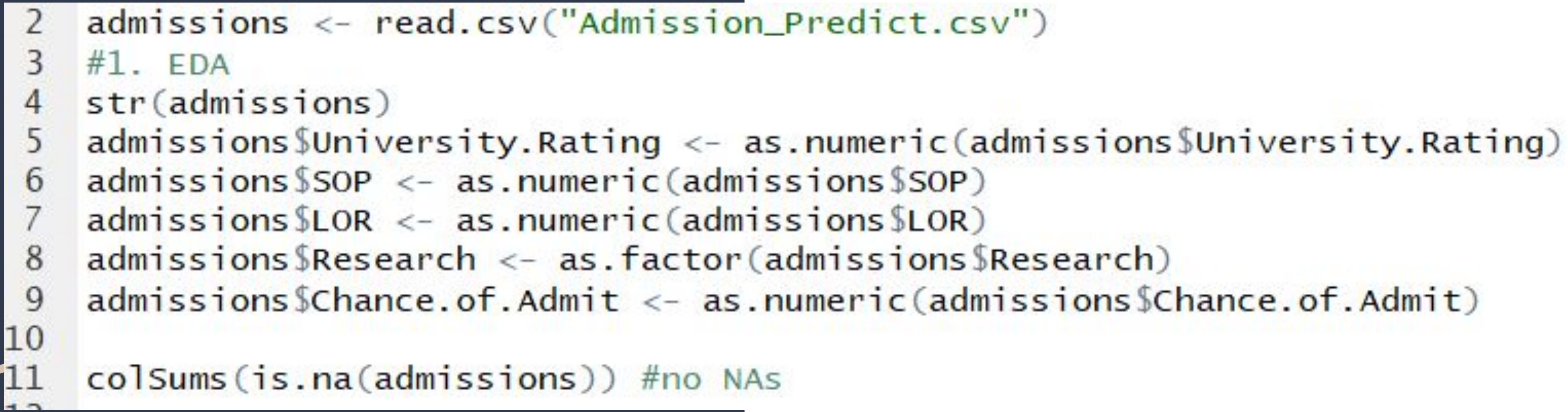
Predicting University Admissions Through Modeling

1. Choosing & Organizing Data

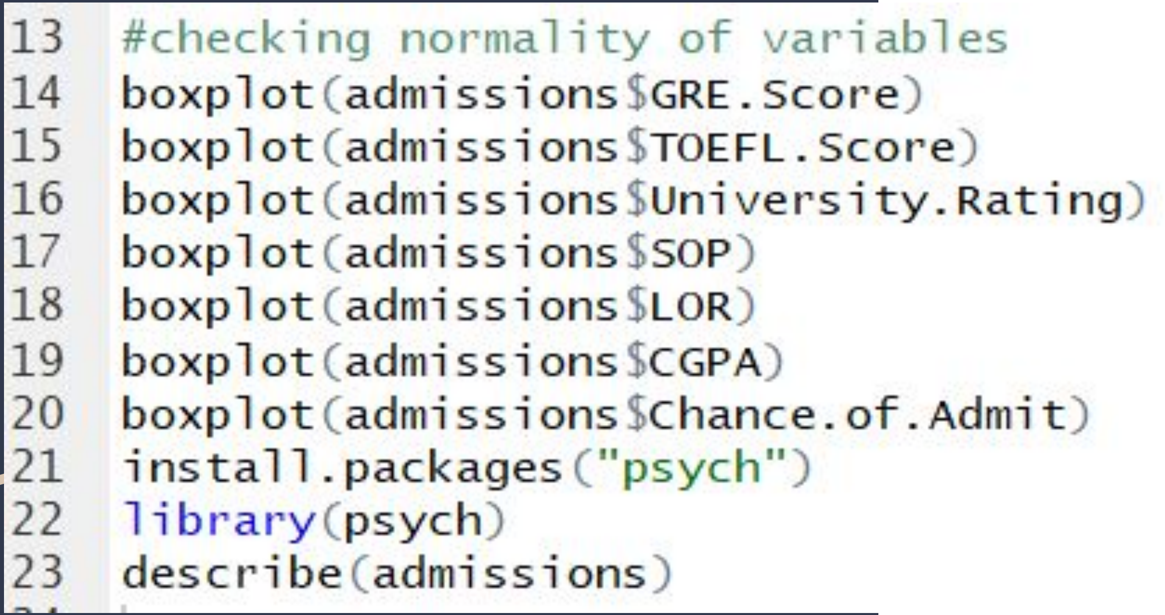


```
admissions <- read.csv("Admission_Predict.csv")
#1. EDA
str(admissions)
admissions$University.Rating <- as.numeric(admissions$University.Rating)
admissions$SOP <- as.numeric(admissions$SOP)
admissions$LOR <- as.numeric(admissions$LOR)
admissions$Research <- as.factor(admissions$Research)
admissions$Chance.of.Admit <- as.numeric(admissions$Chance.of.Admit)

colSums(is.na(admissions)) #no NAs
```

