ECE232E: LARGE SCALE SOCIAL AND COMPLEX NETWORKS: DESIGN AND ALGORITHMS

Project 1: Random Graphs and Random Walks

April 20, 2020

Wanli Gao, UID: 105431975
Yifan Zhang, UID: 805354474
Tianyi Zhao, UID: 804380974

1 Part 1: Generating Random Networks

1.1 Create random networks using ER model

(a)
We create an ER model with n=1000 nodes, and the probability p for drawing an edge between two arbitrary vertices 0.003, 0.004, 0.01, 0.05, and 0.1. The degree distributions for different p is shown below:

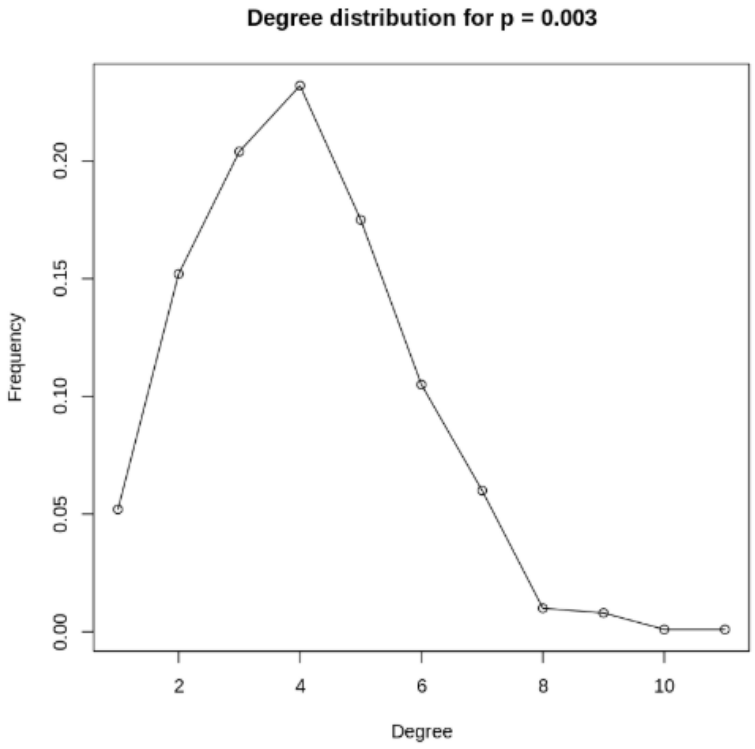

Figure 1.1. Degree distribution for p = 0.003

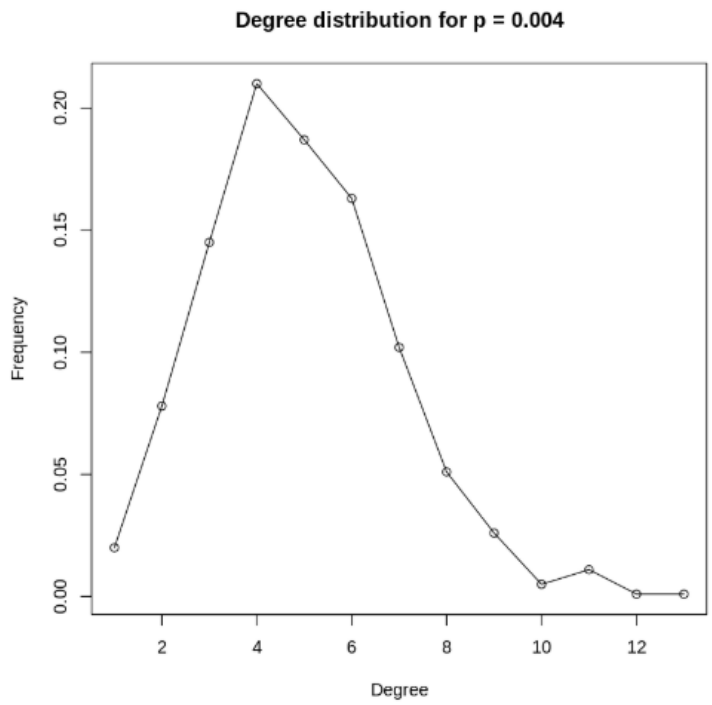

Figure 1.2. Degree distribution for p = 0.004

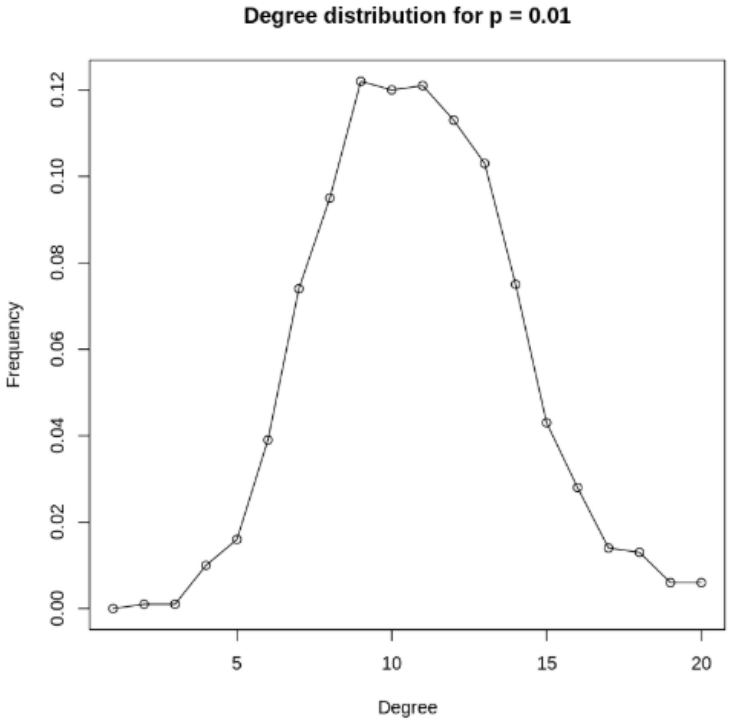

Figure 1.3. Degree distribution for p = 0.01

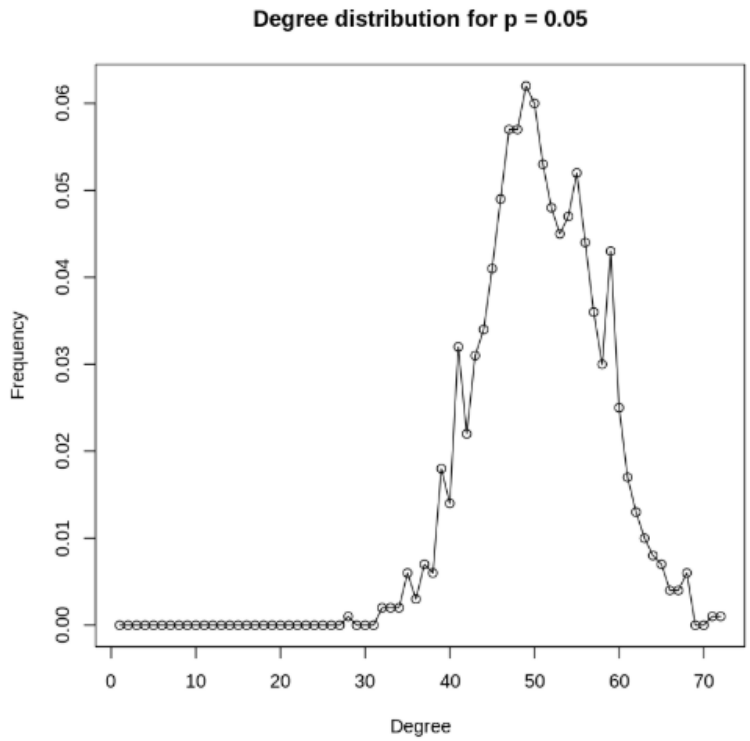

Figure 1.4. Degree distribution for p = 0.05

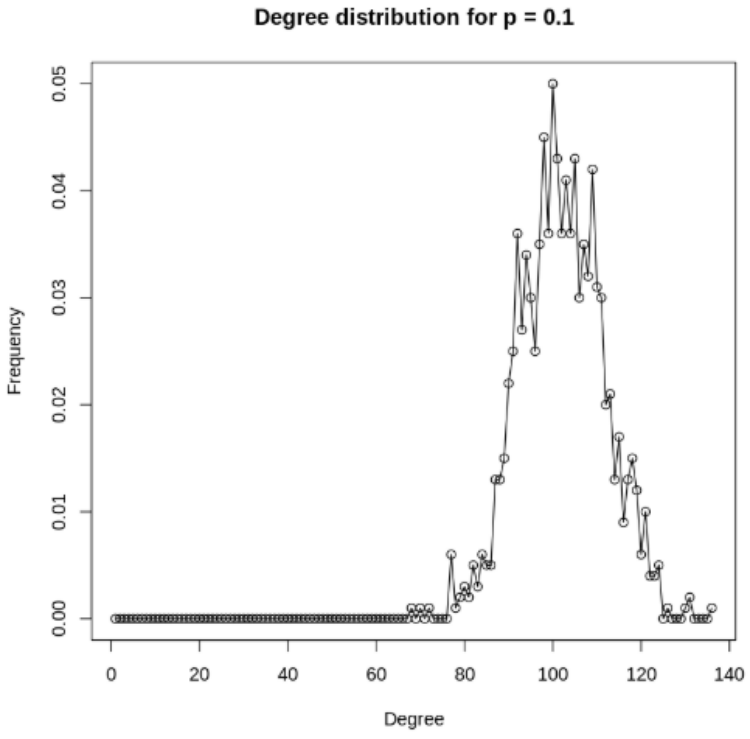

Figure 1.5. Degree distribution for p = 0.1

From the plot we can see that the degree distribution is binomial distribution. This is because for an ER model, the edge formation probability is equal (which is p), so the probability of a node to have degree k can be expressed by:

$$P(degree(v) = k) = \binom{n}{k} p^k (1-p)^{n-k}$$

This is a binomial distribution.

The measured mean and variance of the degree distributions for different p are shown below:

Table 1. Mean and variance of the degree distributions for different p

Probability	0.003	0.004	0.01	0.05	0.1
Mean	2.994	3.916	9.81	49.708	101.024
Variance	2.863	3.889	9.405	46.449	91.707

For a binomial distribution, we can calculate the theoretical mean and variance of the degree distributions for different probability using the following fomula:

$$\mu = (n-1)p$$
$$\sigma^2 = (n-1)p(1-p)$$

Theoretical mean and variance of the degree distributions for different probability is shown below:

Table 2. Theoretical mean and variance of the degree distributions for different p

Probability	0.003	0.004	0.01	0.05	0.1
Theoretical mean	2.997	3.996	9.99	49.95	99.9
Theoretical variance	2.988	3.980	9.890	47.453	89.91

We can see that the measured mean and variance of the degree distributions are very close to the theoretical values.

(b) In this part, we generate a network using ER model, for each p and n = 1000, the statistics for probability and Diameter measurement of ER networks is shown below:

Table 3. Probability and Diameter measurement of ER networks

p value	Is Always Connected	Connected Probability	Diameter of GCC
0.003	FALSE	0	14
0.004	FALSE	0	11
0.01	TRUE	0.957	5
0.05	TRUE	1	3
0.1	TRUE	1	3

