Capstone Project Data Wrangling Parker

Cassandra Parker

7/25/2019

Introduction

In my last few years of teaching Calculus I have noticed an increase in the D,W, F rates amongst students not only in Algebra sequences, but also Calculus. Students are not as prepared and sometimes have weak background of Algebra entering into the course. Calculus 1 occupies a unique position as a gateway course to science, technology, engineering, and mathematics (STEM) degrees. Almost all STEM majors need to take at least the first course in a traditional Calculus sequence. Hence for many students, this first course in the Calculus sequence is either an obstacle that they cannot overcome or a discouragement to continue in their current degree path. Many students may have felt that they were strong in mathematics in high school, but after their first college course in Calculus, they become discouraged in their abilities to continue from the unexpected rigor of the course.

Students that enroll in CMAT 111(Calculus 1), can be categorized into two groups after completion of the course; those that are successful and able to proceed to the next course by earning a grade of “C” or better (pass) and those that are not successful and unable to proceed to the next course by earning below a “C” (fail). The Mathematics Association of America (MAA) has reported the national average of unsuccessful Calculus 1 students to be 25%.

Meeting with my mentor several times and my colleague Dr. Lewis, I saw an interests to team with Dr. Lewis in exploring propensity score matching to predict/ show the students readiness for the Calculus courses. I will also take a look into if students perform better in Fall versus Spring Semesters. In my initial observation of the data the following was given; grades, majors, names, instructors, Pell grant, 1st generation college students, Old Sat scores, New Sat Scores, ACT, High School GPA, High School, Major, and Intervention. All of the categories are great but the most important information gathered will pertain to the Major, instructor, grades, ACT and SAT score, and intervention. After looking at the structure of the data, it was evident that all categories did not have any data. Not having all the data for the SAT and ACT scores, meant I cannot test the readiness students are for Calculus 1. I started with 371 observations and 18 variables. Out of the 371 variables only 117 of them had SAT scores. This approach would make me lose nearly half of my data set. However, this does give to opportunity in dive into a better understanding of propensity score matching.

Propensity Score matching is a matching technique that attempts to estimate the effectiveness of a treatment group versus a control group through observational statistics. This method also reduces the biasness due to cofounding variable.

Now, looking at the data set, I will see if the intervention in the calculus I course is effective. The intervention at Clark Atlanta University began Spring 2018 with a course redesign using the adaptive learning; Assessment and LEarning in Knowledge Spaces (ALEKS). The goal of the effort was to improve students’ mastery and the use of mathematical concepts through course redesign, assessment, and implementation using ALEKS which will enhance student mastery of learning outcomes, retention, and persistence rates in the undergraduate STEM degree programs at CAU. The ALEKS course product used was “Prep for Calculus,” a course that is designed to help students to develop the prerequisite foundation needed to learn Calculus I. Calculus I students in the course were divided into two groups. One section of the course was exposed (treatment group) to the ‘intervention”, while the other students learned Calculus I with previous teaching methods (control group).

Data wrangling is necessary to “clean up” data in order to analyze your information. The packages below were necessary for this part of the Data Wrangling project. Load the necessary packages. I had to install a few packages before loading.

My clients for this project will be senior leaders (presidents and vice presidents for academic affairs) at Clark Atlanta University. I can also include myself and the Mathematics Department Chair. The analysis of the data would help all of us see if the intervention is improving the grade D and F rates.

I will use the latest data requested from the Office of Planning, Assessment & Institutional Research at Clark Atlanta. Fall terms of 2017 and 2019 school will be used. Since propensity scoring is matching students that have the same characteristics. My approach is to clean data, and analyze scores based on SAT Mathematics Composite Scores and ACT, GPA. As well as determine the effictiveness of the Intervention being used at Clark Atlanta.

In propensity score matching or using any crucial statisics it is best that your varables are listed as 1 and 0 for yes and no respectively. The variables used are listed below:

Instructor - the person that taught the class Major - the student’s major area of study Grade - the letter grade that the student earned in the course Gender - the student’s self reported gender male = 0, female = 1 Race - Black/ African American =1, Others (Non-Resident/ Alien, unknown) = 0 1st Generation - The students that are the first to attend college. First = 0, Not the First to Attend College = 1 Score - The numeric grade that corresponds to the letter grade (90=A, 80=B, 70=C, 60=D, 50=F). Intervention - Determines the student group. Treatment =1, Control =0 Pell - The student is eligible to receive a grant from the federal government. Yes =1, No =0 Standardized Score - The standardized z-score for numeric grades.

Data wrangling

Data wrangling is necessary to “clean up” data in order to analyze your information. The packages below were necessary for this part of the Data Wrangling project. Load the necessary packages. I had to install a few packages before loading.

Data set that was provided from Clark Atlanta University has been cleaned. The techniques used were necessary in removing variables and changing names of columns, adding variables and few calculations.

The original data frame included blank variables. There were some irrelevant information that was included in the original data frame that would not be necessary for propensity score matching algorithm. Removing these variables reduce the data frame to 15 variables, though there are some variable that could be still removed, keeping them would add to interesting data.

Changing the column Names:There are 18 variables included in the data frame, to which majority of the variables went through a name change. As the data was imported into R studio, there were several names that were too long and had complexities of “_”. Concise name were given to provide more accurate information in the columns and make the names shorter.

Adding Variables and Data: Two variables were necessary for building the model. A treatment column and a column that displayed standardized z scores for the final grade earned. In the treatment column, an if-else statement was written to place a 1 for the observation that was treated and a 0 otherwise. This information helped to accurately and easily determine the control and treatment groups. Freshman students had various scores for their Sat. Since there is a new version of the SAT, it was best to convert the Old Sat scores to the new SAT scores and combine them in one column. Since there are only 117 observations of this it is best to not eliminate this information.

Standardized z-score: In the standardized z-score column, a function was written to include a standardized score for each observation in the “Score” column of the data frame. This standardized z-score normalizes the data and has a mean of 0 and a standard deviation of 1. It represents the signed fractional number of standard deviations by which the value of an observation or data point lies above or below the mean value of the data set that is measured. Values above the mean have positive standard scores, while values below the mean have negative standard scores. Adding this column increased the data frame to 15 variables.

```
library(tidyverse)
```

Load the Fall Semester data for 2017 and 2018.

```
capdat <- read_csv("capdatar.csv")
```

First take a look at our data. We will notice the number of variables, observations, and column names before proceeding.

```
glimpse(capdat)
```

```
## Observations: 371
## Variables: 18
## $ ACADEMIC_TERM              <dbl> 201709, 201709, 201709, 201709, 201...
## $ COURSE_NUMBER              <chr> "CMAT 111", "CMAT 111", "CMAT 111",...
## $ COURSE_INSTRUCTOR          <chr> "Harlemon, Maxine", "Jalloh, Mohame...
## $ STUDENT_ID                 <dbl> 900743209, 900721877, 900737740, 90...
## $ `STUDENT_NAME (L, F, MI)`  <chr> "Turner, Kailen A", "Devers, Ciera ...
## $ DECLARED_MAJOR_DURING_TERM <chr> "Biology", "Computer Science", "Bio...
## $ COURSE_FINAL_GRADE         <chr> "A", "A", "C", "A", "B", "C", "C", ...
## $ `Num grade`                <dbl> 90, 90, 70, 90, 80, 70, 70, 50, 80,...
## $ NEW_SAT_MATH_SCORE         <dbl> 690, 650, 590, 580, 570, 570, 570, ...
## $ OLD_SAT_MATH_SCORE         <dbl> 670, 650, 590, 560, 560, 560, 550, ...
## $ ACT_COMPOSITE_SCORE        <dbl> 31, 30, 27, 25, 25, 25, 24, 24, 24,...
## $ HIGH_SCHOOL                <chr> "Las Vegas Academy Intl Studies", "...
## $ HSGPA                      <dbl> 4.60, 3.42, 2.63, 3.60, 2.44, 4.22,...
## $ GENDER                     <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ RACE_ETHNICITY             <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ PELL_RECIPIENT             <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ `1st_GENERATION`           <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Intervention               <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
```

It is best to select the variables of interest. Therefore, remove the columns that are not necessary. Since all data is already Calculus courses from Fall semesters 2017 and 2018, we can eliminate those columns along with a few other columns, such as highschool and hsgpa.Let us only select the variables of interest for us.

```
capdat<-dplyr::select(capdat, -c(COURSE_NUMBER, STUDENT_ID, ACADEMIC_TERM,  HIGH_SCHOOL, HSGPA))
```

```
view(capdat)
```

Another concept in data wrangling is to make things more simple and clear. Rename the cloumn names to be more intuitive.

```
colnames(capdat) <- c('Instructor', 'Student', 'Major', 'Grade', 'Score', 'NewSAT', 'OldSAT', 'A
CT', 'Gender', 'Race', 'Pell', 'Generation', 'Intervention' )
```

Old Sat Scores need to be converted to format with new Sat Scores. If score is between, 300-399 then add 50 points, if score 400-499 then add 40 points, if score between 500-599 then add 30 points, if score between 600-699, add 20 points.

```
capdat <- capdat %>% mutate(NewSATScore =
  case_when(
    OldSAT >= 300 & OldSAT < 400 ~ (OldSAT + 50),
    OldSAT >= 400 & OldSAT < 500 ~ (OldSAT + 40),
    OldSAT >= 500 & OldSAT < 600 ~ (OldSAT + 30),
    OldSAT >= 600 & OldSAT < 700 ~ (OldSAT + 20)
  )
)
```

Include a function to the added column that calculates the standardized score. We can use the `scale` function for doing the same.

```
capdat <- capdat %>% mutate(
  Std_Score = as.numeric(scale(Score))
)
```

Create a csv file of the clean data frame for the project submission.

```
write.csv(capdat, file = "capdat.csv")
```

Exploratory Data Analysis

Now that we have our clean data, we can start exploring it. **Exploratory Data Analysis** or EDA, is where we start asking questions about our data and start answering them through plots and graphs. Before answering questions, it is also a good idea to get a feel of how many missing values we have in our dataset. The `colSums()` function is one way of doing it.

```
colSums(is.na(capdat))
```

```
##   Instructor      Student        Major        Grade        Score
##            0            0            0            0            0
##       NewSAT       OldSAT          ACT       Gender         Race
##          295          271          256            0           15
##         Pell   Generation Intervention  NewSATScore    Std_Score
##            0            0            0          271            0
```

We can see from looking at that data that there are a few columns missing data. Plotting data we will use most data that is not missing any values to draw some different observations. Take a look at a few factors, plot grades versus intervention.

Seeing that `Grade` has no missing values, we can base our analysis on this variable. Let us see based on the `Grade` the performance of the students who received intervention or not.

```
(grade_tbl <- capdat %>% select(Grade, Intervention) %>% table())
```

```
##      Intervention
## Grade  0  1
##     A 78 40
##     B 61 40
##     C 46 30
##     D 33  6
##     F 26 11
```

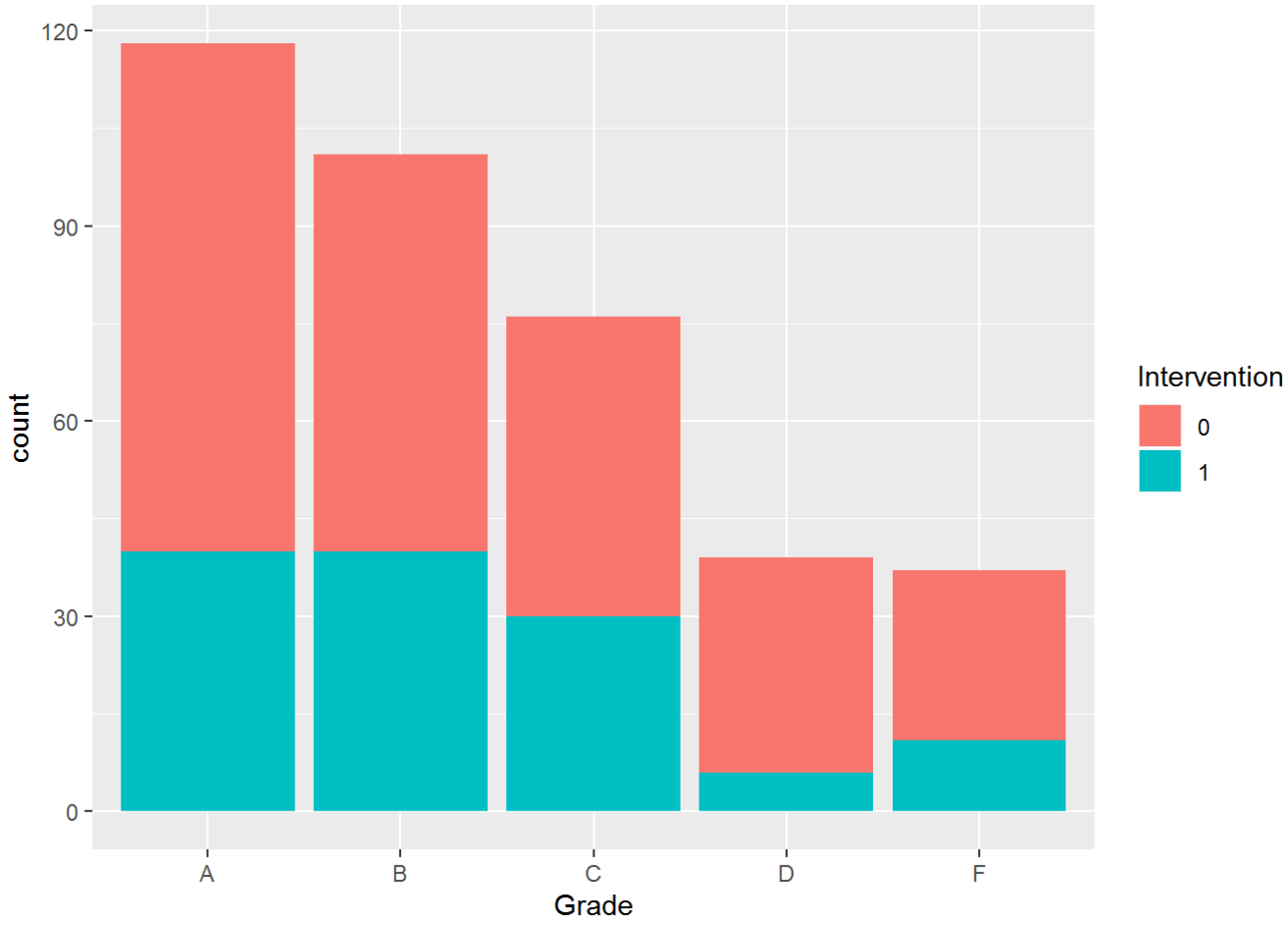
We can also see the above information as a proportion.

```
grade_tbl %>% prop.table()
```

```
##      Intervention
## Grade          0          1
##     A 0.21024259 0.10781671
##     B 0.16442049 0.10781671
##     C 0.12398922 0.08086253
##     D 0.08894879 0.01617251
##     F 0.07008086 0.02964960
```

Take a look at a few factors, plot grades versus intervention.

```
capdat$Intervention <- as.factor(capdat$Intervention)
ggplot(capdat, aes(x = Grade, fill = Intervention)) + geom_bar()
```

