers between 0 and 1. Spectral clustering by the normal matrix gives good experimental results, but little work discussed clusterability of communities in the normal matrix space except [Ng et al., 2001]. Starting from the block-wise normalized matrix representing a graph with k disconnected components, Ng and *et al.* derived the bound of the leading eigenvalues and eigenvectors. However, for the normal matrix, it is challenging to link the perturbation in the normal spectral space directly to the changes of graph's topology, due to the fact that the normal matrix has each link normalized by the nodes' degrees.

Different from the Laplacian matrix or the normal matrix, the properties of the adjacency eigenspace have received much less attention. However, many spectral properties of the adjacency matrix are closely related with the characteristics of the graph [Costa et al., 2007]. For example, the eigenvalues of the adjacency matrix encode information about the cycles of a network as well as its diameter. The maximum degree, chromatic number, clique number, and extend of branching in a connected graph are all related to λ_1 , the largest eigenvalue of the adjacency matrix. In [Wang et al., 2003], it was shown that the epidemic threshold for a network under virus propagation is closely related to λ_1 . In [Ying et al., 2011; Ying and Wu, 2009c], the authors utilized the spectra of the adjacency matrix to characterize and capture the graph randomness. They gave a framework which can quantify graph non-randomness at

edge, node, and the overall graph levels. They used the spectral coordinates of nodes in the first k -dimensional spectral space, where k corresponds the number of communities, to derive the non-randomness measure and showed quantitative comparisons between various social networks with different sizes and densities or between different snapshots of a dynamic social network. They also showed that neither Laplacian nor the normal matrices can give a concise expression of a similar measure at the node level. In their work, the authors also showed their observations of line orthogonality pattern in the adjacency spectral space, and they suggested that such a pattern is associated with the presence of a clear community structure in the graph. However, they did not give theoretical explanation on when and why the observed pattern exists and how the pattern in the spectral space is connected with the structure in real graphs.

In this dissertation, we conduct a thorough theoretical study on adjacency spectral analysis. We offer rigorous proofs of the intimate relation between the topological structures in a social network and the spectral properties of its adjacency matrix. We analyze a broad range of social networks including friendship networks, alliance and war networks, and distrusted networks. The matrix perturbation theory plays a crucial role in the study of spectral analysis. The theory explains how the perturbation on a matrix affects its spectral system such as eigenvalues and eigenvectors. General matrix perturbation theory is complicated and conditions are abstract. We derive a simplified version in the context of social network analysis. We consider the real graph as a perturbation result from a graph in an ideal case that has a more clear structure and is easier to analyze. Based on the matrix perturbation theory, we then

approximate the eigenvectors of the real graph by those of a graph in the ideal case.

Focusing on unsigned graphs, we describe the general community structure as a perturbation from a graph with k disconnected communities. The adjacency matrix of the observed graph, \tilde{A} , is then divided into two parts: $\tilde{A} = A + E$, where A is the adjacency matrix of the graph with k disconnected communities (i.e., A is diagonal k -block matrix) and E is the perturbation matrix denoting the edges among communities. The magnitude of E is usually smaller than that of A because the inter-community connection is relatively sparse.

Based on the matrix perturbation theory, we approximate the eigen-pairs (eigenvalues and eigenvectors) of \tilde{A} by the eigen-pairs of the hidden matrix A and perturbation matrix E . Hence, we can assess the effect of the perturbation (E) on the coordinate of each node in the adjacency spectral space. We follow the framework in [Ying et al., 2011] and use the first k -dimensional spectral space to analyze the graph. We call the first k leading eigenvectors as principal eigenvectors. These eigenvectors have the largest eigenvalues in magnitude. In the spectral space of the hidden k -block graph (A), nodes of k communities stay on k canonical axes spanned by principal eigenvectors. When E is added, the approximated eigenvectors of \tilde{A} have clear relationship with A and E : the k lines rotate from the axes and those nodes connected to other communities are pulled away from their lines by the inter-community edges. As a result, we are able to demonstrate the line orthogonality pattern in the spectral space of \tilde{A} observed in [Ying et al., 2011]. We also derive the conditions on which the pattern exists, and give explicit formula to quantify the rotation and deviation of the nodes in the adjacency spectral space caused by the perturbation matrix E .

This methodology is significantly different from the spectral analysis of the Laplacian or the normal matrix, in which the cluster patterns are usually demonstrated via some graph-cut optimization problem. We also examine the spectral spaces of the Laplacian matrix and the normal matrix through the perturbation framework. We find that the line orthogonality pattern in general does not hold in the Laplacian eigenspace or the normal eigenspace. We further provide theoretical explanations.

1.2 Spectral Analysis in Signed Graphs

Most social network analysis approaches focused on unsigned graphs, but the relationship could be inherently negative to express distrust or dislike. In contrast to the extensive studies on social networks that restrict to only positive relationship between individuals, we extend the study to signed networks with both positive and negative relationships. We similarly apply the matrix perturbation theory to investigate the impacts of introducing negative edges and explore the patterns in the spectral space of the graph’s adjacency matrix.

For a signed graph, $a_{ij} = -1$ in the adjacency matrix A if there exists a negative relationship between the nodes i and j . In [Kunegis et al., 2010], the authors showed spectral analysis of signed graphs by the extended definition of the Laplacian matrix[Hou et al., 2003]: $\mathcal{L} = |D| - A$ where $|D|$ is $\text{diag}(|d_1|, \dots, |d_n|)$ and $|d_i|$ is the total number of positive or negative edges from node i . The authors showed the different drawings of graph based on eigenvectors from A , $D - A$, and $|D| - A$. They then extended Ratio Cut and Normalized Cut to signed graphs. They also showed link sign prediction via different graph kernels. However, they did not relate the structures in signed graphs with patterns in the spectral space directly.

In a signed graph, positive and negative edges are two forces that behave differently. Positive edges group people together and usually exist inside a community while negative edges separate people into different communities and usually exist outside communities. These two forces sometimes work together harmonically. The k -balanced graph is the special case to describe the situation. However, in other cases, these two forces may not work together. The graph may be dominated by positive inner-community edges or negative inter-community edges. We discuss them respectively to show the clusterable patterns in signed graphs.

- k -Balanced Signed Graphs: In this type of signed graphs, the community structure is determined by both positive inner-community edges and negative inter-community edges. The idea k -balanced signed graphs has all inner-community edges positive and all inter-community edges negative. We treat it as a perturbed variant of a k -block graph. The k -block graph has been discussed in the unsigned graphs as the graph with k well separated communities. We then show that communities in k -balanced signed graphs are distinguishable in the spectral space of its signed adjacency matrix, even when negative connections between communities are dense. In the real world networks, a small number of negative inner-community edges or positive inter-community edges may exist in the graph and violate the balance of the graph. We apply the matrix perturbation theory to examine the unbalanced signed graph perturbed from the k -balanced signed graphs.
- Signed Graphs with Dominated Positive Inner-community Edges: In this type of

signed graphs, the community structure is mainly determined by positive inner-community edges and there can be both positive and negative edges between communities. It is straightforward to see the communities are separable in the spectral space if inter-community edges are sparse regardless of signs. We are particularly interested in the case when the inter-community connection is dense. Intuitively, the negative inter-community edges would offset the effect of the positive inter-community edges. Starting from the k -block graph as the ideal case, we theoretically demonstrate that, when the graph contains dense positive and negative inter-community edges of equal magnitude, the communities are still distinguishable in the adjacency spectral space.

- **Partite-dominated Signed Graphs:** In this type of signed graphs, the community structure is mainly determined by negative inter-community edges. The ideal case is called a k -partite graph where there is no inner-community connection and only dense negative inter-community connection. In real networks, the relationship such as distrust or dispute mainly appears between communities and rarely exist inside communities. If we use -1 to denote the negative relationship, we find that the largest eigenvalue in magnitude of the adjacency matrix is negative. After removing the effect of the first eigenvalue and eigenvector, we get a matrix similar to that of signed graphs with dominated positive inner-community edges. We follow similar procedures to approximate its eigenvectors. With properly re-orthogonalizing the eigenvectors, we give the approximated eigenvectors of k -partite graphs and we theoretically demonstrate that

such graphs have k orthogonal clusters in the adjacency spectral space.

These three types of graphs are the clusterable signed graphs. General clusterable signed graphs are close to one of the three or their combinations. We are the first to study the clusterability of the community structure in the adjacency space of signed networks. We conduct theoretical explanation of the patterns for different types of clusterable signed graphs and verify our findings with illustrative examples.

Based on our findings in both unsigned and signed graphs, we develop an efficient and effective graph partition algorithm *UniAdjCluster*. The idea of the algorithm is to project node coordinates in the adjacency spectral space to the unit sphere in the spectral space and then apply the k -means algorithm to find the clusters. The empirical evaluations on synthetic graphs and real-world social networks demonstrate the effectiveness of our graph partition algorithm. Calculation of the eigenvectors of an $n \times n$ matrix takes in general a number of operations $O(n^3)$, which is almost inapplicable for large networks. However, in *UniAdjCluster*, we only need to calculate the first k eigen-pairs where $k \ll n$. Furthermore, adjacency matrices in our context are usually sparse. The Arnoldi/Lanczos algorithm[Golub and Van Loan, 1996] generally needs $O(n)$ rather than $O(n^2)$ floating point operations at each iteration. The algorithm also outperforms those clustering methods based on the Laplacian or the normal spectrum. The main reason is that the leading adjacency eigenpairs are generally more robust under moderate noises due to the large magnitude of the eigenvalues. Unlike adjacency eigenvalues, the leading eigenvalues of the Laplacian and normal matrices are typically less than 1 in magnitude, which results in instability

under noise.

1.3 Anomalies and Signals

In some other situations, it is of great interest for data analysts or data owners to detect small abnormal subgraphs compared to the whole graph. A small and subtle subgraph often indicates a subset of users that are extremely active or the attackers that behave differently from the normal users. We refer the embedded small and subtle subgraphs as *signals or anomalies* and the large graph as *background*. Those small and subtle signals, which are structurally dissimilar to the background, are often hidden within graph communities and can not be revealed in the global structure of the graph. Traditional topology-based detection methods (e.g., [Eberle and Holder, 2007; Noble and Cook, 2003]) often fail to locate those subtle signals. A question is then raised here: whether adjacency spectral analysis can detect the subtle signals.

The adjacency matrix A of a real graph that is composited of two parts: the background graph B and the embedded signal S . In the adjacency spectral space, embedded signals exhibit the pattern significantly different from the community structure. Because of their small magnitude, embedded signals usually affect the leading eigenvectors too little to detect. We then focus on exploring the eigenvectors with smaller eigenvalues to detect subtle anomalies. We call these eigenvectors with smaller eigenvalues as minor eigenvectors.

We first demonstrate that the anomaly nodes have extreme values in some minor eigenvectors compared to background nodes. When the background graph is an *Erdos-Renyi* graph and there is only one signal, we give the approximated first two eigenvectors. We then discuss the conditions when the difference between signal

entries and background entries is large enough to detect the signal on these two eigenvectors. We then extend our theoretical studies to the general case where multiple anomalies are embedded in a general background graph. These eigenvectors appear after the k -th eigenvectors. However, they are not necessarily the eigenvectors right after the k -th. We develop an algorithm that leverages the kurtosis (rather than the L_1 -norm of eigenvector in [Miller et al., 2010]) to find those eigenvectors that best reflect the signals. The kurtosis metric describes the extremeness in the distribution of eigenvector entries caused by embedded signals. Our results do not rely on the assumption that the background graph is generated by some parametric model as in [Miller et al., 2010]. Our theoretical analysis and empirical evaluations on both synthetic data and real social networks show effectiveness of our approach to detecting subtle signals.

1.4 Privacy Preserving Graph Reconstruction

Different from biological networks, social networks have the privacy issue. Many social networks contain sensitive information of their users. In [Backstrom et al., 2007; Hay et al., 2007], the authors discussed the challenges in privacy protection of social networks. They pointed out that even the naive node-anonymized network does not guarantee privacy. Adversaries may re-identify a targeted individual from the anonymized graph by exploiting some known structural information of his neighborhood. Various randomization techniques were later developed such as edge modification [Liu and Terzi, 2008; Zhou and Pei, 2008; Zou et al., 2009], edge randomization [Hay et al., 2007; Ying and Wu, 2008, 2009a,b], and clustering-based generalization [Bhagat et al., 2009; Campan and Truta, 2008; Cormode et al., 2008; Hay

et al., 2008; Zheleva and Getoor, 2007]. These above anonymization approaches have been shown as a necessity, in addition to naive anonymization, to preserve privacy in publishing social network data. However, in order to protect the privacy of the individual entry under feature based attacks or structural attacks, a medium or large perturbation is needed [Hay et al., 2007; Ying and Wu, 2008] and hence the utility of the released randomized graph (in terms of topological features) is significantly lost in the randomized graph. We focus on whether we can reconstruct a graph from the edge randomized graph such that accurate feature values can be recovered. In our study, we find that the adjacency spectral space is relatively stable under the perturbation and principal eigenvectors capture the information of major components. We apply low rank approximation of the adjacency matrix to reconstruct the graph that is randomly added and deleted the same number of edges. We then give a novel solution to determine the (approximate) optimal rank, a key parameter in our reconstruction algorithm. We explicitly assess effects of perturbation on the accuracy of the reconstructed feature values. What is more, one surprising finding is that, for most social networks, the reconstructed networks do not incur further disclosure risks of individual privacy than the released randomized graphs. It has much difference from the numerical data setting. Our further investigation shows that only networks with low ranks or a small number of dominant eigenvalues may incur further privacy disclosure due to reconstruction.

1.5 Paper Organization and Datasets

The dissertation is organized as follows:

In Chapter 2, we first present the general matrix perturbation theory and derive a

simplified version in our context. We then apply the theory and show that communities in unsigned graphs exhibit as orthogonal lines in the adjacency spectral space. We demonstrate the conditions when this pattern holds and why it generally disappears in the Laplacian and the normal eigenspaces. We develop a graph partition algorithm, *AdjCluster*, to utilize our theoretical findings to analyze unsigned graphs.

In Chapter 3, we extend our study to signed graphs. We first study the k -balanced graphs and their variants. We apply the matrix perturbation theory and show that these signed graphs are clusterable in the adjacency spectral space. We then study signed graphs with dominated positive inner-community edges and we are especially interested in the case when the inter-community edges are dense. We show that such graphs with equal magnitude of positive and negative edges still keep similar structures decided by the inner-community edges. Finally, for partite-dominated signed graphs, we transform them to the form of signed graphs with dominated positive inner-community edges and conduct a similar analysis procedure to show their clusterability in adjacency spectral space. We run empirical evaluations to verify our theoretical findings and extend *AdjCluster* to *UniAdjCluster* to partition communities of signed graphs.

In Chapter 4, we explore how minor eigenvectors capture the subtle signals and develop an algorithm to detect them. We start with the *Erdos-Renyi* random graphs with one signal and give approximated forms of the principal eigenvectors and the minor eigenvectors respectively. We derive formulas to show the difference of signal entries from the background entries. We then extend the study to general cases with multiple signals. Our algorithm does not require the assumption of background graph

as in [Miller et al., 2010]. We evaluate both synthetic data and real social networks to show the effectiveness of the algorithm.

In Chapter 5, we discuss the privacy issue in publishing social networks and review the state-of-the-art anonymization methods on network data. We explore spectral properties of the graph data and show why noise could be separated from the perturbed graph using low rank approximation. We also show the key difference between numerical data and network data reconstruction through empirical evaluations and theoretical justifications.

Finally, in Chapter 6, we summarize our work and contributions. The adjacency matrix has many good properties in spectral analysis. It has long been underestimated due to the lack of study of the fundamental properties. We show the theoretical proofs based the simplified perturbation theory to support the applications of adjacency spectral analysis. In future work, we discuss further exploration of adjacency spectral properties in more social networks. We will compare our algorithms with other existing algorithms and analyze networks with very large scale. We believe that there is also promising applications in privacy preserving data mining.

In this dissertation, we use various synthetic data and real network datasets. Synthetic data will be discussed in each chapter. The following are the details of the real network datasets.

- Polbooks: US political books dataset⁵ contains frequent co-purchasing records of US politics books sold by the online bookseller Amazon.com. It has 105 nodes and 441 edges. Each node represents a books published during 2004 presiden-

⁵<http://www-personal.umich.edu/~mejn/netdata/>

tial election and each edge indicates that two books are frequently purchased together. Nodes are separated into groups with labels: liberal, neutral, or conservative by Mark Newman based on a reading of the descriptions and reviews of the books posted on Amazon.

- Polblogs: Political Blogs dataset⁶, collected by by Adamic and Glance[Adamic and Glance, 2005], contains incoming and outgoing links and posts during the time of the 2004 presidential election among US political blogs. It contains 1222 nodes and 16714 edges. Each Node represents a political blog and each edge indicates that two blogs have links or blog posts between them. The nodes are labeled as either liberal or conservative.
- Enron: Enron email network was built from email corpus of a real organization over the course covering a 3 years period. We used a pre-processed version of the dataset provided by [Shetty and Adibi, 2004]. It contains 252,759 emails from 151 Enron employees, mainly senior managers. In the graph *Enron* we focus on emails sent *from and to* these 151 people and the semantics of an edge (u, v) in such a graph is that there have been at least five email communications between u and v , which results in 869 edges.
- Facebook: We use the dataset provided by Bimal Viswanath *et al.*[Viswanath et al., 2009], which contains a subset of all of the user-to-user links from the Facebook New Orleans networks. They crawled the dataset by the breadth-first-search algorithm: they started from a single user, visited all the friends

⁶<http://www-personal.umich.edu/~mejn/netdata/>

who have their profiles visible and use these friends as new start points to repeat the same process. It contains 63731 nodes and 817090 edges. Each node represents a Facebook user and each edge indicates that two users are connected on Facebook as friends.

- **Correlates of War:** The Correlates of War[Ghosn et al., 2004] dataset over the period 1993-2001 contains international relationships such as alliance and dispute among different countries and areas. The dataset Formal Alliances (v3.03) records formal alliance among different countries. There are three types of alliance: defense pact (Type I), neutrality and non-aggression pact (Type II), and ententes (Type III). The dataset Militarized Interstates Disputes (v3.1) records all instances that one state threatened, displayed, or used force against another, e.g., border extension between Colombia and Venezuela and Turkish groups entering Iraqi territory. There are five levels of dispute: no militarized action (Level 1), threat to use force (Level 2), display of force (Level 3), use of force (Level 4), and war (Level 5). For those disputes that involve different levels of actions, we use the highest level to represent the level of dispute. We construct a signed graph where military alliances are represented by positive edges and disputes by negative edges. We use the alliance of defense pact (Type I) to construct the cliques of positive edges and the disputes of use of force (Level 4) and war (Level 5) to construct the bipartite graph of negative edges. When a positive edge conflicts with a negative edge, we treat the negative edge with higher priority. This is because the use of force breaks the alliance.

The adjacency matrix of the constructed signed graph contains 159 nodes with 1093 positive edges and 155 negative edges.

- **Epinions:** Epinions is a consumers opinion site where the users have their block lists of some other users whose reviews are usually inaccurate or not valuable[Massa and Avesani, 2004]. We specially extract the two way distrust relationship, i.e., user i has user j on his/her block list while user j also has user i on the block list. We then get a graph with 2027 nodes representing the users and 2383 edges representing the distrust relationship.
- **AstroPh:** Astro Physics Collaboration Network dataset⁷ is from the e-print arXiv and covers scientific collaborations between authors who submitted papers to Astro Physics category. It contains 18772 nodes and 396160 edges. Each node represents an author and each edge indicates that two authors have coauthored for at least one paper.

⁷<http://snap.stanford.edu>

CHAPTER 2: SPECTRAL ANALYSIS OF COMMUNITY STRUCTURE IN UNSIGNED GRAPHS

In this chapter, we first focus on the graphs with only one type of relation. We first simplify the general matrix perturbation theory in the context of social network analysis. We conduct theoretical studies based on it to demonstrate: why this line orthogonality pattern holds in the adjacency eigenspace and disappears in the Laplacian and the normal eigenspaces. We then utilize the orthogonal line pattern to develop a graph partition algorithm. Empirical evaluations on synthetic data and real-world social networks validate our theoretical findings and show the effectiveness of our graph partition algorithm.

2.1 Introduction

Social networks have received much attention these days. To understand and utilize the information in a social network, researches have developed various methods to capture the structure and characteristics of the network from different perspectives. Among them, spectral analysis of the adjacency matrix and its variants (e.g., Laplacian matrix and normal matrix) has shown intimate relationship between the combinatorial characteristics of a graph and the algebraic spectral properties of its matrix [Seary and Richards, 2003]. Different from the Laplacian matrix or normal matrix, the properties of the adjacency eigenspace received much less attention except some recent work [Prakash et al., 2010; Ying and Wu, 2009c]. It was shown by

Prakash et al. [Prakash et al., 2010] that the singular vectors of mobile call graphs exhibit an EigenSpokes pattern wherein, when plotted against each other, they have clear, separate lines that neatly align along specific axes. The authors suggested that EigenSpokes are associated with the presence of a large number of tightly-knit communities embedded in very sparse graphs. Ying and Wu [Ying and Wu, 2009c] showed that node coordinates in the adjacency eigenspace of a graph with well structured communities form quasi-orthogonal lines (not necessarily axes aligned) and developed a framework to quantify importance (or non-randomness) of a node or a link by exploiting the line orthogonality property. However, no theoretical analysis was presented [Prakash et al., 2010; Ying and Wu, 2009c] to demonstrate why and when this line orthogonality property holds.

Before we go to the theoretical study, we use a real network to illustrate the line orthogonality pattern.

Figure 2.1(a) plots the *polbooks* dataset [Krebs, 2006] with 105 nodes, 441 edges, and two clear communities. Figure 2.1(b) plots node coordinates projected in the 2-D spectral space of the adjacency matrix. We can observe from Figure 2.1(b) that the majority of vertices projected in the 2-D spectral space distribute along two straight and quasi-orthogonal lines. It matches the topological structure that there exist two communities with sparse edges connecting them. The first up-trend line consists of most nodes in red color while the second down-trend line consists of most nodes in blue color. White nodes, which correspond to either noise nodes or bridging nodes, distribute either around the origin or between two quasi-orthogonal lines in the projected space. We removed all edges across two communities and plotted node

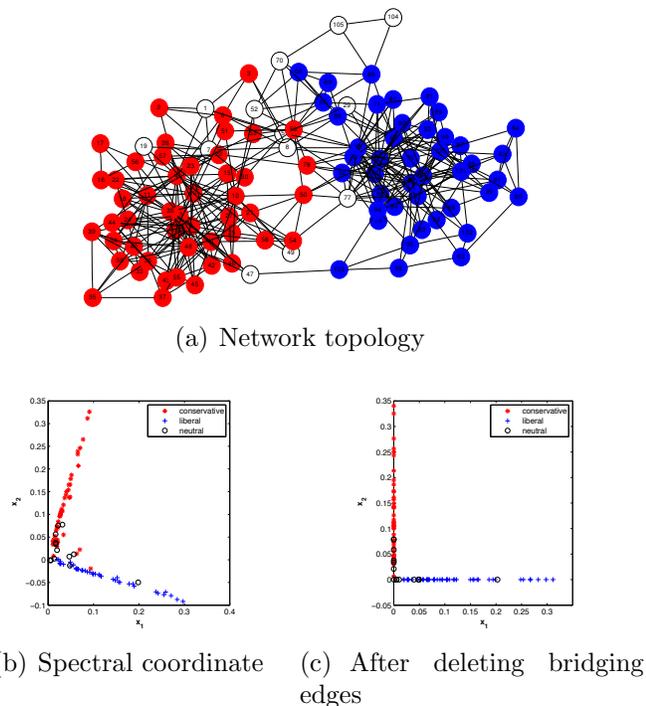


Figure 2.1: The political books network [Krebs, 2006]: nodes represent books about US politics sold by the Amazon.com while edges represent frequent co-purchasing of books by the same buyers on Amazon. Each node is labeled as “liberal” (blue), “neutral” (white), or “conservative” (red) by Mark Newman

coordinates in Figure 2.1(c). We can see clearly that there exist two axes-aligned lines where each line contains nodes from the same community. Generally speaking, if there is k communities in the graph, we can observe k quasi-orthogonal lines. In the following, we conduct theoretical study of the relation between these two patterns.

2.2 Notation

The network considered in this chapter is binary, symmetric, connected, and without self-loops. It can be represented as the symmetric adjacency matrix $A_{n \times n}$ with $a_{ij} = 1$ if node i is connected to node j and $a_{ij} = 0$ otherwise. Let λ_i be the i -th largest eigenvalue of A and \mathbf{x}_i the corresponding eigenvector. x_{ij} denotes the j -th entry of \mathbf{x}_i . Formula 2.1 illustrates our notation. The eigenvector \mathbf{x}_i is represented as a column vector. The leading eigenvectors \mathbf{x}_i ($i = 1, \dots, k$) corresponding to the

largest k eigenvalues contain most topological information of the original graph in the spectral space. The k -dimensional spectral space is spanned by $(\mathbf{x}_1, \dots, \mathbf{x}_k)$. When we project node u in the k -dimensional subspace with \mathbf{x}_i as the basis, the row vector $\boldsymbol{\alpha}_u = (x_{1u}, x_{2u}, \dots, x_{ku})$ is its coordinate of in this subspace. We call $\boldsymbol{\alpha}_u$ the spectral coordinate of node u . The eigenvector \mathbf{x}_i becomes the canonical basis denoted by $\xi_i = (0, \dots, 0, 1, 0, \dots, 0)$, where the i -th entry of ξ_i is 1.

$$\begin{array}{cccc}
 & \mathbf{x}_1 & \mathbf{x}_i & \mathbf{x}_k & \mathbf{x}_n \\
 & & \downarrow & & \\
 \boldsymbol{\alpha}_u \rightarrow & \left(\begin{array}{ccc|ccc}
 x_{11} & \cdots & x_{i1} & \cdots & x_{k1} & \cdots & x_{n1} \\
 \vdots & & \vdots & & \vdots & & \vdots \\
 \hline
 x_{1u} & \cdots & x_{iu} & \cdots & x_{ku} & \cdots & x_{nu} \\
 \hline
 \vdots & & \vdots & & \vdots & & \vdots \\
 x_{1n} & \cdots & x_{in} & \cdots & x_{kn} & \cdots & x_{nn}
 \end{array} \right) & & (2.1)
 \end{array}$$

2.3 Spectral Perturbation

Spectral perturbation analysis deals with the change of the graph spectra (eigenvalues and eigenvector components) when the graph is perturbed. For a symmetric $n \times n$ matrix A with a symmetric perturbation E , the matrix after perturbation can be written as $\tilde{A} = A + E$. Let λ_i be the i -th largest eigenvalue of A with its eigenvector \mathbf{x}_i . Similarly, $\tilde{\lambda}_i$ and $\tilde{\mathbf{x}}_i$ denote the eigenvalue and eigenvector of \tilde{A} . It has been shown that the perturbed eigenvector $\tilde{\mathbf{x}}_i$ can be approximated by a linear function involving all original eigenvectors (refer to Theorem V.2.8 in [Stewart and Sun, 1990]). We quote it as below.

Lemma 2.1: Let $U = (\mathbf{x}_1, \dots, \mathbf{x}_{i-1}, \mathbf{x}_{i+1}, \dots, \mathbf{x}_n)$, $S = \text{diag}(\lambda_1, \dots, \lambda_{i-1}, \lambda_{i+1}, \dots, \lambda_n)$, and $\beta_{ij} = \mathbf{x}_i^T E \mathbf{x}_j$. The eigenvector $\tilde{\mathbf{x}}_i$ ($i = 1, \dots, k$) can be approximated as:

$$\tilde{\mathbf{x}}_i \approx \mathbf{x}_i + U(\lambda_i I - S)^{-1} U^T E \mathbf{x}_i \quad (2.2)$$

when the following conditions hold:

1. $\delta = |\lambda_i - \lambda_{i+1}| - \|\mathbf{x}_i^T E \mathbf{x}_i\|_2 - \|U^T E U\|_2 > 0$;
2. $\gamma = \|U^T E \mathbf{x}_i\|_2 < \frac{1}{2}\delta$.

We simplify its approximation by only using the leading k eigenvectors when the first k eigenvalues are significantly greater than the rest ones. Based on the simplified approximation shown in Theorem 2.1, we are able to prove the line orthogonality pattern in the adjacency eigenspace.

Theorem 2.1: Assume that the conditions in Lemma 2.1 hold. Further assume that $|\lambda_i| \gg |\lambda_j|$, for any $i = 1, \dots, k$ and $j = k + 1, \dots, n$. Then, the eigenvector $\tilde{\mathbf{x}}_i$ ($i = 1, \dots, k$) can be approximated as:

$$\tilde{\mathbf{x}}_i \approx \mathbf{x}_i + \sum_{j=1; j \neq i}^k \frac{\beta_{ji}}{\lambda_i - \lambda_j} \mathbf{x}_j + \frac{1}{\lambda_i} E \mathbf{x}_i. \quad (2.3)$$

Proof. With Lemma 2.1, we have

$$\begin{aligned} \tilde{\mathbf{x}}_i &\approx \mathbf{x}_i + U(\lambda_i I - S)^{-1} U^T E \mathbf{x}_i \\ &= \mathbf{x}_i + \sum_{j=1; j \neq i}^n \frac{\beta_{ji}}{\lambda_i - \lambda_j} \mathbf{x}_j \\ &= \mathbf{x}_i + \sum_{j=1; j \neq i}^k \frac{\beta_{ji}}{\lambda_i - \lambda_j} \mathbf{x}_j + \sum_{j=k+1}^n \frac{\beta_{ji}}{\lambda_i - \lambda_j} \mathbf{x}_j. \end{aligned}$$

Since $|\lambda_i| \gg |\lambda_j|$ for all $i = 1, \dots, k$ and $j = k + 1, \dots, n$, $\frac{\beta_{ji}}{\lambda_i - \lambda_j} \approx \frac{\beta_{ji}}{\lambda_i}$, and we further have

$$\tilde{\mathbf{x}}_i \approx \mathbf{x}_i + \sum_{j=1, j \neq i}^k \frac{\beta_{ji}}{\lambda_i - \lambda_j} \mathbf{x}_j + \sum_{j=k+1}^n \frac{\beta_{ji}}{\lambda_i} \mathbf{x}_j. \quad (2.4)$$

Note that

$$\begin{aligned} \sum_{j=k+1}^n \frac{\beta_{ji}}{\lambda_i} \mathbf{x}_j &\approx \sum_{j=1}^n \frac{\beta_{ji}}{\lambda_i} \mathbf{x}_j = \sum_{j=1}^n \frac{\mathbf{x}_j^T E \mathbf{x}_i}{\lambda_i} \mathbf{x}_j \\ &= \frac{1}{\lambda_i} \sum_{j=1}^n \langle E \mathbf{x}_i, \mathbf{x}_j \rangle \mathbf{x}_j = \frac{1}{\lambda_i} E \mathbf{x}_i. \end{aligned} \quad (2.5)$$

The last equality of Equation 2.5 is because \mathbf{x}_j ($j = 1, \dots, n$) forms an orthogonal basis of the n -dimensional space, and $\mathbf{x}_j^T E \mathbf{x}_i$ is just the projection of vector $E \mathbf{x}_i$ onto one of the basis vector \mathbf{x}_j . Combining Equation 2.4 and Equation 2.5, we get Equation 2.3. \square

2.4 Spectral Analysis of Graph Topology

Social networks usually exhibit community structures. Communities are loosely defined as collections of individuals who interact unusually frequently. For a graph contains k communities, its adjacency matrix \tilde{A} can be divided into two parts: the adjacency matrix A of a graph with k disconnected communities and the perturbation E denoting the edges between communities. The magnitude of E is usually smaller than that of A .

2.4.1 Graphs with k Disconnected Communities

For a graph with k disconnected communities C_1, \dots, C_k of size n_1, \dots, n_k respectively ($\sum_i n_i = n$), its adjacency matrix A can be written as a block-wise diagonal

matrix:

$$A = \begin{pmatrix} A_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & A_k \end{pmatrix}, \quad (2.6)$$

where A_i is the $n_i \times n_i$ adjacency matrix of C_i . Let λ_{C_i} be the largest eigenvalue of A_i in magnitude with eigenvector $\mathbf{x}_{C_i} \in \mathbb{R}^{n_i}$. Without loss of generality, we assume $\lambda_{C_1} > \dots > \lambda_{C_k}$. Since the entries of A_i are all non-negative, with Perron-Frobenius theorem [Stewart and Sun, 1990], λ_{C_i} is positive and all the entries \mathbf{x}_{C_i} are non-negative. When C_i contains one dominant component or does not have a clear inner-community structure, the magnitude of λ_{C_i} is significantly larger than the rest eigenvalues of A_i [Chung et al., 2003]. Hence when the k disconnected communities are comparable, $\lambda_i = \lambda_{C_i}$, $i = 1, \dots, k$ (the eigenvalues and eigenvectors of A_i are naturally the eigenvalues of A). Here we call two communities C_i and C_j are comparable if both of the second largest eigenvalues of A_i and A_j are smaller than λ_{C_i} and λ_{C_j} . Two communities are not comparable when one of them contains either too few edges or nodes and hence does not contribute much to the graph topology.

Lemma 2.2: For a graph with k disconnected comparable communities as shown in Equation 2.6, for all $i = 1, \dots, k$ and $j = k + 1, \dots, n$, $\lambda_i \gg |\lambda_j|$. The first k

eigenvectors of A have the following form:

$$(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k) = \begin{pmatrix} \mathbf{x}_{C_1} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{x}_{C_2} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{x}_{C_k} \end{pmatrix},$$

and all the entries of \mathbf{x}_i are non-negative.

When we project each node in the subspace spanned by $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k$, we have the following result.

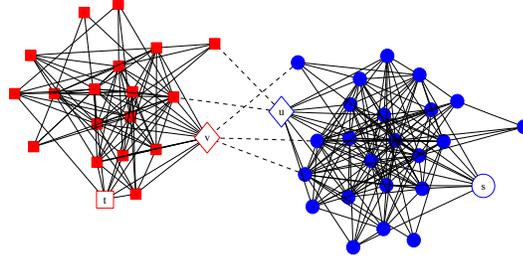
Proposition 2.1: For a graph with k disconnected comparable communities as shown in Equation 2.6, spectral coordinates of all nodes locate on the k axes ξ_1, \dots, ξ_k where $\xi_i = (0, \dots, 0, 1, 0, \dots, 0)$ is the canonical basis and the i -th entry of ξ_i is 1. Specifically, for any node $u \in C_i$, its spectral coordinate has the form

$$\boldsymbol{\alpha}_u = (0, \dots, 0, x_{iu}, 0, \dots, 0). \quad (2.7)$$

The position of non-zero x_{iu} in Equation 2.7 indicates the community that node u belongs to; and the value of x_{iu} indicates the weight or importance of node u within the community C_i and hence captures the magnitude of belongings.

