Question 2: University_Data_Analysis

Pavithra Senthilkumar

2024-01-21

1. Loading the University Student Data

```
data<-read.csv("university_sci.csv",sep=",")
```

2. Dimensions of the Dataset and Finding Missing Elements

```
dim(data)
```

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

The Data has a total of 395 records with 33 columns or variables.

To find the missing elements in the dataframe, each vector/column is looked up for any missing values.

Assuming and declaring the initial number of missing values as 0. (missing=0)

```
#length(data) gives the column
#the loop goes over each vector or column.
#the is.na() function returns a logical response, which is summed up for each vector

for(i in 1:length(data))
{
  missing = 0
  missing=sum(is.na(data[i]))
  print(c('no.of missing values in',names(data[i]),missing))
}
```

```
## [1] "no.of missing values in" "University"
## [3] "0"
## [1] "no.of missing values in" "sex"
## [3] "0"
## [1] "no.of missing values in" "age"
## [3] "0"
## [1] "no.of missing values in" "address"
## [3] "0"
## [1] "no.of missing values in" "famsize"
## [3] "0"
## [1] "no.of missing values in" "Pstatus"
## [3] "0"
## [1] "no.of missing values in" "Medu"
## [3] "0"
```

```
## [1] "no.of missing values in" "Fedu"
## [3] "0"
## [1] "no.of missing values in" "Mjob"
## [3] "0"
## [1] "no.of missing values in" "Fjob"
## [3] "0"
## [1] "no.of missing values in" "reason"
## [3] "0"
## [1] "no.of missing values in" "guardian"
## [3] "0"
## [1] "no.of missing values in" "traveltime"
## [3] "0"
## [1] "no.of missing values in" "studytime"
## [3] "0"
## [1] "no.of missing values in" "failures"
## [3] "0"
## [1] "no.of missing values in" "universitysup"
## [3] "0"
## [1] "no.of missing values in" "famsup"
## [3] "0"
## [1] "no.of missing values in" "paid"
## [3] "0"
## [1] "no.of missing values in" "activities"
## [3] "0"
## [1] "no.of missing values in" "nursery"
## [3] "0"
## [1] "no.of missing values in" "higher"
## [3] "0"
## [1] "no.of missing values in" "internet"
## [3] "0"
## [1] "no.of missing values in" "romantic"
## [3] "0"
## [1] "no.of missing values in" "famrel"
## [3] "0"
## [1] "no.of missing values in" "freetime"
## [3] "0"
## [1] "no.of missing values in" "goout"
## [3] "0"
## [1] "no.of missing values in" "Dalc"
## [3] "0"
## [1] "no.of missing values in" "Walc"
## [3] "0"
## [1] "no.of missing values in" "health"
## [3] "0"
## [1] "no.of missing values in" "absences"
## [3] "0"
## [1] "no.of missing values in" "G1"
## [3] "0"
## [1] "no.of missing values in" "G2"
## [3] "0"
## [1] "no.of missing values in" "G3"
## [3] "0"
```

The result shows that there are no missing values in the dataset.

3. Checking the Datatypes of each column

```
#when the function str() is applied to a dataframe it returns all columns's type

str(data)
```

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

The columns are of Integer and Character Types. Integer columns are :

age,medu,fedu,traveltime,studytime,failures,famrel,freetime, goout,Dalc,Walc,health,absences,G1,G2,G3

Character columns are:

University, Sex, Address Famsize, PStatus,MJob, FJob, Reason,Guardian,UniversitySup,familysup,paid,activities,nursery, higher,internet,romantic

