

Predicting Student Success Using Machine Learning

2022-06-26 by Norah Rayfield

```
#Initial Analysis
```

```
#All code can be found in Github at https://github.com/RayfieldNorah/CIND-820.git.
```

```
# Get the data, check data types of the attributes and install all needed packages and Libraries
#install.packages("ggplot2")
#install.packages("ggcorrplot")
#install.packages("dplyr")
#install.packages("rlang")
#install.packages("magrittr")
#install.packages("caret")
#install.packages("psych")
```

```
library(InformationValue)
```

```
## Warning: package 'InformationValue' was built under R version 4.1.3
```

```
library(ISLR)
```

```
## Warning: package 'ISLR' was built under R version 4.1.3
```

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.1.3
```

```
## Loading required package: ggplot2
```

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

```
##
## Attaching package: 'caret'
```

```
## The following objects are masked from 'package:InformationValue':
##
##     confusionMatrix, precision, sensitivity, specificity
```

```
library(ggcorrplot)
library(ggplot2)
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(rlang)
```

```
## Warning: package 'rlang' was built under R version 4.1.3
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.1.3
```

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

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

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

```
library(psych)
```

```
## Warning: package 'psych' was built under R version 4.1.3
```

```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
library(magrittr)
```

```
##
## Attaching package: 'magrittr'
```

```
## The following object is masked from 'package:rlang':  
##  
##     set_names
```

```
library(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 4.1.3
```

```
##  
## Attaching package: 'tidyr'
```

```
## The following object is masked from 'package:magrittr':  
##  
##     extract
```

```
library(factoextra)
```

```
## Warning: package 'factoextra' was built under R version 4.1.3
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```
library(devtools)
```

```
## Warning: package 'devtools' was built under R version 4.1.3
```

```
## Loading required package: usethis
```

```
## Warning: package 'usethis' was built under R version 4.1.3
```

```
library(mlbench)
```

```
## Warning: package 'mlbench' was built under R version 4.1.3
```

```
library(car)
```

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

```
## Loading required package: carData
```

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

```
##  
## Attaching package: 'car'
```

```
## The following object is masked from 'package:psych':  
##  
##     logit
```

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

```
library(class)  
library(rpart)  
library(rpart.plot)
```

```
## Warning: package 'rpart.plot' was built under R version 4.1.3
```

```
library(RColorBrewer)  
library(rattle)
```

```
## Warning: package 'rattle' was built under R version 4.1.3
```

```
## Loading required package: tibble
```

```
## Loading required package: bitops
```

```
##  
## Attaching package: 'bitops'
```

```
## The following object is masked from 'package:rlang':  
##  
##     %|%
```

```
## Rattle: A free graphical interface for data science with R.  
## Version 5.5.1 Copyright (c) 2006-2021 Togaware Pty Ltd.  
## Type 'rattle()' to shake, rattle, and roll your data.
```

```
library(performance)
```

```
## Warning: package 'performance' was built under R version 4.1.3
```

```

mathdata=read.table("student-mat.csv",sep=";",header=TRUE,
                    stringsAsFactors = TRUE)
langdata=read.table("student-por.csv",sep=";",header=TRUE, stringsAsFactors = TRUE)

str(mathdata)

```

```

## 'data.frame':   395 obs. of  33 variables:
## $ school    : Factor w/ 2 levels "GP","MS": 1 1 1 1 1 1 1 1 1 1 ...
## $ sex        : Factor w/ 2 levels "F","M": 1 1 1 1 1 2 2 1 2 2 ...
## $ age         : int  18 17 15 15 16 16 17 15 15 ...
## $ address    : Factor w/ 2 levels "R","U": 2 2 2 2 2 2 2 2 2 2 ...
## $ famsize    : Factor w/ 2 levels "GT3","LE3": 1 1 2 1 1 2 2 1 2 1 ...
## $ Pstatus    : Factor w/ 2 levels "A","T": 1 2 2 2 2 2 2 1 1 2 ...
## $ 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        : Factor w/ 5 levels "at_home","health",...: 1 1 1 2 3 4 3 3 4 3 ...
## $ Fjob        : Factor w/ 5 levels "at_home","health",...: 5 3 3 4 3 3 3 5 3 3 ...
## $ reason      : Factor w/ 4 levels "course","home",...: 1 1 3 2 2 4 2 2 2 2 ...
## $ guardian    : Factor w/ 3 levels "father","mother",...: 2 1 2 2 1 2 2 2 2 2 ...
## $ traveltimes: 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  : Factor w/ 2 levels "no","yes": 2 1 2 1 1 1 1 2 1 1 ...
## $ famsup     : Factor w/ 2 levels "no","yes": 1 2 1 2 2 2 1 2 2 2 ...
## $ paid        : Factor w/ 2 levels "no","yes": 1 1 2 2 2 2 1 1 2 2 ...
## $ activities: Factor w/ 2 levels "no","yes": 1 1 1 2 1 2 1 1 1 2 ...
## $ nursery     : Factor w/ 2 levels "no","yes": 2 1 2 2 2 2 2 2 2 2 ...
## $ higher     : Factor w/ 2 levels "no","yes": 2 2 2 2 2 2 2 2 2 2 ...
## $ internet   : Factor w/ 2 levels "no","yes": 1 2 2 2 1 2 2 1 2 2 ...
## $ romantic   : Factor w/ 2 levels "no","yes": 1 1 1 2 1 1 1 1 1 1 ...
## $ 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 ...

```

```

str(langdata)

```

```

## 'data.frame': 649 obs. of 33 variables:
## $ school : Factor w/ 2 levels "GP","MS": 1 1 1 1 1 1 1 1 1 1 ...
## $ sex    : Factor w/ 2 levels "F","M": 1 1 1 1 1 2 2 1 2 2 ...
## $ age    : int 18 17 15 15 16 16 16 17 15 15 ...
## $ address: Factor w/ 2 levels "R","U": 2 2 2 2 2 2 2 2 2 2 ...
## $ famsize: Factor w/ 2 levels "GT3","LE3": 1 1 2 1 1 2 2 1 2 1 ...
## $ Pstatus: Factor w/ 2 levels "A","T": 1 2 2 2 2 2 1 1 2 1 ...
## $ 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   : Factor w/ 5 levels "at_home","health",...: 1 1 1 2 3 4 3 3 4 3 ...
## $ Fjob   : Factor w/ 5 levels "at_home","health",...: 5 3 3 4 3 3 3 5 3 3 ...
## $ reason : Factor w/ 4 levels "course","home",...: 1 1 3 2 2 4 2 2 2 2 ...
## $ guardian: Factor w/ 3 levels "father","mother",...: 2 1 2 2 1 2 2 2 2 2 ...
## $ traveltime: 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 0 0 0 0 0 0 0 0 ...
## $ schoolsup: Factor w/ 2 levels "no","yes": 2 1 2 1 1 1 1 1 2 1 1 ...
## $ famsup  : Factor w/ 2 levels "no","yes": 1 2 1 2 2 2 1 2 2 2 ...
## $ paid    : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
## $ activities: Factor w/ 2 levels "no","yes": 1 1 1 2 1 2 1 1 1 2 ...
## $ nursery : Factor w/ 2 levels "no","yes": 2 1 2 2 2 2 2 2 2 2 ...
## $ higher  : Factor w/ 2 levels "no","yes": 2 2 2 2 2 2 2 2 2 2 ...
## $ internet: Factor w/ 2 levels "no","yes": 1 2 2 2 1 2 2 1 2 2 ...
## $ romantic: Factor w/ 2 levels "no","yes": 1 1 1 2 1 1 1 1 1 1 ...
## $ 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 4 2 6 0 0 6 0 2 0 0 ...
## $ G1     : int 0 9 12 14 11 12 13 10 15 12 ...
## $ G2     : int 11 11 13 14 13 12 12 13 16 12 ...
## $ G3     : int 11 11 12 14 13 13 13 13 17 13 ...

```

Check for any missing values.

```
colSums(is.na(mathdata))
```

	school	sex	age	address	famsize	Pstatus	Medu
##	0	0	0	0	0	0	0
##	Fedu	Mjob	Fjob	reason	guardian	traveltime	studytime
##	0	0	0	0	0	0	0
##	failures	schoolsup	famsup	paid	activities	nursery	higher
##	0	0	0	0	0	0	0
##	internet	romantic	famrel	freetime	goout	Dalc	Walc
##	0	0	0	0	0	0	0
##	health	absences	G1	G2	G3		
##	0	0	0	0	0		

```
colSums(is.na(langdata))
```

```
## school sex age address famsize Pstatus Medu
## 0 0 0 0 0 0 0
## Fedu Mjob Fjob reason guardian traveltime studytime
## 0 0 0 0 0 0 0
## failures schoolsup famsup paid activities nursery higher
## 0 0 0 0 0 0 0
## internet romantic famrel freetime goout Dalc Walc
## 0 0 0 0 0 0 0
## health absences G1 G2 G3
## 0 0 0 0 0
```

```
# There are no missing values.
```

```
#Look at descriptive stats for the data
summary(mathdata)
```

```

## school sex      age      address famsize Pstatus     Medu
## GP:349 F:208   Min.    :15.0   R: 88   GT3:281   A: 41   Min.    :0.000
## MS: 46  M:187  1st Qu.:16.0   U:307   LE3:114   T:354   1st Qu.:2.000
##                               Median :17.0
##                               Mean    :16.7
##                               3rd Qu.:18.0
##                               Max.    :22.0
##                               Max.    :4.000
##       Fedu      Mjob      Fjob      reason    guardian
## Min.    :0.000  at_home : 59  at_home : 20  course    :145  father: 90
## 1st Qu.:2.000  health  : 34  health  : 18  home     :109  mother:273
## Median :2.000  other   :141  other   :217  other    : 36  other  : 32
## Mean    :2.522  services:103 services:111 reputation:105
## 3rd Qu.:3.000  teacher : 58  teacher : 29
## Max.    :4.000
## traveltime   studytime   failures   schoolsup  famsup    paid
## Min.    :1.000  Min.    :1.000  Min.    :0.0000  no :344  no :153  no :214
## 1st Qu.:1.000  1st Qu.:1.000  1st Qu.:0.0000 yes: 51  yes:242 yes:181
## Median :1.000  Median :2.000  Median :0.0000
## Mean    :1.448  Mean    :2.035  Mean    :0.3342
## 3rd Qu.:2.000  3rd Qu.:2.000  3rd Qu.:0.0000
## Max.    :4.000  Max.    :4.000  Max.    :3.0000
## activities nursery higher internet romantic famrel
## no :194    no : 81  no : 20  no : 66  no :263  Min.    :1.000
## yes:201   yes:314 yes:375 yes:329 yes:132  1st Qu.:4.000
##                               Median :4.000
##                               Mean   :3.944
##                               3rd Qu.:5.000
##                               Max.   :5.000
##       freetime   goout      Dalc      Walc
## Min.    :1.000  Min.    :1.000  Min.    :1.000  Min.    :1.000
## 1st Qu.:3.000  1st Qu.:2.000  1st Qu.:1.000  1st Qu.:1.000
## Median :3.000  Median :3.000  Median :1.000  Median :2.000
## Mean    :3.235  Mean    :3.109  Mean    :1.481  Mean    :2.291
## 3rd Qu.:4.000  3rd Qu.:4.000  3rd Qu.:2.000  3rd Qu.:3.000
## Max.    :5.000  Max.    :5.000  Max.    :5.000  Max.    :5.000
##       health      absences   G1        G2
## Min.    :1.000  Min.    : 0.000  Min.    : 3.00  Min.    : 0.00
## 1st Qu.:3.000  1st Qu.: 0.000  1st Qu.: 8.00  1st Qu.: 9.00
## Median :4.000  Median : 4.000  Median :11.00  Median :11.00
## Mean    :3.554  Mean    : 5.709  Mean    :10.91  Mean    :10.71
## 3rd Qu.:5.000  3rd Qu.: 8.000  3rd Qu.:13.00  3rd Qu.:13.00
## Max.    :5.000  Max.    :75.000  Max.    :19.00  Max.    :19.00
##       G3
## Min.    : 0.00
## 1st Qu.: 8.00
## Median :11.00
## Mean    :10.42
## 3rd Qu.:14.00
## Max.    :20.00

```

```
summary(langdata)
```

```
## school sex age address famsize Pstatus Medu
## GP:423 F:383 Min. :15.00 R:197 GT3:457 A: 80 Min. :0.000
## MS:226 M:266 1st Qu.:16.00 U:452 LE3:192 T:569 1st Qu.:2.000
## Median :17.00
## Mean :16.74
## 3rd Qu.:18.00
## Max. :22.00
## Fedu Mjob Fjob reason guardian
## Min. :0.000 at_home :135 at_home : 42 course :285 father:153
## 1st Qu.:1.000 health : 48 health : 23 home :149 mother:455
## Median :2.000 other :258 other :367 other : 72 other : 41
## Mean :2.307 services:136 services:181 reputation:143
## 3rd Qu.:3.000 teacher : 72 teacher : 36
## Max. :4.000
## traveltime studytime failures schoolsup famsup paid
## Min. :1.000 Min. :1.000 Min. :0.0000 no :581 no :251 no :610
## 1st Qu.:1.000 1st Qu.:1.000 1st Qu.:0.0000 yes: 68 yes:398 yes: 39
## Median :1.000 Median :2.000 Median :0.0000
## Mean :1.569 Mean :1.931 Mean :0.2219
## 3rd Qu.:2.000 3rd Qu.:2.000 3rd Qu.:0.0000
## Max. :4.000 Max. :4.000 Max. :3.0000
## activities nursery higher internet romantic famrel
## no :334 no :128 no : 69 no :151 no :410 Min. :1.000
## yes:315 yes:521 yes:580 yes:498 yes:239 1st Qu.:4.000
## Median :4.000
## Mean :3.931
## 3rd Qu.:5.000
## Max. :5.000
## freetime goout Dalc Walc health
## Min. :1.00 Min. :1.000 Min. :1.000 Min. :1.00 Min. :1.000
## 1st Qu.:3.00 1st Qu.:2.000 1st Qu.:1.000 1st Qu.:1.00 1st Qu.:2.000
## Median :3.00 Median :3.000 Median :1.000 Median :2.00 Median :4.000
## Mean :3.18 Mean :3.185 Mean :1.502 Mean :2.28 Mean :3.536
## 3rd Qu.:4.00 3rd Qu.:4.000 3rd Qu.:2.000 3rd Qu.:3.00 3rd Qu.:5.000
## Max. :5.00 Max. :5.000 Max. :5.000 Max. :5.00 Max. :5.000
## absences G1 G2 G3
## Min. : 0.000 Min. : 0.0 Min. : 0.000 Min. : 0.00
## 1st Qu.: 0.000 1st Qu.:10.0 1st Qu.:10.00 1st Qu.:10.00
## Median : 2.000 Median :11.0 Median :11.00 Median :12.00
## Mean : 3.659 Mean :11.4 Mean :11.57 Mean :11.91
## 3rd Qu.: 6.000 3rd Qu.:13.0 3rd Qu.:13.00 3rd Qu.:14.00
## Max. :32.000 Max. :19.0 Max. :19.00 Max. :19.00
```

```
describe(mathdata)
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew
##	school*	1	395	1.12	0.32	1	1.02	0.00	1	2	1 2.38
##	sex*	2	395	1.47	0.50	1	1.47	0.00	1	2	1 0.11
##	age	3	395	16.70	1.28	17	16.63	1.48	15	22	7 0.46
##	address*	4	395	1.78	0.42	2	1.85	0.00	1	2	1 -1.33
##	famsize*	5	395	1.29	0.45	1	1.24	0.00	1	2	1 0.93
##	Pstatus*	6	395	1.90	0.31	2	1.99	0.00	1	2	1 -2.59
##	Medu	7	395	2.75	1.09	3	2.82	1.48	0	4	4 -0.32
##	Fedu	8	395	2.52	1.09	2	2.53	1.48	0	4	4 -0.03
##	Mjob*	9	395	3.17	1.23	3	3.21	1.48	1	5	4 -0.33
##	Fjob*	10	395	3.28	0.86	3	3.32	0.00	1	5	4 -0.36
##	reason*	11	395	2.26	1.21	2	2.20	1.48	1	4	3 0.41
##	guardian*	12	395	1.85	0.54	2	1.84	0.00	1	3	2 -0.11
##	traveltime	13	395	1.45	0.70	1	1.31	0.00	1	4	3 1.59
##	studytime	14	395	2.04	0.84	2	1.96	0.00	1	4	3 0.63
##	failures	15	395	0.33	0.74	0	0.14	0.00	0	3	3 2.37
##	schoolsupt	16	395	1.13	0.34	1	1.04	0.00	1	2	1 2.20
##	famsupt	17	395	1.61	0.49	2	1.64	0.00	1	2	1 -0.46
##	paid*	18	395	1.46	0.50	1	1.45	0.00	1	2	1 0.17
##	activities*	19	395	1.51	0.50	2	1.51	0.00	1	2	1 -0.04
##	nursery*	20	395	1.79	0.40	2	1.87	0.00	1	2	1 -1.46
##	higher*	21	395	1.95	0.22	2	2.00	0.00	1	2	1 -4.08
##	internet*	22	395	1.83	0.37	2	1.91	0.00	1	2	1 -1.78
##	romantic*	23	395	1.33	0.47	1	1.29	0.00	1	2	1 0.70
##	famrel	24	395	3.94	0.90	4	4.04	1.48	1	5	4 -0.94
##	freetime	25	395	3.24	1.00	3	3.23	1.48	1	5	4 -0.16
##	goout	26	395	3.11	1.11	3	3.09	1.48	1	5	4 0.12
##	Dalc	27	395	1.48	0.89	1	1.27	0.00	1	5	4 2.17
##	Walc	28	395	2.29	1.29	2	2.15	1.48	1	5	4 0.61
##	health	29	395	3.55	1.39	4	3.69	1.48	1	5	4 -0.49
##	absences	30	395	5.71	8.00	4	4.24	5.93	0	75	75 3.64
##	G1	31	395	10.91	3.32	11	10.80	4.45	3	19	16 0.24
##	G2	32	395	10.71	3.76	11	10.84	2.97	0	19	19 -0.43
##	G3	33	395	10.42	4.58	11	10.84	4.45	0	20	20 -0.73
##			kurtosis	se							
##	school*		3.68	0.02							
##	sex*		-1.99	0.03							
##	age		-0.03	0.06							
##	address*		-0.24	0.02							
##	famsize*		-1.14	0.02							
##	Pstatus*		4.71	0.02							
##	Medu		-1.10	0.06							
##	Fedu		-1.21	0.05							
##	Mjob*		-0.69	0.06							
##	Fjob*		0.98	0.04							
##	reason*		-1.40	0.06							
##	guardian*		0.15	0.03							
##	traveltime		2.27	0.04							
##	studytime		-0.04	0.04							
##	failures		4.89	0.04							
##	schoolsupt		2.86	0.02							
##	famsupt		-1.79	0.02							

```
## 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
```

```
describe(langdata)
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew
##	school*	1	649	1.35	0.48	1	1.31	0.00	1	2	1 0.64
##	sex*	2	649	1.41	0.49	1	1.39	0.00	1	2	1 0.37
##	age	3	649	16.74	1.22	17	16.70	1.48	15	22	7 0.41
##	address*	4	649	1.70	0.46	2	1.74	0.00	1	2	1 -0.85
##	famsize*	5	649	1.30	0.46	1	1.25	0.00	1	2	1 0.89
##	Pstatus*	6	649	1.88	0.33	2	1.97	0.00	1	2	1 -2.29
##	Medu	7	649	2.51	1.13	2	2.53	1.48	0	4	4 -0.03
##	Fedu	8	649	2.31	1.10	2	2.27	1.48	0	4	4 0.21
##	Mjob*	9	649	2.94	1.25	3	2.93	1.48	1	5	4 -0.19
##	Fjob*	10	649	3.22	0.86	3	3.29	0.00	1	5	4 -0.53
##	reason*	11	649	2.11	1.19	2	2.02	1.48	1	4	3 0.56
##	guardian*	12	649	1.83	0.52	2	1.83	0.00	1	3	2 -0.20
##	traveltime	13	649	1.57	0.75	1	1.43	0.00	1	4	3 1.24
##	studytime	14	649	1.93	0.83	2	1.85	1.48	1	4	3 0.70
##	failures	15	649	0.22	0.59	0	0.07	0.00	0	3	3 3.08
##	schoolsupt	16	649	1.10	0.31	1	1.01	0.00	1	2	1 2.57
##	famsupt	17	649	1.61	0.49	2	1.64	0.00	1	2	1 -0.46
##	paid*	18	649	1.06	0.24	1	1.00	0.00	1	2	1 3.69
##	activities*	19	649	1.49	0.50	1	1.48	0.00	1	2	1 0.06
##	nursery*	20	649	1.80	0.40	2	1.88	0.00	1	2	1 -1.52
##	higher*	21	649	1.89	0.31	2	1.99	0.00	1	2	1 -2.55
##	internet*	22	649	1.77	0.42	2	1.83	0.00	1	2	1 -1.26
##	romantic*	23	649	1.37	0.48	1	1.34	0.00	1	2	1 0.55
##	famrel	24	649	3.93	0.96	4	4.05	1.48	1	5	4 -1.10
##	freetime	25	649	3.18	1.05	3	3.19	1.48	1	5	4 -0.18
##	goout	26	649	3.18	1.18	3	3.20	1.48	1	5	4 -0.01
##	Dalc	27	649	1.50	0.92	1	1.28	0.00	1	5	4 2.13
##	Walc	28	649	2.28	1.28	2	2.14	1.48	1	5	4 0.63
##	health	29	649	3.54	1.45	4	3.67	1.48	1	5	4 -0.50
##	absences	30	649	3.66	4.64	2	2.80	2.97	0	32	32 2.01
##	G1	31	649	11.40	2.75	11	11.38	2.97	0	19	19 0.00
##	G2	32	649	11.57	2.91	11	11.56	2.97	0	19	19 -0.36
##	G3	33	649	11.91	3.23	12	12.04	2.97	0	19	19 -0.91
##					kurtosis	se					
##	school*			-1.60	0.02						
##	sex*			-1.87	0.02						
##	age			0.05	0.05						
##	address*			-1.28	0.02						
##	famsize*			-1.21	0.02						
##	Pstatus*			3.23	0.01						
##	Medu			-1.27	0.04						
##	Fedu			-1.12	0.04						
##	Mjob*			-0.83	0.05						
##	Fjob*			1.17	0.03						
##	reason*			-1.25	0.05						
##	guardian*			0.17	0.02						
##	traveltime			1.08	0.03						
##	studytime			0.02	0.03						
##	failures			9.70	0.02						
##	schoolsupt			4.64	0.01						
##	famsupt			-1.79	0.02						

