Student Exam Predictions

Kevin Vanderwater

2023-04-05

Executive Summary

This analysis was completed in an attempt to predict test scores based on 8 categories. The data used for this assessment is not copyrighted however comes from Dr. Royce Kimmons using his online data generator running it 5 times using n=1000 for an overall sample size of 5000. The data can you derived from:
http://roycekimmons.com/tools/generated_data/exams

For this assessment it was decided that Root Mean Square Error (RMSE) to assess the overall predictive strength of the model. The database used was a relatively small data set of 5000 observations, over eight variables (culminating in nine with the incorporation of an average column). For this data set the goal was to met an RMSE in the range of .1 to .6 for a fit of about 75%. Anything below that range would need to be checked for over fitting of the model, and above that not having relevance to meet the objective.

#Libraries

```
library(tidyverse)
library(caret)
library(ggthemes)
library(dplyr)
library(caTools)
library(randomForest)
library(e1071)

library(latexpdf)
library(tinytex)
```

#Data download from github

```
student_exams <-
read.csv("https://raw.githubusercontent.com/chemicalburn09/Student-
Exams/main/studentexams.csv")
```

#Data Pre-processing ##structure

```
str(student_exams)

## 'data.frame':    5000 obs. of  8 variables:
##  $ gender       : chr  "male" "male" "female" "male" ...
##  $ ethnicity    : chr  "group D" "group C" "group C" "group C" ...
```

```
##  $ parental_edu : chr  "bachelor's degree" "some high school" "bachelor's
degree" "associate's degree" ...
##  $ lunch        : chr  "standard" "standard" "standard" "free/reduced" ...
##  $ prep_course  : chr  "none" "completed" "none" "none" ...
##  $ math_score   : int  76 68 84 51 40 54 40 56 75 97 ...
##  $ reading_score: int  73 65 97 53 45 54 41 69 85 98 ...
##  $ writing_score: int  65 64 94 54 41 52 49 66 86 91 ...

head(student_exams)

##   gender ethnicity       parental_edu        lunch prep_course math_score
## 1   male   group D  bachelor's degree     standard        none         76
## 2   male   group C   some high school     standard   completed         68
## 3 female   group C  bachelor's degree     standard        none         84
## 4   male   group C associate's degree free/reduced        none         51
## 5 female   group C   some high school free/reduced        none         40
## 6   male   group A        high school     standard        none         54
##   reading_score writing_score
## 1            73            65
## 2            65            64
## 3            97            94
## 4            53            54
## 5            45            41
## 6            54            52

dim(student_exams)

## [1] 5000    8
```

##blank/NA check

```
sum(is.na(student_exams))

## [1] 0
```

During this phase of data pre-processing the columns were switched to factors. As well as adding a new column with the average of the three tests.

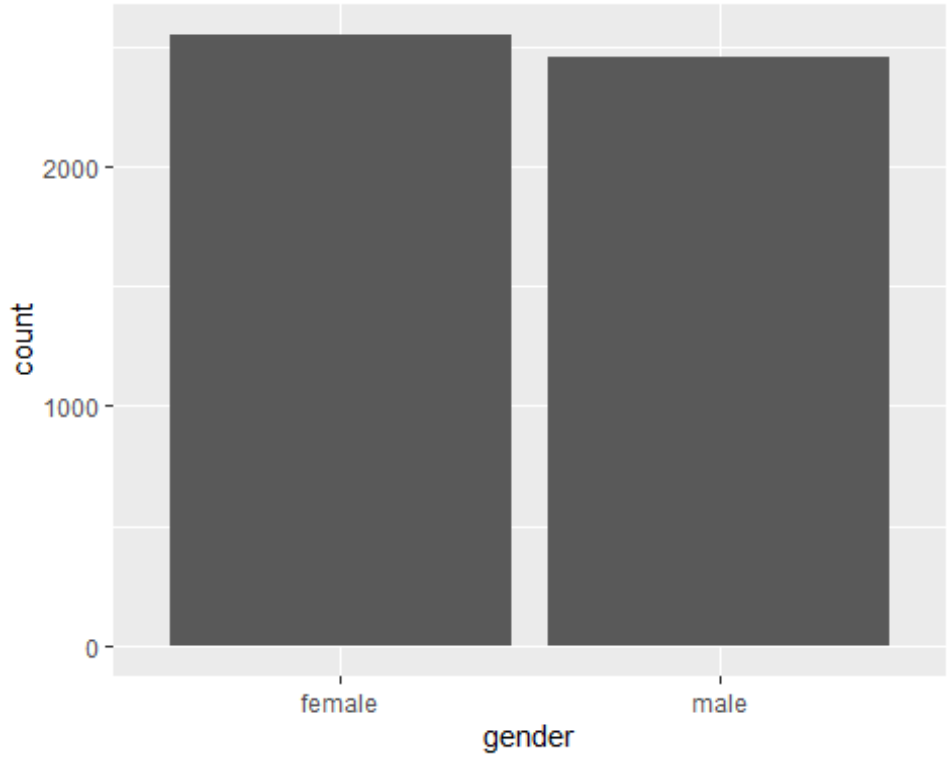
```
student_exams$gender <- as.factor(student_exams$gender)
student_exams$ethnicity <- as.factor(student_exams$ethnicity)
student_exams$parental_edu <- as.factor(student_exams$parental_edu)
student_exams$lunch <- as.factor(student_exams$lunch)
student_exams$prep_course <- as.factor(student_exams$prep_course)

#Create new column for average score
student_exams$avg_score <- apply(student_exams[,6:8],1,mean)
```

