Social Network Analysis
Spring 2019
MS(DS)
Assignment 1
Submission Date: Feb 18, 2019

Problem 1

Load the Wikipedia voting network (file attached) in Snap. Note that Wikipedia is a directed network. Formally, we consider the Wikipedia network as a directed graph G = (V,E), with node set V and edge set E where (edges are ordered pairs of nodes). An edge (a, b) means that user a voted on user b.

Compute and print the following statistics

Number of nodes
Number of edges
Number of self-edges
Number of directed edges (i.e. count edge (a,b) if $a \neq b$)
Number of undirected edges (i.e count edges such that (a, b) and (b, a) both are included in E)
Number of nodes with zero indegree
Number of nodes with zero outdegree
Node id with maximum degree
Number of strongly connected component
Size of largest connected component
Number of weakly connected components
Size of largest weakly connected component

Also plot the outdegree distribution. Each data point is a pair (x, y) where x is a positive integer and y is the number of nodes in the network with outdegree equal to x

Problem 2

One of the goals of network analysis is to find mathematical models that characterize real-world networks and that can then be used to generate new networks with similar properties. In this problem, we will explore two famous models Erdos-Renyi and Small World and compare them to real-world data from an academic collaboration network. Note that in this problem all networks are undirected.

Erdos-Renyi Random graph ($G_{n,m}$ random network): Generate a random instance of this model by using n nodes and picking m edges at random. Write code to construct instances of this model, i.e., do not call a SNAP function.

Small-World Random Network: Generate an instance from this model as follows: begin with n nodes arranged as a ring, i.e., imagine the nodes form a circle and each node is connected to its two direct neighbors (e.g., node 0 is connected to nodes 1 and n-1), giving us n edges. Next, connect each node to the neighbors of its neighbors (e.g., node 0 is also connected to nodes 2 and n-2). This gives us another n edges. Finally, randomly select x pairs of nodes not yet connected and add an edge between them. In total, this will make total edges = n*2+x edges. Write code to construct instances of this model, i.e., do not call a SNAP function.

Real-World Collaboration Network: Nodes in this network represent authors of research papers on the arXiv in high energy physics theory. There is an edge between two authors if they have co-authored at least one paper together.

Generate a random graph from both the Erdos-Renyi and Small-World models. Note that number of nodes and number of edges in both these random graph must be equal to the total number of nodes and edges (other than self loops) in the real network (collaboration network). Also read in the collaboration network (file attached). Delete all of the loops (self-edges) in the collaboration network. Compute the following:

1. Plot the degree distribution of all three networks in the same plot on a log-log scale. In other words, generate a plot with the horizontal axis representing node degrees and the vertical axis representing the proportion of nodes with a given degree (by log-log scale" we mean that both the horizontal and vertical axis must be in logarithmic scale).

2. Compute and print the average clustering coefficient of the three networks. For this question, write your own implementation to compute the clustering coefficient, instead of using a built-in SNAP function.

3. Compute and print the size of largest connected component in all three networks. Also print the fraction of nodes and edges in the largest connected component. For this you can use the SNAP functions.

4. Compute and print the diameter of all three networks.

Important Note:
Submit your code files at google classroom as one single file. If you have more than one files compress them and submit them as single file. The name of file must your complete roll number. Your code must be properly commented.
This is an individual assignment and the code submitted must be your own contribution. Any sort of plagiarism will be dealt seriously and may lead to severe consequences including negative marking.