

Prediction of Secondary School Student Grades Using UCI Data

Jeong Sukchan

11/21/2021

<Structure of the Report>

1. Introduction

Abstract

1.1. Problem Identification

1.2. Goal

1.3. Key Steps

1.4. Dataset

1.4.1. Installing necessary packages

1.4.2. Data collection

1.4.3. Dataset Information

2. Method and Analysis

2.1. Data Preparation

2.2. EDA(Exploratory Data Analysis)

2.2.1. Overview of the Dataset

2.2.2. Data Analysis and Visualization

2.3. Data Pre-processing_Data Modification and Cleaning

2.3.1. Checking NA (Null Values)

2.3.2. Checking Outliers

2.3.3. Data Partition

3. Modelling

3.1. Model 1: Guessing Model

3.2. Model 2: Logistic regression Model

3.3. Model 3: Simplified Logistic Regression with Significant Variables

3.4. Model 4: Cross-validated Decision Tree

3.5. Model 5: Cross-validated KNN Model

3.6. Model 6: Random Forest Model

4. Results

5. Conclusion

1. Introduction

Summary

In this paper, we analyze a dataset at UCI website. We collect a data named “student-mat” and get information from that website. The goal of this paper is to predict students’ final performance result of either pass or fail. The reason for this goal is that educators might help academically poor students who has higher probability to fail in advance. To achieve the goal, the aim of this paper is to build a good model with high accuracy result for binary classification data approaching some machine learning algorithms.

In the section 2(Method and Analysis), we overview the dataset and analyze relation among variables. From data analysis we find out directly influential variables that affect the result of the dependent variable. These are sex, age, address, famsize, Medu, Fedu, Mjob, Fjob, traveltime, studytime, failures, schoolsup, paid, higher, romantic, goout, G1 and G2. They are directly correlated with G3 compared to other variables. In addition, we find out indirectly influential variables. These are famsup, school, internet, Dalc, Walc, freetime and gauardian variables. They are correlated with the influential variables that is correlated with G3 directly. Therefore, excluded variables are Pstatus, reason, famsup, activities, nursery, famrel, health and absences variables that are less correlated with the dependent variable directly and indirectly. Before builing up several algorithms, we modify our dataset by checking NA and outliers; by splitting it into trainset and testset.

In the section 3(Modelling), We approach some algorithms such as Guessing, Regression, Decision Trees, KNN and Random Forest method training each model with trainset and predicting with testset. The models are evaluated by accuracy instrument. In the section 4(Results), the result table is shown on which we select the best optimal model. In the section 5(Conclusion), we conclude by presenting summary, potential impact, limitation of the report and future work.

1.1. Problem Identification

It is no doubt that students and educators have in common regarding getting a good grade. Especially, educators may wonder how to help out students who have the high probability to fail the exam in advance. Then, which factors are critical to predict their grades? Which model would tell their final grade well before taking the final exam?

1.2. Goal

The goal is to predict students' final performance result of either pass or fail. To achieve the goal, the aim of this paper is to build a good model with a high accuracy results for binary data.

1.3. Key Steps

We will go through the 4 steps.

- Data import and preparation
- Data Analysis and data pre-process
- Modelling
- Evaluation

1.4. Dataset

The reason to choose our dataset is the accessibility which we can get from UCI Machine Learning Repository easily.

1.4.1. Installing necessary packages

Before we start importing our dataset, we will install the necessary packages that we need for analysis and modelling.

```
# Create package to need for our report
pkg<-c("tidyverse", "dslabs","dplyr","ggplot2","caret","data.table","lattice","readr","magrittr",
"skimr","Hmisc","psych","gridExtra","doBy", "corrplot", "pheatmap", "ROCR", "gplots", "irr", "jani
tor", "PerformanceAnalytics", "rpart", "rattle")

# Create package to need for installation
new_pkg<-pkg[!(pkg %in% rownames(installed.packages()))]

# if there are packages uninstalled, install the package at a time
if(length(new_pkg))install.packages(new_pkg, dependencies = TRUE)

# Road the package at a time
suppressMessages(sapply(pkg, require, character.only=TRUE)) # suppress the complicated messages.
```

```
## Warning: package 'skimr' was built under R version 4.1.1
```

```
## Warning: package 'doBy' was built under R version 4.1.1
```

```
## Warning: package 'ROCR' was built under R version 4.1.1
```

```
## Warning: package 'gplots' was built under R version 4.1.1
```

```
## Warning: package 'irr' was built under R version 4.1.1
```

```
## Warning: package 'lpSolve' was built under R version 4.1.1
```

```
## Warning: package 'janitor' was built under R version 4.1.1
```

```
## Warning: package 'PerformanceAnalytics' was built under R version 4.1.1
```

```
## Warning: package 'xts' was built under R version 4.1.1
```

```
## Warning: package 'rattle' was built under R version 4.1.1
```

```
## Warning: package 'bitops' was built under R version 4.1.1
```

##	tidyverse	dsmlabs	dplyr
##	TRUE	TRUE	TRUE
##	ggplot2	caret	data.table
##	TRUE	TRUE	TRUE
##	lattice	readr	magrittr
##	TRUE	TRUE	TRUE
##	skimr	Hmisc	psych
##	TRUE	TRUE	TRUE
##	gridExtra	doBy	corrplot
##	TRUE	TRUE	TRUE
##	pheatmap	ROCR	gplots
##	TRUE	TRUE	TRUE
##	irr	janitor	PerformanceAnalytics
##	TRUE	TRUE	TRUE
##	rpart	rattle	
##	TRUE	TRUE	

1.4.2. Data collection

The dataset were imported from UCI Machine Learning Respository website:

<http://archive.ics.uci.edu/ml/datasets/Student+Performance#>

(<http://archive.ics.uci.edu/ml/datasets/Student+Performance#>)

The file is located at: <http://archive.ics.uci.edu/ml/machine-learning-databases/00320/student.zip>

(<http://archive.ics.uci.edu/ml/machine-learning-databases/00320/student.zip>)

The file name is “student-mat.csv”

```
# Download the dataset in our local computer and load the data

# An example of the path of downloaded and unzipped folder in a local computer: "C:/datascience/r/
projects/performance/data/student/"

# Load the data and store "mat" objective
mat <- read.table("C:/datascience/r/projects/performance/data/student/student-mat.csv", sep=";", header=TRUE)
```

1.4.3. Dataset Information

The dataset information is at the website where our dataset was downloaded. The dataset has 395 observations and 33 variables.

Attribute Information

1 school - student's school (binary: 'GP' - Gabriel Pereira or 'MS' - Mousinho da Silveira)

2 sex - student's sex (binary: 'F' - female or 'M' - male)

3 age - student's age (numeric: from 15 to 22)

4 address - student's home address type (binary: 'U' - urban or 'R' - rural)

5 famsize - family size (binary: 'LE3' - less or equal to 3 or 'GT3' - greater than 3)

6 Pstatus - parent's cohabitation status (binary: 'T' - living together or 'A' - apart)

7 Medu - mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 - 5th to 9th grade, 3 - secondary education or 4 - higher education)

8 Fedu - father's education (numeric: 0 - none, 1 - primary education (4th grade), 2 - 5th to 9th grade, 3 - secondary education or 4 - higher education)

9 Mjob - mother's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')

10 Fjob - father's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')

11 reason - reason to choose this school (nominal: close to 'home', school 'reputation', 'course' preference or 'other')

12 guardian - student's guardian (nominal: 'mother', 'father' or 'other')

13 travelttime - home to school travel time (numeric: 1 - <15 min., 2 - 15 to 30 min., 3 - 30 min. to 1 hour, or 4 - >1 hour)

14 studytime - weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)

15 failures - number of past class failures (numeric: n if 0<=n<3, else 4)

16 schoolsup - extra educational support (binary: yes or no)

17 famsup - family educational support (binary: yes or no)

18 paid - extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)

19 activities - extra-curricular activities (binary: yes or no)

20 nursery - attended nursery school (binary: yes or no)

21 higher - wants to take higher education (binary: yes or no)

22 internet - Internet access at home (binary: yes or no)

23 romantic - with a romantic relationship (binary: yes or no)

24 famrel - quality of family relationships (numeric: from 1 - very bad to 5 - excellent)

25 freetime - free time after school (numeric: from 1 - very low to 5 - very high)

26 goout - going out with friends (numeric: from 1 - very low to 5 - very high)

27 Dalc - workday alcohol consumption (numeric: from 1 - very low to 5 - very high)

28 Walc - weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)

29 health - current health status (numeric: from 1 - very bad to 5 - very good)

30 absences - number of school absences (numeric: from 0 to 93)

31 G1 - first period grade (numeric: from 0 to 20)

31 G2 - second period grade (numeric: from 0 to 20)

32 G3 - final grade (numeric: from 0 to 20, output target)

No missing value

Dependent variable

Among the variables, G3 will be the dependent variable which is a final score after G1 and G2 evaluations. The grading point scale is 20 in which 0 is the lowest and 20 is the perfect.

Data Type

Attributes are either character(binary or nominal) or numeric.

- Binary character: school, sex, address, famsize, Pstatus, schoolsup, famsup, paid, activities, nursery, higher, internet, romantic
- Numeric variables: age, Medu, Fedu, travelttime, studytime, failures, famrel, freetime, goout, Dalc, Walc, health, absences, G1, G2, G3
- Nominal character (with five levels): Mjob, Fjob, reason, guardian

2. Method and Analysis

In this section, the process and technique are the following.

- Data preparation: We will modify our data into appropriate several forms so that they can be applied to specific algorithms.
- Overviewing of the data: We will try to get a whole picture of properties and relations of variables.
- Data pre-processing: We will check Null Values (NA) and outliers, and split datasets into trainsets and testsets before modelling.

2.1. Data Preparation

The “mat” dataset is ready for analysis. But we will modify it for specific uses by creating several datasets. We will use them accordingly in EDA and modelling. Further we will clean and modify more in the “Data Pre-processing” section and other sections as necessary.

Create a new dataset called "new_mat" for changing numeric G1, G2 and G3 variables into binary variables. Let's define G3>= 8 as "1" meaning "pass" and otherwise, "0" meaning "fail". The new_mat dataset has a binary dependant variable whereas the mat dataset has a numeric dependant variable.

We will use both accordingly.

```
new_mat<-mat
```

```
new_mat$G3<-ifelse(new_mat$G3>=8, 1, 0)
```

```
new_mat$G1<-ifelse(new_mat$G1>=8, 1, 0)
```

```
new_mat$G2<-ifelse(new_mat$G2>=8, 1, 0)
```

Create a factor dataset called "fnew_mat".

```
fnew_mat<-new_mat
```

```
fnew_mat$school<-factor(mat$school, levels=c("GP","MS"))
```

```
fnew_mat$sex <- factor(mat$sex, levels=c("F","M"))
```

```
fnew_mat$age <- factor(mat$age, levels=c("15","16","17","18","19","20","21","22"))
```

```
fnew_mat$address <- factor(mat$address, levels=c("U","R"))
```

```
fnew_mat$famsize <- factor(mat$famsize, levels=c("LE3","GT3"))
```

```
fnew_mat$Pstatus <- factor(mat$Pstatus, levels=c("T","A"))
```

```
fnew_mat$Medu <- factor(mat$Medu, levels=c("0","1","2","3","4"))
```

```
fnew_mat$Fedu <- factor(mat$Fedu, levels=c("0","1","2","3","4"))
```

```
fnew_mat$Mjob <- factor(mat$Mjob, levels=c("teacher","health","services","at_home","other"))
```

```
fnew_mat$Fjob <- factor(mat$Fjob, levels=c("teacher","health","services","at_home","other"))
```

```
fnew_mat$reason <- factor(mat$reason, levels=c("home","reputation","course","other"))
```

```
fnew_mat$guardian <- factor(mat$guardian, levels=c("mother","father","other"))
```

```
fnew_mat$traveltime <- factor(mat$traveltime, levels=c("1","2","3","4"))
```

```
fnew_mat$studytime <- factor(mat$studytime, levels=c("1","2","3","4"))
```

```
fnew_mat$failures <- factor(mat$failures, levels=c("0","1","2","3","4"))
```

```
fnew_mat$schoolsup <- factor(mat$schoolsup, levels=c("yes","no"))
```

```
fnew_mat$famsup <- factor(mat$famsup, levels=c("yes","no"))
```

```
fnew_mat$paid <- factor(mat$paid, levels=c("yes","no"))
```

```
fnew_mat$activities <- factor(mat$activities, levels=c("yes","no"))
```

```
fnew_mat$nursery <- factor(mat$nursery, levels=c("yes","no"))
```

```
fnew_mat$higher <- factor(mat$higher, levels=c("yes","no"))
```

```
fnew_mat$internet <- factor(mat$internet, levels=c("yes","no"))
```

```
fnew_mat$romantic <- factor(mat$romantic, levels=c("yes","no"))
```

```
fnew_mat$famrel <- factor(mat$famrel, levels=c("1","2","3","4","5"))
```

```
fnew_mat$freetime <- factor(mat$freetime, levels=c("1","2","3","4","5"))
```

```
fnew_mat$goout <- factor(mat$goout, levels=c("1","2","3","4","5"))
```

```
fnew_mat$Dalc <- factor(mat$Dalc, levels=c("1","2","3","4","5"))
```

```
fnew_mat$Walc <- factor(mat$Walc, levels=c("1","2","3","4","5"))
```

```
fnew_mat$health <- factor(mat$health, levels=c("1","2","3","4","5"))
```

```
fnew_mat$absences<-ifelse(new_mat$absence<2, 0,
```

```
ifelse(new_mat$absences>=3 & new_mat$absences<10, 1,
```

```
ifelse(new_mat$absences >=10 & new_mat$absences <20, 2,
```

```
ifelse(new_mat$absences >=20 & new_mat$absences <40, 3, 4
```

```
)))
```

```
fnew_mat$absences<-factor(new_mat$absences, levels=c("0","1","2","3","4"))
```

```
fnew_mat$G1<-factor(new_mat$G1, levels=c("0","1"), labels=c("fail","pass"))
```

```
fnew_mat$G2<-factor(new_mat$G2, levels=c("0","1"), labels=c("fail","pass"))
```

```
fnew_mat$G3<-factor(new_mat$G3, levels=c("0","1"), labels=c("fail","pass"))
```

Create a binomial dataset where G3 is binomial dependent variable and others are factors.

```

bi<-fnew_mat
bi$G1<-new_mat$G1
bi$G1<-as.factor(new_mat$G1)
bi$G2<-new_mat$G2
bi$G2<-as.factor(new_mat$G2)
bi$G3<-new_mat$G3
bi$G3<-as.factor(new_mat$G3)

# Create a numeric dataset called "num" for analyzing correlation among numeric attributes
num<-dplyr::select(mat, age, Medu, Fedu, traveltime, studytime, failures, famrel, freetime, goout,
Dalc, Walc,health,absences, G1, G2, G3)

# Create numeric dataset called "allnum" for all variables
allnum<-mat
allnum$school<-as.numeric(ifelse(fnew_mat$school=="GP", 0, 1))
allnum$sex<-as.numeric(ifelse(fnew_mat$sex=="F", 1, 0))
allnum$address<-as.numeric(ifelse(fnew_mat$address=="U", 1, 0))
allnum$famsize<-as.numeric(ifelse(fnew_mat$famsize=="LE3", 0, 1))
allnum$Pstatus<-as.numeric(ifelse(fnew_mat$Pstatus=="T", 1, 0))
allnum$Mjob<-as.numeric(ifelse(fnew_mat$Mjob=="teacher", 4,
                                ifelse(fnew_mat$Mjob=="health", 3,
                                        ifelse(fnew_mat$Mjob=="services", 2,
                                                ifelse(fnew_mat$Mjob=="at_home", 1, 0)))))
allnum$Fjob<-as.numeric(ifelse(fnew_mat$Fjob=="teacher", 4,
                                ifelse(fnew_mat$Fjob=="health", 3,
                                        ifelse(fnew_mat$Fjob=="services", 2,
                                                ifelse(fnew_mat$Fjob=="at_home", 1, 0)))))
allnum$reason<-as.numeric(ifelse(fnew_mat$reason=="home", 3,
                                ifelse(fnew_mat$reason=="reputation", 2,
                                        ifelse(fnew_mat$reason=="course", 1, 0)))))
allnum$guardian<-as.numeric(ifelse(fnew_mat$guardian=="mother", 2,
                                ifelse(fnew_mat$guardian=="father", 1, 0)))
allnum$schoolsup<-as.numeric(ifelse(fnew_mat$schoolsup=="yes", 1, 0))
allnum$famsup<-as.numeric(ifelse(fnew_mat$famsup=="yes", 1, 0))
allnum$paid<-as.numeric(ifelse(fnew_mat$paid=="yes", 1, 0))
allnum$activities<-as.numeric(ifelse(fnew_mat$activities=="yes", 1, 0))
allnum$nursery<-as.numeric(ifelse(fnew_mat$nursery=="yes", 1, 0))
allnum$higher<-as.numeric(ifelse(fnew_mat$higher=="yes", 1, 0))
allnum$internet<-as.numeric(ifelse(fnew_mat$internet=="yes", 1, 0))
allnum$romantic<-as.numeric(ifelse(fnew_mat$romantic=="yes", 1, 0))

# Create a new dataset called "ex_mat, ex_num, ex_new_mat, ex_fnew_mat" for removing G1 and G2 because of too obviously decisive indicators of the dependent variable and because of a wider application in other cases where many schools might not have the results of the previous tests
ex_num<-num[, c(1:13,16)]
ex_new_mat<-new_mat[, c(1:30,33)]
ex_bi<-bi[, c(1:30,33)]

```

2.2. EDA (Exploratory Data Analysis)

2.2.1. Overview of the Dataset

Let's take a look an overall picture of the “mat” dataset through statistical analysis and get some insights from it. We will use different packages for statistical analysis because each package has the unique way with different statistical displays.

a. Statistical analysis

```
str(mat)
```

```
## 'data.frame':   395 obs. of  33 variables:
## $ school      : chr  "GP" "GP" "GP" "GP" ...
## $ sex         : chr  "F" "F" "F" "F" ...
## $ age         : int   18 17 15 15 16 16 16 17 15 15 ...
## $ address     : chr  "U" "U" "U" "U" ...
## $ famsize     : chr  "GT3" "GT3" "LE3" "GT3" ...
## $ Pstatus     : chr  "A" "T" "T" "T" ...
## $ Medu        : int   4 1 1 4 3 4 2 4 3 3 ...
## $ Fedu        : int   4 1 1 2 3 3 2 4 2 4 ...
## $ Mjob        : chr  "at_home" "at_home" "at_home" "health" ...
## $ Fjob        : chr  "teacher" "other" "other" "services" ...
## $ reason      : chr  "course" "course" "other" "home" ...
## $ guardian    : chr  "mother" "father" "mother" "mother" ...
## $ travelttime : int   2 1 1 1 1 1 1 2 1 1 ...
## $ studytime   : int   2 2 2 3 2 2 2 2 2 2 ...
## $ failures    : int   0 0 3 0 0 0 0 0 0 0 ...
## $ schoolsup   : chr  "yes" "no" "yes" "no" ...
## $ famsup      : chr  "no" "yes" "no" "yes" ...
## $ paid        : chr  "no" "no" "yes" "yes" ...
## $ activities  : chr  "no" "no" "no" "yes" ...
## $ nursery     : chr  "yes" "no" "yes" "yes" ...
## $ higher      : chr  "yes" "yes" "yes" "yes" ...
## $ internet    : chr  "no" "yes" "yes" "yes" ...
## $ romantic    : chr  "no" "no" "no" "yes" ...
## $ famrel      : int   4 5 4 3 4 5 4 4 4 5 ...
## $ freetime    : int   3 3 3 2 3 4 4 1 2 5 ...
## $ goout       : int   4 3 2 2 2 2 4 4 2 1 ...
## $ Dalc        : int   1 1 2 1 1 1 1 1 1 1 ...
## $ Walc        : int   1 1 3 1 2 2 1 1 1 1 ...
## $ health      : int   3 3 3 5 5 5 3 1 1 5 ...
## $ absences    : int   6 4 10 2 4 10 0 6 0 0 ...
## $ G1          : int   5 5 7 15 6 15 12 6 16 14 ...
## $ G2          : int   6 5 8 14 10 15 12 5 18 15 ...
## $ G3          : int   6 6 10 15 10 15 11 6 19 15 ...
```

