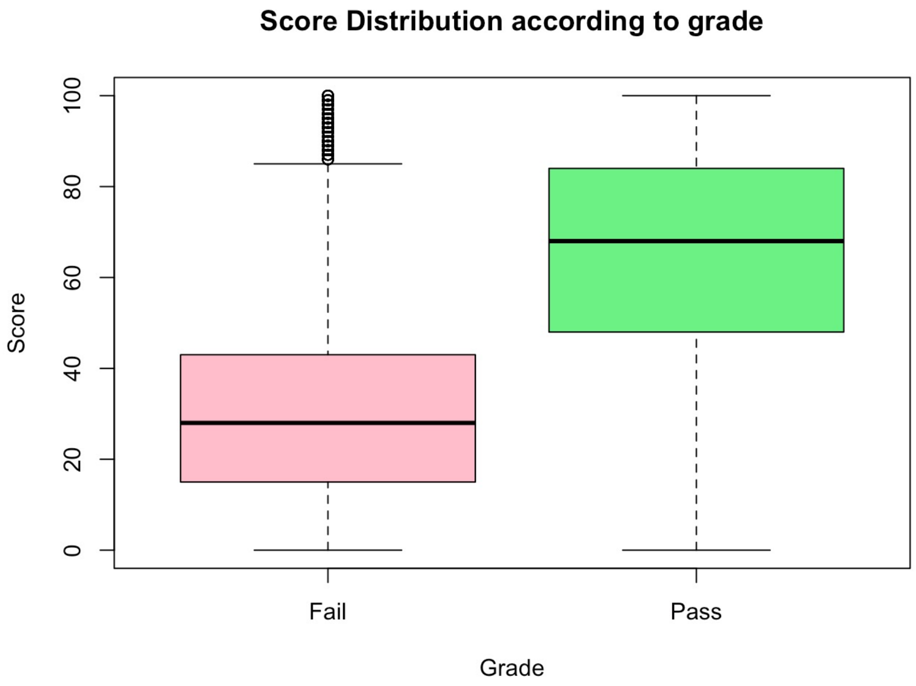
PREDICTING PROF. MOODY'S CLASS' GRADES

PRESENTATION BY-

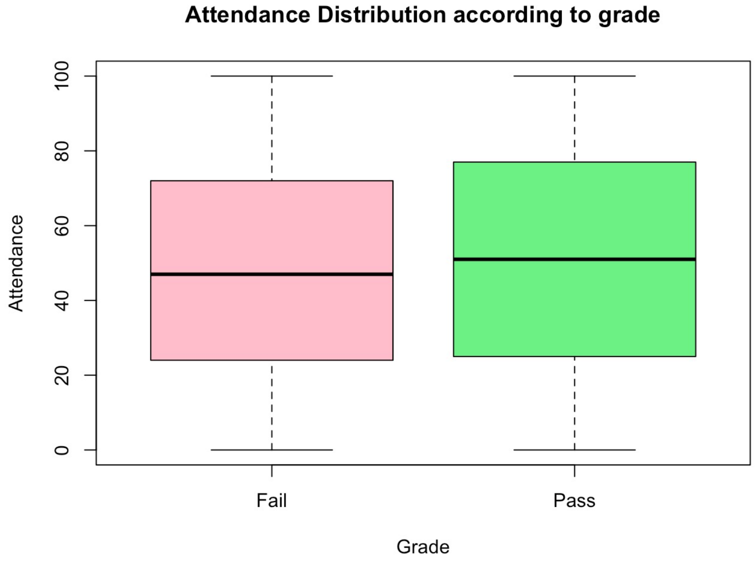
MUSKAN BURMAN

TRAINING DATA

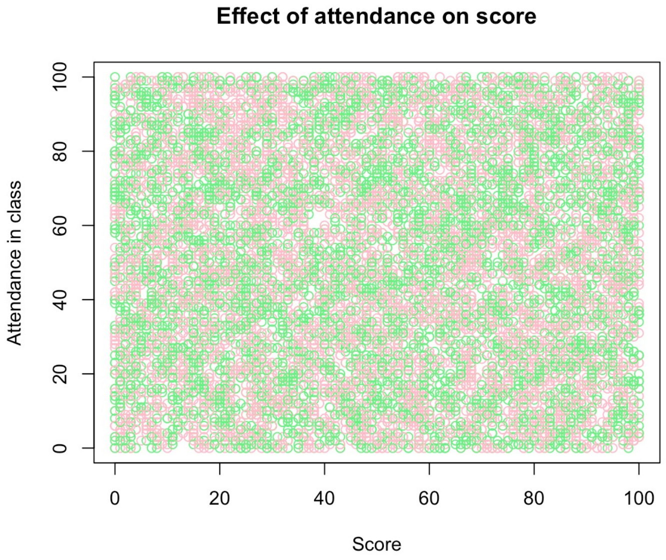
- I started by plotting the score distribution according to the grade, of the training data. Looking at this plot, we can see that there is no clear distinction between the pass and fail grades according to the scores of students, and there are several outliers present as well.
