CIND 123 - Data Analytics: Basic Methods Fall 2021 Assignment 3

Assignment 3 (10%)

[Alyzeh Jiwani]

[D20, 501106857]
Instructions

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. Review this website for more details on using R Markdown http://rmarkdown.rstudio.com.

Use RStudio for this assignment. Complete the assignment by inserting your R code wherever you see the string "#INSERT YOUR ANSWER HERE".

When you click the **Knit** button, a document (PDF, Word, or HTML format) will be generated that includes both the assignment content as well as the output of any embedded R code chunks.

Submit **both** the rmd and generated output files. Failing to submit both files will be subject to mark deduction.

Sample Question and Solution

Use `seq()` to create the vector $(2, 4, 6, \ldots, 20)$.

```
#Insert your code here.
seq(2,20,by = 2)
```

```
##  [1]  2  4  6  8 10 12 14 16 18 20
```

```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1
## v purrr   0.3.4
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(gridExtra)
```

```
##
## Attaching package: ’gridExtra’

## The following object is masked from ’package:dplyr’:
##
##     combine
```

```
library(cowplot)
library(dplyr)
library(broom)
library(ggpubr)
```

```
##
## Attaching package: ’ggpubr’

## The following object is masked from ’package:cowplot’:
##
##     get_legend
```

```
library(moments)
```

Note:

You will use ‘Admission_Predict.csv’ for Assignment-3. This dataset includes the data of the applicants of an academic program. Each application has a unique serial number, which represents a particular student. The dataset contains several parameters which are considered important during the application for Masters Programs. The parameters included are :

1) GRE Scores (out of 340)

2) TOEFL Scores (out of 120)

3) University Rating (out of 5)

4) Statement of Purpose (SOP) (out of 5)

5) Letter of Recommendation (LOR) Strength (out of 5)

6) Undergraduate GPA (out of 10)

7) Research Experience (either 0 or 1)

8) Chance of Admit (ranging from 0 to 1)

Download “Admission_Predict.csv” dataset and load it as ‘data’.

```
data <- read.csv('Admission_Predict.csv', header = TRUE, sep =',')
head(data)
```

```
##   Serial.No. GRE.Score TOEFL.Score University.Rating SOP LOR CGPA Research
## 1          1       337         118                 4 4.5 4.5 9.65        1
## 2          2       324         107                 4 4.0 4.5 8.87        1
## 3          3       316         104                 3 3.0 3.5 8.00        1
## 4          4       322         110                 3 3.5 2.5 8.67        1
## 5          5       314         103                 2 2.0 3.0 8.21        0
## 6          6       330         115                 5 4.5 3.0 9.34        1
##   Chance.of.Admit
## 1            0.92
## 2            0.76
## 3            0.72
## 4            0.80
## 5            0.65
## 6            0.90
```

Question 1 (30 points in total)

a) i- Display the first three rows in this dataset.(1 point)

```
head(data, 3)
```

```
##   Serial.No. GRE.Score TOEFL.Score University.Rating SOP LOR CGPA Research
## 1          1       337         118                 4 4.5 4.5 9.65        1
## 2          2       324         107                 4 4.0 4.5 8.87        1
## 3          3       316         104                 3 3.0 3.5 8.00        1
##   Chance.of.Admit
## 1            0.92
## 2            0.76
## 3            0.72
```

ii - Display the structure of all variables.(1 point)

```
#str(data)
str(data$SOP)
```

```
##  num [1:400] 4.5 4 3 3.5 2 4.5 3 3 2 3.5 ...
```

