Bharath Goud Nadimpally

Student Performance Data & Student Alcohol Consumption

Problem A ::Data gathering and integration

```
library(readr)
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
```

```
frameA <- read.csv("C:/Users/CDMStudent14/Documents/R/Assignment05/student_data1.csv")
frameB <- read.csv("C:/Users/CDMStudent14/Documents/R/Assignment05/student_data2.csv")
student_data <- rbind(frameA, frameB)
str(student_data)
```

```
## 'data.frame':    790 obs. of  33 variables:
##  $ school    : chr  "GP" "GP" "GP" "GP" ...
##  $ sex       : chr  "F" "F" "F" "F" ...
##  $ age       : int  18 17 15 15 16 16 16 17 15 15 ...
##  $ address   : chr  "U" "U" "U" "U" ...
##  $ famsize   : chr  "GT3" "GT3" "LE3" "GT3" ...
##  $ Pstatus   : chr  "A" "T" "T" "T" ...
##  $ Medu      : int  4 1 1 4 3 4 2 4 3 3 ...
##  $ Fedu      : int  4 1 1 2 3 3 2 4 2 4 ...
##  $ Mjob      : chr  "at_home" "at_home" "at_home" "health" ...
##  $ Fjob      : chr  "teacher" "other" "other" "services" ...
##  $ reason    : chr  "course" "course" "other" "home" ...
##  $ guardian  : chr  "mother" "father" "mother" "mother" ...
##  $ 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 : 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 ...
```

```
summary(student_data)
```

```
##     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
```

```
head(student_data)
```

```
##   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
```

1. school - student's school (binary: 'GP' - Gabriel Pereira or 'MS' - Mousinho da Silveira)
2. sex - student's sex (binary: 'F' - female or 'M' - male)
3. age - student's age (numeric: from 15 to 22)
4. address - student's home address type (binary: 'U' - urban or 'R' - rural)
5. famsize - family size (binary: 'LE3' - less or equal to 3 or 'GT3' - greater than 3)
6. Pstatus - parent's cohabitation status (binary: 'T' - living together or 'A' - apart)
7. Medu - mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
8. Fedu - father's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
9. Mjob - mother's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
10. Fjob - father's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
11. reason - reason to choose this school (nominal: close to 'home', school 'reputation', 'course' preference or 'other')
12. guardian - student's guardian (nominal: 'mother', 'father' or 'other')
13. 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. These grades are related with the course subject, Math or Portuguese:

    G1 - first period grade (numeric: from 0 to 20) G2 - second period grade (numeric: from 0 to 20) G3 - final grade (numeric: from 0 to 20, output target)

Problem B :: Data Exploration

```
str(student_data)
```

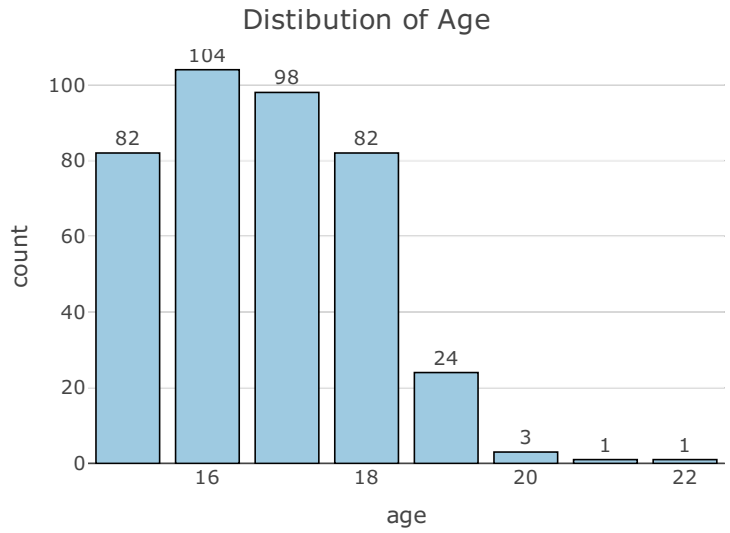
```
## 'data.frame':    395 obs. of  33 variables:
##  $ school    : chr  "GP" "GP" "GP" "GP" ...
##  $ sex       : chr  "F" "F" "F" "F" ...
##  $ age       : int  18 17 15 15 16 16 16 17 15 15 ...
##  $ address   : chr  "U" "U" "U" "U" ...
##  $ famsize   : chr  "GT3" "GT3" "LE3" "GT3" ...
##  $ Pstatus   : chr  "A" "T" "T" "T" ...
##  $ Medu      : int  4 1 1 4 3 4 2 4 3 3 ...
##  $ Fedu      : int  4 1 1 2 3 3 2 4 2 4 ...
##  $ Mjob      : chr  "at_home" "at_home" "at_home" "health" ...
##  $ Fjob      : chr  "teacher" "other" "other" "services" ...
##  $ reason    : chr  "course" "course" "other" "home" ...
##  $ guardian  : chr  "mother" "father" "mother" "mother" ...
##  $ 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 : 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 ...
```

```
library(plotly)
```

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

```
##
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':
##
##     last_plot

## The following object is masked from 'package:stats':
##
##     filter

## The following object is masked from 'package:graphics':
##
##     layout
```

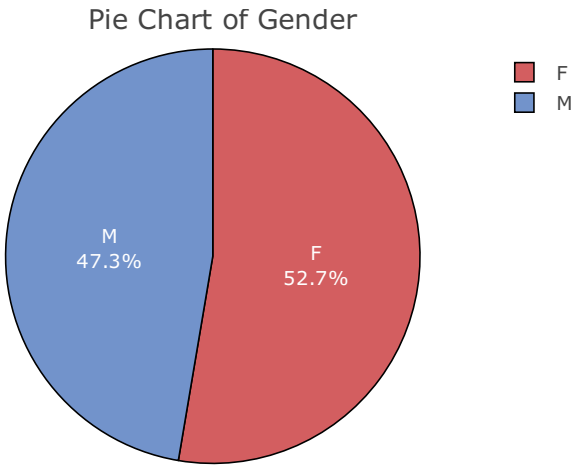
```
student_data %>%
  group_by(age)%>%
  summarize(count = n()) %>%
  plot_ly(x =~age, y=~count, type = 'bar',
        text = ~count,
        textposition = 'outside',
        marker = list(color = 'rgb(158,202,225)',
          line = list(color = 'black',
                width = 1.0))) %>%
  layout(title = 'Distibution of Age')
```



```
student_data_gender_Stat <- student_data %>%
  group_by(sex) %>%
  summarise(count = n(),
            percentage = round((n()/ nrow(student_data)), digits = 4))
student_data_gender_Stat
```

```
## # A tibble: 2 x 3
##   sex   count percentage
##   <chr> <int>      <dbl>
## 1 F       208      0.527
## 2 M       187      0.473
```

```
colors <- c('rgb(211,94,96)','rgb(114,147,203)')
Gender_PieChart <- plot_ly(data = student_data_gender_Stat, labels = ~sex, values = ~percentage,
                type = 'pie', sort = F,
                textposition = 'inside',
                textinfo = 'label+percent',
                insidetextfont = list(color = 'White'),
                hoverinfo = 'text',
                text = ~count,
                marker = list(colors = colors,
                line = list(color = 'Black', width = 1)),
                showlegend = TRUE)
Gender_PieChart <- Gender_PieChart %>% layout(title = 'Pie Chart of Gender')
Gender_PieChart
```



```
student_data$Dalc <- as.factor(student_data$Dalc)
plyr::mapvalues
```

```
## function (x, from, to, warn_missing = TRUE)
## {
##     if (length(from) != length(to)) {
##         stop("`from` and `to` vectors are not the same length.")
##     }
##     if (!is.atomic(x) && !is.null(x)) {
##         stop("`x` must be an atomic vector or NULL.")
##     }
##     if (is.factor(x)) {
##         levels(x) <- mapvalues(levels(x), from, to, warn_missing)
##         return(x)
##     }
##     mapidx <- match(x, from)
##     mapidxNA <- is.na(mapidx)
##     from_found <- sort(unique(mapidx))
##     if (warn_missing && length(from_found) != length(from)) {
##         message("The following `from` values were not present in `x`: ",
##             paste(from[!(1:length(from) %in% from_found)], collapse = ", "))
##     }
##     x[!mapidxNA] <- to[mapidx[!mapidxNA]]
##     x
## }
## <bytecode: 0x00000000294e0288>
## <environment: namespace:plyr>
```

```
student_data$Dalc <- plyr::mapvalues(student_data$Dalc,
                                from = 1:5,
                                to = c("Very Low", "Low", "Medium", "High", "Very High"))

student_data$Walc <- as.factor(student_data$Walc)
student_data$Walc <- plyr::mapvalues(student_data$Walc,
                                from = 1:5,
                                to = c("Very Low", "Low", "Medium", "High", "Very High"))

alcohol.d <- as.data.frame(table(frameB$Dalc))
par.d <- as.numeric(alcohol.d$Freq)
names(par.d) <- alcohol.d$Var1
par.d <- round(par.d/10)

waffle.col <- c("#00d27f","#adff00","#f9d62e","#fc913a","#ff4e50")
library(waffle)
c1 <- waffle(par.d, rows=5,
             #use_glyph="glass",
             size=2,
             title = "Workday alcohol consumption among students",
             glyph_size=8,
             xlab="1 glass == 10 students",
             colors=waffle.col,
             legend_pos= "top"
             )
```

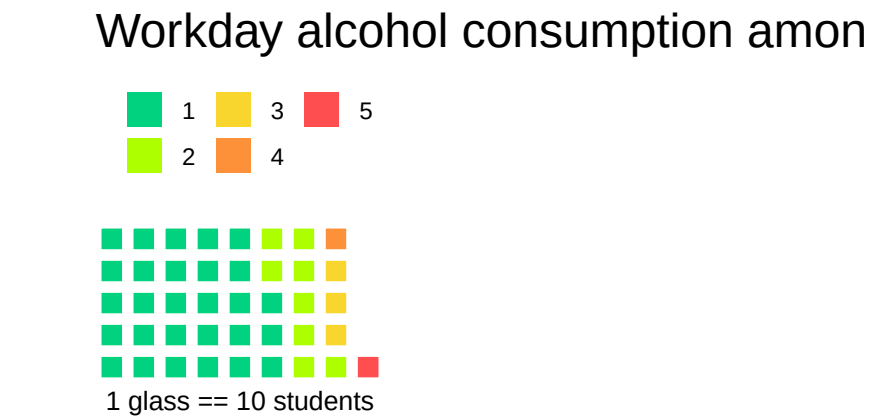
```
alcohol.w <- as.data.frame(table(student_data$Walc))
par.w <- as.numeric(alcohol.w$Freq)
names(par.w) <- alcohol.w$Var1
par.w <- round(par.w/10)

c2 <- waffle(par.w, rows=5,
             #use_glyph="glass",
             size=2,
             title = "Weekend alcohol consumption among students",
             glyph_size=8,
             xlab="1 glass == 10 students",
             colors=waffle.col,
             legend_pos= "top"
             )
require("ggplot2")
require("gridExtra")
```

```
## Loading required package: gridExtra
```

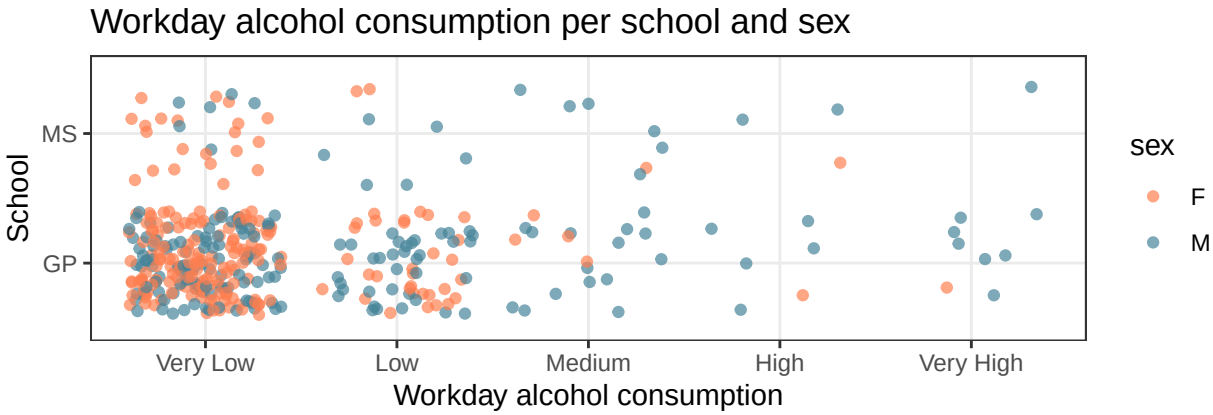
```
##
## Attaching package: 'gridExtra'
```

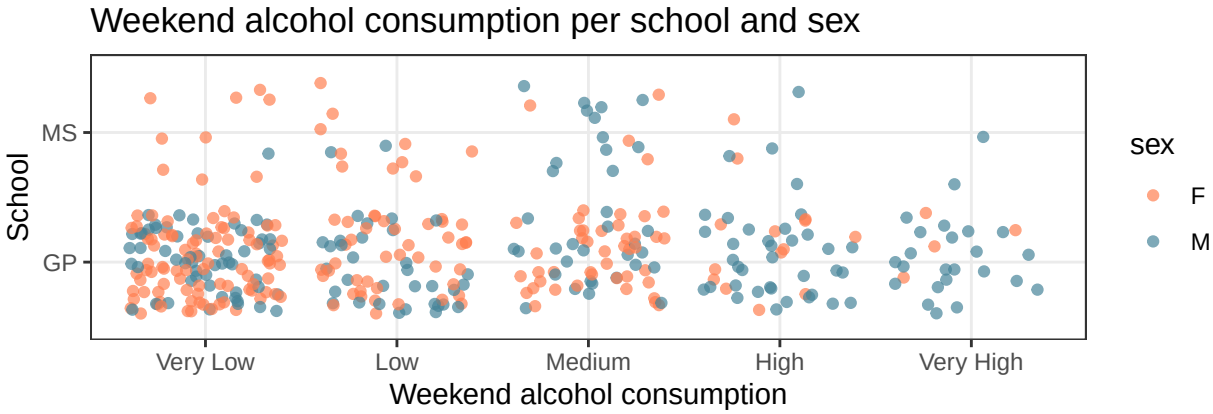
```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
grid.arrange(c1,c2, nrow=2)
```



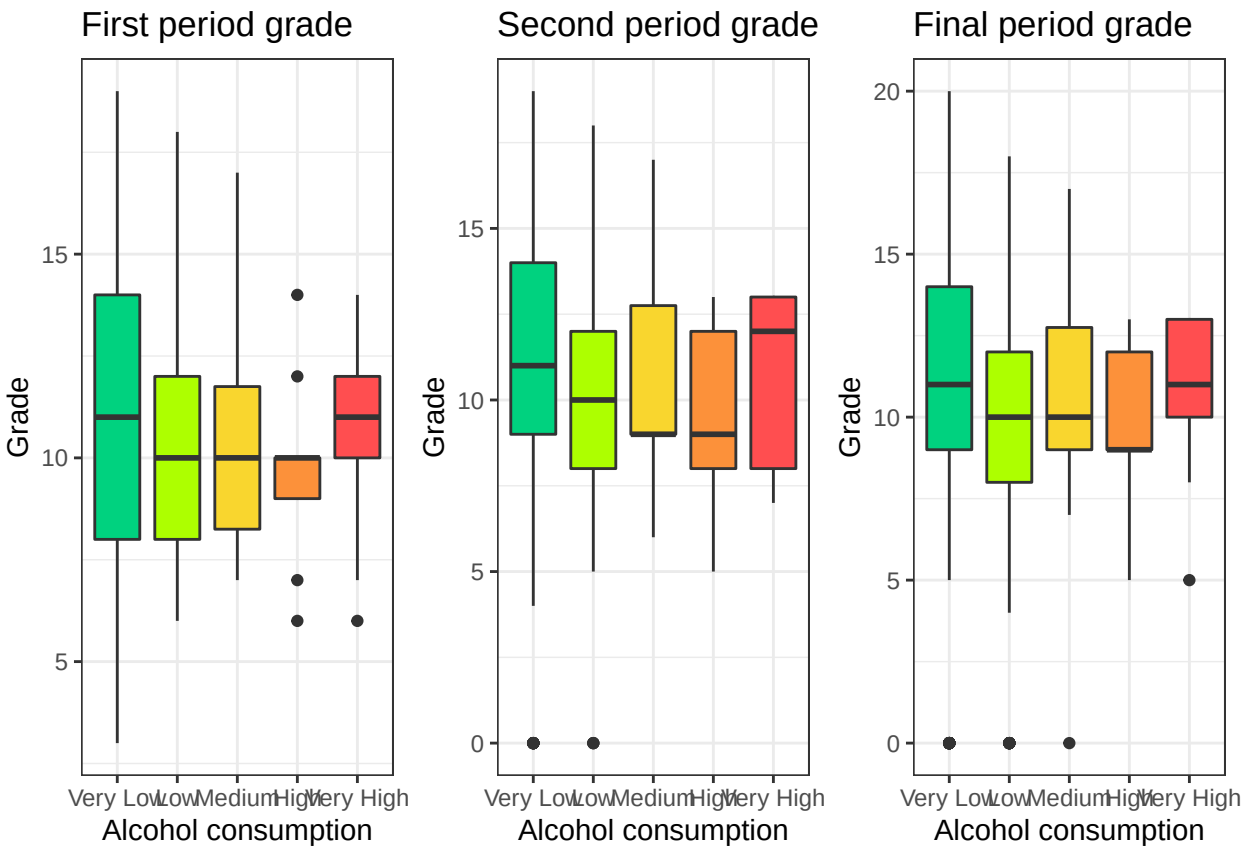

```
c3 <- ggplot(student_data, aes(x=Dalc, y=school, color=sex))+
      geom_jitter(alpha=0.7)+
       scale_colour_manual(values=c("#ff7f50", "#468499"))+
      theme_bw()+
      xlab("Workday alcohol consumption")+
      ylab("School")+
      ggtitle("Workday alcohol consumption per school and sex")


c4 <- ggplot(student_data, aes(x=Walc, y=school, color=sex))+
      geom_jitter(alpha=0.7)+
       scale_colour_manual(values=c("#ff7f50", "#468499"))+
      theme_bw()+
      xlab("Weekend alcohol consumption")+
      ylab("School")+
      ggtitle("Weekend alcohol consumption per school and sex")

grid.arrange(c3,c4, nrow=2)
```





