STA 141A Final Project
Group 35

<u>Group Leader:</u>

Caitlyn Maung (clmaung@ucdavis.edu)

<u>Group Members</u>:

Kyle Fong (khfong@ucdavis.edu)

Anduo Wang (duowang@ucdavis.edu)

Mandy Yu(guiyu@ucdavis.edu)

Contribution: Research question #1 (Caitlyn, Anduo), Research question #2 (Caitlyn, Kyle), Research question #3 (Mandy)

Introduction:

For this project, we will be using the Student Performance dataset from the UC Irvine Machine Learning Repository. The dataset provides students' scores in secondary education (grades 7-12) of two Portuguese schools. The variables of the data include student grades as well as demographic, social, and school-related features, such as school, parents' education, parents' occupation, extracurricular activities, study time, etc. The data was collected using school reports and questionnaires in Portuguese schools. Out of the two datasets provided on students' performance in the subjects of Math and Portuguese, we will be using the dataset on students' math scores. Our goal is to determine which variables are most influential towards a student's performance in their math class. Then, we create a model that can take in future data and generate predictions on what performance that student is likely to have in the class.

Data Base:

This dataset has 33 variables with categorical, numerical, and binary variables. Before we apply the model to the data, here's a small explanation of each variable.

Categorical variables:

- <u>Mjob</u>: student's mother's job ("teacher" for teacher, "health" for health care related, "services" for civil services, "at_home" for stayed at home, "other" for all other jobs)

- Fjob: student's father's job ("teacher" for teacher, "health" for health care related, "services" for civil services, "at_home" for stayed at home, "other" for all other jobs)
- reason: the reason why the student attends this school ("home" for close to home, "reputation" for school reputation, "course" for school course preference, "other" for all other reasons)
- guardian: student's guardian ("mother" for mother, "father" for father, "other" for all other people)

Numerical variables:

- age: student's age (from 15 to 22)
- Medu: student's mother's education level (0 = none, 1 = primary education, 2 = 5th to 9th grade, 3 = secondary education, 4 = higher education)
- Fedu: student's father's education level (0 = none, 1 = primary education, 2 = 5th to 9th grade, 3 = secondary education, 4 = higher education)
- traveltime: commute time from home to school (1 = less than 15 mins, 2 = 15 to 30 mins, 3 = 30 mins to 1 hour, 4 = longer than 1 hour)
- studytime: weekly study time (1 = less than 2 hours, 2 = 2 to 5 hours, 3 = 5 to 10 hours, 4 = greater than 10 hours)
- failures: the number of failed classes in the past (from 1 to 3, anything greater than 3 is 4)
- famrel: the quality of family relationships (ranking from 1 to 5, very bad to excellent)
- freetime: student's freetime after school (ranking from 1 to 5, very low to very high)
- goout: student's time going out with friend (ranking from 1 to 5, very low to very high)
- Dalc: student's workday alcohol consumption (ranking from 1 to 5, very low to very high)
- Walc: student's weekend alcohol consumption (ranking from 1 to 5, very low to very high)
- health: student's current health status (ranking from 1 to 5, very bad to very good)
- absences: number of absences students have (from 0 to 93)
- G1: first marking period grade (form 0 to 20)
- G2: second marking period grade (from 0 to 20)
- G3: final grade (from 0 to 20)

Binary variables:

- school: student's school ("GP" for Gabriel Pereira, "MS" for Moushinho da Silveria)
- sex: student's gender ("F" for female, "M" for male)
- address: student's address type ("U" for urban, "R" for rural)
- famsize: student's family size ("LE3" for less or equal to 3, "GT3" for greater than 3)
- Pstatus: student's parents' cohabitation status ("T" for together, "A" for apart)
- schoolsup: if the student receives extra educational support ("yes" or "no")
- famsup: whether the student receives family educational support ("yes" or "no")
- Paid: extra paid classes within the course subject ("yes" or "no")
- activities: if the student doing extracurricular activities ("yes" or "no")

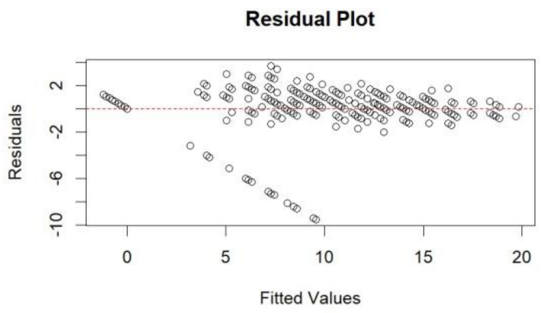
- nursery: if the student attended nursery school (“yes” or “no”)
- higher: if the student wants to take higher education (“yes” or “no”)
- internet: if the student has access to home internet (“yes” or “no”)
- romantic: if the student has a romantic relationship (“yes” or “no”)

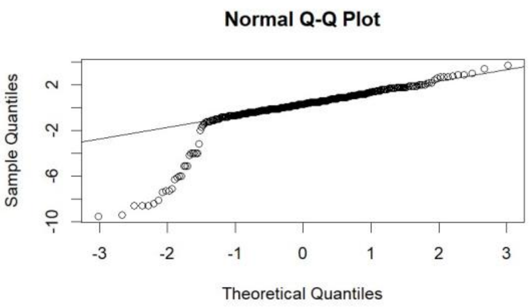
Key Questions:

- How do the first and second-period grades affect the final-period grades?
- What factors have the greatest influence on the grades that students achieve in secondary education?
- How do parents’ education level and occupation affect student performance?

Research Question #1

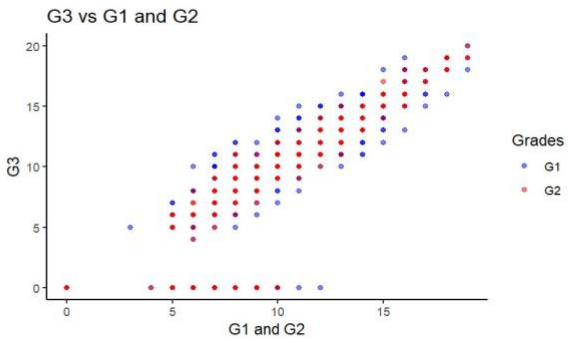
Our first step in exploring our data was analyzing the relationship between first-period grades (G1) and second-period grades (G2) with final-period grades (G3). We first build our model1 with G1 and G2 as the explanatory variables and G3 as the response variable. According to the summary statistics of the model, we see that the adjusted R-squared is 0.8213 and the p-value is less than $2.2 * 10^{-16}$, which indicates that the predictors G1 and G2 are statistically significant. We then plotted a residual plot and found that the errors have non-constant variance as there are several outliers that significantly deviate from y = 0. Therefore, this violates the assumption of constant variance for a linear regression model.

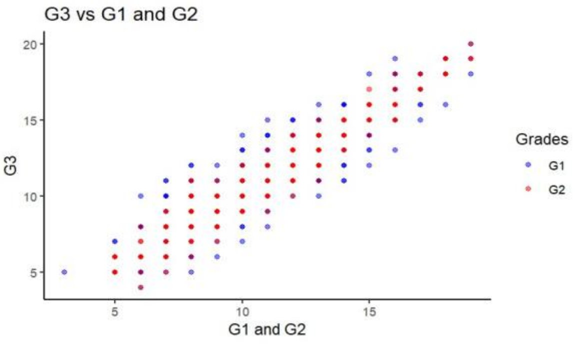




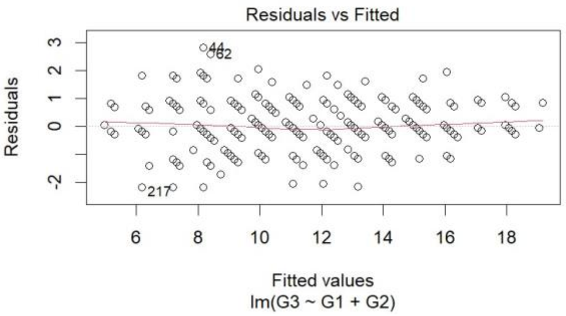
We also assessed the plot of the normality of the errors and found that the data is left-skewed and therefore violates the assumption of normality of errors. To identify the cause of the non-normality of errors, we decided to create a scatterplot of G1 and G2 vs. G3, and we found a strong correlation between G1, G2, and G3. The G1 grades are represented by blue dots and the G2 grades are represented by the red dots. We observed that G2 seems to have a more constant positive linear relationship with G3, compared to G1. However, there are several outliers where G3 = 0. We looked into the dataset and realized that many students who got 0 on G2 continued to get 0 on G3, and before that, they had relatively good grades for G1. We have a high suspicion that this is because the students are no longer in the school/system, therefore it is meaningless to include those data points.

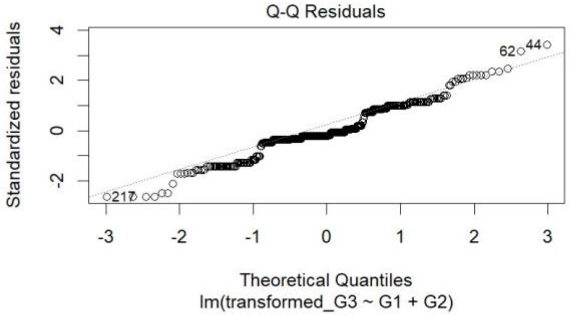
As a result, we decided to remove these points from the data. After removing those points, we can find a more definitive positive linear trend in the data. According to the summary statistics, the new model after removing outliers indicates an adjusted R-squared of 0.9343 (higher than the adjusted R-squared of the original model). It also results in a p-value that remains less than 2.2 * 10^{-16}.





The residual plot of the new model seems to show that the assumption of constant variance is now met. Although the normality of the errors seems to appear jagged, the points are still close to the line. In addition, we attempted to do a Box-Cox transformation but it didn't change the normality at all.



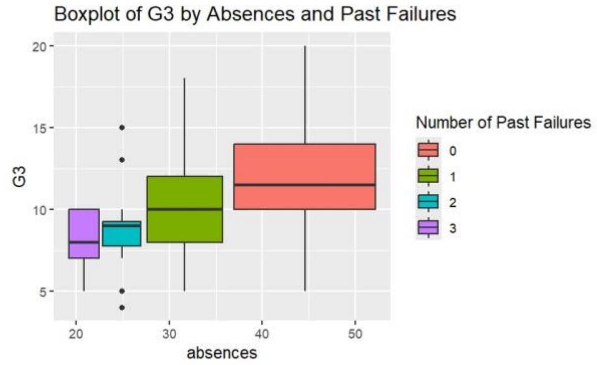


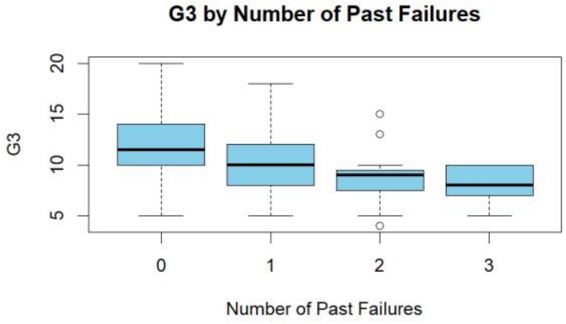
.

Thus, we conclude that G1 and G2 have a possible effect on G3, people who get high scores in the first and second period tend to have higher final grades as well.

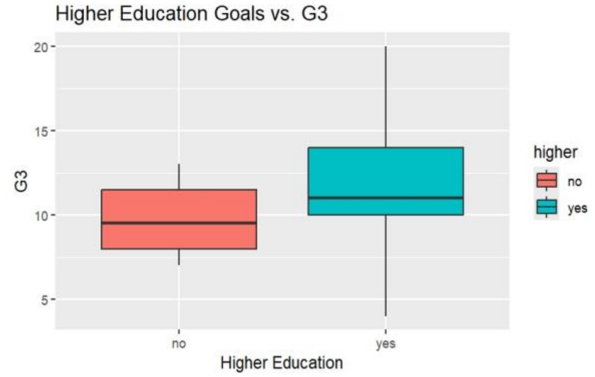
Research Question #2

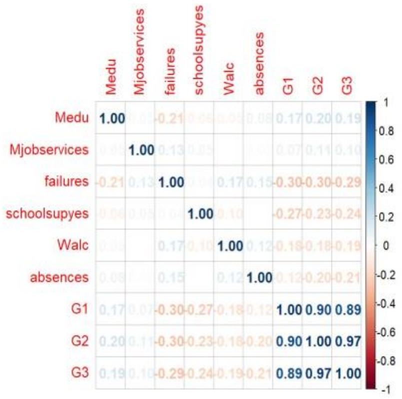
We found that there appears to be a negative correlation between G3 and the number of past failures a student had (with a correlation of -0.2938), compared to G3 and the number of absences (with a correlation of -0.2131). The correlation coefficient indicates that the more past failures a student has, the lower their final grade is.



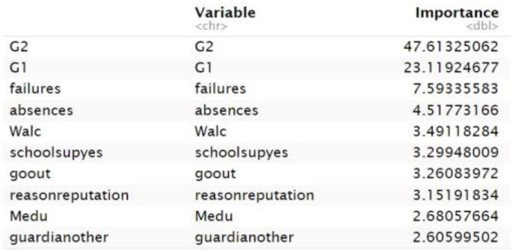


Students who plan to pursue higher education also tend to have higher final period grades, with a median final grade of around 11 out of 20 and a correlation of 0.1134. In addition, the correlation matrix displays G1 and G2 as the most significant variables for G3, along with other variables such as absences, weekend alcohol consumption, school support, and mother's education and job working for civil services.





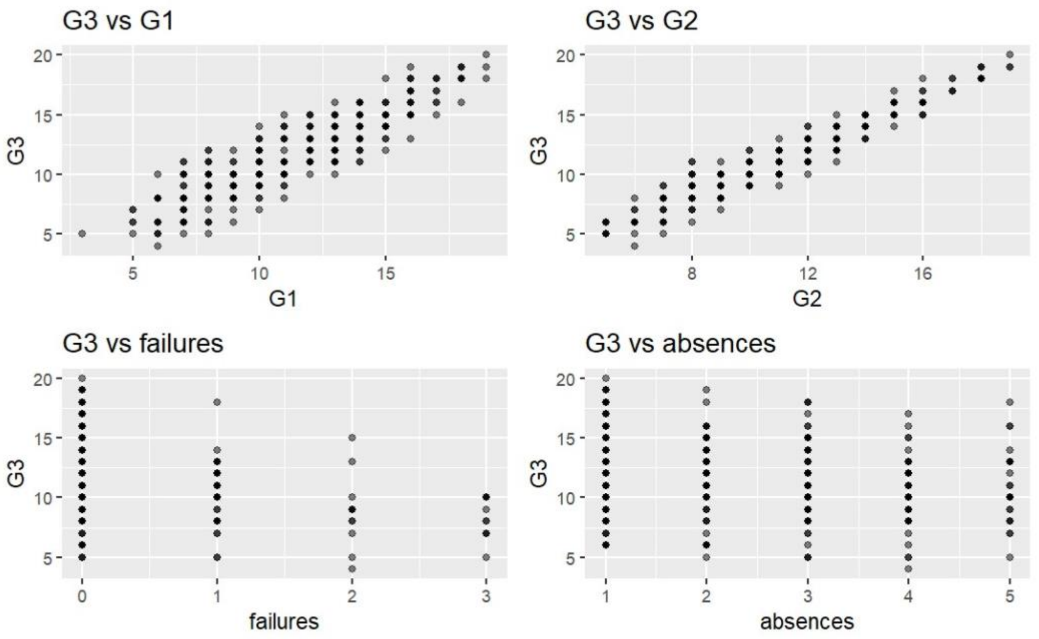
We decided to change all categorical variables into numerical variables using label encoding, which allowed us to train a random forest model on our data. In doing so, we were able to extract the importance values assigned to each variable during the random forest model training and turn it into a data frame. Displayed below are the variables with the highest importance.