```
## paid*          11.66 0.01
## activities*   -2.00 0.02
## nursery*      0.31 0.02
## higher*       4.50 0.01
## internet*     -0.41 0.02
## romantic*     -1.71 0.02
## famrel         1.32 0.04
## freetime       -0.41 0.04
## goout          -0.87 0.05
## Dalc           4.28 0.04
## Walc           -0.78 0.05
## health          -1.13 0.06
## absences        5.70 0.18
## G1              0.02 0.11
## G2              1.63 0.11
## G3              2.66 0.13
```

```
write.csv(summary(mathdata),"mathsummary.csv")
write.csv(summary(langdata),"langsummary.csv")
```

```
# Calculating frequency of multiple variables
mod_frame <- apply(mathdata, 2 , table)

print ("Math Frequencies")
```

```
## [1] "Math Frequencies"
```

```
print (mod_frame)
```

```

## $school
##
## GP MS
## 349 46
##
## $sex
##
## F M
## 208 187
##
## $age
##
## 15 16 17 18 19 20 21 22
## 82 104 98 82 24 3 1 1
##
## $address
##
## R U
## 88 307
##
## $famsize
##
## GT3 LE3
## 281 114
##
## $Pstatus
##
## A T
## 41 354
##
## $Medu
##
## 0 1 2 3 4
## 3 59 103 99 131
##
## $Fedu
##
## 0 1 2 3 4
## 2 82 115 100 96
##
## $Mjob
##
## at_home health other services teacher
## 59 34 141 103 58
##
## $Fjob
##
## at_home health other services teacher
## 20 18 217 111 29
##
## $reason
##

```

```
##      course      home      other reputation
##      145        109       36        105
##
## $guardian
##
## father mother  other
##    90     273     32
##
## $traveltime
##
##    1     2     3     4
## 257 107   23     8
##
## $studytime
##
##    1     2     3     4
## 105 198   65    27
##
## $failures
##
##    0     1     2     3
## 312  50   17    16
##
## $schoolsup
##
## no yes
## 344 51
##
## $famsup
##
## no yes
## 153 242
##
## $paid
##
## no yes
## 214 181
##
## $activities
##
## no yes
## 194 201
##
## $nursery
##
## no yes
## 81 314
##
## $higher
##
## no yes
## 20 375
```

```

##  

## $internet  

##  

## no yes  

## 66 329  

##  

## $romantic  

##  

## no yes  

## 263 132  

##  

## $famrel  

##  

## 1 2 3 4 5  

## 8 18 68 195 106  

##  

## $freetime  

##  

## 1 2 3 4 5  

## 19 64 157 115 40  

##  

## $goout  

##  

## 1 2 3 4 5  

## 23 103 130 86 53  

##  

## $Dalc  

##  

## 1 2 3 4 5  

## 276 75 26 9 9  

##  

## $Walc  

##  

## 1 2 3 4 5  

## 151 85 80 51 28  

##  

## $health  

##  

## 1 2 3 4 5  

## 47 45 91 66 146  

##  

## $absences  

##  

## 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19  

## 115 3 65 8 53 5 31 7 22 3 17 3 12 3 12 3 7 1 5 1  

## 20 21 22 23 24 25 26 28 30 38 40 54 56 75  

## 4 1 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  

##  

## $G1  

##  

## 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19  

## 1 1 7 24 37 41 31 51 39 35 33 30 24 22 8 8 3

```

```
##  
## $G2  
##  
## 0 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19  
## 13 1 15 14 21 32 50 46 35 41 37 23 34 13 5 12 3  
##  
## $G3  
##  
## 0 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20  
## 38 1 7 15 9 32 28 56 47 31 31 27 33 16 6 12 5 1
```

```
modl_frame <- apply(langdata, 2 , table)  
  
print ("Language Frequencies")
```

```
## [1] "Language Frequencies"
```

```
print (modl_frame)
```

```

## $school
##
## GP MS
## 423 226
##
## $sex
##
## F M
## 383 266
##
## $age
##
## 15 16 17 18 19 20 21 22
## 112 177 179 140 32 6 2 1
##
## $address
##
## R U
## 197 452
##
## $famsize
##
## GT3 LE3
## 457 192
##
## $Pstatus
##
## A T
## 80 569
##
## $Medu
##
## 0 1 2 3 4
## 6 143 186 139 175
##
## $Fedu
##
## 0 1 2 3 4
## 7 174 209 131 128
##
## $Mjob
##
## at_home health other services teacher
## 135 48 258 136 72
##
## $Fjob
##
## at_home health other services teacher
## 42 23 367 181 36
##
## $reason
##

```

```
##      course      home      other reputation
##      285        149       72       143
##
## $guardian
##
## father mother  other
##   153    455     41
##
## $traveltime
##
##   1   2   3   4
## 366 213  54  16
##
## $studytime
##
##   1   2   3   4
## 212 305  97  35
##
## $failures
##
##   0   1   2   3
## 549  70  16  14
##
## $schoolsup
##
## no yes
## 581  68
##
## $famsup
##
## no yes
## 251 398
##
## $paid
##
## no yes
## 610  39
##
## $activities
##
## no yes
## 334 315
##
## $nursery
##
## no yes
## 128 521
##
## $higher
##
## no yes
## 69 580
```

```

##  

## $internet  

##  

## no yes  

## 151 498  

##  

## $romantic  

##  

## no yes  

## 410 239  

##  

## $famrel  

##  

## 1 2 3 4 5  

## 22 29 101 317 180  

##  

## $freetime  

##  

## 1 2 3 4 5  

## 45 107 251 178 68  

##  

## $goout  

##  

## 1 2 3 4 5  

## 48 145 205 141 110  

##  

## $Dalc  

##  

## 1 2 3 4 5  

## 451 121 43 17 17  

##  

## $Walc  

##  

## 1 2 3 4 5  

## 247 150 120 87 45  

##  

## $health  

##  

## 1 2 3 4 5  

## 90 78 124 108 249  

##  

## $absences  

##  

## 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 18 21 22  

## 244 12 110 7 93 12 49 3 42 7 21 5 12 1 8 2 10 3 2 2  

## 24 26 30 32  

## 1 1 1 1  

##  

## $G1  

##  

## 0 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19  

## 1 2 5 9 33 42 65 95 91 82 72 71 35 22 16 7 1

```

```

##  

## $G2  

##  

##  0   5   6   7   8   9   10  11  12  13  14  15  16  17  18  19  

##  7   3   7  16  40  72  83 103  86  80  54  38  25  20  14   1  

##  

## $G3  

##  

##  0   1   5   6   7   8   9   10  11  12  13  14  15  16  17  18  19  

## 15   1   1   3  10  35  35  97 104  72  82  63  49  36  29  15   2

```

#Exploratory Analysis and Visualizations

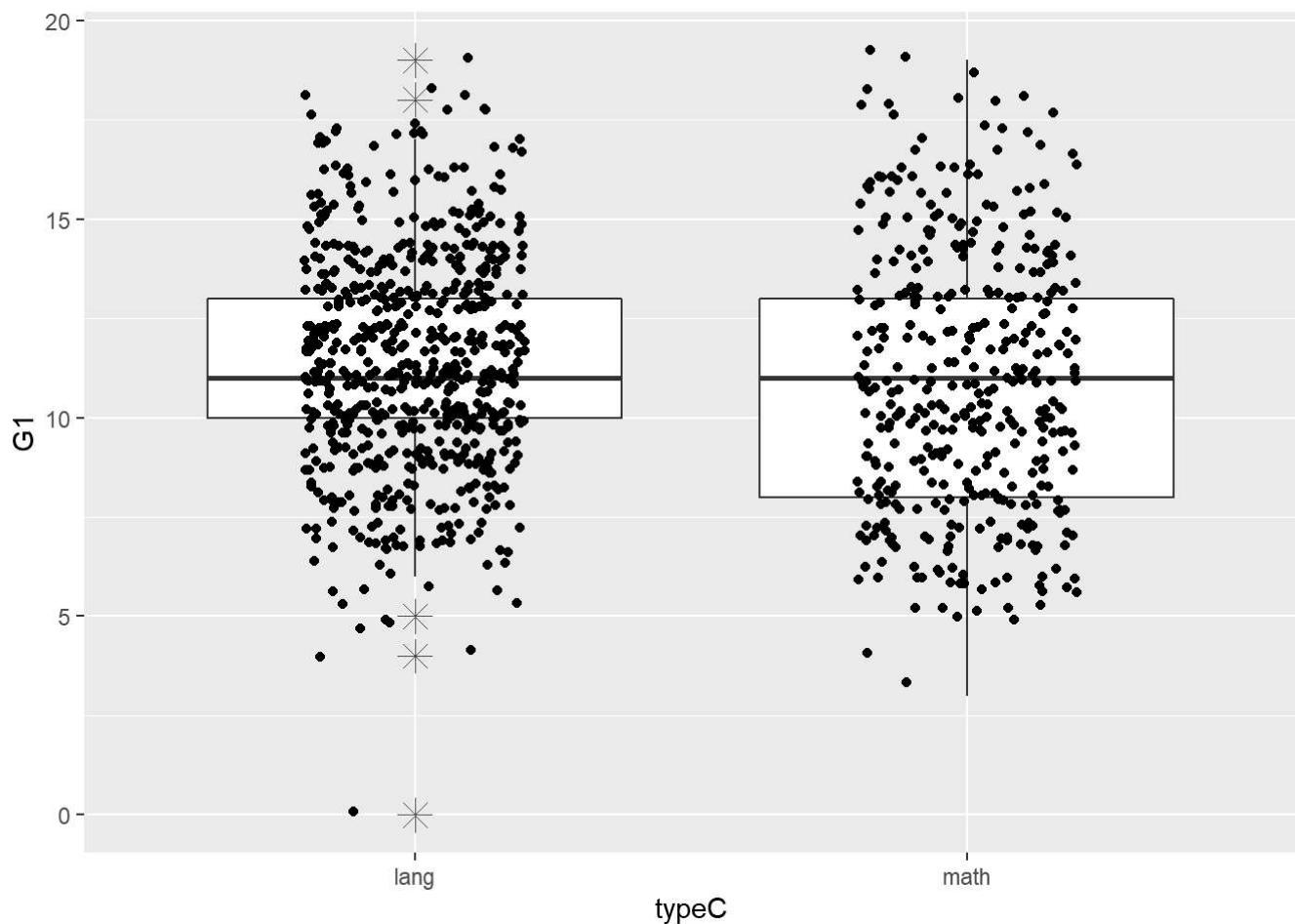
```

#Add type column and append both files
langdata$typeC <- 'lang'
mathdata$typeC <- 'math'
appendedDf <- rbind(mathdata, langdata)

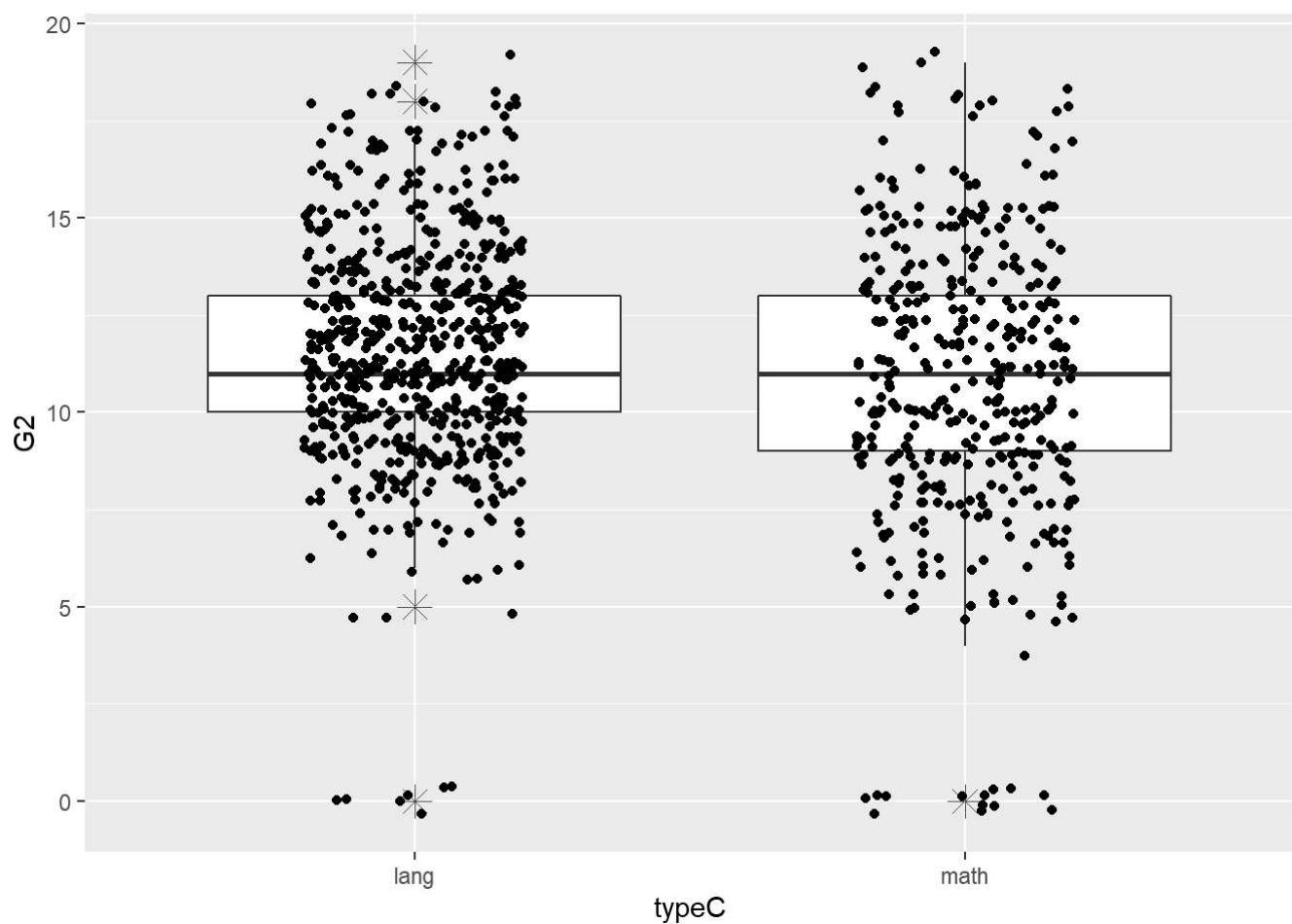
#Take a closer look at the continuous variables
continuousgr <- (appendedDf[,c(31,32,33,34)])  
  

ggplot(continuousgr, aes(x=typeC, y=G1)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
               outlier.size=4) + geom_jitter(shape=16, position=position_jitter(0.2))

