# STAT 3675Q Homework 6

Due date: **Friday, April 15, 2022, 11:59pm**

Hari Patchigolla

## STAT 3675Q Final Project - Analyzing Correlations between Various Colleges in America

### Introduction

Every year thousands upon thousands of high school seniors from across the country apply to numerous universities with hopes of attending a dream school, or a school with a good program in their field of interest. I too have been in that position before, hence the motivation for this project. Student want to have adequent access to do data to based their decisions off of.

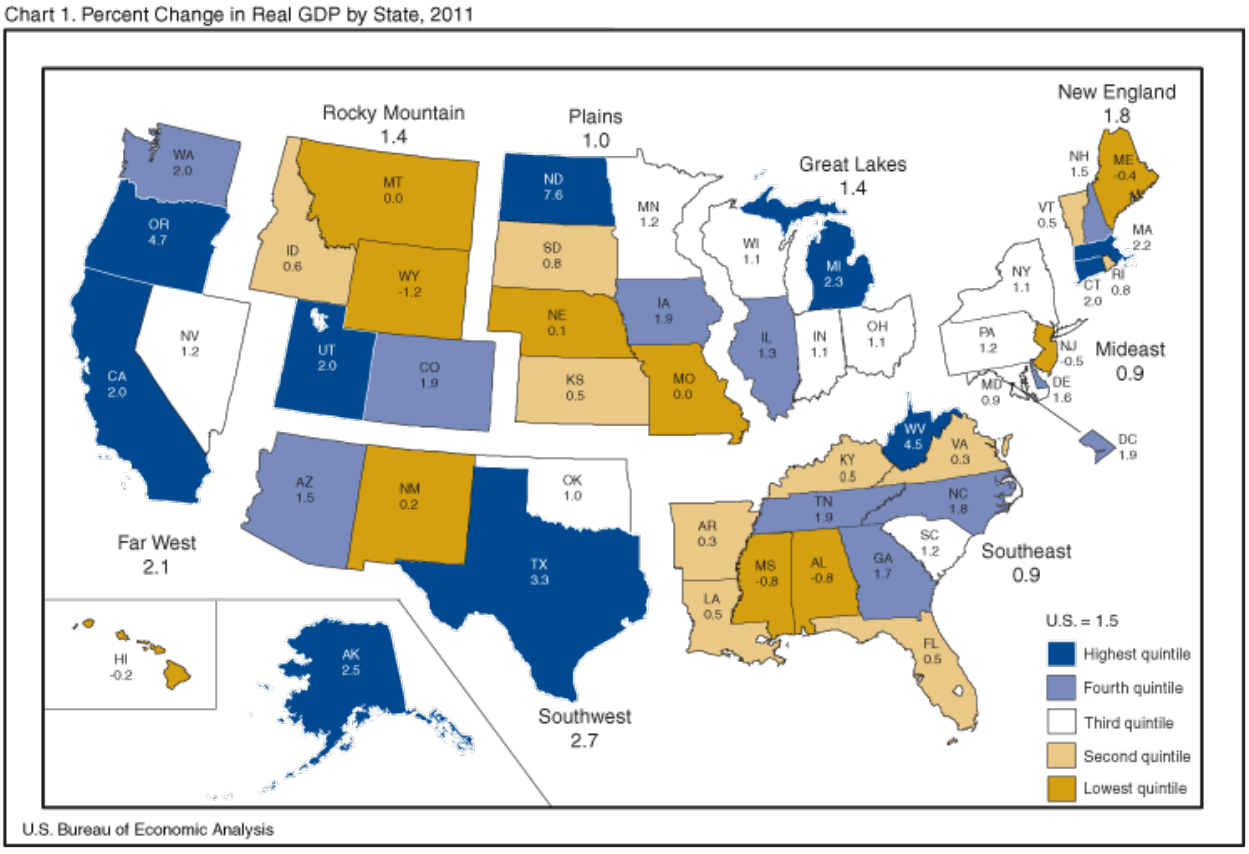
For this analysis I am Web Scraping data off of www.money.com. Money is an independent, advertiser-supported website and their editors "research hundreds of sources and contact hundreds of the most respected experts in each industry to get the most relevant information to help others make the right purchasing decision." The data consists of various/useful metrics of the the best colleges in America ranked by value (as determined by the website). In this first section I create the dataframe that consists of all the data I want to collect from the website.

Features of the dataset -

***est_full_Price_2020_2021*** : The estimated full price for the entire 2020/2021 academic year ***est_price_for_students_who_receive_aid*** : The estimated full price for the entire 2020/2021 academic year for students that receive some form of aid
***average_price_for_low_income_students*** : Avg. Price for Low Income Students
***acceptance_rate*** : Acceptance rate
***Median_SAT_Score*** : Median SAT Score
***Median_ACT_Score*** : Median ACT Score
***enrollment*** : Number of Students Enrolled
***percent_of_need_met*** : Percent of Student that have the need met
***percent_of_students_who_get_merit_grants*** : Percent of students who get merit aids
***average_merit_grant*** : Avg. Merit Grant
***graduation_rate*** : Graduation Rate

***average_time_to_a_degree*** : Avg. Time to a Degree
***average_student_debt*** : Avg. Student Debt
***average_salary_within_5_years*** :Avg. Salary within 5 Years
***percent_earning_more_than_28000*** : Percent earning more that $28,000
***percent_of_students_who_get_any_grants***: Percent of Students who get any grants
***percent_of_students_with_need_who_get_grants*** : Percent of Students with Need who get Grants ***SAT_ACT_required_for_Fall_2021*** : SAT/ACT Required
***regular_application*** : Reg Application deadline
***college_names*** : College Name
***College_Location_Town*** : College Location (Town)
***College_Location_States***: College Location (State)

This dataset contains a large number of variables and rows (each of which represent a different college). Initially my research goal/question was to elucidate statistically significant differences in within the colleges of different regions in USA :




In other words, my initial goal was to understand if there was a significant difference within certain features of the dataframe for various regions of USA. For example: 1. Is there a significant difference between the enrollment in top colleges (determined by money.com) between universities in New England region and Far West Region? 2. Is it harder to get into a college (based off off acceptance rate) in the Southeast region versus the Rocky Mountain Region? 3. Do colleges in the Southwest Region have a higher graduation rate than those

in the New England region?

However, as I was going about this project, I decided to only study the New England region and any statistically significant differences in the universities within the states of New England.

To study these difference I will be using ANOVA tests and t-tests. Please look at the respective sections to see the research questions.

Lastly, I will be implementing a logistic regression model that can predict if a university in USA requires the SAT/ACT (more details, such as my motivation for doing this, are in the specified section)

Please continue reading to find answers to my research questions and analysis.

## Creating Dataset

The `rvest` library is used for web scraping and the `dplyr` library is used for its pipe lining functionality.

```
Attaching package: 'dplyr'
```

```
The following objects are masked from 'package:stats':

    filter, lag
```

```
The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

Here I am using the `read_html()` function in `rvest` to read an html document specified by a url.

```
link <- paste("https://money.com/best-colleges/")
pg <-  read_html(link)
```

The `html_nodes()` function allows me to get the html tags by specifying a specific CSS selector. And `html_text()` gets the text in an html node.

The below code shows how the html page `pg` (from the previous code block) is being used to extract certain information like College Name, Graduation Rate, etc. of all the colleges in this link. The CSS selectors were chosen using the following Chrome Extension.

```
college_names <- pg %>% html_nodes("._1RI9D22X") %>% html_text() #This gets all the col
Est_price_2020_21_without_aid <- pg %>% html_nodes("td:nth-child(2)") %>% html_text()
Est_price_with_average_grant <- pg %>% html_nodes("td:nth-child(3)") %>% html_text()
percent_of_students_who_get_any_grants <- pg %>% html_nodes("td:nth-child(4)") %>% html_
graduation_rate <- pg %>% html_nodes("td:nth-child(5)") %>% html_text()
average_student_debt <- pg %>% html_nodes("td:nth-child(6)") %>% html_text()
early_career_earnings <- pg %>% html_nodes("td:nth-child(7)") %>% html_text()
college_location <- pg %>% html_nodes("._1OtZ7j1R") %>% html_text()
```

Here I create a dataframe and get all the information form a specific college by extracting the href attribute and going to that link. Money has a page dedicated to information on a specific college here is a link to MIT's page on Money. Notice how there are many different features like Enrollment, Acceptance Rate, Est. price for students who receive aid, etc. Also notice how every college from the intial link has a similar page with the same information (here is the link for Harvard, for Univeristy of Florida). All of this information is being extracted and then put into a dataframe.

```
df <- data.frame(matrix(ncol = 19, nrow = 0))
baselink <- "https://money.com"

for (i in 1:748){
  sel <- paste("tr:nth-child(", i, ") ._1RI9D22X", sep="")
  restlink <- pg %>% html_nodes(sel) %>% html_attr("href")
  gotolink <- paste(baselink, restlink, sep="")
  datapg <- read_html(gotolink)

  vals <- datapg %>% html_nodes(".small-3") %>% html_text()
  vals2 <- datapg %>% html_nodes(".small-2") %>% html_text()

  temp <- append(vals, vals2)
  df <- rbind(df, temp)
}
```

Write the csv file to save it and rename the dataframe to `college_df`.

```
write.csv(df,"college_data.csv", row.names = FALSE)
college_df <- df
rm(df)
```

Change the names of the `college_df` to match what they really represent.

```
coln <- c("est_full_Price_2020_2021",
          "est_price_for_students_who_receive_aid",
```

```
            "average_price_for_low_income_students",
            "acceptance_rate",
            "median_SAT_ACT_score",
            "SAT_ACT_required_for_Fall_2021",
            "enrollment",
            "percent_of_need_met",
            "percent_of_students_who_get_merit_grants",
            "average_merit_grant",
            "graduation_rate",
            "average_time_to_a_degree",
            "average_student_debt",
            "average_salary_within_5_years",
            "percent_earning_more_than_28000",
            "early_decision_application",
            "regular_application",
            "percent_of_students_who_get_any_grants",
            "percent_of_students_with_need_who_get_grants"
)

names(college_df) <- coln
```

Add on information from the intial link that was not in any specific college site into `college_df`.

```
college_df["Est_price_2020_21_without_aid"] <- Est_price_2020_21_without_aid
college_df["college_names"] <- college_names
college_df["Est_price_with_average_grant"] <- Est_price_with_average_grant
college_df["early_career_earnings"] <- early_career_earnings
college_df["college_location"] <- college_location
```

Check the structure of the dataframe. Notice how all the columns of character vectors. This is because the `html_text()` function returns a string.

We now have to preprocess the data to change the type and replace missing values. Also notice how values that should be numeric have a dollar sign and commas or even percent symbols. This all needs to be changed.