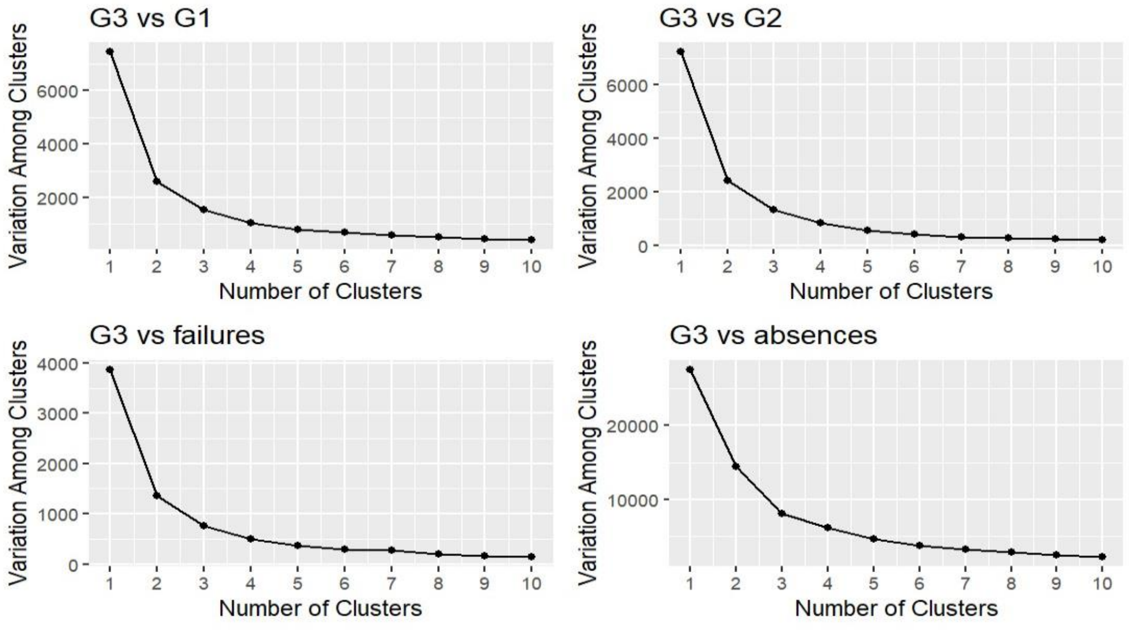
	Variable <chr>	Importance <dbl>
G2	G2	47.61325062
G1	G1	23.11924677
failures	failures	7.59335583
absences	absences	4.51773166
Walc	Walc	3.49118284
schoolsupyes	schoolsupyes	3.29948009
goout	goout	3.26083972
reasonreputation	reasonreputation	3.15191834
Medu	Medu	2.68057664
guardianother	guardianother	2.60599502

With this table, we can choose which variables to use to train our model based on either how much significance we want for our variables, cutting off the variables at a certain threshold, or deciding how many features we want first and choosing the variables from the top of the list.

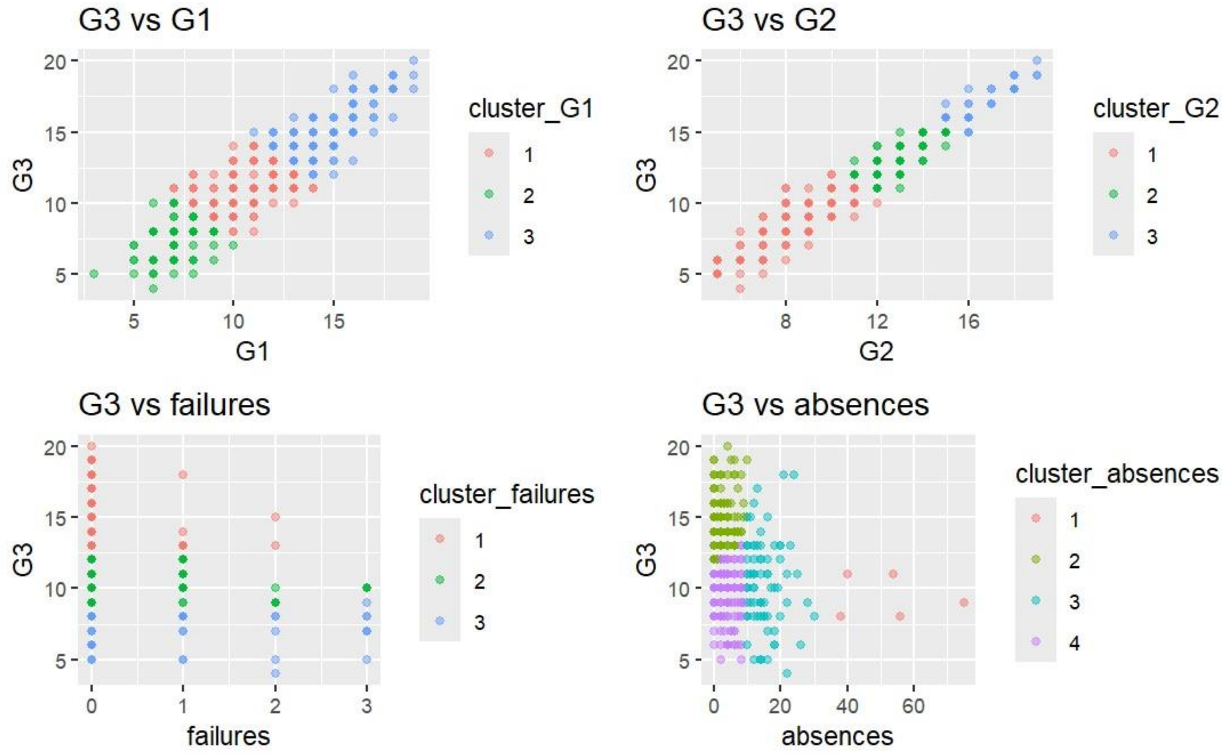
By using K-means clustering, we can group similar data points with each other in order to understand the underlying patterns of the data. In order to do this, we first created scatter plots between G3 and some of the top variables in terms of importance, such as G1, G2, failures, and absences.



We then made elbow plots for each graph to determine how many clusters would minimize the variability within the clusters.



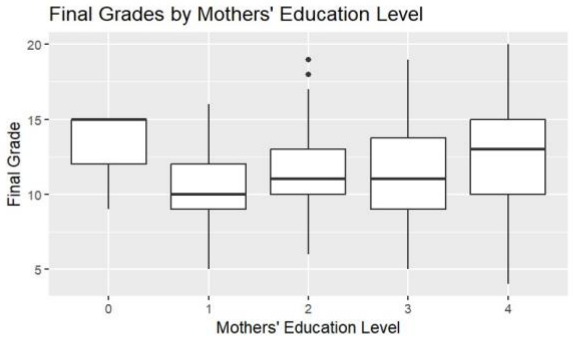
Using the optimal amount of clusters for each variable, we used k-means clustering to create graphs that separated the data points into different clusters.

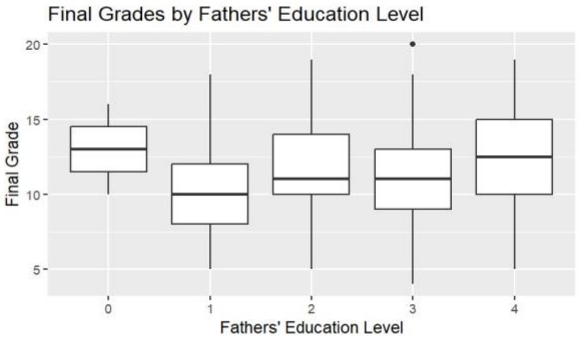


By looking at these graphs, students can be grouped together with other students in similar situations, and the school can predict if they are on the right track to passing the class, or if they might need a little bit of assistance getting there. For instance, in the orange group for G3 vs absences, it can be seen that no students with over 38 absences score over a 12 in G3, so the school can offer additional assistance to students who are missing a lot of classes.

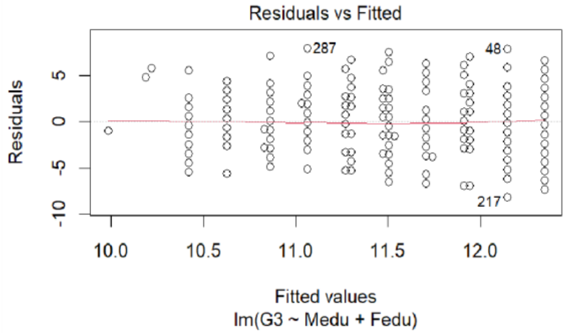
Research Question #3

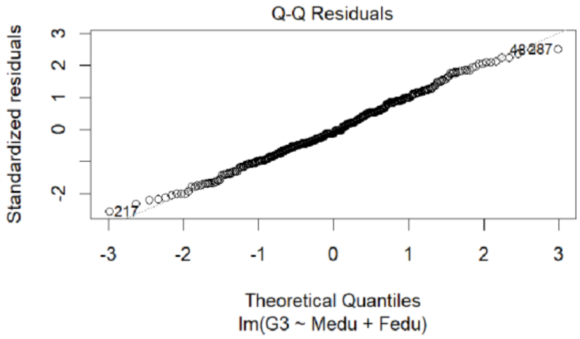
We did a multiple linear regression model to see the relationship between parents' education level and job types vs student's final grades. From the boxplots, we can see that students tend to get a higher score when the level is 0 (no education) or 4 (higher education) for both the father and mother's education levels, with a correlation coefficient of 0.1903081 for mothers and 0.1588105 for fathers.



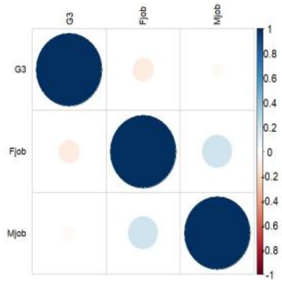


The residuals graph shows equal variance and normality is checked by looking at the QQ graph, thus the linear regression model assumptions are met.





With the model summary, we can see that the p-value for the mother's education is 0.0241, and the p-value for the father's education is 0.2986. Thus we can conclude that the mother's education level does have an effect on students' final grades, and the father's education level has no effect. This is confirmed by the importance table shown above, as the mother's education level is the only one at the top.



Next, we transform the categorical variables "Fjob" and "Mjob" into numerical variables by assigning each job type to a number. We found out that parents' jobs do not have a significant effect on student's grades, as shown in the correlation matrix.

Thus we conclude that parents' education level has a small positive correlation with students' final grades, and their occupations have no effect on the students' final grades.

Conclusion

In this research project, we investigated the influential factors on secondary education students' math performance using the Student Performance dataset from the UC Irvine Machine Learning Repository.

For our first research question, we focused on the influence of early grades on final grades. We found a strong correlation between the first (G1) and second-period (G2) grades and the final-period (G3) grades, with an adjusted R-squared of 0.8213, indicating that G1 and G2 are significant predictors of G3. Moreover, after identifying and removing outliers, the adjusted R-squared improved to 0.9343, confirming a stronger predictive relationship. This suggests that students who perform well in earlier periods are likely to maintain or improve their performance by the end of the term.

Beyond G1 and G2, other variables such as past failures, absences, and intentions to pursue higher education also significantly impact final grades. Our correlation analysis showed a negative correlation between G3 and the number of past failures (-0.2938) and absences (-0.2131). Moreover, students aspiring to higher education generally had higher final grades, with a positive correlation of 0.1134. Using a random forest model, we identified key variables such as absences, weekend alcohol consumption, school support, and mother's education and job as significant predictors of final grades.

With curiosity about the effect of parents' education on children's grades, we did a multiple regression analysis on parents’ education levels and occupations with students’ final grades, and we found that the mother’s education level has a statistically significant impact on student's final grades, with a p-value of 0.0241. In contrast, the father’s education level and parents’ occupations did not show a significant effect. This finding underscores the influence of maternal education on students' academic performance in math.

By examining these findings, we believe that schools can predict students’ performance and develop targeted strategies to support their educational journeys and foster an environment that is beneficial to their educational success.

Code Appendix

```
data=read.table("student-mat.csv",sep=";",header=TRUE) library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```
library(gridExtra)
```

```
## Warning: package 'gridExtra' was built under R version 4.3.3
```

```
library(car)
```

```
## Warning: package 'car' was built under R version 4.3.3

## Loading required package: carData

## Warning: package 'carData' was built under R version 4.3.3
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.3.3

##
## Attaching package: 'dplyr'

## The following object is masked from 'package:car':
##
##      recode

## The following object is masked from 'package:gridExtra':
##
##      combine

## The following objects are masked from 'package:stats':
##
##      filter, lag

## The following objects are masked from 'package:base':
##
##      intersect, setdiff, setequal, union
```

```
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.3.3
## corrplot 0.92 loaded
```

```
#install.packages("randomForest")
#install.packages("caret") library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 4.3.3
```

```
## randomForest 4.7-1.1
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

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

```
## The following object is masked from 'package:gridExtra':
##
##     combine
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
library(caret)
```

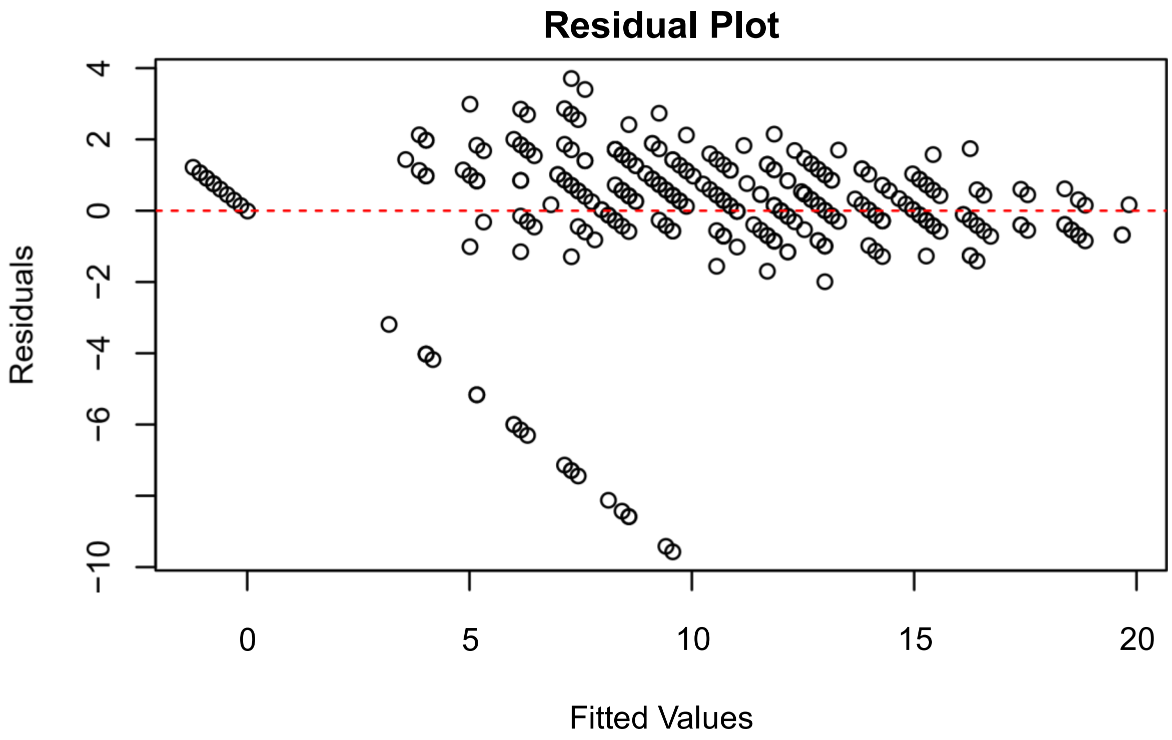
```
## Warning: package 'caret' was built under R version 4.3.3
```