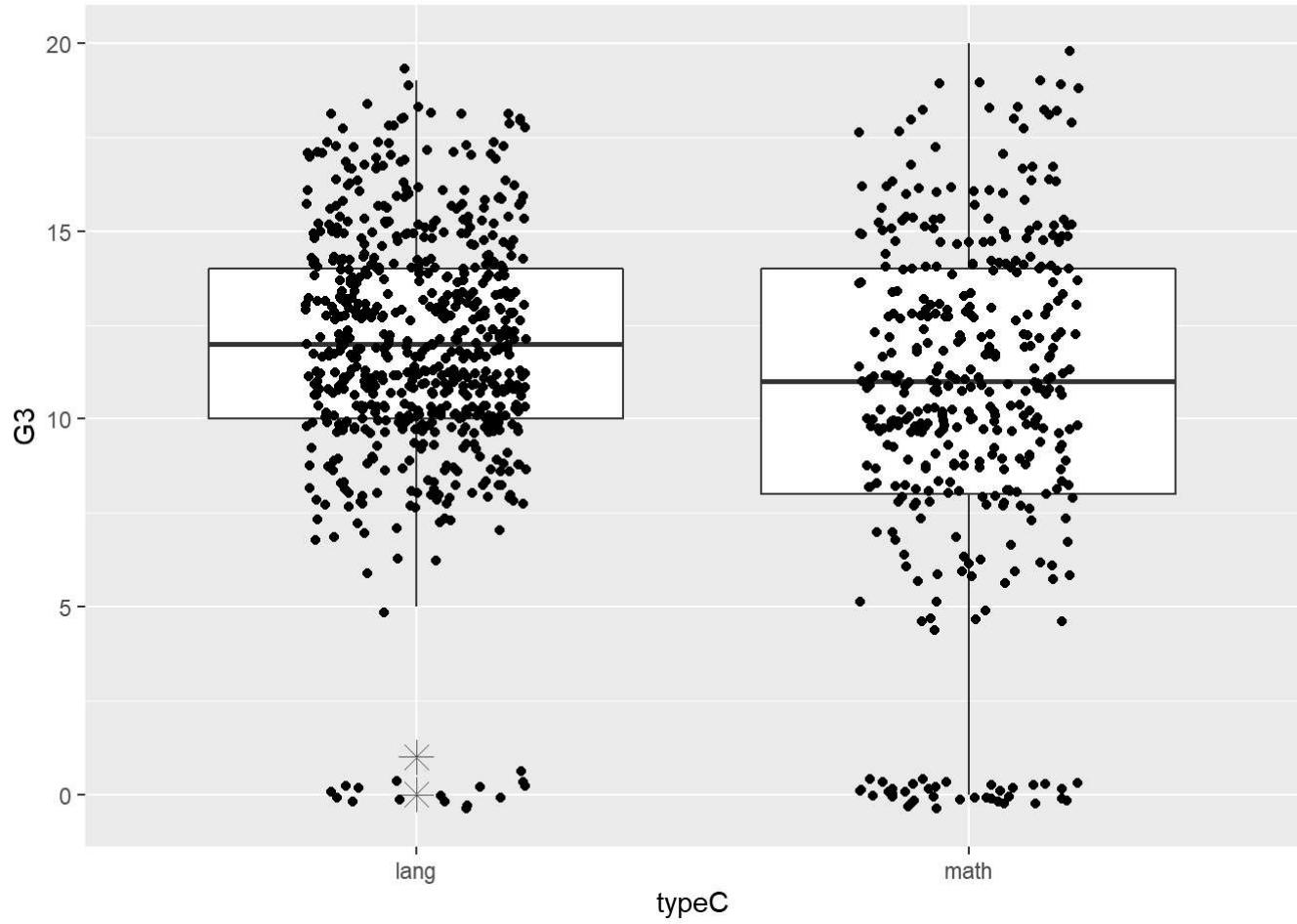
```



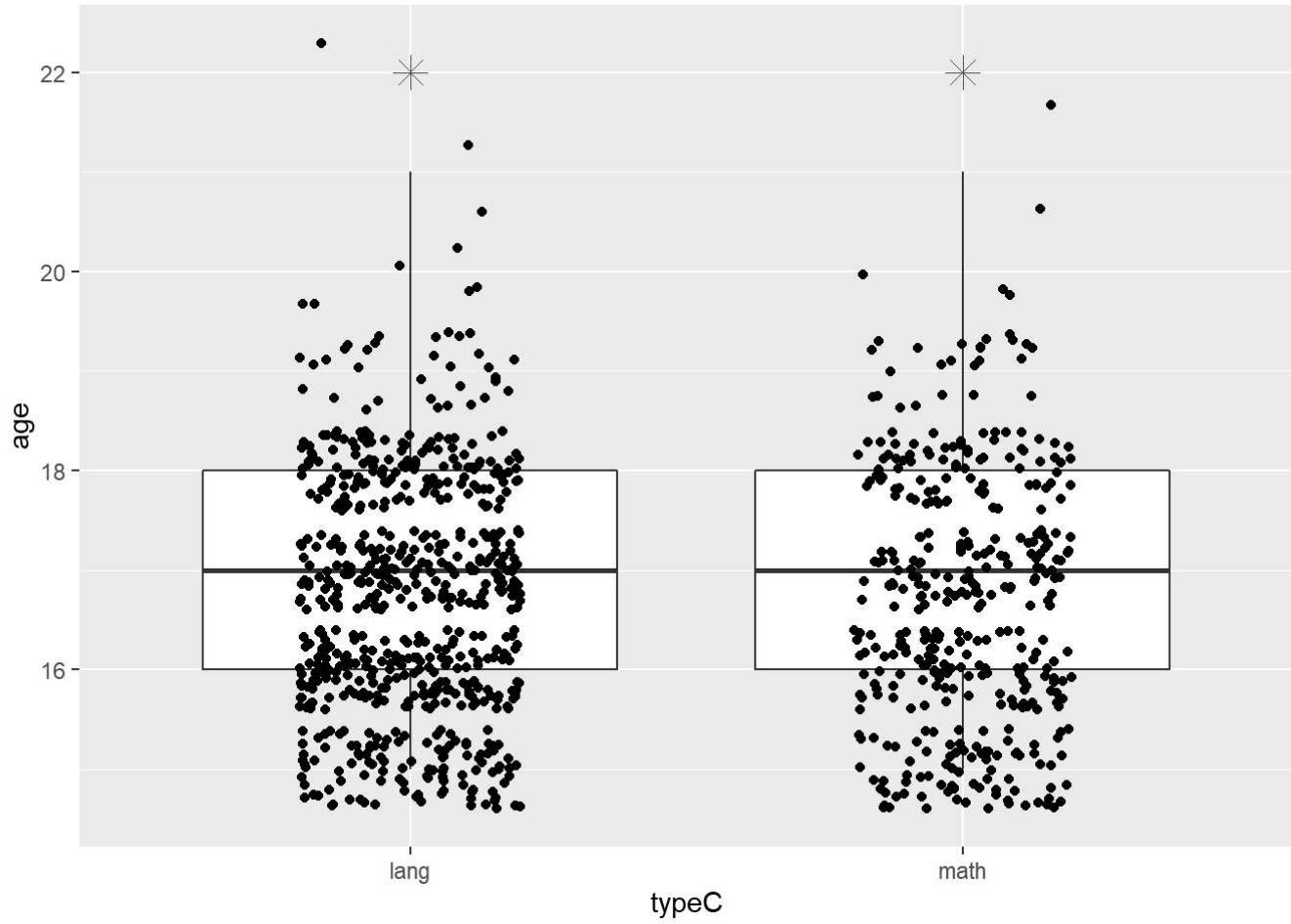
```
ggplot(continuousgr, aes(x=typeC, y=G2)) +  
  geom_boxplot(outlier.colour="red", outlier.shape=8,  
               outlier.size=4) + geom_jitter(shape=16, position=position_jitter(0.2))
```



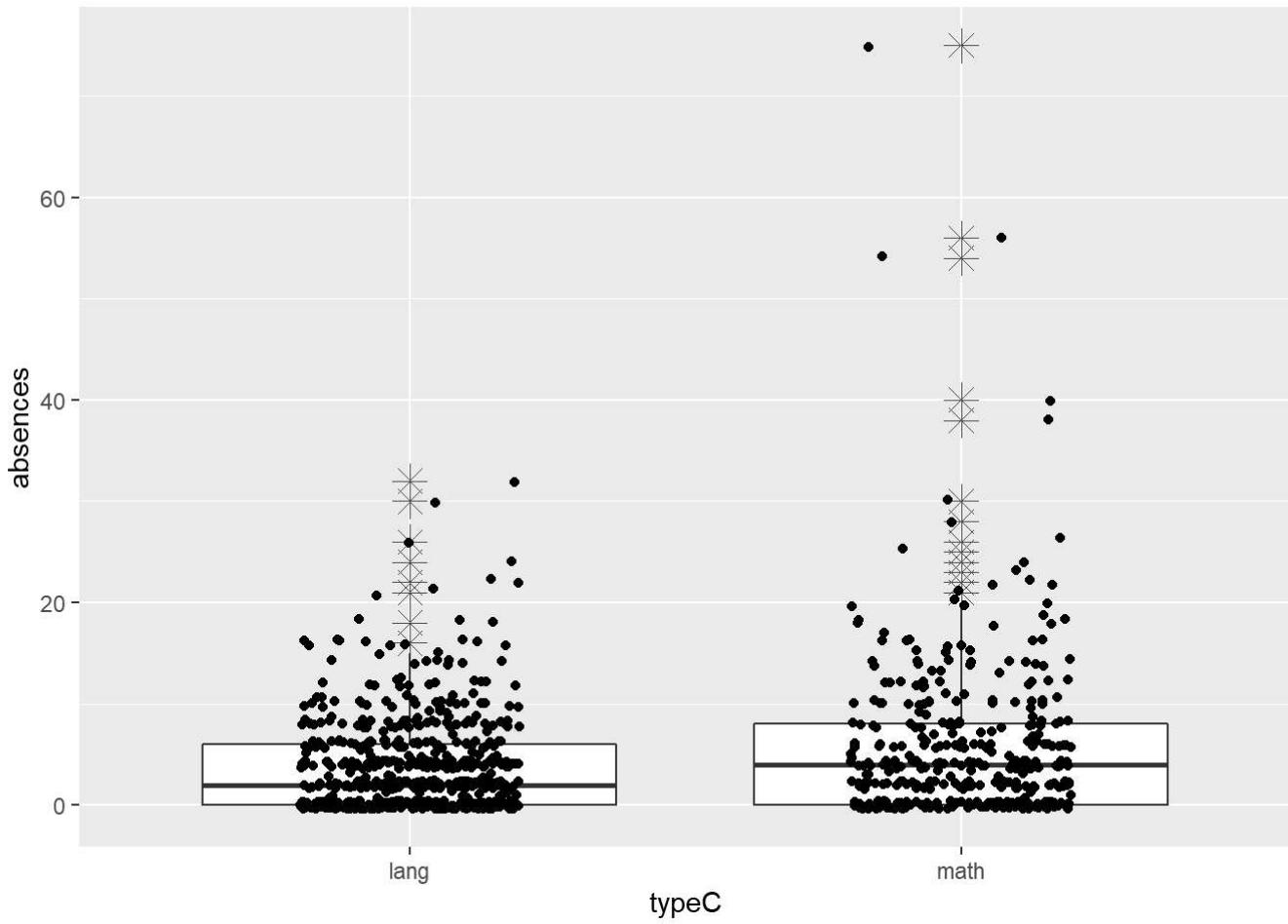
```
ggplot(continuousgr, aes(x=typeC, y=G3)) +  
  geom_boxplot(outlier.colour="red", outlier.shape=8,  
               outlier.size=4) + geom_jitter(shape=16, position=position_jitter(0.2))
```



```
continuousa <- (appendedDf[,c(3,30,34)])
ggplot(continuousa, aes(x=typeC, y=age)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
               outlier.size=4) + geom_jitter(shape=16, position=position_jitter(0.2))
```



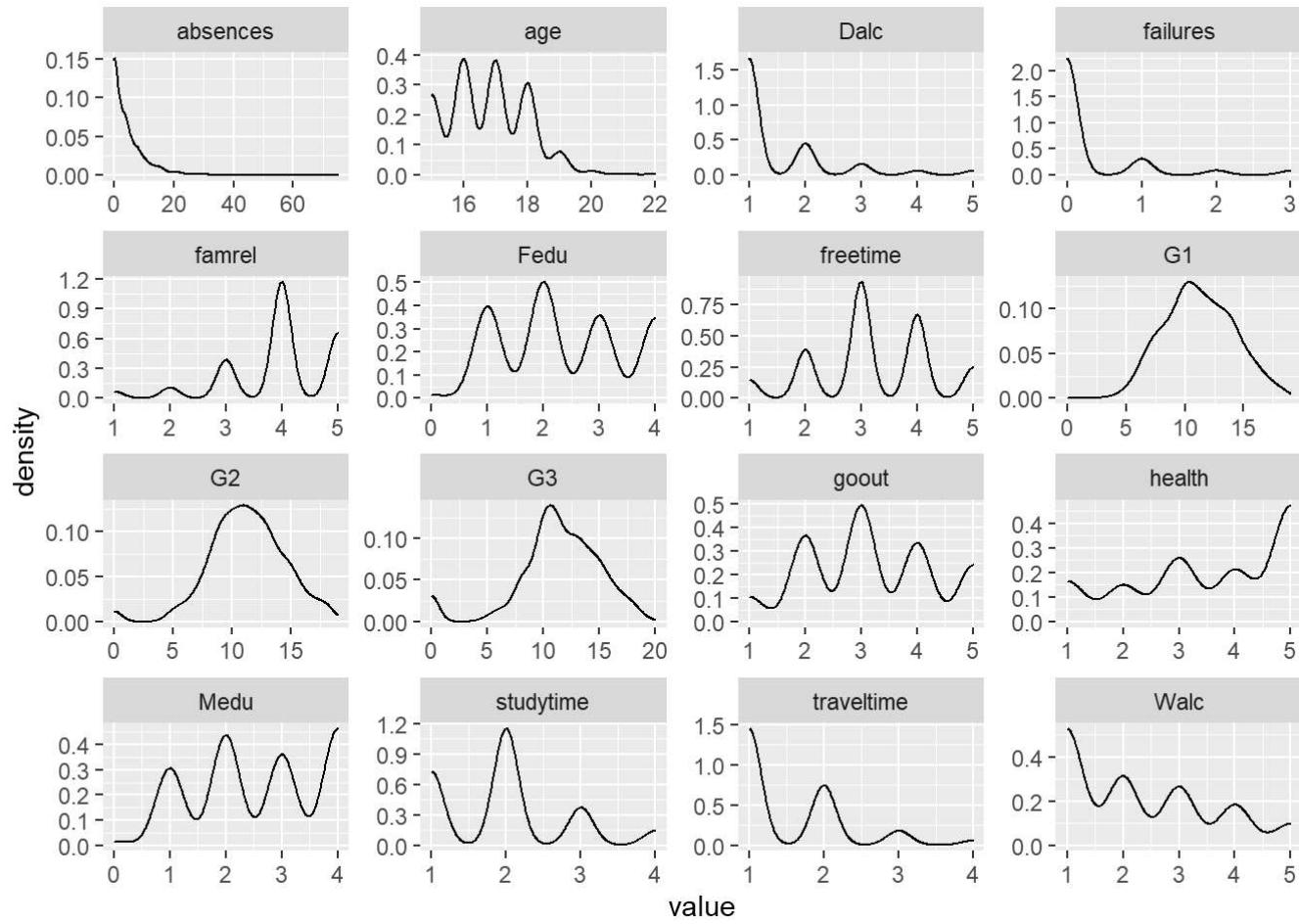
```
ggplot(continuousa, aes(x=typeC, y=absences)) +  
  geom_boxplot(outlier.colour="red", outlier.shape=8,  
               outlier.size=4) + geom_jitter(shape=16, position=position_jitter(0.2))
```



```
#some outliers were identified
```

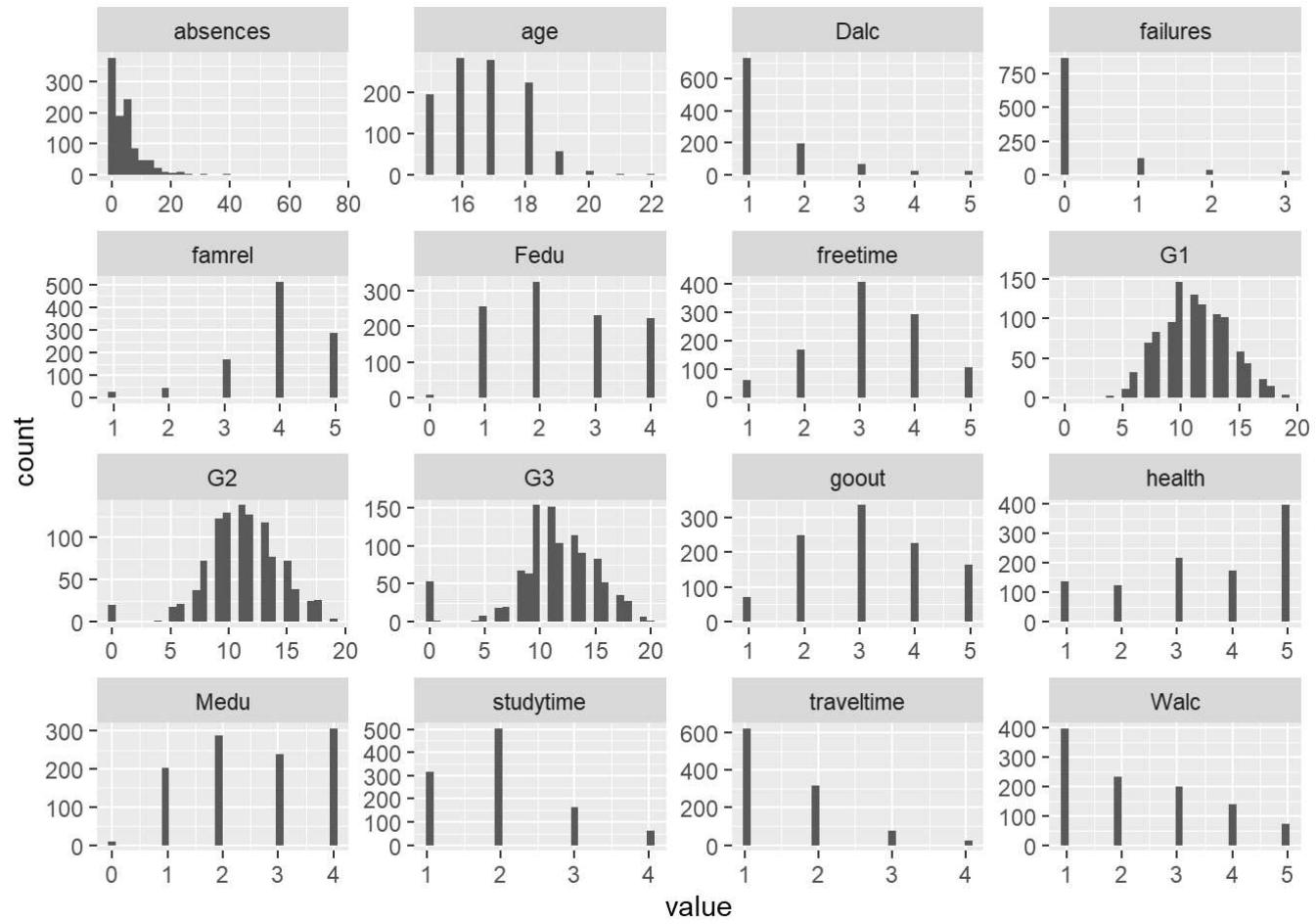
```
#Look at the discrete variables
```

```
appendedDf %>%
  select_if(is.numeric) %>%
  gather() %>%
  ggplot(aes(x = value)) +
  geom_density() +
  facet_wrap(~key, scales = 'free')
```



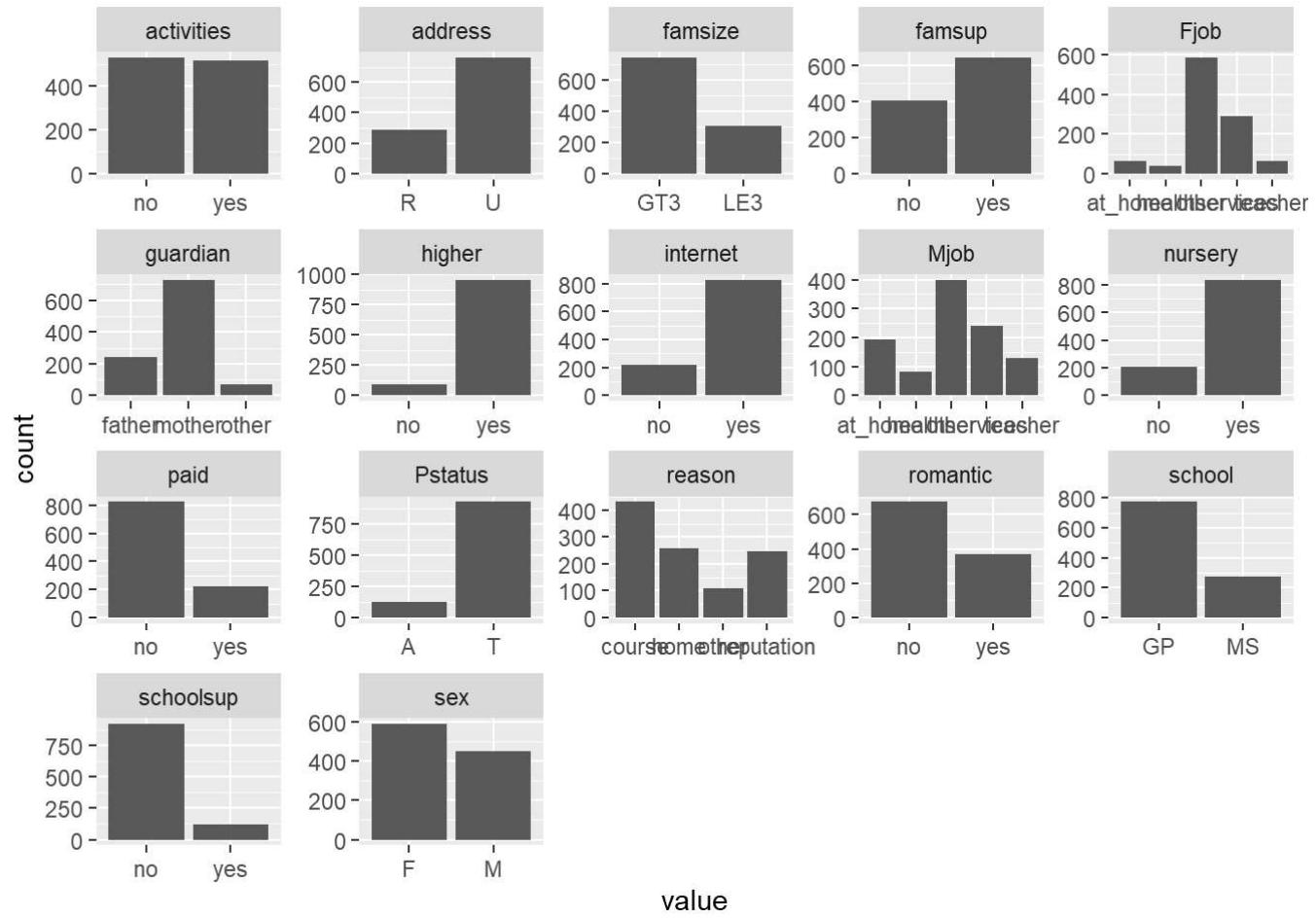
```
appendedDf %>%
  select_if(is.numeric) %>%
  gather() %>%
  ggplot(aes(x = value)) +
  geom_histogram() +
  facet_wrap(~key, scales = 'free')
```

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

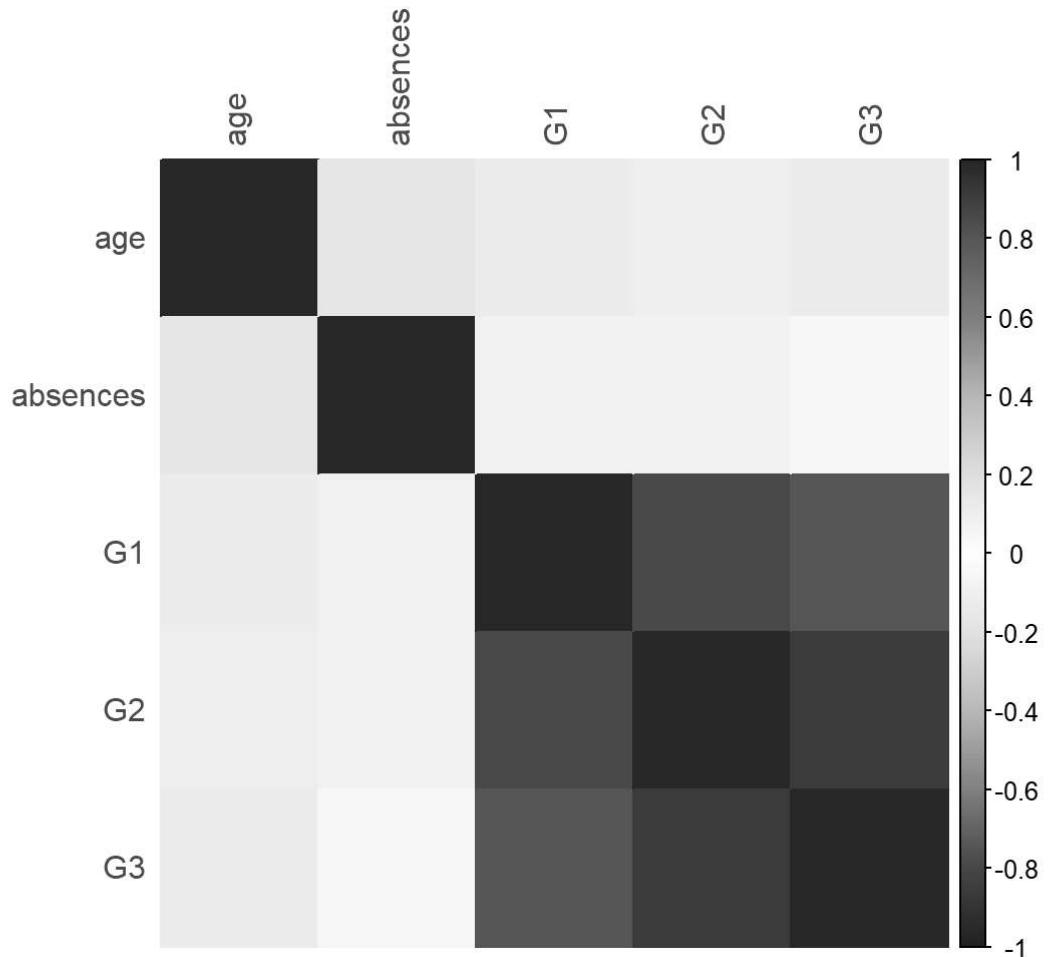


