Statistical Learning Final Report

Alberto Calabrese, Eleonora Mesaglio, Greta d'Amore Grelli

2024-05-20

Contents

Introduction

This project understands how the student's performance (test scores) is affected by other variables such as:

- Gender
- Ethnicity
- Parental level of education
- Lunch
- Test preparation course.

GOAL:

To understand the influence of the parent's background, test preparation etc on students' performance.

We use the mean score as the target variable we want to predict or we can do the mean between the 3 scores and use it as performance indicator

Libraries

We will use only the corrplot library to visualize the correlation matrix.

```
library(corrplot)
```

Data

The data set we are going to use is the “Students Performance in Exams” dataset from Kaggle.

It contains 1000 rows and 8 columns.

```
students <- read.csv("Data/study_performance.csv", sep = ",", header = TRUE)

# Overview of the data
summary(students)
```

```
##     gender          race_ethnicity     parental_level_of_education
##  Length:1000        Length:1000        Length:1000
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##     lunch           test_preparation_course   math_score     reading_score
##  Length:1000        Length:1000             Min.   :  0.00   Min.   : 17.00
##  Class :character   Class :character        1st Qu.: 57.00   1st Qu.: 59.00
##  Mode  :character   Mode  :character        Median : 66.00   Median : 70.00
##                                             Mean   : 66.09   Mean   : 69.17
##                                             3rd Qu.: 77.00   3rd Qu.: 79.00
##                                             Max.   :100.00   Max.   :100.00
##  writing_score
##  Min.   : 10.00
##  1st Qu.: 57.75
##  Median : 69.00
##  Mean   : 68.05
##  3rd Qu.: 79.00
##  Max.   :100.00
```

```
dim(students)
```

```
## [1] 1000    8
```

We created a copy of the original dataset to work on it.

```
stud <- students
```

Data Transformation

We will transform the categorical variables into factors by using the `as.factor()` function.

```
stud$gender <- as.factor(stud$gender)
table(stud$gender)
```

```
##
## female   male
##    518    482
```

```
stud$race_ethnicity <- as.factor(stud$race_ethnicity)
table(stud$race_ethnicity)
```

```
## 
## group A group B group C group D group E 
##      89     190     319     262     140
```

```r
stud$parental_level_of_education <- as.factor(stud$parental_level_of_education)
table(stud$parental_level_of_education)
```

```
## 
## associate's degree  bachelor's degree        high school    master's degree 
##                222                118                196                 59 
##       some college   some high school 
##                226                179
```

```r
stud$lunch <- as.factor(stud$lunch)
table(stud$lunch)
```

```
## 
## free/reduced     standard 
##          355          645
```

```r
stud$test_preparation_course <- as.factor(stud$test_preparation_course)
table(stud$test_preparation_course)
```

```
## 
## completed      none 
##       358       642
```

Creating dummy variables

We will create dummy variables for the categorical variables.

First of all, we create the dummy variables for each categorical variable.

Then, we combine all the dummy variables into one new data frame that we called `stud__just_dummy`, which contains only dummy variables.

Finally, we create a new copy of the dataset `stud` called `stud_dummy`, in which we have the original numeric variable and we replace the original one with the new variable dummy.

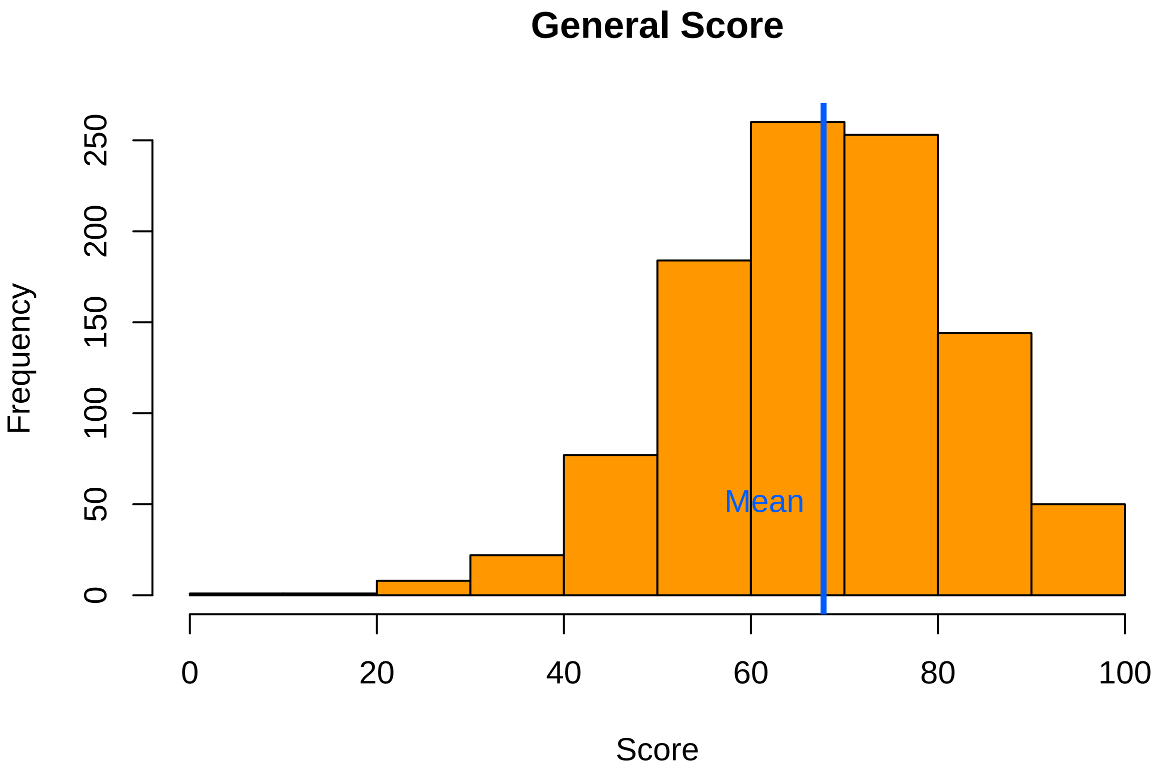
```r
gender_dummy <- model.matrix(~ gender - 1, data = stud)
race_dummy <- model.matrix(~ race_ethnicity - 1, data = stud)
education_dummy <- model.matrix(~ parental_level_of_education - 1, data = stud)
lunch_dummy <- model.matrix(~ lunch - 1, data = stud)
test_prep_dummy <- model.matrix(~ test_preparation_course - 1, data = stud)

stud__just_dummy <- cbind(gender_dummy, race_dummy, education_dummy,
                          lunch_dummy, test_prep_dummy)

stud_dummy <- stud

stud_dummy <- subset(stud_dummy, select = -c(gender, race_ethnicity,
                                             parental_level_of_education, lunch,
                                             test_preparation_course))

stud_dummy <- cbind(stud_dummy, gender_dummy, race_dummy, education_dummy,
                    lunch_dummy, test_prep_dummy)
```

New variable general score

We create a new variable called `general_score` which is the mean of the three scores.

This will be our target variable.

```
scores <- c("math_score", "reading_score", "writing_score")
stud$general_score <- (stud$math_score + stud$reading_score + stud$writing_score) /
  length(scores)

y <- stud$general_score

# Plot the histogram of the general score
hist(y, main = "General Score", xlab = "Score", col = "#ff9800")
abline(v = mean(y), col = "#005cff", lwd = 3)
text(mean(y), 50, "Mean", col = "#005cff", pos = 2)
```



Most of the students get a general score between 60 and 80 out of 100

New binary variable pass/fail

We create a new binary variable called `pass_exam` which is 1 if the general score is greater than 60 and 0 otherwise.

```
stud$pass_exam <- ifelse(stud$general_score > 60, 1, 0)
table(stud$pass_exam)
```