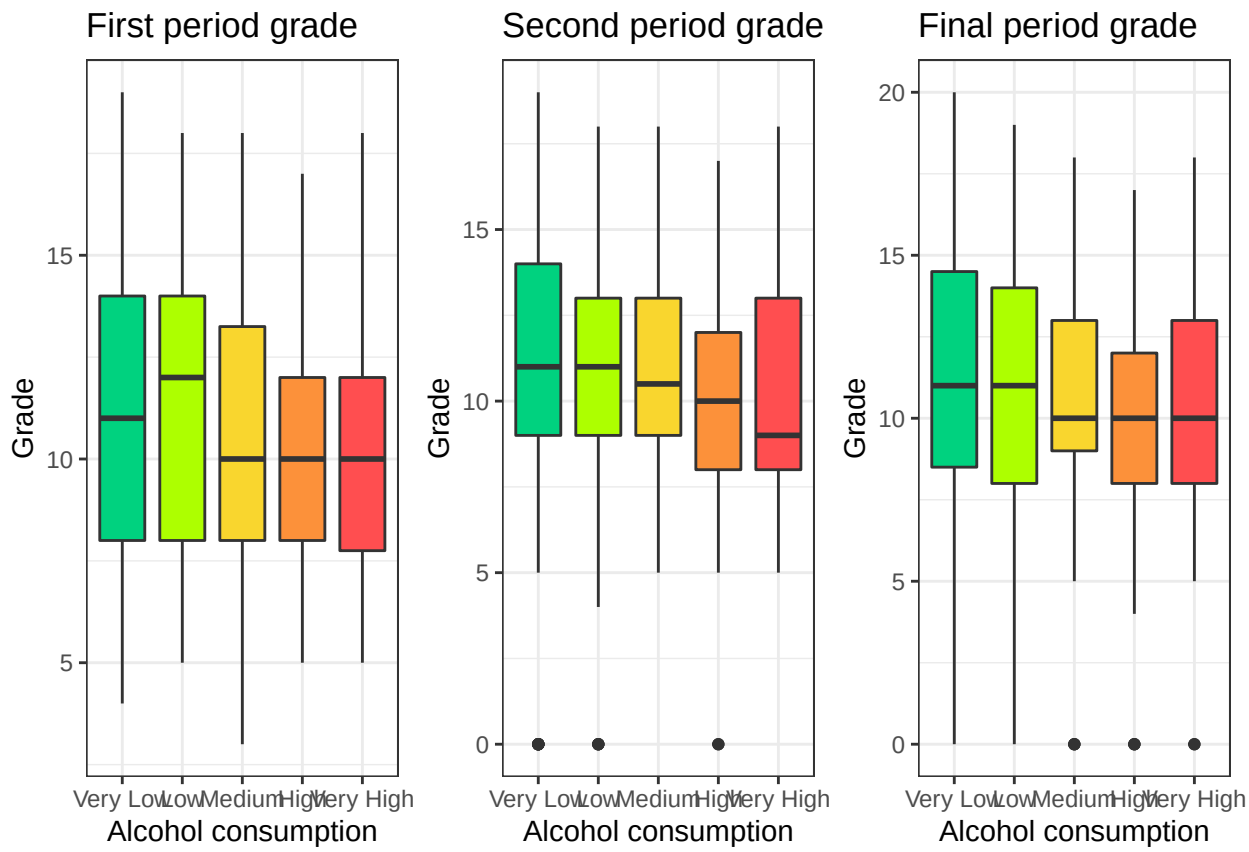

```
#workday
c5 <- ggplot(student_data, aes(x=Dalc, y=G1, fill=Dalc))+
      geom_boxplot()+
      theme_bw()+
      theme(legend.position="none")+
      scale_fill_manual(values=waffle.col)+
      xlab("Alcohol consumption")+
      ylab("Grade")+
      ggtitle("First period grade")

c6 <- ggplot(student_data, aes(x=Dalc, y=G2, fill=Dalc))+
      geom_boxplot()+
      theme_bw()+
      theme(legend.position="none")+
      scale_fill_manual(values=waffle.col)+
      xlab("Alcohol consumption")+
      ylab("Grade")+
      ggtitle("Second period grade")

c7 <- ggplot(student_data, aes(x=Dalc, y=G3, fill=Dalc))+
      geom_boxplot()+
      theme_bw()+
      theme(legend.position="none")+
      scale_fill_manual(values=waffle.col)+
      xlab("Alcohol consumption")+
      ylab("Grade")+
```

```
      ggtitle("Final period grade")

grid.arrange(c5,c6,c7,ncol=3)
```

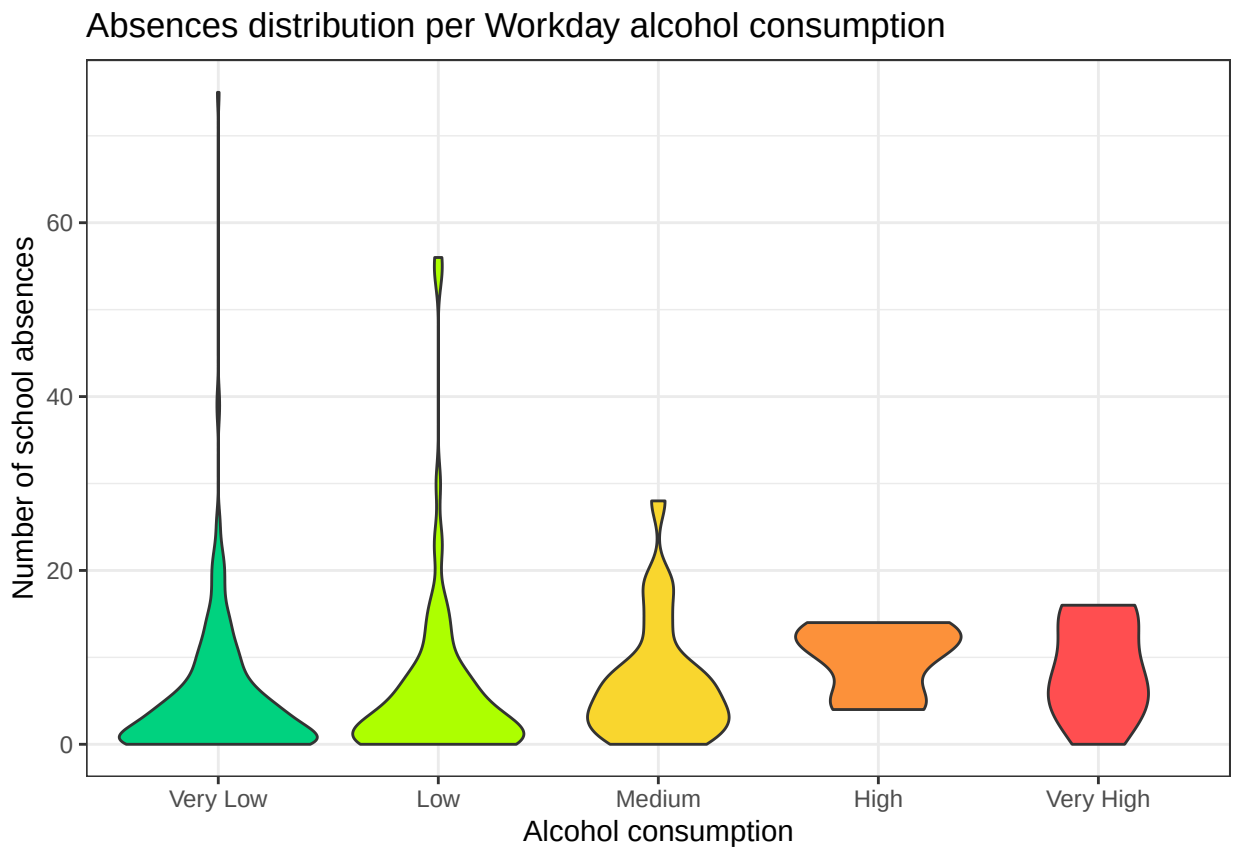



```
#weekend
c8 <- ggplot(student_data, aes(x=Walc, y=G1, fill=Walc))+
      geom_boxplot()+
      theme_bw()+
      theme(legend.position="none")+
      scale_fill_manual(values=waffle.col)+
      xlab("Alcohol consumption")+
      ylab("Grade")+
      ggtitle("First period grade")

c9 <- ggplot(student_data, aes(x=Walc, y=G2, fill=Walc))+
      geom_boxplot()+
      theme_bw()+
      theme(legend.position="none")+
      scale_fill_manual(values=waffle.col)+
      xlab("Alcohol consumption")+
      ylab("Grade")+
      ggtitle("Second period grade")

c10 <- ggplot(student_data, aes(x=Walc, y=G3, fill=Walc))+
      geom_boxplot()+
```

```
      theme_bw()+
      theme(legend.position="none")+
      scale_fill_manual(values=waffle.col)+
      xlab("Alcohol consumption")+
      ylab("Grade")+
      ggtitle("Final period grade")

grid.arrange(c8,c9,c10,ncol=3)
```

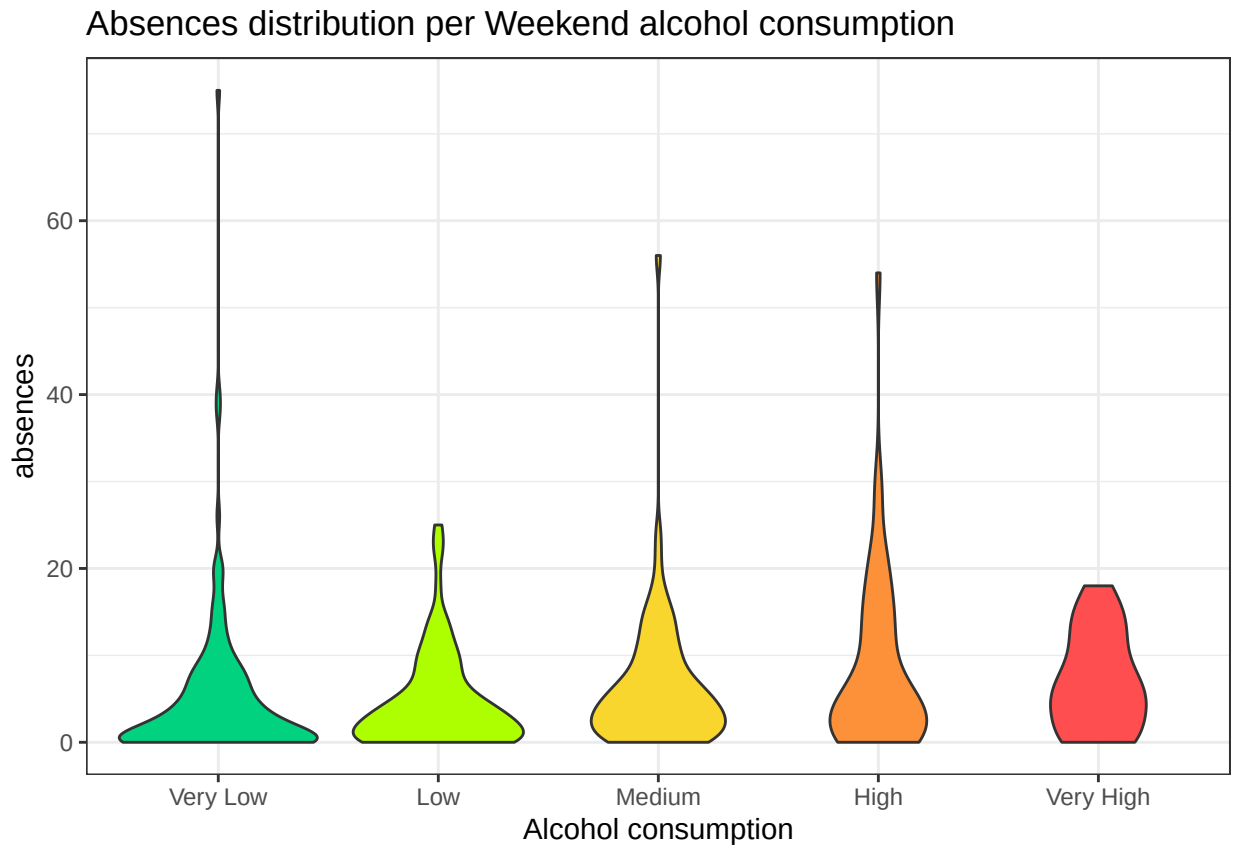



```
ggplot(student_data, aes(x=Dalc, y=absences, fill=Dalc))+
      geom_violin()+
      scale_fill_manual(values = waffle.col)+
      theme_bw()+
      theme(legend.position="none")+
      ggtitle("Absences distribution per Workday alcohol consumption")+
      xlab("Alcohol consumption")+
      ylab("Number of school absences")
```



