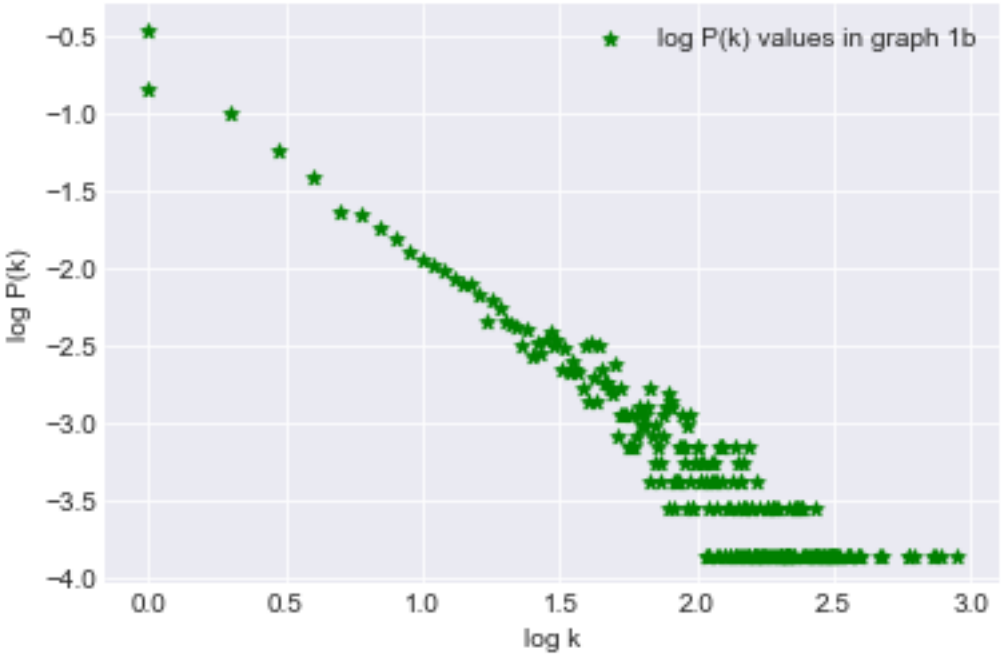
MLN ASSIGNMENT 1 REPORT

Problem 1a:

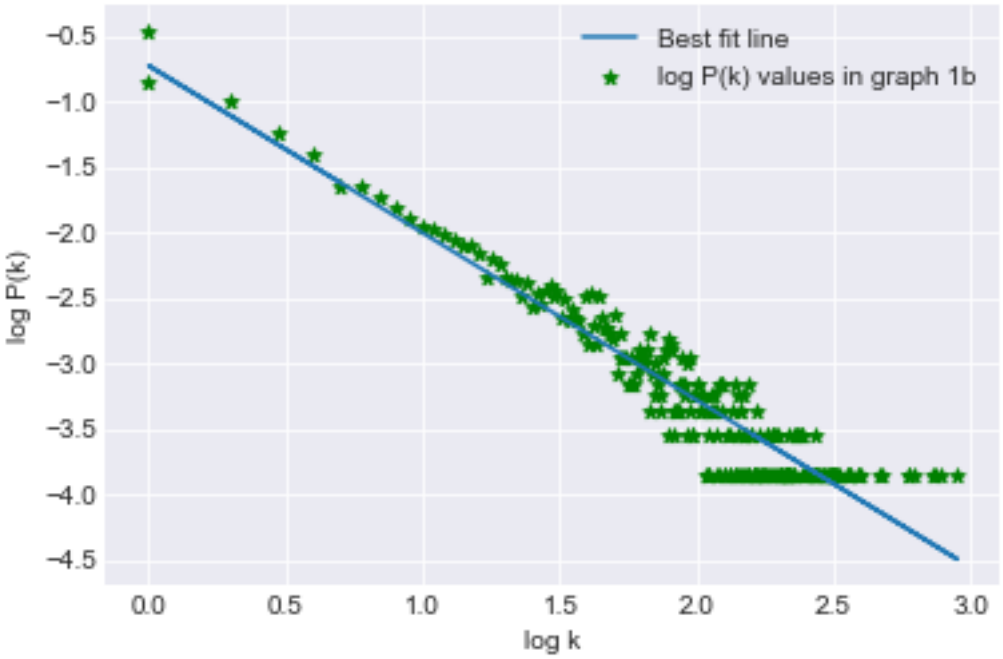
1. Total no. of nodes = 7115
2. Number of nodes having a self-loop = 0
3. Number of edges in the graph that are not self-loops = 103689
4. Number of unique pairs of vertices having an edge between them = 100762
5. The no. of unique bidirectional edges are = 2927
6. The no. of nodes with zero indegree is = 4734
7. The no. of nodes with zero outdegree is = 1005
8. Number of connected components in the network = 5840
9. The no. of nodes with indegree greater than 10 is = 1906
10. The no. of nodes with outdegree greater than 10 is = 1612

Problem 1b:

1. Here the P(k) value denotes the probability of a node having out-degree k.
   P(k) = Nk / N
   Where Nk is the no. of nodes with degree k and N is the total no. of nodes



2. Now linear regression is used to calculate the slope and y-intercept values of the line that best fits the out-degree distribution on a log-log scale.