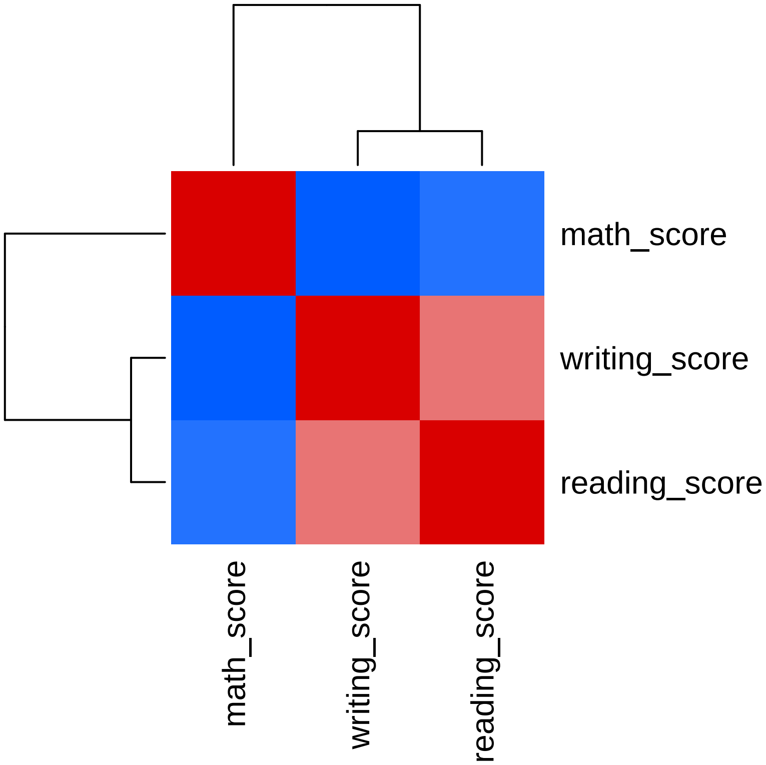
```
##
##   0   1
## 293 707
```

Correlation Analysis

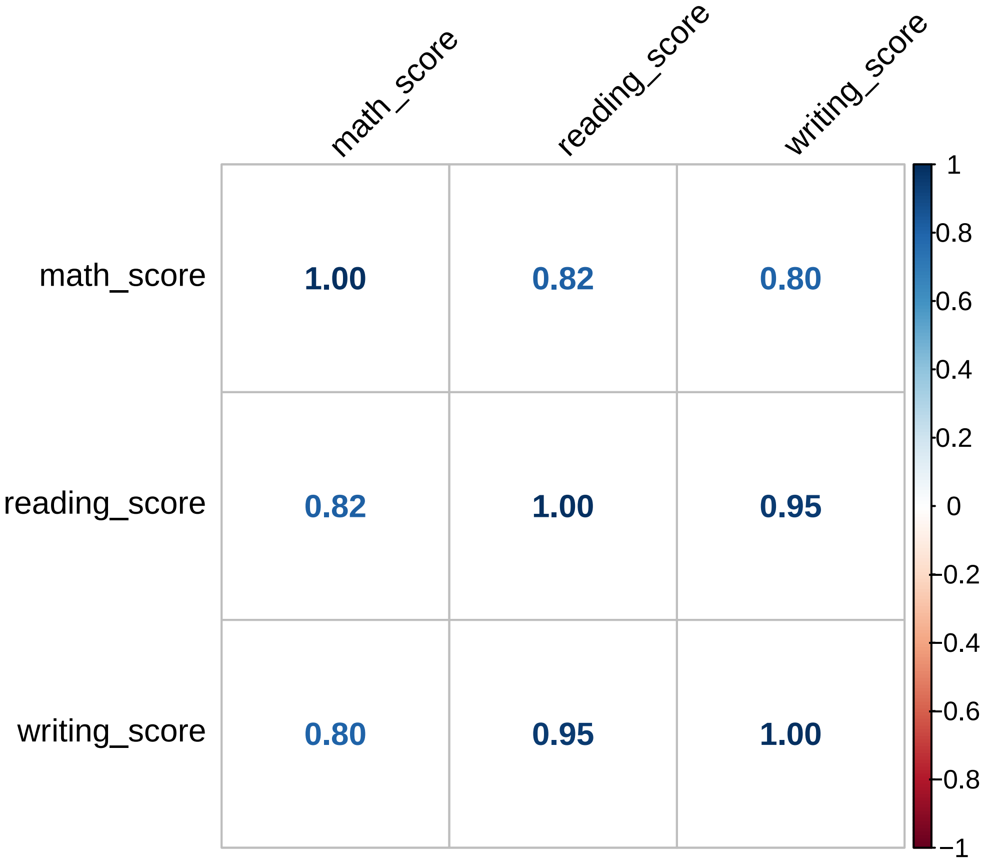
Correlations between numeric variables

Correlations Analysis between the numeric variable only Select only the numeric variables from the dataset stud Work now on the dataset stud_dummy

```
# Select only the numeric variables from the dataset stud
stud_numeric_original <- stud_dummy[, c("math_score", "reading_score", "writing_score")]

# Heatmap
# Displays the correlation matrix using a heatmap
heatmap(cor(stud_numeric_original),
        col = colorRampPalette(c("#005cff", "#fbfbfb", "#d90000"))(100),
        symm = TRUE,
        margins = c(10, 10),
        cexRow = 1.2,
        cexCol = 1.2)
```



```
# Displays the correlation matrix using a corrplot
corrplot(cor(stud_numeric_original), method = "number", tl.col = "black",
         tl.srt = 45, addCoef.col = "black")
```



We notice that there is a strong positive correlation between all the 3 score that means the increase of, an average, score also increase the other

Correlations between numeric and dummy variables

Correlations Analysis between the numeric and dummy variables

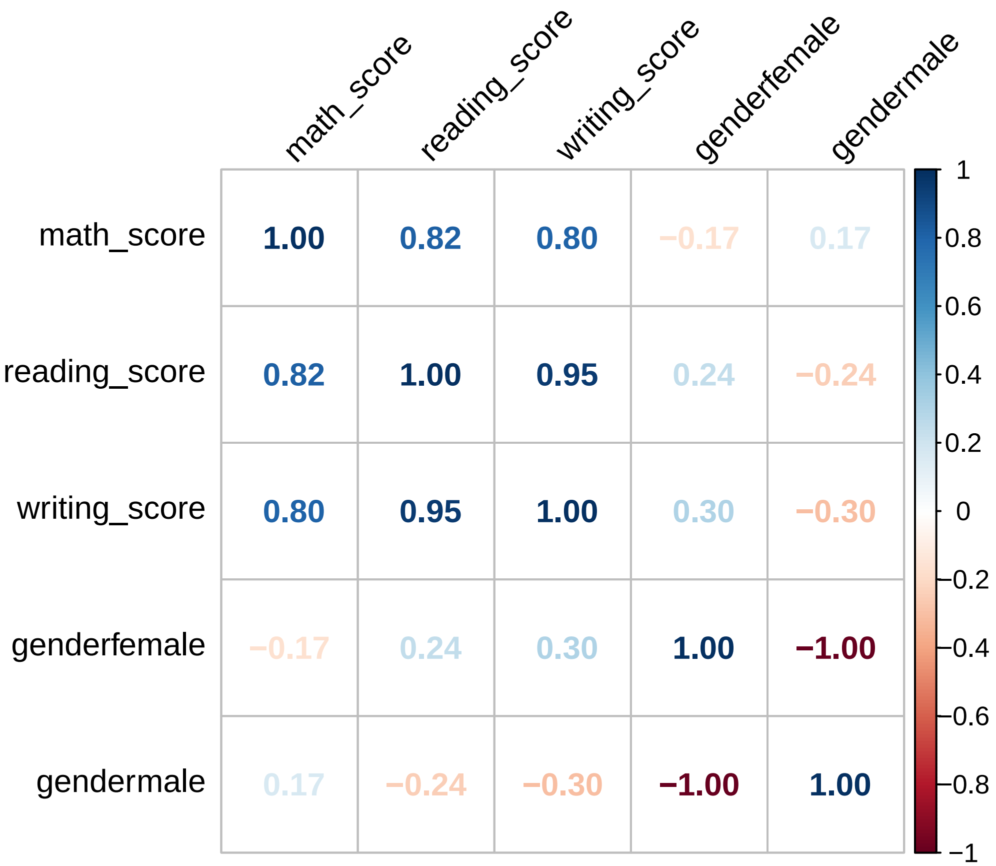
We want to see if and how scores changes within other variables.

