Intro to Stats, Mini-Project 4

In this technology assignment mini-project, we will focus on how to use R to perform multiple regression analysis.

At the end, you will be asked to hand in your R script, complete with answers that have been written into comments. Details on how to do this are at the end of this document.

I anticipate this taking between 90-120 minutes, assuming no hiccups in your technology! [If there are technology hiccups, we'll figure it out as we go.]

Setting up the R Script

Create a new R Script and write the following commands at the top as a way to "title" your Script:

```
# #
# # Mini Project 4
# #
# # FirstName LastName
# #
# # TodaysDate
# #
```

Additionally, please load the `ggplot2` and the `dplyr` packages.

```
> library(ggplot2)
> library(dplyr)
```

I will ask you to use this R script when writing your commands. Save your RScript as "LastnameFirstname4.r" You can save the R script and open it later.

I will ask you to write every line of R code in the R script, and include commentary (#) where appropriate to explain your code or answer the questions in the technology project. This is the document I want you to hand in for this assignment. More details at the end of this project prompt.

First look at a new (and famous!) data set

We will now turn our attention to the data set `iris`, one of the more famous data sets in data analysis. The iris flower data set looks at three different species of Iris (Iris setosa, Iris virginica and Iris versicolor), and records four different measurements of different parts of the flowers (the length and the width of the sepals and petals, measured in centimeters).

The data set was introduced by the British statistician and biologist Ronald Fisher in his 1936 paper "The use of multiple measurements in taxonomic problems as an example of linear discriminant analysis."[1]

1. This data set is pre-loaded in R under the name `iris`. However, we will be manipulating this data set, so we should give it a unique name like `IrisData`. Try typing in the following command:

   > IrisData <-iris

2. Use your usual commands (like `names()`, `dim()`, and `head()`) to get a sense of the data set and its variables. Using the comments, list the 5 variables in this data set, and what kind of variable (quantitative or categorical) each variable is.

3. Let's begin with a simple linear regression, predicting Sepal Width as a function of Sepal Length.Try typing in the following commands:

   > table(IrisData$Species)
   > ggpplot(IrisData, aes(x=Sepal.Length, y=Sepal.Width)) +
   geom_point()+
   geom_smooth(method = "lm")
   > lm.Width.Length < - lm(IrisData$Sepal.Width ~ IrisData$Sepal.Length)
   > lm.Width.Length

[1] It's a little surprising that this data set is so famous - at least, it's surprising to me - but it's fame is due to the fact that it is an excellent data set to explore multiple regression concepts!

4. In comments: Describe your linear regression. What is the equation of the line? (Write it in the form "y = mx + b".) What is the slope of your line, and what does it mean (in real-world terms)? What is the intercept of your line, and what does it mean (in real-world terms)?

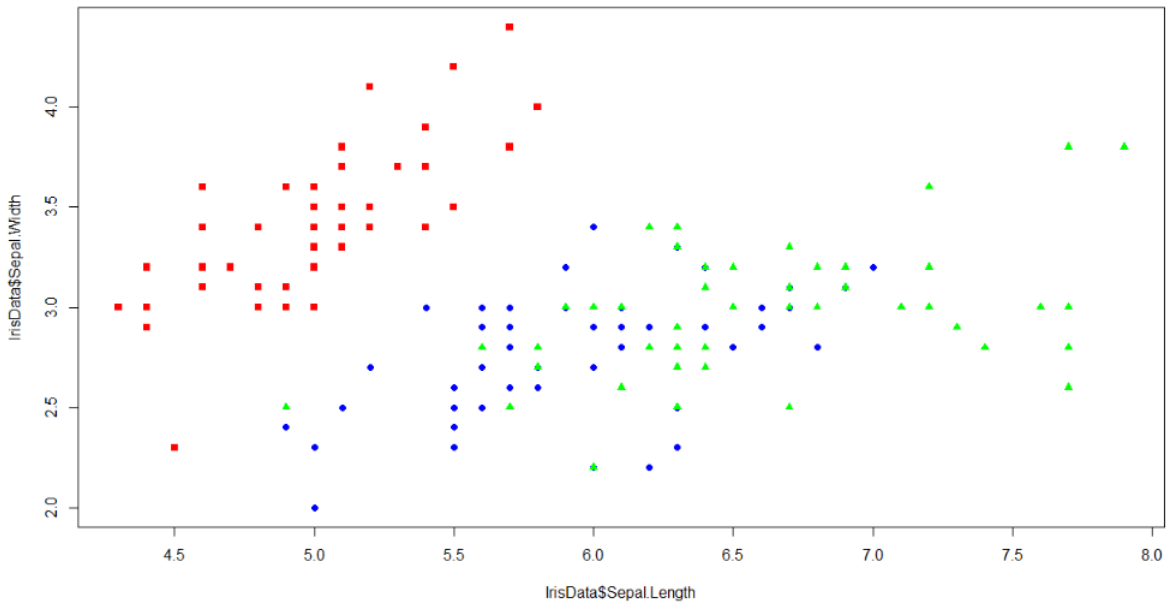
Motivating our decision to use Multiple Regression

To motivate our work for the rest of this sheet, we want to color-code the points in our scatterplot according to the species of iris. This will let us see that one type of iris has a much different pattern than the other two!