```
appendedDf %>%
  select_if(is.factor) %>%
  gather() %>%
  ggplot(aes(x = value)) +
  geom_bar() +
  facet_wrap(~key, scales = 'free')
```

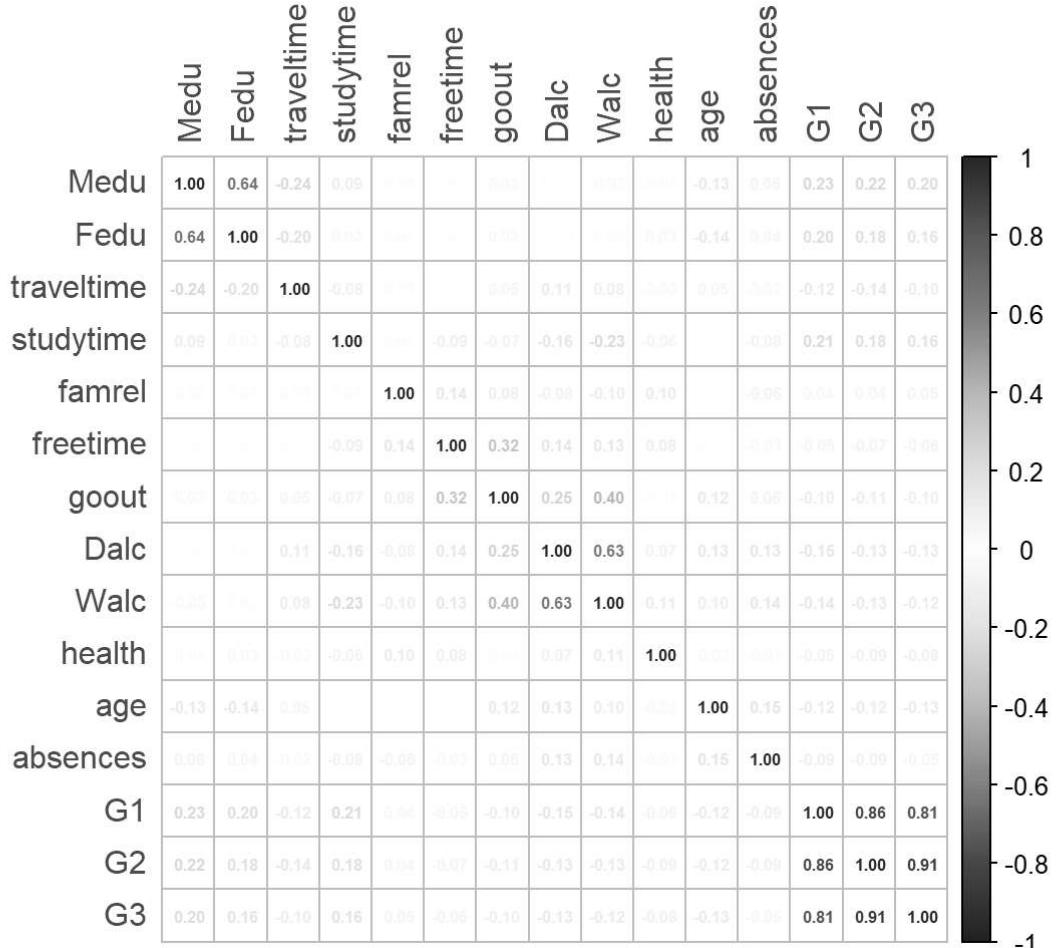
```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```



```
#Look at correlations between variables
corrdf <- (appendedDf[,c(3,30,31,32,33)])
myplot = cor(corrdf)
corrplot(myplot, method = 'color')
```



```
corrdf2 <- (appendedDf[,c(7, 8, 13, 14, 24, 25, 26, 27, 28, 29, 3, 30, 31, 32, 33)])
myplot = cor(corrdf2)
corrplot(myplot, method = 'number', number.cex = 0.5)
```



```
corrplot(myplot, order = 'AOE', addCoef.col = 'black', tl.pos = 'd', cl.pos = 'n', number.cex = 0.5)
```

studytime	0.01	0.21	0.18	0.16	0.09	0.03	-0.09	-0.07	-0.23	-0.16	-0.08	-0.06	-0.01	-0.08	
famrel	0.04	0.04	0.05	0.02	0.01	0.14	0.08	-0.1	-0.08	-0.06	0.1	0.01	-0.01		
G1	0.21	0.04	0.86	0.81	0.23	0.2	-0.05	-0.1	-0.14	-0.15	-0.09	-0.06	-0.12	-0.12	
G2	0.18	0.04	0.86	0.91	0.22	0.18	-0.07	-0.11	-0.13	-0.13	-0.09	-0.09	-0.12	-0.14	
G3	0.16	0.05	0.81	0.91	0.2	0.16	-0.06	-0.1	-0.12	-0.13	-0.05	-0.08	-0.13	-0.1	
Medu	0.09	0.02	0.23	0.22	0.2	Medium	0.64	0	0.03	-0.03	0	0.06	-0.01	-0.13	-0.24
Fedu	0.03	0.01	0.2	0.18	0.16	0.64	0	0.03	0.02	0	0.04	0.03	-0.14	-0.2	
freetime	-0.09	0.14	-0.05	-0.07	-0.06	0	0.32	0.32	0.13	0.14	-0.03	0.08	0	-0.01	
goout	-0.07	0.08	-0.1	-0.11	-0.1	0.03	0.03	0.32	0.4	0.25	0.06	-0.01	0.12	0.05	
Walc	-0.23	-0.1	-0.14	-0.13	-0.12	-0.03	0.02	0.13	0.4	0.63	0.14	0.11	0.1	0.08	
Dalc	-0.16	-0.08	-0.15	-0.13	-0.13	0	0	0.14	0.25	0.63	0.13	0.07	0.13	0.11	
absences	-0.08	-0.06	-0.09	-0.09	-0.05	0.06	0.04	-0.03	0.06	0.14	0.01	0.03	0.15	-0.02	
health	-0.06	0.1	-0.06	-0.09	-0.08	-0.01	0.03	0.08	-0.01	0.11	0.07	-0.03	-0.03	-0.03	
age	-0.01	0.01	-0.12	-0.12	-0.13	-0.13	-0.14	0	0.12	0.1	0.13	0.15	-0.03	0.05	
traveltime	-0.08	-0.01	-0.12	-0.14	-0.1	-0.24	-0.2	-0.01	0.05	0.08	0.11	-0.02	-0.03	0.05	

#The grade variables, G1, G2 and G3 are highly positively correlated
#absences and age show some correlation
#Mother's and Father's education show a positive correlation to each other as well

```
#change binary values to numeric
appendedDf$schoolsup<-ifelse(appendedDf$schoolsup=="yes",1,0)
appendedDf$sex<-ifelse(appendedDf$sex=="F",1,0)
appendedDf$address<-ifelse(appendedDf$address=="U",1,0)
appendedDf$famsize<-ifelse(appendedDf$famsize=="GT3",1,0)
appendedDf$school<-ifelse(appendedDf$school=="GP",1,0)
appendedDf$famsup<-ifelse(appendedDf$famsup=="yes",1,0)
appendedDf$paid<-ifelse(appendedDf$paid=="yes",1,0)
appendedDf$activities<-ifelse(appendedDf$activities=="yes",1,0)
appendedDf$nursery<-ifelse(appendedDf$nursery=="yes",1,0)
appendedDf$higher<-ifelse(appendedDf$higher=="yes",1,0)
appendedDf$internet<-ifelse(appendedDf$internet=="yes",1,0)
appendedDf$romantic<-ifelse(appendedDf$romantic=="yes",1,0)
appendedDf$Pstatus<-ifelse(appendedDf$Pstatus=="T",1,0)
appendedDf$typeC<-ifelse(appendedDf$typeC=="math",1,0)
str(appendedDf)
```

```

## 'data.frame': 1044 obs. of 34 variables:
## $ school : num 1 1 1 1 1 1 1 1 1 1 ...
## $ sex    : num 1 1 1 1 1 0 0 1 0 0 ...
## $ age    : int 18 17 15 15 16 16 16 17 15 15 ...
## $ address : num 1 1 1 1 1 1 1 1 1 1 ...
## $ famsize : num 1 1 0 1 1 0 0 1 0 1 ...
## $ Pstatus : num 0 1 1 1 1 1 1 0 0 1 ...
## $ 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   : Factor w/ 5 levels "at_home","health",...: 1 1 1 2 3 4 3 3 4 3 ...
## $ Fjob   : Factor w/ 5 levels "at_home","health",...: 5 3 3 4 3 3 3 5 3 3 ...
## $ reason  : Factor w/ 4 levels "course","home",...: 1 1 3 2 2 4 2 2 2 2 ...
## $ guardian: Factor w/ 3 levels "father","mother",...: 2 1 2 2 1 2 2 2 2 2 ...
## $ traveltime: 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: num 1 0 1 0 0 0 0 1 0 0 ...
## $ famsup  : num 0 1 0 1 1 1 0 1 1 1 ...
## $ paid    : num 0 0 1 1 1 1 0 0 1 1 ...
## $ activities: num 0 0 0 1 0 1 0 0 0 1 ...
## $ nursery : num 1 0 1 1 1 1 1 1 1 1 ...
## $ higher  : num 1 1 1 1 1 1 1 1 1 1 ...
## $ internet: num 0 1 1 1 0 1 1 0 1 1 ...
## $ romantic: num 0 0 0 1 0 0 0 0 0 0 ...
## $ 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 ...
## $ typeC  : num 1 1 1 1 1 1 1 1 1 1 ...

```

```

allnums <- (appendedDf[,c(1,2,3,4,5,6,7,8,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,
31,32,33,34)])
head(allnums)

```

```

## school sex age address famsize Pstatus Medu Fedu traveltime studytime
## 1     1   1 18      1     1     0    4    4          2       2
## 2     1   1 17      1     1     1    1    1          1       2
## 3     1   1 15      1     0     1    1    1          1       2
## 4     1   1 15      1     1     1    4    2          1       3
## 5     1   1 16      1     1     1    3    3          1       2
## 6     1   0 16      1     0     1    4    3          1       2
## failures schoolsup famsup paid activities nursery higher internet romantic
## 1     0         1     0     0        0     1     1     0     0
## 2     0         0     1     0        0     0     1     1     0
## 3     3         1     0     1        0     1     1     1     0
## 4     0         0     1     1        1     1     1     1     1
## 5     0         0     1     1        0     1     1     0     0
## 6     0         0     1     1        1     1     1     1     0
## famrel freetime goout Dalc Walc health absences G1 G2 G3 typeC
## 1     4         3     4     1     1     3       6    5    6    6    1
## 2     5         3     3     1     1     3       4    5    5    6    1
## 3     4         3     2     2     3     3      10    7    8   10    1
## 4     3         2     2     1     1     5       2   15   14   15    1
## 5     4         3     2     1     2     5       4   6   10   10    1
## 6     5         4     2     1     2     5      10   15   15   15    1

```

```

#normalize numeric attributes
min_max_norm <- function(x){
  (x - min(x)) / (max(x) - min(x))
}

datanorm <- allnums %>%
  mutate(across(c(3,7,8,9,10,11,20,21,22,23,24,25,26,27,28),min_max_norm))

```

```

#Dimensionality reduction
#PCA
dataMatrix <- data.matrix(datanorm[,c(1:28,30)])
data_PCA <- princomp(dataMatrix )
summary(data_PCA, loading = T)

```

```

## Importance of components:
##                               Comp.1      Comp.2      Comp.3      Comp.4      Comp.5
## Standard deviation      0.6357896 0.57702940 0.51305614 0.50349052 0.48864428
## Proportion of Variance 0.1159827 0.09553492 0.07552593 0.07273591 0.06850968
## Cumulative Proportion  0.1159827 0.21151761 0.28704354 0.35977945 0.42828913
##                               Comp.6      Comp.7      Comp.8      Comp.9      Comp.10
## Standard deviation     0.46745478 0.45973252 0.39706605 0.38898417 0.36755530
## Proportion of Variance 0.06269682 0.06064245 0.04523679 0.04341403 0.03876249
## Cumulative Proportion  0.49098595 0.55162840 0.59686519 0.64027923 0.67904172
##                               Comp.11     Comp.12     Comp.13     Comp.14     Comp.15
## Standard deviation     0.35232605 0.34016364 0.33324504 0.31022173 0.30692421
## Proportion of Variance 0.03561687 0.03320031 0.03186352 0.02761282 0.02702891
## Cumulative Proportion  0.71465859 0.74785890 0.77972241 0.80733523 0.83436414
##                               Comp.16     Comp.17     Comp.18     Comp.19     Comp.20
## Standard deviation     0.29392149 0.27496059 0.26334861 0.24498790 0.21890642
## Proportion of Variance 0.02478729 0.02169238 0.01989887 0.01722089 0.01374939
## Cumulative Proportion  0.85915143 0.88084381 0.90074268 0.91796357 0.93171295
##                               Comp.21     Comp.22     Comp.23     Comp.24     Comp.25
## Standard deviation     0.21185506 0.20681283 0.20061719 0.172991986 0.163746861
## Proportion of Variance 0.01287787 0.01227217 0.01154789 0.008586541 0.007693293
## Cumulative Proportion  0.94459082 0.95686299 0.96841087 0.976997416 0.984690709
##                               Comp.26     Comp.27     Comp.28     Comp.29
## Standard deviation     0.153228376 0.143044479 0.076275739 0.05998360
## Proportion of Variance 0.006736661 0.005870951 0.001669318 0.00103236
## Cumulative Proportion  0.991427370 0.997298321 0.998967640 1.00000000
## 

## Loadings:
##          Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9
## school      0.473           0.234           0.154   0.150           0.325
## sex        -0.111  -0.723           0.154   0.225  -0.178   0.216  -0.178
## age         0.337           0.506   0.216           0.211   0.133   0.106
## address     0.337           0.506   0.216           0.211   0.133   0.106
## famsize     -0.197   0.120  -0.324  -0.269   0.376   0.645  -0.342
## Pstatus      -0.156  -0.127           0.136           -0.146
## Medu        0.207           0.105           -0.227  -0.109
## Fedu        0.173           -0.193
## traveltime   -0.135          -0.125
## studytime    -0.169
## failures
## schoolsup    -0.104           0.149
## famsup       0.216  -0.399   0.191          -0.417  -0.652           0.178   0.177
## paid         0.352           -0.195  -0.358          -0.120  -0.137           -0.173
## activities   0.186   0.252   0.754  -0.223          0.270  -0.336   0.195
## nursery      -0.130          -0.119  -0.360  -0.766
## higher       0.121
## internet     0.272           0.155   0.144   0.109  -0.136   0.217           -0.651
## romantic    -0.105  -0.123   0.333  -0.221   0.752  -0.273   0.244  -0.117   0.201
## famrel
## freetime     0.117
## goout        0.108           -0.122           -0.188
## Dalc         0.155           -0.163
## Walc         0.274           -0.220   0.129   0.114  -0.132

```

```

## health           0.113                         0.429
## absences
## G1
## G2                  0.109
## typeC      0.479     -0.391 -0.461  0.214  0.122 -0.153
##          Comp.10 Comp.11 Comp.12 Comp.13 Comp.14 Comp.15 Comp.16 Comp.17
## school      0.179     0.426  0.346  0.149  0.330  0.163  0.174
## sex         -0.260    0.299 -0.172  0.123                0.138        -0.149
## age                    0.102
## address     -0.475   -0.416 -0.135 -0.165
## famsize     -0.139                               0.139
## Pstatus
## Medu        0.125     -0.117  0.348 -0.187                -0.103        -0.149
## Fedu
## traveltim  0.105
## studytim   0.114                               0.169        0.379
## failures   -0.129     0.115 -0.130 -0.121                0.132        -0.108
## schoolsup   0.281     0.193                -0.244 -0.790        -0.286
## famsup      -0.140    0.128 -0.125                0.111
## paid         -0.176   -0.223                0.401 -0.381        0.391        0.129
## activities  -0.140
## nursery     -0.331    0.178  0.175 -0.201                -0.340        0.240
## higher      0.153     -0.200  0.218  0.101                -0.340        0.240
## internet    0.233     0.302                -0.369 -0.183                0.145
## romantic    0.123
## famrel
## freetime    -0.169
## goout       -0.409    0.197                0.201                0.159        -0.107        0.403
## Dalc        -0.159    0.110                0.184
## Walc        -0.328    0.233                0.343  0.196                -0.210
## health       0.388   -0.749 -0.167
## absences
## G1          0.122
## G2          0.122             0.113
## typeC
##          Comp.18 Comp.19 Comp.20 Comp.21 Comp.22 Comp.23 Comp.24 Comp.25
## school      0.132
## sex
## age         -0.179
## address     0.120
## famsize    -0.169
## Pstatus     0.308
## Medu        0.254   -0.157                0.118        -0.648
## Fedu        0.291   -0.102    0.137
## traveltim -0.172    0.123   0.582   0.642  -0.303    0.205
## studytim  -0.287   -0.785
## failures   -0.133
## schoolsup
## famsup
## paid        0.219
## activities
## nursery    -0.112

```

```

## higher      -0.433   0.451                  0.435   0.246
## internet    -0.162
## romantic
## famrel       0.228   0.154   0.487   -0.603   -0.178   0.299
## freetime     0.301          -0.533   0.284   -0.222   0.354
## goout        0.114          0.231          0.284   -0.528
## Dalc         -0.151   -0.101          -0.112   -0.197   0.311      -0.318
## Walc         -0.269          -0.218   -0.158   0.111          0.172
## health
## absences
## G1                   -0.263   -0.114   -0.493
## G2                   -0.303   -0.127   -0.578
## typeC      -0.159          -0.107
## Comp.26  Comp.27  Comp.28  Comp.29
## school
## sex
## age        0.501   0.664
## address
## famsize
## Pstatus
## Medu        -0.251   0.161
## Fedu        0.206   -0.124
## traveltim
## studytim
## failures    -0.322   -0.326
## schoolsup
## famsup
## paid
## activties
## nursery
## higher
## internet
## romantic
## famrel
## freetime      0.131
## goout        -0.146
## Dalc         0.552   -0.490
## Walc         -0.420   0.289
## health
## absences      -0.992
## G1           -0.109          -0.750
## G2           -0.109   -0.110          0.658
## typeC

```

```

score <- data_PCA$scores
head(score)