```
## Loading required package: lattice
```

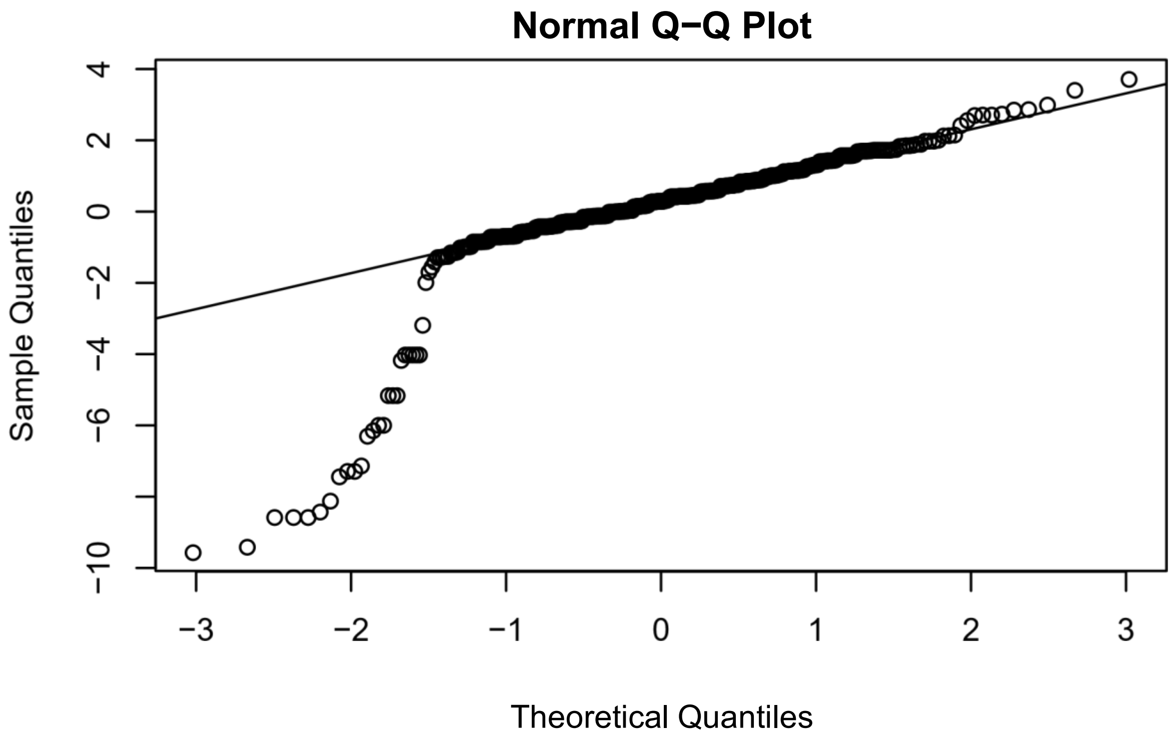
```
#fit a model to see if g1 and g2 affect g3 model1 = lm(G3 ~ G1

+ G2, data = data) summary(model1)
```

```
##
## Call:
## lm(formula = G3 ~ G1 + G2, data = data)
##
## Residuals:
##     Min      1Q Median      3Q     Max
## -9.5713 -0.3888 0.2885 0.9725 3.7089
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.83001    0.33531 -5.458 8.57e-08 ***
## G1                    0.15327         0.05618          2.728 0.00665 **
## G2                    0.98687              0.04957 19.909 < 2e-16 ***
## ---
## Signif. codes: 0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 1.937 on 392 degrees of freedom ## Multiple R-
squared: 0.8222, Adjusted R-squared: 0.8213
## F-statistic: 906.1 on 2 and 392 DF, p-value: < 2.2e-16
```

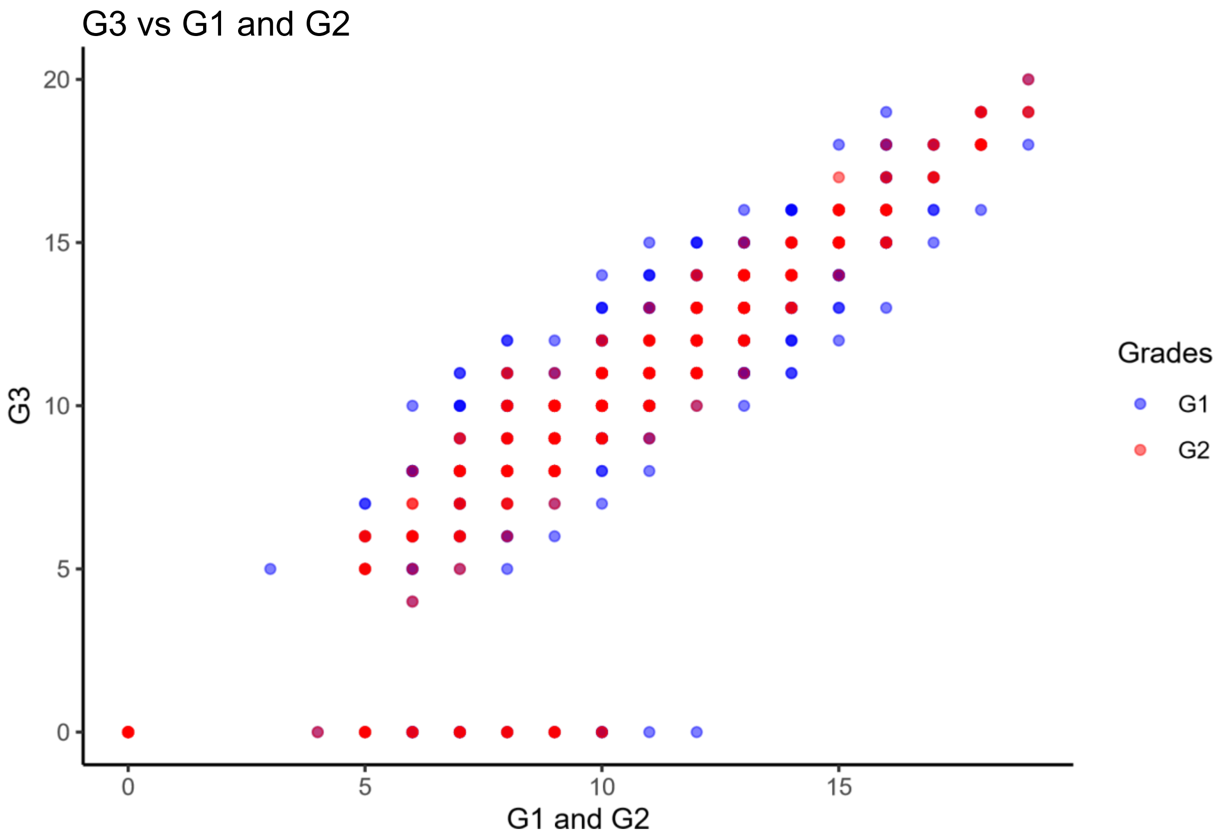
```
#residual graph
plot(model1$fitted.values, model1$residuals, xlab = "Fitted Values",
ylab = "Residuals", main = "Residual Plot") abline(h = 0, col = "red", lty =
2)
```



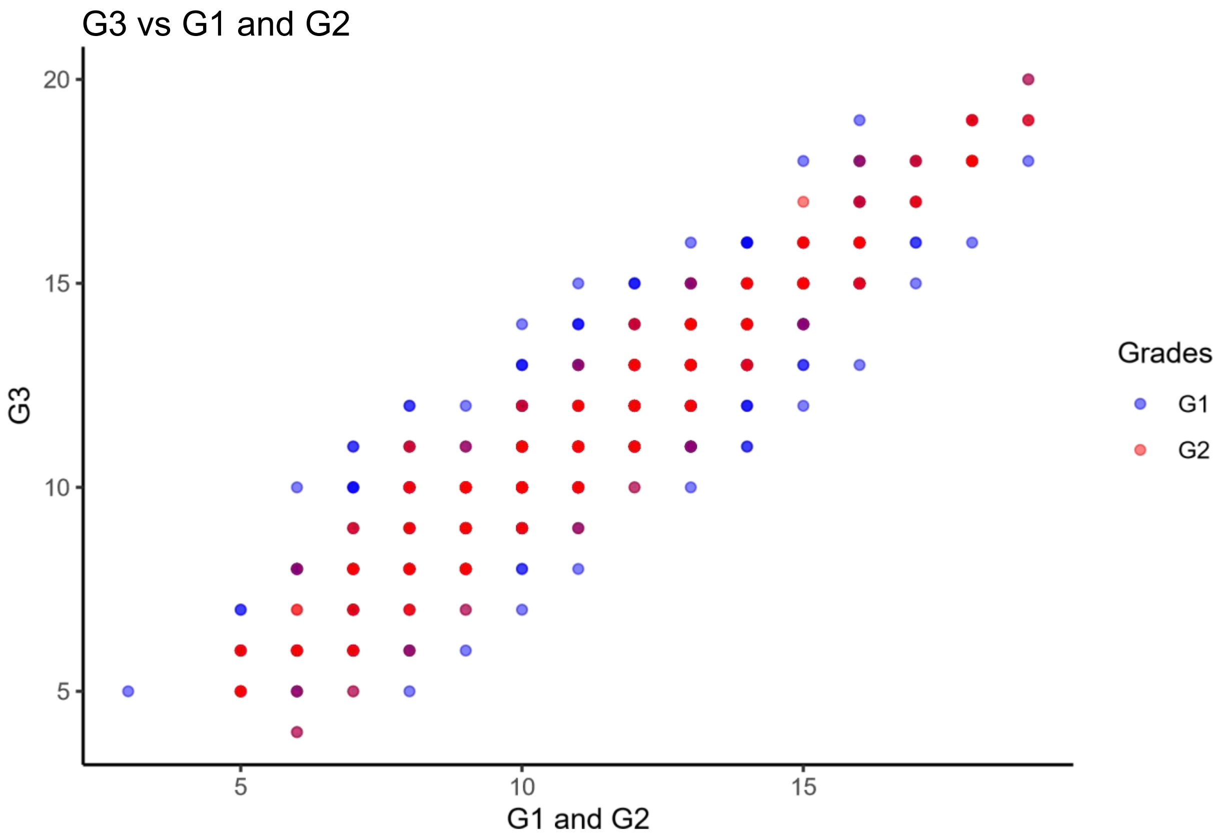
```
qqnorm(model1$residuals)
qqline(model1$residuals)
```



```
#graph G1 and G2 against G3 ggplot(data, aes(x =
G1, y = G3)) +
   geom_point(aes(color = "G1"), alpha = 0.5) +
   geom_point(aes(x = G2, y = G3, color = "G2"), alpha = 0.5) + labs(x = "G1 and G2", y =
   "G3", title = "G3 vs G1 and G2") +
   theme_classic() +
   scale_color_manual(name = "Grades", values = c("G1" = "blue", "G2" = "red"),
                                labels = c("G1", "G2"))
```

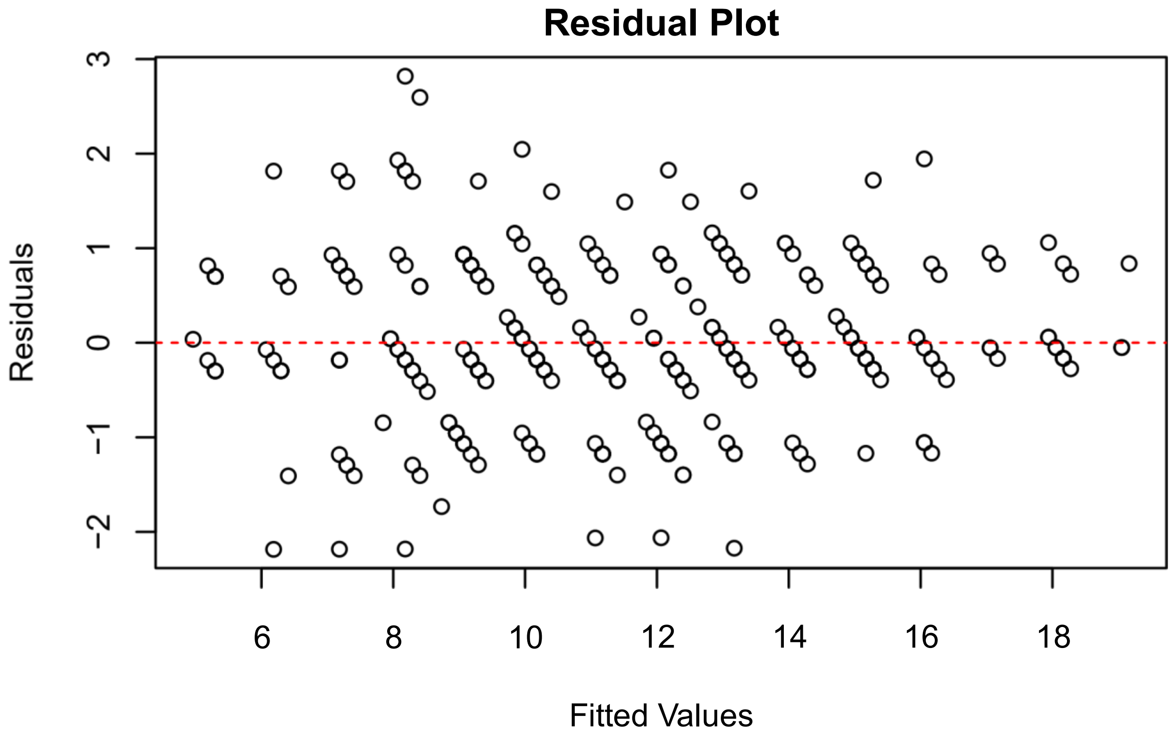


```
#remove all the data that y = 0 reduced.data =

data[data$G3 != 0, ]

#redo the model
ggplot(reduced.data, aes(x = G1, y = G3)) +
  geom_point(aes(color = "G1"), alpha = 0.5) +
  geom_point(aes(x = G2, y = G3, color = "G2"), alpha = 0.5) + labs(x = "G1 and G2", y =
  "G3", title = "G3 vs G1 and G2") +
  theme_classic() +
  scale_color_manual(name = "Grades", values = c("G1" = "blue", "G2" = "red"),
                     labels = c("G1", "G2"))
```

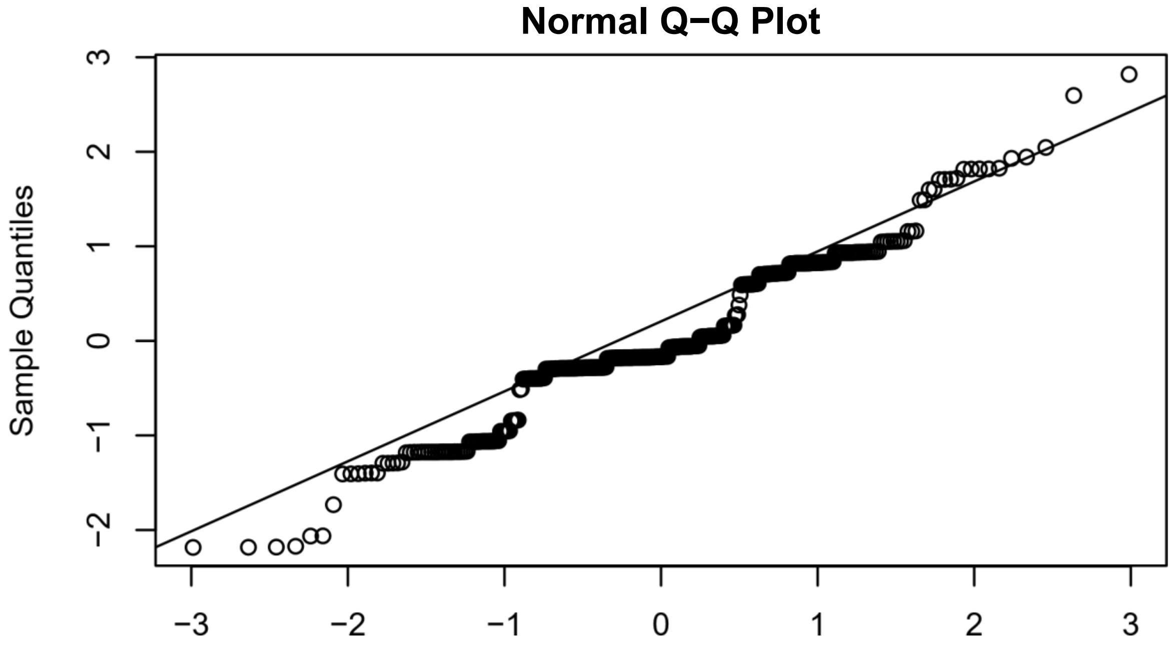