We're looking for linear relationships

Gender correlation

Calculate the correlation matrix between the numeric variables and the dummy `gender` variable

```
corrplot(cor(cbind(stud_numeric_original, gender_dummy)), method = "number",
         tl.col = "black", tl.srt = 45, addCoef.col = "black")
```

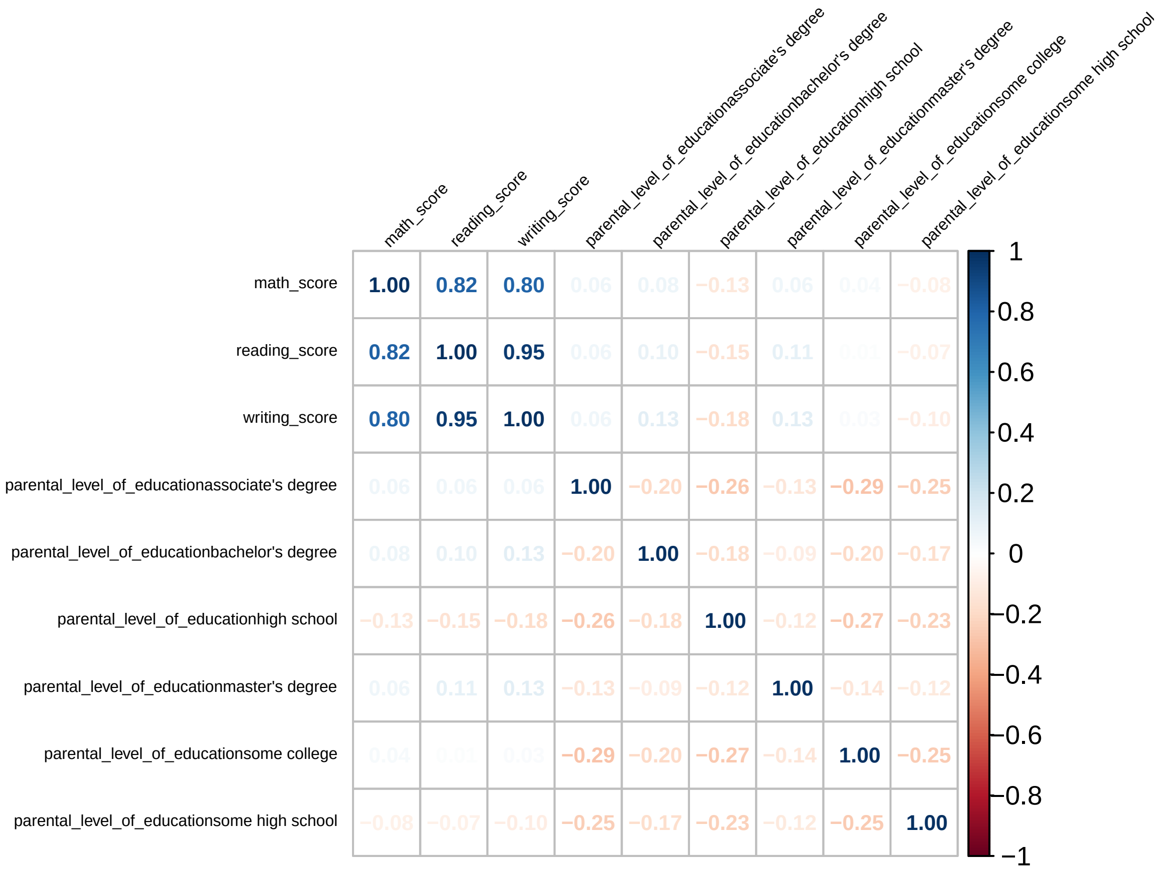


Negative correlation (-0.17) indicates that there is an inverse relationship between gender and math scores. That siggests us that there in general a trend for math score to be sliglty worse for female compared to male

Education correlation

Calculate the correlation matrix between the numeric variables and the dummy `education`

```
corrplot(cor(cbind(stud_numeric_original, education_dummy)), method = "number",
         tl.col = "black", tl.srt = 45,
         addCoef.col = "black", tl.cex = 0.5, number.cex = 0.7)
```

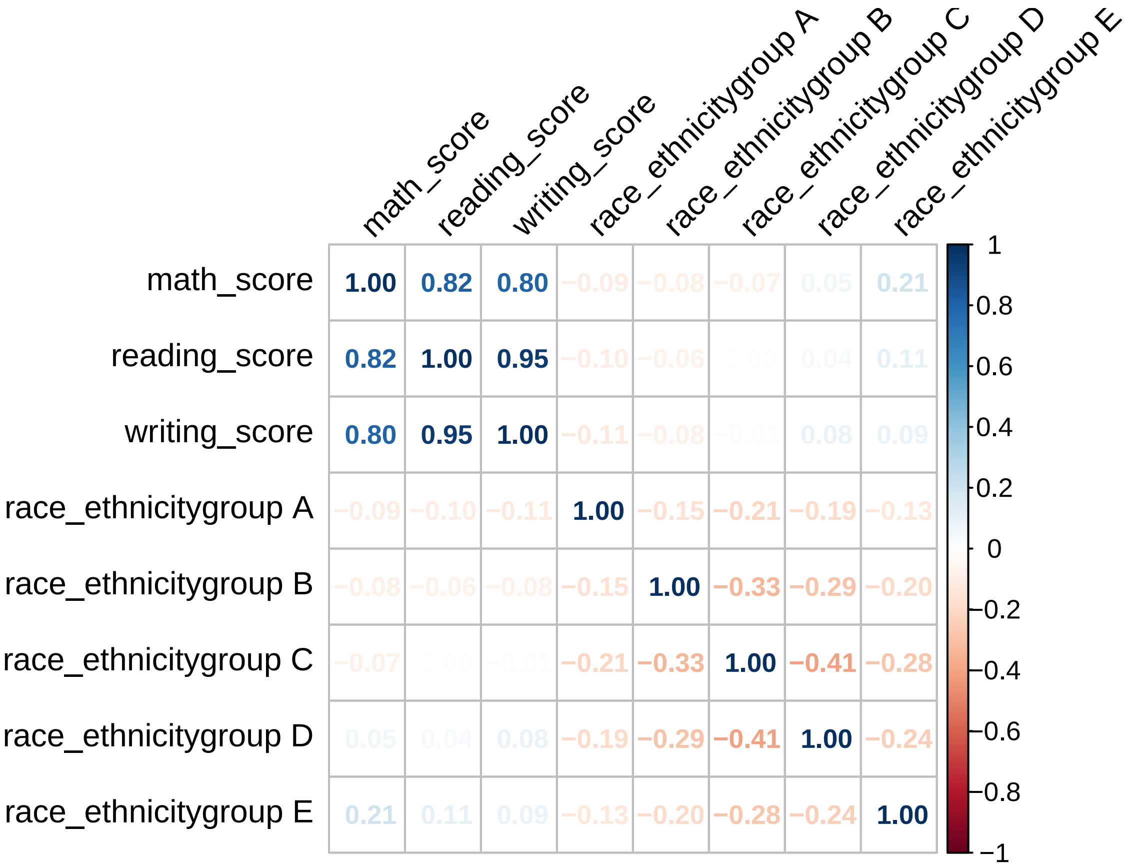


There is no significant correlations. That means that the level of education of the parents does not affect the scores of the students

Race correlation

Calculate the correlation matrix between the numeric variables and the dummy `race`

```
corrplot(cor(cbind(stud_numeric_original, race_dummy)), method = "number",
         tl.col = "black", tl.srt = 45, addCoef.col = "black", number.cex = 0.8)
```

