ASSIGNMENT 2

Group H

11/28/2019

library(markovchain)

## Package: markovchain  
## Version: 0.8.0  
## Date: 2019-09-13  
## BugReport: http://github.com/spedygiorgio/markovchain/issues

library(Matrix)  
  
#States space  
phoneStates <- c("Off","On","Lock","Unlock")  
phoneStates

## [1] "Off" "On" "Lock" "Unlock"

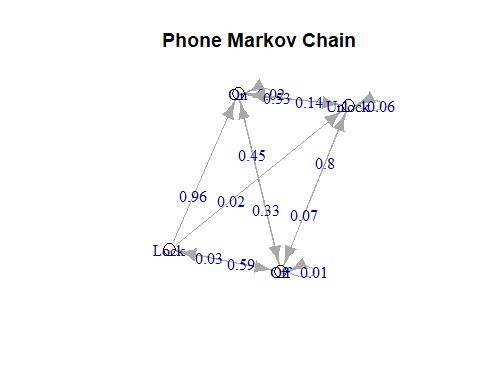
#Forming transition matrix  
phoneMatrix <- matrix(data = c(0.51/100,33.04/100,59.4/100,7.05/100,45.32/100,2.03/100,0,52.65/100,2.83/100,95.64/100,0,1.53/100,80.50/100,13.58/100,0,5.92/100),  
byrow = TRUE, nrow = 4,dimnames = list(phoneStates, phoneStates))  
  
mcphone <- new("markovchain", states = phoneStates, byrow = TRUE,  
transitionMatrix = phoneMatrix, name = "Phone")  
  
mcphone

## Phone   
## A 4 - dimensional discrete Markov Chain defined by the following states:   
## Off, On, Lock, Unlock   
## The transition matrix (by rows) is defined as follows:   
## Off On Lock Unlock  
## Off 0.0051 0.3304 0.594 0.0705  
## On 0.4532 0.0203 0.000 0.5265  
## Lock 0.0283 0.9564 0.000 0.0153  
## Unlock 0.8050 0.1358 0.000 0.0592

#Markov chain  
is(mcphone,"markovchain")

## [1] TRUE

#Transition plot  
plot(mcphone,main="Phone Markov Chain")



#Initial conditions: the state it starts in time, t=0  
#Start from Off  
s0= matrix(c(1,0,0,0),nrow = 1,byrow = T)  
s1=s0\*mcphone  
s1

## Off On Lock Unlock  
## [1,] 0.0051 0.3304 0.594 0.0705

#Start from On  
s0= matrix(c(0,1,0,0),nrow = 1,byrow = T)  
s1=s0\*mcphone  
s1

## Off On Lock Unlock  
## [1,] 0.4532 0.0203 0 0.5265

#Start from Lock  
s0= matrix(c(0,0,1,0),nrow = 1,byrow = T)  
s1=s0\*mcphone  
s1

## Off On Lock Unlock  
## [1,] 0.0283 0.9564 0 0.0153

#Start from Unlock  
s0= matrix(c(0,0,0,1),nrow = 1,byrow = T)  
s1=s0\*mcphone  
s1

## Off On Lock Unlock  
## [1,] 0.805 0.1358 0 0.0592

#Steady state  
phoneMatrix

## Off On Lock Unlock  
## Off 0.0051 0.3304 0.594 0.0705  
## On 0.4532 0.0203 0.000 0.5265  
## Lock 0.0283 0.9564 0.000 0.0153  
## Unlock 0.8050 0.1358 0.000 0.0592

fvalue=function(phoneMatrix,n)  
{  
 pn=phoneMatrix  
   
 for(i in 2:n)  
 {  
 pd=pn  
 pn=phoneMatrix%\*%pd  
 print(i)  
 round(pn,4)  
 print(pn)  
 }  
  
}  
  
fvalue(phoneMatrix,20)

