Network Structure Analysis based on Spectral Clustering: eigenspectrum augmentation

Zhongling Xu
School of Information Science and Technology
ShanghaiTech University
Shanghai, China
xuzhl2@shanghaitech.edu.cn

Abstract

In the past decades, spectral clustering(SC) has become one of the most effective clustering algorithms, which helps to understand and to visualize the functional characteristics of graphs. Spectral Graph Theory aims to analyze the "spectrum" of matrix representing a graph. Here "spectrum" refers to the eigenvectors of a graph (ordered by the magnitude/strength of their corresponding eigenvalues). In this project, we plan to apply spectral graph theory to clustering problems, which is of great significance in network structure analysis.

Index Terms

Network, Spectral Clustering, adjacency matrix, Laplacian matrix, structured graph

I. Introduction

A. Background

Graph theory plays a central role in data analysis with a variety of applications ranging from biology to social network. One of the most common and natural tasks in graph theory is cluster detection in which we identify the similarities within groups on a given network. In this task, our goal is to define a "good" partition of a graph and efficiently identify such a partition. Basically, a good partition should maximize the number of within-group connections and minimize between-group connections. One possible measurement for how well the graph is partitioned is the ratio cut of this partition. [**?**]
To find a partition satisfying this criterion, we need to perform spectral clustering. Generally, the spectral clustering algorithms mainly consist of three basic stages:

1) Pre-processing: Construct a matrix representation of the graph.
2) Decomposition: Compute eigenvalues and eigenvectors of the matrix, and then map each point to a lower-dimensional representation based on one or more eigenvectors.
3) Grouping: Assign points to two or more clusters, based on the new representation.

Although a general pipeline of implementing spectral clustering is given above, there can be quite a few extensions or alternatives with regard to the traditional method, and we will give a brief overview of some of them in later sections.
In this project, we first present an introduction of some preliminary theories and definitions related to spectral clustering including Laplacian matrix, cut, minimum-cut criterion, conductance, partitioning methods and so on. Then we collect some ideas in existing work and make criticism of their contributions. Finally, we expect to propose our own new contribution and get some deep insight into a specific scene which can be analyzed via matrix graph representation and spectral-related theory.

B. Notations and Definitions

1) **Graph notation**
Let $G = (V, E)$ be an undirected graph with vertex set $V = \{v_1, \dots, v_n\}$. Assume that the graph G is weighted such that each edge between two vertices v_i and v_j carries a non-negative weight $w_{ij} \geq 0$. The weighted adjacency matrix of the graph is denoted as $W = (w_{ij})_{i,j=1,\dots,n}$ and $w_{ij} = 0$ means that the vertices v_i and v_j are not connected. As G is undirected, we require that W should be symmetric ($w_{ij} = w_{ji}$). The degree of a vertex $v_i \in V$ is defined as

$$d_i = \sum_{j=1}^{n} w_{ij}$$

The above definition actually only sums over all vertices adjacent to v_i since for all other vertices v_j the weight w_{ij} is 0.
The degree matrix D is defined as the diagonal matrix with the degrees $d_1, \dots, d_n$ on the diagonal entries. Given a subset of vertices $A \subset V$, we denote its complement $V \backslash A$ by $\overline{A}$. We consider two ways of measuring the "size" of a subset $A \subset V$:

- $|A|$:= the number of vertices in A
- vol(A) := $\sum_i d_i$

Intuitively, $|A|$ measures the size of A by its number of vertices while vol(A) measures the size of A by the weights of its edges.

A subset $A \subset V$ of a graph is connected if any two vertices in A can be joined by a path such that all intermediate points also lie in A. A subset A is called a connected component if it is connected and if there are no connections between vertices in A and $\overline{A}$. The sets $A_1, \ldots, A_k$ form a partition of the graph if $A_i \cap A_j = \emptyset$ and $A_1 \cup \cdots \cup A_k = V$.

2) **Graph Laplacians and their basic properties**

The unnormalized graph Laplacian matrix is defiend as

$$L = D - W$$

We summarize some of the most important facts needed for spectral clustering as follows:

- For every vector $f \in \mathbb{R}^n$, we have $f'Lf = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2$.
- L is symmetric and positive semi-definite.
- The smallest eigenvalue of L is 0 and the corresponding eigenvector is the constant one vector $\mathbb{1}$.
- L has n non-negative, real-valued eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \ldots \lambda_n$.