```
#model after outliers are removed model2 = lm (G3 ~ G1 +

G2, data = reduced.data) summary(model2)
```

```
##
## Call:
## lm(formula = G3 ~ G1 + G2, data = reduced.data)
##
## Residuals:
##         Min       1Q Median        3Q        Max ## -
2.1845 -0.2927 -0.1673 0.7056 2.8190
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|) ## (Intercept) 0.19482
0.16572 1.176 0.240541
## G1 0.11167 0.03133 3.564 0.000415 *** ## G2 0.88661 0.03226
27.486 < 2e-16 ***
## ---
## Signif. codes: 0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.8272 on 354 degrees of freedom
## Multiple R-squared: 0.9347, Adjusted R-squared: 0.9343
## F-statistic: 2533 on 2 and 354 DF, p-value: < 2.2e-16 plot(model2$fitted.values,
model2$residuals, xlab = "Fitted Values", ylab = "Residuals", main = "Residual Plot") abline(h
= 0, col = "red", lty = 2)
```

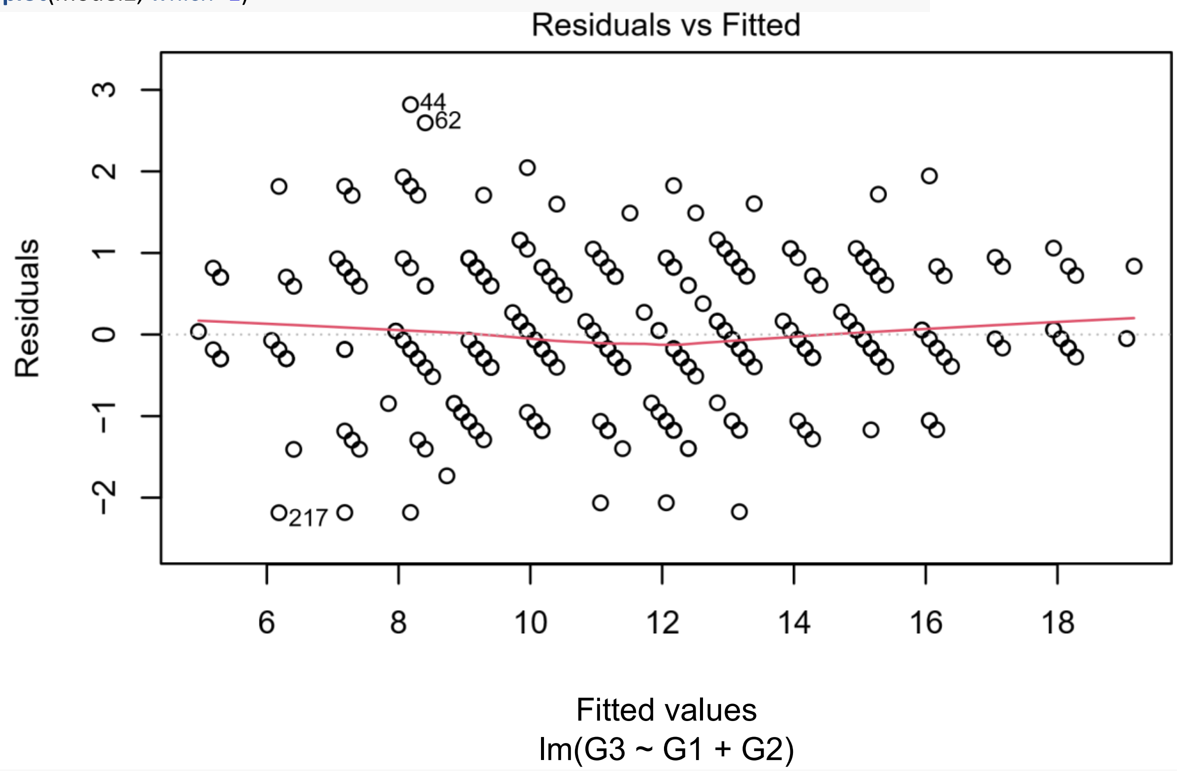


```
qqnorm(model2$residuals)
qqline(model2$residuals)
```



Theoretical Quantiles

```
#Box-Cox transformation didn't change normality that much? model2 = lm(G3 ~ G1 +
G2, data = reduced.data)
plot(model2, which=1)
```



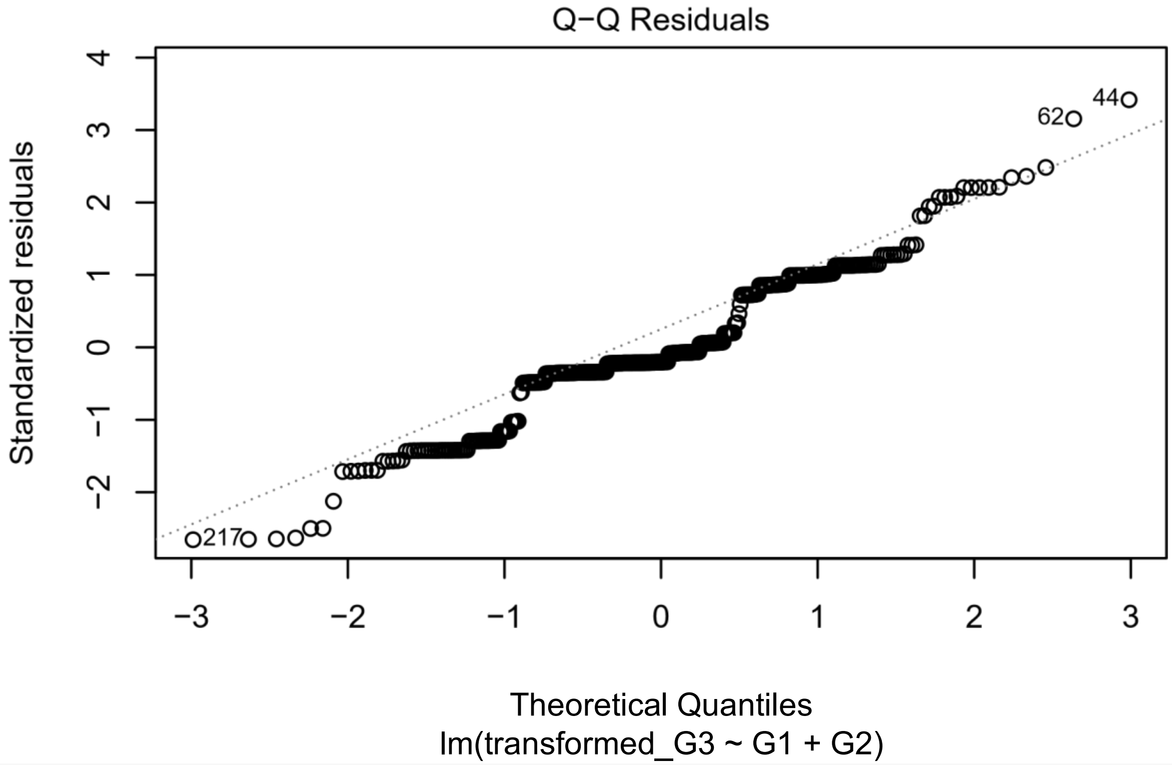
```
#perform a power transformation on G3 using the powerTransform() function p1 =
powerTransform(model2)
#prints coefficients of the power transformation coef(p1, round=TRUE)
```

```
## Y1
## 1
```

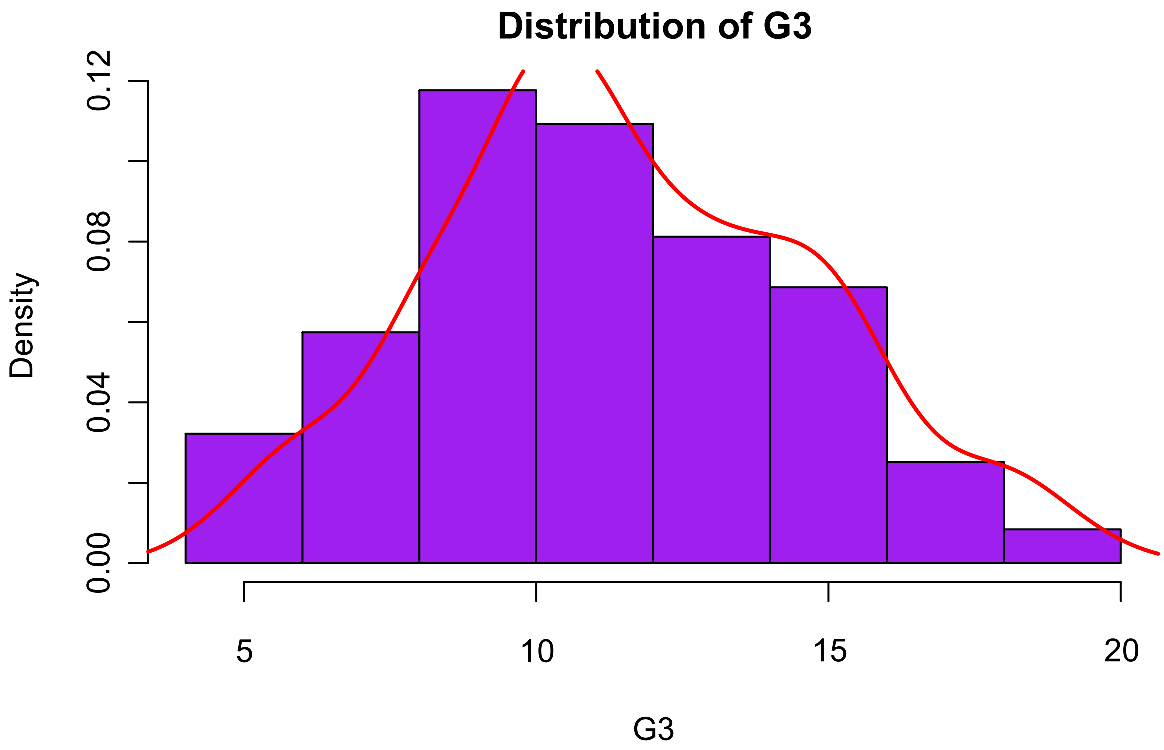
```
#perform Box-Cox transformation on G3 using the estimated lamdba value from the power transformation transform
bcPower(reduced.data$G3, p1$roundlam)
#fit a new model with transformed G3

new_model = lm(transformed_G3 ~ G1 + G2, data = reduced.data) summary(new_model)
```

```
##
## Call:
## lm(formula = transformed_G3 ~ G1 + G2, data = reduced.data)
##
## Residuals:
##          Min       1Q Median         3Q        Max ## -
2.1845 -0.2927 -0.1673 0.7056 2.8190
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.80518                       0.16572 -4.859 1.78e-06 ***
## G1 0.11167 0.03133 3.564 0.000415 *** ## G2 0.88661 0.03226
27.486 < 2e-16 ***
## ---
## Signif. codes: 0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.8272 on 354 degrees of freedom
## Multiple R-squared: 0.9347, Adjusted R-squared: 0.9343 ## F-statistic: 2533
on 2 and 354 DF, p-value: < 2.2e-16
```

```
plot(new_model, which=2)
```



```
#histogram of G3 hist(reduced.data$G3,
col="purple", prob = TRUE, xlab = "G3",
main = "Distribution of G3")
lines(density(reduced.data$G3),
        lwd = 2, col =
        "red")
```



```
cor(reduced.data$G1, reduced.data$G3)
## [1] 0.891805
```

```
cor(reduced.data$G2, reduced.data$G3)
```

```
## [1] 0.9655825
```

```
# strong correlations
```

```
#the more absences a student has, the less the grade is cor(reduced.data$failures,
reduced.data$G3)
```

```
## [1] -0.2938309
```

```
summary(lm(G3 ~ failures, reduced.data))
```

```
##
## Call:
## lm(formula = G3 ~ failures, data = reduced.data)
##
## Residuals:
##      Min      1Q Median      3Q     Max
## -6.9074 -1.9074 -0.0837 2.0926 8.0926
##
```

```
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 11.9074      0.1764 67.492 < 2e-16 *** ## failures          -
1.4119    0.2438 -5.792 1.53e-08 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.09 on 355 degrees of freedom
## Multiple R-squared: 0.08634,                          Adjusted R-squared: 0.08376
## F-statistic: 33.55 on 1 and 355 DF, p-value: 1.534e-08
```

```
#almost 0 correlation
cor(reduced.data$freetime, reduced.data$G3)
```

```
## [1] -0.02158866
```

```
summary(lm(G3 ~ freetime, reduced.data))
```

```
##
## Call:
## lm(formula = G3 ~ freetime, data = reduced.data)
##
## Residuals:
##          Min       1Q Median          3Q         Max ## -
7.4719 -2.4719 -0.5408 2.4592 8.3903
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 11.74744                   0.57566 20.407          <2e-16 ***
## freetime             -0.06888            0.16931 -0.407            0.684
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.232 on 355 degrees of freedom
## Multiple R-squared: 0.0004661, Adjusted R-squared: -0.00235
## F-statistic: 0.1655 on 1 and 355 DF, p-value: 0.6844
```

```
#the more a student goes out, the more their grade decreases
cor(reduced.data$goout, reduced.data$G3) # weak neg correlation
```

```
## [1] -0.1773828
```

```
summary(lm(G3 ~ goout, reduced.data)) # negative relationship
```

```
##
## Call:
## lm(formula = G3 ~ goout, data = reduced.data)
##
```

```
## Residuals:
##           Min        1Q Median           3Q          Max ## -
6.6251 -2.1002 -0.5255 2.4247 7.9496
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13.1500        0.5076 25.907 < 2e-16 *** ## goout               -
0.5249    0.1546 -3.396 0.000761 ***
## ---
## Signif. codes: 0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 3.181 on 355 degrees of freedom
## Multiple R-squared: 0.03146,                              Adjusted R-squared: 0.02874
## F-statistic: 11.53 on 1 and 355 DF, p-value: 0.0007612
```

```
#summary statistics for G3 vs. internet summary(lm(G3 ~ internet,
reduced.data))
```

```
##
## Call:
## lm(formula = G3 ~ internet, data = reduced.data)
##
## Residuals:
##           Min        1Q Median           3Q          Max ## -
7.6823 -1.7069 -0.6823 2.3177 8.3177
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10.7069                      0.4218 25.385          <2e-16 ***
## internetyes             0.9754            0.4609        2.116          0.035 *
## ---
## Signif. codes: 0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 3.212 on 355 degrees of freedom
## Multiple R-squared: 0.01246,                              Adjusted R-squared: 0.009678
## F-statistic: 4.479 on 1 and 355 DF, p-value: 0.03501
```

```
#higher education has correlation with G3
cor(ifelse(reduced.data$higher == 'yes', 1, 0), reduced.data$G3)
```

```
## [1] 0.1134186
```

```
summary(lm(G3 ~ higher, reduced.data))
```

```
##
## Call:
## lm(formula = G3 ~ higher, data = reduced.data)
##
## Residuals:
```