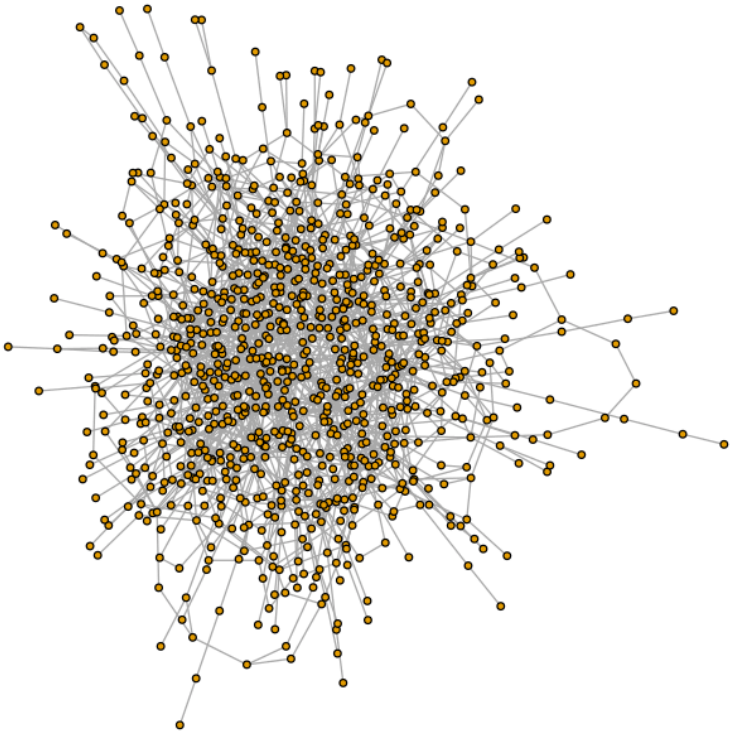

Figure 1.6. GCC for p = 0.003

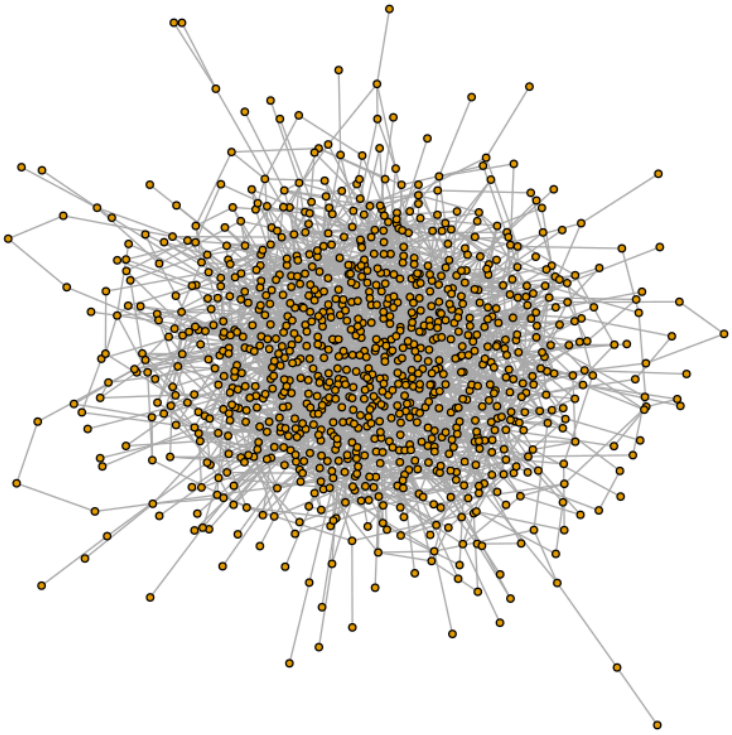

Figure 1.7. GCC for p = 0.004

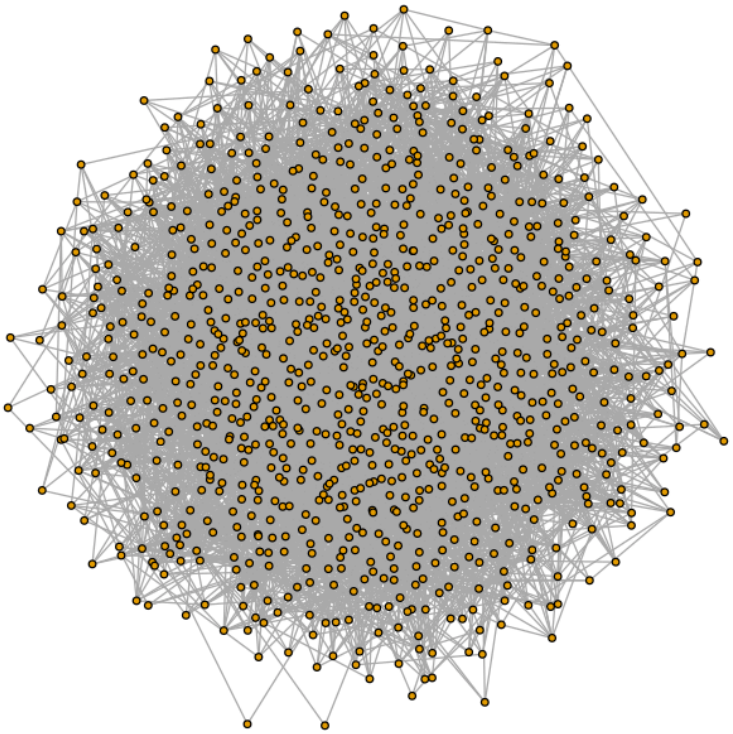

Figure 1.8. GCC for p = 0.01

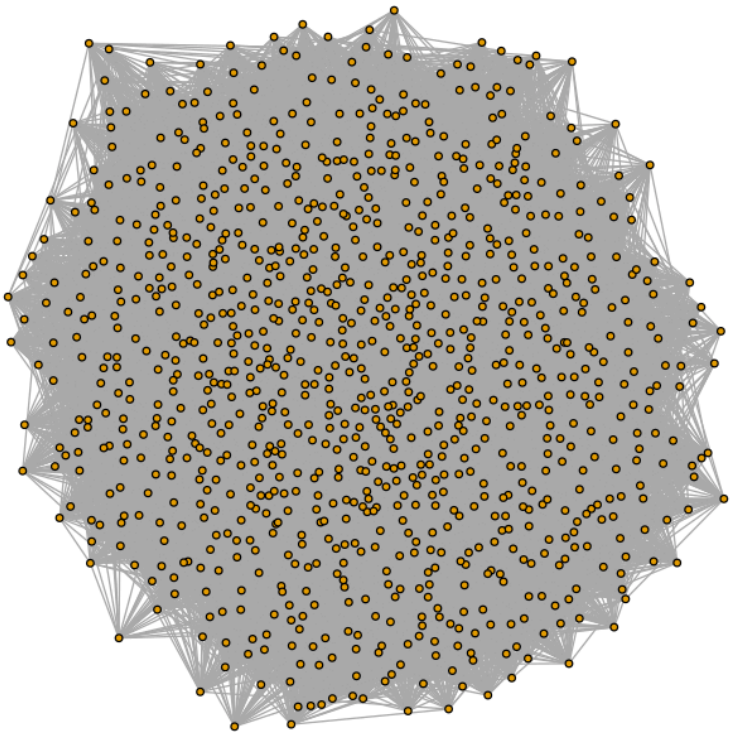

Figure 1.9. GCC for p = 0.05

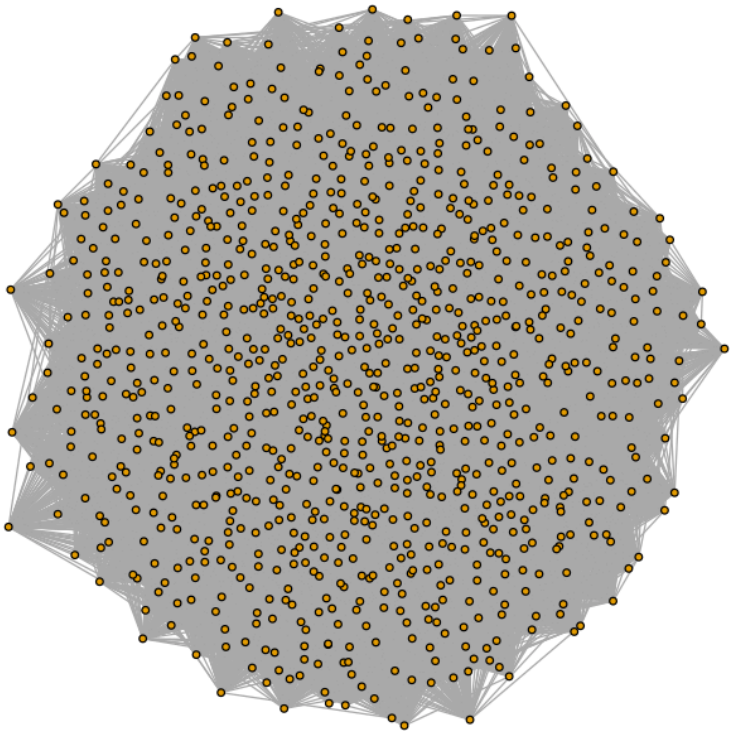

Figure 1.10. GCC for p = 0.1

From the figures and the table shown above, it seems that the diameter of GCC for non-connected network decrease as p value increases. Besides, it is clear that the connected probability increases as p value increases. Finally, we conclude that the graph is all connected if p value is high enough (where p value is around 0.05).

(c)As we discussed before, diameter of GCC is nonlinear as p value increases. Given P = O($\frac{1}{n}$) and P = O($\frac{\ln n}{n}$), as well as n - 1000, we plot the normalized GCC sizes vs p, and a line of the average normalized GCC sizes for each p along with the scatter plot as below:

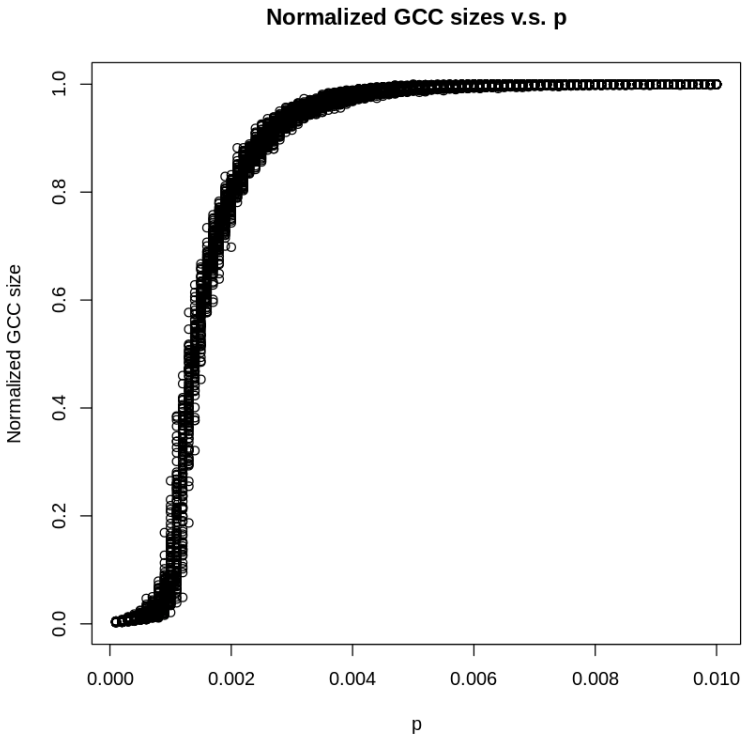

Figure 1.11. normalized GCC sizes vs p

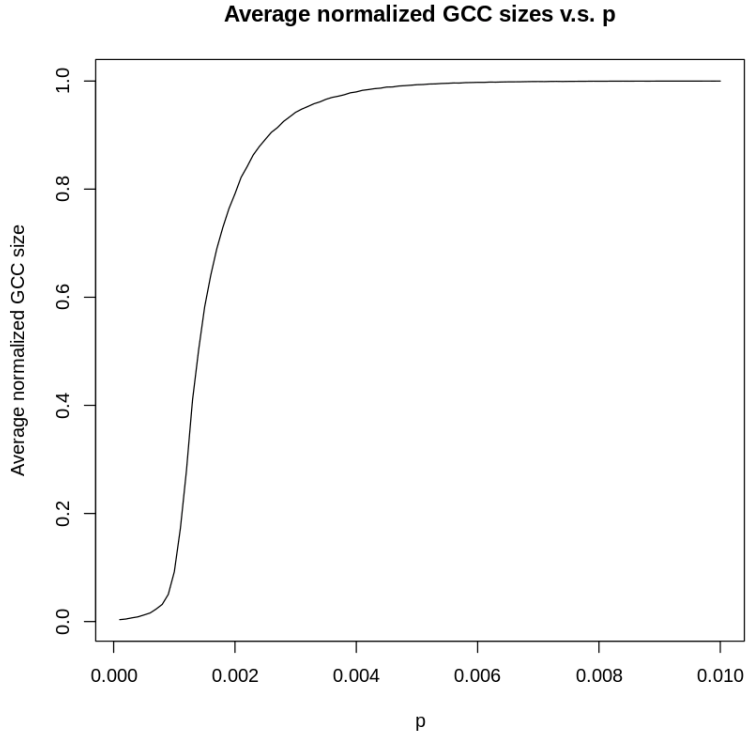

Figure 1.12. average normalized GCC sizes for each p

(i) From the figures above we can clearly see that the value of p where a giant connected component starts to emerge is about 0.001. We define emergence as where the slope of the line looks close to 0. They match with theoretical values mentioned or derived in lectures.

(ii) The giant connected component takes up over 99 percent of the nodes in almost every experiment when p value is around 0.007.

(d)

We sweep over the number of nodes, n, ranging from 100 to 10000, with step = 100, and plot the expected size of the GCC of ER networks with n nodes and edge-formation probabilities p = c/n.

(i) For c = 0.5:

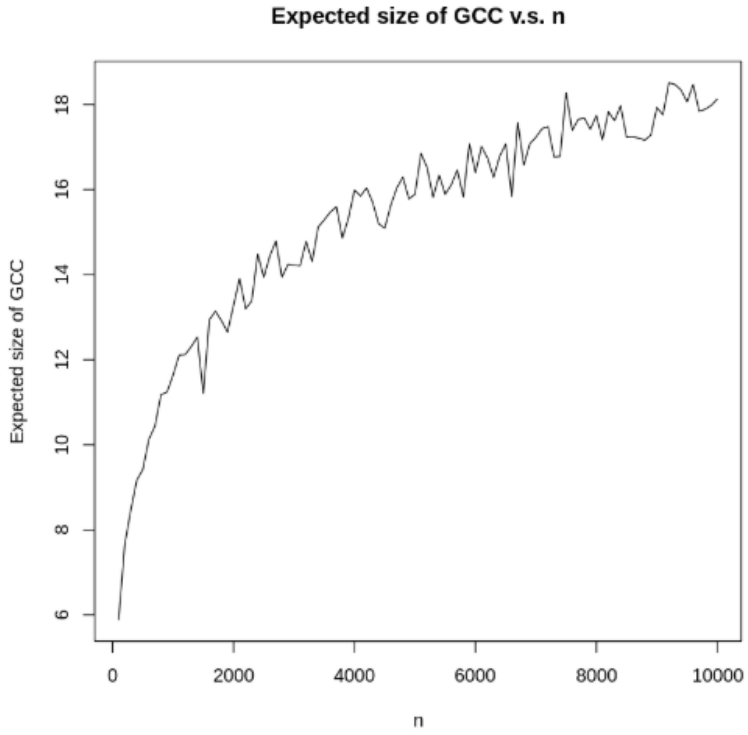

Figure 1.13. Expected size of GCC v.s. n for c = 0.5

From the plot we can see the relation between the expected GCC size and n follows the log trend. This is because when c = 0.5, the expected size of GCC is of O(ln n).

(ii) For c = 1:

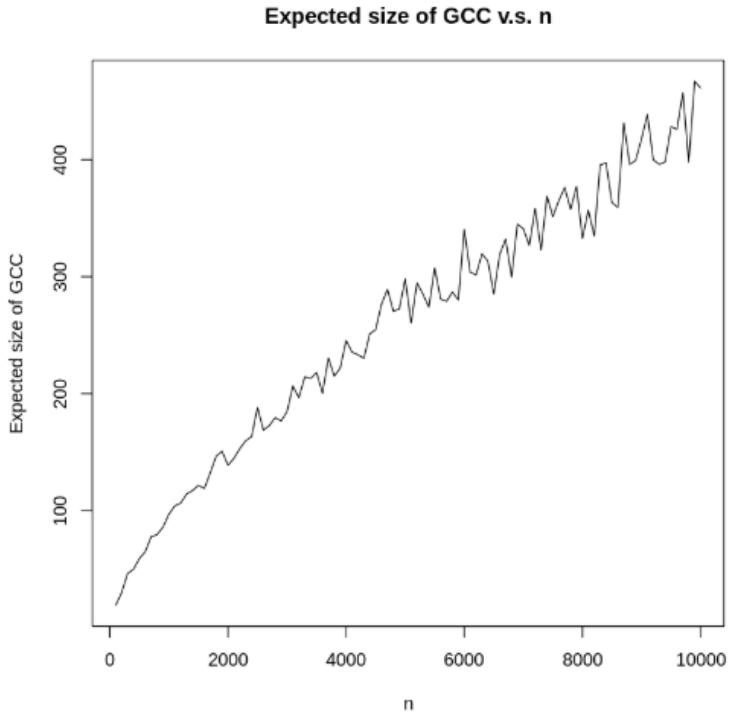

Figure 1.14. Expected size of GCC v.s. n for c = 1

From the plot we can see the relation between the expected GCC size and n is closer to linear trend. This is because when c = 1, the expected size of GCC is of O($\sqrt{n}$).

(iii) For c = 1.1, 1.2, 1.3:

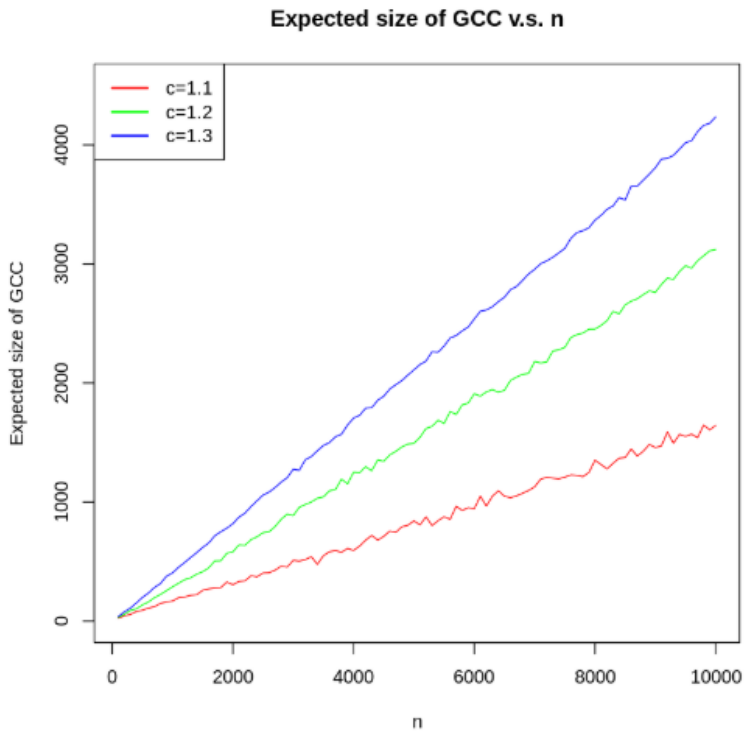

Figure 1.15. Expected size of GCC v.s. n for c = 1.1, 1.2, 1.3

From the plot we can see the relation between the expected GCC size and n shows very good linearity. This is because when c > 1, the expected size of GCC is of O(n). The slope is larger with higher c value.

1.2 Create networks using preferential attachment model

(a)We created an undirected network with n = 1000 nodes, with preferential attachment model, where each new node attaches m = 1 old nodes.
Since we observed that the network is connected in each of the instance we have, so we can definitely say that a graph with these parameters is always connected. The figure below shows our randomly generated network.

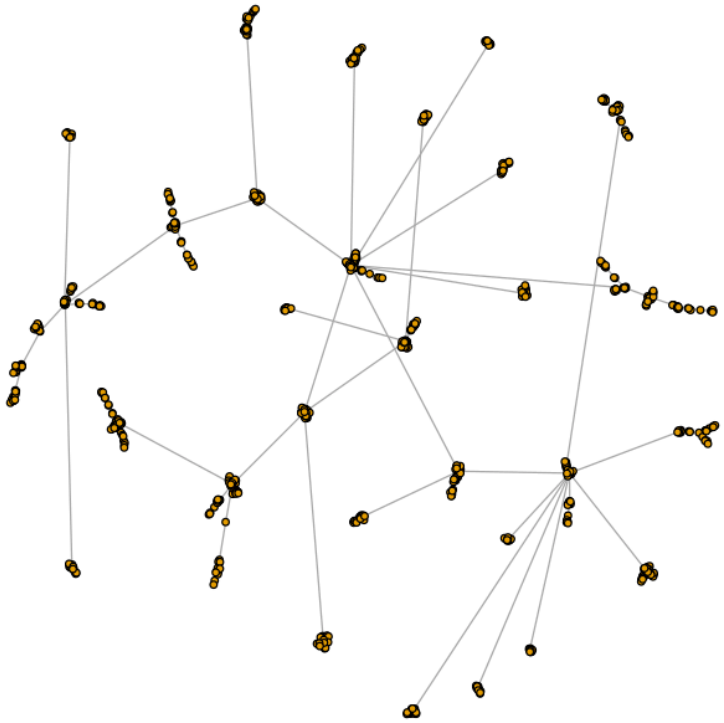

Figure 1.16. Randomly-generated network

(b)
We use the fast greedy method to find the community structure for the network with n = 1000 nodes, which is shown below:

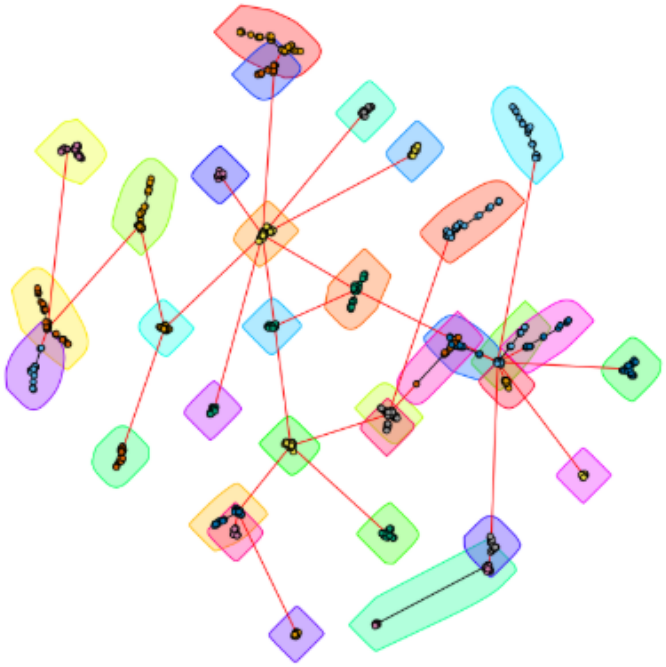

Figure 1.17. Community structure for undirected network with n = 1000

The modularity is 0.933.
(c)
We generate a larger network with 10000 nodes, which is shown below:

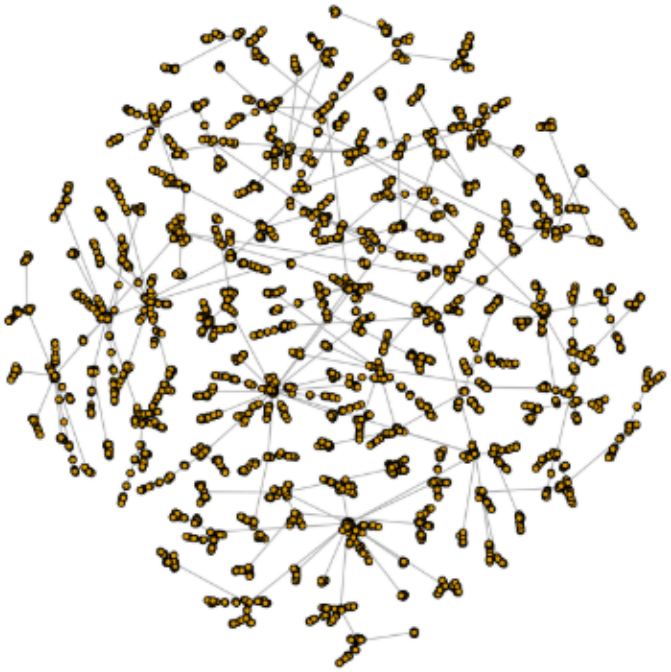

Figure 1.18. Larger network with n = 10000

We use the fast greedy method to find the community structure for the network with n = 10000 nodes, which is shown below:

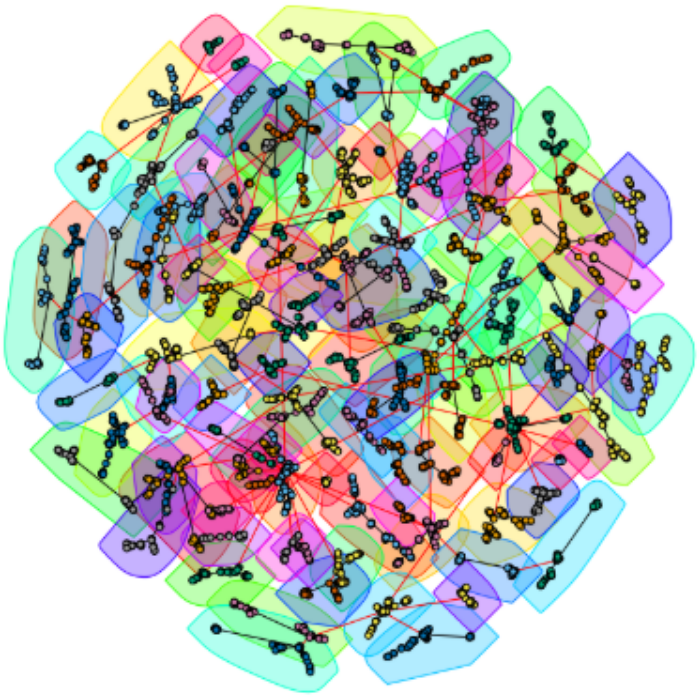

Figure 1.19. Community structure for undirected network with n = 10000

The modularity is 0.978.

(d) We plot the degree distribution in a log-log scale for n = 1000, 10000, with linear regression used for estimating the slope of the plot:

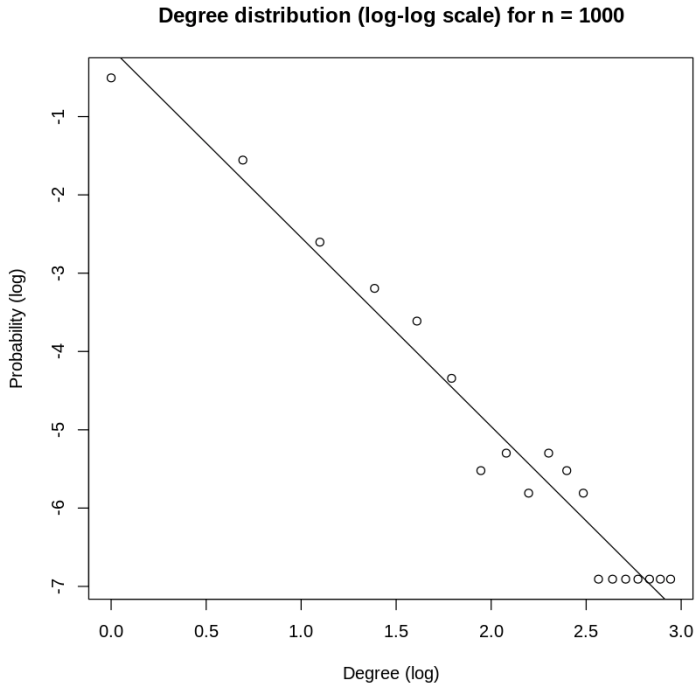

Figure 1.20. Degree distribution (log-log scale) for n = 1000

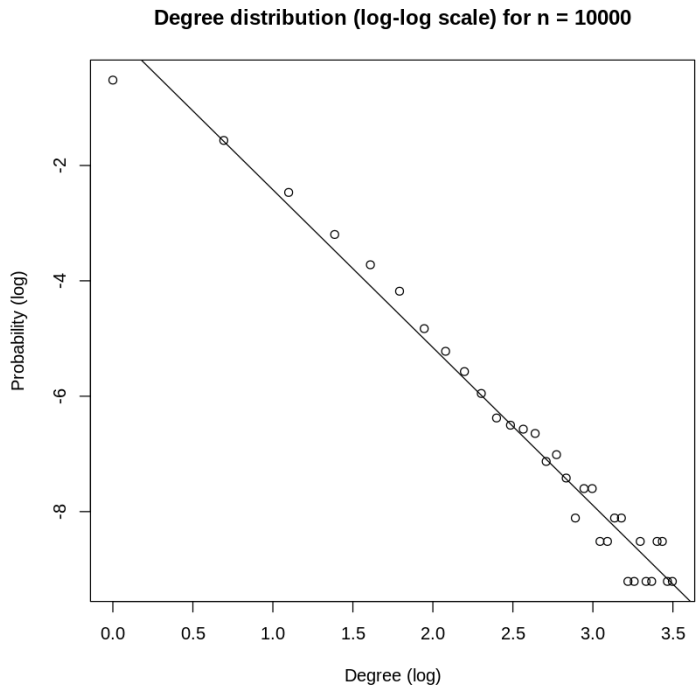

Figure 1.21. Degree distribution (log-log scale) for n = 10000

By using linear regression, the estimation for the slope of the plot (n = 1000) is -2.41; the estimation for the slope of the plot (n = 10000) is -2.74.

(e)
We randomly pick a node i, and then randomly pick a neighbor j of that node. Then we plot the degree distribution of nodes j that are picked with this process, in the log-log scale.The distribution in log-log scale is not linear.
For the network with n = 1000 nodes:

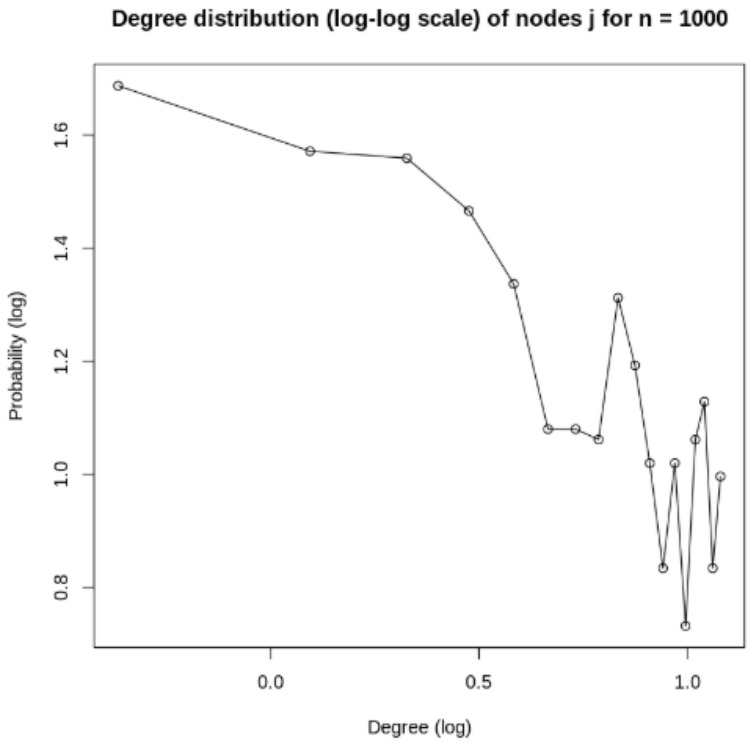

Figure 1.22. Degree distribution (log-log scale) of nodes j for n = 1000

For the network with n = 10000 nodes:

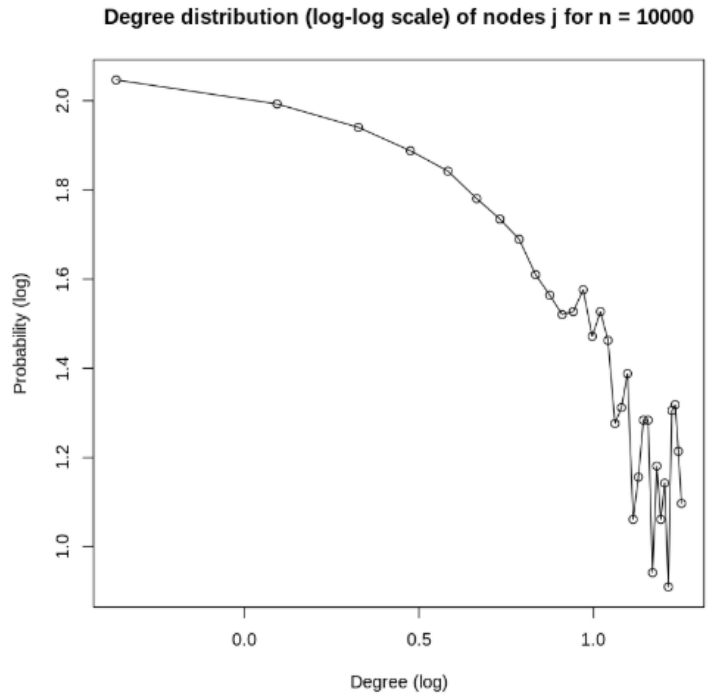

Figure 1.23. Degree distribution (log-log scale) of nodes j for n = 10000

(f)The relationship between the age of nodes and their expected degree is plotted as below:

Figure 1.24. Expected degree v.s. Age of nodes(n = 1000)

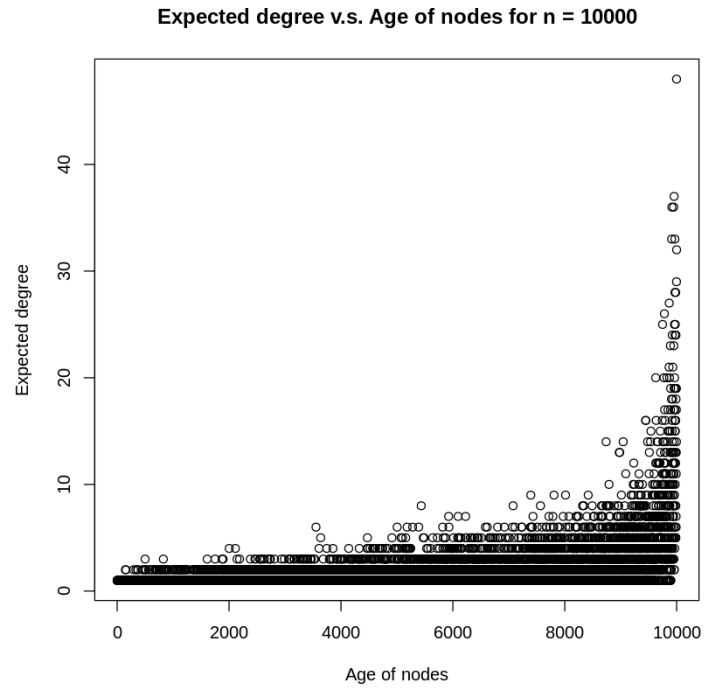

Figure 1.25. Expected degree v.s. Age of nodes(n = 10000)

In the figures shown above, we can conclude that as the age of nodes increases, there is an exponential increase for the expected degree.

(g)

For m = 2:

(a) We created an undirected network with n = 1000 nodes, with preferential attachment model, where

each new node attaches m = 2 old nodes.

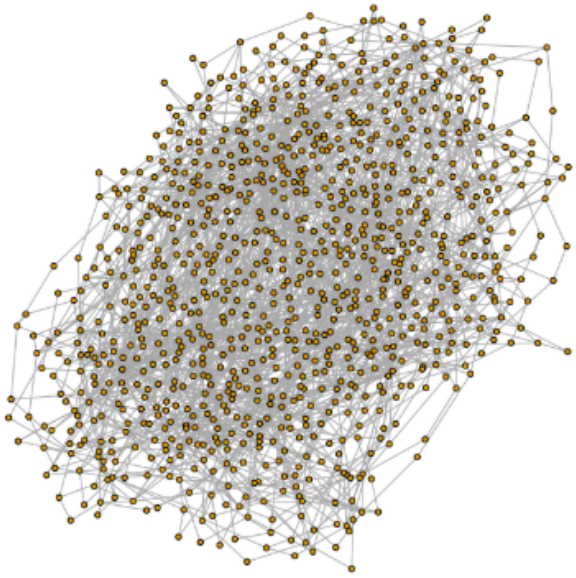

Figure 1.26. Undirected network with n = 1000, m = 2

The network is connected. There are more edges compared with m = 1.

(b) For n = 1000, the community structure is:

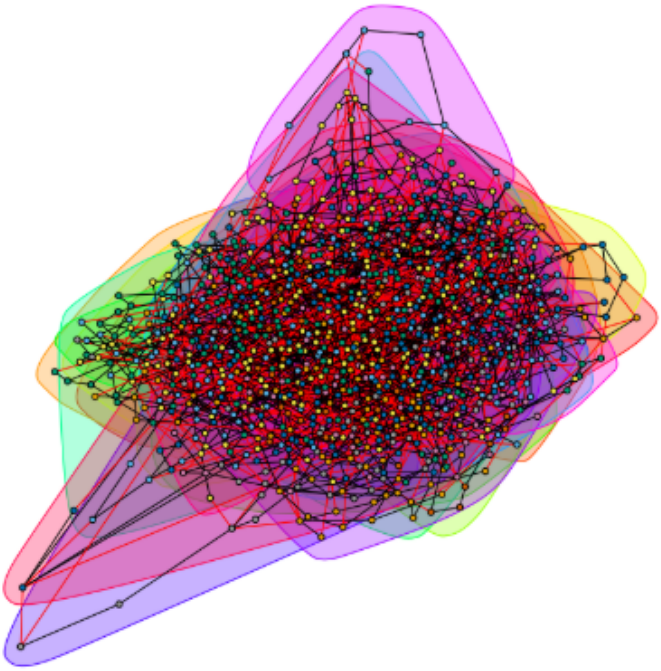

Figure 1.27. Community structure for undirected network with n = 1000, m = 2

The modularity is 0.523, which is lower than the modularity of m = 1.

(c) We generate a larger network with 10000 nodes, m = 2, which is shown below:

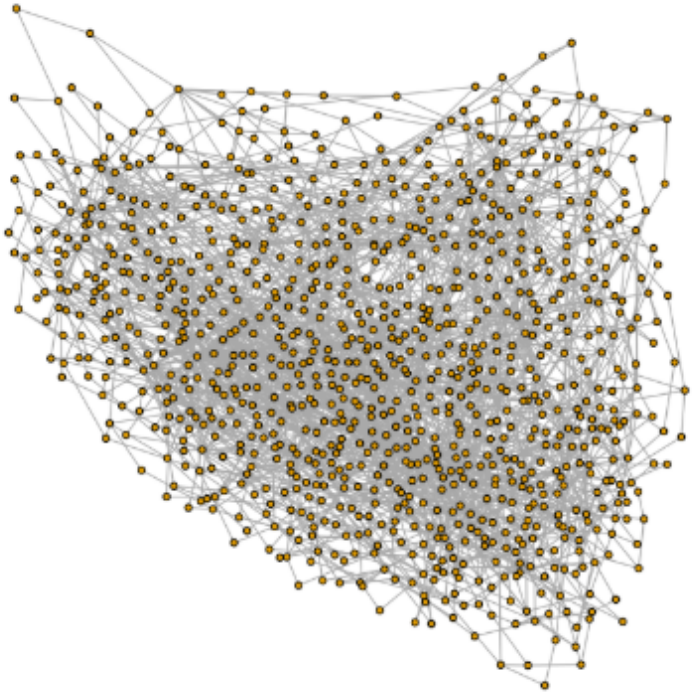

Figure 1.28. Larger network with n = 10000, m = 2

The community structure is:

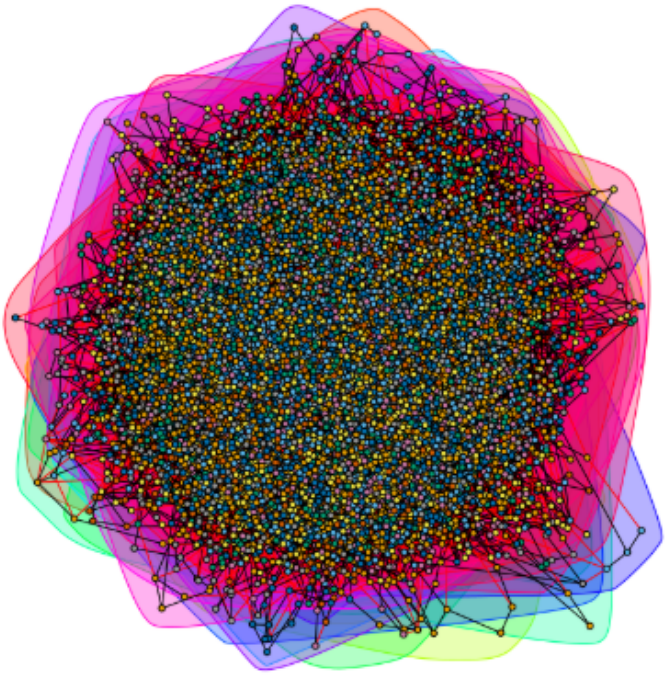

Figure 1.29. Community structure for undirected network with n = 10000, m = 2

The modularity is 0.532. This is also lower than m = 1.