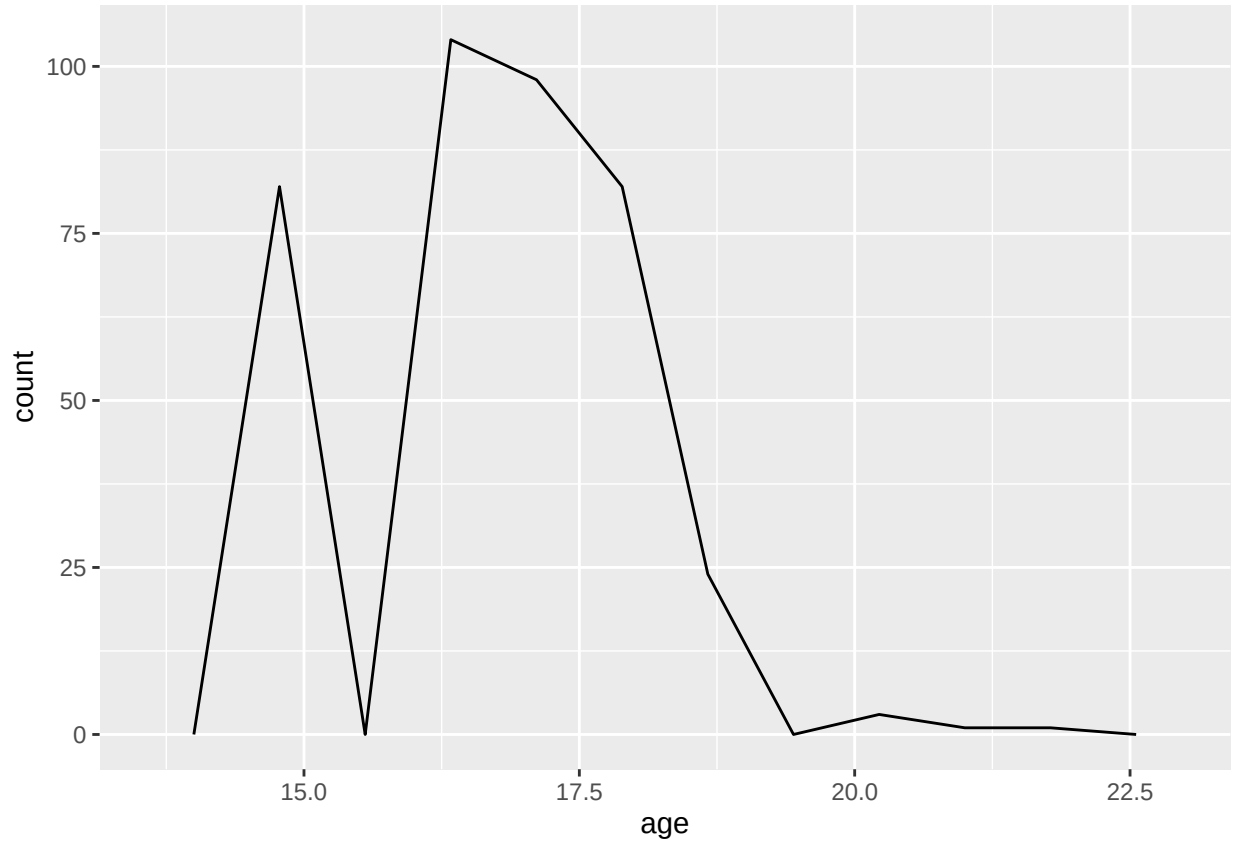
4. Data Distributions

Distribution of Student Age

```r
library(ggplot2)
library(dplyr)
```

```
##
## 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
```

```r
#Age Distribution with frequency plot

data %>%
  ggplot(aes(x=age)) +
  geom_freqpoly(bins=10)
```



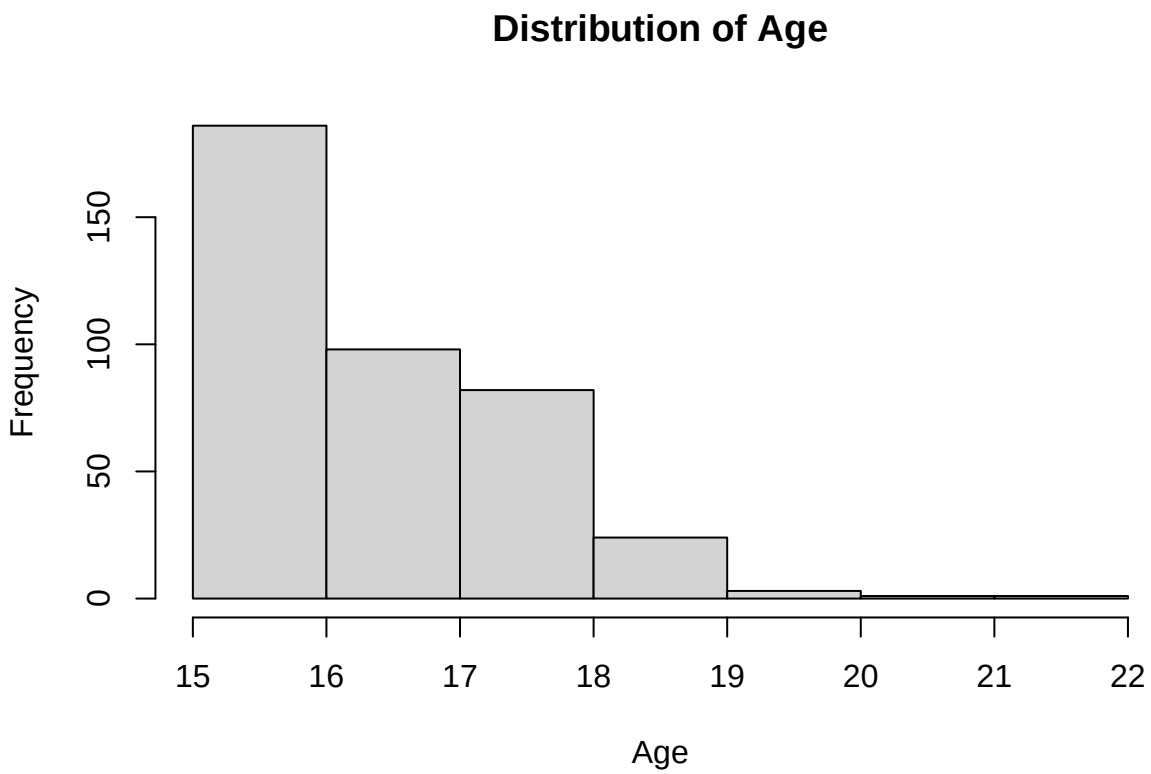

```r
summary(data['age'])
```

```
##       age
```

```
##  Min.   :15.0
##  1st Qu.:16.0
##  Median :17.0
##  Mean   :16.7
##  3rd Qu.:18.0
##  Max.   :22.0
```

To get a better picture, fitting a histogram to see the distribution for the above age data.

```
age<-data$age
hist(age,main="Distribution of Age",xlab ="Age",breaks=6)
```



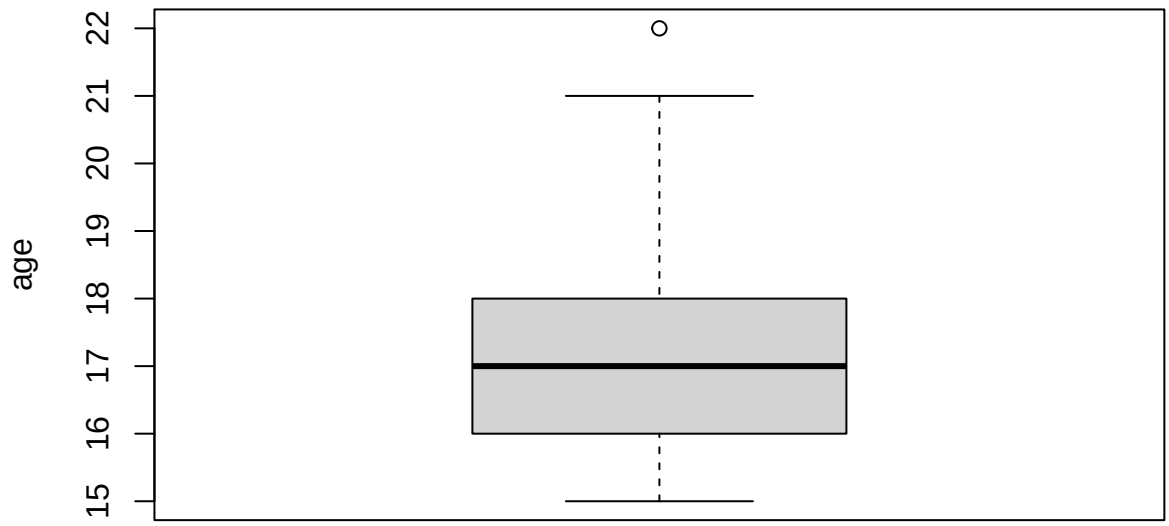
To get to know the percentile distribution,

```
quantile(data[['age']],p=c(0.05,0.25,0.5,0.75,0.95))
```

```
##  5% 25% 50% 75% 95%
##  15  16  17  18  19
```