```
str(college_df)
```

```
'data.frame':   739 obs. of  24 variables:
 $ est_full_Price_2020_2021                    : chr  "$71,800" "$73,400" "$65,800" "$31
 $ est_price_for_students_who_receive_aid      : chr  "$19,800" "$18,000" "$16,900" "$17
 $ average_price_for_low_income_students       : chr  "$7,900" "$1,300" "$2,700" "$4,100
 $ acceptance_rate                             : chr  "7%" "4%" "5%" "23%" ...
```

```
 $ median_SAT_ACT_score                          : chr  "1540/35" "1500/34" "1510/34" "142
 $ SAT_ACT_required_for_Fall_2021                : chr  "no" "no" "no" "no" ...
 $ enrollment                                    : chr  "4,550" "7,080" "5,300" "30,080" .
 $ percent_of_need_met                           : chr  "100%" "100%" "100%" "93%" ...
 $ percent_of_students_who_get_merit_grants      : chr  "N/A" "N/A" "N/A" "11%" ...
 $ average_merit_grant                           : chr  "N/A" "$13,250" "N/A" "$5,570" ...
 $ graduation_rate                               : chr  "94%" "94%" "96%" "92%" ...
 $ average_time_to_a_degree                      : chr  "4.1 years" "4.3 years" "4.1 years
 $ average_student_debt                          : chr  "$12,500" "$11,340" "$9,850" "$16,
 $ average_salary_within_5_years                 : chr  "$81,400" "$72,700" "$70,200" "$63
 $ percent_earning_more_than_28000               : chr  "93%" "91%" "89%" "85%" ...
 $ early_decision_application                    : chr  "Nov 1" "Nov 1" "N/A" "Nov 1" ...
 $ regular_application                           : chr  "Jan 1" "Jan 3" "Jan 1" "Feb 1" ..
 $ percent_of_students_who_get_any_grants        : chr  "65%" "59%" "59%" "51%" ...
 $ percent_of_students_with_need_who_get_grants: chr  "98%" "96%" "100%" "81%" ...
 $ Est_price_2020_21_without_aid                 : chr  " $71,800" " $73,400" " $65,800" "
 $ college_names                                 : chr  "Massachusetts Institute of Techno
 $ Est_price_with_average_grant                  : chr  " $19,800" " $18,000" " $16,900" "
 $ early_career_earnings                         : chr  " $81,400" " $72,700" " $70,200" "
 $ college_location                              : chr  "Cambridge, MA" "Stanford, CA" "Pr
```

# Cleaning Dataset

## Changing Types of the Columns

the `tidyr` library is used later to split certain columns in the `college_df`

```
library(tidyr)
```

Missing values in `college_df` are currently represented as "N/A" or "NA" (as characters and not actual NA values) so this function below takes in a dataframe and an column number and replaces all "N/A" and "NA" with an actual NA and returns a new vector.

```
replace_with_null <- function(data, ind){
  temp <- trimws(data[,ind])
  return(replace(temp, which(temp == "NA" | temp == "N/A"), NA))
}
```

The `cleandfcol` function below is a bit complicated so I will not go into the tiny details of it, but it essentially takes in a dataframe and a column number and and returns the proper format of the column. For example, in the columns that contains values like ["$4,839" ,"$5,674" ...] it will remove the dollar sign and comma and return them as a numeric

vector. Or for a column of percents [“7%”, “23%”, . . . ] it will remove the “%” and return a numeric vector.

```
cleandfcol <- function(data, ind){
  vec <- replace_with_null(data, ind)
  nonavec <- vec[!is.na(vec)]
  if (sum(substr(nonavec, start = 3, stop = 3) == "%") > 0){
    vec <- gsub("%", "", vec)
    return(as.numeric(vec))
  }
  else if (sum(substr(nonavec, start = 1, stop = 1) == "$") > 0){
    vec <- sub('.', '', gsub(",", "", vec))
    return(as.numeric(vec))
  }
  else if (sum(grepl(",", data[,ind], fixed = TRUE)) > 0){
    vec <- gsub(",", "", vec)
    return(as.numeric(vec))
  }
  else {
    vec <- gsub("years", "", vec)
    return(as.numeric(vec))
  }
}
```

Here we go through all the columns that need to be formatted (like the enrollment, acceptance_rate, etc.) and properly convert them into a numeric vector and replace `college_df` with the returned vector.

```
for (i in c(1, 2, 3, 4, 7, 8, 9, 10, 11, 12, 13, 14, 15, 18, 19, 20, 22, 23)){
  college_df[,i] <- cleandfcol(college_df, i)
}
```

In the below two code blocks I use the `separate()` function in the `tidyr` library to split the `median_SAT_ACT_score` and `college_location` columns.

```
college_df <- separate(college_df, 5,
                       c("Median_SAT_Score", "Median_ACT_Score"), "/",
                       remove=TRUE, convert = TRUE)
```

```
college_df <- separate(college_df, 25,
                       c("College_Location_Town", "College_Location_States"), ",",
                       remove=TRUE, convert = TRUE)
college_df$College_Location_States <- trimws(college_df$College_Location_States)
```

The `replace_with_null` function is used replace "NA" to NA in the remaining columns (such as the categorical or character columns)

```
for (i in c(7, 17, 18, 22, 25)){
  college_df[,i] <- replace_with_null(college_df, i)
}
```

Here I remove duplicated rows and columns in `college_df` Source

```
college_df <- college_df[!duplicated(college_df), ]
college_df <- college_df[!duplicated(as.list(college_df))]
```

Notice how all the columns are now of the proper type and format.

```
str(college_df)
```

```
'data.frame':   739 obs. of  23 variables:
 $ est_full_Price_2020_2021                    : num  71800 73400 65800 31000 76000 3420
 $ est_price_for_students_who_receive_aid      : num  19800 18000 16900 17600 20700 1860
 $ average_price_for_low_income_students       : num  7900 1300 2700 4100 NA 8400 2400 4
 $ acceptance_rate                             : num  7 4 5 23 9 26 6 10 30 41 ...
 $ Median_SAT_Score                            : int  1540 1500 1510 1420 1510 1420 1520
 $ Median_ACT_Score                            : int  35 34 34 32 34 32 34 34 30 28 ...
 $ SAT_ACT_required_for_Fall_2021              : chr  "no" "no" "no" "no" ...
 $ enrollment                                  : num  4550 7080 5300 30080 6600 ...
 $ percent_of_need_met                         : num  100 100 100 93 100 100 100 100 83
 $ percent_of_students_who_get_merit_grants    : num  NA NA NA 11 3 3 NA 10 2 4 ...
 $ average_merit_grant                         : num  NA 13250 NA 5570 NA ...
 $ graduation_rate                             : num  94 94 96 92 96 94 97 94 86 86 ...
 $ average_time_to_a_degree                    : num  4.1 4.3 4.1 4.2 4.1 4.1 4.1 4.1 4.
 $ average_student_debt                        : num  12500 11340 9850 16610 11500 ...
 $ average_salary_within_5_years               : num  81400 72700 70200 63700 67800 6140
 $ percent_earning_more_than_28000             : num  93 91 89 85 92 88 88 89 84 84 ...
 $ early_decision_application                  : chr  "Nov 1" "Nov 1" NA "Nov 1" ...
 $ regular_application                         : chr  "Jan 1" "Jan 3" "Jan 1" "Feb 1" ..
 $ percent_of_students_who_get_any_grants      : num  65 59 59 51 56 43 53 67 58 66 ...
 $ percent_of_students_with_need_who_get_grants: num  98 96 100 81 95 87 100 99 95 97 ..
 $ college_names                               : chr  "Massachusetts Institute of Techno
 $ College_Location_Town                       : chr  "Cambridge" "Stanford" "Princeton"
 $ College_Location_States                     : chr  "MA" "CA" "NJ" "MI" ...
```

##Exploring and Handling Missing Values

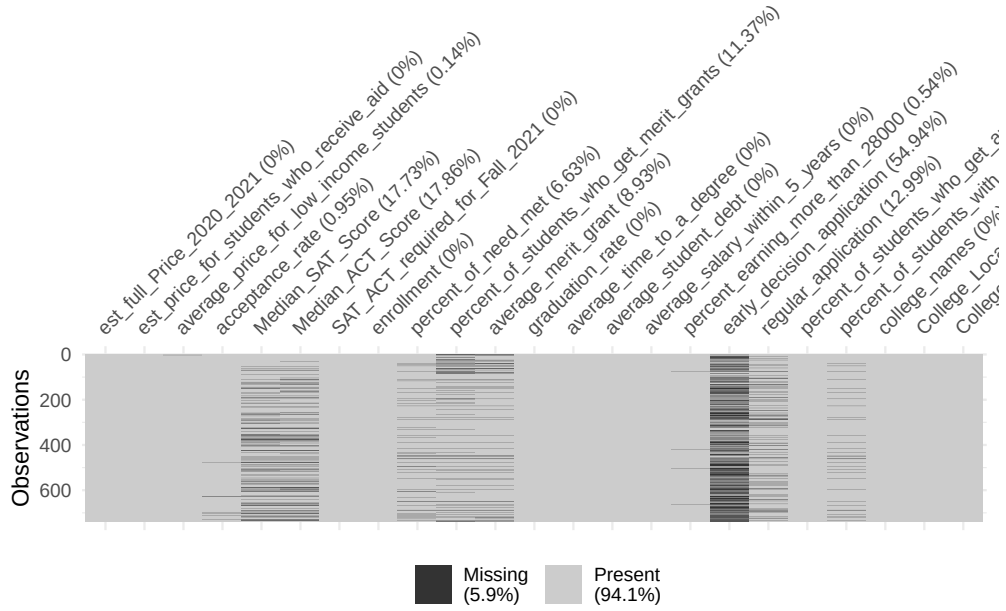
```
Warning: package 'visdat' was built under R version 4.1.3
```

```
Warning: package 'naniar' was built under R version 4.1.3
```

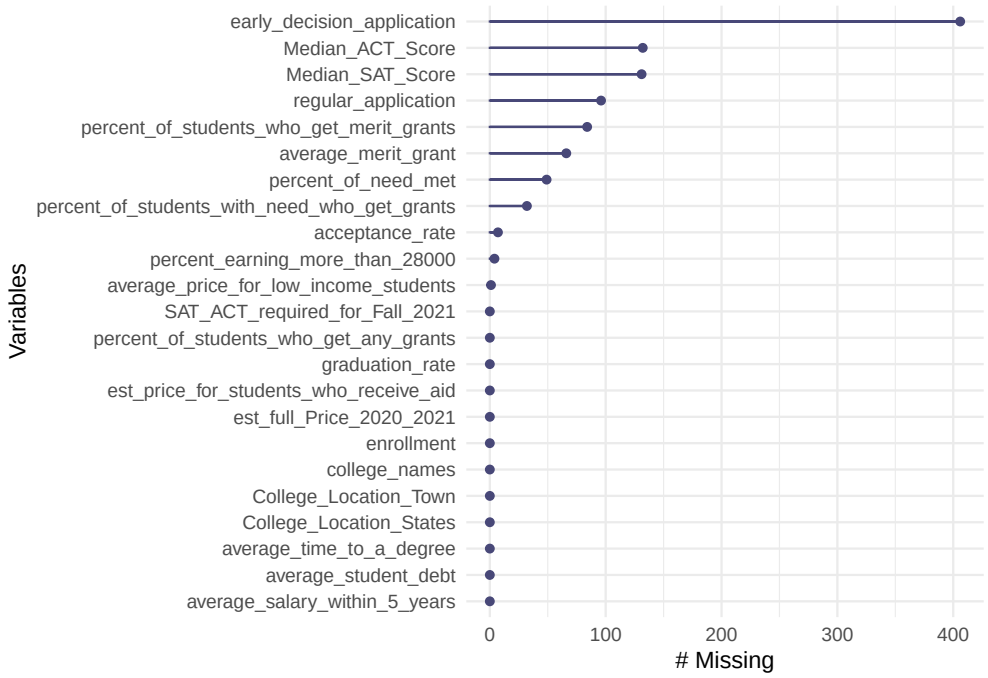
```
Warning: package 'missForest' was built under R version 4.1.3
```

Currently the `college_df` dataset has 1008 missing values with almost 5.9% of its data missing.

```
vis_miss(college_df)
```




```
gg_miss_var(college_df)
```




```
sum(is.na(college_df))
```

```
[1] 1008
```

Because the `early_decision_application` has more than 50% of its data missing (more than 400 missing values) it is dropped.

```
college_df = subset(college_df, select = -c(early_decision_application))
```

Data is split into numeric and character datasets (for imputation via the `missForest` library). It uses a random forest trained on the observed values of a data matrix to predict the missing values.

