# Binary Clustering Using the Fiedler Vector

**Matthew Sun**

## 1 Abstract

The purpose of this assignment is to explore the use of the Fiedler's vector as a means of performing binary clustering on the vertices of a graph. This is done by first calculating finding the Laplacian matrix of the adjacency matrix, then finding the eigenvector corresponding to the second smallest eigenvalue (the Fiedler value) of the Laplacian Matrix. The signs of the entries of the Fiedler vector are used to categorize the vertices into one of two clusters. It was found that with the given data set, the algorithm generated an effective binary clustering of the vertices with a sparse amount of connections between clusters and a dense amount within. The Fielder vector can be used to effectively perform binary clustering given a suitable data set but is limited to graphs that have relatively distinct binary partitions.

## 2 Introduction

The objective of the assignment was to determine whether the Fiedler vector of a Laplacian matrix corresponding to a given graph can be used to perform binary clustering on the vertices.

Binary clustering is the process of grouping the vertices of a graph into two clusters. A graph $G$ is defined as an ordered pair $G = (V, E)$ where:

- $V$ is a set of vertices.
- $E$ is a set of edges, which are unordered pairs, where $E \subseteq \{(x, y) \mid x, y \in V \text{ and } x \neq y\}$.

The adjacency matrix of a graph $A(G)$ is a binary and symmetric matrix of size $n \times n$ where $n = |V|$. A(G) has entries

$$a_{ij} := \begin{cases} 1 \text{ if } (V_i, V_j) \in E \\ \\ 0 \text{ if } (V_i, V_j) \notin E \end{cases} \tag{1}$$

The categorization of the vectors to one of the two clusters is identified by the signs of the Fiedler vector (vertices with negative values go to the first cluster and the other vertices go to the second cluster). The Fiedler vector $\vec{F}$ is the eigenvector corresponding to the second smallest eigenvalue of the Laplacian matrix of a graph $L(G)$. The Laplacian matrix is symmetric, real, diagonally dominant, and positive semi-definite implying that the eigenvalues of a Laplacian matrix are always non-negative values. It describes the adjacency of the graph and the degrees of each vertex. The Laplacian matrix for a given graph $G$ is $L(G) := D(G) - A(G)$. $D(G)$ is the diagonal degree matrix of $A(G)$ where each diagonal entry $d_{ii}$ is the degree of each vertex $v_i$. It is defined as $D(G) := diag(A(G) \cdot \vec{1})$.

The scientific question to be explored is: Can the Fiedler vector be used to effectively cluster the vertices of a given graph? In this case, a clustering is effective if there exists a minimal number of edges between the two clusters and a larger number of edges between vertices of the same cluster. The effectiveness of the algorithm was tested through visual observation of the generated clustering.

## 3 Methods

Let $G(V, E)$ be a graph defined by a list of pairs of integers. Let the set of edges $E$ be the set containing every $(x, y)$ pair in said list. Let the set of vertices $V = \{v_i \mid i \in \mathbb{Z}, 1 \leq i \leq \text{max integer value in the list}\}$. The adjacency matrix of size $n \times n$ where, $n = |V|$, was generated by iterating through the list representing the edges of $G$. By definition, connections are denoted with a 1 for an edge and 0 for no edge at the corresponding entry $d_{ij}$ where $(v_i, v_j) \in E$. The diagonal matrix was calculated by taking the sum

of each row of the adjacency matrix and placing them on the diagonals of a matrix of the same size, leaving all other entries as 0. The Laplacian matrix is then calculated using the definition $L(G) := D(G) - A(G)$.

The Fiedler vector is the eigenvector of the Laplacian matrix that corresponds to the second smallest eigenvalue of the Laplacian matrix. The clustering was done by going through the Fiedler vector and separating the vertices based on the signs of each value. The negative values of the Fiedler vector are assigned -1 and the non-negative values are assigned 1. This step is not necessary in general but it is simpler for the program to sort the vertices into two sets this way.

The clusters were then plotted and the effectiveness of the clustering was observed. The clustering of the test case shown in the assignment instructions appears as two circle-shaped graphs with very dense connectivity among the vertices of each cluster with a sparse amount of edges between the two clusters. As such, the evaluation of the output was based on the qualities of the clustering observed in the example given.

# 4 Results

**Table 1:** The two clusters of vertices using "20ms154.txt"

| | **Vertices** | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **Cluster 1** | 1 | 2 | 6 | 7 | 9 | 11 | 13 | 15 | 17 | 19 |
| **Cluster 2** | 3 | 4 | 5 | 8 | 10 | 12 | 14 | 16 | 18 | 20 |

