1.

(a)

g <- random.graph.game(1000,0.01,directed=F)

(b)

|  |
| --- |
| **[Code]** |
| g <- random.graph.game(1000,0.01,directed=F)  avg\_dist <- c(0)  var\_dist <- c(0)  std\_dist <- c(0)  num\_walker <- 100  shortest\_paths <- shortest.paths(g)  for (i in 1:100) {  rw <- netrw(g, num\_walker, start.node = sample(0:(vcount(g) - 1) + as.integer((sessionInfo())$otherPkgs$igraph$Version >= "0.6"), num\_walker, replace = TRUE), damping = 1, weights = NULL, T = i, seed = NULL, output.walk.path = TRUE, output.walkers = 0:( num\_walker - 1), output.visit.prob = TRUE, output.nodes = 0:(vcount(g) - 1), output.device = "memory", walk.path.file = "walk\_path.txt", visit.prob.file = "visit\_prob.txt", local.pagerank = FALSE, teleport.prob = NULL)    tmp\_dist <- c(0)  for (j in 1:num\_walker) {  tmp\_dist[j] <- shortest\_paths[rw[[2]][1,j], rw[[2]][i,j]]+1  }    avg\_dist[i] <- mean(tmp\_dist)  var\_dist[i] <- mean((tmp\_dist-mean(tmp\_dist))^2) # Variance  std\_dist[i] <- sqrt(mean((tmp\_dist-mean(tmp\_dist))^2)) # Standard Deviation  }  plot(avg\_dist)  plot(var\_dist)  plot(std\_dist) |

|  |  |
| --- | --- |
| **[Graph]** | |
| Average |  |
| Variance |  |
| Standard  Deviation |  |

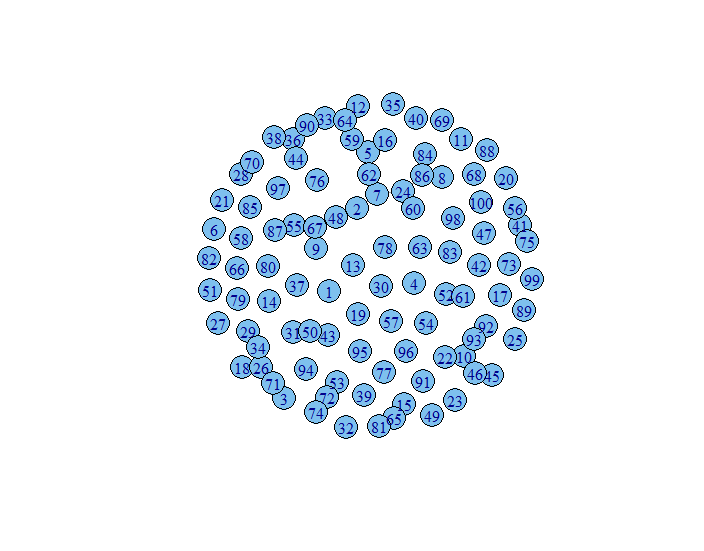
(c)

(d)

n=100

diameter = 13

|  |
| --- |
| **[n=100]** |
| g <- random.graph.game(100,0.01,directed=F)  d <- diameter(g)  avg\_dist <- c(0)  var\_dist <- c(0)  std\_dist <- c(0)  num\_walker <- 100  shortest\_paths <- shortest.paths(g)  for (i in 1:100) {  rw <- netrw(g, num\_walker, start.node = sample(0:(vcount(g) - 1) + as.integer((sessionInfo())$otherPkgs$igraph$Version >= "0.6"), num\_walker, replace = TRUE), damping = 1, weights = NULL, T = i, seed = NULL, output.walk.path = TRUE, output.walkers = 0:( num\_walker - 1), output.visit.prob = TRUE, output.nodes = 0:(vcount(g) - 1), output.device = "memory", walk.path.file = "walk\_path.txt", visit.prob.file = "visit\_prob.txt", local.pagerank = FALSE, teleport.prob = NULL)    tmp\_dist <- c(0)  for (j in 1:num\_walker) {  tmp\_dist[j] <- shortest\_paths[rw[[2]][1,j], rw[[2]][i,j]]+1  }    avg\_dist[i] <- mean(tmp\_dist)  var\_dist[i] <- mean((tmp\_dist-mean(tmp\_dist))^2) # Variance  std\_dist[i] <- sqrt(mean((tmp\_dist-mean(tmp\_dist))^2)) # Standard Deviation  }  plot(avg\_dist)  plot(var\_dist)  plot(std\_dist) |



