Assignment 3

Spencer Moon

11/13/2017

Problem 1

The data was loaded using the following code:

```
# Load necessary libraries for this exercise
library(expm)

# Import data
markov <- read.table('markov100.txt')
markov <- as.matrix(markov)
```

Part A

```
# Create state vector with first element as 1 (currently at State 1)
a <- c(rep(0, dim(markov)[1]))
a[1] <- 1

# Find probability of being in State 5 after 10 transitions
trans <- 10
result <- a %*% (markov %^% trans)
result[5]
```

```
## [1] 0.045091
```

The probability of being in State 5 after 10 transitions from State 1 is 4.5%.

Part B

```
# Create state vector with first element as 1 (currently at State 1)
a <- c(rep(0, dim(markov)[1]))
prob <- 1/3
a[c(1,2,3)] <- prob

# Find probability of being in State 5 after 10 transitions
trans <- 10
result <- a %*% (markov %^% trans)
result[10]
```

```
## [1] 0.08268901
```

The probability of being in State 10 after 10 transitions from State 1, 2, or 3 with equal probability is 8.3%.

Part C

```r
# Replace the last row of (P transpose - I) by a vector of ones
Q <- t(markov) - diag(dim(markov)[1])
Q[dim(markov)[1],] <- c(rep(1, dim(markov)[1]))

# Solve for pi
rhs <- c(rep(0, dim(markov)[1]))
rhs[dim(markov)[1]] <- 1
Pi <- solve(Q) %*% rhs
Pi[1]
```

```
## [1] 0.01256589
```

The steady state probability of State 1 is 1.3%.

Part D

```r
B <- markov[-100, -100]
Q <- diag(dim(markov)[1] - 1) - B
e = c(rep(1, dim(markov)[1] - 1))
m = solve(Q) %*% e
m[1]
```

```
## [1] 254.9395
```

The mean first passage time from State 1 to State 100 is 254.94.

Problem 2

The data was loaded using the following code:

```r
web <- read.table('webtraffic.txt', header = TRUE)
```

Part A

```r
# Create vector of column sums
Traffic <- as.vector(colSums(web))

# Create 9 by 9 matrix from vector
Traffic <- matrix(Traffic, nrow = 9, ncol = 9, byrow = TRUE)
Traffic
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
##  [1,]    0  447  553    0    0    0    0    0    0
##  [2,]    0   23  230  321    0    0    0    0   63
##  [3,]    0  167   43  520    0    0    0    0   96
##  [4,]    0    0    0   44  158  312  247    0  124
##  [5,]    0    0    0    0   22   52   90  127  218
##  [6,]    0    0    0    0   67   21    0  294   97
##  [7,]    0    0    0    0    0   94    7  185   58
##  [8,]    0    0    0    0  262    0    0   30  344
```

```
## [9,]    0    0    0    0    0    0    0    0    0
```

Part B

```r
# Change one matrix element
Traffic[9,1] = 1000
P <- Traffic/rowSums(Traffic)
P
```

```
##       [,1]       [,2]       [,3]       [,4]      [,5]       [,6]
##  [1,]    0 0.44700000 0.55300000 0.00000000 0.0000000 0.00000000
##  [2,]    0 0.03610675 0.36106750 0.50392465 0.0000000 0.00000000
##  [3,]    0 0.20217918 0.05205811 0.62953995 0.0000000 0.00000000
##  [4,]    0 0.00000000 0.00000000 0.04971751 0.1785311 0.35254237
##  [5,]    0 0.00000000 0.00000000 0.00000000 0.0432220 0.10216110
##  [6,]    0 0.00000000 0.00000000 0.00000000 0.1398747 0.04384134
##  [7,]    0 0.00000000 0.00000000 0.00000000 0.0000000 0.27325581
##  [8,]    0 0.00000000 0.00000000 0.00000000 0.4119497 0.00000000
##  [9,]    1 0.00000000 0.00000000 0.00000000 0.0000000 0.00000000
##             [,7]       [,8]      [,9]
##  [1,] 0.00000000 0.00000000 0.0000000
##  [2,] 0.00000000 0.00000000 0.0989011
##  [3,] 0.00000000 0.00000000 0.1162228
##  [4,] 0.27909605 0.00000000 0.1401130
##  [5,] 0.17681729 0.24950884 0.4282908
##  [6,] 0.00000000 0.61377871 0.2025052
##  [7,] 0.02034884 0.53779070 0.1686047
##  [8,] 0.00000000 0.04716981 0.5408805
##  [9,] 0.00000000 0.00000000 0.0000000
```

Part C

```r
# Replace the last row of (P transpose - I) by a vector of ones
Q <- t(P) - diag(dim(P)[1])
Q[dim(P)[1],] <- c(rep(1, dim(P)[1]))

# Solve for pi
rhs <- c(rep(0, dim(P)[1]))
rhs[dim(P)[1]] <- 1
Pi <- solve(Q) %*% rhs
Pi
```

```
##             [,1]
##  [1,] 0.15832806
##  [2,] 0.10085497
##  [3,] 0.13077897
##  [4,] 0.14012033
##  [5,] 0.08058898
##  [6,] 0.07583914
##  [7,] 0.05446485
##  [8,] 0.10069664
##  [9,] 0.15832806
```

Part D

```r
# Create average time vector
time <- c(0.1, 2, 3, 5, 5, 3, 3, 2, 0)

# Calculate average time
sum(Pi * time)
```

```
## [1] 2.305731
```

The average time sepnt on the website is 2.3 minutes.

Part E

```r
# Adjust for additional links
Traffic[2,6] <- .3 * Traffic[2,3]
Traffic[2,3] <- .7 * Traffic[2,3]
Traffic[2,7] <- .2 * Traffic[2,4]
Traffic[2,4] <- .8 * Traffic[2,4]

# Recalculate probability matrix
P <- Traffic/rowSums(Traffic)

# Replace the last row of (P transpose - I) by a vector of ones
Q <- t(P) - diag(dim(P)[1])
Q[dim(P)[1],] <- c(rep(1, dim(P)[1]))

# Solve for pi
rhs <- c(rep(0, dim(P)[1]))
rhs[dim(P)[1]] <- 1
Pi2 <- solve(Q) %*% rhs

# Check variance differences of Pi and Pi2
var(Pi)
```

```
##             [,1]
## [1,] 0.001410675
```

```r
var(Pi2)
```

```
##             [,1]
## [1,] 0.001219604
```

The variance for Pi2 is slightly lower, which indicates that the traffic has become more balanced with the new links.