ID2222 Data Mining Homework 4

GIOVANNI MANFREDI SEBASTIANO MENEGHIN
gioman | meneghin @kth.se

04th December 2023

1 Introduction

Our project consist of the implementation in MatLab of the K-eigenvector Algorithm [1] for two datasets containing the list edges of two graphs, thus creating two graphs having numerous non-overlapping communities between their nodes.

As first step, we enter the data from the two datasets present in our project folder (they are presented in the file *.zip* since they are not retrievable online, if accessing on KTH's Canvas platform). Then the inserted data are processed, according to the algorithm specifications and the graphs are visualized, before and after the clusterization process, along with the values of the primary eigenvalues.

The nature of the two datasets as well as the results, are presented in the following sections, respectively in section 2, section 3 and section 4. The project is completely runnable using the files provided in the submitted file *.zip*, having a valid license of MatLab.

2 First Dataset

The first dataset was created by Rob Burt. He located and analysed the 1966 data on medical innovation gathered by Colman, Katz, and Menzel. They collected data from physicians in four towns in Illinois: Peoria, Bloomington, Quincy, and Galesburg.

The Adjacency Matrix A of the first is presented in Figure 1. The matrix shows the presence of edges between the connected nodes of the graph: a blue point means the presence of one edge, whereas a white area means the lack of connections. What we can infer from this graph is the presence of four different communities, each of them having a different size.

3 Second Dataset

The second dataset is instead a synthetic graphs created for the sake of testing the K-eigenvector Algorithm [1].

The Adjacency Matrix A of the first is presented in Figure 2. The matrix shows the presence of edges between the connected nodes of the graph: a blue point means the presence of one edge, whereas a white area means the lack of connections. In this case, we cannot capture at first glance any trivial division in communities.