Proposition (Number of connected components) Let G be an undirected graph with non-negative weights. Then the multiplicity k of the eigenvalue 0 of L equals the number of connected components $A_1, \ldots, A_k$ in the graph. The eigenspace of eigenvalue 0 is spanned by the indicator vectors $\mathbb{1}_{A_1}, \ldots, \mathbb{1}_{A_k}$ of those components.

3) **Graph cut point of view**

The intuition of clustering is to separate points in different groups according to their similarities. Therefore, we want to find a partition of the graph such that the edges between different groups have a very low weight (which means that points in different clusters are dissimilar from each other) and the edges within a group have high weight (which means that points within the same cluster are similar to each other).

For two disjoint subsets $A, B \subset V$, we define

$$cut(A, B) = \sum_{i \in A, j \in B} w_{ij}$$

This definition sums over sets of edges with one endpoint in each group. Our objective is to minimize weight of connections between two groups, which is finding

$$\arg\min_{A,B} \ cut(A, B)$$

However, some problems lie in this criterion: we only consider external cluster connections but we ignore internal cluster connectivity. Hence we introduce an alternative concept called *conductance*:

$$\phi(A, B) = \frac{cut(A, B)}{min(vol(A), vol(B))}$$

Conductance can be interpreted as the connectivity between groups relative to the density of each group. By intuition, it produces more balanced partitions since its definition takes into consideration both external and internal connections. Suppose we are further interested in finding a good k-way partition, we need to define the *ratio cut* of a given partition. Given a k-way partition of the vertices $\{V_i\}_{i=1}^k (\cup_{i=1}^k V_i = V)$, the ratio cut of this partition is defined to be

$$\text{RatioCut}(\{V_i\}_{i=1}^k) = \sum_{i=1}^{k} \frac{Cut(V_i, \overline{V_i})}{|V_i|}$$

The ratio cut measures the connections among the subgraphs normalized by the size of the subgraphs. The purpose of the normalization is to discourage unbalanced partitions. Finding a k-way partition that has the minimum ratio cut is presumed to be a NP-hard problem and spectral clustering is a natural relaxation to this NP-hard problem.

II. Overview of existing work

[1] is based on the two-step spectral clustering method widely used for graph partitioning, which consists of Laplacian eigenmap and a rounding step. The authors' goal is to study when is spectral clustering able to find the global solution to the minimum ratio cut problem. By combining two results, they give a condition under which spectral clustering is guaranteed to output the global solution to the minimum ratio cut problem. The two results are stated as follows:

1) A condition is given which naturally depends on the intra- and inter- cluster connectivities of a given partition under which the partition is the solution to the minimum ratio cut problem.
2) The authors developed a deterministic two-to-infinity norm perturbation bound for the invariant subspace of the graph Laplacian that corresponds to the k smallest eigenvalues.

[6] discusses some unresolved issues of traditional spectral clustering methods. First, there are a wide variety of algorithms that use the eigenvectors in slightly different ways. Second, many of these algorithms have no proof that they will actually compute a reasonable clustering. The authors present a simple spectral clustering algorithm and give conditions under which it can be expected to do well using tools from matrix perturbation theory. They also show some good experimental results on challenging clustering problems.

[4] criticizes the limitation of traditional objectives of graph clustering algorithms, which is to find clusters with low conductance. First, the traditional objectives are only applicable for undirected graphs. Second, they are also incapable to take the relationships between clusters into account. To overcome these shortcomings, the authors present a nearly-linear time algorithm for digraph (directed graph) clustering based on the Hermitian matrix representation of digraphs, and further show that the proposed algorithm can be implemented in sublinear time under reasonable assumptions. The main purpose of proposing the algorithm is to find *higher-order* structure among the clusters, which is characterised by the imbalance of the edge directions between clusters. To illustrate the significance of their theoretical work, the authors conduct extensive experiments on the UN Comtrade Dataset and the results demonstrate how the clusters (sets of countries) relate to each other with respect to their import and export records and also how these clusters evolve over time in accordance with known facts in international trade.

[5] discusses about sparsification and quantization of spectral clustering methods. First the related background is given: sparsifying, quantizing, and/or performing other entry-wise nonlinear operations can have numerous benefits such as speeding up iterative algorithms for core numerical linear algebra problems and providing nonlinear filters to design state-of-the-art neural network models. The authors exploit tools from random matrix theory to make precise statements about how the eigen spectrum of a matrix changes under nonlinear transformations. They show that very little change occurs in the informative eigenstructure even under drastic sparsification/quantization, and consequently that very little downstream performance loss occurs with very aggressively sparsified or quantized spectral clustering. Finally they illustrate how these results depend on the nonlinearity, characterize a phase transition beyond which spectral clustering becomes possible and show when such nonlinear transformations can introduce spurious non-informative eigenvectors.

