

Homework 4: FISH 553 Advanced R

Open a new script in R and put the following information at the top using comments:

```
# Name: First Last  
# Homework 4
```

Complete the tasks below and be sure to label each question with comments as before. When your script is complete, save it as LastName_Homework4.R, then clear your workspace (Workspace/Clear all) and run through your script again to make sure you don't have any 'object not found' errors. Then go to the course website on Canvas to submit your R script.

Question 1 (60 points)

Question 1a) [30 points] One person is randomly shuffling a deck of 52 cards and taking 7 cards. What is the probability that they select four aces? To answer, simulate a pack of 52 cards (four each of 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K, A), and use repeated resampling. Pay careful attention to whether you should sample with replacement or without replacement. (Advanced: you can check your answers using a hypergeometric distribution.)

Question 1b) [30 points] Generalize the game so that $1 \leq n \leq 7$ players are each dealt 7 cards. What is the probability that one or more players will receive four aces? (This is a little tricky since you need to make sure that different players can't be dealt the same card.)

Question 2 (40 points)

The following piece of code is slow to run. Understand how the code works, and rewrite it to make it return the answer faster. Use microbenchmark() to test how fast your code is compared to the original. (A copy of the code is in "HW4 slowcode.r" on the website.)

```
#Question 2  
#SLOW VERSION of code  
func1 <- function(xvec) {  
  nitens <- length(xvec)  
  sumofdata <- 0  
  for (i in 1:nitens) {  
    sumofdata <- sumofdata + xvec[i]  
  }  
  mean.is <- sumofdata/nitens  
  print(mean.is) #just checking interim output  
  return(mean.is)  
}  
  
func2 <- function(xarray) {  
  ncol.s <- ncol(xarray)  
  nrow.s <- nrow(xarray)  
  results <- vector(length=ncol.s)  
  for (i in 1:ncol.s) {  
    temp.data <- xarray[1:nrow.s,i]  
    mean.of.data <- func1(temp.data)  
    results[i] <- mean.of.data  
  }  
  print(results)  
  num.bigger <- 0  
  for (j in 1:ncol.s) {  
    if (results[j] > 0) {  
      num.bigger <- num.bigger + 1  
    }  
  }  
}
```

```
    }  
    prop.bigger <- num.bigger/ncols  
    return(prop.bigger)  
}  
niter <- 1000  
xarray <- matrix(data=rnorm(niter), ncol=100)  
func2(xarray=xarray)
```