Student's Performance

Promi Roy

2022-09-16

Introduction to Data

Data Source - https://www.kaggle.com/code/scubethoven/student-grade-prediction/data (https://www.kaggle.com/code/scubethoven/student-grade-prediction/data)

Attribute Information:

1. school - student's school (binary: 'GP' - Gabriel Pereira or 'MS' - Mousinho da Silveira)
2. sex - student's sex (binary: 'F' - female or 'M' - male)
3. age - student's age (numeric: from 15 to 22)
4. address - student's home address type (binary: 'U' - urban or 'R' - rural)
5. famsize - family size (binary: 'LE3' - less or equal to 3 or 'GT3' - greater than 3)
6. Pstatus - parent's cohabitation status (binary: 'T' - living together or 'A' - apart)
7. Medu - mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 - primary education (5th grade to 9th grade), 3 - secondary education or 4 - higher education)
8. Fedu - father's education (numeric: 0 - none, 1 - primary education (4th grade), 2 - primary education (5th to 9th grade), 3 - secondary education or 4 - higher education)
9. Mjob - mother's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
10. Fjob - father's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
11. reason - reason to choose this school (nominal: close to 'home', school 'reputation', 'course' preference or 'other')
12. guardian - student's guardian (nominal: 'mother', 'father' or 'other')
13. traveltime - home to school travel time (numeric: 1 - <15 min., 2 - 15 to 30 min., 3 - 30 min. to 1 hour, or 4 - >1 hour)
14. studytime - weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)
15. failures - number of past class failures (numeric: n if 1<=n<3, else 4)
16. schoolsup - extra educational support (binary: yes or no)
17. famsup - family educational support (binary: yes or no)
18. paid - extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)

19. activities - extra-curricular activities (binary: yes or no)
20. nursery - attended nursery school (binary: yes or no)
21. higher - wants to take higher education (binary: yes or no)
22. internet - Internet access at home (binary: yes or no)
23. romantic - with a romantic relationship (binary: yes or no)
24. famrel - quality of family relationships (numeric: from 1 - very bad to 5 - excellent)
25. freetime - free time after school (numeric: from 1 - very low to 5 - very high)
26. goout - going out with friends (numeric: from 1 - very low to 5 - very high)
27. Dalc - workday alcohol consumption (numeric: from 1 - very low to 5 - very high)
28. Walc - weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)
29. health - current health status (numeric: from 1 - very bad to 5 - very good)
30. absences - number of school absences (numeric: from 0 to 93)
31. G1 - 1st period grade
32. G2 - 2nd period grade
33. G3 - Final grade (Target variable)

Exploratory Data Analysis

```
#Loading data
df<-read.csv("C:/Users/Promi/OneDrive/Desktop/student-mat.csv", header=T)
df=data.frame(df)
attach(df)
head(df)
```

```
##   school sex age address famsize Pstatus Medu Fedu     Mjob     Fjob     reason
## 1     GP   F  18       U     GT3       A    4    4  at_home  teacher     course
## 2     GP   F  17       U     GT3       T    1    1  at_home    other     course
## 3     GP   F  15       U     LE3       T    1    1  at_home    other      other
## 4     GP   F  15       U     GT3       T    4    2   health services       home
## 5     GP   F  16       U     GT3       T    3    3    other    other       home
## 6     GP   M  16       U     LE3       T    4    3 services    other reputation
##   guardian traveltime studytime failures schoolsup famsup paid activities
## 1   mother          2         2        0       yes     no   no         no
## 2   father          1         2        0        no    yes   no         no
## 3   mother          1         2        3       yes     no  yes         no
## 4   mother          1         3        0        no    yes  yes        yes
## 5   father          1         2        0        no    yes  yes         no
## 6   mother          1         2        0        no    yes  yes        yes
##   nursery higher internet romantic famrel freetime goout Dalc Walc health
## 1     yes    yes       no       no      4        3     4    1    1      3
## 2      no    yes      yes       no      5        3     3    1    1      3
## 3     yes    yes      yes       no      4        3     2    2    3      3
## 4     yes    yes      yes      yes      3        2     2    1    1      5
## 5     yes    yes       no       no      4        3     2    1    2      5
## 6     yes    yes      yes       no      5        4     2    1    2      5
##   absences G1 G2 G3
## 1        6  5  6  6
## 2        4  5  5  6
## 3       10  7  8 10
## 4        2 15 14 15
## 5        4  6 10 10
## 6       10 15 15 15
```

```
# Summary of the data
summary(df)
```

```
##    school              sex                 age         address
##  Length:395         Length:395         Min.   :15.0   Length:395
##  Class :character   Class :character   1st Qu.:16.0   Class :character
##  Mode  :character   Mode  :character   Median :17.0   Mode  :character
##                                        Mean   :16.7
##                                        3rd Qu.:18.0
##                                        Max.   :22.0
##    famsize            Pstatus              Medu            Fedu
##  Length:395         Length:395         Min.   :0.000   Min.   :0.000
##  Class :character   Class :character   1st Qu.:2.000   1st Qu.:2.000
##  Mode  :character   Mode  :character   Median :3.000   Median :2.000
##                                        Mean   :2.749   Mean   :2.522
##                                        3rd Qu.:4.000   3rd Qu.:3.000
##                                        Max.   :4.000   Max.   :4.000
##     Mjob               Fjob               reason            guardian
##  Length:395         Length:395         Length:395         Length:395
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##    traveltime      studytime        failures       schoolsup
##  Min.   :1.000   Min.   :1.000   Min.   :0.0000   Length:395
##  1st Qu.:1.000   1st Qu.:1.000   1st Qu.:0.0000   Class :character
##  Median :1.000   Median :2.000   Median :0.0000   Mode  :character
##  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
##    famsup              paid            activities          nursery
##  Length:395         Length:395         Length:395         Length:395
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##    higher            internet           romantic            famrel
##  Length:395         Length:395         Length:395         Min.   :1.000
##  Class :character   Class :character   Class :character   1st Qu.:4.000
##  Mode  :character   Mode  :character   Mode  :character   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
```

```
# Checking null values
is.null(df)
```

```
## [1] FALSE
```

There are no null values in the dataset.

```
#Checking unique values for eah attribute
list_unique<-lapply(df, unique)
list_unique
```

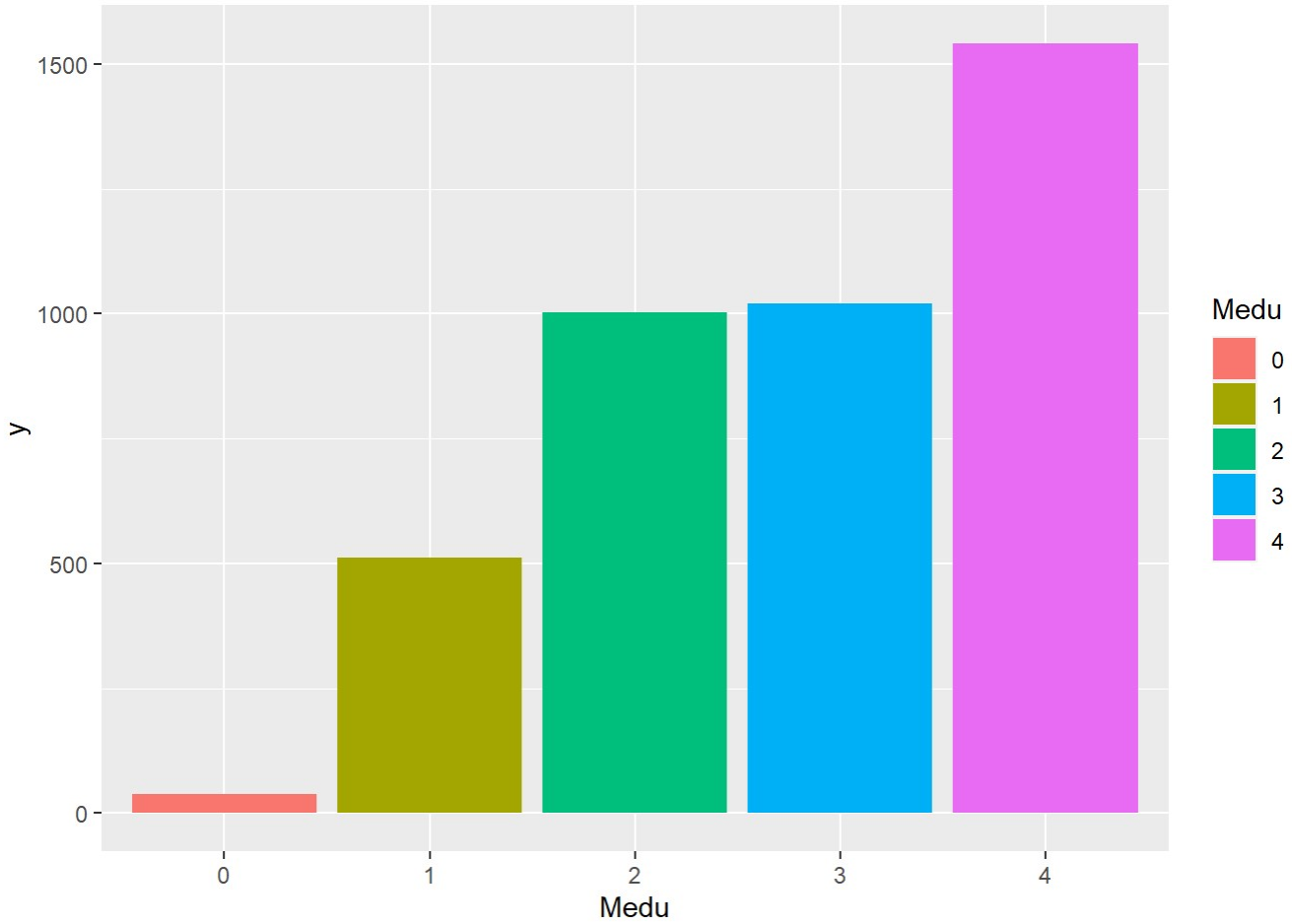
```
## $school
## [1] "GP" "MS"
##
## $sex
## [1] "F" "M"
##
## $age
## [1] 18 17 15 16 19 22 20 21
##
## $address
## [1] "U" "R"
##
## $famsize
## [1] "GT3" "LE3"
##
## $Pstatus
## [1] "A" "T"
##
## $Medu
## [1] 4 1 3 2 0
##
## $Fedu
## [1] 4 1 2 3 0
##
## $Mjob
## [1] "at_home"  "health"   "other"    "services" "teacher"
##
## $Fjob
## [1] "teacher"  "other"    "services" "health"   "at_home"
##
## $reason
## [1] "course"     "other"      "home"       "reputation"
##
## $guardian
## [1] "mother" "father" "other"
##
## $traveltime
## [1] 2 1 3 4
##
## $studytime
## [1] 2 3 1 4
##
## $failures
## [1] 0 3 2 1
##
## $schoolsup
## [1] "yes" "no"
##
## $famsup
## [1] "no"  "yes"
##
```

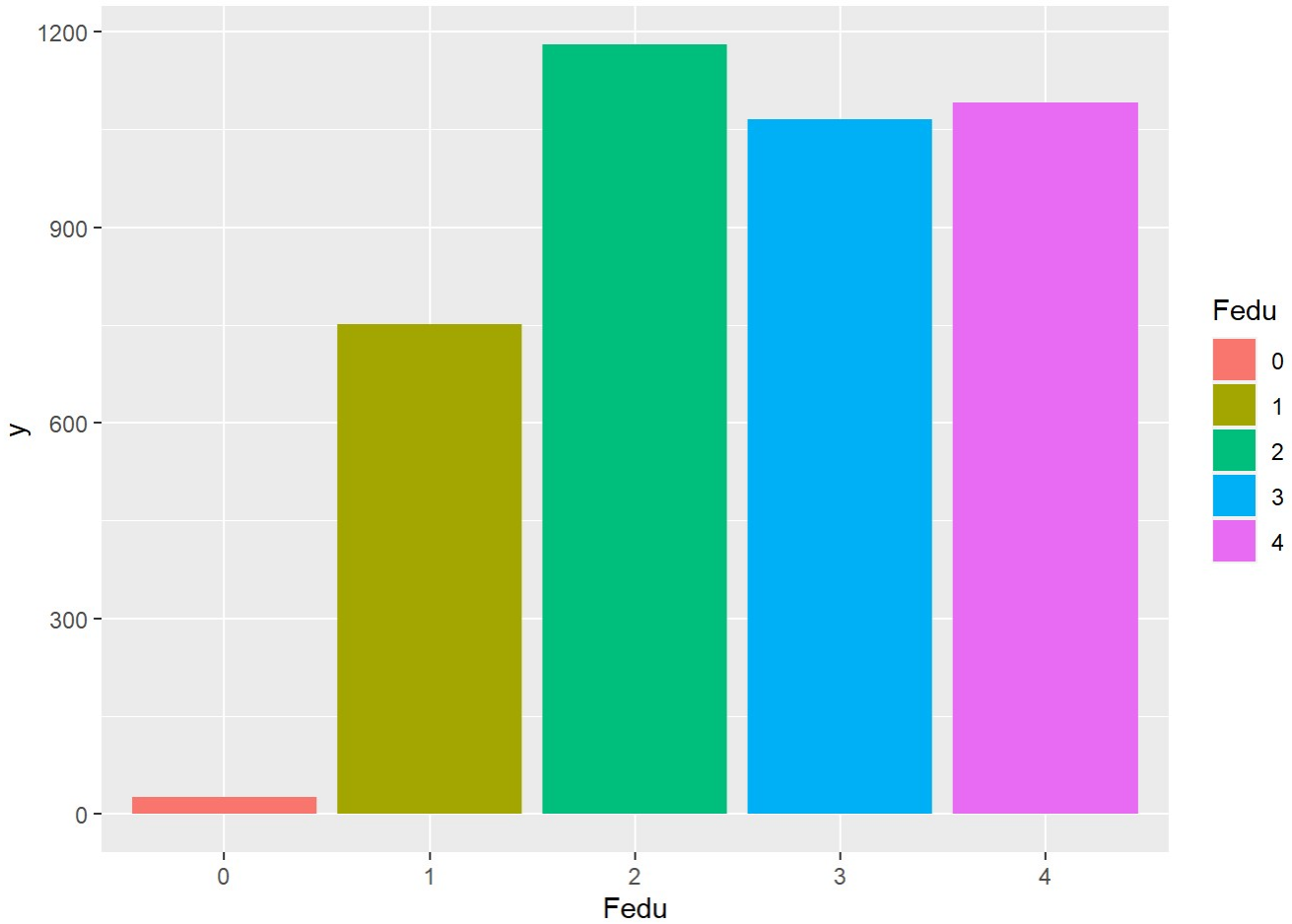
```
## $paid
## [1] "no"  "yes"
##
## $activities
## [1] "no"  "yes"
##
## $nursery
## [1] "yes" "no"
##
## $higher
## [1] "yes" "no"
##
## $internet
## [1] "no"  "yes"
##
## $romantic
## [1] "no"  "yes"
##
## $famrel
## [1] 4 5 3 1 2
##
## $freetime
## [1] 3 2 4 1 5
##
## $goout
## [1] 4 3 2 1 5
##
## $Dalc
## [1] 1 2 5 3 4
##
## $Walc
## [1] 1 3 2 4 5
##
## $health
## [1] 3 5 1 2 4
##
## $absences
##  [1]  6  4 10  2  0 16 14  7  8 25 12 54 18 26 20 56 24 28  5 13 15 22  3 21  1
## [26] 75 30 19  9 11 38 40 23 17
##
## $G1
##  [1]  5  7 15  6 12 16 14 10 13  8 11  9 17 19 18  4  3
##
## $G2
##  [1]  6  5  8 14 10 15 12 18 16 13  9 11  7 19 17  4  0
##
## $G3
##  [1]  6 10 15 11 19  9 12 14 16  5  8 17 18 13 20  7  0  4
```

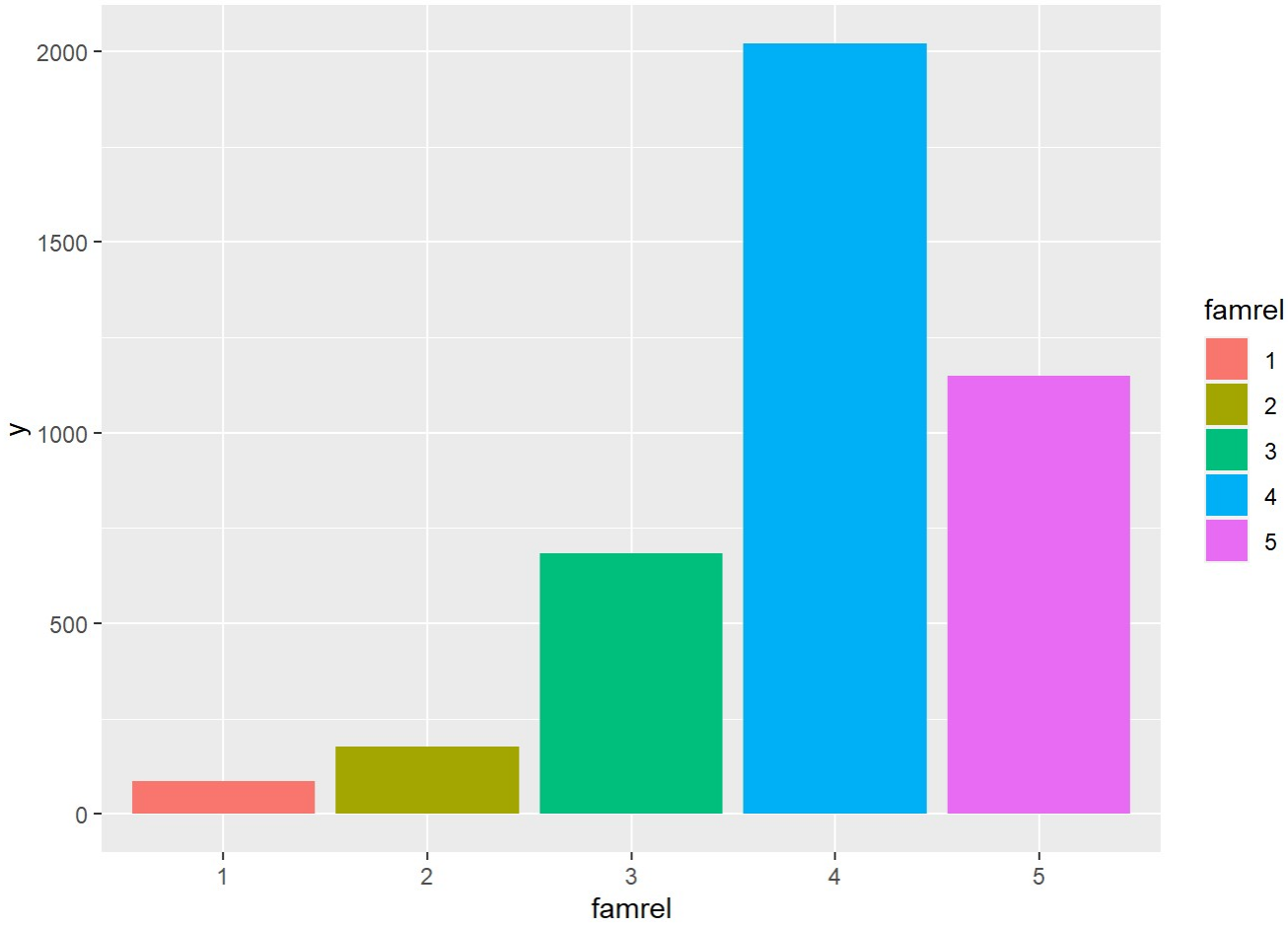
```
#Converting categorical variables to factor
names <- c(1:2,4:12,16:29)
df[,names] <- lapply(df[,names] , factor)
str(df)
```

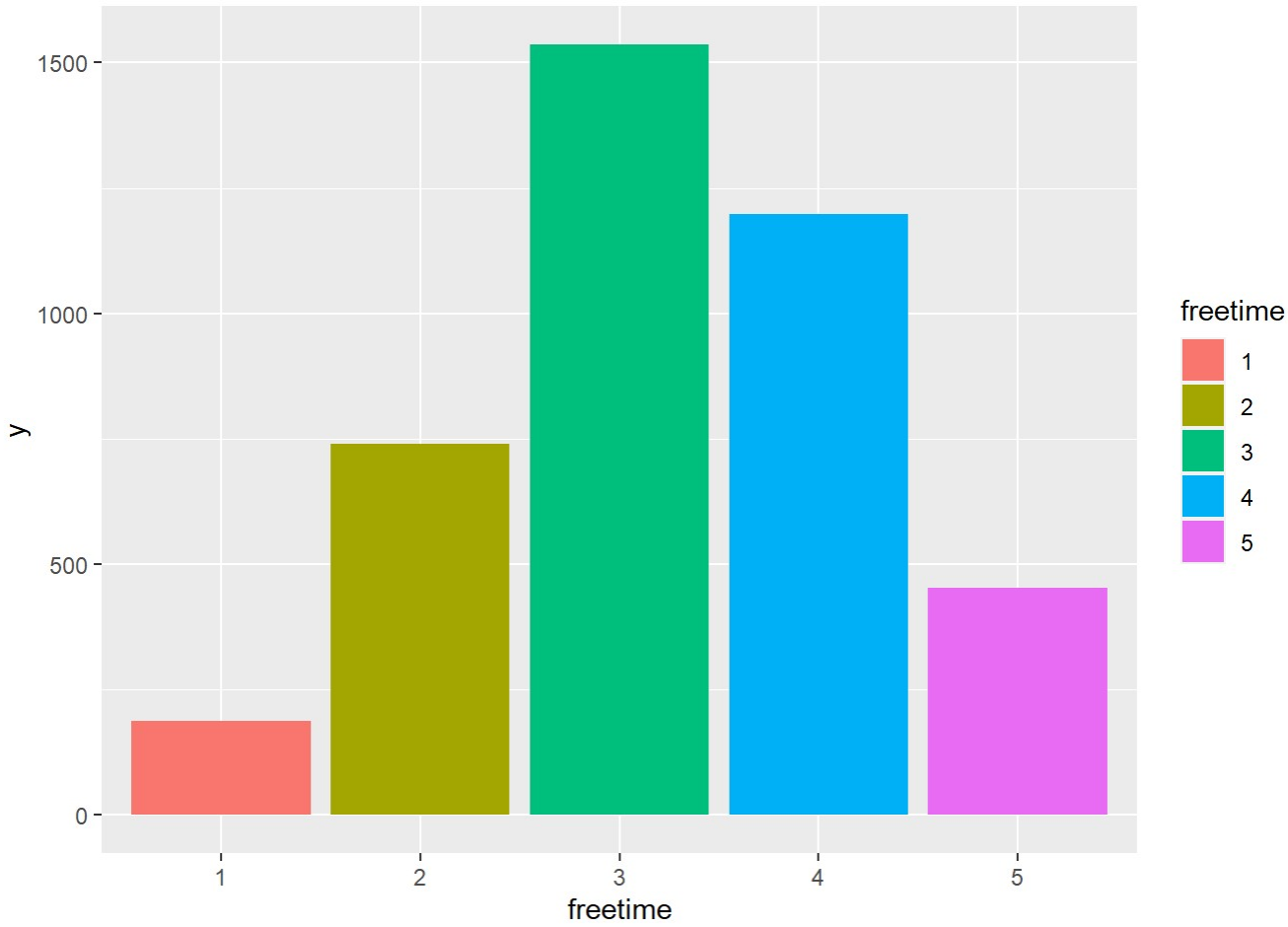
```
## 'data.frame':    395 obs. of  33 variables:
##  $ school    : Factor w/ 2 levels "GP","MS": 1 1 1 1 1 1 1 1 1 1 ...
##  $ sex       : Factor w/ 2 levels "F","M": 1 1 1 1 1 2 2 1 2 2 ...
##  $ age       : int  18 17 15 15 16 16 16 17 15 15 ...
##  $ address   : Factor w/ 2 levels "R","U": 2 2 2 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      : Factor w/ 5 levels "0","1","2","3",..: 5 2 2 5 4 5 3 5 4 4 ...
##  $ Fedu      : Factor w/ 5 levels "0","1","2","3",..: 5 2 2 3 4 4 3 5 3 5 ...
##  $ 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 : 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    : Factor w/ 5 levels "1","2","3","4",..: 4 5 4 3 4 5 4 4 4 5 ...
##  $ freetime  : Factor w/ 5 levels "1","2","3","4",..: 3 3 3 2 3 4 4 1 2 5 ...
##  $ goout     : Factor w/ 5 levels "1","2","3","4",..: 4 3 2 2 2 2 4 4 2 1 ...
##  $ Dalc      : Factor w/ 5 levels "1","2","3","4",..: 1 1 2 1 1 1 1 1 1 1 ...
##  $ Walc      : Factor w/ 5 levels "1","2","3","4",..: 1 1 3 1 2 2 1 1 1 1 ...
##  $ health    : Factor w/ 5 levels "1","2","3","4",..: 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 ...
```

