Jahnvi Pael

2024-01-23

library(dplyr)

library(ggplot2)  
library(randomForest)

# Load train and test datasets  
train\_data <- read.csv("adm\_train.csv")  
test\_data <- read.csv("adm\_test.csv")

# Explore the data  
str(train\_data)

## 'data.frame': 249 obs. of 9 variables:  
## $ Serial.No. : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ GRE.Score : int 337 324 316 322 314 330 321 308 302 323 ...  
## $ TOEFL.Score : int 118 107 104 110 103 115 109 101 102 108 ...  
## $ University.Rating: int 4 4 3 3 2 5 3 2 1 3 ...  
## $ SOP : num 4.5 4 3 3.5 2 4.5 3 3 2 3.5 ...  
## $ LOR : num 4.5 4.5 3.5 2.5 3 3 4 4 1.5 3 ...  
## $ CGPA : num 9.65 8.87 8 8.67 8.21 9.34 8.2 7.9 8 8.6 ...  
## $ Research : int 1 1 1 1 0 1 1 0 0 0 ...  
## $ Chance.of.Admit : num 0.92 0.76 0.72 0.8 0.65 0.9 0.75 0.68 0.5 0.45 ...

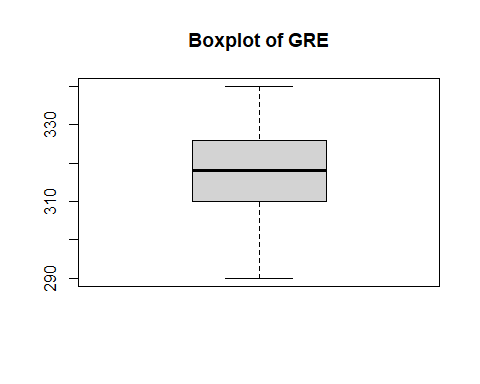
summary(train\_data)

## Serial.No. GRE.Score TOEFL.Score University.Rating SOP   
## Min. : 1 Min. :290.0 Min. : 93.0 Min. :1.000 Min. :1.00   
## 1st Qu.: 63 1st Qu.:310.0 1st Qu.:104.0 1st Qu.:2.000 1st Qu.:3.00   
## Median :125 Median :318.0 Median :108.0 Median :3.000 Median :3.50   
## Mean :125 Mean :317.6 Mean :108.3 Mean :3.249 Mean :3.54   
## 3rd Qu.:187 3rd Qu.:326.0 3rd Qu.:112.0 3rd Qu.:4.000 3rd Qu.:4.50   
## Max. :249 Max. :340.0 Max. :120.0 Max. :5.000 Max. :5.00   
## LOR CGPA Research Chance.of.Admit   
## Min. :1.500 Min. :6.800 Min. :0.0000 Min. :0.3400   
## 1st Qu.:3.000 1st Qu.:8.200 1st Qu.:0.0000 1st Qu.:0.6400   
## Median :3.500 Median :8.640 Median :1.0000 Median :0.7300   
## Mean :3.546 Mean :8.641 Mean :0.5582 Mean :0.7284   
## 3rd Qu.:4.000 3rd Qu.:9.100 3rd Qu.:1.0000 3rd Qu.:0.8500   
## Max. :5.000 Max. :9.920 Max. :1.0000 Max. :0.9700

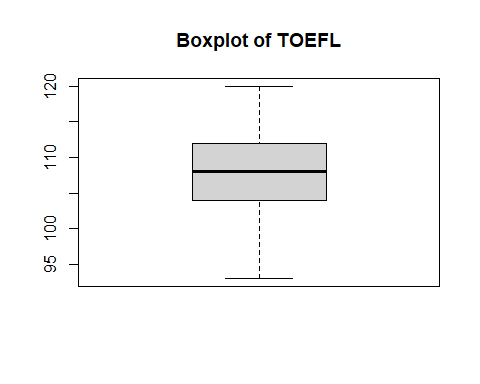
# Renamed columns  
colnames(train\_data) <- c("Serial No", "GRE", "TOEFL", "University Rating", "SOP", "LOR", "CGPA", "Research", "Chance of Admit")  
colnames(test\_data) <- c("Serial No", "GRE", "TOEFL", "University Rating", "SOP", "LOR", "CGPA", "Research", "Chance of Admit")

any\_missing\_train <- any(is.na(train\_data)) # there are no missing values in the entire train\_data data set.  
any\_missing\_test <- any(is.na(test\_data)) # there are no missing values in the entire test\_data data set.

