Homework: Ch 01

STAT 4510/7510

Due Thursday, January 27, 11:59 pm

Instructions: Please list your name and student number clearly. In order to receive credit for a problem, your solution must show sufficient detail so that the grader can determine how you obtained your answer.

Submit a single pdf file for your final outcome. There are several way to do that.

- Use MS Word or MS Powerpoint, then transform it to pdf - copy the codes and the output you obtained in RStudio and paste them to a MS Word or Powerpoint file. Once you complete the homework, you can transform those files to pdf using "Export" or "Save As".
- Use R Markdown - I don't recommend this option for a beginner of R, but if you would like to use more inherent features of RStudio to create a pdf document, please enroll in the course "Introduction_to_R_for_Statistical_Learning", by the link provided in our Canvas website, and watch videos in Module 11. For your information, this current document is written by R Markdown.

All R code should be included, as well as all output produced. Upload your work to the Canvas course site.

Problem 1

Complete 2.3 Lab: Introduction to R, found on pages 42 - 51. *(You are expected to simply work through the textbook lab as written and execute the commands. Include all commands and output in your homework submission.)*

Problem 2

The following codes simulate data shown in page 5 of lecture slide Chapter 2.

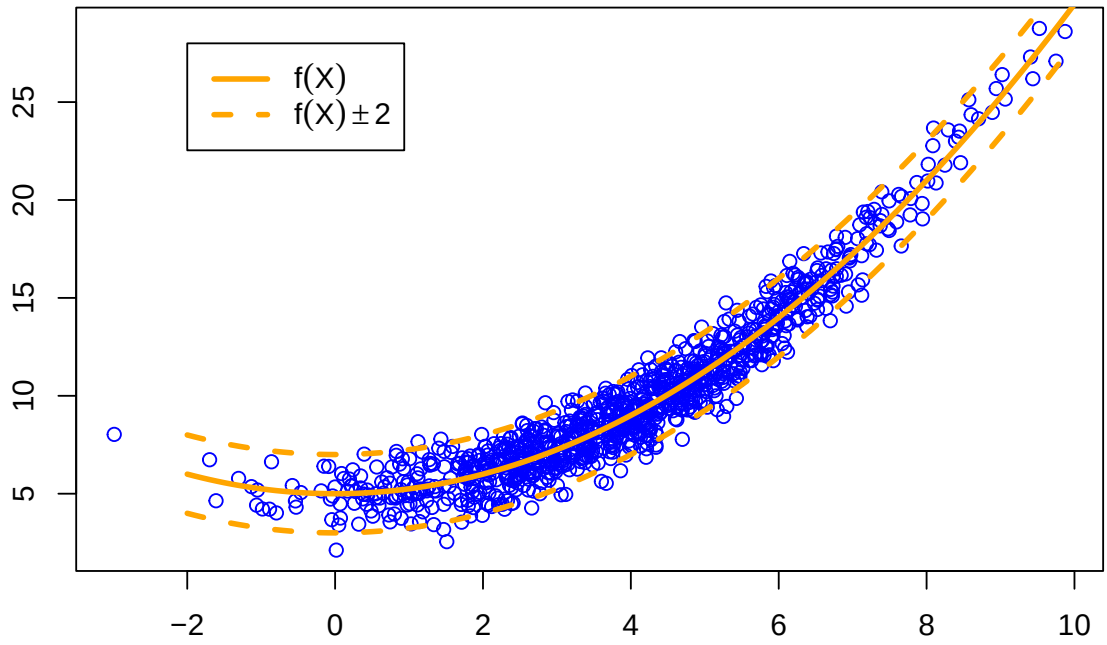
```
# observation
set.seed(45107510)
x <- rnorm(n=1000, mean=4, sd=2) # locations of x
ep <- rnorm(n=1000, mean=0, sd=1) # error terms
y <- 5+0.25*x^2+ep
p1 <- plot(x, y, xlab="", ylab="", col="blue")

# true underlying function
xGrid <- seq(-2, 10, by=0.1)
fTrue <- 5+0.25*xGrid^2
p2 <- lines(xGrid, fTrue, col="orange", lwd=3)

# plus minus two standard deviation
plusTwoSd <- fTrue + 2
minusTwoSd <- fTrue - 2
p3 <- lines(xGrid, plusTwoSd, col="orange", lwd=3, lty=2)
p4 <- lines(xGrid, minusTwoSd, col="orange", lwd=3, lty=2)

# legend
```

```
legend(-2, 28, legend=c(expression(f(X)), expression(f(X) %+-% 2)),
       col=c("orange", "orange"), lty=1:2, lwd=c(3,3))
```



a) When we want to know more about a function of `R`, we can use `?functionname`. In the code above, the function `rnorm` is used to create a sample with size 1000 from a normal distribution for the error term. Investigate this function by running `?rnorm` in your console. Explain what this function do and what each argument represent.

b) Suppose that we could find the estimated f using this data as $\hat{f}(X) = 4.8 + 0.27X^2$ by some statistical learning techniques. Suppose that we want to make a prediction of Y for a new value of X given as $X = 8.5$. Find the reducible error for the prediction?

c) The magnitude of the irreducible prediction error involved in this simulated data can be controlled by the argument `sd` (standard deviation) of `rnorm` function used for the error terms. Change `sd` to `2` and draw the plot again. What do you observe?

Problem 3

The file `Credit.csv` contains the information on customers of a credit card company.

a) Install and load the package `ISLR`. Find the description on the data set `Credit` by `?Credit`.

b) Load the data set to the session and call it `credit` so that you can see the data frame from the window of Global Environment. (You can directly read the data set from the `ISLR` package. But for consistency, please use the `csv` file provided by the Canvas website.)

c) Produce three scatter plots `Income` vs. `Balance`, `Age` vs. `Balance`, and `Limit` vs. `Balance` by using the function `plot`.

d) Produce `str` and summary table of the data set. Some variables in the data set have the type of `chr`. Change the types of such variables to factor using the `as.factor()` command. For example you can change the type of the variable `Own` to factor by

```
credit$Own <- as.factor(credit$Own)
```

e) Produce a `str` and summary table of the data set again. What do you see now for the factor variables?

f) Use the function `hist` to produce histograms of `Income` and `Age`. Add a title and axis labels to each plot, and use a different color for each histogram.

g) Use the function `boxplot` to produce boxplots of `Limit` and `Balance`. Add a title and axis labels to each plot, and use a different color for each boxplot.