

Assignment 4: Graph Spectra

Filippo Boiani <boiani@kth.se>
Riccardo Sibani <rsibani@kth.se>

2 Dec 2017

NOTE

The code can be found at the following link: <https://github.com/filippoboiani/data-mining>

Introduction

In this assignment we implement the k-eigenvectors algorithm (or spectral clustering) to find clusters in a graph. The algorithm is taken from the paper “On Spectral Clustering: Analysis and an algorithm” by Ng, Jordan and Weiss.

The algorithm is then tested on two input files: example1.dat and example2.dat

How to Run

- via console, open the assignment directory
- run the following set of commands

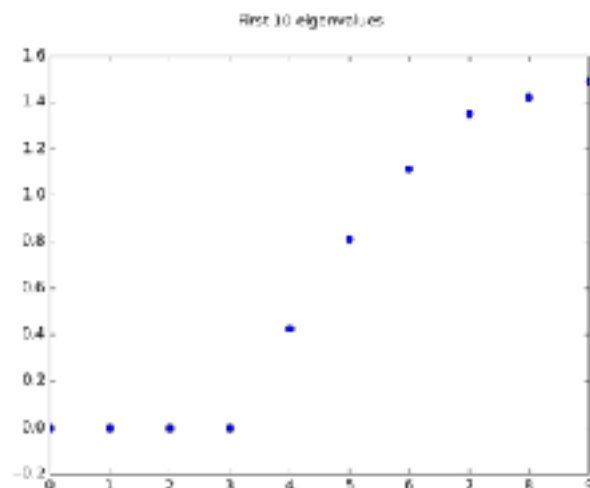
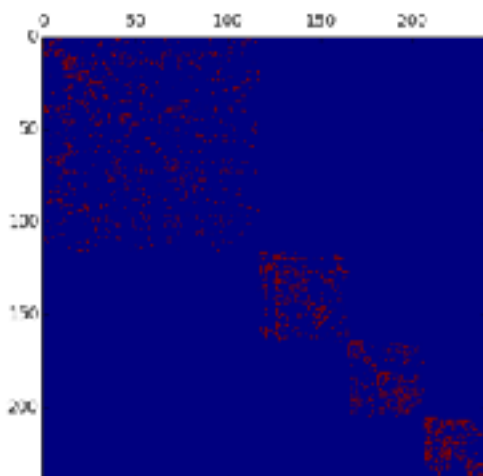
```
cd graph_spectra  
python spectra.py
```

Approach

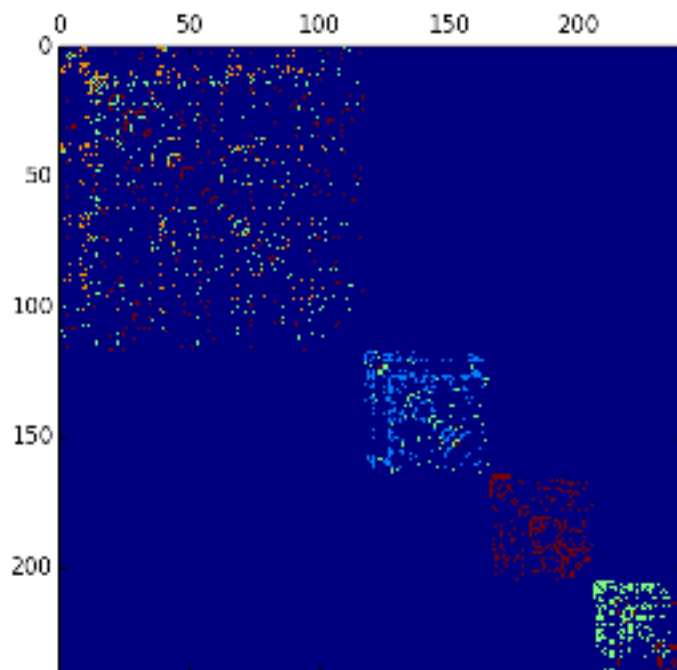
the repository contains the file spectra.py and follows all the steps proposed by the algorithm.

Results

EXAMPLE1.DAT

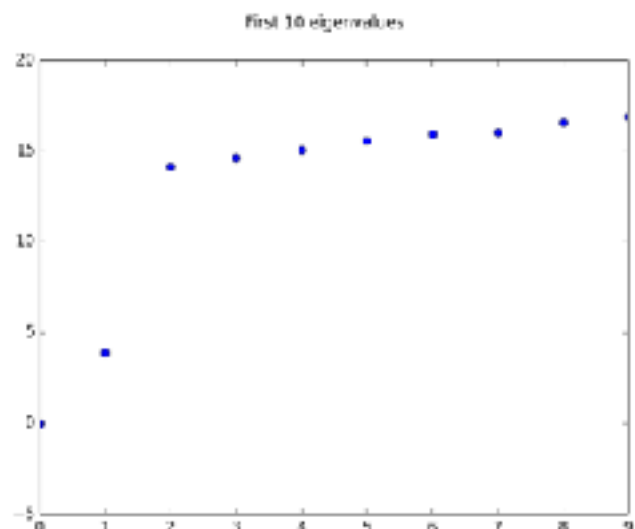
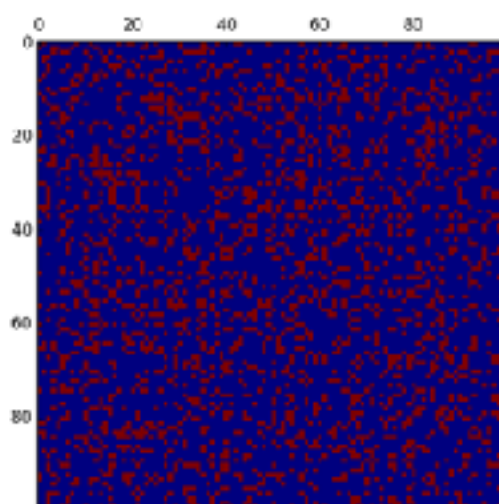


Analysing the adjacency matrix and the first eigenvalues of its normalisation, we can see that the first 4 are equal. This means that there are going to be 4 disconnected components for this graph.



After running the clustering algorithm the output should be something similar to the image above. We see different communities and how these are connected to one another.

EXAMPLE2.DAT



The eigen-gap between the first two eigenvectors suggest that the graph is well connected. Therefore it is difficult to see different and segregated communities. In fact, the algorithm doesn't show any cluster.