Two Dimensional Case

For a graph with two disconnected communities C_1 and C_2 of size n_1 and n_2 respectively ($n = n_1 + n_2$), its adjacency matrix A and two leading eigenvectors can be



(a) Topology snapshot

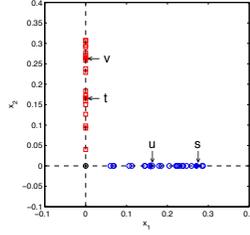
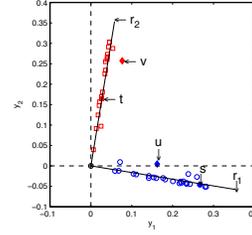
(b) Disconnected graph A (c) Perturbed graph \tilde{A}

Figure 2.2: Illustration example: A graph with two communities

written as follows:

$$A = \begin{pmatrix} A_1 & \mathbf{0} \\ \mathbf{0} & A_2 \end{pmatrix}, (\mathbf{x}_1, \mathbf{x}_2) = \begin{pmatrix} x_{11} & 0 \\ \vdots & \vdots \\ x_{1n_1} & 0 \\ 0 & x_{2,n_1+1} \\ \vdots & \vdots \\ 0 & x_{2n} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_{C_1} & \mathbf{0} \\ \mathbf{0} & \mathbf{x}_{C_2} \end{pmatrix}$$

All the nodes from C_1 lie on the line that passes through the origin and the point $(1, 0)$ and nodes from C_2 lie on the line that passes through the origin and the point $(0, 1)$.

We show a synthetic graph with two disconnected communities in Figure 2.2(a).

The solid lines are links within each community. Figure 2.2(b) shows the spectral

coordinates in the 2-D scatter plot when the two communities are disconnected. Blue

circles represent the 25 nodes from one community and red square nodes represent

the 20 nodes from the other community. We can see that all nodes lie along the two axes.

2.4.2 Spectral Properties of Observed Graphs

The observed graph \tilde{A} can be written as $\tilde{A} = A + E$, where A is as shown in Equation 2.6 and E denotes the edges across communities. Based on Theorem 2.1, we derive the approximation of the perturbed spectral coordinate α_u , which is determined by the original spectral coordinate of itself and that of its neighbors in other communities.

Theorem 2.2: Denote an observed graph as $\tilde{A} = A + E$ where A is as shown in Equation 2.6 and E denotes the edges across communities. For a node $u \in C_i$, let Γ_u^j denote its neighbors in C_j for $j \neq i$, and $\Gamma_u^i = \emptyset$. The spectral coordinate of u can be approximated as

$$\alpha_u \approx x_{iu} \mathbf{r}_i + \left(\sum_{v \in \Gamma_u^1} \frac{e_{uv} x_{1v}}{\lambda_1}, \dots, \sum_{v \in \Gamma_u^k} \frac{e_{uv} x_{kv}}{\lambda_k} \right) \quad (2.8)$$

where scalar x_{iu} is the only non-zero entry in its original spectral coordinate shown in Equation 2.7, e_{uv} is the (u, v) entry of E , and \mathbf{r}_i is the i -th row of the following matrix

$$R = \begin{pmatrix} 1 & \frac{\beta_{12}}{\lambda_2 - \lambda_1} & \cdots & \frac{\beta_{1k}}{\lambda_k - \lambda_1} \\ \frac{\beta_{21}}{\lambda_1 - \lambda_2} & 1 & \cdots & \frac{\beta_{2k}}{\lambda_k - \lambda_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\beta_{k1}}{\lambda_1 - \lambda_k} & \frac{\beta_{k2}}{\lambda_2 - \lambda_k} & \cdots & 1 \end{pmatrix}. \quad (2.9)$$

Proof. With Theorem 2.1, the leading k eigenvectors of \tilde{A} can be approximated as

$$\tilde{\mathbf{x}}_i \approx \mathbf{x}_i + \sum_{j=1; j \neq i}^k \frac{\beta_{ji}}{\lambda_i - \lambda_j} \mathbf{x}_j + \frac{1}{\lambda_i} E \mathbf{x}_i.$$

Putting the k columns together, we have

$$(\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_k) \approx (\mathbf{x}_1, \dots, \mathbf{x}_k)R + E\left(\frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_k}{\lambda_k}\right). \quad (2.10)$$

Note that when A can be partitioned as in Equation 2.6, and the original coordinate α_u has only one non-zero entry x_{iu} as shown in Equation 2.7, the u -th row of $(\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_k)$ in Equation 2.10 can be simplified as:

$$\begin{aligned} \alpha_u &\approx x_{iu} \left(\frac{\beta_{i1}}{\lambda_1 - \lambda_i}, \dots, \frac{\beta_{i,i-1}}{\lambda_{i-1} - \lambda_i}, 1, \frac{\beta_{i,i+1}}{\lambda_{i+1} - \lambda_u}, \dots, \frac{\beta_{ik}}{\lambda_k - \lambda_i} \right) \\ &+ \left(\frac{1}{\lambda_1} \sum_{v \in C_1} e_{uv} x_{1v}, \dots, \frac{1}{\lambda_k} \sum_{v \in C_k} e_{uv} x_{kv} \right), \\ &= x_{iu} \mathbf{r}_i + \left(\sum_{v \in \Gamma_u^1} \frac{e_{uv} x_{1v}}{\lambda_1}, \dots, \sum_{v \in \Gamma_u^k} \frac{e_{uv} x_{kv}}{\lambda_k} \right). \end{aligned}$$

□

Note that e_{uv} in the right hand side (RHS) of Equation 2.8 can be further removed since $e_{uv} = 1$ in our setting. We include e_{uv} there for extension to general perturbations. Our next result shows that spectral coordinates also locate along k quasi-orthogonal lines \mathbf{r}_i (the i -th row of R), instead of exactly on the axes ξ_i when the graph is disconnected.

Proposition 2.2: For a graph $\tilde{A} = A + E$, spectral coordinates form k approximately orthogonal lines. Specifically, for any node $u \in C_i$, if it is not directly connected with other communities, α_u lies on the line \mathbf{r}_i ; otherwise, α_u deviates from lines \mathbf{r}_i ($i = 1, \dots, k$), where \mathbf{r}_i is the i -th row of matrix R shown in Equation 2.9.

Proof. First we prove that node $u \in C_i$ locates on the line \mathbf{r}_i . When node u has no connections to other communities, the second part of the RHS of Equation 2.8 is $\mathbf{0}$.

Hence $\alpha_u \approx x_{iu}\mathbf{r}_i$. When node u has some connections outside C_i , the second part of its spectral coordinate in Equation 2.8 is not equal to $\mathbf{0}$, and it thus deviates from line \mathbf{r}_i .

Next we prove that lines \mathbf{r}_i are approximate orthogonal. Let $W = R - I$, then $W^T + W = \mathbf{0}$ since $\beta_{ij} = \beta_{ji}$. Hence $R^T R = (I + W^T)(I + W) = I - W^T W$. The (i, j) entry of matrix $W^T W$ is $\sum_{t \neq i, j} \frac{\beta_{it}}{\lambda_t - \lambda_i} \frac{\beta_{tj}}{\lambda_j - \lambda_t}$. Note that the conditions of Theorem 2.1 imply that $\beta_{it} = \mathbf{x}_i^T E \mathbf{x}_t$ is much smaller than $|\lambda_t - \lambda_i|$, and hence $W^T W \approx \mathbf{0}$. Then, $R^T R \approx I$, and we prove the orthogonality property. \square

Two Dimensional Case

Nodes from C_1 lie along line \mathbf{r}_1 , while nodes from C_2 lie along line \mathbf{r}_2 , where

$$\mathbf{r}_1 = \left(1, \frac{\beta_{12}}{\lambda_2 - \lambda_1}\right), \quad \mathbf{r}_2 = \left(\frac{\beta_{21}}{\lambda_1 - \lambda_2}, 1\right).$$

Note that \mathbf{r}_1 and \mathbf{r}_2 are orthogonal since $\mathbf{r}_1 \mathbf{r}_2^T = 0$. For nodes that have connections to the other community, e.g., nodes u and v shown in Figure 2.2(a), their spectral coordinates scatter between two lines. For node u , its spectral coordinate can be approximated as

$$\alpha_u \approx x_{1u} \left(1, \frac{\beta_{12}}{\lambda_2 - \lambda_1}\right) + \left(0, \frac{\sum_{v \in \Gamma_u^2} x_{2v}}{\lambda_2}\right). \quad (2.11)$$

Its spectral coordinate jumps away from line \mathbf{r}_1 . The magnitude of jump is determined by spectral coordinates of its connected nodes in the community C_2 , as shown by the second parts of RHS of Equation 2.11. Since the jump vector is non-negative, node u gets closer to line \mathbf{r}_2 . Similarly, we can see for node v jumps towards line \mathbf{r}_1 . In

Figure 2.2(c), we can also see that both \mathbf{r}_1 and \mathbf{r}_2 rotate clockwise from the original axes. This is because $\beta_{12} = \mathbf{x}_1^T E \mathbf{x}_2 = \sum_{i,j} e_{ij} x_{1i} x_{2j} > 0$. There is a negative angle θ between line \mathbf{r}_1 and x -axis since $\tan \theta = \frac{\beta_{12}}{\lambda_2 - \lambda_1} < 0$.

2.4.3 Discussion

The line orthogonality property shown in Theorem 2.2 and Proposition 2.2 are based on the approximation shown in Theorem 2.1. Recall that Theorem 2.1 is derived from Lemma 2.1 which involves two conditions. The two conditions in Lemma 2.1 are naturally satisfied if the eigen-gap of any k leading eigenvalues is greater than $3\|E\|_2$ ($\|E\|_2$ is the largest eigenvalue of E), which guarantees the relative smaller change and the order of the eigenvectors preserved after perturbation. For condition 1, it is easy to verify that $\|\mathbf{x}_i^T E \mathbf{x}_i\|_2 = 0$. Since $\|U^T E U\|_2 \leq \|E\|_2$ for graph A with k disconnected comparable communities, the condition holds when the eigengap $\lambda_i - \lambda_{i+1}$ is greater than $\|E\|_2$. For condition 2, we can see $\|U^T E \mathbf{x}_i\|_2$ is also much smaller than $\|E\|_2$. Hence, condition 2 is satisfied when the eigengap $\lambda_i - \lambda_{i+1}$ is greater than $3\|E\|_2$. Note that $\|E\|_2$ is bounded by the maximum row sum of E and tends to be small when the perturbation edges are randomly added. We can even further relax the conditions. To have the line orthogonality property well preserved, we only need the subspace spanned by the first k eigenvectors stable, which is satisfied when the gap $\lambda_k - \lambda_{k+1}$ is large (Corollary V.2.2 in [Stewart and Sun, 1990]).

Theorem 2.1 requires that $|\lambda_i| \gg |\lambda_j|$, which is naturally satisfied for adjacency matrix A with k disconnected communities in our context. Please refer to Lemma 2.2. In Section 2.6, we will run empirical evaluations to show how well these conditions are met in real social networks and show the effect when conditions do not strictly

hold.

In this chapter, we assume the observed graph contains k comparable communities as well as some edges across communities and each community itself does not have a clear inner community structure. The assumption guarantees the largest eigenvalue of each community turns out to be one of the leading k eigenvalues of the observed graph. This assumption can be relaxed since large graphs usually contain communities at multiple levels [Karypis and Kumar, 1996]. In practice, the line orthogonality pattern still holds as long as the lower-layer community structure is not as significant as that of upper-layers, in which cases the second largest eigenvalue of every community is smaller than the largest eigenvalues of any other $k - 1$ communities.

2.4.4 Laplacian and Normal Eigenspaces

Our perturbation framework based on the adjacency eigenspace utilizes the eigenvectors of the largest k eigenvalues, which are more stable (due to large eigen-gaps) under perturbation. In this section, we examine the spectral spaces of the Laplacian matrix or the normal matrix and show why the line orthogonality pattern generally does not hold.

Recall that the Laplacian matrix \mathcal{L} is defined as $\mathcal{L} = D - A$, where $D = \text{diag}(d_1, \dots, d_n)$ and d_i is the degree of node i . The normal matrix \mathcal{N} is defined as $\mathcal{N} = D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$. We can easily derive that, for the block-wise diagonal graph, the spectral coordinate of node $u \in C_i$ in the Laplacian eigenspace is $(0, \dots, 1, \dots, 0)$ where the i -th entry is 1, indicating the node u 's community whereas the coordinate in the normal eigenspace is $(0, \dots, \sqrt{d_u}, \dots, 0)$. Note that the k eigenvectors corresponding to the smallest eigenvalues of \mathcal{L} capture the community structure. However, Lemma 2.1 is not ap-

plicable to $\tilde{\mathcal{L}}$ in general under perturbation, because the gap between the k smallest eigenvalues and the rest ones is too small and the two conditions in Lemma 2.1 are violated. For the normal matrix, all the eigenvalues of \mathcal{N} are between 1 and -1 . The conditions in Lemma 2.1 do not hold either because the eigen-gaps is generally smaller than $\|\Delta\mathcal{N}\|_2$. Hence it is impossible to explicitly express the perturbed spectral coordinates using the original ones and the perturbation matrix in the Laplacian or normal eigenspace. As a result, the line orthogonality disappears in the Laplacian or the normal eigenspace.

2.5 Adjacency Eigenspace based Clustering

In this section, we present a community partition algorithm, *AdjCluster*, which utilizes the line orthogonality pattern in the spectral space of the adjacency matrix. When a graph contains k clear communities, there exist k quasi-orthogonal lines in the k -dimensional spectral space and each line corresponds to a community in the graph. The spectral coordinate α_u should be close to the line corresponding to the community that the node u belongs to. In general, the idea of fitting k orthogonal lines directly in the k -dimensional space is complex. As shown in Algorithm 1, we project each spectral coordinate α_u to the unit sphere in the k -dimensional subspace by normalizing α_u to its unit length (line 3). We expect to observe that nodes from one community form a cluster on the unit sphere. Hence there will be k well separated clusters on the unit sphere. We apply the clustering k -means algorithm on the unit sphere to produce a partition of the graph (line 4).

To evaluate the quality of the partition and determine the k , we use the classic

Algorithm 1 *AdjCluster*: Adjacency Eigenspace based Clustering

Input: A, K **Output:** Clustering results

- 1: Compute $\mathbf{x}_1, \dots, \mathbf{x}_K$ by the eigen-decomposition of A with $\lambda_1 > \dots > \lambda_K$
 - 2: **for** $k = 2, \dots, K$ **do**
 - 3: $\boldsymbol{\alpha}_u = (x_{1u}, \dots, x_{ku})$ and $\bar{\boldsymbol{\alpha}}_u = \frac{\boldsymbol{\alpha}_u}{\|\boldsymbol{\alpha}_u\|}$;
 - 4: Apply k -means algorithm on $\{\bar{\boldsymbol{\alpha}}_u\}_{u=1, \dots, n}$;
 - 5: Compute fitting statistics from k -means algorithm ;
 - 6: **end for**
 - 7: Output partitions under k with the best fitting statistics.
-

Davies-Bouldin Index (*DBI*) [Davies and Bouldin, 1979]:

$$DBI = \frac{1}{k} \sum_{i=1, j \neq i}^k \left(\frac{\sigma_i + \sigma_j}{d(C_i, C_j)} \right). \quad (2.12)$$

where k is the number of clusters, σ_i is the average distance of all nodes in i -th cluster to centroid of C_i and $d(C_i, C_j)$ is the distance between centroids of C_i and C_j . The low *DBI* indicates output clusters with low intra-cluster distances and high inter-cluster distances. When the graph contains k clear communities, we expect to have the minimum *DBI* after applying k -means in the k -dimensional spectral space. We also expect all the angles between centroids of the output clusters are close to 90° since spectral coordinates form quasi-orthogonal lines in the determined k -dimensional spectral space. However, in the subspace spanned by fewer or more eigenvectors, the coordinates scatter in the spaces and do not form clear orthogonal lines, hence we will not obtain a very good fit after applying the k -means on the unit sphere.

Calculation of the eigenvectors of an $n \times n$ matrix takes in general a number of operations $O(n^3)$, which is almost inapplicable for large networks. However, in our algorithm here, we only need to calculate the first K eigen-pairs. We can determine

the appropriate K as examining the eigen-gaps [Stewart and Sun, 1990]. Furthermore, adjacency matrices in our context are usually sparse. The Arnoldi/Lanczos algorithm [Golub and Van Loan, 1996] generally needs $O(n)$ rather than $O(n^2)$ floating point operations at each iteration.

2.6 Empirical Evaluation

We use several real network data sets in our evaluation: Political books and Political blogs¹, Enron², and Facebook dataset [Viswanath et al., 2009]. We also generate two synthetic graphs: *Syn-1* and *Syn-2*. The *Syn-1* has 5 communities with the number of nodes 200, 180, 170, 150, and 140 respectively, and each community is generated separately with a power law degree distribution with the parameter 2.3. We add cross community edges randomly and keep the ratio between inter-community edges and inner-community edges as 20% in *Syn-1*. *Syn-2* is the same as the *Syn-1* except that we increase the number of links between community C_4 and C_5 to 80%. As a result, the *Syn-2* has four communities.

2.6.1 Line Orthogonality Property

We use spectral plots to check the line orthogonality property in various networks. We can clearly observe from Figures 2.3(a), 2.3(b), and 2.3(c) that for *Syn-1* there exist five orthogonal lines in the spectral space spanned by $\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_5$ and nodes from the same community (denoted by different colors) lie on the same line. For *Syn-2*, we can see from Figure 2.3(d) and 2.3(e) that communities also exhibit clear orthogonal lines in the adjacency spectral space spanned by $\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_4$, and each

¹<http://www-personal.umich.edu/~mejn/netdata/>

²<http://www.cs.cmu.edu/~enron/>

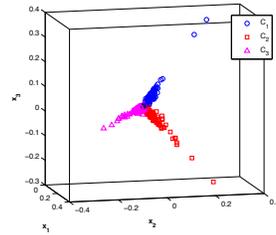
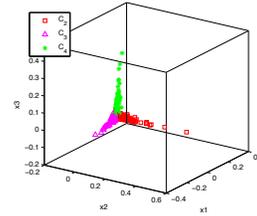
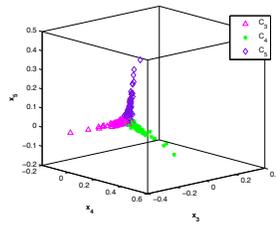
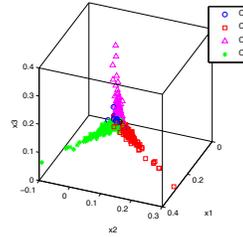
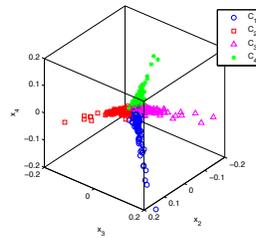
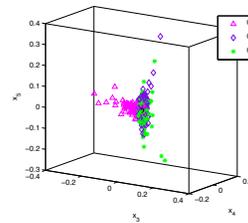
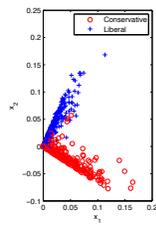
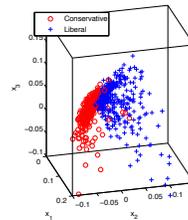
(a) *Syn-1*: $\tilde{x}_1, \tilde{x}_2, \tilde{x}_3$ (b) *Syn-1*: $\tilde{x}_2, \tilde{x}_3, \tilde{x}_4$ (c) *Syn-1*: $\tilde{x}_3, \tilde{x}_4, \tilde{x}_5$ (d) *Syn-2*: \tilde{x}_1, \tilde{x}_2 and \tilde{x}_3 (e) *Syn-2*: \tilde{x}_2, \tilde{x}_3 and \tilde{x}_4 (f) *Syn-2*: $\tilde{x}_3, \tilde{x}_4, \tilde{x}_5$ (g) *Polblogs*: \tilde{x}_1, \tilde{x}_2 (h) *Polblogs*: $\tilde{x}_1, \tilde{x}_2, \tilde{x}_3$

Figure 2.3: Illustration examples of line orthogonality property

line corresponds to one community. We are particularly interested in the subspace spanned by $\tilde{\mathbf{x}}_3, \tilde{\mathbf{x}}_4, \tilde{\mathbf{x}}_5$ for *Syn-2*. As shown in Figure 2.3(f), we can observe that there is no clear line orthogonality pattern, which matches our theoretical analysis since there are actually four communities in *Syn-2*. We also show the 2-D and 3-D spectral plots of *Polblogs* data. As we already knew, there are two communities in this network. Hence, we can observe that spectral coordinates form two orthogonal lines in the subspace spanned by $\tilde{\mathbf{x}}_1$ and $\tilde{\mathbf{x}}_2$ as shown in Figure 2.3(g). However, there is no clear line orthogonality pattern when we add the additional eigenvector $\tilde{\mathbf{x}}_3$, as shown in Figure 2.3(h).

Table 2.1: Statistics of the spectra for some networks. δ values for both the Laplacian and the normal matrices (shown in bold) and $\|\Delta\mathcal{L}\|_2$ for the Laplacian matrix and $\|\Delta\mathcal{N}\|_2$ for the normal matrix (shown in italic) violate conditions in Lemma 2.1.

	<i>Polbooks</i>	<i>Polblogs</i>	<i>Syn-1</i>	<i>Syn-2</i>
Adjacency matrix				
γ	0.59	6.95	3.87	3.16
δ	3.08	30.8	2.44	3.23
$ \lambda_k - \lambda_{k+1} $	5.82	39.6	7.65	8.26
$\ E\ _2$	2.78	13.61	6.99	6.61
Laplacian matrix				
γ	1.54	12.1	4.10	4.11
δ	-11.7	-73.5	-23.7	-25.37
$ \mu_k - \mu_{k+1} $	0.24	0.16	0.30	0.30
$\ \Delta\mathcal{L}\ _2$	<i>11.2</i>	<i>69.3</i>	<i>15.8</i>	<i>15.64</i>
Normal matrix				
γ	0.144	0.15	0.24	0.27
δ	-0.526	-0.29	-1.04	-1.07
$ \nu_k - \nu_{k+1} $	0.139	0.07	0.20	0.20
$\ \Delta\mathcal{N}\ _2$	<i>0.650</i>	<i>0.35</i>	<i>0.76</i>	<i>0.78</i>

Our theoretical analysis in Section 2.4.4 showed that the orthogonality pattern does not hold in either the Laplacian or the normal eigenspace because their small eigen-gap values affect the stability of the spectral space (Recall the conditions in

Lemma 2.1 and Theorem 2.1). Table 2.1 shows the calculated values of γ , δ , eigengap, and the magnitude of perturbations in adjacency, the Laplacian, and the normal eigenspaces for various networks. We can see that for adjacency matrices, all the networks generally satisfy conditions, which explains line orthogonality patterns in their adjacency eigenspaces. However, for the Laplacian or the normal matrices, none of networks satisfies the conditions. For example, all δ values for the Laplacian or the normal matrix (shown in bold) are less than zero, violating Condition 1 in Lemma 2.1; all values of $\|\Delta\mathcal{L}\|_2$ or $\|\Delta\mathcal{N}\|_2$ (shown in italic) are less than their corresponding eigengaps, incurring the violation of Condition 2 in Lemma 2.1; and the eigengaps ($|\mu_k - \mu_{k+1}|$, $|\nu_k - \nu_{k+1}|$) are relatively small, violating the condition in Theorem 2.1. Hence, the orthogonality pattern does not hold in the Laplacian or the normal eigenspaces.

2.6.2 Quality of Community Partition

Table 2.2: Statistics of networks and partition quality of *AdjCluster* (“ k ” is the number of communities, “*DBI*” is the Davies-Bouldin Index, “Angle” is the average angle between centroids, and “*Q*” is the modularity.)

Dataset	n	m	k	<i>DBI</i>	Angle	<i>Q</i>
<i>Syn-1</i>	840	4917	5	0.45	80.7°	0.37
<i>Syn-2</i>	840	5389	4	0.49	76.5°	0.34
<i>Polbooks</i>	105	441	2	0.15	83.8°	0.45
<i>Polblogs</i>	1222	16714	2	0.17	90.4°	0.42
<i>Enron</i>	148	869	6	0.59	88.9°	0.48
<i>Facebook</i>	63392	816886	9	0.83	83.6°	0.51

Table 2.2 shows the quality of our graph partition algorithm *AdjCluster*. The algorithm chooses the value of k that incurs the minimum *DBI* for each network data set. For a network with a clear community structure, we expect that the *DBI* is small, the modularity is away from zero, and the average angle is close to 90° since

there exist k quasi-orthogonal lines in the spectral space. We can see from Table 2.2 that all networks show relatively clear community structures. Notice the networks all have DBI lower than 1. It indicates there is relative strong community structure in each network. We also notice that all the average angles of the real world datasets are over 80° . This result verifies the orthogonality of the lines or clusters in the spectral space.

Table 2.3: Accuracy (%) of clustering results (“Lap” denotes the geometric Laplacian clustering, “NCut” denotes the normalized cut, “HE’” denotes the modularity based clustering, and SpokEn denotes EigenSpoke.)

Dataset	AdjCluster	Lap	NCut	HE’	SpokEn
<i>Syn-1</i>	90.8	57.5	84.4	49.1	40.2
<i>Syn-2</i>	85.1	62.8	80.1	45.9	44.7
<i>Polbooks</i>	96.7	93.5	96.7	88.0	93.5
<i>Polblogs</i>	94.7	58.8	95.3	92.4	91.9

The original data descriptions of *Polbooks* and *Polblogs* (and *Syn-1/Syn-2*) provide node-community relations. So we are able to compare different algorithms in terms of accuracy. The accuracy is defined as $\frac{\sum_{i=1}^k |C_i \cap \hat{C}_i|}{n}$ where \hat{C}_i denotes the i -th community produced by different algorithms. In our experiment, we compare our algorithm, *AdjCluster*, with four graph partition algorithms: one Laplacian based algorithm (the geometric spectral clustering) [John R. Gilbert and Teng, 1998], one normal based algorithm (the normalized cut [Shi and Malik, 2000]), one modularity based agglomerative clustering algorithm (HE’ [Wakita and Tsurumi, 2007]), and the EigenSpoke algorithm (SpokEn [Prakash et al., 2010]). Table 2.3 shows the accuracy values on the above four networks. Note that we cannot report accuracy values for Enron or Facebook since we do not know about their exact true community partitions. We can see that the quality of the partitioning produced by our algorithm, *AdjCluster*,

is better than (or comparable with) that produced by the normalized cut in terms of accuracy. On the contrary, the Laplacian spectrum based algorithm, the modularity based agglomerative clustering algorithm, and the EigenSpoke algorithm produce significant low accuracy values, which matches our theoretical analysis.

2.7 Summary

In this chapter, we show why node points in a graph with k communities exhibit the k orthogonal line pattern in the spectral subspace spanned by the principal eigenvectors of the graph's adjacency matrix. We treat the observed graph as a perturbation variant from the graph with k disconnected communities. The latter has the first k eigenvectors of a clear pattern: each has nonzero entries only for nodes within one specific community. We apply the matrix perturbation theory to get the approximated eigenvectors of the former. We then construct spectral coordinates by eigenvectors and show the nodes in adjacency spectral space. Specifically we show that 1) spectral coordinates of nodes with no direct links to other communities locate exactly on the orthogonal lines; 2) spectral coordinates of nodes with links to other communities deviate from lines; and 3) for a network with k communities there exist k orthogonal lines (and each community forms one line) in the spectral subspace formed by the first k eigenvectors of the adjacency matrix. We further give explicit formula (as well as its conditions) to quantify how much orthogonal lines rotate from the canonical axes and how far spectral coordinates of nodes with direct links to other communities deviate from the line of their own community. We also examine the spectral spaces of the Laplacian matrix and the normal matrix under the same perturbation framework. Our findings show that the line orthogonality pattern in general does not hold in the

Laplacian or the normal spectral space. We conduct empirical evaluations on both synthetic data and real-world social networks and validate our theoretical findings. We also present a graph partition algorithm, *AdjCluster*. The idea is to project nodes coordinates to the unit sphere and apply the classic k -means to find cluster. The line orthogonality property ensures the effectiveness of this graph partition algorithm and the empirical evaluations show competitive analytic results compare with many existing algorithms. The work in this chapter was published in the 22nd International Joint Conference on Artificial Intelligence [Wu et al., 2011].

In next chapter, we similarly apply the matrix perturbation theorem to study the spectral properties of signed graphs. The structures of signed graphs are more complicated due to the introduction of negative edges into the graph. We divide signed graphs into three basic types and show the patterns that exhibit in their adjacency spectral spaces. Based on the theoretical results, we extend graph partition algorithm *AdjCluster* to *UniAdjCluster* to cover the graph partition problem in signed graphs.

CHAPTER 3: SPECTRAL ANALYSIS OF COMMUNITY STRUCTURE IN SIGNED GRAPHS

Previous studies on social networks are often focused on networks with only positive relationships between individual nodes. As a significant extension, we conduct the spectral analysis on graphs with both positive and negative relationships. Specifically, we investigate the different impacts of positive and negative edges to the graph and community related patterns in the spectral space of the graph's adjacency matrix.

3.1 Introduction

In the last chapter, we have shown the community structure is closely related with the adjacency spectral properties in unsigned graphs. In some networks, however, relationships between two nodes could be inherently negative to express distrust or dislike. In contrast to the extensive studies on social networks that restrict to only positive relationships between individuals, in this chapter we study signed networks with both positive and negative relationships.

In anthropology and sociology, signed networks were originally used to model friendship and enmity [Davis, 1967; Hage and Harary, 1983]. The psychologists use -1 , 0 , and 1 to represent disliking, indifference, and liking, respectively. Graph topology of signed networks can then be expressed as an adjacency matrix where an entry is 1 (or -1) if the relationship is positive (or negative) and 0 if the relationship is absent. Positive and negative edges work differently in the networks. Positive rela-

tionships such as friendship or cooperation usually group people together and they mostly exist inside communities. Negative relationships such as distrust or dispute usually separate people into different groups and they mostly exist among communities. To form a community, positive and negative edges sometimes work harmonically. k -balanced signed graphs, first introduced in social psychology, describe this special status: there are only positive inner-community edges and negative inter-community edges in the graph. Our theoretical results show that communities in a k -balanced signed graph are distinguishable in the spectral space of its signed adjacency matrix, even with dense negative edges among communities. This is very different from that of unsigned graphs where communities tend to mix together when connections between communities become dense. In other cases, positive and negative edges may not work so well together. Some graphs are dominated by positive inner-community edges. We are specially interested in the situations when these graphs have dense inter-community edges. We will show that the negative inter-community edges offset the rotation effect of the positive inter-community edges. When positive and negative inter-community edges are of equal magnitude, the graph still keeps a similar community structure as that decided by the inner-community edges. Some other graphs may have dominated negative inter-community edges and no/few inner-community edges. We call them as partite-dominated signed graphs and these graphs are of very different structures compared with all the graphs we discussed before. They do not appear to be close to block-wise diagonal matrices. However, we will show that the ideal case of partite-dominated signed graphs, the k -partite graph, can be transformed to a similar type of signed graphs with dominated positive inner-community

edges and we can apply similar analytic procedures.

These three types of signed graphs cover the community structures decided by both positive and negative edges, by dominated positive edges only and by dominated negative edges only. In the real world situation, various types of community structures may exist in the same graph. By sorting eigenvalues in the sequence of their magnitude, we find the principal eigenvectors well capture the community structure. The only difference from unsigned graphs is that some of the eigenvalues of signed graphs are negative. We thus extend our graph partition algorithm *AdjCluster* to *UniAdjCluster* to include the large negative eigenvalues. We then run empirical evaluation on several synthetic data and real social networks to validate our theoretic results.

3.2 Notation

In this chapter, $A_{n \times n}$ represents the adjacency matrix of a signed graph G . $a_{ij} = 1$ if there is a positive edge between the nodes i and j , $a_{ij} = -1$ if there is a negative edge between the nodes i and j , and $a_{ij} = 0$ otherwise. A has n real eigenvalues. Let λ_i be the i -th largest eigenvalue of A with the eigenvector \mathbf{x}_i , $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n|$. Let x_{ij} denote the j -th entry of \mathbf{x}_i . The spectral decomposition of A is $A = \sum_i \lambda_i \mathbf{x}_i \mathbf{x}_i^T$.

Recall formula 2.1 in Chapter 2 as above. The eigenvector \mathbf{x}_i is represented as a column vector. There usually exist k leading eigenvalues that are significantly greater than the remaining ones for networks with k well-separated communities. We call the row vector $\boldsymbol{\alpha}_u = (x_{1u}, x_{2u}, \dots, x_{ku})$ the spectral coordinate of node u in the k -dimensional subspace spanned by $(\mathbf{x}_1, \dots, \mathbf{x}_k)$. This subspace contains most topological information of the original graph. We denote the i -th canonical basis as $\xi_i = (0, \dots, 0, 1, 0 \dots, 0)$, where the i -th entry of ξ_i is 1 and all other entries are zero.

$$\begin{array}{cccc}
\mathbf{x}_1 & & \mathbf{x}_i & & \mathbf{x}_k & & \mathbf{x}_n \\
& & \downarrow & & & & \\
\mathbf{\alpha}_u \rightarrow & \left(\begin{array}{ccc|ccc}
x_{11} & \cdots & x_{i1} & \cdots & x_{k1} & \cdots & x_{n1} \\
\vdots & & \vdots & & \vdots & & \vdots \\
\hline
x_{1u} & \cdots & x_{iu} & \cdots & x_{ku} & \cdots & x_{nu} \\
\hline
\vdots & & \vdots & & \vdots & & \vdots \\
x_{1n} & \cdots & x_{in} & \cdots & x_{kn} & \cdots & x_{nn}
\end{array} \right)
\end{array}$$

Let E be a symmetric perturbation matrix, and \tilde{A} be the adjacency matrix after perturbation, i.e., $\tilde{A} = A + E$. Similarly, let μ_i be the i -th largest eigenvalue of \tilde{A} with eigenvector $\tilde{\mathbf{x}}_i$, and \tilde{x}_{ij} is the j -th entry of $\tilde{\mathbf{x}}_i$. Row vector $\mathbf{\alpha}_u = (\tilde{x}_{1u}, \dots, \tilde{x}_{ku})$ is the spectral coordinate of node u after perturbation.