|  |  |
| --- | --- |
| **[Graph, n=100]** | |
| Average |  |
| Variance |  |
| Standard  Deviation |  |

n = 10000

diameter = 3

|  |
| --- |
| **[n=10000]** |
| g <- random.graph.game(10000,0.01,directed=F)  d <- diameter(g)  avg\_dist <- c(0)  var\_dist <- c(0)  std\_dist <- c(0)  num\_walker <- 100  shortest\_paths <- shortest.paths(g)  for (i in 1:100) {  rw <- netrw(g, num\_walker, start.node = sample(0:(vcount(g) - 1) + as.integer((sessionInfo())$otherPkgs$igraph$Version >= "0.6"), num\_walker, replace = TRUE), damping = 1, weights = NULL, T = i, seed = NULL, output.walk.path = TRUE, output.walkers = 0:( num\_walker - 1), output.visit.prob = TRUE, output.nodes = 0:(vcount(g) - 1), output.device = "memory", walk.path.file = "walk\_path.txt", visit.prob.file = "visit\_prob.txt", local.pagerank = FALSE, teleport.prob = NULL)    tmp\_dist <- c(0)  for (j in 1:num\_walker) {  tmp\_dist[j] <- shortest\_paths[rw[[2]][1,j], rw[[2]][i,j]]+1  }    avg\_dist[i] <- mean(tmp\_dist)  var\_dist[i] <- mean((tmp\_dist-mean(tmp\_dist))^2) # Variance  std\_dist[i] <- sqrt(mean((tmp\_dist-mean(tmp\_dist))^2)) # Standard Deviation  }  plot(avg\_dist)  plot(var\_dist)  plot(std\_dist) |

|  |  |
| --- | --- |
| **[Graph, n=10000]** | |
| Average |  |
| Variance |  |
| Standard  Deviation |  |

(e)

|  |
| --- |
| **[Code]** |
| g <- random.graph.game(1000,0.01,directed=F)  plot(degree.distribution(g))  num\_walker <- 1  nei\_deg <- c(0)  for (i in 1:100) {  rw <- netrw(g, num\_walker, start.node = sample(0:(vcount(g) - 1) + as.integer((sessionInfo())$otherPkgs$igraph$Version >= "0.6"), num\_walker, replace = TRUE), damping = 1, weights = NULL, T = i, seed = NULL, output.walk.path = TRUE, output.walkers = 0:( num\_walker - 1), output.visit.prob = TRUE, output.nodes = 0:(vcount(g) - 1), output.device = "memory", walk.path.file = "walk\_path.txt", visit.prob.file = "visit\_prob.txt", local.pagerank = FALSE, teleport.prob = NULL)    nei\_deg[i] <- degree(g, rw[[2]][i,1])  }  # Make Plot the Degree Distribution  h <- hist(nei\_deg, breaks = 10, plot=FALSE)  h$counts = h$counts/sum(h$counts)  plot(h, ylab="degree.distribution") |

|  |
| --- |
| **[Degree Distribution]** |
|  |

|  |
| --- |
| **[Degree Distribution of the nodes reached at the end of the random walk]** |
|  |

2. Random walk on networks with fat-tailed degree distribution

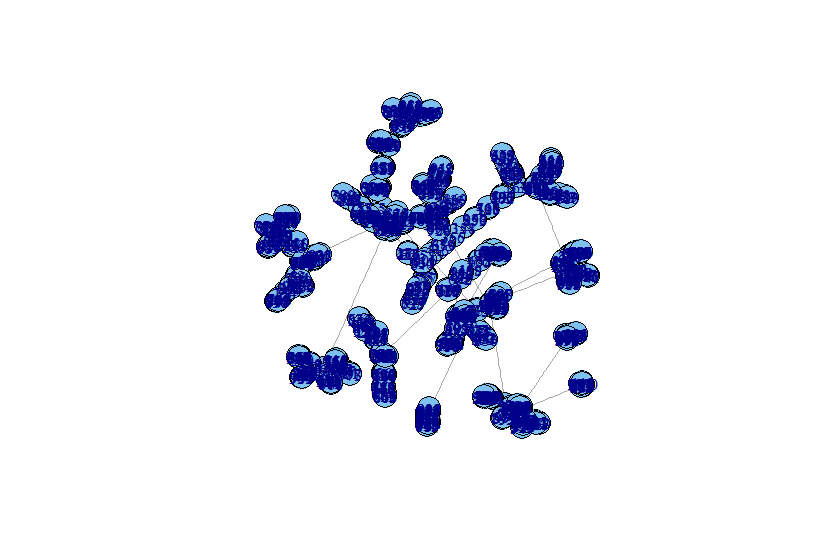
(a)

|  |
| --- |
| **[Code]** |
| power = -3  numnodes = 1000  g<-barabasi.game(numnodes, power, directed = FALSE) |

