STA 6923 Homework 2, Fall 2021

Due date: Tuesday, September 14, 11:59 pm

Instruction: Answer your questions item by item. Write your conclusions clearly and concisely. You need to provide two files as follows.

- Your R code file, with comments on your code
- Your full homework report, written by R markdown pdf (or html) file. Before each output, you should keep your code line as well by using "Echo = TRUE", or you can define it in the beginning.

From text book (ISL)

Chapter 2, page 52:

- Conceptual problems:
  - 1, 2, 3
- Applied problems:
  - 10

Not from the text book

1. Suppose you have following functions. Write an R function to each of them and then make a plot for each one

   a) $f(x) = 2 + 3x^2 - x$, in the range of $(-10, 10)$.

   b) $f(x) = \frac{1}{B(2,3)} x(1-x)^2$, for $0 < x < 1$, where $B(\alpha, \beta)$ is a bete function, in the range of $(0, 1)$

2. Create a data frame with the following command

   >set.seed(123)

   >df = data.frame(x1 = rnorm(10), x2 = rpois(10,3), x3 = runif(10,-1,1), x4 = rgamma(10,2,3))

   a) Obtain the means of all columns using apply().

   b) Add another column, named $c5$, which is 1 for all $x1 \geq 0$ and 0 otherwise.

   c) Draw boxplots of x_2 for different $c5$ values.

3. In this problem, you search the internet using "auto_mpg dataset" to find an automobile mpg data. Most likely you would be able to find it in either at "kaggle", or "UCI Machine Learning Repository." Note that you do not use the original data.

   a) Create a new project and import the data into your RStudio. Show the first 3 rows of the data by using *head* command.

   b) Check the classes of your variables by using *sapply* command. What are the classes of **horsepower**, **model_year** and **name**?

   c) From the original data, **horsepower** is supposed to be numeric. Do you see any problem? In R, any missing value is labeled as "NA". Try to clean the data (actually the **horsepower** column) and replace any character to "NA".[1]

[1] To check whether there is non-numeric data in column x, you can use any(is.na(as.numeric(x))), or if you want to know

d) Do a summary analysis of the data (numeric variables) by checking each variable's range, extreme values, mean, median, standard deviation, etc. Check the correlations among the variables and plot pairwise graph between each two variables by using command *pairs*.

e) Create a two-variable data, with only *acceleration* and *mpg* in it. Make a scatter plot between them by using *mpg* as y-axis variable. Do you see strong correlation between the two variables? In addition, what is a correlation?

f) Run a linear regression between the variables in part e) and use *mpg* as the response variable, *acceleration* as the predictor. What's your conclusion for this analysis? In addition, add the regression line to the plot in part e).

g) Using the regression in part f), make a prediction of *mpg* for each *acceleration* values in the data. Draw a scatter plot between the original *mpg* and the predicted *mpg*. Comment.

h) MSE is an abbreviation for **Mean Squared Error**, which is the average of the squared differences between the estimated and the truth value (or observed value). For the results in part g), treat the original *mpg* as true values, and predicted *mpg* as estimates. Find the MSE of this prediction.

i) The Locally Estimated Scatterplot Smoothing, or LOESS, is a moving regression to fit data more smoothly. Use the **loess** function in R to make a LOESS regression between *acceleration* and *mpg*. What is the MSE of the prediction in this case? Comment, including the results in part h). Add the LOESS regression line into the graph you drew for part h)

j) Using **summary** to check the result of your LOESS regression in part i). The **span**, in the *Control settings*, is a smoothing parameter. Now try to run another (or more if you like) LOESS regression by adding **span** option in your loess command. Comment on the results.

which ones, use which(is.na(as.numeric(x))). In addition, if you want to change all the character in a numeric data to "NA", use command x = as.numeric(as.character(x)).