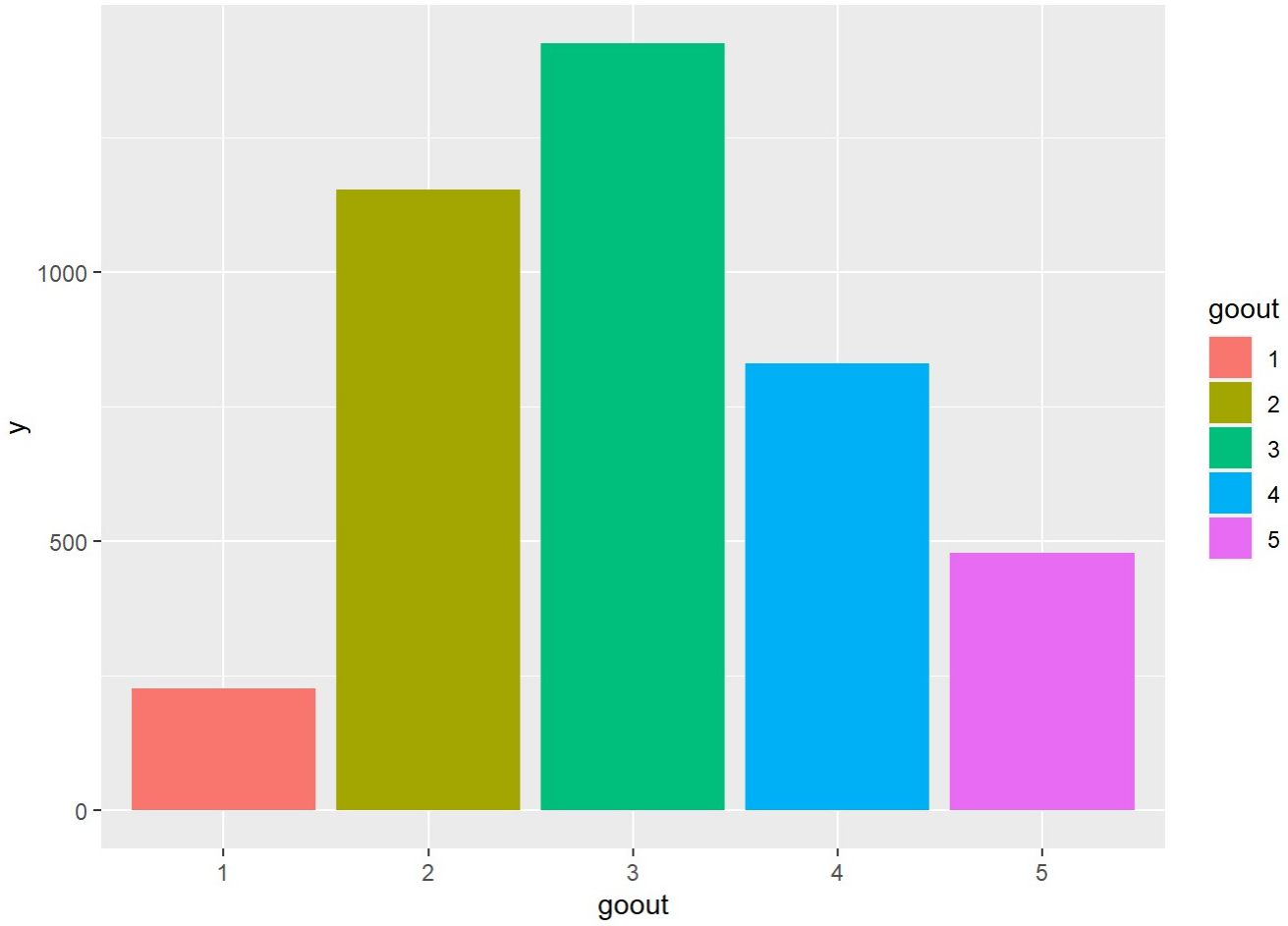
```
plots <- list()
p_names = c("Medu","Fedu","famrel","freetime","goout","Dalc","Walc","health","school","sex","
address","famsize","Pstatus","Mjob","Fjob","reason","guardian","schoolsup","famsup","paid","a
ctivities","nursery","higher","internet","romantic")
for(nm in p_names) {
  plots[[nm]] <- ggplot(data= df,aes_string(x =nm,y=G3,fill=nm))+geom_bar(stat = "identity")
  print(plots[[nm]])
}
```