3.3 Spectral Properties of k -Balanced Signed Graphs and Their Variants

The k -balanced graph is one type of signed graphs that have received extensive examinations in social psychology. It was shown that the stability of sentiments is equivalent to k -balanced (clusterable). A necessary and sufficient condition for a signed graph to be k -balanced is that the signed graph does not contain a cycle with exactly one negative edge [Davis, 1967].

Definition 3.1: Graph G is a k -balanced graph if the node set V can be divided into k non-trivial disjoint subsets such that V_1, \dots, V_k , edges connecting any two nodes from the same subset are all positive, and edges connecting any two nodes from different subsets are all negative.

The k node sets, V_1, \dots, V_k , naturally form k communities denoted by C_1, \dots, C_k respectively. Let $n_i = |V_i|$ ($\sum_i n_i = n$), and A_i be the $n_i \times n_i$ adjacency matrix of community C_i . After ordering the nodes properly, the adjacency matrix \tilde{A} of a k -balanced graph can be written as:

$$B = A + E, \quad \text{where} \quad A = \begin{pmatrix} A_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & A_k \end{pmatrix}, \quad (3.1)$$

and E represents the negative edges across communities. More generally, $e_{uv} = 1(-1)$ if a positive(negative) edge is added between the nodes u and v , and $e_{uv} = 0$ otherwise.

For a graph with k disconnected communities, its adjacency matrix A is shown in Equation 3.1. Recall our discussion about such graphs in last chapter. When the k communities are comparable in size, the first k eigenvectors of A are as follows:

$$(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k) = \begin{pmatrix} \mathbf{x}_{C_1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{x}_{C_2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{x}_{C_k} \end{pmatrix} \quad (3.2)$$

where \mathbf{x}_{C_i} is the first eigenvector for community C_i . Now, consider the node u in community C_i . The spectral coordinate of node u is just the u -th row of the matrix in Equation 3.2. Then, we have

$$\boldsymbol{\alpha}_u = (0, \dots, 0, x_{iu}, 0, \dots, 0), \quad (3.3)$$

where $x_{iu} > 0$ is the only non-zero entry of $\boldsymbol{\alpha}_u$. A graph with k disconnected comparable communities has spectral coordinates of all nodes located on k positive half-axes

of canonical basis ξ_1, \dots, ξ_k and nodes from the same community locate on the same half axis.

Let Γ_u^i ($i = 1, \dots, k$) be the set of nodes in C_i that are newly connected to node u by perturbation E : $\Gamma_u^i = \{v : v \in C_i, e_{uv} = \pm 1\}$. Recall that in last chapter we derived several theoretical results on general graph perturbation. We include the approximation of spectral coordinates below as a basis for our spectral analysis of signed graphs.

Theorem 3.1: Let A be a block-wise diagonal matrix as shown in Equation 3.1, and E be a symmetric perturbation matrix satisfying $\|E\|_2 \ll \lambda_k$. Let $\beta_{ij} = \mathbf{x}_i^T E \mathbf{x}_j$. For a graph with the adjacency matrix $\tilde{A} = A + E$, the spectral coordinate of an arbitrary node $u \in C_i$ can be approximated as

$$\boldsymbol{\alpha}_u \approx x_{iu} \mathbf{r}_i + \left(\sum_{v \in \Gamma_u^1} \frac{e_{uv} x_{1v}}{\lambda_1}, \dots, \sum_{v \in \Gamma_u^k} \frac{e_{uv} x_{kv}}{\lambda_k} \right) \quad (3.4)$$

where scalar x_{iu} is the only non-zero entry in its original spectral coordinate shown in Equation 3.3, and \mathbf{r}_i is the i -th row of matrix R in Equation 3.5:

$$R = \begin{pmatrix} 1 & \frac{\beta_{12}}{\lambda_2 - \lambda_1} & \dots & \frac{\beta_{1k}}{\lambda_k - \lambda_1} \\ \frac{\beta_{21}}{\lambda_1 - \lambda_2} & 1 & \dots & \frac{\beta_{2k}}{\lambda_k - \lambda_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\beta_{k1}}{\lambda_1 - \lambda_k} & \frac{\beta_{k2}}{\lambda_2 - \lambda_k} & \dots & 1 \end{pmatrix}. \quad (3.5)$$

3.3.1 Moderate Inter-community Edges

Proposition 3.1: Let $B = A + E$ where A has k disconnected communities and $\|E\|_2 \ll \lambda_k$ and E is non-positive. We have the following properties:

1. If node $u \in C_i$ is not connected to any C_j ($j \neq i$), α_u lies on the half-line \mathbf{r}_i that starts from the origin, where \mathbf{r}_i is the i -th row of matrix R shown in Equation 3.5. The k half-lines are approximately orthogonal to each other.
2. If node $u \in C_i$ is connected to node $v \in C_j$ ($j \neq i$), α_u deviate from \mathbf{r}_i . Moreover, the angle between α_u and \mathbf{r}_j is an obtuse angle.

To illustrate Proposition 3.1, we now consider a 2-balanced graph. Suppose that a graph has two communities and we add some sparse edges between two communities. For node $u \in C_1$ and $v \in C_2$, with Equation 3.4, the spectral coordinates can be approximated as

$$\alpha_u \approx x_{1u}\mathbf{r}_1 + \left(0, \frac{1}{\lambda_2} \sum_{v \in \Gamma_u^2} e_{uv}x_{2v}\right), \quad (3.6)$$

$$\alpha_v \approx x_{2v}\mathbf{r}_2 + \left(\frac{1}{\lambda_1} \sum_{u \in \Gamma_v^1} e_{uv}x_{1u}, 0\right), \quad (3.7)$$

where $\mathbf{r}_1 = \left(1, \frac{\beta_{12}}{\lambda_2 - \lambda_1}\right)$ and $\mathbf{r}_2 = \left(\frac{\beta_{21}}{\lambda_1 - \lambda_2}, 1\right)$.

The Item 1 of Proposition 3.1 is apparent from Equation 3.6 and Equation 3.7. For those nodes with no inter-community edges, the second parts of the right-hand side (RHS) of Equation 3.6 and Equation 3.7 are 0 since all e_{uv} 's are 0, and hence they lie on the two half-lines \mathbf{r}_1 (nodes in C_1) and \mathbf{r}_2 (nodes in C_2). Note that \mathbf{r}_1 and \mathbf{r}_2 are orthogonal since $\mathbf{r}_1\mathbf{r}_2^T = 0$.

Next, we explain Item 2 of Proposition 3.1. Consider the inner product

$$\langle \alpha_u, \mathbf{r}_2 \rangle = \alpha_u \mathbf{r}_2^T = \frac{1}{\lambda_2} \sum_{v \in \Gamma_u^2} e_{uv}x_{2v}.$$

If node $u \in C_1$ has some negative links to C_2 ($e_{uv} = -1$), $\langle \alpha_u, \mathbf{r}_2 \rangle$ is thus negative.

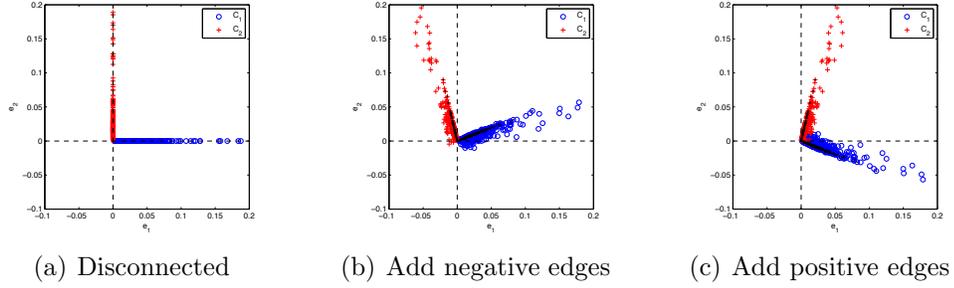


Figure 3.1: *Syn-2Bal*: rotation and deviation with inter-community edges ($p = 0.05$)

In other words, α_u lies outside the two half-lines \mathbf{r}_1 and \mathbf{r}_2 .

Another interesting pattern is the direction of rotation of the two half lines. For the 2-balanced graph, \mathbf{r}_1 and \mathbf{r}_2 rotate counter-clockwise from the axis ξ_1 and ξ_2 . Notice that all the added edges are negative ($e_{uv} = -1$), and hence $\beta_{12} = \beta_{21} = \mathbf{x}_1^T E \mathbf{x}_2 = \sum_{u,v=1}^n e_{uv} x_{1u} x_{2v} < 0$. Therefore, $\frac{\beta_{12}}{\lambda_2 - \lambda_1} > 0$, $\frac{\beta_{21}}{\lambda_1 - \lambda_2} < 0$, which implies that \mathbf{r}_1 and \mathbf{r}_2 have an counter-clockwise rotation from the basis.

Comparison with Adding Positives Edges

When the added edges are all positive ($e_{uv} = 1$), we can derive the following two properties in a similar manner:

1. Nodes with no inter-community edges lie on the k half-lines. (When $k = 2$, the two half-lines exhibit a clockwise rotation from the axes.)
2. For node $u \in C_i$ that connects to node $v \in C_j$, α_u and \mathbf{r}_j form an acute angle.

Figure 3.1 shows the scatter plot of the spectral coordinates for a synthetic graph, *Syn-2Bal*. *Syn-2Bal* is a 2-balanced graph with 600 and 400 nodes in each community. We generate *Syn-2Bal* and modify its inter-community edges via the same method as synthetic data set *Syn-3Bal* in Section 3.3.4. As we can see in Figure 3.1(a), when

the two communities are disconnected, the nodes from C_1 and C_2 lie on the positive part of axis ξ_1 and ξ_2 respectively. We then add a small number of edges connecting the two communities ($p = 0.05$). When the added edges are all negative, as shown in Figure 3.1(b), the spectral coordinates of the nodes from the two communities form two half-lines respectively. The two quasi-orthogonal half-lines rotate counter-clockwise from the axes. Nodes with negative inter-community edges lie outside the two half-lines. On the contrary, if we add positive inter-community edges, as shown in Figure 3.1(c), the nodes from two communities display two half-lines with a clockwise rotation from the axes, and nodes with inter-community edges lie between the two half-lines.

3.3.2 Increase the Number of Inter-community Edges

Theorem 3.1 holds when the magnitude of perturbation is moderate. With perturbation of large magnitude, we can divide the perturbation matrix into several perturbation matrices of small magnitude and approximate the eigenvectors step by step. More general, the perturbed spectral coordinate of a node u can be approximated as

$$\boldsymbol{\alpha}_u \approx \boldsymbol{\alpha}_u R + \sum_{v=1}^n e_{uv} \boldsymbol{\alpha}_v \Lambda^{-1}, \quad (3.8)$$

where $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_k)$.

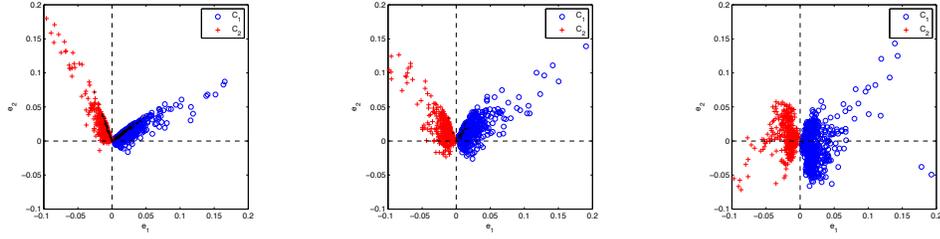
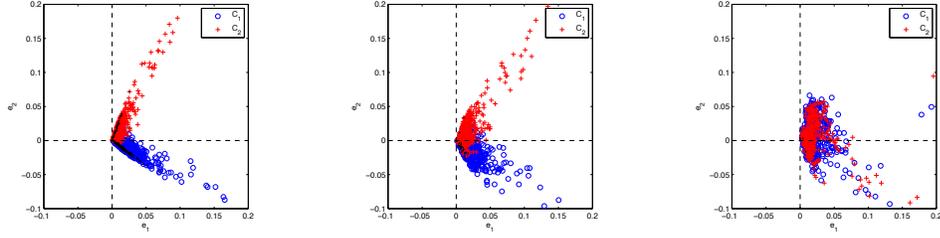
One property implied by Equation 3.8 is that, after adding negative inter-community edges, nodes from different communities are still separable in the spectral space. Note that R is close to an orthogonal matrix, and hence the first part of RHS of Equation 3.8 specifies an orthogonal transformation. The second part of RHS of Equation 3.8

specifies a deviation away from the position after the transformation. Note that when the inter-community edges are all negative ($e_{uv} = -1$), the deviation of α_u is just towards the negative direction of α_v (each dimension is weighted with λ_i^{-1}). Therefore, after perturbation, the nodes u and v are further separable from each other in the spectral space. The consequence of this repellency caused by adding negative edges is that nodes from different communities stay away from each other in the spectral space. As the magnitude of the noise increases, more nodes deviate from the half-lines r_i , and the line pattern eventually disappears.

Positive Large Perturbation

When the added edges are positive, we can similarly observe the opposite phenomenon: more nodes from the two communities are “pulled” closer to each other by the positive inter-community edges and are finally mixed together, indicating that the well-separable communities merge into one community.

Figure 3.2 shows the spectral coordinate of *Syn-2Bal* when we increase the number of inter-community edges ($p = 0.1, 0.3$ and 1). For the first row (Figure 3.2(a) to Figure 3.2(c)), we add negative inter-community edges in *Syn-2Bal*, and for the second row (Figures 3.2(d) to 3.2(f)), we add positive inter-community edges. As we add more and more inter-community edges, no matter positive or negative, more and more nodes deviate from the two half-lines, and finally the line pattern diminishes in Figure 3.2(c) or Figure 3.2(f). When adding positive inter-community edges, the nodes deviate from the lines and hence finally mix together as show in Figure 3.2(f), indicating that two communities merge into one community. Different from adding positive edges, which

(a) Negative edges ($p = 0.1$) (b) Negative edges ($p = 0.3$) (c) Negative edges ($p = 1$)(d) Positive edges ($p = 0.1$) (e) Positive edges ($p = 0.3$) (f) Positive edges ($p = 1$)Figure 3.2: *Syn-2Bal* with different types and sizes of inter-community edges

mixes the two communities in the spectral space, adding negative inter-community edges “pushes” the two communities away from each other. This is because nodes with negative inter-community edges lie outside the two half-lines as shown in Figure 3.2(a) and Figure 3.2(b). Even when $p = 1$, as shown in Figure 3.2(c), two communities are still clearly separable in the spectral space.

3.3.3 Approximate k -Balanced Signed Graphs

For unbalanced graphs that are close to k -balanced graphs, their topologies can be considered as perturbations on balanced graphs with some negative connections within communities and some positive connections across communities. Therefore, we can divide an approximated k -balanced signed graphs into three parts

$$B = A + E_{\text{in}} + E_{\text{out}}, \quad (3.9)$$

where A is a non-negative block-wise diagonal matrix as shown in Equation 3.1, E_{in} represents the negative edges within communities, and E_{out} represents the both negative and positive inter-community edges.

Add Negative Inner-Community Edges

For the block-wise diagonal matrix A , we first discuss the case where a small number of negative edges are added within the communities. E_{in} is also a block-wise diagonal. Hence $\beta_{ij} = \mathbf{x}_i^T E_{\text{in}} \mathbf{x}_j = 0$ for all $i \neq j$, and matrix R caused by E_{in} in Equation 3.5 is reduced to the identity matrix I .

Consider that we add one negative inner-community edge between the nodes $u, v \in C_i$. Since $R = I$, only λ_i and \mathbf{x}_i are involved in approximating α_u and α_v :

$$\begin{aligned} \alpha_u &= (0, \dots, 0, \tilde{x}_{iu}, 0, \dots, 0), & \tilde{x}_{iu} &\approx x_{iu} - \frac{x_{iv}}{\lambda_i} \\ \alpha_v &= (0, \dots, 0, \tilde{x}_{iv}, 0, \dots, 0), & \tilde{x}_{iv} &\approx x_{iv} - \frac{x_{iu}}{\lambda_i}. \end{aligned}$$

Without loss of generality, assume $x_{iu} > x_{iv}$, and we have the following properties when adding $e_{uv} = -1$:

1. Both the nodes u and v move towards the negative part of axis ξ_i after perturbation: $\tilde{x}_{iu} < x_{iu}$ and $\tilde{x}_{iv} < x_{iv}$.
2. Node v moves farther than u after perturbation: $|\tilde{x}_{iv} - x_{iv}| > |\tilde{x}_{iu} - x_{iu}|$.

The two preceding properties imply that, for those nodes close to the origin, adding negative edges would “push” them towards the negative part of axis ξ_i , and a small number of nodes can thus lie on the negative part of axis ξ_i , i.e., $\tilde{x}_{iu} < 0$.

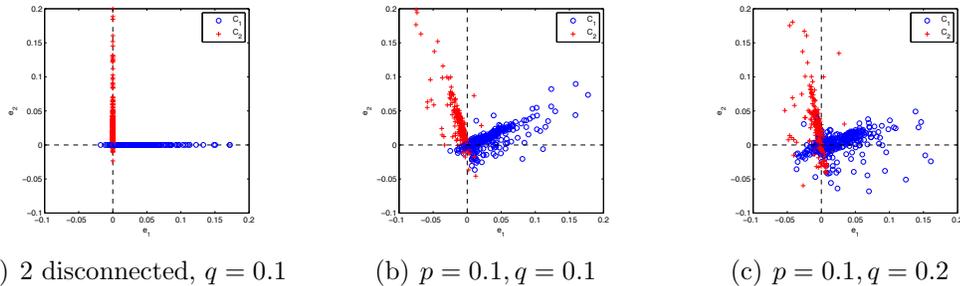


Figure 3.3: Spectral coordinates of unbalanced graphs perturbed from *Syn-2Bal*

Add Inter-Community Edges

The spectral perturbation caused by adding E_{out} to matrix $A + E_{\text{in}}$ can be complicated. Notice that $(A + E_{\text{in}})$ is still a block-wise matrix, and we can still apply Theorem 3.1 and conclude that, when E_{out} is moderate, the major nodes from k communities form k lines in the spectral space and nodes with inter-community edges deviate from the lines.

It is difficult to give the explicit form of the lines and the deviations, because x_{iu} and the inter-community edges can be either positive or negative. However, we expect that the effect of adding negative edges on positive nodes is still dominant in determining the spectral pattern, because most nodes lie along the positive part of the axes and the majority of inter-community edges are negative. Communities are still distinguishable in the spectral space. The majority of nodes in one community lie on the positive part of the line, while a small number of nodes may lie on the negative part due to negative connections within the community.

We make graph *Syn-2Bal* unbalanced by flipping the signs of a small proportion q of the edges. When the two communities are disconnected, as shown in Figure 3.3(a), after flipping $q = 0.1$ inner-community edges, a small number of nodes lie on the

negative parts of the two axes. Figure 3.3(b) shows the spectral coordinates of the unbalanced graph generated from balanced graph *Syn-2Bal* ($p = 0.1, q = 0.1$). Since the size of the inter-community edges is small, we can still observe two orthogonal lines in the scatter plots. The negative edges within the communities cause a small number of nodes lie on the negative parts of the two lines. Nodes with inter-community edges deviate from the two lines. For Figure 3.3(c), we flip more edge signs ($p = 0.1, q = 0.2$). We can observe that more nodes lie on the negative parts of the lines, since more inner-community edges are changed to negative. The rotation angles of the two lines are smaller than that in Figure 3.3(b). This is because the positive inter-community edges “pull” the rotation clockwise a little, and the rotation we observe depends on the effects from both positive and negative inter-community edges.

3.3.4 Evaluation on k -Balanced Signed Graphs

Synthetic Balanced Graph

Dataset *Syn-3Bal* is a synthetic 3-balanced graph generated from the power law degree distribution with the scaling exponent 2.5. The three communities of *Syn-3Bal* contain 600, 500, 400 nodes, and 4131, 3179, 2037 edges respectively. All the 13027 inter-community edges are set to be negative. We delete the inter-community edges randomly until a proportion p of them remain in the graph. The parameter p is the ratio of the size of inter-community edges to that of the inner-community edges. If $p = 0$ there are no inter-community edges. If $p = 1$, inner- and inter-community edges have the same size.

Figure 3.4 shows the change of spectral coordinates of *Syn-3Bal*, as we increase

the size of inter-community edges. When there are no negative links ($p = 0$), the scatter plot of the spectral coordinates is shown in Figure 3.4(a). The disconnected communities display three orthogonal half-lines. Figure 3.4(b) shows the spectral coordinates when the size of inter-community edges is moderate ($p = 0.1$). We can see the nodes form three half-lines that rotate a certain angle, and some of the nodes deviate from the lines. Figures 3.4(c) and 3.4(d) show the spectral coordinates when we increase the size of inter-community edges ($p = 0.3, 1$). We can observe that, as more inter-community edges are added, more and more nodes deviate from the lines. However, nodes from different communities are still separable from each other in the spectral space.

We also add positive inter-community edges on *Syn-3Bal* for comparison, and the spectral coordinates are shown in Figures 3.4(e) and 3.4(f). We can observe that, different from adding negative edges, as the size of inter-community edges (p) increases, nodes from the three communities get closer to each other, and completely mix in one community in Figure 3.4(f).

Synthetic Approximate k -Balanced Signed Graphs

To generate an unbalanced graph that close to k -balanced signed graphs, we randomly flip the signs of a small proportion q of the inner- and inter-community edges of a balanced graph, i.e., the parameter q is the proportion of unbalanced edges given the partition. We first flip edge signs of a small size of inter-community edges. Figures 3.5(a) and 3.5(b) show the spectral coordinates after we flip $q = 10\%$ and $q = 20\%$ edge signs on *Syn-3Bal* with $p = 0.1$. We can observe that, even the graph is

unbalanced, nodes from the three communities exhibit three lines starting from the origin, and some nodes deviate from the lines due to the inter-community edges.

We then flip edge signs of a large size of inter-community edges. Figure 3.5(c) shows the spectral coordinates after we flip $q = 20\%$ edge signs on *Syn-3Bal* with $p = 1$. We can observe that the line pattern diminishes because of the large number of inter-community edges. However, the nodes from three communities are separable in the spectral space, indicating that the unbalanced edges do not greatly change the patterns in the spectral space.

Comparison with The Laplacian Spectrum

The signed Laplacian matrix is defined as $L = \bar{D} - A$ where $\bar{D}_{n \times n}$ is a diagonal degree matrix with $\bar{D}_{ii} = \sum_{j=1}^n |A_{ij}|$ [Kunegis et al., 2010]. Note that the unsigned Laplacian matrix is defined as $L = D - A$ where $D_{n \times n}$ is a diagonal degree matrix with $D_{ii} = \sum_{j=1}^n A_{ij}$. The eigenvectors corresponding to the k smallest eigenvalues of the Laplacian matrix also reflect the community structure of a signed graph: the k communities form k clusters in the Laplacian spectral space. However, eigenvectors associated with the smallest eigenvalues are generally instable to noise according to the matrix perturbation theory [Stewart and Sun, 1990]. Hence, when it comes to real-world networks, the communities may no longer form distinguishable clusters in the Laplacian spectral space.

Figure 3.6(a) shows the Laplacian spectrum of a balanced graph, *Syn-3Bal* with $p = 0.1$. We can see that the nodes from the three communities form 3 clusters in the spectral space. However, the Laplacian spectrum is less stable to the noise. Figures

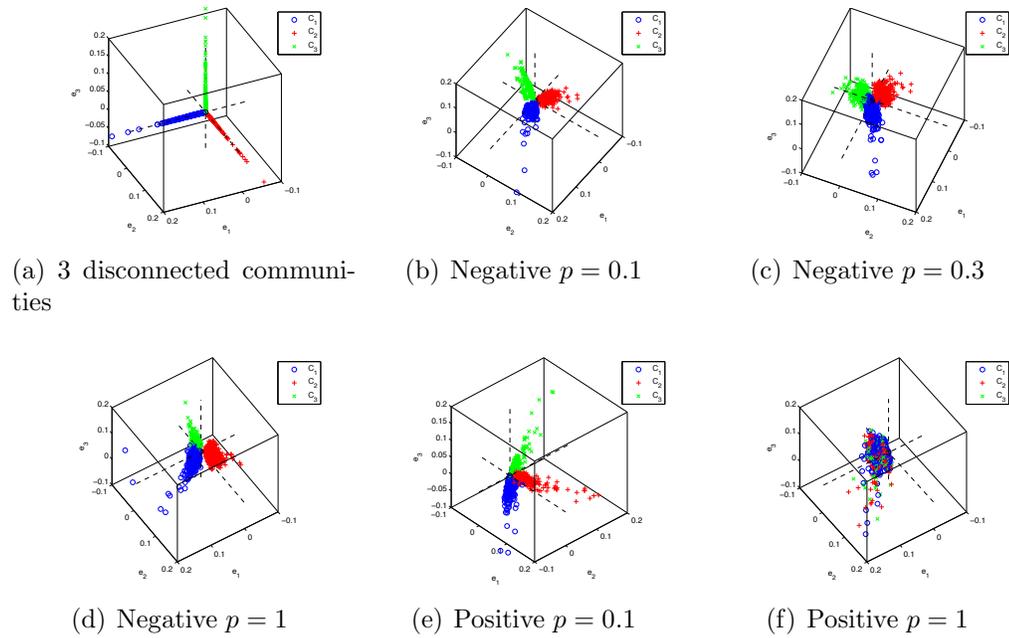


Figure 3.4: The spectral coordinates of the 3-balanced graph *Syn-3Bal*. (b)-(d): add negative inter-community edges; (e)-(f): add positive inter-community edges

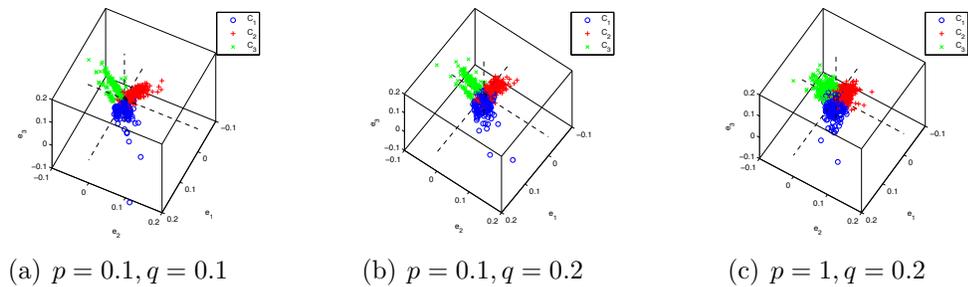


Figure 3.5: The spectral coordinates of a unbalanced synthetic graph generated via flipping signs of inner- and inter-community edges of *Syn-3Bal* with $p = 0.1$ or 1

3.6(b) and 3.6(c) plot the Laplacian spectra of the unbalanced graphs perturbed from *Syn-3Bal*. We can observe that C_1 and C_2 are mixed together in Figure 3.6(b) and all the three communities are not separable from each other in Figure 3.6(c). For comparison, the adjacency spectra of the corresponding graphs were shown in Figure 3.5(b) and Figure 3.5(c) respectively where we can observe that the three communities are well-separable in the adjacency spectral space.

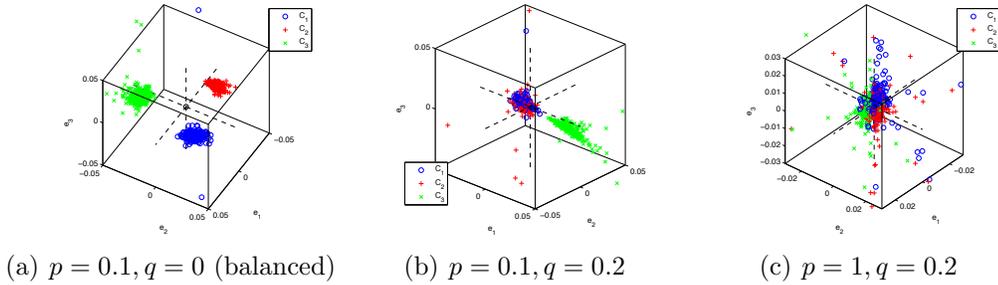


Figure 3.6: The Laplacian spectral space of signed graphs

3.4 Signed Graphs with Dominated Positive Inner-community Edges

In k -balanced signed graphs, the positive and negative edges work harmonically to form the communities. However, k -balanced signed graphs are only one type of special signed graphs. In other cases, the positive and negative edges may not work so harmonically and they may offset some effect of the edges with different signs. In this section, we focus on the signed graph with dominated positive inner-community edges. From the starting point as a graph with k disconnected communities, we add moderate inter-community edges of mixed signed edges. The graph still shows k orthogonal lines in the adjacency spectral space by Theorem 2.1. If we continue to add a large number of edges with different signs, the conditions of Theorem 2.1 may not hold. In the following, we specially interested in this situation when the

inter-community edges are dense. For these signed graphs, we show that there exist k orthogonal clusters in the spectral space spanned by the k principal eigenvectors if the newly added inter-community edges are nearly half positive and half negative. In k -balanced signed graphs, the communities are hostile to each other due to the negative inter-community edges while in unbalanced graphs with equally positive and negative edges, the communities are neutral to each other.

We separate the signed graphs with dominated positive inner-community edges into two parts. A is the adjacency matrix of a 0-1 graph with k disconnected communities. E is the perturbation matrix that contains the positive and negative edges added across communities. The conditions of Theorem 2.1 do not put restriction on the values of the entries. If E contains only a small number of edges, we can apply Theorem 2.1 directly to get the approximation of the principal eigenvectors of the original graphs. $\beta_{ji} = \mathbf{x}_j^T E \mathbf{x}_i = \sum x_{ju} e_{uv} x_{vi}$ where e_{uv} represents the newly added edge. According to Equation 3.2, all entries of \mathbf{x}_i are non-negative. So β_{ji} can be very small when the positive and negative edges are of equal quantity. When the magnitude of edges added across communities is large, we derive the following result:

Corollary 3.1: Denote the adjacency matrix of a graph as $\tilde{A} = A + E$ where A is as shown in Equation 3.1 and E denotes the signed edges across communities. If E can be decomposed into $E = E_r + E_t$ satisfying:

- E_r contains equal numbers of positive and negative edges so that $\mathbf{x}_j^T E_r \mathbf{x}_i$ is almost zero;
- E_t contains mostly one kind of edges and E_t satisfies the condition in Theorem

2.1;

we conclude that the leading k eigenvectors of \tilde{A} can be approximated as

$$\tilde{\mathbf{x}}_i \approx \mathbf{x}_i + \sum_{j=1, j \neq i}^k \frac{\tilde{\beta}_{ji}}{\lambda_i - \lambda_j} \mathbf{x}_j + \frac{1}{\lambda_i} E \mathbf{x}_i + \frac{1}{\lambda_i^2} E_t E_r \mathbf{x}_i + \sum_{j=1, j \neq i}^k \frac{\tilde{\beta}_{ji} E_r}{(\lambda_i - \lambda_j) \lambda_j} \mathbf{x}_j \quad (3.10)$$

where

$$\tilde{\beta}_{ji} \approx \mathbf{x}_j^T E_t \mathbf{x}_i + \frac{1}{\lambda_i} \mathbf{x}_j^T E_t E_r \mathbf{x}_i + \frac{1}{\lambda_j} \mathbf{x}_j^T E_r E_t \mathbf{x}_i.$$

Proof. E_r can be decomposed into a series of E_s for $s = 1, \dots, t-1$ where each E_s contains a very small number of edges and $\mathbf{x}_j^T E_s \mathbf{x}_i \approx 0$. We can apply Theorem 2.1 to get the approximated eigenvectors. Since $\mathbf{x}_j^T E_1 \mathbf{x}_i \approx 0$, the approximated eigenvectors for $A + E_1$ are $\tilde{\mathbf{x}}_i^{(1)} \approx \mathbf{x}_i + \frac{E_1}{\lambda_i} \mathbf{x}_i$. Because the edges are not added inside the communities, the eigenvalues remain almost the same: $\tilde{\lambda}_i \approx \lambda_i + \mathbf{x}_j^T E_1 \mathbf{x}_i = \lambda_i$. E_2 has a similar condition with E_1 . Thus the approximated eigenvectors of $A + E_1 + E_2$ are

$$\tilde{\mathbf{x}}_i^{(2)} \approx \mathbf{x}_i^{(1)} + \frac{E_2}{\lambda_i} \mathbf{x}_i^{(1)} \approx \mathbf{x}_i + \frac{E_1 + E_2}{\lambda_i} \mathbf{x}_i + \frac{E_2 E_1}{\lambda_i^2} \mathbf{x}_i. \quad (3.11)$$

$\|E_1\|_2$ and $\|E_2\|_2$ are small compared with λ_i and $\|E_2 E_1\|_2 \leq \|E_1\|_2 \|E_2\|_2$, so $\frac{E_2 E_1}{\lambda_i^2} \mathbf{x}_i$ has much smaller value than $\frac{E_1 + E_2}{\lambda_i} \mathbf{x}_i$. So the last term in Equation 3.11 is negligible. Then $\tilde{\mathbf{x}}_i^{(2)} \approx \mathbf{x}_i + \frac{E_1 + E_2}{\lambda_i} \mathbf{x}_i$. We follow similar procedure and get the approximated eigenvectors of $A + \sum_{s=1}^{t-1} E_s$:

$$\tilde{\mathbf{x}}_i^{(t-1)} \approx \mathbf{x}_i + \frac{1}{\lambda_i} E_r \mathbf{x}_i \quad (3.12)$$

When E_t satisfies the condition of Theorem 2.1, we have the approximated eigen-

vectors of $A + E$:

$$\tilde{\mathbf{x}}_i \approx \tilde{\mathbf{x}}_i^{(t-1)} + \sum_{j=1, j \neq i}^k \frac{\tilde{\beta}_{ji}}{\lambda_i - \lambda_j} \tilde{\mathbf{x}}_j^{(t-1)} + \frac{1}{\lambda_i} E_t \tilde{\mathbf{x}}_i^{(t-1)} \quad (3.13)$$

where

$$\begin{aligned} \tilde{\beta}_{ji} &= \left(\tilde{\mathbf{x}}_j^{(t-1)} \right)^T E_t \tilde{\mathbf{x}}_i^{(t-1)} \\ &\approx \mathbf{x}_j^T E_t \mathbf{x}_i + \frac{1}{\lambda_i} \mathbf{x}_j^T E_t E_r \mathbf{x}_i + \frac{1}{\lambda_j} \mathbf{x}_j^T E_r E_t \mathbf{x}_i + \frac{1}{\lambda_i \lambda_j} \mathbf{x}_j^T E_r E_t E_r \mathbf{x}_i \end{aligned}$$

As $\lambda_i \lambda_j$ is much larger than $\|E_t\|_2$, we omit the last term from $\tilde{\beta}_{ji}$ since it is much smaller than the second and third terms. We plug Equation 3.12 into Equation 3.13 and have:

$$\begin{aligned} \tilde{\mathbf{x}}_i &\approx \mathbf{x}_i + \frac{E_r}{\lambda_i} \mathbf{x}_i + \sum_{j=1, j \neq i}^k \frac{\tilde{\beta}_{ji}}{\lambda_i - \lambda_j} \left(\mathbf{x}_j + \frac{E_r}{\lambda_j} \mathbf{x}_j \right) + \frac{E_t}{\lambda_i} \left(\mathbf{x}_i + \frac{E_r}{\lambda_i} \mathbf{x}_i \right) \\ &= \mathbf{x}_i + \sum_{j=1, j \neq i}^k \frac{\tilde{\beta}_{ji}}{\lambda_i - \lambda_j} \mathbf{x}_j + \frac{1}{\lambda_i} E \mathbf{x}_i + \frac{1}{\lambda_i^2} E_t E_r \mathbf{x}_i + \sum_{j=1, j \neq i}^k \frac{\tilde{\beta}_{ji} E_r}{(\lambda_i - \lambda_j) \lambda_j} \mathbf{x}_j \end{aligned}$$

□

We write Equation 3.10 in matrix form:

$$\begin{aligned} (\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_k) &\approx (\mathbf{x}_1, \dots, \mathbf{x}_k) \tilde{R} + E \left(\frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_k}{\lambda_k} \right) \\ &+ E_t E_r \left(\frac{\mathbf{x}_1}{\lambda_1^2}, \dots, \frac{\mathbf{x}_k}{\lambda_k^2} \right) + E_r \left(\frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_k}{\lambda_k} \right) (\tilde{R} - I) \end{aligned} \quad (3.14)$$

where

$$\tilde{R} = \begin{pmatrix} 1 & \frac{\tilde{\beta}_{12}}{\lambda_2 - \lambda_1} & \cdots & \frac{\tilde{\beta}_{1k}}{\lambda_k - \lambda_1} \\ \frac{\tilde{\beta}_{21}}{\lambda_1 - \lambda_2} & 1 & \cdots & \frac{\tilde{\beta}_{2k}}{\lambda_k - \lambda_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\tilde{\beta}_{k1}}{\lambda_1 - \lambda_k} & \frac{\tilde{\beta}_{k2}}{\lambda_2 - \lambda_k} & \cdots & 1 \end{pmatrix}.$$

When $E_t = 0$, we have $\tilde{\beta}_{ji} \approx 0$ so that $(\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_k) \approx (\mathbf{x}_1, \dots, \mathbf{x}_k) + E_r \left(\frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_k}{\lambda_k} \right)$.