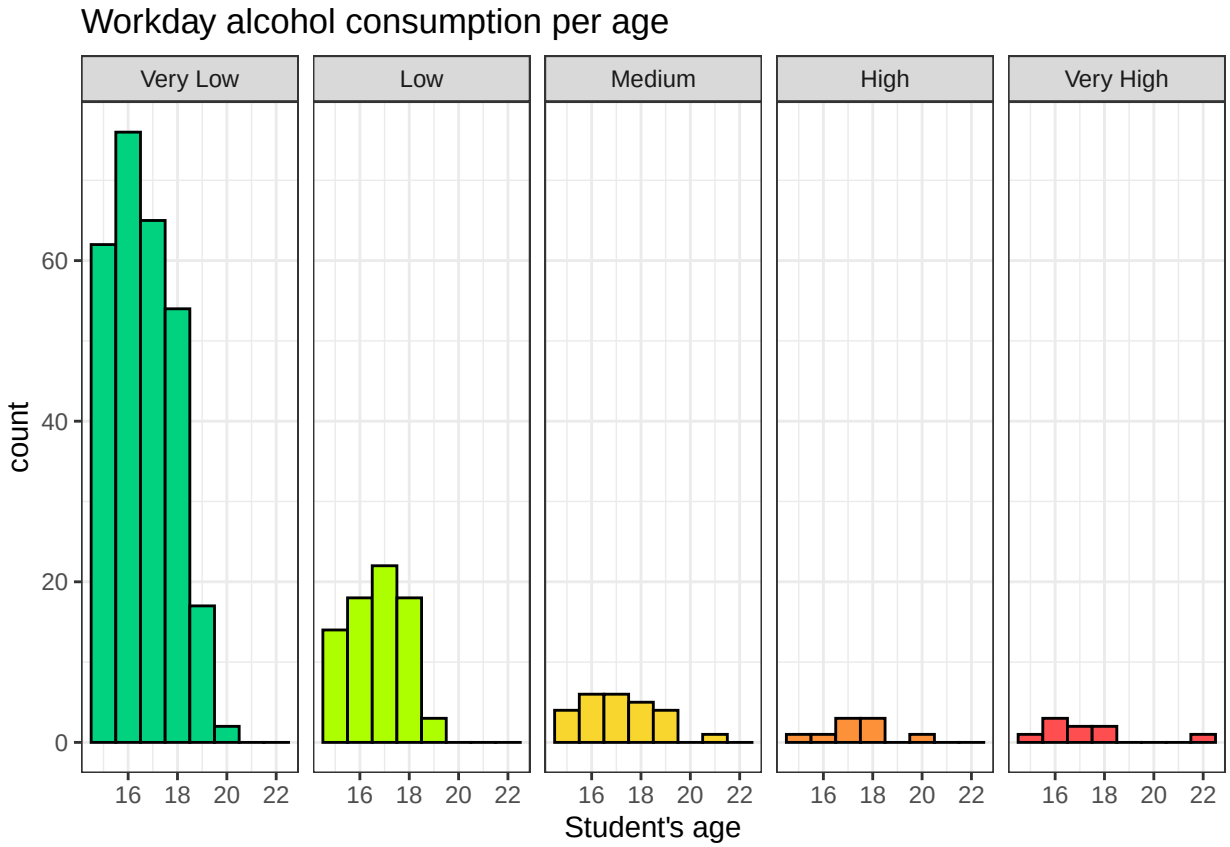
The very high alcohol consumption category has an interesting shape as it expends while others tend to decrease. We can also notice it is nicely shaped as a bottle

```
ggplot(student_data, aes(x=Walc, y=absences, fill=Walc))+
      geom_violin()+
      scale_fill_manual(values = waffle.col)+
      theme_bw()+
      theme(legend.position="none")+
      ggtitle("Absences distribution per Weekend alcohol consumption")+
      xlab("Alcohol consumption")
```



Alcohol consumption and student's age

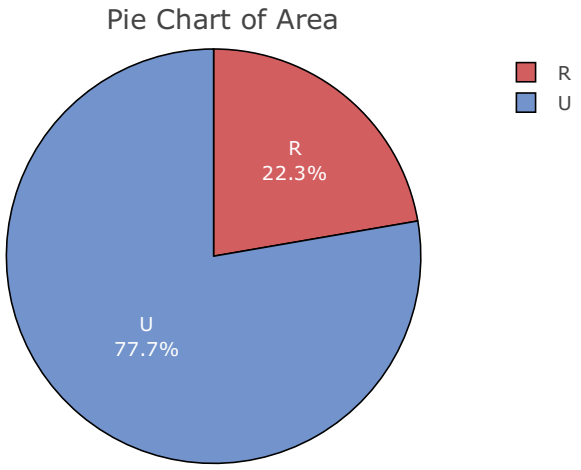
```
ggplot(student_data, aes(x=age, fill=Dalc))+
      geom_histogram(binwidth=1, colour="black")+
      facet_grid(~Dalc)+
      scale_fill_manual(values= waffle.col)+
      theme_bw()+
      theme(legend.position="none")+
      ggtitle("Workday alcohol consumption per age")+
      xlab("Student's age")
```




```
student_data_area_Stat <- student_data %>%
  group_by(address) %>%
  summarise(count = n(),
            percentage = round((n()/ nrow(student_data)), digits = 4))
student_data_area_Stat
```

```
## # A tibble: 2 x 3
##   address count percentage
##   <chr>   <int>      <dbl>
## 1 R          88      0.223
## 2 U         307      0.777
```

```
colors <- c('rgb(211,94,96)','rgb(114,147,203)')
Area_PieChart <- plot_ly(data = student_data_area_Stat, labels = ~address, values = ~percentage,
                type = 'pie', sort = F,
                textposition = 'inside',
                textinfo = 'label+percent',
                insidetextfont = list(color = 'White'),
                hoverinfo = 'text',
                text = ~count,
                marker = list(colors = colors,
                line = list(color = 'Black', width = 1)),
                showlegend = TRUE)
Area_PieChart <- Area_PieChart %>% layout(title = 'Pie Chart of Area')
Area_PieChart
```

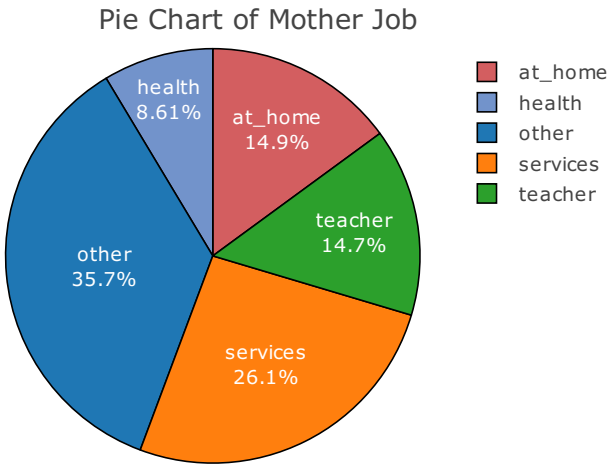



```
student_data_Mjob_Stat <- student_data %>%
  group_by(Mjob) %>%
  summarise(count = n(),
            percentage = round((n()/ nrow(student_data)), digits = 4))
student_data_Mjob_Stat
```

```
## # A tibble: 5 x 3
##   Mjob     count percentage
##   <chr>    <int>      <dbl>
## 1 at_home     59     0.149
## 2 health      34     0.0861
## 3 other      141     0.357
## 4 services   103     0.261
## 5 teacher     58     0.147
```

```
colors <- c('rgb(211,94,96)','rgb(114,147,203)')
Mjob_PieChart <- plot_ly(data = student_data_Mjob_Stat, labels = ~Mjob, values = ~percentage,
                type = 'pie', sort = F,
                textposition = 'inside',
                textinfo = 'label+percent',
                insidetextfont = list(color = 'White'),
                hoverinfo = 'text',
                text = ~count,
                marker = list(colors = colors,
                line = list(color = 'Black', width = 1)),
```

```
                showlegend = TRUE)
Mjob_PieChart <- Mjob_PieChart %>% layout(title = 'Pie Chart of Mother Job')
Mjob_PieChart
```

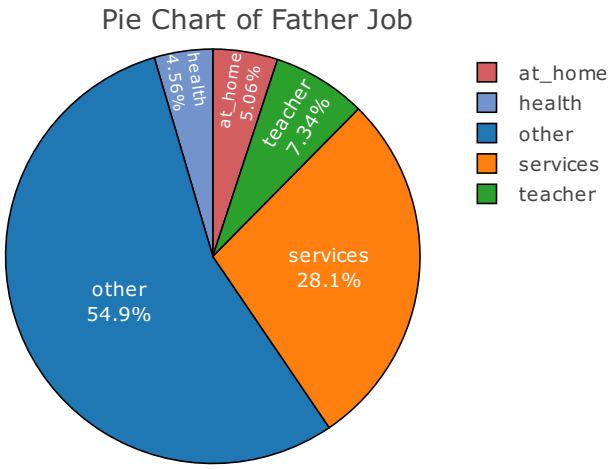



```
student_data_Fjob_Stat <- student_data %>%
  group_by(Fjob) %>%
  summarise(count = n(),
            percentage = round((n()/ nrow(student_data)), digits = 4))
student_data_Fjob_Stat
```

```
## # A tibble: 5 x 3
##   Fjob     count percentage
##   <chr>    <int>      <dbl>
## 1 at_home     20     0.0506
## 2 health      18     0.0456
## 3 other      217     0.549
## 4 services   111     0.281
## 5 teacher     29     0.0734
```

```
colors <- c('rgb(211,94,96)','rgb(114,147,203)')
Fjob_PieChart <- plot_ly(data = student_data_Fjob_Stat, labels = ~Fjob, values = ~percentage,
                type = 'pie', sort = F,
                textposition = 'inside',
                textinfo = 'label+percent',
```

```
                insidetextfont = list(color = 'White'),
                hoverinfo = 'text',
                text = ~count,
                marker = list(colors = colors,
                line = list(color = 'Black', width = 1)),
                showlegend = TRUE)
Fjob_PieChart <- Fjob_PieChart %>% layout(title = 'Pie Chart of Father Job')
Fjob_PieChart
```

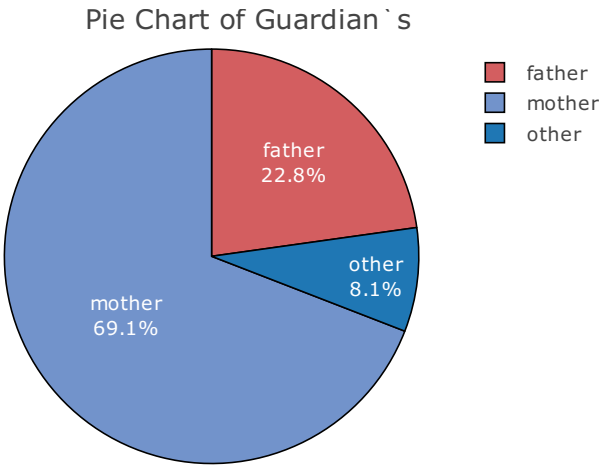



```
student_data_guardian_Stat <- student_data %>%
  group_by(guardian) %>%
  summarise(count = n(),
            percentage = round((n()/ nrow(student_data)), digits = 4))
student_data_guardian_Stat
```

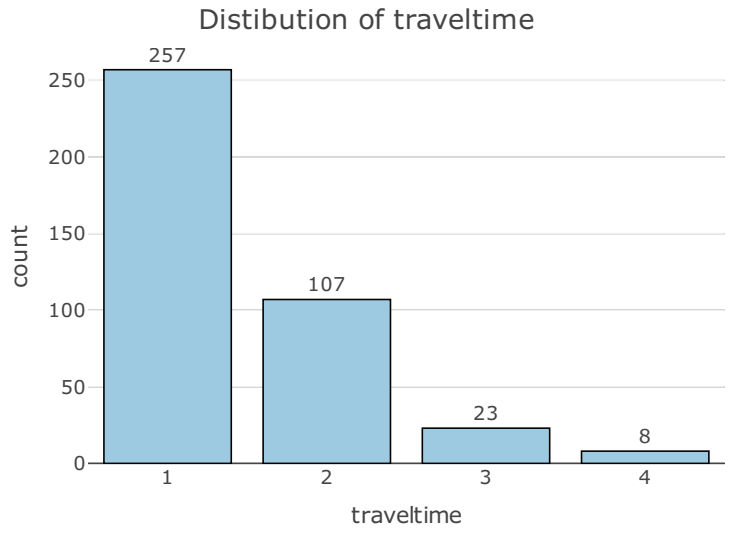
```
## # A tibble: 3 x 3
##   guardian count percentage
##   <chr>    <int>      <dbl>
## 1 father      90      0.228
## 2 mother     273      0.691
## 3 other       32      0.081
```

```
colors <- c('rgb(211,94,96)','rgb(114,147,203)')
Guardian_PieChart <- plot_ly(data = student_data_guardian_Stat, labels = ~guardian, values = ~percentage
```

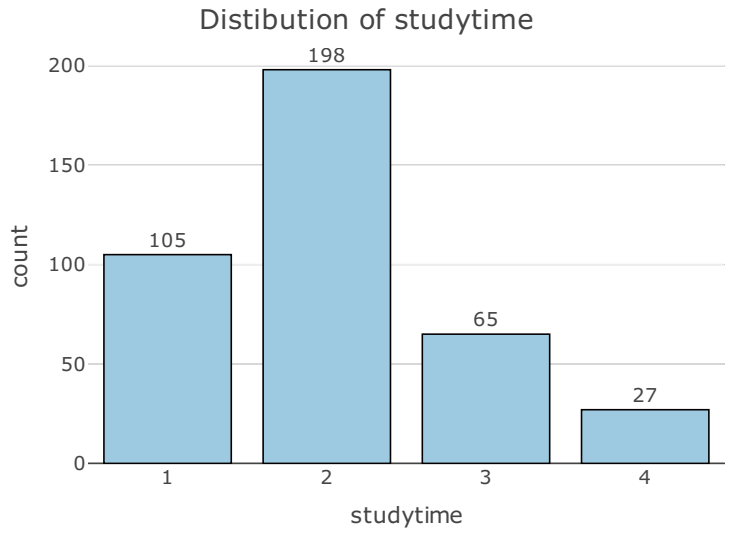
```
                type = 'pie', sort = F,
                textposition = 'inside',
                textinfo = 'label+percent',
                insidetextfont = list(color = 'White'),
                hoverinfo = 'text',
                text = ~count,
                marker = list(colors = colors,
                line = list(color = 'Black', width = 1)),
                showlegend = TRUE)
Guardian_PieChart <- Guardian_PieChart %>% layout(title = 'Pie Chart of Guardian`s')
Guardian_PieChart
```




```
student_data %>%
  group_by(traveltime)%>%
  summarize(count = n()) %>%
  plot_ly(x =~traveltime, y=~count, type = 'bar',
        text = ~count,
        textposition = 'outside',
        marker = list(color = 'rgb(158,202,225)',
          line = list(color = 'black',
                 width = 1.0))) %>%
  layout(title = 'Distibution of traveltime')
```

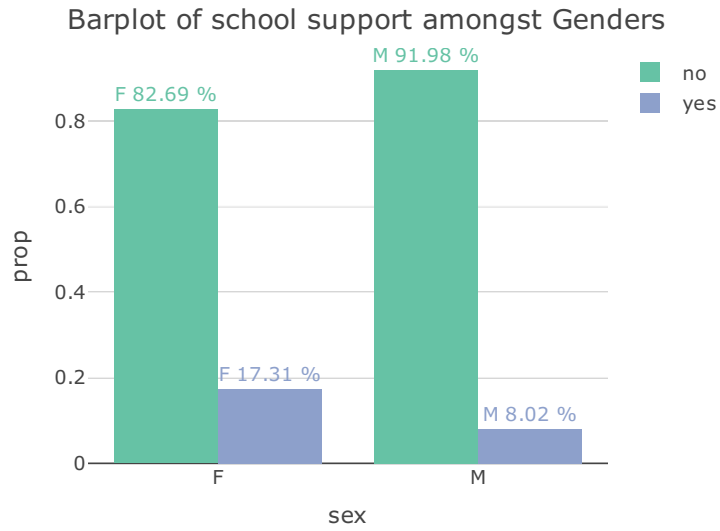



```
student_data %>%
  group_by(studytime)%>%
  summarize(count = n()) %>%
  plot_ly(x =~studytime, y=~count, type = 'bar',
        text = ~count,
        textposition = 'outside',
        marker = list(color = 'rgb(158,202,225)',
          line = list(color = 'black',
                 width = 1.0))) %>%
  layout(title = 'Distibution of studytime')
```



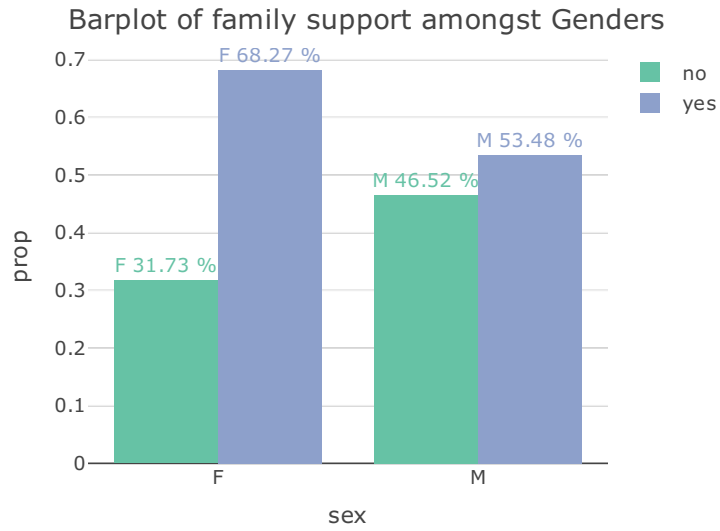

```
student_data %>%
  count(sex, schoolsup, sort = F) %>%
  group_by(sex) %>%
  mutate(prop = round((n / sum(n)),digits = 4)) %>%
  plot_ly(x = ~sex, y=~prop, color = ~schoolsup, type = "bar",
          text = ~paste(sex, prop*100 ,'%'),
          textposition = 'outside') %>%
    layout(barmode = 'Stacked',
           title = 'Barplot of school support amongst Genders')
```

```
## Warning in RColorBrewer::brewer.pal(N, "Set2"): minimal value for n is 3, returning requested palett

## Warning in RColorBrewer::brewer.pal(N, "Set2"): minimal value for n is 3, returning requested palett
```