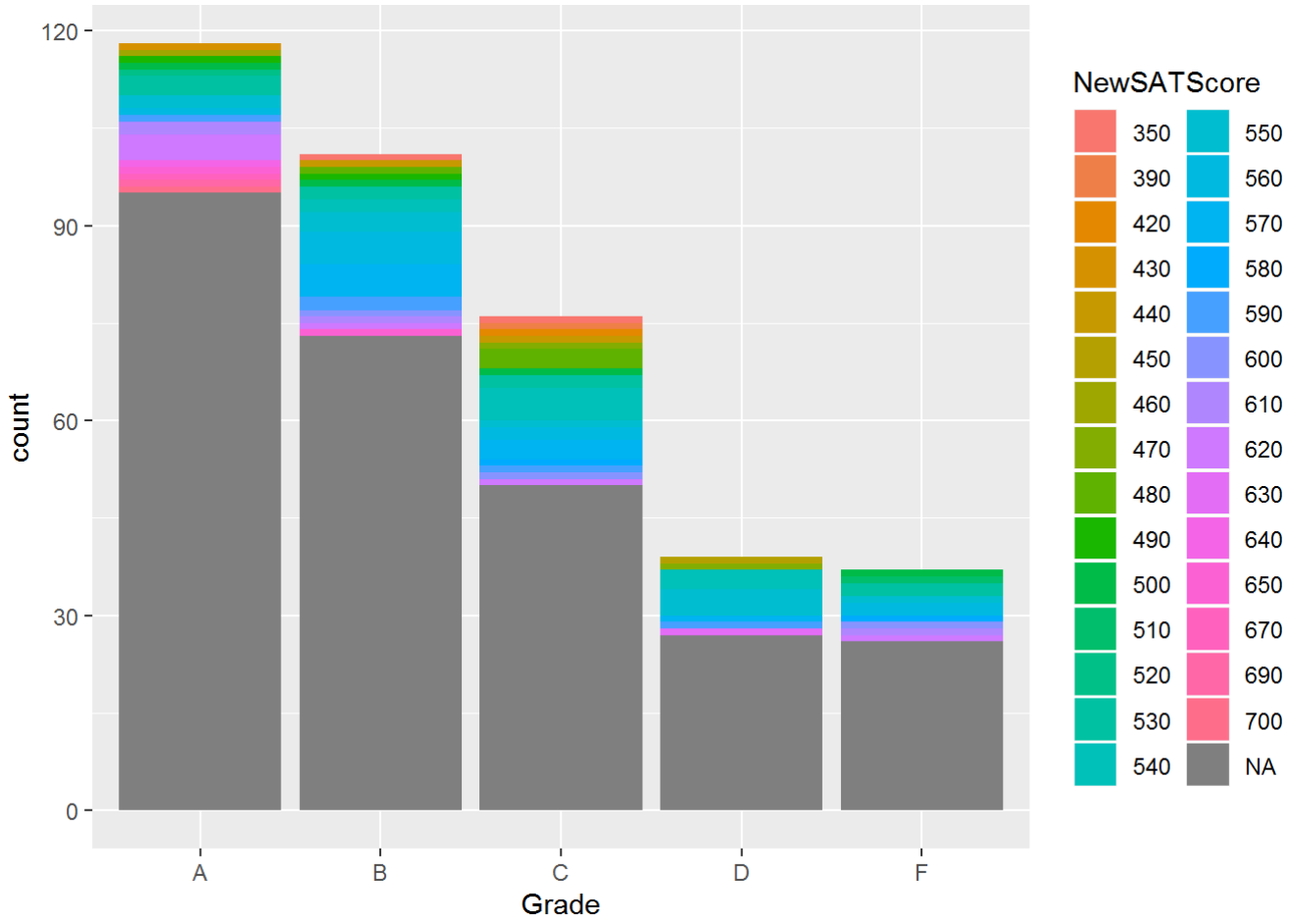



```
capdat %>% select(Intervention) %>% table() %>% prop.table()
```

```
## .
##         0         1
## 0.6576819 0.3423181
```

Our proportion tells us that 65% of the students did not have intervention (control group)and 34% did. Though the control group has a significant amount of students/observations, you can see the drastic decrease in the D and F grade categories. This is great defeat.

```
capdat$NewSATScore <- as.factor(capdat$NewSATScore)
ggplot(capdat, aes(x = Grade, fill = NewSATScore)) + geom_bar()
```



As we can see from above in our column sums, there are quite a few missing values for the NewSat scores. Whereas `Score` and subsequently `Std_Score` have no missing values. It makes sense to work with the scores and Std scores. However we may revisit looking at the SAT scores since they simply deal with the Freshman class on this later. Were the freshman students prepared based upon SAT or ACt scores, did they need as much intervention? Perhaps at a later project we can determine this there for well will clean our data of these values.

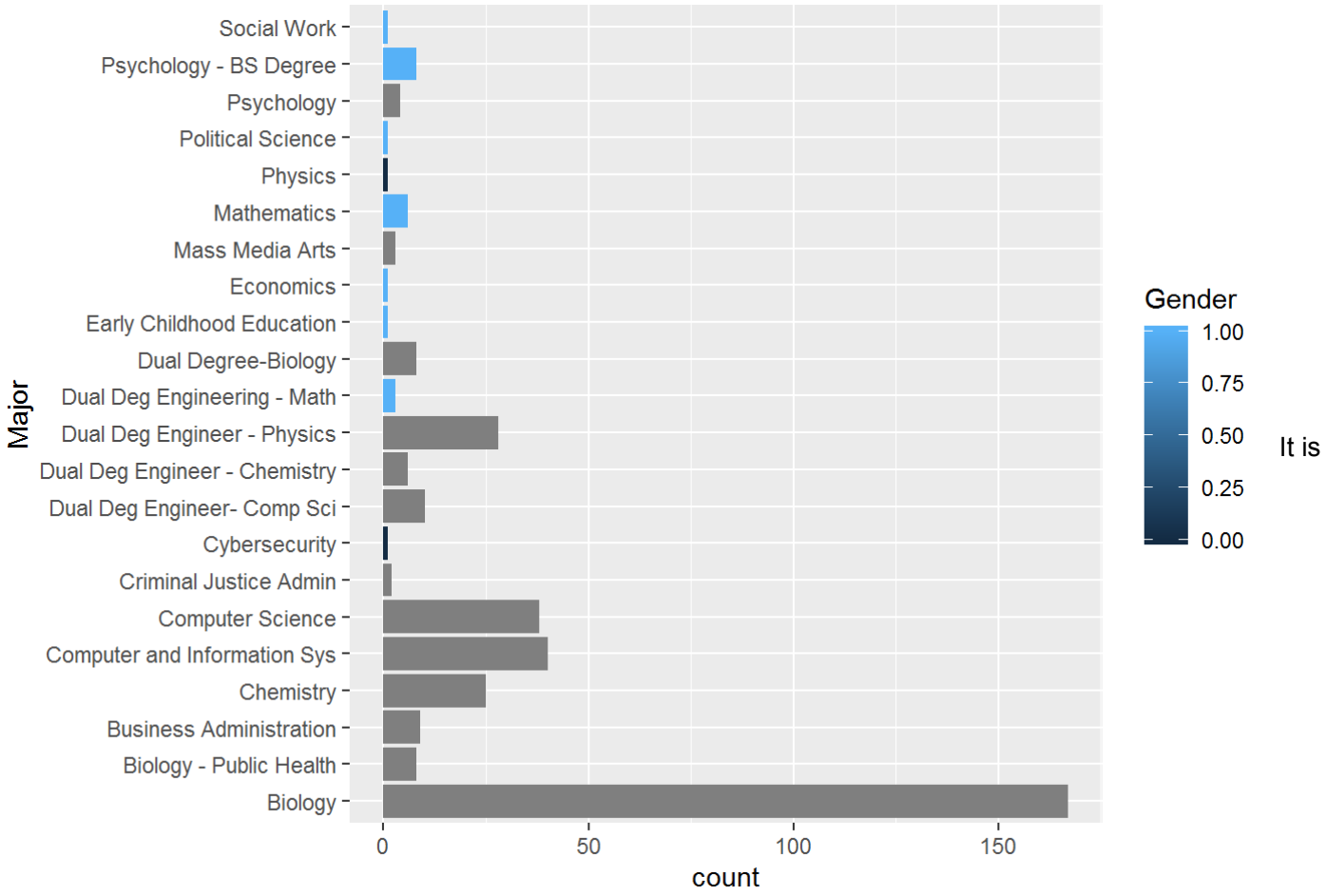
```
capdat<-dplyr::select(capdat, -c(NewSATScore, OldSAT, NewSAT, ACT))
```

Now, lets take a look at the majors and gender.

```
capdat %>% select(Gender) %>% table() %>% prop.table()
```

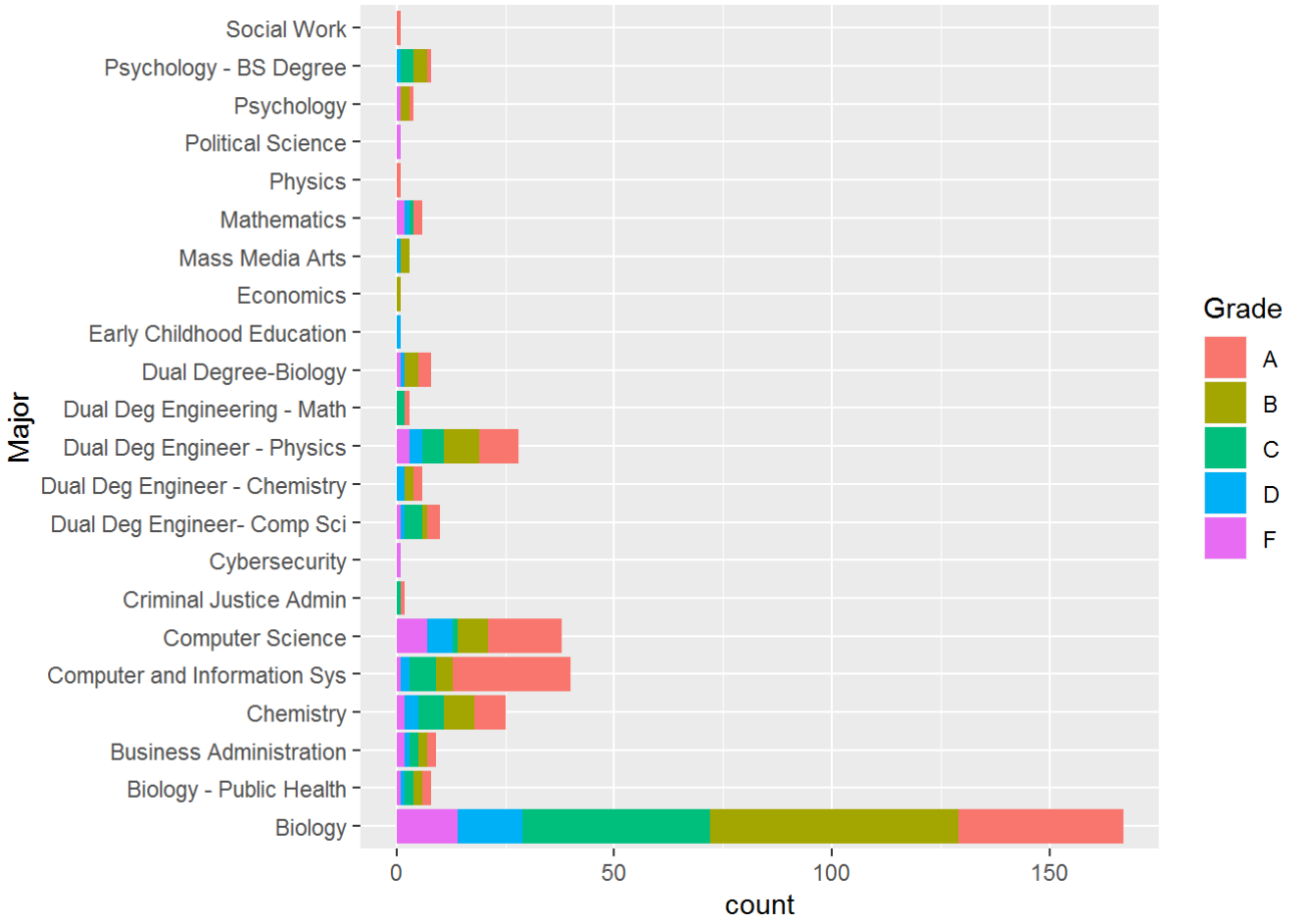
```
## .
##         0         1
## 0.3180593 0.6819407
```

```
ggplot(capdat, aes(x=Major, fill = Gender)) + geom_bar() + coord_flip()
```



no suprise there are more females than males since Clark atlanta is 72% female and 28% male. Majority majors with the exception of computer sciences fields are mostly female driven.It can be seen that majority of the majors are Biology, computer science, conmputer and Information Systems, Chemistry, and Dual degree Physics. Perhaps looking at the majors versus the grades would be a good observation as well.

```
capdat$Major <- as.factor(capdat$Major)
ggplot(capdat, aes(x = Major, fill = Grade)) + geom_bar()+ coord_flip()
```



Looking at the heavily populated majors we might be able to dig a little deeper and looking at those soley? Here below we can see exactly how many students are in each major. Biology being the largest number of students, is primarily because most students who major in Biology want to pursue their careers in the medical fields. Computer Information Systems, Computer Science, dual Degree-Physics and Chemistry are listed with the top five majors as well.

```
capdat %>% group_by(Major) %>% count() %>% arrange(desc(n))
```

```
## # A tibble: 22 x 2
## # Groups:   Major [22]
##    Major                            n
##    <fct>                        <int>
##  1 Biology                        167
##  2 Computer and Information Sys    40
##  3 Computer Science                38
##  4 Dual Deg Engineer - Physics     28
##  5 Chemistry                       25
##  6 Dual Deg Engineer- Comp Sci     10
##  7 Business Administration          9
##  8 Biology - Public Health          8
##  9 Dual Degree-Biology              8
## 10 Psychology - BS Degree           8
## # ... with 12 more rows
```

Since we have 22 rows of different majors, lets Call a new data frame major_count and look at the majors greater than 7. Then print out the result.

```
major_count <- capdat %>% group_by(Major) %>% count() %>% filter(n >7) %>%arrange(desc(n))
major_count
```