There are 395 observations of 33 variables. We can see the content of levels for character variable.

b. Statistical analysis

```
Hmisc::describe(mat)
```

```

## mat
##
## 33 Variables      395 Observations
## -----
## school
##      n missing distinct
##    395      0      2
##
## Value      GP      MS
## Frequency  349    46
## Proportion 0.884 0.116
## -----
## sex
##      n missing distinct
##    395      0      2
##
## Value      F      M
## Frequency  208   187
## Proportion 0.527 0.473
## -----
## age
##      n missing distinct      Info      Mean      Gmd
##    395      0      8    0.948    16.7    1.411
##
## lowest : 15 16 17 18 19, highest: 18 19 20 21 22
##
## Value      15      16      17      18      19      20      21      22
## Frequency    82    104     98     82     24      3      1      1
## Proportion 0.208 0.263 0.248 0.208 0.061 0.008 0.003 0.003
## -----
## address
##      n missing distinct
##    395      0      2
##
## Value      R      U
## Frequency    88   307
## Proportion 0.223 0.777
## -----
## famsize
##      n missing distinct
##    395      0      2
##
## Value      GT3      LE3
## Frequency  281   114
## Proportion 0.711 0.289
## -----
## Pstatus
##      n missing distinct
##    395      0      2
##
## Value      A      T
## Frequency   41   354

```

```

## Proportion 0.104 0.896
## -----
## Medu
##      n missing distinct      Info      Mean      Gmd
##    395         0         5    0.927    2.749    1.213
##
## lowest : 0 1 2 3 4, highest: 0 1 2 3 4
##
## Value      0      1      2      3      4
## Frequency   3     59    103     99    131
## Proportion 0.008 0.149 0.261 0.251 0.332
## -----
## Fedu
##      n missing distinct      Info      Mean      Gmd
##    395         0         5    0.936    2.522    1.216
##
## lowest : 0 1 2 3 4, highest: 0 1 2 3 4
##
## Value      0      1      2      3      4
## Frequency   2     82    115    100     96
## Proportion 0.005 0.208 0.291 0.253 0.243
## -----
## Mjob
##      n missing distinct
##    395         0         5
##
## lowest : at_home health other services teacher
## highest: at_home health other services teacher
##
## Value      at_home health other services teacher
## Frequency   59      34    141    103      58
## Proportion  0.149   0.086  0.357   0.261   0.147
## -----
## Fjob
##      n missing distinct
##    395         0         5
##
## lowest : at_home health other services teacher
## highest: at_home health other services teacher
##
## Value      at_home health other services teacher
## Frequency   20      18    217    111      29
## Proportion  0.051   0.046  0.549   0.281   0.073
## -----
## reason
##      n missing distinct
##    395         0         4
##
## Value      course      home      other reputation
## Frequency   145      109      36      105
## Proportion  0.367   0.276   0.091   0.266
## -----

```

```

## guardian
##      n missing distinct
##    395      0      3
##
## Value      father mother  other
## Frequency    90    273    32
## Proportion 0.228 0.691 0.081
## -----
## traveltime
##      n missing distinct      Info      Mean      Gmd
##    395      0      4    0.704    1.448    0.6406
##
## Value      1      2      3      4
## Frequency  257   107   23    8
## Proportion 0.651 0.271 0.058 0.020
## -----
## studytime
##      n missing distinct      Info      Mean      Gmd
##    395      0      4    0.85    2.035    0.8772
##
## Value      1      2      3      4
## Frequency  105   198   65   27
## Proportion 0.266 0.501 0.165 0.068
## -----
## failures
##      n missing distinct      Info      Mean      Gmd
##    395      0      4    0.505    0.3342    0.5642
##
## Value      0      1      2      3
## Frequency  312   50   17   16
## Proportion 0.790 0.127 0.043 0.041
## -----
## schoolsup
##      n missing distinct
##    395      0      2
##
## Value      no  yes
## Frequency  344   51
## Proportion 0.871 0.129
## -----
## famsup
##      n missing distinct
##    395      0      2
##
## Value      no  yes
## Frequency  153  242
## Proportion 0.387 0.613
## -----
## paid
##      n missing distinct
##    395      0      2
##

```

```

## Value          no   yes
## Frequency      214  181
## Proportion 0.542 0.458
## -----
## activities
##           n missing distinct
##        395         0         2
##
## Value          no   yes
## Frequency      194  201
## Proportion 0.491 0.509
## -----
## nursery
##           n missing distinct
##        395         0         2
##
## Value          no   yes
## Frequency       81  314
## Proportion 0.205 0.795
## -----
## higher
##           n missing distinct
##        395         0         2
##
## Value          no   yes
## Frequency       20  375
## Proportion 0.051 0.949
## -----
## internet
##           n missing distinct
##        395         0         2
##
## Value          no   yes
## Frequency       66  329
## Proportion 0.167 0.833
## -----
## romantic
##           n missing distinct
##        395         0         2
##
## Value          no   yes
## Frequency      263  132
## Proportion 0.666 0.334
## -----
## famrel
##           n missing distinct      Info      Mean      Gmd
##        395         0         5      0.855      3.944      0.9204
##
## lowest : 1 2 3 4 5, highest: 1 2 3 4 5
##
## Value          1      2      3      4      5
## Frequency       8     18     68    195    106

```

```

## Proportion 0.020 0.046 0.172 0.494 0.268
## -----
## freetime
##      n missing distinct      Info      Mean      Gmd
##    395      0        5    0.907    3.235    1.085
##
## lowest : 1 2 3 4 5, highest: 1 2 3 4 5
##
## Value      1      2      3      4      5
## Frequency   19     64    157    115    40
## Proportion 0.048 0.162 0.397 0.291 0.101
## -----
## goout
##      n missing distinct      Info      Mean      Gmd
##    395      0        5    0.934    3.109    1.236
##
## lowest : 1 2 3 4 5, highest: 1 2 3 4 5
##
## Value      1      2      3      4      5
## Frequency   23    103    130     86    53
## Proportion 0.058 0.261 0.329 0.218 0.134
## -----
## Dalc
##      n missing distinct      Info      Mean      Gmd
##    395      0        5    0.652    1.481    0.7524
##
## lowest : 1 2 3 4 5, highest: 1 2 3 4 5
##
## Value      1      2      3      4      5
## Frequency  276     75     26      9      9
## Proportion 0.699 0.190 0.066 0.023 0.023
## -----
## Walc
##      n missing distinct      Info      Mean      Gmd
##    395      0        5    0.923    2.291    1.409
##
## lowest : 1 2 3 4 5, highest: 1 2 3 4 5
##
## Value      1      2      3      4      5
## Frequency  151     85     80     51     28
## Proportion 0.382 0.215 0.203 0.129 0.071
## -----
## health
##      n missing distinct      Info      Mean      Gmd
##    395      0        5    0.929    3.554    1.534
##
## lowest : 1 2 3 4 5, highest: 1 2 3 4 5
##
## Value      1      2      3      4      5
## Frequency   47     45     91     66    146
## Proportion 0.119 0.114 0.230 0.167 0.370
## -----

```

```

## absences
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    395      0      34    0.968    5.709    7.026    0.0    0.0
##      .25      .50      .75      .90      .95
##      0.0      4.0      8.0     14.0     18.3
##
## lowest :  0  1  2  3  4, highest: 38 40 54 56 75
## -----
## G1
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    395      0      17    0.992    10.91    3.788      6      7
##      .25      .50      .75      .90      .95
##      8      11      13      16      16
##
## lowest :  3  4  5  6  7, highest: 15 16 17 18 19
##
## Value      3      4      5      6      7      8      9     10     11     12     13
## Frequency    1      1      7     24     37     41     31     51     39     35     33
## Proportion 0.003 0.003 0.018 0.061 0.094 0.104 0.078 0.129 0.099 0.089 0.084
##
## Value      14     15     16     17     18     19
## Frequency   30     24     22      8      8      3
## Proportion 0.076 0.061 0.056 0.020 0.020 0.008
## -----
## G2
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    395      0      17    0.992    10.71    4.165     5.0     6.0
##      .25      .50      .75      .90      .95
##      9.0     11.0     13.0     15.0     16.3
##
## lowest :  0  4  5  6  7, highest: 15 16 17 18 19
##
## Value      0      4      5      6      7      8      9     10     11     12     13
## Frequency   13      1     15     14     21     32     50     46     35     41     37
## Proportion 0.033 0.003 0.038 0.035 0.053 0.081 0.127 0.116 0.089 0.104 0.094
##
## Value      14     15     16     17     18     19
## Frequency   23     34     13      5     12      3
## Proportion 0.058 0.086 0.033 0.013 0.030 0.008
## -----
## G3
##      n missing distinct      Info      Mean      Gmd      .05      .10
##    395      0      18    0.992    10.42    4.992     0.0     5.0
##      .25      .50      .75      .90      .95
##      8.0     11.0     14.0     15.6     17.0
##
## lowest :  0  4  5  6  7, highest: 16 17 18 19 20
##
## Value      0      4      5      6      7      8      9     10     11     12     13
## Frequency   38      1      7     15      9     32     28     56     47     31     31
## Proportion 0.096 0.003 0.018 0.038 0.023 0.081 0.071 0.142 0.119 0.078 0.078
##

```

```
## Value      14    15    16    17    18    19    20
## Frequency   27    33    16     6    12     5     1
## Proportion 0.068 0.084 0.041 0.015 0.030 0.013 0.003
## -----
```

We can clearly check the number of missing data, frequency and proportion of levels of each variables. We will analyze some variables in the end of this section.

c. Statistical analysis

```
psych::describe(mat)
```



```
## paid*          -1.98 0.03
## activities*    -2.00 0.03
## nursery*       0.12 0.02
## higher*        14.71 0.01
## internet*      1.16 0.02
## romantic*     -1.51 0.02
## famrel         1.09 0.05
## freetime      -0.33 0.05
## goout          -0.79 0.06
## Dalc           4.65 0.04
## Walc          -0.81 0.06
## health        -1.03 0.07
## absences       21.31 0.40
## G1            -0.71 0.17
## G2             0.59 0.19
## G3             0.37 0.23
```

The table above provides a very comprehensive list of the descriptive statistics for our dataset. As we are just going to scroll across the table, it shows us the variable number, “n” which indicates the number of observations for each variable, “mean” which indicates the average value, “standard deviation”, “median”, “trimmed”, “mad” which means the mean absolute deviation, “max”, “min” which means the minimum value, “range” which means between the two, “skew” means the measure of how skewed our dataset is, “kurtosis” and “se” which indicates the standard error. This statistics are very useful for the numeric variables but not for the categorical ones which are denoted with asterisks in the table.

d. Statistical analysis

```
skimr::skim(mat)
```

Data summary

Name	mat
Number of rows	395
Number of columns	33
<hr/>	
Column type frequency:	
character	17
numeric	16
<hr/>	
Group variables	None


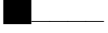



Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
school	0	1	2	2	0	2	0

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
sex	0	1	1	1	0	2	0
address	0	1	1	1	0	2	0
famsize	0	1	3	3	0	2	0
Pstatus	0	1	1	1	0	2	0
Mjob	0	1	5	8	0	5	0
Fjob	0	1	5	8	0	5	0
reason	0	1	4	10	0	4	0
guardian	0	1	5	6	0	3	0
schoolsup	0	1	2	3	0	2	0
famsup	0	1	2	3	0	2	0
paid	0	1	2	3	0	2	0
activities	0	1	2	3	0	2	0
nursery	0	1	2	3	0	2	0
higher	0	1	2	3	0	2	0
internet	0	1	2	3	0	2	0
romantic	0	1	2	3	0	2	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
age	0	1	16.70	1.28	15	16	17	18	22	
Medu	0	1	2.75	1.09	0	2	3	4	4	
Fedu	0	1	2.52	1.09	0	2	2	3	4	
traveltime	0	1	1.45	0.70	1	1	1	2	4	
studytime	0	1	2.04	0.84	1	1	2	2	4	
failures	0	1	0.33	0.74	0	0	0	0	3	
famrel	0	1	3.94	0.90	1	4	4	5	5	
freetime	0	1	3.24	1.00	1	3	3	4	5	
goout	0	1	3.11	1.11	1	2	3	4	5	
Dalc	0	1	1.48	0.89	1	1	1	2	5	
Walc	0	1	2.29	1.29	1	1	2	3	5	

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
health	0	1	3.55	1.39	1	3	4	5	5	
absences	0	1	5.71	8.00	0	0	4	8	75	
G1	0	1	10.91	3.32	3	8	11	13	19	
G2	0	1	10.71	3.76	0	9	11	13	19	
G3	0	1	10.42	4.58	0	8	11	14	20	

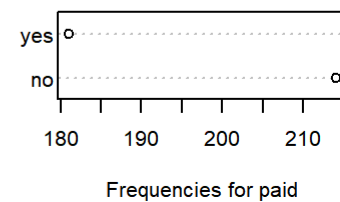
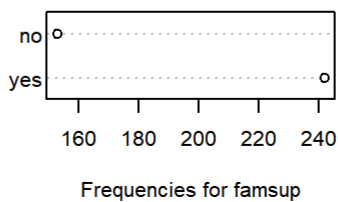
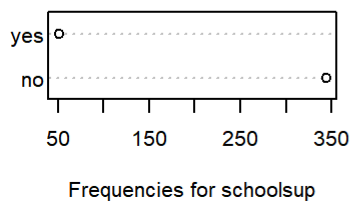
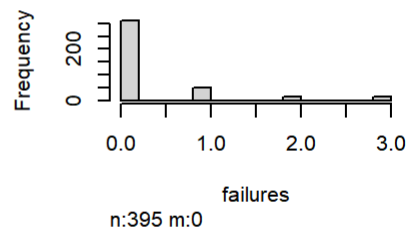
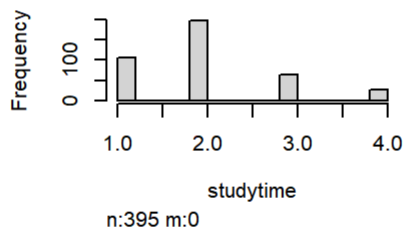
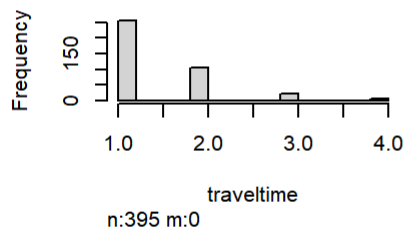
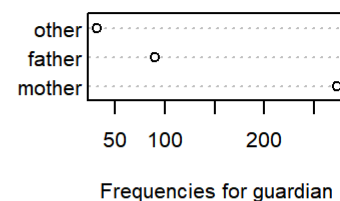
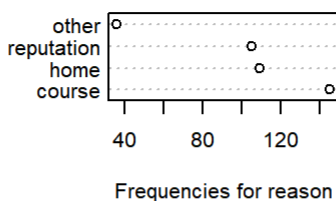
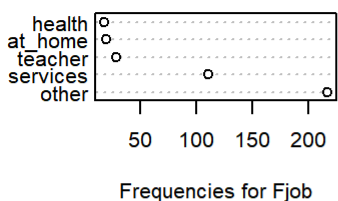
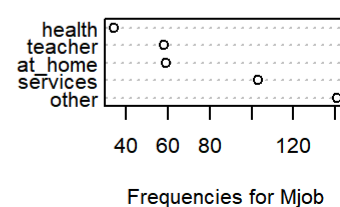
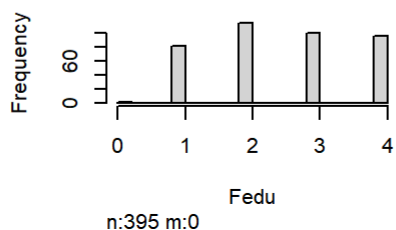
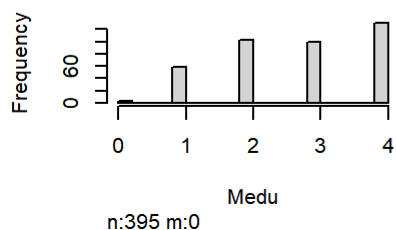
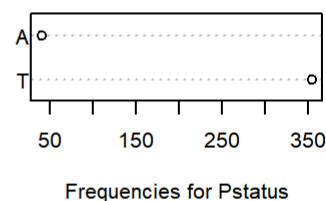
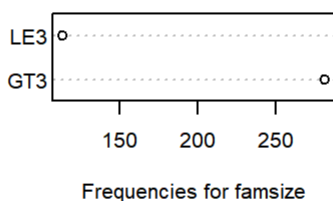
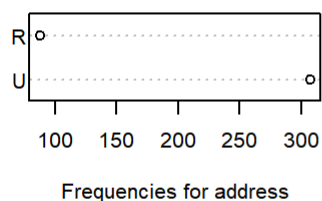
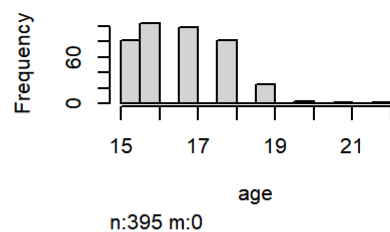
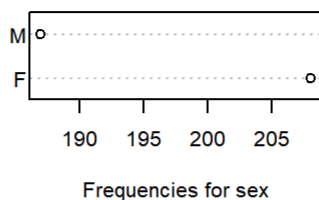
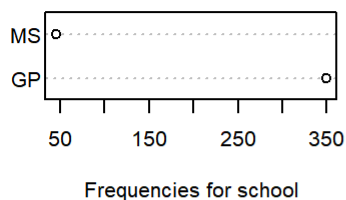
We can quickly assess our data features and types. It goes even deeper by each character variables arranged according to data types. The above two tibbles are grouped by variable types: character and numeric. It gives some information about the number of missing data, completed rate, distribution of the values. It also shows unique values for character variables, whereas it displays distribution such as mean, standard deviation, quantile and a little histogram for numeric variables.

