Intensive Computation – Homework 3

23 April 2021

Exercise 1

Write the **function `epair`** that:

- takes a matrix as **input**
- computes the largest absolute eigenvalue and the corresponding eigenvector implementing the **Power method**
- gives as **output** the maximum absolute eigenvalue and the corresponding eigenvector

Write the **function `deflation`** that:

- takes as **input:** a matrix, the largest eigenvalue and the corresponding eigenvector
- computes the second largest eigenvalue and the corresponding eigenvector implementing one **Deflation method**
- gives as **output** the second eigenvalue and the corresponding eigenvector

To compare the results obtained calling the two functions **`epair`** and **`deflation`** with those produced by the function **`eig`** of Matlab, compute the difference between the largest eigenvalue obtained with the two methods (error) averaging on sets of random matrices of different size (for example 25, 50, 75, 100).
Plot the values of errors on a graph, showing also the *error bar* (to show the minimum and maximum values).

Exercise 2 – Gould index for eigenvector centrality and Fiedler eigenvector for Graph Partitioning

Choose one of the two following exercises

a) Write a script that:
   - defines a **random adjacency matrix M** corresponding to a graph, representing a set of towns and the travel routes among these towns
   - determines the most accessible town using the Gould index, calling the **function `epair`** to obtain the principal eigenvector x1 on the appropriately modified adjacency matrix
   Reference: http://matrixapps.blogspot.it/2010/07/gould-index-matrix-application-to.html

b) Write a script that:
   - partitions the graph into two parts according to **the Fiedler eigenvector** (eigenvector corresponding to the second smallest eigenvector) obtained by considering the Laplacian matrix and using the **function `epair`** and the **function `deflation`**
   - gives as **output** four subplots in a graphical window, one with the representation of the considered graph where the most accessible town is highlighted, the other three with the two parts of the bi-partitioned graph are represented using different colors using the three method to divide the Fiedler eigenvector (positive-negative, mean, median).

Observations Use **`graph`** to define graphs and **`gplot`** to draw.

Exercise 3 – Hooke model 1D

Write a script that:

- Considers a one-dimensional system with at least 4 particles (mass m=1 and same fixed diameter for all particles, fixed spring length d, fixed elastic constant k)
- Simulates the evolution of the system according to the Hooke's Law model, using:
  - the solution with Euler or Verlet (at your choice) method for differential equation
  - the exact solution obtained using eigenvalues and eigenvectors
- Produces the picture of the motion of each particle with time on the horizontal axes and particle position on the vertical axis, for both the approximated solution and the exact solution (an example of graph for four particles is shown in Figure 6 page 15 on Molecular Dynamics - L. Fosdick)
- Produces a film of the particles' motion.

Optional - Exercise 4 – Hard sphere model 1D

Write a script that:

- Considers a one-dimensional system with at least 5 particles (mass m=1 and given diameter)
- Simulates the evolution of the system according to the Hard Sphere model, assigning the position and velocity to each particle, and assuming that the motion line is limited by a barrier both on the left and on the right sides
- Produces a film of the particles' motion.