Assignment - 02

“Web and Social Computing (IT752)”

Submitted by

Tarushi Jat (202IT029)

Submitted to

Dr. Sowmya Kamath

MASTER OF TECHNOLOGY
in
INFORMATION TECHNOLOGY

Department Of Information Technology

National Institute Of Technology, Surathkal

Task Description

Consider the real-world information networks of different scales that you used for the Assignment-1, Apply the formal models listed below on your datasets and verify the inherent properties exhibited by the graph data chosen by you.

1. Erdos-Renyi Model
2. Watts-Strogatz Model
3. Barabasi-Albert Model

Dataset Used for the Task

Dataset 1 : Social Networks Dataset : soc-sign-bitcoin-otc

Nodes are the people who trade using Bitcoin on a platform called Bitcoin OTC. Since Bitcoin users are anonymous, there is a need to maintain a record of users' reputation to prevent transactions with fraudulent and risky users. Members of Bitcoin OTC rate other members in a scale of -10 (total distrust) to +10 (total trust).
Therefore, there will be an **edge** between any 2 nodes if one user gives rating to the other user.

Number of Nodes = 5881
Number of Edges = 21492

Dataset 2 : Collaboration Networks Dataset: ca-GrQc

This dataset covers scientific collaborations between authors papers submitted to the General Relativity and Quantum Cosmology category. In this collaboration Network, **nodes** are the authors in the network and If an author *i* co-authored a paper with author *j*, the graph contains an undirected **edge** from *i* to *j*.

Number of Nodes = 5242
Number of Edges = 14496

Dataset 3 : Communication Network : email-Eu-core

This dataset contains the information of incoming and outgoing email between members of a European research institution. There is an **edge** (u, v) in the network if person u sent person v at least one email.

Number of Nodes = 1005 Number of Edges = 16706

Formal Models for Information Network Analysis

Model 1. Erdos-Renyi Model

It is one of the simplest models to explain random complex networks. Erdos Renyi's model is used to create random networks or graphs on social networking. In this model, each edge has a fixed probability of being present and being absent independent of the edges in a network.
Working: We start with all isolated nodes that are having no edges, then we add edges between pairs of nodes one at a time randomly. Now, for adding edge randomly, we have 2 choices:

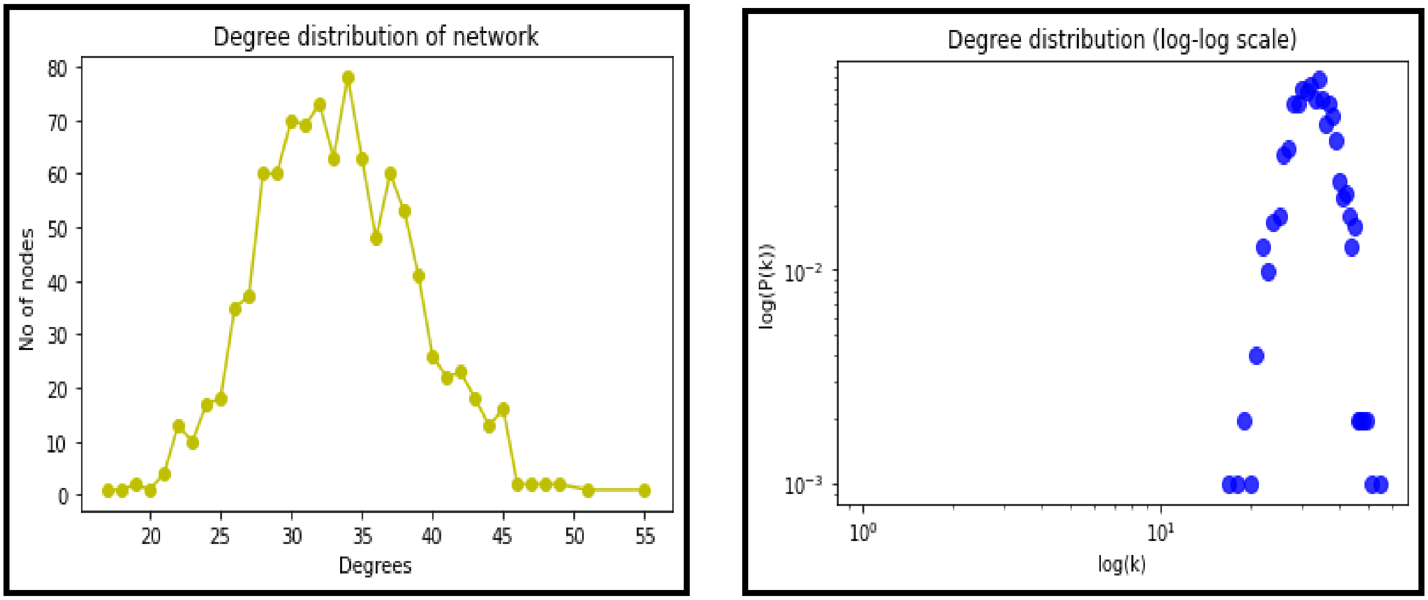
- Randomize Edge Presence
- Randomized Node Pairs

I have implemented the model using **randomized node pairs**.

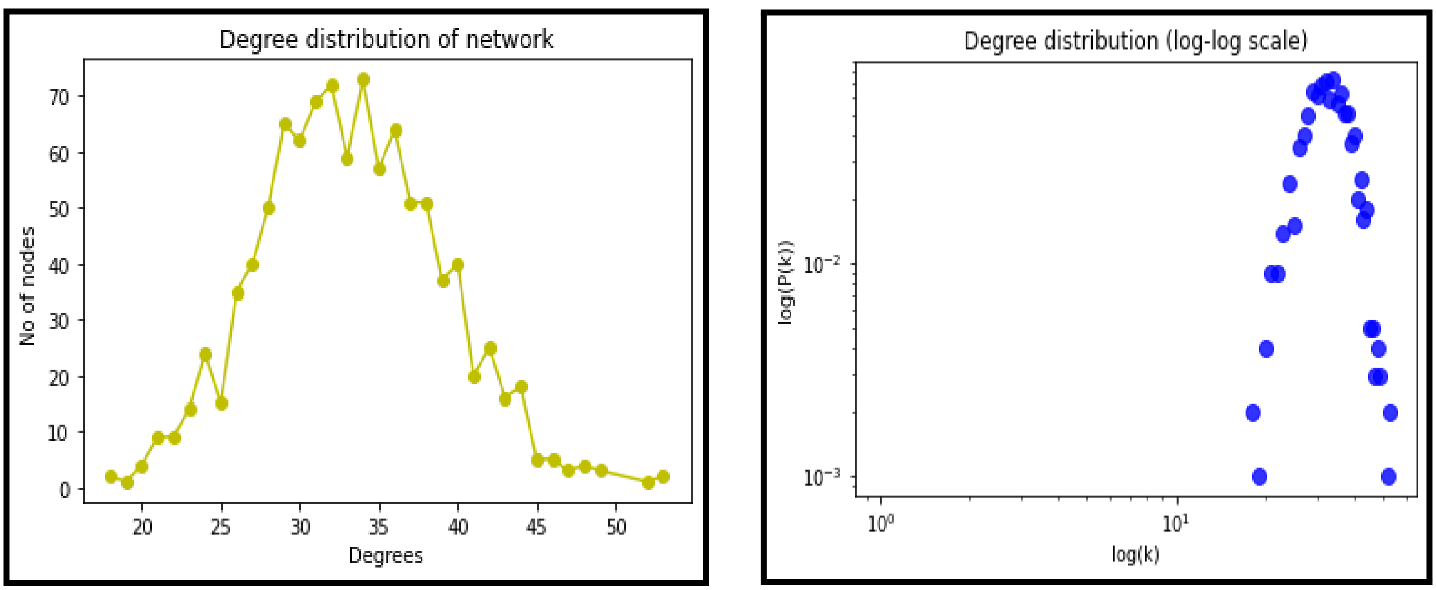
Randomized node pairs pick a pair of nodes at random among the n nodes and add an edge between them if not present already and repeat until exactly m edges have been added. This model is represented by G(n, m).

a) Degree Distribution of Erdos-Renyi Model: The degree of a node in a network is the number of connections it has to other nodes and the degree distribution is the probability distribution of these degrees over the whole network.