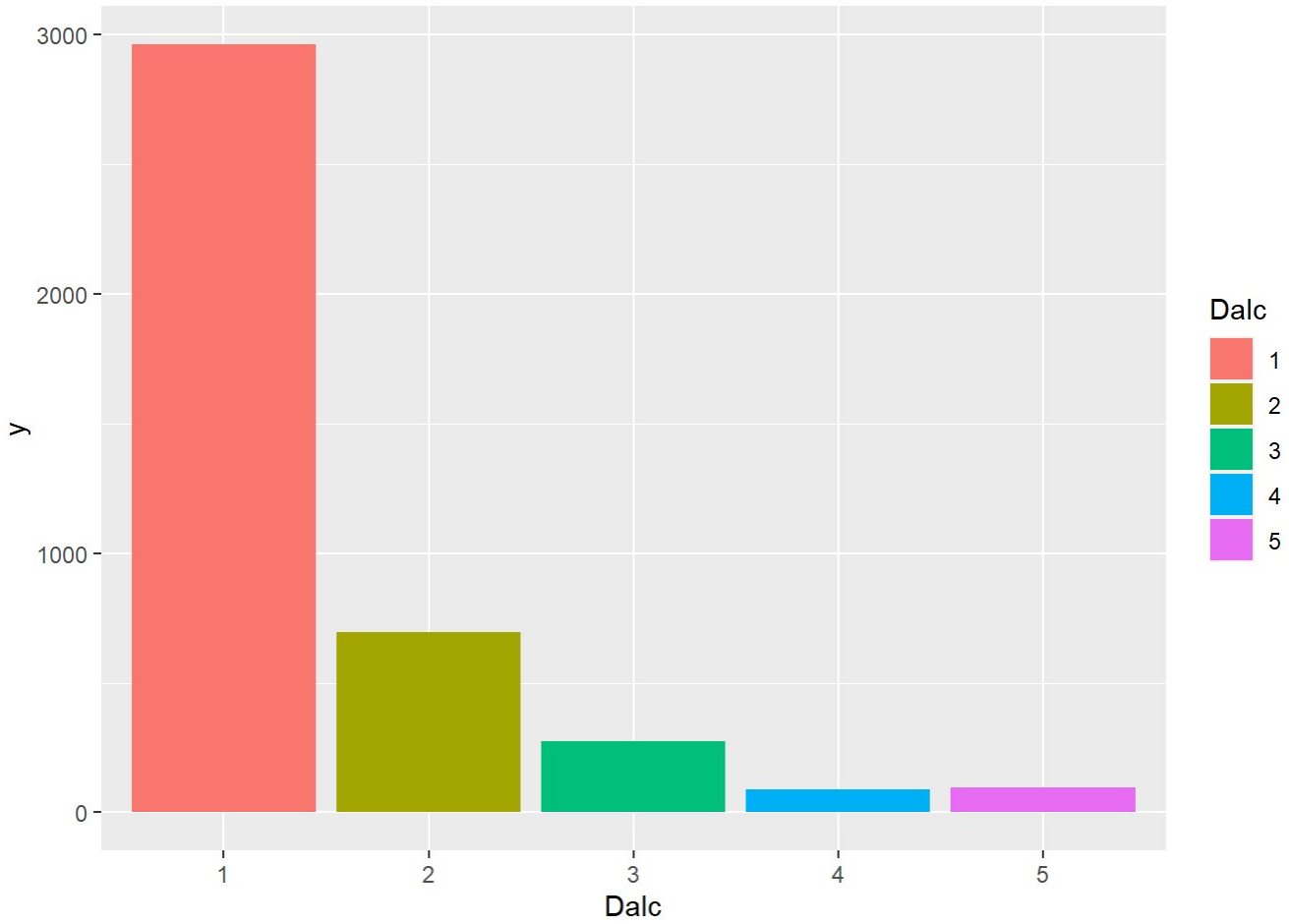

1500
1000
500
0
y
0
1
2
3
4
Medu
Medu
0
1
2
3
4

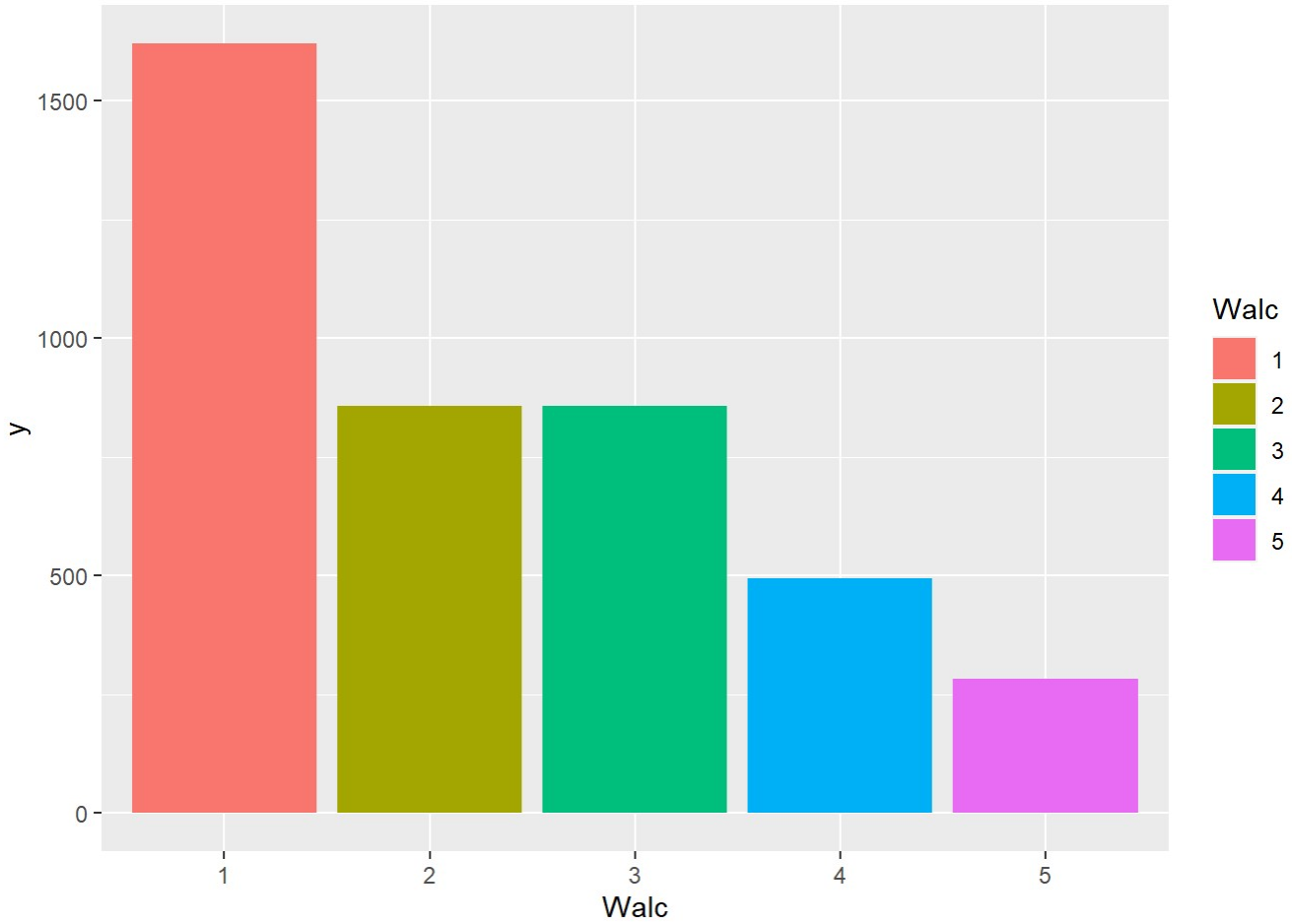

1200
900
600
300
0
y
0
1
2
3
4
Fedu
Fedu
0
1
2
3
4

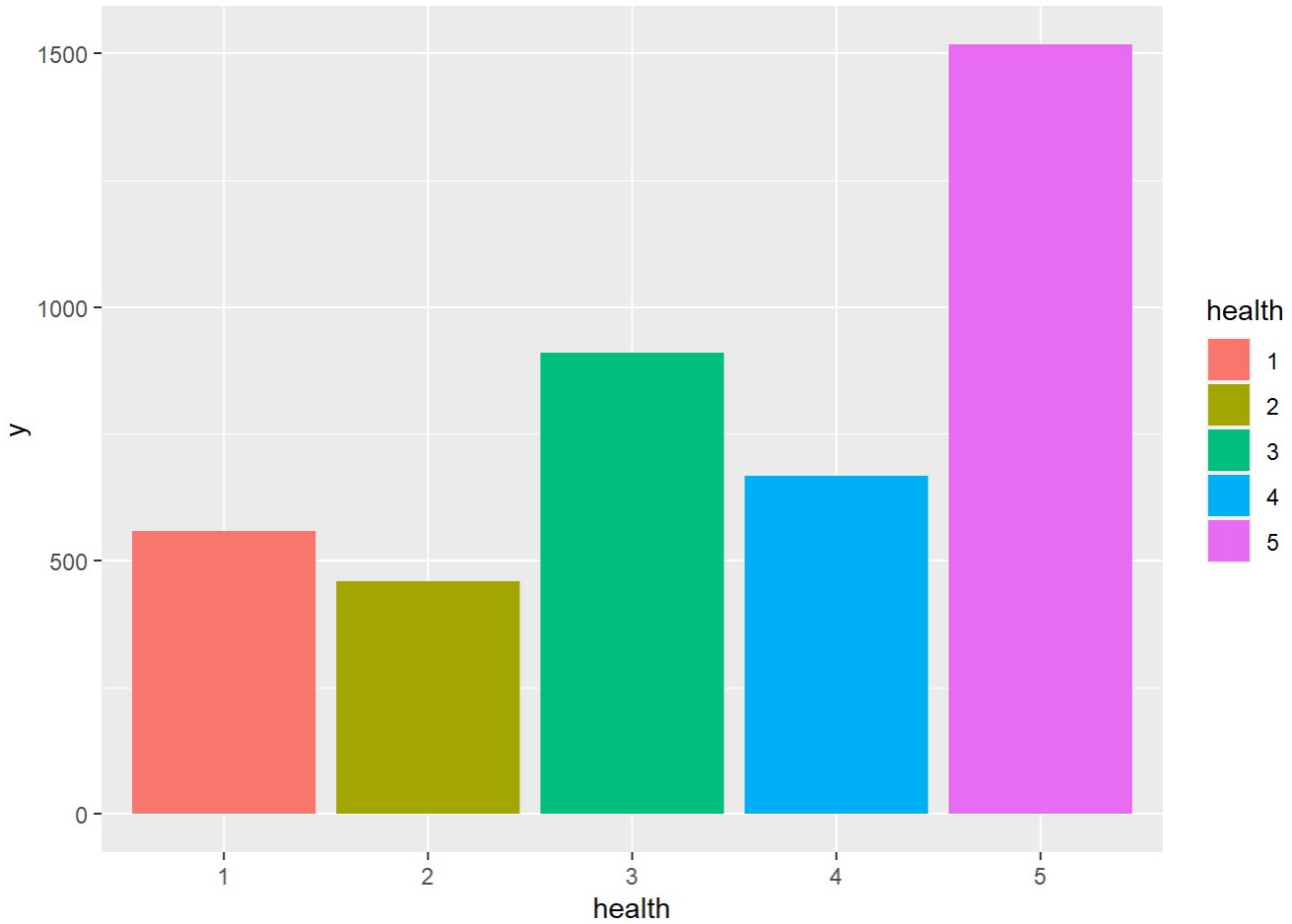

2000
1500
1000
500
0
y
1
2
3
4
5
famrel
famrel
1
2
3
4
5

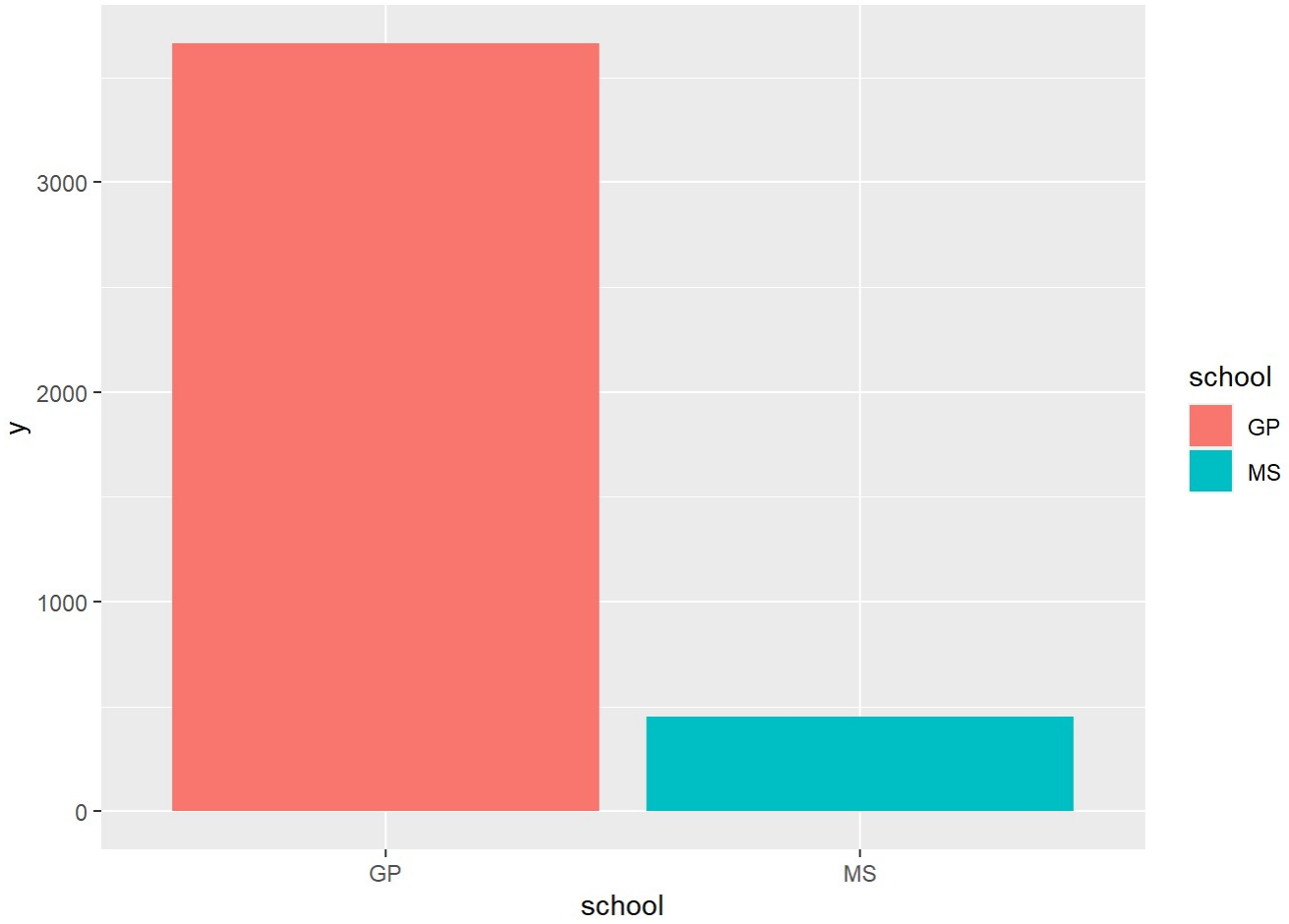

1500
1000
500
0
y
1
2
3
4
5
freetime
freetime
1
2
3
4
5

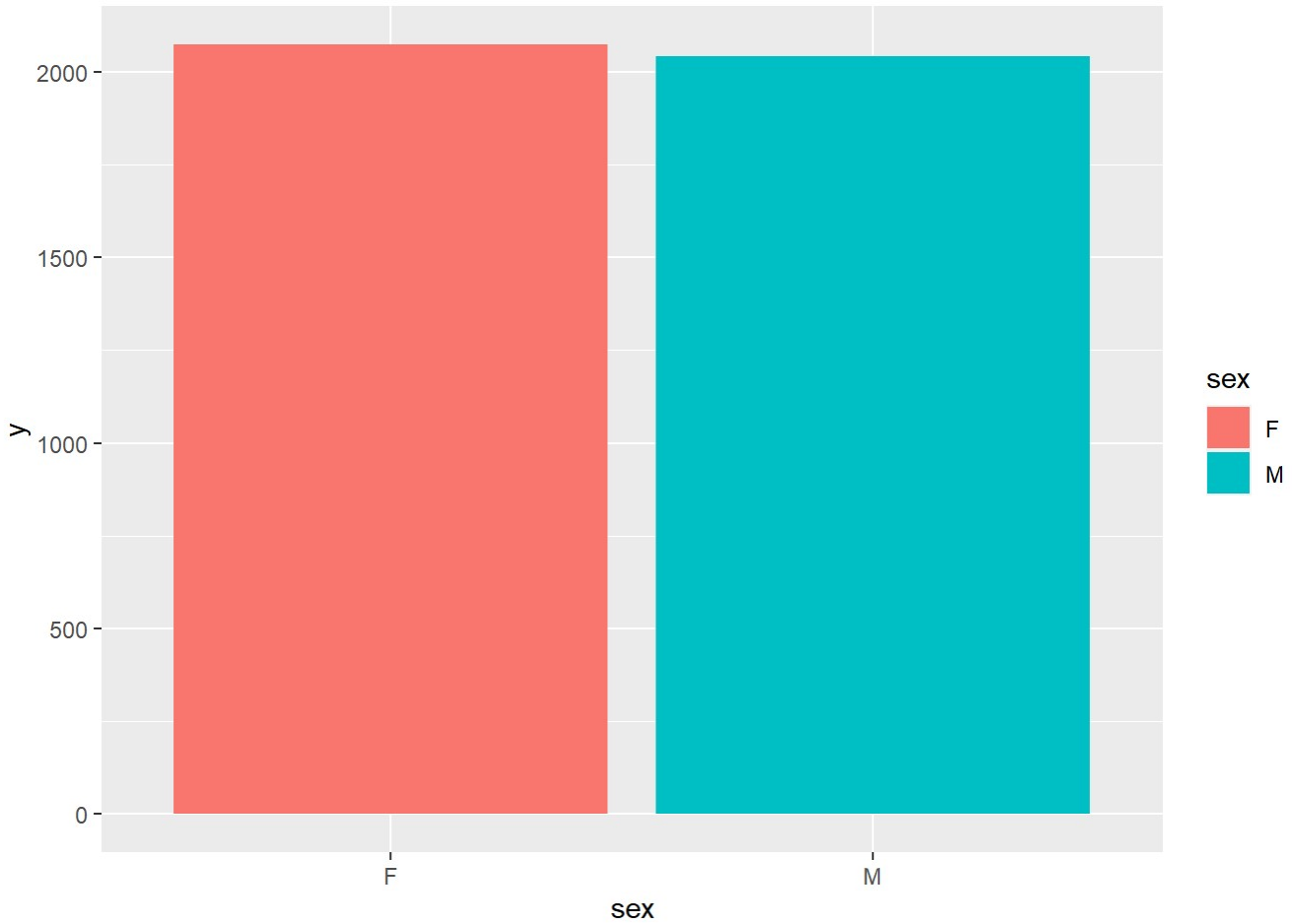

1000
500
0
y
1
2
3
4
5
goout
goout
1
2
3
4
5

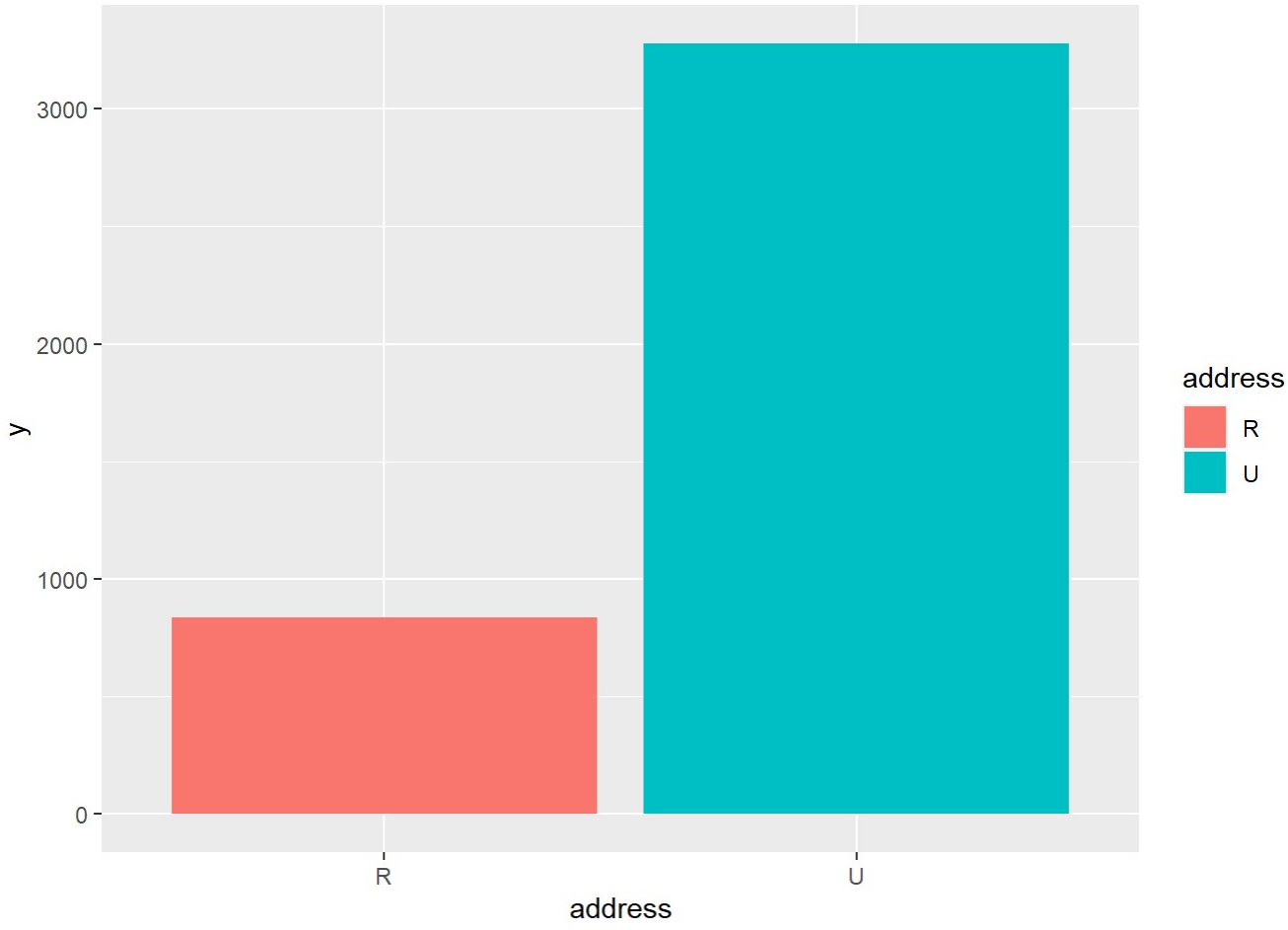

3000
2000
1000
0
y
1
2
3
4
5
Dalc
Dalc
1
2
3
4
5

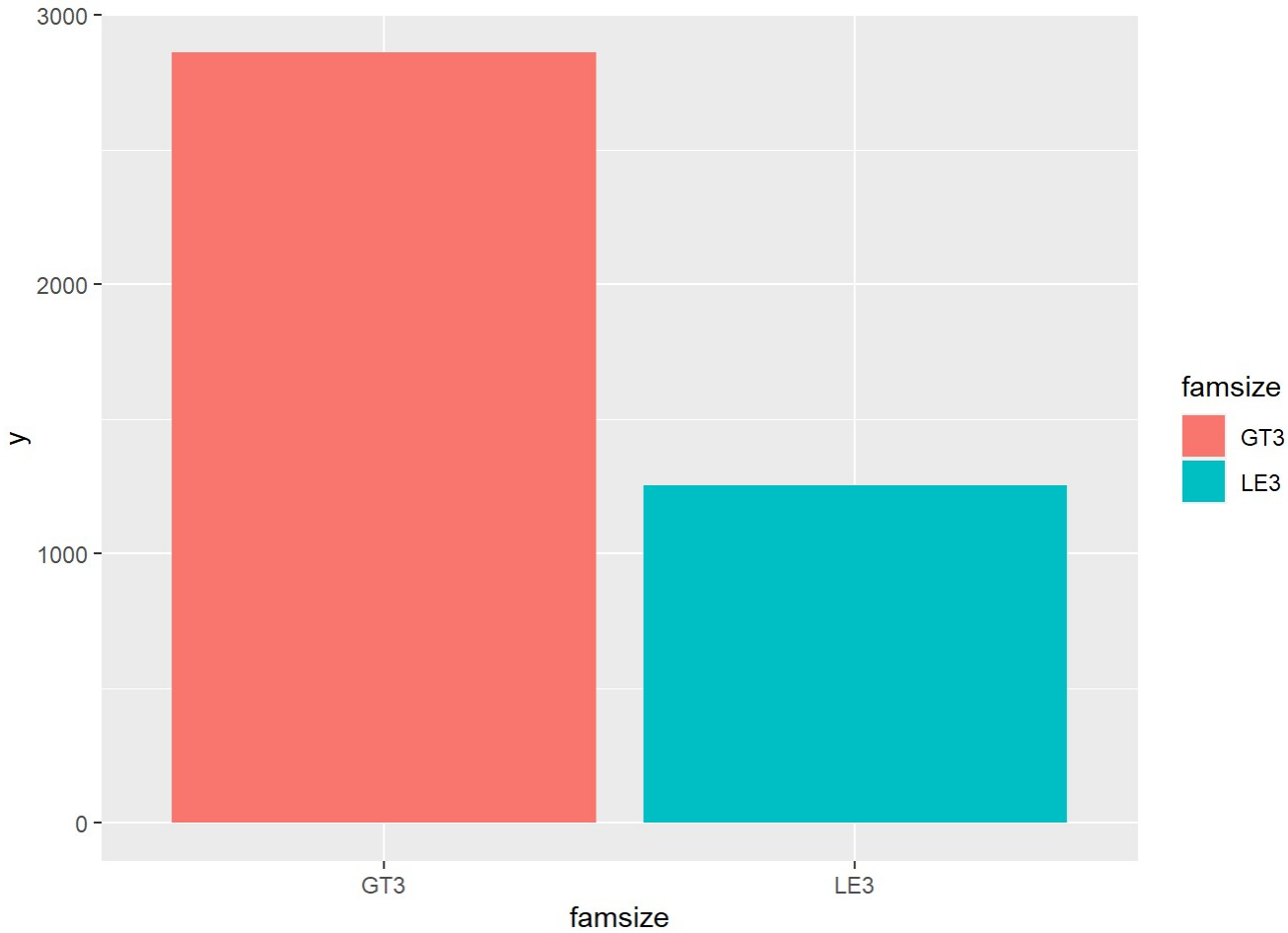

1500
1000
500
0
y
1
2
3
4
5
Walc
Walc
1
2
3
4
5

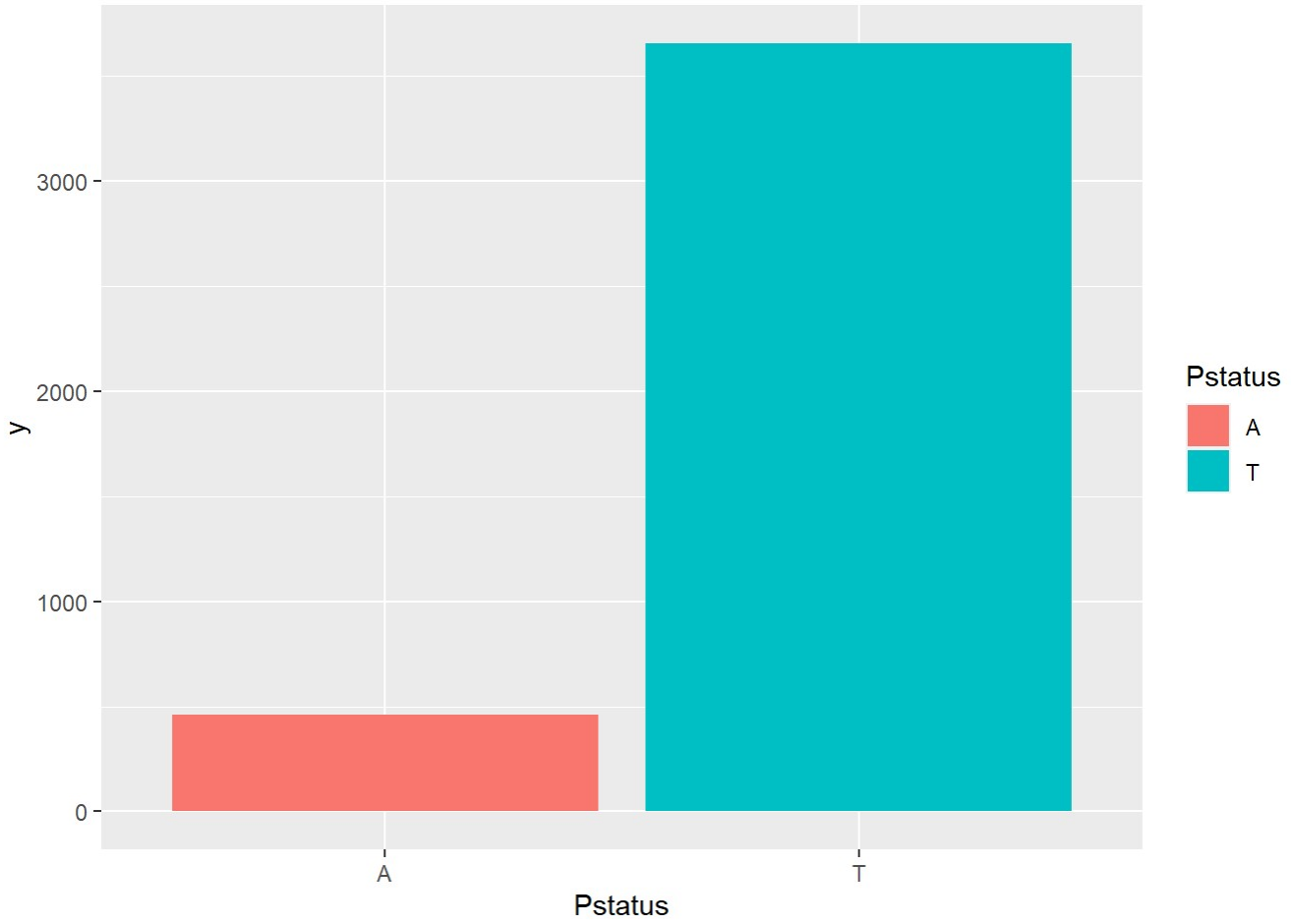

1500
1000
500
0
y
1
2
3
4
5
health
health
1
2
3
4
5

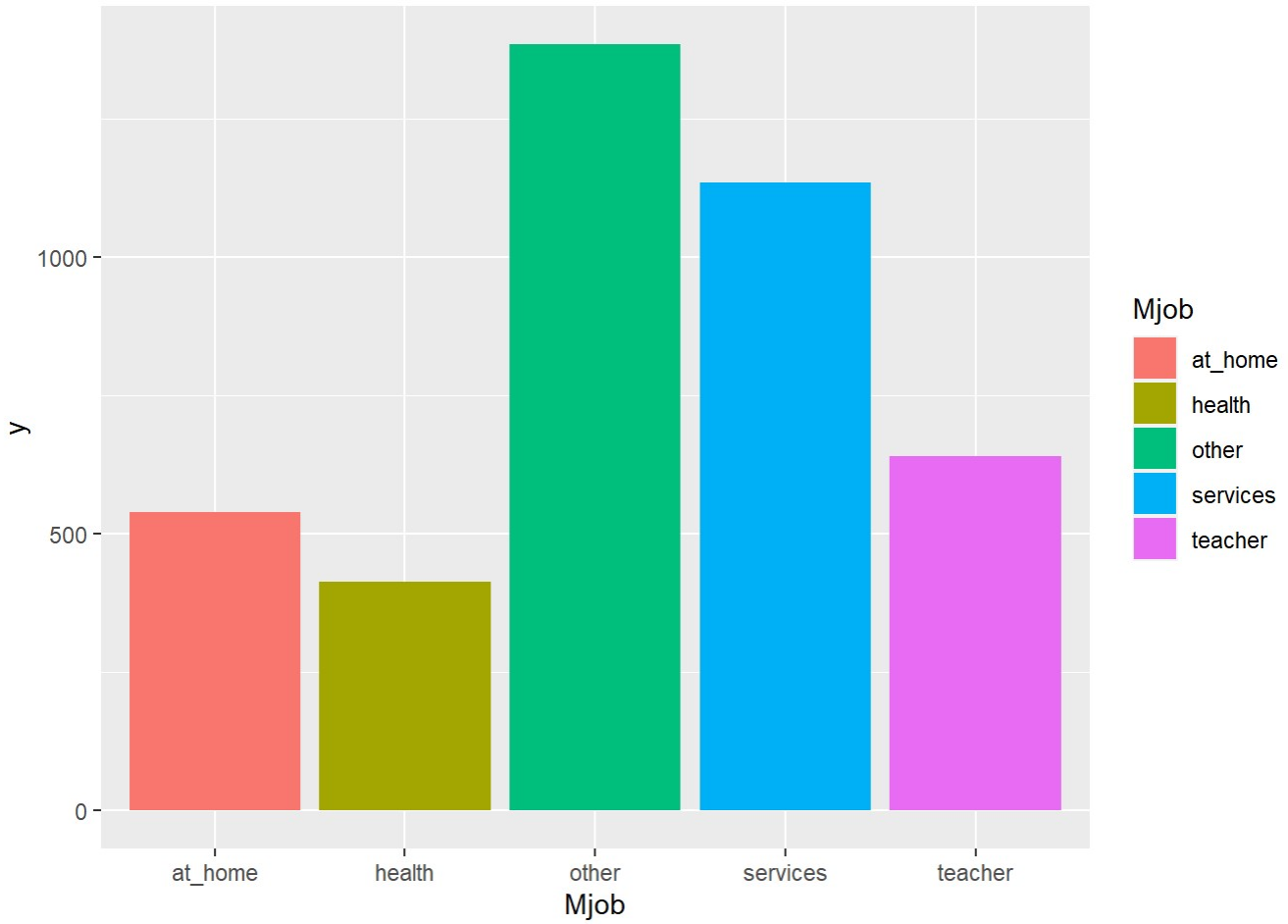

3000
2000