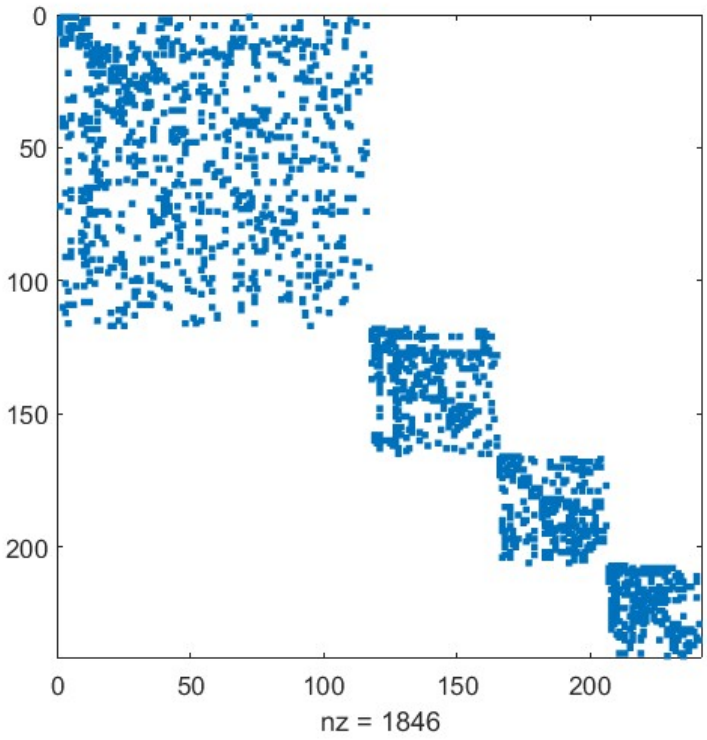


Figure 1: Adjacent Matrix for First Graph

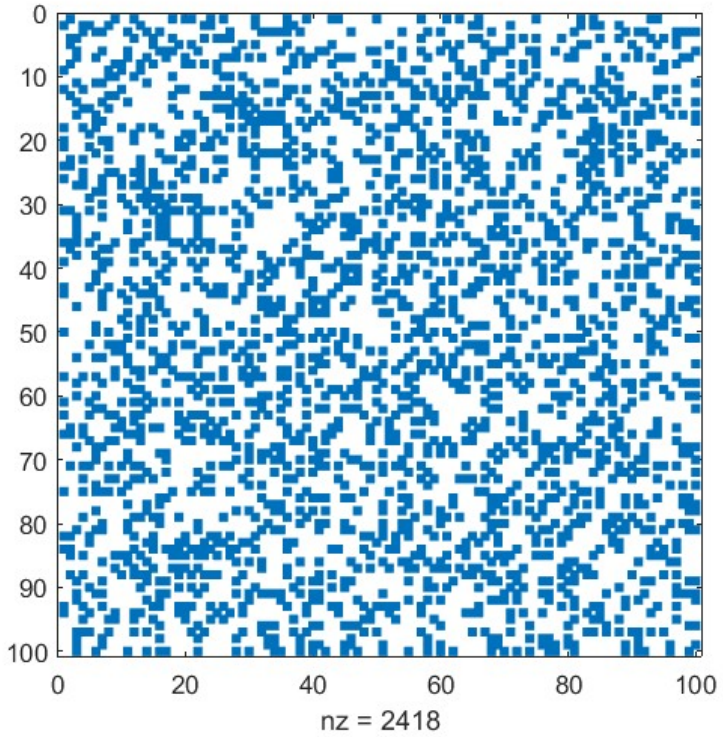


Figure 2: Adjacent Matrix for Second Graph

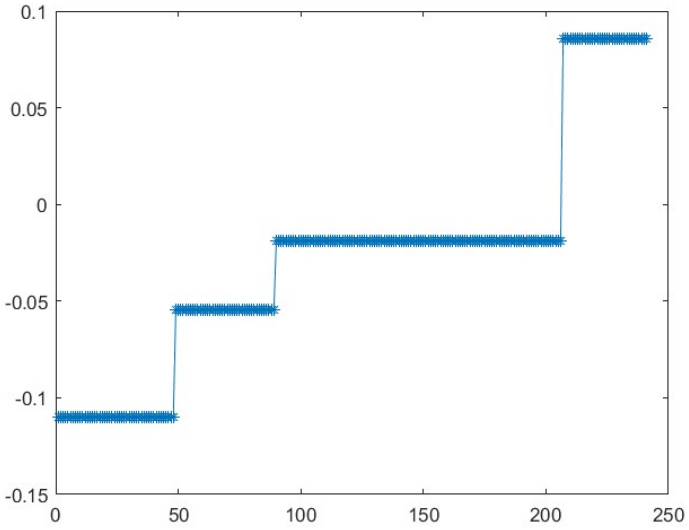
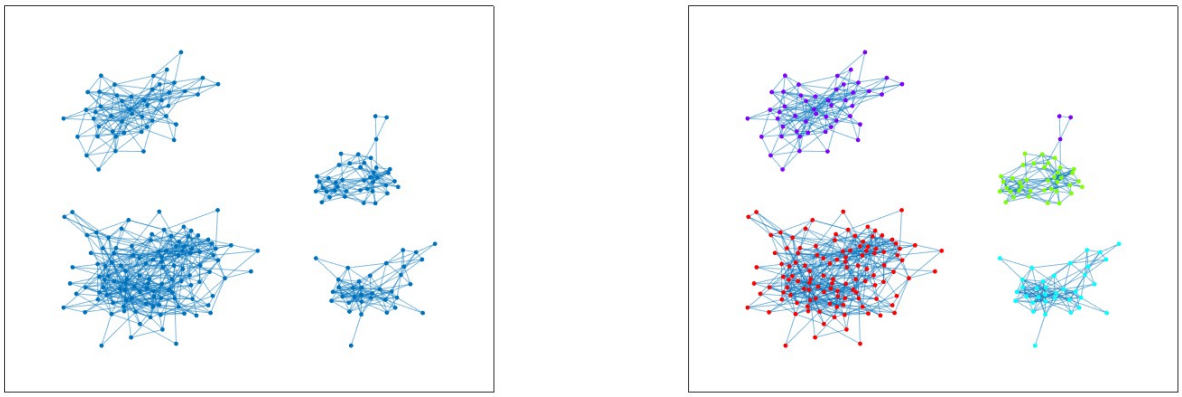