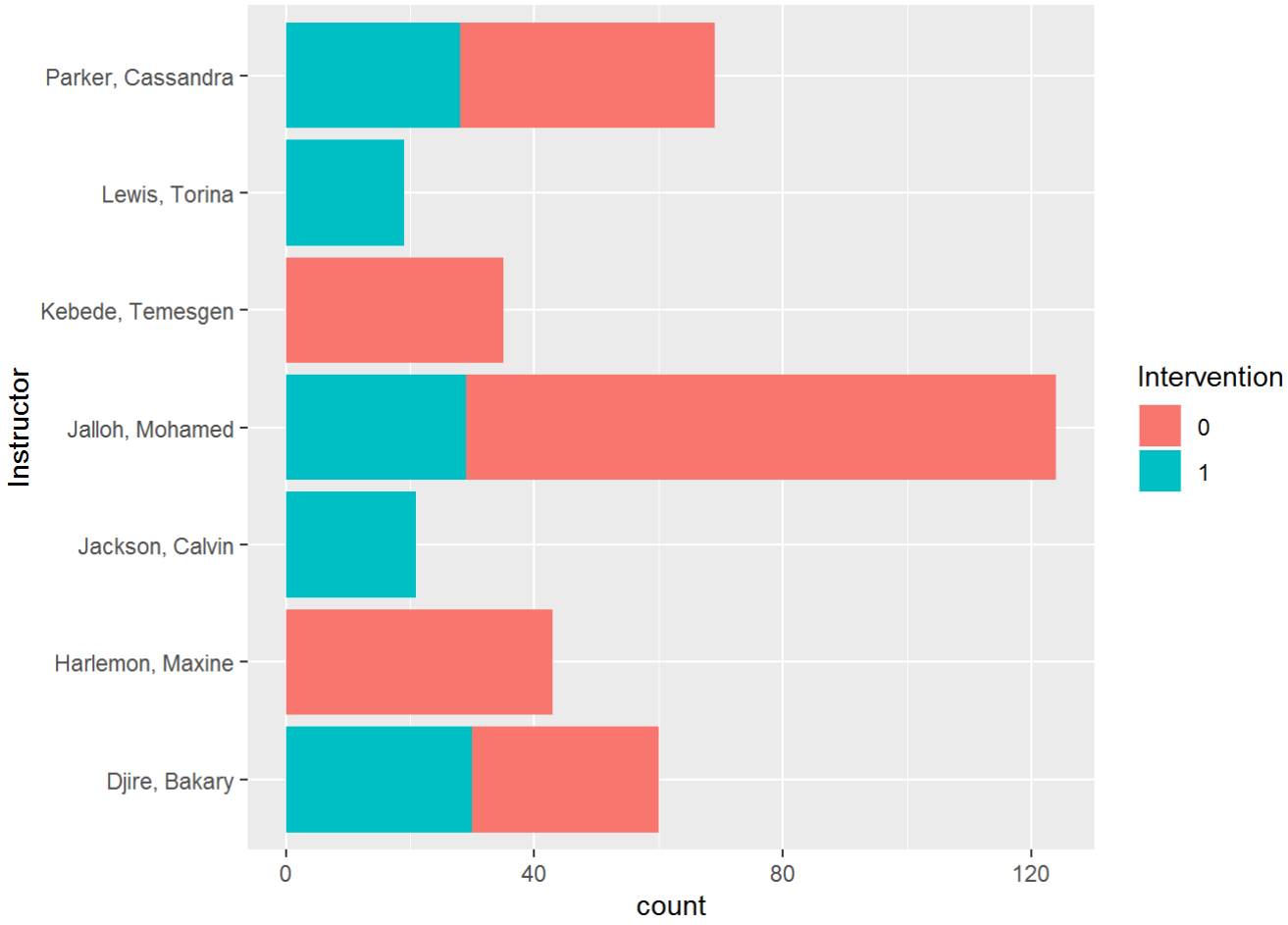
```
## # A tibble: 10 x 2
## # Groups:   Major [10]
##    Major                            n
##    <fct>                        <int>
##  1 Biology                        167
##  2 Computer and Information Sys    40
##  3 Computer Science                38
##  4 Dual Deg Engineer - Physics     28
##  5 Chemistry                       25
##  6 Dual Deg Engineer- Comp Sci     10
##  7 Business Administration          9
##  8 Biology - Public Health          8
##  9 Dual Degree-Biology              8
## 10 Psychology - BS Degree           8
```

Lets take a look at the instructors for a bit. It appears there was a heavy load of of students taught from Mr. Jallohs class. My self second, and Mr. Bakray third. Lets dive a little deeper to see the number of students that had intervention from the courses in respect to the instructor.

```
capdat %>% group_by(Instructor) %>% count() %>%  arrange(desc(n))
```

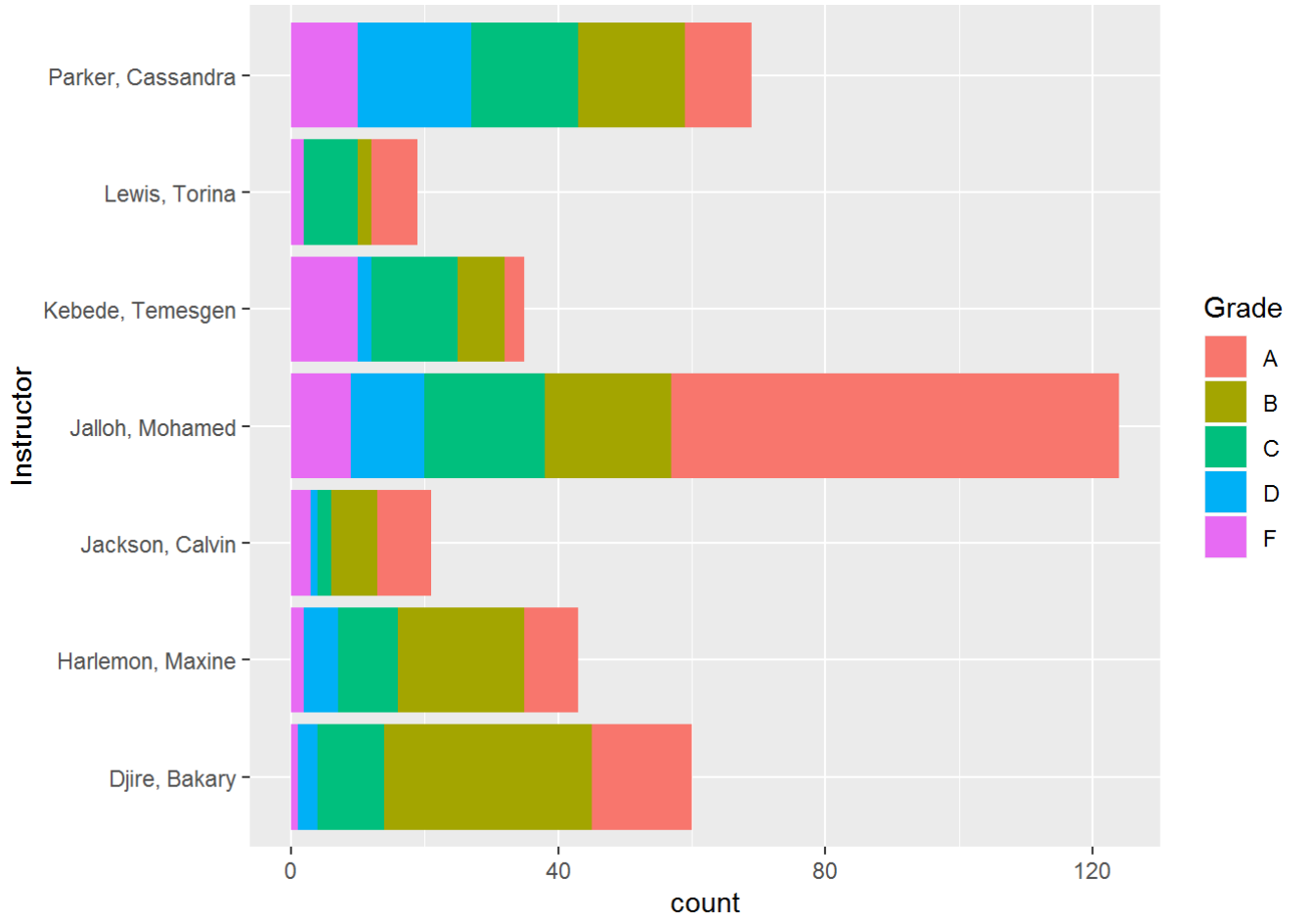
```
## # A tibble: 7 x 2
## # Groups:   Instructor [7]
##   Instructor            n
##   <chr>             <int>
## 1 Jalloh, Mohamed     124
## 2 Parker, Cassandra    69
## 3 Djire, Bakary        60
## 4 Harlemon, Maxine     43
## 5 Kebede, Temesgen     35
## 6 Jackson, Calvin      21
## 7 Lewis, Torina        19
```

```
capdat$Intervention <- as.factor(capdat$Intervention)
ggplot(capdat, aes(x = Instructor, fill = Intervention)) + geom_bar()+coord_flip()
```



Wow, its pretty cool to see that the heavily student populated instructors have the mixture of students with intervention versus not. It is possbile that students take certain professors in school year? Lets take one more look at grades, intervention, and instructor.

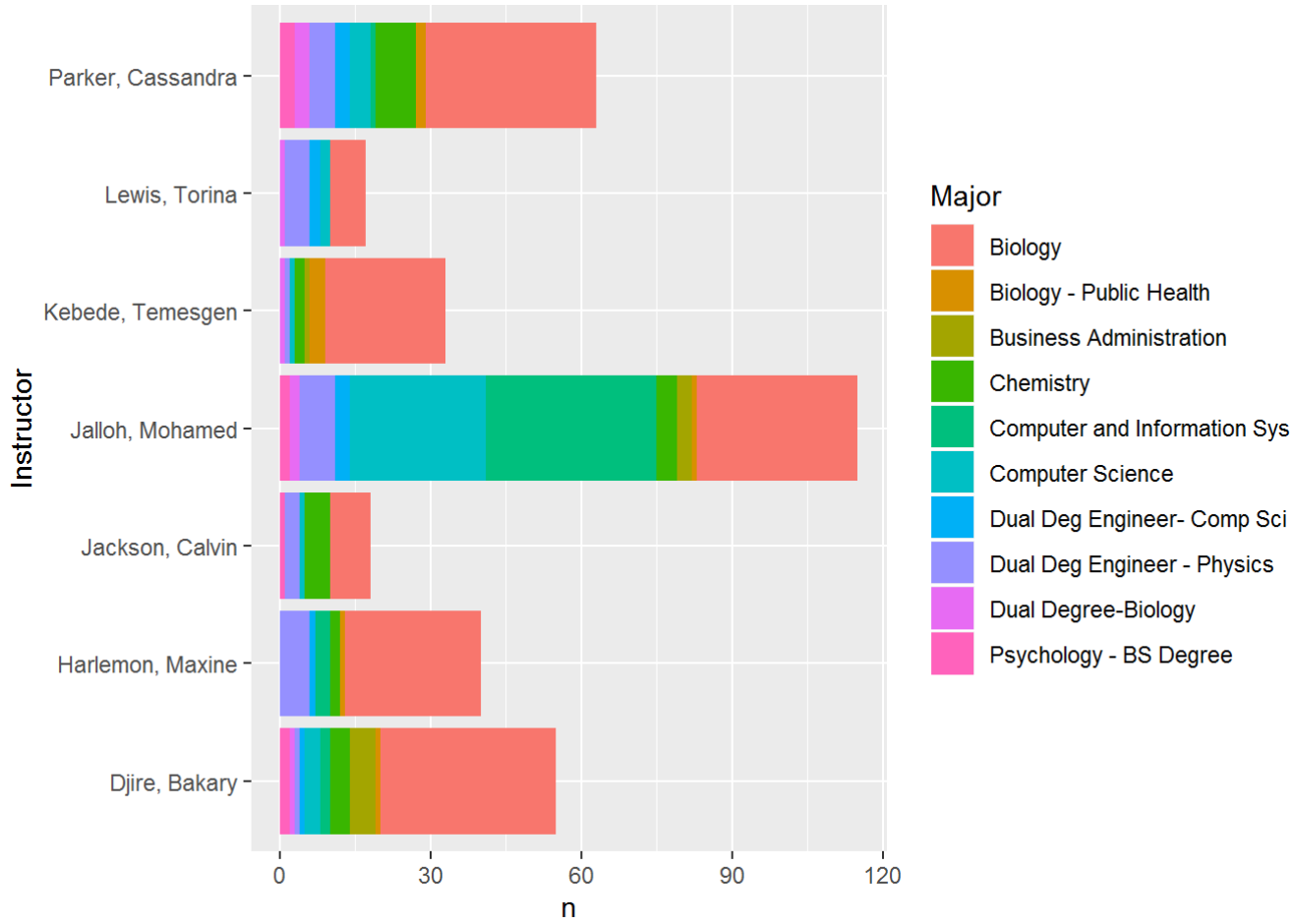
```
capdat$Grade <- as.factor(capdat$Grade)
ggplot(capdat, aes(x = Instructor, fill = Grade)) + geom_bar()+coord_flip()
```




```
inst_maj <- capdat %>% group_by(Instructor, Major) %>% count() %>% arrange(desc(n))
inst_maj
```

```
## # A tibble: 77 x 3
## # Groups:   Instructor, Major [77]
##    Instructor        Major                          n
##    <chr>             <fct>                      <int>
##  1 Djire, Bakary     Biology                       35
##  2 Jalloh, Mohamed   Computer and Information Sys    34
##  3 Parker, Cassandra Biology                       34
##  4 Jalloh, Mohamed   Biology                       32
##  5 Harlemon, Maxine  Biology                       27
##  6 Jalloh, Mohamed   Computer Science              27
##  7 Kebede, Temesgen  Biology                       24
##  8 Jackson, Calvin   Biology                        8
##  9 Parker, Cassandra Chemistry                      8
## 10 Jalloh, Mohamed   Dual Deg Engineer - Physics     7
## # ... with 67 more rows
```

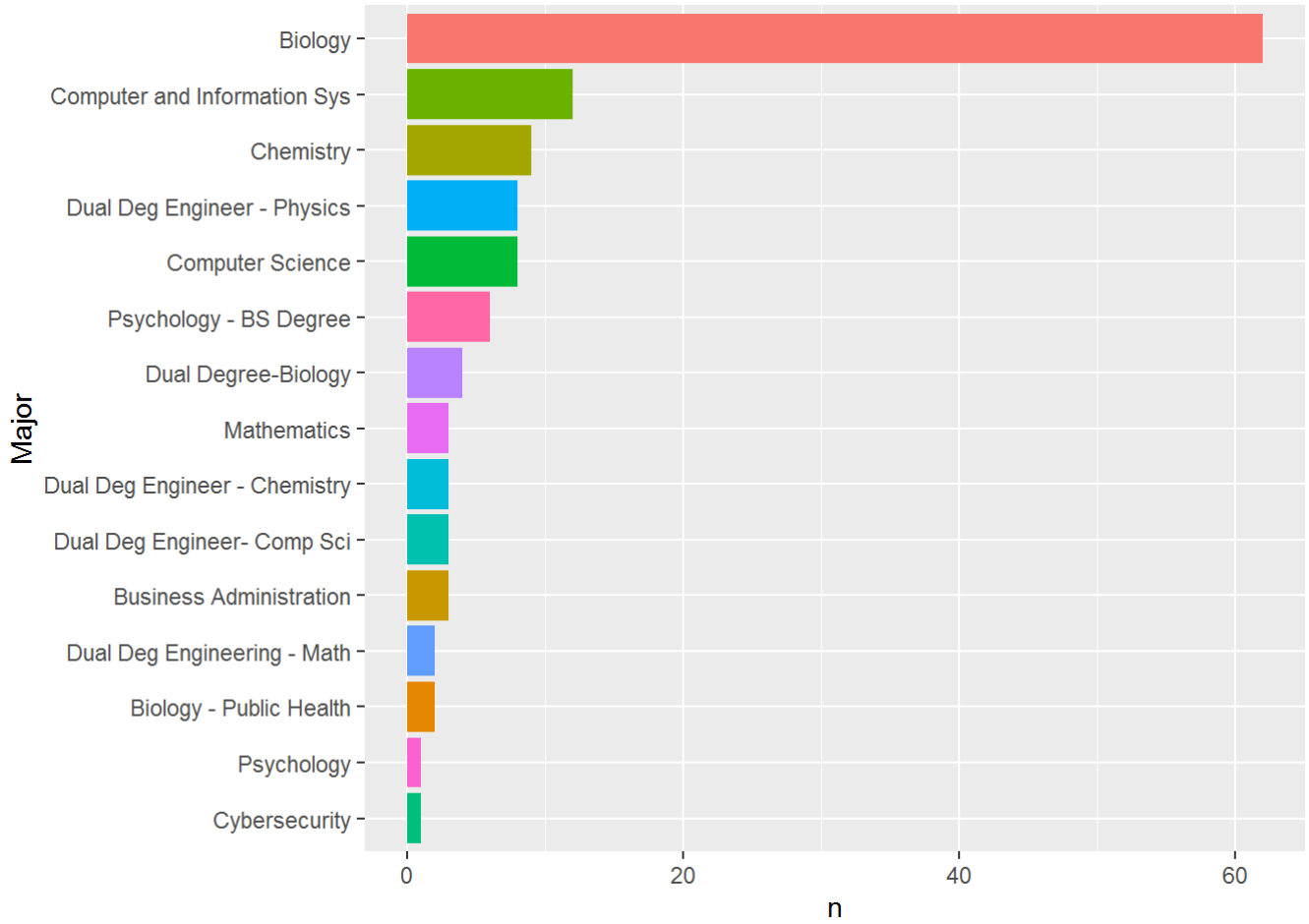
```
ggplot(inst_maj %>% filter(Major %in% unique(major_count$Major)) , aes(x = Instructor, y = n, fi
ll = Major)) + geom_bar(stat = 'identity') +
  coord_flip()
```



This is a great visual representation of the total number of students each professor taught, filled with the students particular major.