## [1] 2  
## Off On Lock Unlock  
## Off 0.2233260 0.58606766 0.0030294 0.18757695  
## On 0.4353438 0.22164807 0.2692008 0.07380735  
## Lock 0.4459013 0.03084298 0.0168102 0.50644551  
## Unlock 0.1133061 0.27676810 0.4781700 0.13175584  
## [1] 3  
## Off On Lock Unlock  
## Off 0.4178300 0.1140543 0.1326556 0.33546001  
## On 0.1697045 0.4158237 0.2585942 0.15587761  
## Lock 0.4244165 0.2328045 0.2648654 0.07791364  
## Unlock 0.2456048 0.5182689 0.0673038 0.16882243  
## [1] 4  
## Off On Lock Unlock  
## Off 0.3276198 0.3127937 0.2481910 0.1113955  
## On 0.3221165 0.3329993 0.1008044 0.2440798  
## Lock 0.1778877 0.4088511 0.2521034 0.1611579  
## Unlock 0.3739388 0.1789641 0.1458893 0.3012078  
## [1] 5  
## Off On Lock Unlock  
## Off 0.2401261 0.3670927 0.1946062 0.1981750  
## On 0.3518951 0.2427426 0.1913372 0.2140252  
## Lock 0.3230651 0.3300707 0.1056653 0.2411989  
## Unlock 0.3296146 0.3076149 0.2221197 0.1406509  
## [1] 6  
## Off On Lock Unlock  
## Off 0.3326293 0.2998232 0.1426349 0.2249126  
## On 0.2895107 0.3332533 0.2090257 0.1682103  
## Lock 0.3483911 0.2472542 0.1919007 0.2124540  
## Unlock 0.2606021 0.3466849 0.1957910 0.1969220  
## [1] 7  
## Off On Lock Unlock  
## Off 0.3226675 0.2829463 0.1975818 0.1968044  
## On 0.2938316 0.3251745 0.1719694 0.2090245  
## Lock 0.2902886 0.3325127 0.2069443 0.1702543  
## Unlock 0.3225098 0.3071372 0.1547976 0.2155554  
## [1] 8  
## Off On Lock Unlock  
## Off 0.2938960 0.3280464 0.1916645 0.1863931  
## On 0.3219991 0.2965400 0.1745360 0.2069249  
## Lock 0.2950865 0.3237035 0.1724315 0.2087786  
## Unlock 0.3187423 0.2901130 0.1915708 0.1995740  
## [1] 9  
## Off On Lock Unlock  
## Off 0.3056401 0.3123827 0.1745742 0.2074030  
## On 0.3075480 0.3074349 0.1912675 0.1937496  
## Lock 0.3211539 0.2973333 0.1752814 0.2062314  
## Unlock 0.2991833 0.3215222 0.1893329 0.1899616  
## [1] 10  
## Off On Lock Unlock  
## Off 0.3150305 0.3024530 0.1815502 0.2009664  
## On 0.3022793 0.3170942 0.1826835 0.1979430  
## Lock 0.3073661 0.3077904 0.1907654 0.1940781  
## Unlock 0.3055169 0.3122518 0.1777149 0.2045164  
## [1] 11  
## Off On Lock Unlock  
## Off 0.3055941 0.3111517 0.1871281 0.1961261  
## On 0.3097628 0.3079093 0.1795539 0.2027741  
## Lock 0.3026897 0.3166058 0.1825754 0.1981291  
## Unlock 0.3127357 0.3050213 0.1814771 0.2007659  
## [1] 12  
## Off On Lock Unlock  
## Off 0.3057497 0.3128879 0.1815229 0.1998395  
## On 0.3094388 0.3078582 0.1839991 0.1987039  
## Lock 0.3096903 0.3079569 0.1797977 0.2025552  
## Unlock 0.3065830 0.3103485 0.1857650 0.1973035  
## [1] 13  
## Off On Lock Unlock  
## Off 0.3093680 0.3081180 0.1816153 0.2008986  
## On 0.3062633 0.3114488 0.1838066 0.1984813  
## Lock 0.3092907 0.3080387 0.1839560 0.1987146  
## Unlock 0.3063000 0.3120546 0.1821103 0.1995351  
## [1] 14  
## Off On Lock Unlock  
## Off 0.3080800 0.3094489 0.1837646 0.1987065  
## On 0.3076897 0.3102582 0.1819204 0.2001317  
## Lock 0.3063517 0.3113638 0.1837187 0.1985658  
## Unlock 0.3087648 0.3088034 0.1819422 0.2004896  
## [1] 15  
## Off On Lock Unlock  
## Off 0.3069727 0.3108082 0.1829995 0.1992195  
## On 0.3084326 0.3091255 0.1827677 0.1996743  
## Lock 0.3077172 0.3102131 0.1819729 0.2000968  
## Unlock 0.3080675 0.3095206 0.1834063 0.1990056  
## [1] 16  
## Off On Lock Unlock  
## Off 0.3079745 0.3098079 0.1823418 0.1998758  
## On 0.3075788 0.3100961 0.1832090 0.1991161  
## Lock 0.3083857 0.3091791 0.1827840 0.1996512  
## Unlock 0.3072358 0.3105035 0.1829921 0.1992686  
## [1] 17  
## Off On Lock Unlock  
## Off 0.3080359 0.3095787 0.1829368 0.1994486  
## On 0.3075775 0.3101800 0.1827018 0.1995407  
## Lock 0.3075847 0.3100942 0.1831811 0.1991400  
## Unlock 0.3078770 0.3098883 0.1824981 0.1997367  
## [1] 18  
## Off On Lock Unlock  
## Off 0.3076052 0.3101054 0.1829733 0.1993160  
## On 0.3079429 0.3097539 0.1827010 0.1996021  
## Lock 0.3075951 0.3101585 0.1827053 0.1995411  
## Unlock 0.3079643 0.3096787 0.1828789 0.1994781  
## [1] 19  
## Off On Lock Unlock  
## Off 0.3077361 0.3099907 0.1827175 0.1995557  
## On 0.3078011 0.3098736 0.1829181 0.1994072  
## Lock 0.3079337 0.3097627 0.1827115 0.1995921  
## Unlock 0.3076724 0.3100324 0.1829308 0.1993645  
## [1] 20  
## Off On Lock Unlock  
## Off 0.3078705 0.3098195 0.1827952 0.1995148  
## On 0.3077039 0.3100103 0.1828339 0.1994520  
## Lock 0.3077973 0.3098793 0.1829126 0.1994107  
## Unlock 0.3077411 0.3099773 0.1827574 0.1995242