iii - Print the descriptive statistics of the admission data to understand the data a little better (min, max, mean, median, 1st and 3rd quartiles). (1 point)

```
summary(data)
```

```
##    Serial.No.      GRE.Score      TOEFL.Score    University.Rating
##  Min.   :  1.0   Min.   :290.0   Min.   : 92.0   Min.   :1.000    
##  1st Qu.:100.8   1st Qu.:308.0   1st Qu.:103.0   1st Qu.:2.000    
##  Median :200.5   Median :317.0   Median :107.0   Median :3.000    
##  Mean   :200.5   Mean   :316.8   Mean   :107.4   Mean   :3.087    
##  3rd Qu.:300.2   3rd Qu.:325.0   3rd Qu.:112.0   3rd Qu.:4.000    
##  Max.   :400.0   Max.   :340.0   Max.   :120.0   Max.   :5.000    
##       SOP            LOR             CGPA          Research     
##  Min.   :1.0   Min.   :1.000   Min.   :6.800   Min.   :0.0000  
##  1st Qu.:2.5   1st Qu.:3.000   1st Qu.:8.170   1st Qu.:0.0000
##  Median :3.5   Median :3.500   Median :8.610   Median :1.0000
##  Mean   :3.4   Mean   :3.453   Mean   :8.599   Mean   :0.5475
##  3rd Qu.:4.0   3rd Qu.:4.000   3rd Qu.:9.062   3rd Qu.:1.0000
##  Max.   :5.0   Max.   :5.000   Max.   :9.920   Max.   :1.0000
##  Chance.of.Admit
##  Min.   :0.3400
##  1st Qu.:0.6400
##  Median :0.7300
##  Mean   :0.7244
##  3rd Qu.:0.8300
##  Max.   :0.9700
```

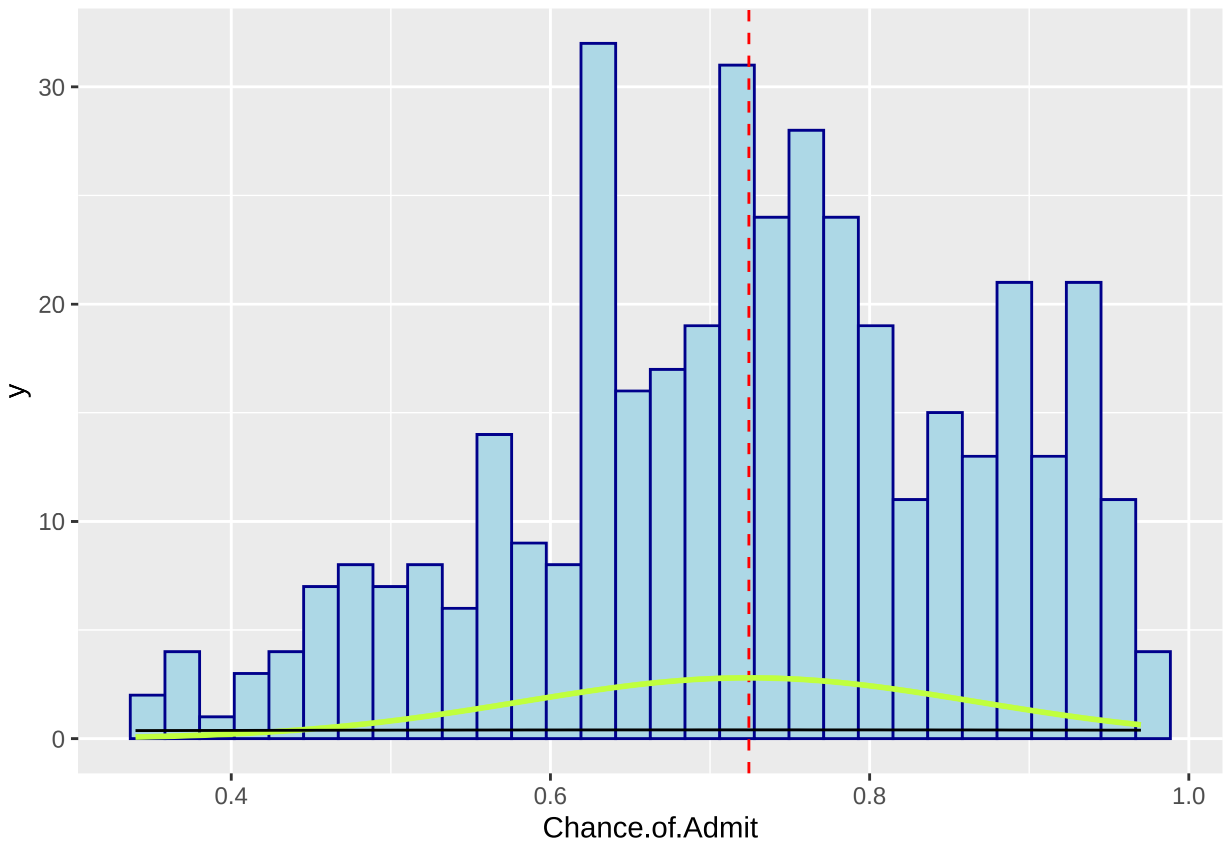
```
sd(data$Chance.of.Admit)
```

```
## [1] 0.1426093
```