5. Try typing in the following command, which assigns colors to different species:

```
> ggpplot(IrisData, aes(x=Sepal.Length, y=Sepal.Width, color=Species)) +
      geom_point()
```

The scatterplot you just created should look (roughly) like this (the symbols might be different):



From our previous scatterplot, it's clear that one species (the species Setosa, represented by the red dots in the upper left) has a different width/length relationship than the other two species. **Our major goal for the rest of this sheet will be to design a multiple regression that predicts Sepal Width as a function of both the Sepal Length and the Iris species (whether or not it's Setosa)**.

Creating an indicator and interaction variable for the Setosa species

To perform an appropriate regression analysis, we will create an indicator and a interaction variable for the Setosa species. To do this, we must add new variables (ie, new columns) to our dataset using the `mutate` command from the `dplyr` package. However, we also want to learn how to use an "if/else" command in RStudio.

6. We want our first new column to be our indicator variable, meaning we want an entry in this column to be a "1" if the plant species is Setosa, and a 0 otherwise, This is an excellent place to use the `ifelse` command in R. Type in the following command:

   > IrisData <- mutate(IrisData, SetosaIndicator = ifelse(Species == "setosa", 1, 0))

   This says that we assign a number of the variable SetosaIndicator as follows: *IF* the species of Iris is "setosa", we assign a 1. *ELSE* (or otherwise), we assign a 0. This is exactly what we were hoping to accomplish!

7. Next, we want to create our **interaction** variable, which we will name SetosaInteraction. As a reminder, an interaction term is of the form (indicator)*(predictor), and so we achieve this by creating a new variable and assigning it to be the indicator variable times the predictor variable (in this case, sepal length):

   IrisData <- mutate(IrisData, SetosaInteraction = SetosaIndicator*Sepal.Length)

The Multiple Regression Analysis

8. Finally, we can finish our multiple regression analysis! To do this, enter the command:

   > mlm.width <- lm(IrisData$Sepal.Width~IrisData$Sepal.Length+
   IrisData$SetosaIndicator+IrisData$SetosaInteraction)

   > mlm.width

9. What are our three coefficients and our intercept? (write your answer as a comment)

10. Use the coefficients and the intercept to write an *equation* describing the relationship between our four variables. (something of the form "predicted sepal width = ...")

11. Suppose we want to predict the sepal width of a virginica or a virsicolor iris flower based only on its sepal length. What is the equation we would use? (Write your answer as a comment, rounding your slope and intercept to two decimal places. Briefly explain your answer in a comment.)

12. Suppose we want to predict the sepal width of a setosa iris based only on its sepal length. What is the equation we would use? (Write your answer as a comment, rounding your slope and intercept to two decimal places. Briefly explain your answer in a comment.)

13. Add on two `geom_abline()` commands to your ggplot, with slope and intercept specified, to verify that your answers to the previous two problems are correct! If done correctly, you should get two lines of best fit: one that describes the setosa irises, and one that describes the virginica and virsicolor irises.

A Second Multiple Regression Analysis – this one is your choice!

Now comes the fun part: repeat this entire process, using other variables of your choice to do some multiple regression analysis! This means

14. Explicitly identify which quantitative variable you want to estimate, and which quantitative variable you want to predict.

15. Create a scatterplot that compares two these two quantitative variables, with the predictor on the x-axis and the response variable on the y-axis, and with different colors used to represent different `Species`.

16. Create an appropriate indicator variable and interaction variable.

17. Run a multiple regression using the `lm` command. In the comments, write down what your coefficients are.

18. Calculate the equation of each of your two lines of best fit (with one line depending on `Species` being a 1, the other with `Species` being a 0.

19. Graph your two lines using the `abline` command, to see if they line up appropriately in your scatterplot!

I recommend estimating `Sepal.Width` based on `Petal.Width` and `Species`, but you are free to explore and compare any `Species` with two quantitative variables you want!

Handing in this Mini Project: I want you to try to write your work as an R script in R, and save/submit the script as your project.

Please address each question being asked by identifying it using comments. Finally, in comments at the bottom of your RScript, please write the SU Honor Pledge.

Save your RScript as "LastnameFirstname4.r" and submit that through Moodle.