Statistical Learning Final Report

Alberto Calabrese, Eleonora Mesaglio, Greta d'Amore Grelli

2024-05-29

Contents

1 Introduction

Here Eleonora you can write the introduction of the project describing the scope and the data used.

2 Libraries

```
library(corrplot)
library(knitr)
```

3 Data

The data set we are going to use is the "Starbucks ...." dataset from Kaggle.

It contains 242 rows and 18 columns.

```
data <- read.csv("Data/starbucks.csv", header = TRUE, sep = ",")

# Overview of the data
summary(data)
```

```
##  Beverage_category    Beverage         Beverage_prep         Calories
##  Length:242         Length:242         Length:242         Min.   :  0.0
##  Class :character   Class :character   Class :character   1st Qu.:120.0
```

```
##  Mode  :character   Mode  :character   Mode  :character   Median :185.0
##                                                          Mean   :193.9
##                                                          3rd Qu.:260.0
##                                                          Max.   :510.0
##  Total.Fat..g.    Trans.Fat..g.   Saturated.Fat..g.  Sodium..mg.
##  Min.   : 0.000   Min.   :0.000   Min.   :0.0000    Min.   : 0.000
##  1st Qu.: 0.200   1st Qu.:0.100   1st Qu.:0.0000    1st Qu.: 0.000
##  Median : 2.500   Median :0.500   Median :0.0000    Median : 5.000
##  Mean   : 2.905   Mean   :1.307   Mean   :0.0376    Mean   : 6.364
##  3rd Qu.: 4.500   3rd Qu.:2.000   3rd Qu.:0.1000    3rd Qu.:10.000
##  Max.   :15.000   Max.   :9.000   Max.   :0.3000    Max.   :40.000
##  Total.Carbohydrates..g. Cholesterol..mg. Dietary.Fibre..g.   Sugars..g.
##  Min.   :  0.0           Min.   : 0.00    Min.   :0.0000    Min.   : 0.00
##  1st Qu.: 70.0           1st Qu.:21.00    1st Qu.:0.0000    1st Qu.:18.00
##  Median :125.0           Median :34.00    Median :0.0000    Median :32.00
##  Mean   :128.9           Mean   :35.99    Mean   :0.8058    Mean   :32.96
##  3rd Qu.:170.0           3rd Qu.:50.75    3rd Qu.:1.0000    3rd Qu.:43.75
##  Max.   :340.0           Max.   :90.00    Max.   :8.0000    Max.   :84.00
##   Protein..g.     Vitamin.A....DV.   Vitamin.C....DV.   Calcium....DV.
##  Min.   : 0.000   Length:242         Length:242         Length:242
##  1st Qu.: 3.000   Class :character   Class :character   Class :character
##  Median : 6.000   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 6.979
##  3rd Qu.:10.000
##  Max.   :20.000
##  Iron....DV.        Caffeine..mg.
##  Length:242         Length:242
##  Class :character   Class :character
##  Mode  :character   Mode  :character
##
##
##
```

```
str(data)
```

```
## 'data.frame':    242 obs. of  18 variables:
##  $ Beverage_category      : chr  "Coffee" "Coffee" "Coffee" "Coffee" ...
##  $ Beverage               : chr  "Brewed Coffee" "Brewed Coffee" "Brewed Coffee" "Brewed Coffee" ...
##  $ Beverage_prep          : chr  "Short" "Tall" "Grande" "Venti" ...
##  $ Calories               : int  3 4 5 5 70 100 70 100 150 110 ...
##  $ Total.Fat..g.          : num  0.1 0.1 0.1 0.1 0.1 3.5 2.5 0.2 6 4.5 ...
##  $ Trans.Fat..g.          : num  0 0 0 0 0.1 2 0.4 0.2 3 0.5 ...
##  $ Saturated.Fat..g.      : num  0 0 0 0 0 0.1 0 0 0.2 0 ...
##  $ Sodium..mg.            : int  0 0 0 0 5 15 0 5 25 0 ...
##  $ Total.Carbohydrates..g.: int  5 10 10 10 75 85 65 120 135 105 ...
##  $ Cholesterol..mg.       : int  0 0 0 0 10 10 6 15 15 10 ...
##  $ Dietary.Fibre..g.      : int  0 0 0 0 0 0 1 0 0 1 ...
##  $ Sugars..g.             : int  0 0 0 0 9 9 4 14 14 6 ...
##  $ Protein..g.            : num  0.3 0.5 1 1 6 6 5 10 10 8 ...
##  $ Vitamin.A....DV.       : chr  "0%" "0%" "0%" "0%" ...
##  $ Vitamin.C....DV.       : chr  "0%" "0%" "0%" "0%" ...
##  $ Calcium....DV.         : chr  "0%" "0%" "0%" "2%" ...
##  $ Iron....DV.            : chr  "0%" "0%" "0%" "0%" ...
##  $ Caffeine..mg.          : chr  "175" "260" "330" "410" ...
```