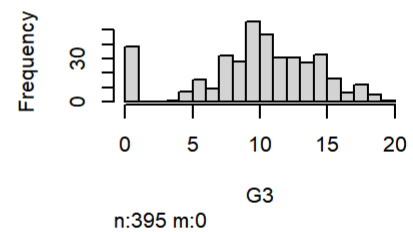
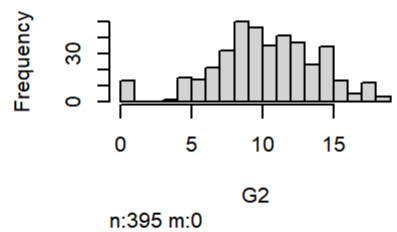
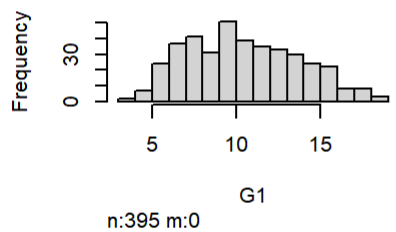
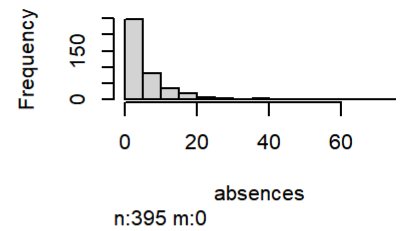
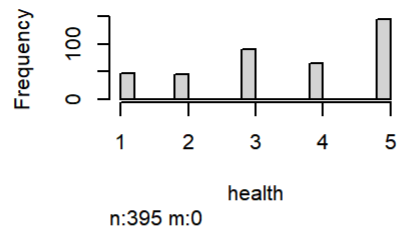
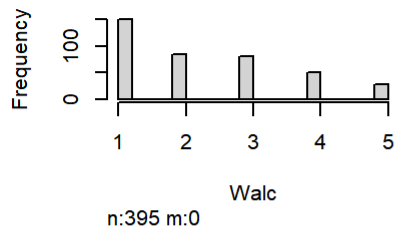
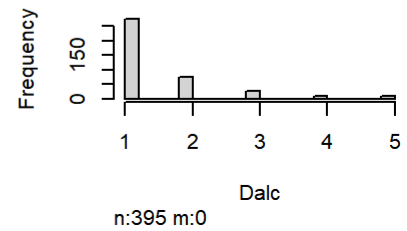
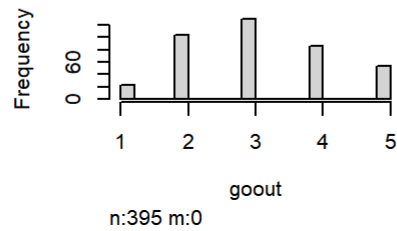
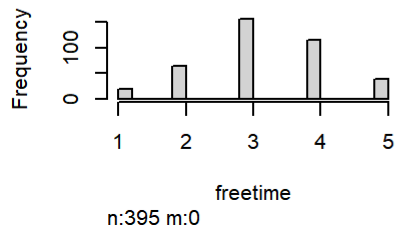
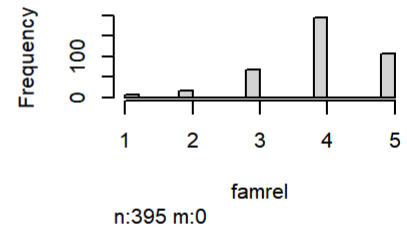
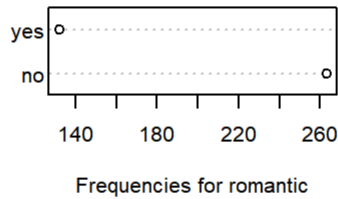
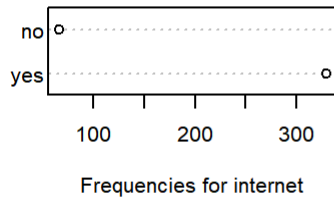
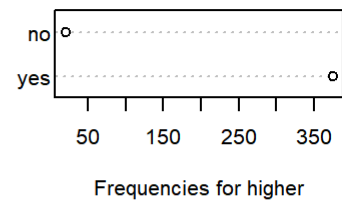
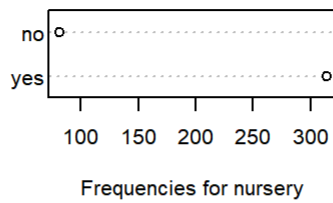
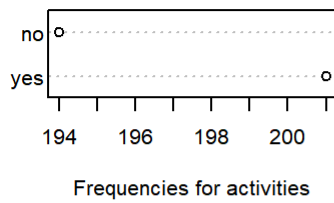
2.2.2. Data Analysis and Visualization

In this section, we will explore the data by asking and answering questions about variables and their relations to get some insights.

a. Show graphs for all variables.

```
# Let's explore overall tendency of each variable displaying multiple histograms
par(mfrow=c(3,3)) # Show 3*3 graphs on each pane
hist.data.frame(mat)
```





- school: Most of the students live in urban areas.

- sex: There are more girls than boys. Hmisc Statistical Analysis tells us proportion of sex: female 0.527; male 0.473.
- age: The most of the students are aged between 15 and 18. From the above statistics in the previous section, there are 29 students older than 18 years old. The average of students' age is 16.7.
- address: Most students live in urban areas rather than in rural areas.
- famsize: More students live with more than 3 family members.
- Pstatus: Most students live with their parents.
- Medu and Fedu: The level of parents' education is widely distributed from primary education to higher education
- Mjob and Fjob: Many parents work in service sections.
- reason: The most reason to choose the school is related to grade such as course and reputation.
- guardian: Most students have guardians. The majority of students think mother is their guardian.
- traveltime: Most students live near their house taking less than 15min.
- studytime: Many students study between 2 and 5 hours a week.
- failures: Most students have not experienced failures of past classes.
- schoolsup: Most students have no extra educational support.
- paid: More students have no extra paid classes in Math.
- activities: More students have extra curricular activities.
- nursery: More students attended nursery school.
- higher: The majority students want to take higher education.
- internet: More students can access internet at home.
- romantic: More students have no romantic relationship.
- famrel: The majority students think the quality of family relationships are more than good.
- goout: The going out with their friends are distributed in a wide range.
- Dalc: Most students did not drink alcohol during weekday.
- Walc: Many students drink alcohol during weekend.
- health: the majority of the students are in good health. from 1 - very bad to 5 - very good. But quite number of students are in bad health conditions: 47 out of 395 for 1 indicator (very bad), 45 for 2.
- absences: Most students attend school but some students has many school absences.
- G1, G2 and G3: They show graphs close to normal distribution. Surprisingly, there are quite a lot of students failed in the end of the course in G3.

```
par(mfrow=c(1,1)) # Return the format of displaying graph showing 1*1 graph on each pane
```

b. How many observations and variables are in the dataset?

```
dim(new_mat)
```

```
## [1] 395 33
```

There are 395 observations and 33 variables.

c. What proportion of individuals is passed?

```
mean(new_mat$G3)
```

```
## [1] 0.8227848
```

d. What is the name of columns of numeric variables?

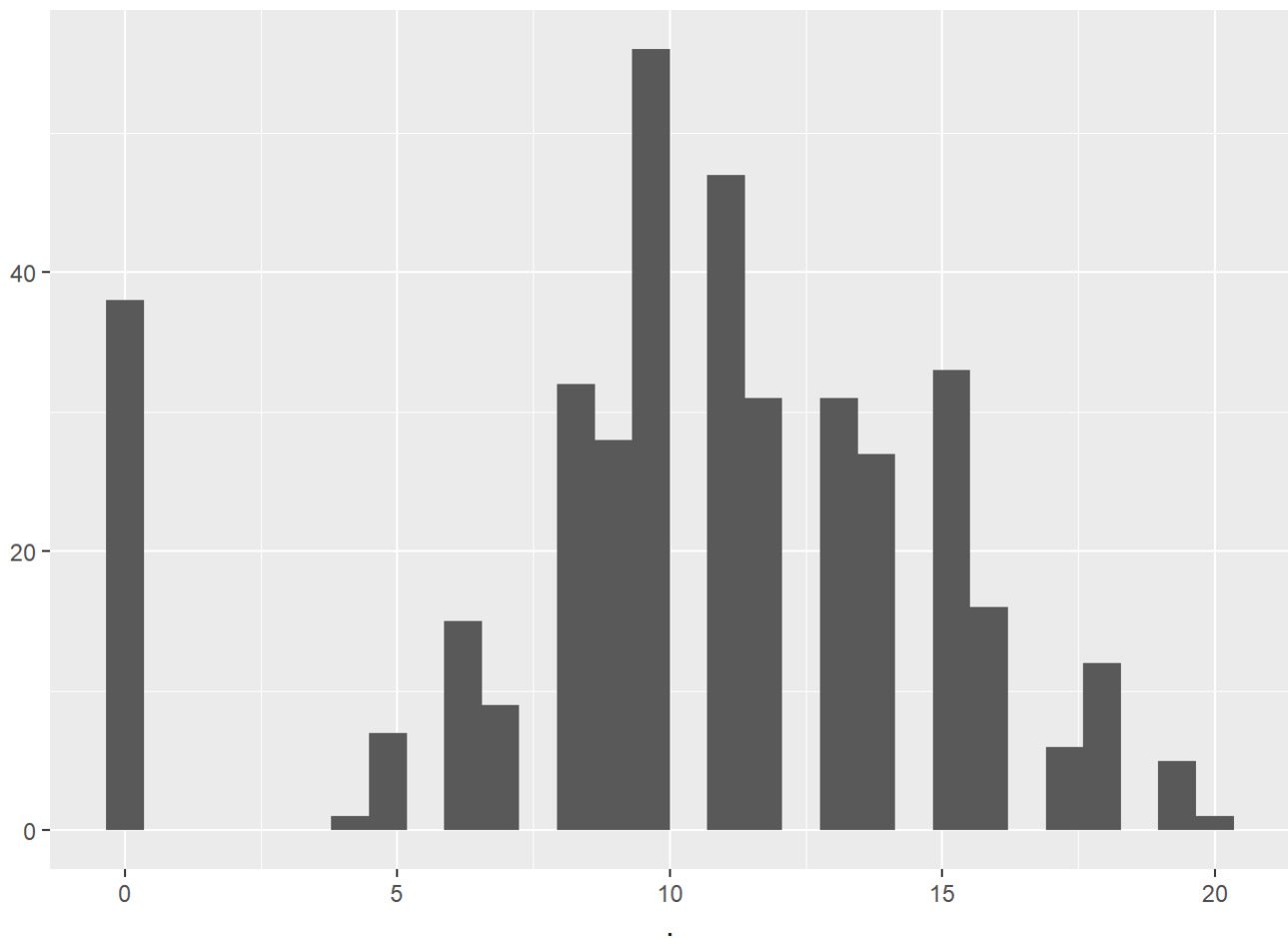
```
names(num)
```

```
## [1] "age"      "Medu"      "Fedu"      "traveltime" "studytime"  
## [6] "failures" "famrel"    "freetime"  "goout"      "Dalc"  
## [11] "Walc"     "health"    "absences"  "G1"         "G2"  
## [16] "G3"
```

e. Show the distribution of the dependent variable G3

```
mat$G3%>%qplot()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
janitor::tabyl(mat$G3)%>%print # Show frequency and percentage in original G3 variable.
```



```
## mat$G3  n    percent
##      0 38 0.096202532
##      4  1 0.002531646
##      5  7 0.017721519
##      6 15 0.037974684
##      7  9 0.022784810
##      8 32 0.081012658
##      9 28 0.070886076
##     10 56 0.141772152
##     11 47 0.118987342
##     12 31 0.078481013
##     13 31 0.078481013
##     14 27 0.068354430
##     15 33 0.083544304
##     16 16 0.040506329
##     17  6 0.015189873
##     18 12 0.030379747
##     19  5 0.012658228
##     20  1 0.002531646
```

```
tabyl(fnew_mat$G3) # Show frequency and percentage in binomial G3 variable.
```

```
## fnew_mat$G3  n    percent
##      fail   70 0.1772152
##      pass  325 0.8227848
```

There are around 18% students that failed the test.

f. Show analysis tibble for whether there is the difference in the characteristics of the numeric variable for G3.

```
num%>%select(age:absences, G3)%>% group_by(G3)%>%summarise_all(mean)
```

```
## # A tibble: 18 x 14
##       G3    age  Medu  Fedu traveltime studytime failures famrel freetime goout
##   <int> <dbl> <dbl> <dbl>      <dbl>      <dbl>      <dbl> <dbl> <dbl> <dbl>
## 1     0  17.1  2.32  2.29      1.61      1.97    0.921  3.84  3.13  3.21
## 2     4   17    4      3        1        2        2      3      4      5
## 3     5  16.7  2.71  2.57      1.14      1.43     1      3.71  4.14  4
## 4     6  16.3  2.73  2.27      1.53      1.93     0      4      3.07  3.27
## 5     7   17    2.78  2      1.67      1.89    1.44    4.11  3.44  3.33
## 6     8  17.2  2.53  2.38      1.5      1.94    0.625  3.91  3.28  3.66
## 7     9  17.2  2.71  2.5      1.36      2.07    0.464  3.86  3.21  3.25
## 8    10  16.6  2.54  2.25      1.68      1.95    0.357  4.12  3.18  3.04
## 9    11  16.6  2.68  2.66      1.36      2.15    0.106  3.85  3.15  2.98
## 10   12  16.5  2.52  2.48      1.42      1.94    0.194  3.90  3.42  2.87
## 11   13  16.8  2.74  2.68      1.55      1.94    0.226  3.71  3.06  2.97
## 12   14  16.5  3.19  2.81      1.19      2.30    0.0370  4.15  3.22  3.19
## 13   15  16.2  3.09  2.76      1.36      2.15    0.0606  3.94  3.21  2.76
## 14   16  16.4  3.31  2.44      1.25      2.12     0      4      3.5  3
## 15   17  16.7  3      3.67      1.33      2.5      0      4      3.17  2.5
## 16   18  16.4  3.42  2.83      1.33      2.08    0.0833  4.17  3.17  3.25
## 17   19  16.2  3.4    2.8      1.2      2      0      4.2  3.8  2.8
## 18   20  16    4      3        1      4      0      4      2    2
## # ... with 4 more variables: Dalc <dbl>, Walc <dbl>, health <dbl>,
## #   absences <dbl>
```

Medu, Fedu, studytime, failures, goout, Walc and absences are significant variables of which failure and absences are the most.

g. Show box plots analyzing whether there is the difference in the level of numeric variables for G3.

```

library(ggplot2); library(gridExtra)#####failed 색깔
p1<-ggplot(data=mat, aes(x=G3, y=age, fill=G3)) + geom_boxplot()
p2<-ggplot(data=mat, aes(x=G3, y=Medu, fill=G3)) + geom_boxplot()
p3<-ggplot(data=mat, aes(x=G3, y=Fedu, fill=G3)) + geom_boxplot()
p4<-ggplot(data=mat, aes(x=G3, y=traveltime, fill=G3)) + geom_boxplot()
p5<-ggplot(data=mat, aes(x=G3, y=studytime, fill=G3)) + geom_boxplot()
p6<-ggplot(data=mat, aes(x=G3, y=failures, fill=G3)) + geom_boxplot()
p7<-ggplot(data=mat, aes(x=G3, y=famrel, fill=G3)) + geom_boxplot()
p8<-ggplot(data=mat, aes(x=G3, y=freetime, fill=G3)) + geom_boxplot()
p9<-ggplot(data=mat, aes(x=G3, y=goout, fill=G3)) + geom_boxplot()
p10<-ggplot(data=mat, aes(x=G3, y=Dalc, fill=G3)) + geom_boxplot()
p11<-ggplot(data=mat, aes(x=G3, y=Walc, fill=G3)) + geom_boxplot()
p12<-ggplot(data=mat, aes(x=G3, y=health, fill=G3)) + geom_boxplot()
p13<-ggplot(data=mat, aes(x=G3, y=absences, fill=G3)) + geom_boxplot()
p14<-ggplot(data=mat, aes(x=G3, y=school, fill=G3)) + geom_boxplot()
p15<-ggplot(data=mat, aes(x=G3, y=sex, fill=G3)) + geom_boxplot()
p16<-ggplot(data=mat, aes(x=G3, y=address, fill=G3)) + geom_boxplot()
p17<-ggplot(data=mat, aes(x=G3, y=famsize, fill=G3)) + geom_boxplot()
p18<-ggplot(data=mat, aes(x=G3, y=Pstatus, fill=G3)) + geom_boxplot()
p19<-ggplot(data=mat, aes(x=G3, y=Mjob, fill=G3)) + geom_boxplot()
p20<-ggplot(data=mat, aes(x=G3, y=Fjob, fill=G3)) + geom_boxplot()
p21<-ggplot(data=mat, aes(x=G3, y=guardian, fill=G3)) + geom_boxplot()
p22<-ggplot(data=mat, aes(x=G3, y=schoolsup, fill=G3)) + geom_boxplot()
p23<-ggplot(data=mat, aes(x=G3, y=famsup, fill=G3)) + geom_boxplot()
p24<-ggplot(data=mat, aes(x=G3, y=paid, fill=G3)) + geom_boxplot()
p25<-ggplot(data=mat, aes(x=G3, y=activities, fill=G3)) + geom_boxplot()
p26<-ggplot(data=mat, aes(x=G3, y=nursery, fill=G3)) + geom_boxplot()
p27<-ggplot(data=mat, aes(x=G3, y=higher, fill=G3)) + geom_boxplot()
p28<-ggplot(data=mat, aes(x=G3, y=internet, fill=G3)) + geom_boxplot()
p29<-ggplot(data=mat, aes(x=G3, y=romantic, fill=G3)) + geom_boxplot()

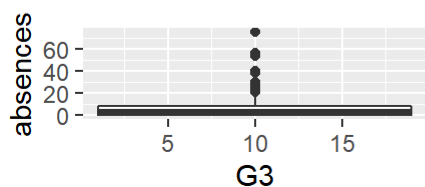
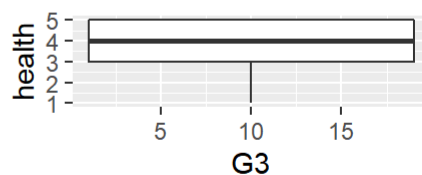
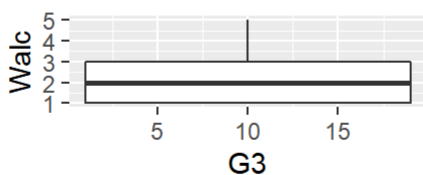
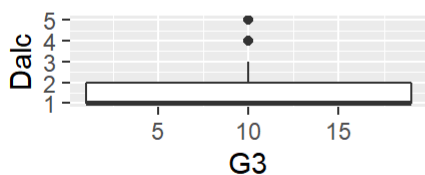
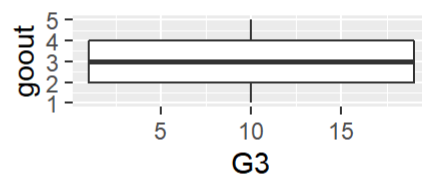
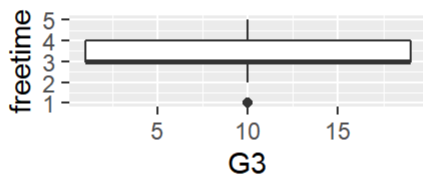
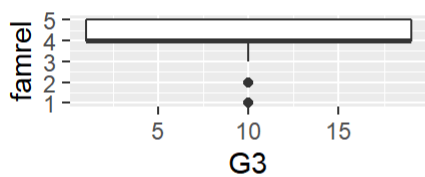
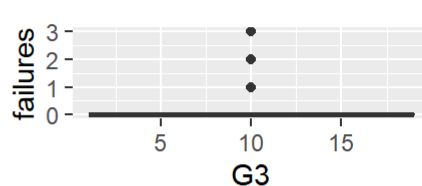
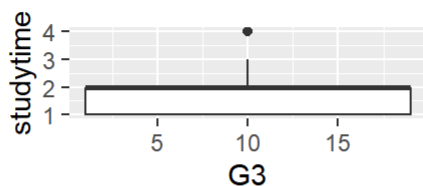
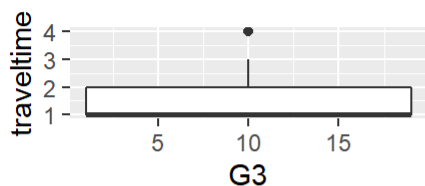
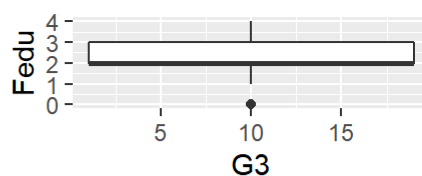
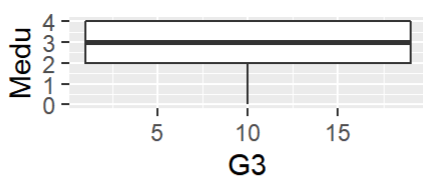
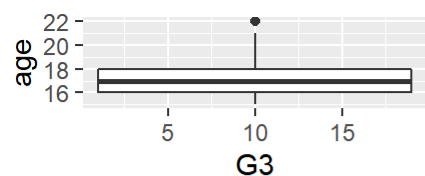
```

```

grid.arrange(p1,p2,p3, p4,p5, p6,p7, p8,p9, p10,p11, p12,p13, ncol=3)

