Graph and Network
Homework 2

Mansee Jadhav - 204567818
Ronak Sumbaly - 604591897

April 20, 2016

Ques 1. Random Walker - Random Networks

Part (a): Creation of Undirected Random Network

A random undirected network was created using **random.graph.game** method of **iGraph** package. The network consists of 1000 nodes with the probability p for drawing an edge between any pair of nodes equal to 0.01.

Part (b): Random Walker from Random Node

The **netrw** package was used to simulate a random walker. The above constructed network was used to select a node at random and randomly walk (1000 walks) by varying **number of steps (t)** ranging from $1 - 50$. The average distance of the walker from the starting vertex to tail vertex at step t is calculated. At the same time we also measure the average standard deviation of this distance. The relationship between the observed values and number of steps is shown below

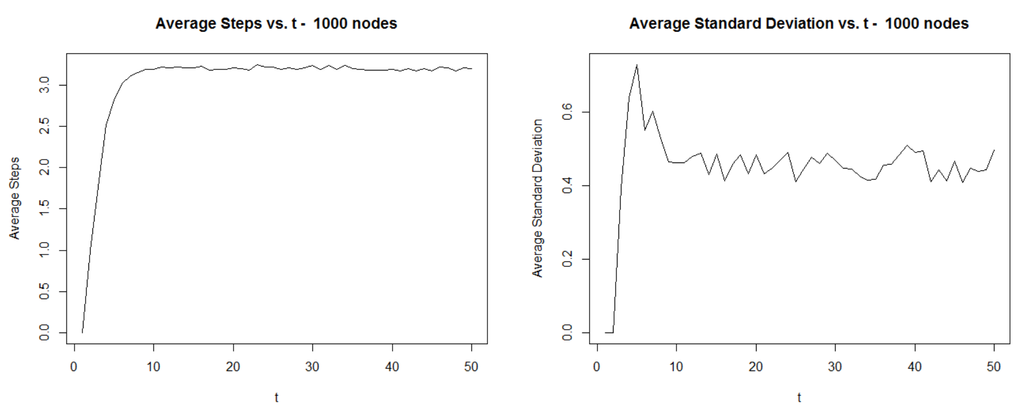

Figure 1: Average Distance & Standard Deviation vs. Number of Steps

Part (c): Comparing random network in D dimensional

A random walker in *D dimensional* has an average distance equal to zero and $\sqrt{\texttt{standard deviation}}$ proportional to $\sqrt{\texttt{number of steps (t)}}$. In D dimensional space the distance can be negative, hence they cancel out positive distances. However in our random graph, all values are positive hence the average distance and average standard deviation converge to 3.0 and 0.4 respectively rather than 0.0.

Part (d): Random networks with 100 and 10000 nodes

By varying the number of nodes to 100 and 10000 we obtained the results of **average number of steps and average standard deviation**, as in Part (b). The graphs obtained are as follows

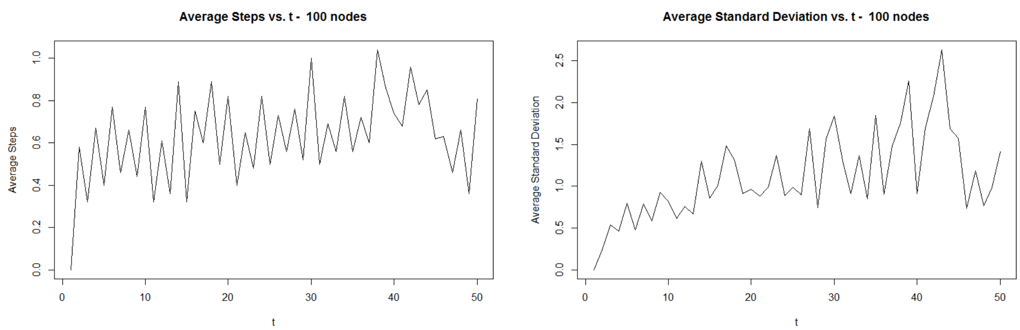

Figure 2: Average Distance & Standard Deviation vs. Number of Steps - Nodes = 100