(b)

|  |
| --- |
| **[Code]** |
| power = -3  numnodes = 1000  g<-barabasi.game(numnodes, power, directed = FALSE)  avg\_dist <- c(0)  var\_dist <- c(0)  std\_dist <- c(0)  num\_walker <- 100  shortest\_paths <- shortest.paths(g)  for (i in 1:100) {  rw <- netrw(g, num\_walker, start.node = sample(0:(vcount(g) - 1) + as.integer((sessionInfo())$otherPkgs$igraph$Version >= "0.6"), num\_walker, replace = TRUE), damping = 1, weights = NULL, T = i, seed = NULL, output.walk.path = TRUE, output.walkers = 0:( num\_walker - 1), output.visit.prob = TRUE, output.nodes = 0:(vcount(g) - 1), output.device = "memory", walk.path.file = "walk\_path.txt", visit.prob.file = "visit\_prob.txt", local.pagerank = FALSE, teleport.prob = NULL)    tmp\_dist <- c(0)  for (j in 1:num\_walker) {  tmp\_dist[j] <- shortest\_paths[rw[[2]][1,j], rw[[2]][i,j]]+1  }    avg\_dist[i] <- mean(tmp\_dist)  var\_dist[i] <- mean((tmp\_dist-mean(tmp\_dist))^2) # Variance  std\_dist[i] <- sqrt(mean((tmp\_dist-mean(tmp\_dist))^2)) # Standard Deviation  }  plot(avg\_dist)  plot(var\_dist)  plot(std\_dist) |

|  |  |
| --- | --- |
| **[Graph]** | |
| Average |  |
| Variance |  |
| Standard  Deviation |  |



(c)

(d)

|  |  |
| --- | --- |
| **[Graph, n=100]** | |
| Graph |  |
| Average |  |
| Variance |  |
| Standard  Deviation |  |

|  |  |
| --- | --- |
| **[Graph, n=10000]** | |
| Graph |  |
| Average |  |
| Variance |  |
| Standard  Deviation |  |

(e)

|  |
| --- |
| **[Code]** |
| power = -3  numnodes = 1000  g<-barabasi.game(numnodes, power, directed = FALSE)  plot(degree.distribution(g))  num\_walker <- 1  nei\_deg <- c(0)  for (i in 1:100) {  rw <- netrw(g, num\_walker, start.node = sample(0:(vcount(g) - 1) + as.integer((sessionInfo())$otherPkgs$igraph$Version >= "0.6"), num\_walker, replace = TRUE), damping = 1, weights = NULL, T = i, seed = NULL, output.walk.path = TRUE, output.walkers = 0:( num\_walker - 1), output.visit.prob = TRUE, output.nodes = 0:(vcount(g) - 1), output.device = "memory", walk.path.file = "walk\_path.txt", visit.prob.file = "visit\_prob.txt", local.pagerank = FALSE, teleport.prob = NULL)    nei\_deg[i] <- degree(g, rw[[2]][i,1])  }  # Make Plot the Degree Distribution  h <- hist(nei\_deg, breaks = 10, plot=FALSE)  h$counts = h$counts/sum(h$counts)  plot(h, ylab="degree.distribution") |

|  |  |
| --- | --- |
| **[Degree Distribution]** | |
| Degree  Distribution |  |
| Degree  Distribution  by  Random Walk |  |

3. PageRank

(a)

|  |
| --- |
| **[Code]** |
| g <- random.graph.game(1000,0.01,directed=F)  num\_walker <- 100  rw <- netrw(g, num\_walker, start.node = sample(0:(vcount(g) - 1) + as.integer((sessionInfo())$otherPkgs$igraph$Version >= "0.6"), num\_walker, replace = TRUE), damping = 1, weights = NULL, T = 100, seed = NULL, output.walk.path = TRUE, output.walkers = 0:( num\_walker - 1), output.visit.prob = TRUE, output.nodes = 0:(vcount(g) - 1), output.device = "memory", walk.path.file = "walk\_path.txt", visit.prob.file = "visit\_prob.txt", local.pagerank = FALSE, teleport.prob = NULL)  degree\_g <- degree(g)  cor(degree\_g,rw[[1]])  plot(degree\_g~rw[[1]]) |

|  |  |
| --- | --- |
| **[Graph]** | |
| Random Walk  Avg Visit Prob |  |
| Degree of Nodes |  |
| y : degree  x : avg visit prob |  |

