Homework 9

Gabriel Lahman

4/7/2019

```
source("finite_markov_chain_functions.R")
```

```
## Warning: package 'expm' was built under R version 3.5.2
```

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

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

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

```
## Warning: package 'markovchain' was built under R version 3.5.2
```

```
## Package:  markovchain
## Version:  0.6.9.14
## Date:     2019-01-20
## BugReport: http://github.com/spedygiorgio/markovchain/issues
```

```
## Warning: package 'DescTools' was built under R version 3.5.2
```

Question 1

1.1

```
data("preproglucacon")
bases = preproglucacon[,2]
est_trans_mat = estimate_transition_matrix(bases)
```

```
##           A         C          G         T
## A 0.3585271 0.1434109 0.16666667 0.3313953
## C 0.3840304 0.1558935 0.02281369 0.4372624
## G 0.3053097 0.1991150 0.15044248 0.3451327
## T 0.2844523 0.1819788 0.17667845 0.3568905
```

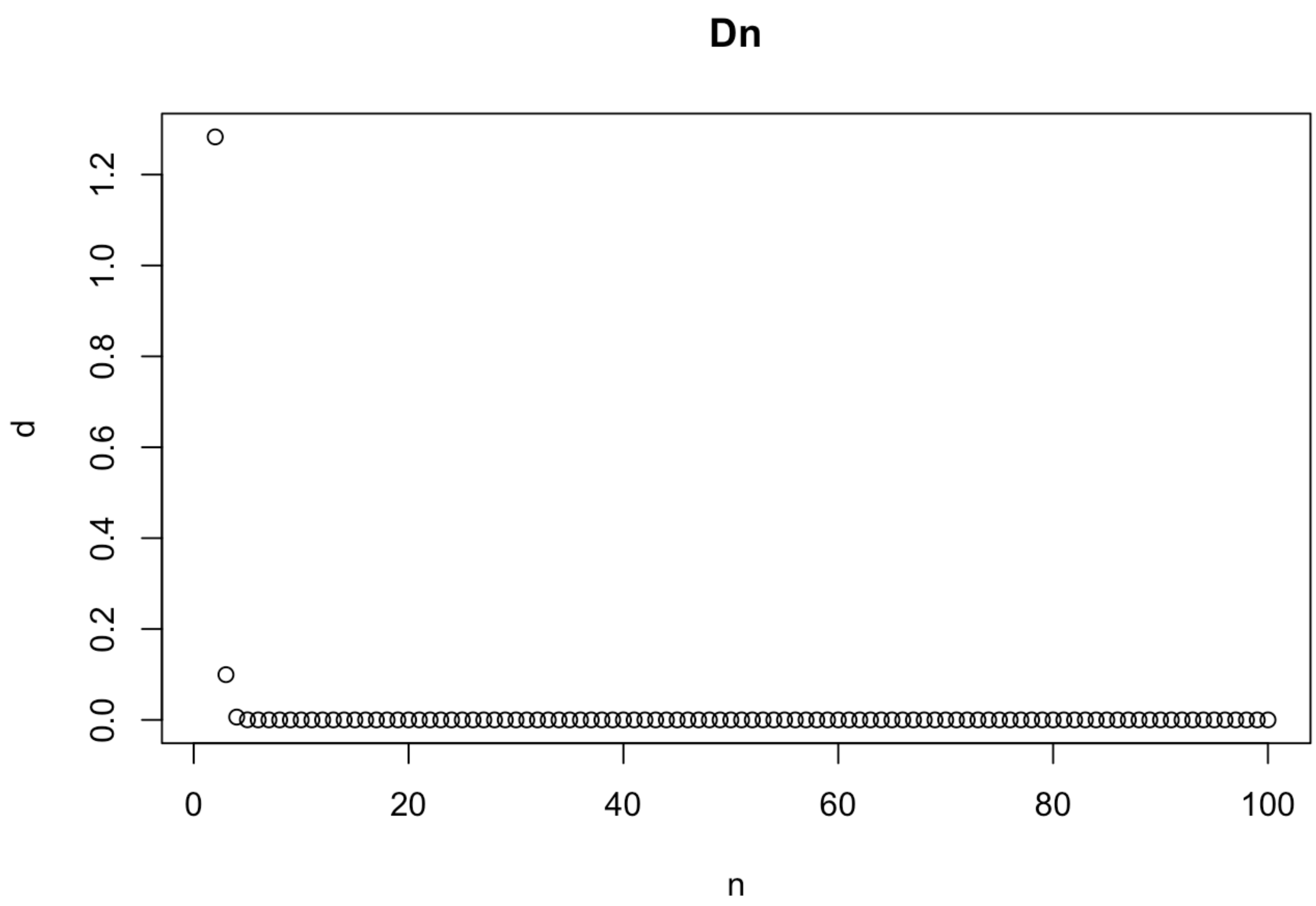
1.2

```
mot = mean_occupancy_time_fun(bases)
```

```
##         A         C         G         T
## 0.3282443 0.1673028 0.1444020 0.3600509
```