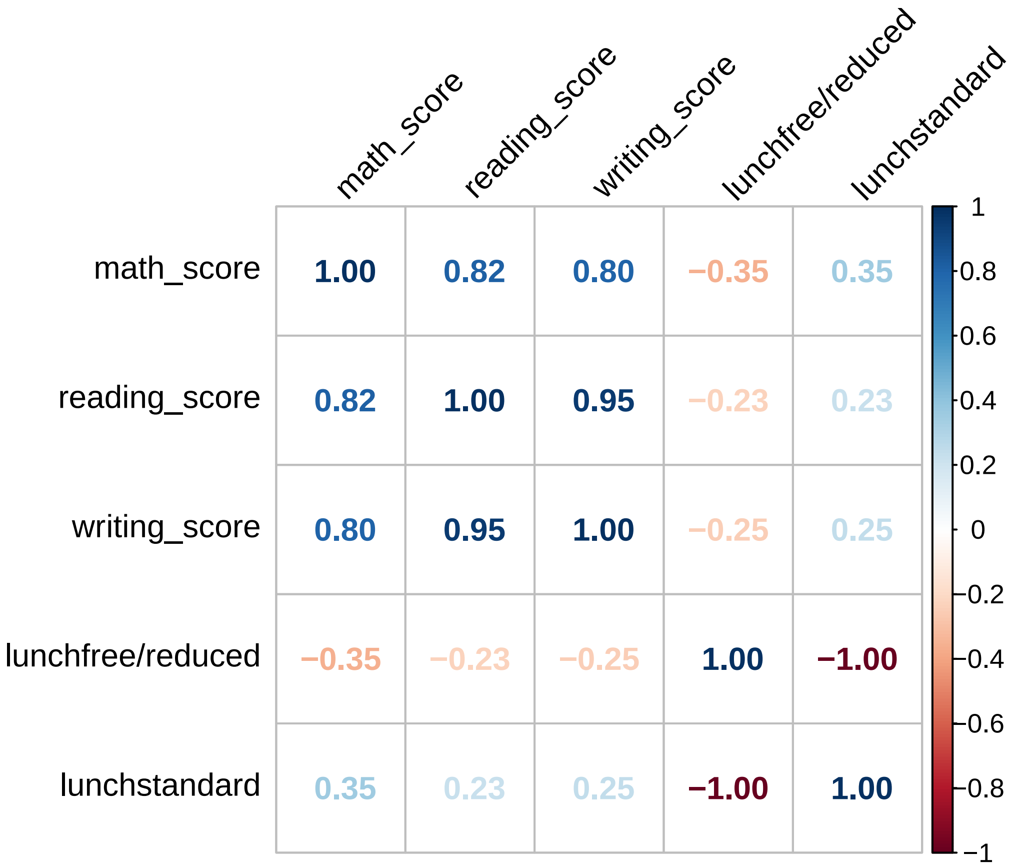


Positive correlation between math score and group E

Lounch correlation

Calculate the correlation matrix between the numeric variables and the dummy `lunch`

```
corrplot(cor(cbind(stud_numeric_original, lunch_dummy)), method = "number",
         tl.col = "black", tl.srt = 45, addCoef.col = "black")
```



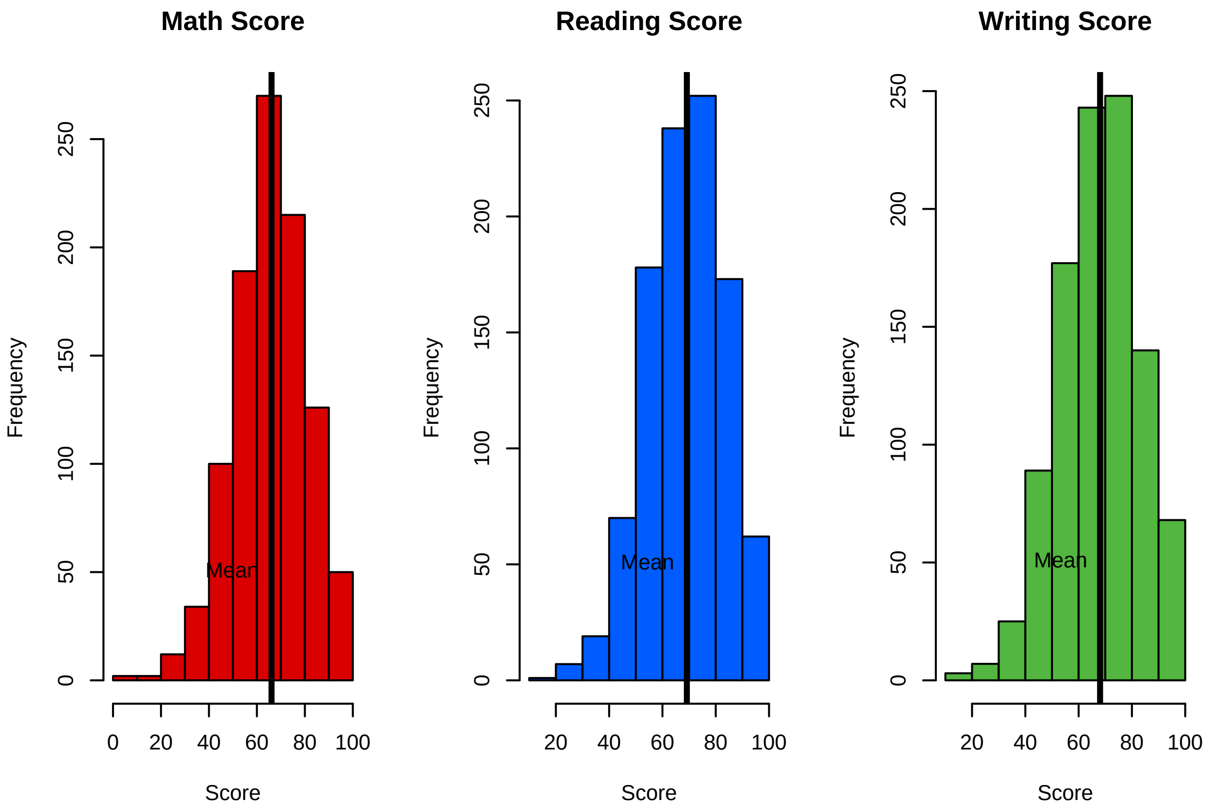
	math_score	reading_score	writing_score	lunchfree/reduced	lunchstandard
math_score	1.00	0.82	0.80	−0.35	0.35
reading_score	0.82	1.00	0.95	−0.23	0.23
writing_score	0.80	0.95	1.00	−0.25	0.25
lunchfree/reduced	−0.35	−0.23	−0.25	1.00	−1.00
lunchstandard	0.35	0.23	0.25	−1.00	1.00

In general perform slighlty better how has a standard meal

Data Visualization

Histograms

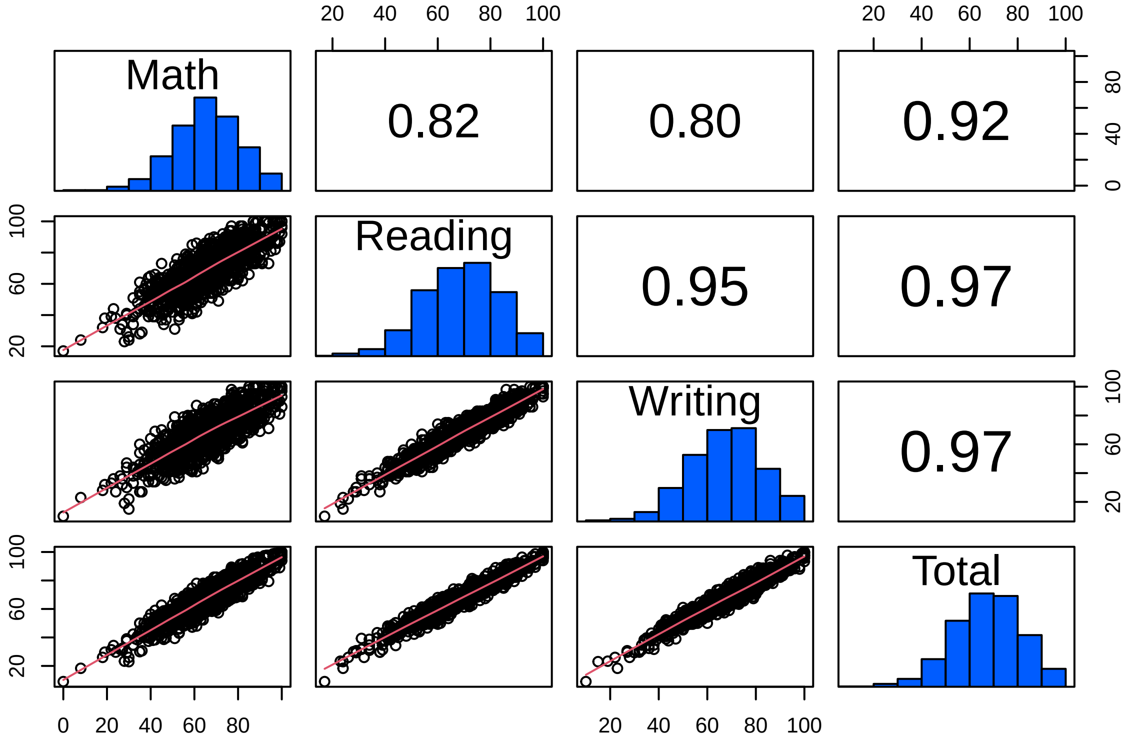
```r
par(mfrow=c(1,3))
hist(stud$math_score, main = "Math Score", xlab = "Score", col = "#d90000")
abline(v = mean(stud$math_score), col = "black", lwd = 3)
text(mean(stud$math_score), 50, "Mean", col = "black", pos = 2)
hist(stud$reading_score, main = "Reading Score", xlab = "Score", col = "#005cff")
abline(v = mean(stud$reading_score), col = "black", lwd = 3)
text(mean(stud$reading_score), 50, "Mean", col = "black", pos = 2)
hist(stud$writing_score, main = "Writing Score", xlab = "Score", col = "#52b640")
abline(v = mean(stud$writing_score), col = "black", lwd = 3)
text(mean(stud$writing_score), 50, "Mean", col = "black", pos = 2)
```