- Thus, the grades of students in Prof. Moody's class depends on factors other than just their scores.
- Let's analyze!



- In order to figure out the relationship between grades of students and the other attributes, I made some plots.

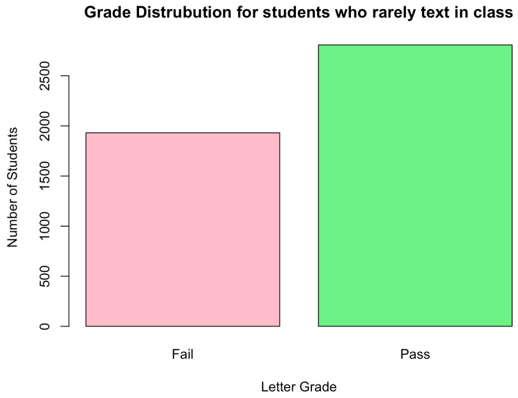


Attendance Distribution according to grade

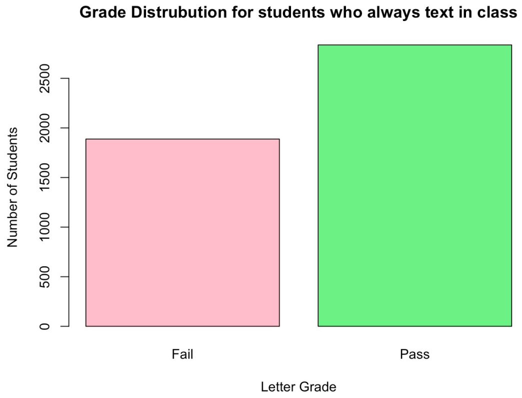


Effect of attendance on score

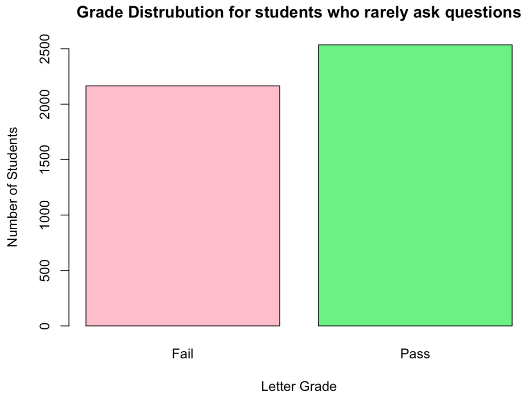
- It is clear from these plots that there is no clear relationship between grade and these factors.