```

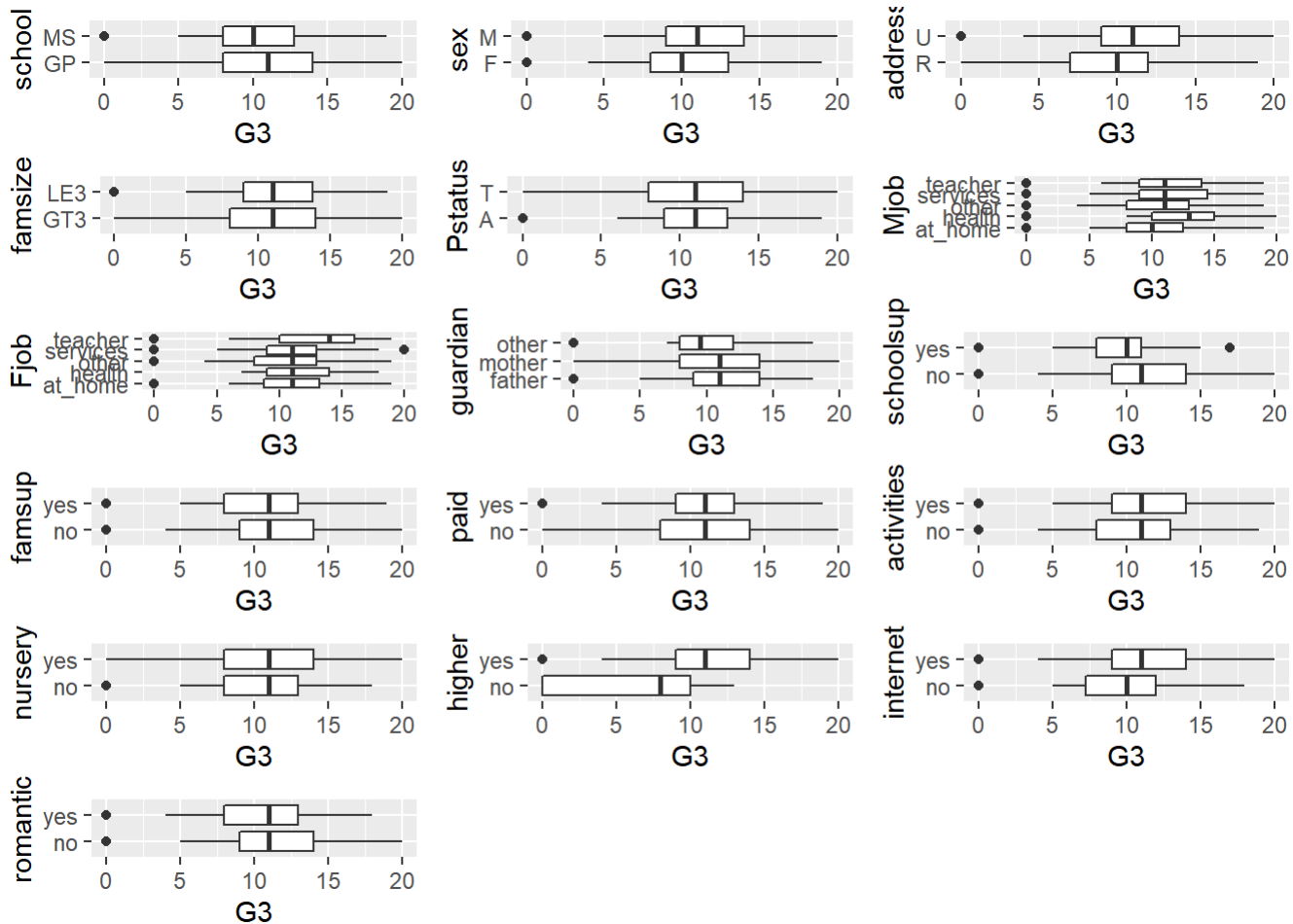
```
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
```



Fedu and failures seems significant variables for G3.

h. Show box plots analyzing whether there is the difference in the level of character variables for G3.

```
grid.arrange(p14,p15, p16,p17, p18,p19, p20,p21,p22,p23,p24,p25,p26,p27,p28,p29, ncol=3)
```



school, sex, address, Mjob, Fjob, gaurdian, schoolsup, higher and internet are factors which affect on results of G3.

Correlation

Let's examine and analyse how our variables are correlated with each other.

i. Show correlation numbers in the numeric variables

```
round(cor(allnum),3)
```

##	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob
## school	1.000	0.012	0.378	-0.280	-0.065	0.046	-0.133	-0.080	-0.065
## sex	0.012	1.000	0.029	0.029	0.090	-0.023	-0.078	-0.035	-0.116
## age	0.378	0.029	1.000	-0.147	-0.038	0.030	-0.164	-0.163	-0.090
## address	-0.280	0.029	-0.147	1.000	-0.072	-0.043	0.139	0.072	0.096
## famsize	-0.065	0.090	-0.038	-0.072	1.000	0.150	0.043	0.059	-0.087
## Pstatus	0.046	-0.023	0.030	-0.043	0.150	1.000	-0.124	-0.089	0.013
## Medu	-0.133	-0.078	-0.164	0.139	0.043	-0.124	1.000	0.623	0.535
## Fedu	-0.080	-0.035	-0.163	0.072	0.059	-0.089	0.623	1.000	0.342
## Mjob	-0.065	-0.116	-0.090	0.096	-0.087	0.013	0.535	0.342	1.000
## Fjob	0.012	0.004	-0.066	0.055	0.047	-0.021	0.215	0.368	0.309
## reason	-0.126	0.010	-0.006	0.119	-0.001	-0.036	0.038	0.018	-0.059
## guardian	-0.076	-0.015	-0.269	-0.070	-0.030	-0.026	0.133	0.005	0.120
## traveltime	0.242	-0.060	0.071	-0.328	-0.063	0.028	-0.172	-0.158	-0.109
## studytime	-0.091	0.306	-0.004	-0.021	0.074	0.024	0.065	-0.009	-0.011
## failures	0.060	-0.044	0.244	-0.079	0.016	-0.003	-0.237	-0.250	-0.104
## schoolsup	-0.140	0.138	-0.252	0.025	0.029	-0.042	-0.036	0.038	-0.097
## famsup	-0.165	0.152	-0.141	0.024	0.113	0.019	0.184	0.185	0.143
## paid	-0.017	0.129	-0.036	0.053	0.014	0.046	0.160	0.087	0.174
## activities	-0.117	-0.100	-0.103	-0.051	0.000	0.097	0.108	0.113	0.129
## nursery	-0.089	0.008	-0.087	0.060	-0.102	-0.091	0.193	0.157	0.145
## higher	-0.024	0.151	-0.209	0.043	0.006	-0.041	0.169	0.175	0.076
## internet	-0.134	-0.044	-0.112	0.217	-0.001	0.070	0.201	0.128	0.172
## romantic	0.061	0.102	0.165	0.005	-0.034	-0.040	0.040	0.016	-0.054
## famrel	-0.048	-0.059	0.054	0.014	0.023	0.025	-0.004	-0.001	-0.031
## freetime	0.033	-0.239	0.016	0.035	-0.018	0.039	0.031	-0.013	0.073
## goout	-0.007	-0.076	0.127	0.069	-0.023	0.003	0.064	0.043	0.012
## Dalc	0.114	-0.268	0.131	-0.093	-0.102	-0.031	0.020	0.002	0.004
## Walc	0.065	-0.274	0.117	-0.101	-0.103	0.006	-0.047	-0.013	0.020
## health	-0.043	-0.144	-0.062	-0.040	0.029	0.022	-0.047	0.015	0.051
## absences	-0.088	0.067	0.175	-0.028	-0.036	-0.135	0.100	0.024	-0.024
## G1	-0.026	-0.092	-0.064	0.070	-0.071	-0.017	0.205	0.190	0.188
## G2	-0.050	-0.091	-0.143	0.126	-0.081	-0.041	0.216	0.165	0.155
## G3	-0.045	-0.103	-0.162	0.106	-0.081	-0.058	0.217	0.152	0.146
##	Fjob	reason	guardian	traveltime	studytime	failures	schoolsup		
## school	0.012	-0.126	-0.076	0.242	-0.091	0.060	-0.140		
## sex	0.004	0.010	-0.015	-0.060	0.306	-0.044	0.138		
## age	-0.066	-0.006	-0.269	0.071	-0.004	0.244	-0.252		
## address	0.055	0.119	-0.070	-0.328	-0.021	-0.079	0.025		
## famsize	0.047	-0.001	-0.030	-0.063	0.074	0.016	0.029		
## Pstatus	-0.021	-0.036	-0.026	0.028	0.024	-0.003	-0.042		
## Medu	0.215	0.038	0.133	-0.172	0.065	-0.237	-0.036		
## Fedu	0.368	0.018	0.005	-0.158	-0.009	-0.250	0.038		
## Mjob	0.309	-0.059	0.120	-0.109	-0.011	-0.104	-0.097		
## Fjob	1.000	-0.086	-0.066	-0.066	0.022	-0.007	0.075		
## reason	-0.086	1.000	-0.096	-0.089	0.084	0.011	0.015		
## guardian	-0.066	-0.096	1.000	-0.063	-0.031	-0.224	0.022		
## traveltime	-0.066	-0.089	-0.063	1.000	-0.101	0.092	-0.009		
## studytime	0.022	0.084	-0.031	-0.101	1.000	-0.174	0.038		
## failures	-0.007	0.011	-0.224	0.092	-0.174	1.000	0.000		
## schoolsup	0.075	0.015	0.022	-0.009	0.038	0.000	1.000		
## famsup	0.055	0.076	-0.014	-0.003	0.145	-0.055	0.105		

## paid	-0.023	0.087	0.053	-0.066	0.167	-0.188	-0.021	
## activities	0.037	0.021	0.035	-0.008	0.090	-0.069	0.046	
## nursery	0.075	0.044	0.153	-0.033	0.081	-0.101	0.046	
## higher	-0.019	0.138	0.004	-0.084	0.175	-0.300	0.054	
## internet	0.031	0.070	0.003	-0.111	0.059	-0.063	-0.010	
## romantic	0.047	-0.005	-0.081	0.022	0.053	0.093	-0.081	
## famrel	-0.035	-0.006	-0.029	-0.017	0.040	-0.044	-0.001	
## freetime	-0.053	-0.112	-0.051	-0.017	-0.143	0.092	-0.045	
## goout	-0.040	-0.017	0.064	0.029	-0.064	0.125	-0.038	
## Dalc	0.057	-0.080	-0.054	0.138	-0.196	0.136	-0.021	
## Walc	-0.067	-0.060	0.024	0.134	-0.254	0.142	-0.087	
## health	0.025	-0.070	0.001	0.008	-0.076	0.066	-0.034	
## absences	-0.018	0.136	-0.043	-0.013	-0.063	0.064	0.023	
## G1	0.152	0.032	0.005	-0.093	0.161	-0.355	-0.213	
## G2	0.115	0.031	0.021	-0.153	0.136	-0.356	-0.117	
## G3	0.091	0.009	0.054	-0.117	0.098	-0.360	-0.083	
##	famsup	paid	activities	nursery	higher	internet	romantic	famrel
## school	-0.165	-0.017	-0.117	-0.089	-0.024	-0.134	0.061	-0.048
## sex	0.152	0.129	-0.100	0.008	0.151	-0.044	0.102	-0.059
## age	-0.141	-0.036	-0.103	-0.087	-0.209	-0.112	0.165	0.054
## address	0.024	0.053	-0.051	0.060	0.043	0.217	0.005	0.014
## famsize	0.113	0.014	0.000	-0.102	0.006	-0.001	-0.034	0.023
## Pstatus	0.019	0.046	0.097	-0.091	-0.041	0.070	-0.040	0.025
## Medu	0.184	0.160	0.108	0.193	0.169	0.201	0.040	-0.004
## Fedu	0.185	0.087	0.113	0.157	0.175	0.128	0.016	-0.001
## Mjob	0.143	0.174	0.129	0.145	0.076	0.172	-0.054	-0.031
## Fjob	0.055	-0.023	0.037	0.075	-0.019	0.031	0.047	-0.035
## reason	0.076	0.087	0.021	0.044	0.138	0.070	-0.005	-0.006
## guardian	-0.014	0.053	0.035	0.153	0.004	0.003	-0.081	-0.029
## traveltime	-0.003	-0.066	-0.008	-0.033	-0.084	-0.111	0.022	-0.017
## studytime	0.145	0.167	0.090	0.081	0.175	0.059	0.053	0.040
## failures	-0.055	-0.188	-0.069	-0.101	-0.300	-0.063	0.093	-0.044
## schoolsup	0.105	-0.021	0.046	0.046	0.054	-0.010	-0.081	-0.001
## famsup	1.000	0.293	-0.002	0.060	0.101	0.104	0.012	-0.020
## paid	0.293	1.000	-0.021	0.102	0.189	0.153	0.006	0.000
## activities	-0.002	-0.021	1.000	0.003	0.096	0.049	0.020	0.041
## nursery	0.060	0.102	0.003	1.000	0.054	0.008	0.027	-0.004
## higher	0.101	0.189	0.096	0.054	1.000	0.020	-0.106	0.024
## internet	0.104	0.153	0.049	0.008	0.020	1.000	0.087	0.033
## romantic	0.012	0.006	0.020	0.027	-0.106	0.087	1.000	-0.064
## famrel	-0.020	0.000	0.041	-0.004	0.024	0.033	-0.064	1.000
## freetime	0.011	-0.064	0.090	-0.025	-0.061	0.051	-0.011	0.151
## goout	-0.016	0.010	0.046	0.005	-0.040	0.074	0.008	0.065
## Dalc	-0.032	0.062	-0.067	-0.085	-0.070	0.036	0.015	-0.078
## Walc	-0.087	0.060	-0.037	-0.100	-0.100	0.012	-0.010	-0.113
## health	0.029	-0.078	0.024	-0.018	-0.016	-0.080	0.026	0.094
## absences	0.024	0.007	-0.014	0.019	-0.056	0.102	0.153	-0.044
## G1	-0.085	0.039	0.057	0.069	0.178	0.072	-0.037	0.022
## G2	-0.059	0.105	0.051	0.068	0.179	0.119	-0.112	-0.018
## G3	-0.039	0.102	0.016	0.052	0.182	0.098	-0.130	0.051
##	freetime	goout	Dalc	Walc	health	absences	G1	G2
## school	0.033	-0.007	0.114	0.065	-0.043	-0.088	-0.026	-0.045

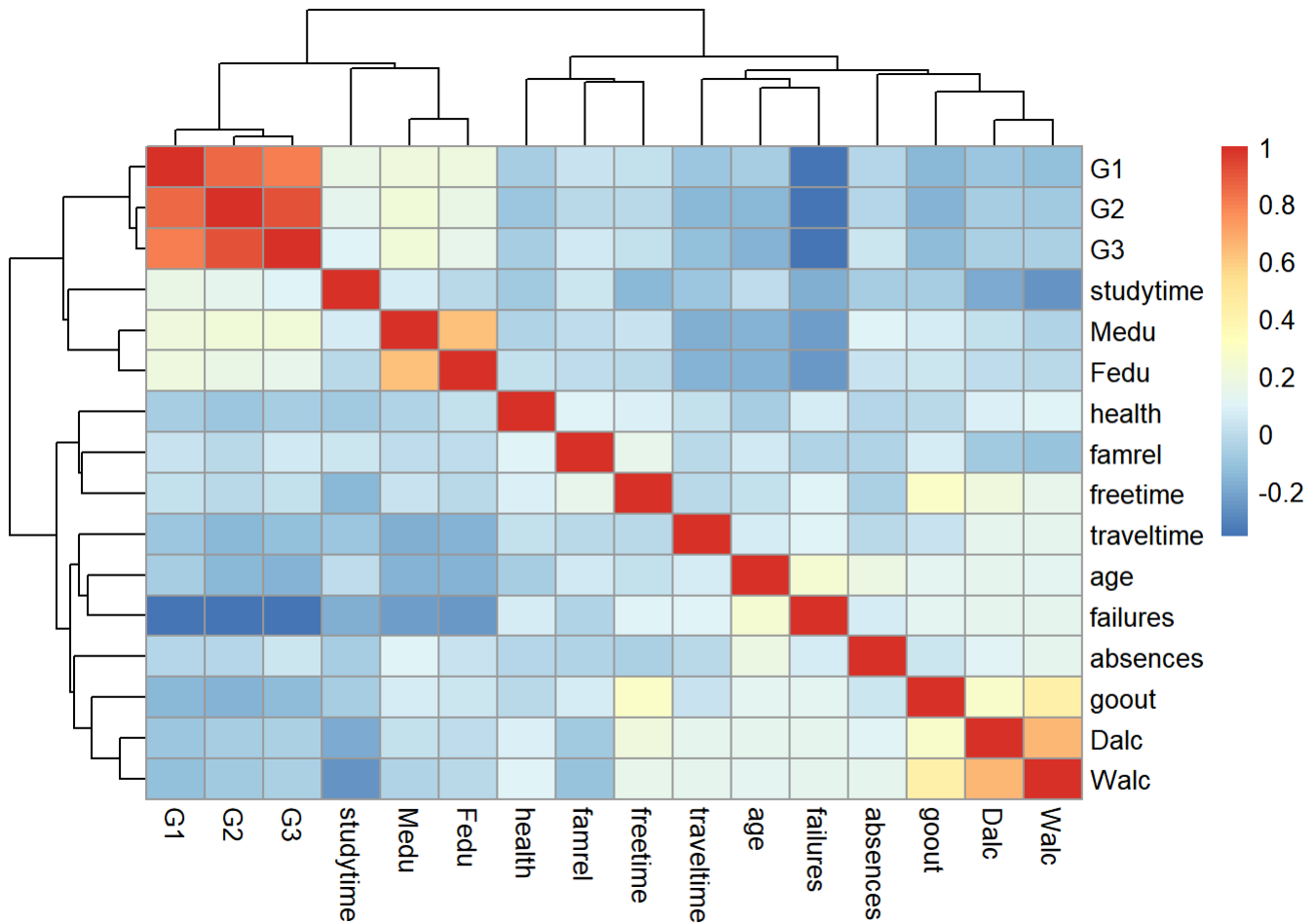
## sex	-0.239	-0.076	-0.268	-0.274	-0.144	0.067	-0.092	-0.091	-0.103
## age	0.016	0.127	0.131	0.117	-0.062	0.175	-0.064	-0.143	-0.162
## address	0.035	0.069	-0.093	-0.101	-0.040	-0.028	0.070	0.126	0.106
## famsize	-0.018	-0.023	-0.102	-0.103	0.029	-0.036	-0.071	-0.081	-0.081
## Pstatus	0.039	0.003	-0.031	0.006	0.022	-0.135	-0.017	-0.041	-0.058
## Medu	0.031	0.064	0.020	-0.047	-0.047	0.100	0.205	0.216	0.217
## Fedu	-0.013	0.043	0.002	-0.013	0.015	0.024	0.190	0.165	0.152
## Mjob	0.073	0.012	0.004	0.020	0.051	-0.024	0.188	0.155	0.146
## Fjob	-0.053	-0.040	0.057	-0.067	0.025	-0.018	0.152	0.115	0.091
## reason	-0.112	-0.017	-0.080	-0.060	-0.070	0.136	0.032	0.031	0.009
## guardian	-0.051	0.064	-0.054	0.024	0.001	-0.043	0.005	0.021	0.054
## traveltime	-0.017	0.029	0.138	0.134	0.008	-0.013	-0.093	-0.153	-0.117
## studytime	-0.143	-0.064	-0.196	-0.254	-0.076	-0.063	0.161	0.136	0.098
## failures	0.092	0.125	0.136	0.142	0.066	0.064	-0.355	-0.356	-0.360
## schoolsup	-0.045	-0.038	-0.021	-0.087	-0.034	0.023	-0.213	-0.117	-0.083
## famsup	0.011	-0.016	-0.032	-0.087	0.029	0.024	-0.085	-0.059	-0.039
## paid	-0.064	0.010	0.062	0.060	-0.078	0.007	0.039	0.105	0.102
## activities	0.090	0.046	-0.067	-0.037	0.024	-0.014	0.057	0.051	0.016
## nursery	-0.025	0.005	-0.085	-0.100	-0.018	0.019	0.069	0.068	0.052
## higher	-0.061	-0.040	-0.070	-0.100	-0.016	-0.056	0.178	0.179	0.182
## internet	0.051	0.074	0.036	0.012	-0.080	0.102	0.072	0.119	0.098
## romantic	-0.011	0.008	0.015	-0.010	0.026	0.153	-0.037	-0.112	-0.130
## famrel	0.151	0.065	-0.078	-0.113	0.094	-0.044	0.022	-0.018	0.051
## freetime	1.000	0.285	0.209	0.148	0.076	-0.058	0.013	-0.014	0.011
## goout	0.285	1.000	0.267	0.420	-0.010	0.044	-0.149	-0.162	-0.133
## Dalc	0.209	0.267	1.000	0.648	0.077	0.112	-0.094	-0.064	-0.055
## Walc	0.148	0.420	0.648	1.000	0.092	0.136	-0.126	-0.085	-0.052
## health	0.076	-0.010	0.077	0.092	1.000	-0.030	-0.073	-0.098	-0.061
## absences	-0.058	0.044	0.112	0.136	-0.030	1.000	-0.031	-0.032	0.034
## G1	0.013	-0.149	-0.094	-0.126	-0.073	-0.031	1.000	0.852	0.801
## G2	-0.014	-0.162	-0.064	-0.085	-0.098	-0.032	0.852	1.000	0.905
## G3	0.011	-0.133	-0.055	-0.052	-0.061	0.034	0.801	0.905	1.000