(d) We plot the degree distribution in a log-log scale for n = 1000, 10000, m = 2, with linear regression used for estimating the slope of the plot:

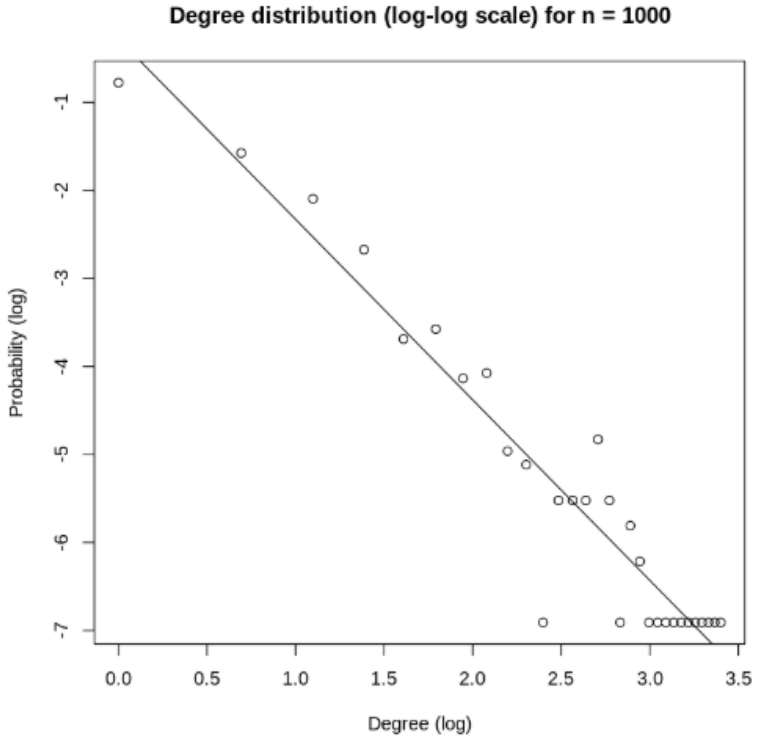

Figure 1.30. Degree distribution (log-log scale) for n = 1000, m = 2

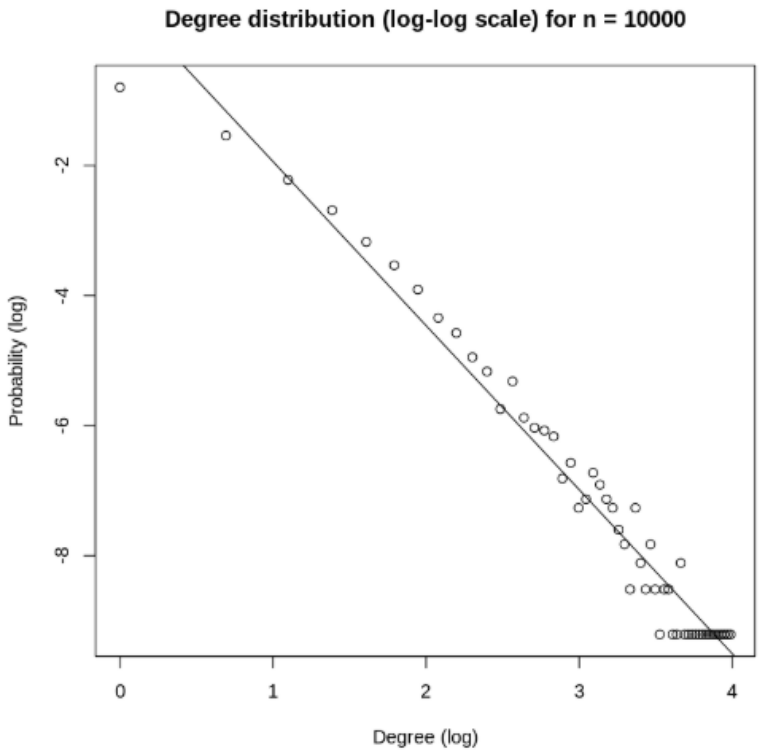

Figure 1.31. Degree distribution (log-log scale) for n = 10000, m = 2

By using linear regression, the estimation for the slope of the plot (n = 1000) is -2.05; the estimation for the slope of the plot (n = 10000) is -2.52. Compared with m = 1, the slope is slightly smaller in absolute value.

(e) We plot the degree distribution of nodes j that are picked with this process, in the log-log scale. The distribution is still not linear.
For the network with n = 1000, m = 2:

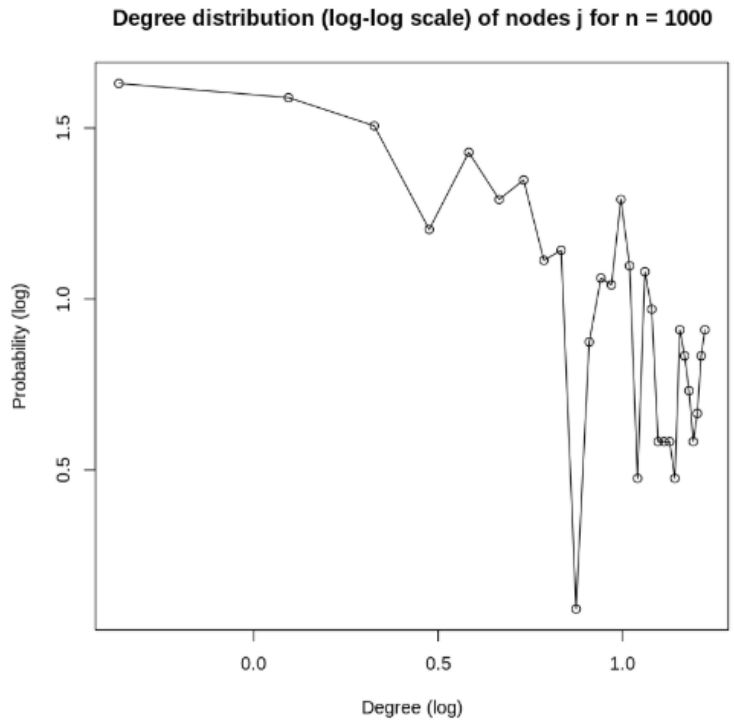

Figure 1.32. Degree distribution (log-log scale) of nodes j for n = 1000, m = 2

For the network with n = 10000, m = 2:

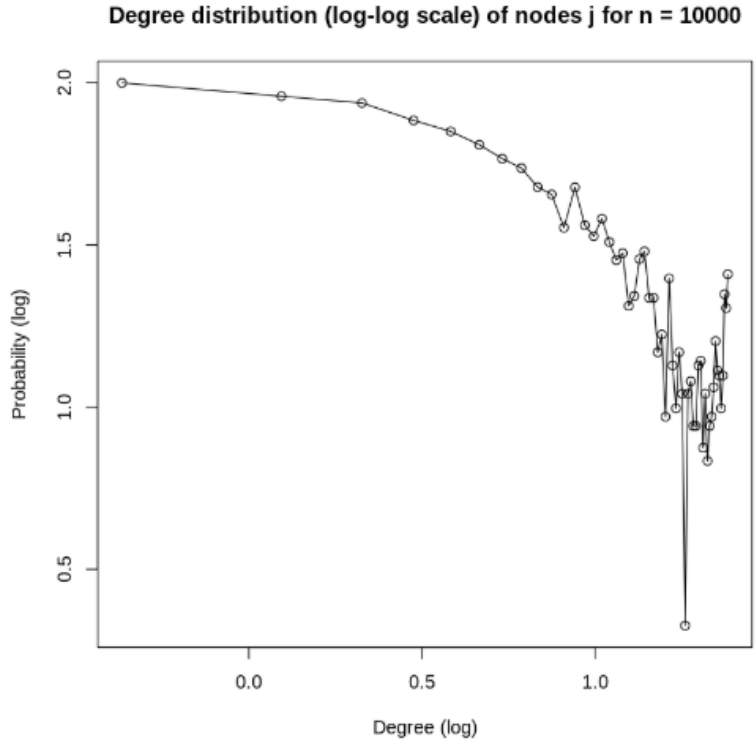

Figure 1.33. Degree distribution (log-log scale) of nodes j for n = 10000, m = 2

(f) The relationship between the age of nodes and their expected degree is plotted as below:

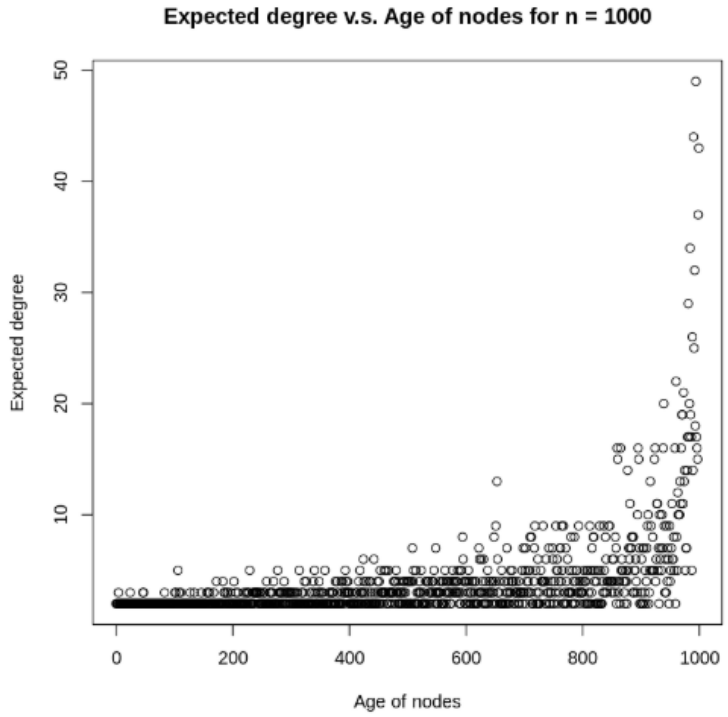

Figure 1.34. Expected degree v.s. Age of nodes (n = 1000, m = 2)

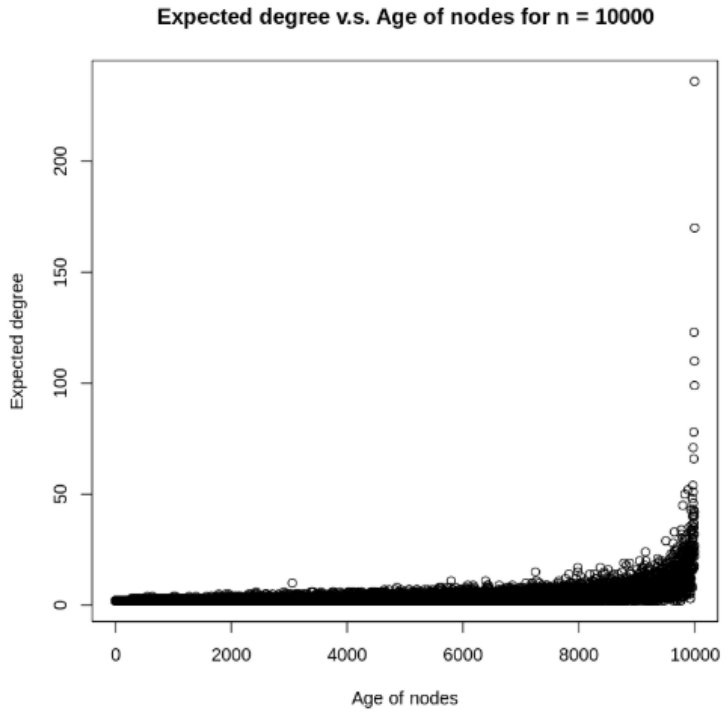

Figure 1.35. Expected degree v.s. Age of nodes (n = 10000, m = 2)

In the figures shown above, we can conclude that as the age of nodes increases, there is an exponential

increase for the expected degree. Compared with m = 1, the increase speed is slower.

For m = 5:
(a) We created an undirected network with n = 1000 nodes, with preferential attachment model, where each new node attaches m = 5 old nodes. There are more edges compared with m = 1 and m = 2.

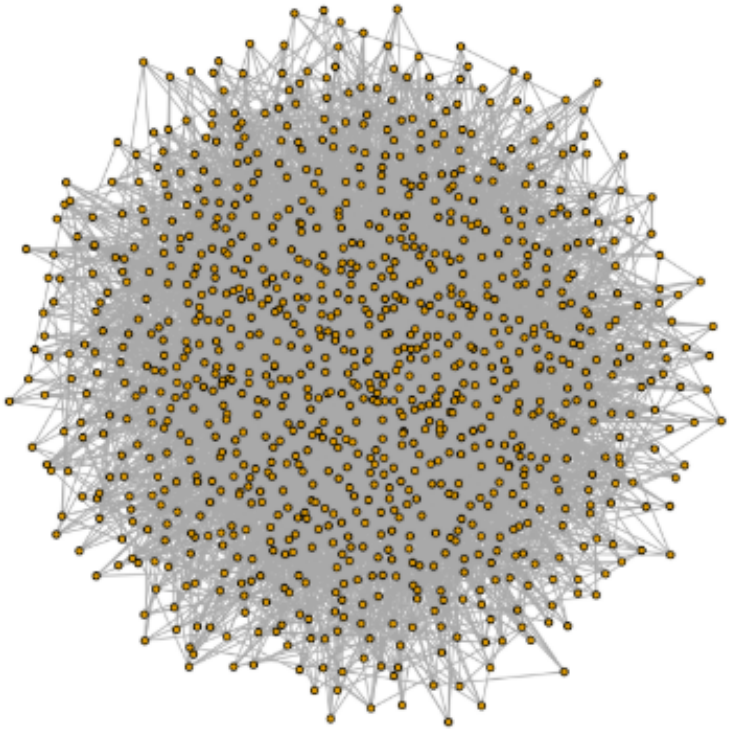

Figure 1.36. Undirected network with n = 1000, m = 5

The network is connected.

(b) For n = 1000, the community structure is:

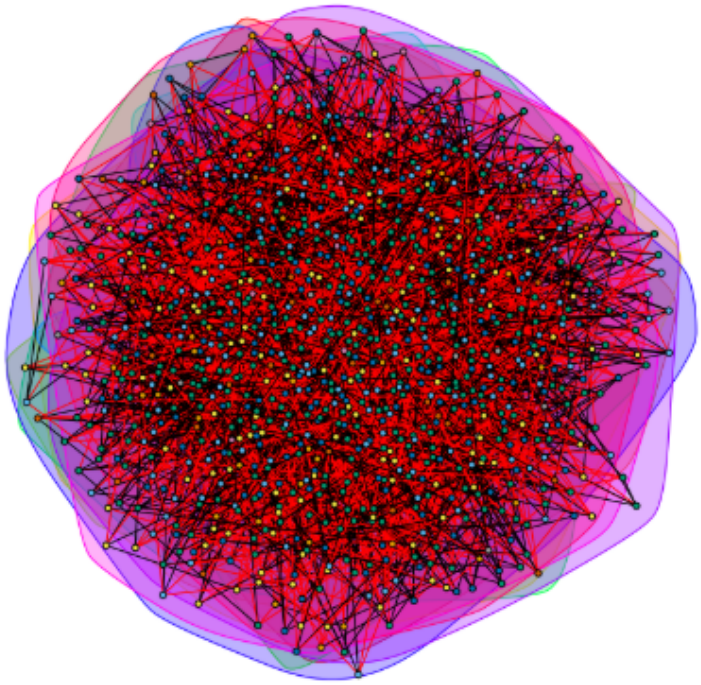

Figure 1.37. Community structure for undirected network with n = 1000, m = 5

The modularity is 0.273, which is lower than the modularity of m = 1 and m = 2.