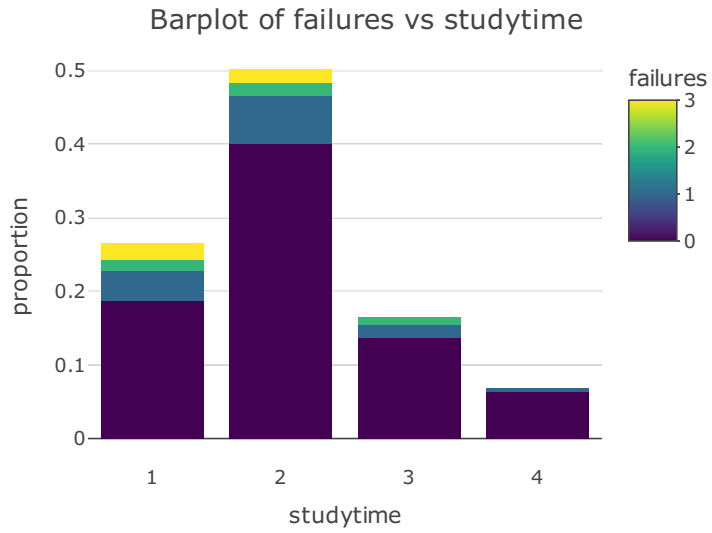

```
student_data %>%
  count(sex, famsup, sort = F) %>%
  group_by(sex) %>%
  mutate(prop = round((n / sum(n)),digits = 4)) %>%
  plot_ly(x = ~sex, y=~prop, color = ~famsup, type = "bar",
          text = ~paste(sex, prop*100 ,'%'),
          textposition = 'outside') %>%
    layout(barmode = 'Stacked',
           title = 'Barplot of family support amongst Genders')
```

```
## Warning in RColorBrewer::brewer.pal(N, "Set2"): minimal value for n is 3, returning requested palett

## Warning in RColorBrewer::brewer.pal(N, "Set2"): minimal value for n is 3, returning requested palett
```

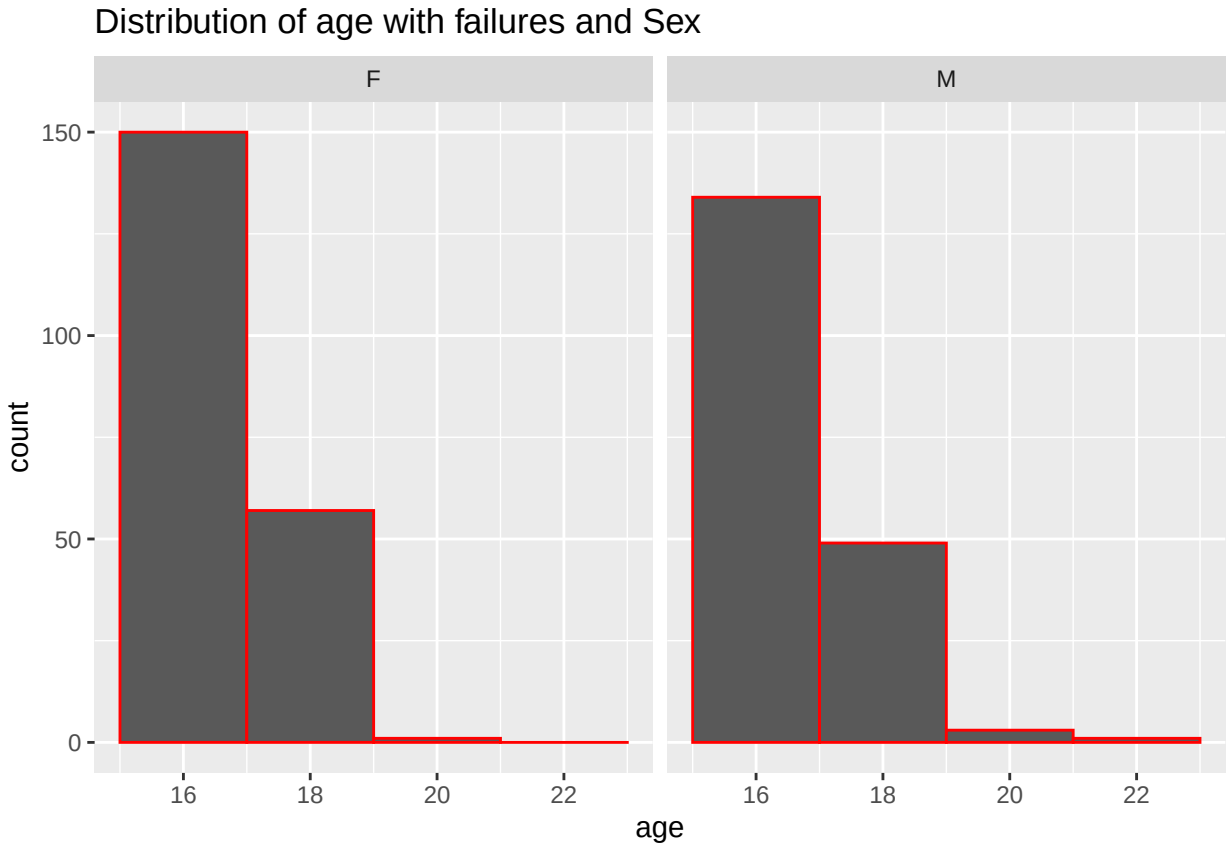



```
student_data %>%
  count(studytime, failures, sort = F) %>%
  mutate(proportion = round((n/sum(n)),digits=4)) %>%
  plot_ly(x =~studytime, y=~proportion, color = ~failures, type = 'bar') %>%
  layout(barmode = 'Group',
         title = 'Barplot of failures vs studytime')
```

```
## Warning: textfont.color doesn't (yet) support data arrays

## Warning: textfont.color doesn't (yet) support data arrays
```



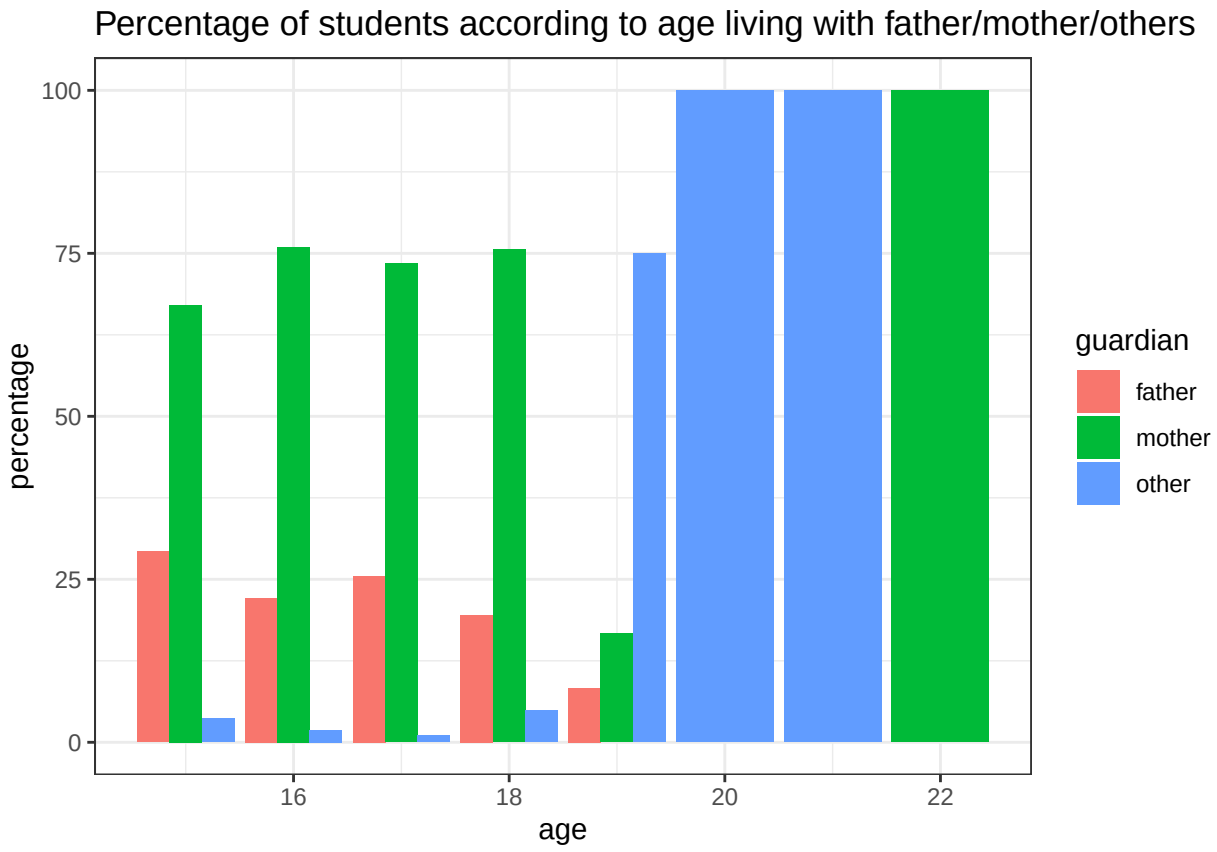

```
student_data %>%
  ggplot(aes(x= age,fill=failures)) +
  geom_histogram(binwidth =2, color="red") +

  xlab("age")+ ggtitle("Distribution of age with failures and Sex")+
  facet_wrap(~sex)
```



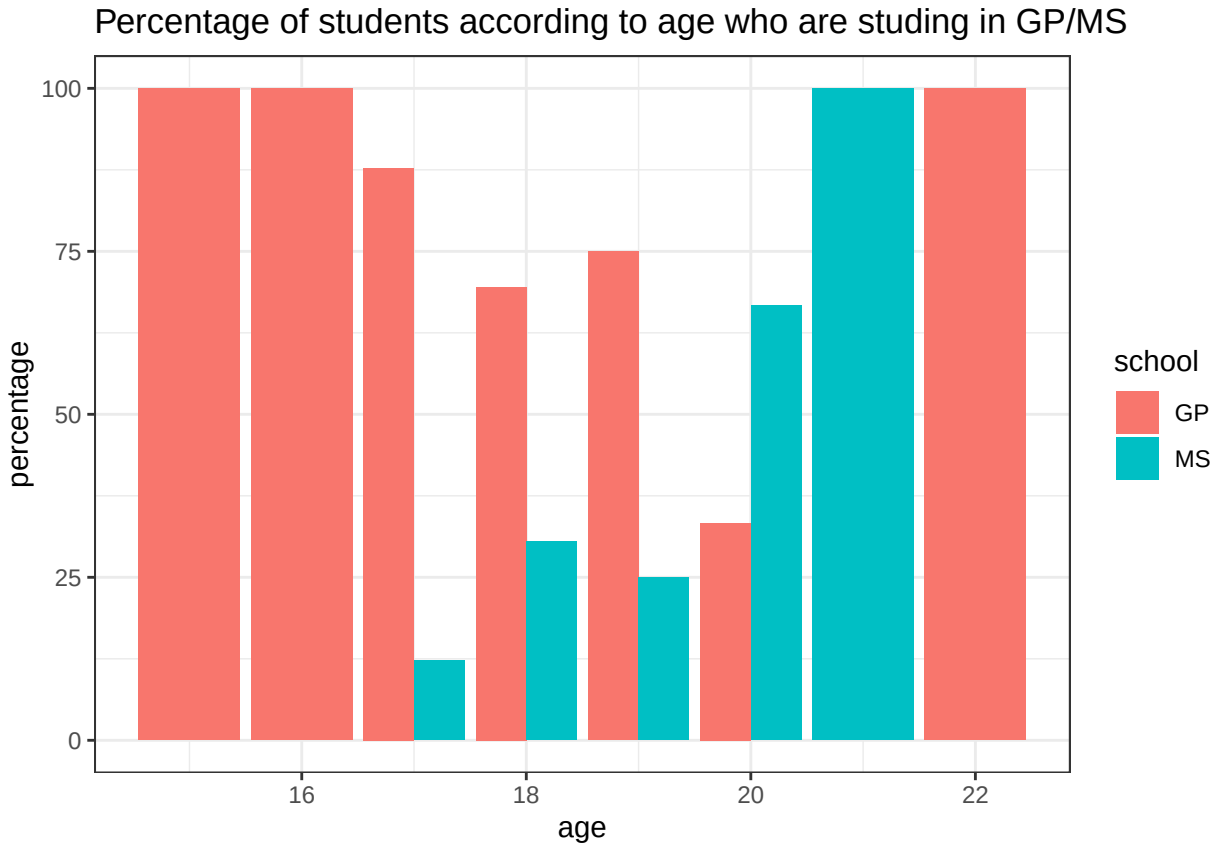

```
library('tidyverse')
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v tibble  3.1.6     v stringr 1.4.0
## v tidyr   1.2.0     v forcats 0.5.1
## v purrr   0.3.4

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x gridExtra::combine() masks dplyr::combine()
## x plotly::filter()     masks dplyr::filter(), stats::filter()
## x dplyr::lag()         masks stats::lag()
```

```
data_2 <- student_data %>%
  group_by(age, guardian) %>%
  tally() %>%
  complete(guardian, fill = list(n =0)) %>%
  mutate(percentage = n / sum(n) *100)

ggplot(data_2, aes(age, percentage, fill = guardian)) +
  geom_bar(stat ='identity', position ='dodge') +
  theme_bw()+ggtitle("Percentage of students according to age living with father/mother/others")
```




```
library('tidyverse')
data_3 <- student_data %>%
  group_by(age, school) %>%
  tally() %>%
  complete(school, fill = list(n =0)) %>%
  mutate(percentage = n / sum(n) *100)

ggplot(data_3, aes(age, percentage, fill = school)) +
  geom_bar(stat ='identity', position ='dodge') +
  theme_bw()+ggtitle("Percentage of students according to age who are studing in GP/MS")
```



problem C:: Data Cleaning

```
student_data <- student_data %>% mutate_all(na_if,"")
summary(student_data)
```

```
##     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   Very Low :276   Very Low :151
##  1st Qu.:3.000   1st Qu.:2.000   Low      : 75   Low      : 85
##  Median :3.000   Median :3.000   Medium   : 26   Medium   : 80
##  Mean   :3.235   Mean   :3.109   High     :  9   High     : 51
##  3rd Qu.:4.000   3rd Qu.:4.000   Very High:  9   Very High: 28
##  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
```

```
colSums(is.na(student_data))
```

```
##     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
```

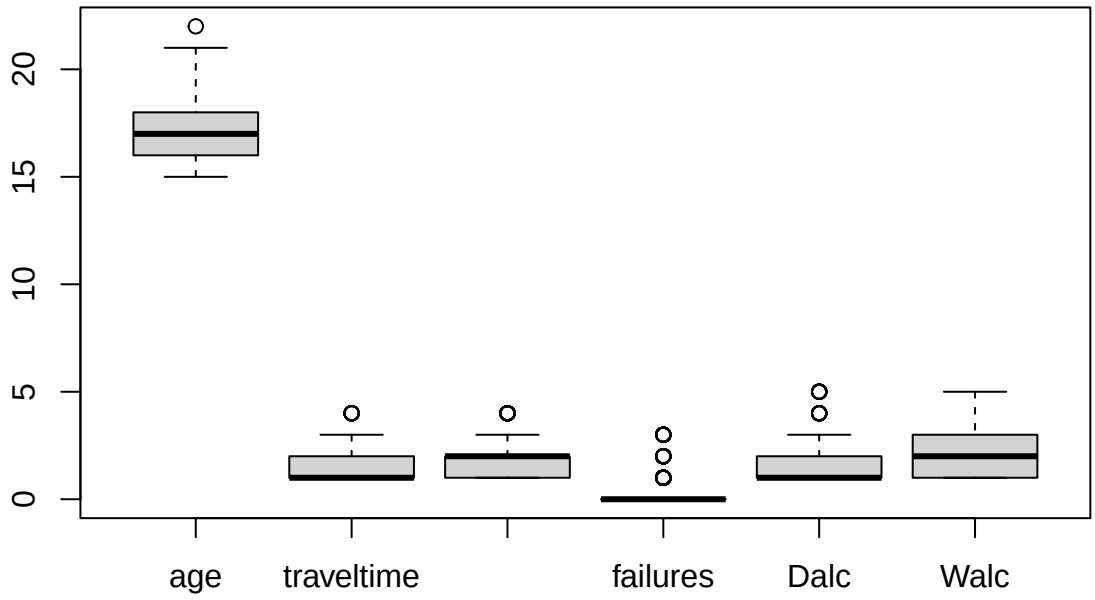
```
##          0          0          0          0          0
```