After raising transition probability matrix to a high enough power, which is 20, we can see that the steady-state distribution is 𝛱 = [ 0.31 0.31 0.18 0.2 ].

#Number of visits in 3 steps  
v=diag(4)+phoneMatrix+phoneMatrix%\*%phoneMatrix+phoneMatrix%\*%phoneMatrix%\*%phoneMatrix  
v

## Off On Lock Unlock  
## Off 1.6462560 1.030522 0.7296850 0.5935370  
## On 1.0582482 1.657772 0.5277950 0.7561850  
## Lock 0.8986178 1.220047 1.2816756 0.5996592  
## Unlock 1.1639109 0.930837 0.5454738 1.3597783

Suppose the screen is off at initial, then the expected number of total visits to Off, On, Lock and Unlock are 1.6462560, 1.030522, 0.7296850 and 0.5935370 respectively.

Suppose the screen is On at initial, then the expected number of total visits to Off, On, Lock and Unlock are 1.0582482, 1.657772, 0.5277950 and 0.7561850 respectively.

Suppose the screen is Lock at initial, then the expected number of total visits to Off, On, Lock and Unlock are 0.8986178, 1.220047, 1.2816756 and 0.5996592 respectively.

Suppose the screen is Unlock at initial, then the expected number of total visits to Off, On, Lock and Unlock are 1.1639109, 0.930837, 0.5454738 and 1.3597783 respectively.

#Reachability  
p<-phoneMatrix  
p1=p  
p2=p1%\*%p1  
p3=p2%\*%p1  
p4=p2%\*%p2  
p5=p3%\*%p2  
p6=p3%\*%p3  
p7=p5%\*%p2  
p8=p4%\*%p4  
p9=p5%\*%p4  
p10=p5%\*%p5  
p11=p10%\*%p1  
p12=p10%\*%p2  
p13=p10%\*%p3  
p14=p10%\*%p4  
p15=p10%\*%p5  
p16=p10%\*%p6  
p17=p10%\*%p7  
p18=p10%\*%p8  
p19=p10%\*%p9  
p20=p16%\*%p4  
p20

## Off On Lock Unlock  
## Off 0.3078705 0.3098195 0.1827952 0.1995148  
## On 0.3077039 0.3100103 0.1828339 0.1994520  
## Lock 0.3077973 0.3098793 0.1829126 0.1994107  
## Unlock 0.3077411 0.3099773 0.1827574 0.1995242

round(p20,2)

## Off On Lock Unlock  
## Off 0.31 0.31 0.18 0.2  
## On 0.31 0.31 0.18 0.2  
## Lock 0.31 0.31 0.18 0.2  
## Unlock 0.31 0.31 0.18 0.2

I=diag(4)  
sum=I+p1+p2+p3+p4+p5+p6+p7+p8+p9+p10+p11+p12+p13+p14+p15+p16+p17+p18+p19  
round(sum,3)

## Off On Lock Unlock  
## Off 6.551 6.028 3.708 3.713  
## On 6.020 6.600 3.393 3.988  
## Lock 5.729 6.261 4.220 3.790  
## Unlock 6.147 5.781 3.469 4.603

is.accessible(mcphone)

## Off On Lock Unlock  
## Off TRUE TRUE TRUE TRUE  
## On TRUE TRUE TRUE TRUE  
## Lock TRUE TRUE TRUE TRUE  
## Unlock TRUE TRUE TRUE TRUE

All states are reachable from On, Off, Lock and Unlock.

#Summary  
summary(mcphone)

## Phone Markov chain that is composed by:   
## Closed classes:   
## Off On Lock Unlock   
## Recurrent classes:   
## {Off,On,Lock,Unlock}  
## Transient classes:   
## NONE   
## The Markov chain is irreducible   
## The absorbing states are: NONE

In this Markov model, all states communicate with each other and they form a closed class = {Off, On, Lock, Unlock}. It is irreducible as the closed class is the set of states in its state space.