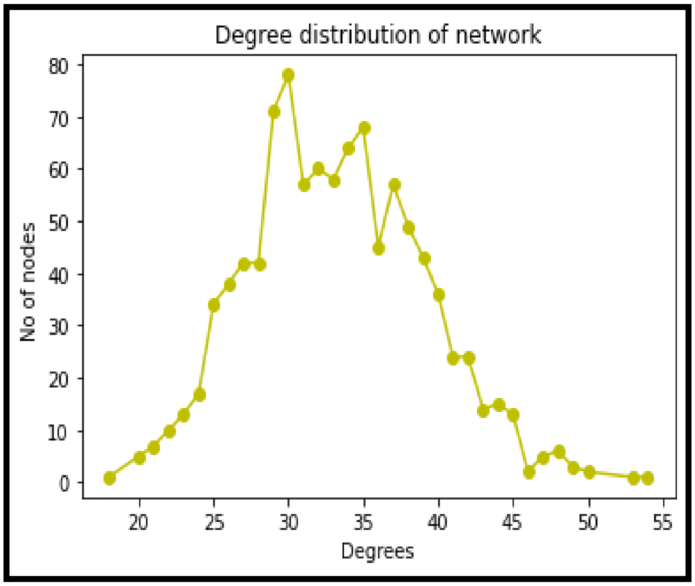
Dataset 1 : soc-sign-bitcoin-otc

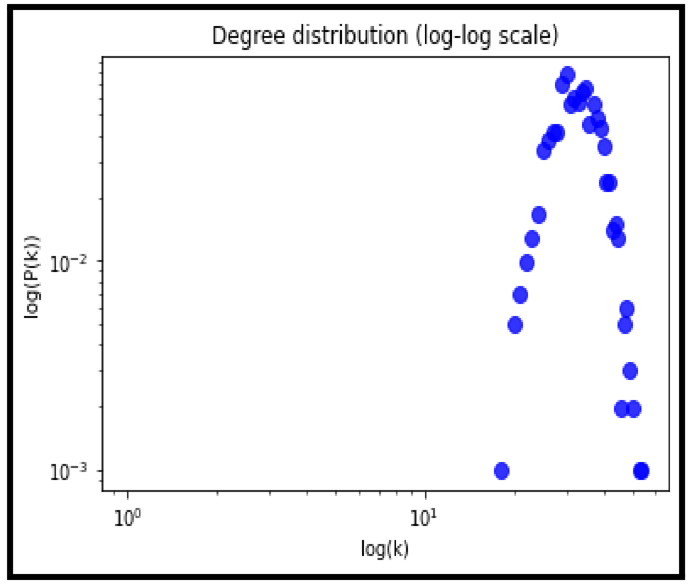


Dataset 2 : ca-GrQc



Dataset 3 : email-Eu-core





*Erdos-Renyi is a **poor predictor of degree distribution** compared to real world networks. It results in Poisson degree distributions that have exponential decay and most real networks exhibit power-law degree distributions that decay much slower than exponential.*

b) Erdos-Renyi Graph or Binomial Graph

Erdos Renyi gives a random graph, which is also known as an Erdos-Rényi graph or a binomial graph. The model chooses each of the possible edges with probability p. To generate the Erdos-Renyi graph, I have used the following function:

erdos_renyi_graph(n,p)

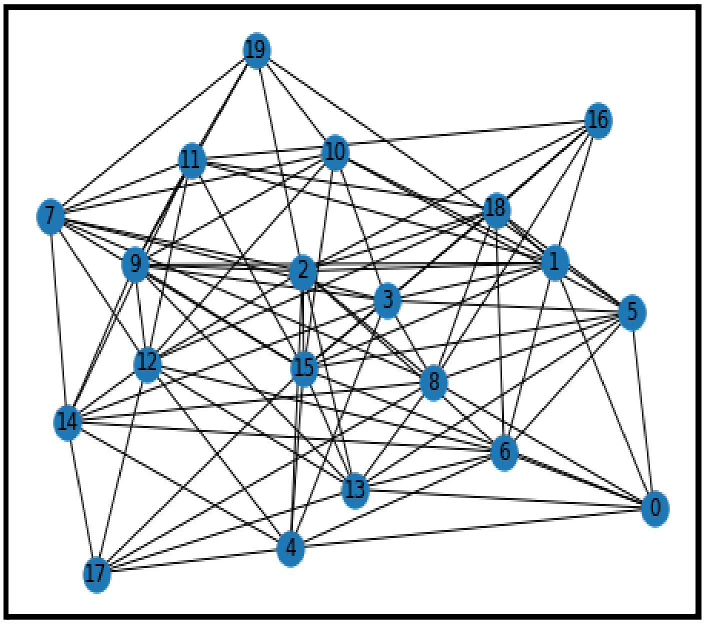
Parameters: *n (int) – The number of nodes.*
p (float) – Probability for edge creation.

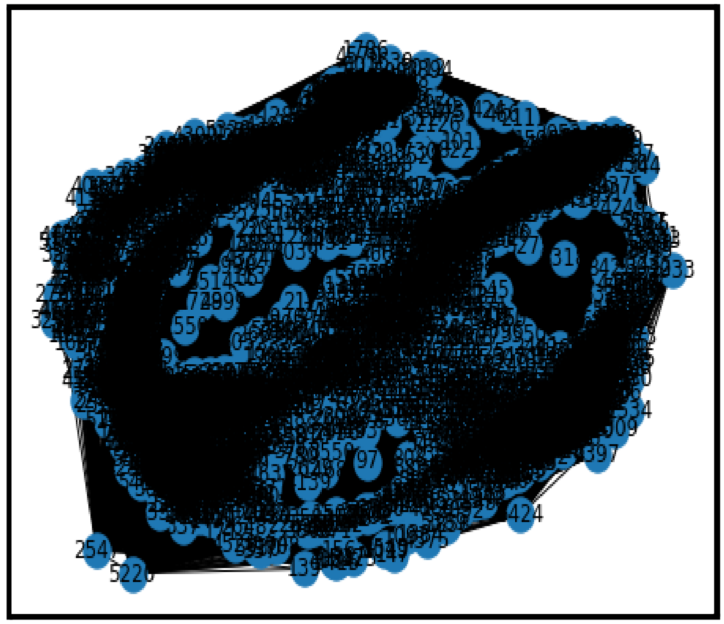
Dataset 1 : soc-sign-bitcoin-otc

Following are the 2 binomial graphs that are generated using the above function. The Erdos-Renyi Graph which is left is generated with 20 nodes with value of p = 0.5. And the

Erdos-Renyi Graph which is in right is generated with all the nodes present in the dataset (which is 5881) with p = 0.5.