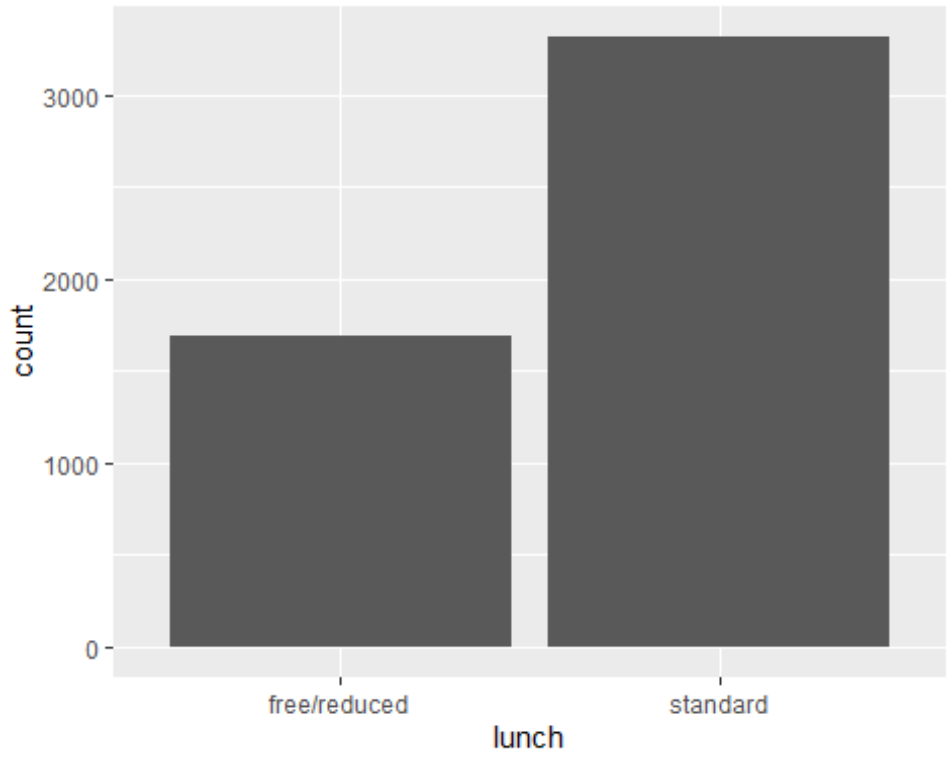
#Exploritory Data Analysis The over arching goal within the EDA process was to plot out the variables so look for similarities that may be evident in the data set.

This bar plot looks at the overall makeup of the gender variable checking to make sure that we have even representation. It shows that we are very close to being evenly distributed between genders.

```
ggplot(student_exams, aes(x = gender)) + geom_bar()
```



