HW1 (due 26 Jan @ 01:00)

Network Analytics Jan-Feb 2021

INDIVIDUAL HOMEWORK: No consulting anyone or books on the first 3 questions. Solutions will be posted but no individual feedback will be given.

Questions 1,2,3: *You develop a feeling for graphs and their manipulation and objects only by doing some theory. For the following give a proof --- i.e., clear rigorous argument in your own words*

1. (10 points) A directed graph is strongly connected if there is a (directed) path from every node to every other node. Show that in a directed strongly connected graph containing more than one node, no node can have a zero indegree or a zero outdegree.

2. (10 points) Show that every tree is a bipartite graph.

3. (10 points) A directed acyclic graph (DAG) is a directed graph without cycles (the underlying graph may have cycles, so it is not a tree). Show that a DAG has a labeling of its nodes (that is a labeling *1, …, n* of its nodes) such that every arc goes from a lower-numbered node to a higher-numbered node. (You need to find a way to do it, and also show that is always possible)
   (These DAGs are ubiquitous--- in neural networks (used for visual recognition), Hidden-Markov models (used for speech recognition) and in general, graphical models.)

4. NetworkX Programming Exercise

Besides the documentation, you are allowed to search forums (but not consult or ask your cohort). The final code has to be written and debugged entirely on your own: Part of the learning objective is to search and choose for the best nx, pandas or numpy function.

PLEASE SUBMIT ALL CODE AS Jupyter html files ONLY

This is based on the data file called HW1_asset_prices.csv. This represents the price movements of a set of assets (bonds, stocks etc., their description is quite irrelevant here).

Economists and investors are very interested in the correlation of asset prices, both to understand risk, as well as (hopefully) find correlation to lagged asset prices for investing. A correlation matrix with N assets is an N × N matrix of correlations. See https://en.wikipedia.org/wiki/Stock_correlation_network for some background information.

Your task is to *visualize* the correlation matrix in network form. You are free to use any python package for your calculations in **a** (numpy, scipy, pandas etc.) as you see fit. For **b** and **c**, I expect you to use the `matplotlib.pyplot` interface in NetworkX.

Tip: Get a skeleton code working to do the tasks (can be done in under 10 lines of code with the right built-in functions in pandas and networkx) and then enhance and explore from there on.

a. (10 points) Calculate the correlation matrix (you may have to use your Stats and Econometrics knowledge here)

b. (10 points) With assets as nodes, plot the matrix as a network. Experiment with the different graph layouts and choose the one that you believe is best, and explain why

c. (10 points) Enhance your plot by representing the thickness of the edges and size of the nodes to give some insight.