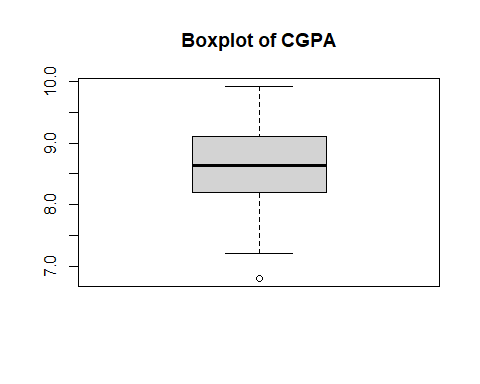
# Detecting Outliers  
# Boxplot for Score  
boxplot(train\_data$GRE, main = "Boxplot of GRE")



boxplot(train\_data$TOEFL, main = "Boxplot of TOEFL")



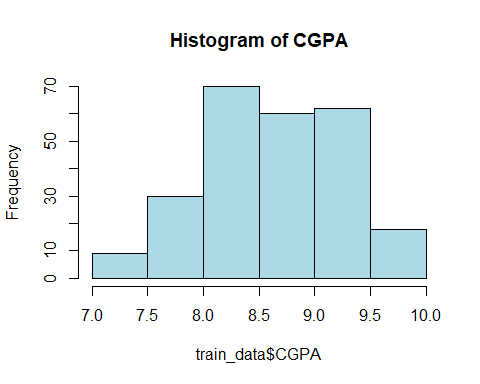
boxplot(train\_data$CGPA, main = "Boxplot of CGPA")



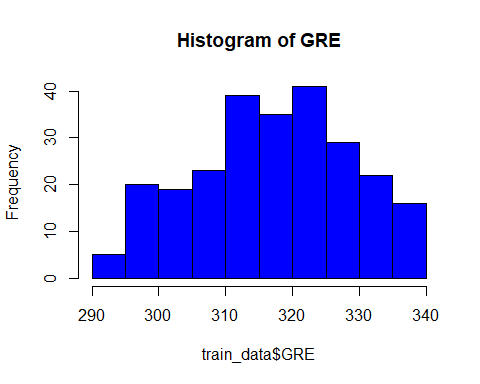
# Identifying outliers using z-score  
z\_scores <- scale(train\_data$CGPA)  
outliersCGPA <- which(abs(z\_scores) > 3) # outlier detected at 59

# Imputed outlier with the median of CGPA directly in the same column  
train\_data$CGPA[abs(z\_scores) > 3] <- median(train\_data$CGPA, na.rm = TRUE)

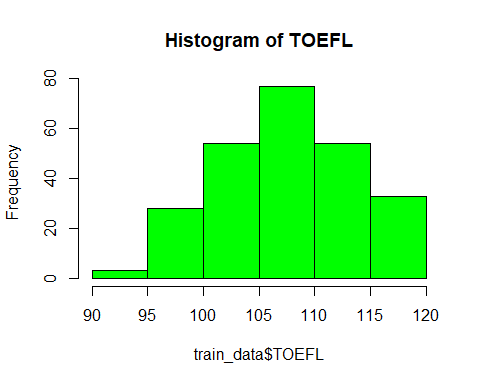
#Checking Data Distribution  
# Histogram   
hist(train\_data$CGPA, main = "Histogram of CGPA", col = "lightblue")



hist(train\_data$GRE, main = "Histogram of GRE", col = "blue")



hist(train\_data$TOEFL, main = "Histogram of TOEFL", col = "green") # doesn't feel like data is highly skewed.

