Analysis and Modeling of Social and Information Networks
CIS 4524/5524, Spring 2021

Assignment 1, due January 28 by 5pm on Canvas

*Please write your name and TUID at the top of your **CANVAS** submission.

Homework Policies (applicable for all assignments):

1. You are required to do the homework problems in order to pass.

2. Understandability of the solution is as desired as correctness.

3. Penalty for late homework assignments submissions is 20% per day. So, do it on time.

4. Solutions are expected to be your own work. Group work is **not allowed** unless explicitly approved for a particular problem. If you obtained a hint with help (e.g., through library work, etc.) acknowledge your source, and write up the solution on your own. Plagiarism and other anti-intellectual behavior will be dealt with severely.

A table attached to this homework is generated according to the background survey for this course. Each of you is representing a raw and column values are 1 if you have some background on (1) Data Mining (CIS 4523/5523) or Machine Learning); (2) Python or R programming; (3) Graphs or Statistics.

Problem 1. [Visualizing a Multilayer Network] In this exercise your job is to visualize a 3-layer course network, where nodes represent students, and each layer corresponds to a single topic network representing one type of relationship among students taking this course (e.g. in layer 1 nodes representing two students should be connected if both students took a Data Mining/Machine Learning course). It is highly recommended to use Python library NetworkX. Check sections 'Creating a graph' and 'Drawing graphs' to see how to visualize a graph. There is also a specific libraries and software to visualize multilayer networks i.e. Pymnet: http://www.mkivela.com/pymnet/ , multinetx: https://github.com/nkoub/multinetx, and MuxViz: http://muxviz.net/ . You may also use any other software or library that is not mentioned. Make a screenshot of your visualized networks.

Problem 2. [Visualizing a Weighted Network] Visualize a network obtained by projecting the 3-layer network from Problem 1 to a single-layer weighted network, where two students are linked by a ***weighted*** edge representing the number of topics both students took.

Problem 3. [Visualization of a Bipartite Network] Visualize a bipartite students-topics network where an edge between a student node and a topic node exists if and only if this student has taken a course on that topic.

Problem 4. [Computing Global Network Properties] For each layer of the 3-layer network constructed in Problem 1 compute the following global network properties:

a) the size and diameter of the network largest connected component,
b) degree distribution,
c) average path length and
d) clustering coefficient.

Repeat this for the network constructed in Problem 3.

All metrics can be calculated using Python library NetworkX, but you can use any freely available packages to compute these properties or you develop your own code.