Figure 3: Eigenvalues of the un-normalized Laplacian matrix L of First Graph



(a) Not clustered graph

(b) Clustered graph

Figure 4: Clusterization of First Graph - 2D

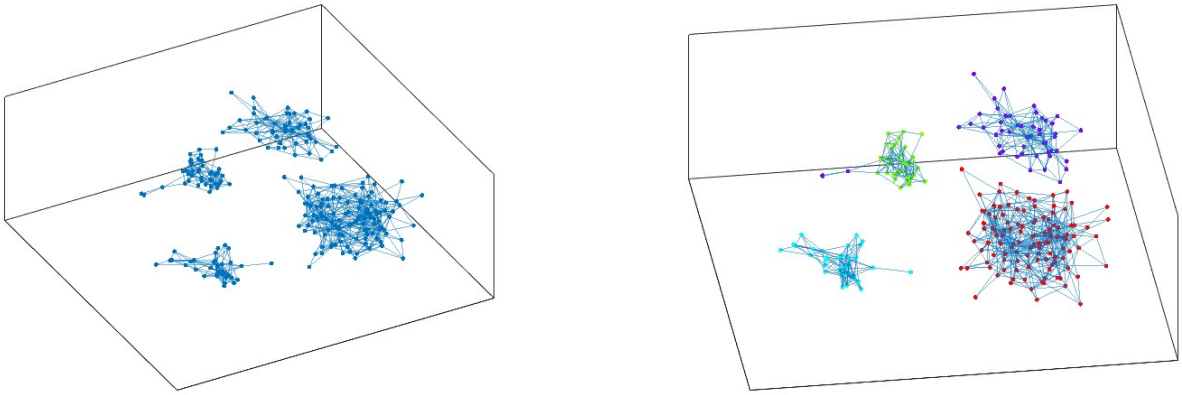
4 Results

4.1 First Graph Results

Even if normally the *paremeter k* is an hyperparameter, from the Adjacency Matrix of the unclustrered first graph (Figure 1) we can infer that the best value for k is $k = 4$.

The Fiedler Vector used to define the clusters has been derived from the normalized Laplacian matrix $L' = (\hat{D}^{-1/2} * A * \hat{D}^{-1/2})$, as defined in the algorithms' paper: here D is the Diagonal Matrix of the column sum of A, where A is the adjacency matrix of the first graph. However, the plot reported in Figure 3, is the spectrum determined by the un-normalized Fiedler Vector, that shows more understandable results since its components are not normalized.

The results of the clusterization are presented in 2D in Figure 5 and in 3D in **??**. It can be clearly seen how the full network has been divided in 4 different clusters/communities, determined by the colours associated to different nodes (green, purple, red and light blue).