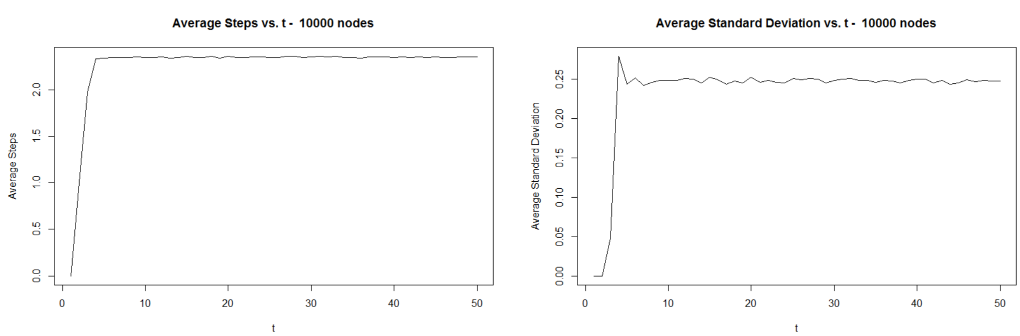

Figure 3: Average Distance & Standard Deviation vs. Number of Steps - Nodes = 10000

The diameters of the networks created were as follows,

node (n)	Diameter (d)
1000	5
100	6
10000	3

Table 1: Diameters of Random Network

It can be seen that for a **smaller diameter** the average distance and average standard deviation **converges at a faster rate** than with a larger diameter. It can also be seen the distribution for each step size for smaller diameter is more confined, indicating the standard deviation will be smaller.

Part (e): Degree distribution of nodes - End of Random Walk

The degree distribution of the nodes reached till the end of the network of 1000 nodes are compared with the degree distribution of the random network. It can be seen that there's not much of difference in the two graphs. The graph obtained were as follows:

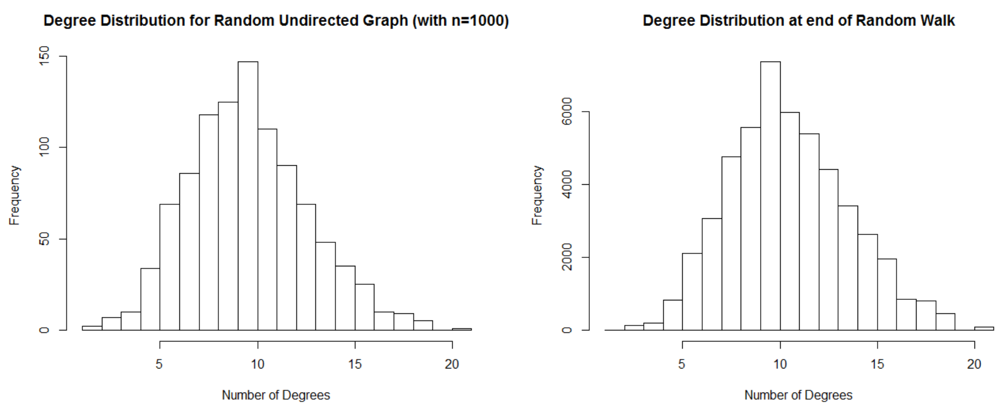

Figure 4: Degree Distribution of Random Network & Nodes at End of Random Walk

Ques 2. Random Walker - Fat-Tailed Degree Distribution

Part (a): Fat - Tailed Distribution Network

The degree distribution resulting from **BA model (barabasi.game())** is scale free, it is power law of form P(k) $\sim k^{-3}$. We created undirected network with 1000 nodes whose degree distribution is proportional to k^{-3} using BA model.

Part (b): Random Walker from Random Node

Employing the same methodology as applied in Ques. 1 we used the **netrw** package to construct a random walker over 1000 walks **varying the step size (t)**. The average distance of the walker from the starting vertex to tail vertex at step t is calculated. At the same time we also measure the average standard deviation of this distance. The relationship between the observed values and number of steps is shown below

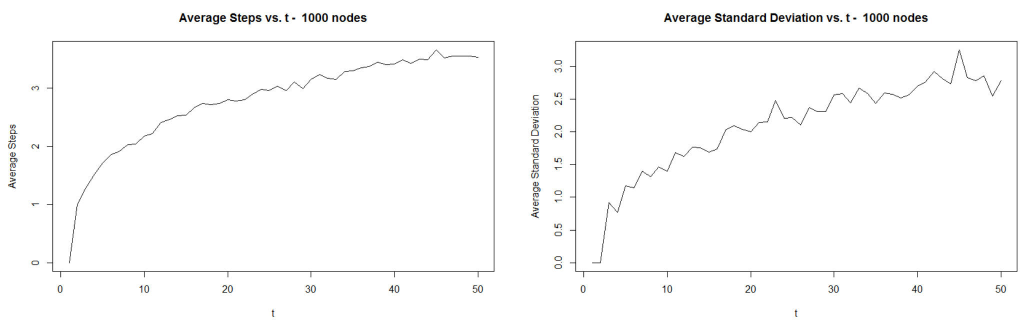

Figure 5: Average Distance & Standard Deviation vs. Number of Steps

Part (c): Comparing Random Network in D dimensional

A random walker in *D dimensional* has an average distance equal to zero and $\sqrt{\texttt{standard deviation}}$ proportional to $\sqrt{\texttt{number of steps (t)}}$. In D dimensional space the distance can be negative, hence they cancel out positive distances. However in our random graph, all values are positive hence the average distance and average standard deviation converge to 3.0 rather than 0.0.