The parameter p in this model is the weighting function, as p increases from 0 to 1, the model becomes more and more likely to include graphs with more edges and less and less likely to include graphs with fewer edges. *In this particular case p = 0.5 corresponds to the case where all 2^($^nC_2$) graphs on n vertices are chosen with equal probability.*

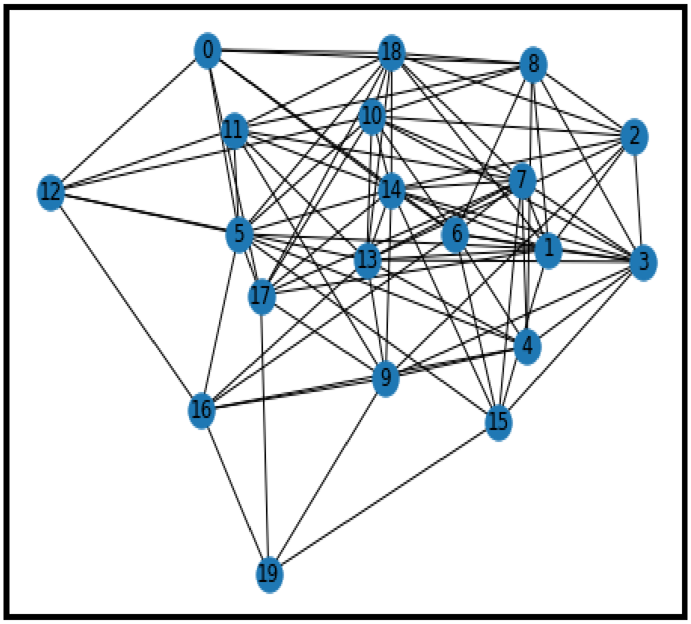


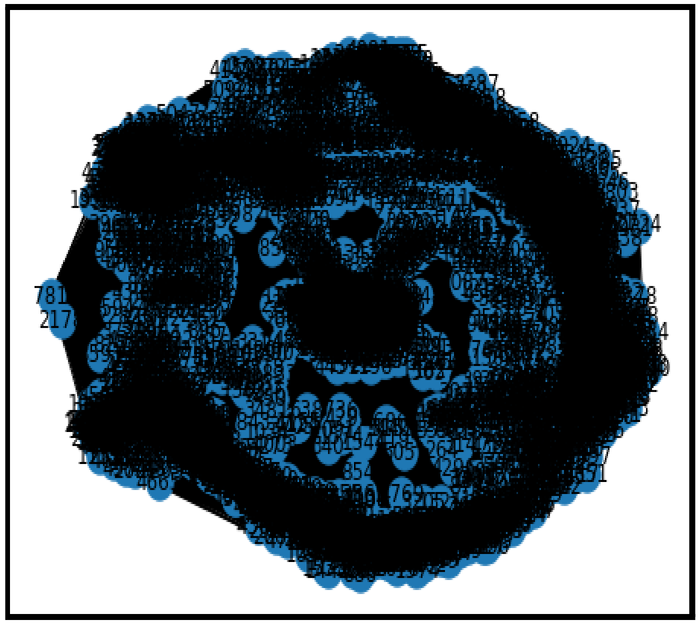


Average Shortest Path	2.2885413569602187
Average Clustering Coefficient	0.032516380271011176

Dataset 2 : ca-GrQc

Following are the 2 binomial graphs for the dataset from the collaboration network that are generated using the above mentioned function. The Erdos-Renyi Graph which is left is generated with 20 nodes with value of p = 0.5. And the Erdos-Renyi Graph which is in right is generated with all the nodes present in the dataset (which is 5242) with p = 0.5.

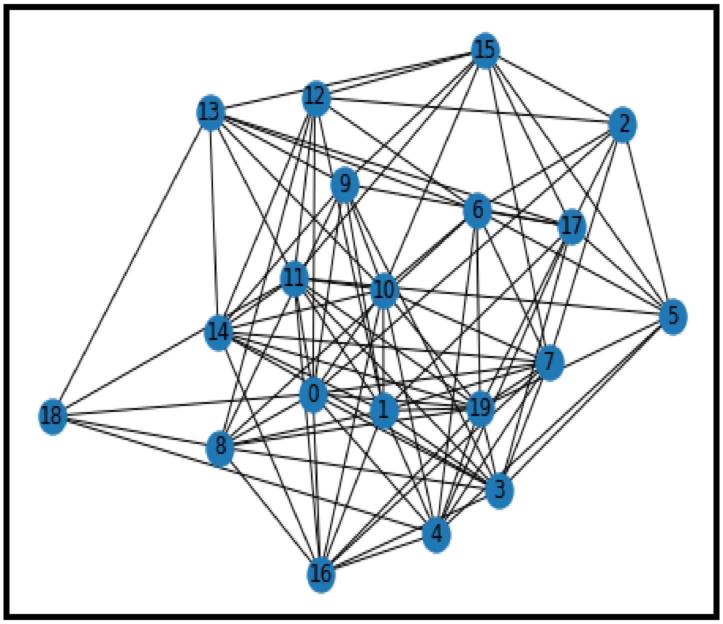


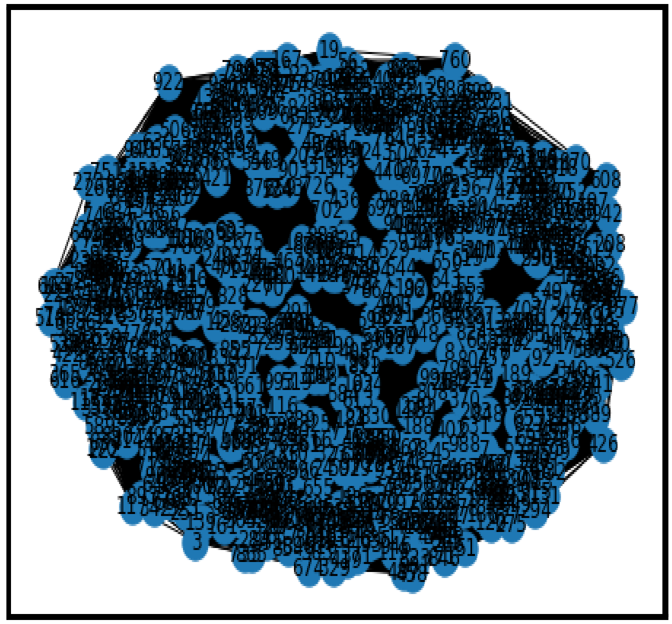


Average Shortest Path	2.2888902103030664
Average Clustering Coefficient	0.03273911525333677

Dataset 3 : email-Eu-core

Following are the 2 binomial graphs that are generated from the social networks dataset using the above mentioned function. The Erdos-Renyi Graph which is left is generated with 20 nodes with value of p = 0.5. And the Erdos-Renyi Graph which is in right is generated with all the nodes present in the dataset (which is 1005) with p = 0.5.



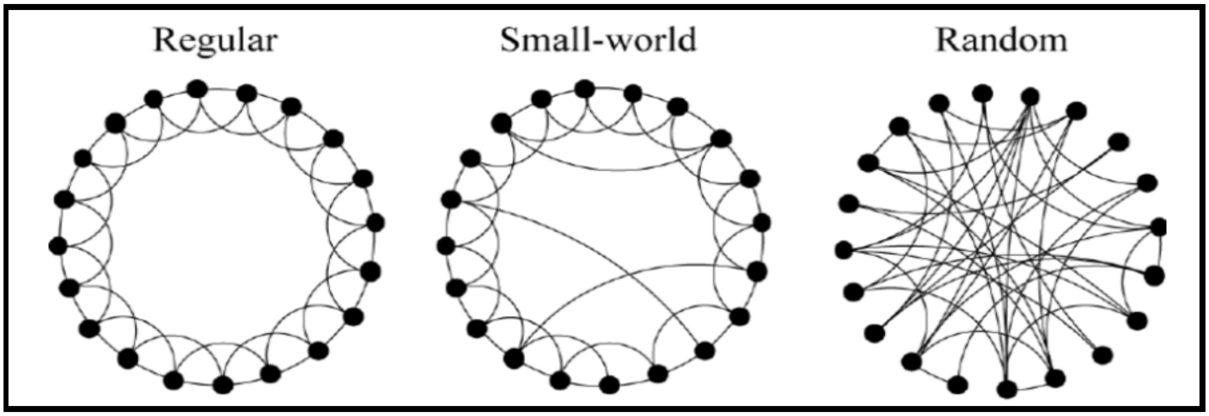


Average Shortest Path	2.2886008205982042
Average Clustering Coefficient	0.033637640082541595