Such signed graphs have communities with neutral relationships. This is a very special type of signed graphs. Even with dense signed edges added across the communities, there is no rotation of the central lines of the communities and the nodes stay close to the canonical axes in the spectral space. When E_r has a large number of edges, the line pattern is lost due to most of the nodes have inter-community edges and derive from the central line. On the other hand, when $E_r = 0$, we have $(\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_k) \approx (\mathbf{x}_1, \dots, \mathbf{x}_k) \tilde{R} + E_t \left(\frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_k}{\lambda_k} \right)$, which is of the same form with Equation 2.10. Notice E_t needs to satisfy the condition of Theorem 2.1 so it usually has very small magnitude. In this situation, the signed graph has sparse inter-community edges with mixed signs. When E contains both parts, Equation 3.10 has two more terms than Equation 2.10. The first term $(\mathbf{x}_1, \dots, \mathbf{x}_k) \tilde{R}$ is related with the impact from the nodes' direct neighbors and the second term $E_t \left(\frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_k}{\lambda_k} \right)$ is related with the impact from the nodes' 2 steps away neighbours. We notice $\left\| \frac{E_t E_r \mathbf{x}_i}{\lambda_j^2} \right\| \leq \frac{\|E\|^2}{\lambda_j^2}$ so that the third term in Equation 3.10 is much smaller than the second term. The fourth term is also small because $\|E_r\|$ and $\|\tilde{R} - I\|$ are small. They can be omitted except for some individual nodes.

Similar as in Theorem 2.2, we can derive the form of spectral coordinates by ex-

tracting the rows of Equation 3.14. We thus have the spectral coordinate of node u as following:

Proposition 3.2: A graph as $\tilde{A} = A + E$ that satisfies the condition of Corollary 3.1, the spectral coordinate of u can be approximated as

$$\begin{aligned} \boldsymbol{\alpha}_u \approx & x_{iu} \tilde{\mathbf{r}}_i + E_u \left(\frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_k}{\lambda_k} \right) + \\ & (E_t E_r)_u \left(\frac{\mathbf{x}_1}{\lambda_1^2}, \dots, \frac{\mathbf{x}_k}{\lambda_k^2} \right) + (E_r)_u \left(\frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_k}{\lambda_k} \right) (\tilde{R} - I) \end{aligned} \quad (3.15)$$

where x_{iu} is the only non-zero entry in its original spectral coordinate shown in Equation 2.7, $(\cdot)_u$ means the u th row of the corresponding matrix.

If $E_t = 0$, we have $\tilde{\beta}_{ji} \approx 0$ so that $\boldsymbol{\alpha}_u \approx (0, \dots, 0, x_{iu}, 0, \dots, 0) + (E_r)_u \left(\frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_k}{\lambda_k} \right)$. $(E_r)_u \mathbf{x}_i$ is the summation of impact of node u 's neighbors in C_i through E_r . Though we assume that the numbers of positive and negative edges between communities in E_r are equal, an individual node u could still have more positive inter-community edges than negative ones or vice versa. Thus the node u could have large jump from the central line $(0, \dots, 0, x_{iu}, 0, \dots, 0)$. E_r contains both positive and negative edges so that some nodes may derive from the central lines in the opposite direction compared with some other nodes. When we have many nodes like node u , the line pattern is no longer kept. We can see clusters stay on the canonical axes.

When $E_r = 0$ and E_t satisfies the condition in Theorem 2.1, the graph has very sparse inter-community connection. Line pattern is still kept. The only difference of signed graph from unsigned graph is that nodes can scatter on both side of the central line instead of just one side.

When neither E_r nor E_t is zero, there are two extra terms in Equation 3.15 compared with Equation 2.8 in Theorem 2.2. These two terms indicate further derivation of the nodes from the central lines.

Proposition 3.3: The spectral coordinates of \tilde{A} form k approximately orthogonal lines or clusters. Specially, for node $u \in C_i$ with no connection with other communities or with similar numbers of positive and negative edges with the same community, α_u lies close to $\tilde{\mathbf{r}}_i$. Otherwise α_u scatters around $\tilde{\mathbf{r}}_i$.

Proof. The proof of orthogonality of $\tilde{\mathbf{r}}$ is similar to that for Proposition 2.2. Since E_t satisfies the conditions in Theorem 2.1, $\tilde{\beta}_{ji}$ is much smaller than $|\lambda_i - \lambda_j|$. So $\tilde{R}^T \tilde{R} \approx I$ and $\tilde{\mathbf{r}}_i$'s are approximately orthogonal. There are two kinds of nodes that still stay on or close to $\tilde{\mathbf{r}}_i$. The first kind of nodes has no connection outside their own community. The second kind of nodes has equal numbers of positive and negative edges so that $E_u \mathbf{x}_i \approx 0$. These two kinds of nodes may not locate exactly on $\tilde{\mathbf{r}}_i$ due to the small perturbation caused by the third and fourth terms in Equation 3.15. When inter-community edges are dense, we have very few nodes without inter-community connection or with exact equal number of positive and negative inter-community edges. Line pattern may be lost. However, since $\tilde{\mathbf{r}}_i$'s are orthogonal, the communities appear to be k clusters that are orthogonal to each other and thus they are clusterable. \square

3.5 Partite-dominated Signed Graphs

The biggest difference of signed graphs from unsigned graphs is the introduce of negative edges. Negative edges work differently compared with positive edges in forming

the communities. In this section, we specially study partite-dominated signed graphs: signed graphs that have dominated negative inter-community edges and no/few inner-community edges. They are very different from all the graphs we studied before. In order to analyze them, we transform them to the graphs similar to signed graphs with dominated positive inner-community edges but with values relaxed to real numbers. With properly applying Corollary 3.1, we are able to give approximated eigenvectors and spectral coordinates for partite-dominated signed graphs.

The graph contains only negative edges so that its largest eigenvalue in magnitude is negative according to Perron-Frobenius theorem [Stewart and Sun, 1990]. The corresponding eigenvector has all the entries nonnegative. The entries on this eigenvector corresponding to different communities overlap with each other unless they have a significant difference in density. To better analyze the graph, we subtract the first eigenpair from the graph and get a new matrix Q . We find that Q has blocks with positive entries on the diagonal and blocks with positive and negative entries off the diagonal, which is very similar to signed graphs with dominated positive inner-community edges. When Q further satisfies all the conditions in Corollary 3.1, we can apply Corollary 3.1 and get its approximated eigenvectors. By orthogonalizing the first k eigenvectors of Q with the removed eigenvector, we can get the approximated eigenvectors of the original k -partite graph, and show the graph clusterable in the spectral space spanned by the k approximated eigenvectors. In the following, we give a formal definition of the ideal case, k -partite graph and theoretically study its spectral properties.

Definition 3.2: A k -partite graph represents a graph with k communities under two constraints. First, there are no links inside the communities. Second, nodes from different communities are densely connected with the negative. The adjacency matrix A_p can be written in the following form with proper permutation of the nodes:

$$A_p = \begin{pmatrix} \mathbf{0} & B_{12} & \cdots & B_{1k} \\ B_{21} & \mathbf{0} & \cdots & B_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ B_{k1} & B_{k2} & \cdots & \mathbf{0} \end{pmatrix}, \quad (3.16)$$

where B_{ij} is the $n_i \times n_j$ matrix representing relationships between community i and community j . We call A_p as a k -partite matrix.

Let $|\zeta_1| > \cdots > |\zeta_n|$ be the eigenvalues of A_p and \mathbf{z}_i be the corresponding eigenvectors. We similarly define the spectral coordinate $\boldsymbol{\alpha}_u$ as in Equation 3.17.

$$\begin{array}{cccc} \mathbf{z}_1 & & \mathbf{z}_i & & \mathbf{z}_k & & \mathbf{z}_n \\ & & \downarrow & & & & \\ \boldsymbol{\alpha}_u \rightarrow & \left(\begin{array}{c|c|c|c} z_{11} & \cdots & z_{i1} & \cdots & z_{k1} & \cdots & z_{n1} \\ \vdots & & \vdots & & \vdots & & \vdots \\ \hline z_{1u} & \cdots & z_{iu} & \cdots & z_{ku} & \cdots & z_{nu} \\ \hline \vdots & & \vdots & & \vdots & & \vdots \\ z_{1n} & \cdots & z_{in} & \cdots & z_{kn} & \cdots & z_{nn} \end{array} \right) & (3.17) \end{array}$$

By the eigen-decomposition of a matrix, $A_p = \sum_{i=1}^n \zeta_i \mathbf{z}_i \mathbf{z}_i^T$. We remove the effect of \mathbf{z}_1 from A_p by introducing $Q = A_p - \zeta_1 \mathbf{z}_1 \mathbf{z}_1^T$. Write $\mathbf{z}_1^T = (\mathbf{z}_{C_1}, \dots, \mathbf{z}_{C_k})^T$ where \mathbf{z}_{C_i} represents the entries of \mathbf{z}_1 corresponding to community C_i . Q contains a block-wise

diagonal matrix:

$$A = \begin{pmatrix} |\zeta_1| \mathbf{z}_{C_1} \mathbf{z}'_{C_1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & |\zeta_1| \mathbf{z}_{C_2} \mathbf{z}'_{C_2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & |\zeta_1| \mathbf{z}_{C_k} \mathbf{z}'_{C_k} \end{pmatrix}.$$

In $E = Q - A$, the corresponding blocks on the diagonal are zero and the off diagonal blocks $B_{ij} - \zeta_1 \mathbf{z}_{C_i} \mathbf{z}'_{C_j}$ represent the connection between C_i and C_j .

A is the same with Equation 3.1 except for the diagonal entries relaxed to real non-negative numbers. E contains both positive and negative real number entries. Thus Q is similar to the signed graph discussed in Corollary 3.1 except for the relaxation of the entries to real numbers. However, the proof of Corollary 3.1 does not require the matrix must have 0-1 entries. So we can apply Corollary 3.1 to Q if the conditions are satisfied. In order to do so, we need to properly decompose E into E_r and E_t . The first k normalized eigenvectors of A are:

$$(\mathbf{x}_1, \dots, \mathbf{x}_k) = \begin{pmatrix} \frac{\mathbf{z}_{C_1}}{\|\mathbf{z}_{C_1}\|_2} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \frac{\mathbf{z}_{C_2}}{\|\mathbf{z}_{C_2}\|_2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \frac{\mathbf{z}_{C_k}}{\|\mathbf{z}_{C_k}\|_2} \end{pmatrix} \quad (3.18)$$

and their corresponding eigenvalues are $\|\mathbf{z}_{C_1}\|_2^2 |\zeta_1|, \dots, \|\mathbf{z}_{C_k}\|_2^2 |\zeta_1|$. Without loss of generality, we assume $\|\mathbf{z}_{C_1}\|_2^2 \geq \dots \geq \|\mathbf{z}_{C_k}\|_2^2$. We also notice that A has the eigenvalues after k th all equal to zero. It means that the gap between k th and

$(k + 1)$ th eigenvalues is big as $\|\mathbf{z}_{C_k}\|_2^2 |\zeta_1|$.

Let

$$q_{ij} = 1 - \frac{|\zeta_1| \|\mathbf{z}_{C_i}\|_2^2 \|\mathbf{z}_{C_j}\|_2^2}{\mathbf{z}_{C_i}^T B_{ij} \mathbf{z}_{C_j}} \quad (3.19)$$

and

$$E_t = \begin{pmatrix} \mathbf{0} & q_{12}B_{12} & \cdots & q_{1k}B_{1k} \\ q_{21}B_{21} & \mathbf{0} & \cdots & q_{2k}B_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ q_{k1}B_{k1} & q_{k2}B_{k2} & \cdots & \mathbf{0} \end{pmatrix}. \quad (3.20)$$

Then $\mathbf{x}_j^T E_r \mathbf{x}_i = \mathbf{x}_j^T (E - E_t) \mathbf{x}_i = |\zeta_1| - (1 - q_{ji}) \frac{\mathbf{z}_{C_j}^T B_{ji} \mathbf{z}_{C_i}}{\|\mathbf{z}_{C_i}\|_2^2 \|\mathbf{z}_{C_j}\|_2^2} = 0$. If E_t is of small magnitude, we can now apply Corollary 3.1 to get the approximated eigenvectors of Q . With proper transformation, we can derive the approximated eigenvectors for A_p .

We conclude our result as follows:

Proposition 3.4: Define q_{ij} and E_t as in Equation 3.19 and Equation 3.20. If E_t satisfies the conditions of Theorem 2.1, the eigenvectors of A_p can be approximated by the following form:

$$(\mathbf{z}_1, \tilde{\mathbf{z}}_2, \cdots, \tilde{\mathbf{z}}_k) \approx (\mathbf{x}_1, \cdots, \mathbf{x}_k) T K + E \left(0, \frac{\mathbf{x}_1}{\|\mathbf{z}_{C_1}\|_2^2 |\zeta_1|}, \cdots, \frac{\mathbf{x}_{k-1}}{\|\mathbf{z}_{C_{k-1}}\|_2^2 |\zeta_1|} \right) K \quad (3.21)$$

where \mathbf{x}_i 's are defined in Equation 3.18

$$T = \begin{pmatrix} \|\mathbf{z}_{C_1}\|_2 & 1 & \cdots & \frac{\tilde{\beta}_{1(k-1)}|\zeta_1|^{-1}}{\|\mathbf{z}_{C_{k-1}}\|_2^2 - \|\mathbf{z}_{C_1}\|_2^2} \\ \|\mathbf{z}_{C_2}\|_2 & \frac{\tilde{\beta}_{21}|\zeta_1|^{-1}}{\|\mathbf{z}_{C_1}\|_2^2 - \|\mathbf{z}_{C_2}\|_2^2} & \cdots & \frac{\tilde{\beta}_{2(k-1)}|\zeta_1|^{-1}}{\|\mathbf{z}_{C_{k-1}}\|_2^2 - \|\mathbf{z}_{C_1}\|_2^2} \\ \vdots & \vdots & \ddots & \vdots \\ \|\mathbf{z}_{C_{k-1}}\|_2 & \frac{\tilde{\beta}_{(k-1)1}|\zeta_1|^{-1}}{\|\mathbf{z}_{C_1}\|_2^2 - \|\mathbf{z}_{C_{k-1}}\|_2^2} & \cdots & 1 \\ \|\mathbf{z}_{C_k}\|_2 & \frac{\tilde{\beta}_{k1}|\zeta_1|^{-1}}{\|\mathbf{z}_{C_1}\|_2^2 - \|\mathbf{z}_{C_k}\|_2^2} & \cdots & \frac{\tilde{\beta}_{k(k-1)}|\zeta_1|^{-1}}{\|\mathbf{z}_{C_{k-1}}\|_2^2 - \|\mathbf{z}_{C_k}\|_2^2} \end{pmatrix}. \quad (3.22)$$

K is an upper triangle matrix decided by Gram-Schmidt process such that TK is orthogonal. The 2nd to k -th eigenvalues of A_p are of the different sign with the first.

Proof. By the definition of E_t in Equation 3.20, $\mathbf{x}_j E_t \mathbf{x}_i = 0$. When the density of edges between C_i and C_j is closed to the global density of edges, q_{ij} is close to zero. We follow the similar procedure as shown in the proof of Corollary 3.1. $\|E_t\|_2$ is bounded by $|\max_{i,j}\{q_{ij}\}\zeta_1|$. When the communities have similar density of inter-community edges, q_{ij} 's are small and E_t satisfies the conditions of Theorem 2.1. The third and fourth terms in Equation 3.14 can be omitted due to their much smaller magnitude compared with the second term. We then have the following approximated eigenvectors for Q :

$$(\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_k) \approx (\mathbf{x}_1, \dots, \mathbf{x}_k)\tilde{R} + E \left(\frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_k}{\lambda_k} \right)$$

where λ_i 's are the eigenvalues of A .

Since the k dimensional subspace spanned by $\tilde{\mathbf{x}}_i$ for $i = 1, \dots, k$ is approximately the one spanned by \mathbf{x}_i for $i = 1, \dots, k$ and \mathbf{z}_1 is of linear combination of \mathbf{x}_i , the subspace spanned by $(\mathbf{z}_1, \tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_k)$ is almost the same with the subspace spanned

by $(\mathbf{z}_1, \tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_{k-1})$. In order to get $\tilde{\mathbf{z}}_2, \dots, \tilde{\mathbf{z}}_k$, we need to orthogonalize $\tilde{\mathbf{x}}_i$ for $i = 1, \dots, k-1$ with \mathbf{z}_1 .

By removing the last column in \tilde{R} and adding one column before the first column, we define T as Equation 3.22 so that

$$(\mathbf{z}_1, \tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_{k-1}) \approx (\mathbf{x}_1, \dots, \mathbf{x}_k)T + E \left(0, \frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_{k-1}}{\lambda_{k-1}} \right).$$

By Gram-Schmidt process, we can find a matrix K such that TK is an orthogonal matrix. When we apply the same K on $(\mathbf{z}_1, \tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_{k-1})$, we write out the following equation:

$$\begin{aligned} & K^T(\mathbf{z}_1, \tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_{k-1})^T(\mathbf{z}_1, \tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_{k-1})K & (3.23) \\ & \approx K^T T^T (\mathbf{x}_1, \dots, \mathbf{x}_k)^T (\mathbf{x}_1, \dots, \mathbf{x}_k) T K \\ & \quad + (\mathbf{x}_1, \dots, \mathbf{x}_k)^T E \left(0, \frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_{k-1}}{\lambda_{k-1}} \right) K \\ & \quad + K^T \left(0, \frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_{k-1}}{\lambda_{k-1}} \right)^T E^T (\mathbf{x}_1, \dots, \mathbf{x}_k) \\ & \quad + K^T \left(0, \frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_{k-1}}{\lambda_{k-1}} \right)^T E^T E \left(0, \frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_{k-1}}{\lambda_{k-1}} \right) K \end{aligned}$$

Since E_i is of small magnitude, $\mathbf{x}_i^T E \mathbf{x}_j \approx 0$ and thus the second, third and fourth terms in Equation 3.23 are close to 0. So we have:

$$K^T(\mathbf{z}_1, \tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_{k-1})^T(\mathbf{z}_1, \tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_{k-1})K \approx I$$

It means that $(\mathbf{z}_1, \tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_{k-1})K$ has approximately orthogonal columns and it can be used as a good approximation for eigenvectors of A_p . So

$$(\mathbf{z}_1, \tilde{\mathbf{z}}_2, \dots, \tilde{\mathbf{z}}_k) \approx (\mathbf{x}_1, \dots, \mathbf{x}_k)TK + E \left(0, \frac{\mathbf{x}_1}{\lambda_1}, \dots, \frac{\mathbf{x}_{k-1}}{\lambda_{k-1}} \right) K. \quad (3.24)$$

Plug in $\lambda_i = \|\mathbf{z}_{C_i}\|_2^2 |\zeta_1|$, we reach our conclusion for eigenvectors.

Since $\tilde{\mathbf{x}}_i^T A_p \tilde{\mathbf{x}}_i = |\zeta_1| (\|\mathbf{z}_{C_i}\|_2^2 - \|\mathbf{z}_{C_i}\|_2^3) > 0$, the estimated eigenvalues for $\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_{k-1}$ positive. Gram-Schmidt process does not change the subspace. So the approximated eigenvalues for $\tilde{\mathbf{z}}_i$'s ($i \geq 2$) are positive. \square

Proposition 3.4 converts the clusterability problem of a k -partite graph A_p to whether E_t is a perturbation small enough. When q_{ij} 's are close to zeros, $\|E_t\|$ is also close to zero and A_p is thus clusterable. Notice $|\zeta_1|$ is the approximated global density and $\frac{\mathbf{z}_{C_i}^T B_{ij} \mathbf{z}_{C_j}}{\|\mathbf{z}_{C_i}\|_2^2 \|\mathbf{z}_{C_j}\|_2^2}$ is the approximated inter-community connection density between C_i and C_j . Small q_{ij} 's mean that all the inter-community connections have similar density.

Extract the uth row from Equation 3.24 and we have the follow approximated form for spectral coordinates of node u belonging to C_i .

Proposition 3.5: For a k -partite graph A_p , the spectral coordinate of node u in community C_i can be approximated as

$$\alpha_u \approx \frac{z_{1u}}{\|\mathbf{z}_{C_i}\|} \mathbf{t}_i K + E_u \left(0, \frac{\mathbf{x}_1}{\|\mathbf{z}_{C_1}\|_2^2 |\zeta_1|}, \dots, \frac{\mathbf{x}_{k-1}}{\|\mathbf{z}_{C_{k-1}}\|_2^2 |\zeta_1|} \right) K,$$

where \mathbf{t}_i is the i th row of T . $\mathbf{t}_i K$ and $\mathbf{t}_j K$ are approximately orthogonal to each other for $i \neq j$. Two nodes in different communities C_i and C_j stay close to the corresponding vectors so the spectral space has k separable clusters that are almost orthogonal to each other.

Proof. TK is an orthogonal matrix due to Gram-Schmidt process. Its rows are orthogonal to each other, i.e., $(\mathbf{t}_i K)(\mathbf{t}_j K)^T = 0$ for $i \neq j$. $E_u \left(0, \frac{\mathbf{x}_1}{\|\mathbf{z}_{C_1}\|_2 |\zeta_1|}, \dots, \frac{\mathbf{x}_{k-1}}{\|\mathbf{z}_{C_{k-1}}\|_2 |\zeta_1|} \right) K$ represents the further derivation of node u from $\mathbf{t}_i K$.

Now we can conclude that a k -partite graph is expected to form k clusters that are approximately orthogonal to each other. \square

General Signed Graphs

We have discussed three types of signed graph: k -balanced signed graphs, signed graphs with dominated positive inner-community edges and partite-dominated signed graphs. They cover the community structures decided by both positive and negative edges, dominated positive edges only or dominated negative edges only. In general signed graphs, these three types of graphs may exist in the same graph. We may observe that some communities have dense positive inner-community edges and inter-community edges of mixed signs while some other communities do not have many inner-community edges but dense negative inter-community edges. We may also observe that we have partite structures inside some communities. The optimal process of eigen-decomposition automatically handles the situations. It will return large positive eigenvalues related with dense positive edges and large negative eigenvalues related with dense negative edges. The corresponding eigenvectors then naturally capture the related structures. With very slight modification from *AdjCluster*, we develop a graph partition algorithm that works for both signed and unsigned graphs.

3.6 Unified Adjacency Eigenspace based Clustering

In this section, we present a graph partition algorithm, *UniAdjCluster*, which utilizes the clusterable patterns in the spectral space of the adjacency matrix. In signed graphs, communities exhibit orthogonal lines/clusters in the adjacency spectral space. We still project the nodes in spectral space to unit sphere to separate different communities. The difference between *AdjCluster* and *UniAdjCluster* is the selection of eigenvectors. If the communities are mostly decided by negative inter-community edges, the adjacency matrix has large negative eigenvalues. We need to include these eigenvectors so that we will not miss important structures. The modification from *AdjCluster* to *UniAdjCluster* is not very big. We put “Unified” in order to emphasize the nicely consistent patterns shown by adjacency spectral space.

As shown in Algorithm 2, we calculate the eigenvectors with the largest eigenvalues in magnitude(line 1). If those eigenvalues are positive, it usually indicates that the graph has some communities with higher density of positive edges inside them. If the largest one of those eigenvalues is negative while the others are positive, it usually indicates that the graph has k -partite structure. We then project each spectral coordinate α_u to the unit sphere in the k -dimensional subspace by normalizing α_u to its unit length (line 3). With the right value of k , we expect to observe that the nodes from one community to form a cluster on the unit sphere. Hence there will be k well separated clusters on the unit sphere. We apply the k -means clustering algorithm on the unit sphere to produce a graph partition(line 4).

In order to evaluate the quality of the partition, we use three measures here: Davies-

Algorithm 2 *UniAdjCluster*: Unified Adjacency Eigenspace based Clustering

Input: A, K
Output: Clustering results

- 1: Compute $\mathbf{x}_1, \dots, \mathbf{x}_K$ by the eigen-decomposition of A with $|\lambda_1| > \dots > |\lambda_K|$
 - 2: **for** $k = 2, \dots, K$ **do**
 - 3: $\boldsymbol{\alpha}_u = (x_{1u}, \dots, x_{ku})$ and $\bar{\boldsymbol{\alpha}}_u = \frac{\boldsymbol{\alpha}_u}{\|\boldsymbol{\alpha}_u\|}$;
 - 4: Apply k -means algorithm on $\{\bar{\boldsymbol{\alpha}}_u\}_{u=1, \dots, n}$;
 - 5: Compute fitting statistics from k -means algorithm ;
 - 6: **end for**
 - 7: Output partitions under k with the best fitting statistics and corresponding eigen-values
-

Bouldin Index (*DBI*) [Davies and Bouldin, 1979], modularity for signed graphs [Traag and Bruggeman, 2009] and average angle between centroids of the output clusters.

We have introduced *DBI* and average angle between centroids in Chapter 2. Recall the low *DBI* indicates output clusters with low intra-cluster distances and high inter-cluster distances. When the graph contains k clear communities, we expect to have the *DBI* reach its minimum after applying k -means in the k -dimensional spectral space. *DBI* can show a better result for a lower dimension than k when some communities are more clear to separate than the others. In such situation, we usually observe a local minimum for some higher dimension other than the global minimum of *DBI*. As for the average angle, we expect all the angles between centroids of the output clusters are close to 90° since spectral coordinates form quasi-orthogonal lines or k orthogonal clusters in the determined k -dimensional spectral space.

The authors in [Traag and Bruggeman, 2009] extended the definition of modularity to signed graphs by giving penalty to absent positive edges inside communities and negative edge inside communities while reward to positive edges inside communities and absent negative edge inside communities. The high modularity indicates a

stronger community structure that contains more positive inner-community edges and more negative inter-community edges. When the graph contains k clear communities, we expect to have modularity reach maximum.

The algorithm is focused on the first level of community structure in the graph. In order to refine the clustering result, we can apply the algorithm once again on one of the communities to do further partition.

3.7 Empirical Evaluation

3.7.1 Datasets

We generate several synthetic graphs for the purpose of illustrating our theoretical results. Recall in Chapter 2, we generate *Syn-1* with 5 communities of with the number of nodes 200, 180, 170, 150 and 140 respectively and each community is generated separately with a power law degree distribution with the parameter 2.3. We extract the inner-community edges of it and get *Syn-0*. It serves as the inner-community connection for *Syn-3*, *Syn-4* and *Syn-5*. *Syn-3* is generated by adding random positive inter-community edges based on *Syn-0*. Denote ratio of inter-community edges over inner-community edges between C_i and C_j as p_{ij} . *Syn-3* has $p_{ij} = 80\%$. We generate *Syn-4* and *Syn-5* by flipping 20% and 50% of inter-community edges in *Syn-3* to negative. In *Syn-6* to *Syn-9* we keep them with the community size in *Syn-0* but remove all the inner-community edges. *Syn-6*, *Syn-7* and *Syn-8* are generated by randomly adding positive edges between two communities with probability 0.2, 0.4 and 0.6 respectively. We also generate *Syn-9*, *Syn-10* and *Syn-11* by adding inner-community edges to *Syn-6* with probability 0.04, 0.1 and 0.16 respectively.

We apply our algorithm *UniAdjCluster* on both the synthetic data and several real

network datasets and give graph partition results in our evaluation. The real datasets include Correlates of War(*COW*) and *Epinions*.

3.7.2 Separation of Communities in Signed Graphs with Dominated Positive Inner-community Edges

In *Syn-3*, we see that the large number of positive inter-community edges make the communities merge together. When negative edges are introduced, our theoretical result shows that even when the number of inter-community edges is large, if we flip nearly half of the edges to negative, we can still get a clear separation of the original community structure.

In order to show this, we first construct two datasets, *Syn-4* and *Syn-5* by flipping 20% and 50% of the inter-community edges of *Syn-3* negative. We then run *UniAdjCluster* on *Syn-4* and *Syn-5*.

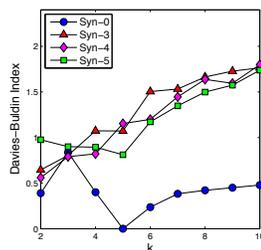


Figure 3.7: Separation of communities for signed graphs with dominated positive inner-community edges

We draw *DBI* against the number of clusters in Figure 3.7 for *Syn-0,3,4,5*. We can see *DBI* for *Syn-0* (blue dots) drops to zero at $k = 5$ and *DBI* for *Syn-5* (green squares) reach the minimum of *DBI* at $k = 5$. These indicate that we can find 5 communities in *Syn-0* and *Syn-5*. *DBI* for *Syn-3* (red triangles) and *DBI* for *Syn-4* (magenta diamond) keep going up when k increases and both of them exceed 1 at

$k = 5$. These indicate that 5 community pattern is lost in *Syn-3* and *Syn-4*. *Syn-4* has much denser positive inter-community edges than negative inter-community edges. E_t is not of small magnitude and communities are merged. *Syn-5* has equal magnitude of positive and negative inter-community edges so that the communities are separable. This verify our theoretical results.

We show the fitting statistics of *UniAdjCluster* in Table 3.1. For *Syn-4*, *DBI* is large and average angle is far from 90° . It indicates the graph does not have a clear 5-community structure. Notice modularity of *Syn-4* is larger than that of *Syn-3*. This is because the modularity rewards negative inter-community edges. For *Syn-5* where nearly half inter-community edges are negative, *DBI* has the minimum value at $k = 5$ and we see the average angle is almost 90° . Modularity of *Syn-5* is much larger than that of *Syn-4*, which indicates a stronger community structure.

3.7.3 Separation of Communities in k -partite Graph

A k -partite graph usually has the largest eigenvalue in magnitude negative and the rest $k - 1$ ones positive. We have shown that a k -partite graph appears to be k orthogonal clusters in the k -dimensional space. We plot *DBI* of *Syn-6*, *Syn-7* and *Syn-8* against the number of clusters. We can see all these graphs with clear 5-partite structure have minimum of *DBI* at $k = 5$. We also notice that with the increase of inter-community connection, *DBI* decreases and it indicates that the denser the inter-community connection is the more clear clusters are in the spectral space.

Figure 3.9 shows the spectral space of *Syn-6* spanned by the first five eigenvectors with largest eigenvalues in magnitude. If we look at the first three dimensions, two of the communities, C_4 and C_5 , are not separable. However, if we look at the fourth

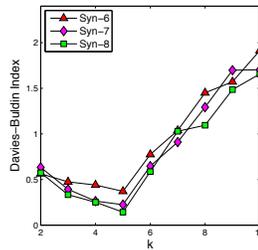


Figure 3.8: k -partite with different density

and the fifth dimensions, we can see they are clearly separated. With moderate inner-community edges added, *Syn-9* does not show big difference compared with *Syn-6*. We then increase the inner-community edges to 0.1 in *Syn-10*. This graph still has a 5-community structure. When we increase the inner-community edges to 0.16 as in *Syn-11*, the graph is totally mixed and 5 community pattern is lost. In Table 3.1, we can see that the k -partite graphs such as *Syn-6*, *Syn-7* and *Syn-8* all have low *DBI* and average angles close to 90° . *Syn-9* is close to a k -partite graph so that the fitting statistics also show it has a clear community structure. *Syn-10* has a much higher *DBI* but not exceed 1. The community structure is still separable. For *Syn-11*, *DBI* exceeds 1 and we can hardly separate nodes for different communities.