```

```

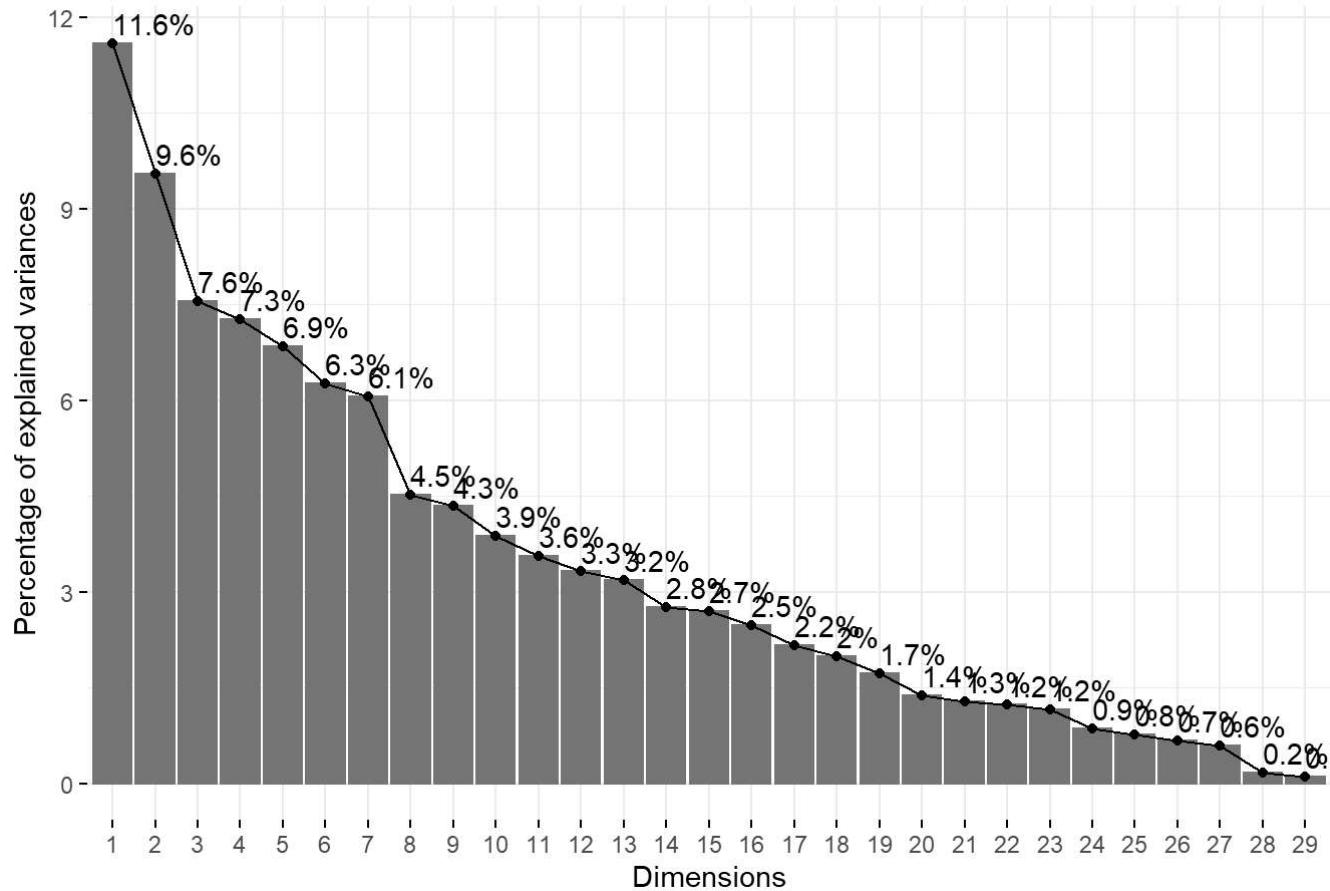
##          Comp.1      Comp.2      Comp.3      Comp.4      Comp.5      Comp.6
## [1,] 0.1515054 -0.42893630 -0.86881252  0.22253378  0.1277206  0.79282786
## [2,] 0.3046240 -0.63415222 -0.64655447 -0.10651901 -0.1798919  0.31372047
## [3,] 0.5053699 -0.09127951 -1.18218724  0.01433697  0.5047999  0.28988917
## [4,] 1.0784900 -0.72547551  0.44613394 -0.58030810  0.5112759  0.05716699
## [5,] 0.6594463 -0.71035163 -0.85236982 -0.38754134 -0.3005554  0.07830783
## [6,] 1.2942678  0.51509143 -0.01910466 -0.06012887 -0.1628791 -0.39522528
##          Comp.7      Comp.8      Comp.9      Comp.10     Comp.11     Comp.12
## [1,] -0.2493474 -0.315565487  0.60308622 -0.36238805  0.18365708  0.3218368
## [2,] 0.5327917  0.841900948 -0.01198026  0.12618812 -0.03023202  0.1115188
## [3,] -0.5814766  0.361614179 -0.14506020 -0.37499463  0.58984346  0.2020173
## [4,] -0.1024558 -0.212525634  0.21270159  0.09564712 -0.17417632 -0.4523031
## [5,] -0.1237801 -0.233653456  0.62890691 -0.37202669 -0.23050015 -0.3911324
## [6,] -0.7198779  0.007348327  0.10984067  0.20791205 -0.25198423 -0.4400668
##          Comp.13     Comp.14     Comp.15     Comp.16     Comp.17     Comp.18
## [1,] 0.8109409 -0.9700917 -0.344777183 -0.07830307  0.17976451  0.10019092
## [2,] -0.4707217 -0.2182965  0.365347413 -0.08163834  0.15307318 -0.09279067
## [3,] -0.4190188  0.1754499 -0.926633870  0.02173663 -0.13458111  0.11587085
## [4,] -0.1633238  0.2743132 -0.052148170  0.15380196 -0.19921495 -0.18166765
## [5,] 0.1641277  0.2983074  0.117507891  0.05713624 -0.14312025  0.24781614
## [6,] -0.3134001  0.1303550 -0.001214135  0.05616535  0.05029108  0.26397352
##          Comp.19     Comp.20     Comp.21     Comp.22     Comp.23     Comp.24
## [1,] -0.09330423  0.38239680  0.14536747  0.07591153  0.000933532  0.13149269
## [2,] 0.17899469  0.08713509 -0.26417304  0.13036238 -0.003550461  0.37130927
## [3,] -0.03852080 -0.14863968 -0.08252605  0.39158903  0.533315745 -0.18452994
## [4,] -0.11302976 -0.10272564  0.10987585 -0.07391771 -0.185166712 -0.06255147
## [5,] 0.06288130 -0.07669153 -0.03884803 -0.02098507  0.104390782  0.24010403
## [6,] 0.03366414 -0.09345173 -0.07196474 -0.16630561  0.098305958 -0.11462794
##          Comp.25     Comp.26     Comp.27     Comp.28     Comp.29
## [1,] -0.07547432  0.25365485  0.25106201  0.03427193  0.06262378
## [2,] -0.06292082  0.19796233  0.06269895  0.02298925  0.03771086
## [3,] 0.13886383 -0.36539676 -0.37171150 -0.07411728  0.01877965
## [4,] -0.27310959 -0.17933860 -0.17068202  0.03009817 -0.04059137
## [5,] -0.00339791 -0.02322869  0.04878757 -0.00215762  0.16728716
## [6,] -0.04864251 -0.13699433  0.05502544 -0.08849720 -0.02340240

```

```

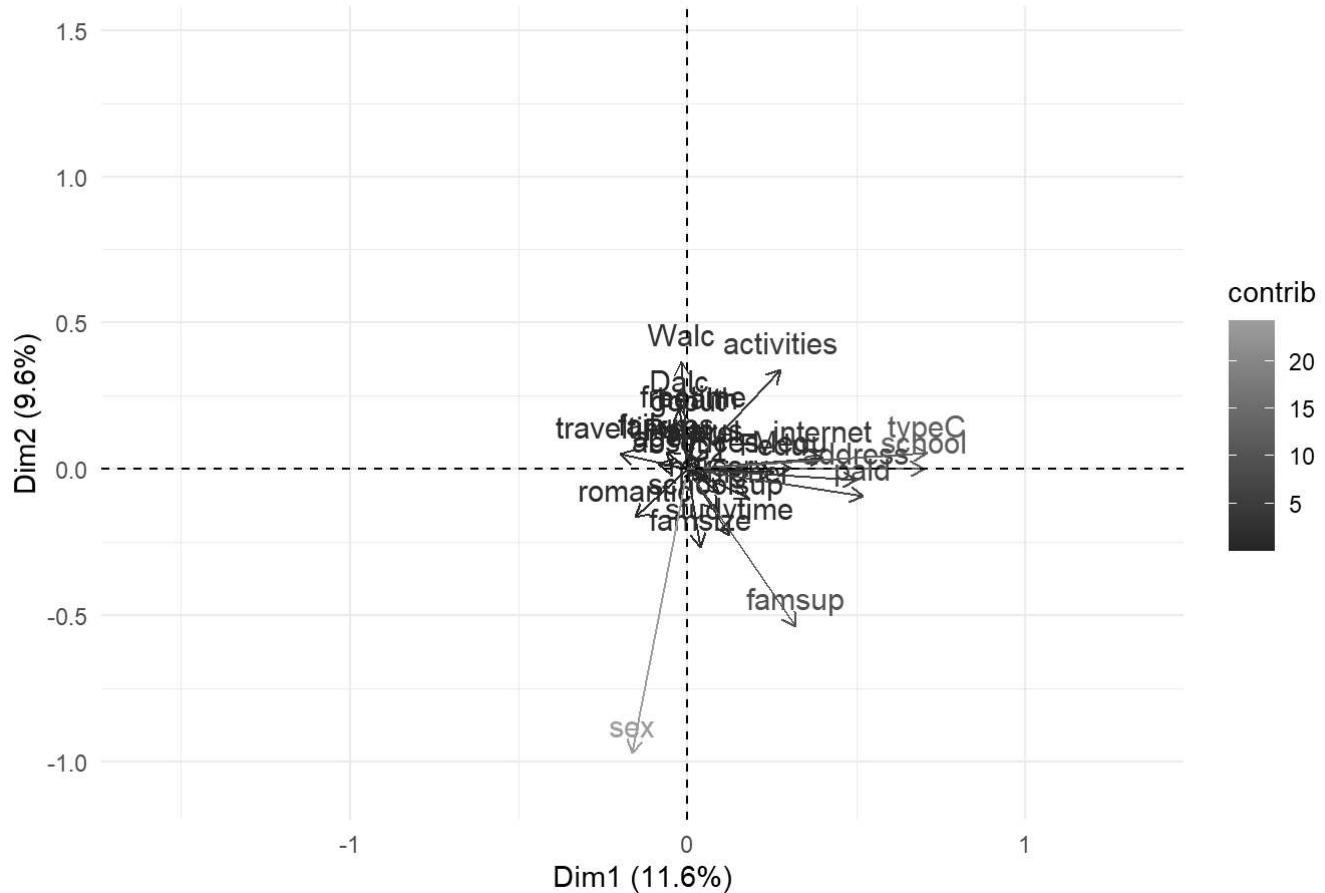
# Plot the dimensions
fviz_screeplot(data_PCA, main=" ", ncp=50, addlabels = TRUE)

```



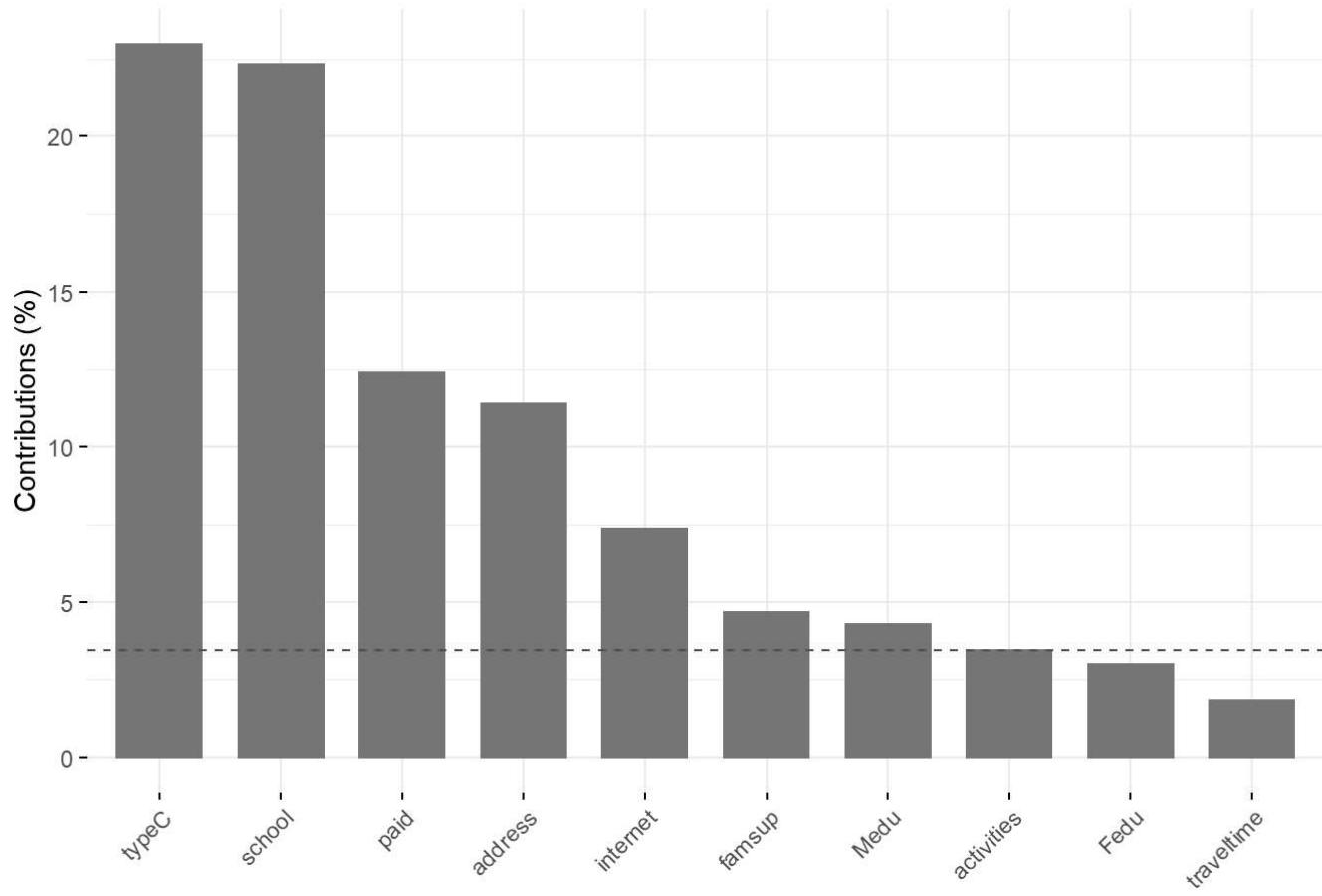
```
# Plot the Principal Components:
fviz_pca_biplot(data_PCA,col.var="contrib", invisible = "ind", habillage ="none", geom = "text",
labelsize=4) + theme_minimal()
```

PCA - Biplot



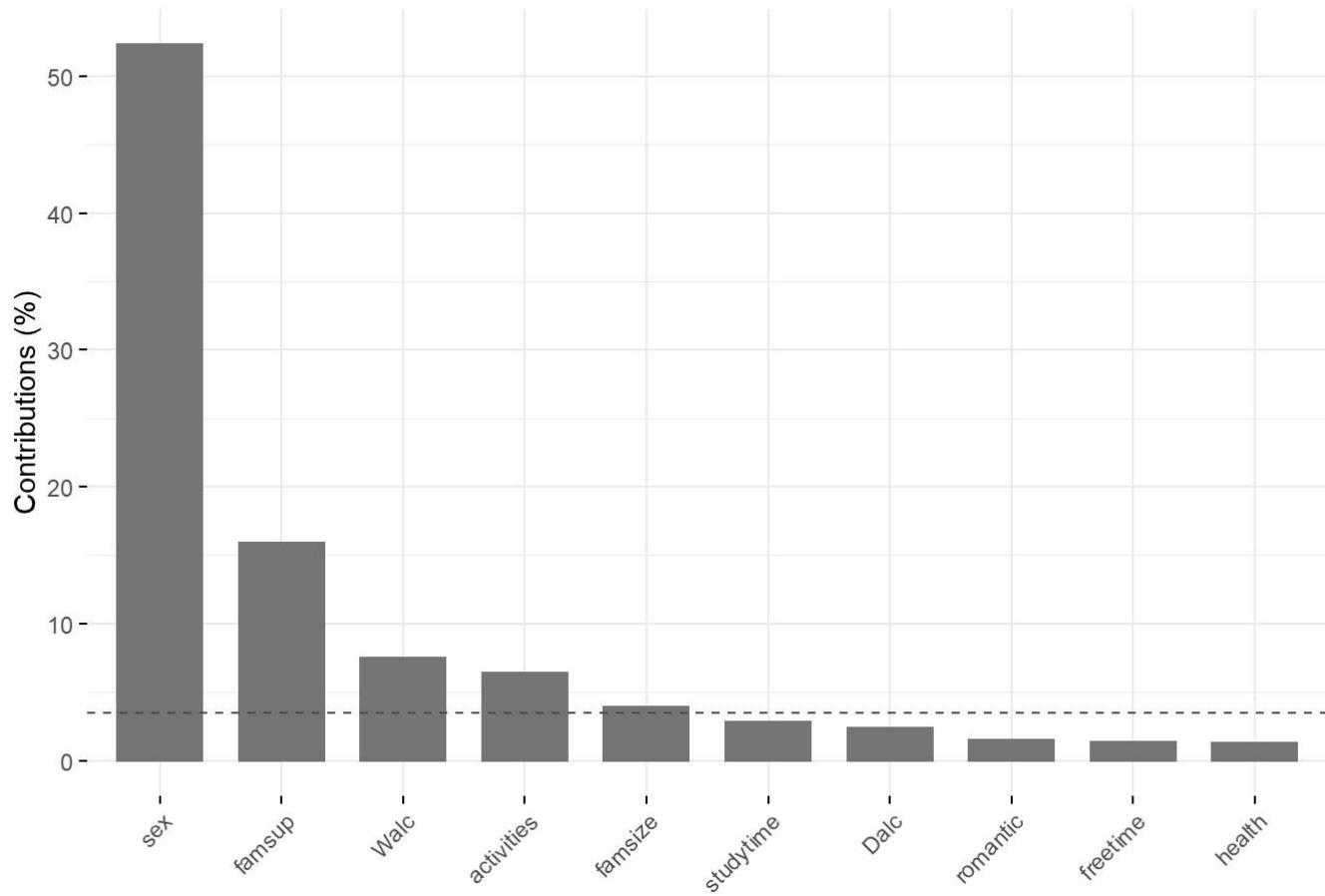
```
# show the contributions of the variables for PC1  
fviz_contrib(data_PCA, choice = "var", axes = 1, top = 10)
```

Contribution of variables to Dim-1



```
# show the contributions of the variables for PC2  
fviz_contrib(data_PCA, choice = "var", axes = 2, top = 10)
```

Contribution of variables to Dim-2



```
#Random Forest dimension reduction
```

```
dataR<-(datanorm[,c(1:30)])
```

```
#add the target column called FinalGradeR
```

```
#grades less than 10 are a failure and assigned a zero and #grades more than or equal to 10 will  
be considered pass and #assigned 1
```

```
dataR <- dataR %>%  
  mutate(FinalGradeR = case_when(  
    G3 < 10 ~ 0,  
    G3 > 9 ~ 1  
)
```

```
head(dataR)
```

```

##   school sex      age address famsize Pstatus Medu Fedu traveltime studytime
## 1      1  1 0.4285714      1      1       0 1.00 1.00  0.3333333 0.3333333
## 2      1  1 0.2857143      1      1       1 0.25 0.25  0.0000000 0.3333333
## 3      1  1 0.0000000      1      0       1 0.25 0.25  0.0000000 0.3333333
## 4      1  1 0.0000000      1      1       1 1.00 0.50  0.0000000 0.6666667
## 5      1  1 0.1428571      1      1       1 0.75 0.75  0.0000000 0.3333333
## 6      1  0 0.1428571      1      0       1 1.00 0.75  0.0000000 0.3333333
##   failures schoolsup famsup paid activities nursery higher internet romantic
## 1      0        1     0     0       0      1     1      0      0
## 2      0        0     1     0       0      0     1      1      0
## 3      1        1     0     1       0      1     1      1      0
## 4      0        0     1     1       1      1     1      1      1
## 5      0        0     1     1       0      1     1      0      0
## 6      0        0     1     1       1      1     1      1      0
##   famrel freetime goout Dalc Walc health absences      G1      G2 G3
## 1    0.75    0.50  0.75  0.00 0.00    0.5 0.08000000 0.2631579 0.3157895 6
## 2    1.00    0.50  0.50  0.00 0.00    0.5 0.05333333 0.2631579 0.2631579 6
## 3    0.75    0.50  0.25  0.25 0.50    0.5 0.13333333 0.3684211 0.4210526 10
## 4    0.50    0.25  0.25  0.00 0.00    1.0 0.02666667 0.7894737 0.7368421 15
## 5    0.75    0.50  0.25  0.00 0.25    1.0 0.05333333 0.3157895 0.5263158 10
## 6    1.00    0.75  0.25  0.00 0.25    1.0 0.13333333 0.7894737 0.7894737 15
##   typeC FinalGradeR
## 1      1      0
## 2      1      0
## 3      1      1
## 4      1      1
## 5      1      1
## 6      1      1

```

```
dataR<-(dataR[,c(1:28, 30, 31)])
```

```
head(dataR)
```

```

##   school sex      age address famsize Pstatus Medu Fedu traveltime studytime
## 1     1    1 0.4285714      1     1       0 1.00 1.00 0.3333333 0.3333333
## 2     1    1 0.2857143      1     1       1 0.25 0.25 0.0000000 0.3333333
## 3     1    1 0.0000000      1     0       1 0.25 0.25 0.0000000 0.3333333
## 4     1    1 0.0000000      1     1       1 1.00 0.50 0.0000000 0.6666667
## 5     1    1 0.1428571      1     1       1 0.75 0.75 0.0000000 0.3333333
## 6     1    0 0.1428571      1     0       1 1.00 0.75 0.0000000 0.3333333
##   failures schoolsup famsup paid activities nursery higher internet romantic
## 1     0        1     0     0       0     1     1     0     0
## 2     0        0     1     0       0     0     1     1     0
## 3     1        1     0     1       0     1     1     1     0
## 4     0        0     1     1       1     1     1     1     1
## 5     0        0     1     1       0     1     1     0     0
## 6     0        0     1     1       1     1     1     1     0
##   famrel freetime goout Dalc Walc health absences      G1      G2 typeC
## 1   0.75    0.50  0.75 0.00 0.00    0.5 0.08000000 0.2631579 0.3157895   1
## 2   1.00    0.50  0.50 0.00 0.00    0.5 0.05333333 0.2631579 0.2631579   1
## 3   0.75    0.50  0.25 0.25 0.50    0.5 0.13333333 0.3684211 0.4210526   1
## 4   0.50    0.25  0.25 0.00 0.00    1.0 0.02666667 0.7894737 0.7368421   1
## 5   0.75    0.50  0.25 0.00 0.25    1.0 0.05333333 0.3157895 0.5263158   1
## 6   1.00    0.75  0.25 0.00 0.25    1.0 0.13333333 0.7894737 0.7894737   1
##   FinalGradeR
## 1     0
## 2     0
## 3     1
## 4     1
## 5     1
## 6     1