```
numintdata <- college_df[,sapply(college_df,is.numeric) | sapply(college_df,is.integer)]
categoricaldata <- college_df[,sapply(college_df,is.character)]
numintdata.imp <- missForest(numintdata)
```

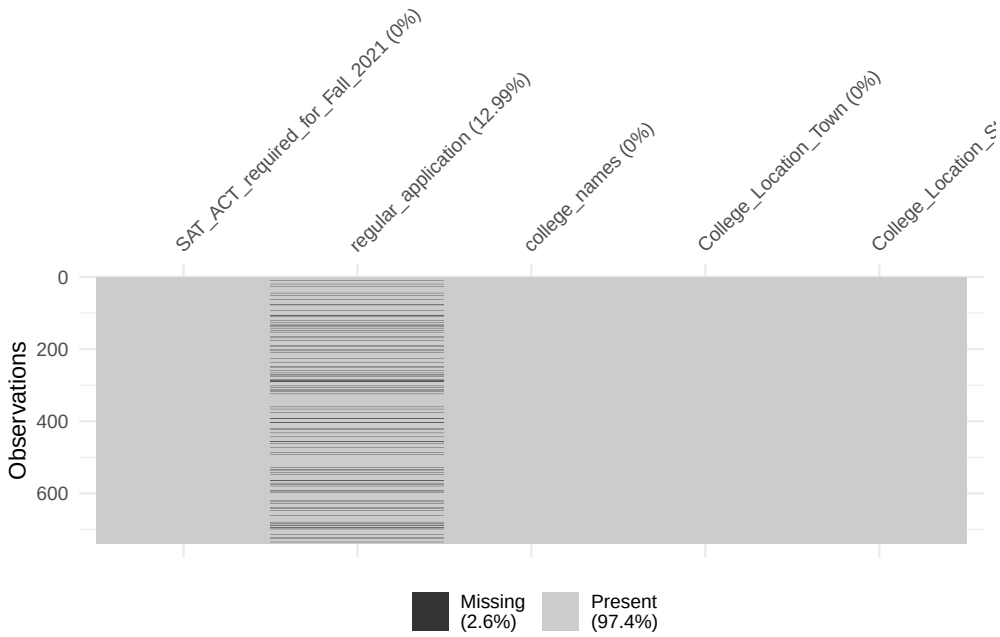
This is the normalized root mean squared error (NRMSE). It is a small value indicated good imputation results.

```
numintdata.imp$OOBerror
```

```
     NRMSE
0.07648893
```

In the categorical dataset only the `regular_application` column has missing values. To impute it I will replace it with the mode of the column.

```
vis_miss(categoricaldata)
```




R does not have a built in function to calculate mode, so I used a function from this link

```
#Source:
calc_mode <- function(x){

  # List the distinct / unique values
  distinct_values <- unique(x)

  # Count the occurrence of each distinct value
  distinct_tabulate <- tabulate(match(x, distinct_values))

  # Return the value with the highest occurrence
  distinct_values[which.max(distinct_tabulate)]
}

mfreq <- calc_mode(categoricaldata$regular_application)
mfreq #The most frequent regular application deadline is rolling
```

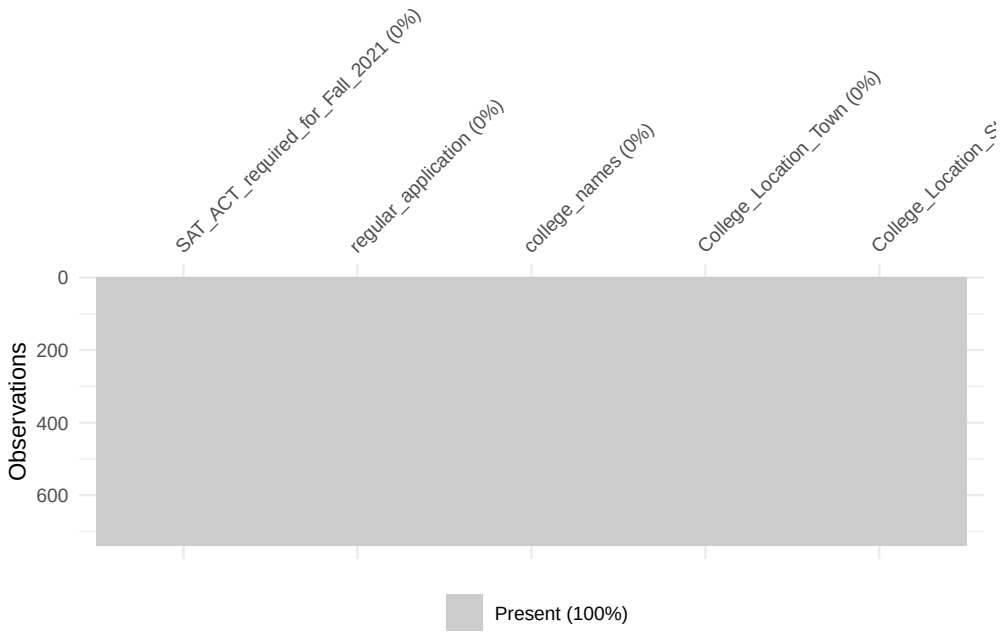
```
[1] "rolling"
```

The following code cell replaces all NA values in `regular_application` with “rolling”

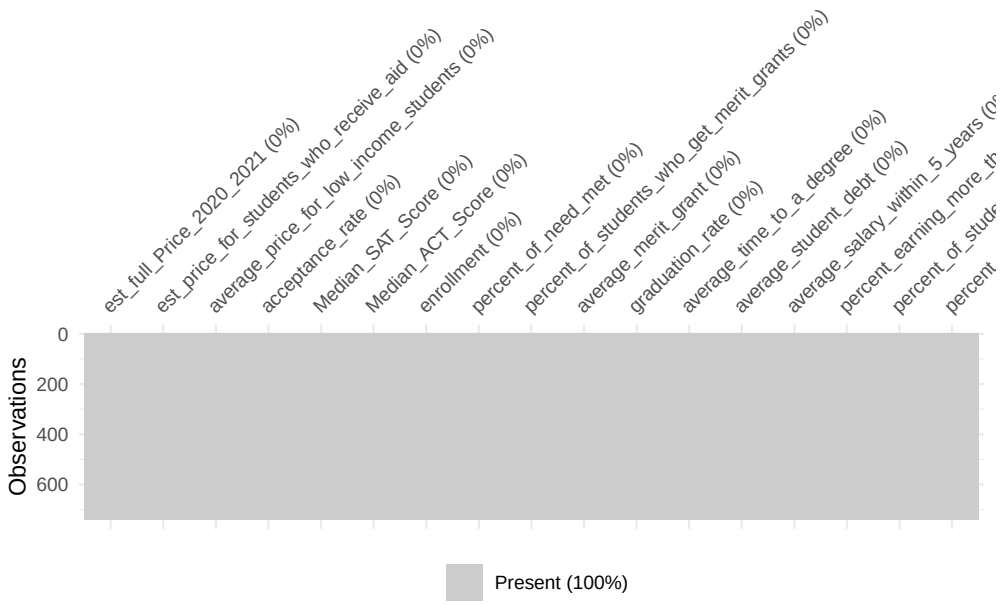
```
categoricaldata$regular_application = if_else(is.na(categoricaldata$regular_application)
                                    mfreq,
                                    categoricaldata$regular_application)
```

Notice how both the numerical and categorical datasets no longer have any missing values

```
vis_miss(categoricaldata)
```



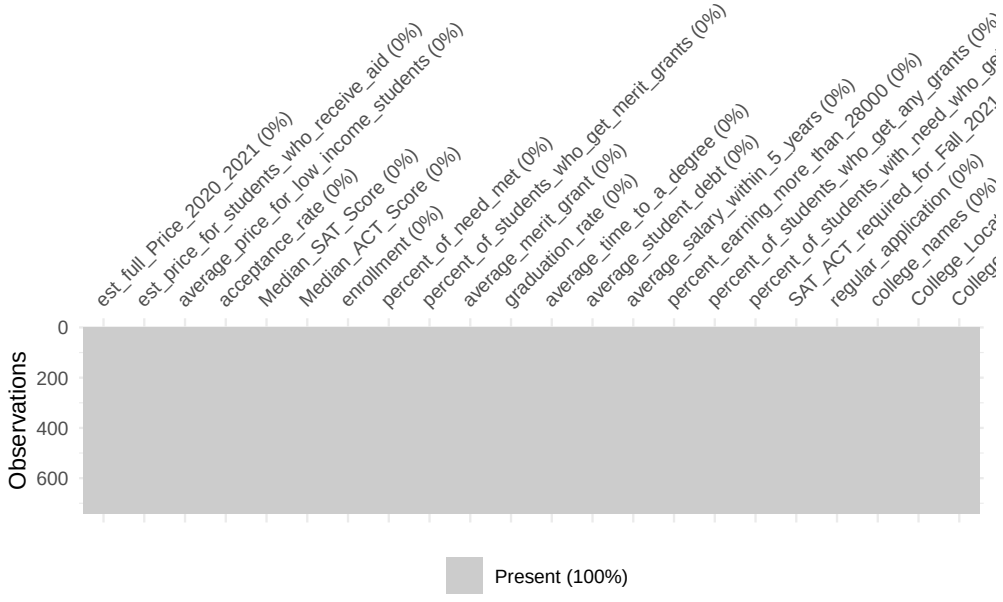

```
vis_miss(numintdata.imp$ximp)
```




```
college_df <- cbind(numintdata.imp$ximp, categoricaldata)
```

Now `college_df` no longer contains missing values.

```
vis_miss(college_df)
```



# Data Visulaization

For the sake of this project as there are many variables to see/understand I will only be focusing on a subset of the dataframe that contians information of the universities within the New England region.

```
Warning: package 'vioplot' was built under R version 4.1.3
```

```
Loading required package: sm
```

```
Package 'sm', version 2.2-5.7: type help(sm) for summary information
```

```
Loading required package: zoo
```

```
Attaching package: 'zoo'
```

```
The following objects are masked from 'package:base':

    as.Date, as.Date.numeric
```

```
Loading required package: mvtnorm
```

```
Loading required package: survival
```

```
Loading required package: TH.data
```

```
Loading required package: MASS
```

```
Attaching package: 'MASS'
```

```
The following object is masked from 'package:sm':

    muscle
```

```
The following object is masked from 'package:dplyr':

    select
```

```
Attaching package: 'TH.data'
```

```
The following object is masked from 'package:MASS':

    geyser

The following object is masked from 'package:sm':

    geyser

Warning: package 'car' was built under R version 4.1.3

Loading required package: carData

Warning: package 'carData' was built under R version 4.1.3

Attaching package: 'car'

The following object is masked from 'package:psych':

    logit

The following object is masked from 'package:dplyr':

    recode
```

Though there is a lot of information below some notable ones are: 1. The trimmed means and means do not seem to change much for each variable suggesting that there are little outliers. 2. The small kurtosis values also suggests lack of outliers. 3. Many of the columns have both a positive or negative skew.