Problem 2a:

1. The no. of weakly connected components in the graph is 10143
   The no. of strongly connected components in the graph is 142474

2. The no. of edges in the largest weakly connected component is 322486
   The no. of nodes in the largest weakly connected component is 131188

3. The page ranks are calculated for all the nodes and a graph is plotted with the different page ranks on the X-axis and the no. of nodes on Y-axis.

   Also, the node with the highest page rank is 992484
   The highest page rank value is 0.013980540412209575

   The plot for the same is shown on the next page:

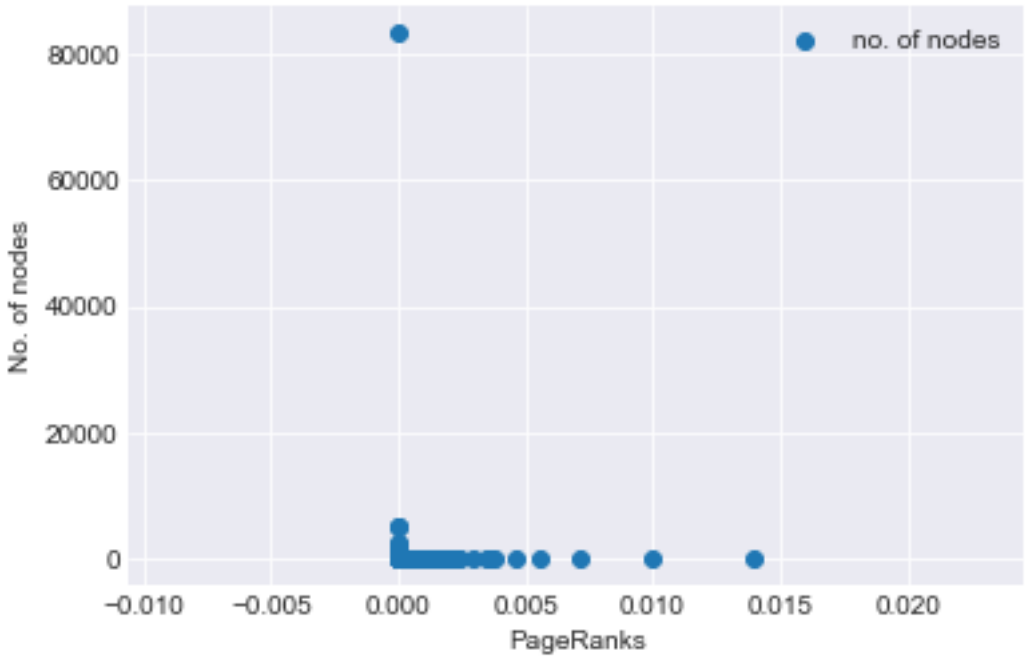


Fig: Plot showing page rank distribution of nodes in the network

4. The HITS algorithm is run on this network using the sanp.py and the top 5 hubs and authorities are found which are:

   Top 5 hubs are:
   Hub 1 = 892029, Hub 2 = 1194415, Hub 3 = 359862, Hub 4 = 648138,
   Hub 5 = 470184

   Top 5 authorities are:
   Authority 1 is 22656, Authority 2 = 157882, Authority 3 = 571407,
   Authority 4 = 57695, Authority 5 is 139985

Problem 2b: Here the HITS algorithm is implemented

1. The top 5 hubs and authorities are found using the following values:
   Tolerance = 0.00004539992
   Max iterations = 100
   The values obtained are:

   The top 5 hubs are:
   Hub 1 = 892029, Hub 2 = 1194415, Hub 3 = 359862, Hub 4 = 648138,
   Hub 5 = 470184

The top 5 authorities are:
Authority 1 = 22656, Authority 2 = 157882, Authority 3 = 571407,
Authority 4 = 57695, Authority 5 = 139985

As can be seen the top 5 hub and authority values obtained here are exactly same in rankings as those obtained in 4th part of problem 2a.
Also, on comparing the respective hub and authority scores obtained here and those obtained in problem 2a 4th part it can be seen that the values are same up to 3-4 decimal values.