3.1 Data Transformation

We have to remove the percentage sign from the data and set the other variables as numeric.

```
# Remove percentage sign from the data
data$Vitamin.C....DV. <- as.numeric(gsub("%", "", data$Vitamin.C....DV.))
data$Calcium....DV. <- as.numeric(gsub("%", "", data$Calcium....DV.))
data$Iron....DV. <- as.numeric(gsub("%", "", data$Iron....DV.))
data$Vitamin.A....DV. <- as.numeric(gsub("%", "", data$Vitamin.A....DV.))

# Set the other variables as numeric
data$Calories <- as.numeric(data$Calories)
data$Trans.Fat..g. <- as.numeric(data$Trans.Fat..g.)
data$Total.Fat..g. <- as.numeric(data$Total.Fat..g.)
data$Cholesterol..mg. <- as.numeric(data$Cholesterol..mg.)
data$Sodium..mg. <- as.numeric(data$Sodium..mg.)
data$Total.Carbohydrates..g. <- as.numeric(data$Total.Carbohydrates..g.)
data$Dietary.Fibre..g. <- as.numeric(data$Dietary.Fibre..g.)
data$Sugars..g. <- as.numeric(data$Sugars..g.)
data$Caffeine..mg. <- as.numeric(data$Caffeine..mg.)
```

3.2 Data Cleaning

We have some NA values in the data. The NA values are in the Caffeine..mg. column. We will remove the rows with NA values.

```
data_cleaned <- data[!is.na(data$Caffeine..mg.),]
```

3.3 Rename Columns

We will rename the cleaned data columns by removing the dots and the unity of measure in order to obtain a more readable dataset.

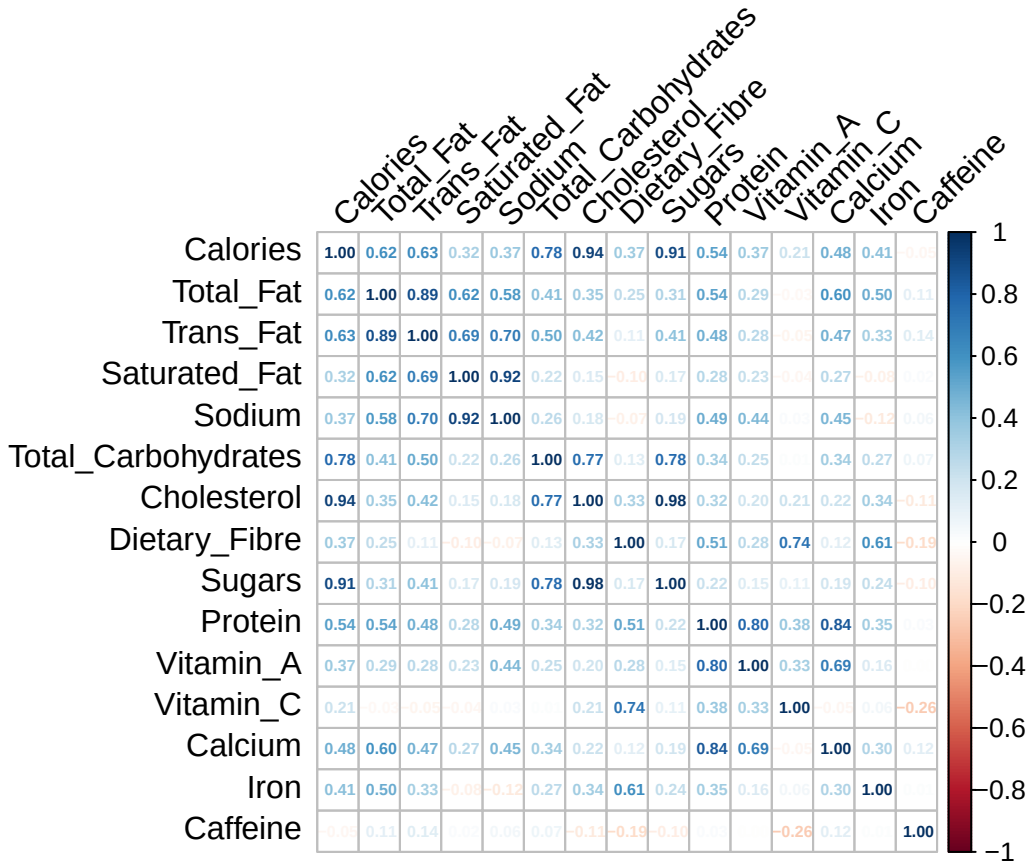
```
colnames(data_cleaned) <- c("Beverage_category", "Beverage",
                            "Beverage_prep", "Calories",
                            "Total_Fat", "Trans_Fat",
                            "Saturated_Fat", "Sodium",
                            "Total_Carbohydrates", "Cholesterol",
                            "Dietary_Fibre", "Sugars",
                            "Protein", "Vitamin_A",
                            "Vitamin_C", "Calcium",
                            "Iron", "Caffeine")
```