1000
0
y
GP
MS
school
school
GP
MS

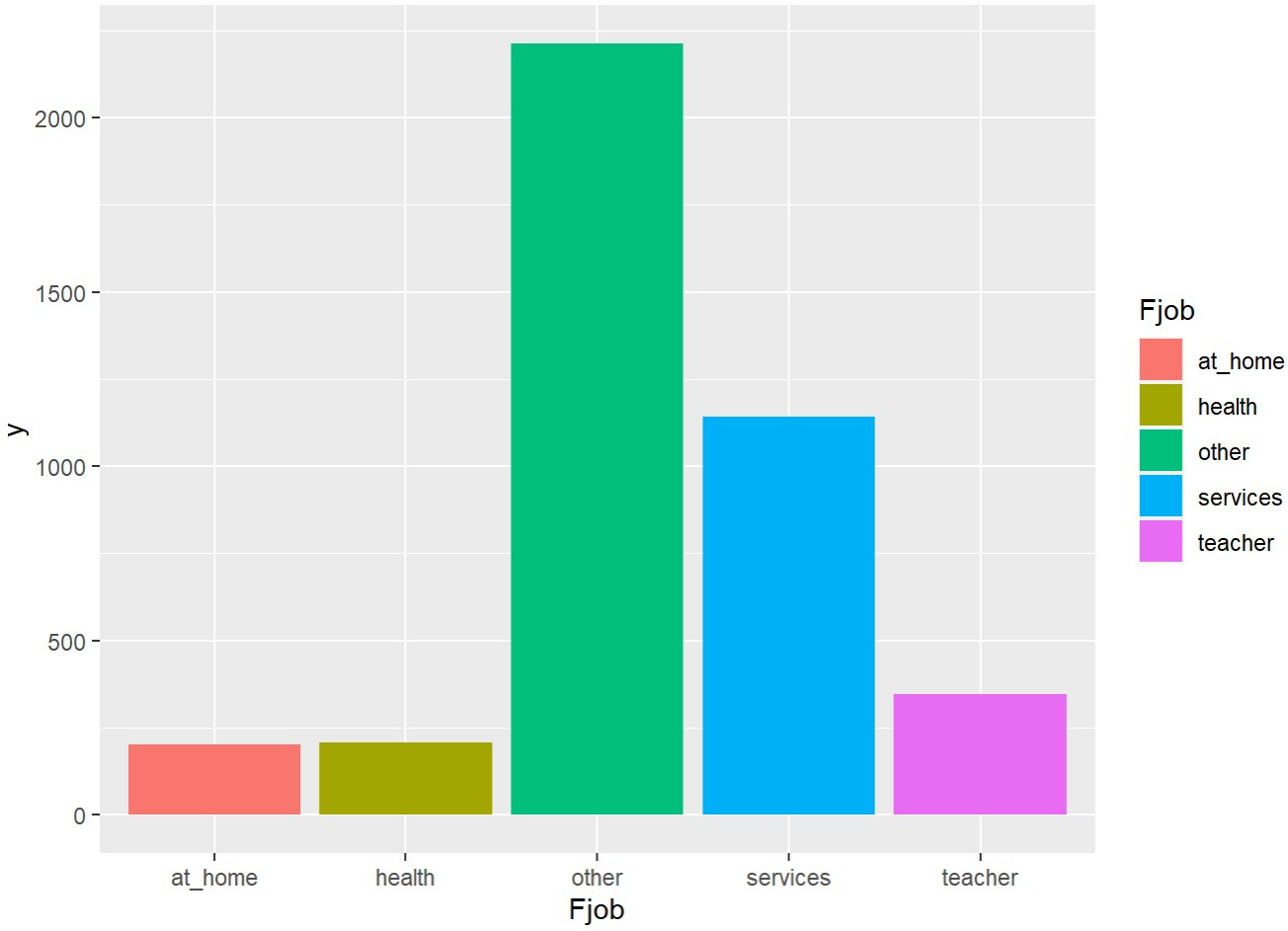

2000
1500
1000
500
0
y
F
M
sex
sex
F
M

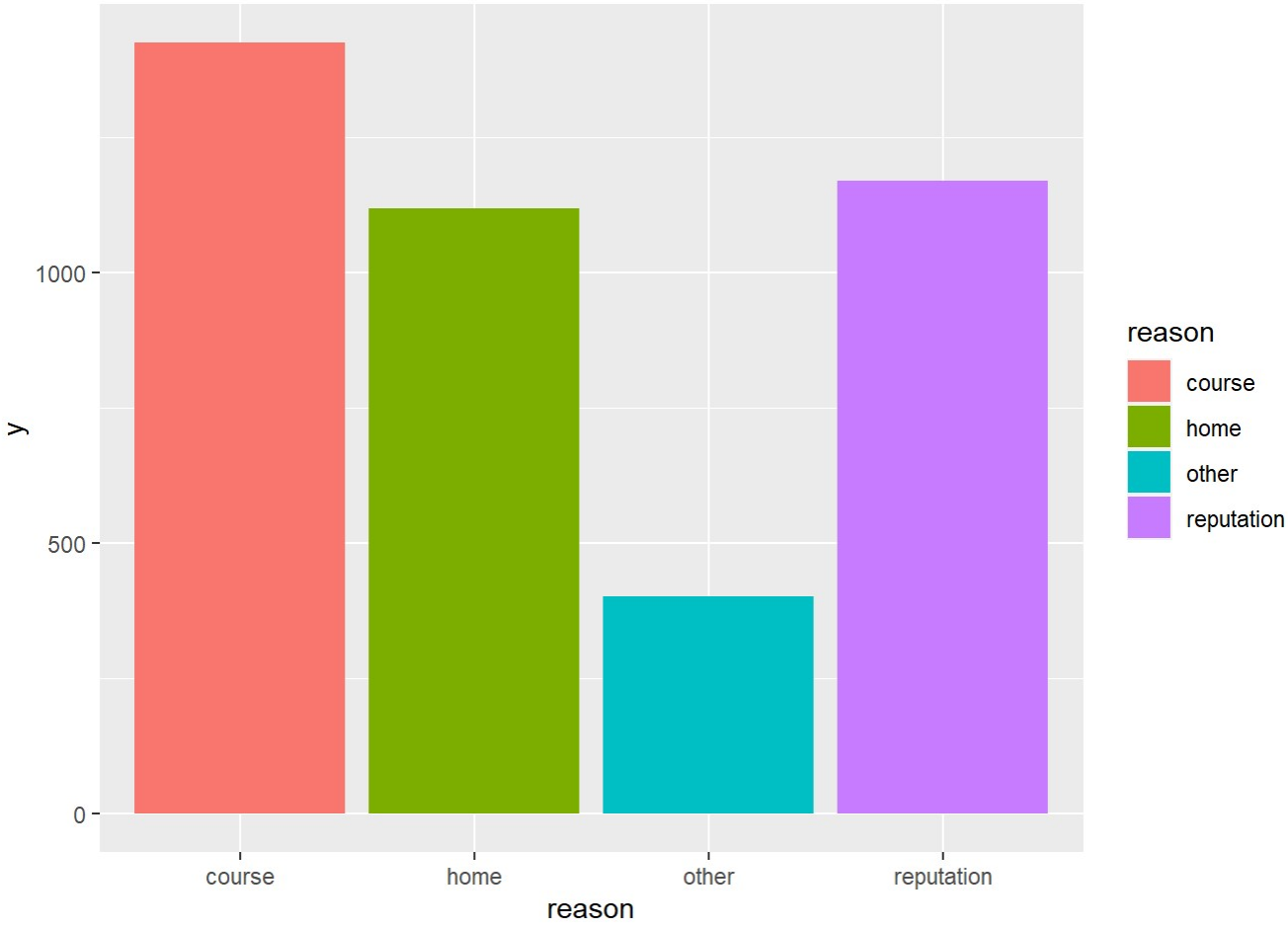

3000
2000
1000
0
y
R
U
address
address
R
U

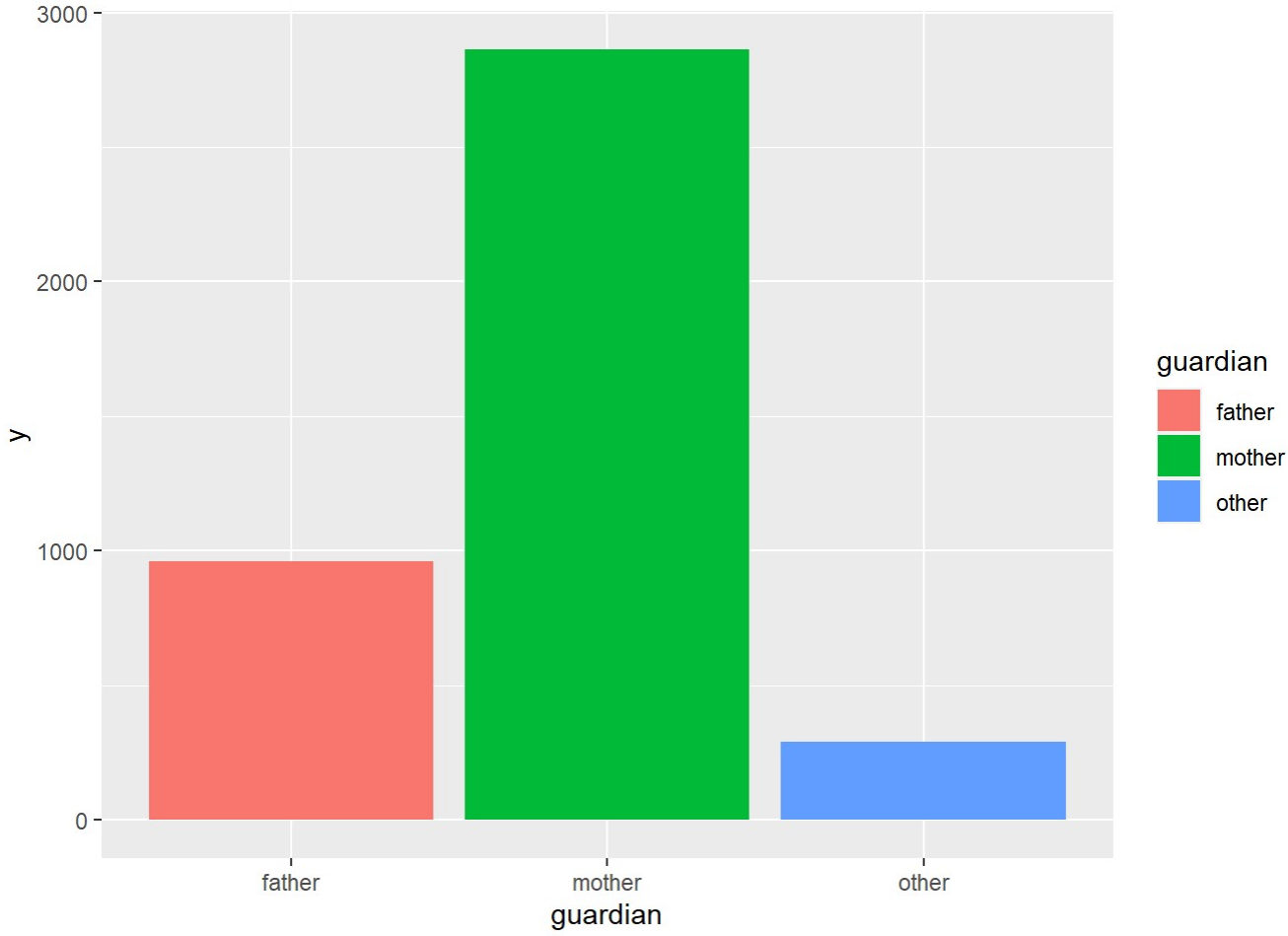

3000
2000
1000
0
y
GT3
LE3
famsize
famsize
GT3
LE3

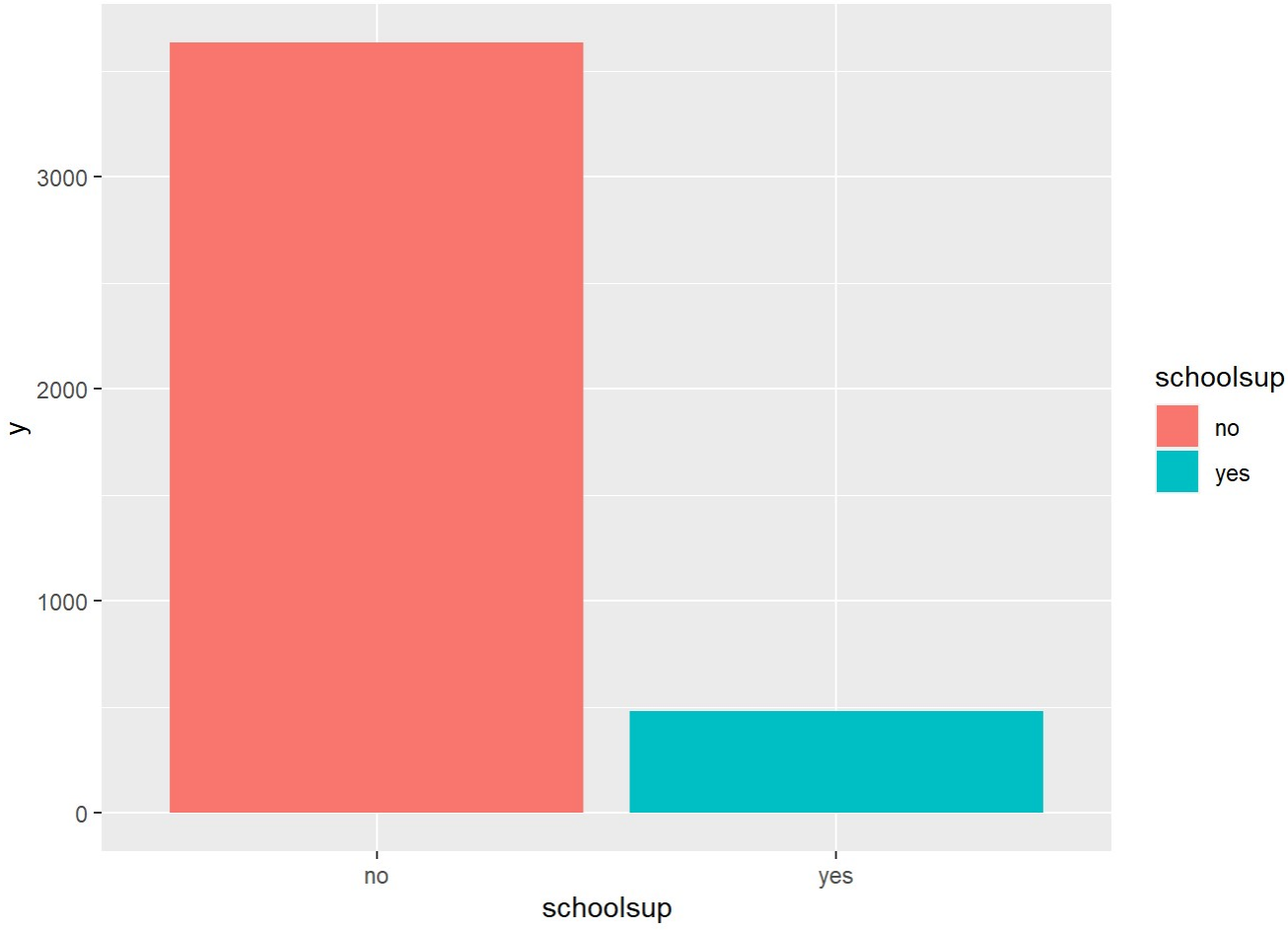

3000
2000
1000
0
y
A
T
Pstatus
Pstatus
A
T

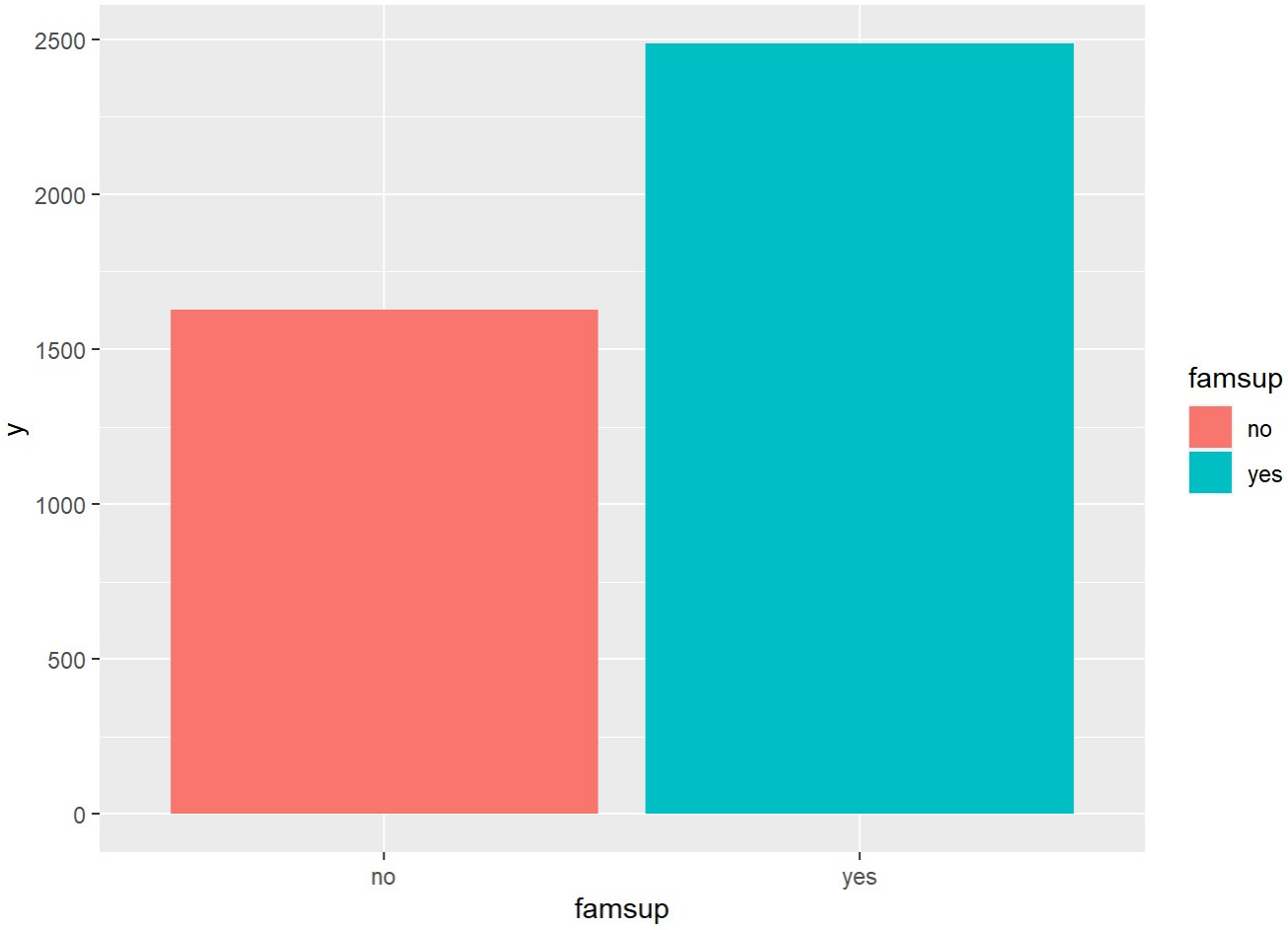

1000
500
0
y
at_home
health
other
services
teacher
Mjob
Mjob
at_home
health
other
services
teacher

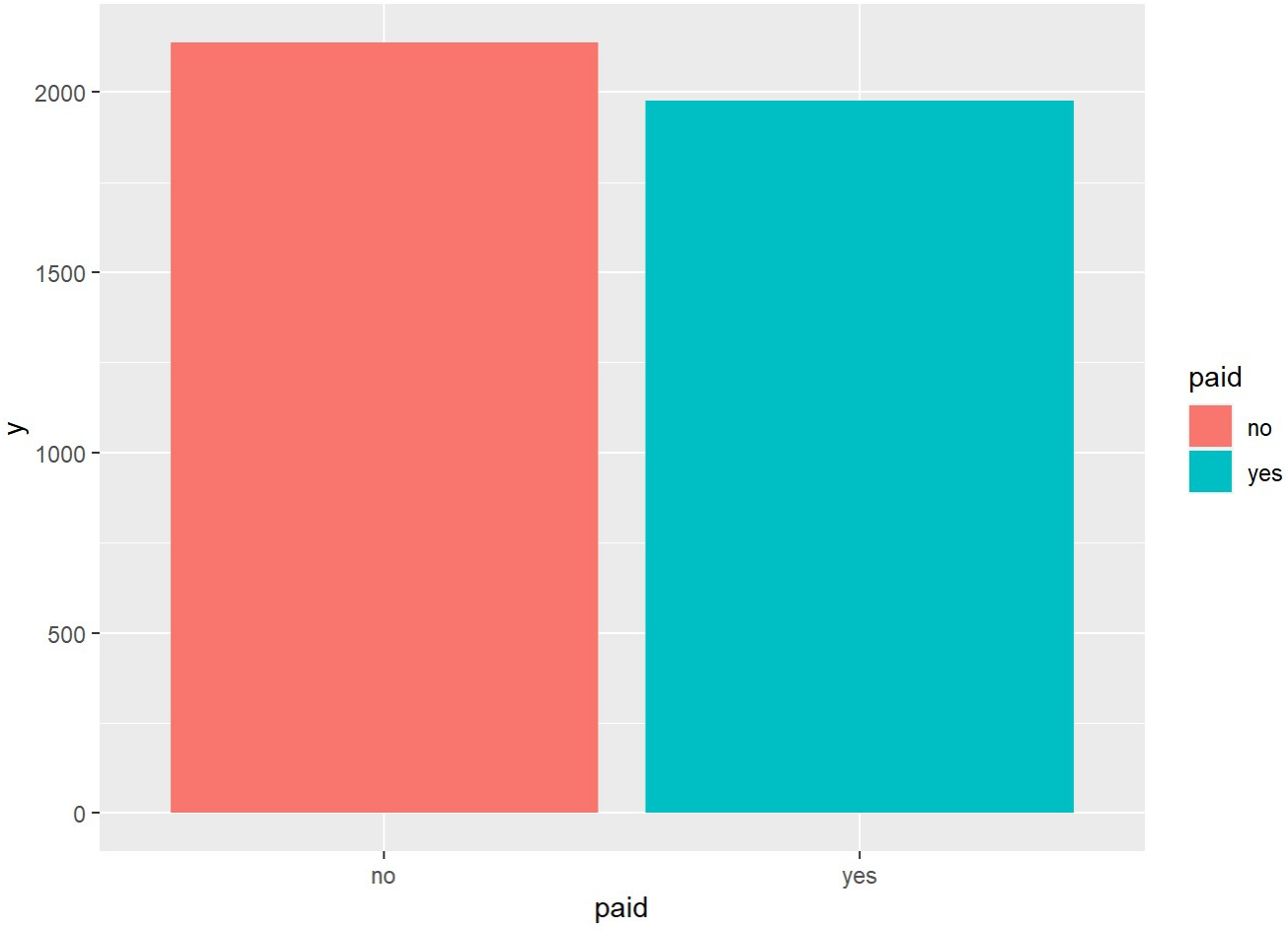

2000
1500
1000
500
0
y
at_home
health
other
services
teacher
Fjob
Fjob
at_home
health
other
services
teacher

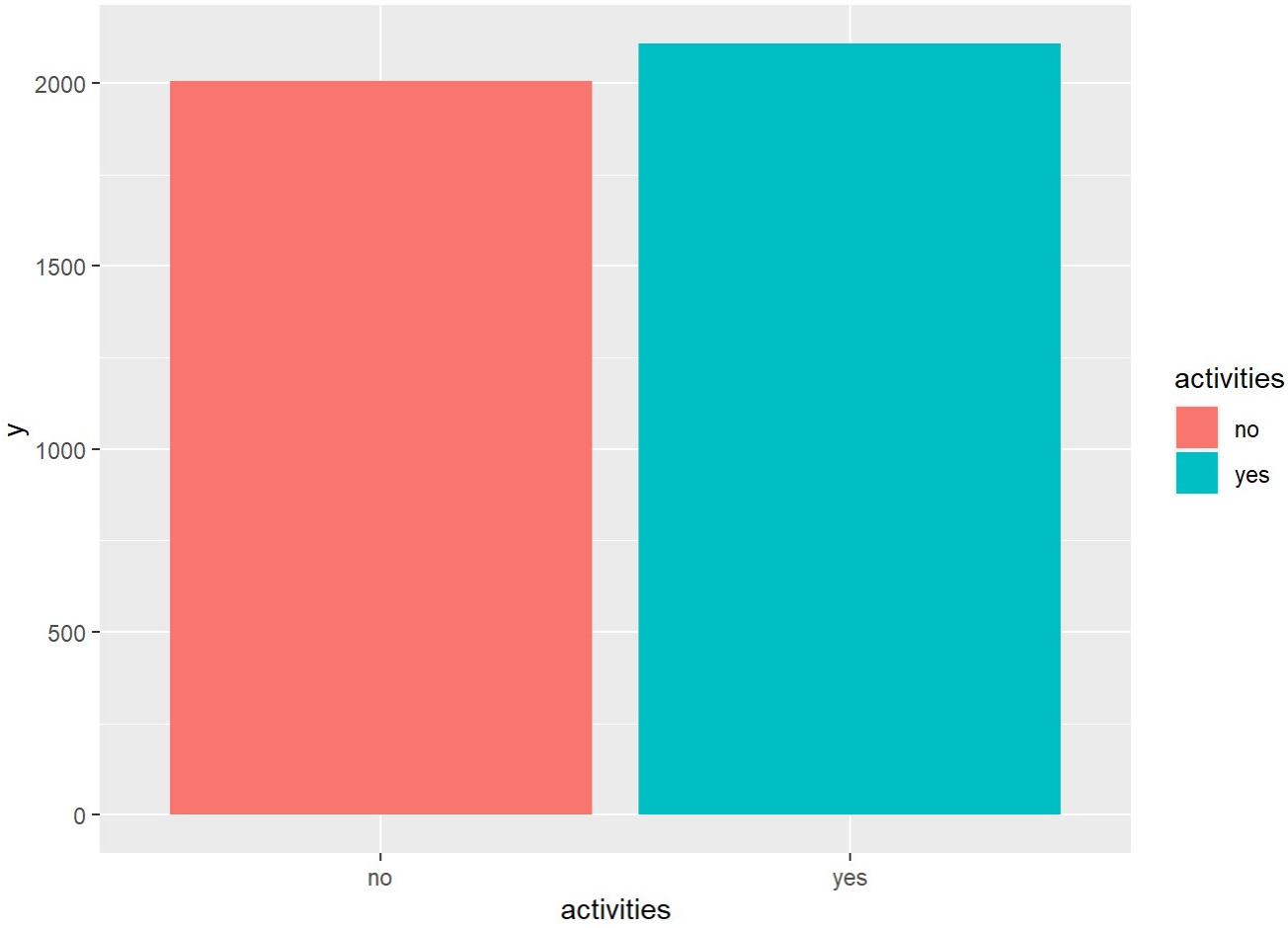

1000
500
0
y
course
home
other
reputation
reason
reason
course
home
other
reputation