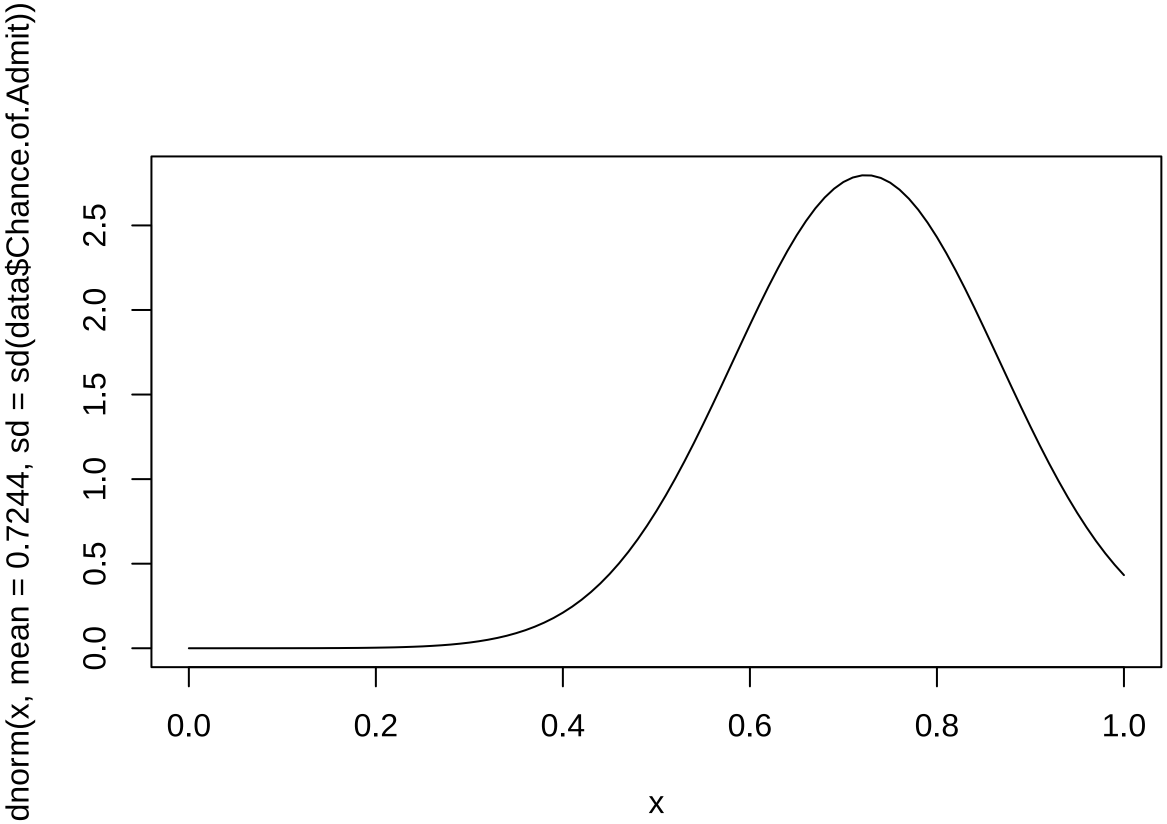
iv - Use a histogram to assess the normality of the ‘Chance.of.Admit’ variable and explain whether it appears normally distributed or not and why? (1 point)

```
ggplot(data, aes(x= Chance.of.Admit))+
  geom_histogram(color="darkblue", fill="lightblue")+
  geom_vline(aes(xintercept=mean(Chance.of.Admit)), color="red",
             linetype="dashed") +
  stat_function(fun = dnorm, args=list(mean = mean(data$Chance.of.Admit, na.rm = TRUE), sd =sd(data$Char
  stat_function( fun = dnorm, args = list(mean = 0.7244))
```

```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```



```
curve(dnorm(x, mean = 0.7244, sd = sd(data$Chance.of.Admit)),from = 0, to = 1)
```



from the graph, the green curve is the curve of the nuormal distribution with the same mean and sd as that of the variable Chance.of.Admit The histogram of the actual data in question is far more peaked than the normal curve