4 Correlation Analysis

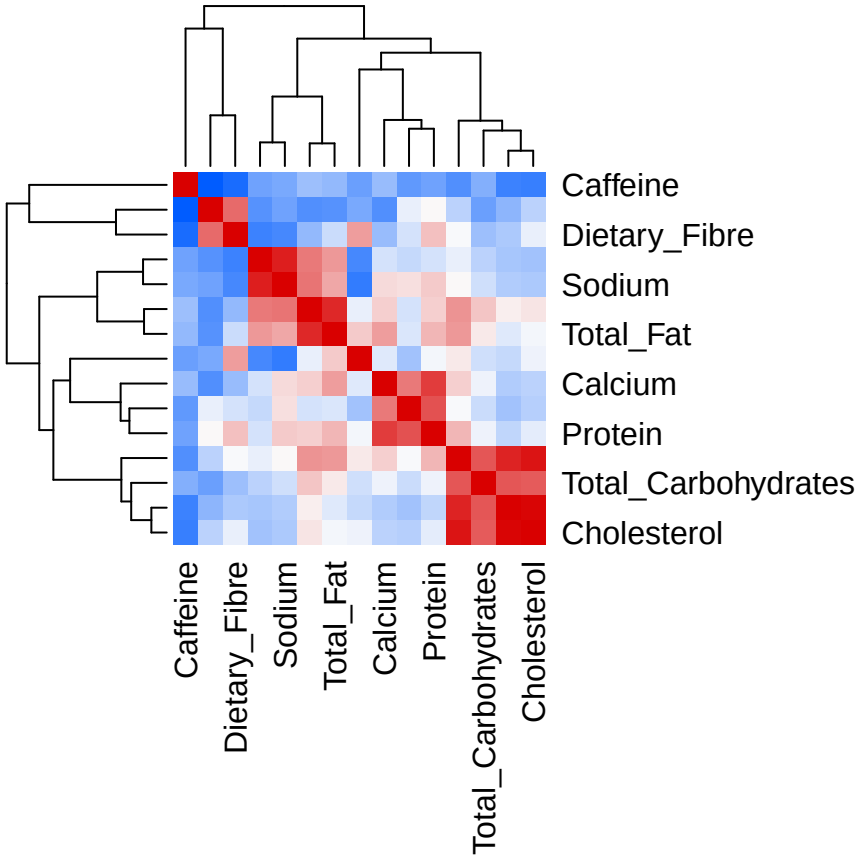
We will calculate the correlation matrix and plot it.

```
# Remove first 3 columns for the correlation matrix since them are categorical
data_num <- data_cleaned[, -c(1:3)]

# Calculate the correlation matrix

correlation_matrix <- cor(data_num)

# Plot the correlation matrix using corrplot
```

```
corrplot(correlation_matrix, method = "number", tl.col = "black",
         tl.srt = 45, addCoef.col = "black", number.cex = 0.5)
```




```
# Heatmap of the correlation matrix
heatmap(cor(data_num),
        col = colorRampPalette(c("#005cff", "#fbfbfb", "#d90000"))(100),
        symm = TRUE,
        margins = c(10, 10),
        cexRow = 1.2,
        cexCol = 1.2)
```



