OMSCS 7280: Network Science Assignment-5

The objective of this assignment is to learn about network models and statistical analysis of network data, covered in Lesson 12 and Lesson 13.

Please submit your Jupyter Notebook **Assignment5-YOURLASTNAME.ipynb and requirement.txt**

Part 1. Modeling the NCAA College Football 2000 Network (65 points)

For the first part of this assignment, you will be working with a network that represents American football games between Division IA colleges during the regular season of Fall 2000.

This network contains 115 nodes and 613 edges.

1.1 Structural Properties of the Graph (18 points)

Here you will be asked to show some of the structural properties of the empirical network.

Load the graph using NetworkX and print the number of nodes and edges to verify that the network is loaded correctly.

A. Calculate the degree sequence (list of degrees of each node) for the network. Plot the degree distribution using a histogram.
B. Identify the community structures with the graph.
   i. To do it, use the Louvain algorithm (as mentioned in Lesson 7) to calculate the best partition. You can use a Python implementation from Louvain Community Detection. Try 10 different resolution parameters from 1 to 10. Compare the partition result with ground truth using normalized mutual information (NMI), and report the resolution value that leads to the highest NMI.
   ii. Based on the communities found with the highest NMI, calculate the inter-community connection density matrix (i.e. matrix P in L12: Generating Networks with Community Structure for the definition). Plot the inter-community connection density matrix as a heatmap.
C. Calculate the network diameter, characteristic path length (CPL), average clustering coefficient, transitivity, and assortativity. Print these values.

1.2 Configuration Model Graph (8 points)

Here you will be working with the Configuration Model graph generator in NetworkX. The main parameter in the configuration model is the degree sequence which you have already calculated for the empirical network above.

A. Generate 100 graphs using the Configuration Model graph generator in NetworkX (In the configuration_model function, use the create_using=nx.Graph() argument to get a Graph and not MultiGraph).
B. Calculate the following properties for each of these 100 graphs: network diameter, CPL, average clustering coefficient, transitivity, and assortativity. Report the distribution of each property among 100 graphs using appropriate plots (histogram or boxplot would be fine).

1.3 Stochastic Block Model Graphs (9 points)

Here you will be working with the Stochastic Block Model generator in NetworkX. This model has two main parameters: a list of community sizes and a matrix representing the inter-community connection density. (Pick the largest connected component if the network you generated is not connected) You have already calculated both for the empirical network in Part 1.1.

A. Generate 100 graphs using the Stochastic Block Model generator in NetworkX.
B. Calculate the following properties for each of these 100 graphs: network diameter, CPL, average clustering coefficient, transitivity, and assortativity. Report the distribution of each property among 100 graphs using appropriate plots.

1.4 Hierarchical Random Graphs (15 points)

Here you will be working with the Hierarchical Random Graphs. We have composed a dendrogram fitted on the empirical network, as in “football-hrg.gml”, using PyHRG. In brief, the dendrogram is formulated as a directed graph. Each leaf node (node with no outgoing edges) in the dendrogram represents a node in the empirical network. Each non-leaf node stores the information about its left/right child (“L” / “R”) and the probability of leaf nodes in the left tree connecting to the leaf nodes in the right tree (“p”), as node attributes.

A. Generate 100 graphs from the dendrogram. Remember that you can infer the probabilities that each pair of leaf nodes will be connected from the dendrogram. To build a graph from the probabilities, you can generate a random number between 0 and 1 for each pair of nodes, and add an edge between them if the random number is smaller than the corresponding probability. Avoid self-loops in the generated networks.
B. Calculate the following properties for each of these 100 graphs: network diameter, CPL, average clustering coefficient, transitivity, and assortativity. Report the distribution of each property among 100 graphs using appropriate plots.

1.5 Best Fit (15 points)

Using a one-sample t-test, we can examine if various features of the empirical network are well-represented in the networks generated by the models used in Part 1.2, 1.3, and 1.4.

A. Report the average value and standard deviation for the diameter, CPL, average clustering coefficient, transitivity, and assortativity values found from the 100 samples of each model.

B. Compare the diameter, CPL, average clustering coefficient, transitivity, and assortativity for the empirical network with the sampled values from each of the graph models using a one-sample t-test. Report the p-values.

C. Which model do you think best approximates the empirical network? Explain your answer.

Part 2. Estimate the number of nodes and edges in Slashdot dataset (35 points)

In this part, we will be working with the Slashdot social network. The file soc-Slashdot0902.txt stores the list of edges in this network. The network has 82,168 nodes and 948,464 edges. Exclude the self-loops in the network and conduct the following analysis:

A. **(10 points)** Use the capture-recapture estimation method to compute the number of nodes in the network by randomly choosing 2,000 nodes each time. Repeat the experiment 1000 times and plot the histogram of the estimated number of nodes in the network (the y-axis is the frequency of occurrence and the x-axis is the estimated number of nodes).

B. **(10 points)** Repeat the same analysis using samples of 500, 1000, 2000, 5000, and 10000 nodes (but you can skip the histogram this time). Plot the estimated number of nodes against the number of sampled nodes. Also, plot the mean±std values over 1000 iterations. Compare it against the actual number of nodes in the network and comment on the trend of estimated values with the sample size.

C. **(15 points)** Estimate the number of edges using the induced sub-graph sampling and Horvitz-Thompson estimator by sampling 5,000 nodes in the network. Repeat this 100 times and plot the histogram of the estimate along with the ground truth (the y-axis is the frequency of occurrence and the x-axis is the estimated number of edges).