CS-E4650 Methods of Data Mining

Exercise 5 / Autumn 2025

Recall that it is forbidden to use any AI tools to solve the tasks, input course material (including tasks and their solutions) to AI tools or otherwise distribute them to other services.

5.1 Basics of social network analysis

Learning goals: Understanding the key ideas of two centrality measures, spectral community detection, and the Girvan–Newman community-detection method.

For this exercise task, you are not expected to write any code.

Let $N = \{A, B, C, D, E, F, G, H, I, L, M\}$ be a set of nodes. Consider the weighted and undirected graph $G = (N, \mathbf{W})$ shown in Figure 1 and specified by the following weight matrix (or weighted adjacency matrix):

$$\mathbf{W} = \begin{pmatrix} 0 & 3 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 5 & 0 & 7 & 2 & 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 7 & 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 & 4 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & 5 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

You may assume that the graph G describes a social network. In this exercise task, you will investigate the centrality of the nodes in G according to two different measures and the partitioning of G into two communities with two different methods.

a) Recall that the weighted degree of a node is obtained by summing the weights of all edges incident to it. Which node has the highest (weighted) degree centrality?

b) Recall that betweenness centrality can take weights into account by defining path length to be the total weight of a path. Which node do you *expect* to have the highest (weighted) betweenness centrality

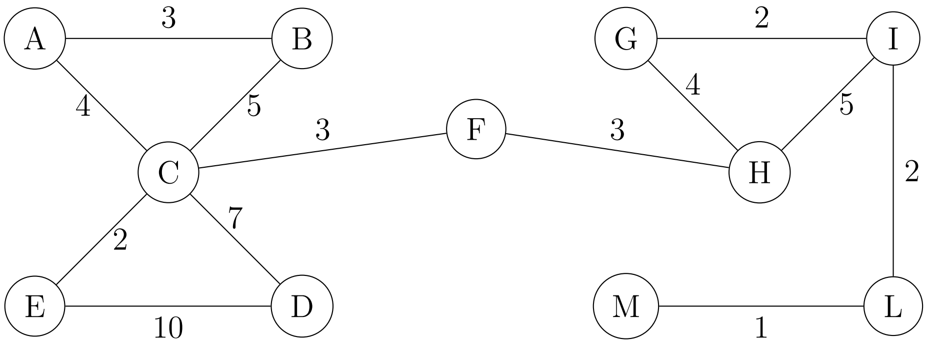

Figure 1: Graph G.

and why? You do not need to compute the scores, use your intuition instead.

c) Approximate the (weighted) betweenness centrality of node F (*Hint: simulate random sampling with sample size* ≈ 10). If the exact (weighted) betweenness centrality of node F is 0.55, what is the approximation error associated with your estimate?

d) Compute the Laplacian matrix associated with graph G, namely $\mathbf{L} = \boldsymbol{\Lambda} - \mathbf{W}$ where $\mathbf{W}$ is the weight matrix given above and $\boldsymbol{\Lambda}$ is a diagonal matrix with $\boldsymbol{\Lambda}_{i,i} = \sum_{j=1}^{n} w_{i,j}$. Check the correctness of your computations by verifying that each row of $\mathbf{L}$ sums up to zero.

e) The eigenvalue decomposition of the Laplacian matrix $\mathbf{L}$ gives the following eigenvalues (sorted in increasing order):

$$\lambda = \begin{pmatrix} 0 & 0.388 & 1.080 & 3.377 & 4.319 & 5.732 & 9.737 & 10.476 & 16.786 & 21.162 & 28.942 \end{pmatrix}$$

The corresponding eigenvectors are the columns of the following matrix:

$$\mathbf{V} = \begin{pmatrix}
0.30 & 0.28 & -0.16 & -0.05 & 0.53 & -0.15 & 0.04 & 0.69 & -0.02 & -0.13 & 0.10 \\
0.30 & 0.28 & -0.16 & -0.04 & 0.45 & -0.11 & -0.01 & -0.72 & -0.04 & -0.20 & 0.15 \\
0.30 & 0.26 & -0.12 & -0.01 & 0.02 & 0.04 & -0.02 & -0.05 & 0.08 & 0.61 & -0.67 \\
0.30 & 0.28 & -0.16 & -0.05 & -0.44 & -0.08 & 0.01 & 0.02 & 0.01 & 0.43 & 0.65 \\
0.30 & 0.29 & -0.17 & -0.06 & -0.57 & -0.12 & 0.02 & 0.05 & -0.05 & -0.61 & -0.30 \\
0.30 & 0.08 & 0.15 & 0.22 & 0.02 & 0.86 & -0.17 & 0.04 & 0.21 & -0.13 & 0.09 \\
0.30 & -0.15 & 0.45 & 0.29 & -0.02 & -0.42 & -0.62 & 0.01 & 0.21 & -0.01 & 0.00 \\
0.30 & -0.11 & 0.37 & 0.20 & -0.00 & 0.04 & 0.23 & 0.00 & -0.81 & 0.06 & -0.02 \\
0.30 & -0.19 & 0.36 & -0.02 & -0.01 & -0.13 & 0.70 & -0.03 & 0.49 & -0.02 & 0.00 \\
0.30 & -0.39 & 0.05 & -0.84 & 0.01 & 0.11 & -0.21 & 0.01 & -0.07 & 0.00 & 0.00 \\
0.30 & -0.63 & -0.62 & 0.35 & -0.00 & -0.02 & 0.02 & -0.00 & 0.00 & -0.00 & 0.00
\end{pmatrix}$$