Plotting boxplots to visualize percentiles better

```
boxplot(data['age'],ylab='age')
```

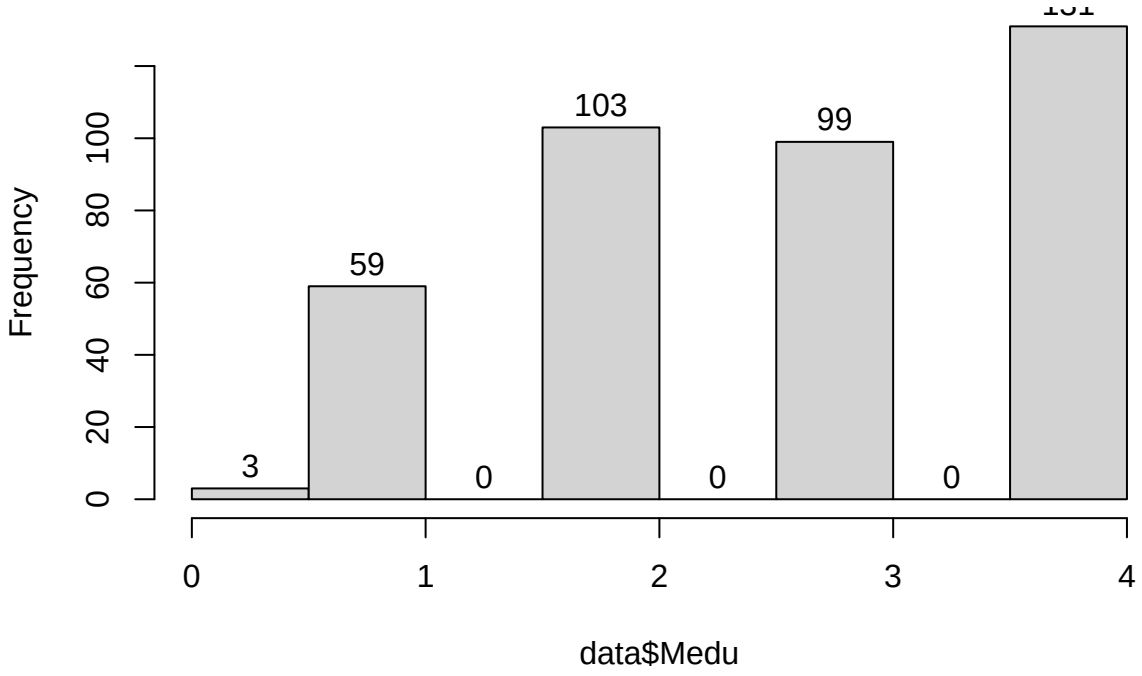


The variable age has an average of 16.7 with maximum age of 22, which is also an outlier. The students are widely spread within the average age of 16 (lower quartile) to 18.0 (upper quartile).

The distribution is more right skewed with ages spread till a maximum of 22, while a majority of the students being around the age 16.7

Distribution of Mother's Education

```
#Mother's Education

hist(data$Medu,main="Mother's Education",labels=TRUE)
```



Percentage of Proportions:

```
f1<-(59/nrow(data))
f2<-(103/nrow(data))
f3<- (99/nrow(data))
f4<-(131/nrow(data))
f_none<-(3/nrow(data))

paste("Primary Education",f1)
```

```
## [1] "Primary Education 0.149367088607595"
```

```
paste("5th to 9th Grade",f2)
```

```
## [1] "5th to 9th Grade 0.260759493670886"
```

```
paste("Secondary Education",f3)
```

```
## [1] "Secondary Education 0.250632911392405"
```

```
paste("Higher Education",f4)
```

```
## [1] "Higher Education 0.331645569620253"
```

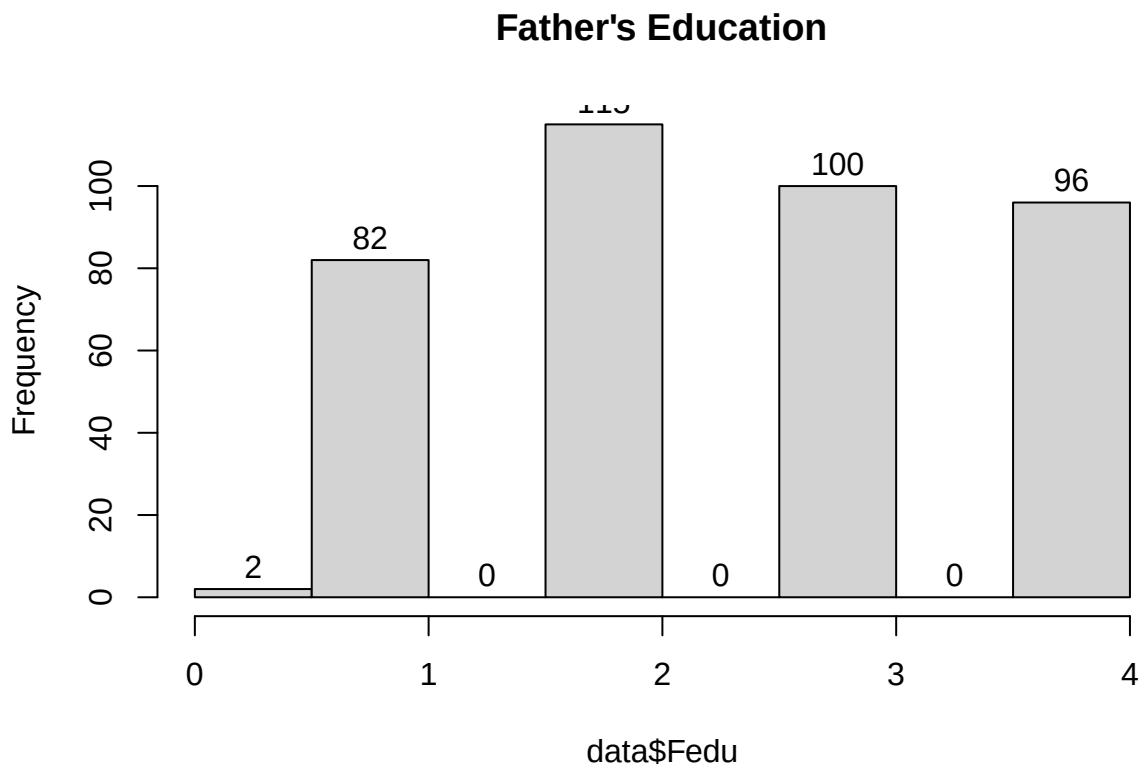
```
paste("None",f_none)
```

```
## [1] "None 0.00759493670886076"
```