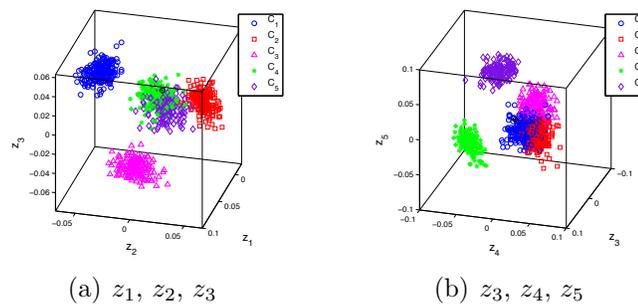


Figure 3.9: Spectral space of *Syn-6* spanned by the first five eigenvectors with largest eigenvalues in magnitude

Table 3.1: Statistics of networks and partition quality of *UniAdjCluster*

Dataset	n	$m(+/-)$	k	DBI	Q	Angle
<i>Syn-0</i>	840	3260/0	5	0	0.8	90°
<i>Syn-3</i>	840	13738/0	5	1.07	0.05	68.7°
<i>Syn-4</i>	840	11642/2096	5	1.15	0.09	73.0°
<i>Syn-5</i>	840	8708/5030	5	0.82	0.69	89.7°
<i>Syn-6</i>	840	0/59242	5	0.37	0.2	88.8°
<i>Syn-7</i>	840	0/117892	5	0.22	0.2	89.6°
<i>Syn-8</i>	840	0/177303	5	0.14	0.2	89.8°
<i>Syn-9</i>	840	0/62134	5	0.48	0.15	87.6°
<i>Syn-10</i>	840	0/66398	5	0.96	0.09	81.2°
<i>Syn-11</i>	840	0/70601	5	1.3	0.02	74.3°
<i>COW</i>	159	1093/155	5	0.97	0.58	97.5°
<i>Epinions</i>	2027	0/2348	3	0.74	0.14	93.2°

Community Partition

Table 3.1 shows the quality of our graph partition algorithm *UniAdjCluster* on the synthetic datasets and several real world dataset as described above. “ k ” is the number of communities, “ DBI ” is the Davies-Bouldin Index, “Angle” is the average angle between centroids, and “ Q ” is the signed modularity. The synthetic networks with clear community settings all return DBI lower than 1 and average angle close to 90°. The results verify our theoretical findings.

The analysis on real graphs returns interesting results. *COW* is a graph with alliance and dispute among 159 countries from 1993 to 2001. We run *UniAdjCluster* and find 5 communities. We plot the world map based on the output in Figure 3.10¹. The power blocs can be identified as follows: Latin America, the West (including USA, Canada, Australia, and West European countries), Muslim World, West Africa, and China and the former Soviet Union. Latin America countries shared similar culture and they had alliances inside. The West countries not only had alliances inside

¹The figure is generated in Region Map Generator: <http://www.cciyy.com/>

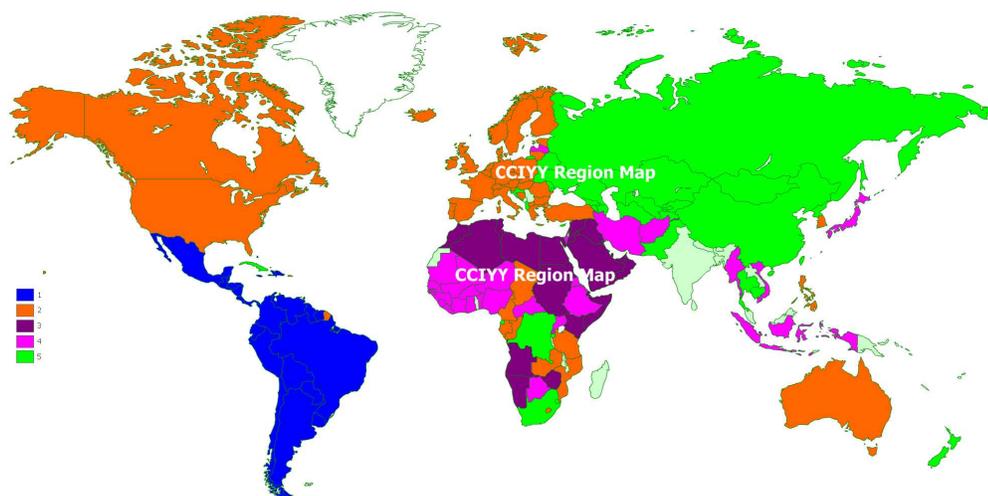


Figure 3.10: Map of the 5 communities in *COW*

but had many disputes with the other communities. China and the former Soviet Union did not have alliances after the cold war, but they shared common enemies so that they are grouped in one community. This is a complicated signed graph mixed with multiple types from the three we have discussed. Our results match the configuration depicted in [Huntington, 1997] with a few notable exceptions. The West African power bloc is absent in the configuration of [Huntington, 1997]. Some other noteworthy differences are South Korea is grouped with the West and South Africa is grouped with China and the Soviet Union.

Epinions is a graph with two-way distrust relationship between any two users in the consumer's review website Epinions.com. We run *UniAdjCluster* and find 3 communities. Its first three eigenvalues are negative, positive and positive. It shows that the negative inter-community can also group people into different communities.

Currently there are few algorithms on graphs with both positive and negative edges. In [Kunegis et al., 2010], the authors extend NormalCut to signed graph by modifying the definition of degree matrix (“SNCut”). In [Traag and Bruggeman, 2009], the authors offer an extended definition of modularity and run various clustering methods such as simulate annealing to maximize the modularity. Their algorithm is called resolution-limit-free community detection algorithm (“RLF-CD”). We run the algorithms on *Syn-4* and *Syn-5*. Our algorithm produce higher accuracy as shown in Table 3.2. We also vary p_{ij} to 50% and filp 20% or 50% of inter-community edges to negative. The results are similar.

We also apply the same two algorithms on the synthetic k -partite graph. With moderate inner-community connection added as in *Syn-9*, we can get a 100% in accuracy from all the algorithms. On *Syn-10*, we notice that *UniAdjCluster* achieves a much higher accuracy than the other two methods. It is because that these two methods mix two of the communities together though the graph still has a 5-community structure. *UniAdjCluster* successfully detects all 5 communities. The results also show that adjacency spectral space is more stable under perturbation than the normal spectral space.

Table 3.2: Accuracy (%) of clustering results

Dataset	UniAdjCluster	SNCut	RLF-CD
<i>Syn-4</i>	68.57%	57.38%	37.5%
<i>Syn-5</i>	88.69%	82.26%	74.44%
<i>Syn-9</i>	100%	100%	100%
<i>Syn-10</i>	91.43%	71.79%	69.52%

3.8 Summary

We conduct theoretical studies based on graph perturbation to examine spectral patterns of signed graphs, which extend our preliminary research in Chapter 2. With the assumption that positive edges mainly exist inside communities and negative edges mainly exist outside communities, each signed graph has its structure decided by balanced positive inner-community edges and negative inter-community edges, dominate positive inner-community edge or dominate negative inter-community edges. We specially discuss the spectral properties of k -balanced signed graphs, signed graphs with dominated positive inner-community edges, partite-dominated signed graphs and their clusterability in the spectral space. Other clusterable signed graphs are usually within a small perturbation from them or their combination. To our best knowledge, these are the first reported findings on showing separability of communities in the spectral space of the signed adjacency matrix. Base on theoretical findings, we develop a unified partition method and verify our theoretical results with synthetic data and real world networks. Part of this chapter was published in Advances in 15th Pacific-Asia Knowledge Discovery and Data Mining Conference[Wu et al., 2011] and International Journal of Social Network Mining[Wu et al., 2012].

CHAPTER 4: WEAK ANOMALIES AND SIGNAL DETECTION

In large social networks, there are some small subgraphs that indicate interesting interaction patterns for a subset of users or capture fraudulent behaviors among attackers. These small subgraphs are usually of much smaller size compared with the whole graph. However, it is often very important for data analysts or network owners to capture them. In this chapter, we explore the application of spectral analysis on the detection of such small subgraphs.

4.1 Introduction

Spectral analysis has been shown as a very effective way to analyze network topology as we see. It provides a global view of the graph and gives a very different statistical framework from traditional Euclidean vector space. The leading eigenvectors of a graph corresponding to the largest eigenvalues contain most global topological information of the graph in the spectral space. We call them principal eigenvectors and call the remaining ones with small eigenvalues as minor eigenvectors. As we explored in the last two chapters, the principal eigenvectors capture community structures in social networks. Naturally, a question will be raised: whether the eigenvectors also capture the smaller signals? The authors in [Miller et al., 2010] showed that some eigenvectors with smaller eigenvalues may capture some signals. But they did not explained why and how the eigenvectors capture the signals. We continue the direc-

tion in their work, i.e., exploring the minor eigenvectors to detect subtle anomalies from a background graph. Such subtle anomaly is often embedded in one particular community. Hence it does not change much the principal eigenvectors. We first demonstrate that when there are such subtle anomalies, there exist some minor eigenvectors with extreme values on some entries corresponding to the anomalies. Under the assumption of the *Erdos-Renyi* random graph model, we derive the formula to show the difference between signal entries and background entries on both principal eigenvectors and minor eigenvectors and give conditions when such difference is large enough on some eigenvectors for us to detect the signal. We then extend our theoretical studies to the general case where multiple anomalies are embedded in a general background graph. When we calculate the eigenvectors in the decreasing order of eigenvalues, we find that the principal eigenvectors and minor eigenvectors of different communities show up in a mixed order. Though the signal is detectable, the particular eigenvector is arranged after a large number of eigenvectors.

We develop an algorithm that uses the kurtosis (rather than the L_1 -norm of eigenvector in [Miller et al., 2010]) to filter out those eigenvectors that capture the signals. The kurtosis metric naturally captures the extremeness in the distribution of eigenvector entries that is caused by embedded signals. We remove the assumption in [Miller et al., 2010] that the background graph is generated using known parameters of a specific model. Our theoretical analysis and empirical evaluations on both synthetic data and real social networks show effectiveness of our approach to detecting subtle signals.

4.2 Preliminary

Let $G = G(V, E)$ be a graph with node set V and edge set E . $n = |V|$ is the number of nodes. A symbol with hat denotes the estimation of the real value. We first revisit some important concepts and theorems in spectral graphs that we will use throughout the paper.

Definition 4.1: Given an $n \times n$ symmetric matrix M and a non-zero n -dimensional vector \mathbf{v} , the rayleigh quotient is defined as

$$R(M, \mathbf{v}) = \frac{\mathbf{v}^T M \mathbf{v}}{\mathbf{v}^T \mathbf{v}}.$$

Rayleigh quotient is closely related with eigenvalues and eigenvectors. When \mathbf{v} is an eigenvector, $R(M, \mathbf{v})$ is the corresponding eigenvalue.

Theorem 4.1: (**Min-max Theorem**) Given an $n \times n$ symmetric matrix M and its eigenvalues $\zeta_1 \geq \zeta_2 \geq \dots \geq \zeta_s \dots \geq \zeta_n$, for all $(n - s + 1)$ -dimensional subspace \mathcal{F} , we have:

$$\zeta_s = \min_{\dim \mathcal{F} = n - s + 1} \max_{\mathbf{v} \in \mathcal{F}, \mathbf{v} \neq \mathbf{0}} R(M, \mathbf{v}), \quad (4.1)$$

and for all s -dimensional subspace \mathcal{F}' , we have:

$$\zeta_s = \max_{\dim \mathcal{F}' = s} \min_{\mathbf{v} \in \mathcal{F}', \mathbf{v} \neq \mathbf{0}} R(M, \mathbf{v}). \quad (4.2)$$

Specially, when \mathcal{F} is the whole n -dimensional space, $\zeta_1 = \max_{\mathbf{v} \in \mathcal{F}} R(M, \mathbf{v})$.

Let $G(n, p)$ denote the random graph generated by the *Erdos-Renyi (ER)* random graph model [Erdős and Rényi, 1959], where n is the number of edges and p is the probability that an edge is included in the graph. Notice when p is not very small,

i.e., $p \geq \frac{2 \log n}{n}$, the random graph $G(n, p)$ is usually connected with high probability [Mitra, 2009].

Theorem 4.2: [Fredi and Komls, 1981] For a *ER* random graph $G(n, p)$, the approximation of the first eigenvalue ζ_1 is:

$$\hat{\zeta}_1 = np. \quad (4.3)$$

The approximation of the first eigenvector \mathbf{w}_1 is:

$$\hat{\mathbf{w}}_1 = \left(\frac{1}{\sqrt{n}}, \dots, \frac{1}{\sqrt{n}} \right)^T. \quad (4.4)$$

The second eigenvalue ζ_2 has an upper bound:

$$\zeta_2 \leq 2\sqrt{np(1-p)} + O\left(n^{\frac{1}{3}} \log n\right). \quad (4.5)$$

Notice when np is large, there is a large gap between ζ_1 and ζ_2 . In [Fredi and Komls, 1981], the authors proved that the first eigenvalue follows an asymptotical normal distribution. For the first eigenvector, the authors in [Mitra, 2009] showed that $\left| \mathbf{w}_1(i) - \frac{1}{\sqrt{n}} \right| \leq c \frac{1}{\sqrt{n}} \frac{\log n}{\log(np)} \sqrt{\frac{\log n}{np}}$ when $p \geq \frac{\log^6 n}{n}$.

Denote $\xi_{\mathbf{w}_1}$ is the error term of the approximation $\hat{\mathbf{w}}_1$, i.e., $\xi_{\mathbf{w}_1} = \mathbf{w}_1 - \hat{\mathbf{w}}_1$. $\|\xi_{\mathbf{w}_1}\|_2$ is about $\frac{1}{\sqrt{np}}$ [Fredi and Komls, 1981], which is much smaller than 1 when np is large.

We can show that the variance $\sigma^2(\xi_{\mathbf{w}_1})$ of entries for $\xi_{\mathbf{w}_1}$ is also very small. $\sigma^2(\xi_{\mathbf{w}_1}) = \sigma^2(\mathbf{w}_1) = \frac{\sum_{i=1}^n \mathbf{w}_1(i)^2}{n} - \left(\frac{\sum_{i=1}^n \mathbf{w}_1(i)}{n} \right)^2$. The first term is $\frac{1}{n}$ due to the normalization of \mathbf{w}_1 . The second term is $\frac{\sqrt{n} + \sum_{i=1}^n \xi_{\mathbf{w}_1}(i)}{n \|\hat{\mathbf{w}}_1 + \xi_{\mathbf{w}_1}\|_2}$. Because $\|\xi_{\mathbf{w}_1}\|_2$ is small, the second term is close to $\frac{1}{n}$. Hence $\sigma^2(\xi_{\mathbf{w}_1})$ is a very small number.

All other eigenvectors represent the noise of the random graph. The mean value

of \mathbf{w}_i ($i \geq 2$) is approximately zero. This is due to their orthogonality with \mathbf{w}_1 . For $i \geq 2$, $\mathbf{w}_i^T \mathbf{w}_1 \approx \frac{1}{\sqrt{n}} \sum_{j=1}^n \mathbf{w}_i(j) \approx 0$. $\sigma^2(\xi_{\mathbf{w}_i}) \approx \frac{\sum_{i=1}^n \mathbf{w}_1(i)^2}{n} = \frac{1}{n}$. We can see that $\sigma^2(\xi_{\mathbf{w}_i})$ ($i \geq 2$) is much larger than $\sigma^2(\xi_{\mathbf{w}_1})$.

4.3 Embedded Signal Detection

We focus on identifying small and subtle signals or anomalies that are not immediately revealed in a graph's principal eigenvectors. We consider the problem of detecting a subgraph embedded in a background as one of detecting a signal from the background. The small size of the signal usually makes it difficult to be detected.

Table 4.1: Notations

	Observed	Background	Signal
Graph	G_A	G_B	G_S
Adj matrix	A	B	S
i -th eigenvalue	λ_i	μ_i	ν_i
i -th eigenvector	\mathbf{x}_i	\mathbf{y}_i	\mathbf{z}_i
No. of nodes	n	n	k

Let $G_B = (V, E)$ denote the background graph of n nodes, i.e., a graph in which no anomaly exists. We define the subgraph $G_S = (V_S, E_S)$ with k nodes as the signal. The observed graph $G_A = (V, E \cup E_S)$ is composed with the background graph G_B and the embedded signal G_S . Let A and B be the adjacency matrix for G_A and G_B and S be their difference: $S = A - B$. Notice that we write S an $n \times n$ symmetric matrix that only has values in the block consisting of the first k rows and the first k columns. Let λ_i be the i -th largest eigenvalue of the observed graph G_A and \mathbf{x}_i the corresponding eigenvector. Similarly let μ_i be the i -th largest eigenvalue and \mathbf{y}_i the corresponding eigenvector of the background graph G_B , and let ν_i be the i -th largest eigenvalue and \mathbf{z}_i the corresponding eigenvector of the signal graph G_S .

Table 4.1 summarizes our notations. We choose to use the adjacency matrix of a graph rather than its variants like Laplacian matrix or modularity matrix because the adjacency matrix is a simple and direct way to present the relations in a network and all information contained in other variants is inherently captured in the adjacency matrix. The adjacency matrices of large social networks are often sparse and many efficient and specialized spectral decomposition algorithms have been developed.

Graphs often contain community structures. For a graph with p communities that are sparsely connected, the first p eigenvectors corresponding to the largest p eigenvalues contain most information of the graph's global community structure. We call them the principal eigenvectors. The remaining eigenvectors corresponding to small eigenvalues are called minor eigenvectors. In this section we conduct theoretical studies on how both principal and minor eigenvectors are changed when signals are added. Specifically, we demonstrate that when there are subtle signals, certain minor eigenvectors have extreme values on some entries and those entries correspond to the signals. We first focus on a simple scenario that both the signal and the background graph follow the simple *ER* model in Section 4.3.1. In this case, only the first eigenvector is the principal one and all remaining eigenvectors are minor ones. In Section 4.3.2, we extend to the general scenario where multiple signals are embedded in a general graph with community structures.

4.3.1 Signal and Background Following ER Model

Assume both the background and the signal follow the *ER* random graph model: $G_B = G(n, p_b)$ and $G_S = G(k, p_s)$. Without loss of generality, we assume the signal is on the first k nodes. According to Equations 4.3 and 4.4, we have approximated

eigenvalues and eigenvectors for B :

$$\widehat{\mu}_1 = np_b, \quad (4.6)$$

$$\widehat{\mathbf{y}}_1 = \left(\frac{1}{\sqrt{n}}, \dots, \frac{1}{\sqrt{n}} \right)^T; \quad (4.7)$$

and for S :

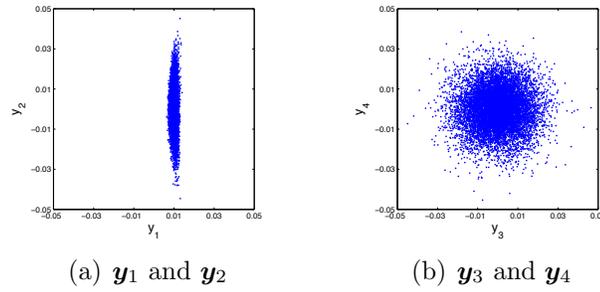
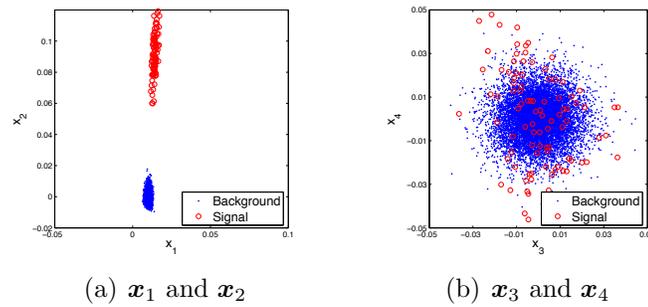
$$\widehat{\nu}_1 = kp_s, \quad (4.8)$$

$$\widehat{\mathbf{z}}_1 = \left(\frac{1}{\sqrt{k}}, \dots, \frac{1}{\sqrt{k}}, 0, \dots, 0 \right)^T. \quad (4.9)$$

Illustrating Example

Throughout this section, we use a synthetic network $G_B = G(10000, 0.01)$ as the background. We add a signal G_S (with varied k and p_s values) to the background. For our generated random graph from $G_B = G(10000, 0.01)$, we have $\mu_1 = 100.99$ and $\mu_2 = 19.98$. The next few eigenvalues are close to μ_2 . We observe that there is a large eigen-gap between the first and the second eigenvectors whereas the eigen-gaps for other adjacent eigenvalues are very small. Figure 4.1 shows the scatter-plot of the first four eigenvectors of the background graph $G(10000, 0.01)$. We can see that the entries of the first eigenvector are located within a narrow range near $\frac{1}{\sqrt{10000}} = 0.01$. The entries of other eigenvectors randomly scatter near zero with a much larger variance than the first eigenvector. This phenomena clearly matches our theoretical justification in Section 4.2.

Figure 4.2 shows the scatter-plot of the first four eigenvectors \mathbf{x}_i ($i = 1, \dots, 4$) after we add a signal with $G_s = (100, 0.3)$. We can see that the signal is not clearly

Figure 4.1: Scatter-plot of the first four eigenvectors of $G(10000, 0.01)$ Figure 4.2: *ER* signal $G(100, 0.3)$

separable by the first eigenvector where signal entries are mixed with the background. However, the signal is well separated from the background by the second eigenvector where it has a large gap between the signal and the background. Figure 4.2(b) shows the signal entries are completely mixed with the background in the scatter-plot of x_3 and x_4 . This is because the signal information is already captured in the previous minor eigenvector x_2 .

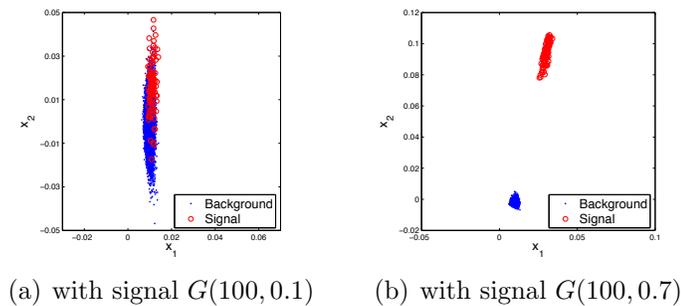
Figure 4.3: Scatter-plot of x_1 and x_2

Figure 4.3(a) shows the scatter-plot of \mathbf{x}_1 and \mathbf{x}_2 for a more subtle signal $G(100, 0.1)$. We can observe that the signal entries are mixed with the background in both the principal eigenvector \mathbf{x}_1 and the minor eigenvector \mathbf{x}_2 . We can not separate the signal from the background due to its subtleness. On the contrary, Figure 4.3(b) shows the scatter-plot for a much stronger signal $G(100, 0.7)$. We can see that the signal entries can be well separated from the background by either the first or the second eigenvector.

Theoretical Analysis

The entries of B and S , b_{ij} and s_{ij} , are binomially distributed random variables with success probability p_b and p_s respectively. We have expectations $E(b_{ij}) = p_b$, $E(s_{ij}) = p_s$ and variances $\sigma^2(b_{ij}) = p_b(1 - p_b)$, $\sigma^2(s_{ij}) = p_s(1 - p_s)$. By the center limit theorem, we can have an accurate estimation of the multiplication of vectors and matrices with much smaller variances than $\sigma^2(b_{ij})$ or $\sigma^2(s_{ij})$. Our goal is to derive the approximated form of eigenvectors of the observed A from eigenvectors of the background graph B and the signal S . Next we present our theoretical results.

Result 4.1: Let $A = B + S$ where A , B and S are the adjacency matrices for G_A , G_B and G_S . $G_B = G(n, p_b)$ and $G_S = G(k, p_s)$. When $k = o(n)$ and $kp_s < \frac{np_b}{1+2\sqrt{\frac{k}{n}}}$, the first eigenvector of A can be expressed as:

$$\mathbf{x}_1 \approx \mathbf{y}_1 + \frac{S\mathbf{y}_1}{\mu_1} \quad (4.10)$$

where the approximation error is $O(\sqrt{\frac{k}{n}})$.

Proof. Let $U = (\mathbf{y}_2, \dots, \mathbf{y}_n)$ and $V = \text{diag}(\mu_2, \dots, \mu_n)$. If we apply Theorem V.2.8 in

[Stewart and Sun, 1990], we have:

$$\mathbf{x}_1 = \mathbf{y}_1 + U(\mu_1 I - V)^{-1} U^T S \mathbf{y}_1, \quad (4.11)$$

Equation 4.11 is quite complex because it involves all the eigenpairs of B . We can further simplify it as:

$$\mathbf{x}_1 = \mathbf{y}_1 + \frac{S \mathbf{y}_1}{\mu_1}, \quad (4.12)$$

In order to apply Theorem V.2.8 in [Stewart and Sun, 1990], the following two conditions need to be satisfied¹:

1. $\delta = |\mu_1 - \mu_2| - \|\mathbf{y}_1^T S \mathbf{y}_1\|_2 - \|U^T S U\|_2 > 0$;
2. $\gamma = \|U^T S \mathbf{y}_1\|_2 < \frac{1}{2} \delta$.

By Equation 4.7, $\|\mathbf{y}_1^T S \mathbf{y}_1\|_2 \approx \sum_{i=1}^k \sum_{j=1}^k \frac{s_{ij}}{n} \approx \frac{k(k-1)p_s}{n}$. U is an $n \times (n-1)$ matrix whose singular value is 1. So $\delta > |\mu_1 - \mu_2| - \|\mathbf{y}_1^T S \mathbf{y}_1\|_2 - \|U^T\|_2 \|S\|_2 \|U\|_2 \approx \mu_1 - \mu_2 - (1 + \frac{k}{n})kp_s > 0$. To satisfy Condition 1, we require:

$$kp_s < \frac{\mu_1 - \mu_2}{1 + \frac{k}{n}}. \quad (4.13)$$

For Condition 2,

$$\|U^T S \mathbf{y}_1\|_2 \leq \|U^T\|_2 \|S \mathbf{y}_1\|_2 \approx \sqrt{\sum_{i=1}^k \left(\sum_{j=1}^k \frac{s_{ij}}{\sqrt{n}} \right)^2} \approx \sqrt{\frac{k}{n}} (k-1)p_s \quad (4.14)$$

¹The 2-norm is the induced matrix norm where $\|A\|_2$ is the largest singular value of A . This norm is sub-multiplicative norm [Stewart and Sun, 1990].

So to satisfy Condition 2, we require

$$kp_s < \frac{\mu_1 - \mu_2}{1 + 2\sqrt{\frac{k}{n} + \frac{k}{n}}} \quad (4.15)$$

Combining Equations 4.13 and 4.15, we have Inequality 4.10 to be held when $kp_s < \frac{\mu_1 - \mu_2}{1 + 2\sqrt{\frac{k}{n} + \frac{k}{n}}}$. By Theorem 4.2, we have $\mu_1 \approx np_b$ and $\mu_2 \leq 2\sqrt{np_b(1 - p_b)}$. So $\mu_1 \gg \mu_2$ when np_b is large. We also assume $k = o(n)$. So the condition can be further simplified to

$$kp_s < \frac{np_b}{1 + 2\sqrt{\frac{k}{n}}} \quad (4.16)$$

At last, we want to discuss about the approximation error. It is divided into two parts. The first part ϵ_1 is related with the higher order terms which are neglected the approximation in Theorem V.2.8 and $\|\epsilon_1\|_2 \sim O(\frac{k}{n})$. The second part $\epsilon_2 = \|U(\mu_1 I - V)^{-1}U^T S\mathbf{y}_1 - \frac{S\mathbf{y}_1}{\mu_1}\|_2 \leq \|U\|_2 \|(\mu_1 I - V)^{-1}\|_2 \|U^T\|_2 \|S\mathbf{y}_1\|_2 + \frac{\|S\mathbf{y}_1\|_2}{\mu_1} \approx \frac{\sqrt{k}kp_s(2\mu_1 - \mu_2)}{\sqrt{n}\mu_1(\mu_1 - \mu_2)}$. By Equation 4.16, $\|\epsilon_2\|_2 \sim O(\sqrt{\frac{k}{n}})$. Combine two parts of error together, the total approximation error is about $O(\sqrt{\frac{k}{n}})$. \square

Denote $\mathbf{x}_1(i)$ as the i -th entry of \mathbf{x}_1 . The first k entries correspond to the embedded signal. We expect they have different values than the rest of entries. Plugging Equations 4.6 and 4.7 into Equation 4.10, we derive the estimation for both signal entries and the background entries of \mathbf{x}_1 .

Corollary 4.1: The entries of the first eigenvector of A can be expressed as:

$$\widehat{\mathbf{x}}_1(i) = \begin{cases} \frac{1+a}{\sqrt{n}} & \text{when } i \leq k \\ \frac{1}{\sqrt{n}} & \text{when } i > k \end{cases} \quad (4.17)$$

where $a = \frac{kp_s}{np_b}$.

Denote $\Delta\hat{\boldsymbol{x}}_1$ the difference between the first k entries with the rest of $\hat{\boldsymbol{x}}_1$. We have $\Delta\hat{\boldsymbol{x}}_1 = \frac{a}{\sqrt{n}}$. When the signal is strong, i.e., $\frac{kp_s}{np_b}$ is large, the difference between the first k entries with the rest of \boldsymbol{x}_1 is sufficient to separate the signal from the background. However, when the signal is weak, $\Delta\hat{\boldsymbol{x}}_1$ is small and the first k entries of \boldsymbol{x}_1 tend to be mixed with the rest entries of \boldsymbol{x}_1 due to the random noise caused by the *ER* model.

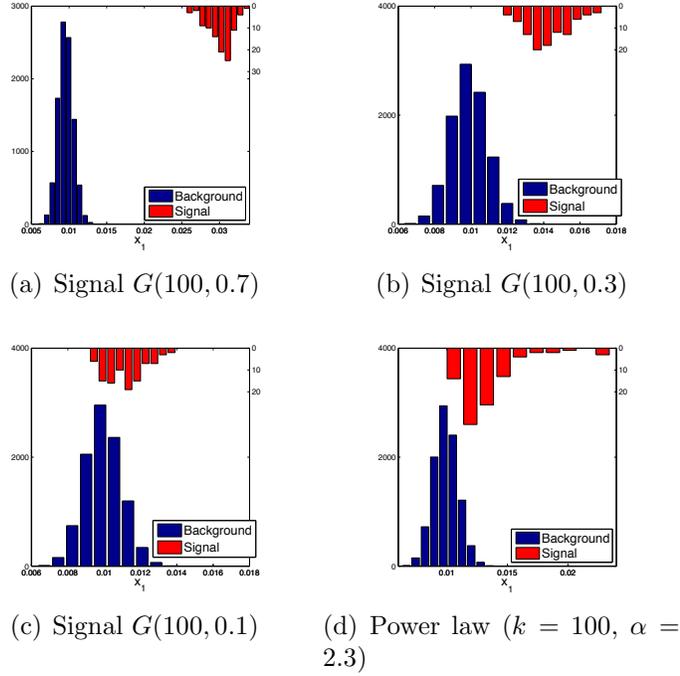


Figure 4.4: Histogram of \boldsymbol{x}_1 to show the difference between the signal entries and background entries

Figure 4.4 shows the histograms of \boldsymbol{x}_1 when the background is embedded with different signals. To make it clear, we show the histogram of the values of background entries in \boldsymbol{x}_1 in the left-down corner (with blue color) and that of the signal entries in the right-up corner (with red color). Each histogram reflects the distribution of

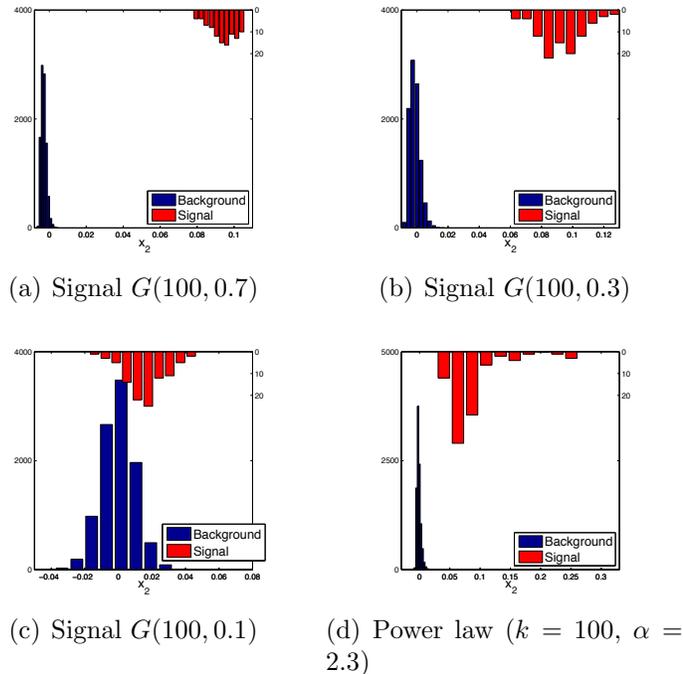


Figure 4.5: Histogram of \mathbf{x}_2 to show the difference between the signal entries and background entries

either signal entries or background entries. We can see a large gap between the signal and the background entries for strong signals such as $G(100, 0.7)$ (shown in Figure 4.4(a)). But for subtle signals such as $G(100, 0.3)$ (shown in Figure 4.4(b)) and $G(100, 0.1)$ (shown in Figure 4.4(b)), there is a large overlap between the signal and the background entries. $\Delta\hat{\mathbf{x}}_1$ is too small for the signal entries to be separated from the background ones. Hence the usage of \mathbf{x}_1 to detect a small signal is very limited.

A natural thought is to check whether the second eigenvector can be used to separate the signal from the background. However, when we derive Result 4.1, we need a large gap in neighboring eigenvalues for the background graph. ER graph does not have a gap large enough between the second and third eigenvalues so we cannot

derive the estimation of the second eigenvector in a similar strategy. Our idea is to construct a new vector, $\mathbf{v} = \frac{\mathbf{z}_1 - (\mathbf{z}_1^T \mathbf{x}_1) \mathbf{x}_1}{\|\mathbf{z}_1 - (\mathbf{z}_1^T \mathbf{x}_1) \mathbf{x}_1\|_2}$, which is orthogonal to \mathbf{x}_1 and is expected to capture most information of the signal. In the following we give the conditions when \mathbf{v} is a good approximation of \mathbf{x}_2 .

We construct an interval and show all the eigenvectors having eigenvalues outside this interval are quite irrelevant with \mathbf{v} . When \mathbf{x}_2 is the only eigenvalue left in this interval, we conclude that \mathbf{v} is a good approximation of \mathbf{x}_2 .

Result 4.2: Let $\mathbf{v} = \frac{\mathbf{z}_1 - (\mathbf{z}_1^T \mathbf{x}_1) \mathbf{x}_1}{\|\mathbf{z}_1 - (\mathbf{z}_1^T \mathbf{x}_1) \mathbf{x}_1\|_2}$. When $k = o(n)$ and $\lambda_3 < kp_s \left(1 - \sqrt{\frac{k}{c^2(n-k)}}\right) < \lambda_2 < kp_s \left(1 + \sqrt{\frac{k}{c^2(n-k)}}\right)$, we have $\mathbf{v}^T \mathbf{x}_2 > \sqrt{1 - c^2}$.

