Social Network Analysis for Computer Scientists Fall 2020 — Assignment 1

http://liacs.leidenuniv.nl/~takesfw/SNACS

Deadline: September 28, 2020

This document contains two exercises that each consist of various numbered questions that together form Assignment 1 of the Social Network Analysis for Computer Scientists course taught at Leiden University.

For each question, the number of points awarded for a 100% correct answer is listed between parentheses. In total, you can obtain 100 points and 10 bonus points. Your assignment grade is computed by dividing your number of points by 10. Please do not be late with handing in your work. You have to hand in the solutions to these exercises **individually**. Discussing the harder questions with fellow students is allowed, but writing down identical solutions is not. Hand in your solutions, typeset using LaTeX, as instructed on the course website.

Clearly and concisely describe how you obtained each answer. Write down any nontrivial assumptions that you make. For the exercises that require some programming, you can use any programming language, scripting language or toolkit. In any case, always clearly describe which toolkit or programming language you used and how you obtained your answer using these tools. Include relevant source code, for example, in an Appendix that you reference in the text where needed. When asked for an algorithm, use simple and consistent pseudo-code.

Questions or remarks? Ask your questions during one of the weekly lectures or lab sessions, on the Brightspace discussion board, or send an e-mail.
Good luck!

Exercise 1: Neighborhoods (40p)

A directed network $G = (V, E)$ consists of a set of nodes V and a set of directed links E. For the number of nodes $|V|$ we use n, and the number of links $|E|$ will be denoted by m. The neighborhood $N(v)$ of a node $v \in V$ is defined as the set of nodes to which v links:

$$N(v) = \{w \in V : (v, w) \in E\}$$

Similarly, the reverse neighborhood $N'(v)$ can be defined as the set of nodes that link to node v:

$$N'(v) = \{u \in V : (u, v) \in E\}$$

The notion of a neighborhood can be extended by defining the neighborhood of a *set* of nodes W as:

$$N(W) = \{w \in V : v \in W \wedge (v, w) \in E\}$$

For convenience, for a node $v \in V$ we say that $N(v) = N(\{v\})$. Next, we say that the k-neighborhood $N_k(W)$ is defined as all nodes that are between 0 and k steps away from nodes in W. For the case $k = 0$ we have $N_0(W) = W$. Then for $k > 0$ we have:

$$N_k(W) = N(N_{k-1}(W)) \cup N_{k-1}(W)$$

Essentially, the k-neighborhood allows us to apply the neighborhood function to a set of nodes k times. Using these notions, it is possible to define other measures, procedures and algorithms.

(2p) Question 1.1 Give a formal definition of the *indegree* and *outdegree* of a node using the notion of a (reversed) neighborhood.

(3p) Question 1.2 In a directed network, the *combined degree* of a node is the number of neighbors connected to that node through either an incoming or an outgoing link. Give a formal definition of the combined degree using the notion of a (reversed) neighborhood.

(4p) Question 1.3 What aspect of a directed network is measured by the following equation?

$$\frac{1}{m} \sum_{v \in V} |N(v) \cap N'(v)|$$

(6p) Question 1.4 Write down an equation that counts the number of *directed triangles* of three nodes in a given directed network using the notion of a (reversed) neighborhood.

(5p) Question 1.5 Write down an equation that measures the number of nodes that reside in a given network's *largest weakly connected component*, using the notion of a (reversed) neighborhood.

Assume from now on that the network has a symmetric edge set, modeling that it is undirected. Also assume there is one connected component.

(5p) Question 1.6 Give a formal definition of a node's *closeness centrality* value using the notion of a (k-)neighborhood.

(6p) Question 1.7 Networks may contain *cliques*. Write an algorithm that checks if a given set of nodes $W \subseteq V$ in an undirected network is a clique, and whether this clique is *maximal*.

(9p) Question 1.8 In the human contact network of a population, a so-called *predictable pandemic* may take place. Starting from one infected node at week $t = 0$, in each subsequent week $t + 1$, all infected nodes infect all of their uninfected neighbors. Give an equation or an algorithm that computes the minimal and the maximum number of weeks t that it may take before the entire population is infected, regardless of which node was initially infected.

Exercise 2: Mining An Online Social Network (60p)

This is a practical exercise, for which you can use any toolkit or programming language. Two social network datasets can be found at
`https://liacs.leidenuniv.nl/~takesfw/SNACS/medium.tsv` and
`https://liacs.leidenuniv.nl/~takesfw/SNACS/large.tsv`.
Each file contains a list of social network friendships of the form
`userA[tab]userB[newline]`
A line thus represents one directed link from a person identified by `userA` to a person identified by `userB`. You may assume that these identifiers are integers that fit in a 4-byte `signed int` in C++. Processing the file `medium.tsv` should be possible on a student workstation with 16GB memory, using for example GEPHI. A larger online social network is given in the file `large.tsv`, which will likely not be processable using standard toolkits such as GEPHI, requiring the use of for example NETWORKX or another CLI package.

Answer each the following six questions for `medium.tsv` and `large.tsv` (hence, up to Question 2.6, points are also given 2×). Remember to write down how you obtained your answer, for example by including pointers to relevant Appendix source code. Display diagrams with properly scaled axes, labels and captions. A histogram or scatter plot can be generated using GNUPLOT or MATPLOTLIB. Remember to use proper captions, axis labels and scaling in figures.

(2×2p) Question 2.1 How many directed links does this network have?

(2×2p) Question 2.2 How many users (nodes) does this social network have? Hint: a node counts as a node if it is a source or a target of a link.

(2×4p) Question 2.3 Give the indegree and outdegree distribution of this network using a diagram.

(2×4p) Question 2.4 How many weakly connected components and strongly connected components are there? How many nodes and links do the largest strongly and largest weakly connected component have? (6 answers)

(2×2p) Question 2.5 Give the exact or approximated average clustering coefficient of this network.

(2×6p) Question 2.6 Give the exact or approximated distance distribution of the largest weakly connected component of this network as a diagram.

(20p) Question 2.7 Visualize the social network in `medium.tsv`. Give the size and the color of a node a sensible meaning based on node centrality, and describe your choices. State which visualization algorithm you used and how you have chosen its parameters. Include your visualization as a proper full-page A4 vector graphic PDF in your report.

(10p, bonus) Question 2.8 This dataset contains over 5 million nodes and 1 billion edges:
`/vol/share/groups/liacs/scratch/SNACS/huge.tsv`
`/data/SNACS/huge.tsv`
The files are identical, where the first is in the university-provided remote Linux environment and the second in the LIACS data science lab environment.
Answer Questions 2.1 through 2.6 above for this dataset. You will need to use approximation and/or a more advanced software package and environment (e.g., GRAPH-TOOL or SNAP), or write efficient code yourself.