The correlation varies between 1(positive correlation) and -1(negative correlation). school,Pstatus, reason, guardian, famsup, activities, nursery, internet, famrel, freetime, absences, Dalc and Walc variables are less related with G3 compared to other variables.

j. Visualize the correlations

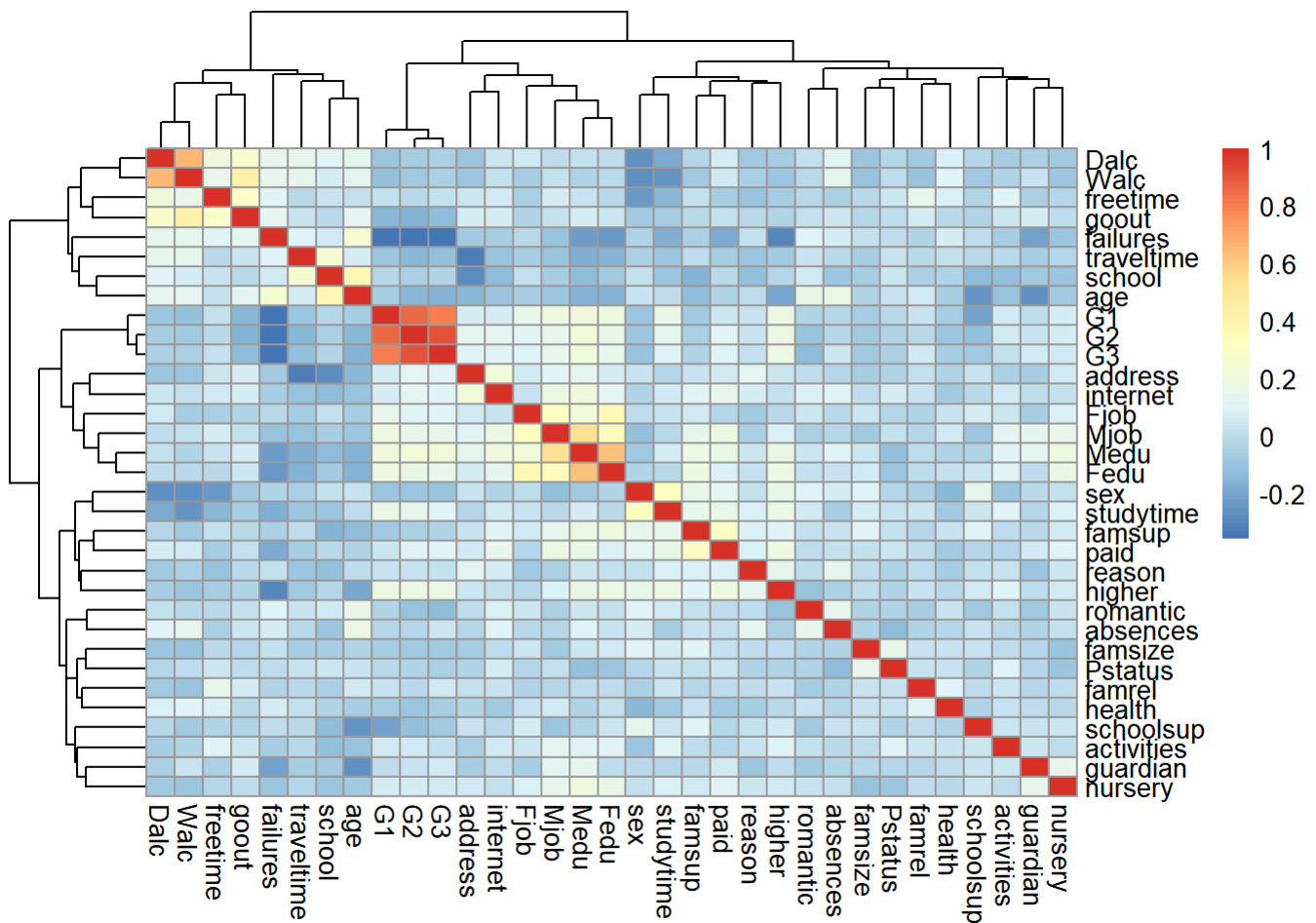
It is difficult to see the correlation in numbers at once. Let's visualize it.

```
pheatmap(cor(num))
```

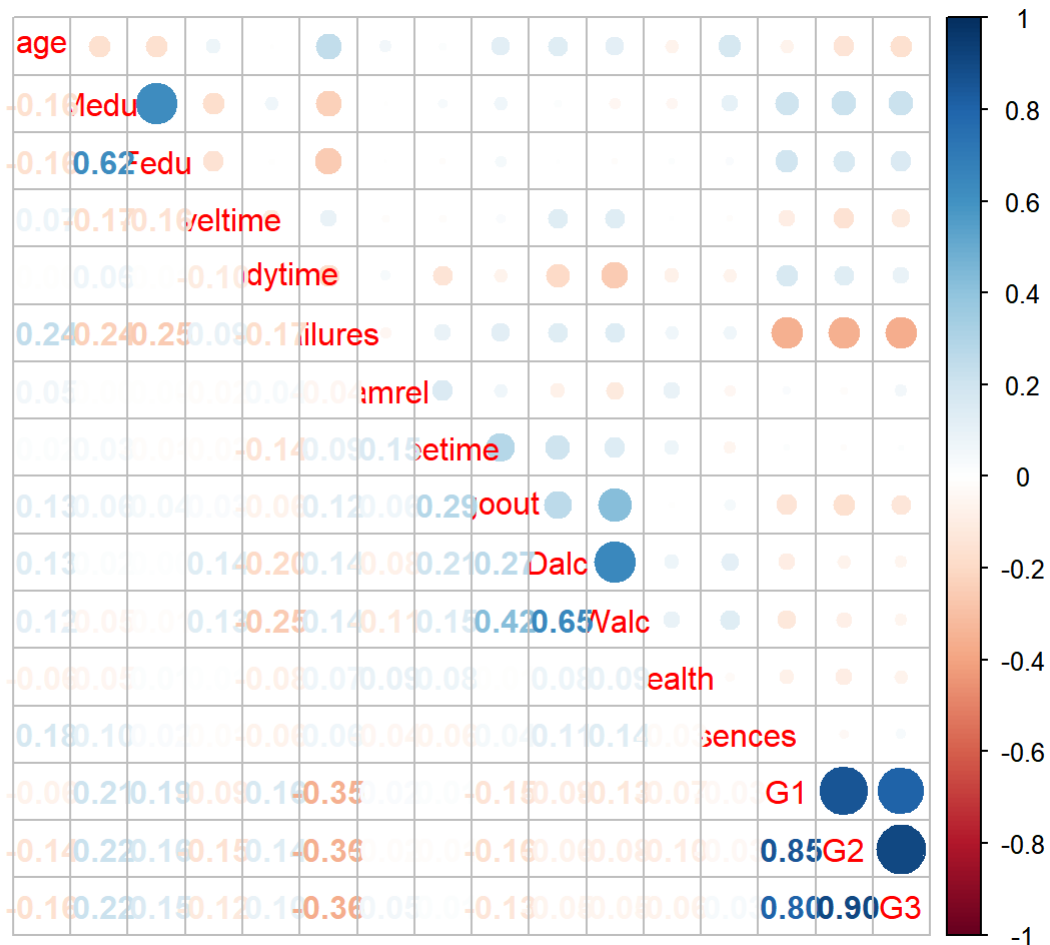
The correlation varies between 1 (positive) and -1 (negative). The dark red color indicates the high correlated positive relation, for example: G1 and G3; G2 and G3. The dark blue color indicates highly correlated negative relation, such as failures and study time.

```
ph heatmap(cor(allnum))
```



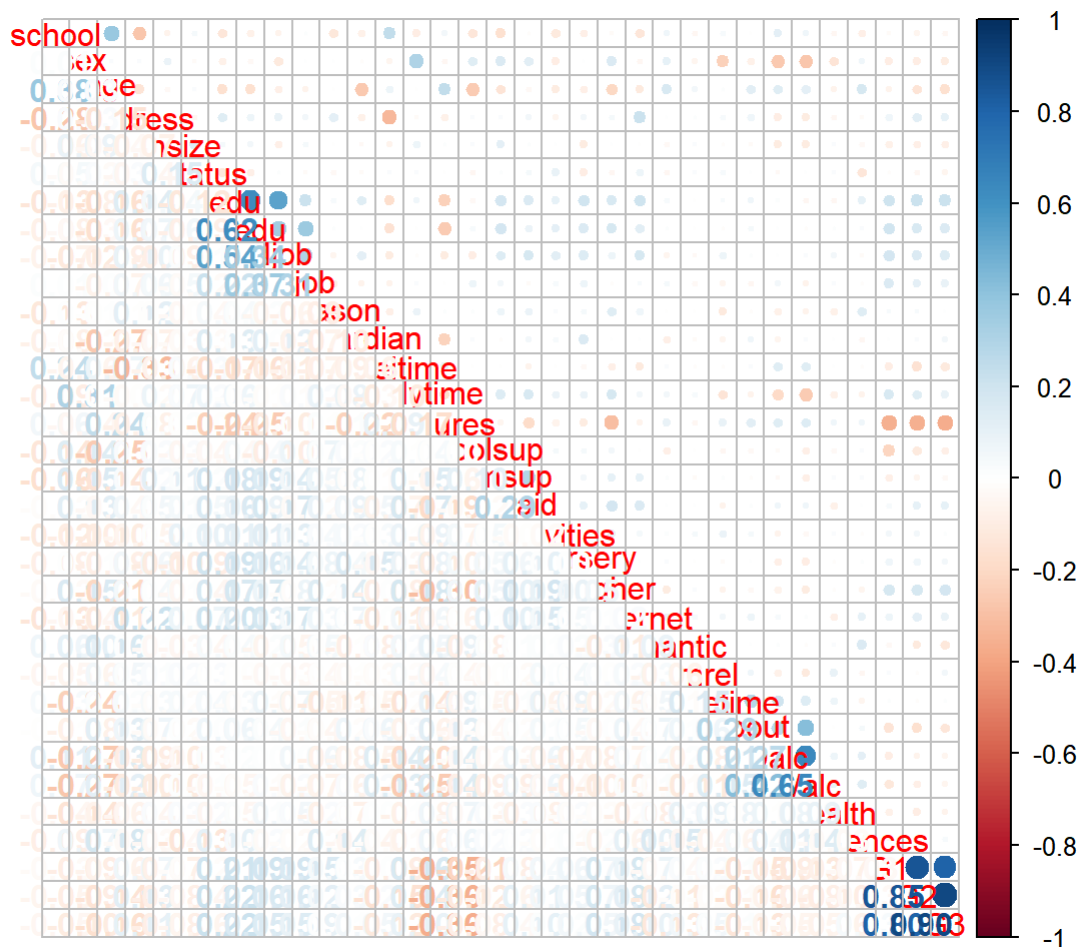
k. Show a heatmap with correlation numbers in the numeric variables

```
corrplot.mixed(cor(num))
```



- Positive correlation with G3: G2, G1, Medu, Fedu, studytime
- Negative correlation with G3: failures, age, freetime, goout
- Positive correlation among variables: Dalc and Walc, Medu and Fedu, goout and Walc/Dalc, freetime and goout/ Dalc, age and failures, G2 and G3, G1 and G2, G1 and G3
- Medu, Fedu, studytime, failures, age, freetime, goout seem to be decisive variables.

```
# Let's think more correlated variables except the above results of correlation from "num" objective
corrplot.mixed(cor(allnum))
```



- Positive correlation with G3 (except the above results): address, Mjob, Fjob, paid, higher
- Negative correlation with G3 (except the above results): sex, famsize, traveltime, schoolsup, romantic
- Positive correlation among variables ($1 > \text{correlation} \geq 0.2$) (except the above results): Medu and Mjob, Fedu and Fjob, Fedu and Mjob, Mjob and Fjob, famsup and paid, school and traveltime, address and internet, G1/G2/G3 and Medu, Medu and internet
- Negative correlation among variables ($-1 < \text{correlation} < -0.2$) (except the above results): failures and G1/G2/G3, failures and higher, failures and Medu/Fedu, failures and guardian, address and traveltime, sex and Dalc/Walc, studytime and Dwal/Walc, sex and freetime, age and guardian/schoolsup/higher

To sum up about correlated variables

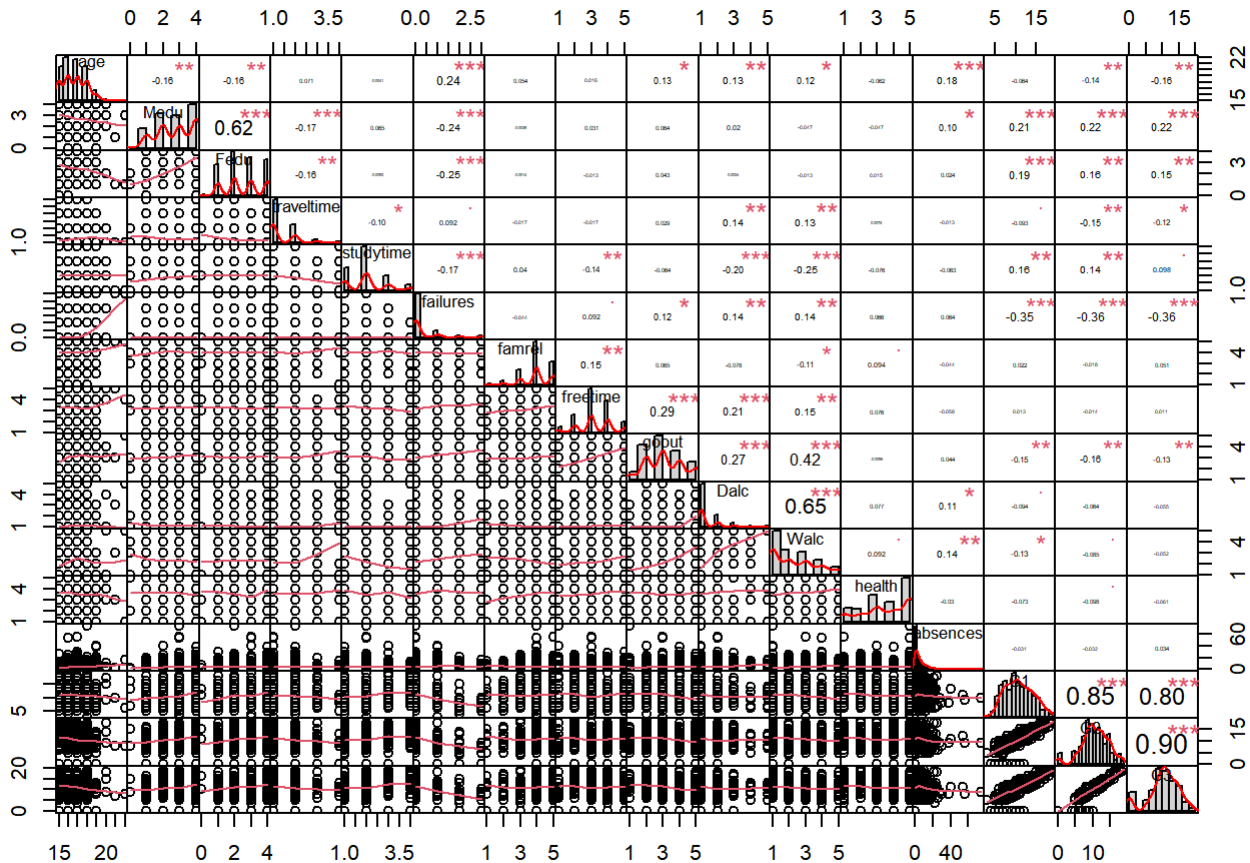
- First, direct influential variables whose absolute value of the relation with G3 is greater than 0.2 are sex, age, address, famsize, Medu, Fedu, Mjob, Fjob, traveltime, studytime, failures, schoolsup, paid, higher, romantic, goout, G1 and G2. They are more correlated with G3 compared to other variables. Whereas school, Pstatus, reason, guardian, famsup, activities, nursery, internet, famrel, freetime, absences, Dalc, Walc and health variables are less related with G3 compared to other variables. We will exclude G1 and G2 from now on. Because we want to know the reason behind poor results except the previous result of any performance.
- Second, indirect influential variables whose absolute value of the correlation with the direct influential variables is greater than 0.2 are famsup, school, internet, Dalc, Walc, freetime and guardian variables. They are more correlated with direct influential variables correlated with G3 directly compared to other variables.
- Third, excluded variables are Pstatus, reason, famsup, activities, nursery, famrel, health and absences. This is because they seem less important variables in consideration with correlation with G3.

I. Create a dataset which is excluded less related variables

```
imp_num<-ex_num[, !(names(ex_num)%in% c("Pstatus", "reason", "famsup", "activities", "nursery", "f
amrel", "health", "absences"))]
```

m. Show correlation table and graph at a time

```
#install.packages("PerformanceAnalytics")
chart.Correlation(num, Histogram=T)
```



Although absences variable has shown in the above graph that it has * correlated with age, Medu, Dalc Walc variables, we will still exclude it because it is less than 0.2

n. Show analysis of relation among decisive variables on G3

The most three direct decisive variables are failure, Medu and studytime. And many variables are related with these decisive variables. Among them, We will examine the following.