Describing & Exploring the Data (EDA)

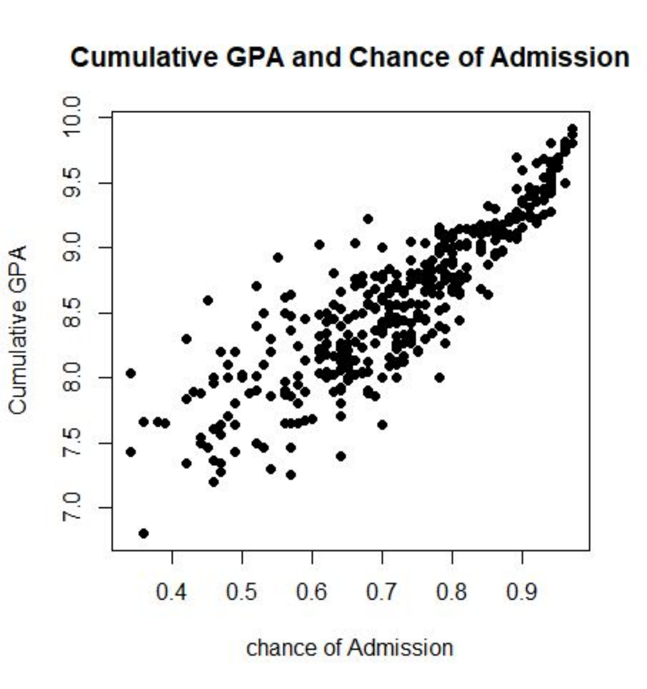
- Boxplots for each variable to observe the distribution of the variables and decide whether transformation is required.
- “Describe” command to see skewness, sd, mean, median, and more statistical values of each variable.