1.3

```
n = 100
x = matrix(nrow = n, ncol = 4)
x[1,] = c(1,0,0,0)
d = rep(NA, n)
for (i in 2:n) {
  x[i,] = x[i - 1,] %*% est_trans_mat
  d[i] = sum(abs(x[i,] - x[i - 1,]))
}
plot(d, xlab = "n", main = "Dn")
```



The plot converges to zero incredibly fast, showing that we reach a stationary matrix in under 10 iterations.

1.4

```
v = mean_recurrence_time_fun(bases)
p = 1/v
```

```
## [1] "p"
```

```
##         A         C         G         T
## 0.3284439 0.1679487 0.1438574 0.3610224
```

```
## [1] "x100"
```

```
## [1] 0.3284532 0.1674093 0.1438574 0.3602801
```

The vectors are pratically equal. ## Question 2

Recurrent States: 1,5,6,4

Transient:3

Absorbing: 6

No ergodic components

Cyclic components: (6), (2,4), (1,5)

Question 3

3.1

$P(Y = (y_1, \ldots, y_8)) = P(Y_1 = y_1)P(Y_4 = y_4)P(Y_3 = y_3|Y_1, Y_4)P(Y_5 = y_5|Y_4)P(Y_2 = y_2|Y_3)P(Y_7 = y_7|Y_3, Y_5)P(Y_8 = y_8|Y_5, Y_6, Y_7)$

3.2

3.3

Question 4

4.1

```
f = function() {
  X = rexp(1)
  Y = rnorm(1)
  Z = rnorm(1, mean = Y, sd = 1/sqrt(X))
  return (c(X,Y,Z))
}
```

4.2

$X|Y, Z \; Gamma(\alpha = 3/2, \beta = \frac{z-y}{2})$

$Y|X, Z \; Normal(\mu = zx, \sigma^2 = \frac{1}{1-x})$

4.3

```
f2 = function(state) {
  x1_y0z0 = rgamma(1, shape = 3/2, rate = (state[[2]] - state[[3]])/ 2)
  y1_x1z0 = rnorm(1, mean = x1_y0z0 * state[[3]], sd = sqrt(abs(1/1 - x1_y0z0)))
  z1_x1y1 = rnorm(1, mean = y1_x1z0, sd = 1 / sqrt(abs(x1_y0z0)))
  return (c(x1_y0z0, y1_x1z0, z1_x1y1))
}
```

4.4

I am not sure my last function is correct as it is producing NA’s, so I did not populate the data.

```
state = f()

#for (i in 1:10000) {
#  print(state)
#  state = f2(state)
#}
```

Question 5

5.1

$E[Y_1] = E[\rho Y_0 + \epsilon_1] = \rho E[Y_0] + E[\epsilon_1] = \rho E[\epsilon_0] + 0 = \rho(0) = 0$