```
psych::describe(college_df)
```

| | vars | n | mean | sd | median |
|---|---|---|---|---|---|
| est_full_Price_2020_2021 | 1 | 739 | 46722.19 | 17748.02 | 48400.0 |
| est_price_for_students_who_receive_aid | 2 | 739 | 24030.31 | 8932.32 | 23100.0 |
| average_price_for_low_income_students | 3 | 739 | 16575.41 | 7745.86 | 15700.0 |
| acceptance_rate | 4 | 739 | 63.26 | 21.30 | 68.0 |
| Median_SAT_Score | 5 | 739 | 1187.19 | 127.93 | 1170.0 |
| Median_ACT_Score | 6 | 739 | 25.34 | 3.78 | 25.0 |
| enrollment | 7 | 739 | 7552.84 | 9441.30 | 3320.0 |
| percent_of_need_met | 8 | 739 | 72.77 | 15.90 | 73.0 |
| percent_of_students_who_get_merit_grants | 9 | 739 | 16.01 | 10.94 | 14.0 |
| average_merit_grant | 10 | 739 | 11843.91 | 7165.01 | 11890.0 |

| | | | | |
|---|---|---|---|---|
| graduation_rate | 11 739 | 68.96 | 12.95 | 68.0 |
| average_time_to_a_degree | 12 739 | 4.27 | 0.21 | 4.2 |
| average_student_debt | 13 739 | 22909.55 | 4221.82 | 24000.0 |
| average_salary_within_5_years | 14 739 | 49957.78 | 7045.81 | 48600.0 |
| percent_earning_more_than_28000 | 15 739 | 77.95 | 7.70 | 78.0 |
| percent_of_students_who_get_any_grants | 16 739 | 75.58 | 17.10 | 77.0 |
| percent_of_students_with_need_who_get_grants | 17 739 | 88.85 | 12.86 | 95.0 |
| SAT_ACT_required_for_Fall_2021* | 18 739 | 1.16 | 0.37 | 1.0 |
| regular_application* | 19 739 | 32.04 | 16.14 | 46.0 |
| college_names* | 20 739 | 369.03 | 213.33 | 369.0 |
| College_Location_Town* | 21 739 | 265.70 | 155.85 | 276.0 |
| College_Location_States* | 22 739 | 25.56 | 13.66 | 27.0 |

| | trimmed | mad | min | max |
|---|---|---|---|---|
| est_full_Price_2020_2021 | 46328.67 | 25500.72 | 13500 | 80400.0 |
| est_price_for_students_who_receive_aid | 23599.49 | 9043.86 | 1200 | 52800.0 |
| average_price_for_low_income_students | 16132.55 | 7413.00 | 100 | 45500.0 |
| acceptance_rate | 65.48 | 17.79 | 4 | 99.0 |
| Median_SAT_Score | 1177.77 | 118.61 | 920 | 1560.0 |
| Median_ACT_Score | 25.12 | 2.97 | 18 | 36.0 |
| enrollment | 5520.98 | 3024.50 | 660 | 80170.0 |
| percent_of_need_met | 73.04 | 14.83 | 3 | 100.0 |
| percent_of_students_who_get_merit_grants | 15.10 | 11.86 | 1 | 94.0 |
| average_merit_grant | 11452.21 | 9103.16 | 120 | 40050.0 |
| graduation_rate | 68.83 | 13.34 | 40 | 98.0 |
| average_time_to_a_degree | 4.24 | 0.15 | 4 | 5.2 |
| average_student_debt | 23408.52 | 4077.15 | 5600 | 40800.0 |
| average_salary_within_5_years | 49236.26 | 5930.40 | 33600 | 81800.0 |
| percent_earning_more_than_28000 | 78.47 | 7.41 | 42 | 94.0 |
| percent_of_students_who_get_any_grants | 76.38 | 22.24 | 30 | 100.0 |
| percent_of_students_with_need_who_get_grants | 91.00 | 7.41 | 30 | 100.0 |
| SAT_ACT_required_for_Fall_2021* | 1.07 | 0.00 | 1 | 2.0 |
| regular_application* | 33.59 | 0.00 | 1 | 47.0 |
| college_names* | 369.00 | 274.28 | 1 | 737.0 |
| College_Location_Town* | 265.52 | 203.12 | 1 | 533.0 |
| College_Location_States* | 25.74 | 16.31 | 1 | 50.0 |

| | range | skew | kurtosis | se |
|---|---|---|---|---|
| est_full_Price_2020_2021 | 66900.0 | 0.07 | -1.36 | 652.87 |
| est_price_for_students_who_receive_aid | 51600.0 | 0.40 | -0.18 | 328.58 |
| average_price_for_low_income_students | 45400.0 | 0.67 | 0.78 | 284.94 |
| acceptance_rate | 95.0 | -0.89 | 0.20 | 0.78 |
| Median_SAT_Score | 640.0 | 0.64 | -0.08 | 4.71 |
| Median_ACT_Score | 18.0 | 0.49 | -0.32 | 0.14 |
| enrollment | 79510.0 | 2.37 | 7.68 | 347.30 |
| percent_of_need_met | 97.0 | -0.38 | 0.64 | 0.58 |
| percent_of_students_who_get_merit_grants | 93.0 | 1.07 | 3.02 | 0.40 |

```
average_merit_grant                          39930.0  0.40    -0.55 263.57
graduation_rate                                 58.0  0.08    -0.61   0.48
average_time_to_a_degree                         1.2  1.46     2.36   0.01
average_student_debt                         35200.0 -0.83     0.90 155.30
average_salary_within_5_years                48200.0  1.10     1.84 259.18
percent_earning_more_than_28000                 52.0 -0.78     1.24   0.28
percent_of_students_who_get_any_grants          70.0 -0.28    -1.12   0.63
percent_of_students_with_need_who_get_grants    70.0 -1.40     1.67   0.47
SAT_ACT_required_for_Fall_2021*                  1.0  1.87     1.49   0.01
regular_application*                            46.0 -0.47    -1.53   0.59
college_names*                                 736.0  0.00    -1.21   7.85
College_Location_Town*                         532.0 -0.04    -1.26   5.73
College_Location_States*                        49.0 -0.16    -1.20   0.50
```

Here we split the `college_df` dataframe to into separate dataframes for all six states in the New England Region (Connecticut, Maine, Massachusetts, New Hampshire, Rhode Island, and Vermont). I have also created a `new_england` dataframe that consists of all the data in all six New England states.

```
CT <- college_df[which(college_df$College_Location_States == "CT"),]
#summary(CT)
ME <- college_df[which(college_df$College_Location_States == "ME"),]
#summary(ME)
MA <- college_df[which(college_df$College_Location_States == "MA"),]
#summary(MA)
NH <- college_df[which(college_df$College_Location_States == "NH"),]
#summary(NH)
RI <- college_df[which(college_df$College_Location_States == "RI"),]
#summary(RI)
VT <- college_df[which(college_df$College_Location_States == "VT"),]
#summary(VT)
new_england <- college_df[which(college_df$College_Location_States %in% c("CT",
                                                                          "MA",
                                                                          "ME",
                                                                          "NH",
                                                                          "RI",
                                                                          "VT")),]
summary(new_england)
```

```
 est_full_Price_2020_2021 est_price_for_students_who_receive_aid
 Min.   :23400            Min.   : 9700
 1st Qu.:33200            1st Qu.:20100
 Median :59900            Median :28000
 Mean   :54717            Mean   :28266
```

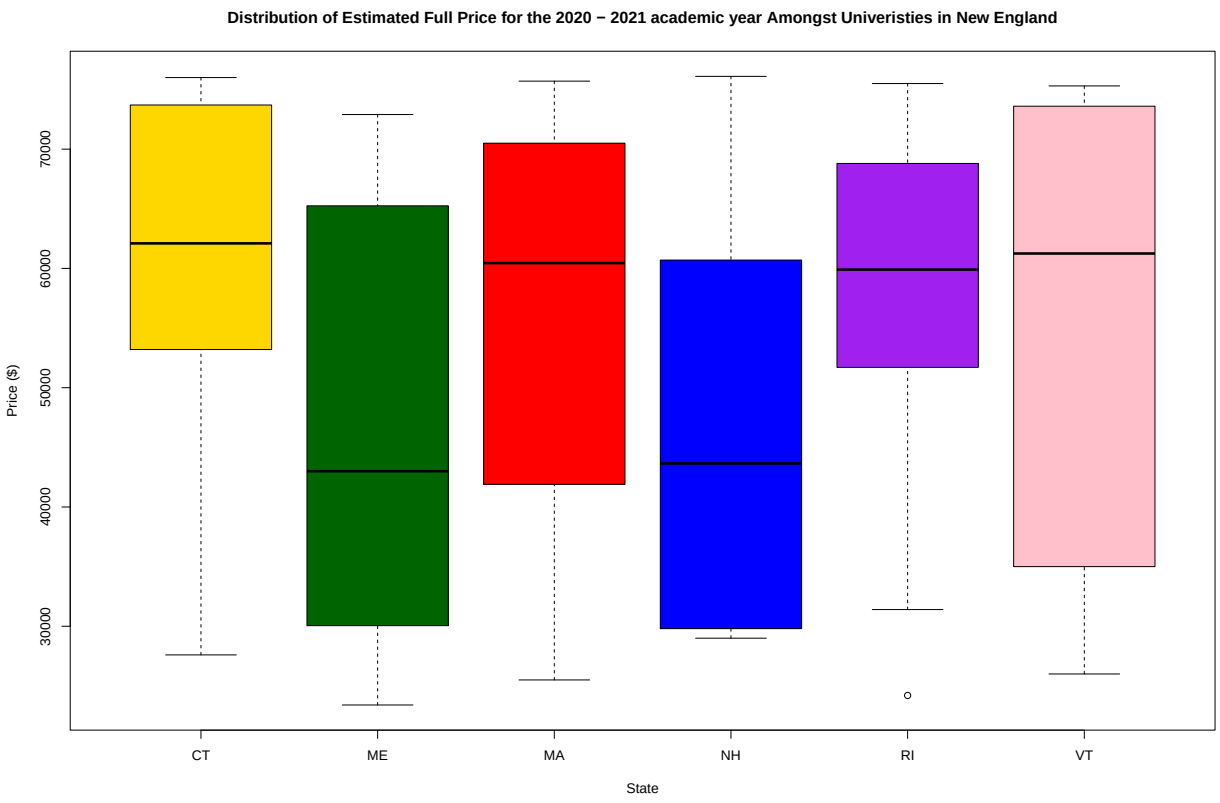
```
3rd Qu.:70700            3rd Qu.:34800
Max.   :76100            Max.   :48100
average_price_for_low_income_students acceptance_rate Median_SAT_Score
Min.   : 1100                         Min.   : 5.00   Min.   : 930
1st Qu.:12800                         1st Qu.:38.00   1st Qu.:1102
Median :17800                         Median :69.00   Median :1190
Mean   :18931                         Mean   :58.84   Mean   :1224
3rd Qu.:24800                         3rd Qu.:80.00   3rd Qu.:1360
Max.   :45500                         Max.   :94.00   Max.   :1540
Median_ACT_Score   enrollment    percent_of_need_met
Min.   :18.00    Min.   :  720   Min.   : 21.00
1st Qu.:23.00    1st Qu.: 1830   1st Qu.: 65.00
Median :26.00    Median : 3030   Median : 75.00
Mean   :26.45    Mean   : 5264   Mean   : 76.46
3rd Qu.:30.22    3rd Qu.: 5600   3rd Qu.: 92.00
Max.   :35.00    Max.   :80170   Max.   :100.00
percent_of_students_who_get_merit_grants average_merit_grant graduation_rate
Min.   : 1.00                            Min.   : 1000       Min.   :46.00
1st Qu.: 6.04                            1st Qu.: 6120       1st Qu.:63.00
Median :14.00                            Median :13343       Median :73.00
Mean   :15.04                            Mean   :12497       Mean   :72.99
3rd Qu.:23.00                            3rd Qu.:16640       3rd Qu.:87.00
Max.   :52.00                            Max.   :38620       Max.   :98.00
average_time_to_a_degree average_student_debt average_salary_within_5_years
Min.   :4.000            Min.   :10600        Min.   :39100
1st Qu.:4.100            1st Qu.:23000        1st Qu.:46800
Median :4.100            Median :25440        Median :51100
Mean   :4.191            Mean   :23497        Mean   :52116
3rd Qu.:4.300            3rd Qu.:26980        3rd Qu.:56000
Max.   :5.000            Max.   :27500        Max.   :81400
percent_earning_more_than_28000 percent_of_students_who_get_any_grants
Min.   :52.00                   Min.   :39.00
1st Qu.:76.00                   1st Qu.:55.00
Median :81.00                   Median :69.00
Mean   :80.47                   Mean   :70.74
3rd Qu.:87.00                   3rd Qu.:88.00
Max.   :94.00                   Max.   :99.00
percent_of_students_with_need_who_get_grants SAT_ACT_required_for_Fall_2021
Min.   : 36.00                               Length:93
1st Qu.: 85.00                               Class :character
Median : 95.00                               Mode  :character
Mean   : 90.23
3rd Qu.: 99.00
Max.   :100.00
regular_application college_names      College_Location_Town
```

```
 Length:93           Length:93          Length:93
 Class :character    Class :character   Class :character
 Mode  :character    Mode  :character   Mode  :character


 College_Location_States
 Length:93
 Class :character
 Mode  :character
```

The following side-by-side boxplot shows that top universites in CT, MA, RI, and VT have a similar median price whereas ME and MH have a lower median full price. Furthermore, universities in VT show the greatest spread. Lastly, there is an outlier the RI, while none of the other states have any outliers.