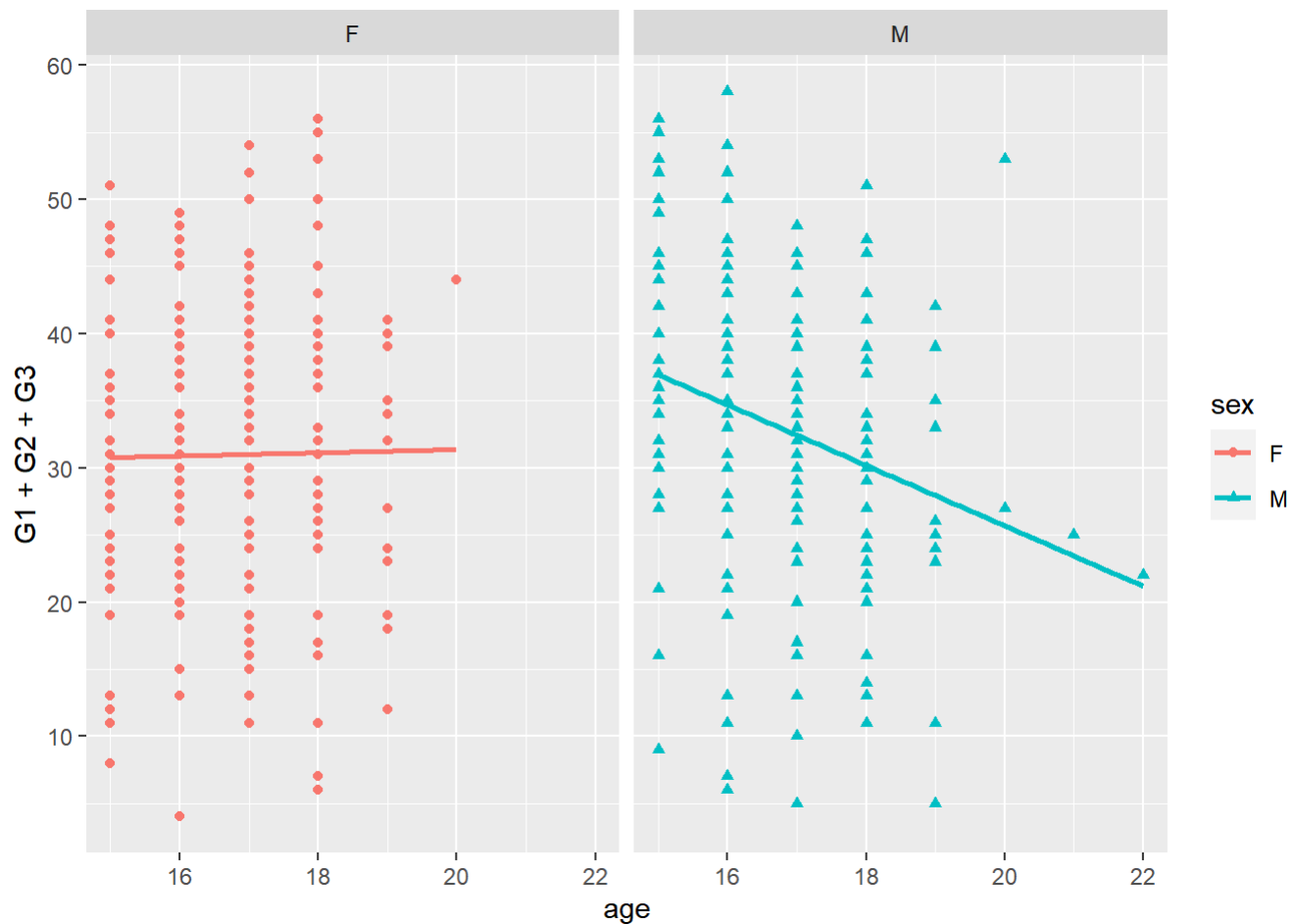
- failure, age and sex(gender)

The reason to choose it is that failure is the most decisive variable on G3. Age has correlation number with failure (0.24) and worth exploring. Sex(gender) has correlation with studytime(0.31) which has correlation with failure.

n-1. Show G1+G2+G3, age and sex relation

```
G1=ggplot(data=mat,aes(x=age, y=G1+G2+G3, col=sex, shape=sex))+geom_point()+geom_smooth(method="l
m",se=F)+facet_grid(~sex)
G1
```

```
## `geom_smooth()` using formula 'y ~ x'
```

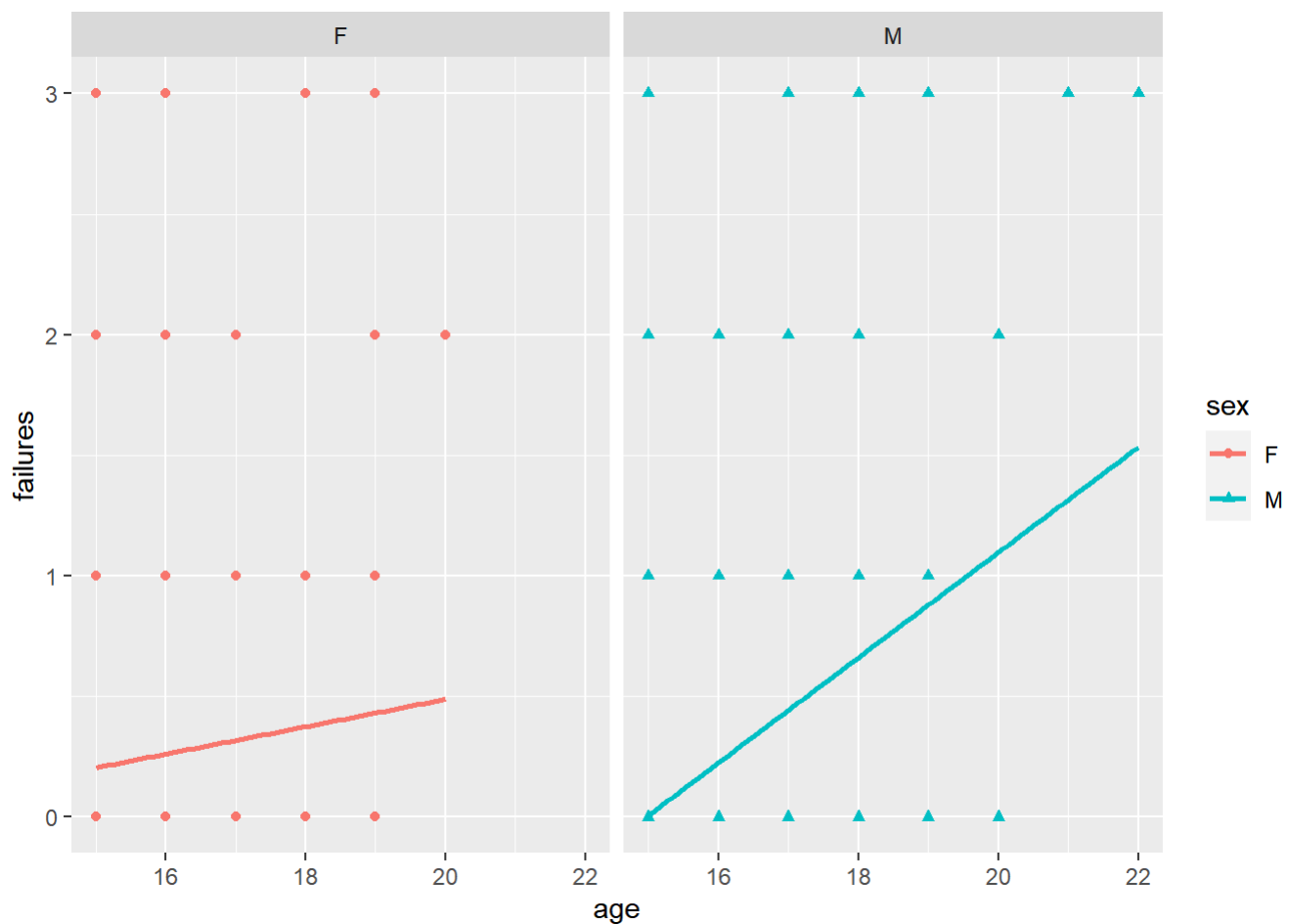


The girls' grades are consistent with their age whereas boys' grades are getting decreased with their age. Behind of the results might be explained with the following questions and codes.

n-2. Why boys are getting bad at their grades with their age?

```
G1=ggplot(data=mat,aes(x=age, y=failures, col=sex, shape=sex))+geom_point()+geom_smooth(method="lm",se=F)+facet_grid(~sex)
G1
```

```
## `geom_smooth()` using formula 'y ~ x'
```



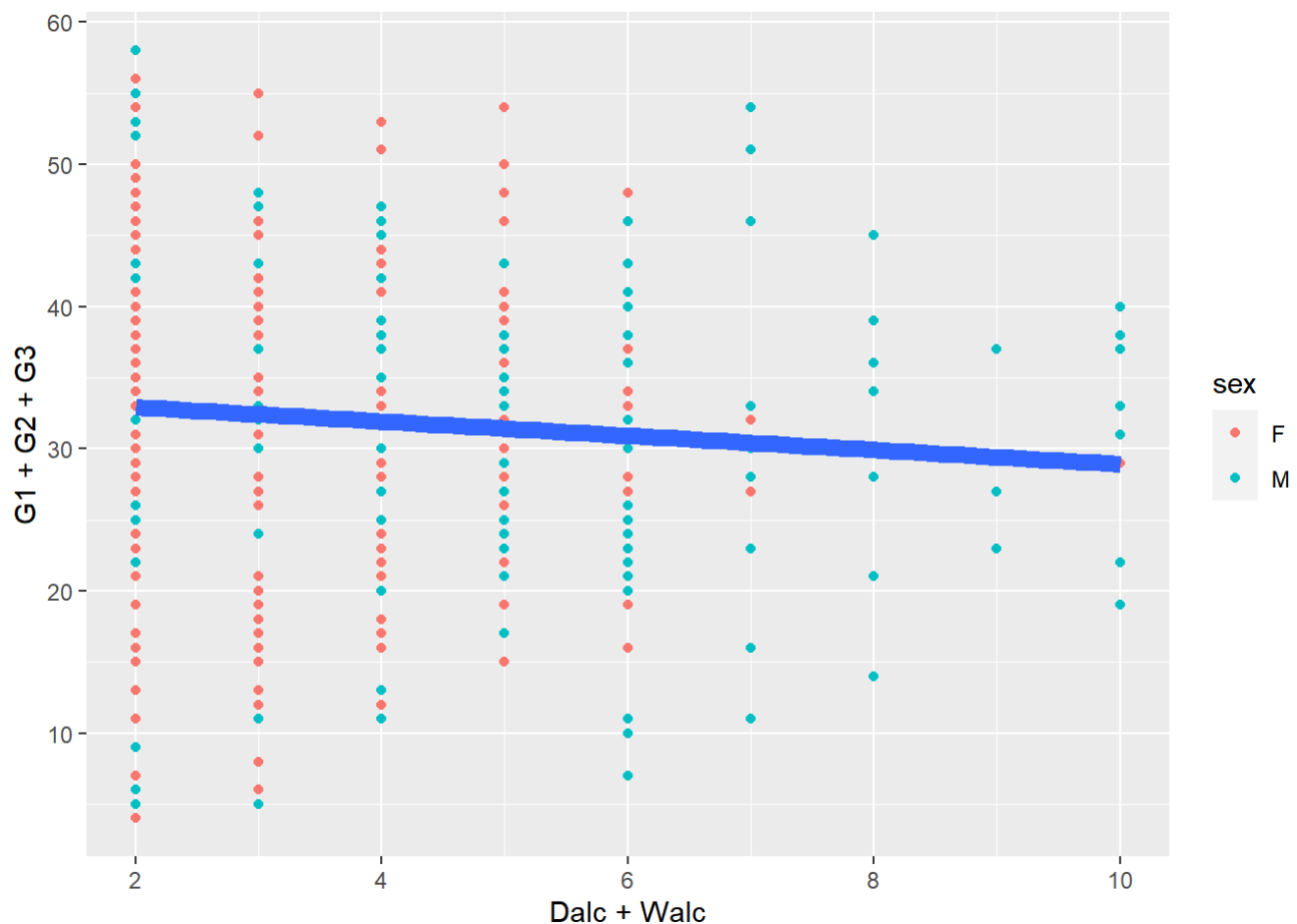
As the students are getting older, they might experience failures which directly affect the results of grades. They might experience more failures than girls. Let's deep into the reason.

n-3. Which variable affect boys' failure?

Let's explore alcohol consumption because one of variables that related with failures are studytime which is also related Dalc and Walc.

```
my_graph <- ggplot(mat, aes(x = Dalc + Walc, y = G1+G2+G3)) +
  geom_point(aes(color = sex)) +
  stat_smooth(method = "lm",
    se = FALSE,
    size = 3)
my_graph
```

```
## `geom_smooth()` using formula 'y ~ x'
```



As the line explains that as alcohol consumption increases the grades decrease. Alcohol intake may lead to frequent confusion and an inability to remember, which results in poor performance. The higher amount of alcohol consumption is from boys. Boys's higher intake of alcohol than girls might affect on their poorer grades than girls.

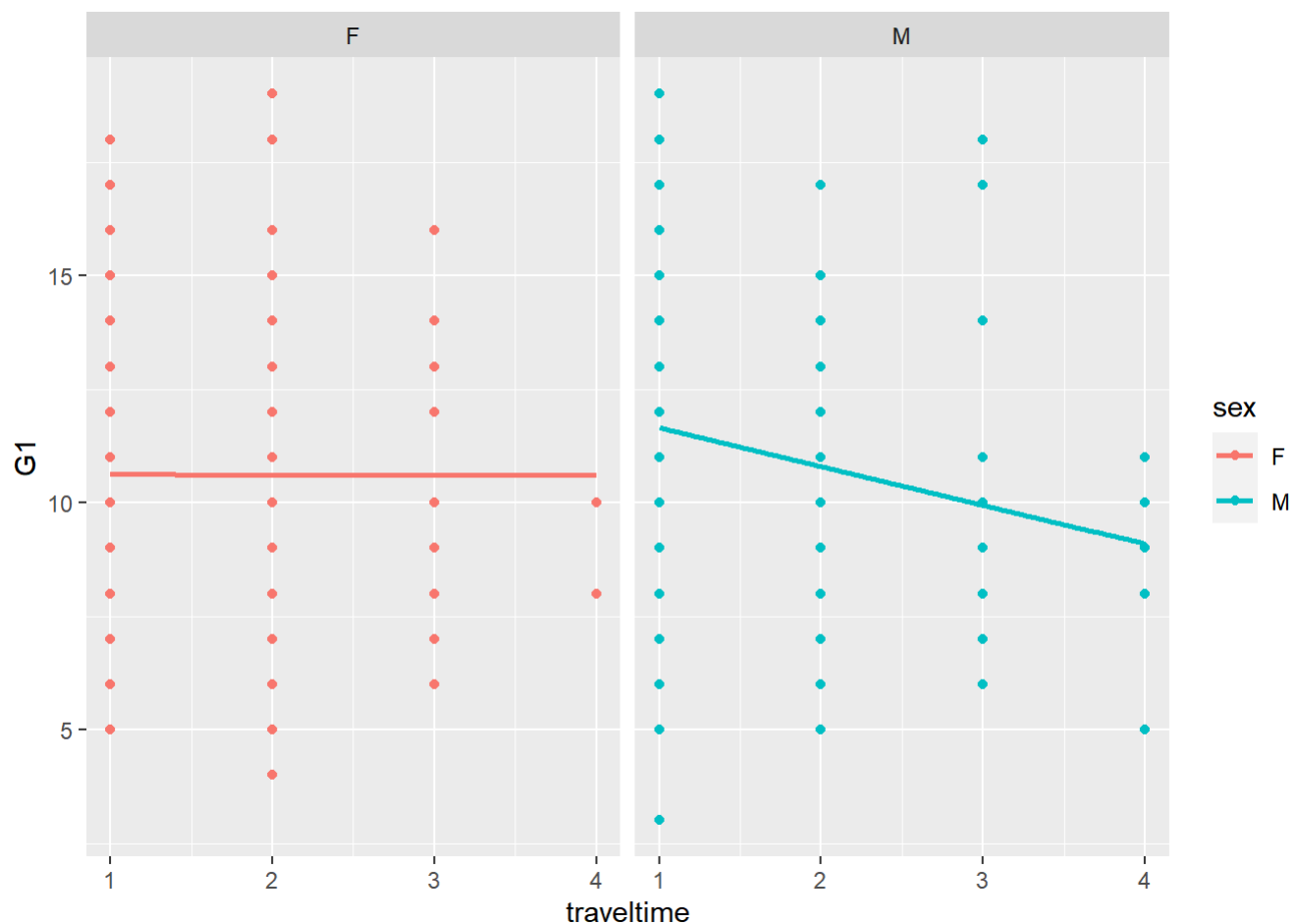
n-4. Might a negatively correlated variable, traveltime affect on male's relatively poor results?

```
table(mat$traveltime)
```

```
##
##  1  2  3  4
## 257 107 23 8
```

```
travel=ggplot(data=mat, aes(x=traveltime, y=G1, col=sex))+geom_point()+geom_smooth(method="lm",se=
F)+facet_grid(~sex)
travel
```

```
## `geom_smooth()` using formula 'y ~ x'
```

As shown above, the negative impact of the traveltime could be largely seen in the boys' performance, the further the male student resides from the school, the less their performance would they get.

Therefore, the difference between results of female and male would be explained with failures, alcohol consumption, traveltime.

2.3. Data Pre-processing_Data Modification and Cleaning

2.3.1. Checking NA (Null Values)

We have already checked the missing values in several statistic tables before. Let's one more check using a simple code.

```
# Check missing data
sapply(mat, anyNA)
```

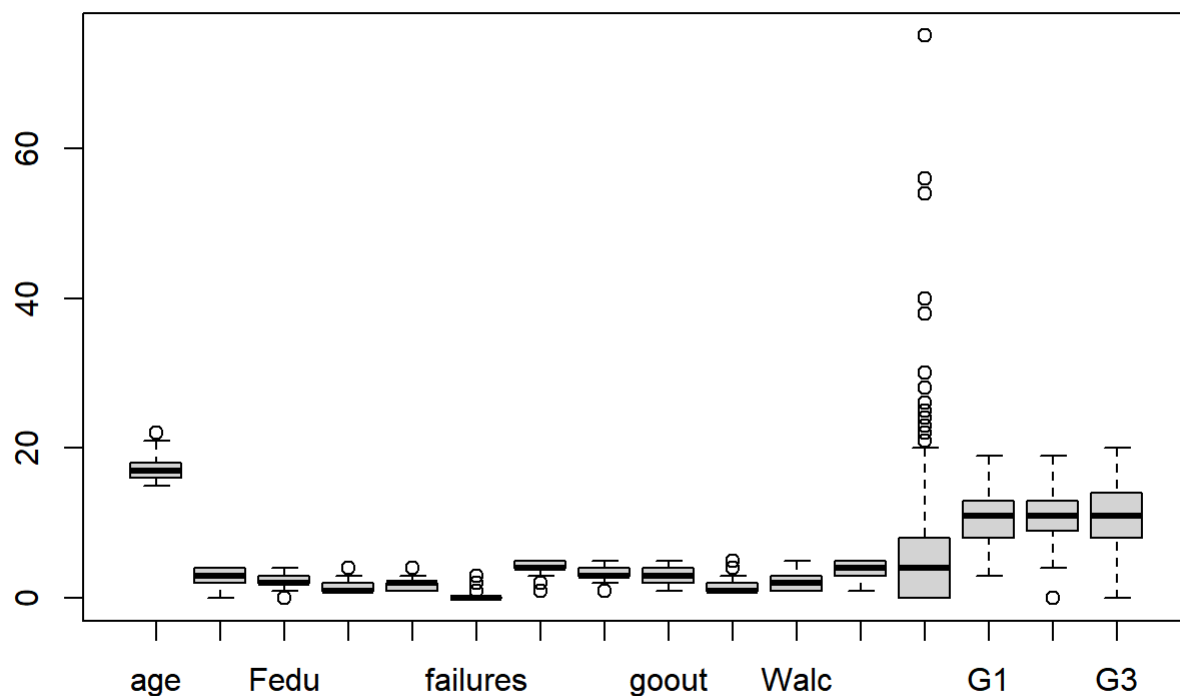
```
## school sex age address famsize Pstatus Medu
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## Fedu Mjob Fjob reason guardian traveltime studytime
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## failures schoolsup famsup paid activities nursery higher
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## internet romantic famrel freetime goout Dalc Walc
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## health absences G1 G2 G3
## FALSE FALSE FALSE FALSE FALSE
```

There is no NA in all columns.

2.3.2. Checking Outliers

```
boxplot(num, main="Multiplot Visualization for numeric variables" )
```

Multiplot Visualization for numeric variables



Absences variable has many outliers.