Partition the nodes in two communities based on the sign of the eigenvector associated with the second smallest eigenvalue (also known as Fiedler vector).

f) What partitioning into two communities do you *expect* the Girvan–Newman algorithm to output and why? You do not need to compute the scores, use your intuition instead.

5.2 PageRank and HITS

Learning goal: The idea of PageRank and HITS algorithms.

Let us consider a toy example, where the entire graph of web pages is presented by Figure 2. Table 1 lists the keywords that occur in the pages. Pages 1–7 are learning material on ranking algorithms, while pages 8–10 are about Star Trek (a sci-fi TV series).

The task is to evaluate PageRank and hubs and authority values of pages. In this task, you can implement the algorithms yourself or use any of the existing PageRank and HITS simulation tools (you can find also online calculators). (For better learning, it is recommended to use a tool that shows how the values are updated at each round.) Note that different tools may use different initialization or scaling, but the top results should be the same.

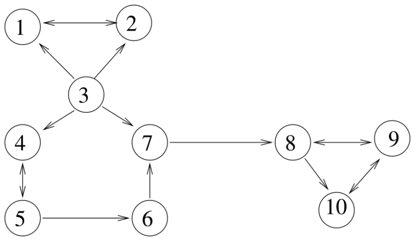

Figure 2: Linkage structure of web pages.

a) Evaluate PageRank values for all pages using teleportation probability $\alpha = 0.10$. Can you explain why certain pages receive high scores? How does the situation change when you increase the teleportation probability?

b) Construct the HITS graph (base set and its edges) for query "PageRank". Include pages in the root set, all pages pointed by the root pages and all pages pointing to the root pages. Evaluate first visually

Table 1: Keywords that occur in pages 1–10.

id	keywords
1	authority, page, in-link, reputable, source
2	hub, page, out-link, good, source
3	PageRank, HITS, ranking, algorithm
4	reputable, page, in-link, PageRank
5	reputation, visit, frequency, random, surfer
6	random, surfer, trap, dead-end
7	PageRank, teleportation, random, surfer, model
8	teleportation, travel, planet
9	Star Trek, tv, series, transporter
10	beam, Scotty, transporter

which are good hub and authority pages in the resulting graph. Then calculate the hub and authority values with some tool and compare the results to your previous intuition. Which pages would be returned to the user, if the search engine used i) HITS or ii) PageRank?

c) Repeat the b) part with query "teleportation"! Is the search working correctly? You may observe signs of "topic drift", where a densely connected component of irrelevant pages becomes included in the query graph and gathers the highest scores.

5.3 Collaborative filtering for movie recommendations

Learning goal: How to use neighbourhood-based collaborative filtering in recommender systems.

Table 2 presents movie ratings by 5 users on 5 movies. The ratings are between 1 (didn't like at all) to 5 (fantastic movie) and 0 means a missing rating (the user hasn't watched the movie). The users are notated $u1, \ldots, u5$ and movies $m1, \ldots, m5$. The task is to apply recommender systems for rating prediction using neighbourhood-based collaborative filtering (see Aggarwal 18.5.2 and an example in the lecture slides).

a) Calculate mean ratings per user. Use all non-missing ratings in the calculation. These are needed in parts b) and c).

b) Calculate required pairwise similarities between users using the modified Pearson correlation r ("Pearson" in Aggarwal Equation 18.12). Use the mean values calculated in part a). Remember that the correlation is calculated only over co-rated movies.

c) Predict the missing ratings using two nearest neighbours ($K = 2$) and an extra requirement that the similarity is $r \geq 0.5$. Tell if the movie is recommended to the user (if the user would like it more than average).

Report if some prediction cannot be made (not enough sufficiently similar neighbours with required ratings).

d) Neighbourhood-based recommendations could also be done in the item-based way. Which way, user-based or item-based, is likely more accurate, if

i) there are a lot of users but relatively few items?

ii) the items are often changed (but the users are relatively stable)?

Justify your answer!

Table 2: Movie ratings (scale 1–5) by 5 users ($u1$–$u5$) on 5 movies ($m1$–$m5$). Special value 0 means a missing rating.