```
skewness(data$Chance.of.Admit)
```

```
## [1] -0.3521213
```

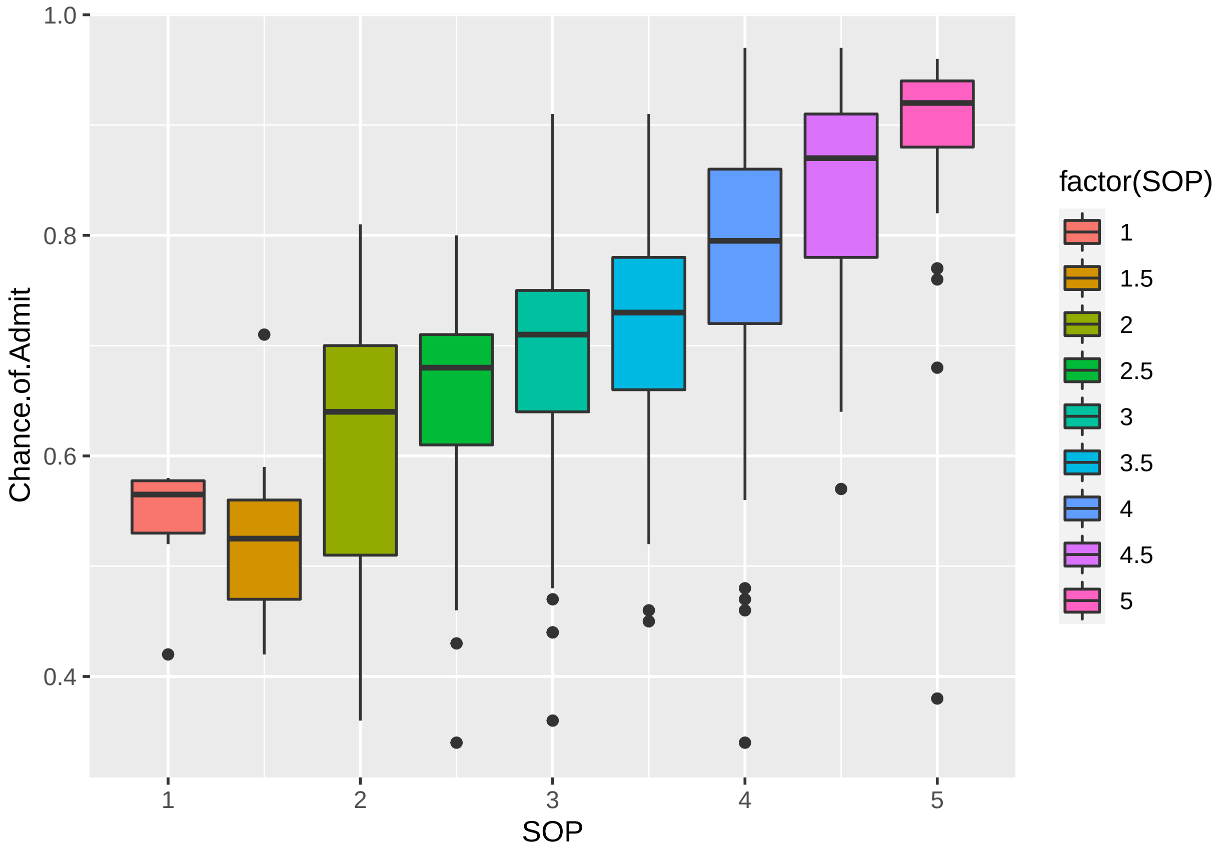
```
kurtosis(data$Chance.of.Admit)
```

```
## [1] 2.600759
```

Our skew value is -0.35.. which indicates a slight skew, but not one significant enough to consider the data not normal. From the kurtosis value, we can see that it is far greater than +1, meaning that the distribution is too peaked. Thus this distribution is considered nonnormal.

b) Create a set of boxplots that shows the distribution of Chance.of.Admit on SOP variables. Use different colors for different SOP score marks. Hint: SOP scores are changing between 1,1.5, to 5, therefore you can use different box colours for each score likewise; 1 (red), 1.5(green), etc. (8 points)

```
data %>%
  ggplot(aes(x=SOP, y=Chance.of.Admit, fill=factor(SOP))) +
  geom_boxplot()
```



c)

i- Find the covariance between the "GRE.Score" and the "Chance.of.Admit". (3 points)