Model 2. Watts-Strogatz Model

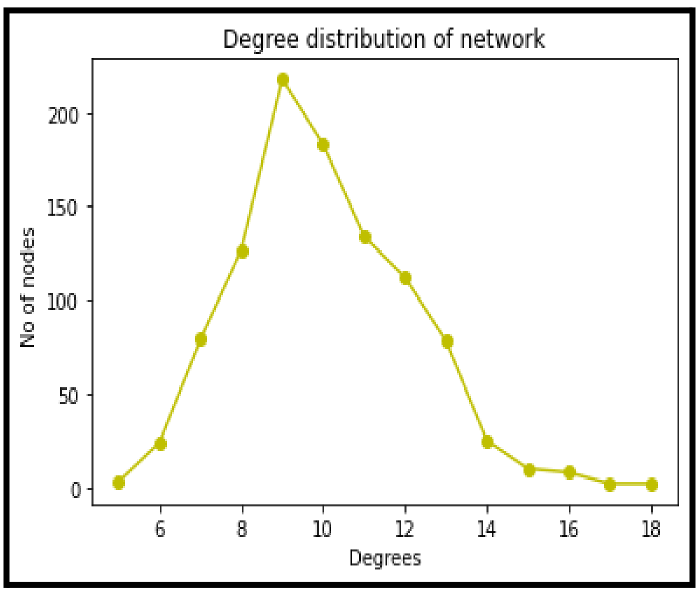
The Watts–Strogatz model is a random graph generation model that produces graphs with **small-world properties**, including **short average path lengths** and **high clustering**. The Watts-Strogatz model is the main mechanism to construct the small-world networks.

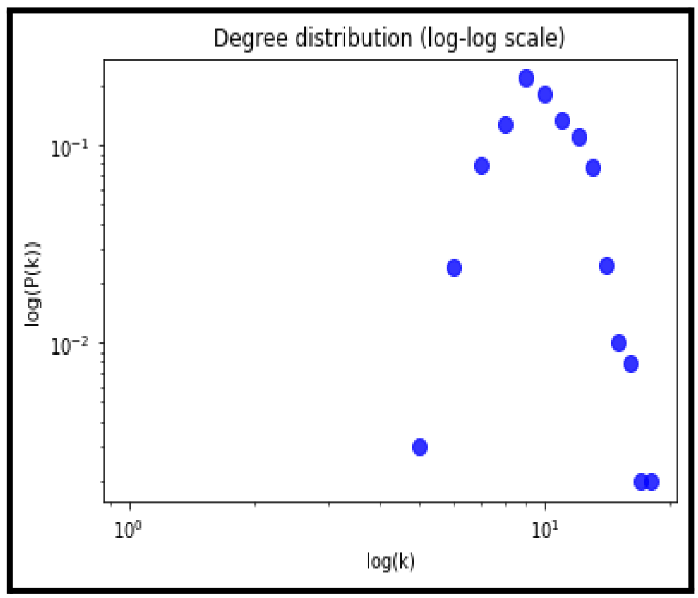
Watts & Strogatz concluded that Small World Networks generally have low average path length but high clustering coefficient. This model is a generative model which starts with a regular graph and rewires its edges randomly to produce graphs with small-world properties.



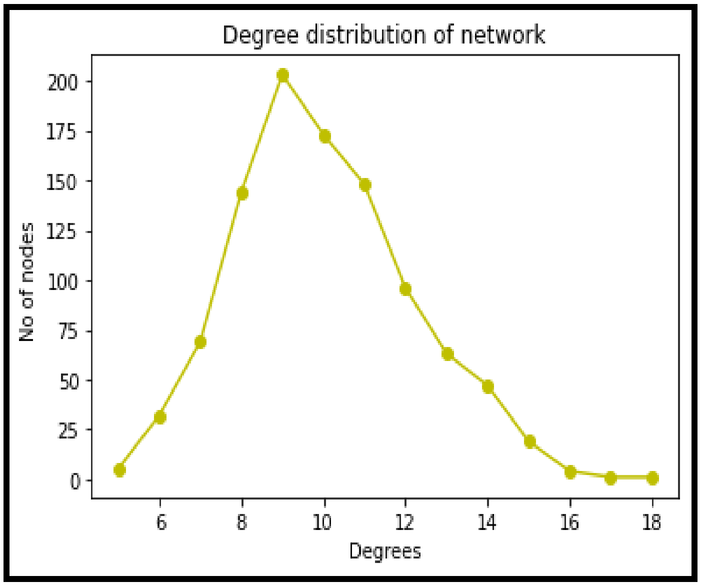
a) Degree Distribution of Watts-Strogatz Model: The degree of a node in a network is the number of connections it has to other nodes and the degree distribution is the probability distribution of these degrees over the whole network.