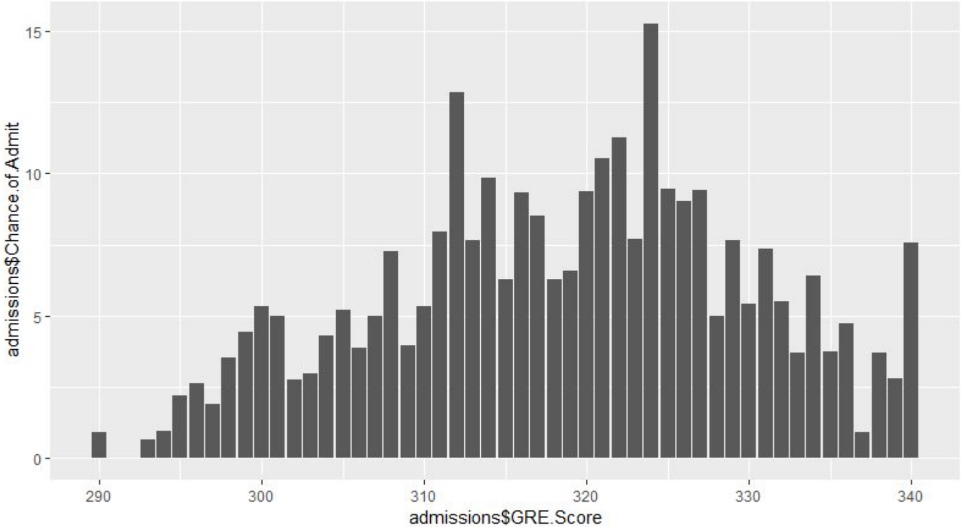


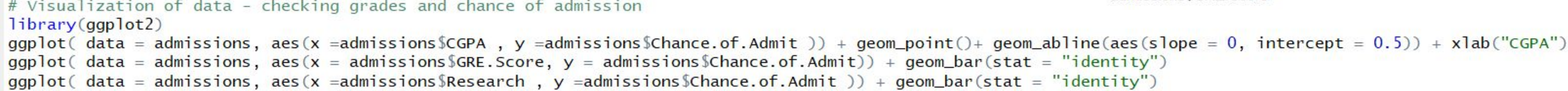
```r
#checking normality of variables
boxplot(admissions$GRE.Score)
boxplot(admissions$TOEFL.Score)
boxplot(admissions$University.Rating)
boxplot(admissions$SOP)
boxplot(admissions$LOR)
boxplot(admissions$CGPA)
boxplot(admissions$Chance.of.Admit)
install.packages("psych")
library(psych)
describe(admissions)
```

Simple Visualization

- Checking variables which look interesting; GPA and GRE Score, plotted against our target variable; Chance of admission

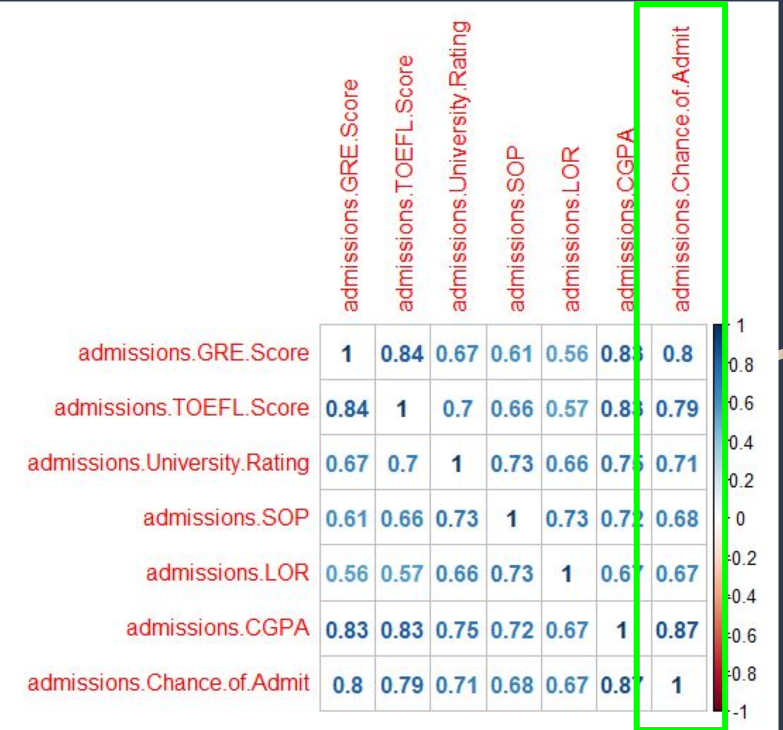






```
# Visualization of data - checking grades and chance of admission
library(ggplot2)
ggplot( data = admissions, aes(x =admissions$CGPA , y =admissions$Chance.of.Admit )) + geom_point()+ geom_abline(aes(slope = 0, intercept = 0.5)) + xlab("CGPA")
ggplot( data = admissions, aes(x = admissions$GRE.Score, y = admissions$Chance.of.Admit)) + geom_bar(stat = "identity")
ggplot( data = admissions, aes(x =admissions$Research , y =admissions$Chance.of.Admit )) + geom_bar(stat = "identity")
```

