MSiA400_Lab_Assignment3

```r
markv = read.table("markov100.txt", header = F)
P <- as.matrix(markv)
```

```r
library(expm)
```

```
## Warning: package 'expm' was built under R version 3.4.2
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'expm'
```

```
## The following object is masked from 'package:Matrix':
##
##     expm
```

Problem 1a

We calculate the probability distribution after 10 transitions given State 1 now by multiplying the current state vector with the probability matrix taken to the power of 10.

```r
statevec <- c(1,rep(0,99))
tenstep <- statevec %*% (P%^%10)
tenstep[5]
```

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

The probability of being in state 5 can be seen to be 0.045.

Problem 1b

After initializing the evenly possible initial states, we calculate the probabilities after 10 steps.

```r
statevec <- c(1/3,1/3,1/3,rep(0,97))
tenstep <- statevec %*% (P%^%10)
tenstep[10]
```

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

The probability of being in State 10 is seen to be 0.082.

Problem 1c

We calculate the steady state probabilities below.

```r
SS <- t(P) - diag(100)
SS[100,] <- rep(1,100)
rhs <- c(rep(0,99),1)
Pi <- solve(SS) %*% rhs
#Steay state prob of 1:
Pi[1]
```

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

We see that the steady state probability of being in state 1 is 0.0126.

Problem 1d

```
B <- P[-100,-100]
Q <- diag(99) - B
e <- c(rep(1,99))
m <- solve(Q) %*% e
m[1]
```

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

The mean first passage time from State 1 to State 100 is seen to be roughly 255.

Problem 2

```
web <- read.table("webtraffic.txt", sep = "\t", header = T)
```

Problem 2a

```
sums <- colSums(web)
Traffic <- matrix(sums, nrow = 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
```

Problem 2b

We observe that column 1 and row 9 are 0s.

```
Traffic[9,1] <- 1000
#P <- Traffic/rowSums(Traffic)
P <- (scale(t(Traffic), center = FALSE, scale = colSums(t(Traffic))))
P <- t(as.matrix(P))
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
## attr(,"scaled:scale")
## [1] 1000  637  826  885  509  479  344  636 1000
```

Problem 2c

```r
SS <- t(P) - diag(9)
SS[9,] <- rep(1,9)
rhs <- c(rep(0,8),1)
Pi <- solve(SS) %*% 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
```

The steady state probability vector is displayed above.

Problem 2d

One way to calculate the total time spent on the website given the traffic data provided would be to calculate the number of visitors to each page and multiply those figures with the average time spent. The column sums would show how many visitors went to that page.

```r
avgtime <- c(0.1,2,3,5,5,3,3,2)
pagevisitors <- colSums(Traffic[,1:8])
totTime <- avgtime %*% pagevisitors/1000
totTime
```

```
##        [,1]
## [1,] 14.563
```

We see that for this data, the total time spent per visitor was on average 14.563 minutes.

Problem 2e

We redirect the number of transitions in the Traffic matrix as given by the question.

```
OrigTraffic <- Traffic
Traffic[2,6] <- Traffic[2,6] + (Traffic[2,3]*0.3)
Traffic[2,3] <- Traffic[2,3] * (1-0.3)
Traffic[2,7] <- Traffic[2,7] + (Traffic[2,3] * 0.2)
Traffic[2,4] <- Traffic[2,4]* (1-0.2)
```

Re-calculating transition probabilities.

```
P <- (scale(t(Traffic), center = FALSE, scale = colSums(t(Traffic))))
P <- t(as.matrix(P))
```

Again we find the steady state probailities.

```
SS <- t(P) - diag(9)
SS[9,] <- rep(1,9)
rhs <- c(rep(0,8),1)
Pi2<- solve(SS) %*% rhs
Pi2
```

```
##             [,1]
##  [1,] 0.16142771
##  [2,] 0.10074592
##  [3,] 0.12245434
##  [4,] 0.12612333
##  [5,] 0.08166198
##  [6,] 0.08328981
##  [7,] 0.05614426
##  [8,] 0.10672495
##  [9,] 0.16142771
```

```
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
```

Comparing the new steady state probabilities to the old ones, we see that there is some improvement from before, but pages 3 and 4 remain dominant compared to the others.