Scatterplot matrix

Scatterplot matrix for numerical variables

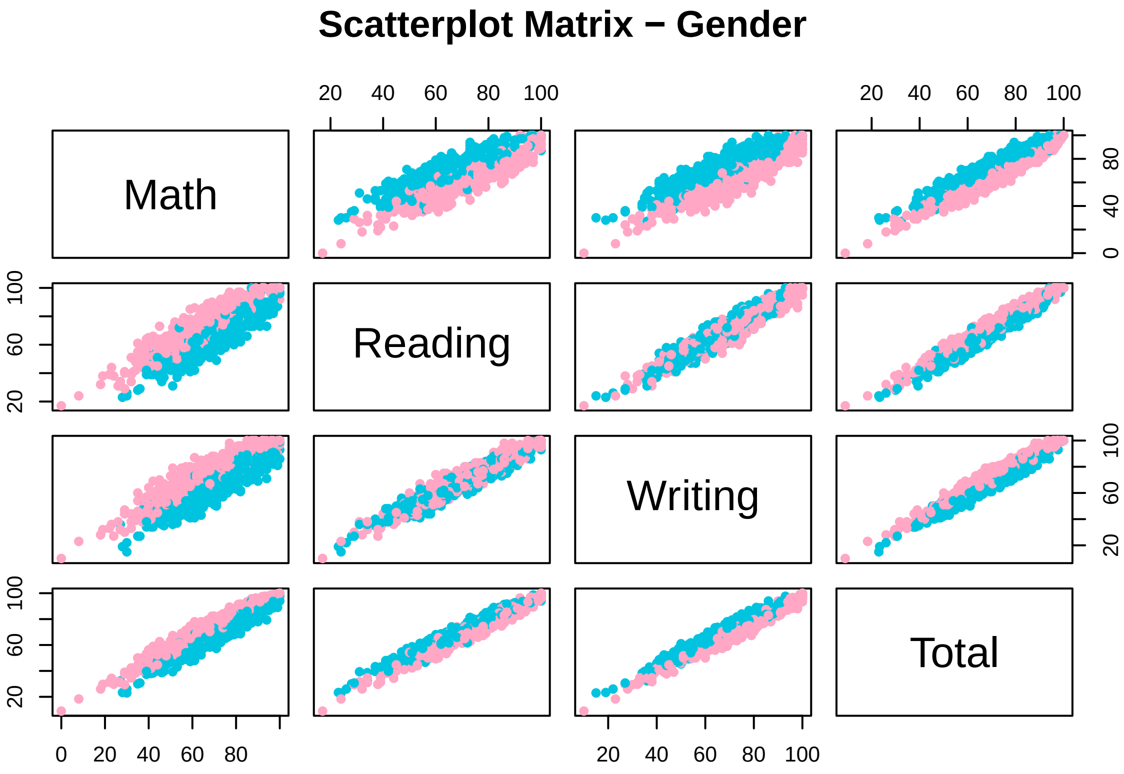
```r
# Selection numerical variables
numerical_vars <- c("math_score", "reading_score", "writing_score", "general_score")

pairs(stud[, numerical_vars],
      diag.panel = panel.hist, # Histograms on the diagonal
      upper.panel = panel.cor, # Correlations above the diagonal
      lower.panel = panel.smooth,# Regression below the diagonal
      labels = c("Math", "Reading", "Writing", "Total"))
```



From the plot we can confirm that there is high correlations between the scores, also with total scores since it results as the mean of other scores regression lines fit very well the cloud of point distributions approximatly normal

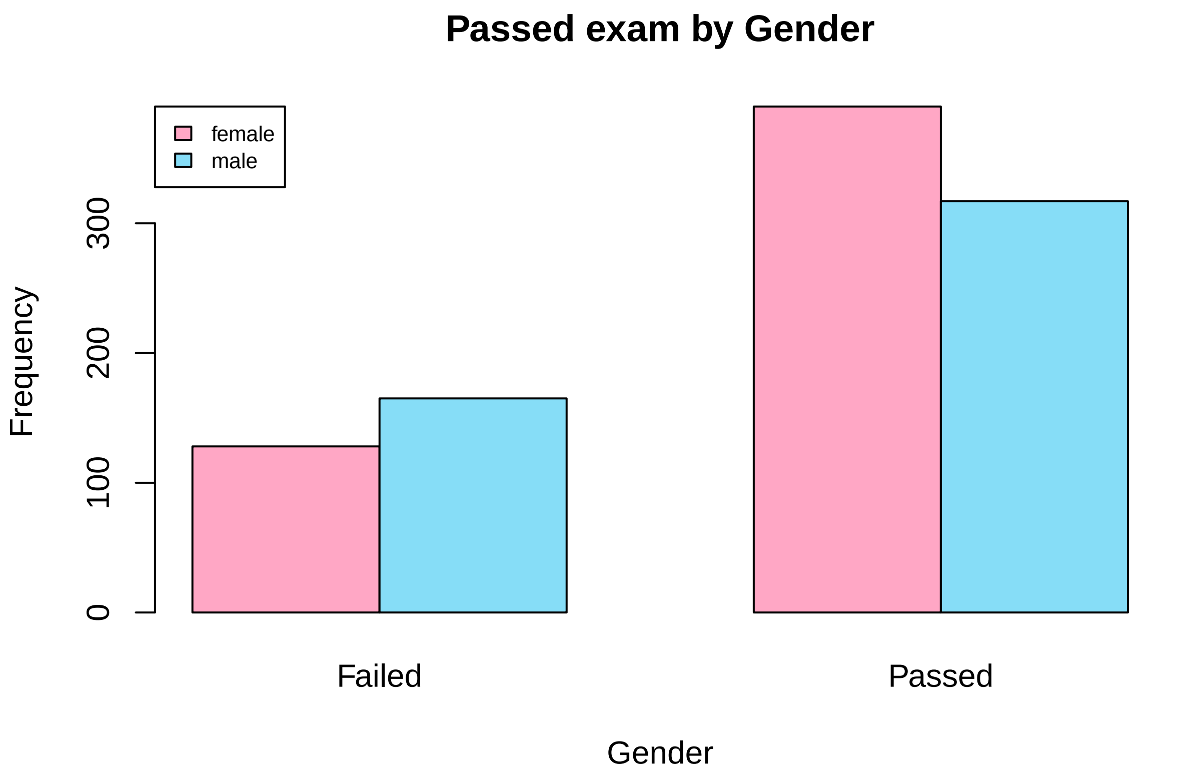
Scatterplot matrix for numerical variables by gender

```r
pairs(stud[, numerical_vars],
      col = ifelse(stud$gender == "female", "#ffa7c5", "#00c4df"),
      pch = 16,
      labels = c("Math", "Reading", "Writing", "Total"),
      main = "Scatterplot Matrix - Gender")
```