Selecting Useful variables/ remove non useful variables

```
student_data_new <- student_data[, c("school","sex","age","address","Mjob","Fjob","guardian","traveltim
summary(student_data_new)
```

```
##     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
##      Mjob               Fjob              guardian           traveltime
##  Length:395         Length:395         Length:395         Min.   :1.000
##  Class :character   Class :character   Class :character   1st Qu.:1.000
##  Mode  :character   Mode  :character   Mode  :character   Median :1.000
##                                                           Mean   :1.448
##                                                           3rd Qu.:2.000
##                                                           Max.   :4.000
##    studytime        failures       schoolsup            famsup
##  Min.   :1.000   Min.   :0.0000   Length:395         Length:395
##  1st Qu.:1.000   1st Qu.:0.0000   Class :character   Class :character
##  Median :2.000   Median :0.0000   Mode  :character   Mode  :character
##  Mean   :2.035   Mean   :0.3342
##  3rd Qu.:2.000   3rd Qu.:0.0000
##  Max.   :4.000   Max.   :3.0000
##         Dalc            Walc
##  Very Low :276   Very Low :151
##  Low      : 75   Low      : 85
##  Medium   : 26   Medium   : 80
##  High     :  9   High     : 51
##  Very High:  9   Very High: 28
##
```

```
boxplot((student_data_new[,c("age","traveltime","studytime","failures","Dalc","Walc")]))
```




```
library(dplyr)
student_data_new_1 <-student_data %>%
  mutate(well_educated_family = cut((Fedu+Medu)/2,
  breaks = c(0, 0.99,1.99,2.99,4),
  labels = c("not educated","less educated","moderatly educated","highly educated")))
#str(student_data_new_1)
summary(student_data_new_1)
```

```
##     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
```

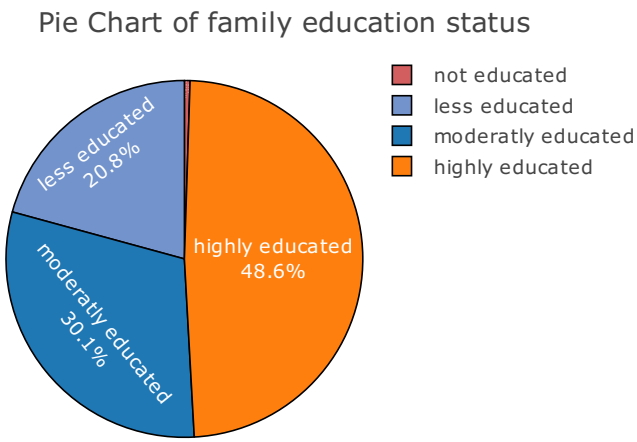
```
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##    traveltime      studytime        failures       schoolsup
##  Min.   :1.000   Min.   :1.000   Min.   :0.0000   Length:395
##  1st Qu.:1.000   1st Qu.:1.000   1st Qu.:0.0000   Class :character
##  Median :1.000   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   Very Low :276   Very Low :151
##  1st Qu.:3.000   1st Qu.:2.000   Low      : 75   Low      : 85
##  Median :3.000   Median :3.000   Medium   : 26   Medium   : 80
##  Mean   :3.235   Mean   :3.109   High     :  9   High     : 51
##  3rd Qu.:4.000   3rd Qu.:4.000   Very High:  9   Very High: 28
##  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                well_educated_family
##  Min.   : 0.00   not educated      :  2
##  1st Qu.: 8.00   less educated     : 82
##  Median :11.00   moderatly educated:119
##  Mean   :10.42   highly educated   :192
##  3rd Qu.:14.00
##  Max.   :20.00
```

```
student_data_educate_Stat <- student_data_new_1 %>%
  group_by( well_educated_family) %>%
  summarise(count = n(),
            percentage = round((n()/ nrow(student_data_new_1)), digits = 4))
student_data_educate_Stat
```

```
## # A tibble: 4 x 3
```

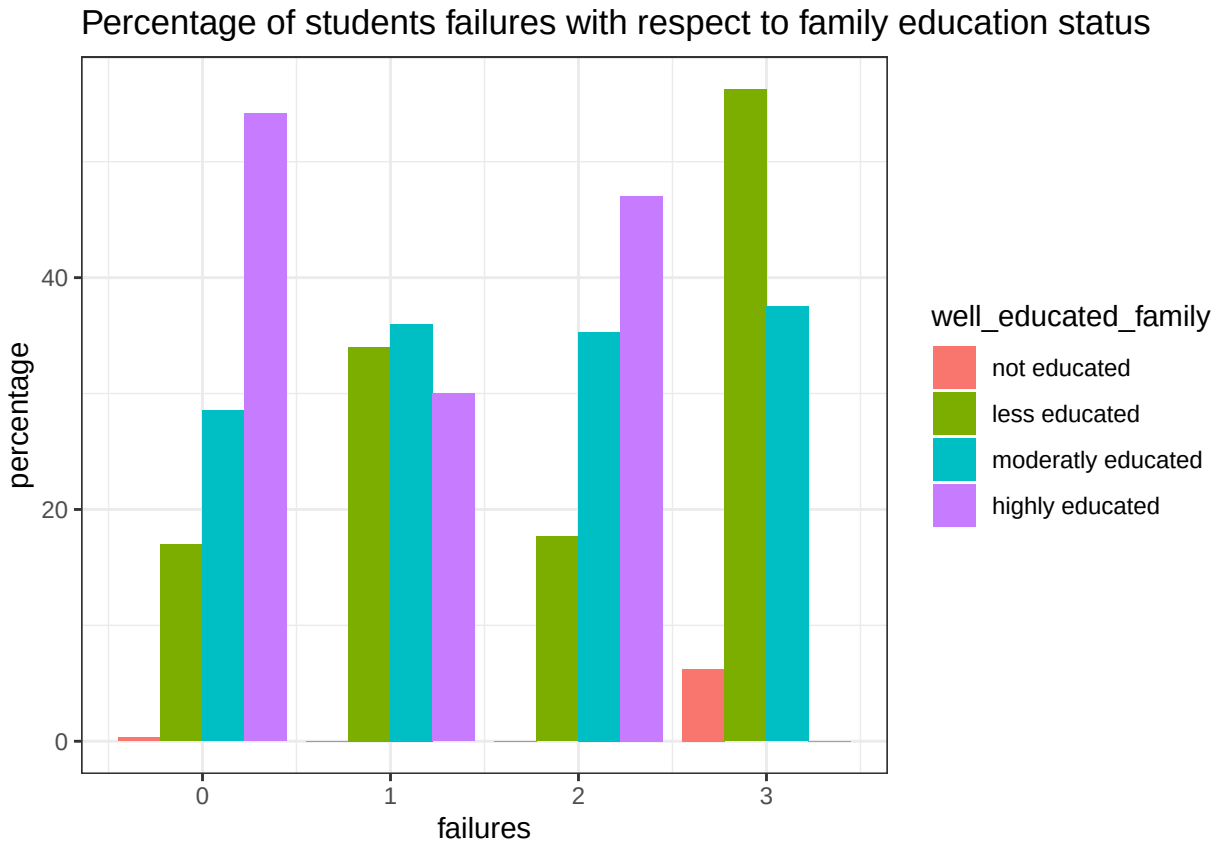
```
##   well_educated_family count percentage
##   <fct>                <int>      <dbl>
## 1 not educated             2     0.0051
## 2 less educated           82     0.208
## 3 moderatly educated     119     0.301
## 4 highly educated        192     0.486
```

```
colors <- c('rgb(211,94,96)','rgb(114,147,203)')
failures_PieChart <- plot_ly(data = student_data_educate_Stat, labels = ~well_educated_family, values =
                type = 'pie', sort = F,
                textposition = 'inside',
                textinfo = 'label+percent',
                insidetextfont = list(color = 'White'),
                hoverinfo = 'text',
                text = ~count,
                marker = list(colors = colors,
                line = list(color = 'Black', width = 1)),
                showlegend = TRUE)
failures_PieChart <- failures_PieChart %>% layout(title = 'Pie Chart of family education status')
failures_PieChart
```




```
library(tidyverse)
library(dplyr)
data_3 <- student_data_new_1 %>%
  group_by(failures, well_educated_family) %>%
```

```
  tally() %>%
  complete(well_educated_family, fill = list(n =0)) %>%
  mutate(percentage = n / sum(n) *100)

ggplot(data_3, aes(failures, percentage, fill = well_educated_family)) +
  geom_bar(stat ='identity', position ='dodge') +
  theme_bw()+ggtitle("Percentage of students failures with respect to family education status")
```



Problem D :: Data Preprocessing

updating the col of well_educated_family in the Student_data_new for dummy data frame student_data_new_1

```
student_data_new["family_education"] <- mutate(student_data_new_1[c(34)])
summary(student_data_new)
```

```
##     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
##      Mjob               Fjob              guardian           traveltime
##  Length:395         Length:395         Length:395         Min.   :1.000
##  Class :character   Class :character   Class :character   1st Qu.:1.000
```

```
##  Mode  :character   Mode  :character   Mode  :character   Median :1.000
##                                                          Mean   :1.448
##                                                          3rd Qu.:2.000
##                                                          Max.   :4.000
##    studytime        failures       schoolsup           famsup
##  Min.   :1.000   Min.   :0.0000   Length:395         Length:395
##  1st Qu.:1.000   1st Qu.:0.0000   Class :character   Class :character
##  Median :2.000   Median :0.0000   Mode  :character   Mode  :character
##  Mean   :2.035   Mean   :0.3342
##  3rd Qu.:2.000   3rd Qu.:0.0000
##  Max.   :4.000   Max.   :3.0000
##        Dalc            Walc              family_education
##  Very Low :276   Very Low :151   not educated      :  2
##  Low      : 75   Low      : 85   less educated     : 82
##  Medium   : 26   Medium   : 80   moderatly educated:119
##  High     :  9   High     : 51   highly educated   :192
##  Very High:  9   Very High: 28
##
```

```
head(student_data_new)
```

```
##   school sex age address     Mjob     Fjob guardian traveltime studytime
## 1     GP   F  18       U  at_home  teacher   mother          2         2
## 2     GP   F  17       U  at_home    other   father          1         2
## 3     GP   F  15       U  at_home    other   mother          1         2
## 4     GP   F  15       U   health services   mother          1         3
## 5     GP   F  16       U    other    other   father          1         2
## 6     GP   M  16       U services    other   mother          1         2
##   failures schoolsup famsup     Dalc     Walc family_education
## 1        0       yes     no Very Low Very Low  highly educated
## 2        0        no    yes Very Low Very Low    less educated
## 3        3       yes     no      Low   Medium    less educated
## 4        0        no    yes Very Low Very Low  highly educated
## 5        0        no    yes Very Low      Low  highly educated
## 6        0        no    yes Very Low      Low  highly educated
```

Creating dummy for well_educated_family(using father education & mother education)

```
library(dplyr)

student_data_new <- student_data %>% mutate(family_education = (Fedu+Medu)/2)

student_data_new <-student_data_new %>%
  mutate(family_education = cut(family_education,
  breaks = c(0, 0.99,1.99,2.99,4),
  labels = c(1,2,3,4)))
```

Create dummy of Gender, 1 => Male 0 => Female

```
student_data_new$Sex<-ifelse(student_data_new$sex=="M",1,0)
```

```r
student_data_new <- student_data_new[, c("Sex","age","traveltime","studytime","failures","family_educati
str(student_data_new)
```

```
## 'data.frame':    395 obs. of  8 variables:
##  $ Sex             : num  0 0 0 0 0 1 1 0 1 1 ...
##  $ age             : int  18 17 15 15 16 16 16 17 15 15 ...
##  $ 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 ...
##  $ family_education: Factor w/ 4 levels "1","2","3","4": 4 2 2 4 4 4 3 4 3 4 ...
##  $ Dalc            : Factor w/ 5 levels "Very Low","Low",..: 1 1 2 1 1 1 1 1 1 1 ...
##  $ Walc            : Factor w/ 5 levels "Very Low","Low",..: 1 1 3 1 2 2 1 1 1 1 ...
```

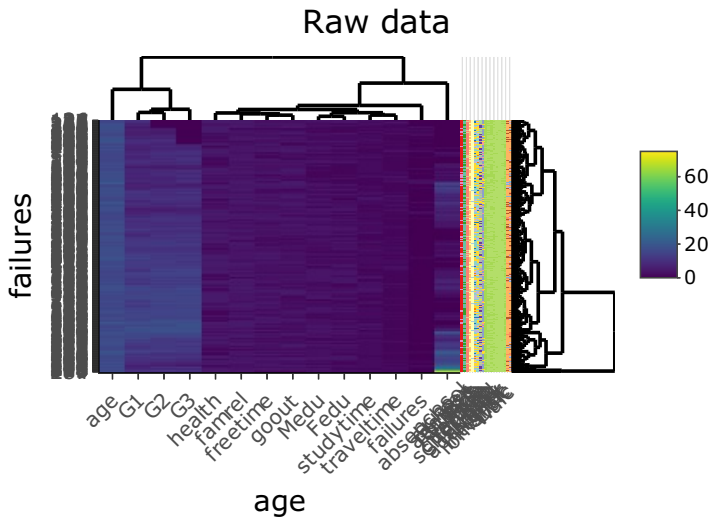
```r
library(heatmaply)
```

```
## Loading required package: viridis
```

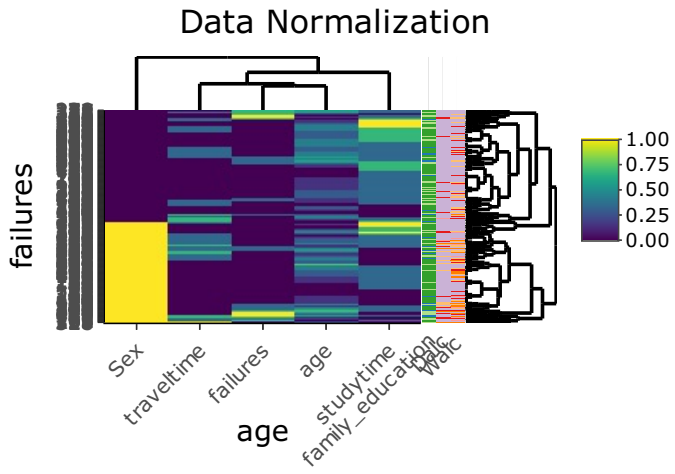
```
## Loading required package: viridisLite
```

```
##
## ======================
## Welcome to heatmaply version 1.3.0
##
## Type citation('heatmaply') for how to cite the package.
## Type ?heatmaply for the main documentation.
##
## The github page is: https://github.com/talgalili/heatmaply/
## Please submit your suggestions and bug-reports at: https://github.com/talgalili/heatmaply/issues
## You may ask questions at stackoverflow, use the r and heatmaply tags:
##   https://stackoverflow.com/questions/tagged/heatmaply
## ======================
```

```r
heatmaply(
  student_data,
  xlab = "age",
  ylab = "failures",
  main = "Raw data"
)
```



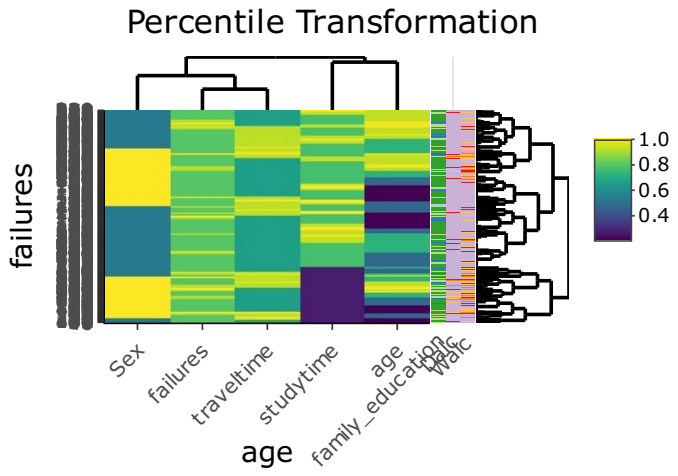

```
heatmaply(
  normalize(student_data_new),
  xlab = "age",
  ylab = "failures",
  main = "Data Normalization"
)
```




```
summary(normalize(student_data_new))
```

```
##       Sex              age           traveltime       studytime
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.1429   1st Qu.:0.0000   1st Qu.:0.0000
##  Median :0.0000   Median :0.2857   Median :0.0000   Median :0.3333
##  Mean   :0.4734   Mean   :0.2423   Mean   :0.1494   Mean   :0.3451
##  3rd Qu.:1.0000   3rd Qu.:0.4286   3rd Qu.:0.3333   3rd Qu.:0.3333
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##     failures      family_education        Dalc            Walc
##  Min.   :0.0000   1:  2            Very Low :276   Very Low :151
##  1st Qu.:0.0000   2: 82            Low      : 75   Low      : 85
##  Median :0.0000   3:119            Medium   : 26   Medium   : 80
##  Mean   :0.1114   4:192            High     :  9   High     : 51
##  3rd Qu.:0.0000                    Very High:  9   Very High: 28
##  Max.   :1.0000
```

```
heatmaply(
  percentize(student_data_new),
  xlab = "age",
  ylab = "failures",
  main = "Percentile Transformation"
)
```



