Everything Starts with Data Lab Exercise #3

Johnny Chiu

11/12/2017

Problem 1

Problem 1

In *Markov100.txt*, the one step transition probability matrix for a Markov chain with 100 states (State 1 to State 100) is given. Note that the data has no heading.

Name of the data set	Markov100
Number of rows	100
Number of columns	100

Problem 1(a)

Suppose we are at State 1 now. Find and display the probability of being in State 5 after 10 transitions.

Problem 1(b)

Suppose we are at one of States 1,2, and 3 with equal probabilities. Find and display the probability of being in State 10 after 10 transitions.

Problem 1(c)

Find the steady state probability of being in State 1.

Problem 1(d)

Find the mean first passage time from State 1 to State 100.

Answer 1(a)

```
# read data
markov <- read.table('_data/markov100.txt', header = FALSE)

P = as.matrix(markov)
a_1 = c(1,rep(0,99))
a_1_t10 = a_1 %*% (P %^%10)
```

The probability of being in State 5 after 10 transitions is 0.045091.

Answer 1(b)

```
a_2 = c(rep(1/3,3),rep(0,97))
a_2_t10 = a_2 %*% (P %^%10)
```

The probability of being in State 10 after 10 transitions is 0.082689.

Answer 1(c)

```
size = dim(P)[1]
Q = t(P)-diag(size)
Q[size,] = rep(1,size)
```

```r
rhs = c(rep(0,size-1),1)
Pi = solve(Q) %*% rhs
```

The steady state probability of being in State 1 is 0.0125659

Answer 1(d)

```r
B =P[1:size-1,1:size-1]
Q = diag(size-1) - B
e = c(rep(1, size-1))
m = solve(Q) %*% e
```

The mean first passage time from state 1 to state 100 is 254.9394631

Problem 2

Problem 2

You are asked to analyze the data from an website with 8 pages (Page 1 - Page 8). Let us assume that there is a virtual page (Page 9) that a visitor must automatically visit when the visitor leaves the website. The visitors always start their visit from Page 1. Let us formulate a Markov chain for this website. The states are defined as

$$S_i = \text{visitor is at Page } i,\ i = 1, \cdots, 9.$$

For example, suppose that a visitor enters the website (hence visit Page 1), moves to Page 3, Page 5, and then leave the website, sequentially. Then, the user visits States S_1, S_3, S_5, and S_9, sequentially.

Please find the attached data *webtraffic.txt*. The data includes the record of 1000 visitors (rows). The data has 81 columns labeled as $t11, t12, \cdots, t19, t21, t22, \cdots, t29, \cdots, t91, t92, \cdots, t99$. The label t_{ij} represents the transition from State i to State j, for $i = 1, \cdots, 9$ and $j = 1, \cdots, 9$. For example, $t12$ is the transition from State 1 to State 2, and $t84$ is the transition from State 8 to State 4. For each visitor (row), it has 1 for column t_{ij} if the visitor makes transition from State i to State j, and it has 0 elsewhere. For example, if a visitor visits States S_1, S_3, S_5, and S_9, sequentially, then the corresponding row has 1 for columns $t13, t35, t59$ and 0 elsewhere.

The summary of the data set is below.

Name of the data set	webtraffic
Type of data	binaries (0,1)
Number of rows	1000
Number of columns	81

Problem 2(a)

Construct 9 by 9 matrix Traffic that counts total traffic between State i to State j for all $i = 1, \cdots, 9$ and $j = 1, \cdots, 9$. Display Traffic.
Hint colSums() adds all rows for each column.

Problem 2(b)

Observe that Traffic has 0's in row 9 and 0's in column 1. Set Traffic[9,1]=1000. Construct the one step transition probability matrix P and display it.

Problem 2(c)

Calculate and display the steady state probability vector Pi.

Problem 2(d)

The following table presents the average time that the visitors spend on each page.

Page	1	2	3	4	5	6	7	8
Avg(minute)	0.1	2	3	5	5	3	3	2

Calculate and display the average time a visitor spend on the website (until she leaves).

Problem 2(e)

In the output of Problem 2(c), observe that Pages 3 and 4 are one of the most crowded pages except Pages 1 and 9. To balance the traffic, the owner of the website decided to create links from Page 2 to Pages 6,7 (hence, from State 2 to States 6,7). By adding the links, the owner anticipates that, from Page 2, 30% of the current outgoing traffic to State 3 would move to State 6, and 20% of the current outgoing traffic to State 4 would move to State 7. Calculate new steady state probability vector Pi2 to check the effect of the new links. Decide if the link helped balancing the traffic by comparing the variance of Pi and Pi2.
Hint Start with matrix Traffic from Problem 2(a).

