Homework Spectral Clustering

Abstract

This is the description of the mandatory homework about Spectral Clustering (HW_SC), course “Computational Linear Algebra for Large Scale Problems”, A.A. 2024-2025.

1 Instructions

The numerical codes that are needed to address the homework must be written using Matlab (preferably) or Python as programming language. The work must be done alone or in a group consisting of two people at most. As a proof of the work done to solve the homework, you are requested to upload:

- a technical report (.pdf file) containing a detailed and exhaustive explanation of how you have solved the different tasks, of the results obtained using the different data provided, the results obtained using other clustering methods you have experimented;
- all the implemented codes together with an *Instruction.txt* file that must explain how to run the codes and reproduce the results presented in the report.

The quality of the code and the effectiveness of the description about how to run it will be part of the evaluation, together with the quality of the report and the correctness of the results. All the material must be included into an archive (zip file or tgz file). The name of the archive must consist of the surname of the student (or the surnames of the two students) and the reference “HW_SC” to the homework (e.g. Surname1_HW_SC.tgz or Surname1_Surname2_HW_SC.tgz o .zip). Do not use spaces in the name of the files. The material must be uploaded 5 days before the oral exam in the section “Elaborati” in the web page of the course. For the first call of the year 2 days before are enough.

2 Requirements

The data-sets of N points to be used are contained in the files *Circle.mat*, *Spiral.mat*, or *Circle.csv* and *Spiral.csv*. The file *Circle.** contains two columns corresponding to x-values and y-values of the points; the file *Spiral.** contains three columns, the third one contains the index of the correct cluster.

Recall that, given a similarity function $s_{i,j}$ for two points, the similarity graph is denoted by $G = (V, E)$, where $V = v_1, ..., v_n$ denotes a non-empty set of vertices and E denotes the set of edges, i.e., a set of pair of vertices. In the similarity graph each vertex $v_i \in V$ represents a data point X_i . An edge between two vertices v_i and v_j exists if the similarity s_{ij} between the corresponding data points X_i and X_j is either positive or larger than a certain threshold (minimum similarity value ϵ for a connection to take place between two data points). We assume that $s_{ij} = s_{ji}$ and that the edge connecting v_i and v_j is weighted by s_{ij}. Consequently, it turns out that the similarity graph is undirected. The weighted adjacency matrix is defined as $\boldsymbol{W}_{ij} = s_{ij}$, if $i \neq j$ and $\boldsymbol{W}_{ij} = 0$, if $i = j$.

For both the data-sets address the following items.

1. Given a set of data points X and the similarity function

$$s_{i,j} = \exp\left(-\frac{||X_i - X_j||^2}{2\sigma^2}\right),$$

construct the *k-nearest neighborhood* similarity graph and its adjacency matrix $\boldsymbol{W}$. Note that the adjacency matrix $\boldsymbol{W}$ has zero-values on the diagonal by definition. Try several values of $k = 10, 20, 40$. Use $\sigma = 1$.

2. Construct the degree matrix $\boldsymbol{D}$ and the Laplacian matrix $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$. The sparse format storage for the matrices $\boldsymbol{W}$, $\boldsymbol{D}$ and $\boldsymbol{L}$ is strongly preferable.

3. Compute the number of connected components of the similarity graph.

4. Compute some small eigenvalues of $\boldsymbol{L}$ and use their values to choose a suitable number of clusters M for the points data-sets.
A self implementation of the numerical method for the eigenvalue approximation is preferable (see optional point 3).

5. Compute the M eigenvectors $u_1, ..., u_M \in \mathbb{R}^N$ that correspond to the M smallest eigenvalues of the Laplacian matrix, and define the matrix $\boldsymbol{U} \in \mathbb{R}^{N \times M}$ with these vectors as columns.
A self implementation of the numerical method for the eigenvalue approximation is preferable (see optional point 3).

6. For $i = 1, ..., N$ let $y_i \in \mathbb{R}^M$ be the vector corresponding to the i-th row of $\boldsymbol{U}$. Cluster the points y_i, $i = 1, ..., N$ in $\mathbb{R}^M$ with the *k-means* algorithm into clusters $C_1, ..., C_M$.

7. Assign the original points in X to the same clusters as their corresponding rows in $\boldsymbol{U}$ and construct the clusters $A_1, ..., A_M$, with $A_i = \{x_j : y_j \in C_i\}$.

8. Plot the clusters of points X with different colors.

9. Compute the clusters for the same set of points with other clustering methods (*k-means*,...) and compare the results.

The following points are optional

1. Look for or create other data-sets in 3D or higher dimension to test spectral clustering.

2. Apply the same process to the normalized symmetric Laplacian matrix $\boldsymbol{L}_{sym} \in \mathbb{R}^{N\times N}$ that is defined as

$$\boldsymbol{L}_{sym} := \boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{L}\boldsymbol{D}^{-\frac{1}{2}} = \boldsymbol{I} - \boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{W}\boldsymbol{D}^{-\frac{1}{2}}.$$

3. Implement the *Inverse Power Method* and the *Deflation Method* to compute the M smallest eigenvalues of a symmetric matrix. See requests 4 and 5.