Part (d): Random Networks with 100 and 10000 nodes

By varying the number of nodes to 100 and 10000 we obtained the results of **average number of steps and average standard deviation**, as in Part (b). The graphs obtained are as follows

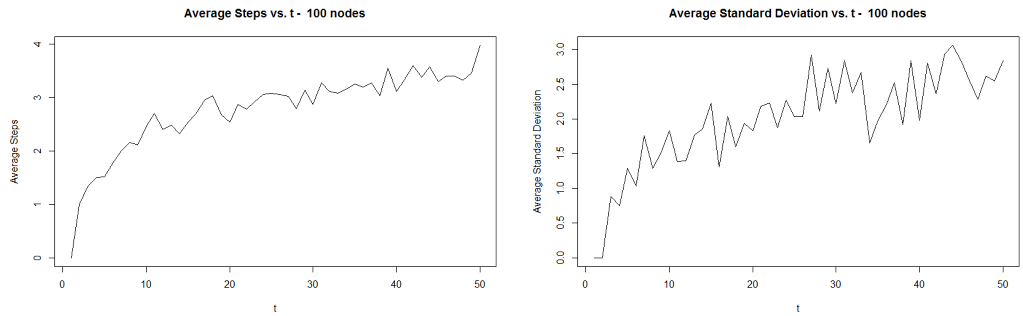

Figure 6: Average Distance & Standard Deviation vs. Number of Steps - Nodes = 100

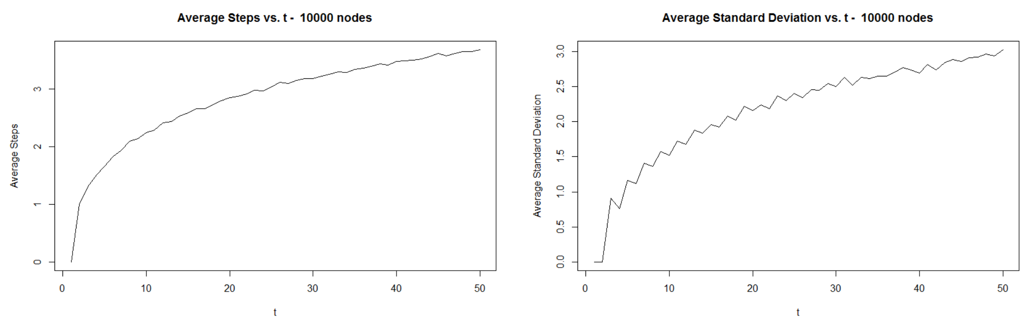

Figure 7: Average Distance & Standard Deviation vs. Number of Steps - Nodes = 10000

The diameter of the random networks were recorded as follows,

node (n)	**Diameter (d)**
1000	18
100	10
10000	24

Table 2: Diameters of Random Network

It can be clearly seen that are the **diameter increases** of the network the time for the distribution to **converge is slower**. **Figure 6** shows that the distribution converges at a much faster rather than **Figure 7** since the diameter has increased two-folds.

Part (e): Degree distribution of nodes - End of Random Walk

The degree distribution of the nodes reached till the end of the network of 1000 nodes are compared with the degree distribution of the random network. It can be seen that there's not much of difference in the two graphs. The graph obtained were as follows:

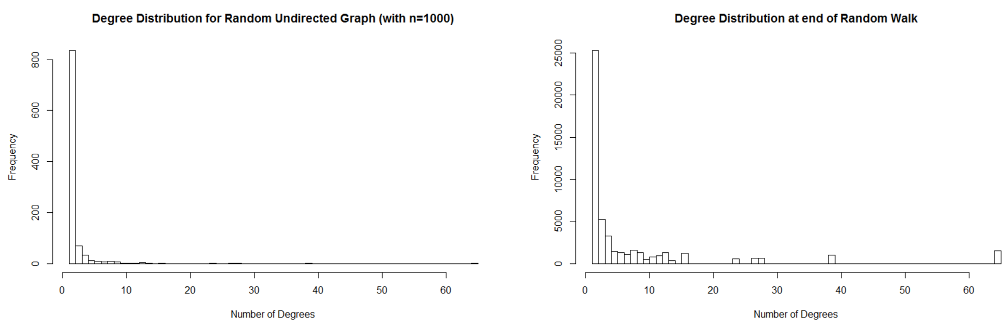

Figure 8: Degree Distribution of Random Network & Nodes at End of Random Walk

Ques 3. PageRank

Part (a): Prob. of Walker Visiting Each Node

Again the **netrw** package was used to simulate a random walker. The above constructed network was used to select on node at random and randomly walk called vertex sequence by varying **number of steps (t)** ranging from $1 - 1000$. We calculated the degree of each node in the graph and then the probability of visiting each node within vertex sequence by using **vertex_sequence$ave.visit.prob**. The correlation between the number of degree and visit probability was calculated as well,