Correlation : 0.6543276

(b)

|  |
| --- |
| **[Code]** |
| g <- random.graph.game(1000,0.01,directed=T)  num\_walker <- 100  rw <- netrw(g, num\_walker, start.node = sample(0:(vcount(g) - 1) + as.integer((sessionInfo())$otherPkgs$igraph$Version >= "0.6"), num\_walker, replace = TRUE), damping = 1, weights = NULL, T = 100, seed = NULL, output.walk.path = TRUE, output.walkers = 0:( num\_walker - 1), output.visit.prob = TRUE, output.nodes = 0:(vcount(g) - 1), output.device = "memory", walk.path.file = "walk\_path.txt", visit.prob.file = "visit\_prob.txt", local.pagerank = FALSE, teleport.prob = NULL)  degree\_g <- degree(g)  plot(rw[[1]])  plot(degree\_g)  cor(degree\_g,rw[[1]])  plot(degree\_g~rw[[1]]) |

|  |  |
| --- | --- |
| **[Graph]** | |
| Random Walk  Avg Visit Prob |  |
| Degree of Nodes |  |
| y : degree  x : avg visit prob |  |

Correlation : 0.507608

(c)

|  |
| --- |
| **[Code]** |
| g <- random.graph.game(1000,0.01,directed=F)  num\_walker <- 100  rw <- netrw(g, num\_walker, start.node = sample(0:(vcount(g) - 1) + as.integer((sessionInfo())$otherPkgs$igraph$Version >= "0.6"), num\_walker, replace = TRUE), damping = 0.85, weights = NULL, T = 100, seed = NULL, output.walk.path = TRUE, output.walkers = 0:( num\_walker - 1), output.visit.prob = TRUE, output.nodes = 0:(vcount(g) - 1), output.device = "memory", walk.path.file = "walk\_path.txt", visit.prob.file = "visit\_prob.txt", local.pagerank = FALSE, teleport.prob = NULL)  degree\_g <- degree(g)  plot(rw[[1]])  plot(degree\_g)  cor(degree\_g,rw[[1]])  plot(degree\_g~rw[[1]]) |

|  |  |
| --- | --- |
| **[Graph]** | |
| Random Walk  Avg Visit Prob |  |
| Degree of Nodes |  |
| y : degree  x : avg visit prob |  |

Correlation : 0.6054806

4. Personalized PageRank

(a, b)

Compare PageRank and Personalized PageRank

|  |
| --- |
| **[Code]** |
| g <- random.graph.game(1000,0.01,directed=T)  num\_walker <- 100  rw <- netrw(g, num\_walker, start.node = sample(0:(vcount(g) - 1) + as.integer((sessionInfo())$otherPkgs$igraph$Version >= "0.6"), num\_walker, replace = TRUE), damping = 0.85, weights = NULL, T = 100, seed = NULL, output.walk.path = TRUE, output.walkers = 0:( num\_walker - 1), output.visit.prob = TRUE, output.nodes = 0:(vcount(g) - 1), output.device = "memory", walk.path.file = "walk\_path.txt", visit.prob.file = "visit\_prob.txt", local.pagerank = FALSE, teleport.prob = NULL)  rw2 <- netrw(g, num\_walker, start.node = sample(0:(vcount(g) - 1) + as.integer((sessionInfo())$otherPkgs$igraph$Version >= "0.6"), num\_walker, replace = TRUE), damping = 0.85, weights = NULL, T = 100, seed = NULL, output.walk.path = TRUE, output.walkers = 0:( num\_walker - 1), output.visit.prob = TRUE, output.nodes = 0:(vcount(g) - 1), output.device = "memory", walk.path.file = "walk\_path.txt", visit.prob.file = "visit\_prob.txt", local.pagerank = TRUE, teleport.prob = NULL)  pg <- personalized.pagerank(g, damping=0.85, weights=NULL, epsilon=1e-5/sum(degree(g, mode="out")), prob=rep(1.,vcount(g))) |

|  |  |
| --- | --- |
| **[Graph]** | |
| Ave Visit Probl  Random Walk  Local PageRank  False |  |
| Ave Visit Probl  Random Walk  Local PageRank  True |  |
| Personalized  Page Rank |  |

(c)