It is noted that most of the student's mothers have completed their higher education, contributing to about 33.1% of the whole population. With 26.07% of proportion of 5th to 9th grade Moms, this becomes the second highest qualification.

Distribution of Father's Education

```
#Father's Education

hist(data$Fedu,main="Father's Education",labels=TRUE)
```



Percentage of Proportions:

```
f1<-(82/nrow(data))
f2<-(115/nrow(data))
f3<- (100/nrow(data))
f4<-(96/nrow(data))
f_none<-(2/nrow(data))

paste("Primary Education",f1)
```

```
## [1] "Primary Education 0.207594936708861"
```

```r
paste("5th to 9th Grade",f2)
```

```
## [1] "5th to 9th Grade 0.291139240506329"
```

```r
paste("Secondary Education",f3)
```

```
## [1] "Secondary Education 0.253164556962025"
```

```r
paste("Higher Education",f4)
```

```
## [1] "Higher Education 0.243037974683544"
```

```r
paste("None",f_none)
```

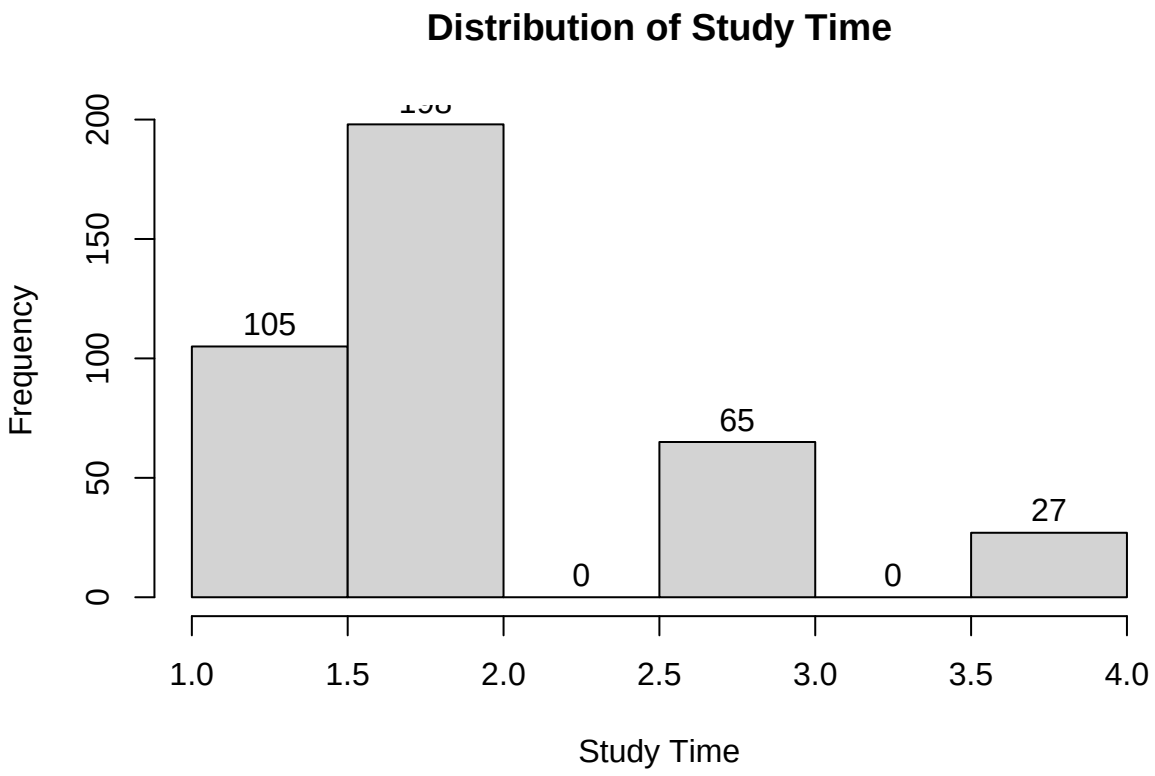
```
## [1] "None 0.00506329113924051"
```

Unlike the mothers, It is noted that most of the student's fathers have completed only until 5th to 9th grade, contributing to the maximum percentage of about 29.11%.

With 25.31% of proportion of Secondary Education, this becomes the second highest qualification in Fathers.