2.3.3. Data Partition

Let's split dataset into trainset and testset.

```
set.seed(3, sample.kind = "Rounding")
```

```
## Warning in set.seed(3, sample.kind = "Rounding"): non-uniform 'Rounding' sampler
## used
```

```
partition <- createDataPartition(new_mat[, 'G3'], times=1, p=0.80, list=FALSE)
new_mat_trainset<-new_mat[partition,]
new_mat_testset<-new_mat[-partition,]

partition <- createDataPartition(ex_new_mat[, 'G3'], times=1, p=0.80, list=FALSE)
ex_new_mat_trainset<-ex_new_mat[partition,]
ex_new_mat_testset<-ex_new_mat[-partition,]

partition <- createDataPartition(ex_bi[, 'G3'], times=1, p=0.80, list=FALSE)
ex_bi_trainset<-ex_bi[partition,]
ex_bi_testset<-ex_bi[-partition,]
```

3. Modelling

In this section, the process and technique are the following.

a. Process and Techniques

We will go through 4 steps:

- Training models with train datasets
- Predicting test datasets
- Evaluating models using accuracy instrument
- Showing accumulated results tables

b. Models

We will apply several machine learning algorithms and build following models to predict whether students would pass or fail the final grade G3.

- Model 1: Guessing Model
- Model 2: Logistic regression Model
- Model 3: Simplified Logistic Regression with Significant Variables
- Model 4: Cross-validated Decision Tree
- Model 5: Cross-validated KNN Model
- Model 6: Random Forest Model

3.1. Model 1: Baseline prediction by randomly guessing the outcome

The simplest prediction method is randomly guessing the outcome, whether that person passed or not by sampling from the vector $c(0,1)$, without using additional predictors. These methods will help us determine whether our machine learning algorithm performs better than chance. How accurate are this method of guessing students' final test?

Let's apply this method to different datasets.

Modeling and Prediction

```
# set.seed(3)
set.seed(1, sample.kind = "Rounding")
```

```
## Warning in set.seed(1, sample.kind = "Rounding"): non-uniform 'Rounding' sampler
## used
```

```
# Guess with equal probability of pass
guess <- sample(c(0,1), nrow(new_mat_trainset), replace = TRUE)
mean(guess == new_mat_testset$G3)
```

```
## [1] 0.5094937
```

```
# set.seed(3)
set.seed(2, sample.kind = "Rounding")
```

```
## Warning in set.seed(2, sample.kind = "Rounding"): non-uniform 'Rounding' sampler
## used
```

```
# guess with equal probability of pass with ex_new_mat dataset
guess <- sample(c(0,1), nrow(ex_new_mat_trainset), replace = TRUE)
mean(guess == ex_new_mat_testset$G3)
```

```
## [1] 0.5
```

```
# set.seed(3)
set.seed(3, sample.kind = "Rounding")
```

```
## Warning in set.seed(3, sample.kind = "Rounding"): non-uniform 'Rounding' sampler
## used
```

```
# guess with equal probability of pass with ex_bi dataset
guess <- sample(c(0,1), nrow(ex_bi_trainset), replace = TRUE)
guess_results<-mean(guess == ex_bi_testset$G3)
guess_results
```

```
## [1] 0.5506329
```

The best result of guessing method was when applying to binary dataset, ex_bi.

Result Table

```
results <- tibble(Method = "Model 1: Guessing Model",
                  Accuracy = guess_results)
results %>% knitr::kable()
```

Method	Accuracy
Model 1: Guessing Model	0.5506329

3.2. Model 2: Logistic Regression Model

Modeling

```
# Regression_glm
set.seed(3, sample.kind = "Rounding") # set.seed function is to ensure that the samples produced are reproducible
```

```
## Warning in set.seed(3, sample.kind = "Rounding"): non-uniform 'Rounding' sampler
## used
```

```
ex_new_mat_log_model<-glm(G3~., data=ex_new_mat_trainset, family=binomial)
ex_new_mat_log_model
```

```
##
## Call: glm(formula = G3 ~ ., family = binomial, data = ex_new_mat_trainset)
##
## Coefficients:
##      (Intercept)      schoolMS      sexM      age
##      2.23771      0.76330      0.94293     -0.06147
##      addressU      famsizeLE3      PstatusT      Medu
##      0.74591     -0.17608     -0.44369     -0.28938
##      Fedu      Mjobhealth      Mjobother      Mjobservices
##      0.23010      1.28452     -0.57569     -0.11914
##      Mjobteacher      Fjobhealth      Fjobother      Fjobservices
##      -0.24950     -1.53160     -1.71661     -1.36784
##      Fjobteacher      reasonhome      reasonother      reasonreputation
##      -1.30942      0.60406      0.09148      0.73495
##      guardianmother      guardianother      traveltime      studytime
##      0.26539      0.23276      0.10769      0.46975
##      failures      schoolsupyes      famsupyes      paidyes
##      -0.75186     -0.24418     -0.55739      0.91928
##      activitiesyes      nurseryyes      higheryes      internetyes
##      0.07638     -0.38931     -0.76442      0.20250
##      romanticyes      famrel      freetime      goout
##      -0.24402      0.30369      0.19856     -0.63679
##      Dalc      Walc      health      absences
##      -0.13257      0.52120      0.10128      0.05499
##
## Degrees of Freedom: 315 Total (i.e. Null); 276 Residual
## Null Deviance: 295.2
## Residual Deviance: 229.9 AIC: 309.9
```

```
summary(ex_new_mat_log_model)
```

```
##
## Call:
## glm(formula = G3 ~ ., family = binomial, data = ex_new_mat_trainset)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.9346   0.1728   0.3856   0.6017   1.8849
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.23771    3.75878   0.595  0.55162
## schoolMS       0.76330    0.66523   1.147  0.25121
## sexM           0.94293    0.44269   2.130  0.03317 *
## age          -0.06147    0.17780  -0.346  0.72952
## addressU       0.74591    0.46703   1.597  0.11024
## famsizeLE3    -0.17608    0.42099  -0.418  0.67576
## PstatusT      -0.44369    0.60754  -0.730  0.46520
## Medu          -0.28938    0.27626  -1.047  0.29487
## Fedu           0.23010    0.23742   0.969  0.33247
## Mjobhealth     1.28452    1.08097   1.188  0.23471
## Mjobother     -0.57569    0.58358  -0.986  0.32390
## Mjobservices  -0.11914    0.66615  -0.179  0.85806
## Mjobteacher   -0.24950    0.89259  -0.280  0.77985
## Fjobhealth    -1.53160    1.58479  -0.966  0.33383
## Fjobother     -1.71661    1.15902  -1.481  0.13858
## Fjobservices  -1.36784    1.19278  -1.147  0.25148
## Fjobteacher   -1.30942    1.31856  -0.993  0.32068
## reasonhome     0.60406    0.47669   1.267  0.20509
## reasonother    0.09148    0.71971   0.127  0.89886
## reasonreputation 0.73495    0.49658   1.480  0.13886
## guardianmother 0.26539    0.48548   0.547  0.58462
## guardianother  0.23276    0.82649   0.282  0.77823
## traveltime     0.10769    0.26782   0.402  0.68760
## studytime      0.46975    0.25161   1.867  0.06191 .
## failures      -0.75186    0.23788  -3.161  0.00157 **
## schoolsupyes   -0.24418    0.49460  -0.494  0.62153
## famsupyes     -0.55739    0.39948  -1.395  0.16293
## paidyes       0.91928    0.42172   2.180  0.02927 *
## activitiesyes  0.07638    0.37284   0.205  0.83767
## nurseryyes    -0.38931    0.46317  -0.841  0.40061
## higheryes     -0.76442    0.88052  -0.868  0.38532
## internetyes    0.20250    0.46008   0.440  0.65984
## romanticyes   -0.24402    0.39948  -0.611  0.54130
## famrel        0.30369    0.19889   1.527  0.12677
## freetime      0.19856    0.19914   0.997  0.31871
## goout        -0.63679    0.19367  -3.288  0.00101 **
## Dalc         -0.13257    0.29741  -0.446  0.65579
## Walc          0.52120    0.22958   2.270  0.02319 *
## health        0.10128    0.13203   0.767  0.44305
## absences      0.05499    0.03117   1.764  0.07775 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 295.24 on 315 degrees of freedom
## Residual deviance: 229.94 on 276 degrees of freedom
## AIC: 309.94
##
## Number of Fisher Scoring iterations: 6
```

Prediction

```
predict_log1<-predict(ex_new_mat_log_model, newdata=ex_new_mat_testset, type="response")
predict_log1<-round(predict_log1)
log_results<-mean(predict_log1==ex_new_mat_testset$G3)
log_results
```

```
## [1] 0.8227848
```

Result Table

```
results <- bind_rows(results,
                      tibble(Method = "Model 2: Logistic Regression Model",
                             Accuracy = log_results))
results %>% knitr::kable()
```

Method	Accuracy
Model 1: Guessing Model	0.5506329
Model 2: Logistic Regression Model	0.8227848

We used the logistic regression algorithm to predict whether the students would pass or not. The output of this model are probabilities to happen with comparison with an unbiased threshold of 0.5. We see that around 82% of results were accurately predicted despite of the fact that we has removed decisive variables such as G1, G2.

According to the summary of the model, the significant predictors are failures, sex, paid, goout, Walc, goout, studytime and absences.

3.3. Model 3: Simplified Logistic Regression Model with Significant Variables

Modeling

Correlation EDA shows that significant variables are sex, age, address, famsize, Medu, Fedu, Mjob, Fjob, traveltime, studytime, failures, schoolsup, paid, higher, romantic and goout. Let's simplify variable and apply to the regression model.

```
set.seed(3, sample.kind = "Rounding") # set.seed function is to ensure that the samples produced a
re reproducible
```

```
## Warning in set.seed(3, sample.kind = "Rounding"): non-uniform 'Rounding' sampler
## used
```

```
sig_log_model<-glm(G3~failures + sex + address + age + Medu + Fedu+ Mjob + Fjob + schoolsup + paid
+ higher + romantic + goout + studytime + traveltime + famsize, data=ex_new_mat_trainset, family=
binomial)
sig_log_model
```

```
##
## Call:  glm(formula = G3 ~ failures + sex + address + age + Medu + Fedu +
##       Mjob + Fjob + schoolsup + paid + higher + romantic + goout +
##       studytime + traveltime + famsize, family = binomial, data = ex_new_mat_trainset)
##
## Coefficients:
## (Intercept)      failures          sexM      addressU          age
##      1.68324      -0.63189       1.12062       0.41892       0.07304
##      Medu      Fedu      Mjobhealth      Mjobother      Mjobservices
##     -0.18921      0.17710       0.95169      -0.35831      -0.04618
##      Mjobteacher      Fjobhealth      Fjobother      Fjobservices      Fjobteacher
##     -0.10448     -0.91705      -1.31523      -1.03912     -1.48354
##      schoolsupyes      paidyes      higheryes      romanticyes      goout
##     -0.39199      0.89304      -0.29514      -0.13208     -0.32342
##      studytime      traveltime      famsizeLE3
##      0.20733      0.11758      0.14503
##
## Degrees of Freedom: 315 Total (i.e. Null);  293 Residual
## Null Deviance:      295.2
## Residual Deviance: 251.4    AIC: 297.4
```

```
summary(sig_log_model)
```



```
##
## Call:
## glm(formula = G3 ~ failures + sex + address + age + Medu + Fedu +
##       Mjob + Fjob + schoolsup + paid + higher + romantic + goout +
##       studytime + traveltime + famsize, family = binomial, data = ex_new_mat_trainset)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.6455   0.2621   0.4401   0.6198   1.8788
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.68324    2.96081   0.569  0.56969
## failures      -0.63189    0.20239  -3.122  0.00180 **
## sexM           1.12062    0.38973   2.875  0.00404 **
## addressU       0.41892    0.40268   1.040  0.29818
## age            0.07304    0.13614   0.536  0.59163
## Medu          -0.18921    0.23960  -0.790  0.42971
## Fedu           0.17710    0.20338   0.871  0.38385
## Mjobhealth     0.95169    0.98230   0.969  0.33263
## Mjobother     -0.35831    0.51707  -0.693  0.48833
## Mjobservices  -0.04618    0.60155  -0.077  0.93881
## Mjobteacher   -0.10448    0.81600  -0.128  0.89812
## Fjobhealth    -0.91705    1.53680  -0.597  0.55069
## Fjobother     -1.31523    1.09147  -1.205  0.22820
## Fjobservices  -1.03912    1.12395  -0.925  0.35521
## Fjobteacher   -1.48354    1.24577  -1.191  0.23371
## schoolsupyes  -0.39199    0.46198  -0.849  0.39616
## paidyes        0.89304    0.37089   2.408  0.01605 *
## higheryes     -0.29514    0.75445  -0.391  0.69565
## romanticyes   -0.13208    0.36040  -0.366  0.71401
## goout         -0.32342    0.14857  -2.177  0.02949 *
## studytime      0.20733    0.22268   0.931  0.35182
## traveltime     0.11758    0.24295   0.484  0.62842
## famsizeLE3     0.14503    0.38494   0.377  0.70636
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 295.24  on 315  degrees of freedom
## Residual deviance: 251.35  on 293  degrees of freedom
## AIC: 297.35
##
## Number of Fisher Scoring iterations: 5
```

Prediction

```

predict_log2<-predict(sig_log_model, newdata=ex_new_mat_testset, type="response")
predict_log2<-round(predict_log2)
sig_log_results<-mean(predict_log2==ex_new_mat_testset$G3)
sig_log_results

```

```
## [1] 0.835443
```

Accuracy has been improved from around 0.82 to 0.84.

Result Table

```

results <- bind_rows(results,
                      tibble(Method = "Model 3: Simplified Logistic Regression Model with Significa
nt Variables",
                             Accuracy = sig_log_results))
results %>% knitr::kable()

```

Method	Accuracy
Model 1: Guessing Model	0.5506329
Model 2: Logistic Regression Model	0.8227848
Model 3: Simplified Logistic Regression Model with Significant Variables	0.8354430

3.4. Model 4: Decision Tree

The Decision Tree(DT) looks like upside down trees where the root of the tree is on top and one goes down step by step towards the leaves. Its internal rule looks like IF-THEN rules in r. Each internal node represents a “test” on an attribute. Each branch represents the outcome of the test and each leaf node represents a class label.

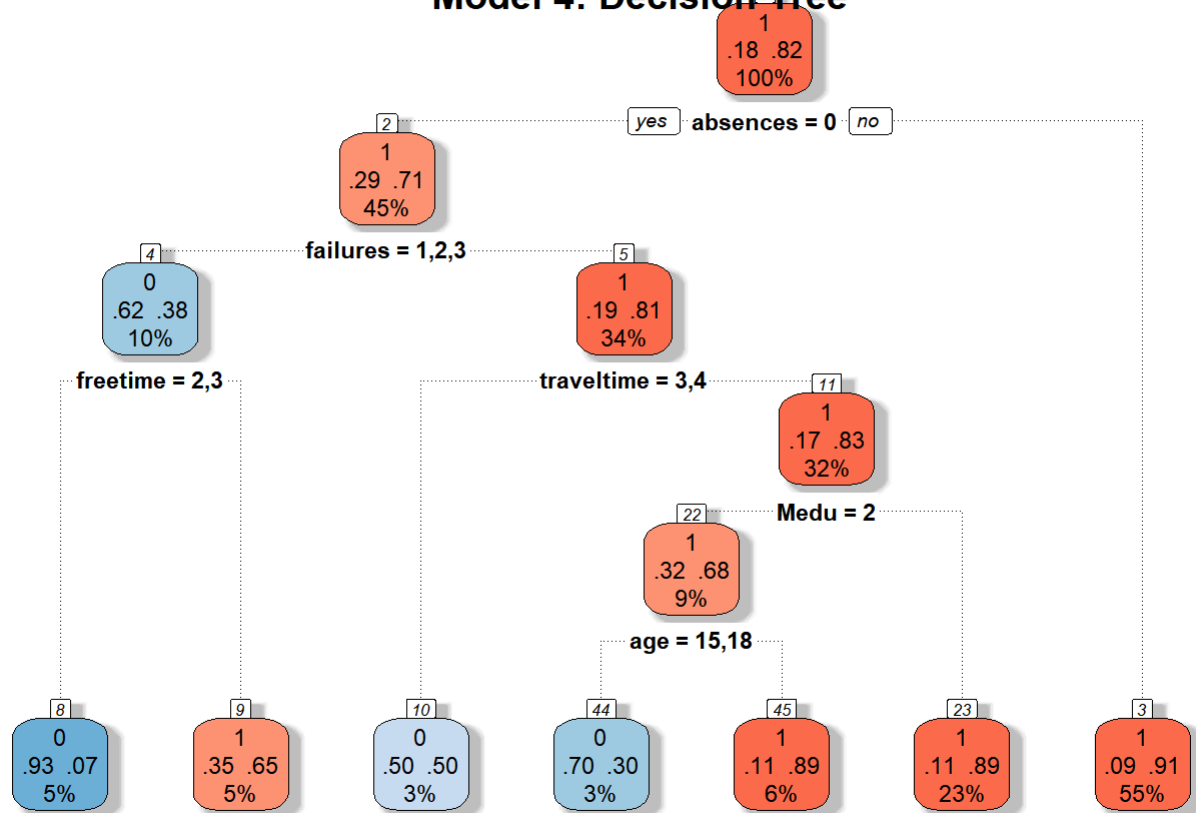
Visualization

```

dtree<-rpart(G3~., data=ex_bi_trainset)
fancyRpartPlot(dtree, main="Model 4: Decision Tree", palettes=c("Blues", "Reds"))

```

Model 4: Decision Tree



Rattle 2021-Nov-21 17:32:48 user

Modeling

```
# Cross validation of the model.
# Let's do? cross validation of the model to assess how the results of a statistical analysis will
# generalize with the optimal cp.
```

```
set.seed(3, sample.kind = "Rounding")
```

```
## Warning in set.seed(3, sample.kind = "Rounding"): non-uniform 'Rounding' sampler
## used
```

```
tc <- trainControl(method = "cv",
                   number = 10)
cp_grid <- expand.grid(cp = seq(0, 0.03, 0.001))
```

```
dtree_cv <- train(G3~.,
                 na.action=na.omit,
                 data = ex_bi_trainset,
                 method = "rpart",
                 trControl = tc,
                 tuneGrid = cp_grid)
```

```
## Warning in (function (kind = NULL, normal.kind = NULL, sample.kind = NULL) :  
## non-uniform 'Rounding' sampler used
```

```
dtree_cv
```

```
## CART
##
## 316 samples
## 30 predictor
## 2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 174, 174, 173, 173, 174, 173, ...
## Resampling results across tuning parameters:
##
##   cp      Accuracy   Kappa
##   0.000  0.7623684  0.2041609
##   0.001  0.7623684  0.2041609
##   0.002  0.7623684  0.2041609
##   0.003  0.7623684  0.2041609
##   0.004  0.7623684  0.2041609
##   0.005  0.7623684  0.2041609
##   0.006  0.7676316  0.1913169
##   0.007  0.7676316  0.1913169
##   0.008  0.7676316  0.1913169
##   0.009  0.7676316  0.1913169
##   0.010  0.7676316  0.1913169
##   0.011  0.7676316  0.1913169
##   0.012  0.7676316  0.1913169
##   0.013  0.7676316  0.1913169
##   0.014  0.7781579  0.1761881
##   0.015  0.7781579  0.1761881
##   0.016  0.7781579  0.1761881
##   0.017  0.7781579  0.1761881
##   0.018  0.7781579  0.1761881
##   0.019  0.7781579  0.1761881
##   0.020  0.7781579  0.1761881
##   0.021  0.7781579  0.1761881
##   0.022  0.7781579  0.1761881
##   0.023  0.7781579  0.1761881
##   0.024  0.7781579  0.1761881
##   0.025  0.7781579  0.1761881
##   0.026  0.7781579  0.1761881
##   0.027  0.7781579  0.1761881
##   0.028  0.7781579  0.1761881
##   0.029  0.7728947  0.1284817
##   0.030  0.7728947  0.1284817
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was cp = 0.028.
```

Prediction

```
# Test the model in the fnew_mat_testset with the optimal value for the model, cp=0.027.
set.seed(3, sample.kind = "Rounding")
```

```
## Warning in set.seed(3, sample.kind = "Rounding"): non-uniform 'Rounding' sampler
## used
```

```
cp_grid <- expand.grid(cp = 0.027)

dtree_cv <- train(G3~.,
                  data = ex_bi_testset,
                  na.action=na.omit,
                  method = "rpart",
                  trControl = tc,
                  tuneGrid = cp_grid)
```

```
## Warning in (function (kind = NULL, normal.kind = NULL, sample.kind = NULL) :
## non-uniform 'Rounding' sampler used
```

```
dtree_cv
```

```
## CART
##
## 79 samples
## 30 predictors
## 2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 45, 46, 46, 46, 46, 46, ...
## Resampling results:
##
##   Accuracy   Kappa
##  0.7833333  -0.025
##
## Tuning parameter 'cp' was held constant at a value of 0.027
```

Result

```
# What is the accuracy on the test set using the cross-validated DT model?
dt_pred <- predict(dtree_cv, ex_bi_testset)
dt_results <- mean(dt_pred == ex_bi_testset$G3) # accuracy of the KNN model on the test set
```

```
## Warning in `==.default`(dt_pred, ex_bi_testset$G3): longer object length is not a
## multiple of shorter object length
```

```
## Warning in is.na(e1) | is.na(e2): longer object length is not a multiple of
## shorter object length
```

```
dt_results
```

```
## [1] 0.8227848
```

Result Table

```
results <- bind_rows(results,  
                      tibble(Method = "Model 4: Cross-validated Decison Tree Model",  
                             Accuracy = dt_results))  
results %>% knitr::kable()
```

Method	Accuracy
Model 1: Guessing Model	0.5506329
Model 2: Logistic Regression Model	0.8227848
Model 3: Simplified Logistic Regression Model with Significant Variables	0.8354430
Model 4: Cross-validated Decison Tree Model	0.8227848

3.5. Model 5: KNN (K-Nearest Neighbors)

Every student is stored. A new student is compared to the stored students. Let's find the k most similar students. The most common grade of the k students is given to a new student.