```
##           Min        1Q Median           3Q          Max ## -
7.5977 -1.7143 -0.5977 2.4023 8.4023
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)              9.7143              0.8583 11.318          <2e-16 ***
## higheryes                1.8834            0.8756        2.151        0.0322 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.211 on 355 degrees of freedom
## Multiple R-squared: 0.01286,                              Adjusted R-squared: 0.01008
## F-statistic: 4.626 on 1 and 355 DF, p-value: 0.03216
```

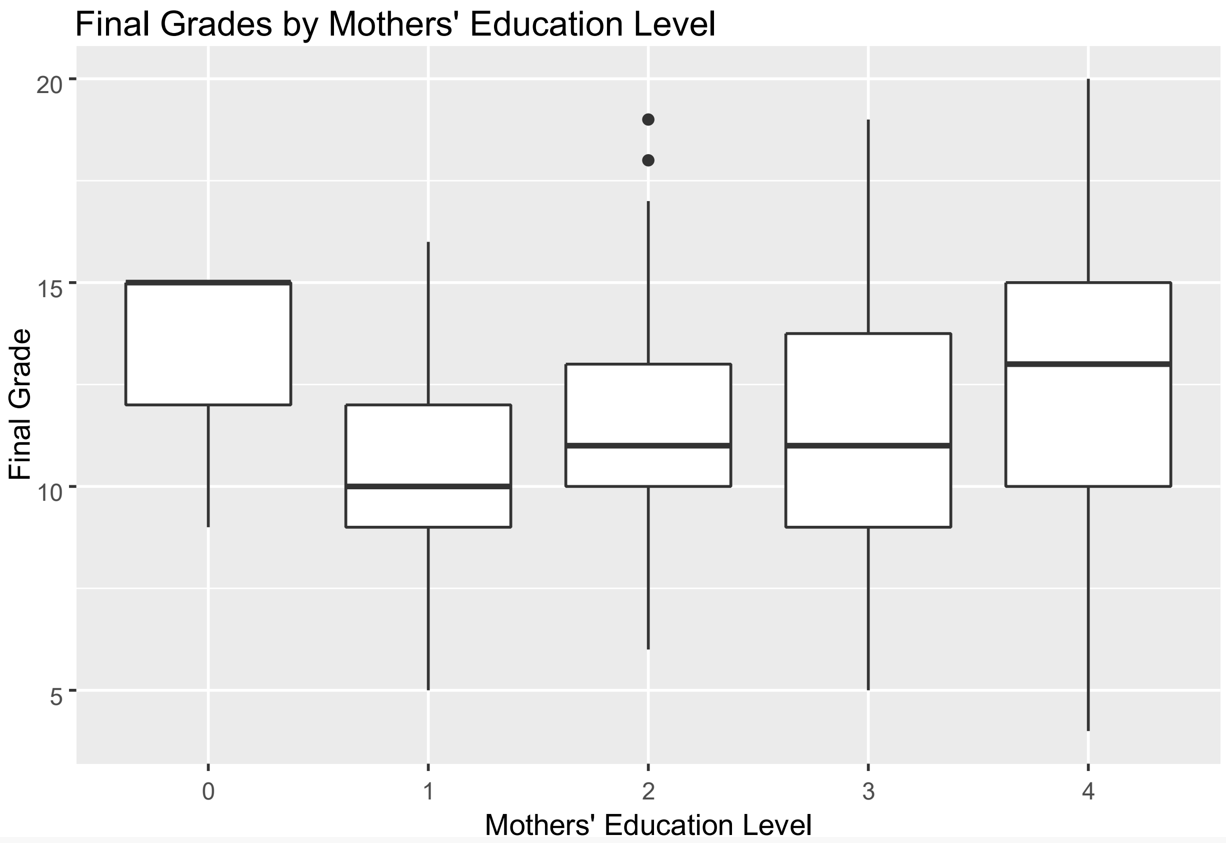
```
# parents education level has correlation with G3 cor(reduced.data$Medu,

reduced.data$G3)
```

```
## [1] 0.1903081
```

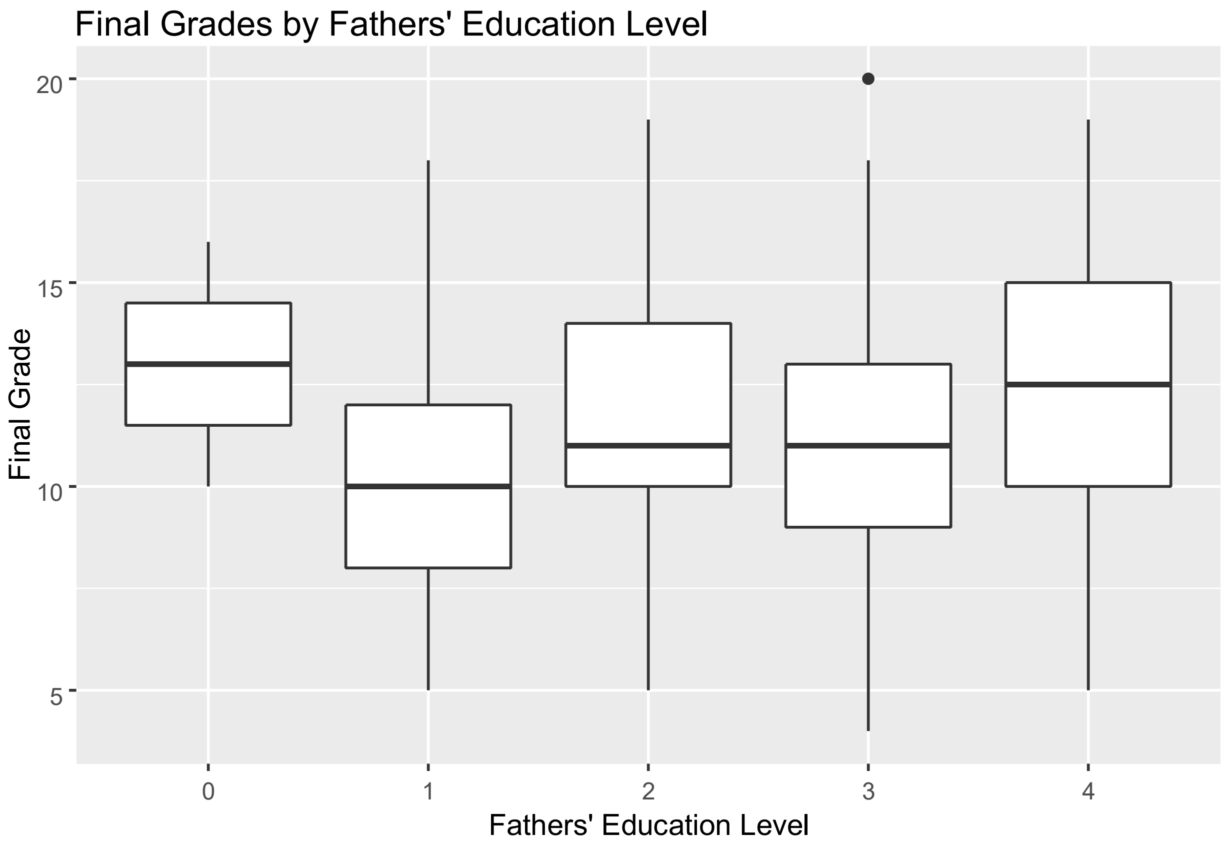
```
cor(reduced.data$Fedu, reduced.data$G3)
```

```
## [1] 0.1588105
```

```
ggplot(reduced.data, aes(x = factor(Medu), y = G3)) + geom_boxplot() +
    labs(title = "Final Grades by Mothers' Education Level", x = "Mothers'
             Education Level", y = "Final Grade")
```



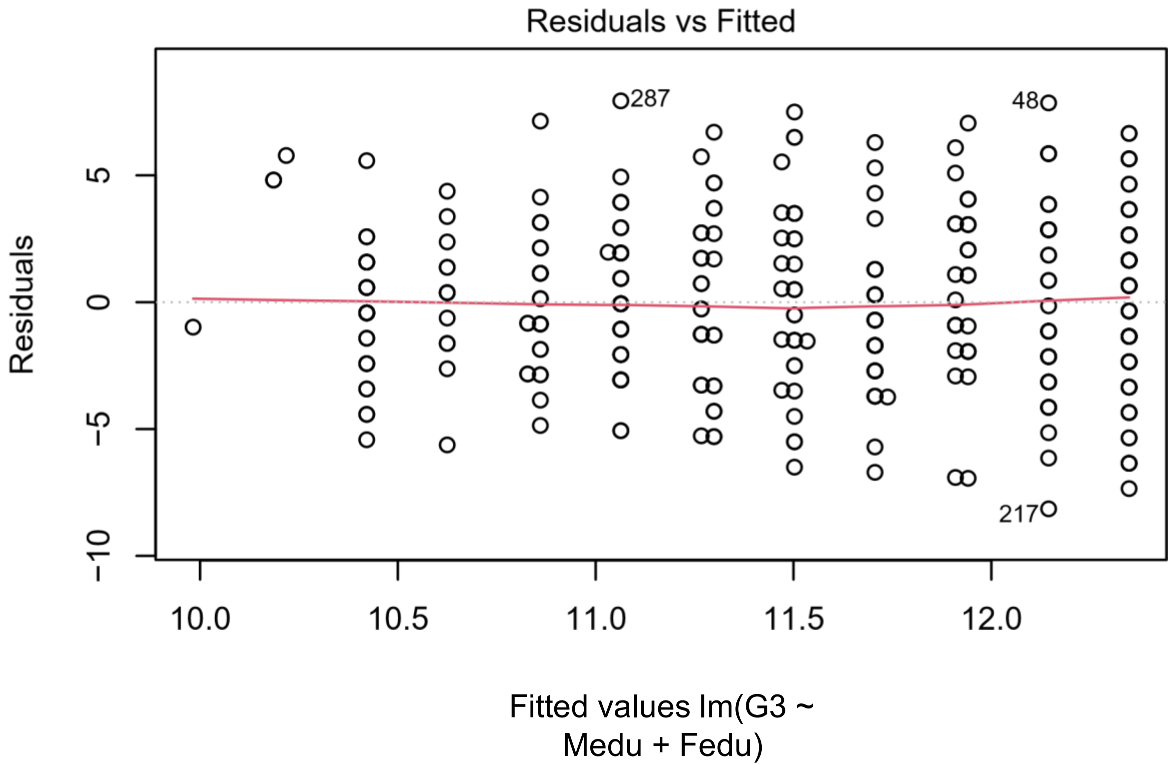
```
ggplot(reduced.data, aes(x = factor(Fedu), y = G3)) + geom_boxplot() +
  labs(title = "Final Grades by Fathers' Education Level", x = "Fathers'
        Education Level", y = "Final Grade")
```

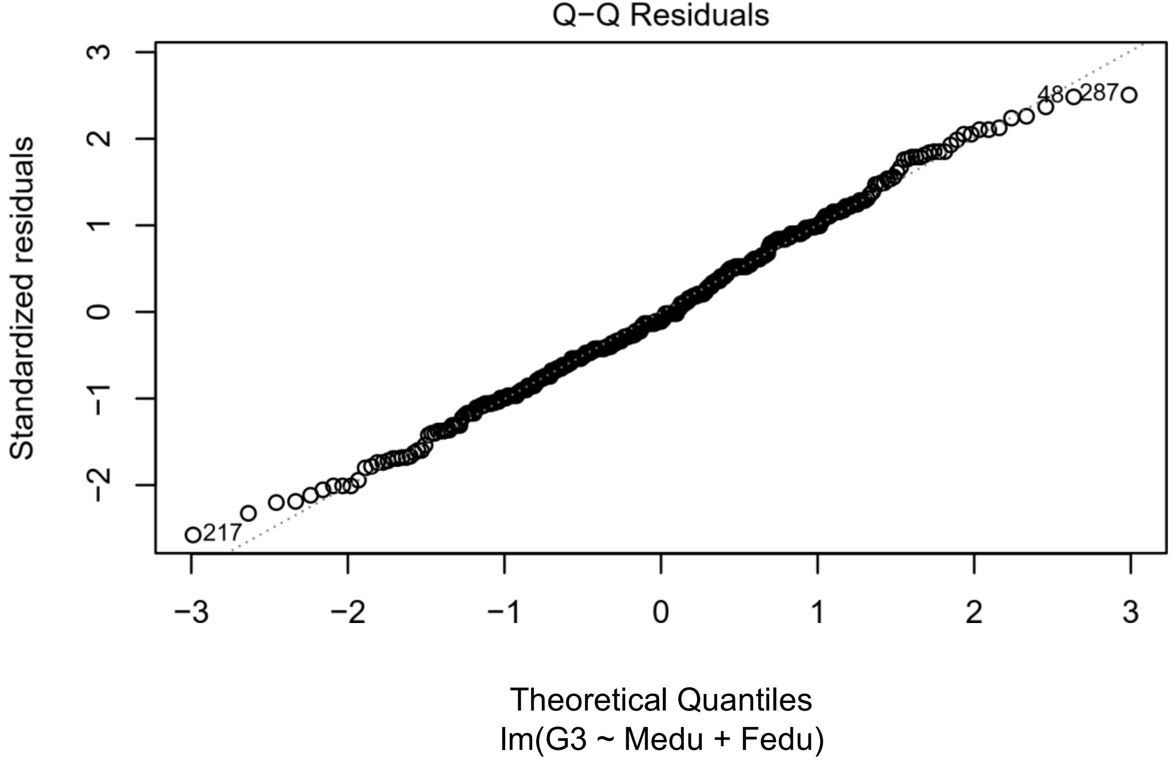


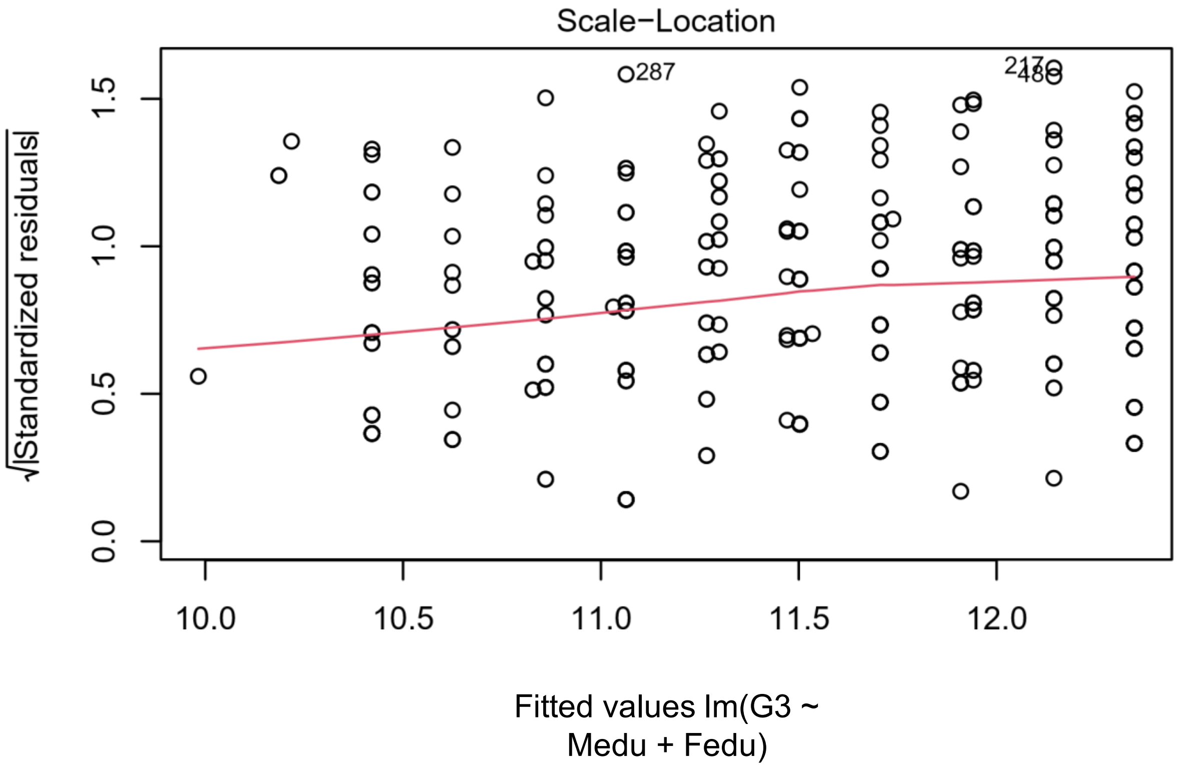
```
model_parent = lm(G3 ~ Medu + Fedu, reduced.data) summary(model_parent)
```

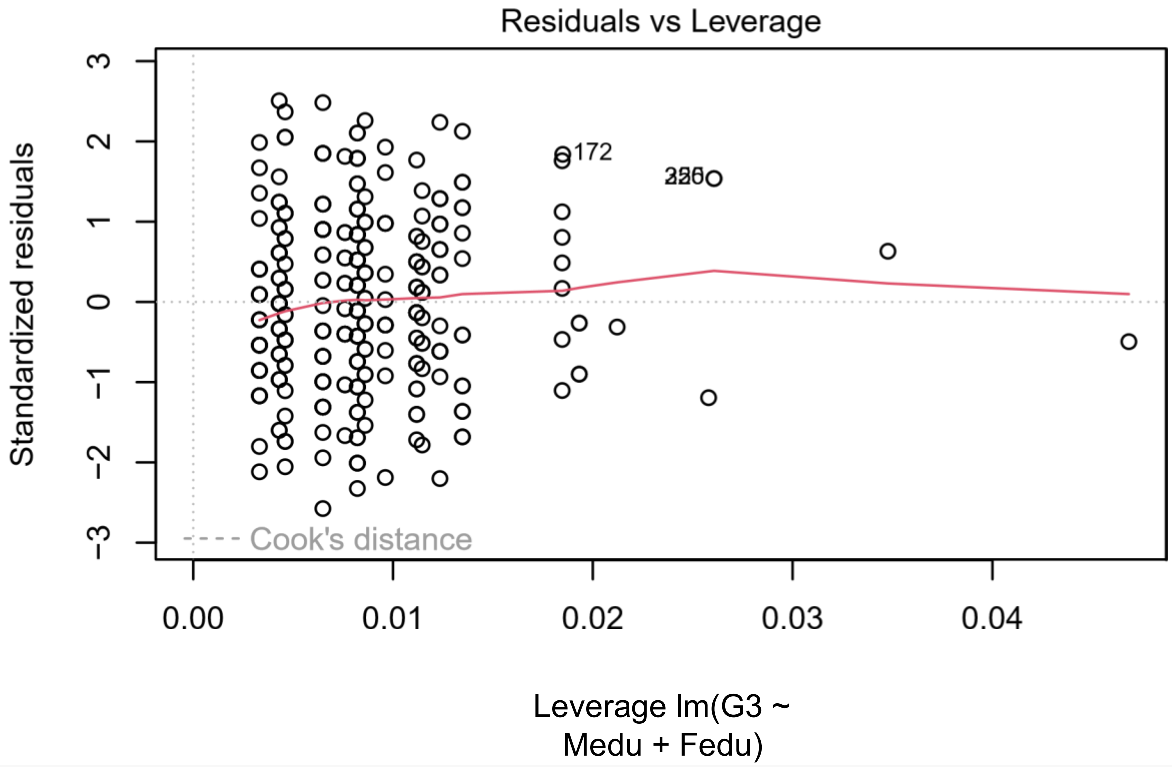
```
##
## Call:
## lm(formula = G3 ~ Medu + Fedu, data = reduced.data)
##
## Residuals:
##          Min       1Q Median         3Q        Max ## -
8.1447 -2.1447 -0.3481 2.1398 7.9364
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            9.7790            0.4911 19.914         <2e-16 ***
## Medu                   0.4389           0.1937       2.266       0.0241 *
## Fedu                   0.2034           0.1954       1.041       0.2986
## ---
## Signif. codes: 0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 3.173 on 354 degrees of freedom
## Multiple R-squared: 0.03916, Adjusted R-squared: 0.03373 ## F-statistic: 7.213 on
2 and 354 DF, p-value: 0.00085
```