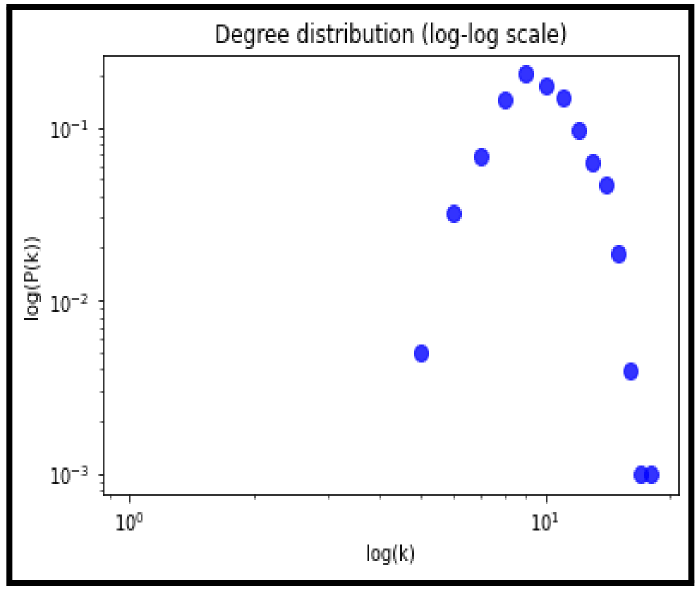
Dataset 1 : soc-sign-bitcoin-otc



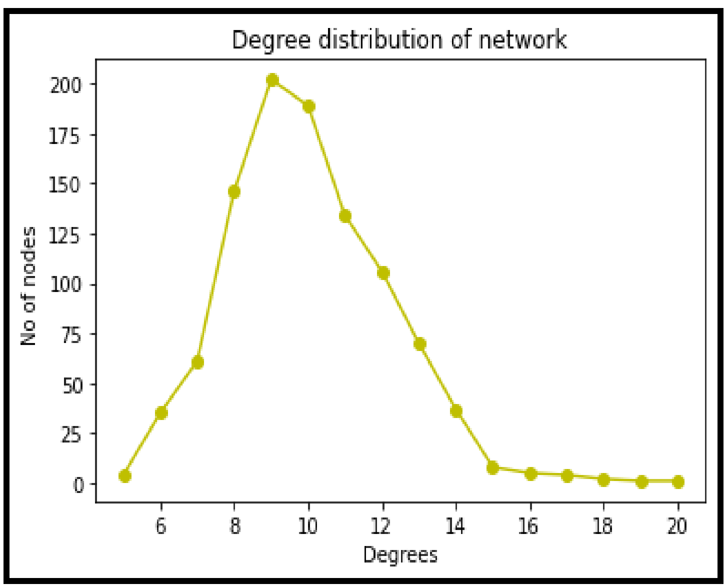


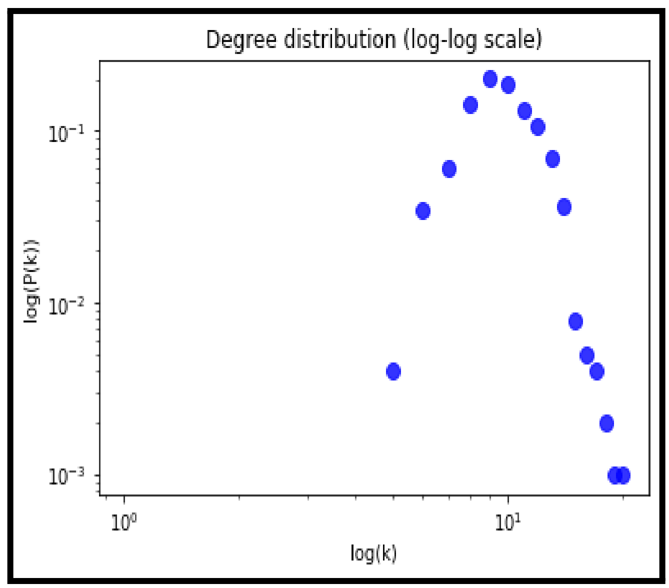
Dataset 2 : ca-GrQc





Dataset 3 : email-Eu-core





b) Watts-Strogatz Graph

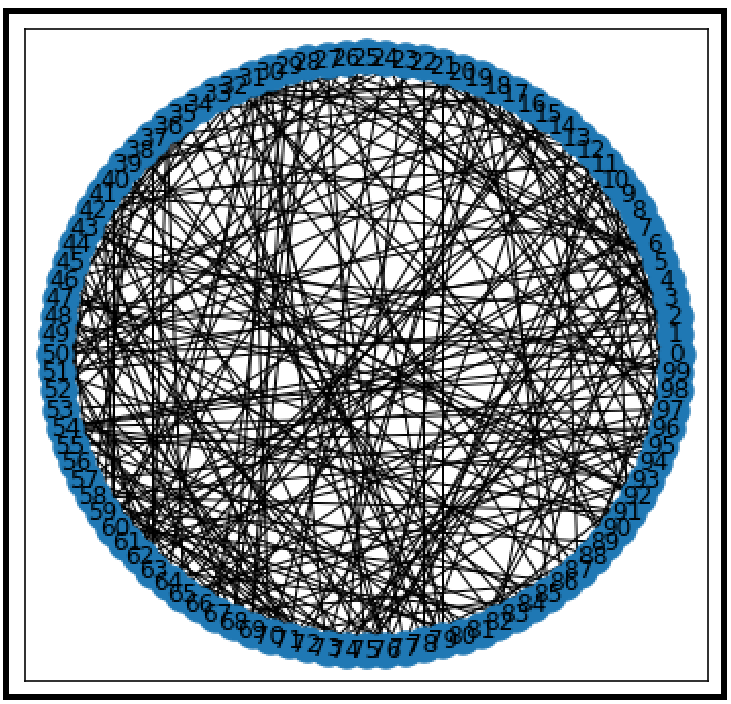
Watts-Strogatz gives a small world graph. **Small World** Phenomenon that we are all connected via a small number of edges. Small World Networks generally have low average path length but high clustering coefficient. To generate the Watts-Strogatz graph, I have used the following function:

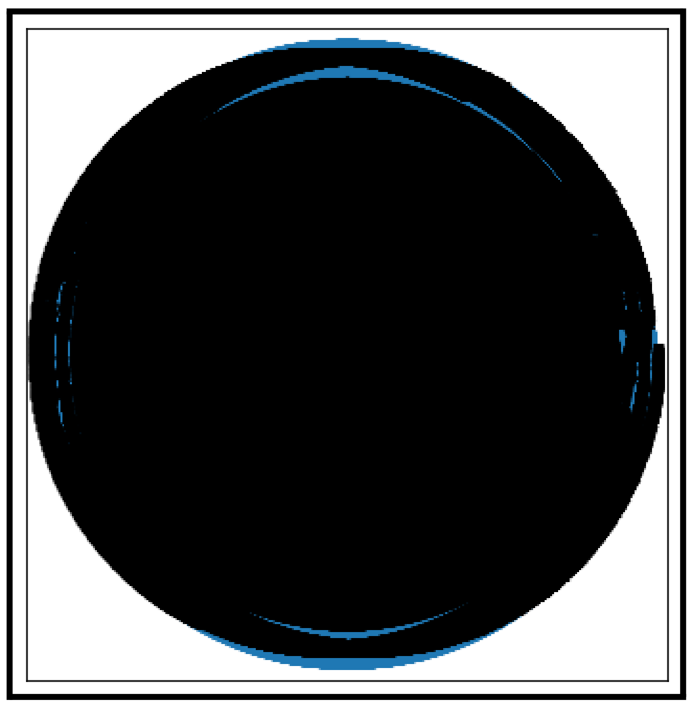
watts_strogatz_graph(n, k, p)

Parameters: *n(int) : The number of nodes*
k(int) : Each node is connected to k nearest neighbors in ring topology
p(float) : probability of rewiring each edge.

Dataset 1 : soc-sign-bitcoin-otc

Following are the 2 Watts-Strogatz graphs that are generated using the above function. The Watts-Strogatz Graph which is left is generated with 100 nodes with value of p = 0.5 and value of k = 10. And the Watts-Strogatz Graph which is in right is generated with all the nodes present in the dataset (which is 5881) with p = 0.5 and value of k = 10.

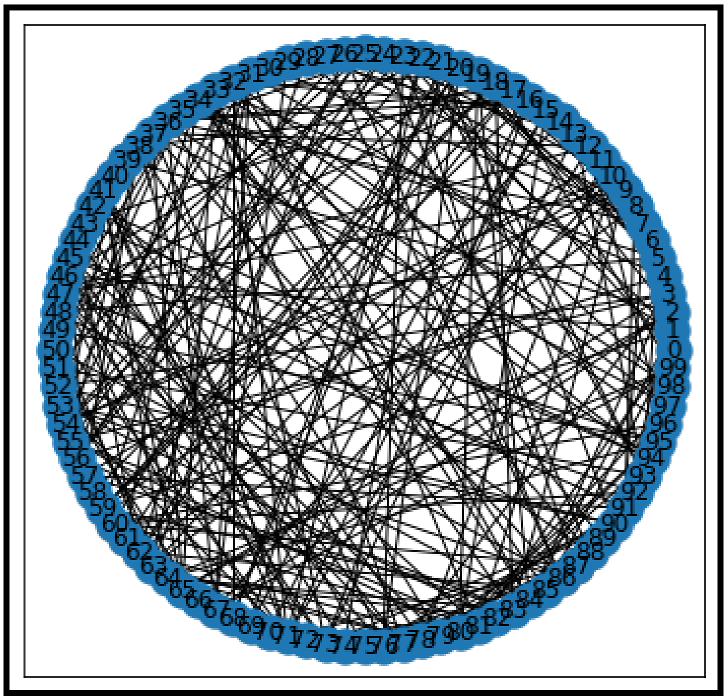


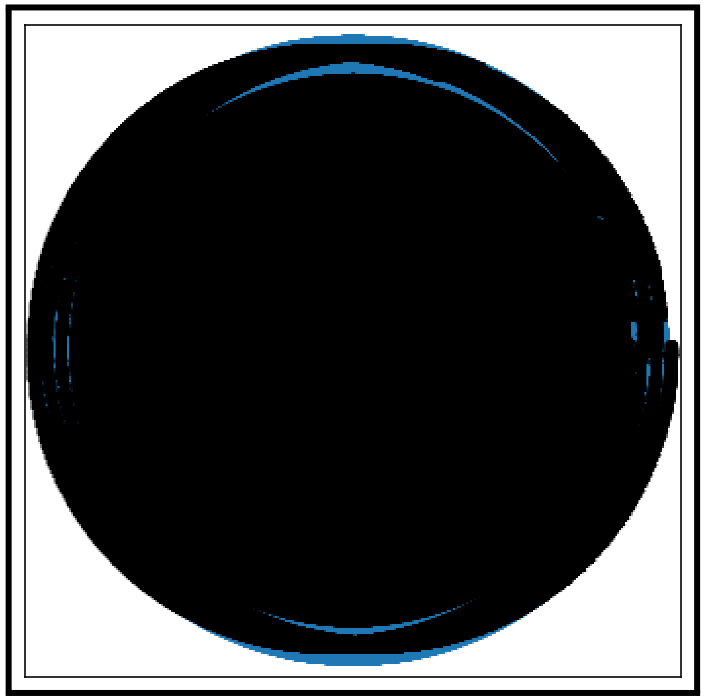


Average Shortest Path	3.310160353610434
Average Clustering Coefficient	0.03939168349616133

Dataset 2 : ca-GrQc

Following are the 2 Watts-Strogatz graphs that are generated using the above function. The Watts-Strogatz Graph which is left is generated with 100 nodes with value of p = 0.5 and value of k = 10. And the Watts-Strogatz Graph which is in right is generated with all the nodes present in the dataset (which is 5242) with p = 0.5 and value of k = 10.