Distribution of Study Time

```r
#Studytime Distribution

study<-data$studytime
hist(study,main="Distribution of Study Time",breaks=6.5,xlab ="Study Time",labels=TRUE)
```



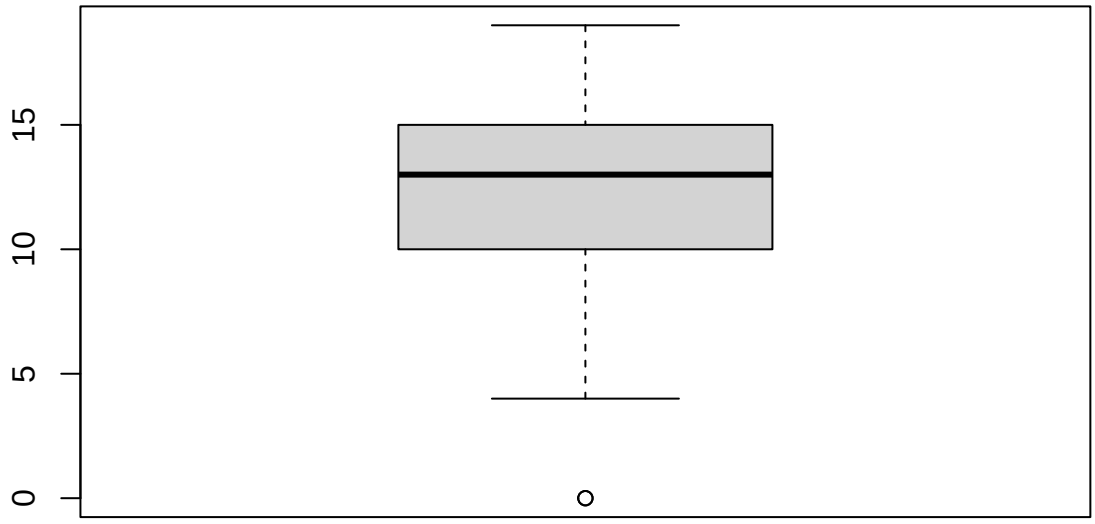
It is noted that most of the students prefer studying for 2 hours. Nearly 6.8% of the students study for 4 hours.

Getting to know the Data Better:

- Is Health affecting Scores?

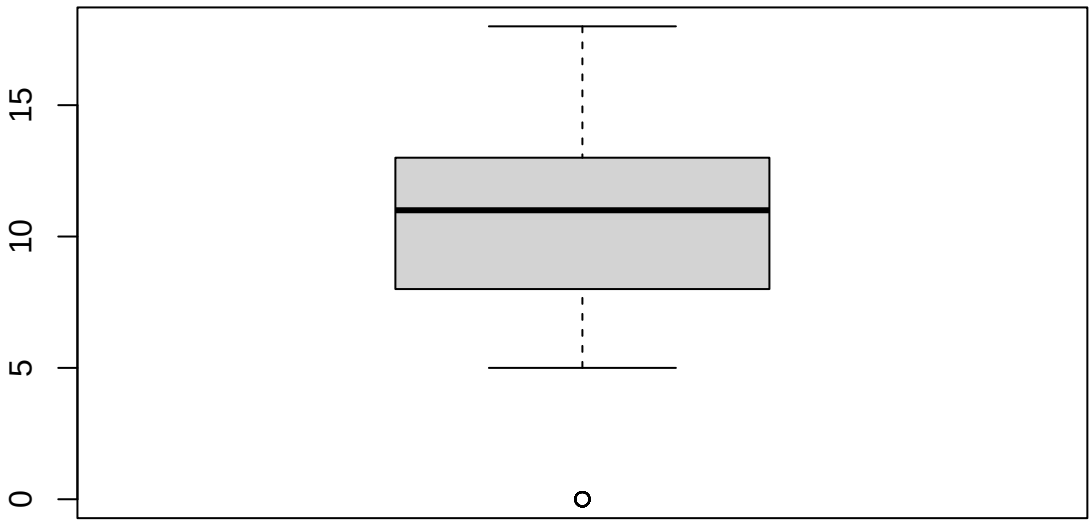
To answer this, plotting a line for the Weakest Health student's final G3 scores to know how the score trend is. Also, plotting the same trend of G3 scores for students in good health condition.

```
Gradehealth1<-data[data$health==1 ,c("health","G3")]

Gradehealth1<-Gradehealth1[order(Gradehealth1$G3,decreasing=FALSE),]



Gradehealth5<-data[data$health==5 ,c("health","G3")]

Gradehealth5<-Gradehealth5[order(Gradehealth5$G3,decreasing=FALSE),]


boxplot(Gradehealth1['G3'], xlab="Final grade Trend with Weak Health")
```



Final grade Trend with Weak Health

```
boxplot(Gradehealth5['G3'], xlab="Final grade Trend with Good Health")
```



Final grade Trend with Good Health

```
summary(Gradehealth1$G3)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   10.00   13.00   11.87   15.00   19.00
```

```
length(Gradehealth1$G3)
```

```
## [1] 47
```

```
summary(Gradehealth5$G3)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0     8.0    11.0    10.4    13.0    18.0
```

```
length(Gradehealth5$G3)
```

```
## [1] 146
```

From the Graph and summary, out of 47 students whose health was bad, the average score of those students is 11.7 in the G3 assesment.

However, for the students with good health condition, the average score is lesser than the average score gained by weak health students. The mean score for good health students is 10.4

The maximum score of the weak health students cohort is 19. Whereas, the maximum score of the good health student cohort is 18.

Therefore, health might not be an impacting factor of the Scores.

- Is past failure related with Grades?

```
failures_3<-data[data$failures==3 ,c("failures","G3")]

failures_0 <-data[data$failures==0,c("failures","G3")]


summary(failures_3$G3)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   7.000   5.688   9.250  10.000
```

```
summary(failures_0$G3)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    9.75   11.00   11.25   14.00   20.00
```

From the summaries, for the students with 3 past failures, the average score turns out to be 5.688 and the average score for the students with no failures in the past is 11.25.

The maximum score for no past failures students is 20. The maximum score for 3 past failures student group is 10.

The past failures might have an influence on grades.

- Are the grades affected due to relationship?

```
rel_yes<-data[data$romantic=="yes", c("romantic","G3")]
rel_no<-data[data$romantic=="no", c("romantic","G3")]

summary(rel_yes$G3)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   8.000  11.000   9.576  13.000  18.000
```

```
length(rel_yes$G3)
```

```
## [1] 132
```

```
summary(rel_no$G3)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    9.00   11.00   10.84   14.00   20.00
```

```
length(rel_no$G3)
```

```
## [1] 263
```

From the spread of the data for both group of students (yes/no in relationship), it is seen that the average scores of students who are not in relationship have higher than those who are not.