(a) Not clustered graph (b) Clustered graph

Figure 5: Clusterization of First Graph - 3D

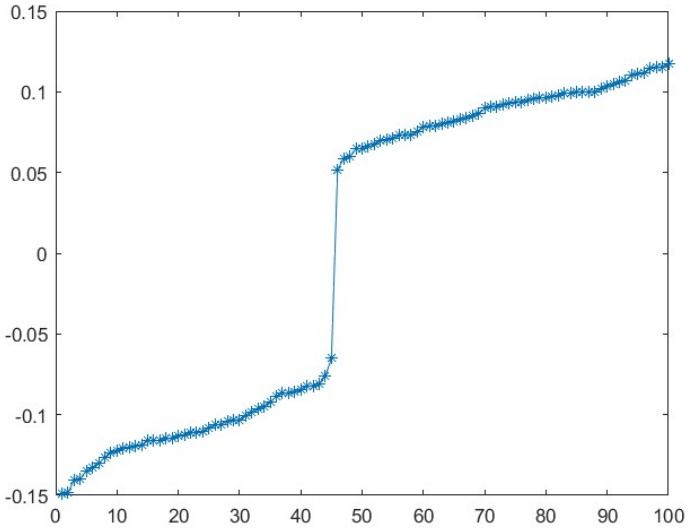


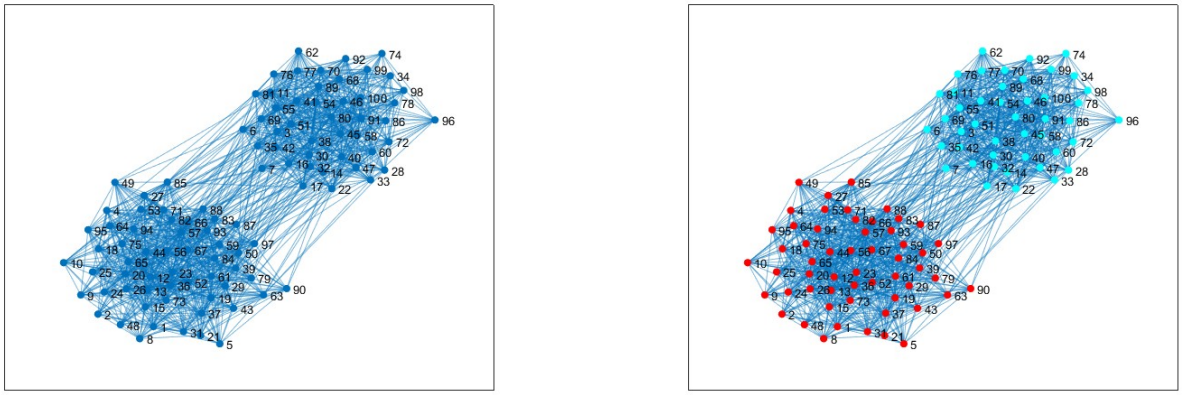
Figure 6: Eigenvalues of the un-normalized Laplacian matrix L of Second Graph

4.2 Second Graph Results

In this particular case, the k selected is $k = 2$, as the tuning of the hyper-parameter k during the beginning of the work on this project suggested.

The Fiedler Vector used to define the clusters has been derived from the normalized Laplacian matrix $L' = (D\hat{}(-1/2)*A*D\hat{}(-1/2))$, as defined in the algorithms' paper: here D is the Diagonal Matrix of the column sum of A, where A is the adjacency matrix of the first graph. However, the plot reported in Figure 6, is the spectrum determined by the un-normalized Fiedler Vector, that shows more understandable results since its components are not normalized.

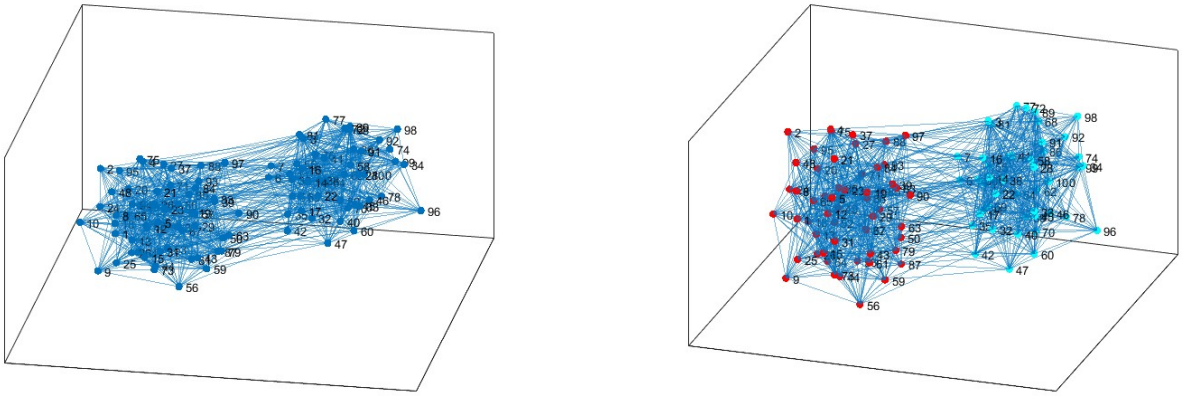
The results of the clusterization are presented in 2D in Figure 7 and in 3D in Figure 8. It can be clearly seen how the full network has been divided in 2 different clusters/communities, determined by the colours associated to different nodes (red and light blue).



(a) Not clustered graph

(b) Clustered graph

Figure 7: Clusterization of Second Graph - 2D



(a) Not clustered graph

(b) Clustered graph

Figure 8: Clusterization of Second Graph - 3D

5 How to run

In order to run the project, you must have a working license of MatLab installed on your own machine. Then, you can use the MatLab application or whichever IDE of your choice that is provided of a MatLab extentions with which you can run your code. Your MatLab environment should also contain the Statistics and Machine Learning Toolbox and a valid license of MatLab.

The datasets are provided in the folder of the projects, so you it is not needed to retrieve them. In order to see the results provided by the MatLab script, you only need to run the code contained in the file *script.m* and wait few seconds for the computation.

References

[1] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS*. MIT Press, 2001, pp. 849–856. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.19.8100