(c) We generate a larger network with 10000 nodes, m = 5, which is shown below:

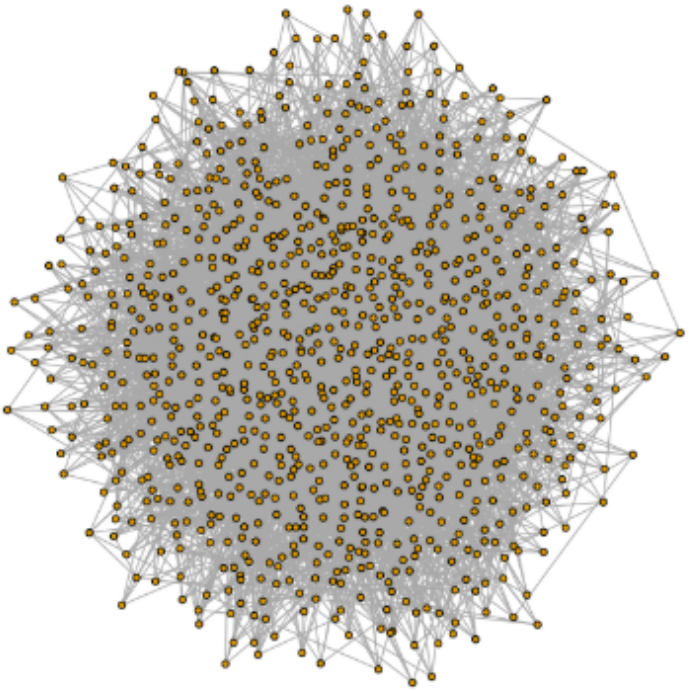

Figure 1.38. Larger network with n = 10000, m = 5

The community structure is:

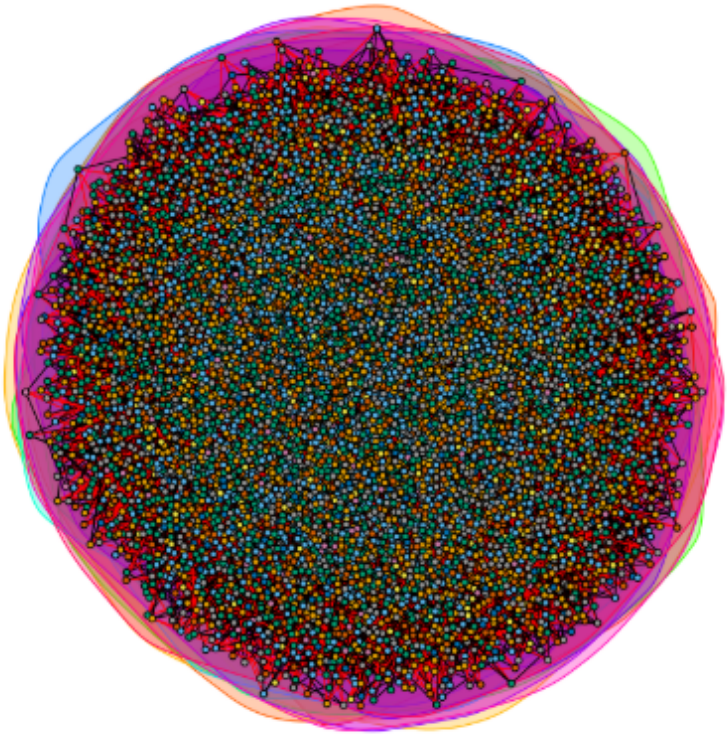

Figure 1.39. Community structure for undirected network with n = 10000, m = 5

The modularity is 0.275, which is also lower than m = 1 and m = 2.

(d) We plot the degree distribution in a log-log scale for n = 1000, 10000, m = 5, with linear regression used for estimating the slope of the plot:

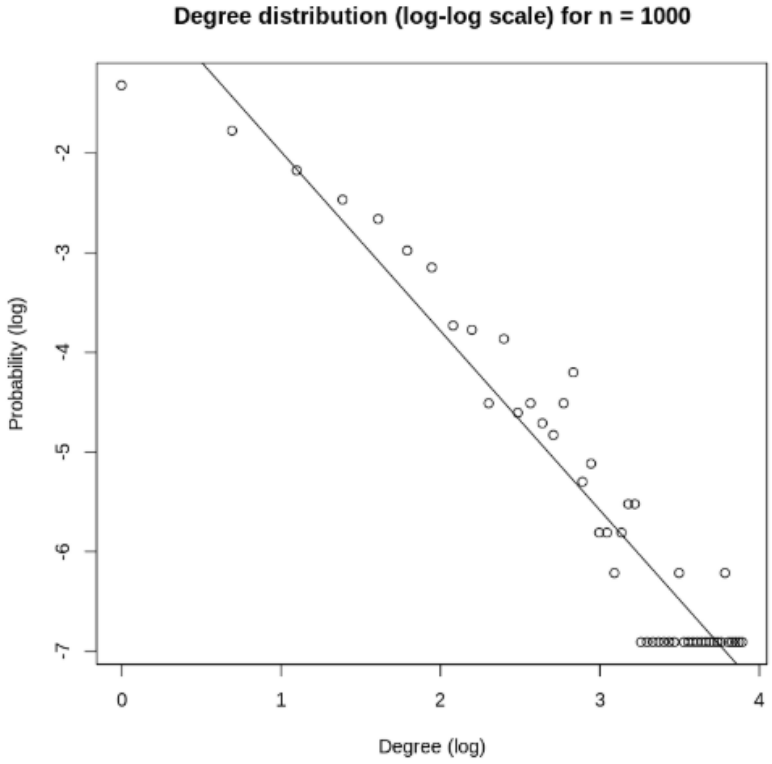

Figure 1.40. Degree distribution (log-log scale) for n = 1000, m = 5

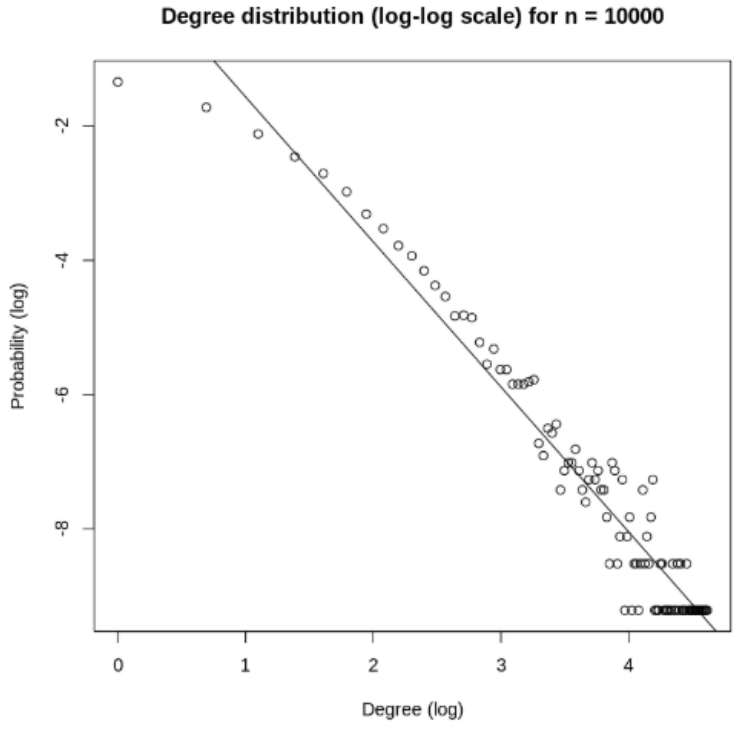

Figure 1.41. Degree distribution (log-log scale) for n = 10000, m = 5

By using linear regression, the estimation for the slope of the plot (n = 1000) is -1.801; the estimation for the slope of the plot (n = 10000) is -2.160. Compared with m = 1 and m = 2, the slope is slightly smaller in absolute value.

(e) We plot the degree distribution of nodes j that are picked with this process, in the log-log scale. For the network with n = 1000, m = 5:

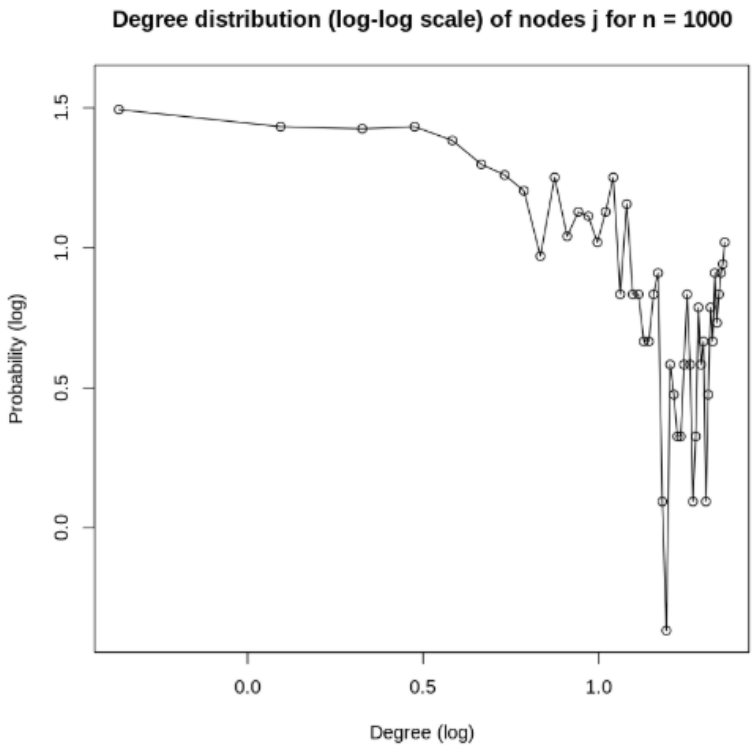

Figure 1.42. Degree distribution (log-log scale) of nodes j for n = 1000, m = 5

For the network with n = 10000, m = 5:

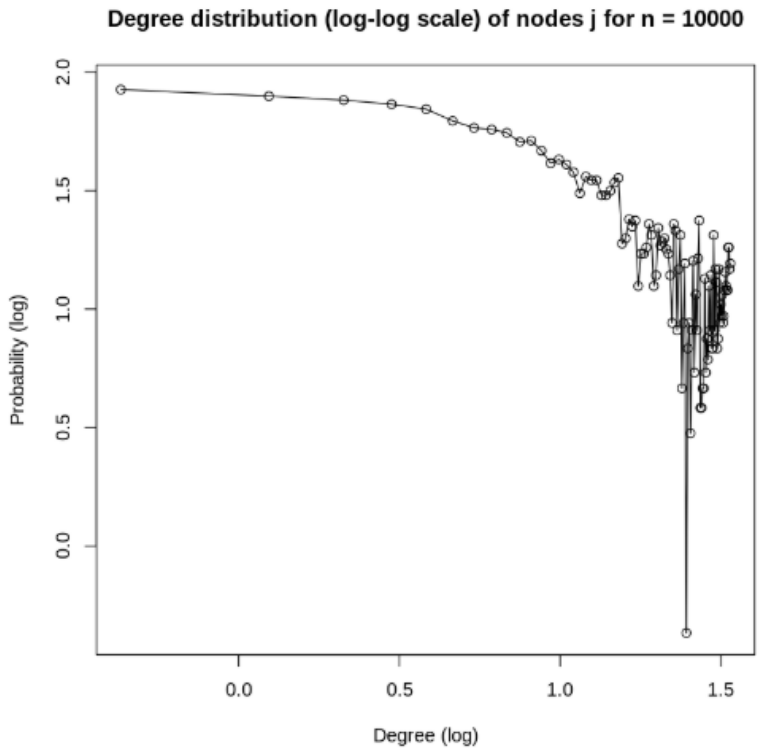

Figure 1.43. Degree distribution (log-log scale) of nodes j for n = 10000, m = 5

(f) The relationship between the age of nodes and their expected degree is plotted as below:

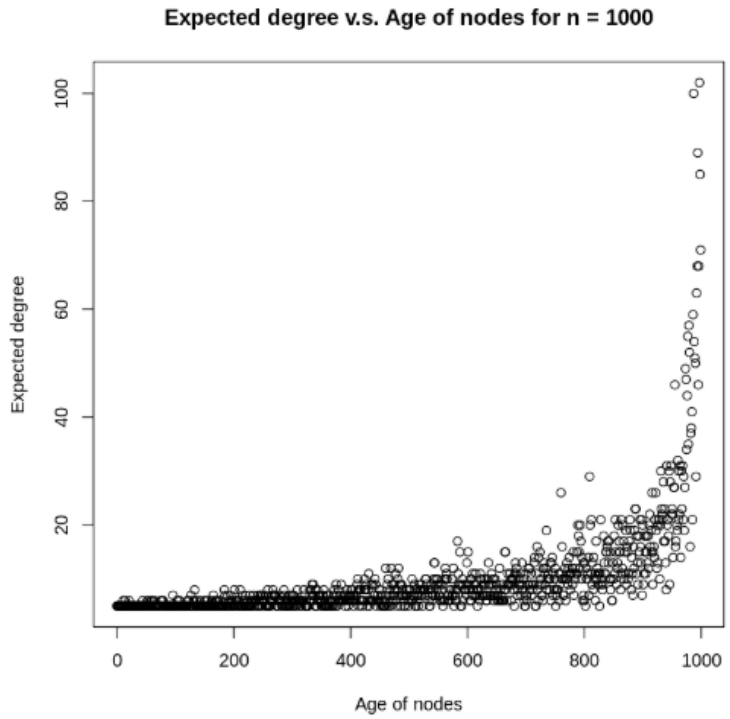

Figure 1.44. Expected degree v.s. Age of nodes (n = 1000, m = 5)

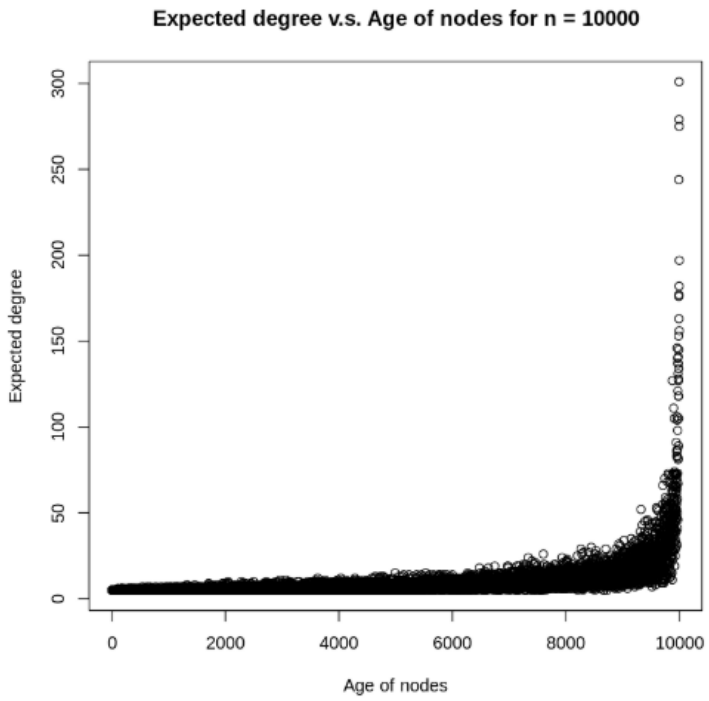

Figure 1.45. Expected degree v.s. Age of nodes (n = 10000, m = 5)

In the figures shown above, we can conclude that as the age of nodes increases, there is an exponential increase for the expected degree. Compared with m = 1 and m = 2, the increase speed is slower.
(g)
For m = 2:
For m = 5:

(h)First, we generate the original network, then we create a new network with the same degree sequence.
With modularity = 0.93:

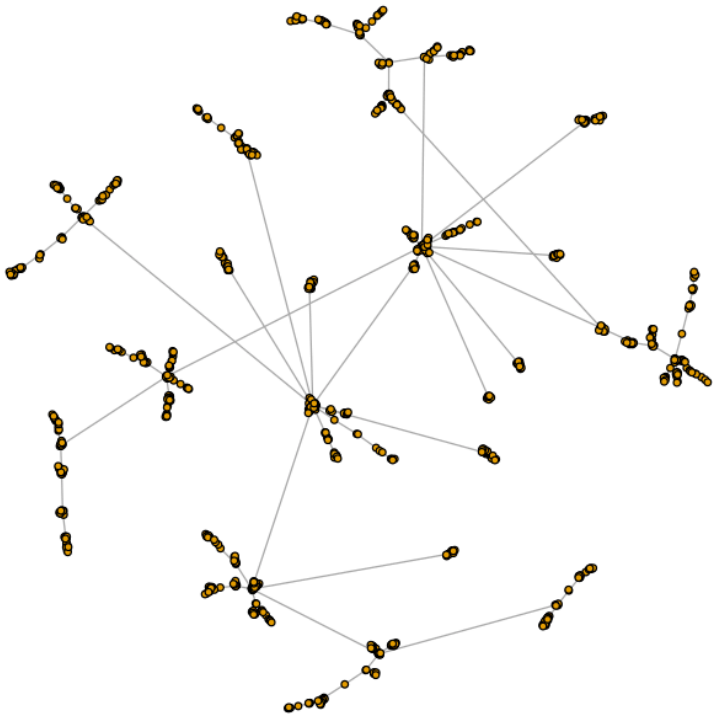

Figure 1.46. Original network (modularity = 0.93)