```

```

for (i in colnames(dataR)){
  dataR[[i]] = factor(dataR[[i]])
}

set.seed(2022)

# define the control using a RF selection function
control <- rfeControl(functions=rffFuncs, method="cv", number=10)
# run the algorithm
results <- rfe(dataR[,1:29], dataR[,30], sizes=c(1:29), rfeControl=control)
# Look at the results
print(results)

```

```

## 
## Recursive feature selection
##
## Outer resampling method: Cross-Validated (10 fold)
##
## Resampling performance over subset size:
##
##   Variables Accuracy  Kappa AccuracySD KappaSD Selected
##       1    0.9157 0.7275    0.01739 0.05989
##       2    0.9157 0.7379    0.01639 0.05138
##       3    0.9157 0.7417    0.01923 0.05762
##       4    0.9233 0.7681    0.01824 0.05516      *
##       5    0.9137 0.7386    0.02774 0.08223
##       6    0.9185 0.7531    0.01900 0.05718
##       7    0.9118 0.7264    0.01882 0.06156
##       8    0.9118 0.7277    0.01652 0.05231
##       9    0.9128 0.7315    0.01912 0.06191
##      10   0.9099 0.7208    0.01062 0.03599
##      11   0.9128 0.7285    0.01399 0.05065
##      12   0.9071 0.7121    0.01378 0.03963
##      13   0.9099 0.7216    0.01467 0.04856
##      14   0.9119 0.7235    0.01275 0.04065
##      15   0.9147 0.7328    0.01481 0.04752
##      16   0.9138 0.7314    0.01650 0.05106
##      17   0.9138 0.7297    0.01218 0.04326
##      18   0.9147 0.7334    0.01474 0.04862
##      19   0.9118 0.7240    0.01814 0.05828
##      20   0.9147 0.7341    0.01851 0.06087
##      21   0.9109 0.7201    0.01997 0.06649
##      22   0.9109 0.7201    0.02035 0.06723
##      23   0.9099 0.7170    0.01602 0.05285
##      24   0.9128 0.7243    0.01949 0.06444
##      25   0.9128 0.7280    0.02021 0.06524
##      26   0.9157 0.7383    0.02032 0.06561
##      27   0.9080 0.7114    0.01609 0.05029
##      28   0.9137 0.7294    0.01997 0.06645
##      29   0.9080 0.7101    0.01607 0.05226
##
## The top 4 variables (out of 4):
##   G2, G1, failures, typeC

```

```

# List the chosen predictors
predictors(results)

```

```

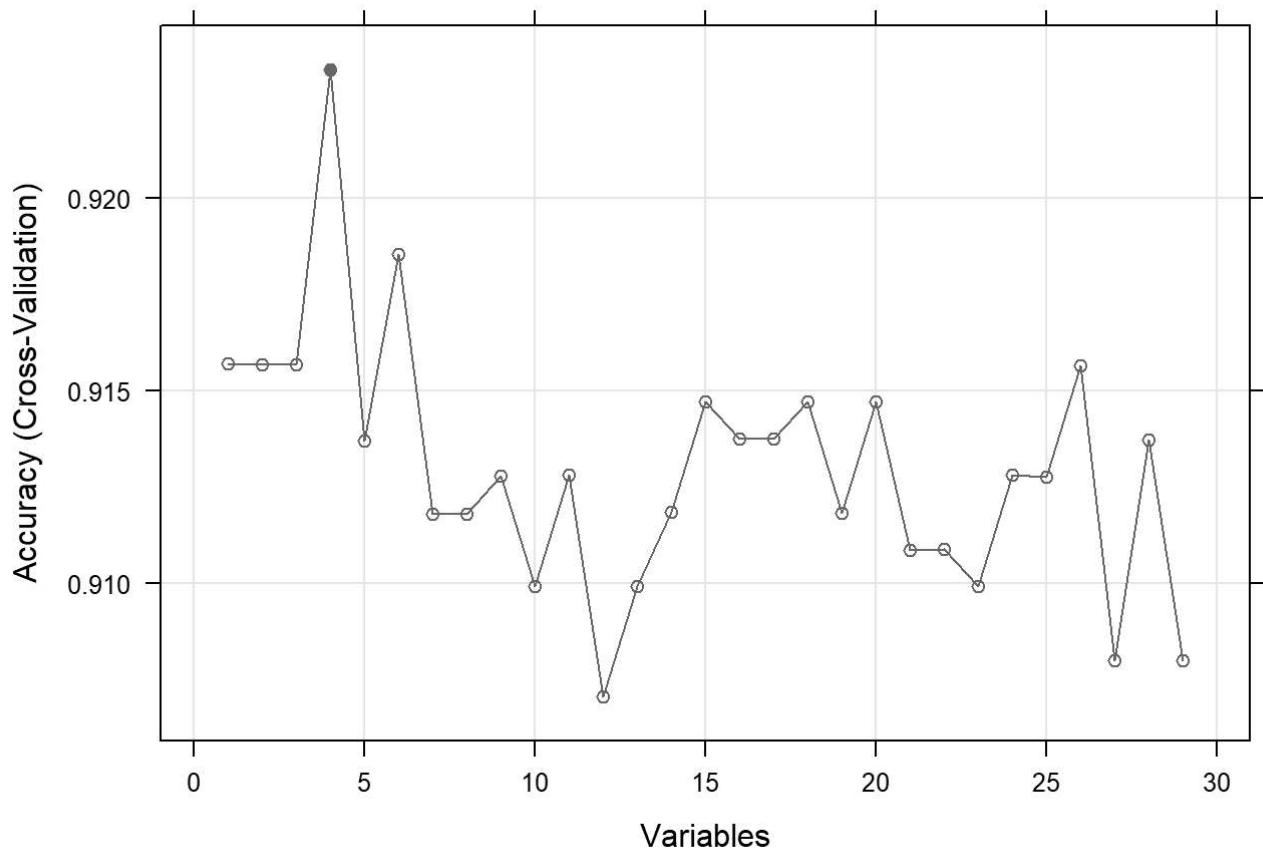
## [1] "G2"        "G1"        "failures"  "typeC"

```

```

# plot the results
plot(results, type=c("g", "o"))

```



```
#normalize numeric attributes
min_max_norm <- function(x){
  (x - min(x)) / (max(x) - min(x))
}
dataN<-(appendedDf[,c(1,2,3,4,5,6,7,8,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34)])
head(dataN)
```

```

## school sex age address famsize Pstatus Medu Fedu traveltime studytime
## 1     1   1 18      1     1     0     4     4          2       2
## 2     1   1 17      1     1     1     1     1          1       2
## 3     1   1 15      1     0     1     1     1          1       2
## 4     1   1 15      1     1     1     4     2          1       3
## 5     1   1 16      1     1     1     3     3          1       2
## 6     1   0 16      1     0     1     4     3          1       2
## failures schoolsup famsup paid activities nursery higher internet romantic
## 1     0         1     0     0          0     1     1     0     0
## 2     0         0     1     0          0     0     1     1     0
## 3     3         1     0     1          0     1     1     1     0
## 4     0         0     1     1          1     1     1     1     1
## 5     0         0     1     1          0     1     1     0     0
## 6     0         0     1     1          1     1     1     1     0
## famrel freetime goout Dalc Walc health absences G1 G2 G3 typeC
## 1     4         3     4     1     1     3          6     5     6     6     1
## 2     5         3     3     1     1     3          4     5     5     6     1
## 3     4         3     2     2     3     3          10    7     8    10     1
## 4     3         2     2     1     1     5          2    15    14    15     1
## 5     4         3     2     1     2     5          4     6    10    10     1
## 6     5         4     2     1     2     5          10    15   15    15     1

```

```

datanorm <- dataN %>%
  mutate(across(c(3,7,8,9,10,11,20,21,22,23,24,25,26,27,28),min_max_norm))

head(datanorm)

```

```

##   school sex      age address famsize Pstatus Medu Fedu traveltime studytime
## 1     1    1 0.4285714      1      1       0 1.00 1.00 0.3333333 0.3333333
## 2     1    1 0.2857143      1      1       1 0.25 0.25 0.0000000 0.3333333
## 3     1    1 0.0000000      1      0       1 0.25 0.25 0.0000000 0.3333333
## 4     1    1 0.0000000      1      1       1 1.00 0.50 0.0000000 0.6666667
## 5     1    1 0.1428571      1      1       1 0.75 0.75 0.0000000 0.3333333
## 6     1    0 0.1428571      1      0       1 1.00 0.75 0.0000000 0.3333333
##   failures schoolsup famsup paid activities nursery higher internet romantic
## 1     0        1    0    0       0      1    1      0      0
## 2     0        0    1    0       0      0    1      1      0
## 3     1        1    0    1       0      1    1      1      0
## 4     0        0    1    1       1      1    1      1      1
## 5     0        0    1    1       0      1    1      0      0
## 6     0        0    1    1       1      1    1      1      0
##   famrel freetime goout Dalc Walc health absences      G1      G2 G3
## 1 0.75    0.50 0.75 0.00 0.00    0.5 0.08000000 0.2631579 0.3157895 6
## 2 1.00    0.50 0.50 0.00 0.00    0.5 0.05333333 0.2631579 0.2631579 6
## 3 0.75    0.50 0.25 0.25 0.50    0.5 0.13333333 0.3684211 0.4210526 10
## 4 0.50    0.25 0.25 0.00 0.00    1.0 0.02666667 0.7894737 0.7368421 15
## 5 0.75    0.50 0.25 0.00 0.25    1.0 0.05333333 0.3157895 0.5263158 10
## 6 1.00    0.75 0.25 0.00 0.25    1.0 0.13333333 0.7894737 0.7894737 15
##   typeC
## 1     1
## 2     1
## 3     1
## 4     1
## 5     1
## 6     1

```

```

#create different sub groups of variables for the ml models

#add the target column
datagroupB <- datanorm %>%
  mutate(FinalGradeR = case_when(
    G3 < 10 ~ 0,
    G3 > 9 ~ 1
  ))

datagroupA <- (datanorm)

datagroupA <- datagroupA %>%
  mutate(FinalGradeR = case_when(
    G3 < 10 ~ 0,
    G3 > 9 ~ 1
  ))
datagroupA <- datagroupA[,c(1:28,30,31)]


datagroupB<-(datanorm[,c(11,27,28,29,30)]) 

#add the target column
datagroupB <- datagroupB %>%
  mutate(FinalGradeR = case_when(
    G3 < 10 ~ 0,
    G3 > 9 ~ 1
  ))
datagroupB <- datagroupB[,c(1:3,5,6)]


datagroupC <- (datanorm[,c(30,1,14,4,18,13,7,15,8,9,29)])
datagroupC <- datagroupC %>%
  mutate(FinalGradeR = case_when(
    G3 < 10 ~ 0,
    G3 > 9 ~ 1
  ))
datagroupC <- datagroupC[,c(1:10,12)]


datagroupLR <- dataN %>%
  mutate(across(c(3,26,27,28,29),min_max_norm))
datagroupLR <- (datagroupLR[,c(3,26,27,28,29)])
```

```

#linear regression model
lm<-lm(G3 ~ age + absences + G1 + G2, data = datagroupLR)
summary(lm)
```

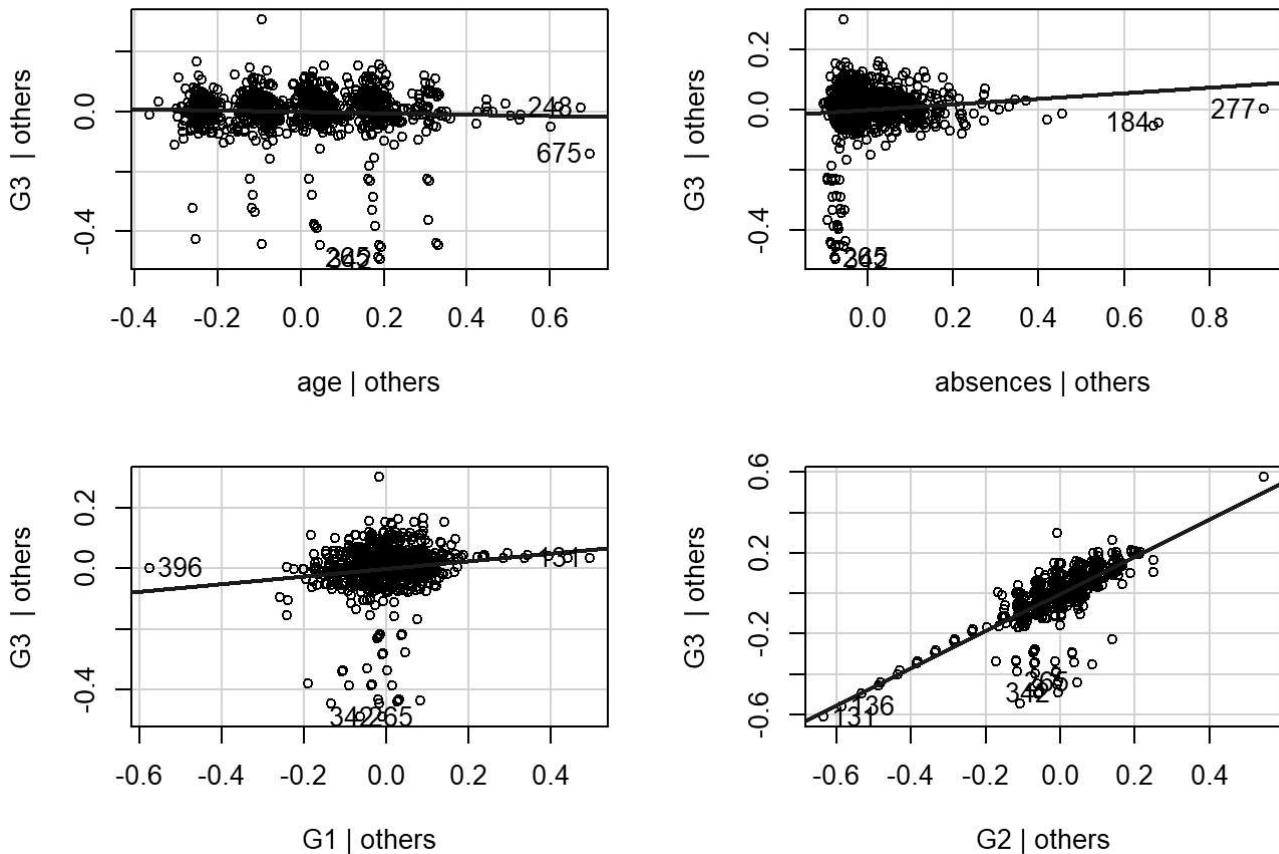
```

## 
## Call:
## lm(formula = G3 ~ age + absences + G1 + G2, data = datagroupLR)
##
## Residuals:
##    Min      1Q  Median      3Q     Max 
## -0.48900 -0.01997  0.00336  0.04108  0.30467 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -0.05274   0.01078  -4.893 1.15e-06 *** 
## age         -0.02215   0.01404  -1.577  0.11503    
## absences     0.09460   0.02994   3.160  0.00162 **  
## G1          0.12773   0.03039   4.203 2.86e-05 *** 
## G2          0.91962   0.02758  33.345 < 2e-16 *** 
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
##
## Residual standard error: 0.07887 on 1039 degrees of freedom
## Multiple R-squared:  0.834, Adjusted R-squared:  0.8334 
## F-statistic: 1305 on 4 and 1039 DF, p-value: < 2.2e-16

```

```
avPlots(lm)
```

Added-Variable Plots



```
#split data into train and test sets
train_index <- sample(1:nrow(datagroupA), 0.7 * nrow(datagroupA))
train.setA <- datagroupA[train_index,]
test.setA <- datagroupA[-train_index,]

train_index <- sample(1:nrow(datagroupB), 0.7 * nrow(datagroupB))
train.setB <- datagroupB[train_index,]
test.setB <- datagroupB[-train_index,]

train_index <- sample(1:nrow(datagroupC), 0.7 * nrow(datagroupC))
train.setC <- datagroupC[train_index,]
test.setC <- datagroupC[-train_index,]
```

```
#build Logistic regression models
glm_modelA <- glm(FinalGradeR ~ ., family = "binomial" (link=logit), data = train.setA)
summary(glm_modelA)
```

```

## 
## Call:
## glm(formula = FinalGradeR ~ ., family = binomial(link = logit),
##      data = train.setA)
## 
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max 
## -3.7718  0.0000  0.0093  0.1253  2.1496 
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept) -19.787987  2.690112 -7.356 1.90e-13 ***
## school       0.881952  0.480217  1.837  0.0663 .  
## sex          0.556339  0.465283  1.196  0.2318    
## age          1.290172  1.184388  1.089  0.2760    
## address      -0.300619  0.423801 -0.709  0.4781    
## famsize      0.406928  0.419358  0.970  0.3319    
## Pstatus      -0.975515  0.739024 -1.320  0.1868    
## Medu         1.615800  0.944574  1.711  0.0872 .  
## Fedu         -1.920646  0.925007 -2.076  0.0379 *  
## traveltime   0.388077  0.781142  0.497  0.6193    
## studytime    -1.347233  0.833561 -1.616  0.1060    
## failures     0.363988  0.719521  0.506  0.6129    
## schoolsup    0.648974  0.533544  1.216  0.2239    
## famsup        -0.009491  0.397175 -0.024  0.9809    
## paid          -0.545610  0.514314 -1.061  0.2888    
## activities   -0.531775  0.387334 -1.373  0.1698    
## nursery       0.147902  0.448228  0.330  0.7414    
## higher        0.534187  0.552998  0.966  0.3341    
## internet     0.239603  0.457583  0.524  0.6005    
## romantic     -0.571870  0.414829 -1.379  0.1680    
## famrel        -0.669258  0.804552 -0.832  0.4055    
## freetime      0.609520  0.790567  0.771  0.4407    
## goout         -1.592999  0.770191 -2.068  0.0386 *  
## Dalc          -1.489725  0.986027 -1.511  0.1308    
## Walc          2.087572  0.857832  2.434  0.0150 *  
## health        -0.461938  0.558120 -0.828  0.4079    
## absences      -5.947909  2.914228 -2.041  0.0413 *  
## G1            10.919820  2.529942  4.316  1.59e-05 *** 
## G2            34.076336  4.479684  7.607  2.81e-14 *** 
## typeC        -0.866669  0.492285 -1.761  0.0783 .  
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
## Null deviance: 777.80 on 729 degrees of freedom
## Residual deviance: 215.66 on 700 degrees of freedom
## AIC: 275.66
## 
## Number of Fisher Scoring iterations: 9

```

```
glm_modelB <- glm(FinalGradeR ~ ., family = "binomial" (link=logit), data = train.setB)
summary(glm_modelB)
```

```
##
## Call:
## glm(formula = FinalGradeR ~ ., family = binomial(link = logit),
##      data = train.setB)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -3.4700  0.0004  0.0203  0.1732  2.2423
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -17.7145   1.9740  -8.974 < 2e-16 ***
## failures      0.5704   0.7424   0.768   0.4423
## G1            8.6592   2.1259   4.073  4.64e-05 ***
## G2           29.6387   3.8221   7.755 8.86e-15 ***
## typeC        -0.8404   0.3494  -2.405   0.0162 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 760.07 on 729 degrees of freedom
## Residual deviance: 224.48 on 725 degrees of freedom
## AIC: 234.48
##
## Number of Fisher Scoring iterations: 8
```

```
glm_modelC <- glm(FinalGradeR ~ ., family = "binomial"(link=logit), data = train.setC)
summary(glm_modelC)
```

```

## 
## Call:
## glm(formula = FinalGradeR ~ ., family = binomial(link = logit),
##      data = train.setC)
## 
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max 
## -2.3057  0.3589  0.4908  0.7850  1.4697 
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept) 0.42664   0.35240   1.211   0.226    
## typeC       -1.59059   0.23074  -6.893 5.45e-12 *** 
## school      1.01950   0.24986   4.080 4.50e-05 *** 
## paid         0.29320   0.24836   1.181   0.238    
## address      0.07596   0.22471   0.338   0.735    
## internet     0.17243   0.23257   0.741   0.458    
## famsup      -0.30036   0.20323  -1.478   0.139    
## Medu        0.44347   0.45163   0.982   0.326    
## activities   0.13604   0.18882   0.720   0.471    
## Fedu        0.55842   0.45390   1.230   0.219    
## traveltime   0.32654   0.41505   0.787   0.431    
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
## Null deviance: 777.80  on 729  degrees of freedom 
## Residual deviance: 706.99  on 719  degrees of freedom 
## AIC: 728.99 
## 
## Number of Fisher Scoring iterations: 4

```

```

test.setA$pred <- predict(glm_modelA, newdata = test.setA, type = 'response')

ProbabilityCutoff <- 0.5
test.setA$pred.probs <- 1-test.setA$pred

test.setA$pred.passed <- ifelse(test.setA$pred > ProbabilityCutoff, 1, 0)

confusionMatrix(as.factor(test.setA$FinalGradeR), as.factor(test.setA$pred.passed))