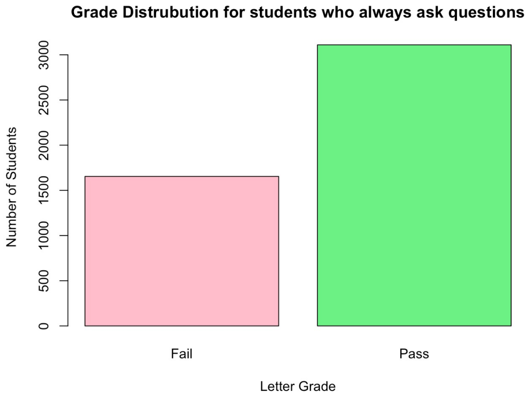
Grade Distrubution for students who rarely text in class



Grade Distrubution for students who always text in class



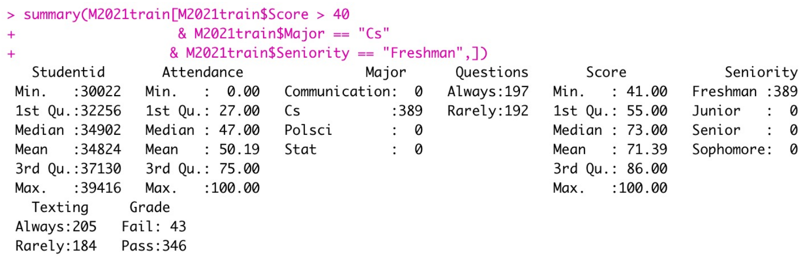
Grade Distrubution for students who rarely ask questions



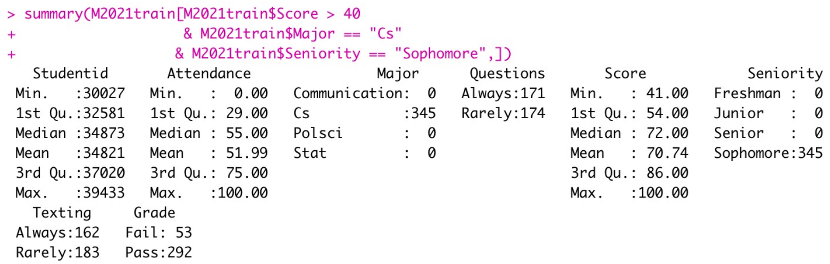
Grade Distrubution for students who always ask questions

PREDICTION MODEL : THE THOUGHT PROCESS

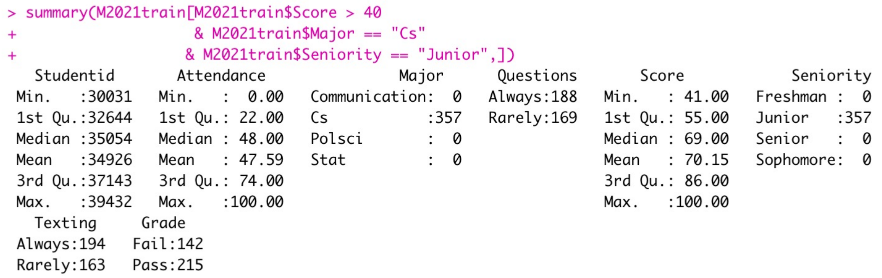
- For the purposes of cross validation and in an attempt to avoid overfitting, I started by randomly dividing by training data set into training and testing data – around 80% for training and 20% for testing.
- Looking at all those plots, I realized that there was no overall straightforward relationship between grade and other attributes.
- So, I decided to subset by the different majors and find possible relationships.
- Looking at the data, I had a hunch that there is some relationship between major and seniority levels.
- Thus, with score > 40 (according to the boxplot of score and grade) and major = CS, I started looking at the summaries for the different levels of seniority.
- We can see here that the number of students who failed is always less than the number of students who passed, for all seniority levels, except for seniors.
- Thus, we can say that CS Seniors with a score < 40 usually fail, rather than pass.