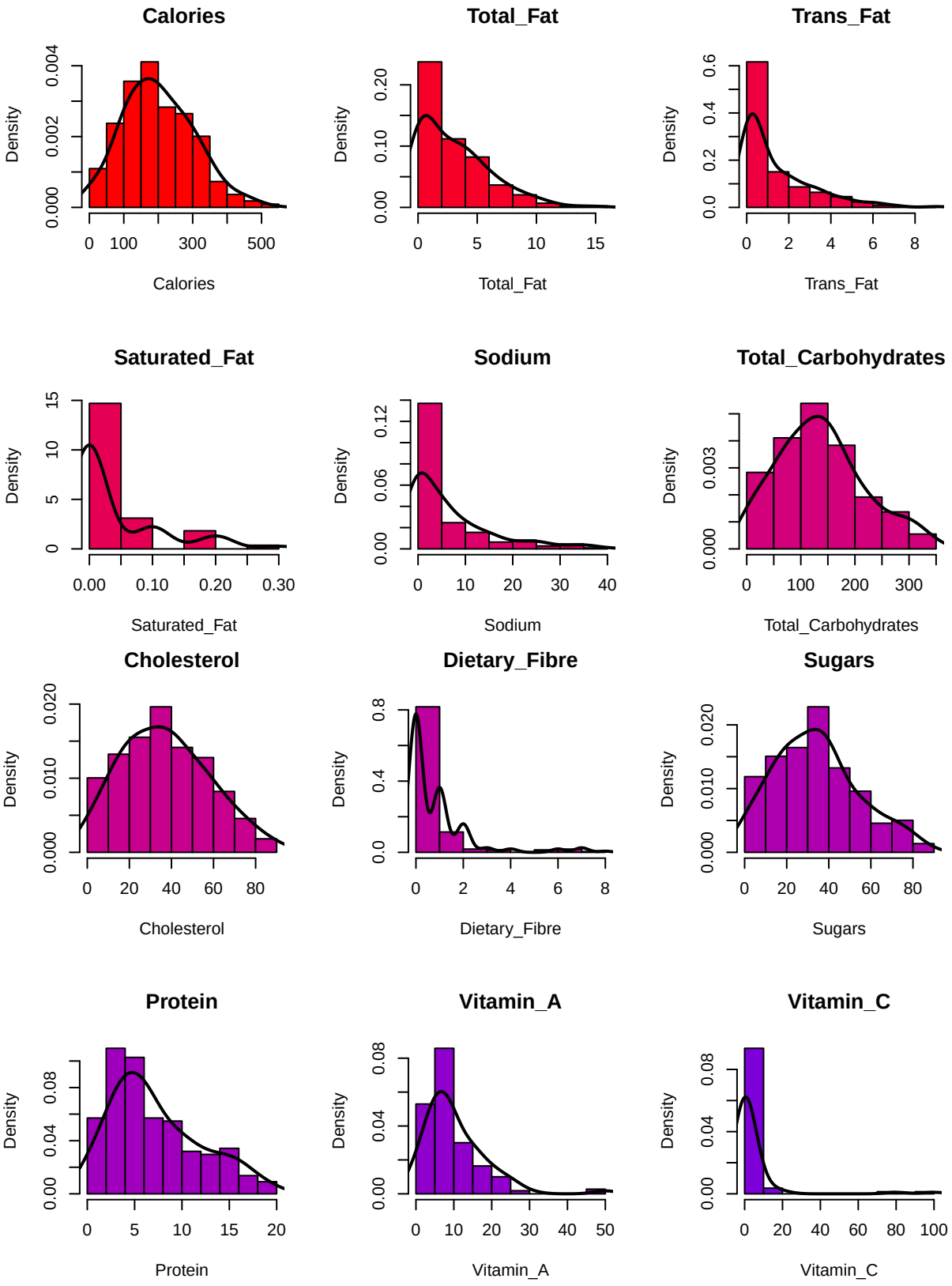
Add some comments on the graph that we obtained.

