Kathmandu University

Department of Computer Science and Engineering

Dhulikhel, Kavre



COMP 314 (Algorithms and Complexity)

Lab 5 Report

Submitted By:

Mani Dumaru

Roll no.: 15

CE 2019 (3rd year. 2nd semester)

Submitted to:

Dr. Rajani Chulyadyo

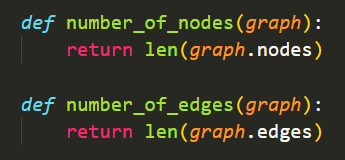
Department of Computer Science and Engineering

Submission Date: 4th June, 2023

Python Graph Libraries:

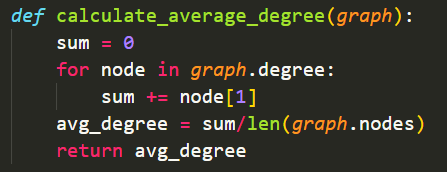
I have used NetworkX python library for this lab work. 6 different networks were downloaded from Network Repository and these networks were imported using the library.

1. Number of Nodes and Edges:



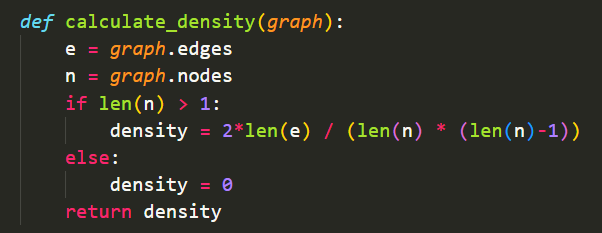
These are the built in attributes of the graph object from the library that returns the number of nodes and edges it contains.

1. Average degree:



The for loop in above code calculates the total degree of the network going through the degree of each node. Then the average degree is returned by this function.

1. Density:

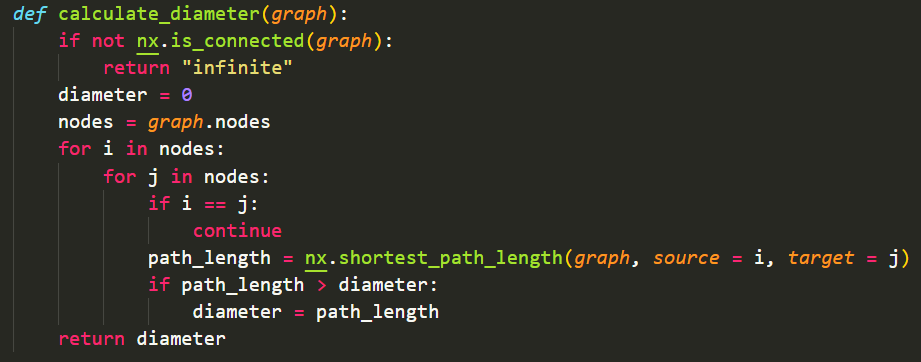


If the number of nodes in the network is greater than 1, its density is calculated by:

density = (2\*number of edges) / [number of vertices \* (number of vertices - 1)]

else the density is zero.

1. Diameter:



Diameter of a network is the shortest path between the two most distanced nodes. For a graph that is not connected (i.e.: there is not path to travel between any two vertices), the diameter is infinite.

1. Going through each node, find the shortest path to all the other nodes.
2. Choose the largest path among these paths as the diameter.
3. Loop through each and every vertex and keep on replacing largest paths of larger size than the current diameter.
4. After the completion of the loop, largest of such path is the final diameter of the network.

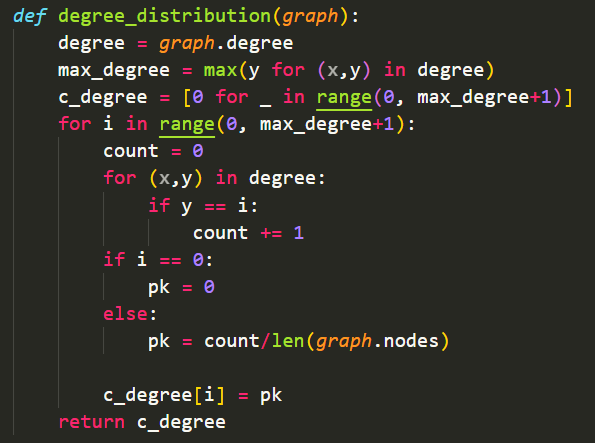
For this lab, calculation of diameter for networks with around 5K vertices took very long time. So, I’ve commented the diameter calculation portion.