This might be a metric that could impact the final scores.

4. Converting 'Guardian' and 'Sex' Variable to Factors

```
cols<-c("sex","guardian")

data[cols]<-lapply(data[cols],factor)

sapply(data,class)
```

```
##    University           sex           age       address       famsize
##   "character"      "factor"     "integer"   "character"   "character"
##       Pstatus          Medu          Fedu          Mjob          Fjob
##   "character"     "integer"     "integer"   "character"   "character"
##        reason      guardian    traveltime     studytime      failures
##   "character"      "factor"     "integer"     "integer"     "integer"
## universitysup        famsup          paid    activities       nursery
##   "character"   "character"   "character"   "character"   "character"
##        higher      internet      romantic        famrel      freetime
##   "character"   "character"   "character"     "integer"     "integer"
##         goout          Dalc          Walc        health      absences
##     "integer"     "integer"     "integer"     "integer"     "integer"
##            G1            G2            G3
##     "integer"     "integer"     "integer"
```

5. How many students are Females? How many student's guardians are not their parents?

```
females<-data[data$sex=="F",c("University","sex")]
length(females$sex)
```

```
## [1] 208
```

There are 208 females.

```
not_parents<-data[data$guardian=="other",c("University","guardian")]
length(not_parents$guardian)
```

```
## [1] 32
```

There are 32 Students whose Guardian is not a parent.

6. For some numerical variables that are integer data types, they may make more sense if they are converted to factors. Give an example and explain why. Do the conversion and show the factor levels.

- If the categorical variable is in the integer format, it must be converted to factors if the number of categories are countable/finite.
- For example, the variable Mother's Education, Father's Education represent the levels of education they have. They represent levels (1-5) with highest as 5 and 1 as lowest. Which when assumed as numeric, they might be misinterpreted or could be taken as numeric weights instead of levels or categories, when proceeding with models. Hence, to avoid such confusions, it is good to convert the categorical numerical variables to factors.

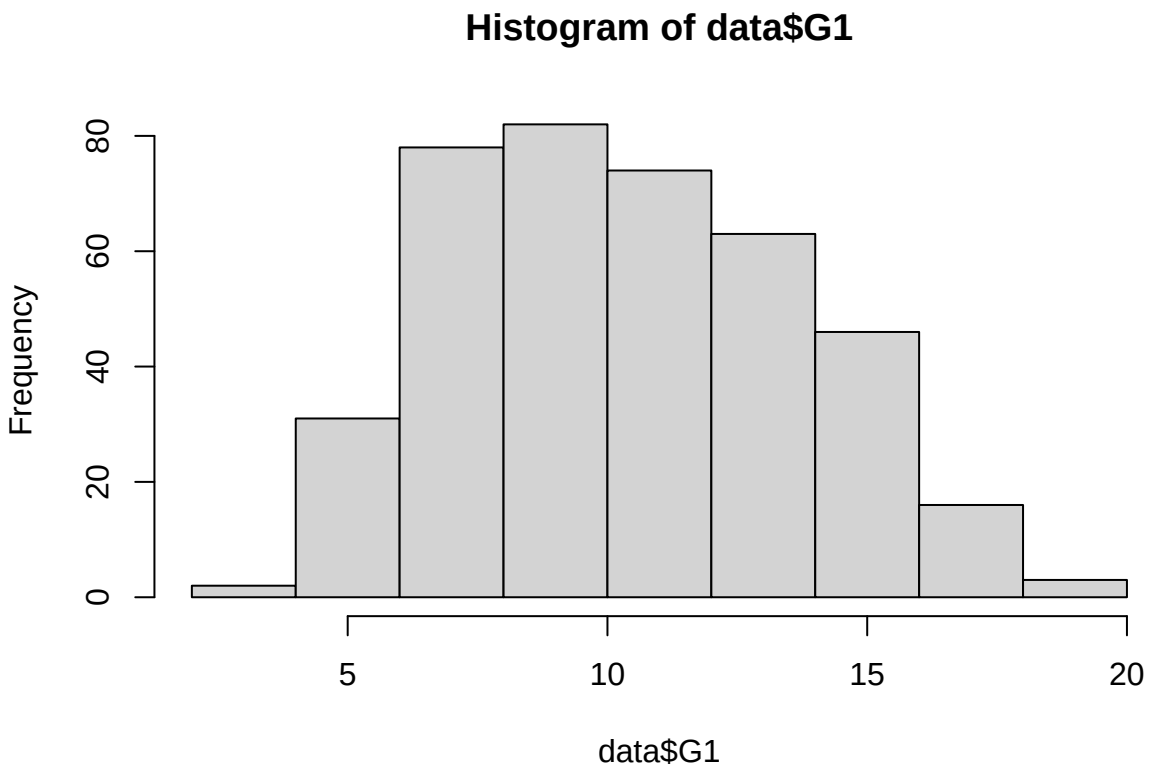
Converting Categorical numerics to factors.

```
cols<-c("Medu","Fedu","traveltime","studytime","failures",
        "famrel","freetime","goout","Dalc","Walc","health")

data[cols]<-lapply(data[cols],factor)
sapply(data,class)
```

```
##    University           sex           age       address       famsize
##   "character"      "factor"     "integer"   "character"   "character"
##       Pstatus          Medu          Fedu          Mjob          Fjob
##   "character"      "factor"      "factor"   "character"   "character"
##        reason      guardian    traveltime     studytime      failures
##   "character"      "factor"      "factor"      "factor"      "factor"
## universitysup        famsup          paid    activities       nursery
##   "character"   "character"   "character"   "character"   "character"
##        higher      internet      romantic        famrel      freetime
##   "character"   "character"   "character"      "factor"      "factor"
##         goout          Dalc          Walc        health      absences
##      "factor"      "factor"      "factor"      "factor"     "integer"
##            G1            G2            G3
##     "integer"     "integer"     "integer"
```