Proof. Let $\mathbf{v} = \frac{\mathbf{z}_1 - (\mathbf{z}_1^T \mathbf{x}_1) \mathbf{x}_1}{\|\mathbf{z}_1 - (\mathbf{z}_1^T \mathbf{x}_1) \mathbf{x}_1\|_2}$. Since $\mathbf{v}^T \mathbf{x}_1 = 0$, we have the decomposition $\mathbf{v} = \sum_{j=2}^n c_j \mathbf{x}_j$ where $\sum_{j=2}^n c_j^2 = 1$. Plug in and we have $\|A\mathbf{v} - q\mathbf{v}\|_2^2 = \sum_{i=2}^n c_i^2 (\lambda_i - q)^2$. So for an arbitrary set of c_i 's we have the upper bound as following:

$$\sum_i c_i^2 \leq \frac{\|A\mathbf{v} - q\mathbf{v}\|_2^2}{\min(\lambda_i - q)^2}.$$

Plug in $A = B + S$ and \mathbf{v} , we want to estimate the value of $\|A\mathbf{v} - q\mathbf{v}\|_2^2$. $\|A\mathbf{v} - q\mathbf{v}\|_2 = \|(B\mathbf{z}_1 + (q - \lambda_1)(\mathbf{z}_1^T \mathbf{x}_1) \mathbf{x}_1) + (S\mathbf{z}_1 - q\mathbf{z}_1)\|_2 / \|\mathbf{z}_1 - (\mathbf{z}_1^T \mathbf{x}_1) \mathbf{x}_1\|_2$. We let $q = \nu_1 \approx kp_s$ so that $S\mathbf{z}_1 - q\mathbf{z}_1 = 0$. By Equation 4.17 and $k = o(n)$, $\lambda_1 = R(A, \mathbf{x}_1) \approx \frac{n^2 p_b + k^2 p_s}{n} \approx np_b$. By Equations 4.9 and 4.10, $\mathbf{z}_1^T \mathbf{x}_1 \approx \sqrt{\frac{k}{n}}$. Thus we have $B\mathbf{z}_1 \approx \lambda_1 (\mathbf{z}_1^T \mathbf{x}_1) \mathbf{x}_1$ and $\|\mathbf{z}_1 - (\mathbf{z}_1^T \mathbf{x}_1) \mathbf{x}_1\|_2 = \sqrt{1 - (\mathbf{z}_1^T \mathbf{x}_1)^2} \approx \sqrt{1 - \frac{k}{n}}$. Finally, we have

$$\|A\mathbf{v} - q\mathbf{v}\|_2 \approx \frac{\|q(\mathbf{z}_1^T \mathbf{x}_1) \mathbf{x}_1\|_2}{\|\mathbf{z}_1 - (\mathbf{z}_1^T \mathbf{x}_1) \mathbf{x}_1\|_2} \approx \nu_1 \sqrt{\frac{k}{n-k}}.$$

For $\lambda_i \notin \left(\nu_1 - \sqrt{\frac{k}{c^2(n-k)}} \nu_1, \nu_1 + \sqrt{\frac{k}{c^2(n-k)}} \nu_1\right)$, the sum of corresponding c_i^2 's is

bounded by c^2 . So when $\lambda_3 < \nu_1 - \sqrt{\frac{k}{c^2(n-k)}}\nu_1 < \lambda_2 < \nu_1 + \sqrt{\frac{k}{c^2(n-k)}}\nu_1$, we have

$$\sum_{i=3}^n c_i^2 \leq \frac{\|A\mathbf{v} - q\mathbf{v}\|_2^2}{(\lambda_3 - q)^2} < c^2.$$

So $c_2 = \sqrt{1 - \sum_{i=3}^n c_i^2} > \sqrt{1 - c^2}$. \square

Result 4.2 provides an approximation of \mathbf{x}_2 . We can derive the following property: $(\mathbf{x}_1^T \mathbf{z}_1)^2 + (\mathbf{v}^T \mathbf{z}_1)^2 = 1$, $\mathbf{x}_i^T \mathbf{z}_1 = 0$ for $i \geq 3$. The signal has no significant information shown on \mathbf{x}_i when $i \geq 3$. We thus regard the eigenvectors and eigenvalues other than the first two do not change. λ_3 thus shares the same upper bound as μ_2 , i.e., $\lambda_3 \leq 2\sqrt{np_b(1-p_b)}$. So the lower bound for the signal to be captured by one of the first eigenvectors can be further simplified as $2\sqrt{np_b(1-p_b)} < kp_s \left(1 - \sqrt{\frac{k}{c^2(n-k)}}\right)$.

In the following, we also provide the estimation for both signal entries and background entries of \mathbf{x}_2 :

Corollary 4.2: The entries of the second eigenvector of A can be expressed as:

$$\hat{\mathbf{x}}_2(i) = \begin{cases} \sqrt{\frac{1}{k} - \frac{1}{n}(1+a)^2} & \text{when } i \leq k \\ -\frac{\frac{\sqrt{k}}{n}(1+a)}{\sqrt{1 - \frac{k}{n}(1+a)^2}} & \text{when } i > k \end{cases} \quad (4.18)$$

where $a = \frac{kp_s}{np_b}$.

So when the conditions in Result 4.2 is satisfied, we expect a gap $\Delta\hat{\mathbf{x}}_2 = \frac{1 - \frac{k}{n}(a+1)(a+2)}{\sqrt{k - \frac{k^2}{n}(1+a)^2}}$ between the signal entries and background entries.

Figure 4.5 shows the histograms of \mathbf{x}_2 for the same signals as shown in Figure 4.4. We still show the histogram of the background nodes in the left-down corner and the histogram of the signal nodes in the right-up corner. We can clearly see that the

signal can be separated from the background for $G(100, 0.7)$ and $G(100, 0.3)$ with a gap at about 0.1. These phenomena validate our theoretical result. Recall that this signal $G(100, 0.3)$ cannot be separated from the background using \mathbf{x}_1 , as shown in Figure 4.4(b). It indicates that minor eigenvectors capture more information about the embedded subtle signal than principal eigenvectors. We also observe in Figure 4.4(c) that even the minor eigenvector \mathbf{x}_2 could not separate the signal from the background (due to violation of the conditions shown in 4.2) when the signal is extremely weak.

Discussion

Denote $\xi_{\mathbf{x}_1}$ is the error term of estimation in Equation 4.17. We can easily derive that $\xi_{\mathbf{x}_1} = \xi_{\mathbf{y}_1} + \mu_1^{-1}S\xi_{\mathbf{y}_1} + \epsilon$. The first two terms are introduced by the error term of \mathbf{y}_1 , $\xi_{\mathbf{y}_1}$, which is caused by the randomness of the *ER* model. From Section 4.2, we know that $\|\xi_{\mathbf{y}_1}\|_2$ is about $\frac{1}{\sqrt{np}}$, which is very small when np is large. $\|\xi_{\mathbf{y}_1} + \mu_1^{-1}S\xi_{\mathbf{y}_1}\|_2 \leq (1 + \mu_1^{-1}\|S\|_2)\|\xi_{\mathbf{y}_1}\|_2 = (1 + \mu_1^{-1}\nu_1)\|\xi_{\mathbf{y}_1}\|_2$. The conditions of Result 1 require $\mu_1 > \nu_1$, so $\|\xi_{\mathbf{y}_1} + \mu_1^{-1}S\xi_{\mathbf{y}_1}\|_2 \leq 2\|\xi_{\mathbf{y}_1}\|_2$. Hence the first two terms are ensured to be small. The last term ϵ represents the higher order terms neglected in the approximation of \mathbf{x}_1 shown in Result 4.1. The higher order terms capture the influence of indirect neighbors. For subtle signals, we can safely omit them in our approximation. However, for strong signals, the first k entries of \mathbf{x}_1 tend to have larger values than we estimate.

Similarly, denote $\xi_{\mathbf{x}_2}$ as the error term of approximating \mathbf{x}_2 in 4.18. The error term of \mathbf{x}_2 is $\xi_{\mathbf{x}_2} \approx \frac{\xi_{z_1} - \mathbf{z}_1^T \xi_{\mathbf{x}_1} \mathbf{x}_1}{\sqrt{1 - \frac{k}{n}(1+a)^2}}$. $\|\xi_{\mathbf{x}_2}\|_2$ is about $\frac{\sqrt{\|\xi_{z_1}\|^2 + \frac{k}{n}\|\xi_{\mathbf{x}_1}\|^2}}{\sqrt{1 - \frac{k}{n}(1+a)^2}}$, which is ensured to be

small. On the other hand, $\Delta \widehat{\boldsymbol{x}}_2 \approx \frac{1}{\sqrt{k}}$, which is significantly larger than $\|\xi_{\boldsymbol{x}_2}\|_2$. Hence, we can use the minor eigenvector \boldsymbol{x}_2 to separate the signal from the background.

Our approximations of \boldsymbol{x}_1 and \boldsymbol{x}_2 shown in Results 4.1 and 4.2 focus on the scenario where both the signal and the background follow ER random graphs. In Section 4.3.2, we will extend to the general case where the background displays a community structure. For other types of signals, there are no concise approximations of eigenvectors \boldsymbol{x}_1 and \boldsymbol{x}_2 . However a general signal is more likely to have a larger eigenvalue than the ER signal with the same density. So we can have an idea how many eigenvectors we need at most. We generate a signal that follows the power law random graph model with the scaling exponent $\alpha = 2.3$. We control the density of the signal as 0.1, which is the same as the *ER* signal $G(100, 0.1)$. At most, we need We add this power law signal to the same background $G(10000, 0.1)$. Figures 4.4(d) and 4.5(d) show the histograms of \boldsymbol{x}_1 and \boldsymbol{x}_2 , respectively. We can observe that the power law signal is well separable from the background using \boldsymbol{x}_2 , which is quite different from that of *ER* signal $G(100, 0.1)$, as shown in Figure 4.5(c).

4.3.2 Detecting Signals from Graph with Community Structure

Social networks usually contain community structures. The background graph B is not necessarily a simple *ER* graph. The embedded signals, which are structurally dissimilar to the background, are not necessarily dense *ER* subgraphs either. Our theoretical results showed that when a signal is strong enough to stand out from the background, there exists some eigenvector with extremely large values on certain entries. Hence, we can use some statistics to filter out those eigenvectors that have extreme values on certain entries. Those eigenvectors tend to capture embedded

signals.

Our algorithm is sketched in Algorithm as follows. Given a graph, compute the eigen-decomposition of its adjacency matrix. For each eigenvector, calculate its kurtosis. Finally we output the eigenvectors with large kurtosis values. In general, eigen-decomposition of an $n \times n$ matrix takes a number of operations $O(n^3)$. In our algorithm, we do not need a full eigen-decomposition. With the user specified size and density of the signal, we only need to calculate a limited number of eigenvectors. Furthermore, adjacency matrices in our context are usually sparse and well structured. We used the Arnoldi/Lanczos algorithm [Golub and Van Loan, 1996] which generally needs $O(n)$ rather than $O(n^2)$ floating point operations at each iteration. The calculation of kurtosis of one eigenvector takes $O(n)$.

Background with Community Structure

When a background graph contains a community structure (i.e., multiple communities that are sparsely connected), we can treat B as a p -block diagonal network (with p disconnected communities C_i for $i = 1, \dots, p$) plus a matrix consisting all cross-community edges. Without loss of generality, we arrange communities in a proper order so that the principal eigenvectors \mathbf{y}_i have large values on the nodes in community C_i . When none of the signals is added in C_i , the principal eigenvector \mathbf{y}_i of B is almost sure to be the eigenvector of A . It is because the entries of $S\mathbf{y}_i$ are much smaller than $B\mathbf{y}_i$ and $A\mathbf{y}_i = B\mathbf{y}_i + S\mathbf{y}_i \approx B\mathbf{y}_i = \mu_i\mathbf{y}_i$.

Next we discuss about how the principal eigenvector \mathbf{y}_i and its associated minor eigenvectors are changed when signals are added in the community C_i . Since the

added signals do not change much other principal eigenvectors \mathbf{y}_j , $j = 1, \dots, p, j \neq i$, we can treat the problem as adding multiple signals to a single-community background graph. When we add q independent signals into G_B , without loss of generality, the adjacency matrix S can be written as a block matrix:

$$S = \begin{pmatrix} S_1 & & & \\ & \ddots & & \\ & & S_q & \\ & & & \mathbf{0} \end{pmatrix}_{n \times n}, \quad (4.19)$$

where each block S_i represents one signal. We arrange S_i in the decreasing order of their largest eigenvalues. Without loss of generality, we assume that the first eigenvalues for S_i are different. Then the first q eigenvectors of S usually have the following form:

$$(\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_q) = \begin{pmatrix} \mathbf{z}_{S_1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{z}_{S_2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{z}_{S_q} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{pmatrix}_{n \times q}, \quad (4.20)$$

where \mathbf{z}_{S_i} is the first eigenvector of S_i . When G_B still has a big gap in eigenvalues and the signals are of small size, we could still have $\mathbf{x}_1 \approx \mathbf{y}_1 + \frac{S\mathbf{y}_1}{\mu_1}$. However, the change on the principal eigenvector is often too small for the purpose of detection. Thus we need to explore the minor eigenvectors. Following a similar strategy, we construct

a series of vectors $\mathbf{v}_i \approx \frac{\mathbf{z}_i - (\mathbf{z}_i^T \mathbf{x}_1) \mathbf{x}_1}{\|\mathbf{z}_i - (\mathbf{z}_i^T \mathbf{x}_1) \mathbf{x}_1\|_2}$. Due to the small size of S_i , $\mathbf{z}_1^T \mathbf{x}_1$ is small so that \mathbf{v}_i is closely collinear with \mathbf{z}_i . By the form of eigenvectors in Equation 4.20, $\mathbf{z}_i^T \mathbf{z}_j = 0$ if $i \neq j$. So $\mathbf{v}_i^T \mathbf{v}_j \approx \mathbf{z}_i^T \mathbf{z}_j = 0$. A similar proof in Result 4.2 can be applied to show \mathbf{v}_i is closed to some eigenvectors within eigenvalues inside a certain interval around the first eigenvalue of S_i when the community is evenly distribution other than the signals. Such interval may overlap with that of other signals. With more communities, such interval may even include some other eigenvectors showing the background noise of other communities. In this way, we want to filter out the useful minor eigenvectors to detect the signals by some extra measures.

Kurtosis vs. L_1 -norm

In our algorithm, we propose the use of kurtosis to identify those eigenvectors that capture embedded signals. In statistics, kurtosis is defined to measure the peakness or infrequent extreme deviations of a set of data. For \mathbf{x}_j , $\kappa(\mathbf{x}_j) = \frac{n \sum_{i=1}^n (\mathbf{x}_j(i) - \bar{x}_j)^4}{(\sum_{i=1}^n (\mathbf{x}_j(i) - \bar{x}_j)^2)^2} - 3$ where $\bar{x}_j = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_j(i)$. The “minus 3” at the end of the formula is a correction to make the kurtosis of the normal distribution equal to zero. A high kurtosis distribution has a sharp peak and long-flat tails, whereas a low kurtosis distribution has a rounded peak and short-thin tails.

In [Miller et al., 2010], the authors used L_1 -norm of eigenvectors of modularity matrix to detect small dense subgraphs. The algorithm calculates the L_1 -norm of each eigenvector, subtracts its expected value, and normalizes by its standard deviation. If any of these modified L_1 -norm values is less than a threshold, the presence of a signal is declared. However, the algorithm assumes that the distribution or statistics of the

Table 4.2: Kurtosis and L_1 -norm of the first four eigenvectors of the observed graphs – $G(10000, 0.01)$ embedded with different signals

G_S	L_1 -norm				Kurtosis			
	$ \mathbf{x}_1 $	$ \mathbf{x}_2 $	$ \mathbf{x}_3 $	$ \mathbf{x}_4 $	$\kappa(\mathbf{x}_1)$	$\kappa(\mathbf{x}_2)$	$\kappa(\mathbf{x}_3)$	$\kappa(\mathbf{x}_4)$
\emptyset	99.50	78.62	78.90	78.96	-0.02	0.37	0.26	0.22
$G(100, 0.1)$	99.49	78.57	77.90	78.84	0.02	0.39	0.82	0.27
$G(100, 0.3)$	99.40	38.53	78.80	78.39	2.63	82.32	0.38	0.69
$G(100, 0.5)$	99.04	33.36	78.49	78.58	22.40	90.84	0.42	0.48
$G(100, 0.7)$	97.83	39.83	78.61	78.99	63.62	92.49	0.39	0.23
$PL(100, 2.3), \nu_1 = 30.4$	99.41	37.03	78.61	79.02	9.1	229.74	0.36	0.21

background graph is given. Otherwise, neither the expected value nor its standard deviation could be computed. Besides, they do not know how many eigenvectors they need to calculate.

We show that the kurtosis is a better metric than L_1 -norm for anomaly detection. First kurtosis is zero for the ER random graph when no signal is embedded. In ER random graphs, all the edges are generated by the same probability. Hence the entries of \mathbf{y}_i approximately follow a normal distribution. $\kappa(\mathbf{y}_i) \approx 0$ for all i no matter what size the graph is. However, for L_1 -norm, $|\mathbf{y}_1| \approx \sqrt{n}$ and $|\mathbf{y}_i| \approx \sqrt{\frac{2n}{\pi}}$ by the half normal distribution when $i \geq 2$. Graphs with different size can have very different L_1 -norm values. As shown in Figure 4.7(b), $|\mathbf{x}_i|$ tends to increase with i . As a result, the algorithm based on L_1 -norm had to assume that the distribution or statistics of the background graph is a-priori given but ours does not need such assumption.

Second, when there is an embedded signal, the signal entries are very different from the other entries. Kurtosis increase dramatically when there is a small portion of entries are different from others and the larger the difference is, the larger the

kurtosis is. We can see from Table 4.2 that both $\kappa(\mathbf{x}_1)$ and $\kappa(\mathbf{x}_2)$ (column 6 and 7) are larger than zero and $\kappa(\mathbf{x}_2)$ is always greater than $\kappa(\mathbf{x}_1)$. This property guarantees that our algorithm can correctly filter out the minor eigenvector that captures more signal information. Our algorithm based on the kurtosis can filter out those strong signals before weak signals.

Meanwhile the L_1 -norm of \mathbf{x}_1 does not change much (column 2) even when the strong signal $G(100, 0.7)$ is added. Although the L_1 -norm of \mathbf{x}_2 (column 3) decreases from the original value 78.62, the change does not capture the magnitude of the signal. For example, $|\mathbf{x}_2| = 33.36$ with signal $G(100, 0.5)$, which is smaller than both $|\mathbf{x}_2| = 38.53$ with a weaker signal $G(100, 0.3)$ and $|\mathbf{x}_2| = 39.83$ with a stronger signal $G(100, 0.7)$. L_1 -norm values can not give a right order to filter out strong signals first. In later evaluation on AstroPh network, we also find that the algorithm based on L_1 -norm fails to detect some very strong signal.

4.4 Empirical Evaluation

4.4.1 Synthetic Graph

Table 4.3: Eigenvalues, L_1 -norm and kurtosis of the graph of 3-community background with 3 signals

	Eigenvalue	L_1 -norm	Kurtosis
1	112.78	110.35	-1.76
2	55.87	89.98	-1.10
3	42.75	64.56	4.25
4	33.6	43.57	134.62
5	25.90	57.28	99.18
56	20.50	89.74	5.18
6 ~ 100	≈ 21	≈ 89	≈ 2

We generate a synthetic graph that contains a 3-community background with 3

embedded signals. The first community is $G_1 = G(10000, 0.1)$, which is the same as our illustrative example. The second community is $G_2 = G(5000, 0.01)$ and the third community is $G_3 = G(2000, 0.02)$. Nodes from different communities are sparsely connected by probability of 0.001 for each edge. We add two *ER* signals, $G(100, 0.3)$ and $G(100, 0.2)$ into community G_1 and a clique of 20 nodes to community G_3 .

We show eigenvalues, L_1 -norm and kurtosis of the first 100 eigenvectors in Table 4.3. In this example, \mathbf{x}_1 , \mathbf{x}_2 , and \mathbf{x}_3 are principal eigenvectors. We observe in Figures 4.6(a) that the three principal eigenvectors clearly capture the global community structure. However, three embedded signals can not be clearly separated from the background. Our algorithm based on kurtosis filters out three eigenvectors (\mathbf{x}_4 , \mathbf{x}_5 , and \mathbf{x}_{56}). We can observe in Figure 4.6(b) that three signals are clearly separated from the background communities using the identified minor eigenvectors. On the contrary, the algorithm based on L_1 -norm can only output two minor eigenvectors, \mathbf{x}_4 and \mathbf{x}_5 , which correspond to the first two signals. The third signal can not be detected because the L_1 -norm of \mathbf{x}_{56} has no significant difference from other minor eigenvectors. However, the kurtosis of \mathbf{x}_{56} is significantly larger than other minor eigenvectors.

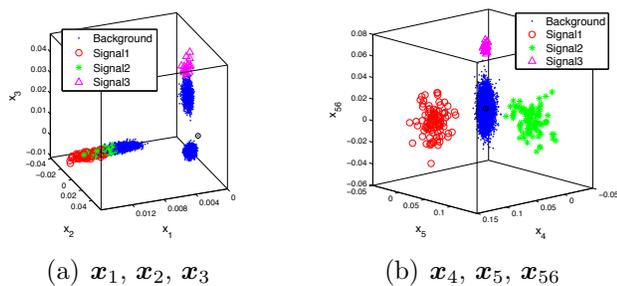


Figure 4.6: 3-Community background with 3 signals

4.4.2 Real Social Networks

In this section we explore whether our algorithm can effectively filter out subtle signals in the real graphs. We use Astro Physic collaboration network from the Stanford Network Analysis Package database².

AstroPh (Astro Physics) collaboration network is from the e-print arXiv and covers scientific collaborations between authors who submitted papers to Astro Physics category. It has 18772 nodes and 396160 edges. We do the eigen-decomposition of its adjacency matrix and calculate kurtosis and L_1 -norm of the first 100 eigenvectors with largest eigenvalues in Figure 4.7.

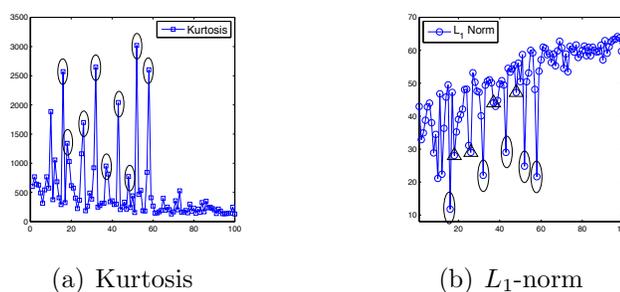


Figure 4.7: AstroPh network

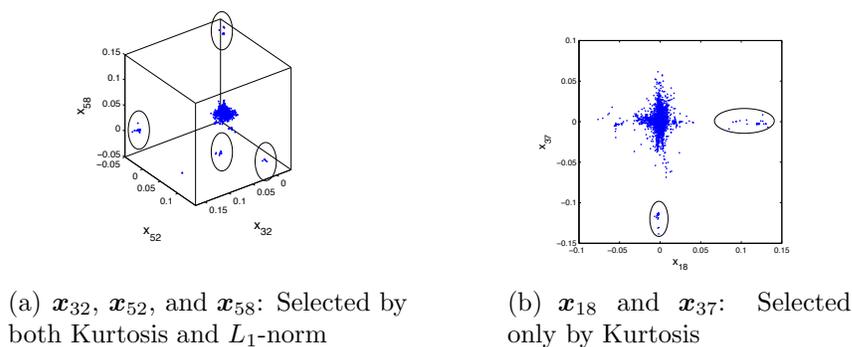


Figure 4.8: Comparison of Kurtosis and L_1 -norm

AstroPh contains a lot of small subgraphs with very high density. Figure 4.7(a)

²<http://snap.stanford.edu>

shows the result of our algorithm. Among the first 100 eigenvectors, we can find about 9 minor eigenvectors that are useful to detect signals. The scatter-plots in Figure 4.8(a) show three eigenvectors selected by both Kurtosis and L_1 -norm. We can detect four cliques with the size varying from 30 to 39 nodes. The selection based on L_1 -norm is difficult because the L_1 -norm of eigenvectors have an increasing tendency, as shown in Figure 4.7(b). In our experiment, we also find that L_1 -norm misses some important signals as we label in Figure 4.7(b). For example, our algorithm identified a clique with 36 nodes based on \mathbf{x}_{37} . However, L_1 -norm fails to detect this signal.

We also instrument some fake anomalies into the AstroPh graph. One anomaly is an *ER* graph with $k = 50$ and $p_s = 0.5$. The second is a power law signal of 50 nodes with the scaling exponent 2.3. The third is a bipartite signal. As we see from Figure 4.9, three more eigenvectors are captured and each captures one embedded anomaly.

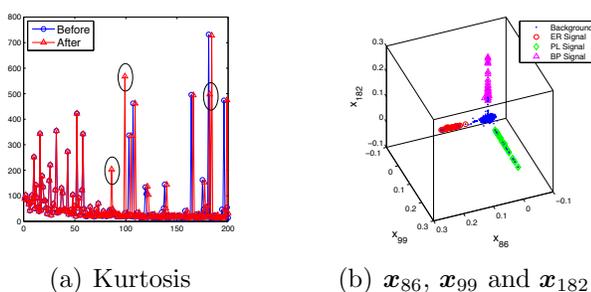


Figure 4.9: Add three signals to AstroPh

4.5 Summary

In this chapter, we have demonstrated the efficacy of using minor eigenvectors of a graph's adjacency matrix to detect subtle anomalies embedded in the background. Under the assumption of the *Erdos-Renyi* random graph model, we derive the explicit formula about how signal entries and background entries of certain eigenvector are

distributed. Our results show that the gap between signal entries and background entries in the minor eigenvector \mathbf{x}_2 is larger than that in the principal eigenvector \mathbf{x}_1 for subtle signals, which provides a theoretical explanation behind the utility of the approach. We further derive the detectability bound for the *Erdos-Renyi* random graph model. Although our theoretical analysis mainly focus on the *Erdos-Renyi* random graph model, we believe theoretical results based on other graph models could be developed. We also propose the use of kurtosis to filter out the eigenvectors that could capture the signals. Our approach remove the assumption in [Miller et al., 2010] that the background graph is generated using a-priori known parameters of a specific model. Empirical evaluations on both synthetic data and real social networks show effectiveness of our approach to detecting subtle signals. Part of this chapter will be published in Journal of Intelligent Information System [Wu et al., 2013].

CHAPTER 5: PRIVACY PRESERVING GRAPH RECONSTRUCTION

One big difference of social networks with others networks(i.e., biology networks) is the privacy issue. The privacy concerns associated with data analysis over social networks have incurred the recent research on privacy-preserving social network analysis, particularly on privacy-preserving publishing social network data. However, perturbations may cause huge lost in utility and the published graphs are totally useless for analysts. In this chapter, we explore the application of the adjacency spectral properties in privacy preserving graph reconstruction. We focus on whether we can reconstruct a graph from the edge randomized graph such that accurate feature values can be recovered. We exploit spectral properties of the graph data and show why noise could be separated from the perturbed graph using low rank approximation. We also show key differences from previous findings of point-wise reconstruction methods on numerical data through empirical evaluations and theoretical justifications.

5.1 Introduction

In a social network, nodes usually correspond to individuals or other social entities, and an edge corresponds to the relationship between two entities. Each entity can have a number of attributes, such as age, gender, income, and a unique identifier. In this chapter, we consider social networks in which node identities (and even entity attributes) are not confidential but sensitive links between individuals are confidential

and should be protected. For example, in a transaction network, an edge denoting a financial transaction between two individuals is considered confidential while nodes corresponding to individual accounts is non-confidential.

To protect privacy, one common practice is to publish a naive node-anonymized version of the network, e.g., by replacing the identifying information of the nodes with random IDs. While the naive node-anonymized network still permits useful analysis, as first pointed out in [Backstrom et al., 2007; Hay et al., 2007], this simple technique does not guarantee privacy since adversaries may re-identify a target individual from the anonymized graph by exploiting some known structural information of his neighborhood.

The state-of-the-art anonymization methods on network data have three categories: K -anonymity privacy preservation via edge modification[Liu and Terzi, 2008; Zhou and Pei, 2008; Zou et al., 2009], edge randomization[Hay et al., 2007; Ying and Wu, 2008, 2009a,b], and clustering-based generalization[Bhagat et al., 2009; Campan and Truta, 2008; Cormode et al., 2008; Hay et al., 2008; Zheleva and Getoor, 2007]. These above anonymization approaches have been shown as a necessity in addition to naive anonymization to preserve privacy in publishing social network data.

We focus on one specific edge randomization strategy, *Rand Add/Del*, which randomly adds one edge followed by deleting another edge and repeats this process for k times. This strategy preserves the total number of edges in the original graph. Figure 5.1 shows the process of graph modification and reconstruction. Recall the the adjacency matrix $A = (a_{ij})_{n \times n}$: $a_{ij} = 1$ if node i and j are connected and $a_{ij} = 0$ otherwise. The edge randomization process can be written in the matrix form $\tilde{A} = A + E$,

where the perturbation matrix E is defined as $e_{ij} = e_{ji} = 1$ if edge (i, j) is added, $e_{ij} = e_{ji} = -1$ if edge (i, j) is deleted, and $e_{ij} = 0$ otherwise. The process of randomization and the randomization parameter k are assumed to be published along with the released graph \tilde{G} .

For randomization approach, there are two fundamentally conflicting requirements: privacy for the individual entry (a_{ij}) and utility of the perturbed data (\tilde{A}). It has been shown in [Hay et al., 2007; Ying and Wu, 2008] that a medium or large perturbation is needed in order to protect the privacy of the individual entry under feature based attacks or structural attacks. However, as shown in our empirical evaluation, the utility of the released randomized graph (in terms of topological features) is significantly lost in the randomized graph when a medium or large perturbation is applied.

To preserve utility, several advanced randomization strategies have been investigated recently. In [Ying and Wu, 2008], Ying and Wu presented a randomization strategy that can preserve the spectral properties of the graph. They presented two spectrum preserving randomization methods, *Spctr Add/Del* and *Spctr Switch*, which keep graph spectral characteristics (i.e., the largest eigenvalue of the adjacency matrix and the second smallest eigenvalue of the Laplacian matrix) not much changed during randomization by examining eigenvector values of nodes to choose where edges are added/deleted or switched. In [Hanhijarvi et al., 2009; Ying and Wu, 2009a], the authors studied the problem of how to generate a synthetic graph matching given features of a real social network in addition to a given degree sequence. They proposed a Markov Chain based feature preserving randomization. Although the proposed advanced randomization strategies generally can preserve more structural properties, it

is very challenging to quantify disclosure risks since the process of feature preserving strategies are complicated.

In this chapter we adopt a different approach. We focus on whether we can reconstruct a graph \hat{G} from the randomized one \tilde{G} such that \hat{G} is closer to the original graph G than \tilde{G} in terms of some feature f , i.e., $|f(\hat{G}) - f(G)| \leq |f(\tilde{G}) - f(G)|$. In particular, we study the use of low rank approximation approach to reconstruct structural features from the randomized graph. We exploit spectral properties of the graph data and show that the noise could be separated from the perturbed graph.

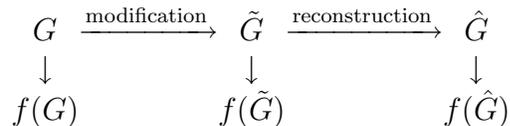


Figure 5.1: The process of graph modification and reconstruction

The rest of this chapter is organized as follows.

In Section 5.2, we first discuss topological features used in this paper and revisit those low rank approximation based reconstruction methods on numerical data. In Section 5.3, we examine the spectra of network data and show the relationship between the positive (negative) eigenvalues and the reconstructed graph structure via low rank approximation. In Section 5.4, we present our low rank approximation based reconstruction algorithm. We also show our novel method to determine the optimal rank for low rank approximation. We conduct empirical evaluations on three real social networks in terms of both privacy and utility in Section 5.5. In Section 5.6, we further examine what type of graphs are sensitive to low rank approximation based reconstruction in terms of privacy protection. Finally we offer our concluding remarks

and point out future directions in Section 5.7.

5.2 Preliminary

5.2.1 Notation and Features

Table 5.1: Notations

n, m	number of nodes and edges
k	number of edges added and deleted
r	number of eigen-pairs in low rank approximation
$A(\tilde{A})$	adjacency matrix of graph G (\tilde{G})
$A_r(\tilde{A}_r)$	rank r approximation of A (\tilde{A})
\hat{A}	adjacency matrix of the reconstructed graph
λ_i, \mathbf{x}_i	the i th largest eigenvalue in magnitude of A and the corresponding eigenvector
E	difference matrix, $E = \hat{A} - A$
ε_1	the largest eigenvalue of E in magnitude

We use the tilde conventions to denote perturbations and use the hat conventions to denote estimations. The original quantity is denoted by the same symbol without a tilde or hat. Table 5.1 summarizes our notations used in this chapter.

In this chapter, we consider the following topological features of the graph:

- λ_1 , the largest eigenvalue of the adjacency matrix A .
- ν_2 , the second largest eigenvalue of the normal matrix $N = D^{-1}A$.
- Q , modularity is defined as the fraction of all edges that lie within communities minus the expected value of the same quantity in a graph generated from a random model which keeps the expected number of degree for each node.
- C , transitivity measure is one type of clustering coefficient measure and char-

acterizes the presence of local loops near a vertex. It is formally defined as $C = 3N_{\Delta}/N_3$, where N_{Δ} is the number of triangles and N_3 is the number of connected triples.

Throughout this chapter, we use the *polblogs* as an example. Recall the *polblogs* network compiles the 16714 links among 1222 US political blogs, based on incoming and outgoing links and posts during the time of the 2004 presidential election [Adamic and Glance, 2005].

5.2.2 Reconstruction Methods on Numerical Data Revisited

The low rank approximation has been well investigated as a point-wise reconstruction method in the numerical setting. In the setting of randomizing numerical data, a data set U with m records of n attributes is perturbed to \tilde{U} by an additive noise data set V with same dimensions as U , i.e., $\tilde{U} = U + V$. A spectral filtering based reconstruction method was first proposed in [Kargupta et al., 2003] to reconstruct original data values from the perturbed data. Similar methods (e.g., PCA based reconstruction method [Huang et al., 2005], SVD based reconstruction method [Guo et al., 2008]) have also been investigated. All methods exploited spectral properties of the correlated data to remove the noise from the perturbed data set. This is because real-world numerical data is usually highly correlated in a low dimensional space while the randomly added noise is distributed (approximately) equally over all dimensions. Then, more accurate aggregate features can be reconstructed by projecting the randomized data into a proper low dimensional space where the majority information of the original data is preserved.