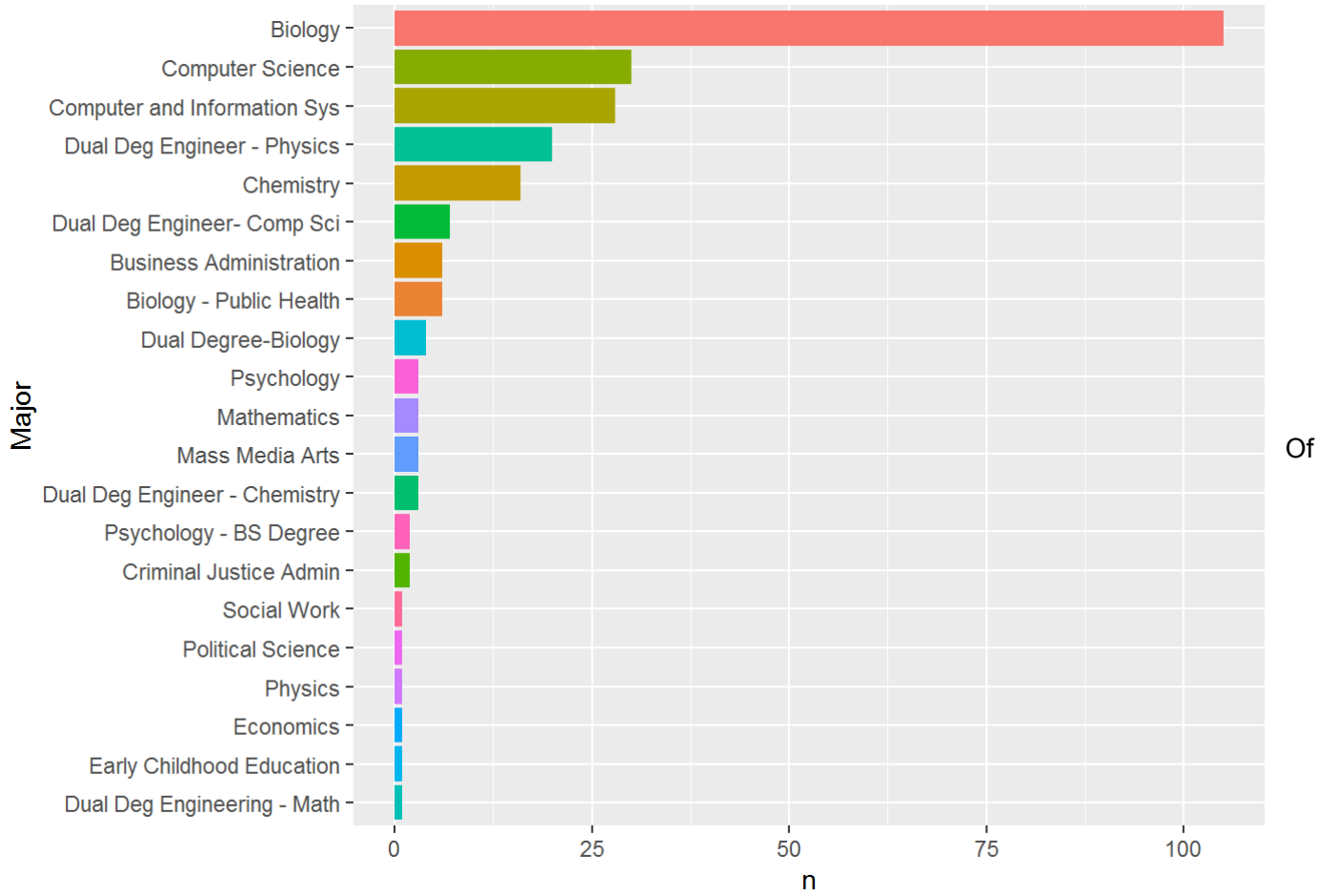
Perhaps we can consider the Intervention control and treated groups in relation to the major

```
capdat %>% filter(Intervention == 1) %>% count(Major) %>%
  ggplot(aes(x = reorder(Major, n), y = n, fill = Major)) + geom_bar(stat = 'identity') + coord_
flip() +
  xlab('Major') + theme(legend.position = "none")
```



No intervention

```
capdat %>% filter(Intervention == 0) %>% count(Major) %>%
  ggplot(aes(x = reorder(Major, n), y = n, fill = Major)) + geom_bar(stat = 'identity') + coord_
flip() +
  xlab('Major') + theme(legend.position = "none")
```



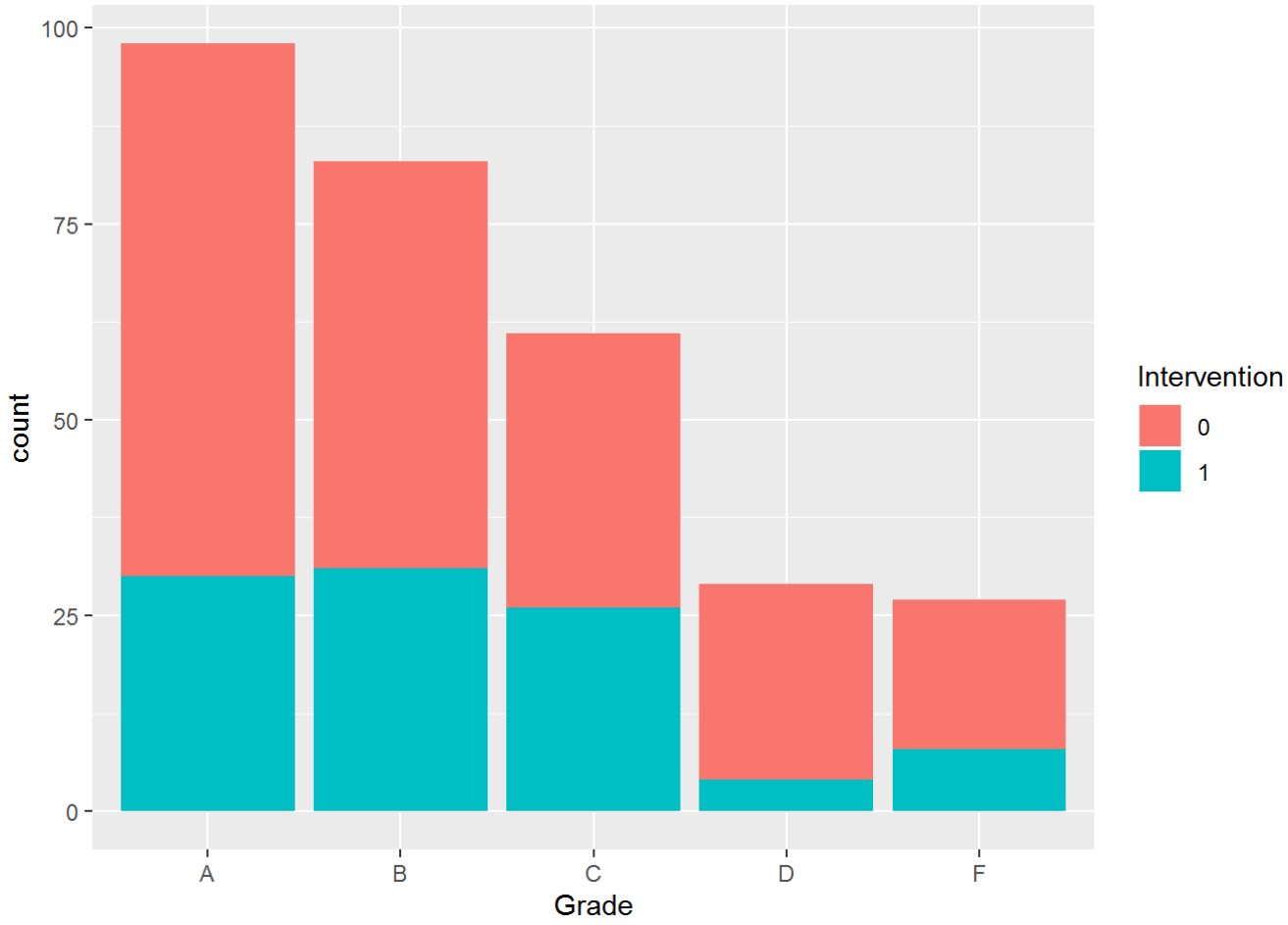
Of course Biology would have the most treated students vs the contol group. However it is interesting to see that top five majors as stated before are mostly from the control group and the invention group, just in a different order.

```
new_data <- filter(capdat, Major == "Biology" | Major == "Computer Science" | Major == "Computer
 and Information Sys" | Major == "Chemistry" | Major == "Dual Deg Engineer - Physics")
```

```
major_count1 <- new_data %>% group_by(Major) %>% count()
major_count1
```

```
## # A tibble: 5 x 2
## # Groups:   Major [5]
##   Major                            n
##   <fct>                        <int>
## 1 Biology                        167
## 2 Chemistry                       25
## 3 Computer and Information Sys    40
## 4 Computer Science                38
## 5 Dual Deg Engineer - Physics     28
```

```
ggplot(new_data, aes(x = Grade, fill = Intervention)) + geom_bar()
```



##Statistical Analysis on Capstone

Now that I have cleaned, wrangled and slightly explored my data. Let's get into the statistical analysis. I have created several different plots in the exploratory analysis. Plunging a little deeper into some statistics from our data, we might find some more insightful information.

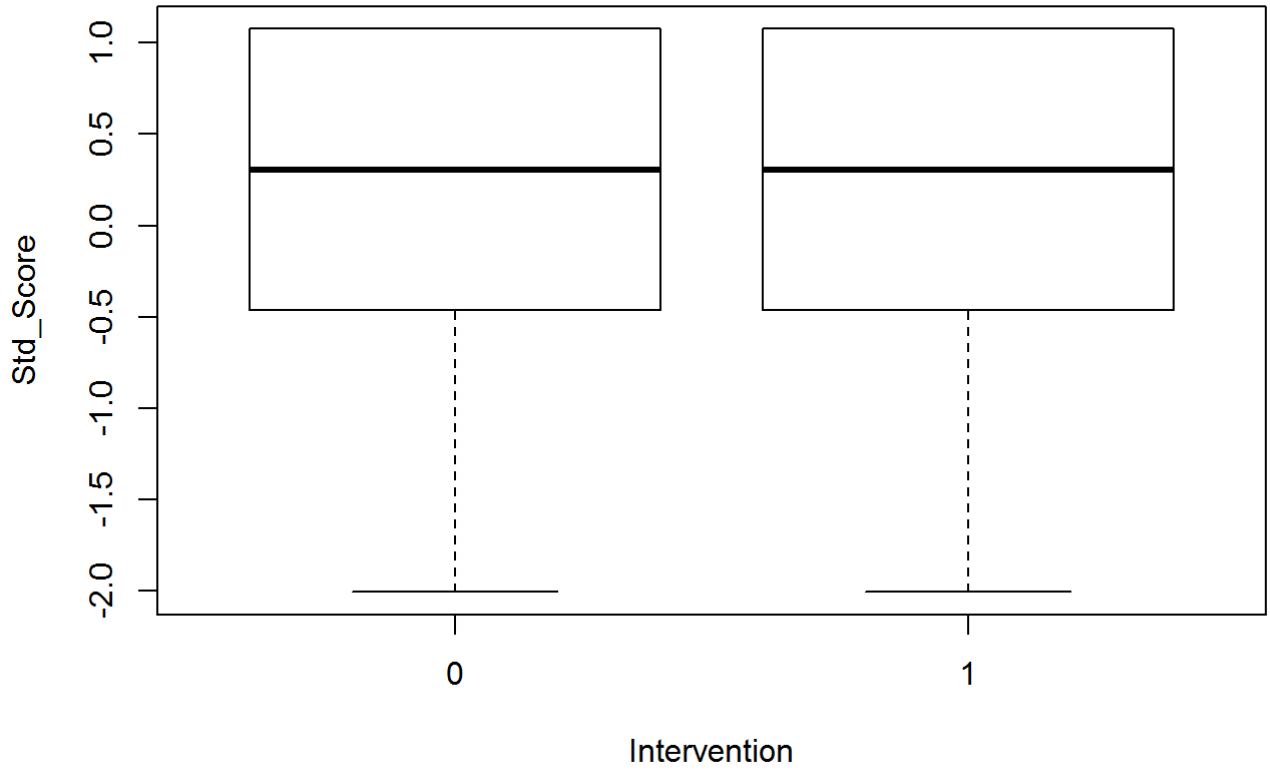
```
dat_stats <- read.csv("capdat.csv")
```

Now, the libraries that are necessary to show the skills learned are loaded into are loaded.