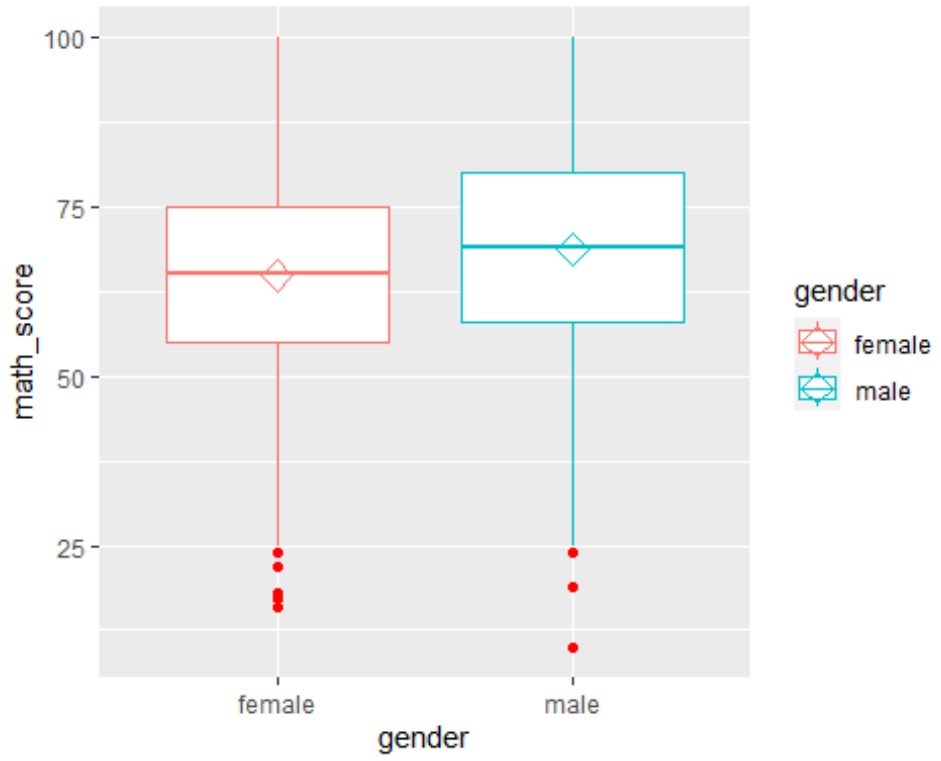

```
ggplot(student_exams, aes(x = lunch)) + geom_bar()
```



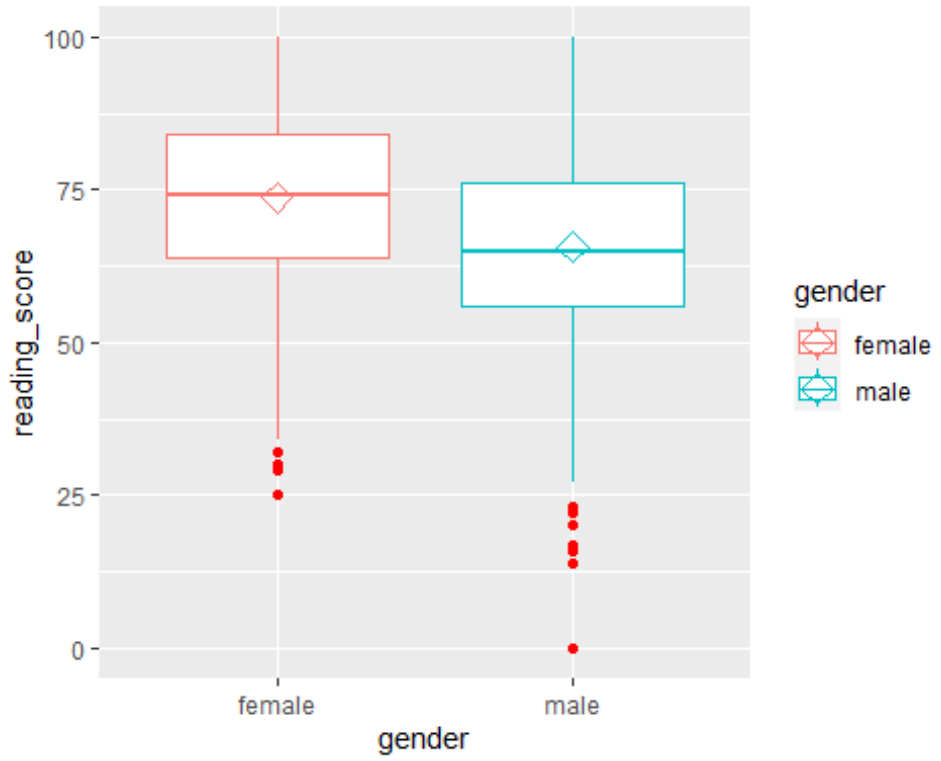
Next, we looked at type of paid/free lunch.