	$m1$	$m2$	$m3$	$m4$	$m5$
u_1	0	4	5	4	4
u_2	4	3	4	3	3
u_3	2	1	3	2	4
u_4	3	2	3	0	3
u_5	4	3	5	4	0

5.4 Homework: Eurovision Song Contest

Recall that this task is homework that is done in groups of 2–3 students – you cannot do the task alone or in a larger group. You can search collaborators in Zulip, exercise sessions, or ask help from the TAs.

Learning goal: Understanding important characteristics of real-world social networks.

In this exercise task, we will perform social network analysis of country-level voting data collected during the final of Eurovision Song Contest 2018 (ESC). In this singing competition, each country is represented by a singer and, to determine the winner, countries are allowed to give a fixed amount of points to other countries of their choice.

The data are given as a graph $G = (N, \mathbf{W})$, where each node represents a country and (directed and weighted) edges represent points assigned from a country to another. More specifically, each edge is represented by a triplet (u, v, w) where u is the source node (country), v is the target node (country) and w is the amount of points given by country u to country v.

You will have to make extensive use of the `networkx` Python package which implements several popular utilities for social network analysis. You are encouraged to look at the documentation of the `networkx` package[1], which contains everything you need for this exercise task!

To start with, you can find in MyCourses example code for loading and visualizing the social network graph.

a) Compute the average clustering coefficient in G (remember to consider edge weights; many `networkx` functions accept a `weight` argument which allows to take weights into account). Then, construct a graph G_{random} 100 times with the same number of edges and nodes and total edge weight as G, but with edge endpoints sampled uniformly at random. (*Hint: you can iterate through the edges of G and re-use the weights of each edge...*). How often do the 100 random graphs have a

[1] `https://networkx.org/documentation/stable/reference/index.html`

higher average clustering coefficient than G? What property of real-world social networks does the comparison of G and the 100 graphs G_{random} in terms of average clustering coefficient reveal?

b) Plot the in-degree distribution of G (remember to consider edge weights, so that in-degree of node i is defined as the sum of the weights of its incoming edges) and (visually) compare it with the degree distribution of a single G_{random} (*Hint: the sample size to estimate the degree distribution is small; use an histogram with a relatively small number of bins for visualization...*). What property of real-world social networks does the comparison of G and G_{random} in terms of degree distribution reveal?

c) Construct an undirected graph G_{u1} by dropping the directions and weights of the edges. Include an edge in G_{u1} only if giving points has been *bidirectional* between the corresponding two countries. Partition G_{u1} into two communities $C_{1,1}$ and $C_{1,2}$ using the Girvan–Newman algorithm. Do you think that the output partitioning is meaningful or do you notice any issue?

d) Construct another undirected graph G_{u2} by again dropping the directions and weights of the edges, but now include an edge if *either one or both* of the countries has given points to the other one. Partition G_{u2} into two communities $C_{2,1}$ and $C_{2,2}$ using the Girvan–Newman algorithm. How are the found communities different from those in the previous subtask? Do you think that the output partitioning is meaningful or do you notice any issue?

e) Construct third undirected graph G_{u3} by using the sum of *unidirectional* points as edge weights. Partition G_{u3} into two communities $C_{3,1}$ and $C_{3,2}$ using the Kernighan–Lin algorithm. Is the issue possibly identified in the previous subtasks for the Girvan–Newman algorithm solved by the Kernighan–Lin algorithm?

f) Form maximal intersections of the communities found in subtasks c)—e) to see what could be the "*cores*" of the two communities and which countries would reside outside of the cores.

g) Does it seem that countries that are geographically close to each other tend to belong to the same mutual voting community? Explain your findings and modify the `draw_eurovision_map` function to visualize the network with nodes colored by their respective communities obtained in all above subtasks c)—f)!

Parts of the report:

1. Cover page/beginning: title (course name and assignment number), names and student ids of all participants of the team.
2. Section “Methods”: Explain with sufficient mathematical notations and visualizations i) average clustering coefficient, ii) edge betweenness, iii) Girvan–Newman algorithm, iv) Kernighan–Lin algorithm.
3. Section “Results”: Present here your answers to all subtasks with visualizations, findings, and analyses. Include a separate subsection for each subtask!
4. Section “Girvan–Newman algorithm and weighted graph”: Explain why the sum of unidirectional points as edge weights might not have been a good idea for the Girvan–Newman algorithm. How instead could the point values have been used?
5. Section “Conclusions”: Summarize your main findings briefly.
6. Section “Appendix”: Include here the code you used to produce your results.

Produce a PDF report containing all parts as described above and submit it in MyCourses before the deadline. You can find a LaTeX template for the report in MyCourses, under section “Exercises”. One submission per group!