```
cov(data$GRE.Score, data$Chance.of.Admit)
```

```
## [1] 1.313271
```

ii- Print or plot the correlation matrix of the data and write down the correlations between the GRE.Score, TOEFL.Score, CGPA and the Chance.of.Admit. (3 points)

```
cor_matrix <- cor(data.frame(data$GRE.Score, data$TOEFL.Score, data$CGPA, data$Chance.of.Admit))
cor_matrix
```

```
##                      data.GRE.Score data.TOEFL.Score data.CGPA
## data.GRE.Score            1.0000000        0.8359768 0.8330605
## data.TOEFL.Score          0.8359768        1.0000000 0.8284174
## data.CGPA                 0.8330605        0.8284174 1.0000000
## data.Chance.of.Admit      0.8026105        0.7915940 0.8732891
##                      data.Chance.of.Admit
## data.GRE.Score                  0.8026105
## data.TOEFL.Score                0.7915940
## data.CGPA                       0.8732891
## data.Chance.of.Admit            1.0000000
```

iii - Interpret the covariance and correlation results obtained from c(i) and c(ii) in terms of the strength and direction of the relationship. (4 points)

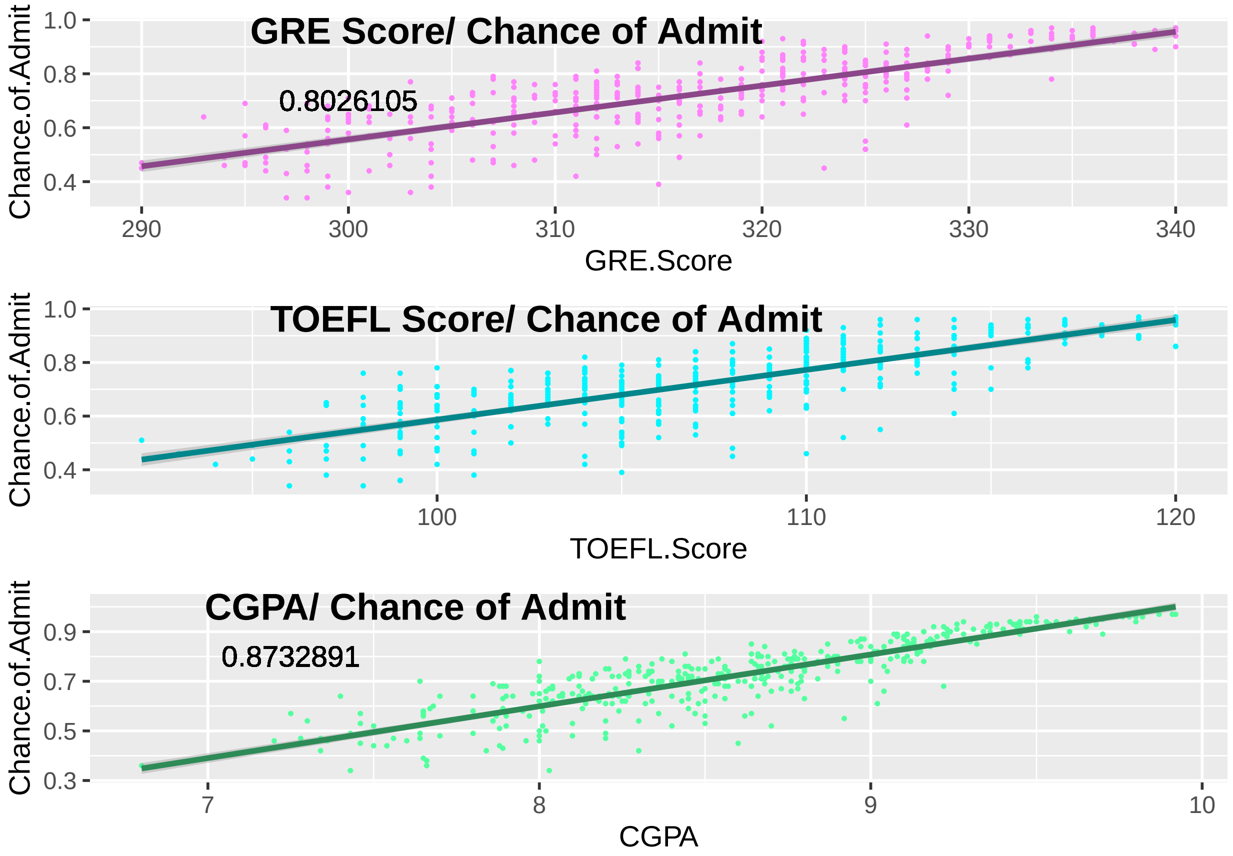
The covariance score between GRE.Score and Chance of Admit is 1.313271. As this is a positive number it implies that an increase in one variable results in an increase in the other, and a decrease in one variable would result in a decrease in the other. i.e. both GRE.score and Chance.of,Admit move together in the same direction when they change Whilst we can see that the variables are positively related it is hard for us to see by how much as covariance values can range from - infinity to + infinity and are dependent on the scales of values of the individual variables. Here our variables are on very different scales thus we must also use the correlation values to interpret their relationship. Their correlation score is 0.8026105, implying that they have a very strong positive correlation.

d) Use ggplot() to plot the graphs to see the relationship between each of three variables (GRE.Score, TOEFL.Score, CGPA) with Chance.of.Admit. (8 points)

