ID2222 Data Mining Homework 4

Graph Spectra

Kevin Dalla Torre `kevindt@kth.se`

Luís Santos `lmpss@kth.se`

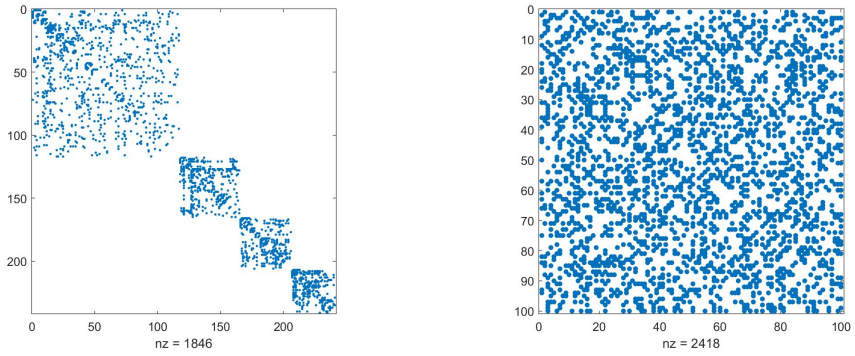
November 2020

1 Introduction

As instructed in this report we have implemented the spectral graph clustering algorithm from the paper Ng et al. (2001) featuring our implementation of K-eigenvector algorithm. We have then used it to analyze two datasets: one based on a real-world graph and another graph constructed synthetically. Since we have edges and not data points in the graph data we are choosing a different option in step one of the algorithm. We are creating the Adjacency matrix instead of the affinity matrix A.

2 Dataset 1

This data was prepared by Rob Burt. He dug out the 1966 data collected by Colman, Katz and Menzel on medical innovation. The adjacency matrix A (sparsity patterns) which was created from this data can be seen in Figure 1a. This image features the existing edges between the nodes, where a blue point signifies the presence of one edge and white areas mean the lack of connections. We can conclude there is one large community and three smaller communities. We clearly see that there are 4 separated communities already in this step since we observe that the adjacency matrix is block diagonal.
We can observe in Figure 2a the un-clustered graph and the clustered one in Figure 2b for dataset 1. From these images we can also see that there are four connected components, since there are no existing edges between communities, and that the algorithm clusters these communities correctly.

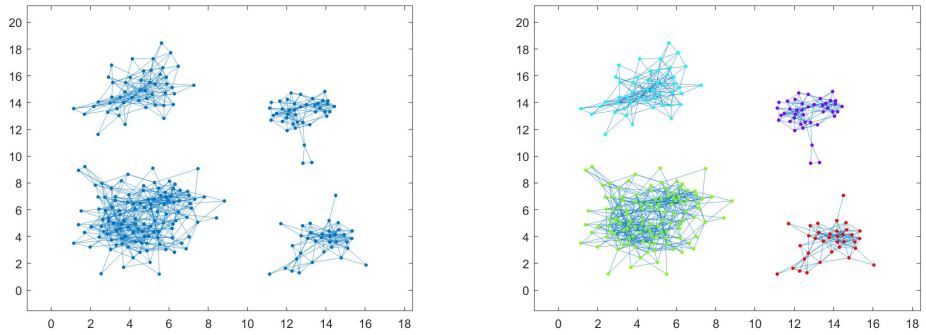


(a) sparsity matrix of dataset 1 (b) sparsity matrix of dataset 2

Figure 1: Sparsity patterns of dataset 1 and 2.

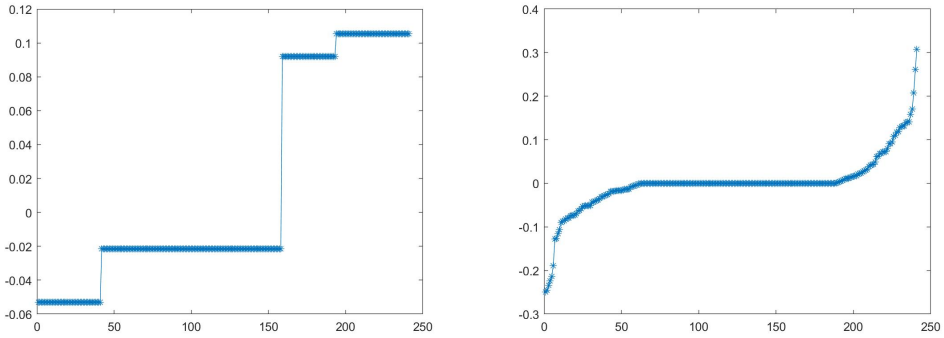
Usually k is a hyper-parameter but in this graph we can infer that the optimal k is 4 right away from either the sparsity pattern or the unclustered data.