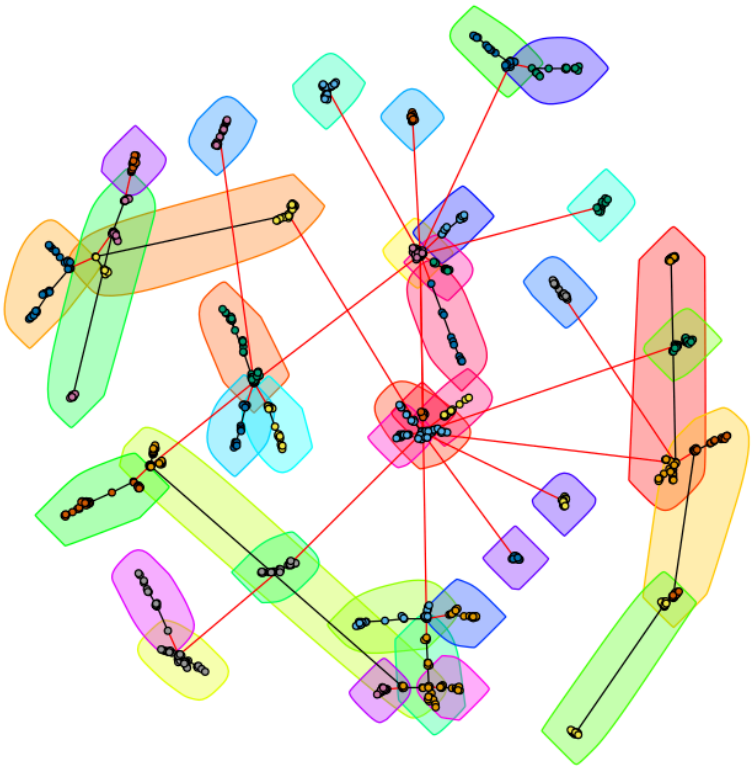

Figure 1.47. New network (modularity = 0.93)

With modularity = 0.85:

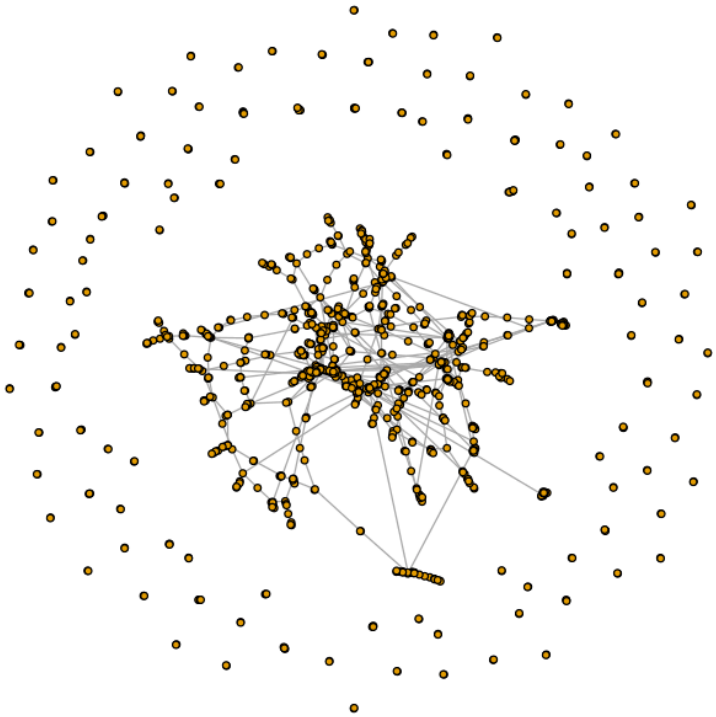

Figure 1.48. Original network (modularity = 0.85)

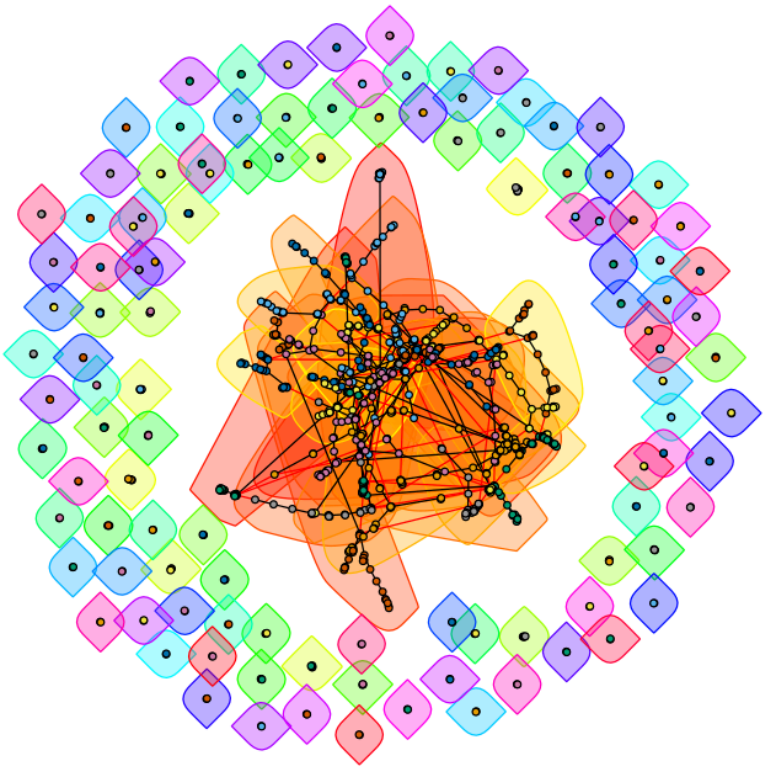

Figure 1.49. New network (modularity = 0.85)

Difference between the two procedures:
In the original, the graph was generated by adding vertices, then connect them to the graph. However,

in the stub-matching, the nodes are not always connected nodes. Since they are created upfront, and the matching stubs are used to connect the graph.

1.3 Create a modified preferential attachment model that penalizes the age of a node

(a)The network is generated with the parameters given by the instruction:

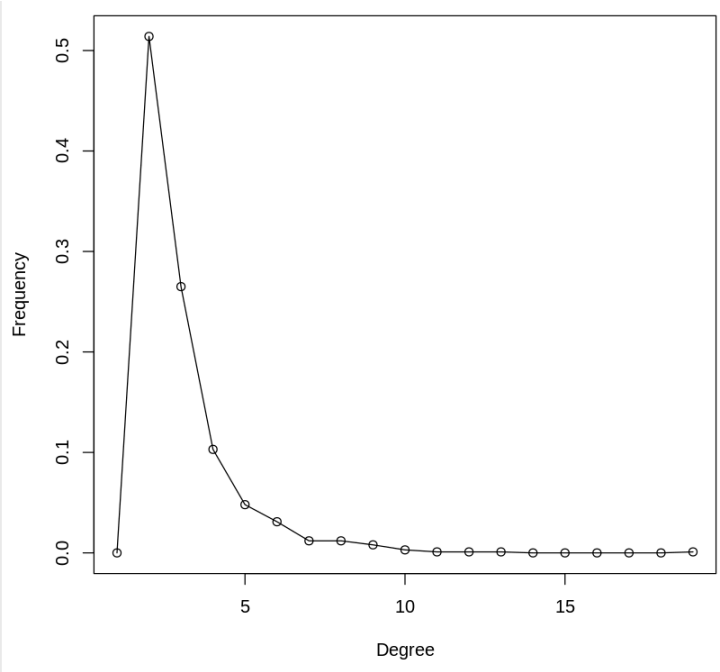

Figure 1.50. Degree v.s. Frequency (n = 1000)

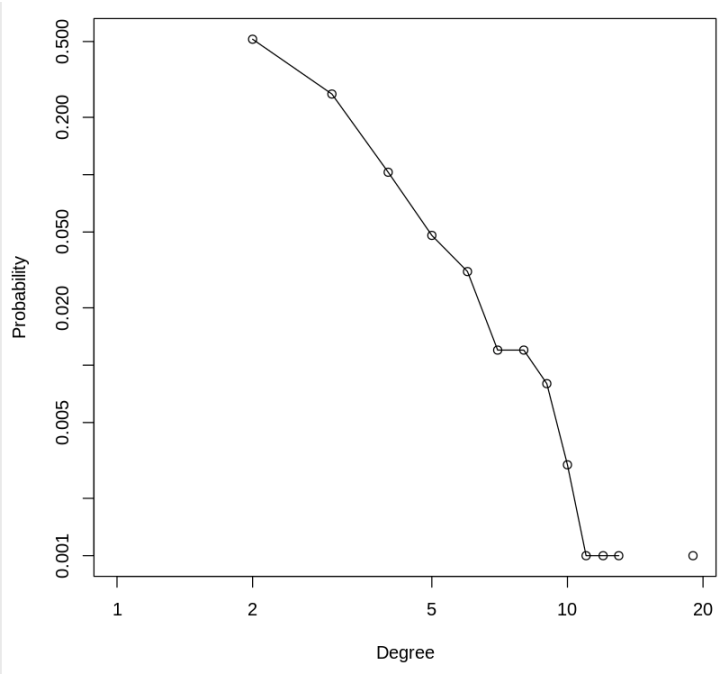

Figure 1.51. Degree v.s. Probability (n = 1000)

We can clearly see that as value of degree increase, the relative frequency vertices value degree logarithmically.
The power law exponent can be calculated by taking the average absolute slope of the log degree distribution, which is around 3.6.
(b)We use fast greedy method to find the community structure, with the modularity = 0.94:

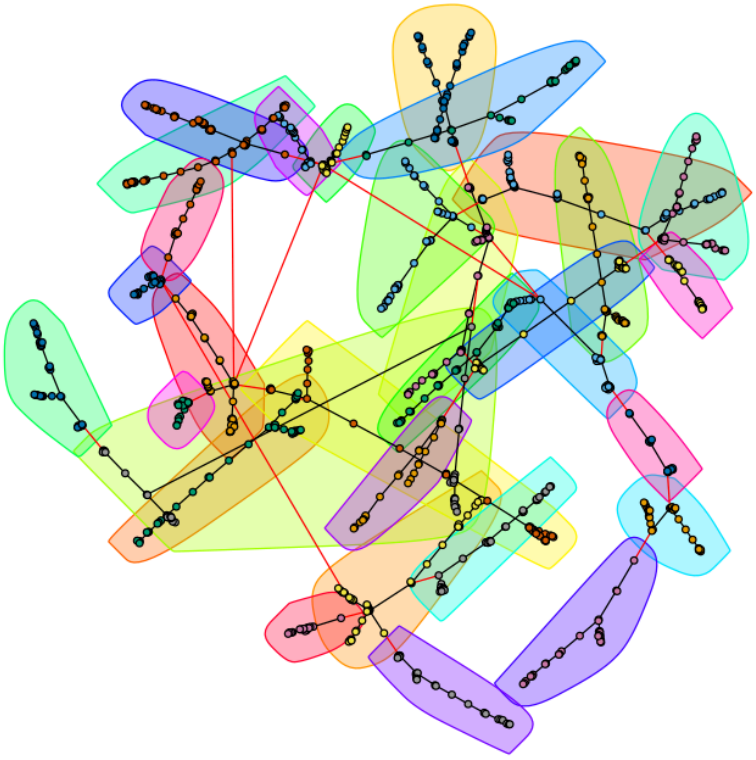

Figure 1.52. Community Structure

2 Part 2 : Random Walk on Networks

2.1 Random walk on ER networks

(a) We create an undirected random network with 1000 nodes, p = 0.01. The network is connected.

(b) We let a random walker start from a randomly selected node. We plot the average distance <s(t)> v.s. t and variance σ^2(t) v.s. t:

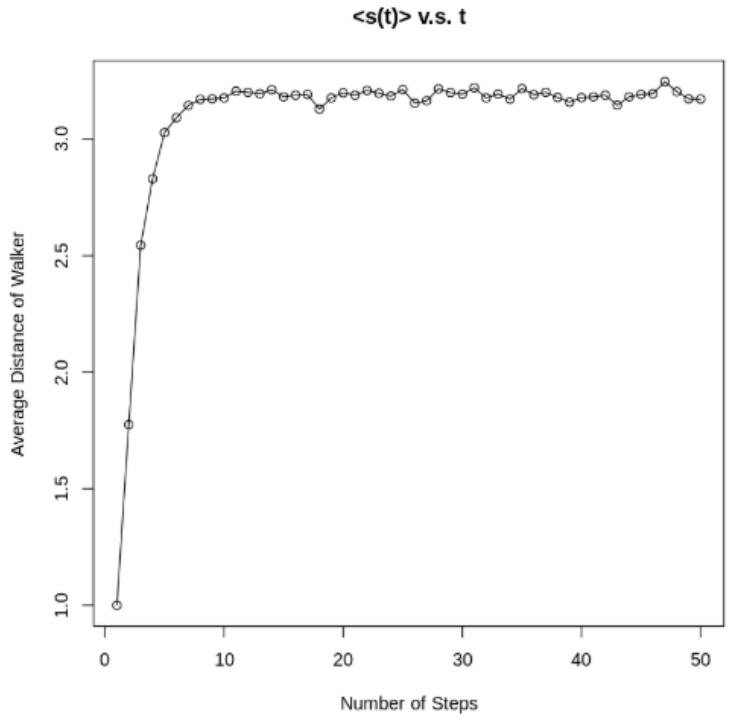

Figure 2.1. Average distance v.s. t

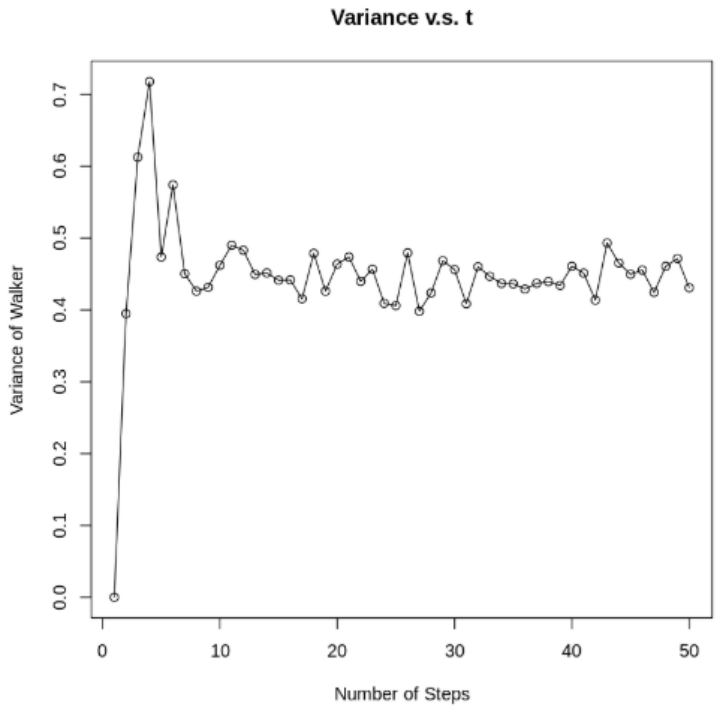

Figure 2.2. Variance v.s. t

(c) We plot the degree distribution of the original network and the nodes reached at the end of the random walk:

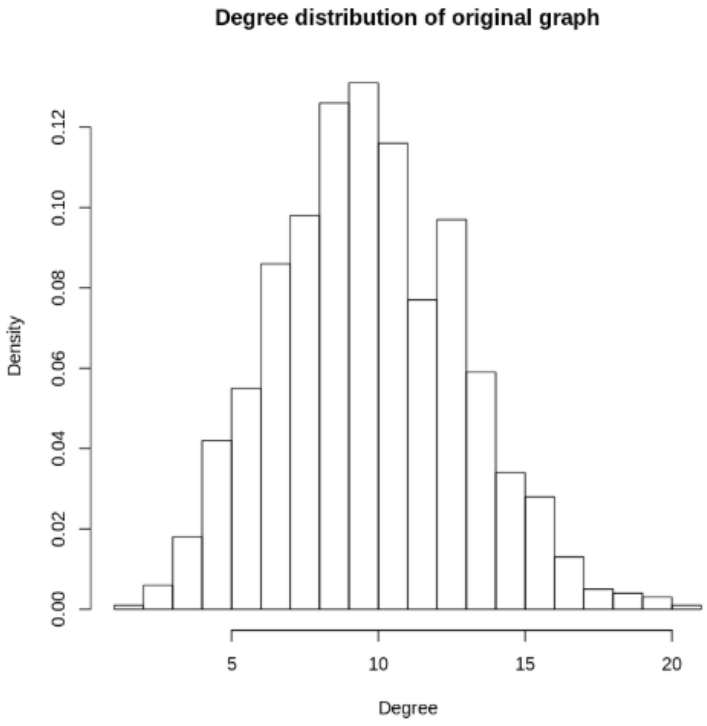

Figure 2.3. Degree distribution of original graph

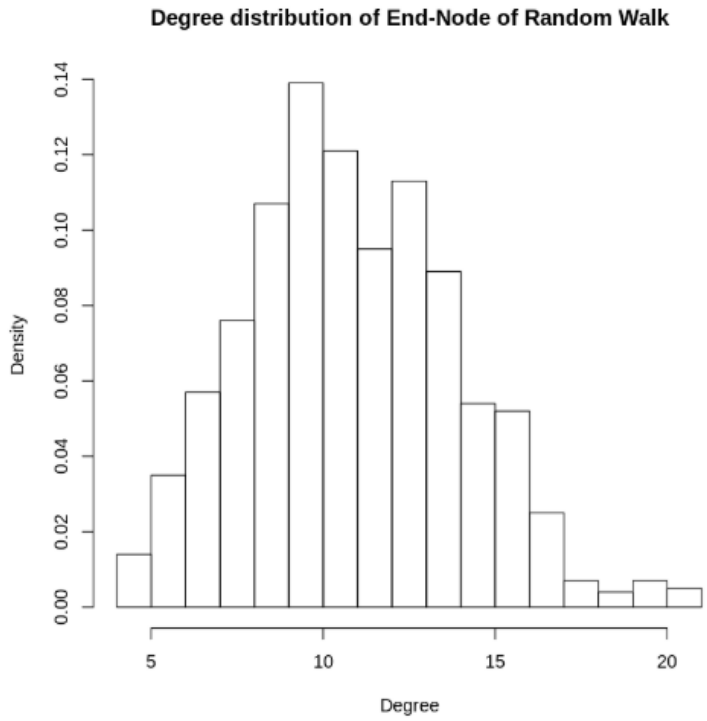

Figure 2.4. Degree distribution of End-Node of Random Walk

We measured the degree distribution of nodes which are reached the end of the random walk. Then we compare them to the degree distribution of the graph.
As we can see in the figures above, the degree distribution of nodes that reached at the end of random walk is similar to the degree distribution of graph, they are all obey binomial distribution.