```
plot(model_parent)
```


Residuals vs Fitted
Residuals
5
0
−5
−10
287
48
217
10.0
10.5
11.0
11.5
12.0
Fitted values lm(G3 ~
Medu + Fedu)


Q−Q Residuals
Standardized residuals
3
2
1
0
−1
−2
48
287
217
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(G3 ~ Medu + Fedu)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
287
217
480
10.0
10.5
11.0
11.5
12.0
Fitted values lm(G3 ~
Medu + Fedu)



```
#turn parent's job into numerical varaibles
temp1 = data.frame(reduced.data$Mjob,reduced.data$Fjob, reduced.data$G3) temp2 =
temp1 %>% mutate(mother = recode(reduced.data$Mjob,
                                          "at_home" = 1, "teacher" = 2,
                                          "services" = 3,
                                          "health" = 4,
                                          "other" = 5))

cor(temp2$mother, temp2$reduced.data.G3)
```

```
## [1] -0.03557432
```

```
summary(lm(reduced.data.G3 ~ mother, temp2))
```

```
##
## Call:
## lm(formula = reduced.data.G3 ~ mother, data = temp2)
##
## Residuals:
##       Min      1Q Median       3Q      Max ## -
7.3944 -2.3944 -0.3944 2.4468 8.5262
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept) 11.79125          0.43386 27.178      <2e-16 ***
## mother          -0.07937      0.11833 -0.671      0.503
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.23 on 355 degrees of freedom
## Multiple R-squared: 0.001266,        Adjusted R-squared: -0.001548
## F-statistic: 0.4498 on 1 and 355 DF, p-value: 0.5029
```

```
temp2 = temp2 %>% mutate(father=
  recode(reduced.data$Fjob,
                              "at_home" = 1, "teacher" = 2,
                              "services" = 3,
                              "health" = 4,
                              "other" = 5))

#create correlation graph

cor(temp2$father, temp2$reduced.data.G3)
```

```
## [1] -0.1070771
```

```
summary(lm(reduced.data.G3 ~ father, temp2))
```

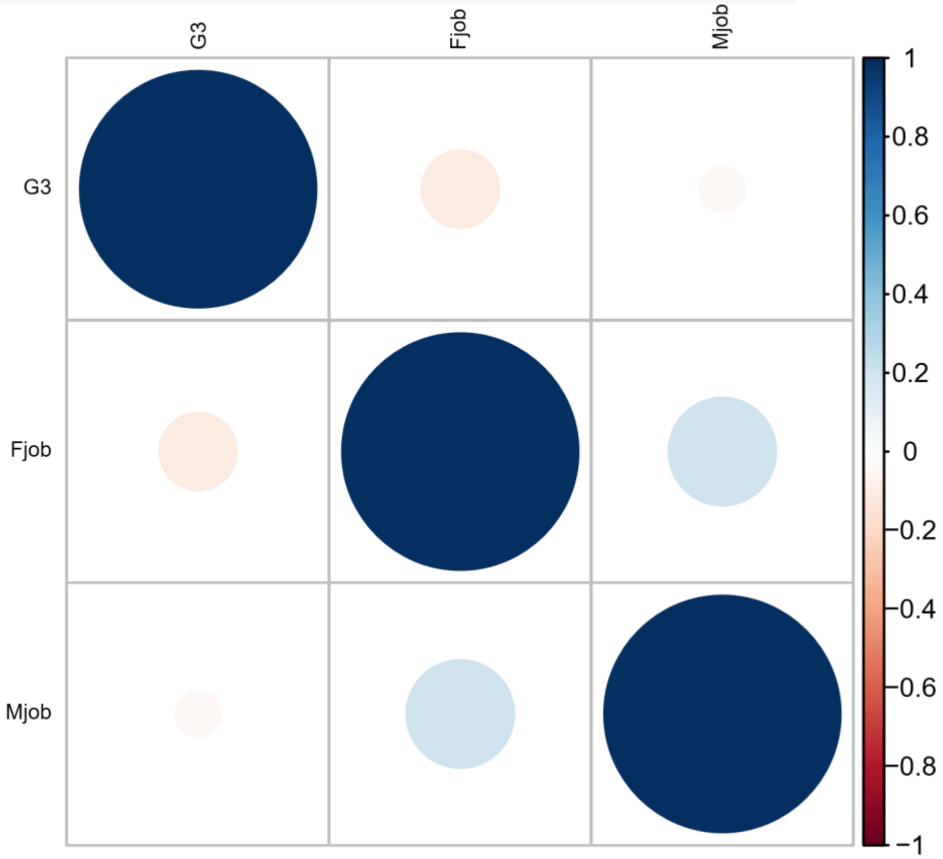
```
##
## Call:
## lm(formula = reduced.data.G3 ~ father, data = temp2)
##
## Residuals:
##     Min     1Q Median     3Q    Max
## -7.241 -2.241 -0.241 2.482 8.204
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 12.6277      0.5700 22.154    <2e-16 ***
## father      -0.2773      0.1367 -2.029    0.0432 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.214 on 355 degrees of freedom
## Multiple R-squared: 0.01147,        Adjusted R-squared: 0.008681
## F-statistic: 4.117 on 1 and 355 DF, p-value: 0.04319
```

```
temp3 = data.frame(temp2$reduced.data.G3, temp2$father, temp2$mother)

job_cor = cor(temp3) print(job_cor)
```

```
## temp2.reduced.data.G3 temp2.father temp2.mother ## temp2.reduced.data.G3
1.00000000 -0.1070771 -0.03557432
## temp2.father                 -0.10707713   1.0000000   0.20656744
## temp2.mother                 -0.03557432   0.2065674   1.00000000
```

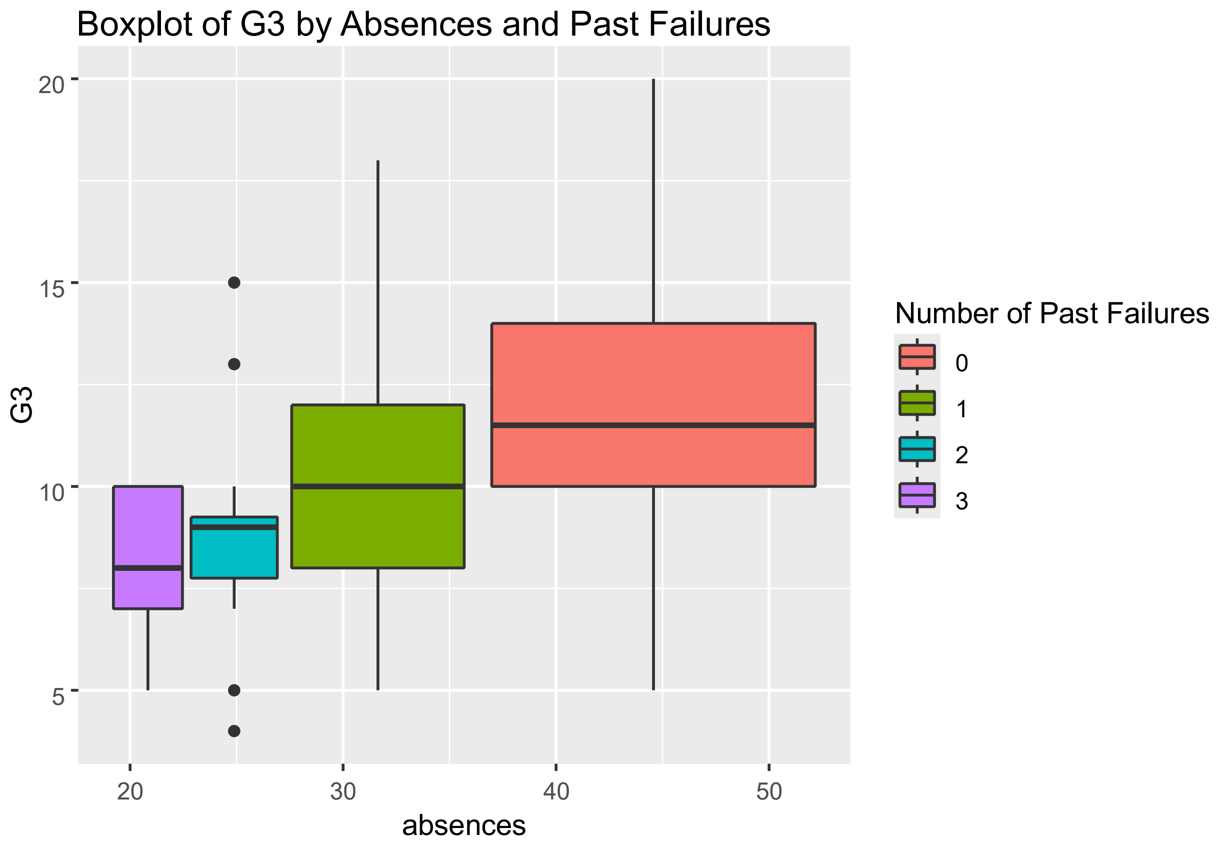
```
name = c("G3","Fjob","Mjob")
```

```
colnames(job_cor) = name rownames(job_cor) = name corrplot(job_cor, method =

"circle", tl.cex = 0.7, tl.col = "black")
```

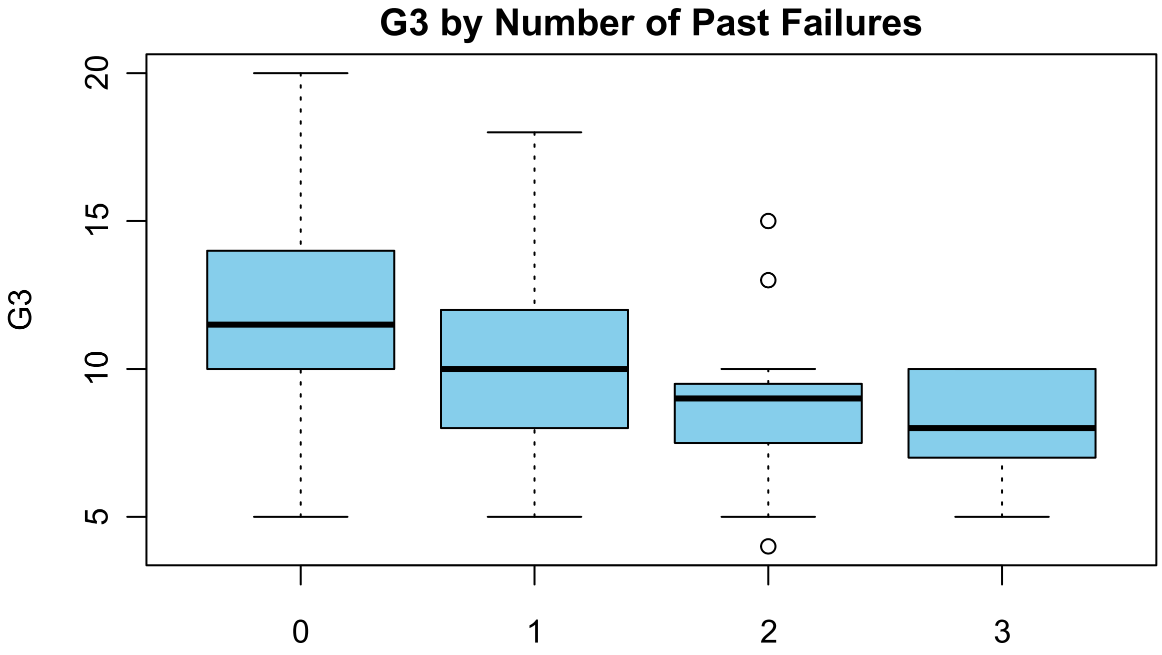


```
#fit 5 different models and calculate the R^2 fit1 =
lm(G3~G1, data = reduced.data) fit2 = lm(G3~G2, data =
reduced.data) fit3 = lm(G3~failures, data = reduced.data)
fit4 = lm(G3~absences, data = reduced.data) fit5 =
lm(G3~schoolsup, data = reduced.data)                                                  ),
rsquares = data.frame(model = c("G3~G1", "G3~G2", "G3~failures", "G3~absences", "G3~higher" R2 =
                        c(summary(fit1)$r.squared, summary(fit2)$r.squared, summary(fit3)$r.squared,      (fit5)$r.squa
                        summary(fit4)$r.squared, summary
rsquares
```