Since there are few theoretical guarantee to recover the underlying partitions of the graph for general models, [2] presents a variant of spectral clustering, called l_1-spectral clustering, performed on a new random model closely related to stochastic block model. The goal of the proposed method is to promote a sparse eigenbasis solution of a l_1 minimization problem revealing the natural structure of the graph.

[7] focuses on the shortfall of previous studies that most of them only focus on spectral clustering tasks with a fixed task set, which cannot incorporate with a new spectral clustering task without accessing to previously learned tasks. The authors aim to explore the problem of spectral clustering in a lifelong machine learning framework, *i.e.*, Lifelong Spectral Clustering (L^2SC). The goal is to efficiently learn a model for a new spectral clustering task by selectively transferring previously accumulated experience from knowledge library. The knowledge library of L^2SC contains two components: 1)orthogonal basis library 2) feature embedding library. As a new spectral clustering task arrives, L^2SC firstly transfers knowledge from both basis library and feature library to obtain encoding matrix, and further redefines the library base over time to maximize performance across all the clustering tasks. Meanwhile, a general online update formulation is derived to alternatively update the basis library and feature library.

[3] considers spectral community detection in the regime of sparse networks with heterogeneous degree distributions. The authors demonstrate that a conveniently parametrized form of regularized Laplacian matrix can be used to perform spectral clustering in sparse networks, without suffering from its degree heterogeneity. Furthermore, they exhibit important connections between this proposed matrix and the now popular *non-backtracking* matrix, the *Bethe-Hessian* matrix, as well as the standard Laplacian matrix. Interestingly, the proposed improved parametrization inherently accounts for the hardness of the classification problem.

[8] concentrates on the idea of *data smoothness*, which refers to that if there exists a sudden change among different objects, they belong to different groups with high probability. The most widely used distance-based similarity for spectral clustering is not suitable for multi-scale data, as the distance varies a lot for clusters with different densities. The authors observe that in real-world scenarios, data in the same cluster tend to present in a smooth manner, which previous algorithms never take into account. Based on the observation, they propose a novel clustering algorithm, which considers the smoothness of data for the first time. The key idea is to cluster tiny clusters, whose centers constitute smooth graphs.

III. Criticism of the existing work

[1] gives theoretical proofs of the certification of the global minimum of ratio cut. The authors developed further conditions under which the performance of spectral clustering is guaranteed. Although the main results of this paper are based on

conclusions in previous work to a great extent, there are still some significant improvement with regard to the invariant subspace perturbation theory, together with rigorous and detailed derivation.
[6] was published in relatively early years and appears quite brief. However, this paper copes with a really general and fundamental aspect of spectral clustering, which is a simple yet feasible algorithm and the conditions under which it can give reasonable outputs. The experimental results are convincing and comprehensive. The authors lay a solid foundation for their successors and provide several meaningful conclusions including the general algorithm and the assumptions that will lead to expected clusterings.
[4] proposes a formulation of high novelty, which overcomes some limitation of traditional spectral clustering methods. The work consists of all-round contributions with regard to the proposed problem formulation including new objective functions, algorithm, analysis and well-designed experiments. Besides, the idea of finding higher-order information in clustering is considered to shed a light on various practical situations such as the analysis of oil trade chain and COVID-19 trend as mentioned in the paper.
[5] show the essential fact that very little change occurs in the informative eigenstructure even under drastic nonlinear operations such as sparsification and quantization. Since these nonlinear operations have great benefits for speeding up iterative algorithms and designing state-of-the-art neural network models, the fact discovered in the paper turns out to have great significance. Furthermore, the authors' work gives technical result characterizing the change in the eigenspectrum under some specific operations. Elaborate derivations and discussions, combined with empirical results on real-world datasets are also attached.
[2] presents a novel variation of spectral clustering algorithm based on promoting a sparse eigenvectors basis that provides information about the structure of the system observed. The paper has a well-organized and complete structure, which consists of traditional graph clustering method, the proposed l_1-spectral clustering algorithm, the applications, the robustness to perturbations and its performance on simulated and real-world datasets.
[7] studies how to add spectral clustering capability into original clustering system without damaging existing capabilities. The novelty of this work is mainly illustrated by the two library components, which play key roles in transferring knowledge in existing libraries to encode the coming spectral clustering task with encoding matrix. The proposed model has not only achieved effective experimental results, but also inspires more lifelong algorithms or models based on knowledge transferring.
[3] provides a critical analysis of the current state-of-the-art spectral algorithms for community detection in sparse graphs. The authors show deep insight into these algorithms in that the seemingly differing methods are actually all related and can be jointly optimized to tackle community detection in heterogeneous and sparse graphs. This paper offers a comprehensive yet conclusive view of progress in the field of spectral clustering in sparse graphs.
[8] studies the performance of considering data smoothness for spectral clustering on data with various sizes and densities. This work is kind of complementing the existing spectral methods in handling multi-scale data, such as ROSC and CAST, under the situation that data is presented in a significant smooth manner. Though this paper singly focuses on multi-scale situations, the authors' idea of data smoothness will possibly be extended to any clustering algorithms. (Since this is not the full version, experimental results have not been provided yet.)