1. Clustering Coefficient:



There is a built-in function in the library that returns all the neighbours of a particular node. Now to find the connections between the neighbours of a particular node, each neighbour is looped through each other and if there exists a path between them, the connection is incremented. In this way the connection and total number of neighbours is calculated and hence using these values, clustering coefficient can be calculated.

1. Degree distribution:

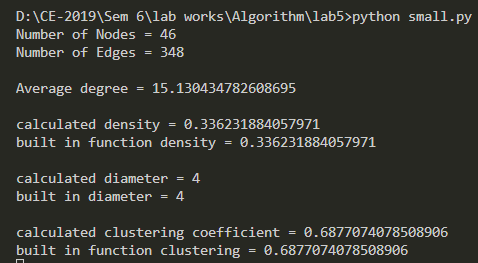


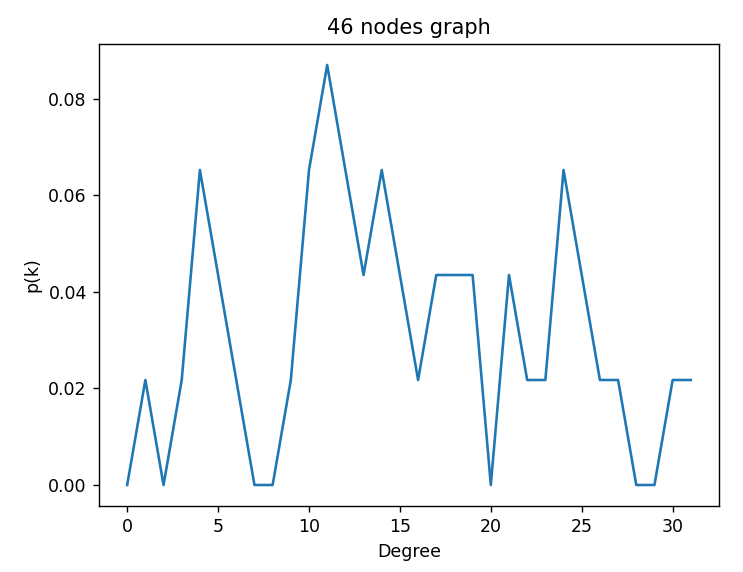
From all the degrees of the network, the maximum degree is found. For each degree from 0 to maximum degree, total number of times that particular degree is repeated is calculated and is divided by total number of nodes in the graph. This gives the degree distribution P(k). These values are then plotted against the degrees itself to form degree distribution graph.

Outputs:

For Task 1:

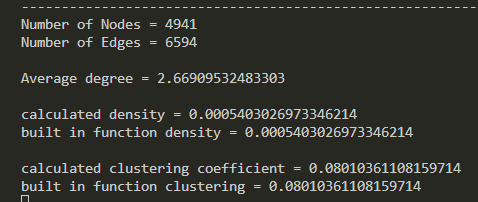
A small graph with only 46 nodes:

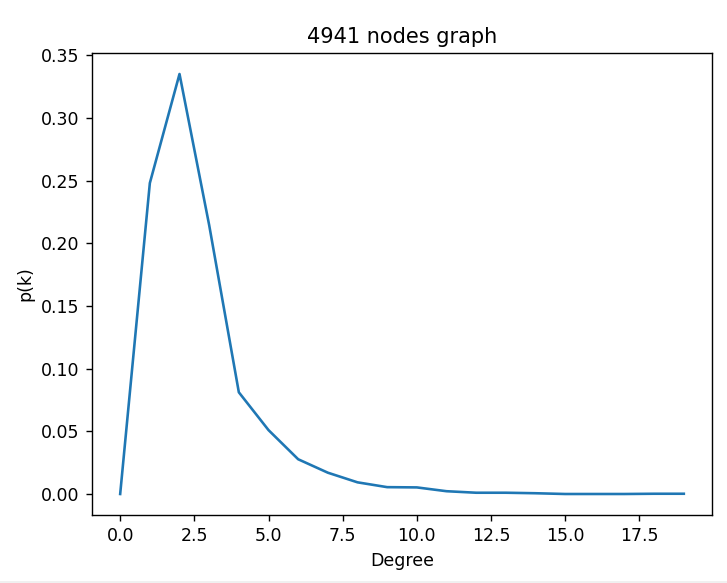




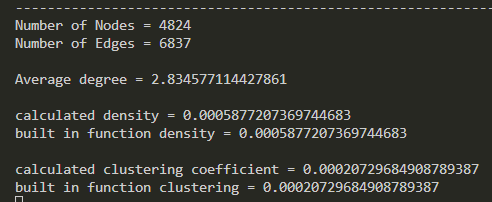
For Task 2:

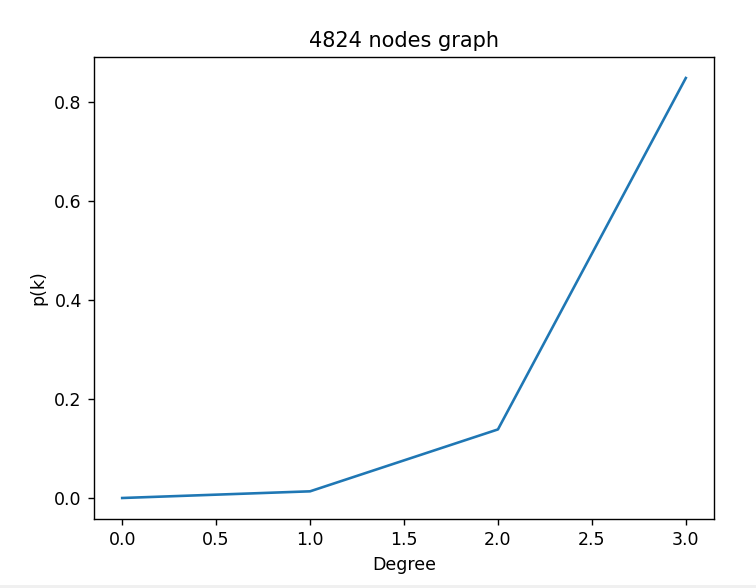
1. inf-power.mtx



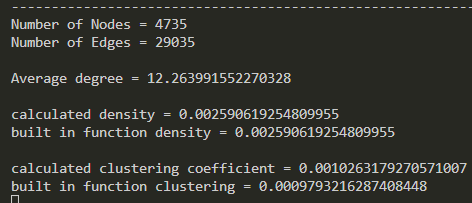


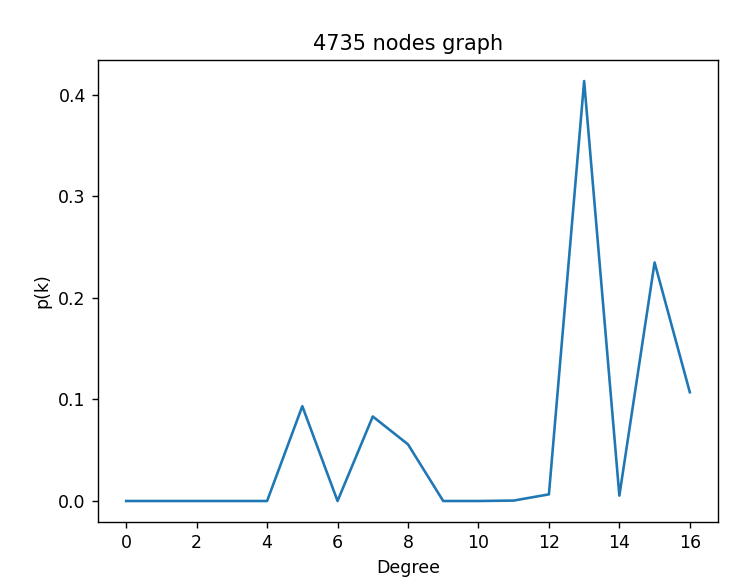
1. uk.mtx



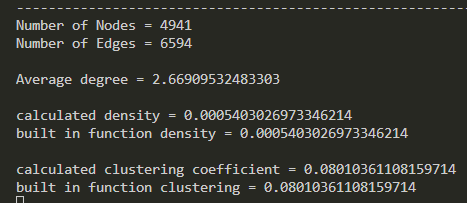


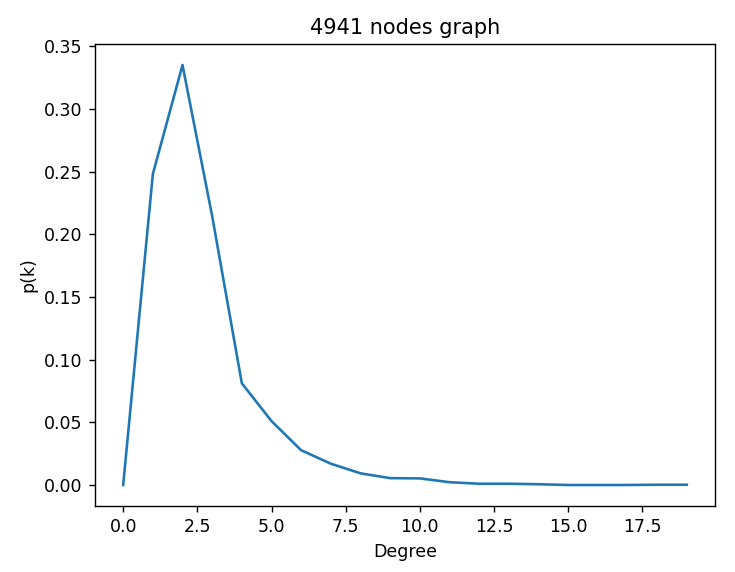
1. n4c5-b7.mtx



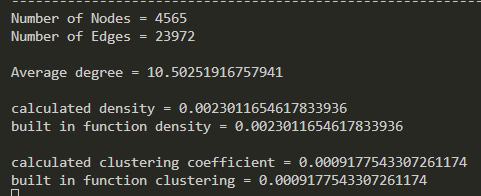


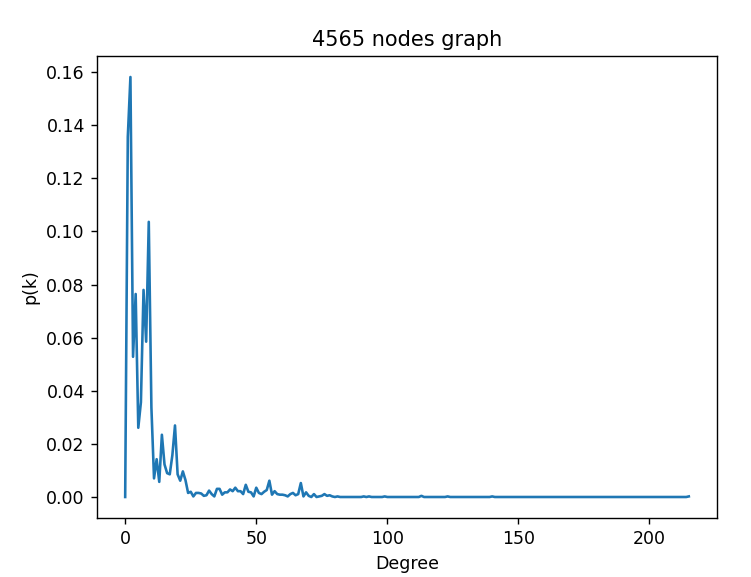
1. USpowerGrid.mtx





1. model3.mtx





Answer the following Questions:

1. From the network properties, what can you say about the networks you selected?

From the networks that I selected, the diameters of networks with many numbers of nodes could not be found. It took a very long amount of time to calculate the diameter of such networks since during the calculation, each node must be visited and its shortest path to all other nodes must be calculated. For this to be done for more than 5 thousand nodes of the graph, it took way too long. Also, the diameter of disconnected graph is infinite since there is no path between two of the nodes. Hence their distance being infinite.

1. Did you find any patterns in the degree of distribution of the networks? In any case, can you come to ay conclusion about the graph from their degree distribution?

For three of the outcomes, lower degree has higher frequency while the increase in degree brings their frequency lower. In the other two cases, higher degree is frequently repeated while as the degree gets lower its frequency also gets lower. Lower degree appears less frequently in such networks. It the frequency of higher degrees are evenly distributed; we can conclude that the graph is dense in nature. While the frequency of lower degree is hight means that the graph is sparse.