```
boxplot(CT$est_full_Price_2020_2021,
        ME$est_full_Price_2020_2021,
        MA$est_full_Price_2020_2021,
        NH$est_full_Price_2020_2021,
        RI$est_full_Price_2020_2021,
        VT$est_full_Price_2020_2021,
        col=c("gold", "darkgreen", "red", "blue","purple", "pink"),
        main="Distribution of Estimated Full Price for the 2020 - 2021 academic year Amo
        names = c("CT", "ME", "MA", "NH", "RI", "VT"),
        xlab="State",
        ylab="Price ($)")
```

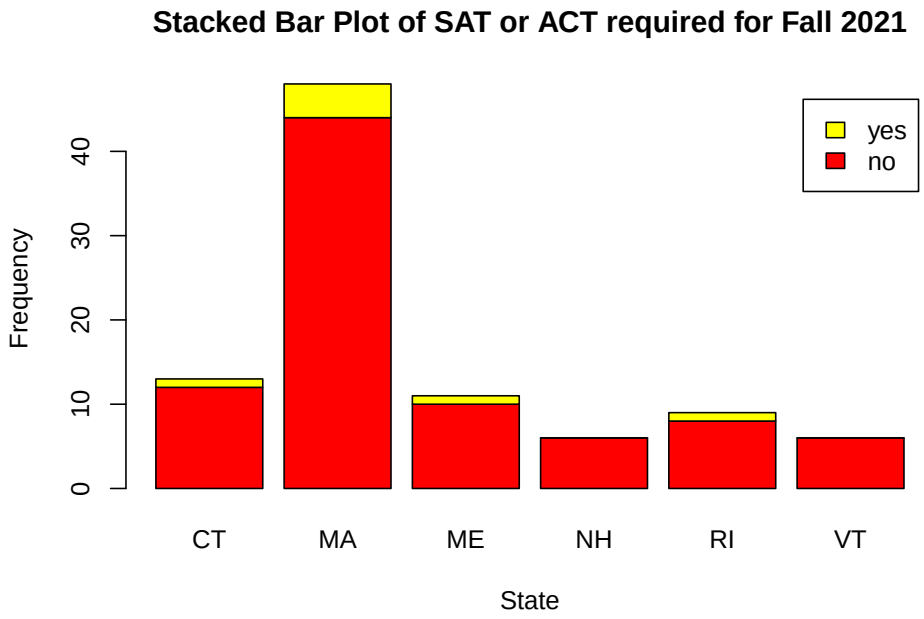



There does not seem to be too much of a difference between the universities of the New England states on requiring the SAT or ACT. In fact, in NH and VT none of the top universities require the SAT or ACT.

```
(tab <- table(new_england$SAT_ACT_required_for_Fall_2021,
              new_england$College_Location_States))
```

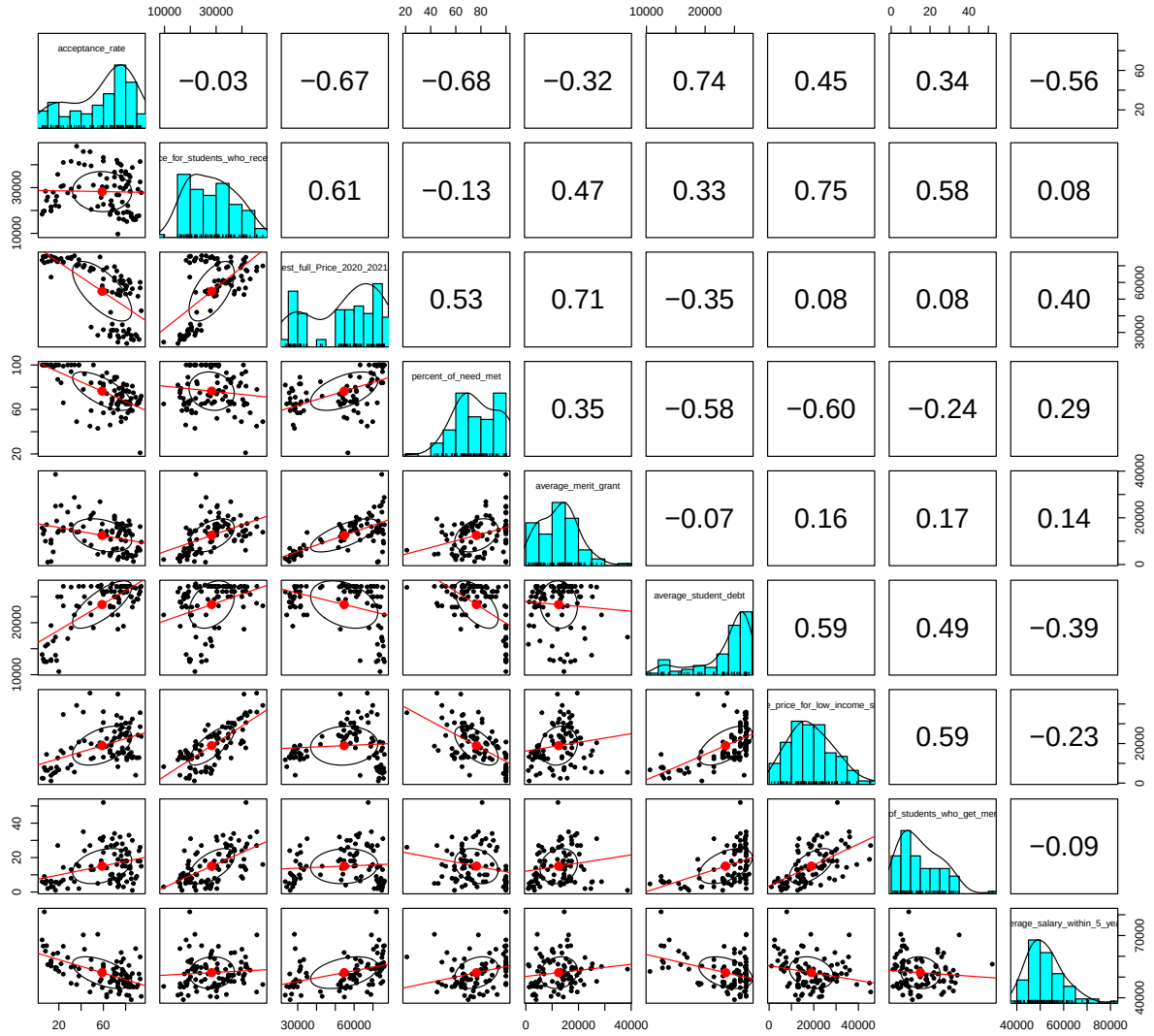
```
     CT MA ME NH RI VT
  no 12 44 10  6  8  6
  yes 1  4  1  0  1  0
```

```
barplot(tab,
        main="Stacked Bar Plot of SAT or ACT required for Fall 2021",
        xlab="State",
        ylab="Frequency",
        col=c("red", "yellow"),
        legend=rownames(tab))
```




The pair plot below shows a lot of information. More specifically in shows that there is a significant positive correlation between `acceptance_rate` and `average_student_debt` (0.74), `percent_of_students_who_get_merit_grants` and `average_price_for_low_income_students` (0.75). Furthermore the distribution of `acceptance_rate` and `est_full_Price_2020_2021` seems to be bi-modal.

```
pairs.panels(new_england[c("acceptance_rate",
                     "est_price_for_students_who_receive_aid",
                     "est_full_Price_2020_2021",
                     "percent_of_need_met",
                     "average_merit_grant",
                     "average_student_debt",
                     "average_price_for_low_income_students",
                     "percent_of_students_who_get_merit_grants",
                     "average_salary_within_5_years")],
             lm = TRUE)
```

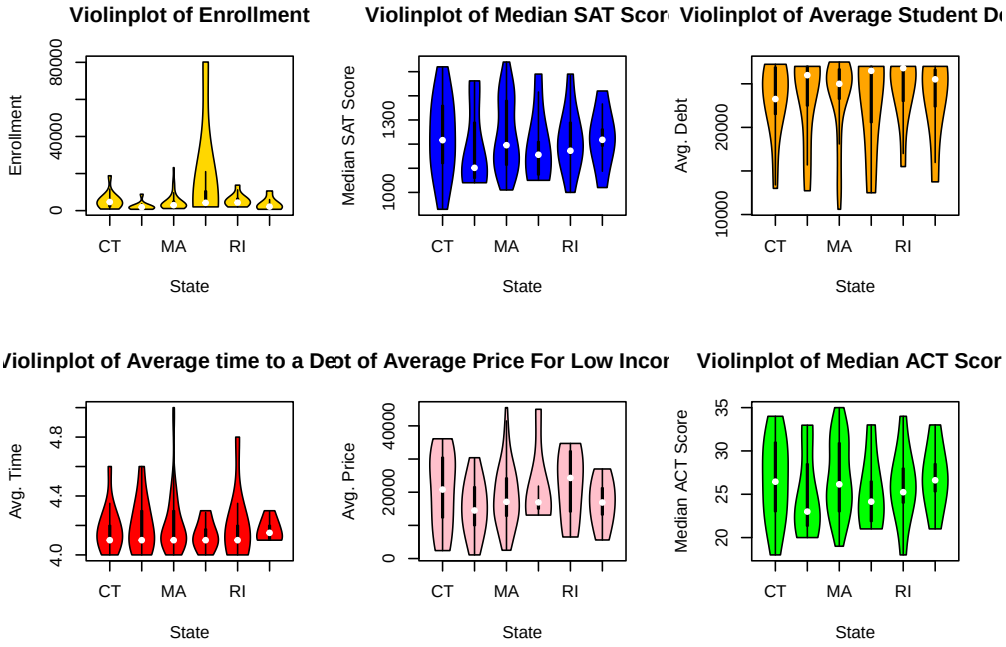



Interpretation of the Violin Plot: 1. There is a large positive skew among all the New England States for enrollment in their colleges. Also the enrollment in the colleges are centered around the median. 2. The Median SAT score seems to be positively skewed for NH while more of the data is centered around the median for VT. 3. All the states in New England have a large negative skew for average student debt (perhaps because a small portion of students get a lot of scholarship money) 4. There is a large positive skew is the average time to a degree for all states 5. The Average Price For Low Income Students seems to be symmetric in VT but positively skewed in MA and NH. 6. ME has a lower median ACT score compared to other universities from other states.

Note: There is a lot of information in this figure, above is just a few significant observations.