Spectral Filtering

The objective of the spectral filtering based approach is to derive the estimation \hat{U} of U from the perturbed data \tilde{U} based on random matrix theory. An explicit filtering procedure is shown below.

1. Calculate the covariance matrix of \tilde{U} by $\tilde{\Sigma} = \tilde{U}^T \tilde{U}$ (assume U has mean equal to 0).
2. The covariance matrix $\tilde{\Sigma}$ is symmetric and positive semi-definite, we apply spectral decomposition on $\tilde{\Sigma}$ to get its i -th largest eigenvalue $\tilde{\lambda}_i$ and the corresponding eigenvector $\tilde{\mathbf{x}}_i$.
3. Derive the eigenvalues information from the covariance matrix of the noise V and choose a proper number of dimensions, r .
4. Let $\tilde{X}_r = [\tilde{\mathbf{x}}_1 \tilde{\mathbf{x}}_2 \cdots \tilde{\mathbf{x}}_r]$, and the orthogonal projection on to the subspace spanned by $\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_r$ is $P_r = \tilde{X}_r \tilde{X}_r^T$. Obtain the estimated data set using $\hat{U} = \tilde{U} P_r$.

SVD

Singular value decomposition decomposes a matrix $U \in \mathbb{R}^{m \times n}$ (say $m \geq n$) as $U = \sum_{i=1}^n \sigma_i \mathbf{p}_i \mathbf{q}_i^T$, where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n$ are the *singular values* and $\mathbf{p}_i \in \mathbb{R}^m$ and $\mathbf{q}_i \in \mathbb{R}^n$ are the left and right *singular vector* of σ_i respectively. Similarly, after perturbation $\tilde{U} = U + V$, we have the SVD of \tilde{U} as $\tilde{U} = \sum_{i=1}^n \tilde{\sigma}_i \tilde{\mathbf{p}}_i \tilde{\mathbf{q}}_i^T$. The SVD reconstruction method simply reconstructs U approximately as $\hat{U} = \tilde{U}_r = \sum_{i=1}^r \tilde{\sigma}_i \tilde{\mathbf{p}}_i \tilde{\mathbf{q}}_i^T$.

It has been shown that the spectral filtering method is equivalent to the SVD reconstruction method [Guo et al., 2008]. We can observe that all spectral based methods reconstruct the original data by projecting the perturbed data onto the projection subspaces that are determined by the first r eigenvectors for the spectral filtering method or by the first r singular vectors for the SVD method. The original spectral filtering algorithm [Kargupta et al., 2003] suggested using $r = \max\{i | \tilde{\lambda}_i \geq \varepsilon_1\}$ to determine the first r eigen components, where ε_1 is the largest eigenvalue of the noise covariance matrix $\text{Cov}(V)$. The authors of [Guo et al., 2008] further proved that using $r = \max\{i | \tilde{\lambda}_i \geq 2\varepsilon_1\}$ can achieve approximately optimal reconstruction for i.i.d. noise. This is because that it only includes the i -th eigen component when the benefit due to inclusion of the i -th component is greater than the loss due to the noise projected on the i -th component, i.e., $\tilde{\lambda}_i \geq 2\varepsilon_1$.

5.3 Low Rank Approximation on Graph Data

The adjacency matrix A discussed here is different from the numerical data set U and the covariance matrix Σ in the following perspectives. First, A is a symmetric 0-1 matrix whereas U is a numerical matrix and the covariance matrix Σ is a semi-definite one. Second, for numerical data, all the eigenvalues of Σ are real and non-negative. For graph data A , the covariance matrix is not properly defined. We can see that in AA^T , the non-zero entry at row i column j means j is 2 steps away from i . When we directly apply eigen-decomposition on the adjacency matrix A , the eigen-decomposition of A contains negative eigenvalues.

In Section 5.3.1, we study the low rank approximation on graph data. In Section 5.3.2, we examine the spectra of graph data and show the relationship between the

topological graph structure and the significant eigen-pairs that may involve both positive and negative eigenvalues.

5.3.1 Low Rank Approximation

Let λ_i be A 's i -th largest eigenvalue in magnitude: $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n|$, and \mathbf{x}_i denotes the eigenvector of λ_i . The rank r approximations of A via the eigen-decomposition are given by:

$$A_r = \sum_{i=1}^r \lambda_i \mathbf{x}_i \mathbf{x}_i^T. \quad (5.1)$$

Among all the matrix with rank no larger than r , the low rank approximation A_r shown in (5.1) is the matrix closest to A in term of the Frobenius norm [Stewart and Sun, 1990]:

$$\|A_r - A\|_F^2 = \min_{\text{rank}(B) \leq r} \|B - A\|_F^2.$$

The key difference between our low rank approximation on graph data and those low rank approximation methods on numerical data is that we rank eigenvalues based on their absolute values and also include those significant negative eigenvalues in the low rank approximation. In Section 5.3.2, we will illustrate the relationship between the graph topology and significant positive and negative eigenvalues.

Because A_r is a real matrix, we need to derive a symmetric 0-1 matrix \hat{A} that is close to A_r . Our strategy is to find the $2m$ largest off-diagonal entries in A_r (note that A and \hat{A} are symmetric) and set the corresponding entries in \hat{A} as 1 and others as 0, i.e.,

$$\hat{A}(i, j) = \begin{cases} 1, & \text{if } A_r(i, j) \text{ is one of the } 2m \\ & \text{largest off-diagonal entries,} \\ 0, & \text{otherwise.} \end{cases} \quad (5.2)$$

By using (5.2), we have the following property.

Property 5.1: If \hat{A} is obtained by (5.2), \hat{A} is the closest adjacency matrix to A_r in term of the Frobenius norm, i.e.,

$$\|\hat{A} - A_r\|_F^2 = \min_{B \in \mathcal{A}_n^m} \|B - A_r\|_F^2,$$

where \mathcal{A}_n^m denotes the set of all symmetric $n \times n$ 0-1 matrices with $2m$ off-diagonal 1's and 0 else where.

The following theory states that the difference between the spectrum of \hat{A} and that of A_r is upper bounded by $\|\hat{A} - A_r\|_F^2$.

Theorem 5.1: [Stewart and Sun, 1990] Given two $n \times n$ symmetric matrices A and E with eigenvalues $\lambda_1 \geq \dots \geq \lambda_n$ and $\varepsilon_1 \geq \dots \geq \varepsilon_n$ respectively. Let $\tilde{\lambda}_1 \geq \dots \geq \tilde{\lambda}_n$ be the eigenvalues of $\tilde{A} = A + E$. Then we have

$$\lambda_i + \varepsilon_n \leq \tilde{\lambda}_i \leq \lambda_i + \varepsilon_1, \quad (5.3)$$

$$\sum_i (\tilde{\lambda}_i - \lambda_i)^2 \leq \|E\|_F^2. \quad (5.4)$$

By minimizing this upper bound, we expect the eigenvalues and eigenvectors of \hat{A} is close to those of A . In fact, many spectral properties, such as eigenvectors, the sum of several eigenvalues, and spectral subspace, are stable when the magnitude of the difference matrix is moderate. For varies spectrum bounds and more details, please refer to [Stewart and Sun, 1990]. Since the graph topology is closely related with eigenvalues and eigenvectors of the graph, we expect that \hat{A} can preserve the major topological information of the original graph.

5.3.2 Leading Eigen-pairs vs. Graph Topology

In this section, we study the relationship between eigen-pairs and graph topology. In particular, we examine the role of positive and negative eigenvalues in graph topology.

Without loss of generality, we partition the node set V into two groups $V_1 = \{1, \dots, n_1\}$ and $V_2 = \{n_1 + 1, \dots, n\}$. Then the adjacency matrix can be partitioned as

$$A = A_{\text{inner}} + A_{\text{inter}} = \begin{pmatrix} A_{11} & \mathbf{0} \\ \mathbf{0} & A_{22} \end{pmatrix} + \begin{pmatrix} \mathbf{0} & A_{12} \\ A_{12}^T & \mathbf{0} \end{pmatrix}, \quad (5.5)$$

where A_{11} and A_{22} represent the edges within V_1 and V_2 respectively, and A_{12} represents the edges between V_1 and V_2 .

Disconnected Communities

In an ideal graph with two disconnected communities, A_{11} and A_{22} are dense matrices of comparable size, and $A_{12} = \mathbf{0}$. Then, all the eigenvalues of A_{11} and A_{22} are eigenvalues of A . Let μ_1 and η_1 be the largest eigenvalue in magnitude of A_{11} and A_{22} with eigenvector \mathbf{y}_1 and \mathbf{z}_1 respectively. μ_1 and η_1 are two eigenvalues of A with eigenvectors $\begin{pmatrix} \mathbf{y}_1 \\ \mathbf{0} \end{pmatrix}$ and $\begin{pmatrix} \mathbf{0} \\ \mathbf{z}_1 \end{pmatrix}$. Note that, by the Perron-Frobenius theorem [Cvetkovic et al., 1997], μ_1 and η_1 must be positive and all entries in \mathbf{y}_1 and \mathbf{z}_1 must be positive. Assume $\mu_1 \geq \eta_1$, then

$$A_1 = \mu_1 \begin{pmatrix} \mathbf{y}_1 \\ \mathbf{0} \end{pmatrix} (\mathbf{y}_1^T, \mathbf{0}) = \begin{pmatrix} \mu_1 \mathbf{y}_1 \mathbf{y}_1^T & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}. \quad (5.6)$$

We can see all large entries only appear among the nodes in V_1 . Similarly, the rank 2 approximation of A is given by

$$A_2 = \begin{pmatrix} \mu_1 \mathbf{y}_1 \mathbf{y}_1^T & \mathbf{0} \\ \mathbf{0} & \eta_1 \mathbf{z}_1 \mathbf{z}_1^T \end{pmatrix}, \quad (5.7)$$

and large entries appear both within V_1 and V_2 . Figure 5.2 shows a synthetic network with 60 nodes and 280 edges. This network contains two disconnected 30-node communities generated via *ER* model with inner-community probability 0.5. The derived graphs \hat{A} by discretizing A_1 and A_2 via (5.2) are shown in Figure 5.2(b) and 5.2(c). For the graph derived from A_1 , all the edges appear in only one of the communities. After adding one more eigen-pair in the low rank approximation, the derived graph shown in Figure 5.2(c) reveals two very clear communities.

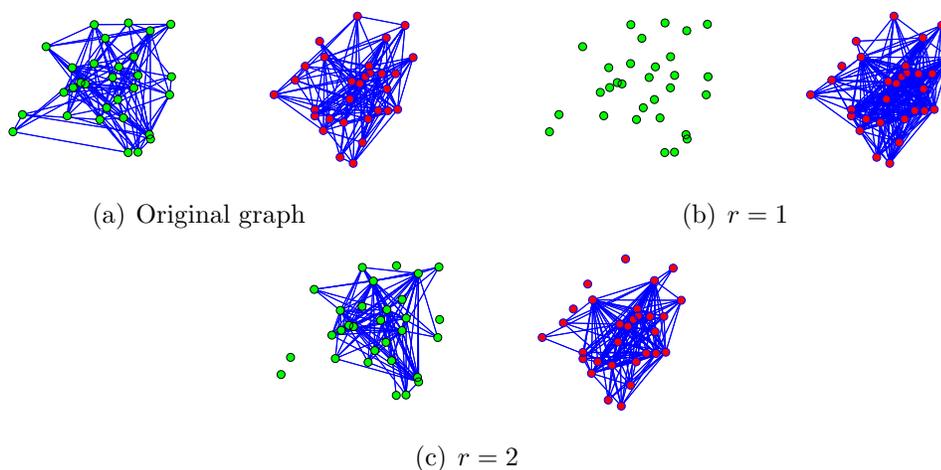


Figure 5.2: Synthetic random graph with two disconnected communities

Bipartite Graphs

The negative eigenvalues are closely related to the bipartite structure of the graph. A bipartite graph is a graph containing two types of nodes, and edges only exist between

two nodes of different types. For a bipartite graph, A_{11} and A_{22} in (5.5) are both zero matrix. The spectrum of A is then fully determined by A_{12} . Let $\sigma \geq 0$ be the largest singular value of A_{12} (note A_{12} is generally a non-square matrix) with right-singular value \mathbf{u} and left-singular value \mathbf{v} . If G is a connected graph, all the entries of \mathbf{u} and \mathbf{v} are positive. It is easy to verify that σ and $-\sigma$ are both the eigenvalues of A with eigenvalue $\begin{pmatrix} \mathbf{u} \\ \mathbf{v} \end{pmatrix}$ and $\begin{pmatrix} -\mathbf{u} \\ \mathbf{v} \end{pmatrix}$ respectively. Similar as (5.6) and (5.7), we can have

$$A_1 = \begin{pmatrix} \sigma \mathbf{u} \mathbf{u}^T & \sigma \mathbf{u} \mathbf{v}^T \\ \sigma \mathbf{v} \mathbf{u}^T & \sigma \mathbf{v} \mathbf{v}^T \end{pmatrix}, \quad A_2 = \begin{pmatrix} \mathbf{0} & 2\sigma \mathbf{u} \mathbf{v}^T \\ 2\sigma \mathbf{v} \mathbf{u}^T & \mathbf{0} \end{pmatrix}.$$

We can see that entries within V_1 and V_2 in A_1 are non-zero, which is significantly different from A . However, as we introduce the leading negative eigenvalue, non-zero entries in A_2 only appear in those entries across two type of nodes.

Figure 5.3(a) shows a synthetic bipartite graph with 60 nodes and 94 edges. Any two nodes of different colors have probability 0.1 to be connected, and nodes of the same color do not connect to each other. The first two eigenvalues are 4.27 and -4.27 respectively. The derived graphs (\hat{A}) from A_1 and A_2 are shown in Figure 5.3(b) and 5.3(c) respectively. We can see that, when only the positive eigenvalue and its eigenvector are involved, many edges connecting two nodes of the same type are falsely introduced in \hat{A} ; and as the negative eigenvalue and its eigenvectors are included, \hat{A} derived from A_2 shown in Figure 5.3(c) correctly reveals the bipartite structure.

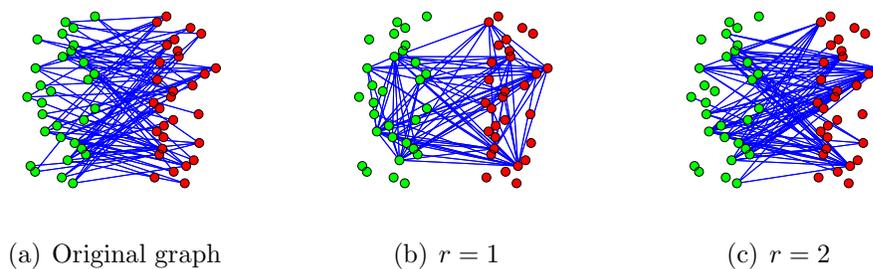


Figure 5.3: Synthetic random quasi-bipartite graph

Real Social Networks

Real world social networks usually have clear connected community structures. In other words, there are few non-zero entries in A_{inter} , i.e., $\|A_{\text{inter}}\|_F^2$ is small. By Theory 5.1, the eigenvalues and eigenvectors of A are close to A_{inner} , and similar to (5.6) and (5.7), the upper right and lower left parts of A_1 and A_2 are close to $\mathbf{0}$.

Figure 5.4(a) shows a synthetic network with 2 clear but connected communities. It is generated by adding inter-community edges with probability 0.05 to the synthetic graph in Figure 5.2(a). The first four eigenvalues are $\lambda_1 = 10.30$, $\lambda_2 = 9.05$, $\lambda_3 = -4.82$, and $\lambda_4 = -4.79$. The $2m$ largest entries in A_2 and A_4 are shown in Figure 5.4(b) and 5.4(c) respectively. Similar as Figure 5.2(c), large entries of A_2 appear in both of the two communities, and no inter-community entries have large values. As two negative eigenvalues λ_3 and λ_4 are included in A_4 , inter-community edges emerge. \hat{A} is closer to the original graph A .

For graphs containing c large communities, the c largest positive eigenvalues corresponds to the communities. If node j and node k belong to the i -th community C_i , the j -th and k -th entry of \mathbf{x}_i (x_{ji} and x_{ki}) tend to be large, which matches the finding by Ying and Wu [Ying and Wu, 2009c] that eigenvectors corresponding to the

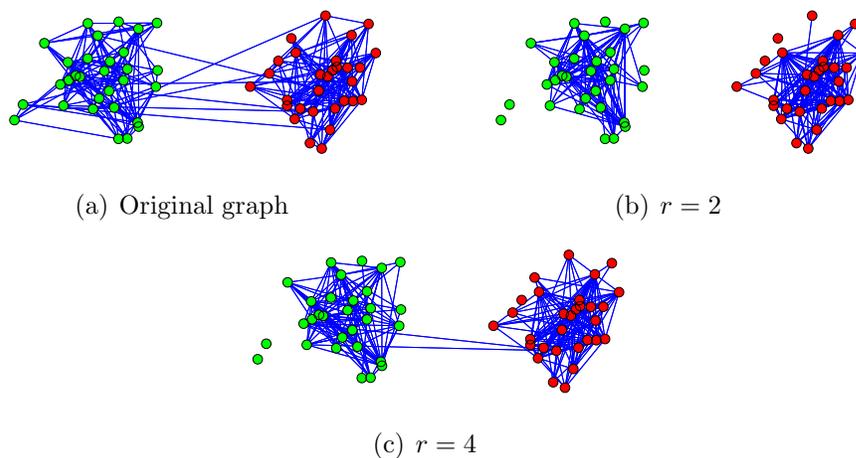


Figure 5.4: Synthetic random graph with two clear but connected communities

large positive eigenvalues of A are good indices of the community partition. Then, the (j, k) entry of matrix $(\lambda_i \mathbf{x}_i \mathbf{x}_i^T)$, which equals to $\lambda_i x_{ji} x_{ki}$, tends to be large. Therefore, large entries in the low rank approximation matrix $A_c = \sum_{i=1}^c \lambda_i \mathbf{x}_i \mathbf{x}_i^T$ would reflect edges within these communities.

Similarly, for a quasi-bipartite graph, A_{inner} has few non-zero entries ($\|A_{\text{inner}}\|_F^2$ is small). Hence, the spectrum of A would have the similar pattern of A_{inter} and some of the leading eigenvalues of A are negative. Besides quasi-bipartite graphs, bowtie graphs [?] or graphs with very skewed degree distribution also have their adjacency matrices close to bipartite graphs (in term of the Frobenius norm). In such graphs, a large number of nodes do not connect to each other directly but through a small number of core nodes and core nodes are well connected to each other. Suppose node set V_1 represents the core nodes, then A_{11} represents the edges among core nodes, and A_{22} represents the edges among non-core nodes. $\|A_{\text{inner}}\|_F^2$ is small because there are few edges in A_{22} and the size of A_{11} is small. By the perturbation theory, the spectrum of a bowtie graph is similar to that of a bipartite graph and has significant

negative eigenvalues.

5.4 Reconstruction from Randomized Graphs

Recall that in the edge randomization process, we randomly add k false edges followed by deleting k true edges. The perturbation can be expressed as a perturbation matrix E where $e_{ij} = e_{ji} = 1$ if edge (i, j) is added, $e_{ij} = e_{ji} = -1$ if edge (i, j) is deleted, and $e_{ij} = 0$ otherwise. The process of randomization and the randomization parameter k are assumed to be published along with the released graph.

In Section 5.4.1, we present our low rank approximation based reconstruction algorithm and show why the algorithm (given an optimal rank r) can reconstruct topological features accurately. In Section 5.4.2, we conduct theoretical analysis and give our procedure to determine the optimal r .

5.4.1 Algorithm

Let $\tilde{\lambda}_i$ be \tilde{A} 's i -th largest eigenvalue in magnitude: $|\tilde{\lambda}_1| \geq |\tilde{\lambda}_2| \geq \dots \geq |\tilde{\lambda}_n|$, and $\tilde{\mathbf{x}}_i$ denotes the eigenvector of $\tilde{\lambda}_i$. The rank r approximation of \tilde{A} is $\tilde{A}_r = \sum_{i=1}^r \tilde{\lambda}_i \tilde{\mathbf{x}}_i \tilde{\mathbf{x}}_i^T$.

The topology of the randomized graph \tilde{A} may be significantly different from that of the original graph A when the magnitude of perturbation is medium or large. However, by choosing an appropriate r , \tilde{A}_r can preserve major topological structures. This is because that \tilde{A}_r only includes those significant eigen-pairs and filters out all noises added in the rest dimensions. Recall that the leading eigen-pairs reflect the dominant structure of the graph, e.g., those eigen-pairs with large positive eigenvalues capture the inner structure of those significant communities and those eigen-pairs with negative eigenvalues capture the inter-community connections. Since \tilde{A} is obtained by randomly adding and deleting edges on A , both strong inner- and inter-community

connections are less affected by the randomization. Therefore, \tilde{A}_r consisting of the leading eigen-pairs can still capture the major topological structures of the original graph.

After low rank approximation, \tilde{A}_r is a real matrix. Similarly we adopt the following strategy to obtain a 0-1 matrix \hat{A} as the reconstructed graph.

$$\hat{A}(i, j) = \begin{cases} 1, & \text{if } \tilde{A}_r(i, j) \text{ is one of the } 2m \\ & \text{largest off-diagonal entries,} \\ 0, & \text{otherwise.} \end{cases} \quad (5.8)$$

We show our graph reconstruction algorithm in Algorithm 3.

Algorithm 3 Graph Reconstruction Algorithm

Input: randomized graph \tilde{A} , randomization parameter k

Output: reconstructed graph \hat{A}

- 1: Calculate $\tilde{\lambda}_i$ and $\tilde{\mathbf{x}}_i$, $|\tilde{\lambda}_1| \geq \dots \geq |\tilde{\lambda}_n|$.
- 2: Calculate λ_1^* using

$$\lambda_1^* = \frac{(mk - mN)\tilde{\lambda}_1 + mk\tilde{\lambda}_0}{kN - mN + mk}$$

$$N = \binom{n}{2} - m, \text{ and } \tilde{\lambda}_0 = \tilde{\mathbf{x}}_1^T(\mathbf{1} - I - \tilde{A})\tilde{\mathbf{x}}_1$$

- 3: $r = 1$;
- 4: **repeat**
- 5: Construct \hat{A} from $\tilde{A}_r = \sum_{i=1}^r \tilde{\lambda}_i \tilde{\mathbf{x}}_i \tilde{\mathbf{x}}_i^T$ by

$$\hat{A}(i, j) = \begin{cases} 1, & \text{if } \tilde{A}_r(i, j) \text{ is one of the } 2m \\ & \text{largest off-diagonal entries,} \\ 0, & \text{otherwise.} \end{cases}$$

- 6: $\hat{\lambda}_1 =$ the largest eigenvalue of \hat{A} in magnitude;
 - 7: $r = r + 1$;
 - 8: **until** $|\hat{\lambda}_1 - \lambda_1^*|$ increases
-

5.4.2 Determine r in Low Rank Approximation based Graph Reconstruction

In the low rank approximation, the different choices of r can significantly affect the accuracy of reconstruction. When r is very small, the topological structure of

the reconstructed \hat{A} may be significantly different from that of the original graph A . This is because too few eigen-pairs are included in reconstruction and not all major structures are captured during the reconstruction. On the other hand, the reconstruction with a large r may introduce too much noise. As a result, the benefit due to the inclusion of major structures is decreased by the loss due to the added noise. Figure 5.5 shows the reconstructed feature values, along with the original and randomized values, for *polblogs* network as the choice of r varies ($k = 0.4m$). When r is very small, the reconstructed feature values are significantly different from the original value, indicating that the topology of \hat{A} is very different from the original graph. As r increases, $f(\hat{A})$ approaches the original value, and for some r , the reconstructed value approximately equals to original value. Further increasing r makes the reconstructed feature values approach to the randomized one, indicating that too much noise is included in \hat{A} . We can see that choosing a proper r is critical in reconstructing graphs.

We would emphasize again that the strategies of determining r in reconstructing numerical data (via comparing $\tilde{\lambda}_i$ with ε_1) is not applicable here. This is because the entries of E can only be 0, 1 and -1 , and the magnitude of E can be very large while k is actually moderate. For example, when we randomly add and delete $k = 0.4m$ edges on *polblogs* network, we can get $\varepsilon_1 = 28.6$, which is greater than almost all $\tilde{\lambda}_i$ except $\tilde{\lambda}_1$ and $\tilde{\lambda}_2$. The strategies of determining r by $r = \max\{i | \tilde{\lambda}_i \geq \varepsilon_1\}$ [Huang et al., 2005] would choose $r = 2$. However, as shown in Figure 5.5, when $r = 2$, the feature values of the reconstructed graph are significantly different from the original value.

One natural idea is to determine r such that $f(\hat{A})$ is approximately equal to $f(A)$ for some feature f . One problem is that feature values of the original graph may not be available to data miners. In general, it is difficult, if not impossible, to derive the accurate estimates of real space feature values (e.g., cluster coefficient, transitivity) from the randomized graph using the statistics of randomization. However, for the spectral feature λ_1 , we can derive the moment estimate of the original values, as shown in our next result.

Result 5.1: Let $N = \binom{n}{2} - m$, and $\tilde{\lambda}_0 = \tilde{\mathbf{x}}_1^T(\mathbf{1} - I - \tilde{A})\tilde{\mathbf{x}}_1$, where $\mathbf{1}$ is a $n \times n$ all 1 matrix and I is the identity matrix. Let λ_1^* denote the moment estimator of λ_1 . If \tilde{A} is obtained by adding k false edges and deleting k true edges, λ_1^* is given by

$$\lambda_1^* = \frac{(mk - mN)\tilde{\lambda}_1 + mk\tilde{\lambda}_0}{kN - mN + mk} \quad (5.9)$$

Proof. Define $\lambda_0 = \mathbf{x}_1^T(\mathbf{1} - I - A)\mathbf{x}_1$ and $\tilde{\lambda}_0 = \tilde{\mathbf{x}}_1^T(\mathbf{1} - I - \tilde{A})\tilde{\mathbf{x}}_1$. Since $\tilde{\lambda}_1 = \tilde{\mathbf{x}}_1^T \tilde{A} \tilde{\mathbf{x}}_1$, we have

$$\mathbf{E}(\tilde{\lambda}_1) = \mathbf{E}(\tilde{\mathbf{x}}_1^T \tilde{A} \tilde{\mathbf{x}}_1) \approx \mathbf{x}_1^T \mathbf{E}(\tilde{A})\mathbf{x}_1. \quad (5.10)$$

We adopt the assumption that $\tilde{\mathbf{x}}_1 \approx \mathbf{x}_1$ in establishing the second equality of (5.10). Since in *Rand Add/Del* every existing (non-existing) edge of A has the same probability to be add (deleted), we have $\mathbf{E}(\tilde{a}_{ij}) = \frac{m-k}{m}$ if $a_{ij} = 1$, and $\mathbf{E}(\tilde{a}_{ij}) = \frac{k}{N}$ if $a_{ij} = 0$ and $i \neq j$, where $N = \binom{n}{2} - m$, i.e.,

$$\mathbf{E}(\tilde{A}) = \frac{m-k}{m}A + \frac{k}{N}(\mathbf{1} - I - A).$$

Continue with (5.10), we have

$$\begin{aligned}\mathbf{E}(\tilde{\lambda}_1) &= \frac{m-k}{m} \mathbf{x}_1^T A \mathbf{x}_1 + \frac{k}{N} \mathbf{x}_1^T (\mathbf{1} - I - A) \mathbf{x}_1 \\ &= \left(1 - \frac{k}{m}\right) \lambda_1 + \frac{k}{N} \lambda_0.\end{aligned}$$

Similarly, we can calculate $\mathbf{E}(\tilde{\lambda}_0)$ and have

$$\begin{aligned}\mathbf{E}(\tilde{\lambda}_1) &= \left(1 - \frac{k}{m}\right) \lambda_1 + \frac{k}{N} \lambda_0, \\ \mathbf{E}(\tilde{\lambda}_0) &= \left(1 - \frac{k}{N}\right) \lambda_1 + \frac{k}{m} \lambda_0.\end{aligned}\tag{5.11}$$

In estimating λ_1 , we substitute $\mathbf{E}(\tilde{\lambda}_1)$ and $\mathbf{E}(\tilde{\lambda}_0)$ with observed $\tilde{\lambda}_1$ and $\tilde{\lambda}_0$, and solving (5.11) for λ_0 and λ_1 , we can get the moment estimator of λ_1 is given by:

$$\lambda_1^* = \frac{(mk - mN)\tilde{\lambda}_1 + mk\tilde{\lambda}_0}{kN - mN + mk}.$$

□

This result is significant since λ_1 is closely related with many real space topological features, such as the maximum degree, chromatic number, clique number, and extend of branching of the graph [Cvetkovic et al., 1997]. Therefore, our algorithm determines r such that the difference between the reconstructed value $\hat{\lambda}_1$ and the estimated value λ_1^* is minimized. We expect that by preserving λ_1 in the reconstructed graph, many other features can also be well reconstructed.

The circled points in Figure 5.5 plot the r value chosen by our method and the corresponding values for the four features. For λ_1 shown in Figure 5.5(a), the reconstructed value is close to original value, indicating that the estimator shown in (5.9) accurately matches the original λ_1 . For other features, the chosen r value may not

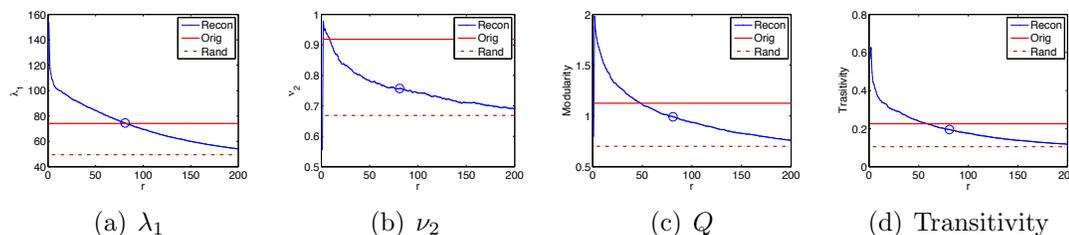


Figure 5.5: Original, randomized and reconstructed features for polblog network, r varies from 1 to 200, $k = 0.4m$

be the optimal for those features. However, the reconstructed features are closer to the original value than the randomized one when r is chosen by our method.

5.5 Empirical Evaluation

In addition to *polblogs* network, we use two network data sets (*polbooks*, *Enron*) in our evaluation. The *polbooks* network contains 105 nodes and 441 edges. Nodes represent books about US politics sold by the online bookseller Amazon.com, and edges represent frequent co-purchasing of books by the same buyers.¹ The *Enron* network was built from email corpus of a real organization over the course covering a 3 years period. We used a pre-processed version of the dataset provided by [Shetty and Adibi, 2004]. This data set contains 252,759 emails from 151 Enron employees, mainly senior managers. We regard there is an edge between node i and j if there are at least 5 emails sent between i and j , which results in 869 edges. The numbers of nodes and edges for three networks are shown in the first row of Table 5.2.

5.5.1 Feature Reconstruction

We focus on four topological features (λ_1 , ν_2 , Q , and C) in our evaluation. For each network data set, we first calculate feature values of the original graph and show them in Table 5.2. We randomize each network data with noise level $\frac{k}{m} = 0.4$. We

¹*polbooks* and *polblogs* are available at <http://www-personal.umich.edu/~mejn/netdata/>.

then apply our low rank approximation based reconstruction algorithm on each randomized graph and calculate the reconstructed feature values from the reconstructed graph. The randomization and reconstruction process repeats 10 times. We report the average results of these 10 rounds in Table 5.2.

We can observe that perturbation with noise level $\frac{k}{m} = 0.4$ significantly changes the feature values in the randomized graphs. It indicates that edge randomization in general cannot well preserve the graph topological structure. However, for all four features on three network data sets, our reconstructed feature values are much closer to the original ones.

To evaluate accuracy of feature reconstruction, we use the following measure.

Definition 5.1: For a graph feature f , define reconstruction quality

$$S_f = 1 - \frac{|f(\hat{A}) - f(A)|}{|f(\tilde{A}) - f(A)|}.$$

$S_f \in (0, 1]$ indicates that the reconstructed feature is closer to the original feature value than the feature value directly calculated from the randomized graph. The larger S_f is, the better the feature is reconstructed. $S_f = 1$ if and only if $f(\hat{A}) = f(A)$, and S_f is close to 1 if $f(\hat{A}) \approx f(A)$.

Table 5.2 shows the reconstruction quality S_f for these four features on three networks ($k = 0.4m$). We can see that all S_f values are above 0.22 and some S_f values are even close to 1, indicating that the majority of topological structure of the original graph has been reconstructed. We also notice that λ_1 is better reconstructed than the other three features. This is because we use the estimate of λ_1 as our target function when we determine r .

Table 5.2: The reconstructed features for three data sets ($k = 0.4m$)

	<i>polbooks</i> (105, 441)				<i>Enron</i> (151, 869)				<i>polblogs</i> (1222, 16714)			
	orig	rand	recon	S_f	orig	rand	recon	S_f	orig	rand	recon	S_f
λ_1	11.9	9.95	12.62	0.65	17.8	14.3	18.3	0.87	74.1	49.5	74.5	0.98
ν_2	0.96	0.72	0.77	0.22	0.89	0.65	0.80	0.63	0.92	0.67	0.76	0.35
Q	0.70	0.45	0.56	0.45	0.56	0.38	0.56	1.00	1.13	0.70	0.99	0.69
C	0.35	0.15	0.20	0.27	0.34	0.15	0.28	0.65	0.23	0.11	0.20	0.75

Effect of Noise Level

In this experiment, we evaluate how the reconstruction accuracy of features is affected by the magnitude of noise. We set noise level $\frac{k}{m} = 0.2, 0.4, 0.6, 0.8$. We report the feature values of the original data sets ($f(A)$), the randomized feature values under different noise levels ($f(\tilde{A})$), and the reconstructed feature values using our algorithm ($f(\hat{A})$) in Table 5.3.

For all features, the difference between $f(\tilde{A})$ and $f(A)$ increases as the magnitude of noise increases. For example, $\tilde{\lambda}_1$ is reduced approximately by half from the original value when $k = 0.6m$ for the *polblogs* network. After reconstruction, all reconstructed feature values are much more accurate than those feature values calculated from randomized graphs. For example, even under noise $\frac{k}{m} = 0.6$, our reconstructed transitivity value (C) is 0.15, which is much closer to the original transitivity value (0.23) than the randomized transitivity value (0.06). This result shows that our low rank approximation based reconstruction method can effectively filter out the noise and preserve the topological structure. We can also observe that the difference between $f(\hat{A})$ and $f(A)$ increases when the magnitude of noise increases, indicating that larger noise causes more loss of feature reconstruction quality. For example, the reconstructed transitivity value decreases to 0.09 under noise level $\frac{k}{m} = 0.8$, but it is

still better than the randomized transitivity value (0.03).