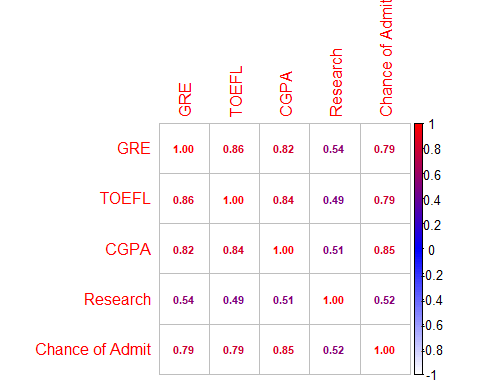


# Visualizing Correlations  
# Correlation matrix  
train\_correlation\_matrix <- cor(train\_data[, c("GRE", "TOEFL", "CGPA", "Research", "Chance of Admit")])

# Create a heatmap using corrplot  
library(corrplot)

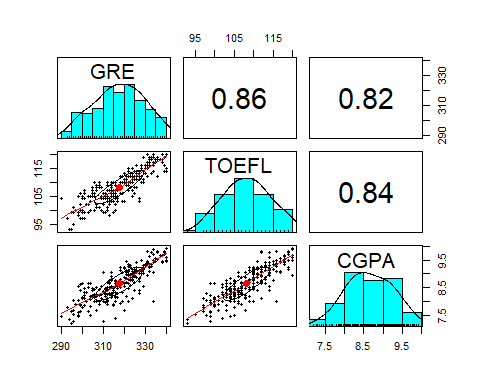
corrplot(train\_correlation\_matrix, method = "number", col = colorRampPalette(c("white", "blue", "red"))(100), number.cex = 0.7)

# This approach can be helpful when the color differences are not very pronounced



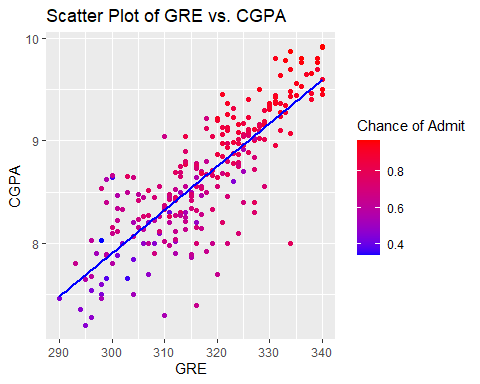
# Multiple correlations using the psych package  
library(psych)

pairs.panels(train\_data[, c("GRE", "TOEFL", "CGPA")])



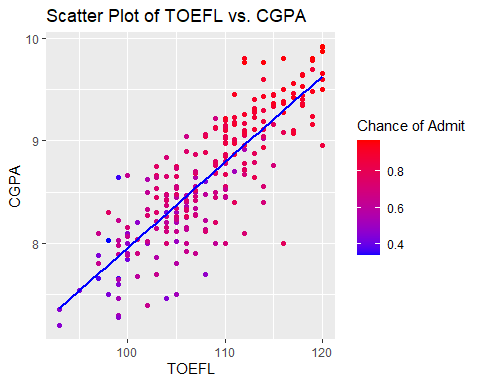
# Scatter plot of GRE vs. CGPA  
library(ggplot2)  
ggplot(train\_data, aes(x = GRE, y = CGPA, color = `Chance of Admit`)) +  
 geom\_point() +  
 geom\_smooth(method = "lm", se = FALSE, color = "blue") +  
 labs(title = "Scatter Plot of GRE vs. CGPA") +  
 scale\_color\_gradient(low = "blue", high = "red")