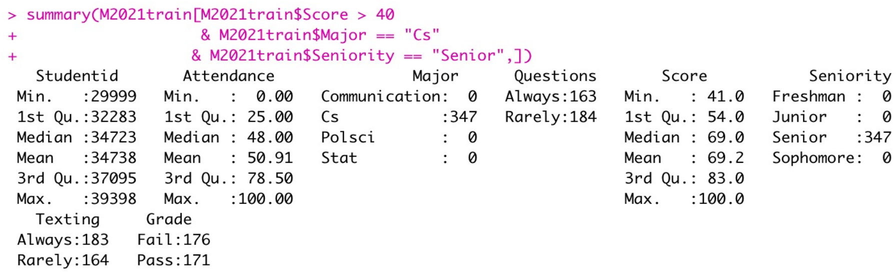
```
> summary(M2021train[M2021train$Score > 40
+                    & M2021train$Major == "Cs"
+                   & M2021train$Seniority == "Freshman",])
   Studentid       Attendance             Major      Questions       Score           Seniority
 Min.   :30022   Min.   :  0.00   Communication:  0   Always:197   Min.   : 41.00   Freshman :389
 1st Qu.:32256   1st Qu.: 27.00   Cs           :389   Rarely:192   1st Qu.: 55.00   Junior   :  0
 Median :34902   Median : 47.00   Polsci       :  0                Median : 73.00   Senior   :  0
 Mean   :34824   Mean   : 50.19   Stat         :  0                Mean   : 71.39   Sophomore:  0
 3rd Qu.:37130   3rd Qu.: 75.00                                    3rd Qu.: 86.00
 Max.   :39416   Max.   :100.00                                    Max.   :100.00
   Texting      Grade
 Always:205   Fail: 43
 Rarely:184   Pass:346
```



```
> summary(M2021train[M2021train$Score > 40
+                    & M2021train$Major == "Cs"
+                   & M2021train$Seniority == "Sophomore",])
   Studentid       Attendance             Major      Questions       Score           Seniority
 Min.   :30027   Min.   :  0.00   Communication:  0   Always:171   Min.   : 41.00   Freshman :  0
 1st Qu.:32581   1st Qu.: 29.00   Cs           :345   Rarely:174   1st Qu.: 54.00   Junior   :  0
 Median :34873   Median : 55.00   Polsci       :  0                Median : 72.00   Senior   :  0
 Mean   :34821   Mean   : 51.99   Stat         :  0                Mean   : 70.74   Sophomore:345
 3rd Qu.:37020   3rd Qu.: 75.00                                    3rd Qu.: 86.00
 Max.   :39433   Max.   :100.00                                    Max.   :100.00
   Texting      Grade
 Always:162   Fail: 53
 Rarely:183   Pass:292
```