```
opar <- par(no.readonly=TRUE)
par(mfrow=c(2,3))
vioplot(CT$enrollment, ME$enrollment, MA$enrollment,
        NH$enrollment, RI$enrollment, VT$enrollment,
```

```
        names = c("CT", "ME", "MA", "NH", "RI", "VT"),
        col="gold",
        main = "Violinplot of Enrollment",
        xlab ="State",
        ylab = "Enrollment")
vioplot(CT$Median_SAT_Score, ME$Median_SAT_Score, MA$Median_SAT_Score,
        NH$Median_SAT_Score, RI$Median_SAT_Score, VT$Median_SAT_Score,
        names = c("CT", "ME", "MA", "NH", "RI", "VT"),
        col="blue",
        main = "Violinplot of Median SAT Score",
        xlab ="State",
        ylab = "Median SAT Score")
vioplot(CT$average_student_debt, ME$average_student_debt, MA$average_student_debt,
        NH$average_student_debt, RI$average_student_debt, VT$average_student_debt,
        names = c("CT", "ME", "MA", "NH", "RI", "VT"),
        col="orange",
        main = "Violinplot of Average Student Debt",
        xlab ="State",
        ylab = "Avg. Debt")
vioplot(CT$average_time_to_a_degree, ME$average_time_to_a_degree, MA$average_time_to_a_d
        NH$average_time_to_a_degree, RI$average_time_to_a_degree, VT$average_time_to_a_d
        names = c("CT", "ME", "MA", "NH", "RI", "VT"),
        col="red",
        main = "Violinplot of Average time to a Degree",
        xlab ="State",
        ylab = "Avg. Time")
vioplot(CT$average_price_for_low_income_students, ME$average_price_for_low_income_studen
        MA$average_price_for_low_income_students, NH$average_price_for_low_income_studen
        RI$average_price_for_low_income_students, VT$average_price_for_low_income_studen
        names = c("CT", "ME", "MA", "NH", "RI", "VT"),
        col="pink",
        main = "Violinplot of Average Price For Low Income Students",
        xlab ="State",
        ylab = "Avg. Price")
vioplot(CT$Median_ACT_Score, ME$Median_ACT_Score, MA$Median_ACT_Score,
        NH$Median_ACT_Score, RI$Median_ACT_Score, VT$Median_ACT_Score,
        names = c("CT", "ME", "MA", "NH", "RI", "VT"),
        col="green",
        main = "Violinplot of Median ACT Score",
        xlab ="State",
        ylab = "Median ACT Score")
```




```
par(opar)
```

In the following tests, a One-way ANOVA test was used. This is because a One-way Anova test allows to test/compare the means of 2 or more groups which is the purpose of the following questions.

**Question: Is there a different between the median SAT and median ACT scores between each of the 6 states in New England in the dataset?** The first test, the null hypothesis is that average Median SAT score for each state in New England is equal and the alt hyp. is that at least two states do not have the same average Median SAT score. the p-value (0.907) is very large indicating that we fail to reject the null hyp. and conclude that there is not significant difference in the average Median SAT score amongst the states of New England at the 0.05 significance level.

The second test, the null hypothesis is that average Median ACT score for each state in New England is equal and the alt hyp. is that at least two states do not have the same average Median ACT score. the p-value (0.832) is very large indicating that we fail to reject the null hyp. and conclude that there is not significant difference in the average Median ACT score amongst the states of New England at the 0.05 significance level.

```
aov.fit1 <- aov(Median_SAT_Score ~ College_Location_States, data = new_england)
summary(aov.fit1)
```

```
                       Df  Sum Sq Mean Sq F value Pr(>F)
College_Location_States  5   35500    7100   0.301  0.911
Residuals               87 2055133   23622
```

```
aov.fit1 <- aov(Median_ACT_Score ~ College_Location_States, data = new_england)
summary(aov.fit1)
```

```
                        Df Sum Sq Mean Sq F value Pr(>F)
College_Location_States  5   44.1    8.81   0.436  0.822
Residuals               87 1757.6   20.20
```

**Question: Is there a different between the median graduation rate between each of the 6 states in New England in the dataset?** The null hyp. for this test is that there is no significant difference between the mean graduation rate among CT, ME, MA, NH, RI, and VT. And the alt hyp. is that there is a significant difference among at least 2 of the states. The p-value (0.865) shows that we fail to reject the null hypothesis and that there is no significant difference between the graduation rates of CT, ME, MA, NH, RI, and VT at the 0.05 significance level.

```
aov.fit2 <- aov(graduation_rate ~ College_Location_States, data = new_england)
summary(aov.fit2)
```

```
                        Df Sum Sq Mean Sq F value Pr(>F)
College_Location_States  5    400   79.96   0.375  0.865
Residuals               87  18573  213.48
```

**Question: Is there a different between the acceptence rate between each of the 6 states in New England in the dataset?** The null hyp. for this test is that there is no significant difference between the mean acceptance rate among CT, ME, MA, NH, RI, and VT. And the alt hyp. is that there is a significant difference among at least 2 of the states. The p-value (0.89) shows that we fail to reject the null hypothesis and that there is no significant difference between the acceptance rates of CT, ME, MA, NH, RI, and VT at the 0.05 significance level.

```
aov.fit3 <- aov(acceptance_rate ~ College_Location_States, data = new_england)
summary(aov.fit3)
```

```
                        Df Sum Sq Mean Sq F value Pr(>F)
College_Location_States  5   1230   245.9   0.335  0.891
Residuals               87  63892   734.4
```

**Question: Is there a different between the enrollment between each of the 6 states in New England in the dataset?** The null hyp. for this test is that there is no significant difference between the mean enrollment among CT, ME, MA, NH, RI, and VT. And the alt hyp. is that there is a significant difference among at least 2 of the states.

The p-value (0.0135) shows that we reject the null hypothesis and that there is significant difference between at least 2 of the states in enrollment at the 0.05 significance level.

```
new_england$College_Location_States <- factor(new_england$College_Location_States)
aov.fit <- aov(enrollment ~ College_Location_States, data = new_england)
summary(aov.fit)
```

```
                        Df    Sum Sq   Mean Sq F value Pr(>F)
College_Location_States  5 1.082e+09 216387902   3.065 0.0135 *
Residuals               87 6.143e+09  70603729
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

To understand which states differ from each other for enrollment we use the `TukeyHSD()` function. The mean enrollment for MA and CT or VT and RI are not significantly different. But the Mean enrollment between NH and CT, NH and MH, and VT and NH are significantly different at the 0.05 significance level.

```
(Tfit <- TukeyHSD(aov.fit))
```

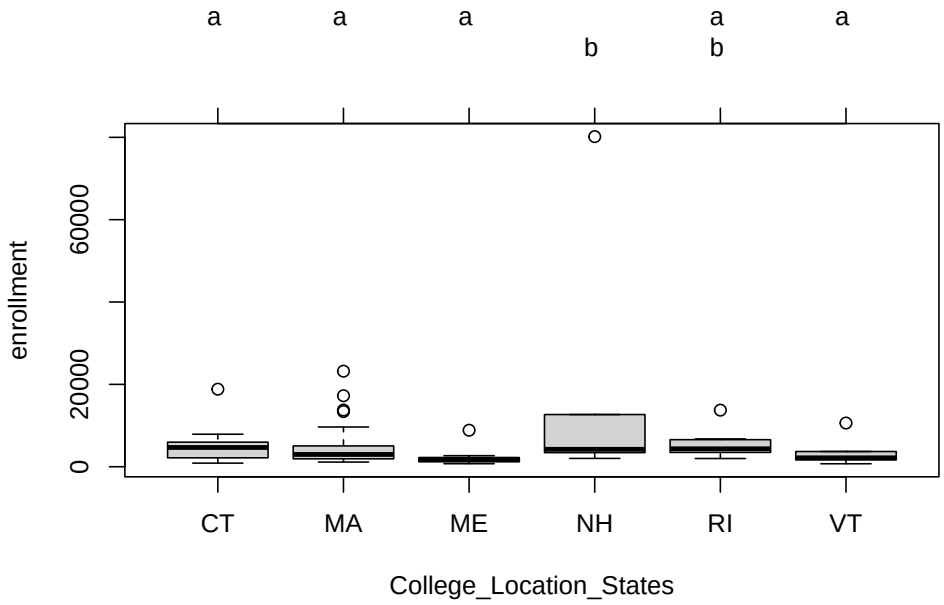
```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = enrollment ~ College_Location_States, data = new_england)

$College_Location_States
             diff         lwr        upr     p adj
MA-CT   -724.0224  -8380.1143  6932.0694 0.9997778
ME-CT  -2992.8671 -13024.5112  7038.7769 0.9527674
NH-CT  12509.1026    423.6196 24594.5855 0.0381054
RI-CT    212.9915 -10405.2579 10831.2408 0.9999999
VT-CT  -1779.2308 -13864.7137 10306.2522 0.9980923
ME-MA  -2268.8447 -10454.3135  5916.6241 0.9653845
NH-MA  13233.1250   2629.9691 23836.2809 0.0060324
RI-MA    937.0139  -7957.6607  9831.6885 0.9996230
VT-MA  -1055.2083 -11658.3643  9547.9476 0.9997147
NH-ME  15501.9697   3074.3772 27929.5622 0.0060694
RI-ME   3205.8586  -7800.2023 14211.9195 0.9572753
VT-ME   1213.6364 -11213.9561 13641.2289 0.9997399
RI-NH -12296.1111 -25201.8615   609.6393 0.0709838
VT-NH -14288.3333 -28425.8746  -150.7921 0.0460803
VT-RI  -1992.2222 -14897.9726 10913.5282 0.9976076
```

The following a visualization of the pariwise comparisons from above. Groups that have the same label don't have significantly different mean enrollment.

```
opar <- par(no.readonly=TRUE)
tuk <- glht(aov.fit, linfct=mcp(College_Location_States="Tukey"))
par(mar=c(5,4,6,2))
plot(cld(tuk, level=.05), col="lightgrey")
```

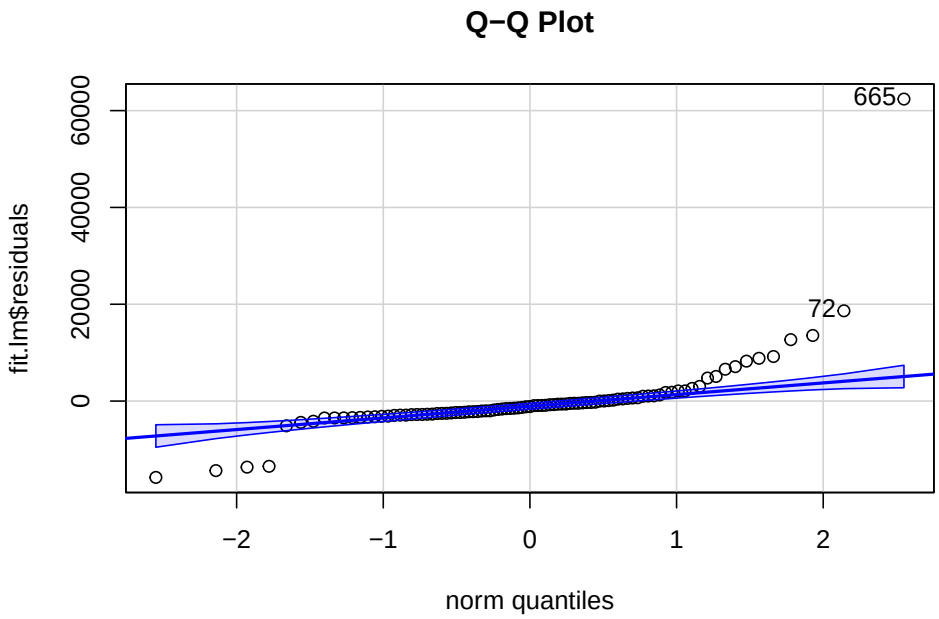



```
par(opar)
```

Because the ANOVA test shows shows some significance, lets see if the assumptions of ANOVA uphold. ANOVA requires that in each group the response is normally distributed, with equal variances.