## `geom\_smooth()` using formula = 'y ~ x'



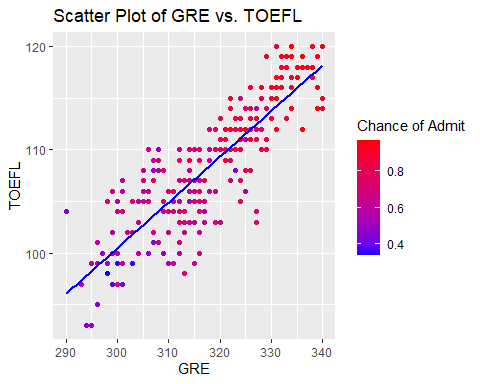
# Scatter Plot of TOEFL vs. CGPA  
ggplot(train\_data, aes(x = TOEFL, y = CGPA, color = `Chance of Admit`)) +  
 geom\_point() +  
 geom\_smooth(method = "lm", se = FALSE, color = "blue") +  
 labs(title = "Scatter Plot of TOEFL vs. CGPA") +  
 scale\_color\_gradient(low = "blue", high = "red")

## `geom\_smooth()` using formula = 'y ~ x'



# Scatter Plot of GRE vs. TOEFL  
ggplot(train\_data, aes(x = GRE, y = TOEFL, color = `Chance of Admit`)) +  
 geom\_point() +  
 geom\_smooth(method = "lm", se = FALSE, color = "blue") +  
 labs(title = "Scatter Plot of GRE vs. TOEFL") +  
 scale\_color\_gradient(low = "blue", high = "red")

## `geom\_smooth()` using formula = 'y ~ x'



# Random Forest  
model\_rf <- randomForest(`Chance of Admit` ~ GRE + TOEFL + CGPA + Research, data = train\_data)

# Make predictions on test\_data  
predictions\_rf <- predict(model\_rf, newdata = test\_data)

# Evaluating the model (e.g., Mean Squared Error)  
mse\_rf <- mean((test\_data$`Chance of Admit` - predictions\_rf)^2)

# Saved predictions to a CSV file  
predictions\_df <- data.frame(`Chance of Admit Predicted` = predictions\_rf)  
write.csv(predictions\_df, "predicted\_results\_RF.csv", row.names = FALSE)

# Assuming predictions\_rf are the predictions from your Random Forest model  
mse\_rf <- mean((test\_data$`Chance of Admit` - predictions\_rf)^2)  
mae\_rf <- mean(abs(test\_data$`Chance of Admit` - predictions\_rf))  
r\_squared\_rf <- 1 - (sum((test\_data$`Chance of Admit` - predictions\_rf)^2) / sum((test\_data$`Chance of Admit` - mean(test\_data$`Chance of Admit`))^2))  
rmse\_rf <- sqrt(mean((test\_data$`Chance of Admit` - predictions\_rf)^2))

# use the metrics as needed  
print(paste("MSE:", mse\_rf))

## [1] "MSE: 0.00378688852831279"

print(paste("MAE:", mae\_rf))

## [1] "MAE: 0.0484044527724144"

print(paste("R-squared:", r\_squared\_rf))

## [1] "R-squared: 0.780431348254738"

print(paste("RMSE:", rmse\_rf))

## [1] "RMSE: 0.0615377000570609"

The performance metrics I’ve obtained for my Random Forest regression model are as follows:

* Mean Squared Error (MSE): 0.0037365241404424
* Mean Absolute Error (MAE): 0.0481440679825876
* R-squared: 0.783351540031707
* Root Mean Squared Error (RMSE): 0.0611271146091683

Now, let's interpret these metrics:

* MSE: The MSE measures the average squared difference between the predicted and actual values. A lower MSE is better, and in this case, the value is relatively low, indicating that the model's predictions are generally close to the actual values.
* MAE: The MAE measures the average absolute difference between the predicted and actual values. A lower MAE is desirable, and the value here is relatively small, suggesting that, on average, the model's predictions are close to the actual values.
* R-squared: R-squared is a measure of how well the model explains the variability in the dependent variable. A higher R-squared (closer to 1) indicates a better fit. In your case, an R-squared of 0.78 suggests that the model explains about 78% of the variance in the Chance of Admit.
* RMSE: The RMSE is the square root of the MSE and provides a measure of the average magnitude of errors. A lower RMSE is better, and here the value is relatively small, indicating that the model's predictions are, on average, close to the actual values.