7. G1 - G3 represent students' grades. Use hist() to get an idea the grade distributions. What do you observe in each distribution? and compare the three.

```
hist(data$G1)
```

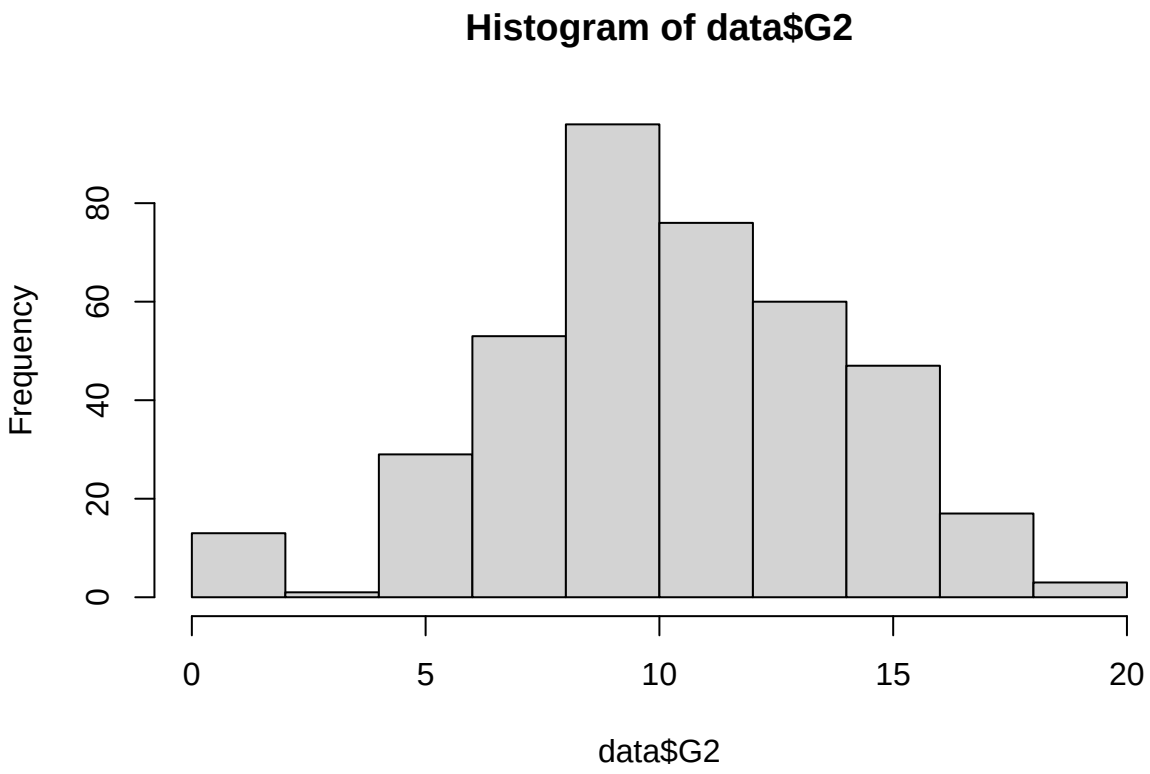



```
summary(data$G1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    3.00    8.00   11.00   10.91   13.00   19.00
```

The Grade 1 distribution resembles a normal distribution with mean and median 10.91 and 11.00 respectively. The distribution is skewed a little to its right, which means few students grades are spread widely higher than the mean.

```
hist(data$G2)
```

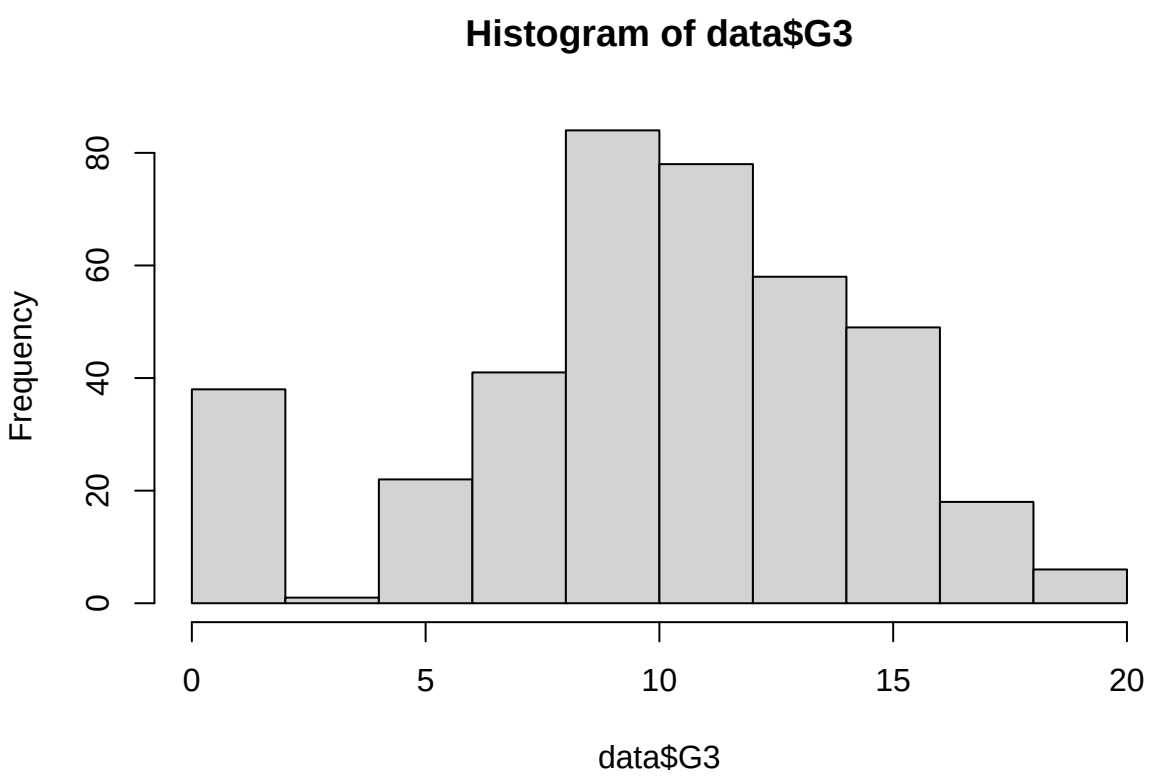



```
summary(data$G2)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    9.00   11.00   10.71   13.00   19.00
```