IV. New Contribution

A. Motivation

Inspired by the existing work about spectral clustering and community detection, now we propose our own novel contribution. The issue we are going to cope with is *intra-cluster degeneration* in dynamic networks. Here intra-cluster degeneration basically refers to the situation that some relationships within a cluster suddenly vanishes, which can be interpreted as removed edges from the view of graph representation. Under such circumstances, the original cluster information about its components or even whether a cluster ever exists may be lost or become hard to detect. The effect of this issue will be amplified if the graph is of low conductance. Suppose we aim to preserve the original clustering structure and recover it after intra-cluster degeneration occurs, we need some methods to prevent the network from suffering from the degeneration case. One of the possible ways to handle it is **eigenspectrum augmentation** proposed by us. The basic idea behind it is that we enhance the weights of edges within clusters and leave the inter-cluster edges unchanged. By doing so, we can distinguish the edges via weight information under degeneration cases. In later sections, we will give some simple examples of how eigenspectrum augmentation works.

B. The tradition pipeline of spectral partitioning algorithm

1) **Pre-processing**:
   Build a Laplacian matrix $\mathbf{L}$ of the given graph. (For simplicity, we do not apply normalized Laplacian here.) The

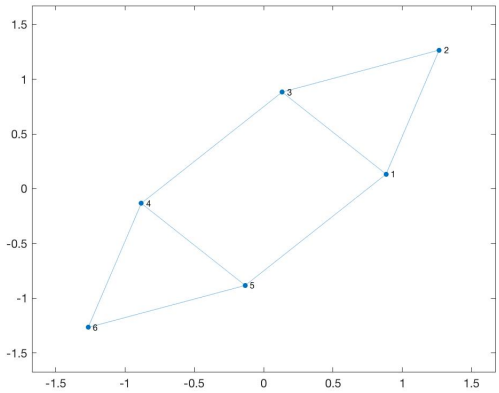

Fig. 1. the graph with adjacency matrix $\mathbf{A}$ and Laplacian matrix $\mathbf{L}$

above graph is constructed by adjacency matrix $\mathbf{A} = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$ and its associated Laplacian matrix $\mathbf{L}$ is

$$\begin{bmatrix} 3 & -1 & -1 & 0 & -1 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ -1 & -1 & 3 & -1 & 0 & 0 \\ 0 & 0 & -1 & 3 & -1 & -1 \\ -1 & 0 & 0 & -1 & 3 & -1 \\ 0 & 0 & 0 & -1 & -1 & 2 \end{bmatrix}$$

2) **Decomposition:**
Find eigenvalues λ and eigenvectors $\mathbf{v}$ of the Laplacian matrix $\mathbf{L}$. Then map vertices to corresponding components of $\mathbf{v_2}$ (the eigenvector corresponding to the second smallest eigenvalue). Continue with the previous example, we have

$$\lambda = \begin{bmatrix} 0 \\ 1 \\ 3 \\ 3 \\ 4 \\ 5 \end{bmatrix}, \qquad \mathbf{v} = \begin{bmatrix} 0.4 & \mathbf{0.3} & -0.5 & -0.2 & -0.4 & -0.5 \\ 0.4 & \mathbf{0.6} & 0.4 & -0.4 & 0.4 & 0 \\ 0.4 & \mathbf{0.3} & 0.1 & 0.6 & -0.4 & 0.5 \\ 0.4 & \mathbf{-0.3} & 0.1 & 0.6 & 0.4 & -0.5 \\ 0.4 & \mathbf{-0.3} & -0.5 & -0.2 & 0.4 & 0.5 \\ 0.4 & \mathbf{-0.6} & 0.4 & -0.4 & -0.4 & 0 \end{bmatrix}$$

and the mapping from vertices to corresponding components of $\mathbf{v_2}$ is $\left|\begin{array}{c|c} 1 & 0.3 \\ 2 & 0.6 \\ 3 & 0.3 \\ 4 & -0.3 \\ 5 & -0.3 \\ 6 & -0.6 \end{array}\right|$.