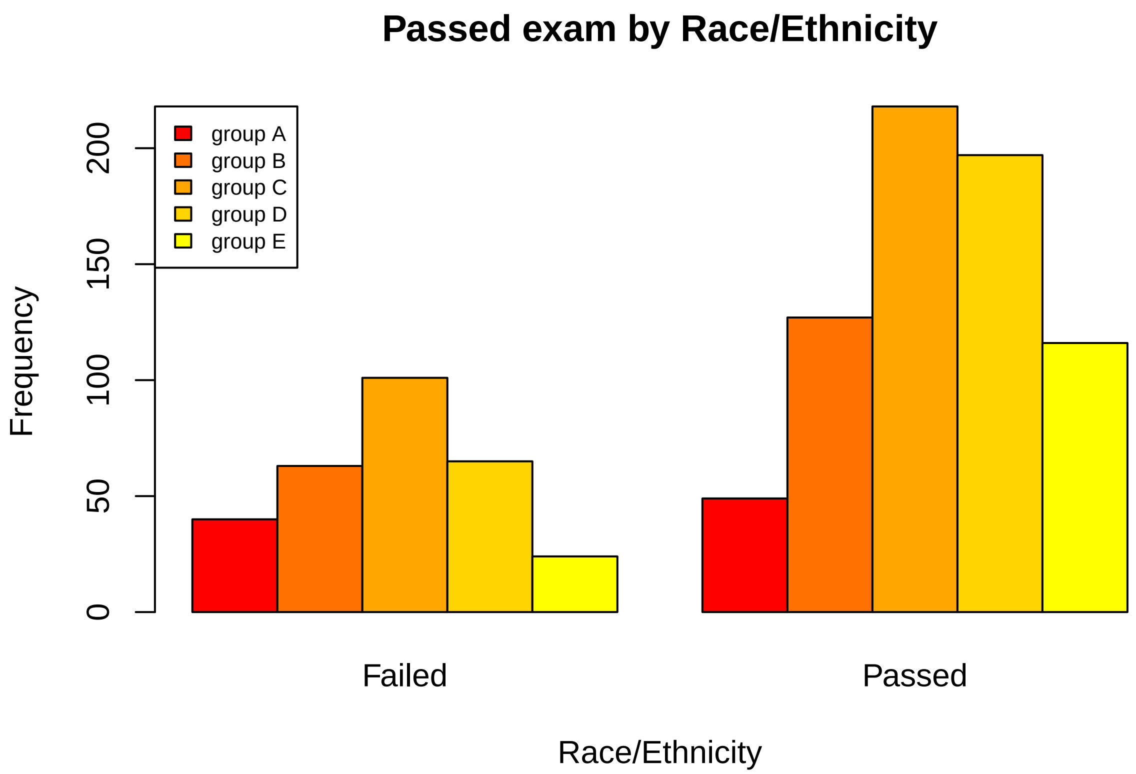


Barplot

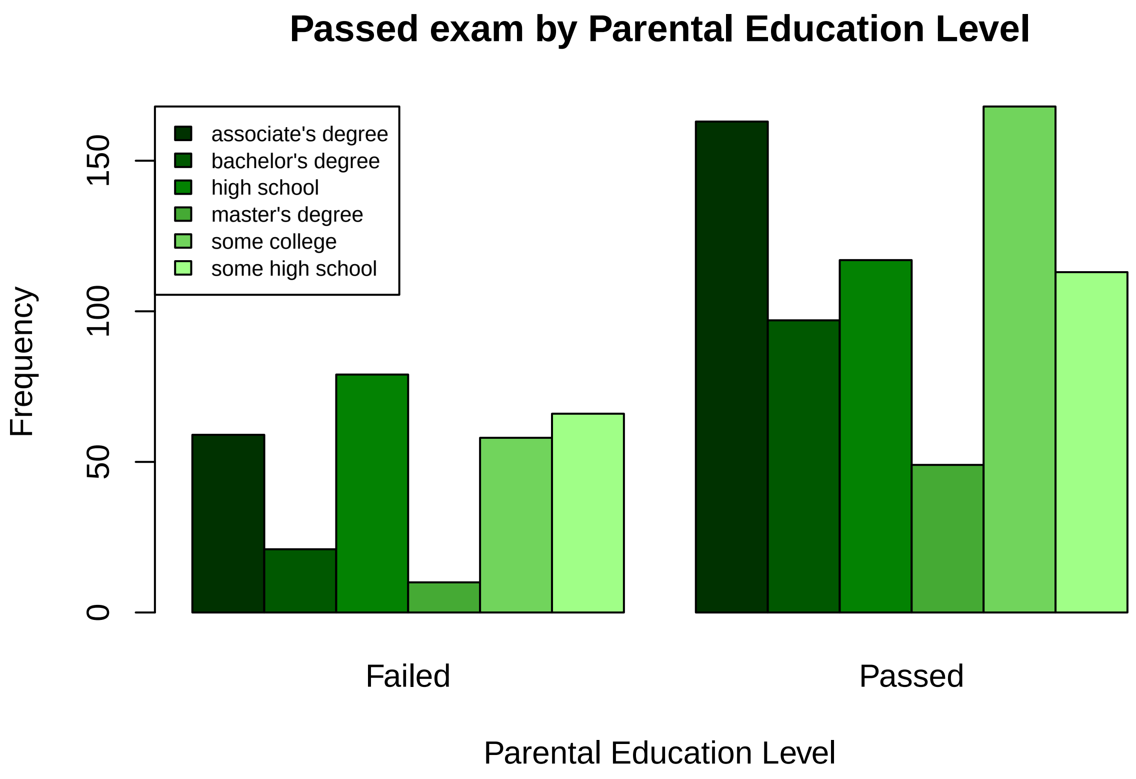
```
# par(mfrow = c(3, 2), mar = c(5, 5, 4, 2))

# Barplot of total score by gender
barplot(table(stud$gender, stud$pass_exam),
        main = "Passed exam by Gender",
        xlab = "Gender", ylab = "Frequency", col = c("#ffa7c5", "#86ddf7"),
        legend = rownames(table(stud$gender, stud$pass_exam)),
        beside = TRUE, axisnames = TRUE, args.legend = list(x = "topleft", cex = 0.7),
        names.arg = c("Failed", "Passed"))
```



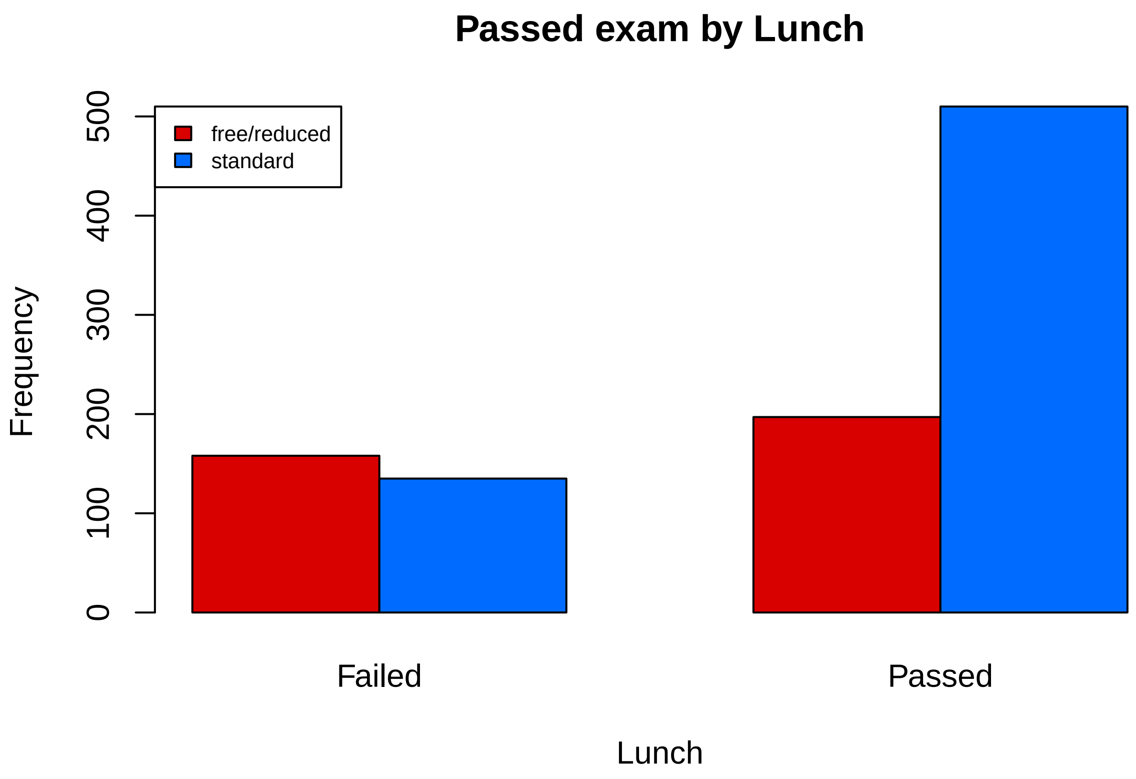
```
# Barplot of total score by race
barplot(table(stud$race_ethnicity, stud$pass_exam),
        main = "Passed exam by Race/Ethnicity",
        xlab = "Race/Ethnicity", ylab = "Frequency",
        col = c('#ff0000', '#ff7100', '#ffa600', '#ffd400', '#ffff00'),
        legend = rownames(table(stud$race_ethnicity, stud$pass_exam)),
        beside = TRUE, axisnames = TRUE, args.legend = list(x = "topleft", cex = 0.7),
        names.arg = c("Failed", "Passed"))
```