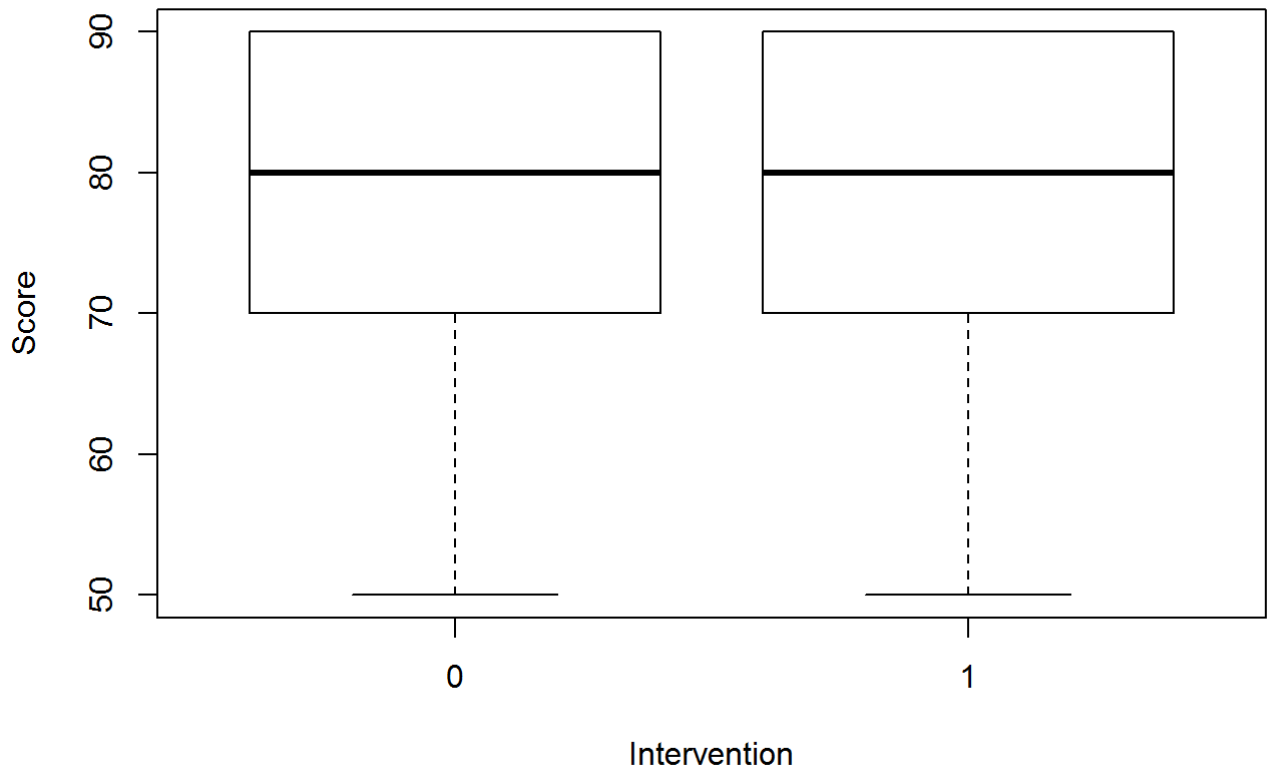
```
library(tidyverse)
```

Lets first create a box plot for the standard scores and scores.

```
boxplot(Std_Score~Intervention, data = dat_stats)
```

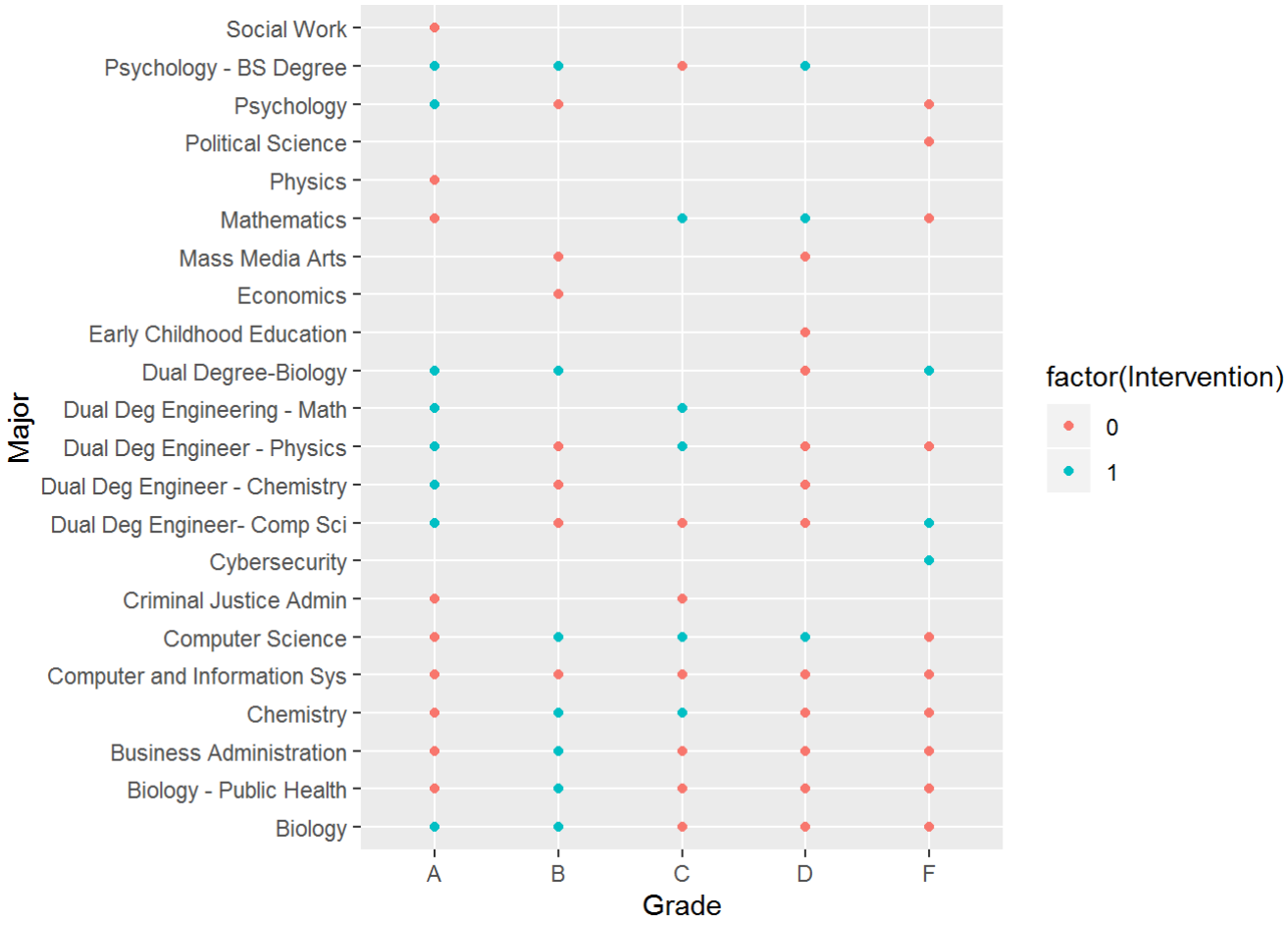



```
boxplot(Score~Intervention, data = dat_stats)
```



Scatter plots are good tools to observe the relationships between variables. The data divided by the Intervention type, is plotted showing the particular earned grade by major.

```
ggplot(dat_stats, aes(x = Grade, y = Major, color=factor(Intervention))) +
  geom_point()
```



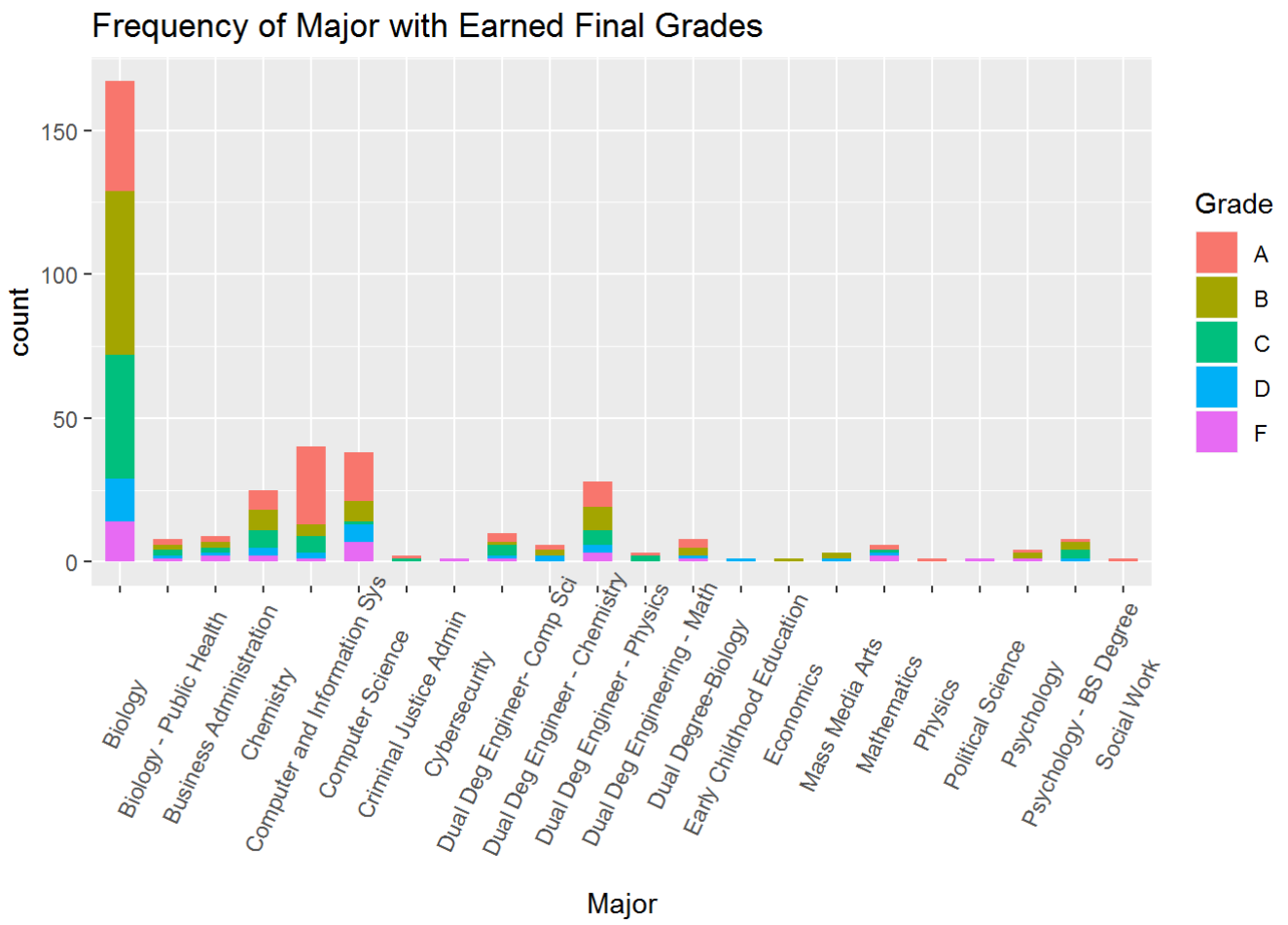
Here is a table of the number of students that earned a particular grade is displayed.

```
table(dat_stats$Grade)
```

```
##
##   A   B   C   D   F
## 118 101  76  39  37
```

This particular histogram determines the major that takes the course more frequently, as well as the breakdown of grades by major.Looking at the plots created, there are quite a few majors with very small numbers. We will need to filter the data to eliminate such values later.

```
dat_hist <- ggplot(dat_stats, aes(Major))
dat_hist + geom_bar(aes(fill=Grade), width = 0.6) +
  theme(axis.text.x = element_text(angle=65, vjust=0.6)) +
  labs(title="Frequency of Major with Earned Final Grades")
```



Let's first determine the difference in means between the control and treated groups. I believe it is vital to our data the we find them separately. Thus the means for these groups are calculated for the outcome variable. We also determine the number of students in each group. The treatement and control group are labelled 1, 0 respectively.

```
dat_stats %>%
  dplyr::group_by(Intervention) %>%
  dplyr::summarise(n_students = n(),
            mean_Score = mean(Score),
            std_error = sd(Score) / sqrt(n_students))
```

```
## # A tibble: 2 x 4
##   Intervention n_students mean_Score std_error
##          <int>      <int>      <dbl>     <dbl>
## 1            0        244       75.4     0.860
## 2            1        127       77.2     1.07
```

Also, run the same above code but this time we use the standard score. The standard score is the standard z-score.

```
dat_stats %>%
  dplyr:: group_by(Intervention) %>%
  dplyr:: summarise(n_students = n(),
            mean_Std_Score = mean(Std_Score),
            std_error = sd(Std_Score) / sqrt(n_students))
```

```
## # A tibble: 2 x 4
##   Intervention n_students mean_Std_Score std_error
##          <int>      <int>          <dbl>     <dbl>
## 1            0        244        -0.0483    0.0662
## 2            1        127         0.0928    0.0824
```

The mean of the observation, or the entire population might have some value to us later.

```
mean(dat_stats$Score)
```

```
## [1] 76.03774
```

I will also attempt to use a t-test. A t test tells you how signiicant the differences between the score and std_score are. In simplest terms it will let us know if the measured averages of the groups could have happened by chance. If our data gives us low p-values such as p < 0.05 then, these are good values and indicate our data did not occur by chance. The greater the p value means the more like the intervention just happened “by chance”. It can also compare the means of the control and treatment groups to determine if their is a difference in the means.

```
with(dat_stats, t.test(Std_Score ~ Intervention))
```

```
##
##  Welch Two Sample t-test
##
## data:  Std_Score by Intervention
## t = -1.3358, df = 280.57, p-value = 0.1827
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.34917054  0.06685382
## sample estimates:
## mean in group 0 mean in group 1
##     -0.04832106      0.09283731
```

```
with(dat_stats, t.test(Score ~ Intervention))
```

```
##
##  Welch Two Sample t-test
##
## data:  Score by Intervention
## t = -1.3358, df = 280.57, p-value = 0.1827
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -4.5372375  0.8687206
## sample estimates:
## mean in group 0 mean in group 1
##        75.40984        77.24409
```

Our p values are greater than 0.05, which means the grades could have happened by chance.

Lets look at the data, eliminating those majors that are extremely low. Let use the data sat that was created in out exploratory anaylsis new_data.

Determine the difference in means between the control and treated groups.

```
new_data %>%
  dplyr::group_by(Intervention) %>%
  dplyr::summarise(n_students = n(),
            AvgScore = mean(Score),
            std_error = sd(Score) / sqrt(n_students))
```

```
## # A tibble: 2 x 4
##   Intervention n_students AvgScore std_error
##   <fct>             <int>    <dbl>     <dbl>
## 1 0                   199     76.3     0.938
## 2 1                    99     77.2     1.18
```

The std_error is quite low, this is actually a good thing. The smaller the error, the less the spread and the more less the spread, the more likely the mean is closest to the population mean.