Correlation between # of Degree and Visit Probability: 0.9942095

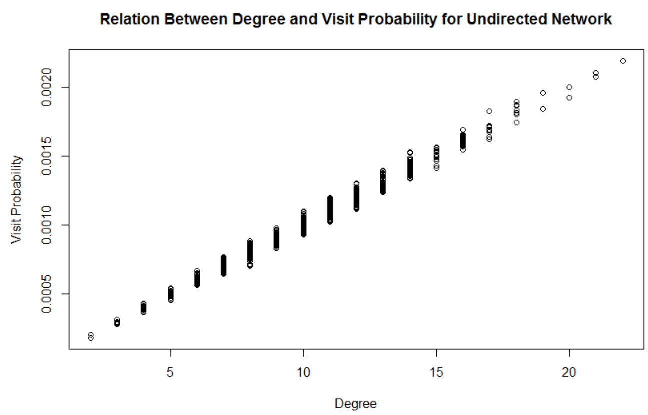

Figure 9: # of Degree vs Visit Probability - Undirected Network

As seen from the above graph and the correlation value, the degree of nodes and their visit probability are linearly dependent as they have high a correlation value.

Part (b): Prob. of Walker Visiting Each Node - Directed Graph

The **netrw** package was used to simulate a random walker. We created a **directed network** of 1000 nodes, where the probability p for drawing an edge between any pair of nodes is 0.01. We calculated the degree of each node in the graph and then the probability of visiting each node within vertex sequence by using **vertex_sequence$ave.visit.prob**

Visit Probability of Nodes	Correlation
total-degree	0.6169737
in-degree	0.884234
out-degree	-0.009215413

Table 3: Correlation between nodes of network and their visit probability

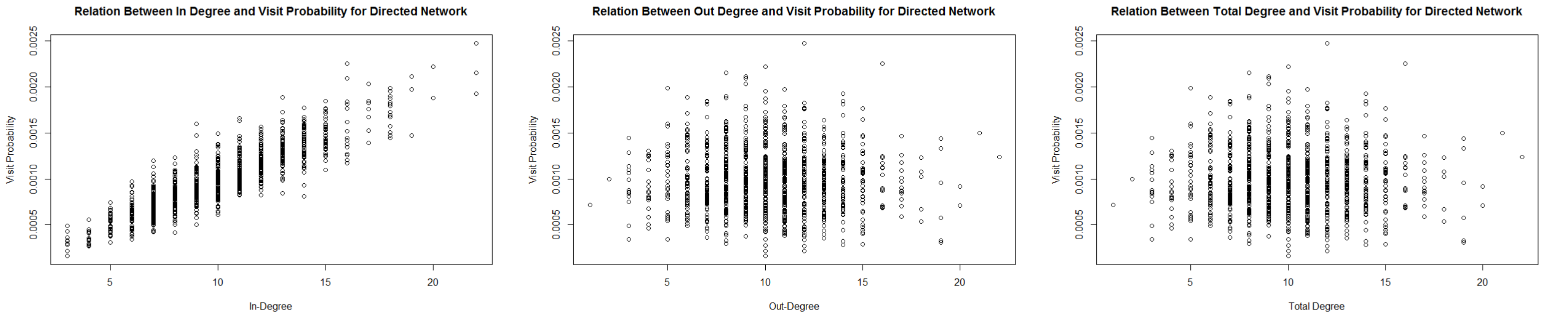

Figure 10: # of Degree vs Visit Probability - Directed Network

We can see that **in-degrees of nodes are highly correlated** with the visit probability of the nodes, as more the number of incoming links the node has, more is the visit probability of the node. The inverse relationship can be seen for the out-degree.