Absorbing state is once the chain reaches the state, it stays there forever and impossible to leave. There is no absorbing state in this Markov model.

transientStates(mcphone)

## character(0)

recurrentClasses(mcphone)

## [[1]]  
## [1] "Off" "On" "Lock" "Unlock"

Recurrent class is if a state is recurrent, then all states in the class must be recurrent. So, the recurrent class in this Markov model is {Off, On, Lock, Unlock}.

#Periodity  
period(mcphone)

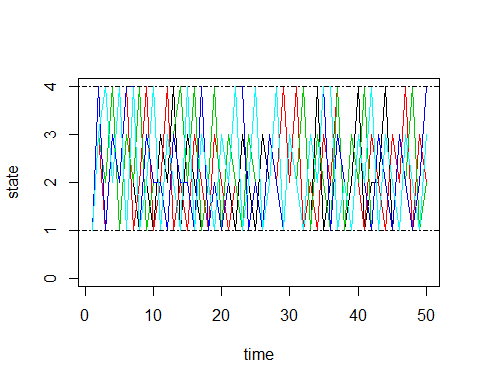
## [1] 1

A state is said to have a period d when the chain can only revisit it after a multiple of d steps. Since d=1, the state is aperiodic.

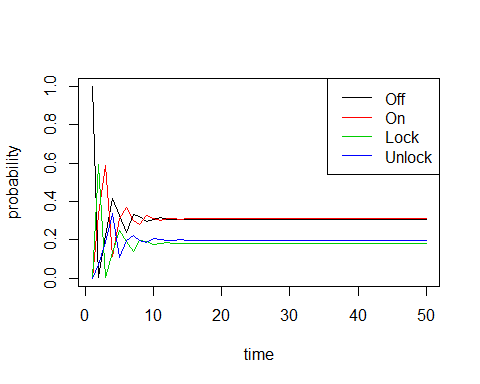
#Limiting distribution  
#Long term visiting rates  
e=matrix(c(1,1,1,1),nrow = 1, byrow = T)  
I=diag(4)  
E=matrix(1,4,4)  
e%\*%solve(I+E-phoneMatrix)

## Off On Lock Unlock  
## [1,] 0.3077797 0.3099211 0.1828211 0.1994782

# simulate discrete Markov chains  
run.mc.sim <- function( P, # probability transition matrix  
 num.iters=50,   
 num.chains=150 )  
 {  
   
 # number of possible states  
 num.states <- nrow(P)  
   
 # states X\_t for all chains  
 states <- matrix(NA, ncol=num.chains, nrow=num.iters)  
   
 # probability vectors pi^n through time  
 all\_probs <- matrix(NA, nrow=num.iters, ncol=num.states)  
   
 # forces chains to start in state 1  
 pi\_0 <- c(1, rep(0, num.states-1))  
  
 # initialize variables for first state   
 P\_n <- P  
 all\_probs[1,] <- pi\_0  
 states[1,] <- 1  
  
 for(t in 2:num.iters) {  
   
 # pi^n for this iteration  
 pi\_n <- pi\_0 %\*% P\_n  
 all\_probs[t,] <- pi\_n  
   
 for(chain\_num in seq\_len(num.chains)) {  
 # probability vector to simulating next state   
 p <- P[ states[t-1,chain\_num], ]  
 states[t,chain\_num] <- which(rmultinom(1, 1, p) == 1)  
 }  
   
 # update probability transition matrix  
 P\_n <- P\_n %\*% P  
 }  
 return(list(all.probs=all\_probs, states=states))  
}  
  
P<-phoneMatrix  
sim1 <- run.mc.sim(P)  
states <- sim1[[2]]  
matplot(states[,1:5], type='l', lty=1, col=1:5, ylim=c(0,4), ylab='state', xlab='time')  
abline(h=1, lty=4)  
abline(h=4, lty=4)



all.probs <- sim1[[1]]  
matplot(all.probs, type='l', col=1:4, lty=1, ylab='probability', xlab='time')  
legend('topright', c('Off', 'On', 'Lock','Unlock'), lty=1, col=1:4)

 The probabilities quickly converge to the stationary distribution, ℼ as we found above.

#First reaching time  
 meanFirstPassageTime(mcphone)

## Off On Lock Unlock  
## Off 0.000000 1.844632 2.799176 4.464597  
## On 1.725823 0.000000 4.524999 3.086001  
## Lock 2.670651 1.092615 0.000000 4.077800  
## Unlock 1.312040 2.641293 4.111216 0.000000

#Expected return time  
#for Off  
Eoff=1/0.3077797  
Eoff

## [1] 3.249077

#for On  
Eon=1/0.3099211  
Eon

## [1] 3.226628

#for Lock  
Elock=1/0.1828211  
Elock

## [1] 5.469828

#for Unlock  
Eunlock=1/0.1994782  
Eunlock

## [1] 5.013079