The grade 2 also shows a normal distribution with median 11 and mean 10.71 respectively. Almost no student got grades between 1 to 5, whereas more than 15 students got 0. This makes the distribution a little uneven to its left.

```
hist(data$G3)
```




```
summary(data$G3)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    8.00   11.00   10.42   14.00   20.00
```

Unlike the G2 and G1, the distribution of G3 follows a normal distribution with two peaks (bimodal), with mean 10.42 and median 11. Nearly 40 students got 0, and no one got between 1-5.

For all the three Grades(G1,G2,G3), it is noted that the median grade remained the same, i.e, 11. All three distribution follows a normal curve. They also have almost near mean values.

8. Using logical expressions and subsetting, calculate the fraction of G3 that are less than its median.

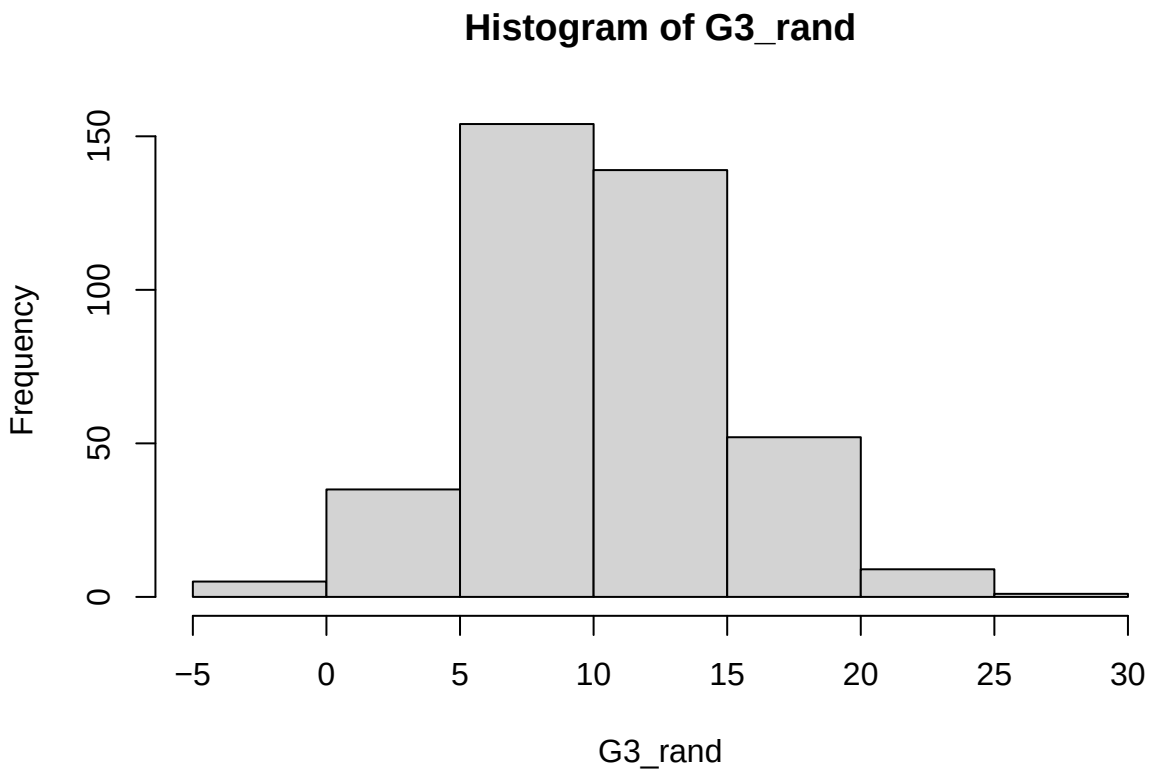
```
data_G3_frac <-data[data$G3<11.00 , c("University","G1","G2","G3")]
new_len_g3<-length(data_G3_frac$G3)
old_len_g3<-nrow(data)

fract_G3<-(new_len_g3/old_len_g3)*100
print(fract_G3)
```

```
## [1] 47.08861
```

About 47.08% of the population(0.47 in fraction), are less than the median score for G3.

9. Create a random normal vector of the same length as G3 using the mean, standard deviation and length of G3. Call this new vector G3_rand. Visually verify that it follows a distribution similar to G3.

```
G3_rand<-rnorm(n=395,mean=10.42,sd=4.58)

hist(G3_rand)
```




```
sd(data$G3)
```

```
## [1] 4.581443
```

```
mean(G3_rand)
```

```
## [1] 10.38687
```

```
sd(G3_rand)
```

```
## [1] 4.572415
```

The random vector of G3, follows a more accurate normal distribution unlike that of G3.

10.Add 'grade_level' to the dataframe as a new column. If a student's final grade (G3) is larger than 14, or between 10 to 14, or less than 10, assign them grade level: A, B, and C respectively. (Hint: ifelse() may be useful

```
data$grade_level<-with(data,ifelse(G3 > 14,"A",
                                                    ifelse(G3<10,"C","B")))

head(data)
```

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

11. Now subset the dataframe according to the 'grade_level' column. (Hint: you can use df[] or subset().) You should have three dataframes. Make sure to give your new data objects meaningful names as discussed in class.

```
data_grade_A<-data[data$grade_level=="A",]

data_grade_B <-data[data$grade_level=="B",]

data_grade_C<-data[data$grade_level=="C",]

dim(data_grade_A)
```

```
## [1] 73 34
```

```
dim(data_grade_B)
```

```
## [1] 192  34
```

```r
dim(data_grade_C)
```

```
## [1] 130  34
```