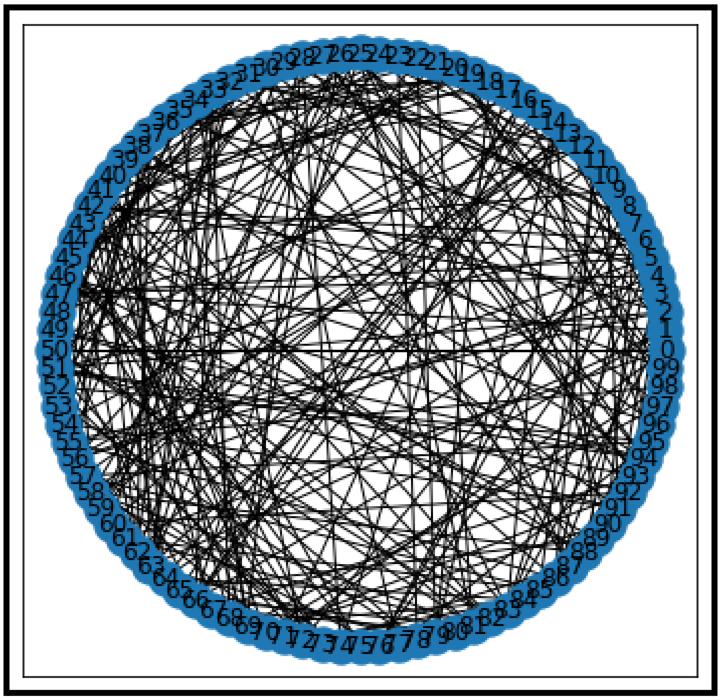


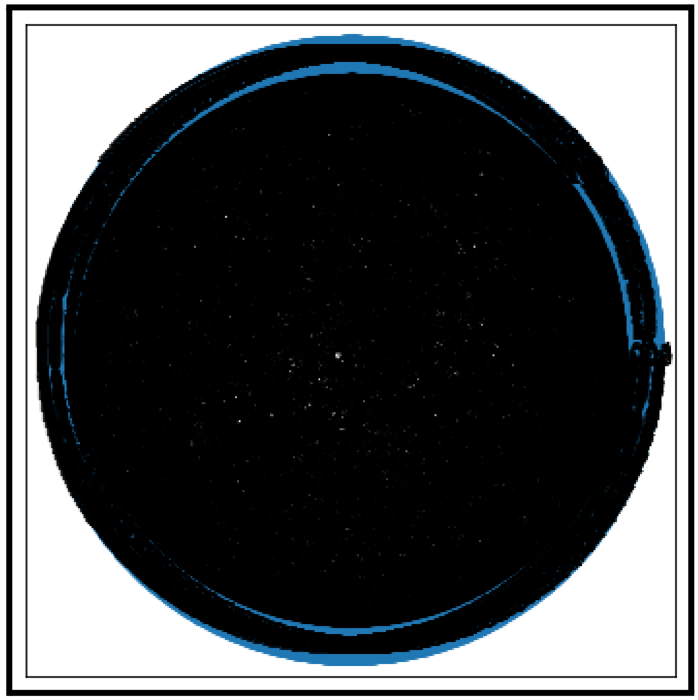


Average Shortest Path	3.308790707815504
Average Clustering Coefficient	0.03964633763141248

Dataset 3 : email-Eu-core

Following are the 2 Watts-Strogatz graphs that are generated using the above function. The Watts-Strogatz Graph which is left is generated with 100 nodes with value of p = 0.5 and value of k = 10. And the Watts-Strogatz Graph which is in right is generated with all the nodes present in the dataset (which is 1005) with p = 0.5 and value of k = 10.





Average Shortest Path	3.309421022378149
Average Clustering Coefficient	0.035902442195127525

Model 3. Barabasi-Albert Model

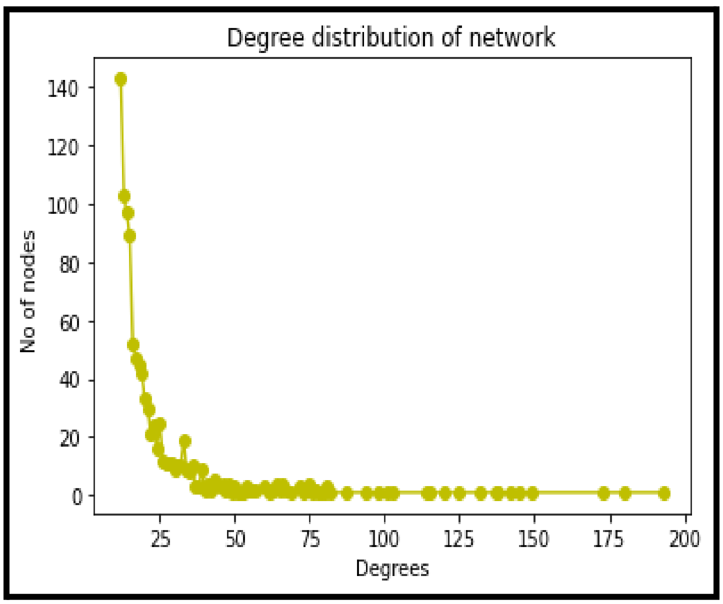
Goal - Start with a small network with a few nodes and understand how it grows by adding new nodes.

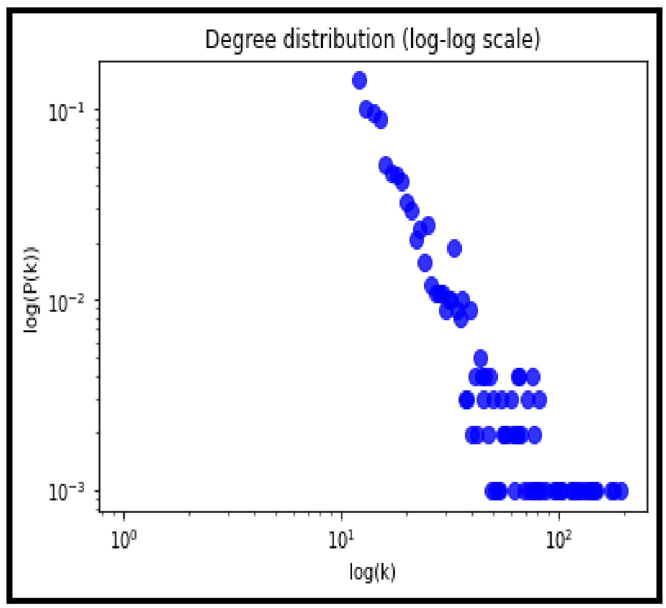
The Barabási–Albert model is one of several proposed models that generate scale-free networks. It incorporates two important general concepts: **growth** and **preferential attachment**. Both growth and preferential attachment exist widely in real networks.

Growth: means that the number of nodes in the network increases over time.
Preferential attachment: means that the more connected a node is, the more likely it is to receive new links/edges [*rich getting richer and the poor getting poorer*].