The next plot looks are the mean and spread within one standard deviation of gender and test scores, also highlighting know outliers in the data (noted as the red dots).

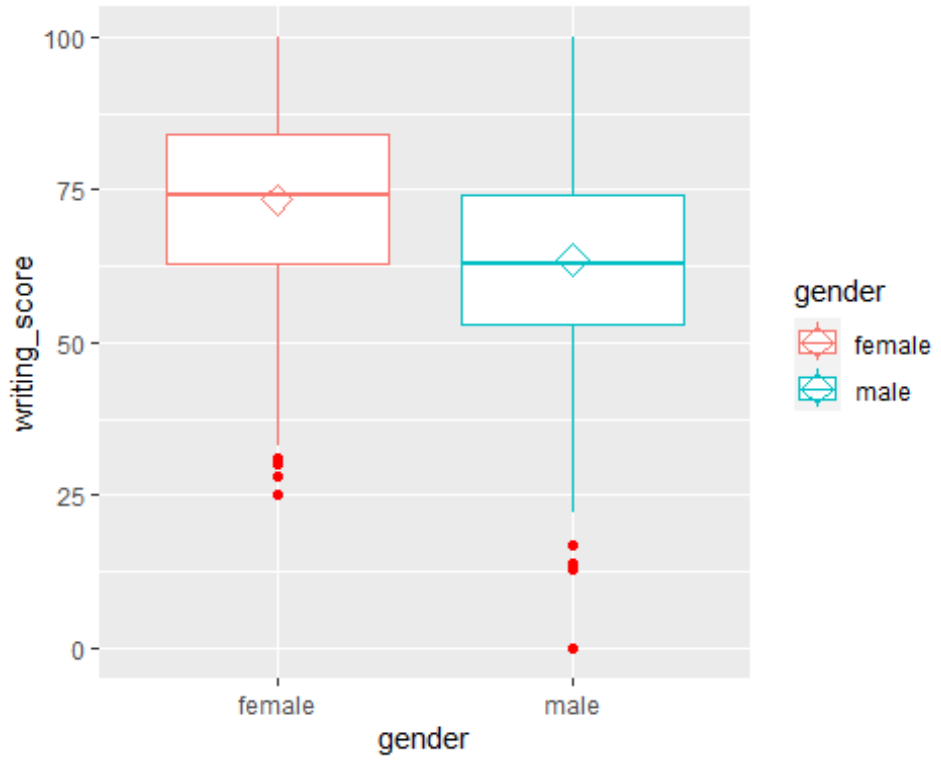
```
ggplot(student_exams, aes(x = gender, y = math_score, color = gender)) +
geom_boxplot(outlier.colour="red") +
  stat_summary(fun = mean, geom="point", shape=23, size=4)
```




```
ggplot(student_exams, aes(x = gender, y = reading_score, color = gender)) +
geom_boxplot(outlier.colour="red") +
  stat_summary(fun = mean, geom="point", shape=23, size=4)
```



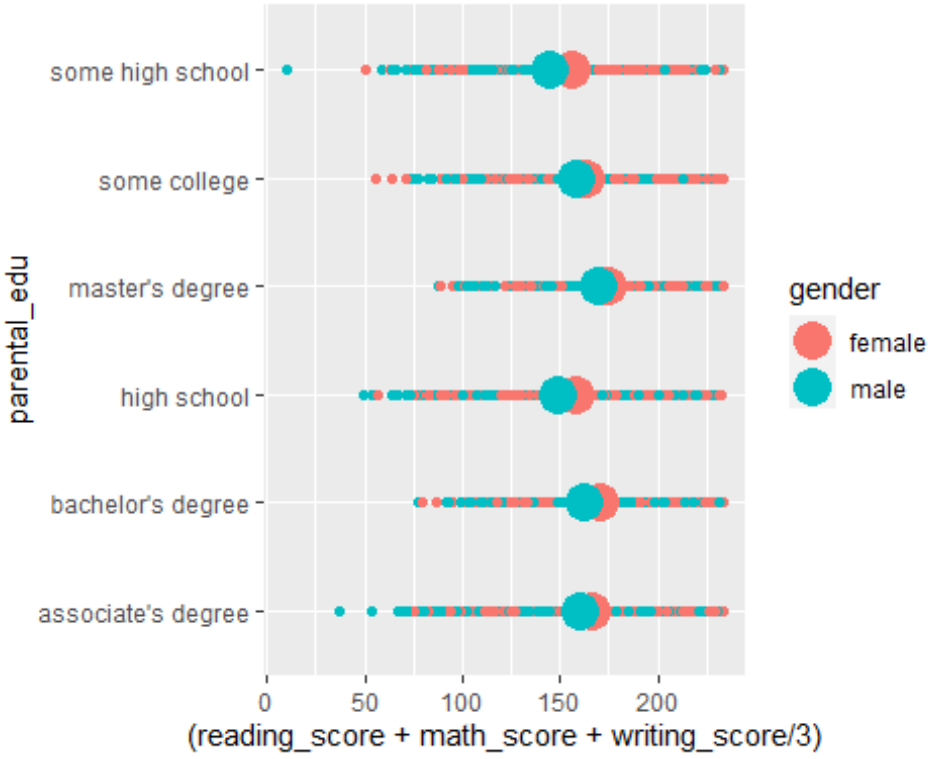
```
ggplot(student_exams, aes(x = gender, y = writing_score, color = gender)) +
geom_boxplot(outlier.colour="red") +
  stat_summary(fun = mean, geom="point", shape=23, size=4)
```