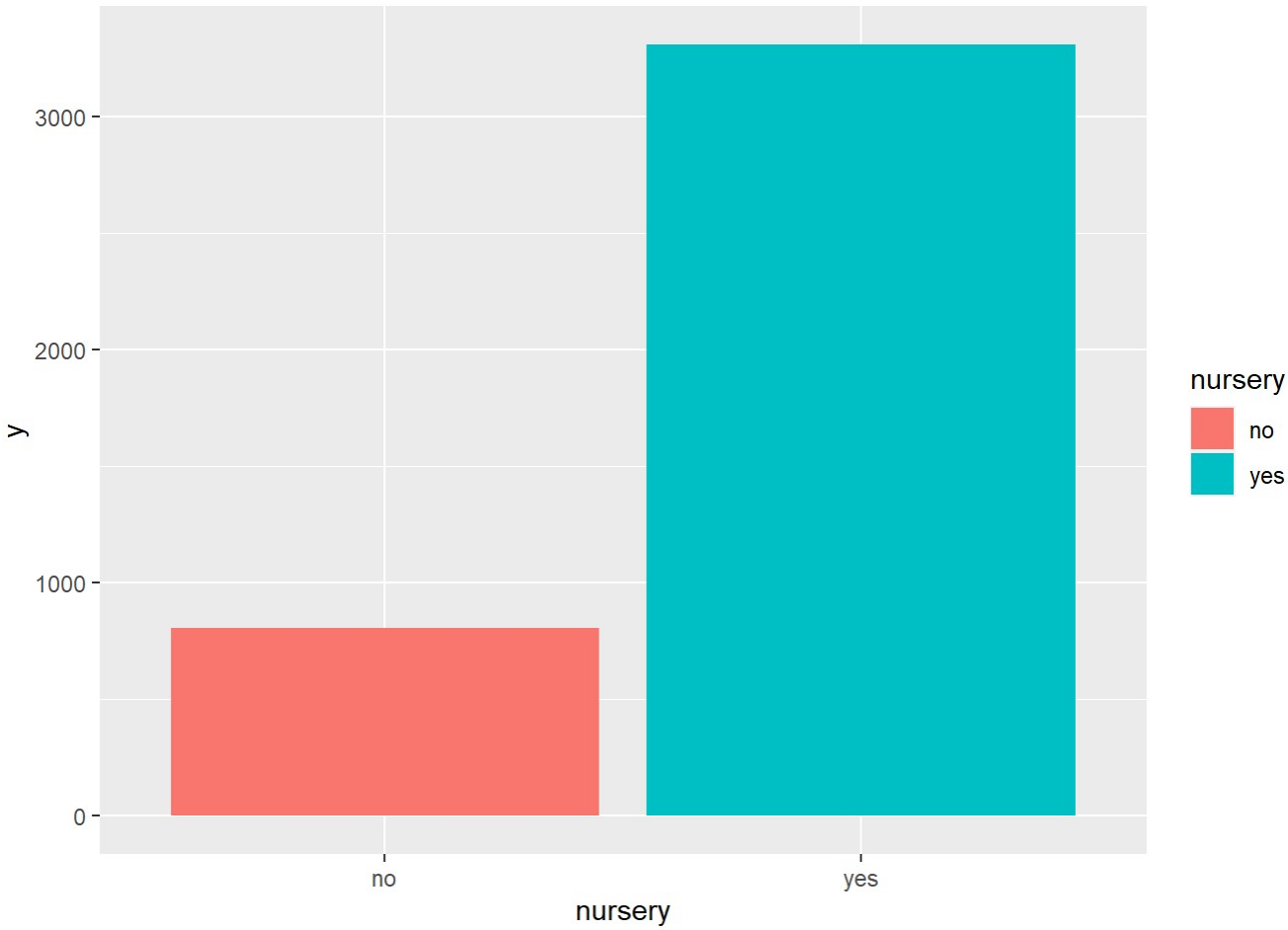

3000
2000
1000
0
y
father
mother
other
guardian
guardian
father
mother
other

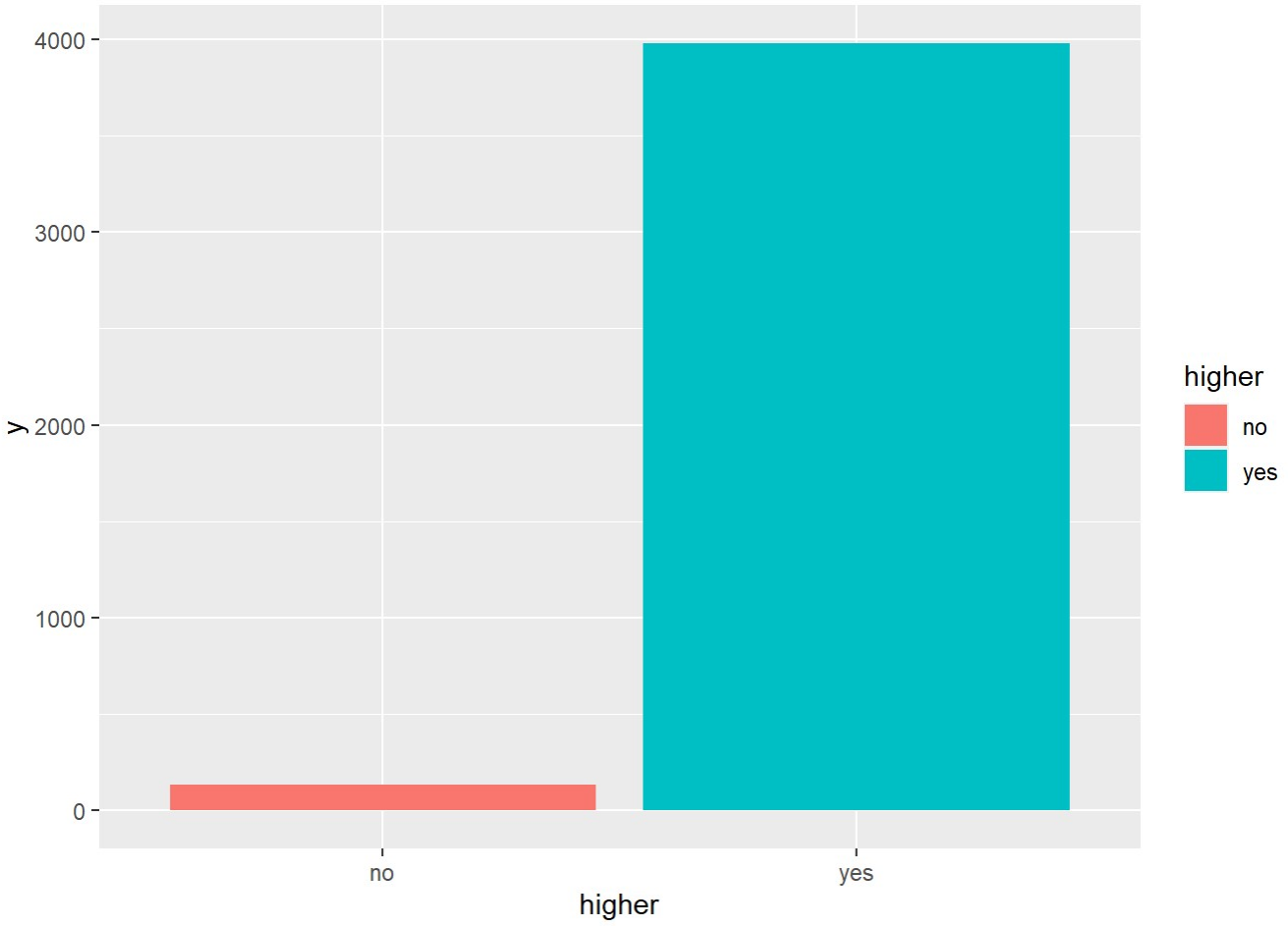

3000
2000
1000
0
y
no
yes
schoolsup
schoolsup
no
yes

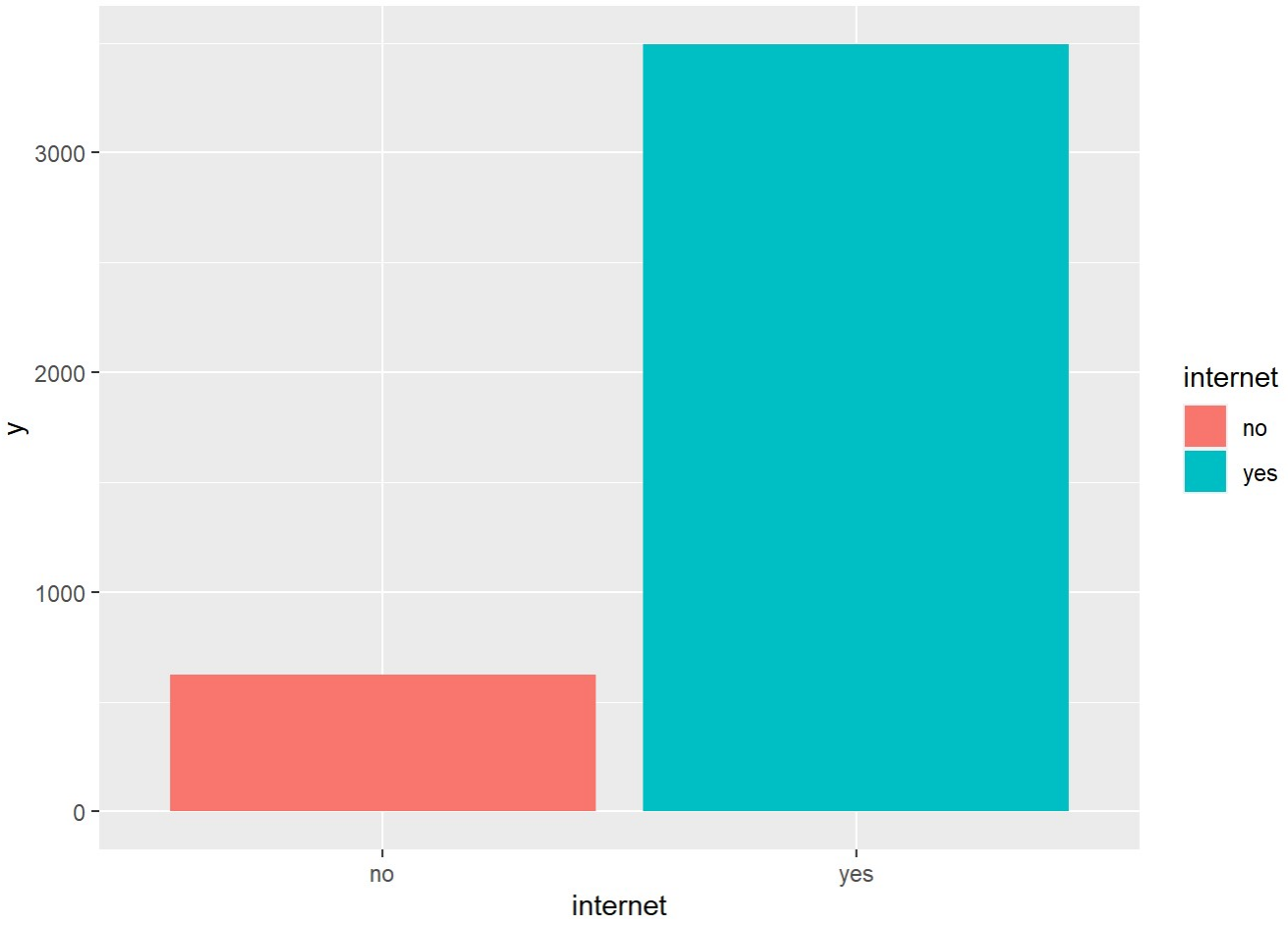

2500
2000
1500
1000
500
0
y
no
yes
famsup
famsup
no
yes

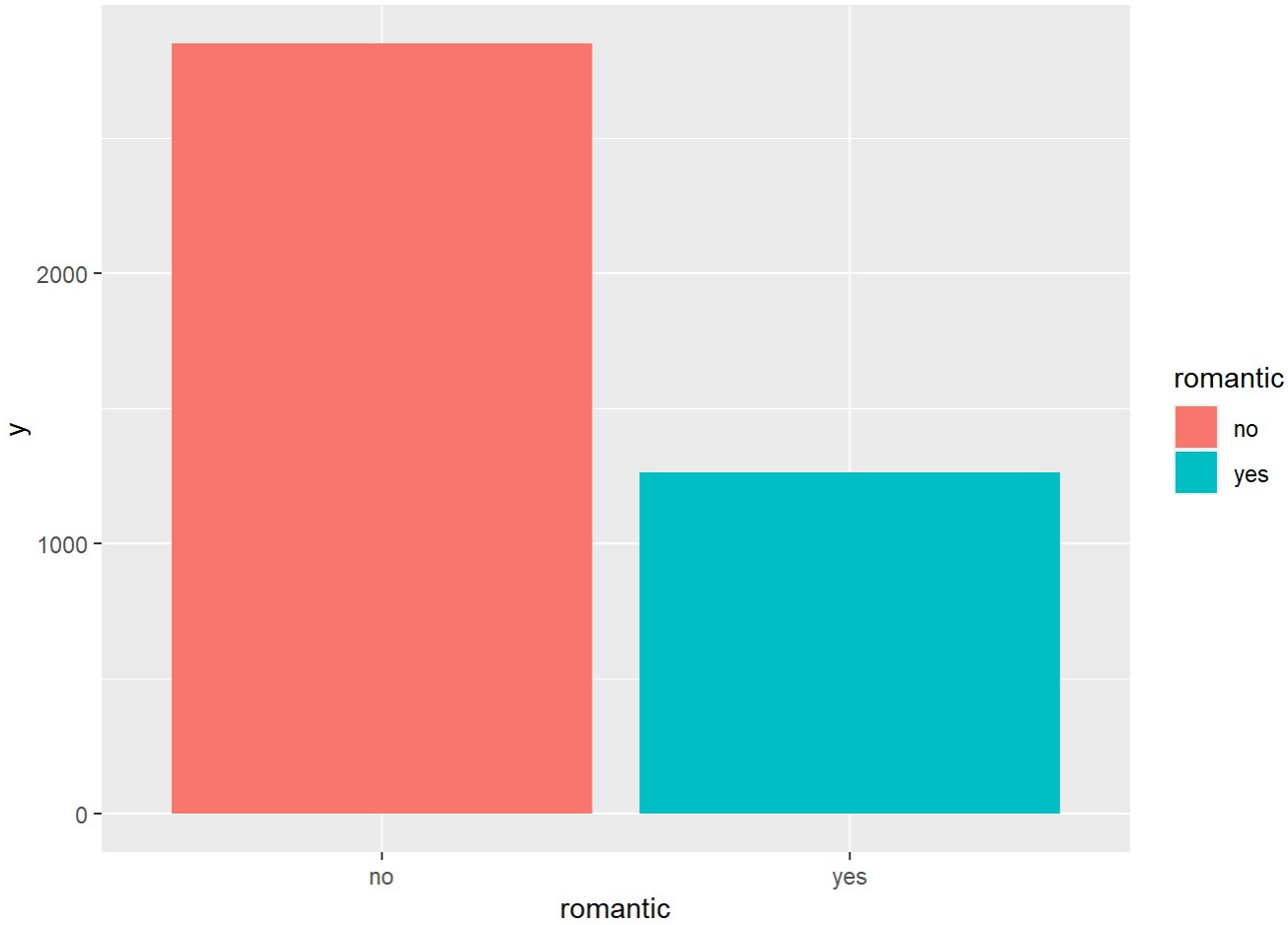

2000
1500
1000
500
0
y
no
yes
paid
paid
no
yes


2000
1500
1000
500
0
y
no
yes
activities
activities
no
yes


3000
2000
1000
0
y
no
yes
nursery
nursery
no
yes


4000
3000
2000
1000
0
y
no
yes
higher
higher
no
yes


3000
2000
1000
0
y
no
yes
internet
internet
no
yes



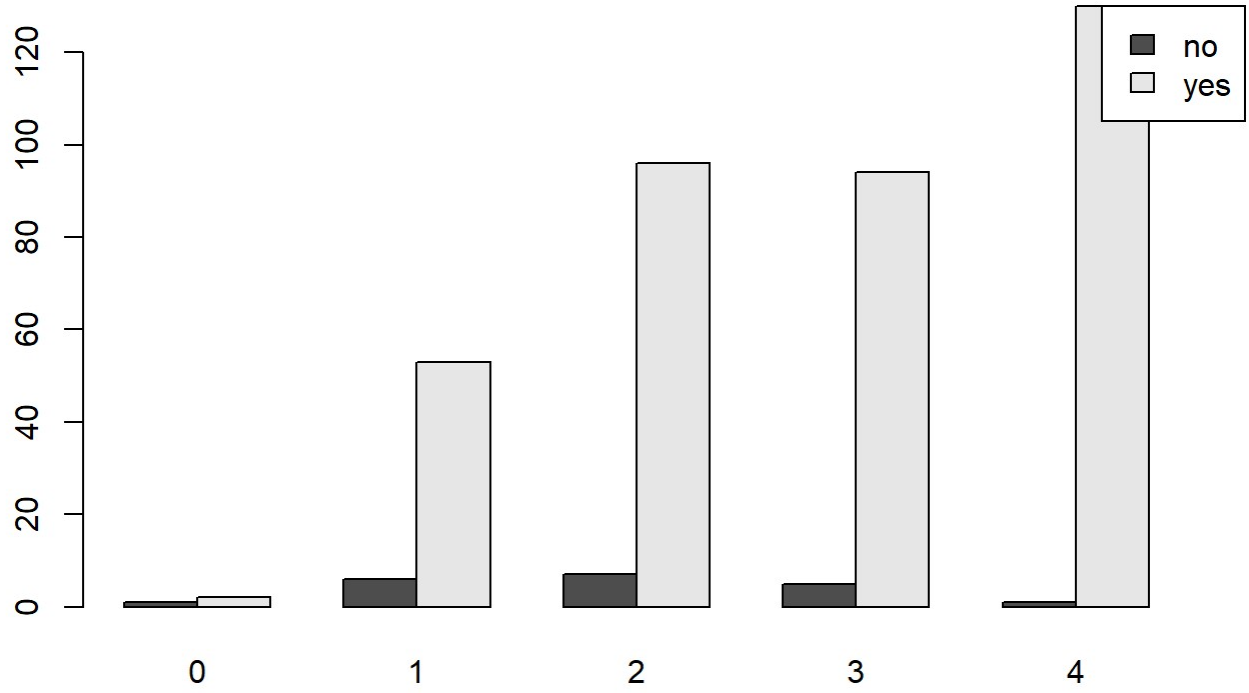
From the bar plots we can see more students are interested to have higher education. Before doing any test we would create some contingency table to check what motivates the students to have higher education. In other words, what other variables are associated with higher education.

We will do a chi Square test to check if other categorical variables are significantly associated with interest in higher education.

Chi-square test can helps us to find out whether a difference between two categorical variables is due to chance or a relationship between them.

Let's see if mother's education has anything to do with student's interest in higher education. Let's create a barplot for both.

```
#Barplot w.r.t mother's education and interest in higher studies
MeduVshigher=table(higher,Medu) #Contingency table
barplot(MeduVshigher,legend.text = TRUE, beside=TRUE, args.legend = list(x = "topright",
                          inset = c(- 0.05, 0)))
```



The bar plot clearly shows the more educated the mother is the more children are interested in higher education.

```
#Chisquare test w.r.t mother's education and interest in higher studies
chisq.test(df$Medu, df$higher)
```

```
## Warning in chisq.test(df$Medu, df$higher): Chi-squared approximation may be
## incorrect
```

```
##
##  Pearson's Chi-squared test
##
## data:  df$Medu and df$higher
## X-squared = 13.87, df = 4, p-value = 0.007721
```

Here the p value is less than 0.05 which concludes there is a significant relationship between mother's education and interest in higher studies.Now, we will try for other categorical variables, if there are significant association between interest in higher education and other categorical variables.

Let us create contingency tables for higher education with other categorical varables.

```
#Contingency table
#i=c(1:2,4:12,16:29)
df_list_f <- function(x) (table(higher, x))
i=c(1:2,4:12,16:20,22:29)
df2 <- df[,i]
lapply(df2, df_list_f)
```

```
## $school
##       x
## higher  GP  MS
##    no   17   3
##    yes 332  43
##
## $sex
##       x
## higher   F   M
##    no    4  16
##    yes 204 171
##
## $address
##       x
## higher   R   U
##    no    6  14
##    yes  82 293
##
## $famsize
##       x
## higher GT3 LE3
##    no   14   6
##    yes 267 108
##
## $Pstatus
##       x
## higher   A   T
##    no    1  19
##    yes  40 335
##
## $Medu
##       x
## higher   0   1   2   3   4
##    no    1   6   7   5   1
##    yes   2  53  96  94 130
##
## $Fedu
##       x
## higher   0   1   2   3   4
##    no    0  10   7   2   1
##    yes   2  72 108  98  95
##
## $Mjob
##       x
## higher at_home health other services teacher
##    no        7      0     7        5       1
##    yes      52     34   134       98      57
##
## $Fjob
##       x
## higher at_home health other services teacher
```

```
##    no        1      0     9        9       1
##    yes      19     18   208      102      28
##
## $reason
##       x
## higher course home other reputation
##    no      10    3     5          2
##    yes    135  106    31        103
##
## $guardian
##       x
## higher father mother other
##    no       4     14     2
##    yes     86    259    30
##
## $schoolsup
##       x
## higher  no yes
##    no   19   1
##    yes 325  50
##
## $famsup
##       x
## higher  no yes
##    no   12   8
##    yes 141 234
##
## $paid
##       x
## higher  no yes
##    no   19   1
##    yes 195 180
##
## $activities
##       x
## higher  no yes
##    no   14   6
##    yes 180 195
##
## $nursery
##       x
## higher  no yes
##    no    6  14
##    yes  75 300
##
## $internet
##       x
## higher  no yes
##    no    4  16
##    yes  62 313
##
```

```
## $romantic
##       x
## higher  no yes
##    no    9  11
##    yes 254 121
##
## $famrel
##       x
## higher   1   2   3   4   5
##    no    0   1   5  10   4
##    yes   8  17  63 185 102
##
## $freetime
##       x
## higher   1   2   3   4   5
##    no    1   1   9   5   4
##    yes  18  63 148 110  36
##
## $goout
##       x
## higher   1   2   3   4   5
##    no    3   4   4   2   7
##    yes  20  99 126  84  46
##
## $Dalc
##       x
## higher   1   2   3   4   5
##    no    9   9   1   0   1
##    yes 267  66  25   9   8
##
## $Walc
##       x
## higher   1   2   3   4   5
##    no    7   2   3   3   5
##    yes 144  83  77  48  23
##
## $health
##       x
## higher   1   2   3   4   5
##    no    1   4   4   3   8
##    yes  46  41  87  63 138
```

The contingency tables gives us a potentiality of having association between interest in higher education and other categorical variables.We need to deep dive to check if there is significant association between interest in higher education with other categorical variables.

```
#Contingency table
df_list_f <- function(x) chisq.test(table(higher, x))
i=c(1:2,4:12,16:20,22:29)
df2 <- df[,i] # df2 contains the columns vs, am, gear and carb
lapply(df2, df_list_f)
```

```
## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect
```

```
## Warning in chisq.test(table(higher, x)): Chi-squared approximation may be
## incorrect
```

```
## $school
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(higher, x)
## X-squared = 0.014947, df = 1, p-value = 0.9027
##
##
## $sex
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(higher, x)
## X-squared = 7.6859, df = 1, p-value = 0.005565
##
##
## $address
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(higher, x)
## X-squared = 0.33171, df = 1, p-value = 0.5647
##
##
## $famsize
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(higher, x)
## X-squared = 6.9167e-29, df = 1, p-value = 1
##
##
## $Pstatus
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(higher, x)
## X-squared = 0.18781, df = 1, p-value = 0.6647
##
##
## $Medu
##
##  Pearson's Chi-squared test
##
## data:  table(higher, x)
## X-squared = 13.87, df = 4, p-value = 0.007721
##
##
## $Fedu
##
##  Pearson's Chi-squared test
```

```
##
## data:  table(higher, x)
## X-squared = 14.216, df = 4, p-value = 0.006636
##
##
## $Mjob
##
##  Pearson's Chi-squared test
##
## data:  table(higher, x)
## X-squared = 8.8482, df = 4, p-value = 0.06501
##
##
## $Fjob
##
##  Pearson's Chi-squared test
##
## data:  table(higher, x)
## X-squared = 3.637, df = 4, p-value = 0.4574
##
##
## $reason
##
##  Pearson's Chi-squared test
##
## data:  table(higher, x)
## X-squared = 10.237, df = 3, p-value = 0.01665
##
##
## $guardian
##
##  Pearson's Chi-squared test
##
## data:  table(higher, x)
## X-squared = 0.16785, df = 2, p-value = 0.9195
##
##
## $schoolsup
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(higher, x)
## X-squared = 0.54863, df = 1, p-value = 0.4589
##
##
## $famsup
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(higher, x)
## X-squared = 3.1262, df = 1, p-value = 0.07704
```

```
##
##
## $paid
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(higher, x)
## X-squared = 12.463, df = 1, p-value = 0.0004152
##
##
## $activities
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(higher, x)
## X-squared = 2.8495, df = 1, p-value = 0.0914
##
##
## $nursery
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(higher, x)
## X-squared = 0.6321, df = 1, p-value = 0.4266
##
##
## $internet
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(higher, x)
## X-squared = 0.0094745, df = 1, p-value = 0.9225
##
##
## $romantic
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(higher, x)
## X-squared = 3.4476, df = 1, p-value = 0.06334
##
##
## $famrel
##
##  Pearson's Chi-squared test
##
## data:  table(higher, x)
## X-squared = 1.5459, df = 4, p-value = 0.8185
##
##
## $freetime
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(higher, x)
## X-squared = 3.93, df = 4, p-value = 0.4156
##
##
## $goout
##
##  Pearson's Chi-squared test
##
## data:  table(higher, x)
## X-squared = 13.067, df = 4, p-value = 0.01095
##
##
## $Dalc
##
##  Pearson's Chi-squared test
##
## data:  table(higher, x)
## X-squared = 10.618, df = 4, p-value = 0.03121
##
##
## $Walc
##
##  Pearson's Chi-squared test
##
## data:  table(higher, x)
## X-squared = 11.249, df = 4, p-value = 0.0239
##
##
## $health
##
##  Pearson's Chi-squared test
##
## data:  table(higher, x)
## X-squared = 2.3865, df = 4, p-value = 0.6651
```

From the chisquare tests which has p value less than 0.05 we have found that sex, mother's education, father's education, reason for choosing school, extra payment for classes, going out, consuming alcohol on workdays and consuming alcohol on weekends are significantly associated with higher education.

```
#Chisquare test
df_list_f <- function(x) chisq.test(table(goout, x))
i=c(19,23,25,27:28)
df2 <- df[,i]
lapply(df2, df_list_f)
```

```
## Warning in chisq.test(table(goout, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(goout, x)): Chi-squared approximation may be
## incorrect

## Warning in chisq.test(table(goout, x)): Chi-squared approximation may be
## incorrect
```