Now, calculate the differences in means for the standardized score “Std_Score” grouping by treatment (1) and control (0) groups for the outcome variable.

```
new_data %>%
  dplyr:: group_by(Intervention) %>%
  dplyr:: summarise(n_students = n(),
            AvgStdScore = mean(Std_Score),
            std_error = sd(Std_Score) / sqrt(n_students))
```

```
## # A tibble: 2 x 4
##   Intervention n_students AvgStdScore std_error
##   <fct>             <int>       <dbl>     <dbl>
## 1 0                   199      0.0188    0.0722
## 2 1                    99      0.0873    0.0911
```

Again a t test tells you how signiicant the differences between the score and std_score are.

```
with(new_data, t.test(Std_Score ~ Intervention))
```

```
##
##  Welch Two Sample t-test
##
## data:  Std_Score by Intervention
## t = -0.58937, df = 217.15, p-value = 0.5562
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2976417  0.1606112
## sample estimates:
## mean in group 0 mean in group 1
##      0.01875212      0.08726739
```

```
with(new_data, t.test(Score ~ Intervention))
```

```
##
##  Welch Two Sample t-test
##
## data:  Score by Intervention
## t = -0.58937, df = 217.15, p-value = 0.5562
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.867655  2.087034
## sample estimates:
## mean in group 0 mean in group 1
##        76.28141        77.17172
```

Our results have a bit of a difference. I anticiapted the p would be less than 0.05.Our p value is quite higher than from our raw data. Could the intevention grades just happen by chance? Perhaps we can do a variance test as well to double check.

```
with(new_data, var.test(Std_Score, Score))
```

```
##
##  F test to compare two variances
##
## data:  Std_Score and Score
## F = 0.0059223, num df = 297, denom df = 297, p-value < 2.2e-16
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.004715403 0.007438160
## sample estimates:
## ratio of variances
##        0.005922324
```

The variance looks good and there is a small interval. At this time we can move onto to the propensity portion using MatchIt. Guiding thru MatchIt, there was toons of information that makes this the perfect package to utlize.

MatchIt implements the suggestions for improving parametric statistical models and reducing model dependence. When matching select duplicate observations from our data, hence this must be done without inducing bias and the is no dependcencey on the outome variable. “The simplest way to obtain good matches is to use one-to-one exact matching, which pairs each treated unit with one control unit for which the values of Xi are identical. However, with many covariates and finite numbers of potential matches, sufficient exact matches often cannot be found.”

For that reason the nearest neighbor in MatchIt will be more applicable. Nearest neighbor matching selects the r best control matches for each individual in the treatment group (excluding those discarded using the discard option). Matching is done using a distance measure specified by the distance option.

Take a quick glimpse at your data to make sure all necessary values have 1 or 0, as well as see if there are any missing values in our catergories that we did not see.

```
glimpse(new_data)
```

```
## Observations: 298
## Variables: 11
## $ Instructor   <chr> "Harlemon, Maxine", "Jalloh, Mohamed", "Parker, C...
## $ Student      <chr> "Turner, Kailen A", "Devers, Ciera D", "Gordon, J...
## $ Major        <fct> Biology, Computer Science, Biology, Biology, Biol...
## $ Grade        <fct> A, A, C, A, B, C, B, D, B, D, F, A, C, A, A, B, B...
## $ Score        <dbl> 90, 90, 70, 90, 80, 70, 80, 60, 80, 60, 50, 90, 7...
## $ Gender       <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
## $ Race         <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
## $ Pell         <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ Generation   <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ Intervention <fct> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0...
## $ Std_Score    <dbl> 1.0744889, 1.0744889, -0.4646439, 1.0744889, 0.30...
```

There are a few variables missing there for we will omit the NA values and call it new_dat. And load MatchIt

```
library(MatchIt)
```

```
## Warning: package 'MatchIt' was built under R version 3.6.1
```

```
new_dat <- na.omit(new_data)
```

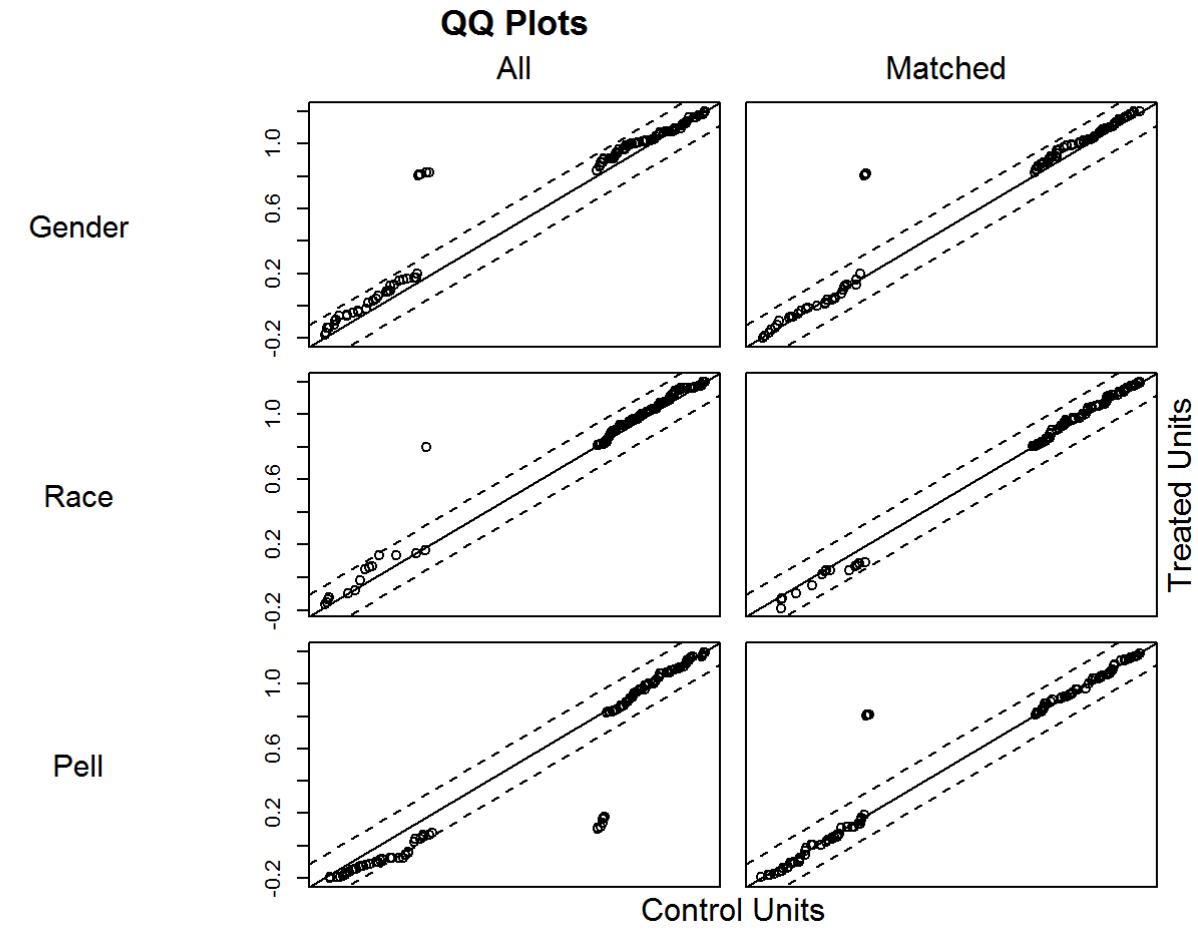
```
glimpse(new_dat)
```

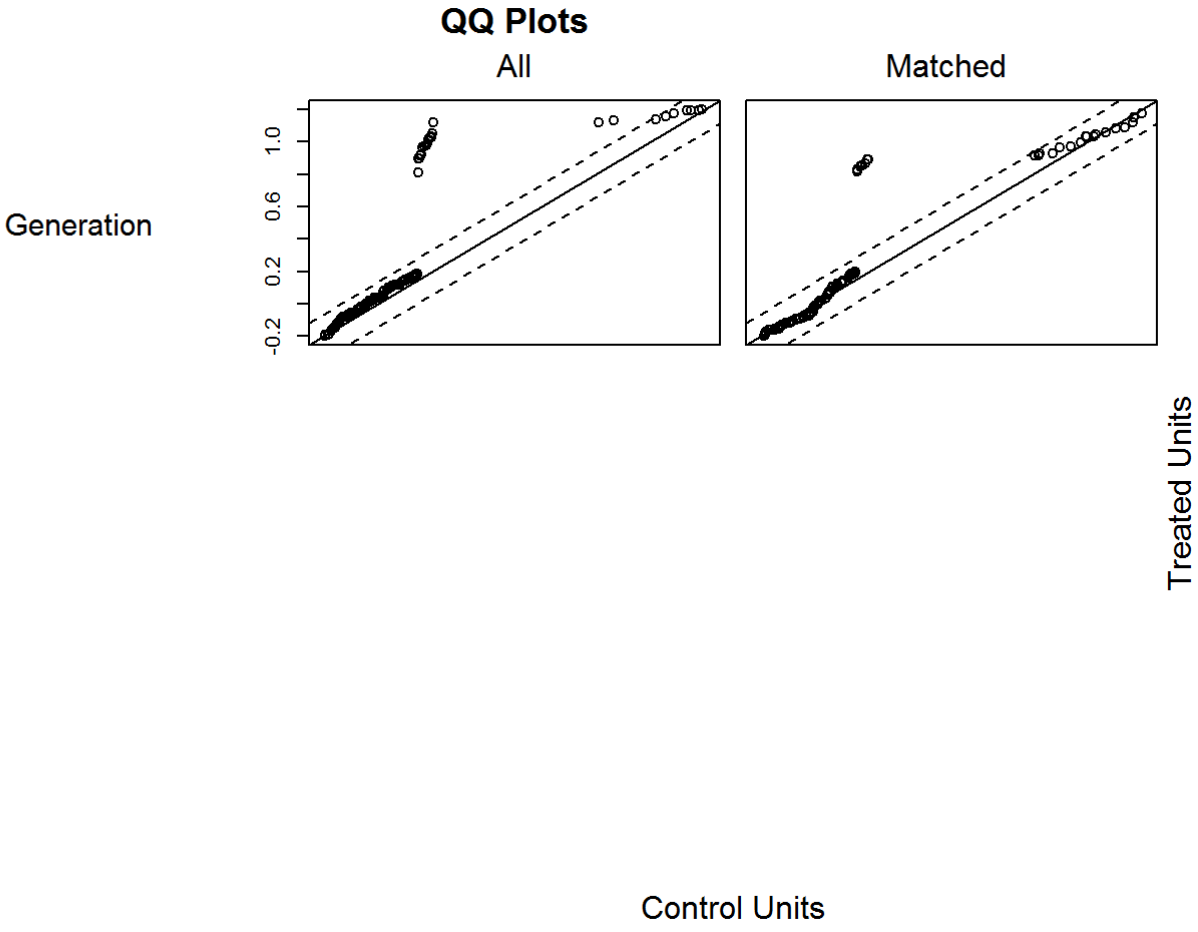
```
## Observations: 285
## Variables: 11
## $ Instructor   <chr> "Harlemon, Maxine", "Jalloh, Mohamed", "Parker, C...
## $ Student      <chr> "Turner, Kailen A", "Devers, Ciera D", "Gordon, J...
## $ Major        <fct> Biology, Computer Science, Biology, Biology, Biol...
## $ Grade        <fct> A, A, C, A, B, C, B, D, B, D, F, A, C, A, A, B, B...
## $ Score        <dbl> 90, 90, 70, 90, 80, 70, 80, 60, 80, 60, 50, 90, 7...
## $ Gender       <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
## $ Race         <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
## $ Pell         <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ Generation   <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ Intervention <fct> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0...
## $ Std_Score    <dbl> 1.0744889, 1.0744889, -0.4646439, 1.0744889, 0.30...
```

```
mod_match <- matchit(Intervention ~  Gender + Race + Pell + Generation,
                     method = 'nearest', data = new_dat)
summary(mod_match)
```

```
##
## Call:
## matchit(formula = Intervention ~ Gender + Race + Pell + Generation,
##     data = new_dat, method = "nearest")
##
## Summary of balance for all data:
##            Means Treated Means Control SD Control Mean Diff eQQ Med
## distance          0.3840        0.3279     0.0944    0.0561  0.0574
## Gender            0.7071        0.6559     0.4764    0.0512  0.0000
## Race              0.8586        0.8495     0.3586    0.0091  0.0000
## Pell              0.5758        0.6452     0.4798   -0.0694  0.0000
## Generation        0.2424        0.0914     0.2890    0.1510  0.0000
##            eQQ Mean eQQ Max
## distance     0.0569  0.1792
## Gender       0.0505  1.0000
## Race         0.0101  1.0000
## Pell         0.0707  1.0000
## Generation   0.1515  1.0000
##
##
## Summary of balance for matched data:
##            Means Treated Means Control SD Control Mean Diff eQQ Med
## distance          0.3840        0.3644     0.1140    0.0195       0
## Gender            0.7071        0.6768     0.4701    0.0303       0
## Race              0.8586        0.8586     0.3502    0.0000       0
## Pell              0.5758        0.5455     0.5005    0.0303       0
## Generation        0.2424        0.1717     0.3791    0.0707       0
##            eQQ Mean eQQ Max
## distance     0.0195  0.1792
## Gender       0.0303  1.0000
## Race         0.0000  0.0000
## Pell         0.0303  1.0000
## Generation   0.0707  1.0000
##
## Percent Balance Improvement:
##            Mean Diff. eQQ Med eQQ Mean eQQ Max
## distance      65.1958     100  65.7072       0
## Gender        40.7643       0  40.0000       0
## Race         100.0000       0 100.0000     100
## Pell          56.3380       0  57.1429       0
## Generation    53.1823       0  53.3333       0
##
## Sample sizes:
##           Control Treated
## All           186      99
## Matched        99      99
## Unmatched      87       0
## Discarded       0       0
```

```
plot(mod_match)
```


QQ Plots
All
Matched
Gender
Race
Pell
-0.2
0.2
0.6
1.0
Treated Units
Control Units


QQ Plots
All
Matched
Generation
-0.2
0.2
0.6
1.0
Treated Units
Control Units

```
mod_match1 <- match.data(mod_match)
write.csv(mod_match1, file= "mod_match")
```

There were 99 matched from the Intervention (treated group), based on the similiar observation. This is still a good number students.

