

CISC 271, Winter 2021

Assignment #1: Binary Clustering of Graph Vertices Due by 11:30AM on Friday, January 22, 2021

Please read the details and instructions carefully before you begin to work on the problem. The question in this assignment is relatively straightforward because it is intended to be a practical introduction to MATLAB and to writing a report in this course. There must be a single results section and a single discussion section on your report. The results section of the report must contain one table and one figure; more of either, or fewer tables, will produce deductions from your grade on this assignment.

Statement of Academic Integrity

This assignment is copyrighted by the instructor, so unauthorized dissemination of this assignment may be a violation of copyright law and may constitute a departure from academic integrity.

Sharing of all or part of a solution to this assignment, whether as code or as a report, will be interpreted as a departure from academic integrity. This includes sharing of the assignment after the due date and after completion of this course.

Learning Outcomes

On completion, a successful student will be able to:

- Use `for` loops in MATLAB
- Transform an edge list to an adjacency matrix and a Laplacian matrix
- Use the Fiedler vector to assign vertices in a graph to one of two clusters

Question 1: Graph Clustering 4% of Final Grade

As described in the course notes, the Fiedler vector of a Laplacian matrix of a graph can be used to cluster the vertices of a graph into two sets. Set 1 contains the indices of vertices that have a negative Fiedler entry and Set 2 contains the indices of vertices that have a positive Fiedler entry.

The problem is: given an edge list, use a `for` loop to create a Laplacian matrix for the graph and then cluster the vertices of the graph. The input is an array in which each row is a pair of integers that index the list of vertices. The output is a vector that indicates the clustering of the vertices, in which each entry is -1 for Set 1 and +1 for Set 2.

For this problem, you will need to modify the instructor's "starter" code that is in the accompanying ZIP file. The starter code will return a constant clustering vector, intended to act as a guide for you in learning how a MATLAB function can return values.

For testing, a randomly generated graph has been provided. Diagrams of the test graph are shown in Figure 1. There are two files that you can use to test your code: `testedge.txt` contains edges of the graph, and `testsets.txt` contains the correct set vector. You can check your results, replacing `xxxxxxxx` with your student number, using the MATLAB commands

```
load('testedge.txt');  
a1_xxxxxxxxx(testedge)
```

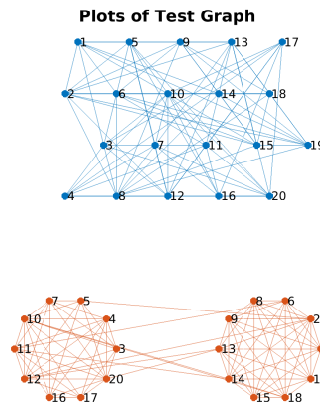


Figure 1: Diagrams of the test graph. The upper part shows the graph in a rectilinear grid. The lower part shows the graph when plotted according to a computed clustering of the vertices.

The clustering vector has entries ± 1 , which is useful for checking your code but which is less useful for a human reader. The clusters of the vertices for the test graph are shown in Table 1.

Table 1: Clusters of vertices for the test graph. Each row contains the vertices of the cluster.

Set	Vertices									
1	1	2	6	8	9	13	14	15	18	19
2	3	4	5	7	10	11	12	16	17	20

Grading Guide:

We will test your code on your individual, randomly generated edge list. The edges for everyone are in the file `alfiles.zip`; your data file is coded as your Queen's netid as indicated in onQ. A correct clustering of "your" graph will be part of your grade for this assignment. This means that your single figure should be the diagram produced for your individual edge list, and your single table should be the clustering of vertices for your individual edge list.

The TA's have been instructed to use this guide when they mark your assignment. Your grade will be based on the numerical results and on the report. The distribution of points for the assignment grade are:

4/16 points: numerical values that are produced by the code and that are presented in the results

4/16 points: quality of the code in the modified "starter" functions, and any other changes in the submission file that was used to generate values and plots for the report

8/16 points: quality of the report, especially including the figures and descriptions; clarity may be assessed, in part, by the written discussion of results

Grading Considerations

- The quality of your report will be considered. You need, at minimum, to conform to the “student version” of the report style in the onQ website; you may wish to consider the “grader version” that we will use for assessing your report.
- The quality of your MATLAB code will be considered. Your code should be appropriately indented, sufficiently commented, and otherwise be appropriate software.
- The output of your code will be considered.
- Your code can use functions provided by MATLAB, but the code that you submit *must* be your original work. You may not use any builtin functions that perform optimization or clustering such as k-means, although you may wish to use such functions as you develop and test your software.
- Code that causes MATLAB to produce an error or warning will result in a failing grade.

What to turn in:

- You will submit your answers electronically as two files. The code will be tested by one or more graders. The PDF report will be read by one or more graders and will be checked, using electronic methods, to ensure that it meets professional standards for originality.
- The code must be in one MATLAB file (`al_XXXXXXXX.m`). This file will contain all of the code needed to verify that the values and tables in the report can be reproduced. The functions must produce the values for your table and for each figure.
- Your functions must take one argument, return one value, and require no user input or action such as using the “enter” key. Running this function should produce, on the console, every value that is in the report; the function should also produce every plot that is in your report, each plot in a separate figure. The function should produce no other values or figures. The graders will compare your computed values to the values in the report and may deduct marks from the report for differences between any reported value/plot and the corresponding computed value/plot.
- The report must be in a single PDF file (`al_XXXXXXXX.pdf`). The PDF file must include a description of how you tested your code. You can also include notes, comments on the problems, and assumptions you have made, as appropriate.
- The assignment must be submitted using the Queen’s “onQ” software.

Policies:

- You must complete these questions individually.
- Although you are allowed to discuss the questions with other students, you must write your own answers and MATLAB code.
- The syllabus standards apply to this assignment.
- Lateness policy applies starting the minute after the submission deadline, at a rate of 20% off the assignment value per calendar day. *Please note: the time in the onQ system is beyond your control, so submitting within an hour of the deadline is inherently a risky process for which you assume full responsibility.*