The next plot compares the average of the three test scores against gender and parental education levels. Not surprising it shows that parents who place a high level of "value" on education have children that receive higher grades. What can be noted here is that in every category females out perform the male students.

```
ggplot(student_exams, aes(x = (reading_score+math_score+writing_score/3), y =
parental_edu, colour = gender)) +
  geom_point() +
  stat_summary(fun = mean, geom = "pointrange", size = 1.5)

## Warning: Removed 12 rows containing missing values (`geom_segment()`).
```



This plot focuses a look at gender, average score, and the ethnicity group. It is unknown from the data provided which ethnicities are represented, however what we can see that their are differences between each group. The genders show similar patterns to test scores and gender above for each category. Due to the 50 point different between group B and group E, this analysis recommends further investigation. Cultural differences are clearly indicated that could be studied in the hopes to identify areas of improvement or targeted resources so ethnic group under performing.

```
ggplot(student_exams, aes(x = (reading_score+math_score+writing_score/3), y =
ethnicity, colour = gender)) +
  geom_point() +
  stat_summary(fun = mean, geom = "pointrange", size = 1.5)

## Warning: Removed 10 rows containing missing values (`geom_segment()`).
```

The last steps before model building is to split the dataset into training and testing partitions using an 80/20 split. As well as scaling the three test scores and average score column to bring them closer to the mean (in each column). This was done because of the unknown in terms of grading that could have accrued as teachers/professors each grade differently. Although all marks are calculated in a 0-100 point scale, scaling takes out some of the potential bias that could have accrued without our intimate known of the data collection process.

```
set.seed(1984)

split = sample.split(student_exams, SplitRatio = 0.80)
student_train = subset(student_exams, split == TRUE)
student_test = subset(student_exams, split == FALSE)

#Scale data (scores)
student_train$math_score <- scale(student_train$math_score)
student_train$writing_score <- scale(student_train$writing_score)
student_train$reading_score <- scale(student_train$reading_score)
student_train$avg_score <- scale(student_train$avg_score)

student_test$math_score <- scale(student_test$math_score)
student_test$writing_score <- scale(student_test$writing_score)
student_test$reading_score <- scale(student_test$reading_score)
student_test$avg_score <- scale(student_test$avg_score)
```

#Model Building

Two types of models were used in an attempt to using the variables provided be able to estimate the score each student received.

```
#Model Building #1 multivariate linear regression math scores
set.seed(1984)

model10 = lm(math_score ~ reading_score + writing_score, data =
student_train)
summary(model10)

##
## Call:
## lm(formula = math_score ~ reading_score + writing_score, data =
student_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.5753 -0.4008 -0.0046  0.4069  1.8085
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.617e-16  9.085e-03    0.00        1
## reading_score 4.607e-01  3.033e-02   15.19   <2e-16 ***
## writing_score 3.729e-01  3.033e-02   12.29   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5666 on 3887 degrees of freedom
## Multiple R-squared:  0.6791, Adjusted R-squared:  0.6789
## F-statistic:  4112 on 2 and 3887 DF,  p-value: < 2.2e-16

varImp(model10)

##                Overall
## reading_score 15.18991
## writing_score 12.29307

y_pred10 <- predict(model10, student_test)
#summary(y_pred)

RMSE(y_pred10, student_test$math_score)

## [1] 0.5598695

##Returned RMSE scaled is 0.5598695
```