```
Gre_Score_Plot <- ggplot(data, aes(x = GRE.Score, y = Chance.of.Admit))+
   geom_point(colour = 'orchid1', size = 0.3)+
   geom_smooth(method=lm, color = 'orchid4')+
   geom_text(x=300, y=0.7,label = cor_matrix[1,4], parse = TRUE)

TOEFL_Score_Plot <- ggplot(data, aes(x = TOEFL.Score, y = Chance.of.Admit))+
  geom_point(colour = 'turquoise1', size = 0.3)+
  geom_text(x=50, y=0.7,label = cor_matrix[2,4])+
  geom_smooth(method=lm, color = 'turquoise4')



CGPA_Plot <- ggplot(data, aes(x = CGPA, y = Chance.of.Admit))+
   geom_point(colour = 'seagreen1', size = 0.3)+
   geom_smooth(method=lm, color = 'seagreen4')+
   geom_text(x=7.25, y=0.8,label = cor_matrix[3,4], parse = TRUE)

plot_grid(Gre_Score_Plot, TOEFL_Score_Plot, CGPA_Plot, labels=c("GRE Score/ Chance of Admit", "TOEFL Sc
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
## ‘geom_smooth()‘ using formula ’y ~ x’
## ‘geom_smooth()‘ using formula ’y ~ x’
```

```
## Warning in as_grob.default(plot): Cannot convert object of class
## element_textelement into a grob.
```



Question 2 (40 points in total)

a)

i- Define the linear regression model between GRE.Score and Chance.of.Admit (3 points)