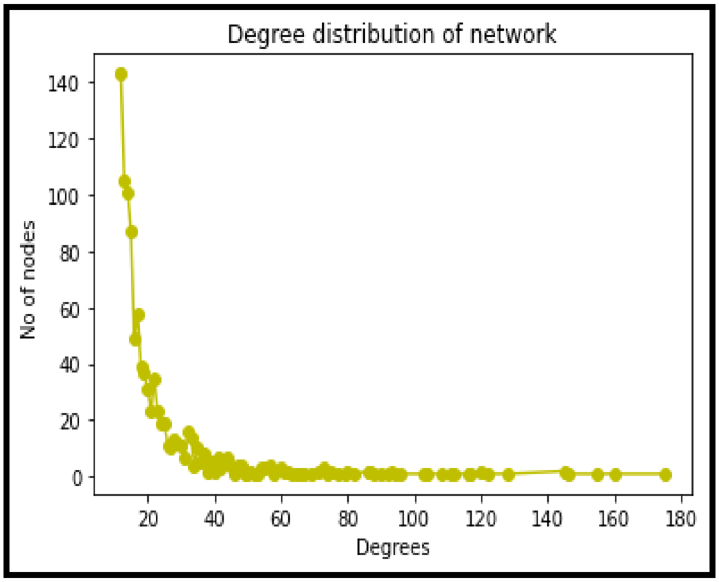
a) Degree Distribution of Barabasi-Albert Model: The degree of a node in a network is the number of connections it has to other nodes and the degree distribution is the probability distribution of these degrees over the whole network.

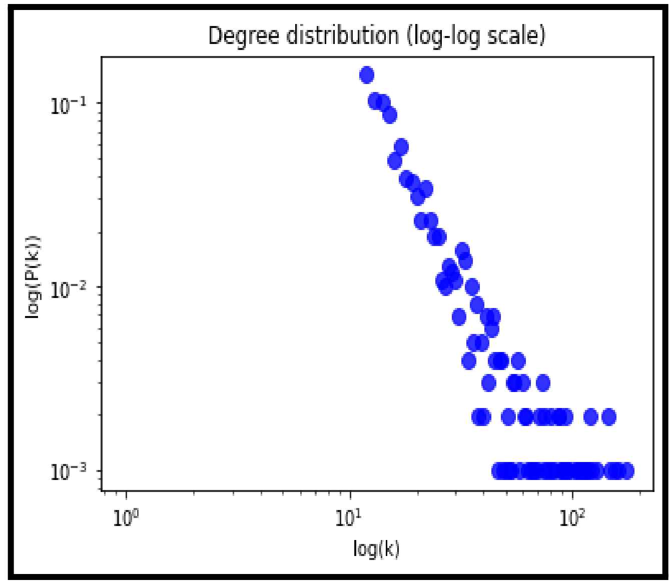
Dataset 1 : soc-sign-bitcoin-otc



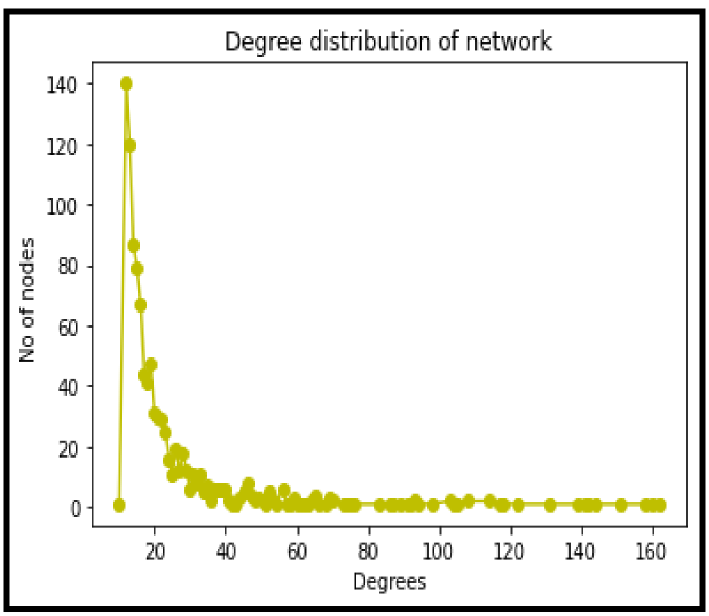


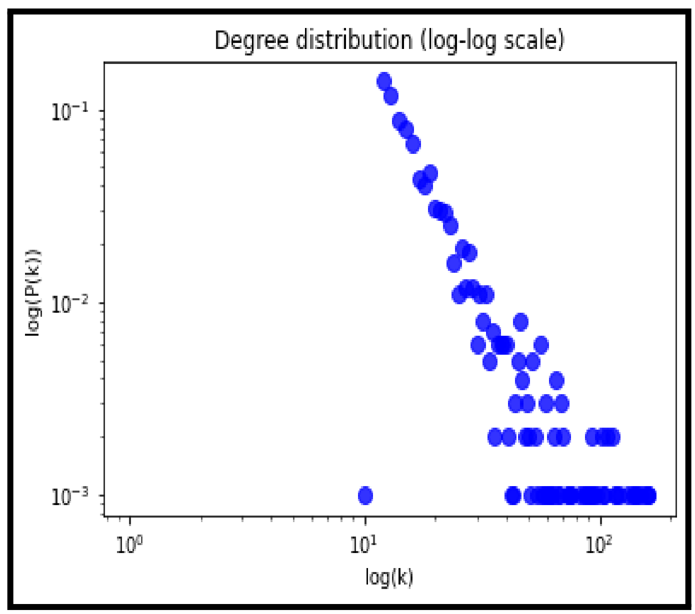
Dataset 2 : ca-GrQc





Dataset 3 : email-Eu-core





b) Barabasi-Albert's Random Graph

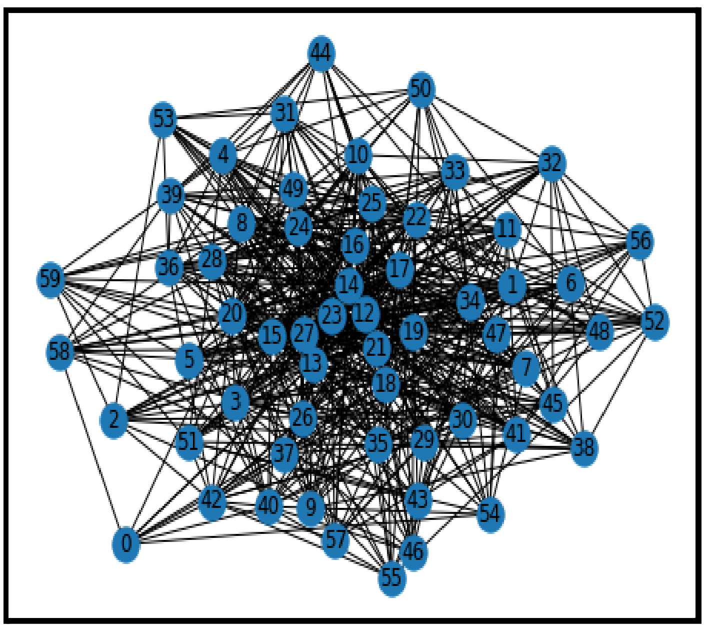
Barabasi-Albert model gives a random graph according to the Barabási–Albert preferential attachment model. A graph of n nodes is grown by attaching new nodes, each with m edges that are preferentially attached to existing nodes with high degree.To generate the Albert-Barabasi Random Graph, I have used the following function:

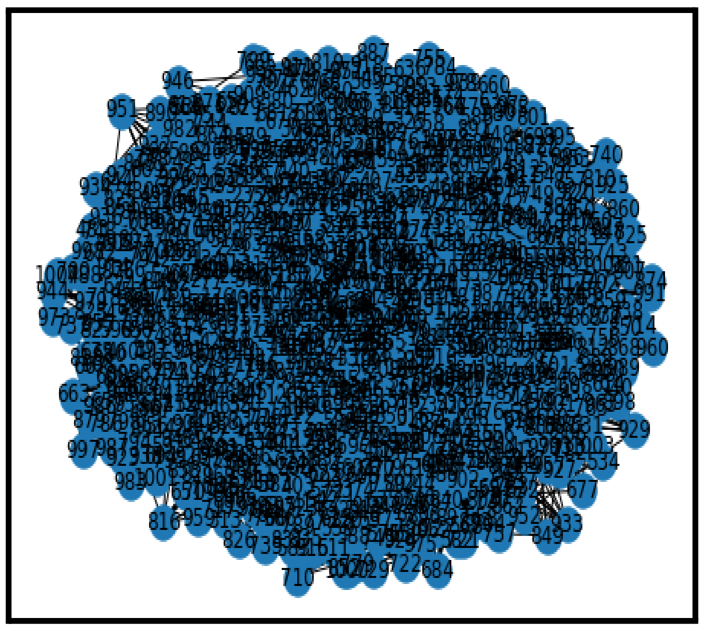
barabasi_albert_graph(n,m)

Parameters: *n (int) – The number of nodes.*
m (int) – Number of edges to attach from a new node to existing nodes.