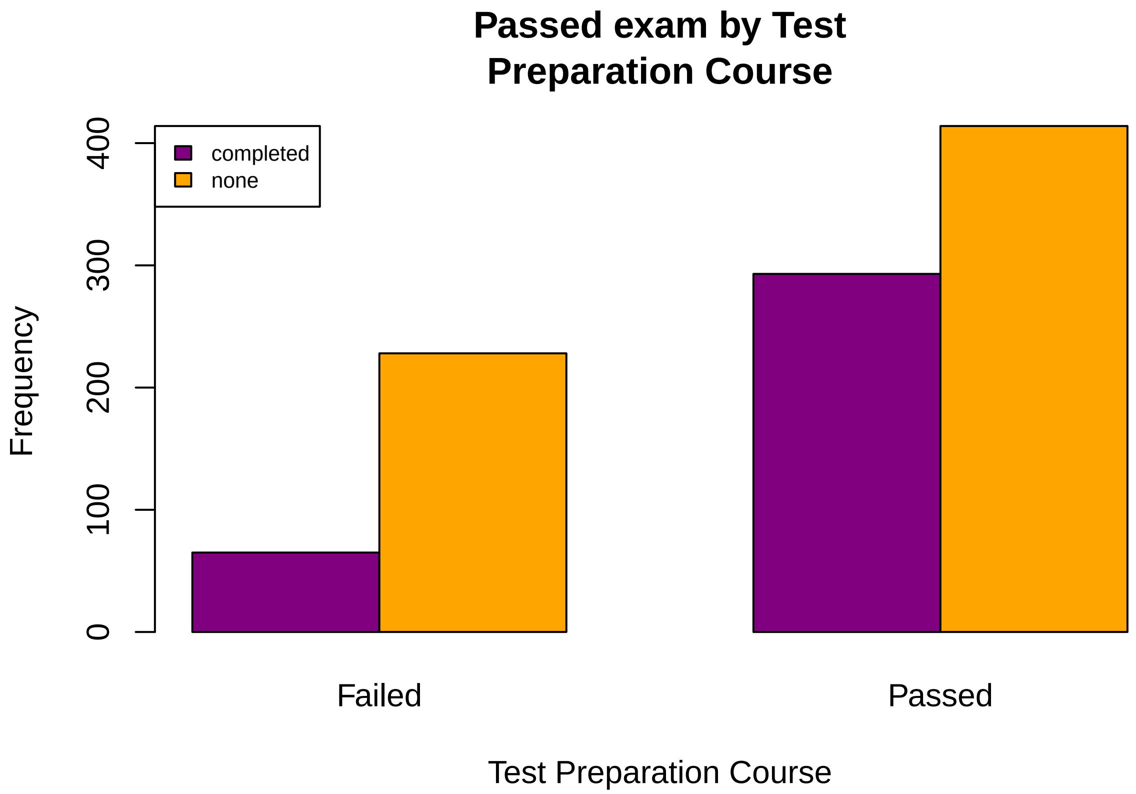
```r
# Barplot of total score by parents level of education
barplot(table(stud$parental_level_of_education, stud$pass_exam),
        main = "Passed exam by Parental Education Level",
        xlab = "Parental Education Level", ylab = "Frequency",
        col = c('#003200', '#005800', '#038202', '#45aa34', '#71d45c', '#a0ff87'),
        legend = rownames(table(stud$parental_level_of_education, stud$pass_exam)),
        beside = TRUE, axisnames = TRUE, args.legend = list(x = "topleft", cex = 0.7),
        names.arg = c("Failed", "Passed"))
```



```
# Barplot of total score by lunch
barplot(table(stud$lunch, stud$pass_exam),
        main = "Passed exam by Lunch",
        xlab = "Lunch", ylab = "Frequency",
        col = c('#d90000', '#006cff'),
        legend = rownames(table(stud$lunch, stud$pass_exam)),
        beside = TRUE, axisnames = TRUE, args.legend = list(x = "topleft", cex = 0.7),
        names.arg = c("Failed", "Passed"))
```



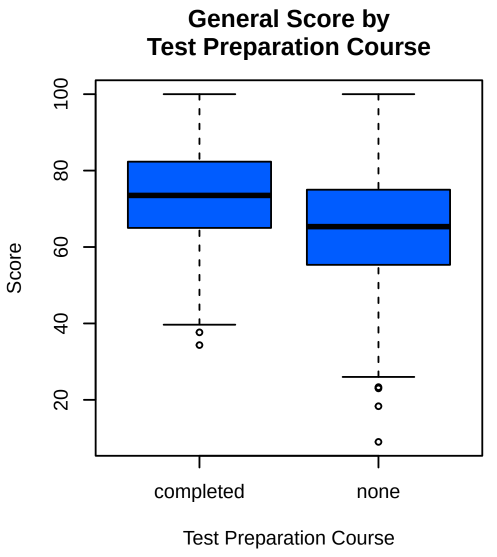
```
# Barplot of total score by pass preparation in the course
barplot(table(stud$test_preparation_course, stud$pass_exam),
        main = "Passed exam by Test\nPreparation Course",
        xlab = "Test Preparation Course", ylab = "Frequency",
        legend = rownames(table(stud$test_preparation_course, stud$pass_exam)),
        beside = TRUE, col = c('#800080', '#ffa500'), axisnames = TRUE,
        args.legend = list(x = "topleft", cex = 0.7),
        names.arg = c("Failed", "Passed"))
```