Selecting Variables

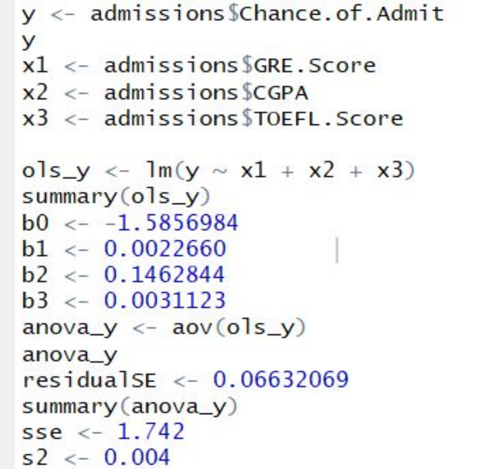


	admissions.GRE.Score	admissions.TOEFL.Score	admissions.University.Rating	admissions.SOP	admissions.LOR	admissions.CGPA	admissions.Chance.of.Admit
admissions.GRE.Score	1	0.84	0.67	0.61	0.56	0.83	0.8
admissions.TOEFL.Score	0.84	1	0.7	0.66	0.57	0.83	0.79
admissions.University.Rating	0.67	0.7	1	0.73	0.66	0.75	0.71
admissions.SOP	0.61	0.66	0.73	1	0.73	0.72	0.68
admissions.LOR	0.56	0.57	0.66	0.73	1	0.67	0.67
admissions.CGPA	0.83	0.83	0.75	0.72	0.67	1	0.87
admissions.Chance.of.Admit	0.8	0.79	0.71	0.68	0.67	0.87	1

- As we know we want to predict the chance of admission, we look for the variables with the highest correlation with that variable.
- 1. CGPA 2. GRE Score 3. TOEFL Score
- Goals: Create a model which can predict the chance of admission taking these 3 variables into account.
- Hypothesis: The higher the score on each of these tests, the higher the chance of admission.

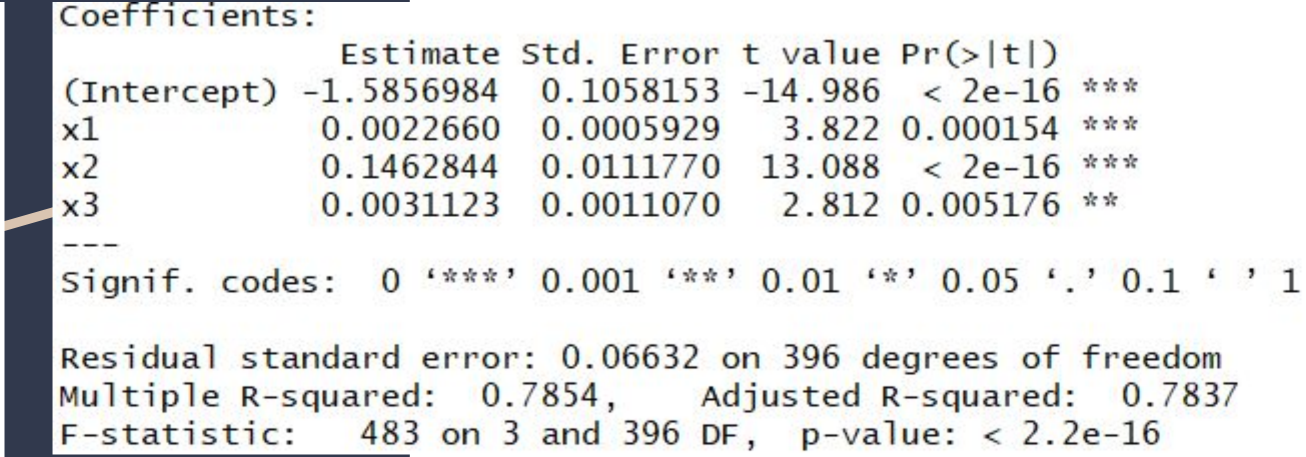
Creating the Model & Finding the Coefficients