```

```

## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0   1
##           0 54 12
##           1 10 238
##
##           Accuracy : 0.9299
##             95% CI : (0.8958, 0.9556)
##   No Information Rate : 0.7962
## P-Value [Acc > NIR] : 3.986e-11
##
##           Kappa : 0.7866
##
## McNemar's Test P-Value : 0.8312
##
##           Sensitivity : 0.8438
##           Specificity : 0.9520
##   Pos Pred Value : 0.8182
##   Neg Pred Value : 0.9597
##           Prevalence : 0.2038
##   Detection Rate : 0.1720
## Detection Prevalence : 0.2102
##   Balanced Accuracy : 0.8979
##
## 'Positive' Class : 0
##

```

```

test.setB$pred <- predict(glm_modelB, newdata = test.setB, type = 'response')

test.setB$pred.probs <- 1-test.setB$pred

test.setB$pred.passed <- ifelse(test.setB$pred > ProbabilityCutoff, 1, 0)

confusionMatrix(as.factor(test.setB$FinalGradeR), as.factor(test.setB$pred.passed))

```

```

## Confusion Matrix and Statistics
##
##             Reference
## Prediction   0    1
##           0 57 16
##           1 16 225
##
##                 Accuracy : 0.8981
##                   95% CI : (0.8592, 0.9292)
##       No Information Rate : 0.7675
##     P-Value [Acc > NIR] : 1.943e-09
##
##                 Kappa : 0.7144
##
## McNemar's Test P-Value : 1
##
##                 Sensitivity : 0.7808
##                 Specificity : 0.9336
##      Pos Pred Value : 0.7808
##      Neg Pred Value : 0.9336
##          Prevalence : 0.2325
##      Detection Rate : 0.1815
## Detection Prevalence : 0.2325
##     Balanced Accuracy : 0.8572
##
## 'Positive' Class : 0
##

```

```

test.setC$pred <- predict(glm_modelC, newdata = test.setC, type = 'response')
test.setC$pred.probs <- 1-test.setC$pred

test.setC$pred.passed <- ifelse(test.setC$pred > ProbabilityCutoff, 1, 0)

confusionMatrix(as.factor(test.setC$FinalGradeR), as.factor(test.setC$pred.passed))

```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0    1
##           0    7  59
##           1   10 238
##
##           Accuracy : 0.7803
##             95% CI : (0.7303, 0.8248)
## No Information Rate : 0.9459
## P-Value [Acc > NIR] : 1
##
##           Kappa : 0.0904
##
## McNemar's Test P-Value : 7.536e-09
##
##           Sensitivity : 0.41176
##           Specificity : 0.80135
## Pos Pred Value : 0.10606
## Neg Pred Value : 0.95968
## Prevalence : 0.05414
## Detection Rate : 0.02229
## Detection Prevalence : 0.21019
## Balanced Accuracy : 0.60656
##
## 'Positive' Class : 0
##
```

```
#build decision tree models
treeA <- rpart(FinalGradeR ~ ., data=train.setA, method="class", control = rpart.control(minsplit = 30, minbucket = 10, cp = 0.001))

summary(treeA)
```

```

## Call:
## rpart(formula = FinalGradeR ~ ., data = train.setA, method = "class",
##       control = rpart.control(minsplit = 30, minbucket = 10, cp = 0.001))
## n= 730
##
##          CP nsplit rel error      xerror      xstd
## 1 0.57926829      0 1.0000000 1.0000000 0.06875826
## 2 0.04573171      1 0.4207317 0.4756098 0.05089391
## 3 0.03658537      3 0.3292683 0.4634146 0.05031418
## 4 0.01219512      4 0.2926829 0.3658537 0.04524887
## 5 0.00100000      5 0.2804878 0.3231707 0.04274910
##
## Variable importance
##      G2      G1 failures   higher   famrel   typeC      age     Walc
##      51      27         9        6        2        1        1        1
## paid
##      1
##
## Node number 1: 730 observations,    complexity param=0.5792683
## predicted class=1 expected loss=0.2246575  P(node) =1
##   class counts: 164 566
##   probabilities: 0.225 0.775
## left son=2 (209 obs) right son=3 (521 obs)
## Primary splits:
##   G2      < 0.5      to the left,  improve=147.956000, (0 missing)
##   G1      < 0.4473684 to the left,  improve=111.337300, (0 missing)
##   failures < 0.1666667 to the right, improve= 32.323040, (0 missing)
##   higher   < 0.5      to the left,  improve= 15.651810, (0 missing)
##   typeC   < 0.5      to the right, improve=  8.972684, (0 missing)
## Surrogate splits:
##   G1      < 0.5      to the left,  agree=0.871, adj=0.550, (0 split)
##   failures < 0.1666667 to the right, agree=0.773, adj=0.206, (0 split)
##   higher   < 0.5      to the left,  agree=0.749, adj=0.124, (0 split)
##   age      < 0.9285714 to the right, agree=0.716, adj=0.010, (0 split)
##   absences < 0.2866667 to the right, agree=0.715, adj=0.005, (0 split)
##
## Node number 2: 209 observations,    complexity param=0.04573171
## predicted class=0 expected loss=0.2727273  P(node) =0.2863014
##   class counts: 152 57
##   probabilities: 0.727 0.273
## left son=4 (120 obs) right son=5 (89 obs)
## Primary splits:
##   G2      < 0.4473684 to the left,  improve=20.216210, (0 missing)
##   G1      < 0.3947368 to the left,  improve=11.142000, (0 missing)
##   typeC   < 0.5      to the right, improve= 5.108027, (0 missing)
##   absences < 0.14      to the right, improve= 2.090096, (0 missing)
##   health   < 0.125     to the left,  improve= 1.783799, (0 missing)
## Surrogate splits:
##   G1      < 0.4473684 to the left,  agree=0.732, adj=0.371, (0 split)
##   Fedu   < 0.875      to the left,  agree=0.589, adj=0.034, (0 split)
##   studytime < 0.5      to the left,  agree=0.589, adj=0.034, (0 split)
##   age     < 0.07142857 to the right, agree=0.584, adj=0.022, (0 split)

```

```

##      paid      < 0.5      to the left, agree=0.584, adj=0.022, (0 split)
##
## Node number 3: 521 observations
##   predicted class=1 expected loss=0.02303263 P(node) =0.7136986
##   class counts:    12    509
##   probabilities: 0.023 0.977
##
## Node number 4: 120 observations
##   predicted class=0 expected loss=0.08333333 P(node) =0.1643836
##   class counts:    110     10
##   probabilities: 0.917 0.083
##
## Node number 5: 89 observations, complexity param=0.04573171
##   predicted class=1 expected loss=0.4719101 P(node) =0.1219178
##   class counts:    42     47
##   probabilities: 0.472 0.528
##   left son=10 (34 obs) right son=11 (55 obs)
## Primary splits:
##   typeC      < 0.5      to the right, improve=3.375593, (0 missing)
##   absences    < 0.14     to the right, improve=2.425373, (0 missing)
##   studytime   < 0.5      to the right, improve=2.145265, (0 missing)
##   famrel     < 0.875     to the right, improve=1.512176, (0 missing)
##   Walc       < 0.625     to the left,  improve=1.488583, (0 missing)
## Surrogate splits:
##   paid        < 0.5      to the right, agree=0.764, adj=0.382, (0 split)
##   famrel     < 0.375     to the left,  agree=0.663, adj=0.118, (0 split)
##   studytime   < 0.5      to the right, agree=0.640, adj=0.059, (0 split)
##   Walc       < 0.875     to the right, agree=0.640, adj=0.059, (0 split)
##   absences    < 0.2533333 to the right, agree=0.640, adj=0.059, (0 split)
##
## Node number 10: 34 observations, complexity param=0.01219512
##   predicted class=0 expected loss=0.3529412 P(node) =0.04657534
##   class counts:    22     12
##   probabilities: 0.647 0.353
##   left son=20 (24 obs) right son=21 (10 obs)
## Primary splits:
##   Walc      < 0.625     to the left,  improve=1.7294120, (0 missing)
##   health    < 0.875     to the left,  improve=1.2053010, (0 missing)
##   G1        < 0.5      to the right, improve=0.9523367, (0 missing)
##   age        < 0.2142857 to the right, improve=0.6943240, (0 missing)
##   famrel    < 0.625     to the left,  improve=0.6627451, (0 missing)
## Surrogate splits:
##   Dalc      < 0.375     to the left,  agree=0.853, adj=0.5, (0 split)
##   traveltime < 0.5      to the left,  agree=0.765, adj=0.2, (0 split)
##   sex        < 0.5      to the right, agree=0.735, adj=0.1, (0 split)
##   Pstatus    < 0.5      to the right, agree=0.735, adj=0.1, (0 split)
##   studytime  < 0.1666667 to the right, agree=0.735, adj=0.1, (0 split)
##
## Node number 11: 55 observations, complexity param=0.03658537
##   predicted class=1 expected loss=0.3636364 P(node) =0.07534247
##   class counts:    20     35
##   probabilities: 0.364 0.636

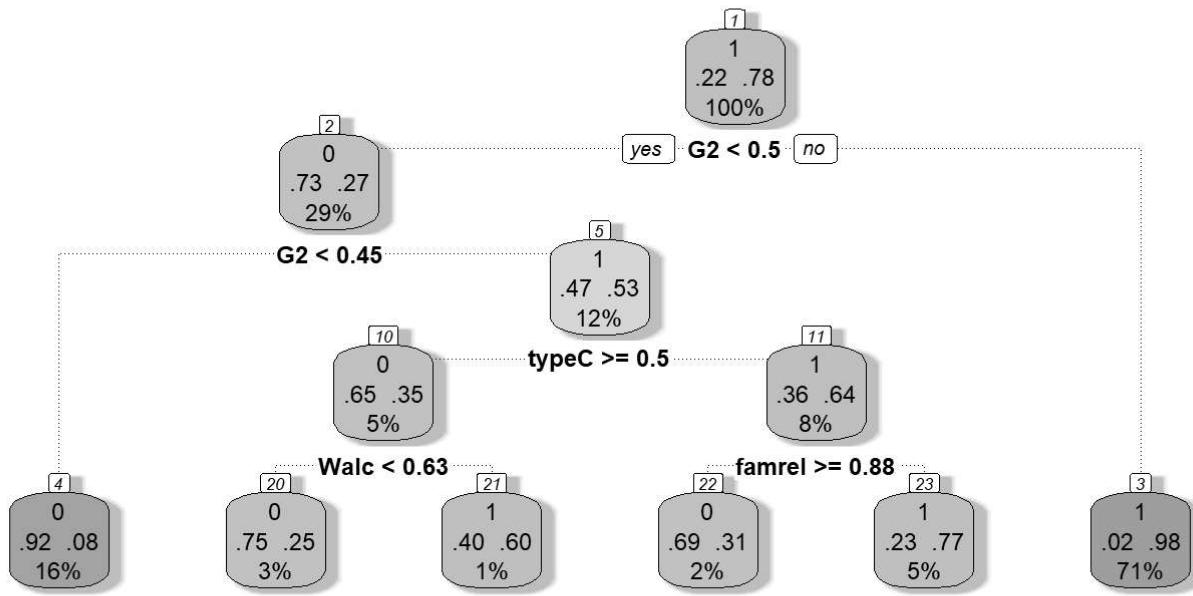
```

```

##  left son=22 (16 obs) right son=23 (39 obs)
## Primary splits:
##   famrel < 0.875      to the right, improve=4.733392, (0 missing)
##   G1      < 0.5       to the left,  improve=2.845850, (0 missing)
##   age     < 0.3571429 to the left,  improve=2.457469, (0 missing)
##   absences < 0.02    to the right, improve=1.498501, (0 missing)
##   health  < 0.875    to the right, improve=1.473064, (0 missing)
## Surrogate splits:
##   age      < 0.07142857 to the left,  agree=0.782, adj=0.250, (0 split)
##   G1      < 0.3947368  to the left,  agree=0.745, adj=0.125, (0 split)
##   traveltimes < 0.8333333 to the right, agree=0.727, adj=0.063, (0 split)
##   schoolsup < 0.5     to the right, agree=0.727, adj=0.063, (0 split)
##   health  < 0.125    to the left,  agree=0.727, adj=0.063, (0 split)
##
## Node number 20: 24 observations
##   predicted class=0  expected loss=0.25  P(node) =0.03287671
##   class counts: 18     6
##   probabilities: 0.750 0.250
##
## Node number 21: 10 observations
##   predicted class=1  expected loss=0.4   P(node) =0.01369863
##   class counts: 4     6
##   probabilities: 0.400 0.600
##
## Node number 22: 16 observations
##   predicted class=0  expected loss=0.3125 P(node) =0.02191781
##   class counts: 11     5
##   probabilities: 0.688 0.312
##
## Node number 23: 39 observations
##   predicted class=1  expected loss=0.2307692 P(node) =0.05342466
##   class counts: 9     30
##   probabilities: 0.231 0.769

```

```
fancyRpartPlot(treeA,caption = "Classification Tree")
```



Classification Tree

```
predictA <- predict(treeA, test.setA, type = 'class')
```

```
table_A<- table(test.setA$FinalGradeR, predictA)
table_A
```

```
##      predictA
##      0   1
##  0  53 13
##  1  9 239
```

```
accuracy_Test <- sum(diag(table_A)) / sum(table_A)
print(paste('Accuracy for tree A', accuracy_Test))
```

```
## [1] "Accuracy for tree A 0.929936305732484"
```

```
treeB <- rpart(FinalGradeR ~ ., data =train.setB, method = 'class', control = rpart.control(minsplit = 30, minbucket = 10, cp = 0.001))
summary(treeB)
```

```

## Call:
## rpart(formula = FinalGradeR ~ ., data = train.setB, method = "class",
##       control = rpart.control(minsplit = 30, minbucket = 10, cp = 0.001))
## n= 730
##
##          CP nsplit rel error      xerror      xstd
## 1 0.64968153      0 1.0000000 1.0000000 0.07070759
## 2 0.03821656      1 0.3503185 0.4394904 0.05034591
## 3 0.00100000      3 0.2738854 0.2738854 0.04051833
##
## Variable importance
##          G2      G1 failures      typeC
##          66      26       6       3
##
## Node number 1: 730 observations,    complexity param=0.6496815
##   predicted class=1 expected loss=0.2150685  P(node) =1
##   class counts:  157  573
##   probabilities: 0.215 0.785
##   left son=2 (116 obs) right son=3 (614 obs)
## Primary splits:
##   G2      < 0.4473684 to the left,  improve=144.81820, (0 missing)
##   G1      < 0.4473684 to the left,  improve=120.53060, (0 missing)
##   failures < 0.1666667 to the right, improve= 25.71011, (0 missing)
##   typeC   < 0.5      to the right, improve= 11.66517, (0 missing)
## Surrogate splits:
##   G1      < 0.4473684 to the left,  agree=0.911, adj=0.440, (0 split)
##   failures < 0.8333333 to the right, agree=0.858, adj=0.103, (0 split)
##
## Node number 2: 116 observations
##   predicted class=0 expected loss=0.06034483  P(node) =0.1589041
##   class counts:  109     7
##   probabilities: 0.940 0.060
##
## Node number 3: 614 observations,    complexity param=0.03821656
##   predicted class=1 expected loss=0.0781759  P(node) =0.8410959
##   class counts:  48  566
##   probabilities: 0.078 0.922
##   left son=6 (75 obs) right son=7 (539 obs)
## Primary splits:
##   G2      < 0.5      to the left,  improve=22.369990, (0 missing)
##   G1      < 0.5526316 to the left,  improve=13.522330, (0 missing)
##   failures < 0.5      to the right, improve= 1.399248, (0 missing)
##   typeC   < 0.5      to the right, improve= 1.156212, (0 missing)
## Surrogate splits:
##   G1 < 0.3947368 to the left,  agree=0.889, adj=0.093, (0 split)
##
## Node number 6: 75 observations,    complexity param=0.03821656
##   predicted class=1 expected loss=0.44  P(node) =0.1027397
##   class counts:  33  42
##   probabilities: 0.440 0.560
##   left son=12 (28 obs) right son=13 (47 obs)
## Primary splits:

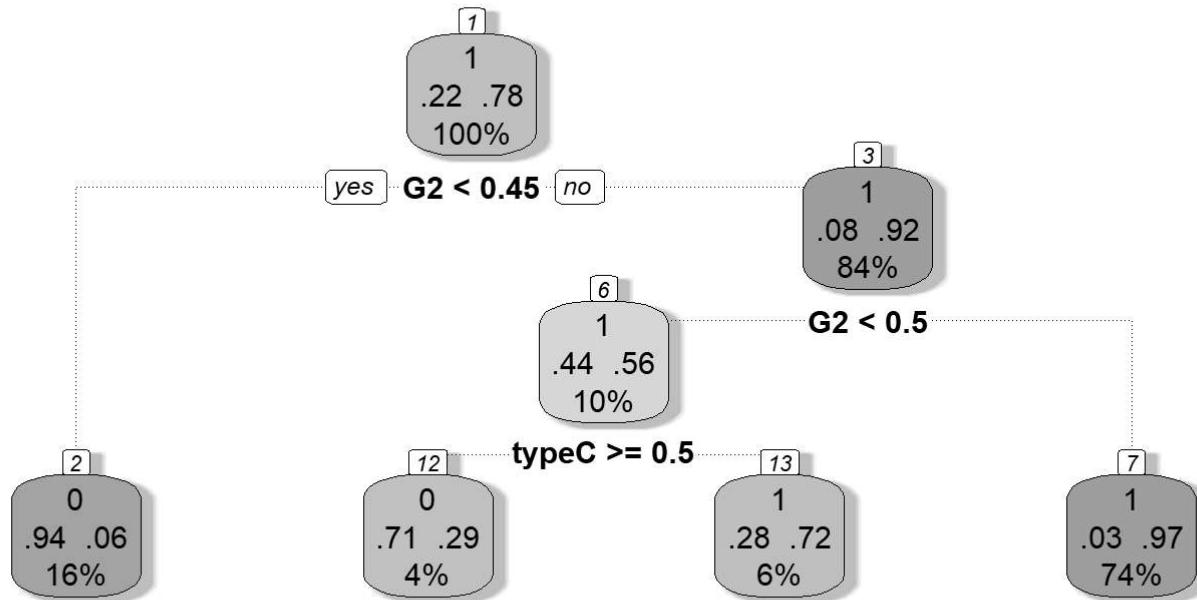
```

```

##      typeC < 0.5      to the right, improve=6.72291800, (0 missing)
##      G1      < 0.3947368 to the left,  improve=2.00218400, (0 missing)
##      failures < 0.1666667 to the left,  improve=0.03505071, (0 missing)
##
## Node number 7: 539 observations
##   predicted class=1  expected loss=0.02782931  P(node) =0.7383562
##   class counts:    15    524
##   probabilities: 0.028  0.972
##
## Node number 12: 28 observations
##   predicted class=0  expected loss=0.2857143  P(node) =0.03835616
##   class counts:    20     8
##   probabilities: 0.714  0.286
##
## Node number 13: 47 observations
##   predicted class=1  expected loss=0.2765957  P(node) =0.06438356
##   class counts:    13    34
##   probabilities: 0.277  0.723

```

```
fancyRpartPlot(treeB, caption = "Classification Tree")
```



Classification Tree

```
predictB <- predict(treeB, test.setB, type = 'class')
```

```
table_B<- table(test.setB$FinalGradeR, predictB)
table_B
```

```
##      predictB
##      0    1
##  0  59  14
##  1 16 225
```

```
accuracy_Test <- sum(diag(table_B)) / sum(table_B)
print(paste('Accuracy for tree B', accuracy_Test))
```

```
## [1] "Accuracy for tree B 0.904458598726115"
```

```
treeC <- rpart(FinalGradeR ~ ., data =train.setC, method = 'class', control = rpart.control(minsplit = 10, minbucket = 10, cp = 0.001))
```

```
summary(treeC)
```

```

## Call:
## rpart(formula = FinalGradeR ~ ., data = train.setC, method = "class",
##       control = rpart.control(minsplit = 10, minbucket = 10, cp = 0.001))
## n= 730
##
##          CP nsplit rel error  xerror      xstd
## 1 0.008130081     0 1.0000000 1.000000 0.06875826
## 2 0.006097561     6 0.9390244 1.085366 0.07074150
## 3 0.003810976     7 0.9329268 1.091463 0.07087565
## 4 0.001000000    15 0.9024390 1.115854 0.07140258
##
## Variable importance
##      typeC    school      paid      Medu      Fedu    famsup    internet
##      24        19        13        10        9        7        6
## traveltimes activities address
##      5         4         4
##
## Node number 1: 730 observations, complexity param=0.008130081
## predicted class=1 expected loss=0.2246575 P(node) =1
##   class counts: 164 566
##   probabilities: 0.225 0.775
## left son=2 (278 obs) right son=3 (452 obs)
## Primary splits:
##   typeC < 0.5      to the right, improve=12.306760, (0 missing)
##   school < 0.5      to the left,  improve= 3.020075, (0 missing)
##   Medu < 0.875      to the left,  improve= 2.405894, (0 missing)
##   Fedu < 0.375      to the left,  improve= 2.066714, (0 missing)
##   internet < 0.5    to the left,  improve= 1.372363, (0 missing)
## Surrogate splits:
##   paid < 0.5      to the right, agree=0.748, adj=0.338, (0 split)
##
## Node number 2: 278 observations, complexity param=0.008130081
## predicted class=1 expected loss=0.3417266 P(node) =0.3808219
##   class counts: 95 183
##   probabilities: 0.342 0.658
## left son=4 (152 obs) right son=5 (126 obs)
## Primary splits:
##   paid < 0.5      to the left,  improve=2.3816750, (0 missing)
##   Medu < 0.375      to the left,  improve=1.4243150, (0 missing)
##   famsup < 0.5      to the right, improve=0.8333849, (0 missing)
##   Fedu < 0.375      to the left,  improve=0.6338959, (0 missing)
##   activities < 0.5    to the left,  improve=0.1875245, (0 missing)
## Surrogate splits:
##   famsup < 0.5      to the left,  agree=0.622, adj=0.167, (0 split)
##   Medu < 0.875      to the left,  agree=0.608, adj=0.135, (0 split)
##   Fedu < 0.875      to the left,  agree=0.554, adj=0.016, (0 split)
##
## Node number 3: 452 observations, complexity param=0.003810976
## predicted class=1 expected loss=0.1526549 P(node) =0.6191781
##   class counts: 69 383
##   probabilities: 0.153 0.847
## left son=6 (153 obs) right son=7 (299 obs)

```