5.2

$Var[Y_1] = Var[\rho Y_0 + \epsilon_1] = \rho^2 Var[Y_0] + \sigma^2_{\epsilon_1} = \rho^2(1) + 1 = \rho^2 + 1$

5.3

$E[Y_2] = E[\rho Y_1 + \epsilon_2] = \rho E[Y_1] + E[\epsilon_2] = \rho(0) + 0 = 0$

5.4

$Var[Y_2] = Var[\rho Y_1 + \epsilon_2] = \rho^2 Var[Y_1] + \sigma^2_{\epsilon_2} = \rho^2(\rho^2 + 1) + 1 = \rho^4 + \rho^2 + 1$

5.5

$E[Y_3] = E[\rho Y_2 + \epsilon_3] = \rho E[Y_2] + E[\epsilon_3] = \rho(0) + 0 = 0$

5.6

$Var[Y_3] = Var[\rho Y_2 + \epsilon_3] = \rho^2 Var[Y_2] + \sigma^2_{\epsilon_3} = \rho^2(\rho^4 + \rho^2 + 1) + 1 = \rho^6 + \rho^4 + \rho^2 + 1$

Question 6

6.1

$E[Y_1] = E[\rho Y_0 + \phi\epsilon_0 + \epsilon_1] = \rho E[\epsilon_0] + \phi E[\epsilon_0] + E[\epsilon_1] = \rho(0) + \phi(0) = 0 = 0$

6.2

$Var[Y_1] = Var[\rho Y_0 + \phi\epsilon_0 + \epsilon_1] = \rho^2 Var[Y_0] + \phi_2 Var[\epsilon_0] + Var[\epsilon_1] = \rho^2 + \phi^2 + 1$

6.3

$E[Y_2] = E[\rho Y_1 + \phi\epsilon_1 + \epsilon_2] = \rho E[\epsilon_1] + \phi E[\epsilon_1] + E[\epsilon_2] = \rho(0) + \phi(0) = 0 = 0$

6.4

$Var[Y_2] = Var[\rho Y_1 + \phi\epsilon_1 + \epsilon_2] = \rho^2 Var[Y_1] + \phi_2 Var[\epsilon_1] + Var[\epsilon_2] = \rho^2(\rho^2 + \phi^2 + 1) + \phi^2 + 1$
$= \rho^4 + \rho^2\phi^2 + \rho^2 + \phi^2 + 1 = \rho^2(\rho^2 + 1) + \phi^2(\rho^2 + 1) + 1 = (\rho^2 + \phi^2)(\rho^2 + 1) + 1$

6.5

$E[Y_3] = E[\rho Y_2 + \phi\epsilon_2 + \epsilon_3] = \rho E[\epsilon_2] + \phi E[\epsilon_2] + E[\epsilon_3] = \rho(0) + \phi(0) + 0 = 0$

6.6

$Var[Y_3] = Var[\rho Y_2 + \phi\epsilon_2 + \epsilon_3] = \rho^2 Var[Y_2] + \phi^2 Var[\epsilon_2] + Var[\epsilon_3]$
$= \rho^2((\rho^2 + \phi^2)(\rho^2 + 1) + 1) + \phi^2 + 1 = \rho^2(\rho^4 + \rho^2 + \rho^2\phi^2 + \phi^2 + 1) + \phi^2 + 1$
$= \rho^6 + \rho^4 + \rho^4\phi^2 + \rho^2\phi^2 + \rho^2 + \phi^2 + 1 = \rho^6 + \rho^4 + \rho^2 + \phi^2(\rho^4 + \rho^2 + 1) + 1$
$= \rho^2(\rho^4 + \rho^2 + \rho^1) + \phi^2(\rho^4 + \rho^2 + 1) + 1$
$= (\rho^2 + \phi^2)(\rho^4 + \rho^2 + 1) + 1$