Below, the normality assumption is violated because the point deviate significantly from the liear line.

```
fit.lm <- lm(enrollment ~ College_Location_States, data = new_england)
qqPlot(fit.lm$residuals, main="Q-Q Plot")
```




```
665  72
 88  10
```

The null hyp for the bartlett test is that the variances are equal, but the low p-value leads us to reject the null hyp. at the 0.05 significance level and accept the alt hyp. that the variances are not equal.

```
bartlett.test(enrollment ~ College_Location_States, data = new_england)
```

```
    Bartlett test of homogeneity of variances

data:  enrollment by College_Location_States
Bartlett's K-squared = 104.61, df = 5, p-value < 2.2e-16
```

From the output, you can see that, after adjusting for multiple testing (Bonferonni), there's an indication of outliers in the data.

```
outlierTest(aov.fit)
```

```
    rstudent unadjusted p-value Bonferroni p
665  16.5132         2.0996e-28   1.9526e-26
```

The one way ANOVA results are not to be used as none of the assumptions are valid.

I will now fit a logistic regression model to predict if the Sat or ACT in required on the ***entire*** `college_df` dataset.

My motivation behind this is because MANY colleges are starting to no longer require the SAT or ACT, and will not consider them as a part of your application even if you send them. As a result having a model that can predict if a college requires SAT or not will be useful.

```
college_df$SAT_ACT_required_for_Fall_2021 <- ifelse(college_df$SAT_ACT_required_for_Fall
```

Split the data into a train and test dataset (with out the categorical variables)

```
var <- names(college_df) %in% c("regular_application",
                                "college_names",
                                "College_Location_Town",
                                "College_Location_States")
set.seed(1000)
train <- sample(nrow(college_df), 0.8*nrow(college_df))
College.train <- college_df[train,][!var]
College.validate <- college_df[-train,][!var]
```

6 of the predictors are significant at the 0.05 level. However there is only a small drop in Null deviance indicating that this logistic regression model is not the best.

```
log_model <- glm(SAT_ACT_required_for_Fall_2021 ~ ., family = binomial(), College.train)
summary(log_model)
```

```
Call:
glm(formula = SAT_ACT_required_for_Fall_2021 ~ ., family = binomial(),
    data = College.train)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.65910  -0.61473  -0.42754  -0.07744   2.74719

Coefficients:
                                          Estimate Std. Error z value
(Intercept)                              9.158e+00  5.414e+00   1.692
est_full_Price_2020_2021                -5.591e-05  2.314e-05  -2.416
est_price_for_students_who_receive_aid  -2.906e-05  4.972e-05  -0.585
average_price_for_low_income_students    8.126e-05  3.929e-05   2.068
acceptance_rate                          3.445e-03  8.948e-03   0.385
Median_SAT_Score                        -6.324e-03  3.863e-03  -1.637
```

```
Median_ACT_Score                                -3.109e-02  1.302e-01  -0.239
enrollment                                      -2.703e-05  2.240e-05  -1.207
percent_of_need_met                             -1.984e-02  9.935e-03  -1.997
percent_of_students_who_get_merit_grants         3.713e-02  1.638e-02   2.267
average_merit_grant                              1.088e-05  3.628e-05   0.300
graduation_rate                                  1.016e-02  2.377e-02   0.428
average_time_to_a_degree                        -5.216e-01  9.155e-01  -0.570
average_student_debt                             2.790e-07  4.373e-05   0.006
average_salary_within_5_years                    3.600e-06  3.062e-05   0.118
percent_earning_more_than_28000                 -3.740e-03  2.221e-02  -0.168
percent_of_students_who_get_any_grants           1.399e-02  1.271e-02   1.101
percent_of_students_with_need_who_get_grants     4.552e-05  1.215e-02   0.004
                                                Pr(>|z|)
(Intercept)                                       0.0907 .
est_full_Price_2020_2021                          0.0157 *
est_price_for_students_who_receive_aid            0.5589
average_price_for_low_income_students             0.0386 *
acceptance_rate                                   0.7002
Median_SAT_Score                                  0.1016
Median_ACT_Score                                  0.8113
enrollment                                        0.2276
percent_of_need_met                               0.0458 *
percent_of_students_who_get_merit_grants          0.0234 *
average_merit_grant                               0.7643
graduation_rate                                   0.6690
average_time_to_a_degree                          0.5688
average_student_debt                              0.9949
average_salary_within_5_years                     0.9064
percent_earning_more_than_28000                   0.8662
percent_of_students_who_get_any_grants            0.2709
percent_of_students_with_need_who_get_grants      0.9970
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 527.71  on 590  degrees of freedom
Residual deviance: 432.92  on 573  degrees of freedom
AIC: 468.92

Number of Fisher Scoring iterations: 6
```

There are 5 false positive and 19 false negatives for the above model.

```
prob <- predict(log_model, College.validate, type="response")
logit.pred <- factor(prob > .5, labels=c("No", "Yes"))
table(logit.pred)
```

```
logit.pred
 No Yes
145   3
```

```
logit.perf <- table(College.validate$SAT_ACT_required_for_Fall_2021, logit.pred,
                    dnn=c("Actual", "Predicted"))
logit.perf
```

```
      Predicted
Actual  No Yes
     0 125   3
     1  20   0
```