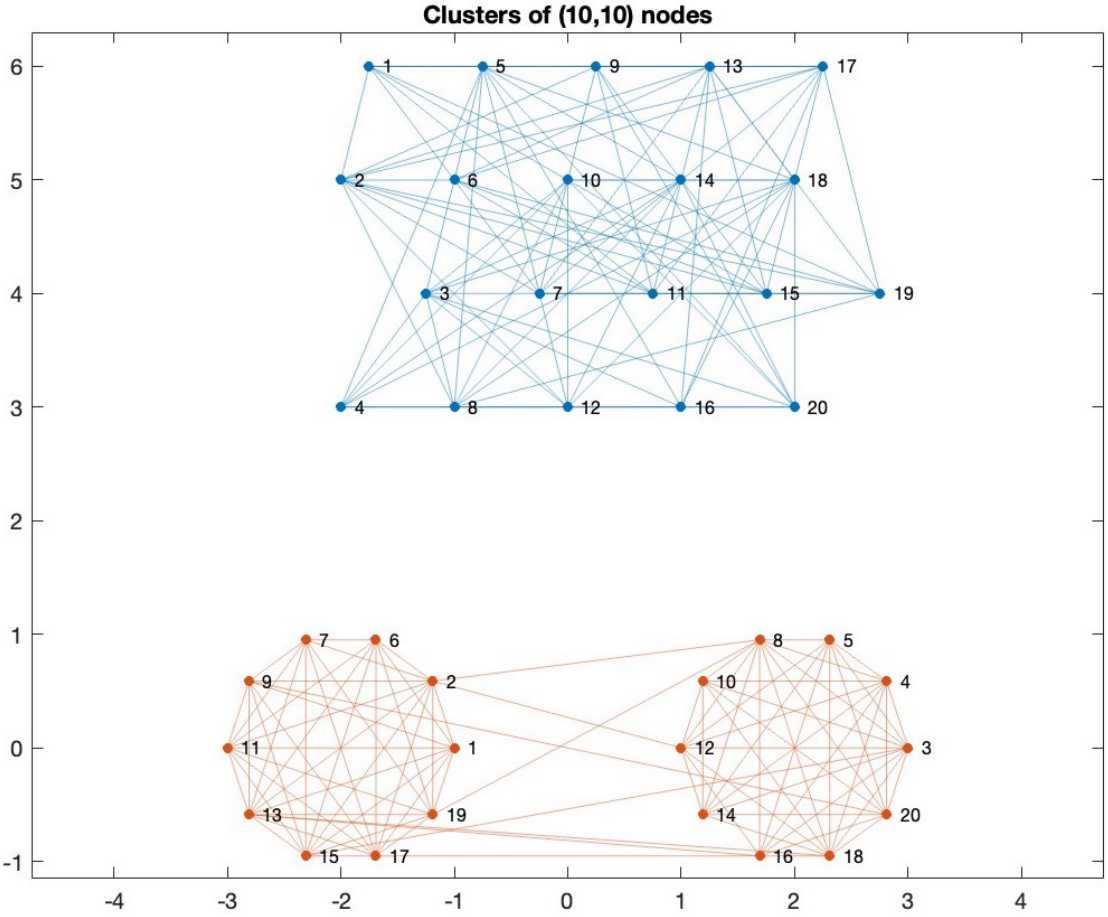


**Figure 1:** Plots of graph represented by "20ms154.txt", original graph in blue, binary clustering in orange

# 5 Discussion

For the given data set "20ms154.txt" that represents the edges of a graph, using the signs of the entries of the Fiedler vector to categorize resulted in an effective binary clustering of the vertices. It can be observed that out of the 91 edges that connect the 20 vertices of the graph, only 8 edges connect vertices from different clusters. This means an overwhelming majority ($\approx 88\%$) of edges belong exclusively to one of the two clusters. Based on the evaluation requirement that the generated clustering should have a sparse amount of edges relative to the amount within a cluster, this binary clustering algorithm using the Fiedler vector to categorize the vertices of a graph was successful, at least for this data set.

Problems in the application of this algorithm arise when dealing with less ideal data sets. If the adjacency of a graph at any arbitrary vertex is identical to any or most other vertices, for example a regular graph. In this case, the signs of the entries of the Fiedler vector would not provide meaningful information on how to perform binary clustering on these vertices since there is no effective binary clustering to begin with. This would limits the algorithm's ability to analyze graphs that are "further" from being two separate components. This relates to the concept of algebraic connectivity, where the magnitude of the Fiedler value describes the algebraic connectivity of the graph [Fie73]. For example, a graph with two separate components meaning that the null space of the Laplacian matrix is of dimension 2. The Fiedler value of such a graph would be 0 since that is its second smallest eigenvalue, indicating that it has low algebraic connectivity. More generally, the algebraic connectivity of a random graph decreases as as the number of vertices increase and increases with the average degree [Hol06]. This ability to measure algebraic connectivity applies itself well into the field of cognitive neuroscience, where the networks of the brain can be described as a graph and be analyzed using the corresponding Laplacian matrix. In one 2014 study [Dai+14], researchers used the Laplacian matrix and its Fiedler value to investigate the brain network patterns of those afflicted with Alzheimer's disease. They found that as the disease progressed, the density of connection between regions decreased which was indicated by the decrease of the Fiedler value. This lead to the conclusion that "diseased brains may be more vulnerable to losses in connections that allow communication between cortical regions, leading to a less robust neural network, at least according to these mathematical metrics." In the context of this assignment's algorithm, one could imagine that as the disease progressed, the edeges between the clusters would become sparser.

With a suitable data set, using the Fiedler vector as a means of performing binary clustering on the vertices of a graph is an effective algorithm. There are limitations to the type of graphs that the algorithm can effectively cluster, regardless this algorithm is able to provide meaningful information about the algebraic connectivity of a graph.

# References

[Fie73] Miroslav Fiedler. "Algebraic connectivity of graphs". In: *Czechoslovak Mathematical Journal* 23.2 (1973), pp. 298–305. DOI: 10.21136/cmj.1973.101168.

[Hol06] Michael Holroyd. "Synchronizability and Connectivity of Discrete Complex Systems". In: (Jan. 2006).

[Dai+14] Madelaine Daianu et al. "Algebraic connectivity of brain networks shows patterns of segregation leading to reduced network robustness in alzheimer's disease". In: *Computational Diffusion MRI* (2014), pp. 55–64. DOI: 10.1007/978-3-319-11182-7_6.