- Adjusted R^2 Value: 0.7837 = good model, 78.3% of data explained through the new model.



```
y <- admissions$Chance.of.Admit
y
x1 <- admissions$GRE.Score
x2 <- admissions$CGPA
x3 <- admissions$TOEFL.Score

ols_y <- lm(y ~ x1 + x2 + x3)
summary(ols_y)
b0 <- -1.5856984
b1 <- 0.0022660
b2 <- 0.1462844
b3 <- 0.0031123
anova_y <- aov(ols_y)
anova_y
residualSE <- 0.06632069
summary(anova_y)
sse <- 1.742
s2 <- 0.004
```



```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -1.5856984  0.1058153 -14.986  < 2e-16 ***
x1           0.0022660  0.0005929   3.822 0.000154 ***
x2           0.1462844  0.0111770  13.088  < 2e-16 ***
x3           0.0031123  0.0011070   2.812 0.005176 ** 
---
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
```

Predicting chance of admission

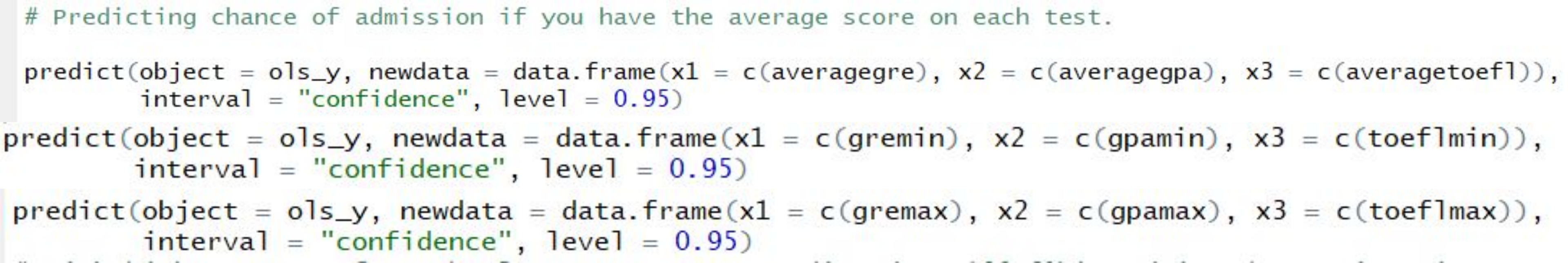
- 1. Using the average test scores from the sample in the dataset to predict chance of admission
  fit lower upper

  0.72435 0.7178308 0.7308692

- 2. Using the minimum test scores to predict chance of admission
  fit lower upper

  0.3524908 0.3315795 0.373402

- 3. Using maximum scores to predict chance
  fit lower upper
- 1.009339 0.9932109 1.025467



```
# Predicting chance of admission if you have the average score on each test.

predict(object = ols_y, newdata = data.frame(x1 = c(averagegre), x2 = c(averagegpa), x3 = c(averagetoefl)),
        interval = "confidence", level = 0.95)
predict(object = ols_y, newdata = data.frame(x1 = c(gremin), x2 = c(gpamin), x3 = c(toeflmin)),
        interval = "confidence", level = 0.95)
predict(object = ols_y, newdata = data.frame(x1 = c(gremax), x2 = c(gpamax), x3 = c(toeflmax)),
        interval = "confidence", level = 0.95)
```

Conclusion

Hypothesis was correct!

Model was able to predict chance of admission based on various combinations of test scores