Here I use a step wise selection to generate a model with fewer variables. Predictor variables are added or removed in order to obtain a model with a smaller AIC value. Notice how the AIC value is 449.48 whereas the previous model has an AIC value of 467.8, shows that the step wise model is a better fit.

```
logRegPrivate.step <- step(log_model, direction="backward")
```

```
Start:  AIC=468.92
SAT_ACT_required_for_Fall_2021 ~ est_full_Price_2020_2021 + est_price_for_students_who_r
    average_price_for_low_income_students + acceptance_rate +
    Median_SAT_Score + Median_ACT_Score + enrollment + percent_of_need_met +
    percent_of_students_who_get_merit_grants + average_merit_grant +
    graduation_rate + average_time_to_a_degree + average_student_debt +
    average_salary_within_5_years + percent_earning_more_than_28000 +
    percent_of_students_who_get_any_grants + percent_of_students_with_need_who_get_grant

                                                Df Deviance    AIC
- percent_of_students_with_need_who_get_grants   1   432.92 466.92
- average_student_debt                           1   432.92 466.92
- average_salary_within_5_years                  1   432.93 466.93
- percent_earning_more_than_28000                1   432.95 466.95
- Median_ACT_Score                               1   432.97 466.97
- average_merit_grant                            1   433.01 467.01
- acceptance_rate                                1   433.07 467.07
- graduation_rate                                1   433.10 467.10
```

```
- average_time_to_a_degree                      1   433.25 467.25
- est_price_for_students_who_receive_aid        1   433.26 467.26
- percent_of_students_who_get_any_grants        1   434.15 468.15
- enrollment                                    1   434.61 468.61
<none>                                              432.92 468.92
- Median_SAT_Score                              1   435.66 469.66
- percent_of_need_met                           1   436.89 470.89
- average_price_for_low_income_students         1   437.06 471.06
- percent_of_students_who_get_merit_grants      1   438.90 472.90
- est_full_Price_2020_2021                      1   439.26 473.26

Step:  AIC=466.92
SAT_ACT_required_for_Fall_2021 ~ est_full_Price_2020_2021 + est_price_for_students_who_r
    average_price_for_low_income_students + acceptance_rate +
    Median_SAT_Score + Median_ACT_Score + enrollment + percent_of_need_met +
    percent_of_students_who_get_merit_grants + average_merit_grant +
    graduation_rate + average_time_to_a_degree + average_student_debt +
    average_salary_within_5_years + percent_earning_more_than_28000 +
    percent_of_students_who_get_any_grants

                                       Df Deviance    AIC
- average_student_debt                  1   432.92 464.92
- average_salary_within_5_years         1   432.93 464.93
- percent_earning_more_than_28000       1   432.95 464.95
- Median_ACT_Score                      1   432.97 464.97
- average_merit_grant                   1   433.01 465.01
- acceptance_rate                       1   433.07 465.07
- graduation_rate                       1   433.10 465.10
- average_time_to_a_degree              1   433.25 465.25
- est_price_for_students_who_receive_aid 1  433.26 465.26
- percent_of_students_who_get_any_grants 1  434.37 466.37
- enrollment                            1   434.63 466.63
<none>                                      432.92 466.92
- Median_SAT_Score                      1   435.67 467.67
- percent_of_need_met                   1   436.98 468.98
- average_price_for_low_income_students 1   437.06 469.06
- percent_of_students_who_get_merit_grants 1 439.00 471.00
- est_full_Price_2020_2021              1   439.51 471.51

Step:  AIC=464.92
SAT_ACT_required_for_Fall_2021 ~ est_full_Price_2020_2021 + est_price_for_students_who_r
    average_price_for_low_income_students + acceptance_rate +
    Median_SAT_Score + Median_ACT_Score + enrollment + percent_of_need_met +
    percent_of_students_who_get_merit_grants + average_merit_grant +
    graduation_rate + average_time_to_a_degree + average_salary_within_5_years +
```

```
    percent_earning_more_than_28000 + percent_of_students_who_get_any_grants

                                          Df Deviance    AIC
- average_salary_within_5_years            1   432.93 462.93
- percent_earning_more_than_28000          1   432.95 462.95
- Median_ACT_Score                         1   432.97 462.97
- average_merit_grant                      1   433.01 463.01
- acceptance_rate                          1   433.07 463.07
- graduation_rate                          1   433.10 463.10
- average_time_to_a_degree                 1   433.26 463.26
- est_price_for_students_who_receive_aid   1   433.26 463.26
- percent_of_students_who_get_any_grants   1   434.37 464.37
- enrollment                               1   434.69 464.69
<none>                                         432.92 464.92
- Median_SAT_Score                         1   435.72 465.72
- percent_of_need_met                      1   436.99 466.99
- average_price_for_low_income_students    1   437.11 467.11
- percent_of_students_who_get_merit_grants 1   439.00 469.00
- est_full_Price_2020_2021                 1   439.75 469.75

Step:  AIC=462.93
SAT_ACT_required_for_Fall_2021 ~ est_full_Price_2020_2021 + est_price_for_students_who_r
    average_price_for_low_income_students + acceptance_rate +
    Median_SAT_Score + Median_ACT_Score + enrollment + percent_of_need_met +
    percent_of_students_who_get_merit_grants + average_merit_grant +
    graduation_rate + average_time_to_a_degree + percent_earning_more_than_28000 +
    percent_of_students_who_get_any_grants

                                          Df Deviance    AIC
- percent_earning_more_than_28000          1   432.95 460.95
- Median_ACT_Score                         1   432.98 460.98
- average_merit_grant                      1   433.03 461.03
- acceptance_rate                          1   433.07 461.07
- graduation_rate                          1   433.11 461.11
- average_time_to_a_degree                 1   433.26 461.26
- est_price_for_students_who_receive_aid   1   433.27 461.27
- percent_of_students_who_get_any_grants   1   434.37 462.37
- enrollment                               1   434.73 462.73
<none>                                         432.93 462.93
- Median_SAT_Score                         1   435.72 463.72
- average_price_for_low_income_students    1   437.12 465.12
- percent_of_need_met                      1   437.15 465.15
- percent_of_students_who_get_merit_grants 1   439.04 467.04
- est_full_Price_2020_2021                 1   439.81 467.81
```

```
Step:  AIC=460.95
SAT_ACT_required_for_Fall_2021 ~ est_full_Price_2020_2021 + est_price_for_students_who_r
    average_price_for_low_income_students + acceptance_rate +
    Median_SAT_Score + Median_ACT_Score + enrollment + percent_of_need_met +
    percent_of_students_who_get_merit_grants + average_merit_grant +
    graduation_rate + average_time_to_a_degree + percent_of_students_who_get_any_grants

                                        Df Deviance    AIC
- Median_ACT_Score                       1   433.00 459.00
- average_merit_grant                    1   433.04 459.04
- acceptance_rate                        1   433.08 459.08
- graduation_rate                        1   433.11 459.11
- est_price_for_students_who_receive_aid 1   433.29 459.29
- average_time_to_a_degree               1   433.30 459.30
- percent_of_students_who_get_any_grants 1   434.46 460.46
- enrollment                             1   434.78 460.78
<none>                                       432.95 460.95
- Median_SAT_Score                       1   435.76 461.76
- percent_of_need_met                    1   437.17 463.17
- average_price_for_low_income_students  1   437.36 463.36
- percent_of_students_who_get_merit_grants 1 439.09 465.09
- est_full_Price_2020_2021               1   440.08 466.08

Step:  AIC=459
SAT_ACT_required_for_Fall_2021 ~ est_full_Price_2020_2021 + est_price_for_students_who_r
    average_price_for_low_income_students + acceptance_rate +
    Median_SAT_Score + enrollment + percent_of_need_met + percent_of_students_who_get_me
    average_merit_grant + graduation_rate + average_time_to_a_degree +
    percent_of_students_who_get_any_grants

                                        Df Deviance    AIC
- average_merit_grant                    1   433.08 457.08
- graduation_rate                        1   433.13 457.13
- acceptance_rate                        1   433.14 457.14
- average_time_to_a_degree               1   433.33 457.33
- est_price_for_students_who_receive_aid 1   433.37 457.37
- percent_of_students_who_get_any_grants 1   434.53 458.53
- enrollment                             1   434.86 458.86
<none>                                       433.00 459.00
- percent_of_need_met                    1   437.27 461.27
- average_price_for_low_income_students  1   437.42 461.42
- percent_of_students_who_get_merit_grants 1 439.15 463.15
- est_full_Price_2020_2021               1   440.11 464.11
- Median_SAT_Score                       1   440.99 464.99
```

```
Step:  AIC=457.08
SAT_ACT_required_for_Fall_2021 ~ est_full_Price_2020_2021 + est_price_for_students_who_r
    average_price_for_low_income_students + acceptance_rate +
    Median_SAT_Score + enrollment + percent_of_need_met + percent_of_students_who_get_me
    graduation_rate + average_time_to_a_degree + percent_of_students_who_get_any_grants

                                          Df Deviance    AIC
- acceptance_rate                          1   433.21 455.21
- graduation_rate                          1   433.26 455.26
- average_time_to_a_degree                 1   433.45 455.45
- est_price_for_students_who_receive_aid   1   433.60 455.60
- percent_of_students_who_get_any_grants   1   434.84 456.84
- enrollment                               1   434.99 456.99
<none>                                         433.08 457.08
- percent_of_need_met                      1   437.27 459.27
- average_price_for_low_income_students    1   437.87 459.87
- percent_of_students_who_get_merit_grants 1   439.29 461.29
- Median_SAT_Score                         1   441.04 463.04
- est_full_Price_2020_2021                 1   442.01 464.01

Step:  AIC=455.21
SAT_ACT_required_for_Fall_2021 ~ est_full_Price_2020_2021 + est_price_for_students_who_r
    average_price_for_low_income_students + Median_SAT_Score +
    enrollment + percent_of_need_met + percent_of_students_who_get_merit_grants +
    graduation_rate + average_time_to_a_degree + percent_of_students_who_get_any_grants

                                          Df Deviance    AIC
- graduation_rate                          1   433.41 453.41
- average_time_to_a_degree                 1   433.61 453.61
- est_price_for_students_who_receive_aid   1   433.70 453.70
- percent_of_students_who_get_any_grants   1   434.92 454.92
- enrollment                               1   435.14 455.14
<none>                                         433.21 455.21
- percent_of_need_met                      1   437.44 457.44
- average_price_for_low_income_students    1   437.94 457.94
- percent_of_students_who_get_merit_grants 1   439.58 459.58
- Median_SAT_Score                         1   441.92 461.92
- est_full_Price_2020_2021                 1   442.30 462.30

Step:  AIC=453.41
SAT_ACT_required_for_Fall_2021 ~ est_full_Price_2020_2021 + est_price_for_students_who_r
    average_price_for_low_income_students + Median_SAT_Score +
    enrollment + percent_of_need_met + percent_of_students_who_get_merit_grants +
    average_time_to_a_degree + percent_of_students_who_get_any_grants
```

```
                                          Df Deviance    AIC
- est_price_for_students_who_receive_aid   1   433.77 451.77
- average_time_to_a_degree                 1   434.07 452.07
- percent_of_students_who_get_any_grants   1   435.13 453.13
- enrollment                               1   435.30 453.30
<none>                                         433.41 453.41
- percent_of_need_met                      1   437.57 455.57
- average_price_for_low_income_students    1   437.96 455.96
- percent_of_students_who_get_merit_grants 1   439.71 457.71
- est_full_Price_2020_2021                 1   442.49 460.49
- Median_SAT_Score                         1   445.36 463.36

Step:  AIC=451.77
SAT_ACT_required_for_Fall_2021 ~ est_full_Price_2020_2021 + average_price_for_low_income
    Median_SAT_Score + enrollment + percent_of_need_met + percent_of_students_who_get_me
    average_time_to_a_degree + percent_of_students_who_get_any_grants

                                          Df Deviance    AIC
- average_time_to_a_degree                 1   434.32 450.32
- enrollment                               1   435.57 451.57
<none>                                         433.77 451.77
- percent_of_students_who_get_any_grants   1   435.92 451.92
- percent_of_need_met                      1   437.70 453.70
- percent_of_students_who_get_merit_grants 1   439.71 455.71
- average_price_for_low_income_students    1   440.88 456.88
- Median_SAT_Score                         1   448.32 464.32
- est_full_Price_2020_2021                 1   448.62 464.62

Step:  AIC=450.32
SAT_ACT_required_for_Fall_2021 ~ est_full_Price_2020_2021 + average_price_for_low_income
    Median_SAT_Score + enrollment + percent_of_need_met + percent_of_students_who_get_me
    percent_of_students_who_get_any_grants

                                          Df Deviance    AIC
<none>                                         434.32 450.32
- percent_of_students_who_get_any_grants   1   436.62 450.62
- enrollment                               1   437.14 451.14
- percent_of_need_met                      1   437.99 451.99
- percent_of_students_who_get_merit_grants 1   440.59 454.59
- average_price_for_low_income_students    1   442.45 456.45
- Median_SAT_Score                         1   448.34 462.34
- est_full_Price_2020_2021                 1   448.84 462.84
```

```
summary(logRegPrivate.step)
```

```
Call:
glm(formula = SAT_ACT_required_for_Fall_2021 ~ est_full_Price_2020_2021 +
    average_price_for_low_income_students + Median_SAT_Score +
    enrollment + percent_of_need_met + percent_of_students_who_get_merit_grants +
    percent_of_students_who_get_any_grants, family = binomial(),
    data = College.train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.7284  -0.6146  -0.4261  -0.0873   2.7990

Coefficients:
                                          Estimate Std. Error z value Pr(>|z|)
(Intercept)                              6.615e+00  2.103e+00   3.145 0.001660
est_full_Price_2020_2021                -5.692e-05  1.549e-05  -3.674 0.000238
average_price_for_low_income_students    6.501e-05  2.305e-05   2.820 0.004797
Median_SAT_Score                        -6.408e-03  1.784e-03  -3.592 0.000328
enrollment                              -3.237e-05  2.110e-05  -1.534 0.125104
percent_of_need_met                     -1.825e-02  9.483e-03  -1.924 0.054299
percent_of_students_who_get_merit_grants 3.527e-02  1.490e-02   2.368 0.017896
percent_of_students_who_get_any_grants   1.645e-02  1.103e-02   1.492 0.135818

(Intercept)                             **
est_full_Price_2020_2021                ***
average_price_for_low_income_students   **
Median_SAT_Score                        ***
enrollment
percent_of_need_met                     .
percent_of_students_who_get_merit_grants *
percent_of_students_who_get_any_grants
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 527.71  on 590  degrees of freedom
Residual deviance: 434.32  on 583  degrees of freedom
AIC: 450.32

Number of Fisher Scoring iterations: 6
```

The step wise model shows that est_full_Price_2020_2021, average_price_for_low_income_students,

Median_SAT_Score, enrollment, percent_of_students_who_get_merit_grants, and percent_of_need_met are all good predictors.

The step wise model has 4 false positives and 20 false negatives. This is better than the previous model

```
prob <- predict(logRegPrivate.step, College.validate, type="response")
logit.pred <- factor(prob > .5, labels=c("No", "Yes"))
table(logit.pred)
```

```
logit.pred
 No Yes
145   3
```

```
logit.perf2 <- table(College.validate$SAT_ACT_required_for_Fall_2021, logit.pred,
                     dnn=c("Actual", "Predicted"))
logit.perf2
```

```
      Predicted
Actual  No Yes
     0 125   3
     1  20   0
```

The performance shows that the two models do that not really differ by much. The step wise model has a lower sensitivity and while has a higher specificity (true negative rate). The regular model also has a higher negative predictive rate.

```
performance <- function(table, n=2){
  if(!all(dim(table) == c(2,2)))
  stop("Must be a 2 x 2 table")
  tn = table[1,1]
  fp = table[1,2]
  fn = table[2,1]
  tp = table[2,2]
  sensitivity = tp/(tp+fn)
  specificity = tn/(tn+fp)
  ppp = tp/(tp+fp)
  npp = tn/(tn+fn)
  hitrate = (tp+tn)/(tp+tn+fp+fn)
  result <- paste("Sensitivity = ", round(sensitivity, n) ,
  "\nSpecificity = ", round(specificity, n),
  "\nPositive Predictive Value = ", round(ppp, n),
  "\nNegative Predictive Value = ", round(npp, n),
```

```
  "\nAccuracy = ", round(hitrate, n), "\n", sep="")
  cat(result)
}

performance(logit.perf)
```

```
Sensitivity = 0
Specificity = 0.98
Positive Predictive Value = 0
Negative Predictive Value = 0.86
Accuracy = 0.84
```

```
performance(logit.perf2)
```

```
Sensitivity = 0
Specificity = 0.98
Positive Predictive Value = 0
Negative Predictive Value = 0.86
Accuracy = 0.84
```

In conclusion. I have shown in the One way ANVOA test shows that there is a significant difference in enrollment among the universities of New England but the the assumptions were not valid. Also, there is no significant difference between median SAT and ACT scores and graduation rate and acceptance rate. The graphs in this pdf show that the distribution of full price is roughly the same for all the states with outliers in RI. Ad the violin plots show positive skewing for enrollment and avg time to a degree while there is a negative skew for avg student debt.

A future step for me would be to analyse other regions of USA the same way I did for New England and then compare different regions within the data set. By doing this I will have a more holistic idea of how this data is.

I encountered many problems. For example when webscraping the data, because there was a lot, it would take a long time to run. But I was able to overcome this by looking through documentation and make sure sure I used proper functions.

Sources:

https://www.dataquest.io/blog/web-scraping-in-r-rvest/ https://cran.r-project.org/web/packages/missForest/missForest.pdf https://cran.r-project.org/web/packages/rvest/rvest.pdf https://www.marsja.se/how-to-remove-duplicates-in-r-rows-columns-dplyr/ https://chrome.google.com/webstore/detail/selectorgadget/mhjhnkcfbdhnjickkkdbjoemdmbfginb?hl=en https://money.com/best-colleges/