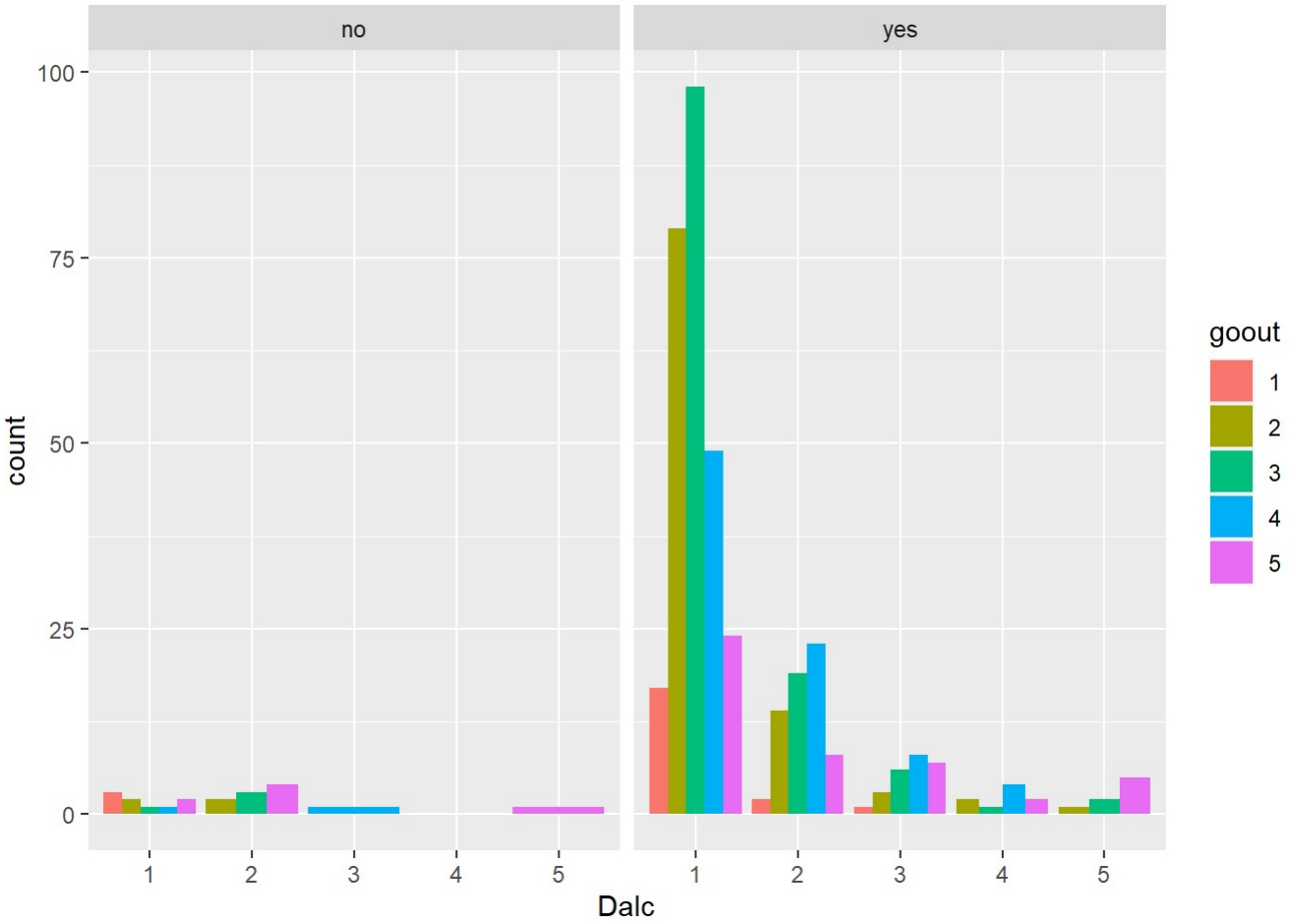
```
## $activities
##
##  Pearson's Chi-squared test
##
## data:  table(goout, x)
## X-squared = 1.5676, df = 4, p-value = 0.8146
##
##
## $romantic
##
##  Pearson's Chi-squared test
##
## data:  table(goout, x)
## X-squared = 1.0439, df = 4, p-value = 0.9031
##
##
## $freetime
##
##  Pearson's Chi-squared test
##
## data:  table(goout, x)
## X-squared = 80.878, df = 16, p-value = 1.156e-10
##
##
## $Dalc
##
##  Pearson's Chi-squared test
##
## data:  table(goout, x)
## X-squared = 48.786, df = 16, p-value = 3.571e-05
##
##
## $Walc
##
##  Pearson's Chi-squared test
##
## data:  table(goout, x)
## X-squared = 116.57, df = 16, p-value < 2.2e-16
```

Wee have checked the association between going out and taking alcohol on working days, alcohol on weekends, involving in a romantic relationship, when they have free time and doing activities.I found that taking

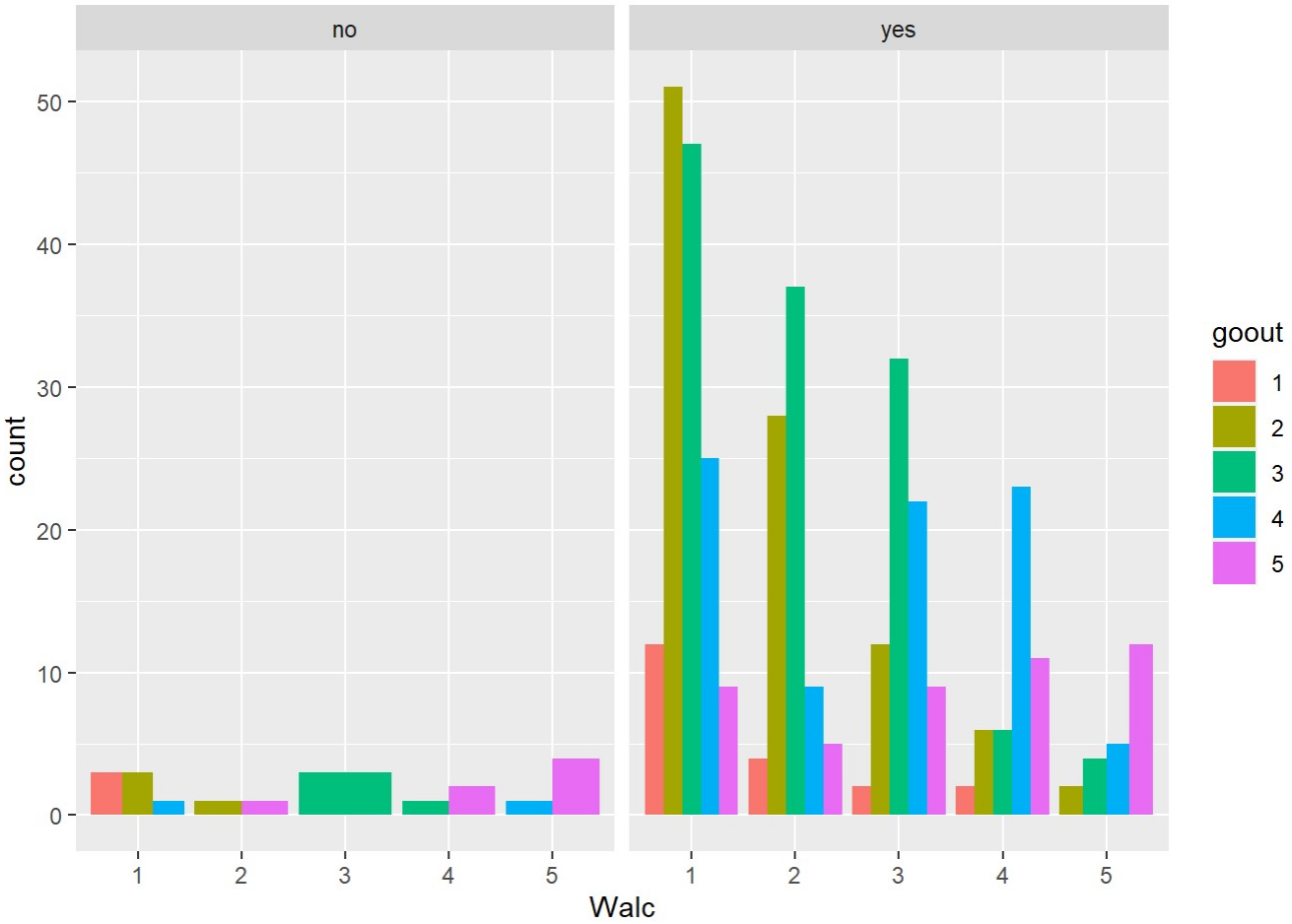
alcohol on working days, alcohol on weekends, and when they have free time are significantly associated with going out.

Now we will create more bar plots with respect to interest in higher education to see any pattern.

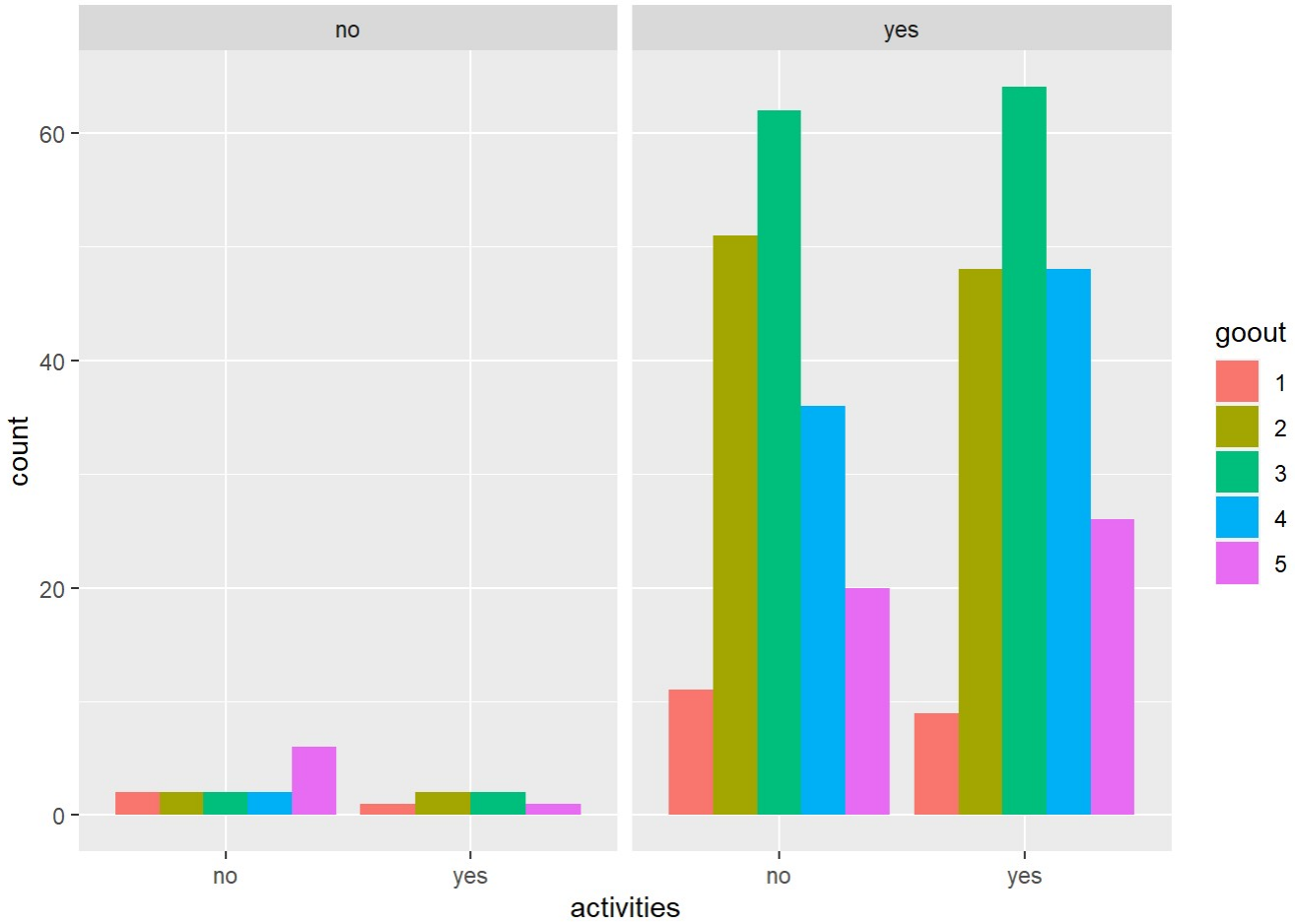
```
ggplot(df) +
  geom_bar(aes(x=Dalc, fill=goout),
           position = "dodge") +
  facet_wrap(~higher) #Interest in higher education, going out with friends, taking alcohol o
n workdays
```



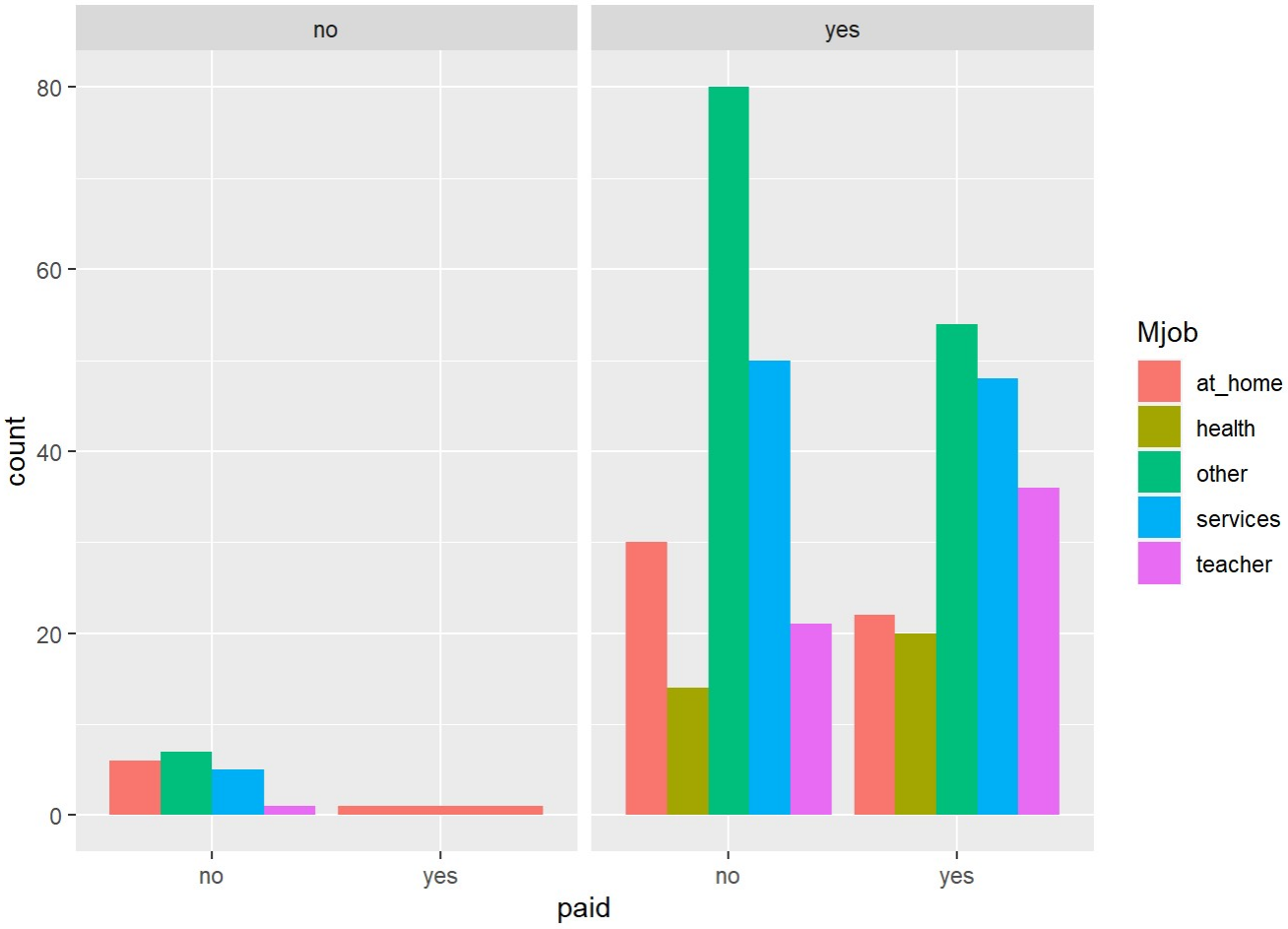

```
ggplot(df) +
  geom_bar(aes(x=Walc, fill=goout),
           position = "dodge") +
  facet_wrap(~higher) #Interest in higher education, going out with friends, taking alcohol o
n weekends
```




```
ggplot(df) +
  geom_bar(aes(x=activities, fill=goout),
           position = "dodge") +
  facet_wrap(~higher)  #Interest in higher education, going out with friends, involved in act
ivities
```




```
ggplot(df) +
  geom_bar(aes(x=paid, fill=Mjob),
           position = "dodge") +
  facet_wrap(~higher)  #Interest in higher education, mother's job, paid for studies
```




```
p_grph <- ggplot(df) +
  geom_bar(aes(x=Fjob, fill=paid),
           position = "dodge") +
  facet_wrap(~higher)  #Interest in higher education, father's job, paid for studies
```

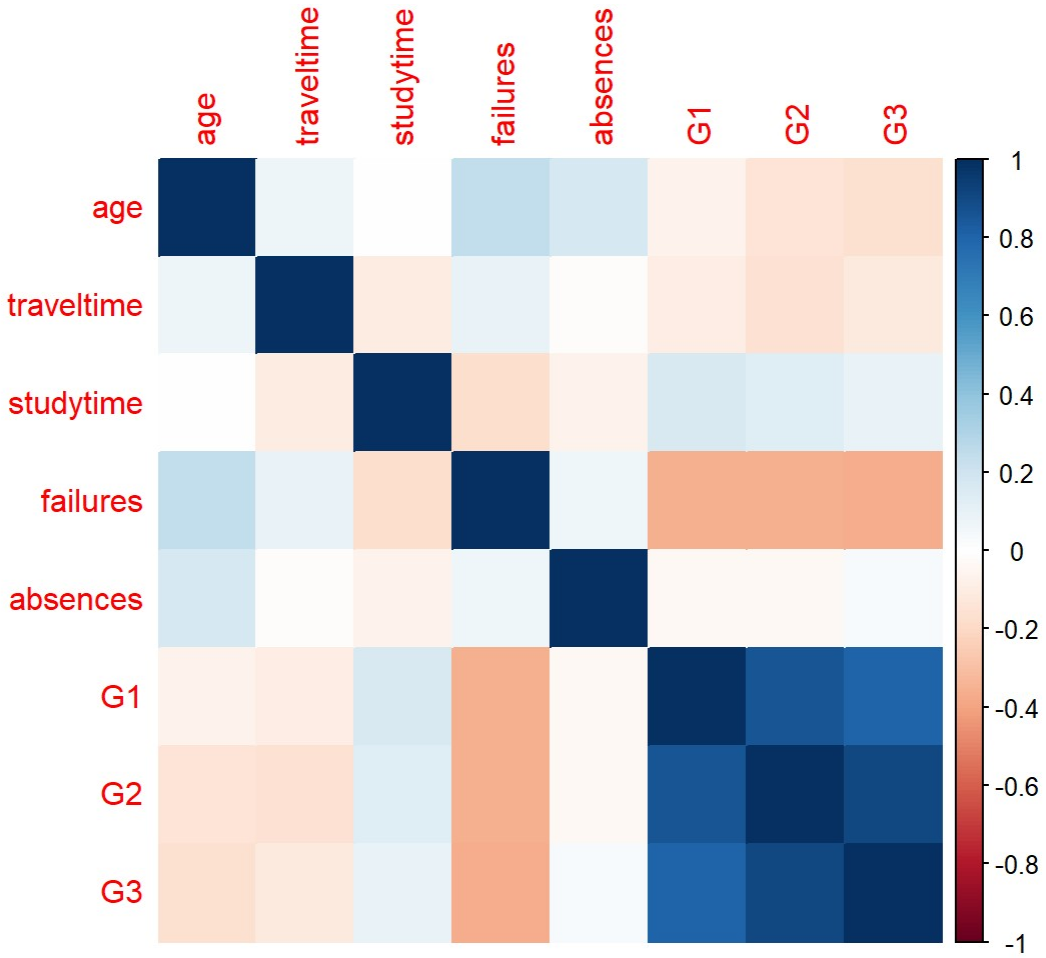
Now it's time to check correlation between continuous variables.We have found that G1 and G2 is highly correlated with G3.

```
library(tidyverse)
dat <- df %>%
  select(age, traveltime, studytime,failures,absences,G1, G2, G3)
cor(dat)
```

```
##                    age  traveltime    studytime    failures    absences
## age         1.000000000  0.07064072 -0.004140037  0.24366538  0.17523008
## traveltime  0.070640721  1.00000000 -0.100909119  0.09223875 -0.01294378
## studytime  -0.004140037 -0.10090912  1.000000000 -0.17356303 -0.06270018
## failures    0.243665377  0.09223875 -0.173563031  1.00000000  0.06372583
## absences    0.175230079 -0.01294378 -0.062700175  0.06372583  1.00000000
## G1         -0.064081497 -0.09303999  0.160611915 -0.35471761 -0.03100290
## G2         -0.143474049 -0.15319796  0.135879999 -0.35589563 -0.03177670
## G3         -0.161579438 -0.11714205  0.097819690 -0.36041494  0.03424732
##                    G1         G2          G3
## age        -0.06408150 -0.1434740 -0.16157944
## traveltime -0.09303999 -0.1531980 -0.11714205
## studytime   0.16061192  0.1358800  0.09781969
## failures   -0.35471761 -0.3558956 -0.36041494
## absences   -0.03100290 -0.0317767  0.03424732
## G1          1.00000000  0.8521181  0.80146793
## G2          0.85211807  1.0000000  0.90486799
## G3          0.80146793  0.9048680  1.00000000
```

```
#pairs.panel(dat,col="red")
```

```
#Checking correlation of continuous variables graphically
library(corrplot)
M<-cor(dat)
corrplot(M, method="color")
```



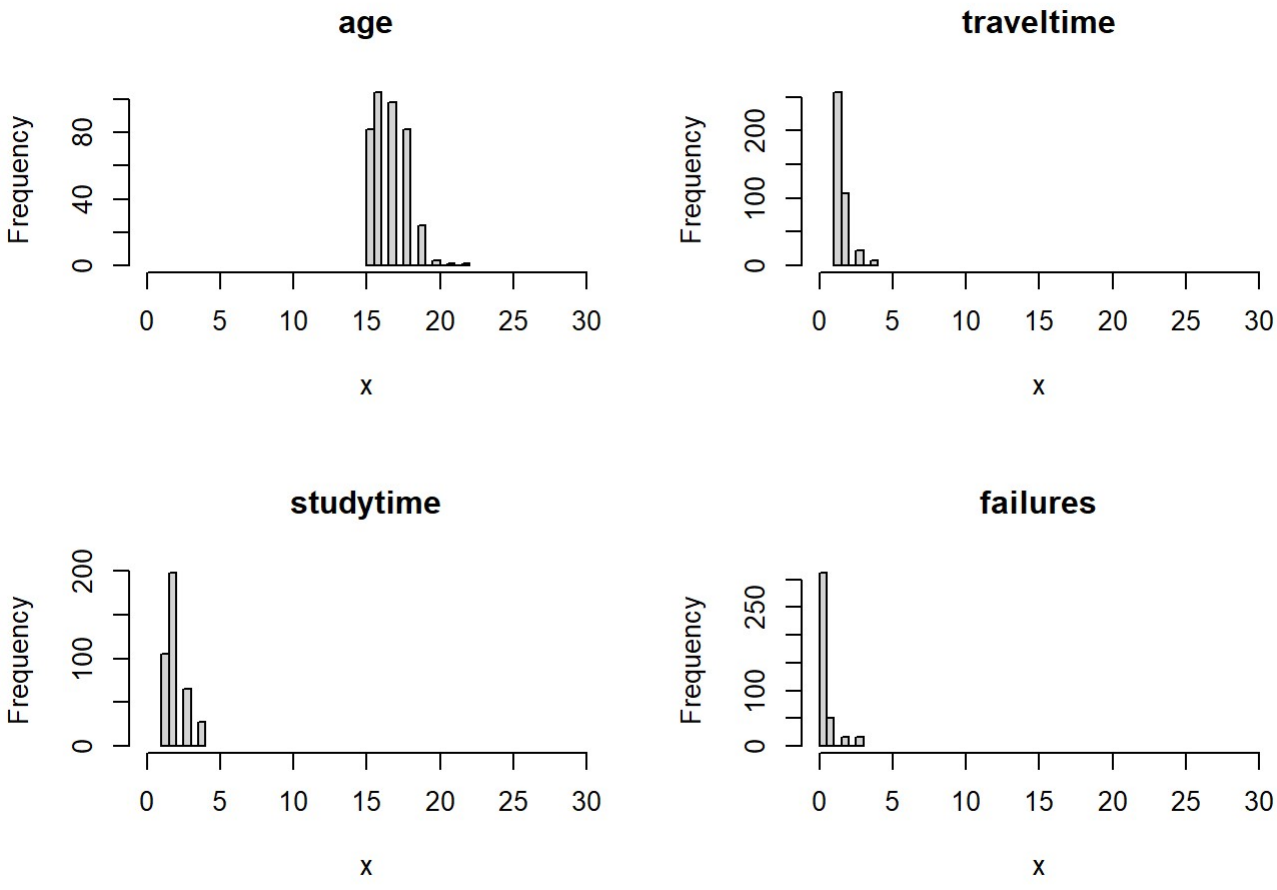
From the above correlation matrix we found that the 1st grade "G1" and the 2nd grade "G2" is correlated and the value is 0.85. As it is more than 70%, we will add G1 and G2.