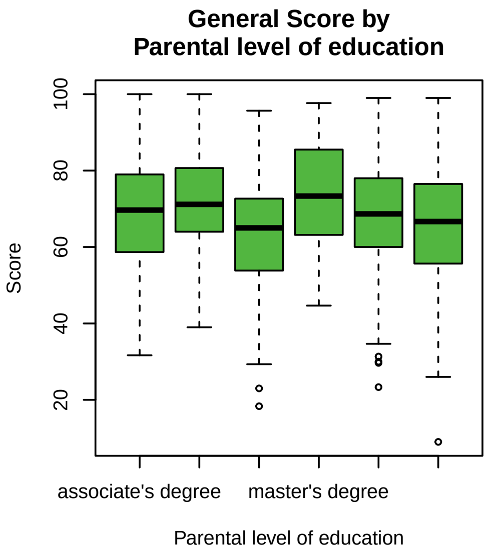


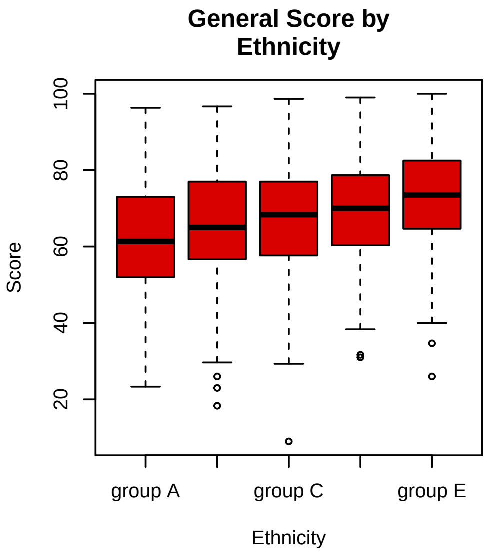
Boxplots

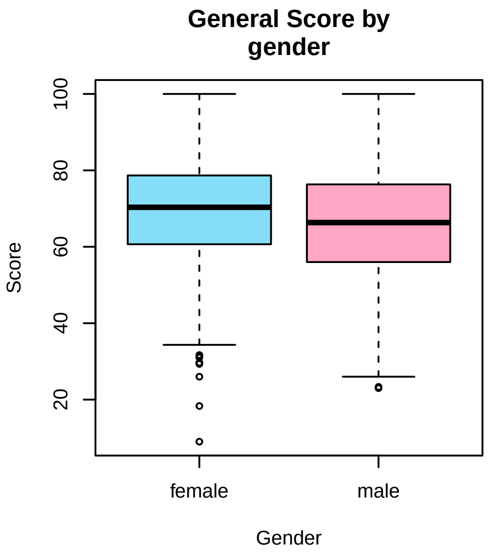
Boxplot of general score

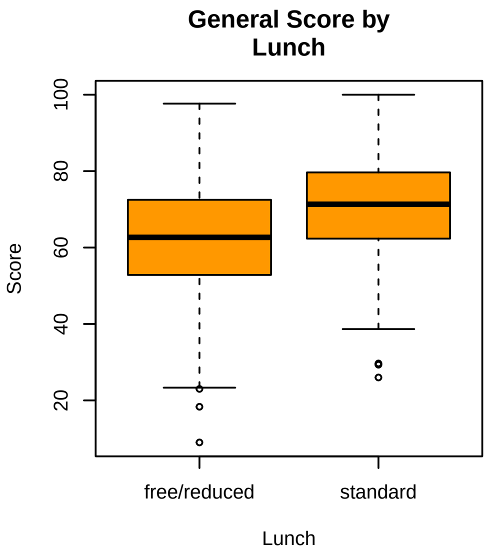
```
# Boxplot of the general score with other variables
par(mfrow=c(3,2))
boxplot(stud$general_score ~ stud$test_preparation_course,
        main = "General Score by\nTest Preparation Course",
        xlab = "Test Preparation Course", ylab = "Score",
        col = "#005cff")
boxplot(stud$general_score ~ stud$parental_level_of_education,
        main = "General Score by\nParental level of education",
        xlab = "Parental level of education", ylab = "Score",
        col = "#52b640")
boxplot(stud$general_score ~ stud$race_ethnicity,
        main = "General Score by\nEthnicity",
        xlab = "Ethnicity", ylab = "Score",
        col = "#d90000")
boxplot(stud$general_score ~ stud$gender,
        main = "General Score by\ngender",
        xlab = "Gender", ylab = "Score",
        col = c("#86ddf7", "#ffa7c5"))
boxplot(stud$general_score ~ stud$lunch,
        main = "General Score by\nLunch",
        xlab = "Lunch", ylab = "Score",
        col = "#ff9800")
boxplot(stud$general_score ~ stud$pass_exam,
        main = "General Score by\nPass_exam",
        xlab = "Passed Exam", ylab = "Score",
        col = "#830783")
```

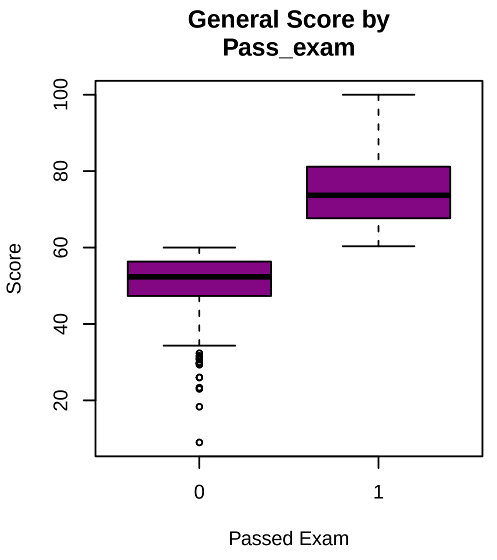

General Score by
Test Preparation Course
Score
100
80
60
40
20
completed
none
Test Preparation Course


General Score by
Parental level of education
Score
100
80
60
40
20
associate's degree
master's degree
Parental level of education


General Score by
Ethnicity
Score
100
80
60
40
20
group A
group C
group E
Ethnicity

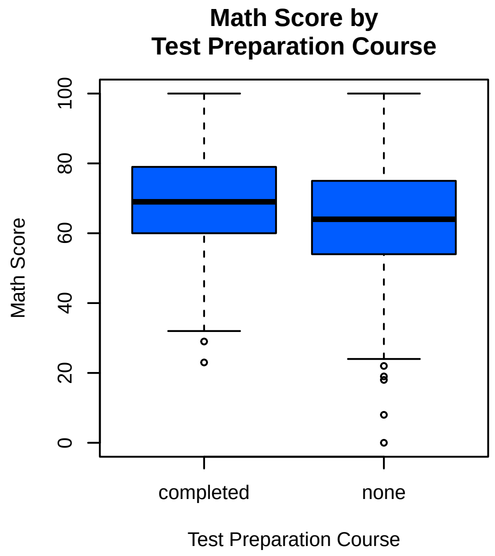

General Score by
gender
Score
100
80
60
40
20
female
male
Gender


General Score by
Lunch
Score
100
80
60
40
20
free/reduced
standard
Lunch

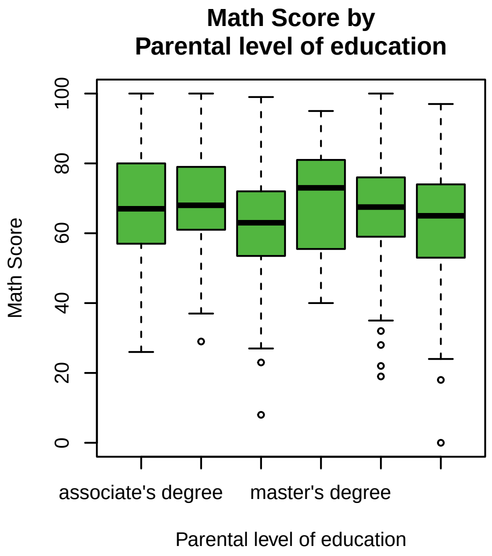

General Score by
Pass_exam
Score
100
80
60
40
20
0
1
Passed Exam

Boxplot of math score

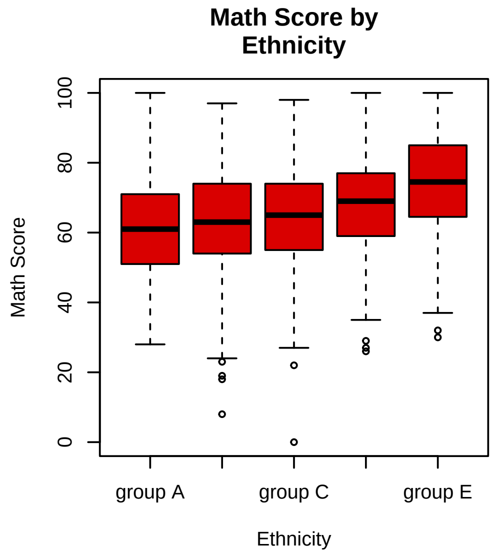
```
# Boxplot of the math score with other variables
par(mfrow=c(3,2))
boxplot(stud$math_score ~ stud$test_preparation_course,
        main = "Math Score by\nTest Preparation Course",
        xlab = "Test Preparation Course", ylab = "Math Score",
        col = "#005cff")
boxplot(stud$math_score ~ stud$parental_level_of_education,
        main = "Math Score by\nParental level of education",
        xlab = "Parental level of education", ylab = "Math Score",
        col = "#52b640")
boxplot(stud$math_score ~ stud$race_ethnicity,
        main = "Math Score by\nEthnicity",
        xlab = "Ethnicity", ylab = "Math Score",
        col = "#d90000")
boxplot(stud$math_score ~ stud$gender,
        main = "Math Score by\nGender",
        xlab = "Gender", ylab = "Math Score",
        col = c("#86ddf7", "#ffa7c5"))
boxplot(stud$math_score ~ stud$lunch,
        main = "Math Score by\nLunch",
        xlab = "Lunch", ylab = "Math Score",
        col = "#ff9800")
```



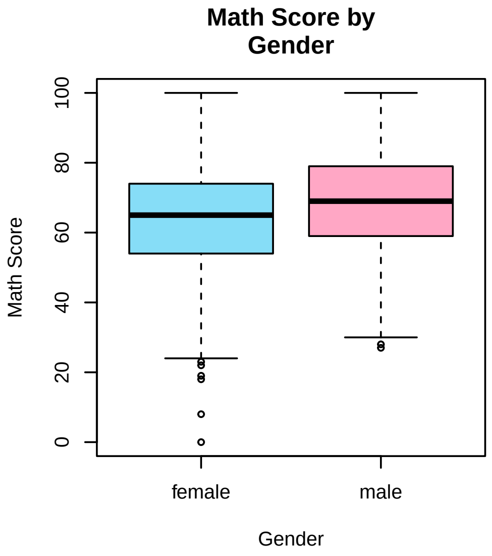
Math Score by
Test Preparation Course



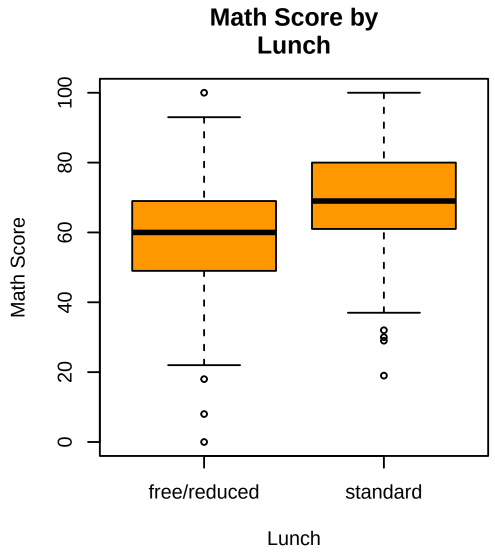
Math Score by
Parental level of education



Math Score by
Ethnicity



Math Score by
Gender



Math Score by
Lunch