Answer 2(a)

```
# read data
web = read.table('_data/webtraffic.txt', header=TRUE)

Traffic_list =  colSums(web)
Traffic = as.matrix(data.frame(split(Traffic_list, 1:9)))
Traffic

##      X1  X2  X3  X4  X5  X6  X7  X8  X9
## t11   0 447 553   0   0   0   0   0   0
## t21   0  23 230 321   0   0   0   0  63
## t31   0 167  43 520   0   0   0   0  96
## t41   0   0   0  44 158 312 247   0 124
## t51   0   0   0   0  22  52  90 127 218
## t61   0   0   0   0  67  21   0 294  97
## t71   0   0   0   0   0  94   7 185  58
```

```
## t81   0    0    0    0 262    0    0   30  344
## t91   0    0    0    0    0    0    0    0    0
```

Answer 2(b)

```
Traffic[9,1] = 1000
P=Traffic/rowSums(Traffic)
P
```

```
##      X1          X2          X3          X4         X5          X6          X7
## t11   0 0.44700000 0.55300000 0.00000000 0.0000000 0.00000000 0.00000000
## t21   0 0.03610675 0.36106750 0.50392465 0.0000000 0.00000000 0.00000000
## t31   0 0.20217918 0.05205811 0.62953995 0.0000000 0.00000000 0.00000000
## t41   0 0.00000000 0.00000000 0.04971751 0.1785311 0.35254237 0.27909605
## t51   0 0.00000000 0.00000000 0.00000000 0.0432220 0.10216110 0.17681729
## t61   0 0.00000000 0.00000000 0.00000000 0.1398747 0.04384134 0.00000000
## t71   0 0.00000000 0.00000000 0.00000000 0.0000000 0.27325581 0.02034884
## t81   0 0.00000000 0.00000000 0.00000000 0.4119497 0.00000000 0.00000000
## t91   1 0.00000000 0.00000000 0.00000000 0.0000000 0.00000000 0.00000000
##              X8         X9
## t11 0.00000000 0.0000000
## t21 0.00000000 0.0989011
## t31 0.00000000 0.1162228
## t41 0.00000000 0.1401130
## t51 0.24950884 0.4282908
## t61 0.61377871 0.2025052
## t71 0.53779070 0.1686047
## t81 0.04716981 0.5408805
## t91 0.00000000 0.0000000
```

Answer 2(c)

```
size = dim(P)[1]
Q = t(P)-diag(size)
Q[size,] = rep(1,size)
rhs = c(rep(0,size-1),1)
Pi = solve(Q) %*% rhs
```

state	steady_prob
P1	0.1583281
P2	0.1008550
P3	0.1307790
P4	0.1401203
P5	0.0805890
P6	0.0758391
P7	0.0544649
P8	0.1006966
P9	0.1583281

Answer 2(d)

```
traffic_view = colSums(Traffic)[1:8]
avg_min = c(0.1,2,3,5,5,3,3,2)
avg_time_per_visitor = sum(traffic_view*avg_min)/1000
```

The average time that a visitor spend on the website is 14.563

Answer 2(e)

```
Traffic2 = Traffic
# From Page 2, 30% of the current outgoing traffic to State 3 would move to
State 6.
Traffic2[2,3] = Traffic[2,3] - Traffic[2,3]*0.3
Traffic2[2,4] = Traffic[2,4] - Traffic[2,4]*0.2

# From Page 2, 20% of the current outgoing traffic to State 4 would move to
State 7.
Traffic2[2,6] = Traffic[2,6] + Traffic[2,3]*0.3
Traffic2[2,7] = Traffic[2,7] + Traffic[2,4]*0.2

P2 = Traffic2/rowSums(Traffic2)
size2 = dim(P2)[1]
Q2 = t(P2)-diag(size2)
Q2[size2,] = rep(1,size2)
rhs2 = c(rep(0,size2-1),1)
Pi2 = solve(Q2) %*% rhs2
```

state	steady_prob
P1	0.1616284
P2	0.1003434
P3	0.1210433
P4	0.1227572
P5	0.0816461
P6	0.0825088
P7	0.0600322
P8	0.1084121
P9	0.1616284

name	variance
Pi	0.0014107
Pi2	0.0012196

Yes, the link helped balancing the traffic, since the variance of Pi2 is lower than Pi.