```
#G3 is more correlated with number of past failures than number of absences ggplot(reduced.data,
aes(absences, G3, fill=factor(failures))) + geom_boxplot() +
  ggtitle("Boxplot of G3 by Absences and Past Failures") + labs(fill = "Number of Past
  Failures")
```

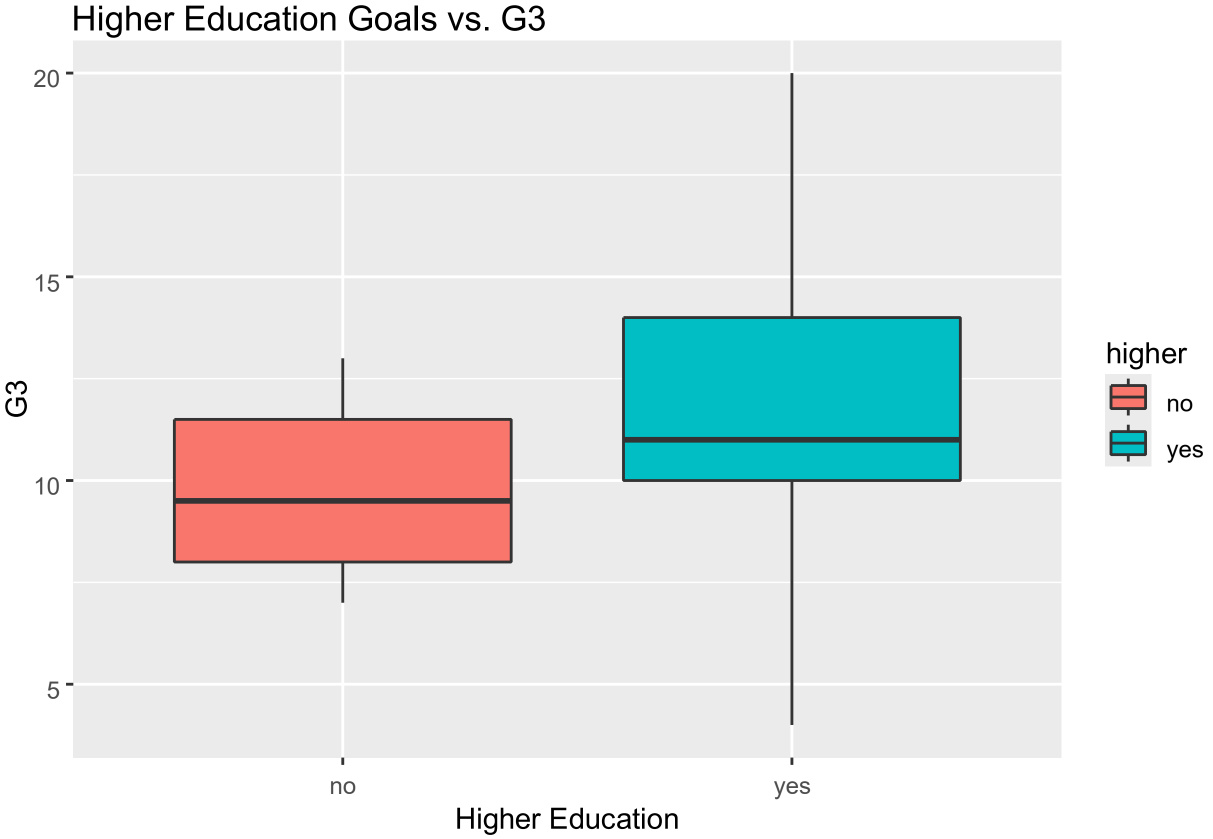


```
#boxplot of G3 by number of past failures boxplot(reduced.data$G3 ~
reduced.data$failures,
              col='skyblue',
              main='G3 by Number of Past Failures', xlab='Number
              of Past Failures', ylab='G3')
```



Number of Past
Failures

```
#boxplots for G3 in relation to whether students want higher education ggplot(data = reduced.data,
aes(x = higher, y = G3, fill = higher)) + geom_boxplot() +
      labs(title = 'Higher Education Goals vs. G3', x = 'Higher Education', y = 'G3')
```



```
#changes all variables into numerical values df_dummy = model.matrix(~ . - 1, data =
reduced.data)
df_dummy = data.frame(df_dummy)
df_dummy
```

```
#set seed to get the same output set.seed(800)
#create a random forest model
rf_model = randomForest(G3 ~ ., data = df_dummy, importance = TRUE)

#find the importance of each feature importance =
importance(rf_model)
importance
```

```
##                         %IncMSE IncNodePurity
## schoolGP           -0.36962232      2.711503
## schoolMS           -0.67810329      3.135983
## sexM                1.06193910     12.584136
## age                 1.53636357     28.940620
## addressU              1.13840753        9.284175
## famsizeLE3            0.57146266        8.230343
## PstatusT              0.77457140        6.403479
## Medu                  2.68057664       33.460566
## Fedu                  0.82165853       20.956425
## Mjobhealth            1.43655230        6.530603
## Mjobother              0.72949287      10.987348
## Mjobservices           0.78279945      10.535891
## Mjobteacher            0.09951471       4.735331
## Fjobhealth             0.01605371       3.026464
## Fjobother              0.08267064       8.278488
## Fjobservices          -0.43856784       6.042378
## Fjobteacher            0.03595527      14.957156
## reasonhome            -0.65869546       7.149836
## reasonother            0.77580580       4.401587
## reasonreputation 3.15191834               8.808678
## guardianmother         0.38728941       6.139268
## guardianother         2.60599502        2.829744
## traveltime           -0.07448962       10.415860
## studytime             0.49246673       28.797625
## failures              7.59335583       72.856441
## schoolsupyes          3.29948009       36.627585
## famsupyes             2.39748224        9.371643
## paidyes               1.76766331       10.123318
## activitiesyes         0.47013477        6.654691
## nurseryyes            0.39060483        8.842885
## higheryes            -0.63963549        1.391632
## internetyes           0.07664341        7.226612
## romanticyes          -0.23680088        7.266316
## famrel                1.81350298       21.358904
## freetime              0.78174477       21.675410
## goout                 3.26083972       34.814364
## Dalc                  0.79630780       15.022408
## Walc                  3.49118284       34.397062
## health                0.33873133       29.165790
## absences              4.51773166       72.617001
## G1                   23.11924677     1020.655772
## G2                   47.61325062     1976.888499
```

```
importance_df = data.frame(Variable = rownames(importance), Importance = importance[, 1]) importance_df =
importance_df[order(importance_df$Importance, decreasing = TRUE), ] print(importance_df)
```

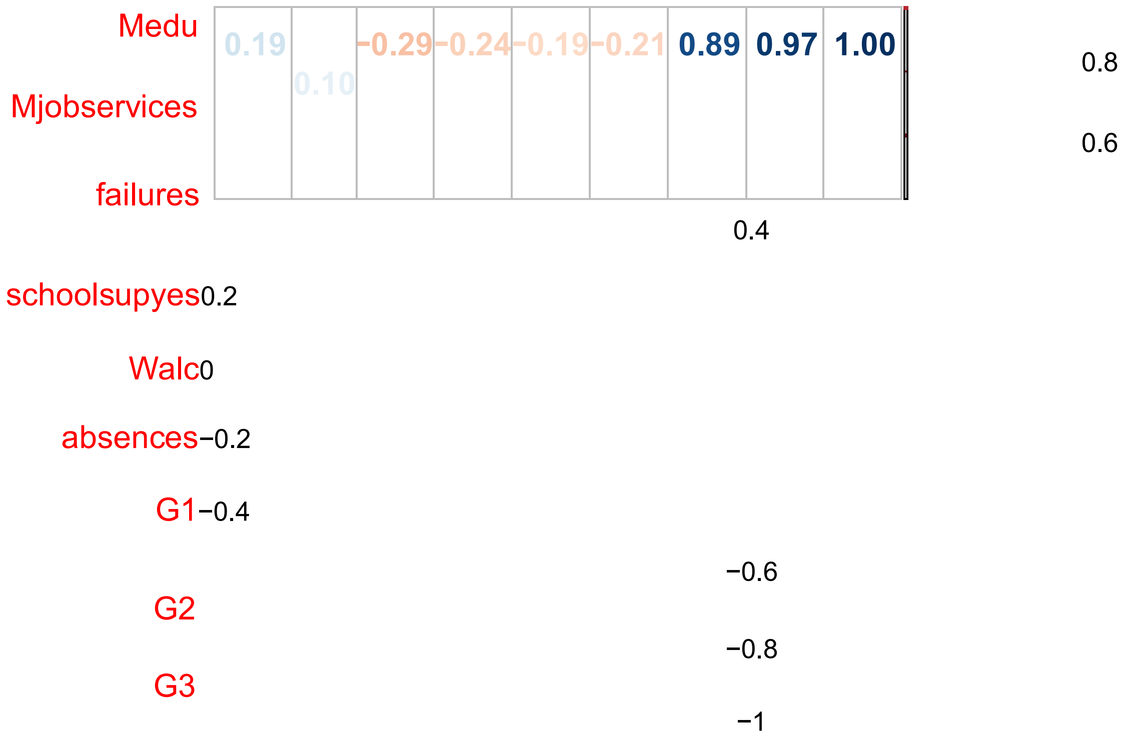
```
##          Variable Importance ## G2            G2 47.61325062
## G1                                          G1 23.11924677
## failures         failures 7.59335583 ## absences     absences
4.51773166
## Walc                                       Walc 3.49118284
## schoolsupyes                    schoolsupyes 3.29948009
## goout goout 3.26083972 ## reasonreputation
reasonreputation 3.15191834
## Medu                                          Medu 2.68057664
## guardianother guardianother 2.60599502 ## famsupyes
famsupyes 2.39748224 ## famrel    famrel 1.81350298 ##
paidyes  paidyes 1.76766331
## age                                            age 1.53636357
## Mjobhealth     Mjobhealth 1.43655230 ## addressU
addressU 1.13840753
## sexM                                          sexM 1.06193910
## Fedu                                          Fedu 0.82165853
## Dalc                                          Dalc 0.79630780
## Mjobservices                     Mjobservices 0.78279945
## freetime                                  freetime 0.78174477
## reasonother                          reasonother 0.77580580
## PstatusT                                  PstatusT 0.77457140
## Mjobother                              Mjobother 0.72949287
## famsizeLE3                            famsizeLE3 0.57146266
## studytime                                studytime 0.49246673
## activitiesyes                   activitiesyes 0.47013477
## nurseryyes                            nurseryyes 0.39060483
## guardianmother             guardianmother 0.38728941
## health                                      health 0.33873133
## Mjobteacher                          Mjobteacher 0.09951471
## Fjobother                              Fjobother 0.08267064
## internetyes                          internetyes 0.07664341
## Fjobteacher                          Fjobteacher 0.03595527
## Fjobhealth                            Fjobhealth 0.01605371
## traveltime                           traveltime -0.07448962
## romanticyes                         romanticyes -0.23680088
## schoolGP                               schoolGP -0.36962232
## Fjobservices                   Fjobservices -0.43856784
## higheryes                             higheryes -0.63963549
## reasonhome                           reasonhome -0.65869546
## schoolMS                               schoolMS -0.67810329
#we can pick which features to use based on how many we want to use and starting from the
#top or choosing all variables over a certain threshold
```

```
#calculate the correlation matrix cor_matrix =
cor(df_dummy)
#print all the correlations between each variable with G3 cor_with_g3 = cor_matrix[,
"G3"]
cor_with_g3 = cor_with_g3[order(abs(cor_with_g3), decreasing = TRUE)] print(cor_with_g3)
```

```
##        G3        G2        G1        failures ##        1.0000000000     0.9655825287
0.8918050310     -0.2938309093
##          schoolsupyes                absences                  Medu                  Walc
##        -0.2383648498        -0.2131285321         0.1903080729         -0.1900538533
##                    goout                    Fedu          Fjobteacher              Mjobother
##        -0.1773827901          0.1588105457         0.1584550259         -0.1425405683
##                      Dalc                      age           Mjobhealth               addressU
##        -0.1406897637         -0.1403717829          0.1345806598          0.1300898332
##               studytime                higheryes             internetyes                        sexM
##         0.1267280984      0.1134185689      0.1116238759      0.1024479378 ##
Mjobservices        traveltime            schoolGP            schoolMS
##          0.1021342377         -0.0997845234          0.0836150926         -0.0836150926
##                    health                Fjobother         guardianother                famsupyes
##   -0.0816909599    -0.0815020834    -0.0694908665    -0.0673049785    ##    activitiesyes
reasonreputation     romanticyes     Mjobteacher     ##     0.0585995095     0.0565727878     -
0.0499435699 0.0453897682 ##      famsizeLE3          famrel     paidyes   PstatusT
## 0.0397442579 0.0377105060 -0.0288986179 -0.0266645220 ## nurseryyes freetime
Fjobservices    guardianmother     ##     0.0265406395     -0.0215886601     -0.0181544574
0.0077275704
##              Fjobhealth              reasonother              reasonhome
##          0.0062411044          -0.0038965449          0.0003721098
```

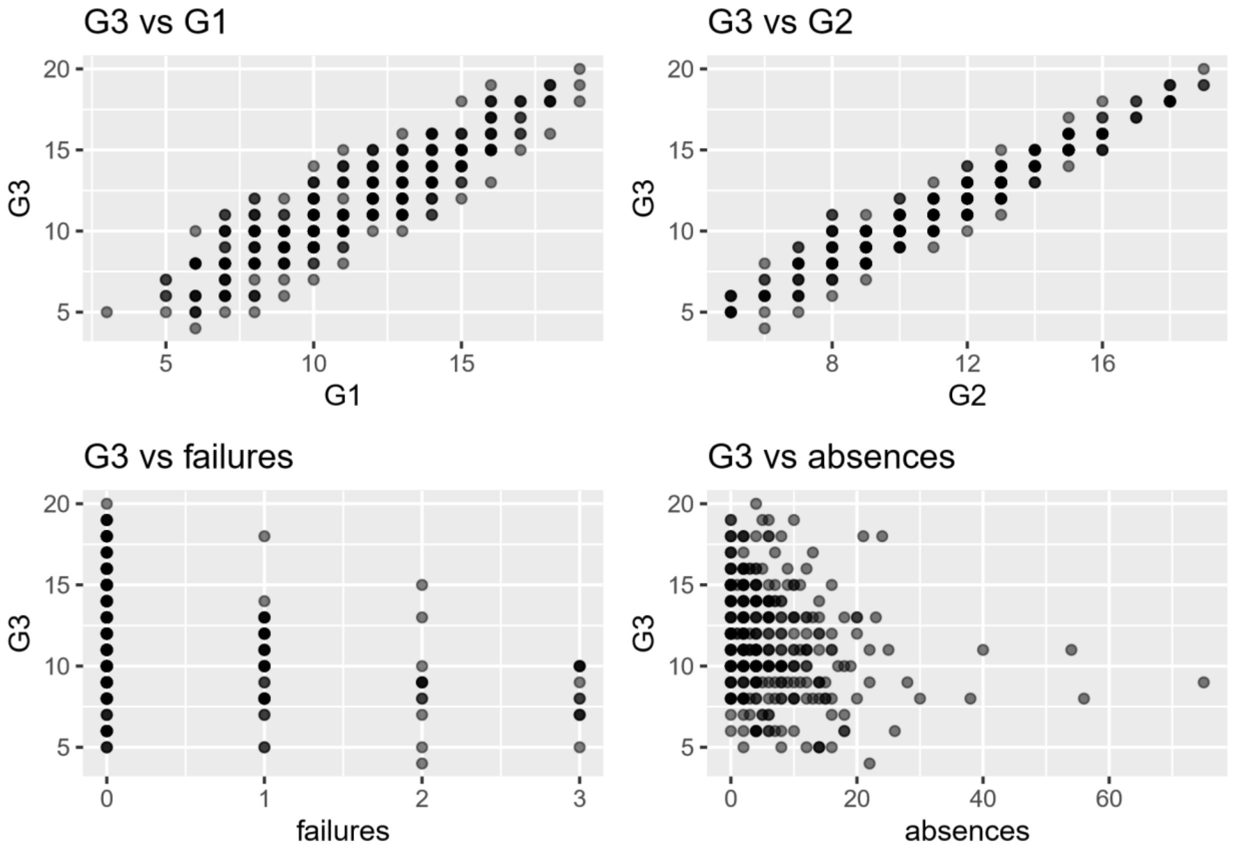
```
#correlation matrix of the most correlated numeric variables
temp = df_dummy[, (names(df_dummy) %in% c("G3", "G1", "G2", "failures", "absences", datamatrix = cor(temp)     "schoolsupyes'
corrplot(datamatrix,method = "number")                                                                                "Me
```

1.00	0.05	-0.21	-0.06	-0.05	0.08	0.17	0.20	0.19
0.05	1.00	0.13	0.05	0.01	0.02	0.07	0.11	0.10
-0.21	0.13	1.00	0.04	0.17	0.15	-0.30	-0.30	-0.29
-0.06	0.05	0.04	1.00	-0.10	0.00	-0.27	-0.23	-0.24
-0.05	0.01	0.17	-0.10	1.00	0.12	-0.18	-0.18	-0.19
0.08	0.02	0.15	0.00	0.12	1.00	-0.12	-0.20	-0.21
0.17	0.07	-0.30	-0.27	-0.18	-0.12	1.00	0.90	0.89
0.20	0.11	-0.30	-0.23	-0.18	-0.20	0.90	1.00	0.97



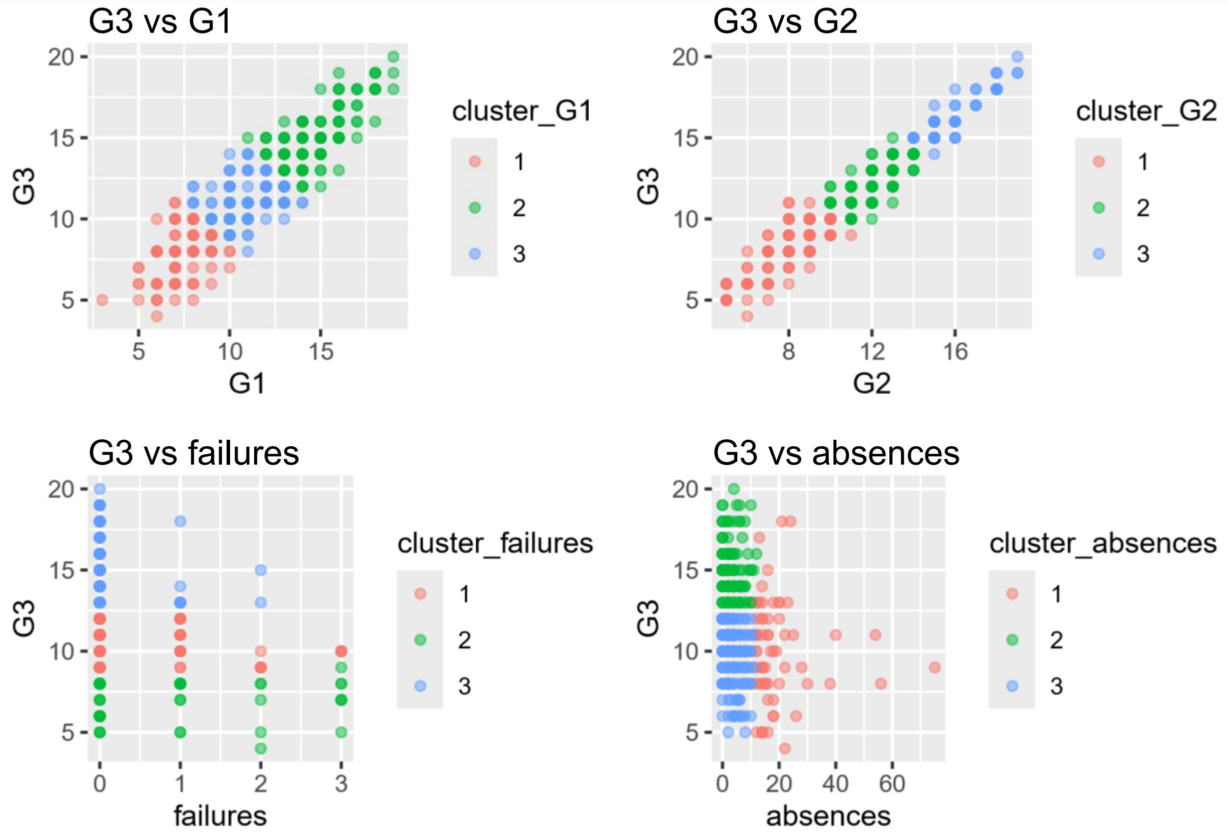
```
# Create the scatter plots
p1 = ggplot(reduced.data, aes(x=G1, y=G3)) + geom_point(alpha=0.5) +
   labs(title="G3 vs G1", x="G1", y="G3")