From Rayleigh Theorem, we have

$$\lambda_2 = \min_{x:x^T\mathbf{v_1}=0} \frac{x^T L x}{x^T x} = \min_{x:x^T\mathbf{v_1}=0} \frac{\sum_{(i,j)\in E}(x_i - x_j)^2}{\sum_i x_i^2}$$

, where $\mathbf{v_1}$ is the eigenvector corresponding to λ_1. (Note that we sort $\lambda_\mathbf{i}$ in ascending order such that $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$) In the above equation, x is orthogonal to the first eigenvector thus we have $\sum_i x_i = 0$. So the interpretation of λ_2 as an optimization problem is that we want to assign values x_i to nodes i such that few edges cross 0.

3) **Grouping:**
After obtaining λ_2 and its corresponding mapping, we sort components of reduced 1-dimensional vector and then identify clusters by splitting the sorted vector in two. The naive approach of choosing a splitting point is simply split at 0 or

TABLE I
CLUSTER A: POSITIVE POINTS

1	0.3
2	0.6
3	0.3

TABLE II
CLUSTER B: NEGATIVE POINTS

4	-0.3
5	-0.3
6	-0.6

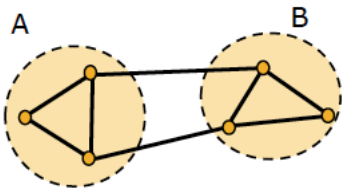

Fig. 2. Clustering

median value. If we wish to get more clusters, we can recursively sort the clusters in two. To illustrate this with the previous example, we can tell cluster A and cluster B directly from the mapping in step 2).
To better visualize the clustering result, we plot the components of $\mathbf{v_2}$.

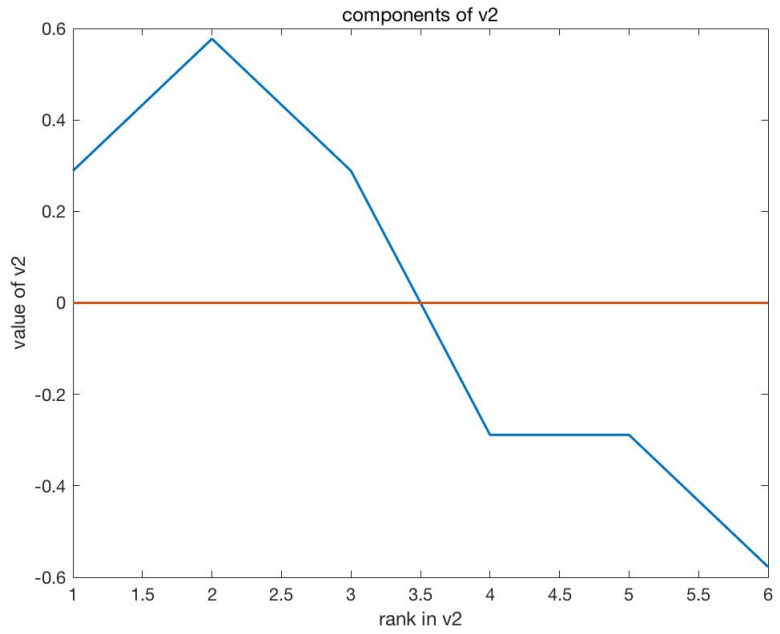

Fig. 3. Sketch plot of components of $\mathbf{v_2}$

C. intra-cluster degeneration case

Suppose some edges within the clusters are removed. With the previous example, assume that two edges are removed. Now the associated adjacency matrix and Laplacian matrix become

$$\mathbf{A}' = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}, \qquad \mathbf{L}' = \begin{bmatrix} 2 & -1 & 0 & 0 & -1 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & -1 & 2 & 0 & -1 \\ -1 & 0 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & -1 & -1 & 2 \end{bmatrix}$$

Following the same procedure as before, we skip the middle steps and directly plot λ_2 of $\mathbf{L}'$:

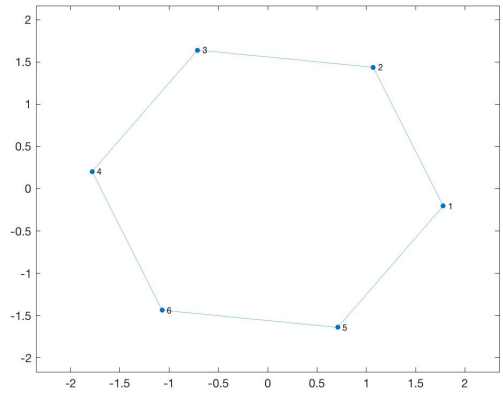

Fig. 4. degeneration case

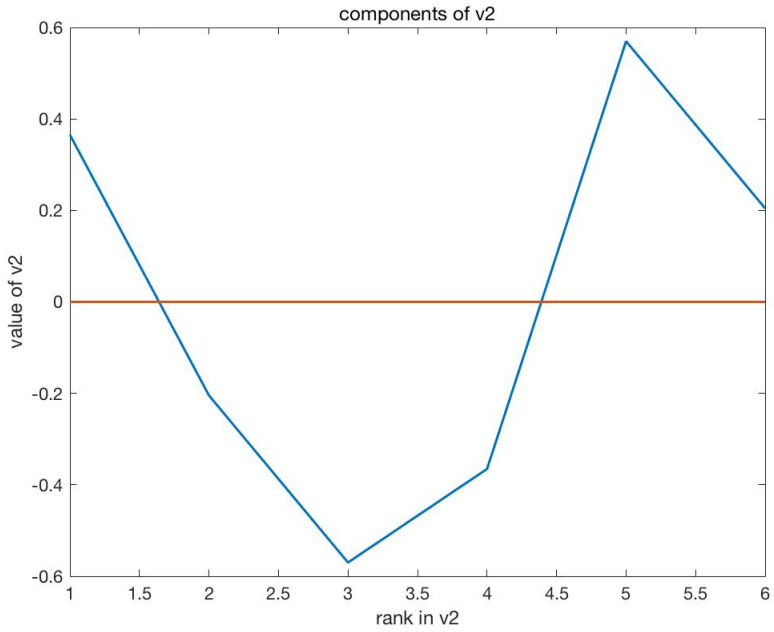

Fig. 5. Sketch plot of $\mathbf{v_2}$ in degeneration case

From the plot we can see that $\lambda_\mathbf{2}$ crosses 0 at two points, thus the clustering information is completely lost and it becomes impossible to tell which cluster a vertex belongs to in the original graph.

D. eigenspectrum augmentation

To overcome the intra-cluster degeneration problem, we apply some modifications to the adjacency matrix as well as the Laplacian matrix of the graph.

Denote the intra-cluster edges in the original graph as $\{E_{intra}\}$ and the inter-cluster edges as $\{E_{inter}\}$. We multiply the weight of E_{intra} by a **augmentation factor** α with $\alpha > 1$ and leave the weight of $\{E_{inter}\}$ unchanged. (Since we deal with unweighted graph here, i.e. graph with weight 1 for edges, it's guaranteed that the modified weights will be larger than before.) In particular, the augmentation factor α controls how strong or robust a cluster is desired to be such that it can tolerate some internal degeneration while still keeps itself as a cluster.

We keep illustrate our scheme with the previous example. Suppose we have applied eigenspectrum augmentation to the original

graph and set $\alpha = 1.5$, we obtain the modified adjacency matrix and Laplacian matrix under the degeneration case:

$$\mathbf{A}'' = \begin{bmatrix} 0 & 1.5 & 0 & 0 & 1 & 0 \\ 1.5 & 0 & 1.5 & 0 & 0 & 0 \\ 0 & 1.5 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1.5 \\ 1 & 0 & 0 & 0 & 0 & 1.5 \\ 0 & 0 & 0 & 1.5 & 1.5 & 0 \end{bmatrix}, \quad \mathbf{L}'' = \begin{bmatrix} 2.5 & -1.5 & 0 & 0 & -1 & 0 \\ -1.5 & 3 & -1.5 & 0 & 0 & 0 \\ 0 & -1.5 & 2.5 & -1 & 0 & 0 \\ 0 & 0 & -1 & 2.5 & 0 & -1.5 \\ -1 & 0 & 0 & 0 & 2.5 & -1.5 \\ 0 & 0 & 0 & -1.5 & -1.5 & 3 \end{bmatrix}$$

