**IST 687 PREP EXERCISE 07**

**Student Name: Bhavish Kumar**

**Prep Exercise No: 07**

**Date Due: 9th October 2019**

1. **Getting Ready: Load and repair median income data**
   1. Download the provided MedianZIP.csv file from Blackboard and read into R-studio into a dataframe called “mydata”. ***HINT: Use read\_csv() to simplify later steps!***

mydata <- read.csv("MedianZIP.csv", stringsAsFactors = FALSE)

### Reading MedianZIP csv into mydata dataframe

mydata$Mean <- as.numeric(mydata$Mean)

### Converting Mean column to numeric values

* 1. **Cleaning up the NAs:** Find and fix the missing data in the Mean column by substituting the value from the Median column in place of the missing mean values. Explain why the median is a reasonable replacement for the mean.

which(is.na(mydata$Mean))

### Obtaining row indexes of rows containing NA values in Mean column

mydata$Mean[is.na(mydata$Mean)] <- mydata$Median[is.na(mydata$Mean)]

### Substituting the NAs with corresponding value from Median column

Median is a reasonable replacement for the mean value because median is a good measure of central tendency and moreover median is not affected by outliers or skewness in the data. Hence since Mean is not available the best possible replacement is median.

* 1. Examine the data with View( ) and add comments explaining what each column contains. Add a comment explaining why the first 2391 zip codes look weird.

View(mydata)

str(mydata)

The data contains 5 columns which are zip, Median, Mean & Pop which contain the zipcode of the place, the Median income of the location, the Mean income of the location and the population of the location respectively.

The first 2391 zip codes look weird because the zip codes contain values of integer data type and also they contain only 4 digits, whereas ideally a zipcode should contain 5 digits as that is the appropriate zip code format.

1. **Merge the median income data with detailed zipcode data**
   1. Code and execute the commands below. Write a paragraph below that explains what this code does.

install.packages("zipcode")  
library(zipcode)  
mydata$zip <- clean.zipcodes(mydata$zip)  
data(zipcode)  
dfNew <- merge(mydata, zipcode, by="zip")

The above code installs the zipcode package at first and uses the clean.zipcodes() function to get all the values in the zip column of mydata dataframe to the correct zipcode format of 5 digits and also converts the datatype to character. It restores leading zeroes to all the zipcodes that have only 4 digits.

The data() function loads in built datasets. The inbuilt zipcode dataset is being used.

A new dataframe is obtained by merging the mydata dataframe and zipcode dataframe on the zip column

1. **Merge the new dataset with stateNameDF data**
   1. Create a new dataframe with the following code:

stateNameDF <- data.frame(state=state.abb, stateName=state.name, center=state.center)

stateNameDF$stateName <- tolower(stateNameDF$stateName)

* 1. Comment each line of the code to explain what it is doing

### 1. Creating a new stateNameDF using the data.frame function and including the vectors state, stateName & center. The state vector gets its values from the abb(abbreviation) column of the inbuilt dataset state. Likewise the vectors stateName and center also get their values from state.name and state.center respectively.

### 2. The values in stateName column are reduced to lowercase using the tolower() function.

* 1. Using steps similar to step 2 create a new dataframe that contains our previous information and the information from the stateNameDF.

new\_df <- merge(dfNew, stateNameDF, by = "state")

### Merging the dfNew dataset and stateNameDF on state column to obtain all the columns.

1. **Examine your new df with the View command.** I

Include a screen shot of the first 10 rows of data and all of your columns.

View(new\_df)