Problem E :: Clustering

PCA projection

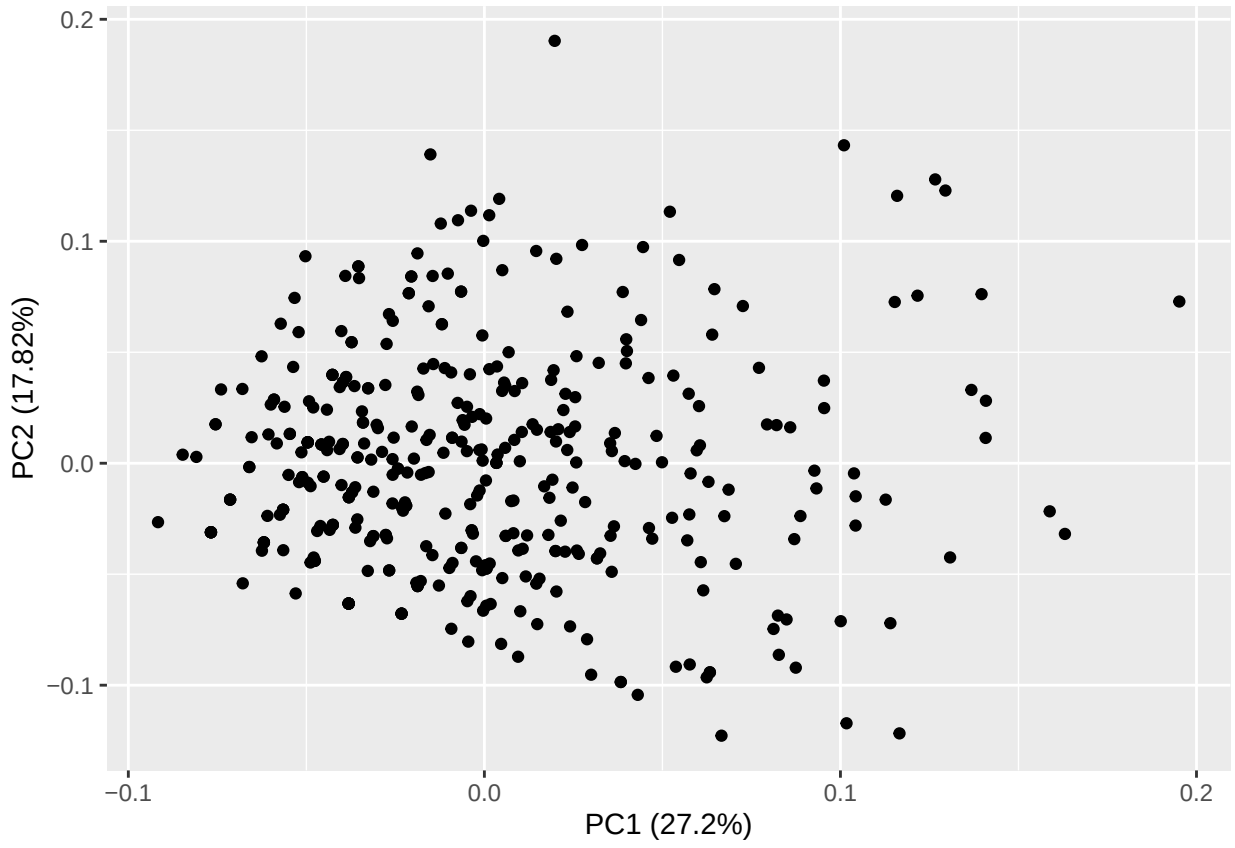
```
student_data_dummy <- data.frame(student_data_new)
#summary(student_data_dummy)

student_data_dummy$Sex <- as.numeric(student_data_dummy$Sex)
student_data_dummy$age <- as.numeric(student_data_dummy$age)
student_data_dummy$traveltime <- as.numeric(student_data_dummy$traveltime)
student_data_dummy$studytime <- as.numeric(student_data_dummy$studytime)
student_data_dummy$failures <- as.numeric(student_data_dummy$failures)
student_data_dummy$family_education <- as.numeric(student_data_dummy$family_education)
student_data_dummy$Dalc <- as.numeric(student_data_dummy$Dalc)
student_data_dummy$Walc <- as.numeric(student_data_dummy$Walc)

str(student_data_dummy)
```

```
## 'data.frame':    395 obs. of  8 variables:
##  $ Sex             : num  0 0 0 0 0 1 1 0 1 1 ...
##  $ age             : num  18 17 15 15 16 16 16 17 15 15 ...
##  $ traveltime      : num  2 1 1 1 1 1 1 2 1 1 ...
##  $ studytime       : num  2 2 2 3 2 2 2 2 2 2 ...
##  $ failures        : num  0 0 3 0 0 0 0 0 0 0 ...
##  $ family_education: num  4 2 2 4 4 4 3 4 3 4 ...
##  $ Dalc            : num  1 1 2 1 1 1 1 1 1 1 ...
##  $ Walc            : num  1 1 3 1 2 2 1 1 1 1 ...
```

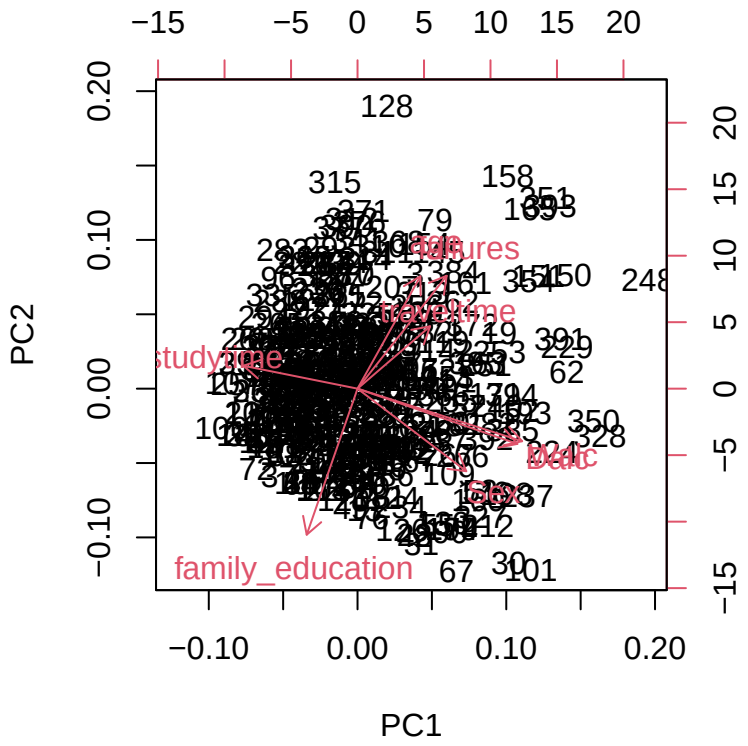
```
#sapply(student_data_dummy, class)
student_data_new.pca <- prcomp(student_data_dummy, center = TRUE,scale. = TRUE)
summary(student_data_new.pca)
```

```
## Importance of components:
##                           PC1    PC2    PC3    PC4     PC5     PC6     PC7
## Standard deviation     1.475 1.1941 1.0160 0.9741 0.85754 0.82707 0.80633
## Proportion of Variance 0.272 0.1782 0.1290 0.1186 0.09192 0.08551 0.08127
## Cumulative Proportion  0.272 0.4502 0.5792 0.6978 0.78976 0.87526 0.95654
##                            PC8
## Standard deviation     0.58968
## Proportion of Variance 0.04346
## Cumulative Proportion  1.00000
```

```
# loading library
library(ggfortify)
student_data_new.pca.plot <- autoplot(student_data_new.pca,
                        data = student_data_dummy,
                        color=Species)

student_data_new.pca.plot
```




```
biplot.student_data_new.pca <- biplot(student_data_new.pca)
```




```
biplot.student_data_new.pca
```

```
## NULL
```

MDS projection

```
student_data_new_mds = smacof::mds(delta = student_data_dummy,ndim = 2, type = "ratio" )
```

```
## Registered S3 methods overwritten by 'proxy':
##   method               from
##   print.registry_field registry
##   print.registry_entry registry

## Warning in df[row(df) > col(df)] <- x: number of items to replace is not a
## multiple of replacement length

## Warning in df[row(df) > col(df)] <- x: number of items to replace is not a
## multiple of replacement length

## Warning in wghts * diss^2: longer object length is not a multiple of shorter
## object length

## Warning in wgths * d * dhat: longer object length is not a multiple of shorter
## object length
```

```
## Warning in dhat - d: longer object length is not a multiple of shorter object
## length
```

```
## Warning in wgths * diss: longer object length is not a multiple of shorter
## object length
```

```
## Warning in dhat - e: longer object length is not a multiple of shorter object
## length
```

```
## Warning in w * Result^2: longer object length is not a multiple of shorter
## object length
```

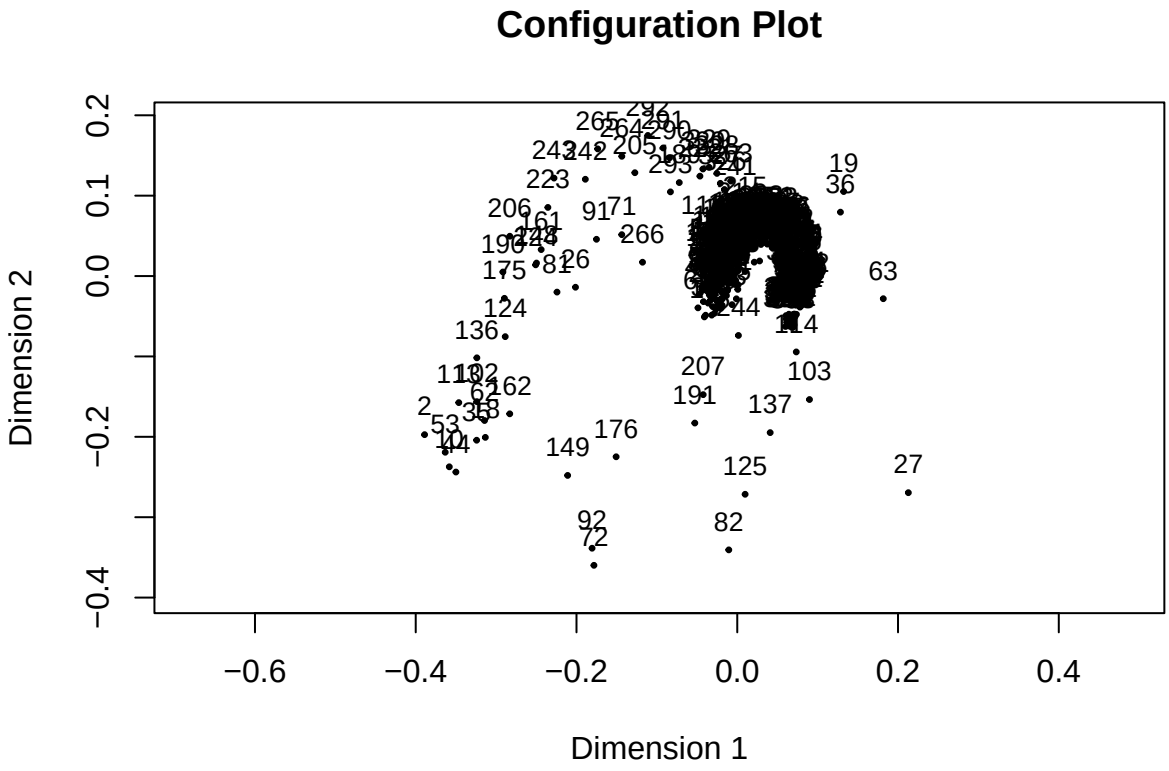
```
## Warning in dhat - e: longer object length is not a multiple of shorter object
## length
```

```
## Warning in dhat - confdiss: longer object length is not a multiple of shorter
## object length
```

```
student_data_new_mds
```

```
##
## Call:
## smacof::mds(delta = student_data_dummy, ndim = 2, type = "ratio")
##
## Model: Symmetric SMACOF
## Number of objects: 395
## Stress-1 value: 0.873
## Number of iterations: 1
```

```
plot(student_data_new_mds)
```



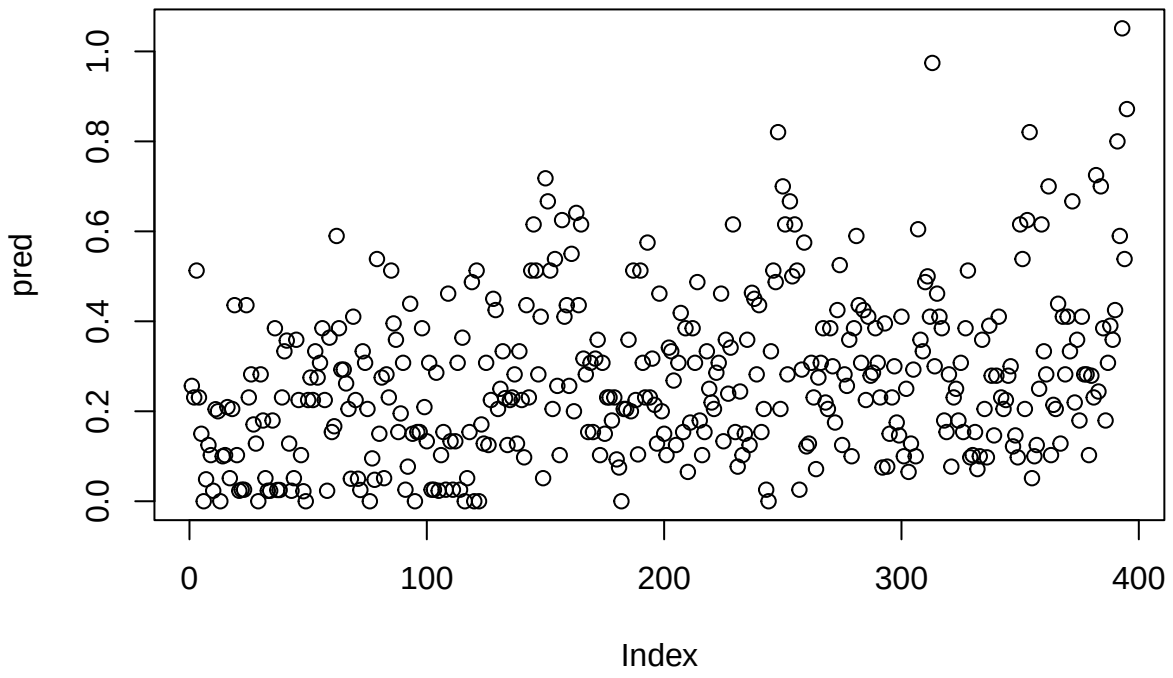

```
set.seed(222)
library(caret)
```

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

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

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
library(gmodels)
fit <- train(failures ~ .,data = student_data_dummy ,method ='knn',tuneLength =20,preProc = c("center",
pred <- predict(fit, newdata = student_data_dummy)
plot(pred)
```




```
summary(pred)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.1333  0.2391  0.2730  0.3846  1.0513
```

```
#pred<-ifelse(pred> 0.5,1,0)
#student_data_dummy_1 <- student_data_dummy %>% slice(1:395)
#student_data_dummy_1<-student_data_dummy_1[,-c(1)]
#colnames(student_data_dummy_1) <- NULL
#rownames(student_data_dummy_1) <- NULL
#conf_mat<-confusionMatrix(as.factor(student_data_dummy_1),as.factor(pred),cutoff = 0.5)
#conf_mat
#conf_mat$table
```

f. Classification

```
library(caTools)
split <- sample.split(student_data_dummy, SplitRatio = 0.75)
train <- subset(student_data_dummy, split==TRUE)
test <- subset(student_data_dummy, split==FALSE)

dim(train)
```

```
## [1] 297   8
```

```r
dim(test)
```

```
## [1] 98  8
```

```r
train_scale <- scale(train[,c("Walc","Dalc","failures")])
test_scale <- scale(test[,c("Walc","Dalc","failures")])

glimpse(train_scale)
```

```
##  num [1:297, 1:3] -1.005 -1.005 0.591 -1.005 -0.207 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : chr [1:297] "1" "2" "3" "4" ...
##   ..$ : chr [1:3] "Walc" "Dalc" "failures"
##  - attr(*, "scaled:center")= Named num [1:3] 2.259 1.438 0.323
##   ..- attr(*, "names")= chr [1:3] "Walc" "Dalc" "failures"
##  - attr(*, "scaled:scale")= Named num [1:3] 1.253 0.795 0.737
##   ..- attr(*, "names")= chr [1:3] "Walc" "Dalc" "failures"
```

RandomForst classifier

```r
set.seed(123)
model_rf <- train(Sex ~., data = student_data_dummy, method ="rf",trControl = trainControl("cv", number
```

```
## Warning in train.default(x, y, weights = w, ...): You are trying to do
## regression and your outcome only has two possible values Are you trying to do
## classification? If so, use a 2 level factor as your outcome column.
```

```
## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?
```

```
## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?

