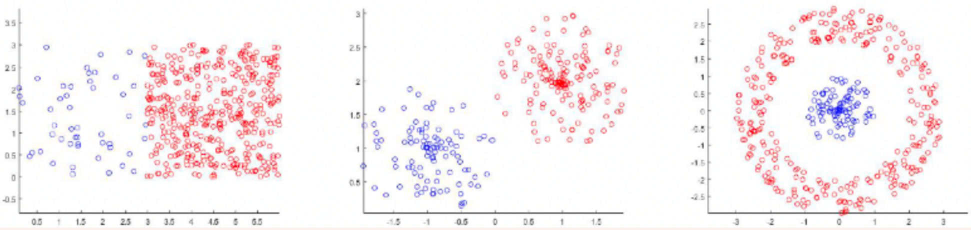
Numerical Linear Algebra 2025/2026 - Challenge 2

Eigenvalue problems applied to clustering and social networks

The goal of the challenge is to **cluster the nodes of a network** into maximally connected components by the approximate solution of **eigenvalue problems**. An overview on the topic of this challenge and the .mtx file of the social network are available here:

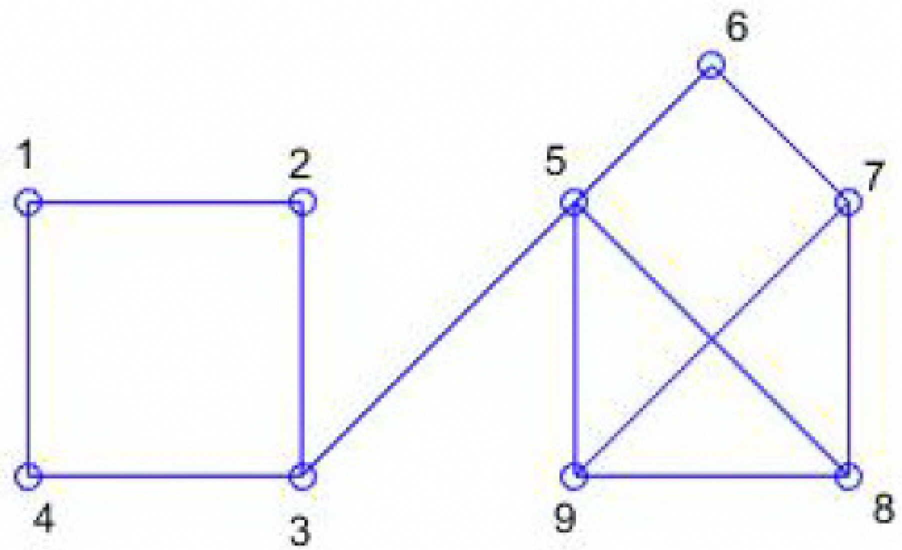
https://webeep.polimi.it/mod/folder/view.php?id=403937

*The clusters above are just example and **do not correspond** to the expected results of the challenge!*



14

Task 1 - We start by considering an example of a small graph. Using *Eigen*, create a new adjacency matrix A_g corresponding to the simple graph here on the right. **Report the Frobenius norm of A_g.** (2 punti)



4.89898

15

Task 2 - (3 punti)

Construct the vector $\boldsymbol{v}_g$ such that each component v_i is the sum of the entries in the i-th row of matrix A_g. Compute the graph Laplacian by performing the following operations:

$$L_g = D_g - A_g,$$

where D_g is diagonal matrix with the vector $\boldsymbol{v}_g$ on the main diagonal and zeros everywhere else. Compute the matrix-vector product $\boldsymbol{y} = L_g\, \boldsymbol{x}$ with $\boldsymbol{x} = [1, 1, \ldots, 1]^T$ and **report the Euclidean norm of $\boldsymbol{y}$. Is matrix L_g symmetric positive definite**?