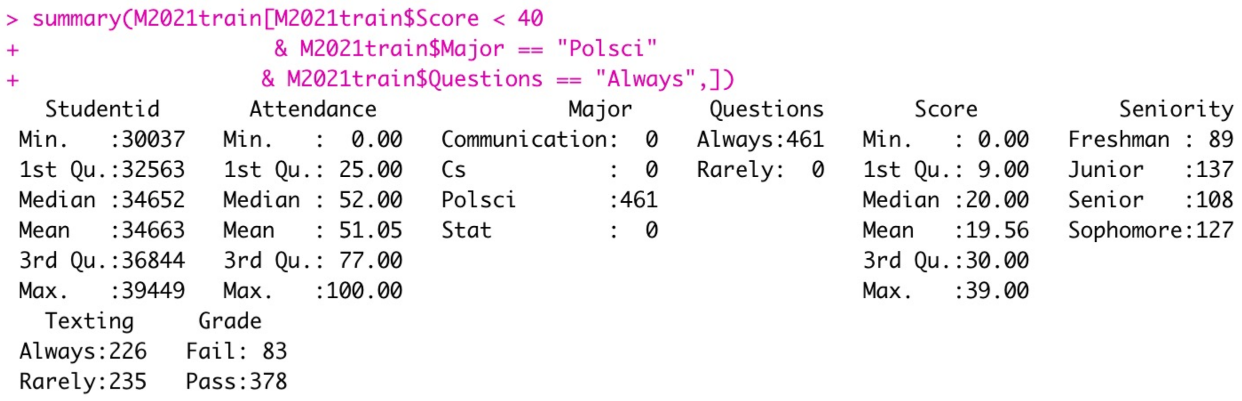
```
> summary(M2021train[M2021train$Score > 40
+                    & M2021train$Major == "Cs"
+                   & M2021train$Seniority == "Junior",])
   Studentid       Attendance             Major      Questions       Score           Seniority
 Min.   :30031   Min.   :  0.00   Communication:  0   Always:188   Min.   : 41.00   Freshman :  0
 1st Qu.:32644   1st Qu.: 22.00   Cs           :357   Rarely:169   1st Qu.: 55.00   Junior   :357
 Median :35054   Median : 48.00   Polsci       :  0                Median : 69.00   Senior   :  0
 Mean   :34926   Mean   : 47.59   Stat         :  0                Mean   : 70.15   Sophomore:  0
 3rd Qu.:37143   3rd Qu.: 74.00                                    3rd Qu.: 86.00
 Max.   :39432   Max.   :100.00                                    Max.   :100.00
   Texting      Grade
 Always:194   Fail:142
 Rarely:163   Pass:215
```



```
> summary(M2021train[M2021train$Score > 40
+                    & M2021train$Major == "Cs"
+                   & M2021train$Seniority == "Senior",])
   Studentid       Attendance             Major      Questions       Score          Seniority
 Min.   :29999   Min.   :  0.00   Communication:  0   Always:163   Min.   : 41.0   Freshman :  0
 1st Qu.:32283   1st Qu.: 25.00   Cs           :347   Rarely:184   1st Qu.: 54.0   Junior   :  0
 Median :34723   Median : 48.00   Polsci       :  0                Median : 69.0   Senior   :347
 Mean   :34738   Mean   : 50.91   Stat         :  0                Mean   : 69.2   Sophomore:  0
 3rd Qu.:37095   3rd Qu.: 78.50                                    3rd Qu.: 83.0
 Max.   :39398   Max.   :100.00                                    Max.   :100.0
   Texting      Grade
 Always:183   Fail:176
 Rarely:164   Pass:171
```

PREDICTION MODEL : THE THOUGHT PROCESS

- I then repeated this process with every combination of major and seniority level, but that did not result in any significant findings.
- Similarly, I tried many many many different combinations, in an attempt to find some relation between these various attributes.
- Using some more free predicting (I spent some time just playing around with the training dataset in Rstudio and Excel and plotting different graphs), I tried using the following combination of score, major and questions:

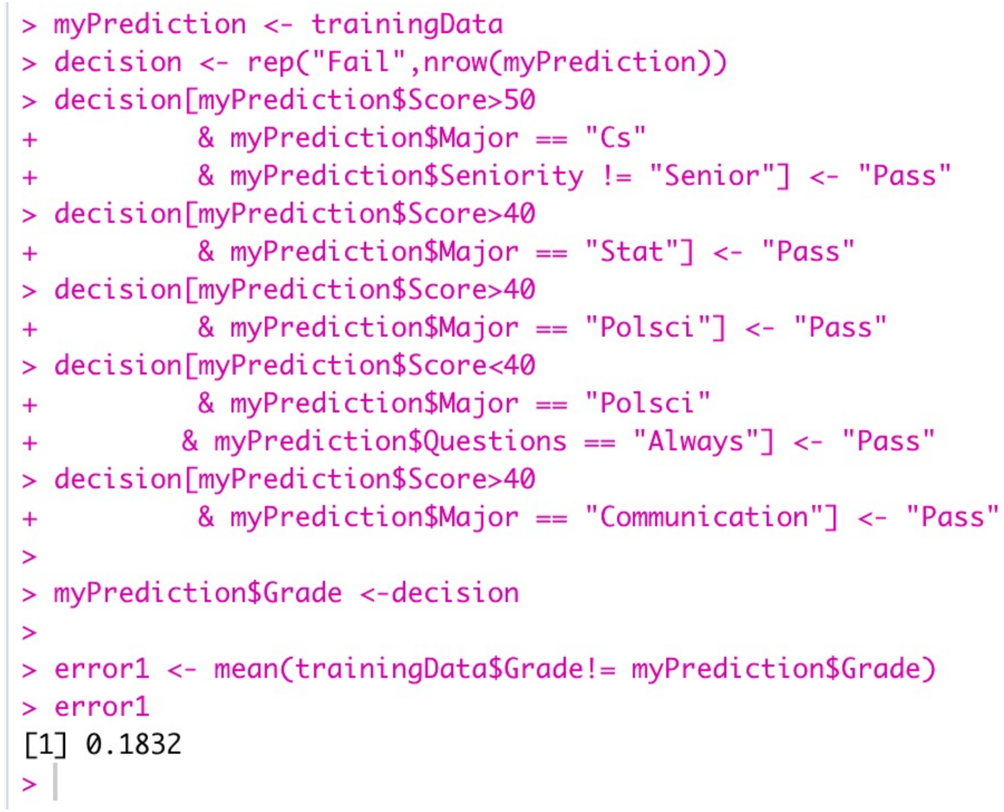


```
> summary(M2021train[M2021train$Score < 40
+                    & M2021train$Major == "Polsci"
+                  & M2021train$Questions == "Always",])
   Studentid       Attendance                Major      Questions        Score          Seniority  
 Min.   :30037   Min.   :  0.00   Communication:  0   Always:461   Min.   : 0.00   Freshman : 89  
 1st Qu.:32563   1st Qu.: 25.00   Cs           :  0   Rarely:  0   1st Qu.: 9.00   Junior   :137  
 Median :34652   Median : 52.00   Polsci       :461                Median :20.00   Senior   :108  
 Mean   :34663   Mean   : 51.05   Stat         :  0                Mean   :19.56   Sophomore:127  
 3rd Qu.:36844   3rd Qu.: 77.00                                    3rd Qu.:30.00                  
 Max.   :39449   Max.   :100.00                                    Max.   :39.00                  
   Texting       Grade    
 Always:226   Fail: 83  
 Rarely:235   Pass:378  
```

- Here, even though the score < 40, PolSci major students who always ask questions are very much more likely to pass than fail.

FINAL PREDICTION MODEL

- Trying similar combinations for all these attributes, and using them in my model, I was finally able to develop my final prediction model.



```
> myPrediction <- trainingData
> decision <- rep("Fail",nrow(myPrediction))
> decision[myPrediction$Score>50
+          & myPrediction$Major == "Cs"
+          & myPrediction$Seniority != "Senior"] <- "Pass"
> decision[myPrediction$Score>40
+          & myPrediction$Major == "Stat"] <- "Pass"
> decision[myPrediction$Score>40
+          & myPrediction$Major == "Polsci"] <- "Pass"
> decision[myPrediction$Score<40
+          & myPrediction$Major == "Polsci"
+         & myPrediction$Questions == "Always"] <- "Pass"
> decision[myPrediction$Score>40
+          & myPrediction$Major == "Communication"] <- "Pass"
>
> myPrediction$Grade <-decision
>
> error1 <- mean(trainingData$Grade!= myPrediction$Grade)
> error1
[1] 0.1832
>
```

- With an error percentage of around 18%, I applied this prediction model to my test data, several times, and was able to attain a stable error percentage of around 18% most of the times.
- Yet again, I took many more attempts at improving my prediction model, but this was the best that I could achieve.
- Submitting this to Kaggle, I earned a score of 0.82565, that is my error came out to be around 17.5% for the test dataset.