## Warning in randomForest.default(x, y, mtry = min(param$mtry, ncol(x)), ...):
## The response has five or fewer unique values. Are you sure you want to do
## regression?
```

```
model_rf$results
```

```
##   mtry      RMSE  Rsquared       MAE     RMSESD RsquaredSD      MAESD
## 1    2 0.4582577 0.1695874 0.4081582 0.02335097 0.06551285 0.02001172
## 2    4 0.4637100 0.1630997 0.3952204 0.02421528 0.05826748 0.02385263
## 3    7 0.4712573 0.1470169 0.3958253 0.02041368 0.04603134 0.02033878
```

```
model_rf$resample
```

```
##        RMSE   Rsquared       MAE Resample
## 1 0.4608212 0.16004012 0.4086370    Fold1
## 2 0.4647301 0.14489658 0.4193828    Fold2
## 3 0.4427950 0.21718319 0.4063267    Fold5
```

```
## 4 0.4920724 0.07890748 0.4299014    Fold4
## 5 0.4308700 0.24690971 0.3765428    Fold3
```

```
model_rf$bestTune
```

```
##   mtry
## 1    2
```

```
model_rf
```

```
## Random Forest
##
## 395 samples
##   7 predictor
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 316, 316, 316, 316, 316
## Resampling results across tuning parameters:
##
##   mtry  RMSE       Rsquared   MAE
##   2     0.4582577  0.1695874  0.4081582
##   4     0.4637100  0.1630997  0.3952204
##   7     0.4712573  0.1470169  0.3958253
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was mtry = 2.
```

```
library(randomForest)
```

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

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

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

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

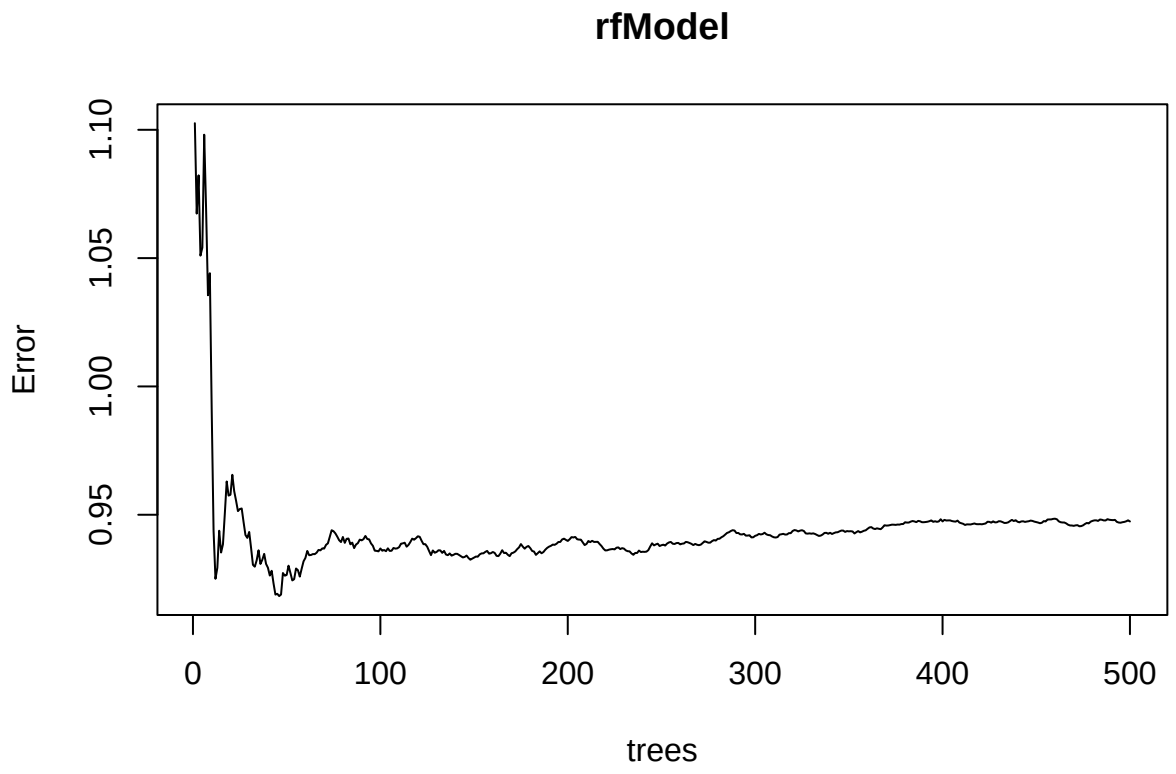
```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

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

```
rfModel=randomForest(Walc~.,data=train,ntree=500,importance=T)
```

```
## Warning in randomForest.default(m, y, ...): The response has five or fewer
## unique values. Are you sure you want to do regression?
```

```
plot(rfModel)
```



Prediction and accuracy

```
predicted.classes <- model_rf %>% predict(student_data_dummy)
#predicted.classes
mean(predicted.classes )
```

```
## [1] 0.7425102
```

```
set.seed(101)
model_ksvm <- train(failures ~., data = student_data_dummy, method ="svmPoly",trControl = trainControl(
model_ksvm
```

```
## Support Vector Machines with Polynomial Kernel
##
## 395 samples
##   7 predictor
```

```
##
## Pre-processing: centered (7), scaled (7)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 355, 356, 355, 356, 356, 356, ...
## Resampling results across tuning parameters:
##
##   degree  scale  C     RMSE       Rsquared    MAE
##   1       0.001  0.25  0.7728276  0.09488669  0.3765950
##   1       0.001  0.50  0.7728438  0.07898067  0.3765909
##   1       0.001  1.00  0.7728340  0.09293894  0.3766052
##   1       0.001  2.00  0.7728407  0.08006682  0.3766060
##   1       0.010  0.25  0.7728270  0.10010504  0.3765981
##   1       0.010  0.50  0.7728352  0.09645751  0.3766012
##   1       0.010  1.00  0.7728226  0.10773985  0.3765914
##   1       0.010  2.00  0.7728222  0.12601386  0.3766080
##   1       0.100  0.25  0.7728305  0.09687563  0.3765891
##   1       0.100  0.50  0.7728335  0.08837733  0.3766233
##   1       0.100  1.00  0.7728370  0.11600740  0.3765936
##   1       0.100  2.00  0.7728546  0.08426962  0.3766209
##   1       1.000  0.25  0.7728404  0.11597753  0.3765864
##   1       1.000  0.50  0.7728377  0.12950503  0.3765944
##   1       1.000  1.00  0.7728238  0.12359287  0.3765807
##   1       1.000  2.00  0.7728292  0.09981936  0.3765913
##   2       0.001  0.25  0.7728234  0.08717102  0.3766460
##   2       0.001  0.50  0.7728059  0.12771646  0.3766128
##   2       0.001  1.00  0.7727634  0.13627669  0.3766285
##   2       0.001  2.00  0.7726915  0.17060997  0.3765789
##   2       0.010  0.25  0.7710305  0.18152807  0.3755726
##   2       0.010  0.50  0.7692392  0.18129408  0.3744666
##   2       0.010  1.00  0.7657025  0.18182962  0.3722270
##   2       0.010  2.00  0.7589118  0.18253987  0.3678333
##   2       0.100  0.25  0.7000752  0.19328155  0.3518723
##   2       0.100  0.50  0.6943078  0.19854061  0.3516414
##   2       0.100  1.00  0.6925718  0.20402479  0.3551591
##   2       0.100  2.00  0.6954707  0.19883616  0.3594427
##   2       1.000  0.25  0.7004241  0.18844834  0.3668180
##   2       1.000  0.50  0.7005769  0.18812101  0.3673068
##   2       1.000  1.00  0.7002695  0.18859134  0.3672023
##   2       1.000  2.00  0.7005816  0.18821017  0.3673661
##   3       0.001  0.25  0.7727836  0.13635805  0.3766342
##   3       0.001  0.50  0.7727484  0.15461901  0.3766268
##   3       0.001  1.00  0.7726122  0.17486532  0.3765380
##   3       0.001  2.00  0.7723935  0.18303035  0.3764359
##   3       0.010  0.25  0.7672971  0.18259396  0.3734275
##   3       0.010  0.50  0.7619979  0.18274838  0.3701512
##   3       0.010  1.00  0.7520832  0.18354911  0.3641442
##   3       0.010  2.00  0.7357928  0.18917222  0.3555576
##   3       0.100  0.25  0.7070624  0.20116600  0.3600522
##   3       0.100  0.50  0.7112400  0.20067046  0.3686341
##   3       0.100  1.00  0.7220873  0.18460826  0.3827148
##   3       0.100  2.00  0.7374684  0.17211103  0.4001033
##   3       1.000  0.25  0.9882413  0.11158347  0.5420091
##   3       1.000  0.50  1.0123210  0.10556031  0.5536925
##   3       1.000  1.00  1.0303351  0.10328910  0.5598437
```

```
##   3       1.000  2.00  1.0363928  0.10313551  0.5615185
##
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were degree = 2, scale = 0.1 and C = 1.
```

```
model_ksvm$bestTune
```

```
##    degree scale C
## 27      2   0.1 1
```

```
model_ksvm$results
```

```
##    degree scale    C      RMSE   Rsquared       MAE    RMSESD RsquaredSD
## 1       1 0.001 0.25 0.7728276 0.09488669 0.3765950 0.1510107 0.07737258
## 2       1 0.001 0.50 0.7728438 0.07898067 0.3765909 0.1510182 0.07448086
## 3       1 0.001 1.00 0.7728340 0.09293894 0.3766052 0.1510408 0.10471526
## 4       1 0.001 2.00 0.7728407 0.08006682 0.3766060 0.1510201 0.08515776
## 5       1 0.010 0.25 0.7728270 0.10010504 0.3765981 0.1510096 0.09801592
## 6       1 0.010 0.50 0.7728352 0.09645751 0.3766012 0.1510400 0.08497039
## 7       1 0.010 1.00 0.7728226 0.10773985 0.3765914 0.1510154 0.10456841
## 8       1 0.010 2.00 0.7728222 0.12601386 0.3766080 0.1510106 0.10897925
## 9       1 0.100 0.25 0.7728305 0.09687563 0.3765891 0.1510214 0.08862853
## 10      1 0.100 0.50 0.7728335 0.08837733 0.3766233 0.1510069 0.06891608
## 11      1 0.100 1.00 0.7728370 0.11600740 0.3765936 0.1510167 0.13987580
## 12      1 0.100 2.00 0.7728546 0.08426962 0.3766209 0.1510169 0.07972509
## 13      1 1.000 0.25 0.7728404 0.11597753 0.3765864 0.1510372 0.09152492
## 14      1 1.000 0.50 0.7728377 0.12950503 0.3765944 0.1510140 0.11551400
## 15      1 1.000 1.00 0.7728238 0.12359287 0.3765807 0.1510173 0.12080326
## 16      1 1.000 2.00 0.7728292 0.09981936 0.3765913 0.1510172 0.09846179
## 17      2 0.001 0.25 0.7728234 0.08717102 0.3766460 0.1510247 0.12455668
## 18      2 0.001 0.50 0.7728059 0.12771646 0.3766128 0.1510194 0.13051325
## 19      2 0.001 1.00 0.7727634 0.13627669 0.3766285 0.1510200 0.14522021
## 20      2 0.001 2.00 0.7726915 0.17060997 0.3765789 0.1510404 0.16327724
## 21      2 0.010 0.25 0.7710305 0.18152807 0.3755726 0.1514213 0.17145506
## 22      2 0.010 0.50 0.7692392 0.18129408 0.3744666 0.1518200 0.17042855
## 23      2 0.010 1.00 0.7657025 0.18182962 0.3722270 0.1526577 0.17047221
## 24      2 0.010 2.00 0.7589118 0.18253987 0.3678333 0.1541788 0.16994581
## 25      2 0.100 0.25 0.7000752 0.19328155 0.3518723 0.1653233 0.16198757
## 26      2 0.100 0.50 0.6943078 0.19854061 0.3516414 0.1661601 0.17331019
## 27      2 0.100 1.00 0.6925718 0.20402479 0.3551591 0.1683053 0.18302403
## 28      2 0.100 2.00 0.6954707 0.19883616 0.3594427 0.1680179 0.18536190
## 29      2 1.000 0.25 0.7004241 0.18844834 0.3668180 0.1720784 0.19037713
## 30      2 1.000 0.50 0.7005769 0.18812101 0.3673068 0.1725000 0.19072037
## 31      2 1.000 1.00 0.7002695 0.18859134 0.3672023 0.1720947 0.19047212
## 32      2 1.000 2.00 0.7005816 0.18821017 0.3673661 0.1722495 0.19029974
## 33      3 0.001 0.25 0.7727836 0.13635805 0.3766342 0.1510304 0.15294532
## 34      3 0.001 0.50 0.7727484 0.15461901 0.3766268 0.1510222 0.18497702
## 35      3 0.001 1.00 0.7726122 0.17486532 0.3765380 0.1510421 0.16424375
## 36      3 0.001 2.00 0.7723935 0.18303035 0.3764359 0.1510996 0.16967295
## 37      3 0.010 0.25 0.7672971 0.18259396 0.3734275 0.1522599 0.16952616
## 38      3 0.010 0.50 0.7619979 0.18274838 0.3701512 0.1534674 0.16935818
## 39      3 0.010 1.00 0.7520832 0.18354911 0.3641442 0.1557990 0.16936083
## 40      3 0.010 2.00 0.7357928 0.18917222 0.3555576 0.1589702 0.16966090
```

```
## 41      3 0.100 0.25 0.7070624 0.20116600 0.3600522 0.1495513 0.17882385
## 42      3 0.100 0.50 0.7112400 0.20067046 0.3686341 0.1456928 0.18358674
## 43      3 0.100 1.00 0.7220873 0.18460826 0.3827148 0.1476389 0.19265471
## 44      3 0.100 2.00 0.7374684 0.17211103 0.4001033 0.1527502 0.19863255
## 45      3 1.000 0.25 0.9882413 0.11158347 0.5420091 0.1834185 0.16996103
## 46      3 1.000 0.50 1.0123210 0.10556031 0.5536925 0.1979974 0.16568977
## 47      3 1.000 1.00 1.0303351 0.10328910 0.5598437 0.2105271 0.16367840
## 48      3 1.000 2.00 1.0363928 0.10313551 0.5615185 0.2175702 0.16380623
##         MAESD
## 1  0.06974325
## 2  0.06970805
## 3  0.06971911
## 4  0.06974812
## 5  0.06973635
## 6  0.06975973
## 7  0.06971904
## 8  0.06973904
## 9  0.06971647
## 10 0.06972268
## 11 0.06972975
## 12 0.06971281
## 13 0.06973245
## 14 0.06969854
## 15 0.06973791
## 16 0.06973943
## 17 0.06972394
## 18 0.06974050
## 19 0.06973758
## 20 0.06972465
## 21 0.06981000
## 22 0.06993336
## 23 0.07016340
## 24 0.07066193
## 25 0.07495461
## 26 0.08011167
## 27 0.08152728
## 28 0.08214315
## 29 0.08720114
## 30 0.08749843
## 31 0.08718936
## 32 0.08739717
## 33 0.06974611
## 34 0.06969228
## 35 0.06971379
## 36 0.06973382
## 37 0.07004847
## 38 0.07036719
## 39 0.07113098
## 40 0.07181545
## 41 0.06878387
## 42 0.06974252
## 43 0.07665158
## 44 0.07880911
## 45 0.10264576
```

```
## 46 0.10597099
## 47 0.10834350
## 48 0.11011521
```

```
model_ksvm$resample
```

```
##         RMSE   Rsquared       MAE Resample
## 1  0.8492008 0.07213739 0.4278968   Fold09
## 2  0.5126191 0.53359788 0.2539450   Fold04
## 3  0.8556262 0.14870157 0.3914130   Fold02
## 4  0.5810740 0.24007649 0.3148586   Fold06
## 5  0.8015943 0.32847065 0.3655605   Fold01
## 6  0.6539944 0.01103448 0.3767948   Fold03
## 7  0.8458655 0.06865421 0.4882967   Fold10
## 8  0.5117379 0.46110957 0.2975512   Fold08
## 9  0.4485541 0.14798931 0.2256851   Fold05
## 10 0.8654517 0.02847630 0.4095890   Fold07
```

```
model_ksvm
```

```
## Support Vector Machines with Polynomial Kernel
##
## 395 samples
##   7 predictor
##
## Pre-processing: centered (7), scaled (7)
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 355, 356, 355, 356, 356, 356, ...
## Resampling results across tuning parameters:
##
##   degree  scale  C     RMSE       Rsquared    MAE
##   1       0.001  0.25  0.7728276  0.09488669  0.3765950
##   1       0.001  0.50  0.7728438  0.07898067  0.3765909
##   1       0.001  1.00  0.7728340  0.09293894  0.3766052
##   1       0.001  2.00  0.7728407  0.08006682  0.3766060
##   1       0.010  0.25  0.7728270  0.10010504  0.3765981
##   1       0.010  0.50  0.7728352  0.09645751  0.3766012
##   1       0.010  1.00  0.7728226  0.10773985  0.3765914
##   1       0.010  2.00  0.7728222  0.12601386  0.3766080
##   1       0.100  0.25  0.7728305  0.09687563  0.3765891
##   1       0.100  0.50  0.7728335  0.08837733  0.3766233
##   1       0.100  1.00  0.7728370  0.11600740  0.3765936
##   1       0.100  2.00  0.7728546  0.08426962  0.3766209
##   1       1.000  0.25  0.7728404  0.11597753  0.3765864
##   1       1.000  0.50  0.7728377  0.12950503  0.3765944
##   1       1.000  1.00  0.7728238  0.12359287  0.3765807
##   1       1.000  2.00  0.7728292  0.09981936  0.3765913
##   2       0.001  0.25  0.7728234  0.08717102  0.3766460
##   2       0.001  0.50  0.7728059  0.12771646  0.3766128
##   2       0.001  1.00  0.7727634  0.13627669  0.3766285
##   2       0.001  2.00  0.7726915  0.17060997  0.3765789
##   2       0.010  0.25  0.7710305  0.18152807  0.3755726
##   2       0.010  0.50  0.7692392  0.18129408  0.3744666
```

```
##   2       0.010  1.00  0.7657025  0.18182962  0.3722270
##   2       0.010  2.00  0.7589118  0.18253987  0.3678333
##   2       0.100  0.25  0.7000752  0.19328155  0.3518723
##   2       0.100  0.50  0.6943078  0.19854061  0.3516414
##   2       0.100  1.00  0.6925718  0.20402479  0.3551591
##   2       0.100  2.00  0.6954707  0.19883616  0.3594427
##   2       1.000  0.25  0.7004241  0.18844834  0.3668180
##   2       1.000  0.50  0.7005769  0.18812101  0.3673068
##   2       1.000  1.00  0.7002695  0.18859134  0.3672023
##   2       1.000  2.00  0.7005816  0.18821017  0.3673661
##   3       0.001  0.25  0.7727836  0.13635805  0.3766342
##   3       0.001  0.50  0.7727484  0.15461901  0.3766268
##   3       0.001  1.00  0.7726122  0.17486532  0.3765380
##   3       0.001  2.00  0.7723935  0.18303035  0.3764359
##   3       0.010  0.25  0.7672971  0.18259396  0.3734275
##   3       0.010  0.50  0.7619979  0.18274838  0.3701512
##   3       0.010  1.00  0.7520832  0.18354911  0.3641442
##   3       0.010  2.00  0.7357928  0.18917222  0.3555576
##   3       0.100  0.25  0.7070624  0.20116600  0.3600522
##   3       0.100  0.50  0.7112400  0.20067046  0.3686341
##   3       0.100  1.00  0.7220873  0.18460826  0.3827148
##   3       0.100  2.00  0.7374684  0.17211103  0.4001033
##   3       1.000  0.25  0.9882413  0.11158347  0.5420091
##   3       1.000  0.50  1.0123210  0.10556031  0.5536925
##   3       1.000  1.00  1.0303351  0.10328910  0.5598437
##   3       1.000  2.00  1.0363928  0.10313551  0.5615185
##
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were degree = 2, scale = 0.1 and C = 1.
```

```
predicted.classes <- model_ksvm %>% predict(student_data_dummy)
mean(predicted.classes)
```