This RMSE score fit into our overall goal as stated above. Determining that by using scores from the reading and writing columns we are able to estimate the math score. This trend continued as we checked the other scores against each other.

```r
#Model Building #2 multivariate linear regression reading scores
set.seed(1984)

model11 = lm(reading_score ~ math_score + writing_score, data =
student_train)
summary(model11)

##
## Call:
## lm(formula = reading_score ~ math_score + writing_score, data =
student_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.04411 -0.19017  0.00136  0.20192  0.98821
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -4.801e-16  4.668e-03    0.00        1
## math_score     1.216e-01  8.007e-03   15.19   <2e-16 ***
## writing_score  8.553e-01  8.007e-03  106.82   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2911 on 3887 degrees of freedom
## Multiple R-squared:  0.9153, Adjusted R-squared:  0.9152
## F-statistic: 2.1e+04 on 2 and 3887 DF,  p-value: < 2.2e-16

varImp(model11)

##                 Overall
## math_score     15.18991
## writing_score 106.81751

y_pred11 <- predict(model11, student_test)
#summary(y_pred)

RMSE(y_pred11, student_test$math_score)

## [1] 0.536046

##Returned RMSE scaled is 0.536046

#MOdel Building #3 multivariate linear regression writing scores
set.seed(1984)

model12 = lm(writing_score ~ math_score + reading_score, data =
student_train)
summary(model12)
```

```
##
## Call:
## lm(formula = writing_score ~ math_score + reading_score, data =
student_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.96864 -0.19736  0.00458  0.19568  1.01890
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.743e-16  4.714e-03    0.00        1
## math_score    1.004e-01  8.165e-03   12.29   <2e-16 ***
## reading_score 8.721e-01  8.165e-03  106.82   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.294 on 3887 degrees of freedom
## Multiple R-squared:  0.9136, Adjusted R-squared:  0.9136
## F-statistic: 2.055e+04 on 2 and 3887 DF,  p-value: < 2.2e-16

varImp(model12)

##                 Overall
## math_score     12.29307
## reading_score 106.81751

y_pred12 <- predict(model12, student_test)
#summary(y_pred)

RMSE(y_pred12, student_test$math_score)

## [1] 0.5214318

##Returned RMSE scaled is 0.5214318
```

This proves two important things, that factors within the data are affecting the outcomes in a manor that we can use them to predict the overall test scores. Each RMSE was within out goal range.

##Random Forest

For the final model the research was done using a random forest regression to determine if by using the whole dataset the math test score could be estimated. It was not necessary to look at all/each test score since the models above showed the correlation between the two columns of data.

```
#Model Building #4 Randomforest
#remove average column so that it doesn't skew results

set.seed(1984)
```

```
classifier = randomForest(x = student_train[,-9],
                          y = student_train$math_score,
                          ntree = 500, random_state = 0)

## Warning in rfout$mse/(var(y) * (n - 1)/n): Recycling array of length 1 in
vector-array arithmetic is deprecated.
##   Use c() or as.vector() instead.

y_pred13 = predict(classifier, student_test[,-9])

RMSE(y_pred13, student_test$math_score)

## [1] 0.1077161

##Returned RMSE scaled is 0.1114175

varImp(classifier)

##                  Overall
## gender         123.40057
## ethnicity       68.70666
## parental_edu    35.40944
## lunch          145.65790
## prep_course     11.67066
## math_score    1911.54721
## reading_score  772.57679
## writing_score  730.50273
```

###Variable Importance The classifier used in the Random Forest model also shows using the variable importance command which classifiers affected the overall model the most. What it shows is that all of the variables with the exception of prep_course displayed varying levels of importance to the math score outcome. Of note the students who could afford school lunch at no reduced/free cost also out performed other students suggesting that family income may also play a large part in the total outcome of good test scores.

#Summary This analysis shows that through 8 variables test scores can be predicted based on factors of a students life outside of the classroom. Community leaders now have the ability to look for way to increase/change their localities to help target areas of improvement that will increase the value of everyone. Ultimately, and increase in education for everyone helps every community.

#References As stated above the data retrieved for this data set was provided by Dr. Royce Kimmons by using his website for data generation. No license or reference is directly required as stated by him, however for greater transparency and future analysis it can be found here: http://roycekimmons.com/tools/generated_data/exams