**IST687 – Homework 4: Creating a Function and Sampling**

**Reminders of things to practice from previous weeks:**

Descriptive statistics mean( ) max( ) min( )

Sequence operator : (For example, 1:4 is shorthand for 1, 2, 3, 4)

Create a function myFunc <- function(myArg) { }

?command Ask R for help with a command

**This week:** Sampling is a process of drawing elements from a larger set. In data science, when analysts work with data, they often work with a sample of the data, rather than all of the data (which we call the population), because of the expense and inconvenience of obtaining all of the data. Imagine trying to collect data on the number of apps installed on every smart phone in the world. It would be much easier, and nearly as good, to work with just a sample of smart phones and calculate some statistics from that sample.

One must be careful, however, because statistics from a sample usually fail to exactly match the characteristics of the population. The goal of this homework is to sample several times from a data set and explore the meaning of the results. Before you get started make sure to read Chapters 8 and 9 of *An Introduction to Data Science*. Don’t forget your comments!

**Part A: Write a function to reveal the distribution of a vector of numeric values**

1. Create a new function ‘printVecInfo’ and have it take one numeric vector as its input argument.
2. Make the function print the following information for the vector supplied in the argument:
   1. Mean
   2. Median
   3. Min & Max
   4. Standard deviation
   5. 0.05 and 0.95 quantiles (Use the quantile( ) function)
3. Test the function with this vector: testVector <- 1:10. Results should look something like this:

[1] 5.5

[1] 10

[1] 1

[1] 3.02765

5% 95%

1.45 9.55

1. Add labels to each element of the function’s output.

**Part B: Read the census dataset**

1. Read in the Census dataset   
   *Hint: reuse the function you created in HW 3*

**Part C: Sample from the state population data frame**

1. Sample 20 observations from states$population and use printVecInfo( ) to display the characteristics of the resulting sample, and then display the results as a histogram.
2. Repeat step five two more times. Each time that you create a sample, run the resulting vector through printVecInfo( ) and create a histogram.
3. Using a block comment, explain in a comment why each result is different.

**Part D: Replicate the sampling**

1. Use the replicate function, to replicate the sampling (described in step 5 above). Replicate the sampling 2000 times, then use printVecInfo( ) to display the characteristics of the resulting replicated sample, and then display the results as a histogram.
2. Repeat step 8 two more times. Each time that you create the replicated sample, run the resulting vector through printVecInfo( ) and create a histogram.
3. Using a block comment, explain why the histograms generated in Part C are different than the histograms generated in Part D