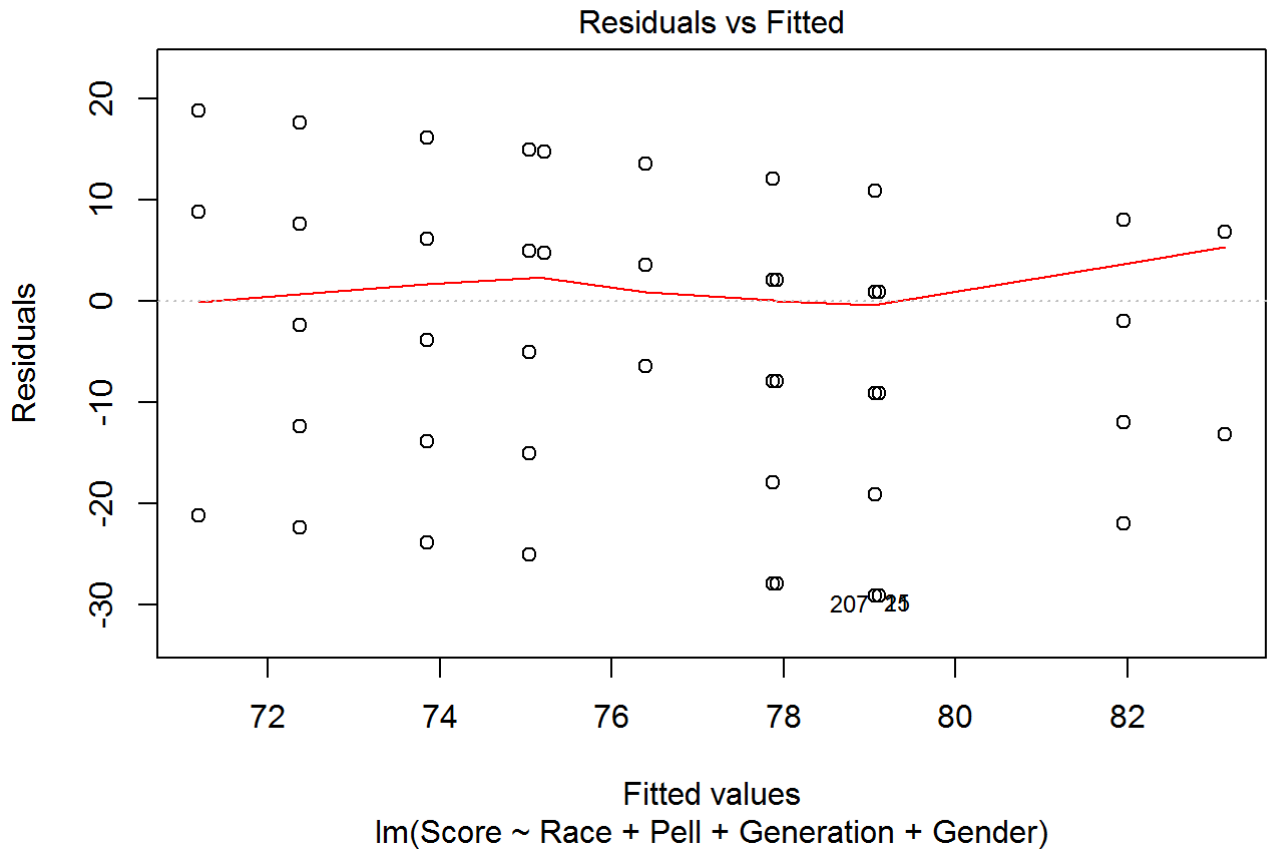
```
lm_treat1 <- lm(Score ~ Race + Pell + Generation + Gender, data = mod_match1)
summary(lm_treat1)
```

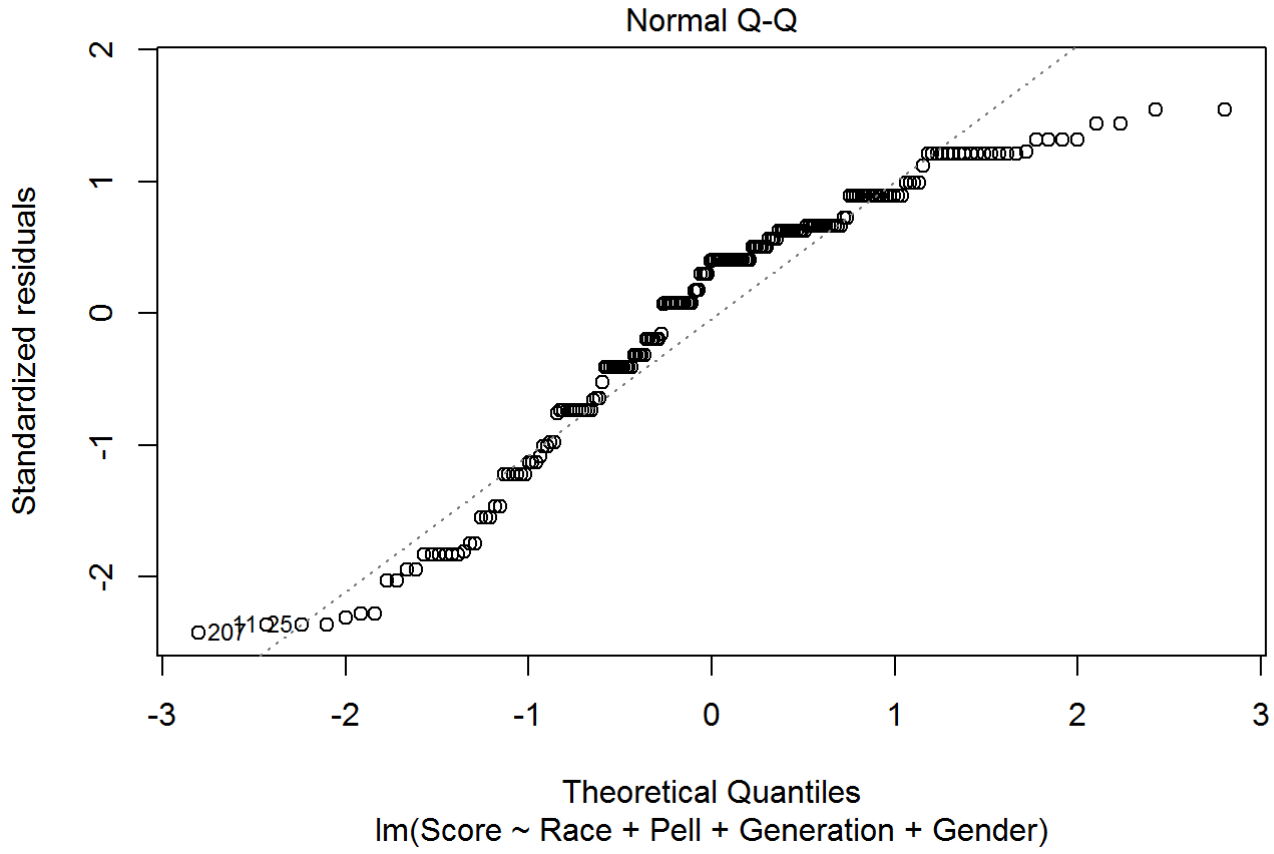
```
##
## Call:
## lm(formula = Score ~ Race + Pell + Generation + Gender, data = mod_match1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.110  -9.061   4.875   8.047  18.811
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   81.953      2.455  33.384   <2e-16 ***
## Race          -4.075      2.835  -1.437   0.1523
## Pell          -4.026      1.895  -2.124   0.0349 *
## Generation    -2.663      2.279  -1.169   0.2440
## Gender         1.183      2.050   0.577   0.5647
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.42 on 193 degrees of freedom
## Multiple R-squared:  0.06176,    Adjusted R-squared:  0.04231
## F-statistic: 3.176 on 4 and 193 DF,  p-value: 0.01482
```

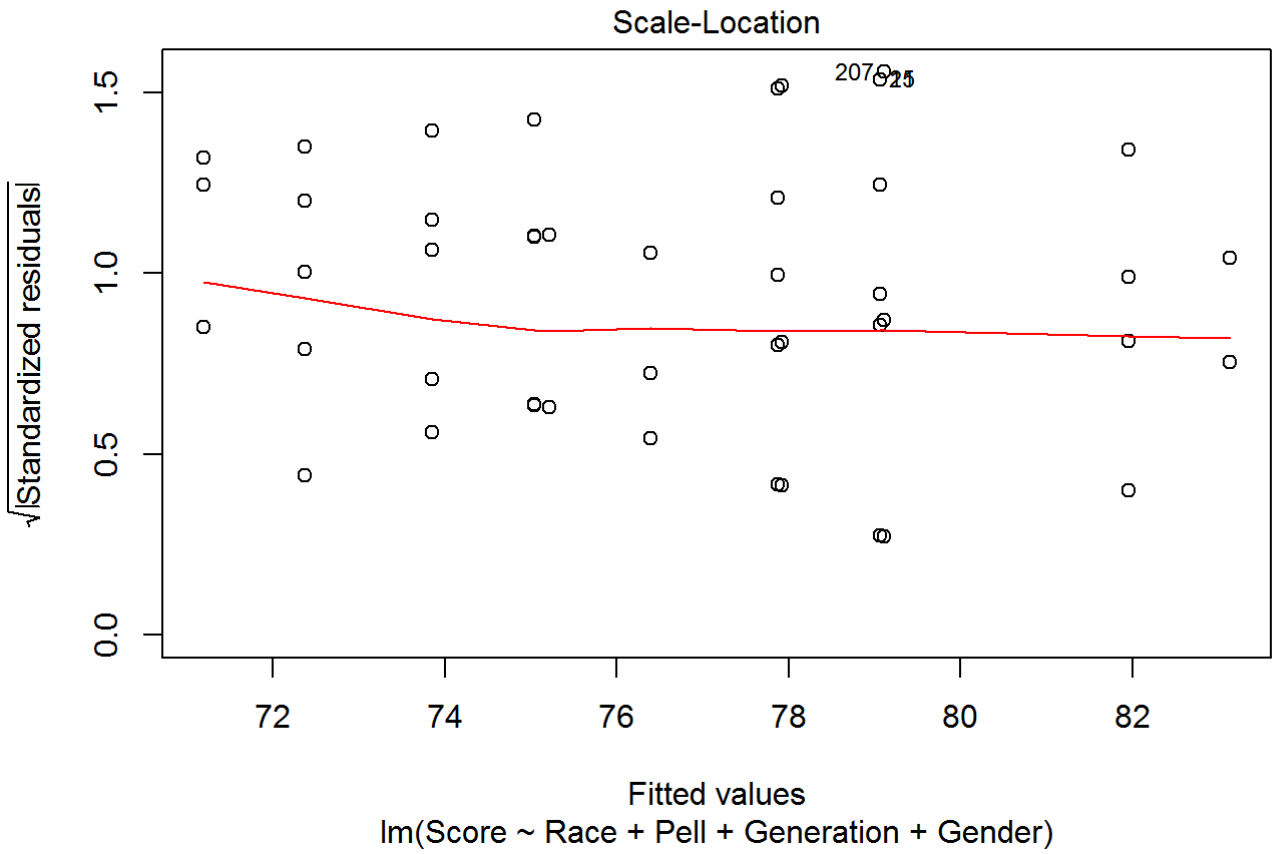
This is the machine learning model.

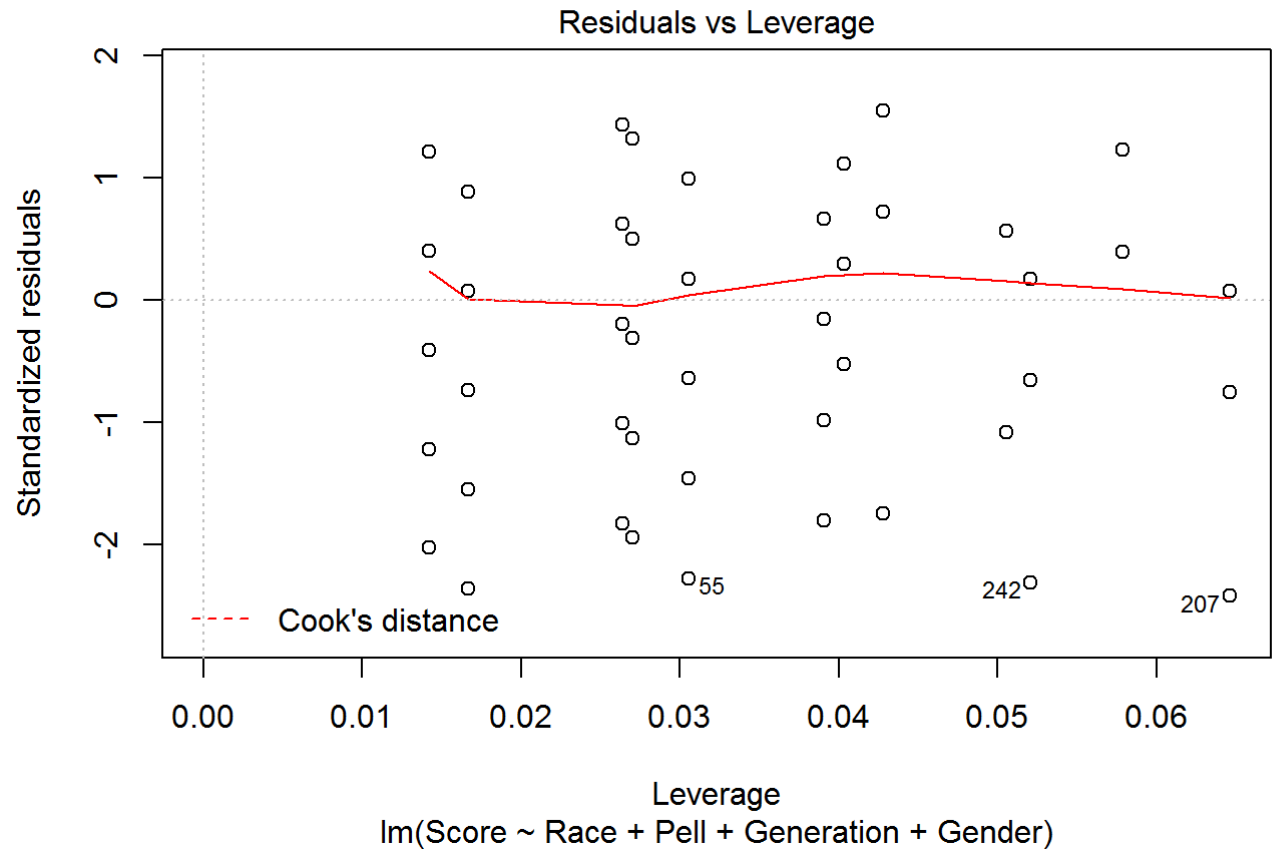
$$y = 82.550x_1 - 5.44(race) - 3.807(Pell) - 2.267(Generation) + 1.431(Gender)$$

```
plot(lm_treat1)
```


Residuals vs Fitted
Residuals
20
10
0
-10
-20
-30
72
74
76
78
80
82
207
25
Fitted values
lm(Score ~ Race + Pell + Generation + Gender)


Normal Q-Q
Standardized residuals
2
1
0
-1
-2
-3
-2
-1
0
1
2
3
207
11
25
Theoretical Quantiles
lm(Score ~ Race + Pell + Generation + Gender)