p2 = ggplot(reduced.data, aes(x=G2, y=G3)) + geom_point(alpha=0.5) +
   labs(title="G3 vs G2", x="G2", y="G3")

p3 = ggplot(reduced.data, aes(x=failures, y=G3)) + geom_point(alpha=0.5) +
   labs(title="G3 vs failures", x="failures", y="G3")
```

```
p4 = ggplot(reduced.data, aes(x=absences, y=G3)) + geom_point(alpha=0.5) +
   labs(title="G3 vs absences", x="absences", y="G3") grid.arrange(p1, p2, p3, p4,

ncol=2)
```



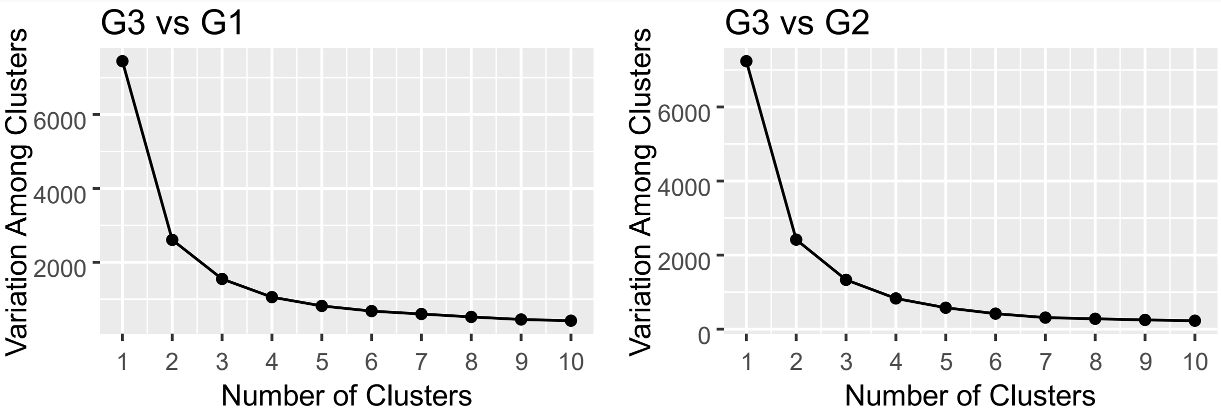
```
k = 3 #set 3 clusters


# G3 vs G1

kmeans_G1 = kmeans(reduced.data[, c("G1", "G3")], centers = k)
reduced.data$cluster_G1 = as.factor(kmeans_G1$cluster) p11 =
ggplot(reduced.data, aes(x=G1, y=G3, color=cluster_G1)) +
geom_point(alpha=0.5) +
   labs(title="G3 vs G1", x="G1", y="G3")

# G3 vs G2
kmeans_G2 = kmeans(reduced.data[, c("G2", "G3")], centers = k)
reduced.data$cluster_G2 = as.factor(kmeans_G2$cluster) p22 =
ggplot(reduced.data, aes(x=G2, y=G3, color=cluster_G2)) +
geom_point(alpha=0.5) +
   labs(title="G3 vs G2", x="G2", y="G3")

# G3 vs failures
```

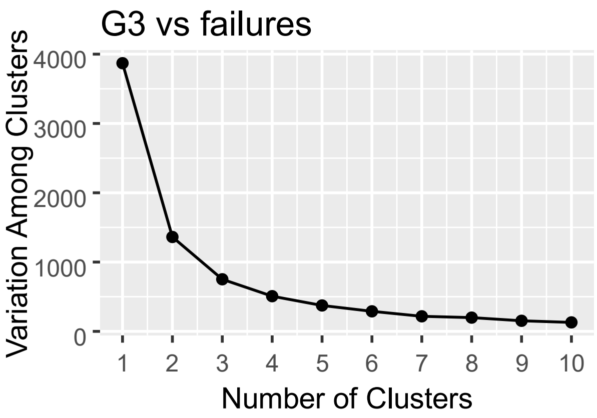
```
kmeans_failures = kmeans(reduced.data[, c("failures", "G3")], centers = k)
reduced.data$cluster_failures = as.factor(kmeans_failures$cluster) p33 = ggplot(reduced.data,
aes(x=failures, y=G3, color=cluster_failures)) + geom_point(alpha=0.5) +
   labs(title="G3 vs failures", x="failures", y="G3")

# G3 vs absences

kmeans_absences = kmeans(reduced.data[, c("absences", "G3")], centers = k)
reduced.data$cluster_absences = as.factor(kmeans_absences$cluster) p44 =
ggplot(reduced.data, aes(x=absences, y=G3, color=cluster_absences)) + geom_point(alpha=0.5)
+
   labs(title="G3 vs absences", x="absences", y="G3") grid.arrange(p11, p22, p33, p44,
ncol=2)
```

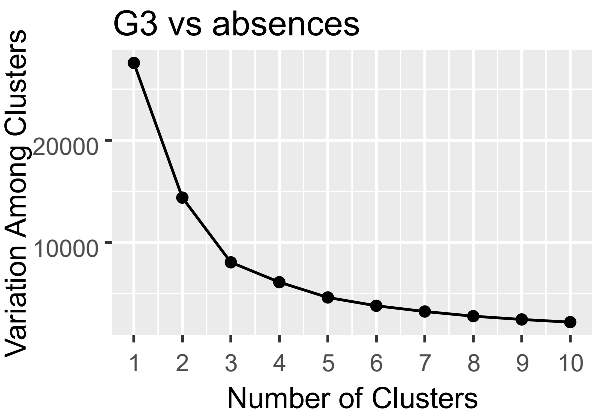


```
# calculate within cluster sum of squares wss = function(reduced.data, max_clusters)
{ wss_values = numeric(max_clusters) for (k in 1:max_clusters) { kmeans_result =
kmeans(reduced.data, centers = k, nstart = 20) wss_values[k] =
kmeans_result$tot.withinss
   }
   return(wss_values)
}
```

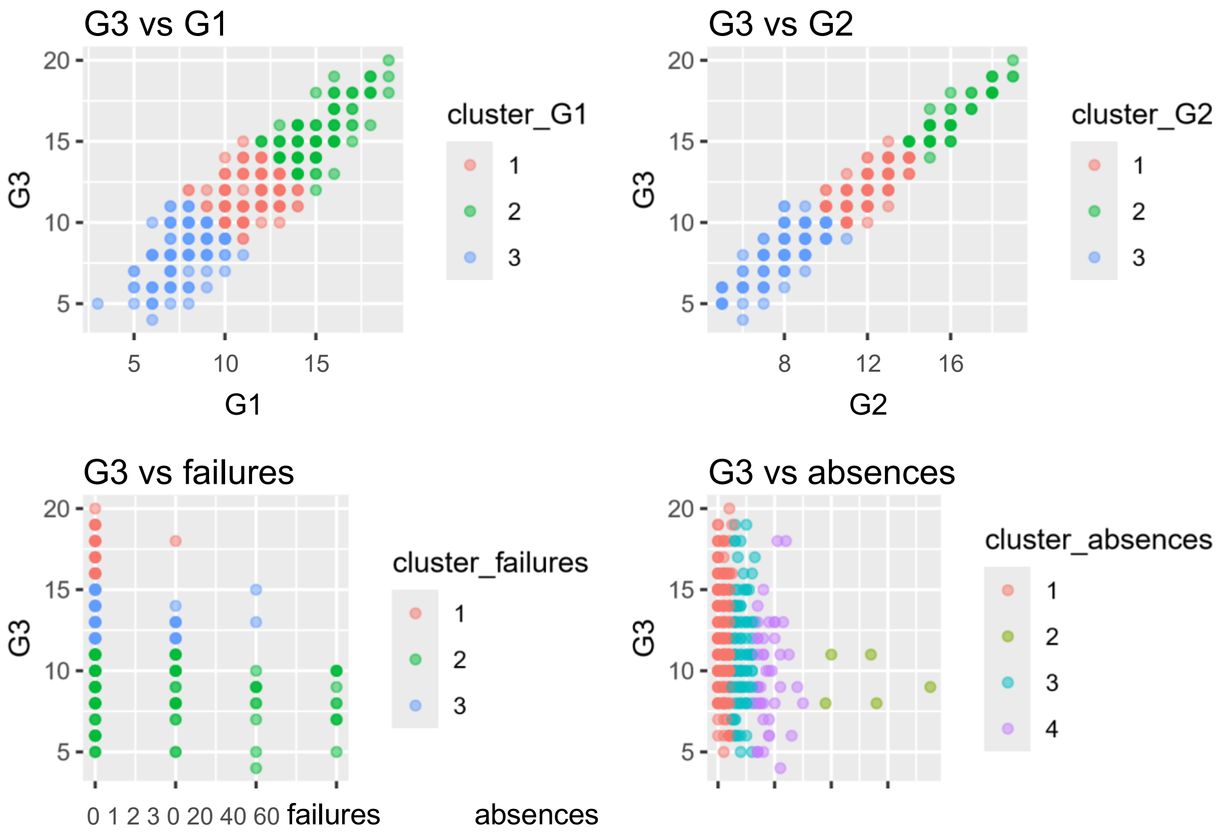
```r
#set a max amount of clusters max_clusters = 10

#find the variance for each set of variables

# G3 vs G1

wss_G1 = wss(reduced.data[, c("G1", "G3")], max_clusters)

# G3 vs G2

wss_G2 = wss(reduced.data[, c("G2", "G3")], max_clusters)

# G3 vs failures

wss_failures = wss(reduced.data[, c("failures", "G3")], max_clusters)

# G3 vs absences

wss_absences = wss(reduced.data[, c("absences", "G3")], max_clusters)

# Create Elbow plots
elbow_plot = function(wss_values, title) { data.frame(k =
   1:max_clusters, wss = wss_values) %>%
      ggplot(aes(x = k, y = wss)) +
      geom_line() + geom_point() +

      labs(title = title, x = "Number of Clusters", y = "Variation Among Clusters") + scale_x_continuous(breaks =
      1:max_clusters)
}

#plot the elbow plots
p1_elbow = elbow_plot(wss_G1, "G3 vs G1") p2_elbow =
elbow_plot(wss_G2, "G3 vs G2")
p3_elbow = elbow_plot(wss_failures, "G3 vs failures") p4_elbow =
elbow_plot(wss_absences, "G3 vs absences") grid.arrange(p1_elbow,
p2_elbow, p3_elbow, p4_elbow, ncol=2)
```







```
k1 = 3 k2
= 3 k3 = 3
k4 = 4


# G3 vs G1
kmeans_G1 = kmeans(reduced.data[, c("G1", "G3")], centers = k1)
reduced.data$cluster_G1 = as.factor(kmeans_G1$cluster) p1 =
ggplot(reduced.data, aes(x=G1, y=G3, color=cluster_G1)) +
geom_point(alpha=0.5) +
   labs(title="G3 vs G1", x="G1", y="G3")

# G3 vs G2
kmeans_G2 = kmeans(reduced.data[, c("G2", "G3")], centers = k2)
reduced.data$cluster_G2 = as.factor(kmeans_G2$cluster) p2 =
ggplot(reduced.data, aes(x=G2, y=G3, color=cluster_G2)) +
geom_point(alpha=0.5) +
   labs(title="G3 vs G2", x="G2", y="G3")

# G3 vs failures

kmeans_failures = kmeans(reduced.data[, c("failures", "G3")], centers = k3)
reduced.data$cluster_failures = as.factor(kmeans_failures$cluster) p3 = ggplot(reduced.data,
aes(x=failures, y=G3, color=cluster_failures)) + geom_point(alpha=0.5) +
   labs(title="G3 vs failures", x="failures", y="G3")
# G3 vs absences

kmeans_absences = kmeans(reduced.data[, c("absences", "G3")], centers = k4)

reduced.data$cluster_absences = as.factor(kmeans_absences$cluster) p4 =

ggplot(reduced.data, aes(x=absences, y=G3, color=cluster_absences)) +
   geom_point(alpha=0.5) + labs(title="G3 vs absences", x="absences", y="G3")

grid.arrange(p1,p2,p3,p4)
```


G3 vs G1
G3
20
15
10
5
5
10
15
G1
cluster_G1
1
2
3
G3 vs G2
G3
20
15
10
5
8
12
16
G2
cluster_G2
1
2
3
G3 vs failures
G3
20
15
10
5
0 1 2 3 0 20 40 60 failures
absences
cluster_failures
1
2
3
G3 vs absences
G3
20
15
10
5
cluster_absences
1
2
3
4