```

## Primary splits:
##   school < 0.5      to the left, improve=9.257059, (0 missing)
##   Fedu    < 0.625    to the left, improve=3.354053, (0 missing)
##   Medu    < 0.875    to the left, improve=3.098157, (0 missing)
##   address < 0.5     to the left, improve=2.188422, (0 missing)
##   internet < 0.5   to the left, improve=2.043080, (0 missing)
## Surrogate splits:
##   address < 0.5     to the left, agree=0.732, adj=0.209, (0 split)
##   internet < 0.5    to the left, agree=0.695, adj=0.098, (0 split)
##   Medu    < 0.375    to the left, agree=0.679, adj=0.052, (0 split)
##   traveltim< 0.5    to the right, agree=0.666, adj=0.013, (0 split)
##
## Node number 4: 152 observations, complexity param=0.008130081
## predicted class=1 expected loss=0.4013158 P(node) =0.2082192
##   class counts: 61 91
##   probabilities: 0.401 0.599
##   left son=8 (30 obs) right son=9 (122 obs)
## Primary splits:
##   Medu    < 0.375    to the left, improve=2.0438450, (0 missing)
##   famsup < 0.5       to the right, improve=1.9579150, (0 missing)
##   Fedu    < 0.375    to the left, improve=0.8399571, (0 missing)
##   internet < 0.5    to the left, improve=0.4743396, (0 missing)
##   school < 0.5     to the left, improve=0.3482972, (0 missing)
## Surrogate splits:
##   school < 0.5     to the left, agree=0.816, adj=0.067, (0 split)
##
## Node number 5: 126 observations
## predicted class=1 expected loss=0.2698413 P(node) =0.1726027
##   class counts: 34 92
##   probabilities: 0.270 0.730
##
## Node number 6: 153 observations, complexity param=0.003810976
## predicted class=1 expected loss=0.2941176 P(node) =0.209589
##   class counts: 45 108
##   probabilities: 0.294 0.706
##   left son=12 (118 obs) right son=13 (35 obs)
## Primary splits:
##   Fedu    < 0.625    to the left, improve=2.9352230, (0 missing)
##   Medu    < 0.375    to the left, improve=1.1557000, (0 missing)
##   paid    < 0.5       to the right, improve=0.9070341, (0 missing)
##   traveltim< 0.1666667 to the left, improve=0.4022532, (0 missing)
##   internet < 0.5    to the left, improve=0.2794118, (0 missing)
## Surrogate splits:
##   Medu < 0.875     to the left, agree=0.804, adj=0.143, (0 split)
##
## Node number 7: 299 observations
## predicted class=1 expected loss=0.08026756 P(node) =0.409589
##   class counts: 24 275
##   probabilities: 0.080 0.920
##
## Node number 8: 30 observations, complexity param=0.006097561
## predicted class=0 expected loss=0.4333333 P(node) =0.04109589

```

```

##      class counts:    17    13
##      probabilities: 0.567 0.433
##      left son=16 (13 obs) right son=17 (17 obs)
##      Primary splits:
##          activities < 0.5      to the right, improve=0.72428360, (0 missing)
##          famsup     < 0.5      to the right, improve=0.16874000, (0 missing)
##          Fedu       < 0.375    to the left,  improve=0.13333330, (0 missing)
##          travelttime < 0.16666667 to the left,  improve=0.03650075, (0 missing)
##          internet    < 0.5      to the left,  improve=0.03333333, (0 missing)
##      Surrogate splits:
##          famsup     < 0.5      to the right, agree=0.733, adj=0.385, (0 split)
##          school     < 0.5      to the right, agree=0.667, adj=0.231, (0 split)
##          address     < 0.5      to the right, agree=0.667, adj=0.231, (0 split)
##          travelttime < 0.5      to the right, agree=0.600, adj=0.077, (0 split)
##
## Node number 9: 122 observations,    complexity param=0.008130081
##   predicted class=1 expected loss=0.3606557 P(node) =0.1671233
##   class counts:    44    78
##   probabilities: 0.361 0.639
##   left son=18 (66 obs) right son=19 (56 obs)
##   Primary splits:
##       famsup     < 0.5      to the right, improve=2.5350220, (0 missing)
##       travelttime < 0.5      to the left,  improve=1.0016890, (0 missing)
##       Fedu       < 0.625    to the left,  improve=0.3100563, (0 missing)
##       internet    < 0.5      to the left,  improve=0.2574874, (0 missing)
##       Medu       < 0.875    to the left,  improve=0.2221948, (0 missing)
##   Surrogate splits:
##       school     < 0.5      to the right, agree=0.598, adj=0.125, (0 split)
##       Fedu       < 0.375    to the right, agree=0.598, adj=0.125, (0 split)
##       Medu       < 0.875    to the right, agree=0.590, adj=0.107, (0 split)
##       internet   < 0.5      to the right, agree=0.557, adj=0.036, (0 split)
##
## Node number 12: 118 observations,    complexity param=0.003810976
##   predicted class=1 expected loss=0.3474576 P(node) =0.1616438
##   class counts:    41    77
##   probabilities: 0.347 0.653
##   left son=24 (59 obs) right son=25 (59 obs)
##   Primary splits:
##       Fedu       < 0.375    to the left,  improve=0.8305085, (0 missing)
##       activities < 0.5      to the right, improve=0.3589282, (0 missing)
##       travelttime < 0.5      to the left,  improve=0.3020352, (0 missing)
##       address     < 0.5      to the left,  improve=0.1308275, (0 missing)
##       Medu       < 0.375    to the left,  improve=0.1130985, (0 missing)
##   Surrogate splits:
##       Medu       < 0.375    to the left,  agree=0.712, adj=0.424, (0 split)
##       internet    < 0.5      to the right, agree=0.585, adj=0.169, (0 split)
##       activities < 0.5      to the right, agree=0.542, adj=0.085, (0 split)
##       travelttime < 0.16666667 to the right, agree=0.534, adj=0.068, (0 split)
##       paid        < 0.5      to the left,  agree=0.525, adj=0.051, (0 split)
##
## Node number 13: 35 observations
##   predicted class=1 expected loss=0.1142857 P(node) =0.04794521

```

```

##      class counts:    4    31
##      probabilities: 0.114 0.886
##
## Node number 16: 13 observations
##      predicted class=0  expected loss=0.3076923  P(node) =0.01780822
##      class counts:    9    4
##      probabilities: 0.692 0.308
##
## Node number 17: 17 observations
##      predicted class=1  expected loss=0.4705882  P(node) =0.02328767
##      class counts:    8    9
##      probabilities: 0.471 0.529
##
## Node number 18: 66 observations,    complexity param=0.008130081
##      predicted class=1  expected loss=0.4545455  P(node) =0.09041096
##      class counts:   30    36
##      probabilities: 0.455 0.545
##      left son=36 (28 obs) right son=37 (38 obs)
## Primary splits:
##      activities < 0.5      to the left,  improve=1.3287760, (0 missing)
##      Medu       < 0.875     to the left,  improve=0.6475006, (0 missing)
##      internet    < 0.5      to the left,  improve=0.4865320, (0 missing)
##      Fedu       < 0.625     to the left,  improve=0.3448337, (0 missing)
##      travelttime < 0.1666667 to the left,  improve=0.3336219, (0 missing)
## Surrogate splits:
##      internet < 0.5      to the left,  agree=0.606, adj=0.071, (0 split)
##      Medu       < 0.625     to the left,  agree=0.591, adj=0.036, (0 split)
##      Fedu       < 0.375     to the left,  agree=0.591, adj=0.036, (0 split)
##
## Node number 19: 56 observations
##      predicted class=1  expected loss=0.25  P(node) =0.07671233
##      class counts:   14    42
##      probabilities: 0.250 0.750
##
## Node number 24: 59 observations,    complexity param=0.003810976
##      predicted class=1  expected loss=0.4067797  P(node) =0.08082192
##      class counts:   24    35
##      probabilities: 0.407 0.593
##      left son=48 (43 obs) right son=49 (16 obs)
## Primary splits:
##      travelttime < 0.1666667 to the right,  improve=0.39027390, (0 missing)
##      internet    < 0.5      to the left,  improve=0.31417530, (0 missing)
##      Medu       < 0.625     to the left,  improve=0.27457630, (0 missing)
##      famsup      < 0.5      to the right, improve=0.13198370, (0 missing)
##      address      < 0.5      to the left,  improve=0.08171913, (0 missing)
## Surrogate splits:
##      Medu < 0.875      to the left,  agree=0.78, adj=0.188, (0 split)
##
## Node number 25: 59 observations,    complexity param=0.003810976
##      predicted class=1  expected loss=0.2881356  P(node) =0.08082192
##      class counts:   17    42
##      probabilities: 0.288 0.712

```

```

##  left son=50 (20 obs) right son=51 (39 obs)
## Primary splits:
##   travelttime < 0.1666667 to the left,  improve=1.58544100, (0 missing)
##   activities < 0.5      to the right,  improve=0.80242360, (0 missing)
##   Medu       < 0.625    to the right,  improve=0.50338980, (0 missing)
##   internet   < 0.5      to the right,  improve=0.11813640, (0 missing)
##   famsup     < 0.5      to the left,   improve=0.03555766, (0 missing)
## Surrogate splits:
##   paid < 0.5      to the right,  agree=0.678, adj=0.05, (0 split)
##
## Node number 36: 28 observations,  complexity param=0.008130081
## predicted class=0 expected loss=0.4285714 P(node) =0.03835616
##   class counts: 16 12
##   probabilities: 0.571 0.429
##  left son=72 (18 obs) right son=73 (10 obs)
## Primary splits:
##   Medu < 0.625    to the right,  improve=0.91428570, (0 missing)
##   Fedu < 0.625    to the right,  improve=0.05274725, (0 missing)
## Surrogate splits:
##   Fedu < 0.625    to the right,  agree=0.75, adj=0.3, (0 split)
##
## Node number 37: 38 observations
## predicted class=1 expected loss=0.3684211 P(node) =0.05205479
##   class counts: 14 24
##   probabilities: 0.368 0.632
##
## Node number 48: 43 observations,  complexity param=0.003810976
## predicted class=1 expected loss=0.4418605 P(node) =0.05890411
##   class counts: 19 24
##   probabilities: 0.442 0.558
##  left son=96 (33 obs) right son=97 (10 obs)
## Primary splits:
##   travelttime < 0.5      to the left,  improve=0.52445380, (0 missing)
##   internet   < 0.5      to the left,  improve=0.43102180, (0 missing)
##   famsup     < 0.5      to the right,  improve=0.01449713, (0 missing)
##   activities < 0.5      to the left,  improve=0.01449713, (0 missing)
##   Medu       < 0.375    to the right,  improve=0.01443053, (0 missing)
##
## Node number 49: 16 observations
## predicted class=1 expected loss=0.3125 P(node) =0.02191781
##   class counts: 5 11
##   probabilities: 0.312 0.688
##
## Node number 50: 20 observations,  complexity param=0.003810976
## predicted class=1 expected loss=0.45 P(node) =0.02739726
##   class counts: 9 11
##   probabilities: 0.450 0.550
##  left son=100 (10 obs) right son=101 (10 obs)
## Primary splits:
##   internet < 0.5      to the right,  improve=0.9, (0 missing)
##   famsup   < 0.5      to the right,  improve=0.1, (0 missing)
## Surrogate splits:

```

```

##      Medu      < 0.375      to the left, agree=0.65, adj=0.3, (0 split)
##      famsup    < 0.5      to the left, agree=0.60, adj=0.2, (0 split)
##      paid      < 0.5      to the right, agree=0.55, adj=0.1, (0 split)
##      activities < 0.5      to the right, agree=0.55, adj=0.1, (0 split)
##
## Node number 51: 39 observations
##   predicted class=1 expected loss=0.2051282 P(node) =0.05342466
##   class counts:     8     31
##   probabilities: 0.205 0.795
##
## Node number 72: 18 observations
##   predicted class=0 expected loss=0.3333333 P(node) =0.02465753
##   class counts:     12      6
##   probabilities: 0.667 0.333
##
## Node number 73: 10 observations
##   predicted class=1 expected loss=0.4  P(node) =0.01369863
##   class counts:     4      6
##   probabilities: 0.400 0.600
##
## Node number 96: 33 observations, complexity param=0.003810976
##   predicted class=1 expected loss=0.4848485 P(node) =0.04520548
##   class counts:     16     17
##   probabilities: 0.485 0.515
##   left son=192 (11 obs) right son=193 (22 obs)
## Primary splits:
##      internet < 0.5      to the left, improve=0.757575800, (0 missing)
##      activities < 0.5      to the left, improve=0.729292900, (0 missing)
##      address    < 0.5      to the left, improve=0.431002300, (0 missing)
##      famsup     < 0.5      to the right, improve=0.129292900, (0 missing)
##      Medu       < 0.375    to the right, improve=0.008658009, (0 missing)
##
## Node number 97: 10 observations
##   predicted class=1 expected loss=0.3  P(node) =0.01369863
##   class counts:     3      7
##   probabilities: 0.300 0.700
##
## Node number 100: 10 observations
##   predicted class=0 expected loss=0.4  P(node) =0.01369863
##   class counts:     6      4
##   probabilities: 0.600 0.400
##
## Node number 101: 10 observations
##   predicted class=1 expected loss=0.3  P(node) =0.01369863
##   class counts:     3      7
##   probabilities: 0.300 0.700
##
## Node number 192: 11 observations
##   predicted class=0 expected loss=0.3636364 P(node) =0.01506849
##   class counts:     7      4
##   probabilities: 0.636 0.364
##

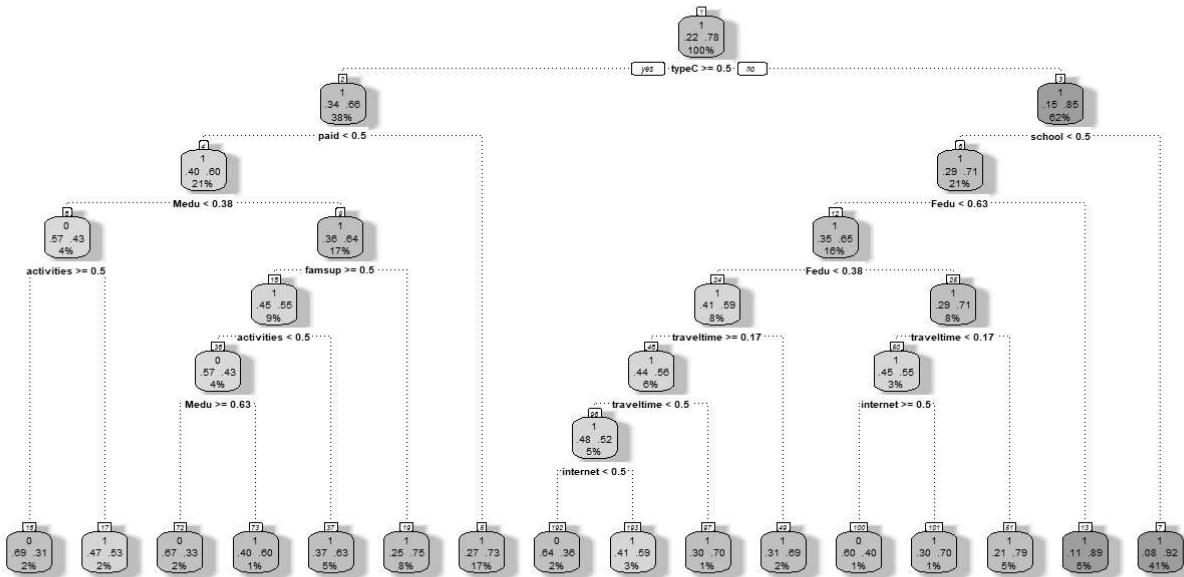
```

```

## Node number 193: 22 observations
##   predicted class=1  expected loss=0.4090909  P(node) =0.03013699
##   class counts:    9    13
##   probabilities: 0.409 0.591

```

```
fancyRpartPlot(treeC, caption = "Classification Tree")
```



Classification Tree

```

predictC <- predict(treeC, test.setC, type = 'class')

table_C<- table(test.setC$FinalGradeR, predictC)
table_C

```

```

##   predictC
##   0    1
##   0   12  54
##   1   13 235

```

```
accuracy_Test <- sum(diag(table_C)) / sum(table_C)
print(paste('Accuracy for tree C', accuracy_Test))
```

```
## [1] "Accuracy for tree C 0.786624203821656"
```

```
# buid kNN model

target_category <- train.setA[,30]
test_category <- test.setA[,30]
test.setA <- test.setA[,1:30]

knnmodel <- knn(train.setA, test.setA, cl=target_category, k = 10)
tab <- table(knnmodel,test_category)
accuracy <- function(x){sum(diag(x))/(sum(rowSums(x))) * 100}
accuracy(tab)
```

```
## [1] 96.17834
```

```
confusionMatrix(table(knnmodel ,test_category))
```

```
## Confusion Matrix and Statistics
##
##          test_category
## knnmodel   0    1
##          0  54   0
##          1 12 248
##
##          Accuracy : 0.9618
##                 95% CI : (0.9342, 0.9801)
## No Information Rate : 0.7898
## P-Value [Acc > NIR] : < 2.2e-16
##
##          Kappa : 0.8767
##
## Mcnemar's Test P-Value : 0.001496
##
##          Sensitivity : 0.8182
##          Specificity : 1.0000
## Pos Pred Value : 1.0000
## Neg Pred Value : 0.9538
##          Prevalence : 0.2102
## Detection Rate : 0.1720
## Detection Prevalence : 0.1720
## Balanced Accuracy : 0.9091
##
## 'Positive' Class : 0
##
```

```
target_category <- train.setB[,5]
test_category <- test.setB[,5]
test.setB <- test.setB[,1:5]

knnmodelB <- knn(train.setB, test.setB, cl=target_category, k = 10)
tab <- table(knnmodelB,test_category)
accuracy <- function(x){sum(diag(x))/(sum(rowSums(x))) * 100}
accuracy(tab)
```

```
## [1] 100
```

```
confusionMatrix(table(knnmodelB ,test_category))
```

```
## Confusion Matrix and Statistics
##
##          test_category
## knnmodelB    0    1
##          0 73   0
##          1   0 241
##
##          Accuracy : 1
##             95% CI : (0.9883, 1)
##     No Information Rate : 0.7675
##     P-Value [Acc > NIR] : < 2.2e-16
##
##          Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
##          Sensitivity : 1.0000
##          Specificity : 1.0000
##     Pos Pred Value : 1.0000
##     Neg Pred Value : 1.0000
##          Prevalence : 0.2325
##     Detection Rate : 0.2325
## Detection Prevalence : 0.2325
##     Balanced Accuracy : 1.0000
##
##     'Positive' Class : 0
##
```

```
target_category <- train.setC[,11]
test_category <- test.setC[,11]
test.setC <- test.setC[,1:11]

knnmodelC <- knn(train.setC, test.setC, cl=target_category, k = 10)
tab <- table(knnmodelC,test_category)
accuracy <- function(x){sum(diag(x))/(sum(rowSums(x))) * 100}
accuracy(tab)
```

```
## [1] 98.40764
```

```
confusionMatrix(table(knnmodelC ,test_category))
```

```
## Confusion Matrix and Statistics
##
##          test_category
## knnmodelC   0    1
##           0  61   0
##           1   5 248
##
##          Accuracy : 0.9841
##             95% CI : (0.9632, 0.9948)
##    No Information Rate : 0.7898
##    P-Value [Acc > NIR] : < 2e-16
##
##          Kappa : 0.9507
##
##  Mcnemar's Test P-Value : 0.07364
##
##          Sensitivity : 0.9242
##          Specificity : 1.0000
##    Pos Pred Value : 1.0000
##    Neg Pred Value : 0.9802
##          Prevalence : 0.2102
##          Detection Rate : 0.1943
##  Detection Prevalence : 0.1943
##          Balanced Accuracy : 0.9621
##
##          'Positive' Class : 0
##
```