STAT380: Assignment 3

Ash Midgley

April 15, 2016

Question 1

```
ci.int = function(xbar,n,sigma){
  # finding the confidence interval
  tmp = cbind(xbar-1.96*sigma/sqrt(n), xbar+1.96*sigma/sqrt(n))
  return(tmp)
}

q1 = function(){
  parameter = 1/3
  mu = (1-parameter)/parameter
  nvec = c(10,100,200,300,400,500,600,700,800,900,1000)
  nlength = length(nvec)
  m = 10000
  covvec = rep(NA, nlength)
  for (j in 1:nlength) {
    mat = matrix(rgeom(m*nvec[j],parameter), m,nvec[j])
    xbar = rowMeans(mat)
    sig = apply(mat, 1, sd)
    confval = ci.int(xbar, nvec[j], sig)
    inOrOut = (confval[,1] < mu) & (confval[,2] > mu)
    covvec[j] = mean(inOrOut)
  }
  plot(nvec, covvec, pch=20, xlab="n", ylab="coverage value",
      main="geometric distribution of n")
}
```

We can see from the plot that as we increase the sample size, the coverage probability increases. The plot also shows us that we generally would need a sample size of at least 100 to get a respectable coverage probability (around 0.94).

Question 2

(a)

```
q2 = function(){
  #female
  meanf = 200
  sigmaf = 10
  nf = 20
  #male
  sigmam = 15
  nm = 20
  meanm = seq(175, 225)
  meanmlength = length(meanm)
  #print(meanm)
  #print(meanmlength)
  m = 1000
  #t test matrix
  tmat = matrix(NA, meanmlength, m)
  #wilcox test matrix
  wilcmat = matrix(NA, meanmlength, m)
  wilcPowCCount=0
  tPowCount = 0
  for(i in 1:meanmlength){
    for(j in 1:m){
      fvalue = rnorm(nf, meanf, sigmaf);
      mvalue = rnorm(nm, meanm[i], sigmam);
      ttest = t.test(fvalue, mvalue)
      wilctest = wilcox.test(fvalue, mvalue)
      ttpVal = ttest$p.value
      wilctpVal = wilctest$p.value
      tmat[i,j] = ttpVal
      wilcmat[i,j] = wilctpVal
      if(ttpVal < 0.05){
        tPowCount = tPowCount+1
      }
      if(wilctpVal < 0.05){
          wilcPowCCount = wilcPowCCount + 1
      }
    }
  }
  print(tmat)
  print(wilcmat)
  tPow = tPowCount/m
  wPow = wilcPowCCount/m
  print(tPow)
  print(wPow)
}
```

By comparison, the 2 power values for the tests are very similar.

Question 3

```r
library(plyr)

findWinner = function(i, j){
  p = plogis(-(i-j)/8)
  win = sample(c("i","j"),1, replace = TRUE, prob=c(p, 1-p))
  #print(win)
  if(win == "i"){
    return(i)
  }else{
    return(j)
  }
}

q3 = function(){
  i = 1
  m=10000
  rounds = c(8,4,2,1)
  winvec = rep(NA, m)
  for(x in 1:m) {
    seedVec = c(1, 16, 8, 9, 4, 13, 5, 12, 11, 6, 14, 3, 10, 7, 15, 2)
    for (i in 1:length(rounds)) {
      nextRound = rep(NA, rounds[i])
      for (j in 1:length(nextRound)) {
        winner = findWinner(seedVec[j * 2 - 1], seedVec[j * 2])
        nextRound[j] = winner
      }
      seedVec = rep(nextRound)
    }
    winvec[x] = seedVec[1]
  }
  winTable = table(winvec)
  t4WinP = sum(winTable[4]/m)
  print(t4WinP)
  t10HigherWinP = sum(winTable[10:16]/m)
  print(t10HigherWinP)
  hist(winvec,breaks=0.5:16.5,freq=FALSE,xlab="Team's seed value",
       ylab="Probability of winning tournament",
       main = "Probability of team's winning tournament")
}
```