Following the same procedure as before, again we skip the middle steps and directly plot λ_2 of $\mathbf{L}''$:

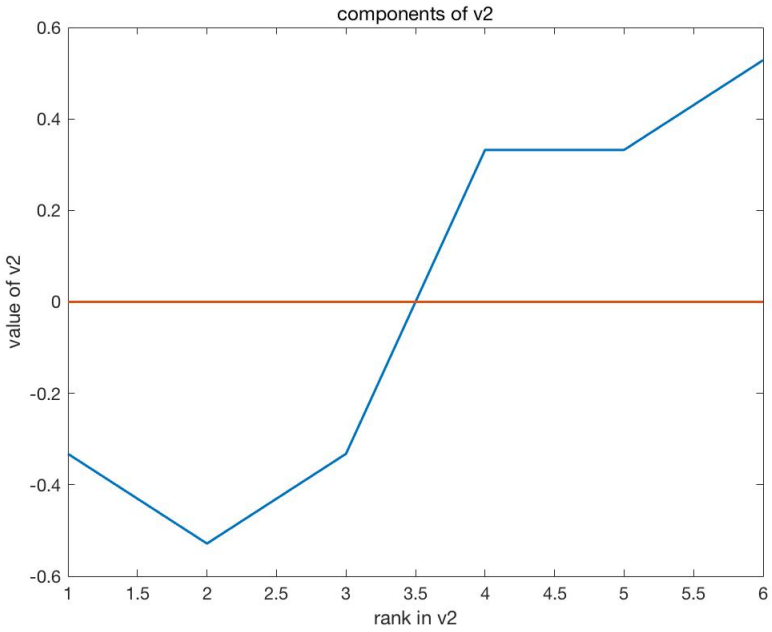

Fig. 6. Sketch plot of $\mathbf{v_2}$ under degeneration case with augmentation

From the above plot, we can see that the cluster structure become clear again even after intra-cluster degeneration.

V. Numerical results

In this section we test our proposed method on a graph with 20 nodes.

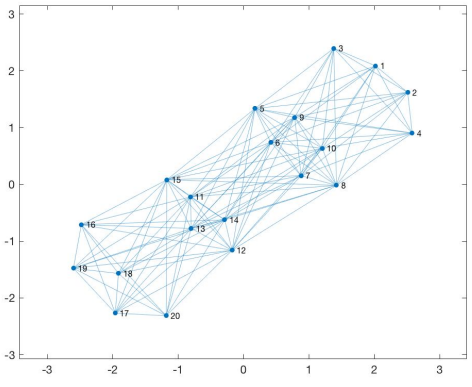

Fig. 7. a graph with 20 nodes

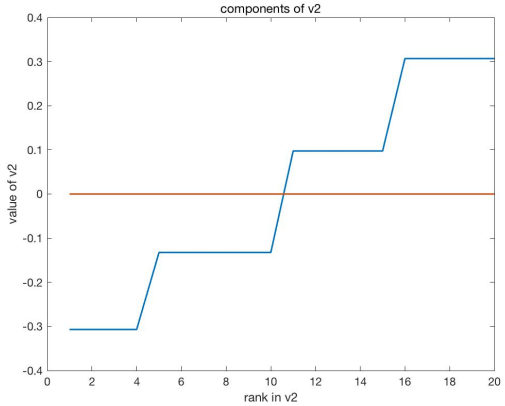

Fig. 8. Sketch plot of components of $\mathbf{v_2}$

Now assume about half edges within each cluster are removed shown as follows:

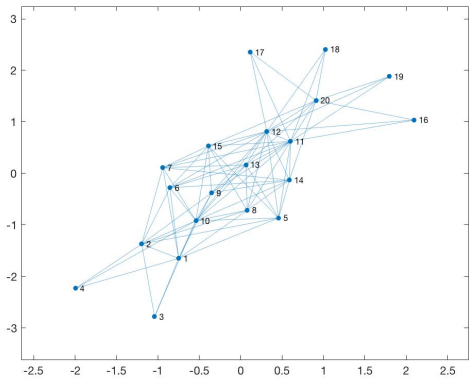

Fig. 9. a graph with 20 nodes under degeneration case

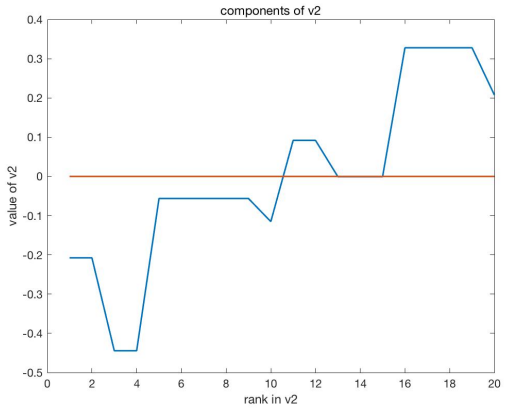

Fig. 10. Sketch plot of components of $\mathbf{v_2}$ under degeneration case

We can see that the clustering or partitioning becomes ambiguous in that the plot of $\lambda_\mathbf{2}$ crosses 0 at several points. Then we apply eigenspectrum augmentation to the original graph with $\alpha = 4$ and plot its components of $\lambda_\mathbf{2}$:

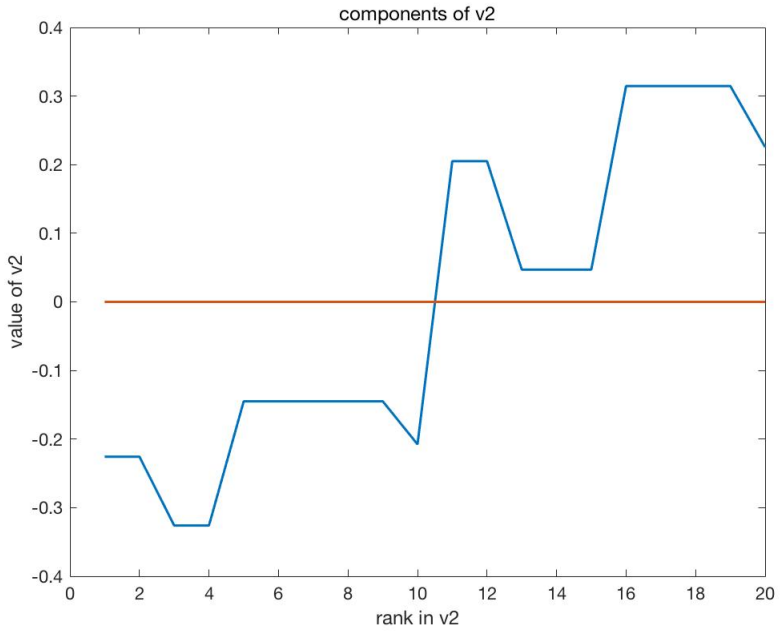

Fig. 11. Sketch plot of components of $\mathbf{v_2}$ under degeneration case with augmentation

It can be observed that the clustering structure is well reconstructed.

VI. Conclusions

In this project we focus on spectral clustering, which is a topic that has been developed rapidly through the past years. Firstly we give an overview of the problem formulation and background, together with some state-of-the-art existing work. Then we propose a new method, namely eigenspectrum augmentation, to avoid losing clustering structure information under the circumstance of intra-cluster degeneration. Though the method is quite simple and intuitive, it performs well in our experimental results as long as the augmentation factor α is properly tuned.

For further future study related to this topic, there are some possible directions that we can think of:

1) Quantify the relationship between the augmentation factor α and the conductance of graphs.
2) Try this method on weighted or directed graphs and test its performance to see how we can modify the original scheme to adapt to other types of graphs.

3) Recursively apply eigenspectrum augmentation to achieve k-way partitioning reconstruction under degeneration cases.

REFERENCES

[1] March Boedihardjo, Shaofeng Deng, and Thomas Strohmer. A performance guarantee for spectral clustering, 2020.
[2] Camille Champion, Blazère Mélanie, Burcelin Rémy, Loubes Jean-Michel, and Risser Laurent. Robust spectral clustering using lasso regularization, 2020.
[3] Lorenzo Dall'Amico, Romain Couillet, and Nicolas Tremblay. A unified framework for spectral clustering in sparse graphs, 2020.
[4] Steinar Laenen and He Sun. Higher-order spectral clustering of directed graphs, 2020.
[5] Zhenyu Liao, Romain Couillet, and Michael W. Mahoney. Sparse quantized spectral clustering, 2020.
[6] Andrew Ng, Michael Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In T. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems*, volume 14, pages 849–856. MIT Press, 2002.
[7] Gan Sun, Yang Cong, Qianqian Wang, Jun Li, and Yun Fu. Lifelong spectral clustering, 2019.
[8] Hengrui Wang, Yubo Zhang, Mingzhi Chen, and Tong Yang. Spectral clustering with smooth tiny clusters, 2020.