```
cor(data$GRE.Score,data$Chance.of.Admit)
```

```
## [1] 0.8026105
```

```
slmodel <- lm(Chance.of.Admit~GRE.Score, data = data)
slmodel
```

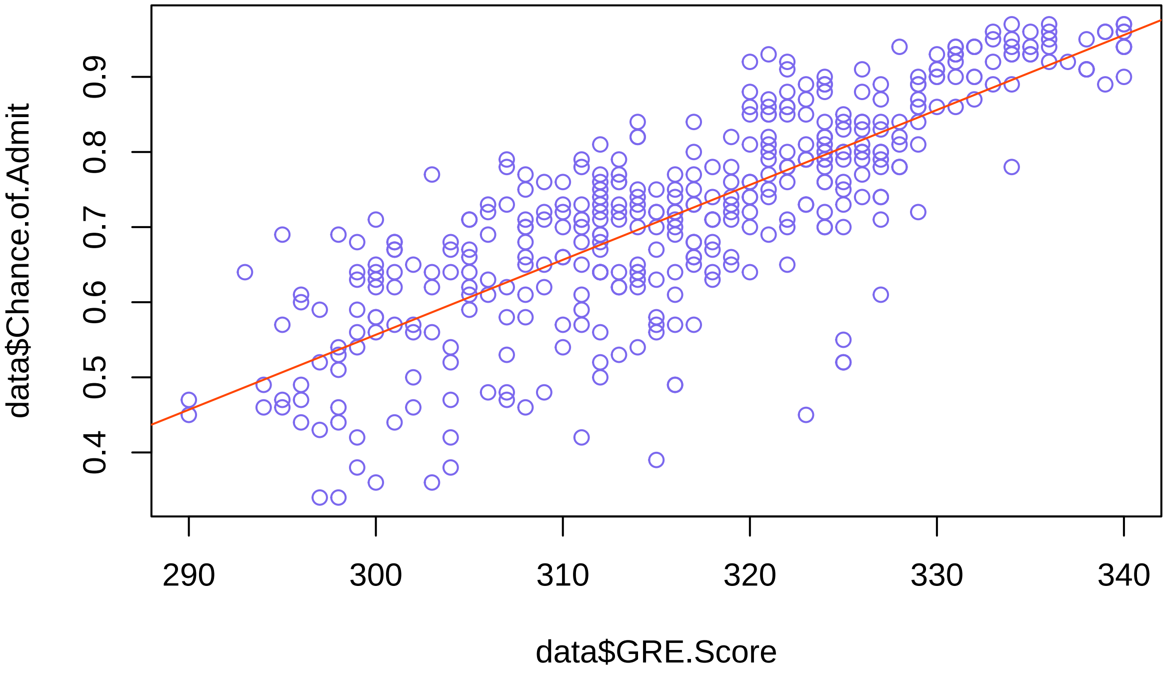
```
##
## Call:
## lm(formula = Chance.of.Admit ~ GRE.Score, data = data)
##
## Coefficients:
## (Intercept)    GRE.Score
##   -2.436084     0.009976
```

```
slmodel$coefficients[1]
```

```
## (Intercept)
##   -2.436084
```

ii - Plot the regression (least-square) line on the same plot.(3 points)

```
plot(data$GRE.Score, data$Chance.of.Admit, col = "mediumslateblue")
abline(lm(Chance.of.Admit~GRE.Score, data = data), col = "orangered1")
```



ii- - Explain the meaning of the slope and y-intercept for the least-squares regression line in Q2(ii). (3 points)

The intercept is the expected value of the chance of admission, when we consider the average gre score of all the students in the dataset. The slope is the effect that the gre score has on the chance of admission. For every increase in chance of admission, the required gre score goes up by 0.0099759

b) Print the results of this model and interpret the results by following questions:

```
summary(slmodel)
```

```
##
## Call:
## lm(formula = Chance.of.Admit ~ GRE.Score, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33613 -0.04604  0.00408  0.05644  0.18339
```

```
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.4360842  0.1178141  -20.68   <2e-16 ***
## GRE.Score    0.0099759  0.0003716   26.84   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08517 on 398 degrees of freedom
## Multiple R-squared:  0.6442, Adjusted R-squared:  0.6433
## F-statistic: 720.6 on 1 and 398 DF,  p-value: < 2.2e-16
```

i - What is the number of observations was the regression run on? (3 points)

```r
length(slmodel$residuals)
```

```
## [1] 400
```

ii - Interpret the R-squared of this regression? (4 points)

The residual standard error is the average amount that the response variable (chance of admit) will deviate from the true regression line (0.08517, or 8.5%) r- squared is how well the regression model fits the observed data. Our R^2 value is 0.6442, i.e. roughly 64% of the variance found in the response variable can be explained by the predictor variable.

iii - Write the regression equation associated with this regression model? (4 points)

```r
# Chance_of_Admit = slmodel$coefficients[1] + (slmodel$coefficients[2]*GRE_Score)
```

c) Use the regression line to predict the chance of admit when GRE score 310. (10 points)

```r
Chance_of_Admit <- slmodel$coefficients[1] + (slmodel$coefficients[2]*310)
Chance_of_Admit
```

```
## (Intercept)
##   0.6564392
```

So the chance of admit is 65.6%

d) From the given Q2(a) linear model between GRE.Score and Chance.of.Admit, what should be GRE score of a student who has 50% of chance of admission?(10 points)

Using our model we have intercept = -2.4360842 and slope = 0.0099759. So our equation is Y (Chance.of.Admit) = -2.4360842 + 0.0099759*X (GRE.Score) we let Y = 0.5 it then follows that: X (GRE.Score) = (0.5 + 2.4360842)/0.0099759