Table 5.3: Reconstruction quality for *polblogs* network at different noise levels

	λ_1 (74.08)		ν_2 (0.92)		Q (1.13)		C (0.23)	
$\frac{k}{m}$	rand	recon	rand	recon	rand	recon	rand	recon
0.2	61.43	75.83	0.77	0.84	0.90	1.11	0.16	0.22
0.4	49.38	74.28	0.66	0.75	0.69	0.98	0.10	0.19
0.6	38.39	71.35	0.54	0.62	0.47	0.78	0.06	0.15
0.8	30.56	60.74	0.40	0.48	0.27	0.50	0.03	0.09

5.5.2 Privacy

One question here is that whether attackers can exploit the reconstructed graph \hat{A} to breach the link privacy. If \hat{A} is similar to A at the entry level, attackers may simply use the value of \hat{a}_{ij} as a guess of the original value a_{ij} (the sensitive link between node i and j). If \hat{A} well matches A at the individual entry level, attackers have high confidence about the existence of the true link between node i and j based on the reconstructed \hat{a}_{ij} .

To measure the average disclosure risk of all link entries, we use the normalized Frobenius distance defined as

$$d(\hat{A}, A) = \frac{\|\hat{A} - A\|_F^2}{4m}.$$

It is easy to verify that $1 - d(\hat{A}, A) = |\hat{E} \cap E|/|\hat{E}|$. In other words, the larger $d(\hat{A}, A)$ is, the lower the disclosure risk is in the reconstructed graph. $d(\hat{A}, A) = 1$ if and only if no edge from the original graph appears in the reconstructed graph. Similarly, we can measure the disclosure risk of the randomized graph as $d(\tilde{A}, A) \equiv \frac{k}{m}$.

Figure 5.6 shows how $d(\hat{A}, A)$ for the three networks changes for different choices of r . We randomize each network data by the noise $k = 0.4m$. For each r , we derive

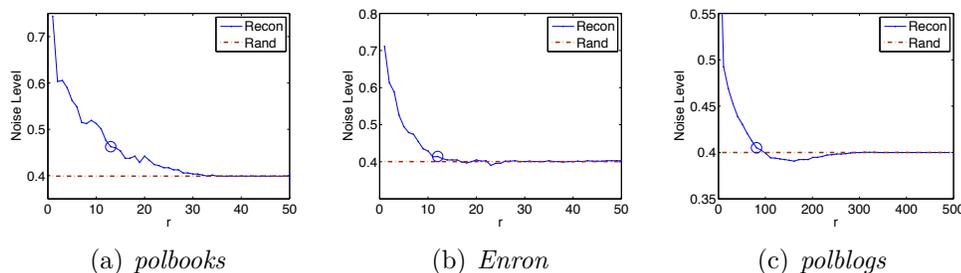


Figure 5.6: $d(\hat{A}, A)$ for the three networks, as r varies ($k = 0.4m$)

the reconstructed graph \hat{A} by discretizing \tilde{A}_r and calculate the normalized distance $d(\hat{A}, A)$. The circled points plots the $d(\hat{A}, A)$ value when r is chosen by our method. As r approaches n , $d(\hat{A}, A)$ thus converges to k/m because \hat{A} approaches \tilde{A} .

One surprising observation is that for both *polbooks* and *Enron* the normalized distance of the reconstructed graph ($d(\hat{A}, A)$) is always above that of the randomized graph ($d(\tilde{A}, A)$) no matter how we choose r as shown in Figure 5.6(a) and 5.6(b). This indicates that the reconstructed graph does not incur any further privacy disclosure than the randomized graph. In Section 5.5.1, we know that the features can be well reconstructed. This is because the risk of privacy disclosure depends on the extent to which \hat{A} matches A individually, while most topological features are overall measures of the graph. It is possible that two graphs have close topological structures but are very different at the individual level, and an accurate reconstruction of features does not necessarily lead to an accurate reconstruction of Frobenius distance. Note that \hat{A} is reconstructed such that it preserves the leading eigenvalue and eigenvectors of A . Therefore, strong structure, which is reflected by the leading eigen-pairs, is preserved in the reconstructed features; and weaker structure indicated by remaining eigen-pairs are neglected along with the noise. Therefore, the reconstruction method

can approximate many original topological features at the global level. However, at the individual level, the neglected eigenvalues and eigenvalues can cause many false edges, and the Frobenius norm distance, which accumulates the difference of each entry, can be very large.

However, for *polblogs*, as shown in Figure 5.6(c), we can observe the normalized distance of the reconstructed graph ($d(\hat{A}, A)$) is a little below that of the randomized graph ($d(\tilde{A}, A)$) for some choices of r . In other words, the reconstructed graph can incur some additional privacy disclosure risks. In the next section, we further investigate what type of graphs may incur additional privacy disclosure risks due to reconstruction.

5.6 Reconstruction Accuracy on Low Rank Graphs

The phenomenon shown in Section 5.5.2 is very different from that in the numerical setting. More accurate individual data can be recovered from the randomized numerical data using those point-wise data reconstruction methods based on low rank approximation [Guo et al., 2008; Huang et al., 2005], which jeopardizes data privacy at the individual level.

Our intuition is that there usually exist strong correlations among attributes in the numerical data and the number of attributes is much smaller than the number of tuples. Hence the numerical data U (or its covariance $\text{Cov}(U)$) has a low rank. On the contrary, for most real social networks, their adjacency matrices have very high ranks. For example, all three networks used in our paper have almost full ranks. Our conjecture is that for social networks with low ranks or with a small number of dominant eigenvalues the reconstructed graph can also be close to the original one at

the individual entry level.

The difference between the reconstructed graph and the original graph can be divided into three components:

$$\begin{aligned} \|A - \hat{A}\|_F &= \|(A - A_r) + (A_r - \tilde{A}_r) + (\tilde{A}_r - \hat{A})\|_F \\ &\leq \|A - A_r\|_F + \|A_r - \tilde{A}_r\|_F + \|\tilde{A}_r - \hat{A}\|_F. \end{aligned} \quad (5.12)$$

$\|A - A_r\|_F$ denotes the low rank approximation error that is determined by those excluded non-significant eigen-pairs; $\|A_r - \tilde{A}_r\|_F$ denotes the randomization error that is determined by the noise added in the subspace spanned by the first r eigenvectors; and $\|\tilde{A}_r - \hat{A}\|_F$ denotes the discretization error when we convert the real matrix \tilde{A}_r to the 0-1 matrix \hat{A} . To decrease $\|A - A_r\|_F$, we tend to choose a large r value. However, a large r value introduces more noise in the projected spectral space, increasing the randomization error $\|A_r - \tilde{A}_r\|_F$.

Hence, if a graph A can be well approximated by A_r with a small r value, both the low rank approximation error ($\|A - A_r\|_F$) and the randomization error ($\|A_r - \tilde{A}_r\|_F$) could be small. In this case, $\tilde{A}_r \approx A_r \approx A$, and \tilde{A}_r is already close to a 0-1 matrix, which then further reduces the discretization error $\|\tilde{A}_r - \hat{A}\|_F$.

For three network data sets used in our paper, we can derive their minimum r values such that $\frac{\|A - A_r\|_F^2}{\|A\|_F^2} \leq \tau$. When $\tau = 0.05$, we have $r = 54$ ($0.51n$) for *polbooks*, $r = 64$ ($0.42n$) for *Enron*, and $r = 348$ ($0.28n$) for *polblogs* network. Since all r values are large, the difference between the reconstructed graph and the original graph at the individual level ($\|A - \hat{A}\|_F$) is still significant, indicating the individual privacy is well protected in the reconstructed graph. However, the feature values can still be

well reconstructed. This is because those non-significant eigen-pairs do not contribute much to the global topological structure although they may significantly affect the Frobenius distance.

To verify our proposition, we construct a series of synthetic graphs

H_t ($t = 2, 5, 10, 50, 100, 200$) from the *polblog* network. We first calculate $A_t = \sum_{i=1}^t \lambda_i \mathbf{x}_i \mathbf{x}_i^T$ and regard its discretized version (using (5.2)) as H_t . We expect that these synthetic graphs H_t have a small number of dominant eigen-pairs. When $\tau = 0.05$, their minimum r values are listed in Table 5.4. For example, for graph H_2 , the number of dominant eigen-pairs is 16, which is much less than that of the original graph A . As a result, when we apply our low rank approximation based reconstruction algorithm on H_2 , the normalized distance is only 0.05, indicating that 95% of original edges are recovered in the reconstructed graph. We can also observe that as t increases, the number of dominant eigen-pairs also increases, and the reconstruction accuracy at the individual entry level decreases. For example, when $t = 200$, the normalized distance is 0.39, which is approximately equal to that of the randomized graph.

Table 5.4: Normalized Frobenius distance of reconstruction for the synthetic graphs from *polblogs* ($k = 0.4m$)

	H_2	H_5	H_{10}	H_{50}	H_{100}	H_{200}	A
min r	16	54	95	179	231	299	348
$d(\hat{H}_t, H_t)$	0.05	0.10	0.15	0.25	0.32	0.39	0.40

Figure 5.7 shows the normalized distance between the reconstructed graph \hat{H}_t and the original graph H_t for different choices of r . The circled points represent the distance values when r is chosen via our method. We can see that, for graphs H_5 ,

H_5 and H_{100} , the normalized distance values ($d(\hat{H}_t, H_t)$) are smaller than that of the randomized graph (k/m) for the majority r values. In particular, the normalized distance values on H_5 could reach as low as 0.1. As t increases, the curve of the normalized distance values on H_t approaches the curve of the original graph A , as shown in Figure 5.7. This phenomenon supports our conjecture: for those graphs with a small number of dominant eigen-pairs, reconstruction can accurately recover the original individual entries, which may seriously jeopardize data privacy.

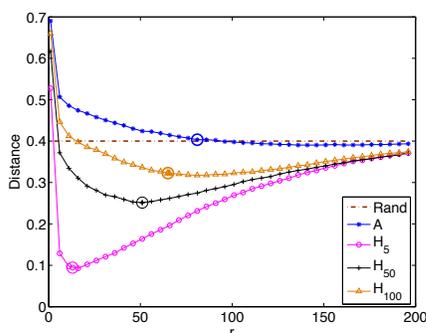


Figure 5.7: The normalized Frobenius distance of the synthetic graphs

We also calculate the feature values of the reconstructed graphs ($f(\hat{H}_t)$) and compare them with the original feature values ($f(H_t)$). Table 5.5 shows the feature reconstruction qualities (S_f) for different features. For all features, S_f values are also higher for those synthetic graphs constructed using small t values. This is because we can have an accurate reconstruction on the individual entry level for those graphs, and hence many global features can be accurately reconstructed. Note that, as shown for the three real networks, the inverse direction is generally not guaranteed. In summary, the dominance of the leading eigen-pairs of a graph plays an important role in reconstructing individual entries as well as global features.

Table 5.5: Feature reconstruction quality of the synthetic graphs from *polblogs* ($k = 0.4m$)

	H_2	H_5	H_{10}	H_{50}	H_{100}	H_{200}	A
S_{λ_1}	1.00	1.00	0.99	0.99	0.98	1.00	1.00
S_{ν_2}	0.99	0.96	0.89	0.67	0.54	0.43	0.35
S_Q	1.00	0.98	0.95	0.83	0.73	0.69	0.66
S_C	0.98	1.00	0.98	0.95	0.83	0.78	0.73

5.7 Summary

In this chapter, we have presented a low rank approximation based reconstruction algorithm, which can well recover feature values from the randomized network data. We have shown the close relationship between graph topological structure and spectral spaces determined by eigen-pairs of the adjacency matrix. We have also presented a novel solution to determine the optimal rank r in reconstruction. Our empirical evaluation results showed that accurate feature values can still be recovered from the randomized graphs even with the large magnitude of noise. One surprising finding is that, for most social networks, the reconstructed networks do not incur further disclosure risks of individual privacy than the released randomized graphs. Our investigation showed that only networks with low ranks or a small number of dominant eigenvalues may incur further privacy disclosure due to reconstruction.

This work is published in 2010 SIAM International Conference on Data Mining[Wu et al., 2010]

CHAPTER 6: CONCLUSIONS AND FUTURE WORK

In this dissertation, we studied the spectral space of the adjacency matrix for different types of social networks. By describing the observed graph as a perturbation from some ideal graph model, we theoretically quantified the effect of noise edges to the adjacency eigenvalues and eigenvectors. We conducted comprehensive evaluations on both real-world and synthetic graphs to verify our theoretical results. Our study showed that the eigenvalues and eigenvectors of the adjacency matrix reflect various aspects of the graph topology, including those global structures such as communities as well as those hidden subtle anomalies. While the Laplacian and the normal matrices were well studied in spectral graph analysis, the adjacency matrix attracted less attention in social network research. In our studies, we found that the adjacency spectral space is more robust under moderate noise than the Laplacian or the normal spectral spaces. Leveraging these theoretical results, we developed algorithms for graph partition, anomaly detection, and graph reconstruction.

We highly leveraged theoretical results from the matrix perturbation theory, which allows us to derive explicit formulas to approximate the eigenvalues and eigenvectors using the perturbation matrix and the original eigenvalues and eigenvectors of the ideal cases. The eigenvector entries of the adjacency matrix associated to the i -th node represent the coordinate in the high dimensional space spanned by the eigenvectors.

We demonstrated that the nodes' positions in the spectral space, e.g., distance to the origin or distance/angles to the coordinates of other nodes, reflect the roles of these nodes in the social network. By approximating the eigenvectors of the observed graphs, we were able to quantify the movement of the nodes' spectral coordinates from the ideal case, and thus revealed the topological features of a real-world graph in its adjacency spectral space.

For the unsigned graph, the ideal case is a graph with k disconnected communities. Its adjacency matrix is a block-wise diagonal matrix with k blocks. Real-world graphs are perturbed variants from this ideal case with moderate inter-community edges added. Recent work [Prakash et al., 2010; Ying and Wu, 2009c] showed the observations of orthogonal line pattern in the adjacency eigenspace spanned by principal eigenvectors and the authors suggested that such a pattern is associated with the presence of a clear community structure in the graph. However, they did not show when and why such a pattern exists. Our theoretical results proved the line orthogonality pattern in the adjacency eigenspace. Specifically we demonstrated the following phenomena:

1. In the spectral space spanned by the principal eigenvectors, the k communities form k straight lines starting from the origin, and each line represents one community.
2. The k lines are (approximately) orthogonal to each other, and all the lines have a rotation from the canonical axes caused by inter-community edges.
3. Nodes with no inter-community edges lie on the lines, whereas nodes with some

inter-community edges deviate from the original lines and move closer to the other lines (the distance of the deviation depends on the inter-community connections).

We further gave the explicit formula for the orthogonal line rotation and nodes' deviation caused by inter-community edges.

Following a similar methodology, we extended the study to signed graphs with both positive and negative relations. We started with the typical signed graphs in sociological study [Davis, 1967; Hage and Harary, 1983; Inohara, 2002]: k -balanced graphs. Such signed graphs contain the following assumption: individuals tend to have the positive relation within the same communities and the negative relation with individuals from other communities. Following the same assumption, we focus on three typical types of clusterable signed graphs and studied them respectively.

- k -Balanced Signed Graphs: These signed graphs have positive inner-community edges and negative inter-community edges. We demonstrated the following phenomena of it:
 1. With moderate negative inter-community edges, k -balanced signed graphs keep the orthogonal line pattern but all the lines have an opposite direction of rotation from the canonical axes compared with unsigned graphs. Nodes with no inter-community edges stay on the lines. Nodes with inter-community edges deviate from the original lines and move further from the other lines.
 2. With dense negative inter-community edges, k -balanced signed graphs are still distinguishable in the adjacency spectral space though the line pattern

is lost. This is very different from unsigned graphs where the communities tend to mix together in the spectral space when inter-community edges are dense.

3. With further adding moderate negative inner-community edges and positive inter-community edges, k -balanced signed graphs are still distinguishable. The majority of nodes in one community lie on the positive part of the line, while a small number of nodes may lie on the negative part due to negative inner-community connections.

- Signed Graphs with Dominated Positive Inner-community Edges: These signed graphs have dominated positive inner-community edges. We showed that the negative inter-community edges offset the rotation effect of the positive inter-community edges. Specially, with dense positive and negative inter-community edges of equal magnitude, the graph still keeps a similar community structure as that decided by the inner-community edges. The communities exhibit as orthogonal clusters in the spectral space spanned by the principal eigenvectors.
- Partite-dominated Signed Graphs: These signed graphs have dominated negative inter-community edges and no/few inner-community edges. We showed that the communities still exhibit as orthogonal clusters in the spectral space spanned by principal eigenvectors. The largest eigenvalue for the k -partite graph has an opposite sign with the rest eigenvalues of the principal eigenvectors.

We can see that communities in both unsigned and signed graphs all exhibit as orthogonal lines/clusters in the spectral space spanned by the principal eigenvectors

with largest eigenvalues in magnitude. Based on our theoretical results, we developed an effective algorithm, *UniAdjCluster*, to partition both unsigned and signed graphs. In our unified methodology, the algorithm is able to discover various structural patterns, such as the community or multi-partite structure. In the algorithm, we leveraged the orthogonality of those clusters in the adjacency spectral space and projected them on to a unit sphere before clustering. In comprehensive evaluations, *UniAdjCluster* achieved higher accuracies on graph partition than those based on the Laplacian or the normal spectral spaces. The complexity of our algorithm is mainly determined by the complexity of the eigen-decomposition of the adjacency matrix. Note that the adjacency matrix is usually sparse and has large eigenvalues, which leads to some efficient algorithms such as Lanczos algorithm to obtain its eigenvalues and eigenvectors.

While the principal eigenvectors capture the global structures as communities, we also showed the efficacy of using minor eigenvectors of a graph's adjacency matrix to detect subtle anomalies embedded in the background. Under the assumption of the *Erdos-Renyi* random graph model, we derived the explicit formula about how signal entries and background entries of certain eigenvector are distributed. Our results showed that the gap between signal entries and background entries in the minor eigenvector is larger than that in the principal eigenvector for subtle signals, which provides a theoretical explanation behind the utility of the approach. We further derived the detectability bound for the *Erdos-Renyi* random graph model. Although our theoretical analysis mainly focused on the *Erdos-Renyi* random graph model, we believe theoretical results based on other graph models could be developed.

We also showed the use of kurtosis to filter out the eigenvectors that could capture the signals. Our approach removed the assumption in [Miller et al., 2010] that the background graph is generated using a specific random graph model with prior-known parameters. Empirical evaluations on both synthetic data and real social networks showed effectiveness of our approach to detecting subtle signals.

From our study, we found that eigenvectors of the adjacency matrix capture the structure of the graph and are quite stable under the perturbation. We utilized these features of the adjacency spectral space to do privacy preserving graph reconstruction. We showed the use of low rank approximation to reconstruct the graph topology from the randomized network and presented a novel solution to determine the (approximate) optimal rank, a key parameter in our reconstruction algorithm. Our empirical evaluation results showed that accurate feature values can still be recovered from the randomized graphs even with the large magnitude of noise. One surprising finding was that, for many social networks, the reconstructed networks do not incur further disclosure risks of individual privacy than the released randomized graphs. This is very different from the numerical data setting. Our further investigation showed that only networks with low ranks or a small number of dominant eigenvalues may incur further privacy disclosure due to reconstruction.

Future Work

The adjacency matrix has many good properties in spectral analysis. It has long been underestimated due to the lack of study of the fundamental properties. Our work mainly contributes to make up this part so that the application of the adjacency matrix would be widely explored. With the rapid development of online social

networks such as Facebook, Twitter and Sina Weibo, we are able to obtain many large real networks. It will be interesting to apply adjacency spectral analysis on these real world data to do graph partition and signal detection. The eigen-decomposition of large graphs may require extraordinary computing time and we will explore how to extend our algorithms to the big data. We will conduct complete comparisons with other recently developed algorithms with *UniAdjCluster*.

In signal detection, we currently assume the embedded signal is highly correlated with a single eigenvector. It is interesting to see whether a much weaker signal that can not be detected by a single eigenvector could be detected by combining multiple eigenvectors together. We are also interested in exploring how the approach works for various types of anomalies. We will also compare the approach with other anomaly detection approaches in practice.

We believe that spectral perturbation is also promising in privacy preserving data mining. In privacy preserving graph reconstruction, we are interested in comparing with other various edge based randomization strategies[Wu et al., 2010]. For example, we will explore whether a similar low rank approximation based reconstruction method can be derived for the *Random Switch* strategy. It is also our conjecture that it is very hard, if not impossible, to figure out reconstruction methods on the released randomized data using K -anonymity schemes. This is because in K -anonymity based modification schemes, modified edge entries are not randomly chosen. For example, the K -degree scheme[Liu and Terzi, 2008] examined the degree sequence of nodes and chose a subset of nodes (that violates the K -degree anonymity property) for edge modification. It will be interesting to compare various randomization strategies in

terms of the tradeoff between privacy and utility. We will also explore differential privacy preserving spectral graph analysis. Differential privacy [Dwork et al., 2006] is a paradigm of post-processing the output of queries such that the inclusion or exclusion of a single individual from the dataset makes no statistical difference to the output results. Differential privacy is usually achieved by directly adding calibrated Laplace noise on the output of the computation. It will be interesting to utilize the spectral properties of the adjacency matrix to develop release mechanisms of graph analysis results that satisfy the differential privacy conditions. We have some preliminary findings as shown in [Wang et al., 2013] and we will continue to explore this topic.

REFERENCES

- ADAMIC, L. AND GLANCE, N. 2005. The political blogosphere and the 2004 us election: divided they blog. In *Proceedings of the WWW Workshop on the Weblogging Ecosystem*.
- BACKSTROM, L., DWORK, C., AND KLEINBERG, J. 2007. Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. In *Proceedings of the international conference on World Wide Web(WWW)*. ACM Press, New York, NY, USA, 181–190.
- BELKIN, M. AND NIYOGI, P. 2002. Laplacian eigenmaps and spectral techniques for embedding and clustering. *Advances in neural information processing systems 1*, 585–592.
- BHAGAT, S., CORMODE, G., KRISHNAMURTHY, B., AND SRIVASTAVA, D. 2009. Class-based graph anaonymization for social network data. In *Proceedings of the International Conference on Very Large Data Base*.
- CAMPAN, A. AND TRUTA, T. M. 2008. A clustering approach for data and structural anonymity in social networks. In *In Privacy, Security, and Trust in KDD Workshop (PinKDD)*.
- CHAN, P., SCHLAG, M., AND ZIEN, J. 1993. Spectral k-way ratio-cut partitioning and clustering. In *Proceedings of the 30th international Design Automation Conference*. ACM, 749–754.
- CHUNG, F., LU, L., AND VU, V. 2003. Eigenvalues of random power law graphs. *Annals of Combinatorics 7*, 21–33.
- CLAUSET, A., NEWMAN, M., AND MOORE, C. 2004. Finding community structure in very large networks. *Physical Review E 70*, 066111.
- CORMODE, G., SRIVASTAVA, D., YU, T., AND ZHANG, Q. 2008. Anonymizing bipartite graph data using safe groupings. In *Proceedings of the VLDB Endowment*. 833–844.
- COSTA, L. F., RODRIGUES, F. A., TRAVIESO, G., AND BOAS, P. R. V. 2007. Characterization of complex networks: A survey of measurements. *Advances In Physics 56*, 167–242.
- CVETKOVIC, D., ROWLINSON, P., AND SIMIC, S. 1997. *Eigenspaces of Graphs*. Cambridge University Press.
- DAVIES, D. AND BOULDIN, D. 1979. A cluster separation measure. *IEEE Tran. on Pattern Analysis and Machine Intelligence 1*, 224–227.

- DAVIS, J. A. 1967. Clustering and structural balance in graphs. *Human Relations* 20, 181–187.
- DING, C., HE, X., ZHA, H., GU, M., AND SIMON, H. 2001. A min-max cut algorithm for graph partitioning and data clustering. In *Proceedings of IEEE International Conference on Data Mining(ICDM)*.
- DWORK, C., MCSHERRY, F., NISSIM, K., AND SMITH, A. 2006. Calibrating noise to sensitivity in private data analysis. In *In Proceedings of the 3rd Theory of Cryptography Conference*. Springer, 265–284.
- EBERLE, W. AND HOLDER, L. B. 2007. Anomaly detection in data represented as graphs. *Intell. Data Anal.* 11, 6, 663–689.
- ERDŐS, P. AND RÉNYI, A. 1959. On random graphs. *Publicationes Mathematicae Debrecen* 6, 290–297.
- FREDI, Z. AND KOMLS, J. 1981. The eigenvalues of random symmetric matrices. *Combinatorica* 1, 233–241.
- GHOSN, F., PALMER, G., AND BREMER, S. 2004. The MID3 Data Set, 1993–2001: Procedures, Coding Rules, and Description. *Conflict Management and Peace Science* 21, 2, 133–154.
- GIRVAN, M. AND NEWMAN, M. 2002. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences* 99, 12 (June), 7821–7826.
- GOLUB, G. H. AND VAN LOAN, C. F. 1996. *Matrix Computations*. The Johns Hopkins University Press.
- GUO, S., WU, X., AND LI, Y. 2008. Determining error bounds for spectral filtering based reconstruction methods in privacy preserving data mining. *Knowl. Inf. Syst.* 17, 2, 217–240.
- HAGE, P. AND HARARY, F. 1983. Structural models in anthropology. *Cambridge University Press*, 56–60.
- HAGEN, L. AND KAHNG, B. 1992. New spectral methods for ratio cut partitioning and clustering. *IEEE Transactions on computer-aided design* 11, 9.
- HANHIJARVI, S., GARRIGA, G. C., AND PUOLAMAKI, K. 2009. Randomization techniques for graphs. In *Proceedings of the SIAM International Conference on Data Mining(SDM)*.
- HAY, M., MIKLAU, G., JENSEN, D., TOWSELY, D., AND WEIS, P. 2008. Resisting structural re-identification in anonymized social networks. In *Proceedings of the VLDB Endowment*.

- HAY, M., MIKLAU, G., JENSEN, D., WEIS, P., AND SRIVASTAVA, S. 2007. Anonymizing social networks. *University of Massachusetts Technical Report 07-19*.
- HOU, Y., LI, J., AND PAN, Y. 2003. On the laplacian eigenvalues of signed graphs. *Linear and Multilinear Algebra* 51, 1, 21–30.
- HUANG, L., YAN, D., JORDAN, M., AND TAFT, N. 2008. Spectral clustering with perturbed data. *Advances in Neural Information Processing Systems (NIPS)*.
- HUANG, Z., DU, W., AND CHEN, B. 2005. Deriving private information from randomized data. In *Proceedings of the ACM SIGMOD Conference on Management of Data*.
- HUNTINGTON, S. 1997. *The clash of civilizations and the remaking of world order*. Pocket Books.
- INOHARA, T. 2002. Characterization of clusterability of signed graph in terms of newcombs balance of sentiments. *Applied Mathematics and Computation* 133, 93–104.
- JOHN R. GILBERT, G. L. M. AND TENG, S.-H. 1998. Geometric mesh partitioning: Implementation and experiments. *SIAM J. Scientific Computing* 19, 2091–2110.
- KARGUPTA, H., DATTA, S., WANG, Q., AND SIVAKUMAR, K. 2003. On the privacy preserving properties of random data perturbation techniques. In *Proceedings of IEEE International Conference on Data Mining(ICDM)*. 99–106.
- KARYPIS, G. AND KUMAR, V. 1996. Parallel multilevel k-way partitioning scheme for irregular graphs. In *Proceedings of the 1996 ACM/IEEE Conference on Supercomputing, 1996*. 35–35.
- KEMPE, D., KLEINBERG, J. M., AND TARDOS, É. 2003. Maximizing the spread of influence through a social network. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD)*. 137–146.
- KREBS, V. 2006. <http://www.orgnet.com/>.
- KUNEGIS, J., SCHMIDT, S., LOMMATZSCH, A., LERNER, J., LUCA, E. W. D., AND ALBAYRAK, S. 2010. Spectral analysis of signed graphs for clustering, prediction and visualization. In *Proceedings of the SIAM International Conference on Data Mining(SDM)*. 559–570.
- LIU, K. AND TERZI, E. 2008. Towards identity anonymization on graphs. In *Proceedings of the ACM SIGMOD Conference*. ACM Press.
- MASSA, P. AND AVESANI, P. 2004. Trust-aware collaborative filtering for recommender systems. *Lecture Notes in Computer Science - International Conference on Cooperative Information Systems (CoopIS)* 3290, 492–508.

- MILLER, B., BLISS, N., AND WOLFE, P. 2010. Subgraph detection using eigenvector l_1 norms. In *Advances in Neural Information Processing Systems 23*. 1633–1641.
- MITRA, P. 2009. Entrywise bounds for eigenvectors of random graphs. *The electronic journal of combinatorics* 16, R131.
- NEWMAN, M. 2003. The structure and function of complex networks. *SIAM Review* 45, 167.
- NEWMAN, M. 2006. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences* 103, 23, 8577–8582.
- NG, A., JORDAN, M., AND WEISS, Y. 2001. On Spectral Clustering: Analysis and an Algorithm. In *Proceedings of Advances in Neural Information Processing Systems(NIPS)*. 849–856.
- NOBLE, C. C. AND COOK, D. J. 2003. Graph-based anomaly detection. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD)*. 631–636.
- POTHEN, A., SIMON, H., AND LIOU, K. 1990. Partitioning sparse matrices with eigenvectors of graphs. *SIAM Journal on Matrix Analysis and Applications* 11, 430.
- PRAKASH, B., SRIDHARAN, A., SESHADRI, M., MACHIRAJU, S., AND FALOUTSOS, C. 2010. EigenSpokes: Surprising Patterns and Scalable Community Chipping in Large Graphs. *Advances in Knowledge Discovery and Data Mining*, 435–448.
- SEARY, A. AND RICHARDS, W. 2003. Spectral methods for analyzing and visualizing networks: an introduction. *National Research Council, Dynamic Social Network Modelling and Analysis: Workshop Summary and Papers*, 209–228.
- SHETTY, J. AND ADIBI, J. 2004. The Enron email dataset database schema and brief statistical report. *Information Sciences Institute Technical Report, University of Southern California*.
- SHI, J. AND MALIK, J. 2000. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence* 22, 8, 888–905.
- SHIGA, M., TAKIGAWA, I., AND MAMITSUKA, H. 2007. A spectral clustering approach to optimally combining numerical vectors with a modular network. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining(KDD)*. 647–656.
- STEWART, G. W. AND SUN, J. 1990. *Matrix Perturbation Theory*. Academic Press.
- STROGATZ, S. 2001. Exploring complex networks. *Nature* 410, 268–276.
- TRAAG, V. AND BRUGGEMAN, J. 2009. Community detection in networks with positive and negative links. *Physics Review E* 80.

- VISWANATH, B., MISLOVE, A., CHA, M., AND GUMMADI, K. P. 2009. On the Evolution of User Interaction in Facebook. In *Proceedings of ACM SIGCOMM Workshop on Social Networks (WOSN)*.
- WAKITA, K. AND TSURUMI, T. 2007. Finding community structure in mega-scale social networks: [extended abstract]. In *Proceedings of the international conference on World Wide Web(WWW)*. 1275–1276.
- WANG, Y., CHAKRABARTI, D., WANG, C., AND FALOUTSOS, C. 2003. Epidemic spreading in real networks: An eigenvalue viewpoint. *Proceedings of the 22nd International Symposium on Reliable Distributed Systems*.
- WANG, Y., WU, X., AND WU, L. 2013. Differential privacy preserving spectral graph analysis. In *Proceedings of Pacific Asia Knowledge Discovery and Data Mining(PAKDD)*.
- WU, L., WU, X., LU, A., AND ZHOU, Z.-H. 2013. A spectral approach to detecting subtle anomalies in graphs. *Journal of Intelligent Information System(JIIS)*.
- WU, L., YING, X., AND WU, X. 2010. Reconstruction from randomized graph via low rank approximation. In *Proceedings of the SIAM International Conference on Data Mining(SDM)*. 60–71.
- WU, L., YING, X., WU, X., LU, A., AND ZHOU, Z.-H. 2011. Spectral analysis of k -balanced signed graphs. In *Proceedings of Pacific Asia Knowledge Discovery and Data Mining(PAKDD) (2)*. 1–12.
- WU, L., YING, X., WU, X., LU, A., AND ZHOU, Z.-H. 2012. Examining spectral space of complex networks with positive and negative links. *International Journal of Social Network Mining(IJSNM) 1, 1*, 91–111.
- WU, L., YING, X., WU, X., AND ZHOU, Z.-H. 2011. Line orthogonality in adjacency eigenspace with application to community partition. In *Proceedings of the International Joint Conference on Artificial Intelligence(IJCAI)*. 2349–2354.
- WU, X., YING, X., LIU, K., AND CHEN, L. 2010. A survey of privacy-preservation of graphs and social networks. In *Managing and Mining Graph Data*. 421–453.
- YING, X., WU, L., AND WU, X. 2011. A spectrum-based framework for quantifying randomness of social networks. *IEEE Transactions on Knowledge and Data Engineering 23, 12*, 1842–1856.
- YING, X. AND WU, X. 2008. Randomizing social networks: a spectrum preserving approach. In *Proceedings of the SIAM International Conference on Data Mining(SDM)*.
- YING, X. AND WU, X. 2009a. Graph generation with prescribed feature constraints. In *Proceedings of the SIAM International Conference on Data Mining(SDM)*.

- YING, X. AND WU, X. 2009b. On link privacy in randomizing social networks. In *Proceedings of Pacific Asia Knowledge Discovery and Data Mining(PAKDD)*.
- YING, X. AND WU, X. 2009c. On randomness measures for social networks. In *Proceedings of the SIAM International Conference on Data Mining(SDM)*.
- ZHELEVA, E. AND GETOOR, L. 2007. Preserving the privacy of sensitive relationships in graph data. In *In Privacy, Security, and Trust in KDD Workshop (PinKDD)*. 153–171.
- ZHOU, B. AND PEI, J. 2008. Preserving privacy in social networks against neighborhood attacks. In *Proceedings of the International Conference on Data Engineering (ICDE)*.
- ZOU, L., CHEN, L., AND ÖZSU, M. T. 2009. K-automorphism: A general framework for privacy preserving network publication. In *Proceedings of International Conference on Very Large Data Base*.