Euclidean norm of y: 0
No, Lg is not symmetric positive definite, but is symmetric SEMI positive definite because its lowest eigenvalue is 0 (or machine eps)

16

Task 3 - Using the proper eigensolver available in the *Eigen* library, find the eigenvalues and eigenvectors of the matrix L_g. **Report the smallest and largest (in modulus) computed eigenvalues.**
(3 punti)

min = 2.04895e-16
max = 5.67038

17

Task 4 - (2 punti)

Observe that one of the eigenvalues of the matrix L_g is zero (this corresponds to an eigenvector proportional to $\boldsymbol{x}$). All other eigenvalues of the matrix are positive. **Report the smallest strictly positive eigenvalue of L_g and the corresponding eigenvector**. Check that the positive entries corresponds to the nodes $\{1; 2; 3; 4\}$ and the negative ones to $\{5; 6; 7; 8; 9\}$ (or viceversa) which would be an intuitive clustering choice.

smallest strictly positive eigenvalue: 0.28031
corresponding eigenvector:
 0.455542
 0.391696
 0.218053
 0.391696
-0.190354
-0.308704
-0.340521
-0.308704
-0.308704

18

Task 5 - (2 punti)

Apply the previous technique to a larger graph for which a clustering strategy is not obvious. Using `Eigen`, load the adjacency matrix A_s stored in the file `social.mtx` corresponding to a network of 351 Facebook friends. The $(i;j)$-th element is equal to 1 if the i-th and j-th friends are also friends with each other and is equal to 0 otherwise. **Report the Frobenius norm of the matrix A_s.**

93.819

19

Task 6 - (3 punti)

Repeat the procedure we performed previously on the smaller graph. Create the matrices D_s and L_s which are the vector of row sums of the matrix A_s and the Laplacian of the corresponding graph, respectively. Check the symmetry of L_s and **report the number of nonzeros entries in L_s.**

Ls is symmetric, nnz = 9153

20

Task 7 - (3 punti)

In order to make the graph laplacian matrix invertible, add a small perturbation to the first diagonal entry of L_s, namely $L_s(1,1) = L_s(1,1) + 0.2$. Export L_s in the `.mtx` format and move it to the `lis-2.1.10/test` folder. Using the proper iterative solver available in the LIS library compute the largest eigenvalue of L_s up to a tolerance of 10^{-8}. **Report the computed eigenvalue and the iterations counts.**

eigenvalue = 6.013370e+01
number of iterations = 2007

21

Task 8 - (2 punti)

Find a **shift** $\mu \in \mathbb{R}$ yielding an acceleration of the previous eigensolver. **Report** μ **and the number of iterations required to achieve a tolerance of** 10^{-8}.

mu = 29.55
Iteration Count : 1063

22

Task 9 - (3 punti)

Use the proper iterative solver available in the LIS library to identify the second smallest positive eigenvalue and the corresponding eigenvector setting a tolerance of 10^{-10}. **Report the computed eigenvalue and the iteration counts**

eigenvalue = 1.789070e+00
number of iterations = 113

23

Task 10 - (2 punti)

Sort the vertices of the graph according to the positive and negative coefficients of the eigenvector computed in the previous point. **Report the number** n_p **of positive entries and the number** n_n **of negative entries.**

np: 52
nn: 299

24

Task 11 - (3 punti)

In Eigen, construct the permutation matrix P which reorders the nodes of the network according to the previous clustering strategy (first the n_p nodes associated to positive coefficients, then the n_n nodes associated to negative ones). Computed the reordered adjacency matrix $A_{ord} = P\ A_s\ P^{\mathrm{T}}$. **Report the number of nonzero entries in the non-diagonal blocks** $A_{ord}(1 : n_p, n_p + 1 : n_p + n_n)$ **and** $A_s(1 : n_p, n_p + 1 : n_p + n_n)$.

nnz in A_ord: 332
nnz in A_s: 1162

25

Task 12 - Comment the obtained results. Has the clustering procedure been effective? (2 punti)

According to Fiedler's observations [1], the eigenvector corresponding to the second smallest eigenvalue (called "algebraic connectivity of the graph") of L_s gives a natural bipartition of an undirected graph, and it is strongly related to the problem of minimizing the cut between two subgraphs N1 and N2 of the original graph.
What we get from permutating the matrix A_s accordingly to the positive and negative entries of the eigenvector is a block matrix A_ord [2], in which in the diagonal block matrices we have the edges between nodes in the same cluster, whereas in the non-diagonal block we get the edges that cross the bipartion.
According to our results, the number of edges across the partition is 332, which is less than the number of edges crossing the partition before the permutation (1162).
So our procedure has well clustered the initial data set. The reason why the number of edges doesn't seem to reduce largely may be related to the fact that the smallest eigenvalue isn't very close to 0, the multiplicity of which indicates the number of components connected within the graph [2].

[1] A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory, M.Fiedler
[2] A Tutorial on Spectral Clustering, U. von Luxburg