Part (c): Teleportation

Previous sections considered damping factor (d = 1). In this section we introduced a **damping factor d = 0.85** for an undirected random network. We calculated the degree of each node in this graph and then the probability of visiting each node within vertex sequence by using **vertex_sequence$ave.visit.prob**. The correlation between the number of degree and visit probability was calculated as well,

Correlation between # of Degree and Visit Probability: 0.9889963

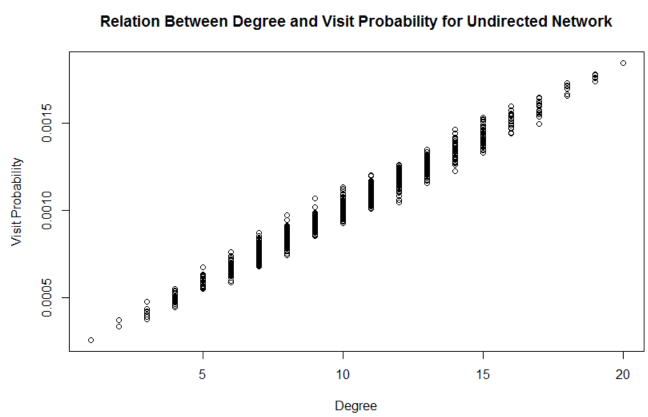

Figure 11: # of Degree vs Visit Probability - Undirected Network with d=0.85

As seen from the above graph and the correlation value, the **degree of nodes and their visit probability** are **linearly dependent** as they have high a correlation value.

Ques 4. Personalized PageRank

Part (a): Simulate PageRank of Nodes

A **directed random network** was created with **1000 nodes** with the probability p for drawing an edge between any pair of nodes is 0.01 and **damping factor (d = 0.85)**. We simulated the page rank of the network by plotting the visit probability of each node in the graph.

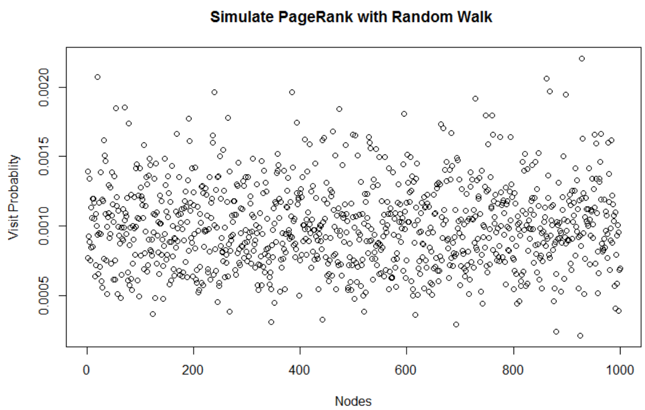

Figure 12: Simulated PageRank - Visit Probabilities

Part (b): Personalized PageRank - Teleportation Probability Proportional to Page Rank

In order to create a **Personalized Pagerank**, a random directed network was generated with a damping factor of 0.85. The visit probabilities of this network was used as teleportation probability to create a new directed network. The purpose of this is to make the page-rank for each node proportional to teleportation probability instead of it being equal to $\frac{1}{N}$.

Correlation between page-rank and page-rank (with teleportation): 0.9903303

The graph below shows the simulated personalized page rank with its visit probabilities.

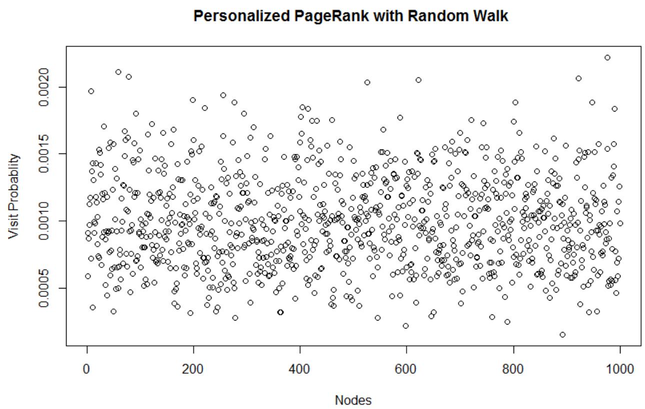

Figure 13: Personalized PageRank - Visit Probabilities

Comparing the results to Part (a), we can see that the graph of visit probabilities is more concentrated with this approach. Thus we can conclude that nodes which have large people interest based on teleportation have large visit probabilities and consequently larger page rank.

Part (c): Modification of PageRank equation by using Self-Enforcement

In order to incorporate the notion of self importance into the PageRank of each node we consider the PageRank equation,

$$\mathtt{Pr}(A) = \frac{(1-d)}{N} + d \sum_{T_{in}} \frac{\mathtt{Pr}(T_{in})}{C(T_{in})}$$

To take into account the effect of this self-enforcement we need to change the **teleportation probability** from $\frac{1}{N}$ to a number which is proportional to PageRank.