```r
predicted_GRE <- (0.5 + 2.4360842)/0.0099759
predicted_GRE
```

```
## [1] 294.3177
```

Looking at our scatterplot and our regression line we can see that this value is indeed plausible.

Question 3 (30 points in total)

a) Use three independent variables (‘GRE.Score’,‘TOEFL.Score’, ‘CGPA’) to build a multiple linear regression model to predict dependent variable ‘Chance.of.Admit’. Display a summary of your model indicating Residuals, Coefficients, etc. Explain the summary results. (8 points)

```
mult <- lm(Chance.of.Admit ~ GRE.Score + TOEFL.Score + CGPA, data = data)
mult
```

```
##
## Call:
## lm(formula = Chance.of.Admit ~ GRE.Score + TOEFL.Score + CGPA,
##     data = data)
##
## Coefficients:
## (Intercept)    GRE.Score  TOEFL.Score         CGPA
##   -1.585698     0.002266     0.003112     0.146284
```

```
summary(mult)
```

```
##
## Call:
## lm(formula = Chance.of.Admit ~ GRE.Score + TOEFL.Score + CGPA,
##     data = data)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.290375 -0.023030  0.008255  0.040153  0.143108
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.5856984  0.1058153 -14.986  < 2e-16 ***
## GRE.Score    0.0022660  0.0005929   3.822 0.000154 ***
## TOEFL.Score  0.0031123  0.0011070   2.812 0.005176 **
## CGPA         0.1462844  0.0111770  13.088  < 2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.06632 on 396 degrees of freedom
## Multiple R-squared:  0.7854, Adjusted R-squared:  0.7837
## F-statistic:   483 on 3 and 396 DF,  p-value: < 2.2e-16
```

b) Write the regression equation associated with this multiple regression model. (8 points)

```
# Chance_of_Admit = mult$coefficients[1] + (mult$coefficients[2]* GRE_Score) + (mult$coefficients[3]*TO
```

c) Using this model:

i- Find the chance of admit for the 3rd student and 23rd students in the dataset. (4 points)

```
#Finding the observed values
s3 <- data$Chance.of.Admit[data$Serial.No.==3]
s3
```

```
## [1] 0.72
```

```
s23 <- data$Chance.of.Admit[data$Serial.No.==23]
s23
```

```
## [1] 0.94
```

```
#Finding the predicted values using the model
s_predict <- data.frame(GRE.Score = c(data$GRE.Score[data$Serial.No.==3],data$GRE.Score[data$Serial.No.=
s_predict
```

```
##   GRE.Score TOEFL.Score CGPA
## 1       316         104  8.0
## 2       328         116  9.5
```

```
predict(mult, s_predict)
```

```
##         1         2
## 0.6242940 0.9082592
```

```
#So, using the model, the chance of admit for student 3 is 62.43%, and the chance of admit for
#student 23 is 90.83%
```

ii- Identify which student (3rd or 23rd) has higher chance than the other and print the difference between the chance of admit of these two students.(3 points)

```
#student 23 has a higher chance of admission

predict(mult, s_predict)[2]-predict(mult, s_predict)[1]
```

```
##         2
## 0.2839652
```

d) Explain the difference between the linear regression models in Question 2 and in Question 3. (7 points)

```
cor_matrix
```

```
##                      data.GRE.Score data.TOEFL.Score data.CGPA
## data.GRE.Score            1.0000000        0.8359768 0.8330605
## data.TOEFL.Score          0.8359768        1.0000000 0.8284174
## data.CGPA                 0.8330605        0.8284174 1.0000000
## data.Chance.of.Admit      0.8026105        0.7915940 0.8732891
##                      data.Chance.of.Admit
## data.GRE.Score                  0.8026105
## data.TOEFL.Score                0.7915940
## data.CGPA                       0.8732891
## data.Chance.of.Admit            1.0000000
```

In question 2 we use a simple linear regression model whereas in question 3 we use a multiple regression model. The simple regression model establishes a relationship between two variables using a straight line. The regression attempts to draw a line that comes closest to the data by finding the slope and intercept that define the line and minimise regression errors. However it is unlikely that a variable is explained by only one other variable. the multiple regression model is supposed to establish the relationship between one dependent variable (chance of admit) and multiple independent variables (CGPA, GRE and TOEFL scores). We assume no major correlation between the independent variables (however from our matrix we can see that GRE, CGPA, and TOEFL all have very strong correlation coefficients)