Statistics 406 Lab 11
GSI: Adam Rothman
Tuesday, November 28

1. Imagine that you flip a possibly biased coin with $P(\text{heads}) = p$ until a heads appears. What is the average number of flips required to get the first heads. The mathematical solution is to notice that the number of flips N is a geometric random variable with parameter p, and $E[N] = 1/p$. Verify this using a simulation in R.

   Solution:

```
## n is maximum length of a sequence of bets for
## our simulation
n <- 1000
reps <- 1e3

## Probability of getting "heads"
p <- 0.7

## Generate a very large sequence of coin flips, reps times
BT <- array( 1*(runif(n*reps) < p), c(n, reps) )

## Initialize the memory that will store the number
## of coin flips needed for the first "heads" to appear
## will need to be a reps long vector
numFlips <- array(1, reps)

## Loop through the repetitions
for ( r in 1:reps )
{
  ## Loop through the flips
  for ( i in 1:n )
  {
    if ( BT[i, r] )
    {
      ## We got a "heads" for this sequence
      break
    }
    ## We got tails on this flip
    numFlips[r] <- numFlips[r] + 1
  }
}
# show the result
print(mean(numFlips))
```

2. Consider a gambler that makes a sequence of bets. She stops when she wins k consecutive bets. Assume the bets are independent, find her expected winnings if for each bet, she wins 1 unit with probability p, and loses one unit with probability $1 - p$.

Solution:

Using tools from probability, let W be her winnings, so $W = \sum_{i=1}^{N} X_i$. N is a stopping time since it is a gambling strategy, thus applying Wald's Equation, $E[W] = E[X_1]E[N]$, thus if $p = 0.5$,$E[W] = 0$.

```
## n should be large enough so that
## k consecutive successes will occur before
## the nth bet
n <- 1000
reps <- 1e3
p <- 0.5

## Generate a very large sequence of bets, reps times
BT <- array( 1*(runif(n*reps) < p), c(n, reps) )

## Number of consecutive wins for gambler to quit
k <-3

## Initialize the memory for the simulations
numTrials <- array(0, reps)
winnings <- array(0, reps)

## A sequence of k ones used to see if she quits
kones <- array(1, k)

## Loop over the repetitions
for ( r in 1:reps )
{
  ## Loop through the sequence of bets
  for ( i in 1:n )
  {
     ## Check if we haven't seen k consecutive wins by n
     if ( (i+k-1) <= n )
     {

       if ( sum(BT[i:(i+k-1), r] == kones) == k && ((i+k-1) <= n) )
       {
         ## she got k consecutive wins
```

```
          winnings[r] <- winnings[r] + k
          numTrials[r] <- numTrials[r] + k
          break
      }else if ( BT[i,r] == 1 )
      {
        ## she won this game, but it isn't the kth consec. win
        winnings[r] <- winnings[r] + 1
      }else
      {
        #she lost this game
        winnings[r] <- winnings[r] - 1
      }
      numTrials[r] <- numTrials[r]+1
    }else
    {
      ## We did not see k consecutive wins
      ## Thus we need to make n larger for the
      ## simulation to work properly
      print("Error n is not large enough")
      break
    }
  } ##END n LOOP
}## END REPS LOOP
## Show our average number of trials and average
## amout of money she won:
print(c(mean(numTrials), mean(winnings)) )
```

3. Simulate a deck of cards in R, find the probability of drawing an Ace off the top of a shuffled deck. This probability is obviously $4/52 = 1/13$.

   Solution:

```
## Create the deck of 52 cards
## Each row of the deck matrix is one card in the deck
deck <- array(0, c(52,2) )

## The first column of the deck matrix is the cards number
deck[,1]=c("Ace",2,3,4,5,6,7,8,9,10, "Jack", "Queen","King")

## The second column of the deck matrix is the cards suit
deck[,2]=c("Clubs", "Diamonds", "Hearts", "Spades")

## We now have our deck as a matrix
## So the third card in our deck is
## deck[3,], etc.

## Compute the prob of drawing an Ace off
## the top of the deck
reps <- 1e6
gotIt <- 0
for (r in 1:reps )
{
  ## shuffle the deck
  deck <- deck[sample(52),]

  ## Add one to our counter if we draw an
  ## Ace off of the top of the deck
  gotIt <- gotIt+1*(deck[1,1] == "Ace" )
}
prb <- gotIt/reps
print(prb)
```