Dataset 1 : soc-sign-bitcoin-otc

Following are the 2 Barabasi-Albert random graphs that are generated using the above function. The Barabasi-Albert random graph which is left is generated with 60 nodes with value of m = 12. And the Barabasi-Albert random Graph which is in right is generated with all the nodes present in the dataset (which is 5881) with m = 12.

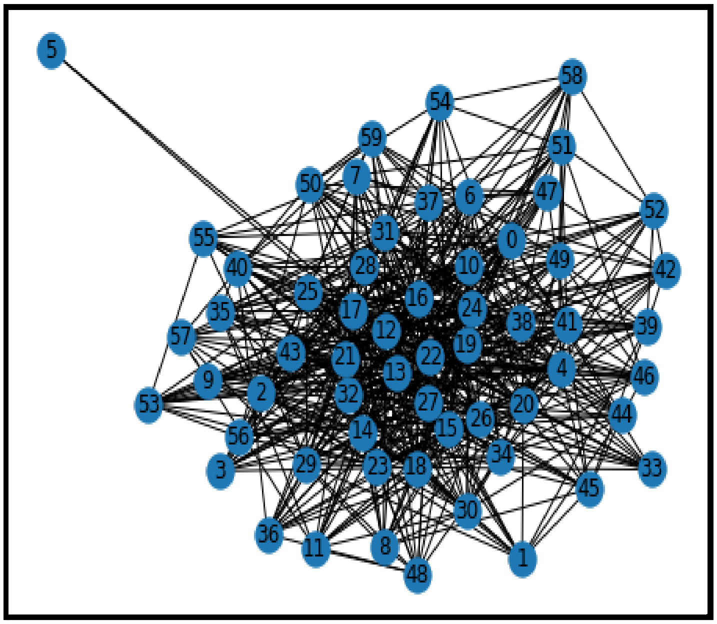


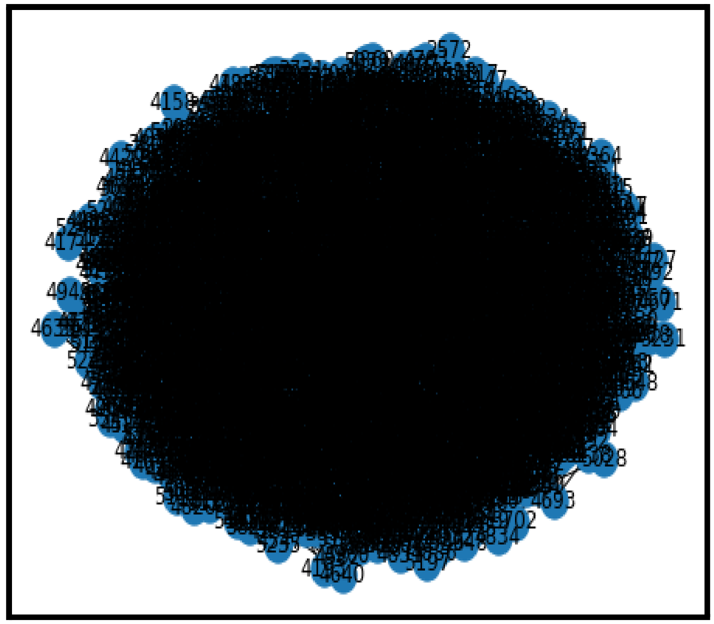


Average Shortest Path	2.4629264038373866
Average Clustering Coefficient	0.06670875197399005

Dataset 2 : ca-GrQc

Following are the 2 Barabasi-Albert random graphs that are generated using the above function. The Barabasi-Albert random graph which is left is generated with 60 nodes with value of m = 12. And the Barabasi-Albert random Graph which is in right is generated with all the nodes present in the dataset (which is 5242) with m = 12.

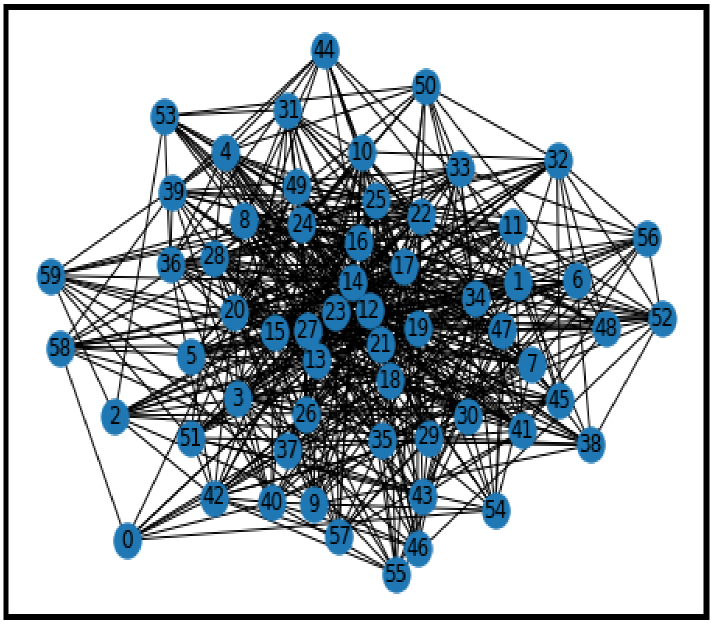


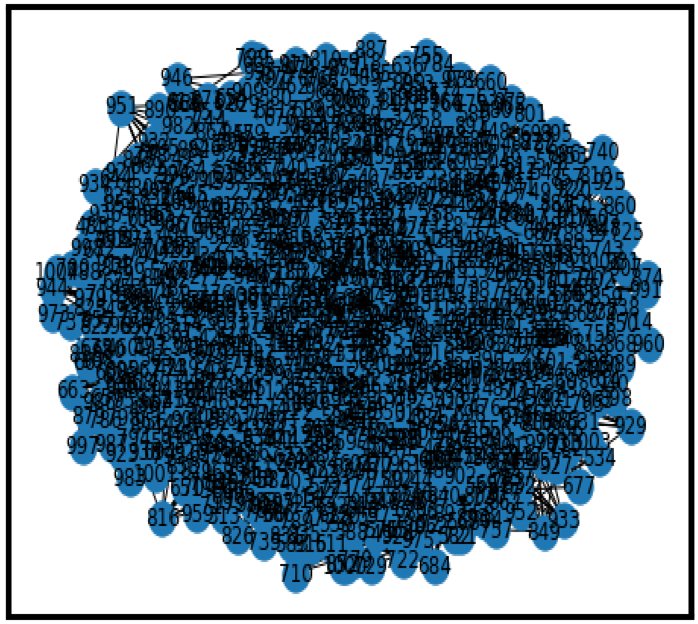


Average Shortest Path	2.4633664347584787
Average Clustering Coefficient	0.06529057234530404

Dataset 3 : email-Eu-core

Following are the 2 Barabasi-Albert random graphs that are generated using the above function. The Barabasi-Albert random graph which is left is generated with 60 nodes with value of m = 12. And the Barabasi-Albert random Graph which is in right is generated with all the nodes present in the dataset (which is 1005) with m = 12.





Average Shortest Path	2.4573209648966325
Average Clustering Coefficient	0.0687039251555958

----------------*--*--------------------