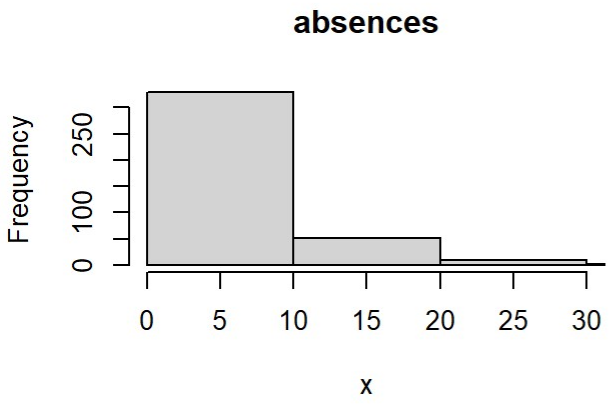
```
df$G <- df$G1+df$G2
df$G
```

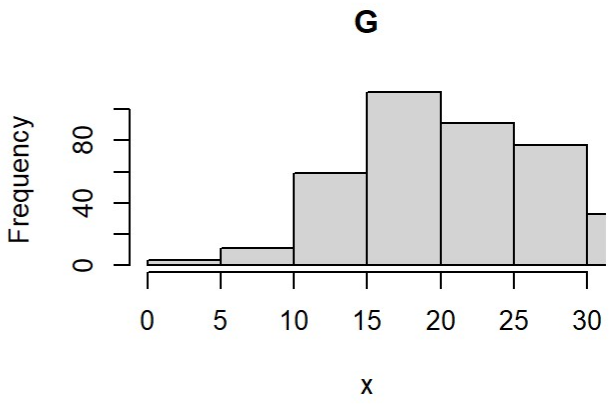
```
##   [1] 11 10 15 29 16 30 24 11 34 29 18 22 28 20 30 28 27 18 11 18 27 27 30 26 19
##  [26] 15 24 31 22 22 20 33 33 18 26 15 31 31 24 27 17 24 37 16 20 16 23 38 30 14
##  [51] 25 24 22 18 23 17 29 29 19 31 21 18 18 19 20 31 26 14 17 32 28 20 14 24 23
##  [76] 18 22 22 16 10 22 21 13 30 19 16 15 27 21 14 14 33 13 21 24 17 26 17 25 16
## [101] 14 33 23 13 34 21 15 34 23 29 37 17 23 37 18 30 24 27 16 27 31 30 26 25 15
## [126] 26 17 15 11 36 12  8 23 23  9 11 10  4 26 32 16 18 20 28  5 19 13 21 13 17
## [151] 11 25 20  5 22 19 28 17 32 22 13 14  7 20 13 23 20 29 13 28 11 28 24 15 21
## [176] 19 26 11 18 20 17 25 33 18 25 24 23 30 15 17 23 16 15 17 27 29 32 18 36 18
## [201] 32 18 18 13 20 19 14 23 18 14 16 24 25 13 18 29 12 12 14 19 12 11 32 25 26
## [226] 17 31 23 18 22 26 22 20 27 16 20 27 25 24 14 24 21  6 25  7 36 24 14  8 28
## [251] 14 17 15 17 20 16 26 22 29 19 35 16 25 19 19 34 18 22 19  6 18 29 22 29 20
## [276] 24 19 18 17 21 16 20 24 17 19 22 36 25 29 28 23 30 24 36 27 26 19 18 27 31
## [301] 22 22 27 34 29 26 35 17 27 22 18 26 24 23 28 24 16 19 22 22 26 20 22 26 31
## [326] 21 29 21 19 28 17 26  7 16 19 31 27 15 31 19 23 20 31 17 21 26 31 20 28 24
## [351] 15 26 15 16 24 19 25 24 20 34 26 25 22 31 23 20 26 13 21 26 14 26 24 11 37
## [376] 16 29 17 30 20 29 13 22 11 11 19 11 12 16 11 18 30 18 23 17
```

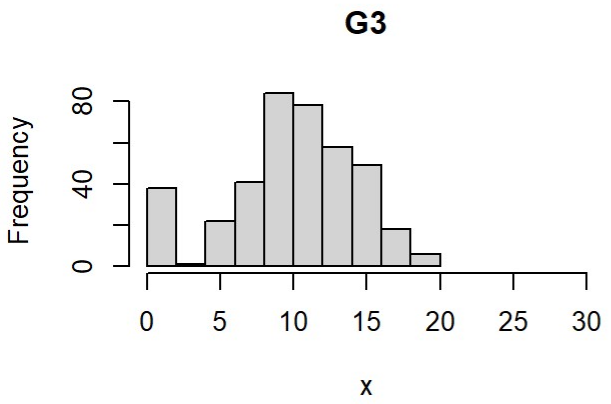
```
df = subset(df, select = -c(G1,G2) )
```

```
#Creating histogram of continuous variables
par(mfrow = c(2, 2))  # Set up a 2 x 2 plotting space

# Create the loop.vector (all the columns)
loop.vector <- c("age","traveltime","studytime","failures","absences","G","G3")

for (i in loop.vector) { # Loop over loop.vector

  # store data in column.i as x
  x <- df[,i]

  # Plot histogram of x
  hist(x,
       main = paste(i),
       xlim = c(0, 30))
}
```









Predictive Data Analysis

Multiple Linear Regression

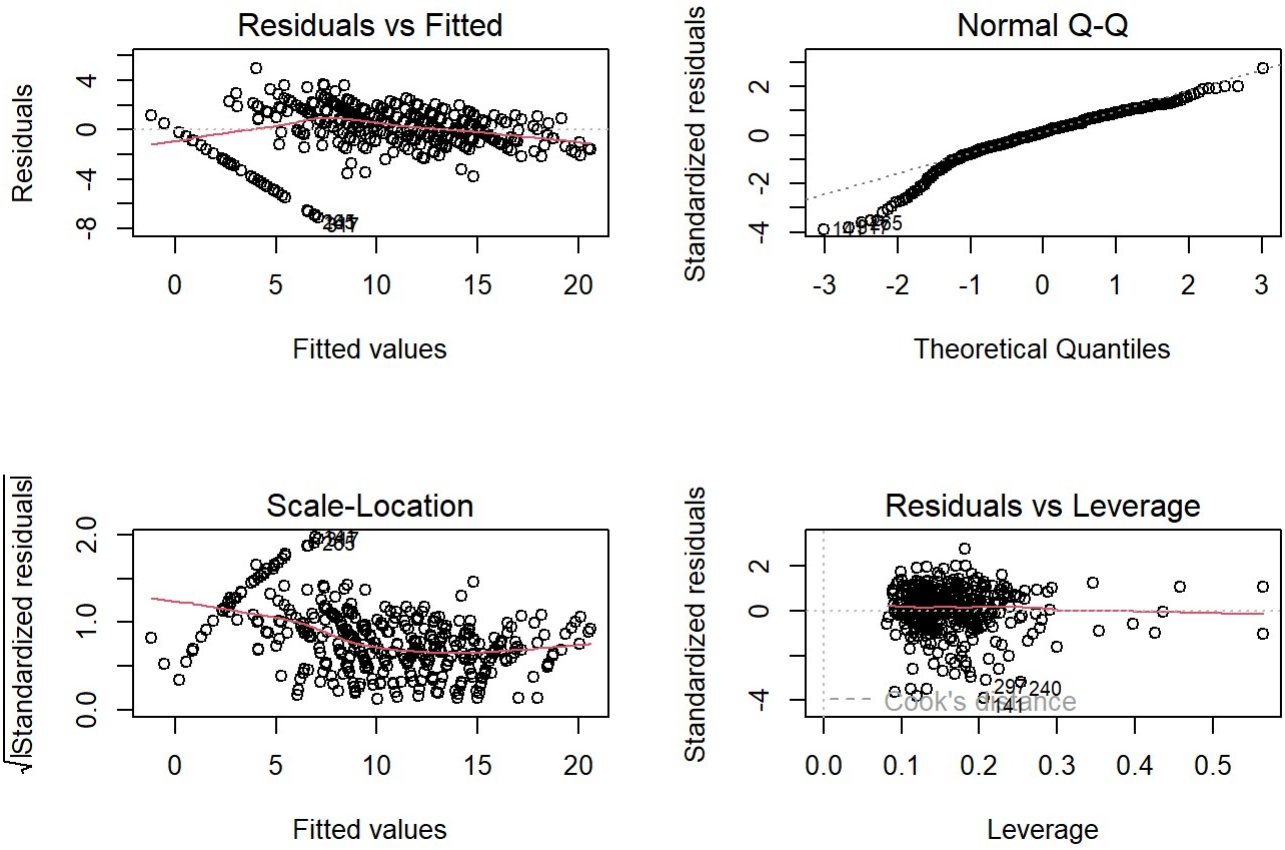
Before doing multiple linear regression, we usually are interested in answering a few important questions. 1. Is at least one of the predictors X_1, X_2, ..., X_p useful in predicting the response?

2. Do all the predictors help to explain Y, or is only subset of the predictors useful?
3. How well does the model fit the data?
4. Given a set of predictor values, what response would should we predict.

Before answering these questions we will first check

1. Non-linearity of the response-predictor relationship
2. Correlation of error terms
3. Non-constant variance of error terms
4. Outliers
5. High-leverage points
6. Colinearity

```
#Ploting the model
model=lm(G3~., data=df)
par(mfrow = c(2, 2))
plot(model)
```




```
#Checking Multicolinearity
#create vector of VIF values
vif_values <- vif(model)
vif_values
```

```
##                GVIF Df GVIF^(1/(2*Df))
## school     1.665170  1        1.290415
## sex        1.598355  1        1.264260
## age        1.936451  1        1.391564
## address    1.526507  1        1.235519
## famsize    1.240299  1        1.113687
## Pstatus    1.257137  1        1.121221
## Medu       6.323713  4        1.259278
## Fedu       3.971724  4        1.188153
## Mjob       5.046677  4        1.224266
## Fjob       3.066243  4        1.150339
## reason     1.797862  3        1.102705
## guardian   1.973116  2        1.185190
## traveltime 1.412766  1        1.188598
## studytime  1.531791  1        1.237656
## failures   1.674411  1        1.293990
## schoolsup  1.288284  1        1.135026
## famsup     1.351998  1        1.162755
## paid       1.447050  1        1.202934
## activities 1.245094  1        1.115838
## nursery    1.248240  1        1.117247
## higher     1.437270  1        1.198862
## internet   1.329965  1        1.153241
## romantic   1.237362  1        1.112368
## famrel     1.790103  4        1.075498
## freetime   2.647874  4        1.129437
## goout      2.984521  4        1.146461
## Dalc       5.094611  4        1.225713
## Walc       6.409862  4        1.261409
## health     2.117556  4        1.098321
## absences   1.340484  1        1.157793
## G          1.608353  1        1.268209
```

From the above image of Normal Q-Q plot we can say the data follows normal distribution.

The image above shows the "Residual vs. Fitted"-plot and the "Scale-Location"-plot for a regression model without heteroscedastic residuals. In other words, the variance of the residuals is the same for all values of the fitted values.

The residual plot is not showing any trend, just some outliers.So we can say there is no correlation among the errors.

From the residual vs. leverage plot we don't see any high leverage points.

Now we will start answering our questions that we have set above. The first question we need to ask whether all of the regression coefficients are zero. So, we test the null hypothesis as

H_0: All the regression coefficients are zero.

H_1: Atleast one regression coefficient is non-zero.

The hypothesis test is performed by computing the F-statistic.

Before starting the process the data is splitted into train and test data. We would build the model based on the training data.

```
#Splitting the data into train and test set
set.seed(0)
parts = createDataPartition(df$G3, p = .8, list = F)
train = df[parts, ]
test = df[-parts, ]
```

```
#Creating model including all variables
fullmodel=lm(G3~., data=train)
summary(fullmodel)
```

```
##
## Call:
## lm(formula = G3 ~ ., data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.5760 -0.8249  0.1818  1.1993  4.5850
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        2.10485    3.61227   0.583  0.56062
## schoolMS           0.67505    0.45680   1.478  0.14072
## sexM               0.09085    0.29806   0.305  0.76076
## age               -0.29648    0.12792  -2.318  0.02126 *
## addressU          -0.22150    0.33785  -0.656  0.51266
## famsizeLE3         0.23081    0.27459   0.841  0.40137
## PstatusT          -0.32414    0.43287  -0.749  0.45467
## Medu1             -0.34971    1.59456  -0.219  0.82658
## Medu2             -0.49383    1.57620  -0.313  0.75431
## Medu3              0.09302    1.59659   0.058  0.95359
## Medu4              0.41486    1.62662   0.255  0.79889
## Fedu1             -0.41652    1.55087  -0.269  0.78848
## Fedu2             -1.25735    1.55014  -0.811  0.41806
## Fedu3             -0.82056    1.54376  -0.532  0.59552
## Fedu4             -1.17119    1.57778  -0.742  0.45859
## Mjobhealth        -0.48383    0.63542  -0.761  0.44711
## Mjobother          0.27466    0.39552   0.694  0.48805
## Mjobservices      -0.02278    0.45813  -0.050  0.96038
## Mjobteacher       -0.58291    0.59647  -0.977  0.32937
## Fjobhealth        -0.12046    0.87876  -0.137  0.89108
## Fjobother         -0.50819    0.58692  -0.866  0.38739
## Fjobservices      -0.44135    0.60023  -0.735  0.46283
## Fjobteacher       -0.62676    0.75759  -0.827  0.40885
## reasonhome        -0.14916    0.32409  -0.460  0.64574
## reasonother        0.77704    0.44593   1.742  0.08264 .
## reasonreputation   0.22705    0.32803   0.692  0.48947
## guardianmother     0.20000    0.31384   0.637  0.52452
## guardianother      0.08837    0.56630   0.156  0.87612
## traveltime        -0.08718    0.19652  -0.444  0.65769
## studytime         -0.25669    0.17563  -1.462  0.14510
## failures          -0.12304    0.19900  -0.618  0.53695
## schoolsupyes       0.55508    0.39435   1.408  0.16049
## famsupyes          0.20750    0.27649   0.750  0.45367
## paidyes            0.25500    0.27818   0.917  0.36019
## activitiesyes     -0.63461    0.26072  -2.434  0.01562 *
## nurseryyes         0.09795    0.30989   0.316  0.75221
## higheryes          0.44413    0.61323   0.724  0.46959
## internetyes        0.23462    0.34766   0.675  0.50038
## romanticyes       -0.46293    0.27686  -1.672  0.09575 .
## famrel2           -0.97115    0.95878  -1.013  0.31208
## famrel3            0.31481    0.83322   0.378  0.70588
```

```
## famrel4           0.26425    0.80549   0.328  0.74313
## famrel5           0.65526    0.82288   0.796  0.42660
## freetime2         0.77741    0.68096   1.142  0.25468
## freetime3         0.62761    0.64988   0.966  0.33509
## freetime4         0.91949    0.66622   1.380  0.16875
## freetime5         0.70445    0.76381   0.922  0.35726
## goout2            0.98849    0.58540   1.689  0.09253 .
## goout3            0.82776    0.59111   1.400  0.16263
## goout4            0.30370    0.61487   0.494  0.62179
## goout5            0.29805    0.66142   0.451  0.65264
## Dalc2            -0.95217    0.36910  -2.580  0.01045 *
## Dalc3            -0.90680    0.56560  -1.603  0.11012
## Dalc4            -1.39084    0.93888  -1.481  0.13975
## Dalc5            -1.57787    1.00383  -1.572  0.11723
## Walc2            -0.05851    0.36213  -0.162  0.87176
## Walc3             0.72314    0.39845   1.815  0.07072 .
## Walc4             0.84977    0.50406   1.686  0.09306 .
## Walc5             2.01322    0.69042   2.916  0.00386 **
## health2          -0.53840    0.52958  -1.017  0.31029
## health3           0.23884    0.45584   0.524  0.60076
## health4           0.33772    0.47962   0.704  0.48199
## health5           0.28327    0.42630   0.664  0.50698
## absences          0.04742    0.01632   2.906  0.00399 **
## G                 0.58946    0.02136  27.601  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.015 on 253 degrees of freedom
## Multiple R-squared:  0.8481, Adjusted R-squared:  0.8097
## F-statistic: 22.07 on 64 and 253 DF,  p-value: < 2.2e-16
```

When there is no relationship between the response and predictors, one would expect to take on a value close to 1. Here our F statistic is 24.38. Since this is larger than 1, it provides compelling evidence against null hypothesis.Also, the p value associated with the F statistic is essentially zero, so we have extremely strong evidence that at least one of the predictors is useful predicting the response variable.This answers our first question.

It is possible that all the predictors are associated with the response but it is more often the case that the response is only associated with a subset of predictors.The task of determining which predictors are associated with response, in order to fit a single model involving only those predictors is referred to as variable selection.

There are automated and efficient classical approaches to choose a smaller set of models to consider. These are forward selection, backward selection, and mixed or stepwise selection. The stepwise selection is a combination of both forward and backward selection. It stats with no variable. Then adds variables one by one. And, if at any point the p value of a variable rises above a certain threshold it was then dropped from the model.

```
#define intercept-only model
intercept_only <- lm(G3 ~ 1, data=train)

#define model with all predictors
all <- lm(G3 ~ ., data=train)

#Reduced model
#perform backward stepwise regression
both <- step(intercept_only, direction='both', scope=formula(all), trace=0)
#view results of backward stepwise regression
both
```

```
##
## Call:
## lm(formula = G3 ~ G + age + absences + activities + famrel +
##     school + Fedu + romantic, data = train)
##
## Coefficients:
##   (Intercept)              G            age       absences  activitiesyes
##       4.98771        0.59694       -0.42440        0.05768       -0.59667
##       famrel2        famrel3        famrel4        famrel5       schoolMS
##      -0.98303        0.14299        0.20120        0.62694        0.77102
##         Fedu1          Fedu2          Fedu3          Fedu4    romanticyes
##      -0.20444       -1.09318       -0.39885       -0.67904       -0.39364
```

```
summary(both)
```

```
##
## Call:
## lm(formula = G3 ~ G + age + absences + activities + famrel +
##     school + Fedu + romantic, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.2128 -0.7021  0.2315  1.2494  4.6575
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    4.98771    2.27761   2.190   0.0293 *
## G              0.59694    0.01704  35.022  < 2e-16 ***
## age           -0.42440    0.10278  -4.129 4.71e-05 ***
## absences       0.05768    0.01419   4.065 6.12e-05 ***
## activitiesyes -0.59667    0.23191  -2.573   0.0106 *
## famrel2       -0.98303    0.87524  -1.123   0.2623
## famrel3        0.14299    0.76676   0.186   0.8522
## famrel4        0.20120    0.73580   0.273   0.7847
## famrel5        0.62694    0.75630   0.829   0.4078
## schoolMS       0.77102    0.38577   1.999   0.0465 *
## Fedu1         -0.20444    1.46002  -0.140   0.8887
## Fedu2         -1.09318    1.44597  -0.756   0.4502
## Fedu3         -0.39885    1.44762  -0.276   0.7831
## Fedu4         -0.67904    1.44431  -0.470   0.6386
## romanticyes   -0.39364    0.25133  -1.566   0.1183
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.004 on 303 degrees of freedom
## Multiple R-squared:  0.8201, Adjusted R-squared:  0.8118
## F-statistic: 98.68 on 14 and 303 DF,  p-value: < 2.2e-16
```

Thus we get our selected predictors that explain our response variable. These are 1st grade, absences from school, family relationship, study time, 2nd grade, activities, school, reason for choosing the school and interest in higher education.

There are different approaches to judge the model fitting. These include Mallow's Cp, Akaike Information Criterion(AIC), Bayesian Information Criterion(BIC), adjusted R^2.

To check how well the model fitted the data we have check R^2 and RSE, two common numerical measures of model fit.Here the R^2 value is close to 1, which means 86% variation in the response variable can be explained by the model.The model hat includes all the predictors has a small increase in R^2 compared to our reduced model.Additionally, The model has the lowest AIC value.However, it turns out that the model has some insignificant predictors.

Also, the full model has RSE 1.88, However , the reduced model is slightly lesser than that.

Interpretation of beta coefficients:

On avereage for 1 score increase in 2nd grade their final grade increases by 0.97 points while keeping all other variables constant.

On avereage for 1 day increase in absence their final grade increases by 0.05 points while keeping all other variables constant.

On avereage for 1 hr increase in studytime their final grade decreases by 0.32 points while keeping all other variables constant.

On avereage for 1 hr increase in age their final grade decreases by 0.28 points while keeping all other variables constant.

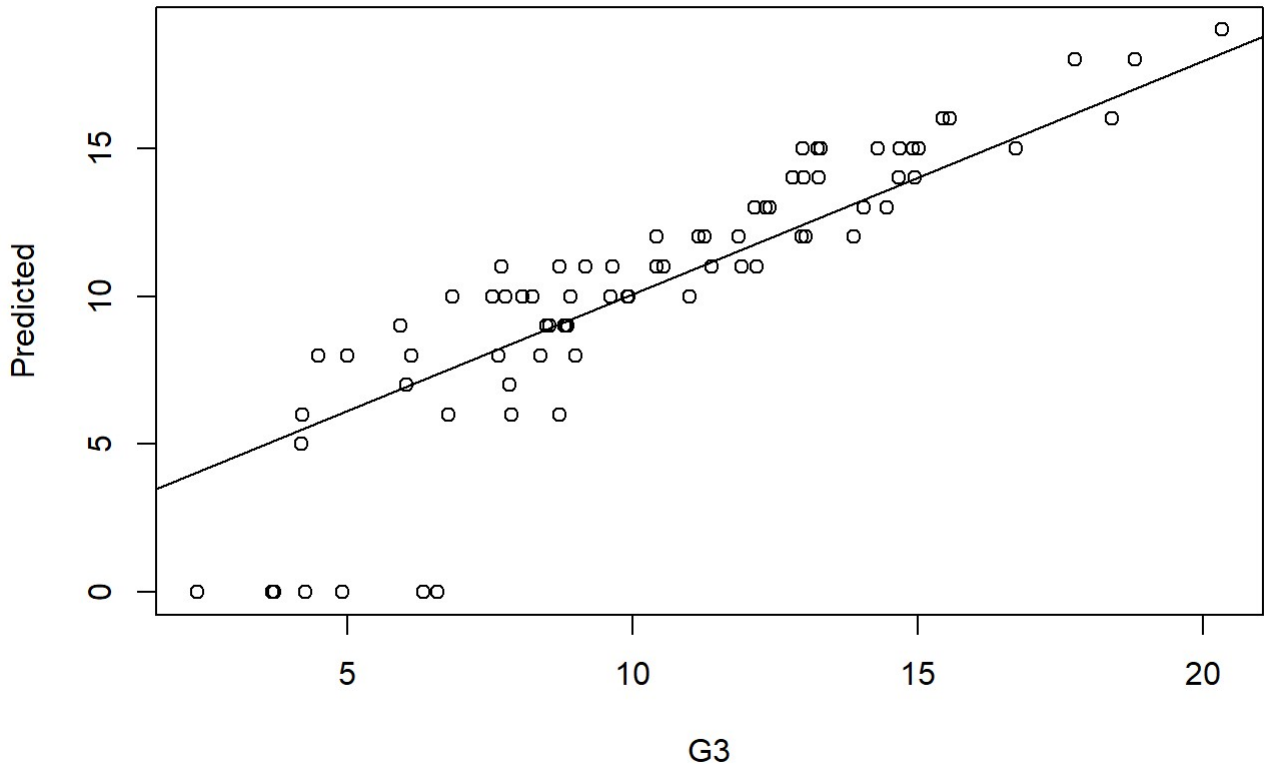
On avereage for 1 hr increase in 1st grade G1 their final grade increases by 0.17 points while keeping all other variables constant.

On average students who do activities gets 0.58 less points in final grade than their reference group.

On average students who are interested in higher education gets 0.86 more points than their reference group.