2. On setting max_iters = 500, the L2 norm values are obtained for hubs and authority separately.
The L2 norm value is obtained by finding the square root of the sum of the squares of the difference of the scores obtained by my implementation and those obtained by library. The values are:

L2 norm of the difference between estimated and true hub scores = 0.25522106810235295

L2 norm of the difference between estimated and true authority scores = 0.2466908447353253

Problem 3a:

A random graph is generated using the Erdos-Renyi model. Here the edges are progressively added to the graph each with a probability value.

No. of nodes in the graph = 5242

No. of edges in the graph = 14484

Problem 3b:

- No. of nodes in the graph = 5242
- No. of edges after adding immediate neighbours= 5242
- No. of edges after adding 2 hop neighbours= 10484
- No. of all possible edges in graph = 13736661
- No. of remaining edges = 13726180
- Finally, no. of edges in the graph after adding 4000 random edges also= 14484

Problem 3c:

No. of nodes in the graph = 5242

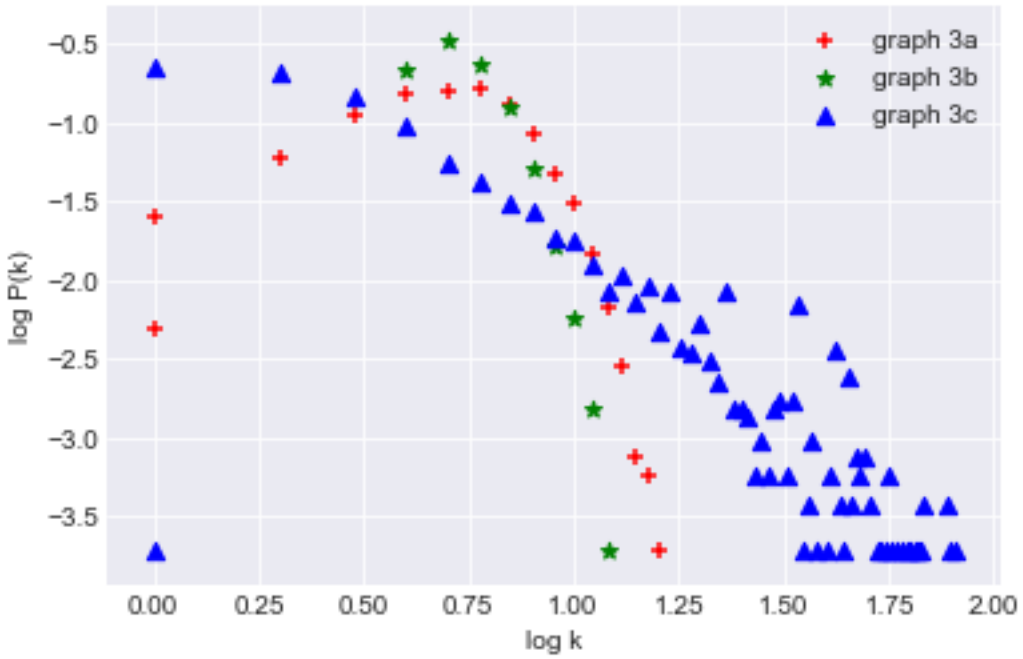
No. of edges in graph excluding repeats and self-loops = 14484

Problem 3d:

The degree distributions for the 3 graphs are shown below.

Here P(k) = Nk / N where Nk is the no. of nodes with degree k and N is the total no. of nodes.

log P(k) values are taken along Y-axis and log k values taken on X-axis.



The analysis of the degree distributions is:

- Graph 3a is a random graph and here the edges are being added to the graph with a probability value p. It can be seen here that there is less no. of nodes with a lower degree and more no. of nodes with a higher degree.
- Graph 3b is not a random graph and in this around 10000 edges have been added first which satisfy a particular condition and then only 4000 edges are added randomly. So, it is like a real-world network and has lesser amount of randomness as compared to graph 3a.
- Graph 3c is a real-world citation network and thus it has edges between nodes only if one of the authors has cited the other. It is not a random network.

Problem 3e:

The average clustering co-efficient values for the 3 graphs are:

Average clustering coefficient of graph a= 0.0031113730188687266

Average clustering coefficient of graph b= 0.2838417972717822

Average clustering coefficient of graph c = 0.529635811052136

Graph-c has the highest clustering co-efficient value as it is a proper real-world network. And real-world networks have a high avg. clustering co-efficient as compared to random networks.

Graph-b has the next highest value as it is like a real-world network up to some extent.

Graph-a has completely been generated randomly and thus has the lowest avg. clustering coefficient.

A high value of avg. clustering coefficient indicates that the neighbours of the nodes in the graph are densely connected.