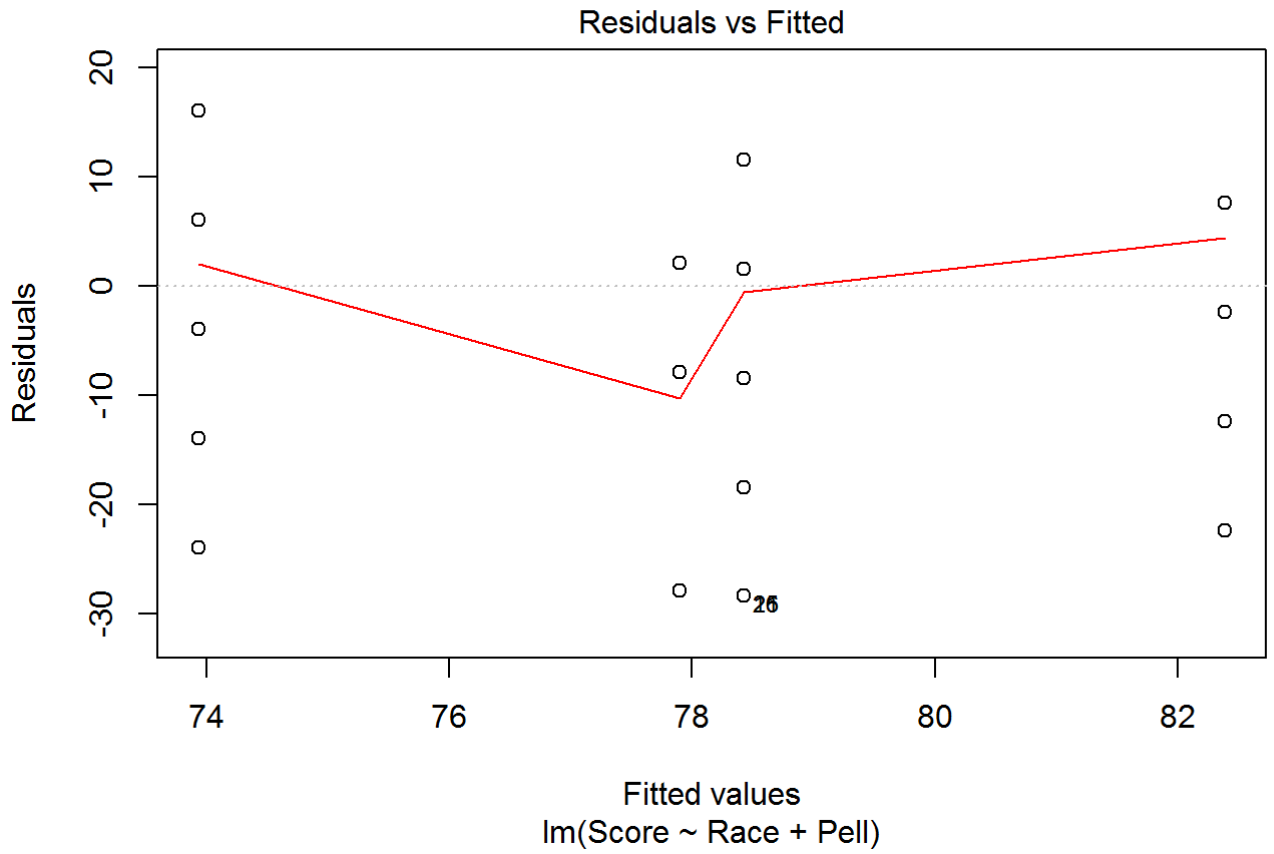


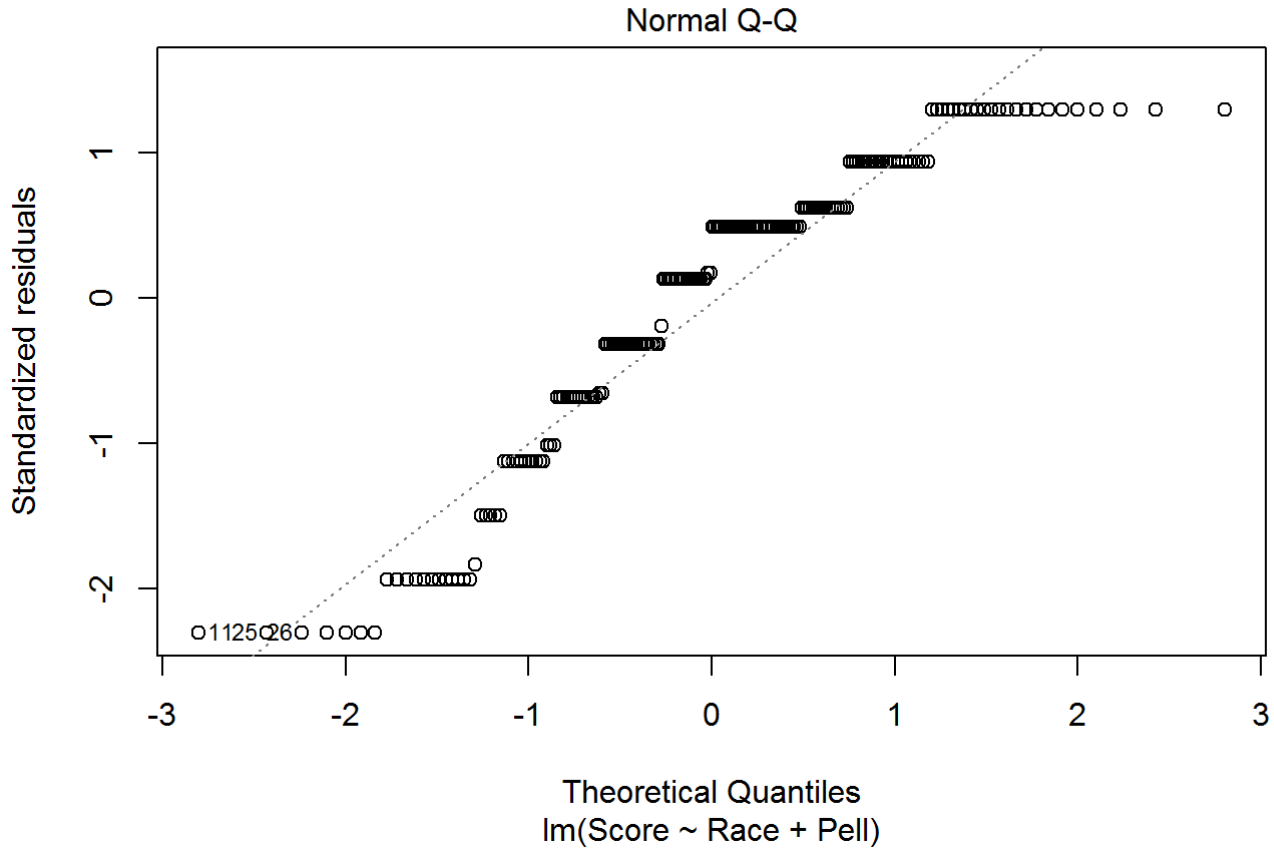
Drop the generation and genderto see if there is a differnce in the significance.

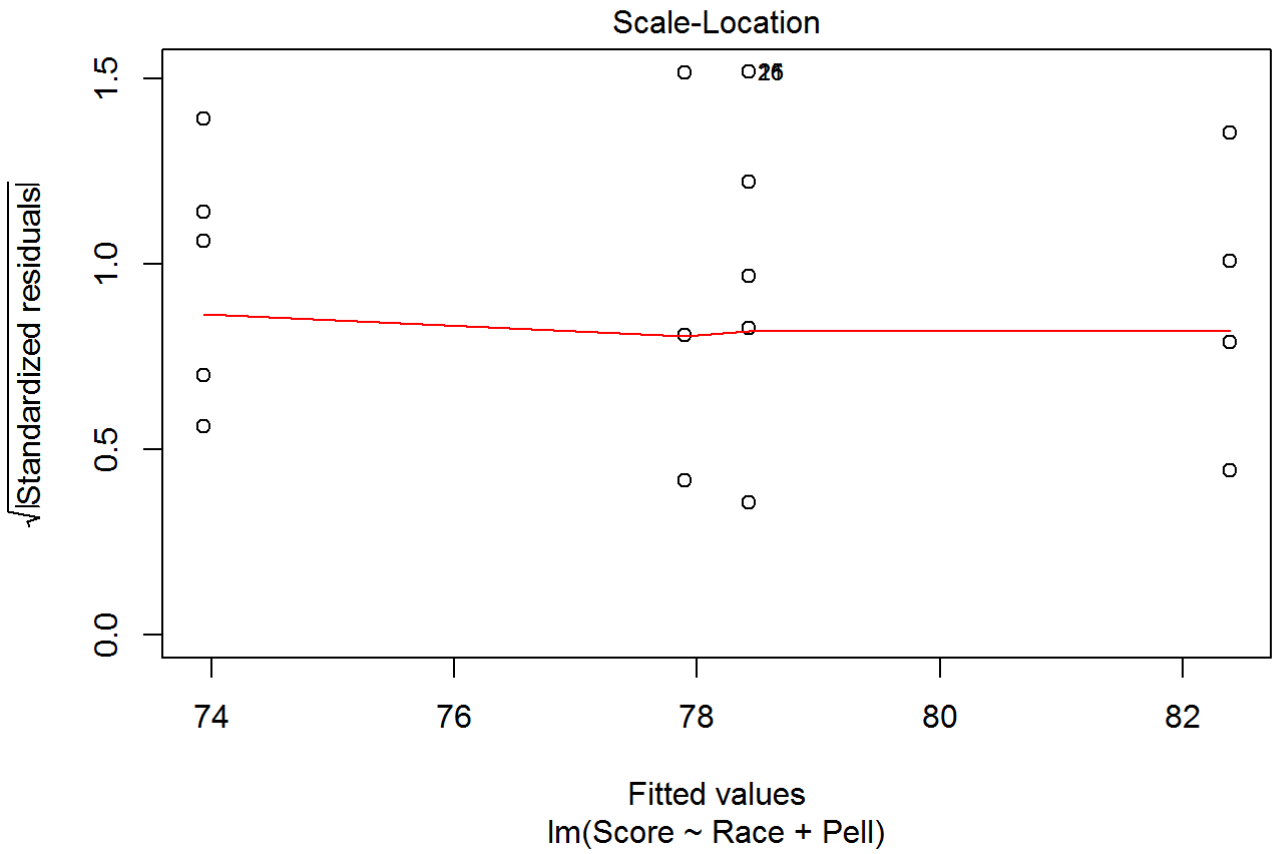
```
lm_treat2 <- lm(Score ~ Race + Pell, data = mod_match1)
summary(lm_treat2)
```

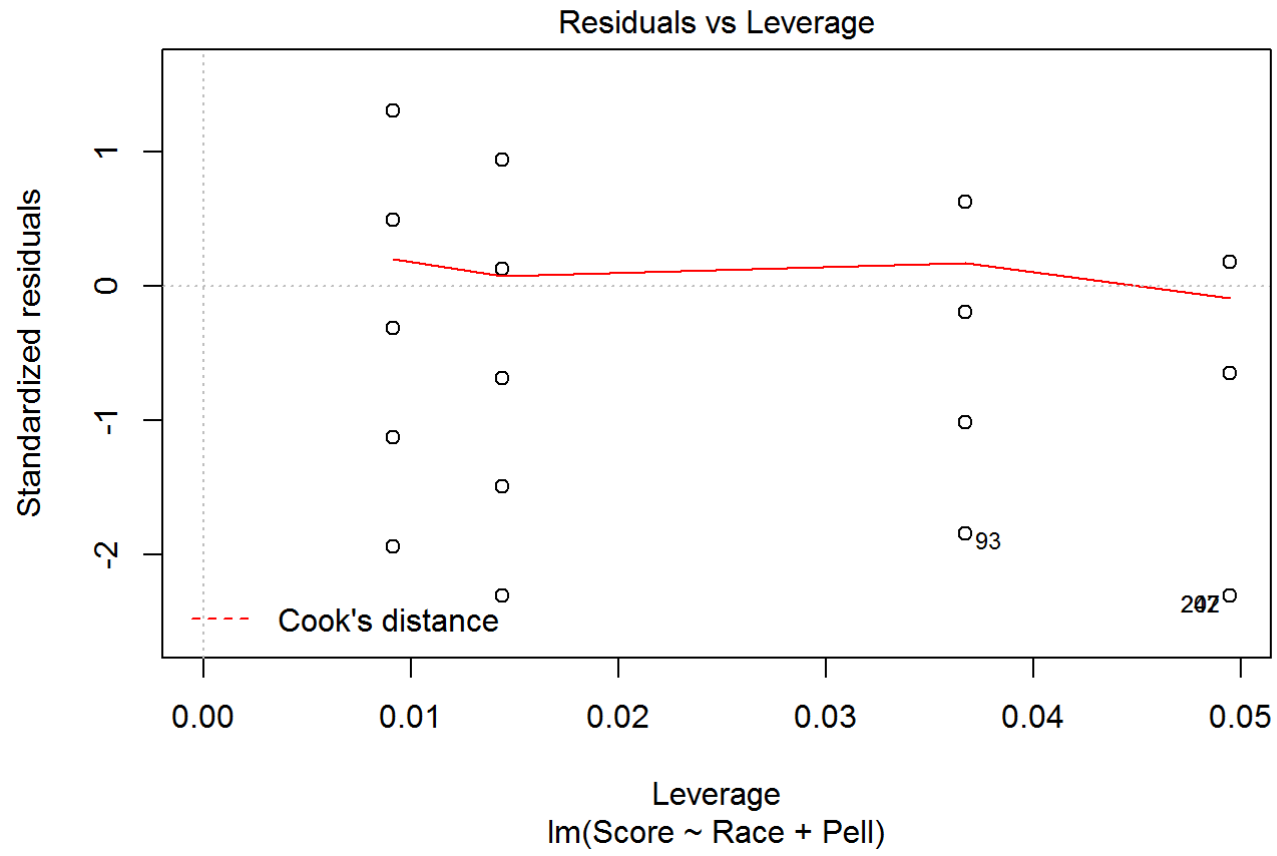
```
##
## Call:
## lm(formula = Score ~ Race + Pell, data = mod_match1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -28.422  -8.421   4.086   7.609  16.070
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   82.391      2.378  34.647   <2e-16 ***
## Race          -3.970      2.638  -1.505   0.1341
## Pell          -4.492      1.852  -2.425   0.0162 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.41 on 195 degrees of freedom
## Multiple R-squared:  0.05386,    Adjusted R-squared:  0.04416
## F-statistic:  5.55 on 2 and 195 DF,  p-value: 0.004525
```

```
plot(lm_treat2)
```


Residuals vs Fitted
Residuals
20
10
0
-10
-20
-30
74
76
78
80
82
26
Fitted values
lm(Score ~ Race + Pell)


Normal Q-Q
Standardized residuals
1
0
-1
-2
-3
-2
-1
0
1
2
3
11
25
26
Theoretical Quantiles
lm(Score ~ Race + Pell)





This is the machine learning model.

$$y = 83.044 - 5.152(race) - 4.207(Pell)$$

The propensity score matching was to predict if the control group performed better in Calculus. After the 198 observations (99 control vs 99 treated) were matched to “Pell”, “Race”, “Generation”, and “Gender” only based on their Mathematical Score, there was some “significance”, but not enough to hold any statistical value. Looking at our R Squared we did not get to near 1% as expected. The Residual standard Error is quite high as well. We would like for that to be in 4, the greater the error the higher the grade could have error. Unfortunately, this model is not a good predictor in either case. However, there is one good statistical information, for instance the p-value. Remember, if our data gives us low p-values such as p < 0.05 then, these are good values and indicate our data did not occur by chance. As we ca see our p-value in terms of “Pell” and “Race” is 2%. Meaning students are not by chance making better grades in the course. This could possibly mean that the intervention maybe be effective. I believe our data is too small for us to get the conclusion we are looking for.

Recommendations

1. Collect data on students utilizing other variable such as SAT, Study time, or some possible variable that influences their grade.
2. Look at the data Fall versus Spring Semesters, as it has been hypothesized that student typically do better in the Fall.
3. Look at other Interventions methods. Perhaps having a co-requiste calculus.

Future Work

What would happen if the data set was larger? What would happen if influencers of grades were added as variables, such as the Study time, tutoring, SAT scores? Does certain instructors make a difference? These would be all things I would collect data for and work on future work for this particular study. In my instructor role I would look at prediction of final grades based on some of the same variables including class time and strictly only online course. On my personal level I would look into Water testing and are students learning more effectively from web enhanced learning platforms versus Adaptive learning platforms?