CS 7280: Network Science

Assignment-2

Learning Objectives

The objective of this assignment is to experiment with the concepts we covered in Module-2:

- Degree distribution
- G(n,p) random networks
- Power-law networks
- Small-world networks
- Clustering coefficient and transitivity
- Average path length, diameter, efficiency
- Assortativity
- Network motifs
- Degree-preserving network randomization

Submission

Please submit your jupyter notebook **Assignment2-YOURLASTNAME.ipynb** with **requirements.txt** so that we may be able to replicate your Python dependencies to run your code as needed. With Anaconda, you can do this by running: **conda list -e > requirements.txt**

Do NOT zip your submission files.

Ensure all graphs and plots are properly labeled with unit labels and titles for x & y axes – there may be point deductions if plots are not properly labeled.

Notes

1) First review the **"powerlaw"** module: https://pypi.org/project/powerlaw/. Please only use what is provided in the "powerlaw" folder – nothing else. Thus, simply type "from powerlaw import *" in your ipynb file. The API is the same as on the website.
2) You are allowed to use the **scipy** package.
3) We recommend you use **"box and whiskers" plots**, so that you can show both the average value and the range of plus/minus one standard deviation around the mean.

Part-0

"blog.txt" represents a directed network that contains hyperlinks between blogs. A node represents a blog and an edge represents a hyperlink between two blogs. The format of each line in the text file consists of two numbers separated by space. The two numbers represent the two nodes and the line

represents an edge from the first node to the second node (i.e., it is a directed and unweighted network). Parse the dataset using **NetworkX** functions.

Part-1 (26 points)

The given network is directed, so repeat the same steps below once with the out-degree and then once with the in-degree.

1. (14 points) Plot the degree distribution in the four possible ways shown in Figure 4.22 of your textbook (separately for out-degree and in-degree) and please ignore all the nodes with degrees 0 in the log-scale plot.
2. (12 points) Because of the presence of **"low-degree saturation"** and **"structural limit"** in real networks, the power-law distribution is usually fit excluding values that are smaller or larger than a certain threshold. Use the **"Fit"** function from **"powerlaw"** to estimate the **exponent of the power-law degree distribution** and **the minimum-x value** for the power-law fit and set **discrete = True** within "Fit" function. Do the estimation twice: once without setting the xmax threshold value, and once setting xmax to remove the maximum outlier value (we recommend you set xmax = 200 for the out-degree distribution and xmax = 300 for the in-degree distribution respectively). We recommend you ignore nodes with zero degree for the log-log scale plots (otherwise the function "Fit" will "complain" giving you several warnings). Overall, you should get 8 values in this part. 4 values for either out-degree or in-degree.

Part-2 (15 points)

1. (6 points) Convert the previous network (from Part-1) into an undirected network. Calculate the **Pearson correlation coefficient** for the *degrees of adjacent nodes*. Based on this analysis, is this network **assortative, disassortative, or neutral**? Make sure that your answer is justified and that you evaluate the statistical significance of your conclusion using the t-test.
2. (9 points) Plot the **average neighbor degree (averaged across all nodes of degree k) as a function of the node degree k**. Calculate the **Pearson correlation coefficient** for node degrees and its average neighbor degrees. Based on this analysis, is this network assortative, disassortative, or neutral? Make sure that your answer is justified and that you evaluate the statistical significance of your conclusion using the **t-test**.

Part-3 (20 points)

1. (8 points) Find **the largest strongly connected component** of the network in Part-1, and convert it to an undirected network. Let us call this undirected network G_0. Create G(n,p) random networks with the same number of nodes and the same number of expected edges as G_0. Then, compare the **diameter** of G_0 with the diameter of 100 such G(n,p) networks (with a plot that shows the distribution of those 100 values – as well as the diameter of G_0).
2. (4 points) Repeat the previous step but this time for the **average shortest path length** instead of the diameter.
3. (8 points) Use the **one-sample t-test** to examine if the diameter of the undirected network is significantly different than the diameter of the random networks at a 95% significance level. Perform the same test for the average shortest path length. Answer for each: Are the values statistically different? Are they within the same order of magnitude?

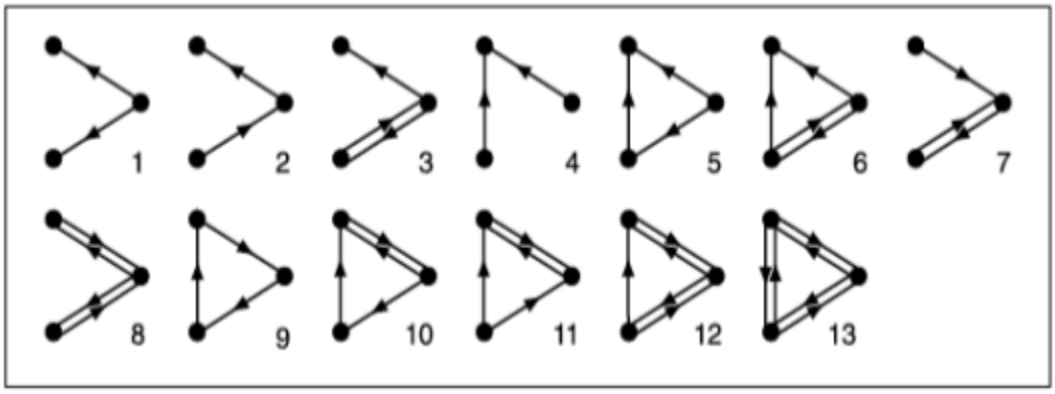
Part-4 (20 points)

1. (5 points) Starting from the network G_0 of Part-3, calculate the **clustering coefficient of each node**, and then plot the C-CDF of the clustering coefficients **(Please show your x-axis and y-axis labels)**. Compare with the corresponding distribution of a random network with the same number of nodes and connection probability (as you did in Part-3). Use the **Kolmogorov-Smirnov** test to compare the two distributions (you can find that test in the **scipy** package).
2. (5 points) Plot the average clustering coefficient **(Please show your x-axis and y-axis labels)**, as a function of the node degree, for both G_0 and for the random network you constructed in Part 4.1. What do you observe?
3. (5 points) Calculate **the transitivity coefficient** of the undirected network. Use a boxplot to compare this value with the transitivity coefficient of the 100 random networks that were constructed in Part-3.
4. (5 points) Combining the results of Part-3 and Part-4, can we conclude that G_0 is **a small-world network**? Why or why not?

Part-5 (15 points)

Recall that there are 13 different types of **directed weakly-connected "node triplets".**

First, count the number of type-5 (feed-forward loop : A->B->C with the additional edge A->C) and type-9 (directed cycle :A->B->C->A) in the **largest strongly connected component** (call it G_1) of the original network.



Second, using **the "directed_configuration_model"** function, generate 10 random networks that have the same number of nodes, edges, in-degree distribution, and out-degree distribution with G_1. Remove any multi-edges between nodes by converting the random networks from "multi-directed graphs" into (single-edge) directed graphs. Also, remove any self-loop edges. Use these 10 random networks to examine statistically which of the previous two triplet types are more (or less) common in G_1 compared to chance.

Part-6 (4 points)

Note that the Transitivity and the Average Clustering Coefficient are two different metrics. They may often be close but there are also some extreme cases in which the two metrics give very different answers. To see that consider a network in which two nodes A and B are connected to each other as well as to every other node. There are no other links. The total number of nodes is n. What would be the

transitivity and average clustering coefficient in this case (you can simplify by assuming that n is quite large)? Points will only be awarded for a mathematical derivation, however you may use code to verify your result.