We first tried to obtain the Fiedler Vector using the normalized Laplacian matrix L as calculated in the suggested paper. This result can be visualized in Figure 3b. We had 4 clusters for this dataset and therefore the expected behaviour was more closely matched with the Fiedler vector on Figure 3a. We obtained this result for the un-normalized Laplacian $L = D - A$, where D is the diagonal matrix from the row sum of A and A is the adjacency matrix.



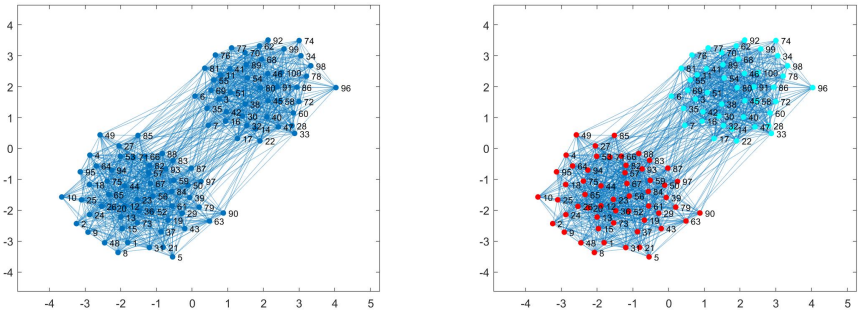
(a) non-clustered data (b) clustered data

Figure 2: Spectral Clustering of dataset 1.



(a) un-normalized Laplacian (b) normalized Laplacian

Figure 3: Fiedler Vector of dataset 1.

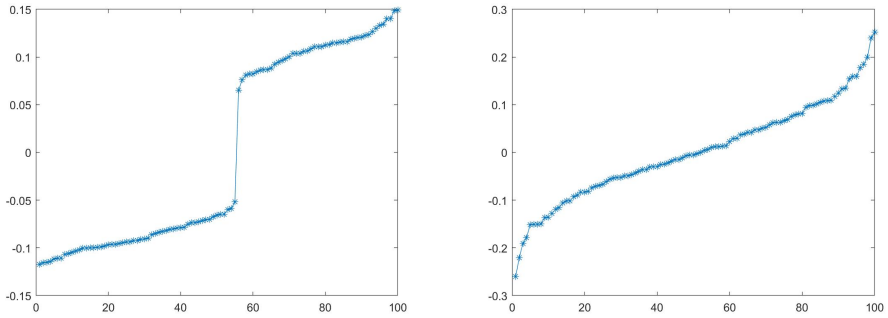


(a) non-clustered data (b) clustered data

Figure 4: Spectral Clustering of dataset 2.

3 Dataset 2

The other dataset is a synthetic graph, and its corresponding sparsity pattern can be seen in Figure 1b. We can not see any clusters directly from this image since it is not block diagonal and it looks like one large, connected component. If we look at Figure 4a it looks like it should contain 2 clusters for this case. However, we also notice edges between these clusters so it is in fact one big connected component. If we look at Figure 4b we can see that the algorithm clusters the data in the desired way without incurring any false positives.
If we look at the Fiedler Vector in Figure 5, we can see that even in this case the un-normalized Laplacian provides a better result. The fact that there are edges between the clusters is also represented in Figure 5a. The Fielder vector shows the partitioning of the graph and if we compare Figure 5a with Figure 3a we can see the difference for the case where there is a clear separation of the clusters and when there is not. We have a non-smooth function in Figure 3a due to the lack of edges between components, in contrast to Figure 5a.



(a) un-normalized Laplacian (b) normalized Laplacian

Figure 5: Fiedler Vector of dataset 2.

References

A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Proceedings of the International Conference on Neural Information Processing Systems*, NIPS'01, Cambridge, MA, USA, 2001.