(d) For the undirected random networks with 10000 nodes, we plot the average distance <s(t)> v.s. t and variance σ^2(t) v.s. t:

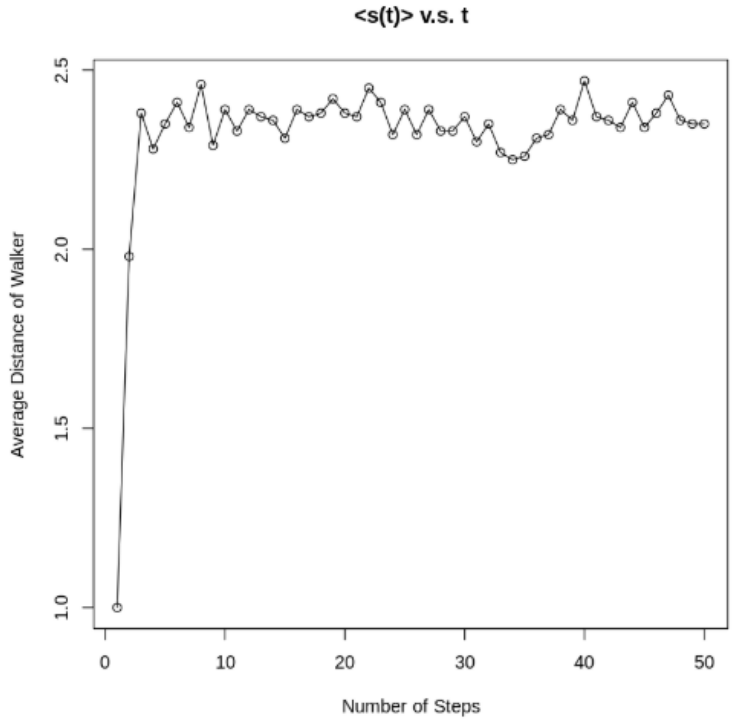

Figure 2.5. Average distance v.s. t

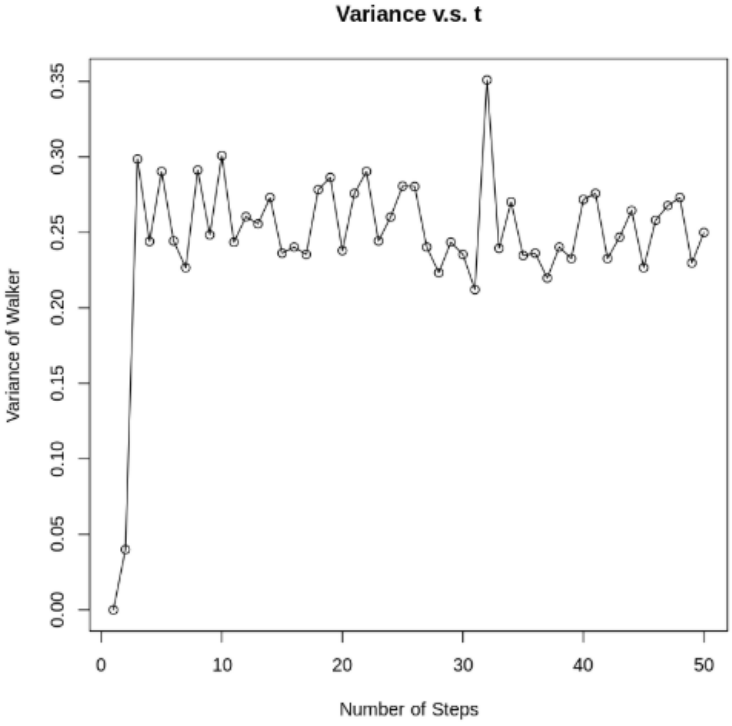

Figure 2.6. Variance v.s. t

From the figures shown above, the shortest distance reaches the steady point around 2.5, we can see it is smaller distance than with 1000 nodes, so as the variance measurement distance. Besides, we conclude that more nodes are added smaller graphs use more steps and larger variance in order to reach the steady point. However, since larger graph contains more degree distribution for random walk, large nodes reach

steady state with shorter distance as well as smaller variance, which is around 0.25.

2.2 Random walk on networks with fat-tailed degree distribution

(a) We generate an undirected preferential attachment network with 1000 nodes, where each new node attaches to m = 1 old nodes.

(b) We let a random walker start from a randomly selected node. We plot the average distance <s(t)> v.s. t and variance σ^2(t) v.s. t:

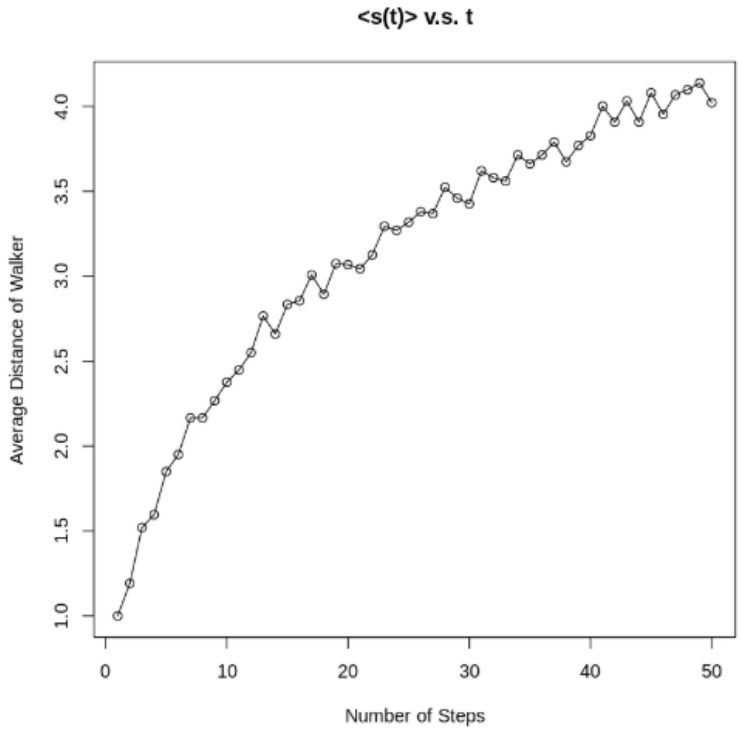

Figure 2.7. Average distance v.s. t

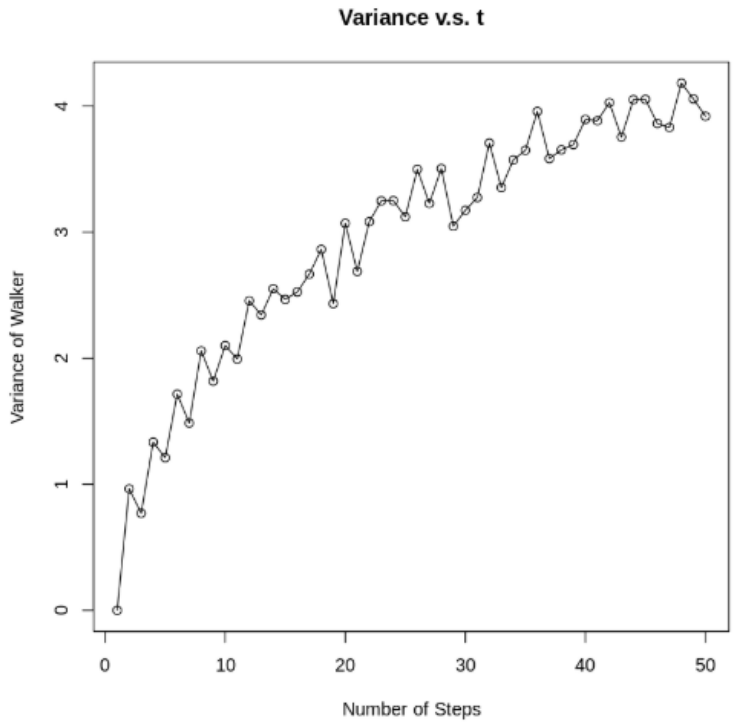

Figure 2.8. Variance v.s. t

(c) We plot the degree distribution of the original network and the nodes reached at the end of the random walk:

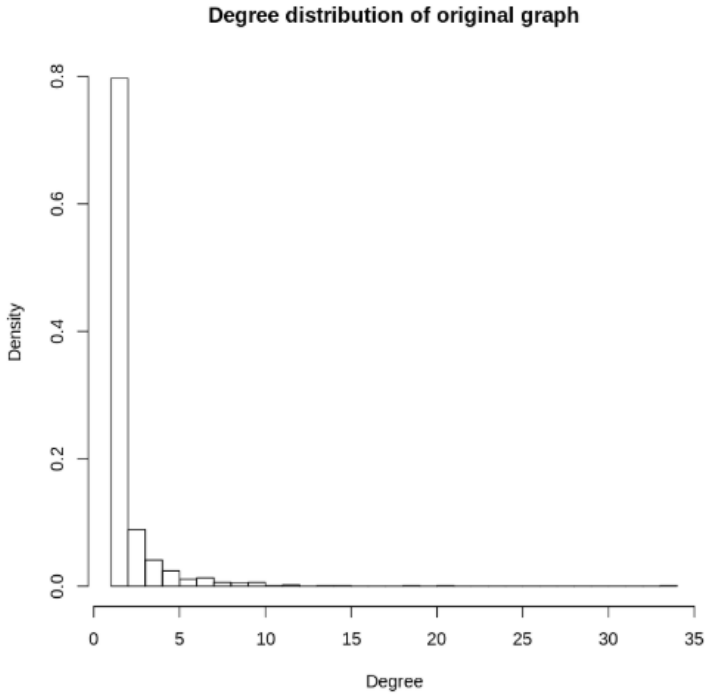

Figure 2.9. Degree distribution of original graph

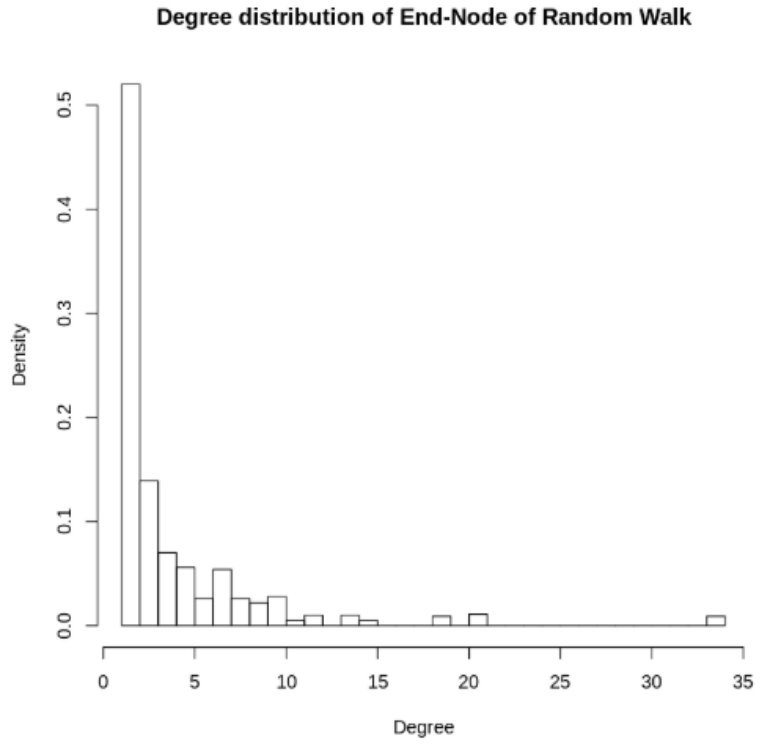

Figure 2.10. Degree distribution of End-Node of Random Walk

Comparison: From these two figures above, we can conclude that the network takes more steps to make the mean shortest path and variance to be stable as the number of nodes increase. And the graph with large diameter has less information containing its degree distribution, so it is reasonable that it requires more steps to reach a stable state.

(d) For the undirected random networks with 100 and 10000 nodes, we plot the average distance <s(t)> v.s. t and variance σ^2(t) v.s. t:
For n = 100, the diameter of GCC is 12:

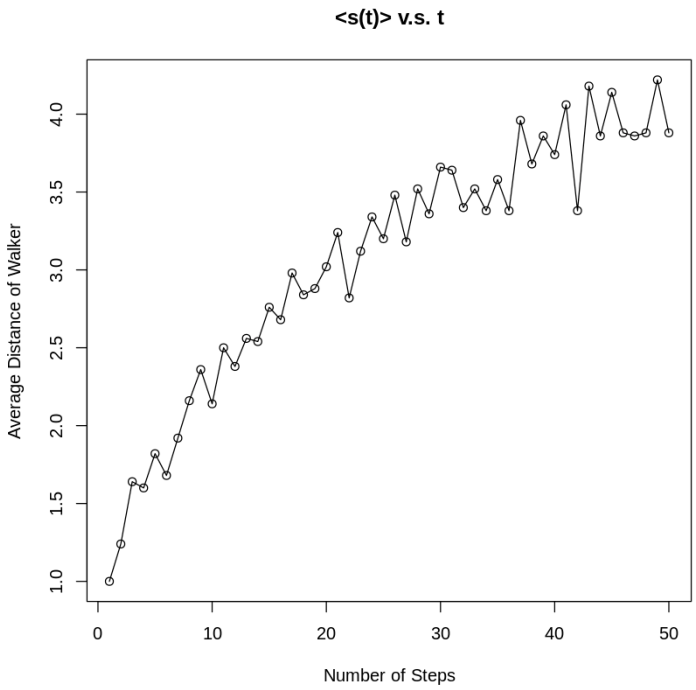

Figure 2.11. Average distance v.s. t, n = 100

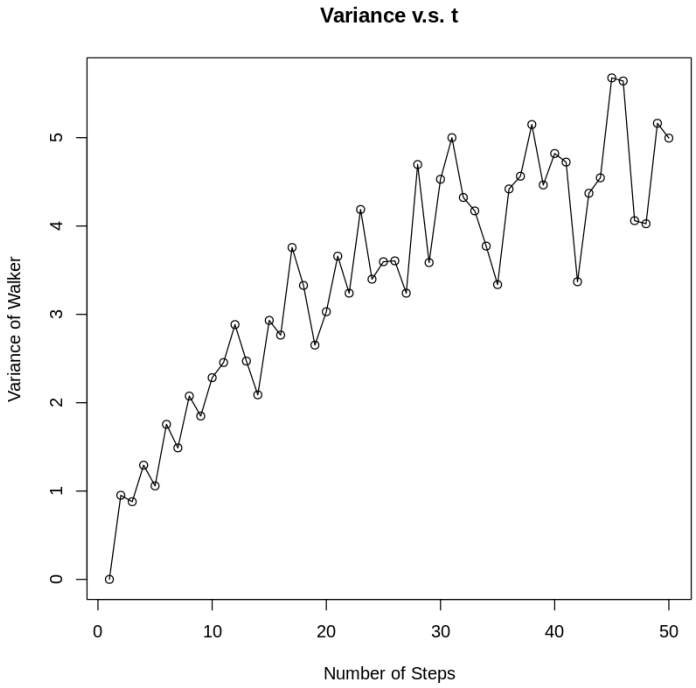

Figure 2.12. Variance v.s. t, n = 100

For n = 10000, the diameter of GCC is 28:

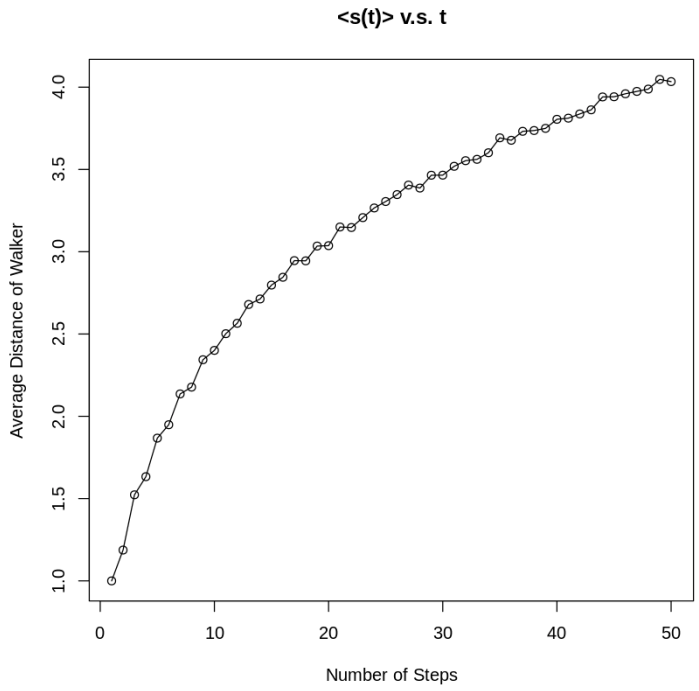

Figure 2.13. Average distance v.s. t, n = 10000

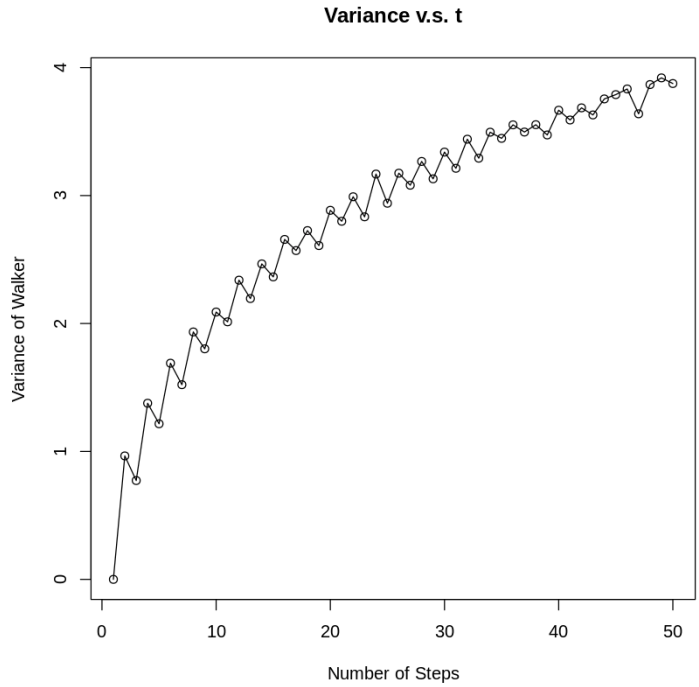

Figure 2.14. Variance v.s. t, n = 10000

We use n = 100 and n = 10000 respectively. Since the network follows the power law distribution. From the figures we shown above, 100 nodes network use less steps to reach steady point. Thus, the degree distribution of the network is in fact display fat-tailed degree distribution. However, network with 10000 nodes use more steps to reach steady point for mean shortest distance and variance.
In conclusion, large diameter graph has less state. The network use more steps to make the mean shortest path and variance to be stabilized, as the number of nodes increases.

2.3 PageRank

(a) We create a directed random network with 1000 nodes, m = 4, using the preferential attachment model. Random walk is run for 1000 iterations and 50 time steps. We measure the probability that the walker visits each node:

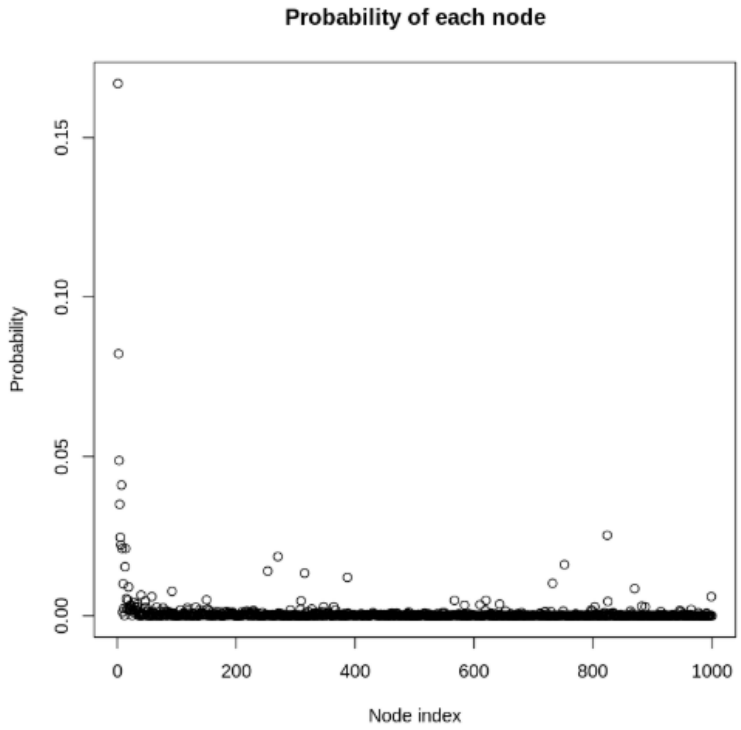

Figure 2.15. Probability of each node

We also plot the probability v.s. graph in-degree:

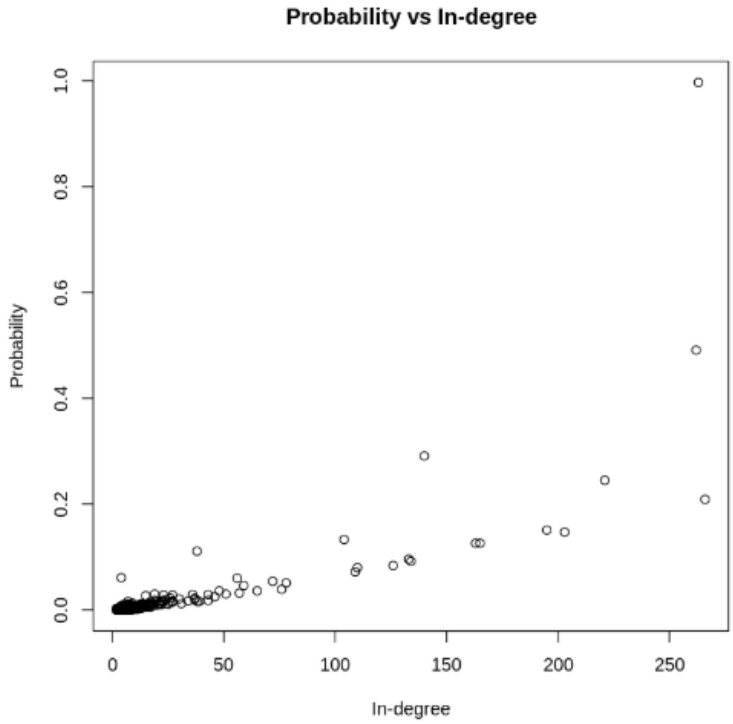

Figure 2.16. Probability v.s. in-degree

From the plot we can see that the probability is related to the in-degree. With higher in-degree the probability is higher, which make sense as the probability to move to ”popular nodes”, which have high in-degree should be higher.

(b) We use a teleportation probability of $\alpha = 0.15$. We perform random walks on the network created in 3(a) and measure the probability that the walker visits each node:

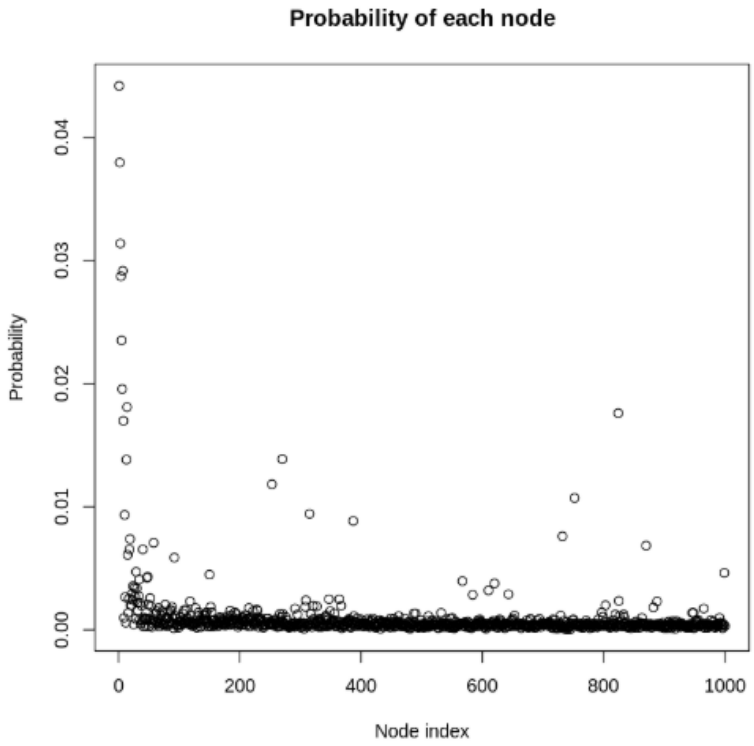

Figure 2.17. Probability of each node

We also plot the probability v.s. graph ln-degree:

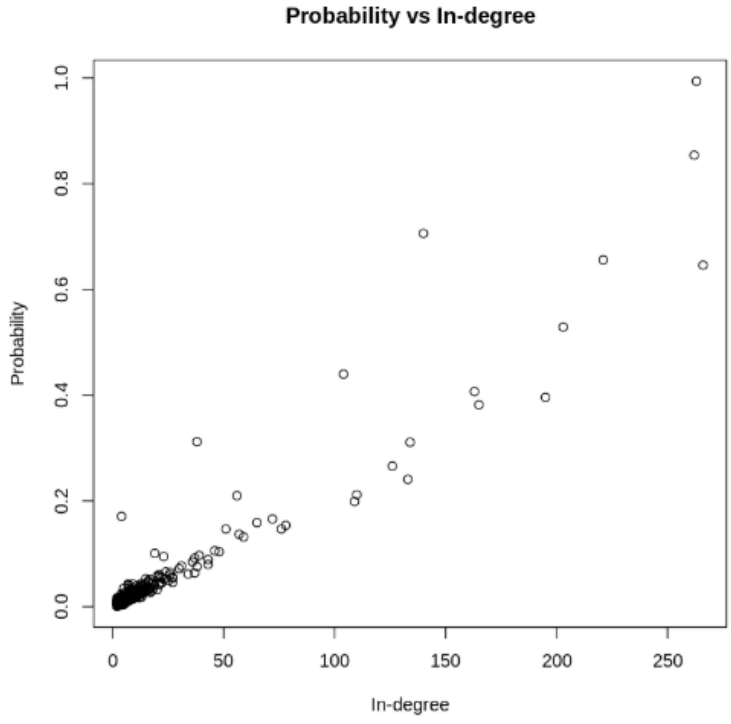

Figure 2.18. Probability v.s. ln-degree

Still, we can see that the probability is related to the in-degree. This is because there are only 15 percent

chance for the random teleportation (which is independent of the node in-degree) to happen.

2.4 Personalized PageRank

(a) We use random walk on network generated in question 3 to simulate a personalized PageRank, where the teleportation probability to each node is proportional to its PageRank. The teleportation probability is $\alpha = 0.15$. We measure the probability that the walker visits each node:

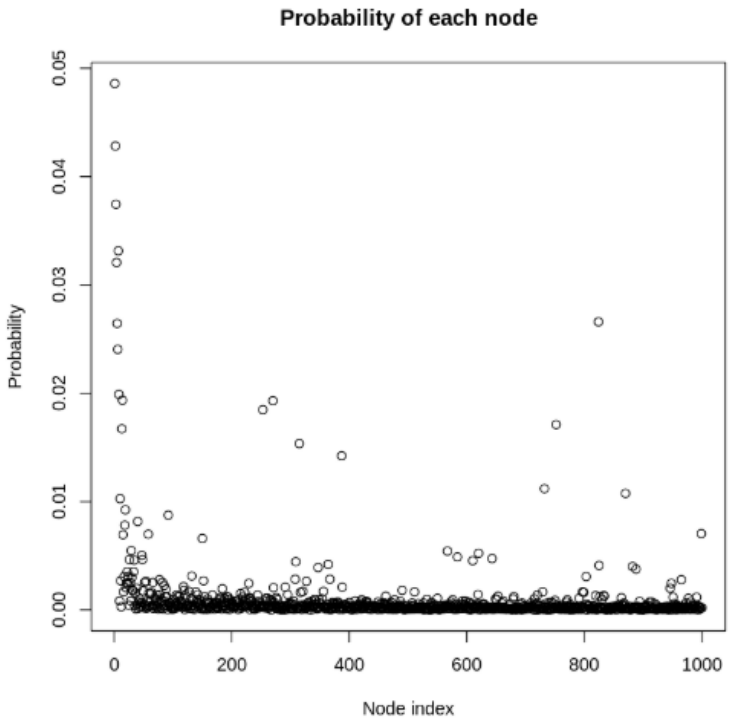

Figure 2.19. Probability of each node

We can see the result is similar to the result in question 3, which makes sense as the chance of visiting nodes with larger PageRank will be larger too.

We also plot the probability v.s. graph in-degree:

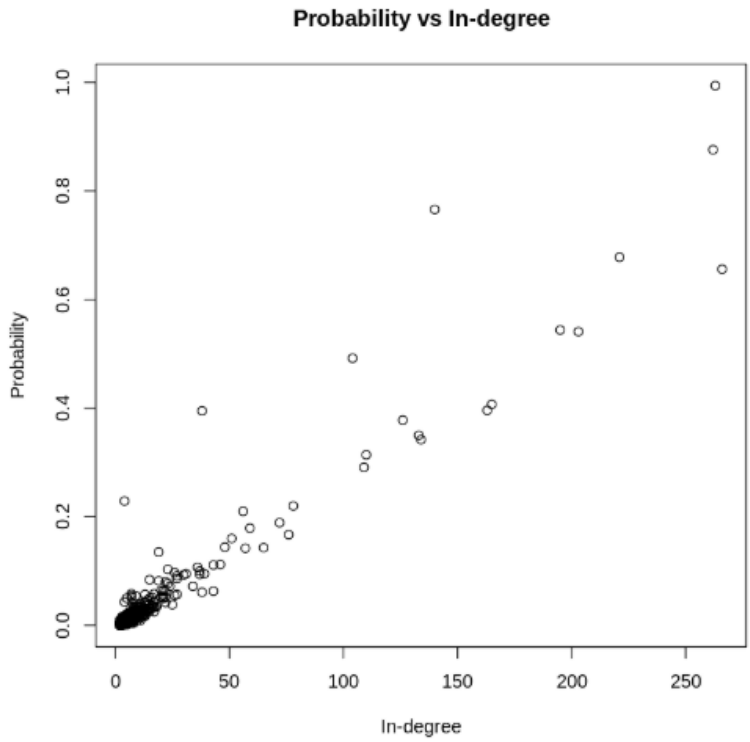

Figure 2.20. Probability v.s. In-degree

We can see that the probability is related to the in-degree. Compared with results in question 3, the probability is more affected when the nodes have higher degree.

(b) We find two nodes in the network with median PageRanks: node 500 and node 501, and repeat part 4(a):

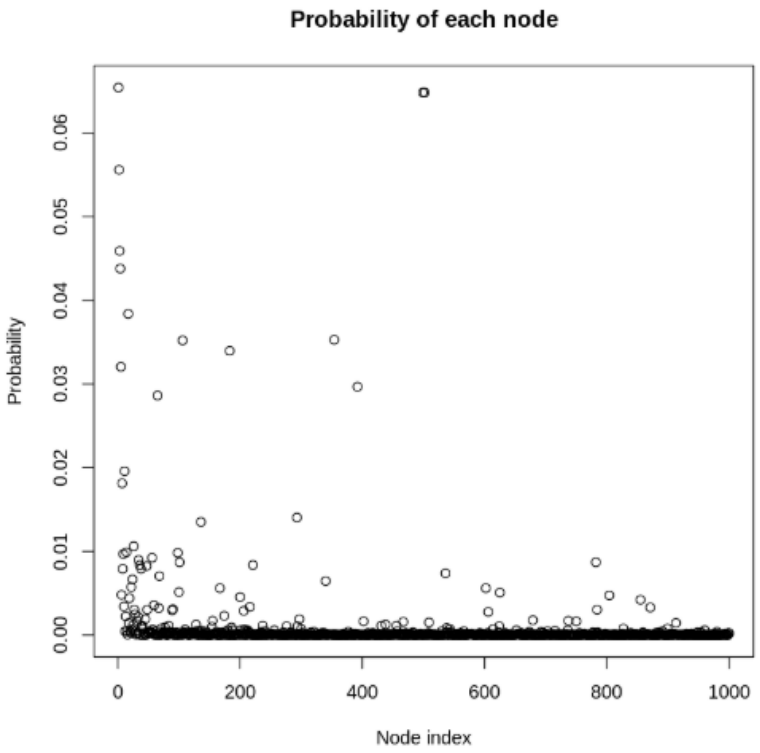

Figure 2.21. Probability of each node

Compared with (a), the probability of visiting the node 500 and 501 is obviously enlarged, as the teleportation happens only on these two nodes.

(c) The PageRank equation can be rewritten as below, where α is the teleportation probability, T is the transition matrix. Pref refers to the matrix of people's interest in different nodes, which should be a normalized vector.

$$PR = (1 - \alpha) * PR * T + \alpha * PR * Pref$$