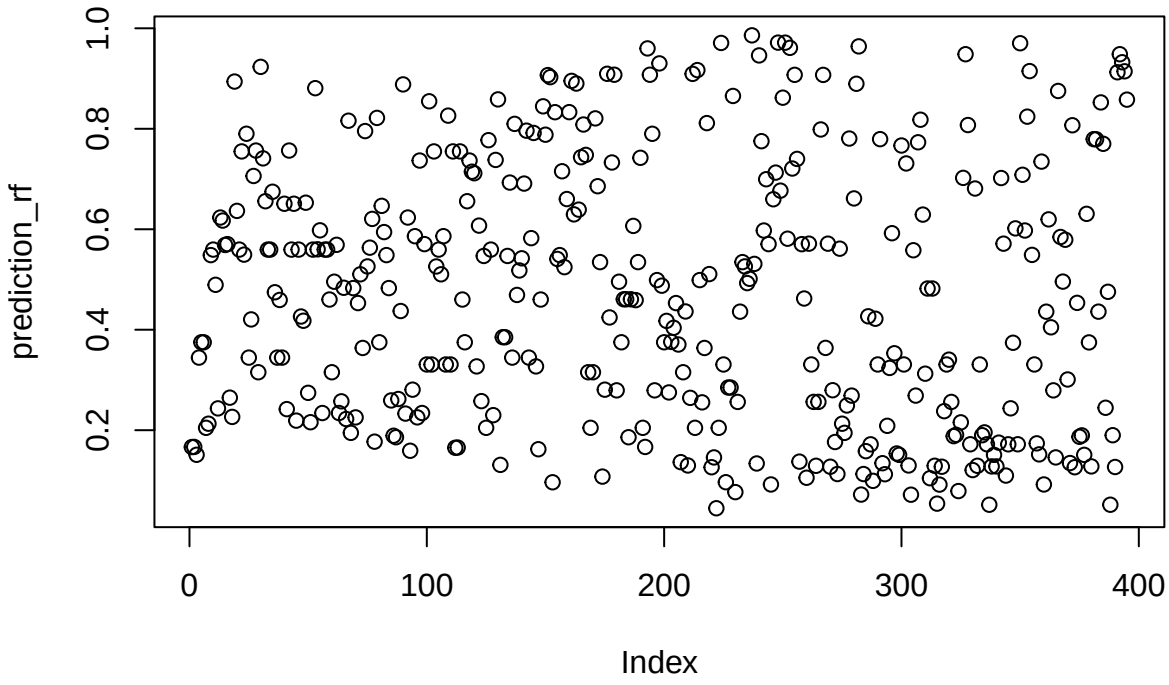
```
## [1] 0.624432
```

As accuracy of Random forest is more than SVM classifier.Therefore we selected Randomforest classifier

```
prediction_rf<-predict(model_rf,newdata = student_data_dummy)
str(prediction_rf)
```

```
##  Named num [1:395] 0.166 0.167 0.151 0.345 0.375 ...
##  - attr(*, "names")= chr [1:395] "1" "2" "3" "4" ...
```

```
plot(prediction_rf)
```




```
summaary(prediction_rf)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.04467 0.23605 0.46213 0.47251 0.68339 0.98597
```

```
auc(student_data_dummy$failures, as.numeric(prediction_rf))
```

```
## Setting levels: control = 15, case = 16
```

```
## Setting direction: controls > cases
```

```
## Area under the curve: 0.7898
```

```
library(rpart)
classifier_data <- rpart(failures ~ Walc, data = student_data_dummy, method="class", minsplit = 10)
summary(classifier_data)
```

```
## Call:
## rpart(formula = failures ~ Walc, data = student_data_dummy, method = "class",
##     minsplit = 10)
##   n= 395
##
##   CP nsplit rel error xerror xstd
## 1  0      0         1      0    0
##
## Node number 1: 395 observations
```

```
##   predicted class=0  expected loss=0.2101266  P(node) =1
##     class counts:   312    50    17    16
##    probabilities: 0.790 0.127 0.043 0.041
```

g. Evaluation

```
pred_1 <- predict(classifier_data, newdata = student_data_dummy)
summary(pred_1)
```

```
##        0                1                2                3
##  Min.   :0.7899   Min.   :0.1266   Min.   :0.04304   Min.   :0.04051
##  1st Qu.:0.7899   1st Qu.:0.1266   1st Qu.:0.04304   1st Qu.:0.04051
##  Median :0.7899   Median :0.1266   Median :0.04304   Median :0.04051
##  Mean   :0.7899   Mean   :0.1266   Mean   :0.04304   Mean   :0.04051
##  3rd Qu.:0.7899   3rd Qu.:0.1266   3rd Qu.:0.04304   3rd Qu.:0.04051
##  Max.   :0.7899   Max.   :0.1266   Max.   :0.04304   Max.   :0.04051
```

```
library(verification)
```

```
## Loading required package: fields
```

```
## Loading required package: spam
```

```
## Spam version 2.8-0 (2022-01-05) is loaded.
## Type 'help( Spam)' or 'demo( spam)' for a short introduction
## and overview of this package.
## Help for individual functions is also obtained by adding the
## suffix '.spam' to the function name, e.g. 'help( chol.spam)'.
```

```
##
## Attaching package: 'spam'
```

```
## The following objects are masked from 'package:base':
##
##     backsolve, forwardsolve
```

```
##
## Try help(fields) to get started.
```

```
##
## Attaching package: 'fields'
```

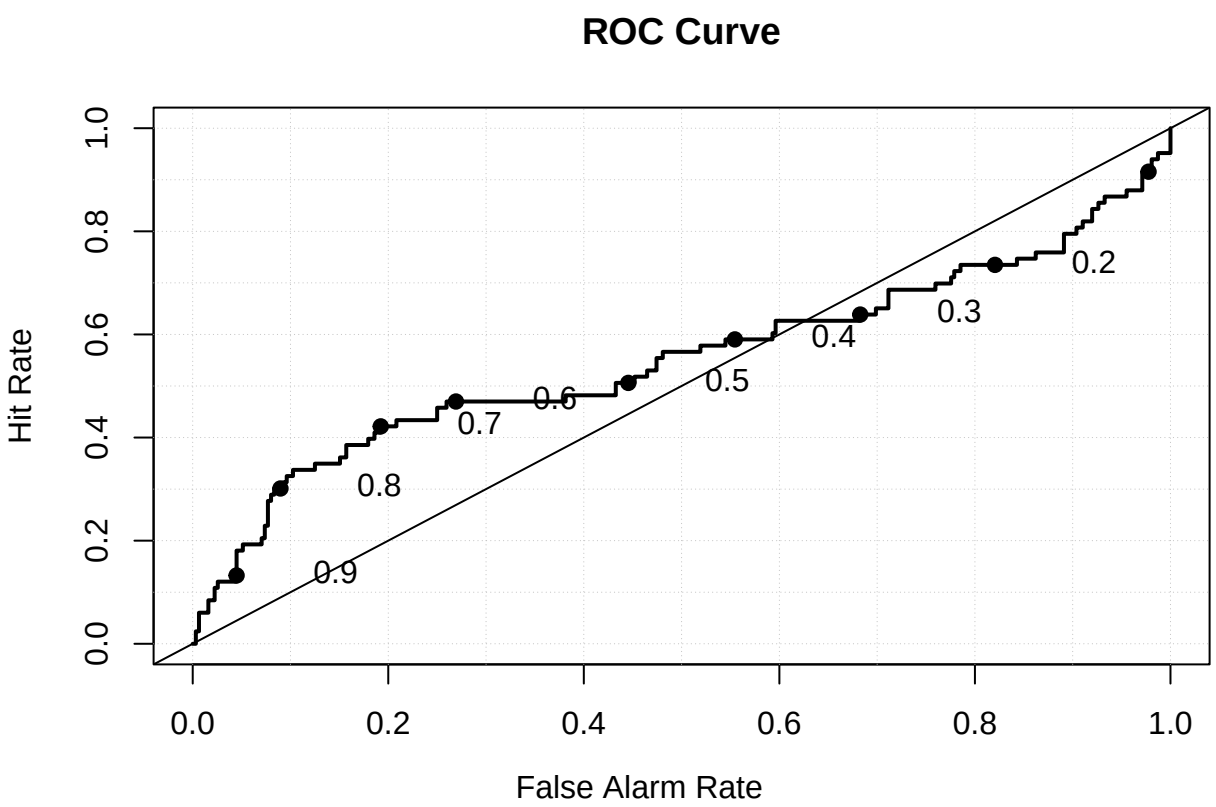
```
## The following object is masked from 'package:ggfortify':
##
##     unscale
```

```
## Loading required package: boot
```

```
##
## Attaching package: 'boot'
```

```
## The following object is masked from 'package:lattice':
##
##     melanoma

## Loading required package: CircStats

## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:plotly':
##
##     select

## The following object is masked from 'package:dplyr':
##
##     select

## Loading required package: dtw

## Loading required package: proxy

##
## Attaching package: 'proxy'

## The following object is masked from 'package:spam':
##
##     as.matrix

## The following objects are masked from 'package:stats':
##
##     as.dist, dist

## The following object is masked from 'package:base':
##
##     as.matrix

## Loaded dtw v1.22-3. See ?dtw for help, citation("dtw") for use in publication.

## Registered S3 method overwritten by 'verification':
##   method    from
##   lines.roc pROC
```

```
data<-data.frame(student_data_dummy)
#names(data)<-c("failures","Walc")
roc.plot(data$failures,prediction_rf)
```



The above plot is ROC and Area under the curve is 0.7898. After doing classification the accuracy has increased from 74% to 789%

h. Report

1. Student Performance Data & Student Alcohol Consumption, data available on kaggle.com is used for prediction. I merged both files to find the percentage of failures with respect to alcohol consumption. New data named as Student_data.

2. In the data exploration dimension, 395 observations with 33 columns are found. Some are categorical and some are integer variables. 48.48% are Integer variables and 51.52% are categorical variables. Some visualizations are done on the data. We found that student consume more alcohol in weekends. The very high alcohol consumption category has an interesting shape as it expands while others tend to decrease. We can also notice it is nicely shaped as a bottle. Interestingly student age with 16 years is highest in all areas like less alcohol to high consumption. But most of the 16 years students live with their mother. And Femal students are getting more support from their family then males. And same with school also. But surprisingly schools are not supporting students more.

3. Removing all the N/A in the data set (student_data), and removing all the non-useful variables. Using Mother education & Father Education, I developed Well Educated family. Because Father & mother are belonging to one family. We can observe that students with high failure rate are belongs to less educated families & less educated families and surprisingly students belongs to high educated families has not failed in exams.

4.Updating the col of well_educated_family in the Student_data_new for dummy data frame student_data_new_1. Creating dummy for well_educated_family(using father education & mother education) with family_education. And Create dummy of Gender, 1 => Male 0 => Female. Applying

"heatmaply" doing visualization for raw data(student_data) and applying visualization on normalized data.

5.PCA is one of the most used unsupervised learning algorithms, Doing PCA on the data set, and using "biplot" a two-dimensional chart that represents the relationship between the rows and columns of a data set. And doing MDS projection.

6.Doing Random forest and SVM to the data set, I got accuracy around 74% for randomforest & around 62% for SVM. Using "rpart" for building classification and regression trees.

7. At last ROC of best Randomforest model is plotted and Area under the curve is calculated ,which is 78.98%. ROC is better performance metrics as compared to Accuracy if data is imbalanced.