Then comes prediction.The prediction has been done in test data and then created a scatter plot with predicted G3 and original G3.

```
predicted<-predict(both, test)
test["predicted"]<-predicted
plot(test$predicted, test$G3, xlab="G3", ylab="Predicted")
abline(lm(test$predicted~test$G3))
```



Then we calculated R^2 on test data. The R^2 in test data is little bit less than R^2 in training data.But still is good enough explain the variation of our data.

```
require(miscTools)
r2 <- rSquared(test$G3, resid = test$G3-test$predicted)
r2
```

```
##           [,1]
## [1,] 0.7940755
```

Stepwise process are not always good when accuracy of the model comes into question.We have some variables that are insignificant. Dropping those variables might hurt our R^2. However it is easy to use as it is automated. So, we will go for another approach which is known as K-fold cross validation.

K-Fold Cross Validation

To evaluate the performance of a model on a dataset, we need to measure how well the predictions made by the model match the observed data.

One commonly used method for doing this is known as k-fold cross-validation, which uses the following approach:

1. Randomly divide a dataset into k groups, or "folds", of roughly equal size.
2. Choose one of the folds to be the holdout set. Fit the model on the remaining k-1 folds. Calculate the test MSE on the observations in the fold that was held out.
3. Repeat this process k times, using a different set each time as the holdout set.
4. Calculate the overall test MSE to be the average of the k test MSE's.

The following code shows how to fit a multiple linear regression model to this dataset in R and perform k-fold cross validation with k = 10 folds to evaluate the model performance:

```
#specify the cross-validation method
ctrl <- trainControl(method = "cv", number = 5)

#fit a regression model and use k-fold CV to evaluate performance
model1 <- train(G3 ~ ., data = df, method = "lm", trControl = ctrl)
```

```
## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading
```

```
#view summary of k-fold CV
print(model1)
```

```
## Linear Regression
##
## 395 samples
##  31 predictor
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 317, 315, 316, 316, 316
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   2.268542  0.7590886  1.672951
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```

Interpretation of the output:

• No pre-processing occurred. That is, we didn't scale the data in any way before fitting the models.

• The resampling method we used to evaluate the model was cross-validation with 5 folds.

• The sample size for each training set was 315 to 316.

• RMSE: The root mean squared error. This measures the average difference between the predictions made by the model and the actual observations. The lower the RMSE, the more closely a model can predict the actual observations.

• R squared: This is a measure of the correlation between the predictions made by the model and the actual observations. The higher the R-squared, the more closely a model can predict the actual observations.

• MAE: The mean absolute error. This is the average absolute difference between the predictions made by the model and the actual observations. The lower the MAE, the more closely a model can predict the actual observations.

```
#view final model
model1$finalModel
```

```
##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Coefficients:
##      (Intercept)          schoolMS              sexM               age
##          2.41260           0.48825           0.03278          -0.22848
##         addressU        famsizeLE3          PstatusT             Medu1
##          0.04270           0.16057          -0.16775          -1.76472
##            Medu2             Medu3             Medu4             Fedu1
##         -1.59834          -1.23918          -0.94742          -0.30399
##            Fedu2             Fedu3             Fedu4        Mjobhealth
##         -1.20785          -0.69798          -1.20346          -0.26343
##        Mjobother      Mjobservices       Mjobteacher        Fjobhealth
##          0.34504           0.15789          -0.27575           0.53444
##        Fjobother      Fjobservices       Fjobteacher        reasonhome
##          0.15173           0.09569          -0.08060          -0.20380
##      reasonother  reasonreputation    guardianmother     guardianother
##          0.56272           0.04903           0.13758          -0.08397
##       traveltime         studytime          failures      schoolsupyes
##         -0.03029          -0.25179          -0.12421           0.79204
##        famsupyes           paidyes     activitiesyes        nurseryyes
##          0.30268           0.24170          -0.34990          -0.06799
##        higheryes       internetyes       romanticyes           famrel2
##         -0.08514           0.07513          -0.57954          -0.96490
##          famrel3           famrel4           famrel5         freetime2
##         -0.02876           0.05759           0.40320           0.10249
##        freetime3         freetime4         freetime5            goout2
##         -0.05260           0.26278           0.22506           0.93339
##           goout3            goout4            goout5             Dalc2
##          1.09316           0.70718           0.50755          -0.70335
##            Dalc3             Dalc4             Dalc5             Walc2
##         -0.39438          -0.79055          -0.90978          -0.22115
##            Walc3             Walc4             Walc5           health2
##          0.40818           0.74832           1.43594          -0.55168
##          health3           health4           health5          absences
##          0.23010           0.11984           0.07182           0.04257
##                G
##          0.60118
```

The final model turns out to be: G3 = 1.88923 -0.02918*(PstatusT) – 1.12538(x2)

```
#view predictions for each fold
model1$resample
```

```
##       RMSE  Rsquared      MAE Resample
## 1 2.336258 0.7179686 1.664518    Fold1
## 2 2.081632 0.7778204 1.593373    Fold2
## 3 2.301911 0.7768479 1.761490    Fold3
## 4 2.150249 0.8076345 1.705587    Fold4
## 5 2.472661 0.7151717 1.639786    Fold5
```

Advantages of K-fold Cross-Validation

• Fast computation speed.

• A very effective method to estimate the prediction error and the accuracy of a model. Disadvantages of K-fold Cross-Validation

• A lower value of K leads to a biased model and a higher value of K can lead to variability in the performance metrics of the model. Thus, it is very important to use the correct value of K for the model (generally K = 5 and K = 10 is desirable).

###Ridge & Lasso Regression

```
#define response variable
y <- df$G3

#define matrix of predictor variables
x <- data.matrix(df[, c(1:31, 32)])
```

```
library(glmnet)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
## Loaded glmnet 4.1-4
```

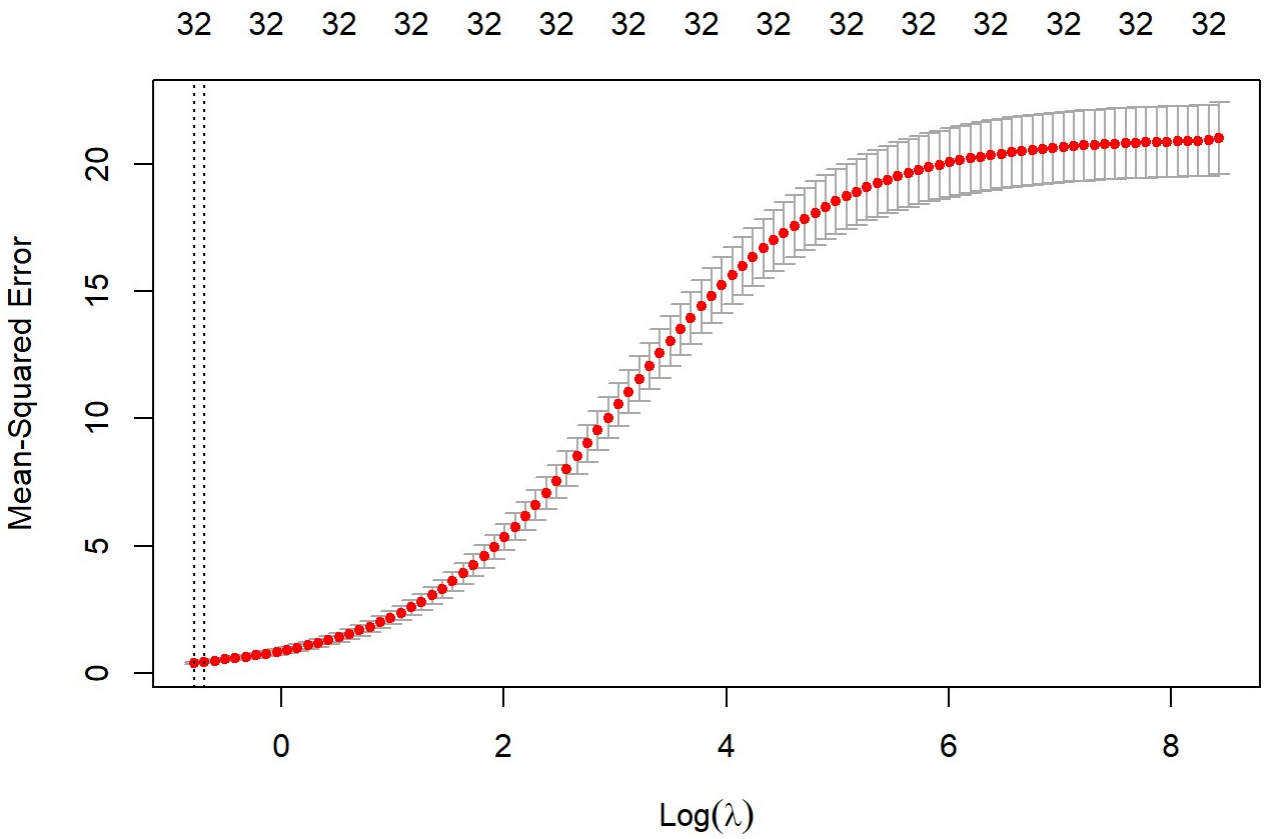
```
#fit ridge regression model
model <- glmnet(x, y, alpha = 0)

#view summary of model
summary(model)
```

```
##           Length Class     Mode
## a0         100   -none-    numeric
## beta      3200   dgCMatrix S4
## df         100   -none-    numeric
## dim          2   -none-    numeric
## lambda     100   -none-    numeric
## dev.ratio  100   -none-    numeric
## nulldev      1   -none-    numeric
## npasses      1   -none-    numeric
## jerr         1   -none-    numeric
## offset       1   -none-    logical
## call         4   -none-    call
## nobs         1   -none-    numeric
```

```
#perform k-fold cross-validation to find optimal lambda value
cv_model <- cv.glmnet(x, y, alpha = 0)

#find optimal lambda value that minimizes test MSE
best_lambda <- cv_model$lambda.min
best_lambda
```

```
## [1] 0.457564
```

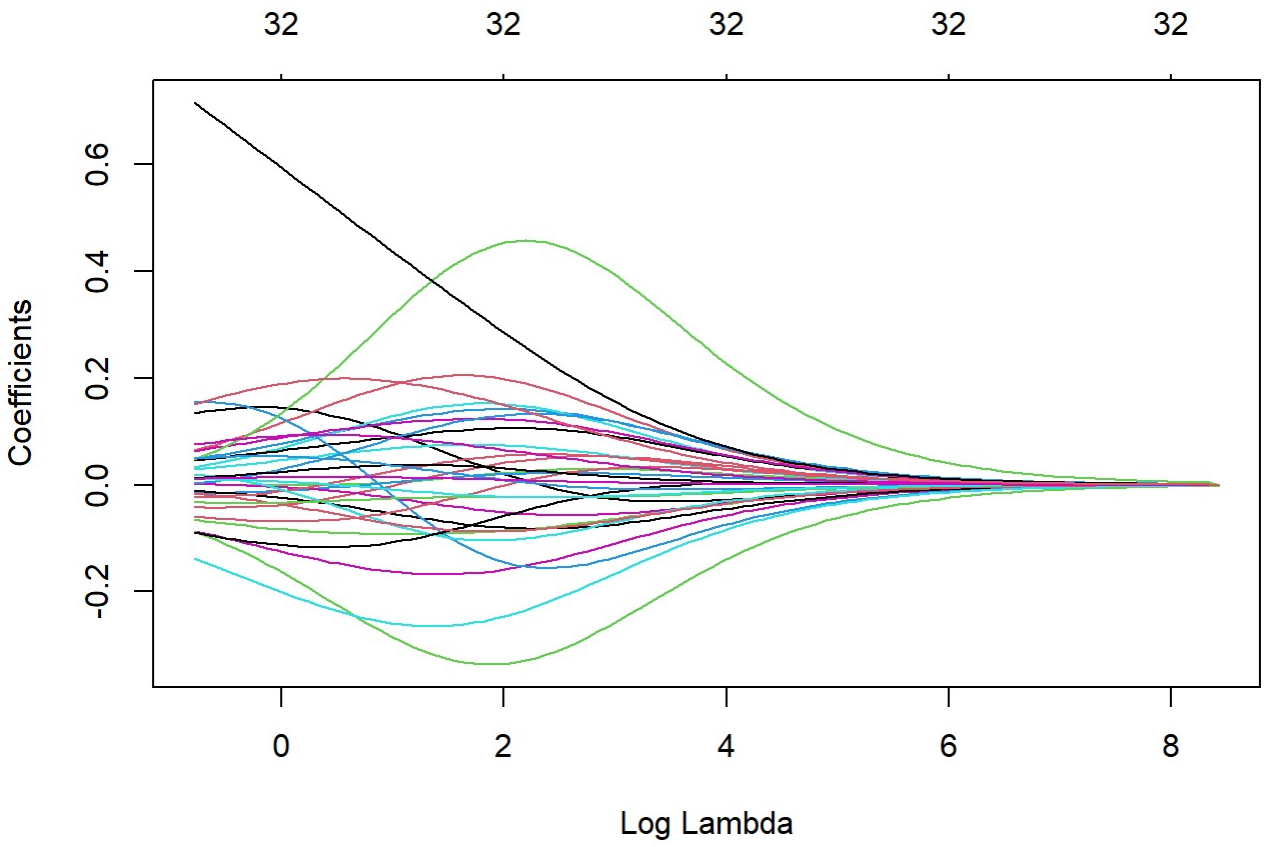
```
#produce plot of test MSE by lambda value
plot(cv_model)
```




```
#find coefficients of best model
best_model <- glmnet(x, y, alpha = 0, lambda = best_lambda)
coef(best_model)
```

```
## 33 x 1 sparse Matrix of class "dgCMatrix"
##                        s0
## (Intercept)  0.2068524404
## school       0.1332062801
## sex          0.0641012819
## age         -0.0645310120
## address      0.0482319076
## famsize      0.0330880138
## Pstatus     -0.0859723196
## Medu         0.0456239958
## Fedu        -0.0423061984
## Mjob         0.0005467531
## Fjob        -0.0155066732
## reason       0.0288070892
## guardian     0.0007394317
## traveltime  -0.0116006104
## studytime   -0.0237694879
## failures    -0.0865882989
## schoolsup    0.1535628155
## famsup       0.0206355348
## paid         0.0630656011
## activities  -0.0887638931
## nursery     -0.0581354416
## higher       0.0472263664
## internet     0.0031711414
## romantic    -0.1374566042
## famrel       0.0752645367
## freetime     0.0125989347
## goout       -0.0148957639
## Dalc        -0.0326246703
## Walc         0.0490382228
## health       0.0097566547
## absences     0.0110924672
## G3           0.7161789937
## G            0.1500618853
```

```
#produce Ridge trace plot
plot(model, xvar = "lambda")
```




```
#use fitted best model to make predictions
y_predicted <- predict(model, s = best_lambda, newx = x)

#find SST and SSE
sst <- sum((y - mean(y))^2)
sse <- sum((y_predicted - y)^2)

#find R-Squared
rsq <- 1 - sse/sst
rsq
```

```
## [1] 0.9848606
```

```
#define response variable
y <- df$G3

#define matrix of predictor variables
x <- data.matrix(df[, c(1:31, 32)])
```

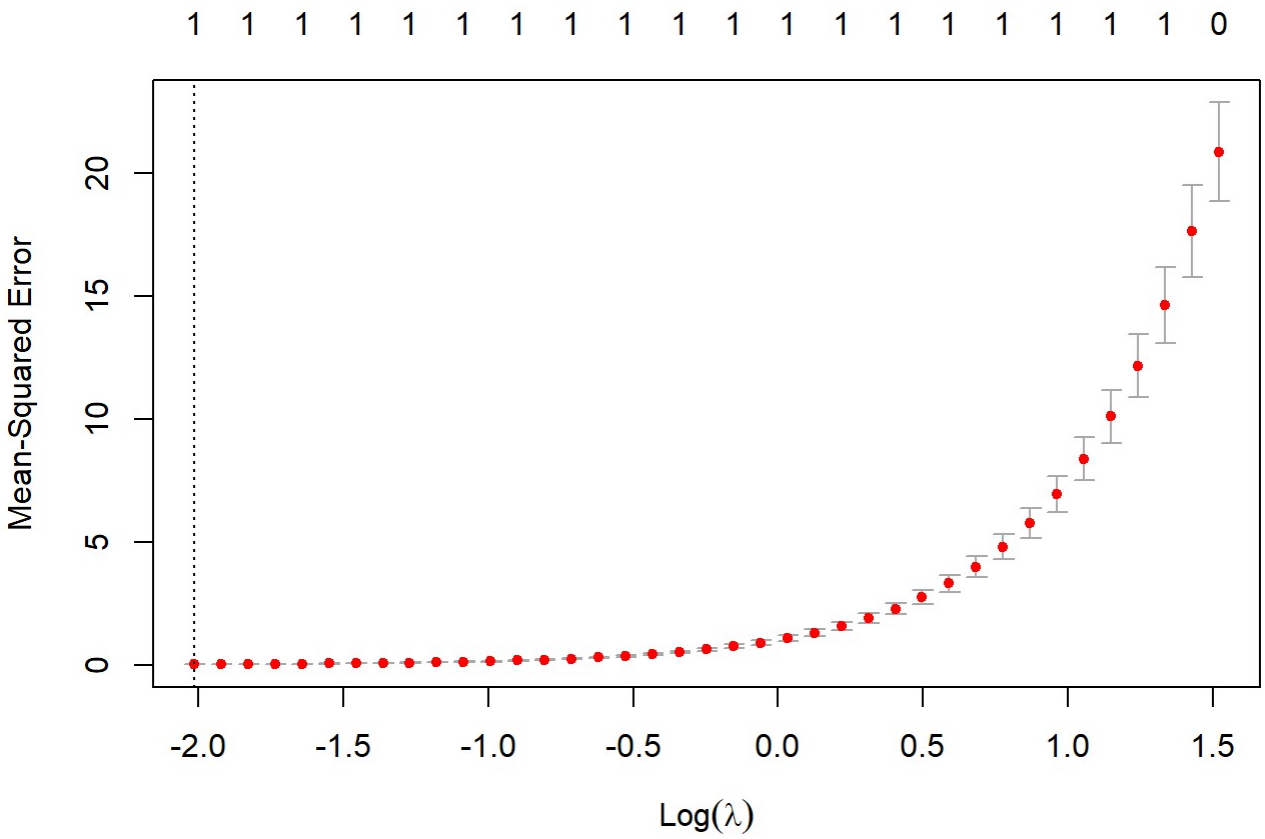
```
library(glmnet)

#fit ridge regression model
model <- glmnet(x, y, alpha = 1)

#view summary of model
summary(model)
```

```
##           Length Class     Mode
## a0          39   -none-    numeric
## beta      1248   dgCMatrix S4
## df          39   -none-    numeric
## dim          2   -none-    numeric
## lambda      39   -none-    numeric
## dev.ratio   39   -none-    numeric
## nulldev      1   -none-    numeric
## npasses      1   -none-    numeric
## jerr         1   -none-    numeric
## offset       1   -none-    logical
## call         4   -none-    call
## nobs         1   -none-    numeric
```

```
#perform k-fold cross-validation to find optimal lambda value
cv_model <- cv.glmnet(x, y, alpha = 1)

#find optimal lambda value that minimizes test MSE
best_lambda <- cv_model$lambda.min
best_lambda
```

```
## [1] 0.1333823
```

```
#produce plot of test MSE by lambda value
plot(cv_model)
```

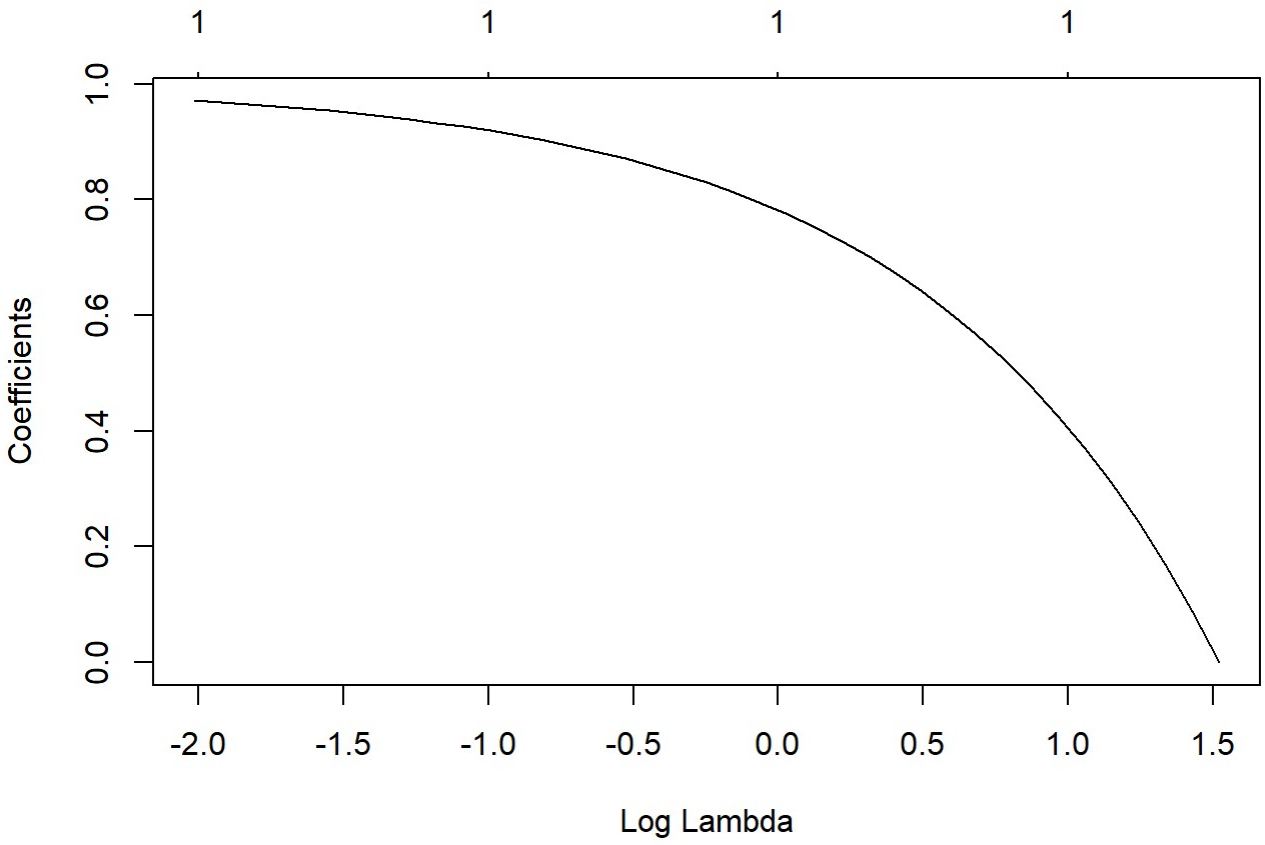



```
#find coefficients of best model
best_model <- glmnet(x, y, alpha = 1, lambda = best_lambda)
coef(best_model)
```

```
## 33 x 1 sparse Matrix of class "dgCMatrix"
##                     s0
## (Intercept) 0.3036083
## school      .
## sex         .
## age         .
## address     .
## famsize     .
## Pstatus     .
## Medu        .
## Fedu        .
## Mjob        .
## Fjob        .
## reason      .
## guardian    .
## traveltime  .
## studytime   .
## failures    .
## schoolsup   .
## famsup      .
## paid        .
## activities  .
## nursery     .
## higher      .
## internet    .
## romantic    .
## famrel      .
## freetime    .
## goout       .
## Dalc        .
## Walc        .
## health      .
## absences    .
## G3          0.9708495
## G           .
```

```
#produce Ridge trace plot
plot(model, xvar = "lambda")
```

```
## Warning in plotCoef(x$beta, lambda = x$lambda, df = x$df, dev = x$dev.ratio, : 1
## or less nonzero coefficients; glmnet plot is not meaningful
```




```
#use fitted best model to make predictions
y_predicted <- predict(model, s = best_lambda, newx = x)

#find SST and SSE
sst <- sum((y - mean(y))^2)
sse <- sum((y_predicted - y)^2)

#find R-Squared
rsq <- 1 - sse/sst
rsq
```

```
## [1] 0.9991502
```