Modeling

```
# Let's use caret to train a decision tree with the rpart method.  
# The caret package performs cross validation for us and lets us train different algorithms using  
  simlart syntax.
```

```
# Predict  
#set.seed(6)  
set.seed(6, sample.kind = "Rounding") # if using R 3.6 or later
```

```
## Warning in set.seed(6, sample.kind = "Rounding"): non-uniform 'Rounding' sampler  
## used
```

```
train_knn_cv <- train(G3 ~ .,  
                      method = "knn",  
                      data = ex_bi_trainset, na.action=na.omit,  
                      tuneGrid = data.frame(k = seq(3, 51, 2)),# Try tuning with k=seq(3,51,2).  
                      trControl = trainControl(method = "cv", number = 10, p = 0.9))# 10-fold cross  
validation where each partition consists of 10% of the total.
```

```
## Warning in (function (kind = NULL, normal.kind = NULL, sample.kind = NULL) :  
## non-uniform 'Rounding' sampler used
```

```
# Optimal value of k using cross-validation  
train_knn_cv$bestTune # parameter that maximized the estimated accuracy
```

```
##      k  
## 25 51
```

The optimal value of k is 35.

```
print(train_knn_cv)
```



```
## k-Nearest Neighbors
##
## 316 samples
## 30 predictor
## 2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 173, 174, 174, 173, 174, 173, ...
## Resampling results across tuning parameters:
##
## k Accuracy Kappa
## 3 0.7578363 -0.005739167
## 5 0.7830702 -0.026086957
## 7 0.7880702 -0.017391304
## 9 0.7880702 -0.017391304
## 11 0.7980702 0.000000000
## 13 0.7930702 -0.008695652
## 15 0.7980702 0.000000000
## 17 0.7980702 0.000000000
## 19 0.7980702 0.000000000
## 21 0.7980702 0.000000000
## 23 0.7980702 0.000000000
## 25 0.7980702 0.000000000
## 27 0.7980702 0.000000000
## 29 0.7980702 0.000000000
## 31 0.7980702 0.000000000
## 33 0.7980702 0.000000000
## 35 0.7980702 0.000000000
## 37 0.7980702 0.000000000
## 39 0.7980702 0.000000000
## 41 0.7980702 0.000000000
## 43 0.7980702 0.000000000
## 45 0.7980702 0.000000000
## 47 0.7980702 0.000000000
## 49 0.7980702 0.000000000
## 51 0.7980702 0.000000000
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 51.
```

Results

```
# What is the accuracy on the test set using the cross-validated KNN model?
# Accuracy
knn_cv_pred <- predict(train_knn_cv, ex_bi_testset)
# mean(knn_cv_pred == fnew_mat_test_set$G3)
knn_cv_results<-mean(knn_cv_pred== ex_bi_testset$G3) # accuracy of the KNN model on the test set
```

```
## Warning in `==.default`(knn_cv_pred, ex_bi_testset$G3): longer object length is
## not a multiple of shorter object length
```

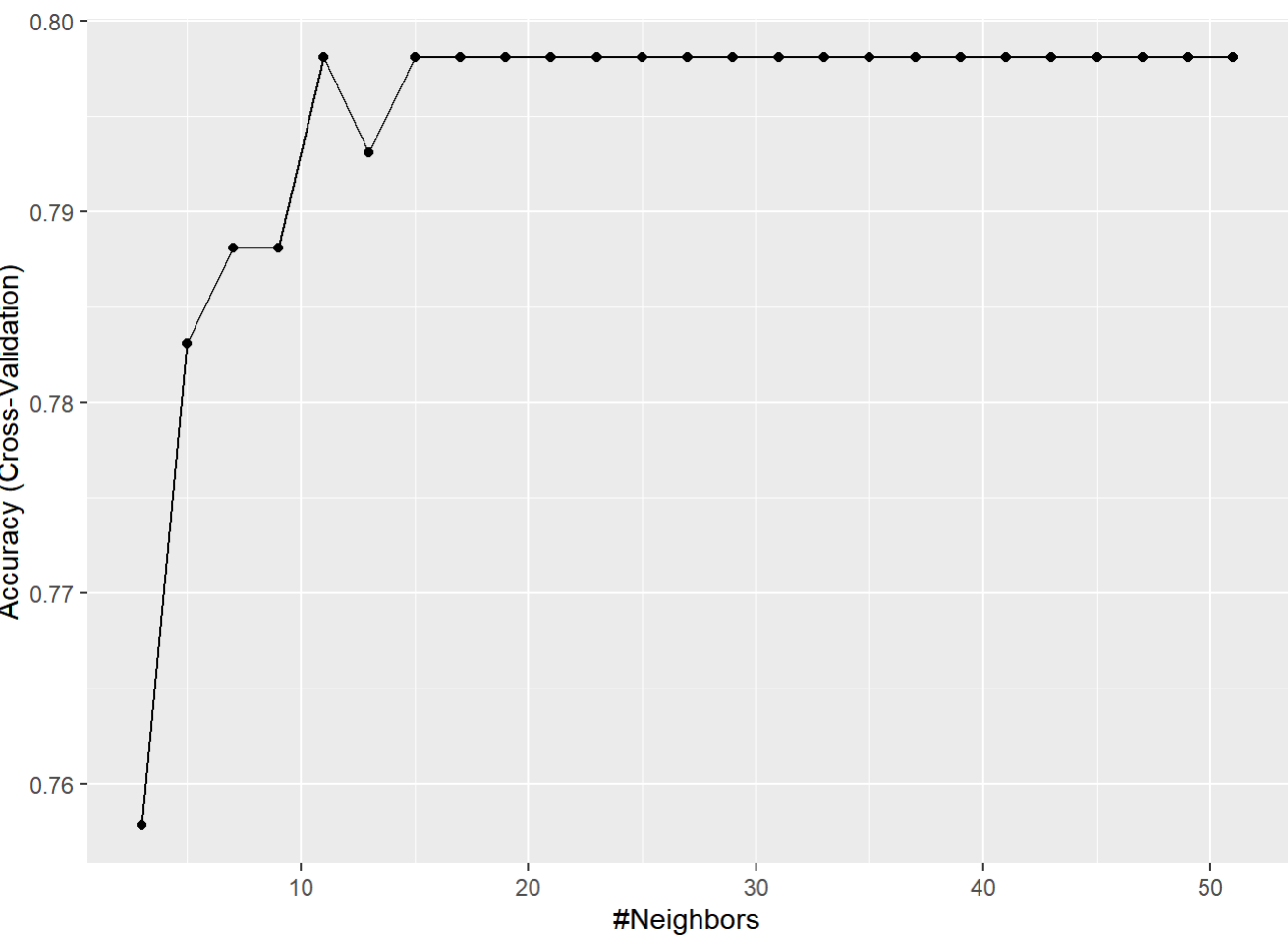
```
## Warning in is.na(e1) | is.na(e2): longer object length is not a multiple of
## shorter object length
```

```
knn_cv_results
```

```
## [1] 0.8227848
```

Visualization

```
ggplot(train_knn_cv)
```



Results Table

```
results <- bind_rows(results,
                      tibble(Method = "Model 5: Cross-validated KNN Model",
                             Accuracy = knn_cv_results))
results %>% knitr::kable()
```

Method	Accuracy
--------	----------

Method	Accuracy
Model 1: Guessing Model	0.5506329
Model 2: Logistic Regression Model	0.8227848
Model 3: Simplified Logistic Regression Model with Significant Variables	0.8354430
Model 4: Cross-validated Decison Tree Model	0.8227848
Model 5: Cross-validated KNN Model	0.8227848

3.6. Model 6. Random Forest Model

The Random Forest (RF) (Breiman 2001) is an ensemble unpruned Decision Tree(DT). Each tree is based on a random feature selection from bootstrap training samples and the RF predictions are built by averaging the outputs of the trees.

Modeling

```
# Set the seed to 14. Use the caret train() function with the rf method to train a random forest.
# Test values of mtry = seq(1:7). Set ntree to 100.
# What mtry value maximizes accuracy?
```

```
#set.seed(14)
set.seed(14, sample.kind = "Rounding")    # simulate R 3.5
```

```
## Warning in set.seed(14, sample.kind = "Rounding"): non-uniform 'Rounding'
## sampler used
```

```
train_rf <- train(G3 ~ .,
                  data = ex_bi_trainset,
                  na.action=na.omit,
                  method = "rf",
                  ntree = 100,
                  tuneGrid = data.frame(mtry = seq(1:7)))
```

```
## Warning in (function (kind = NULL, normal.kind = NULL, sample.kind = NULL) :
## non-uniform 'Rounding' sampler used
```

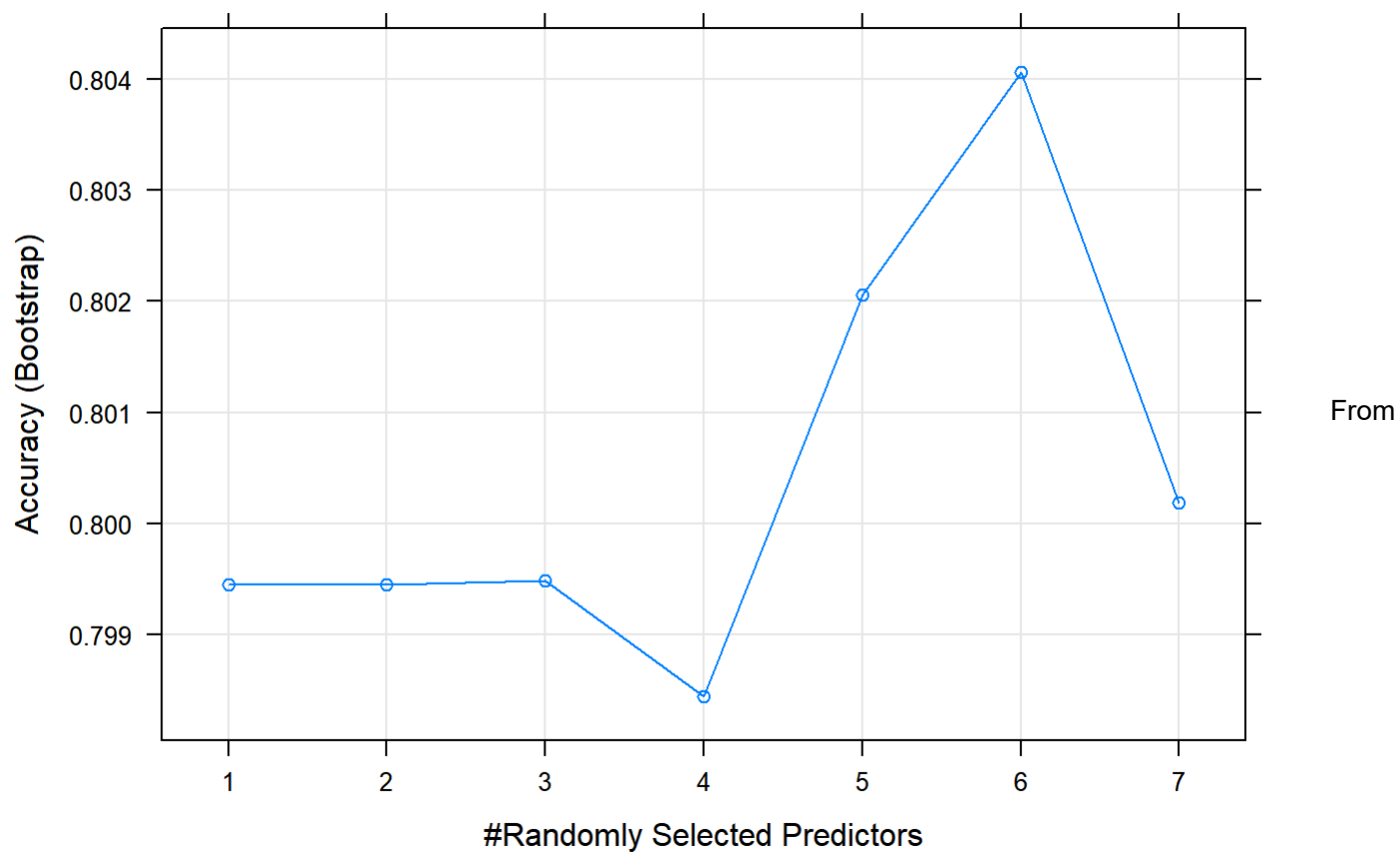
```
train_rf$bestTune
```

```
##      mtry
## 6      6
```

Visualization

```
# What is the best value for mtry? Lets try to find the optimal value for mtry parameter (Number of variables randomly sampled as candidates at each split).
```

```
plot(train_rf)
```



the graph above and from the results, we can conclude that optimal number of variables is 6

```
print(train_rf)
```

```
## Random Forest
##
## 316 samples
## 30 predictor
## 2 classes: '0', '1'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 193, 193, 193, 193, 193, ...
## Resampling results across tuning parameters:
##
## mtry Accuracy Kappa
## 1 0.7994464 0.00000000
## 2 0.7994464 0.00000000
## 3 0.7994827 0.01494237
## 4 0.7984391 0.03448286
## 5 0.8020542 0.07629003
## 6 0.8040673 0.09208346
## 7 0.8001894 0.10313475
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 6.
```

Result

```
# What is the accuracy of the random forest model on the test set?
rf_preds <- predict(train_rf, ex_bi_testset)
rf_results <- mean(rf_preds == ex_bi_testset$G3)
```

```
## Warning in `==.default`(rf_preds, ex_bi_testset$G3): longer object length is not a
## multiple of shorter object length
```

```
## Warning in is.na(e1) | is.na(e2): longer object length is not a multiple of
## shorter object length
```

```
rf_results
```

```
## [1] 0.7721519
```

Results Table

```
results <- bind_rows(results,
  tibble(Method = "Model 6: Random Forest Model",
    Accuracy = rf_results))
results %>% knitr::kable()
```

Method

Accuracy

Method	Accuracy
Model 1: Guessing Model	0.5506329
Model 2: Logistic Regression Model	0.8227848
Model 3: Simplified Logistic Regression Model with Significant Variables	0.8354430
Model 4: Cross-validated Decison Tree Model	0.8227848
Model 5: Cross-validated KNN Model	0.8227848
Model 6: Random Forest Model	0.7721519

```
# What is the most important variable?
varImp(train_rf)
```

```
## rf variable importance
##
## only 20 most important variables shown (out of 79)
##
## Overall
## romantico 100.00
## paidno 94.05
## failures1 92.59
## failures2 77.48
## nurseryno 59.59
## higherno 56.37
## health2 50.96
## absences2 49.79
## goout2 46.99
## guardianother 45.49
## freetime3 44.39
## addressR 44.30
## failures3 44.05
## activitiesno 42.99
## sexM 41.68
## Mjobat_home 40.78
## Medu1 40.35
## absences4 39.96
## Fedu1 38.13
## famsupno 38.12
```

Some important variables in the above table are failures, paid and goout as the same as correlation graphs but we can see slightly different result. For example, we excluded absences because of relatively smaller correlation coefficient but we can see absences variable is one of the important variable in rf model. The most striking feature is romantic is the most important variable in rf model unlike correlation table.

4. Results

Now, let's see which method has performed the best among guessing, Regression, DT, KNN and RF.

```
# Show the final result
results
```

```
## # A tibble: 6 x 2
##   Method                                     Accuracy
##   <chr>                                     <dbl>
## 1 Model 1: Guessing Model                   0.551
## 2 Model 2: Logistic Regression Model       0.823
## 3 Model 3: Simplified Logistic Regression Model with Significant Varia~ 0.835
## 4 Model 4: Cross-validated Decison Tree Model 0.823
## 5 Model 5: Cross-validated KNN Model       0.823
## 6 Model 6: Random Forest Model            0.772
```

The best model is Simplified Logistic Regression Model with Significant Variables with accuracy of around 0.84.

5. Conclusion

Summary

The goal of this paper was to predict students' final performance result of either pass or fail at the thought that educators might help students who has higher probability to fail in advance. To achieve the goal, the aim of this paper was to build a good model with high accuracy result for binary classification data and continuous data to approach some machine learning algorithms. During EDA, we had attempted to get influential predictors with which we wanted to build model such as Regreesion.

We collected a data named "student-mat" and get information from UCI website. We overviewed of the dataset and analyzed relation among variables. From data analysis we could find out important and influential variables that affect the result of the dependent variable. They are sex, age, address, famsize, Medu, Fedu, Mjob, Fjob, traveltime, studytime, failures, schoolsup, paid, higher, romantic, goout, G1 and G2. They are correlated with G3 compared to other variables. Indirect influential variables are famsup, school, internet, Dalc, Walc, freetime and gauardian variables. They are correlated with the influential variables that is correlated with G3 directly. Excluded variables are Pstatus, reason, famsup, activities, nursery, famrel, health and absences variables that are less correlated with G3 directly and indirectly.

Before builing up several algorithms, we modified our dataset by checking NA and outliers and by splitting into trainset and testset. We approached some algorithms such as Guessing, Regression, Decision Trees, KNN and Random Forest method training each model with trainset and predicting with testset. The models were evaluated by accuracy instrument and the best optimal model was selected: Simplified Logistic Regression Model with Significant Variables with accuracy of around 0.84.

Potential Impact

This paper may have potential impact in that we get some implied information in education field from the raw data, which would make educators to approach presumable week students and give more attention. This might make contribution to achieve equity in education domain. This model can be used for predicting students with a fairly decent accuracy.

Limiation

We modified dataset into several types and applied in accordance with a model that requires specific type of data. Despite of the fact of using different type of dataset, the content remained the same meaning. We just wanted to find out the best model. However, it would be more useful to compare if we use the same data type. In addition, we

may apply to other algorithms that we don't know yet. Further study is needed for a better promising model.

Further work:

To get a better result, we may add more data and apply other unexplored machine learning methods. In addition, more research is need to study on consideration on how educators can supplement the decisive and deficit factors on their grades. Also, real data related either Singapore or Korea is needed in order to apply in practice. Further, research on online learning environment would be potential especially in the pandemic situation like Covid-19.

Reference

P. Cortez and A. Silva. Using Data Mining to Predict Secondary School Student Performance. In A. Brito and J. Teixeira Eds., Proceedings of 5th FUTURE BUSINESS TECHNOLOGY Conference (FUBUTEC 2008) pp. 5-12, Porto, Portugal, April, 2008, EUROESIS, ISBN 978-9077381-39-7.