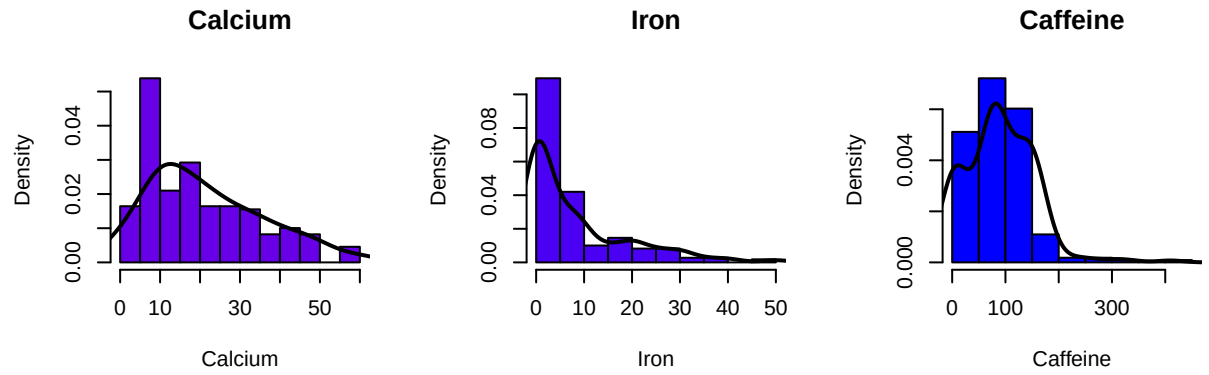
5 Data Visualization

5.1 Histograms

We will plot some histograms to visualize the data.

```
# Histogram of the data with density distribution
par(mfrow = c(2, 3))
col <- c('#ff0000', '#f70028', '#ee0040', '#e50055', '#dc0069',
         '#d2007b', '#c7008d', '#bb009e', '#ae00ae', '#a000be',
         '#8f00cc', '#7d00da', '#6700e7', '#4900f3', '#0000ff')
for (i in 1:ncol(data_num)) {
  hist(data_num[, i], main = colnames(data_num)[i],
       xlab = colnames(data_num)[i], col = col[i], freq = FALSE)
  dens <- density(data_num[, i], na.rm=TRUE, adjust=1.25)
  lines(dens, col = "black", lwd = 2)
}
```


Calories
Density
0.000
0.002
0.004
0
100
300
500
Calories
Total_Fat
Density
0.00
0.10
0.20
0
5
10
15
Total_Fat
Trans_Fat
Density
0.0
0.2
0.4
0.6
0
2
4
6
8
Trans_Fat
Saturated_Fat
Density
0
5
10
15
0.00
0.10
0.20
0.30
Saturated_Fat
Sodium
Density
0.00
0.06
0.12
0
10
20
30
40
Sodium
Total_Carbohydrates
Density
0.000
0.003
0
100
200
300
Total_Carbohydrates
Cholesterol
Density
0.000
0.010
0.020
0
20
40
60
80
Cholesterol
Dietary_Fibre
Density
0.0
0.4
0.8
0
2
4
6
8
Dietary_Fibre
Sugars
Density
0.000
0.010
0.020
0
20
40
60
80
Sugars
Protein
Density
0.00
0.04
0.08
0
5
10
15
20
Protein
Vitamin_A
Density
0.00
0.04
0.08
0
10
20
30
40
50
Vitamin_A
Vitamin_C
Density
0.00
0.04
0.08
0
20
40
60
80
100
Vitamin_C



5.2 Pairplot

We will plot a pairplot to visualize the relationship between the variables.

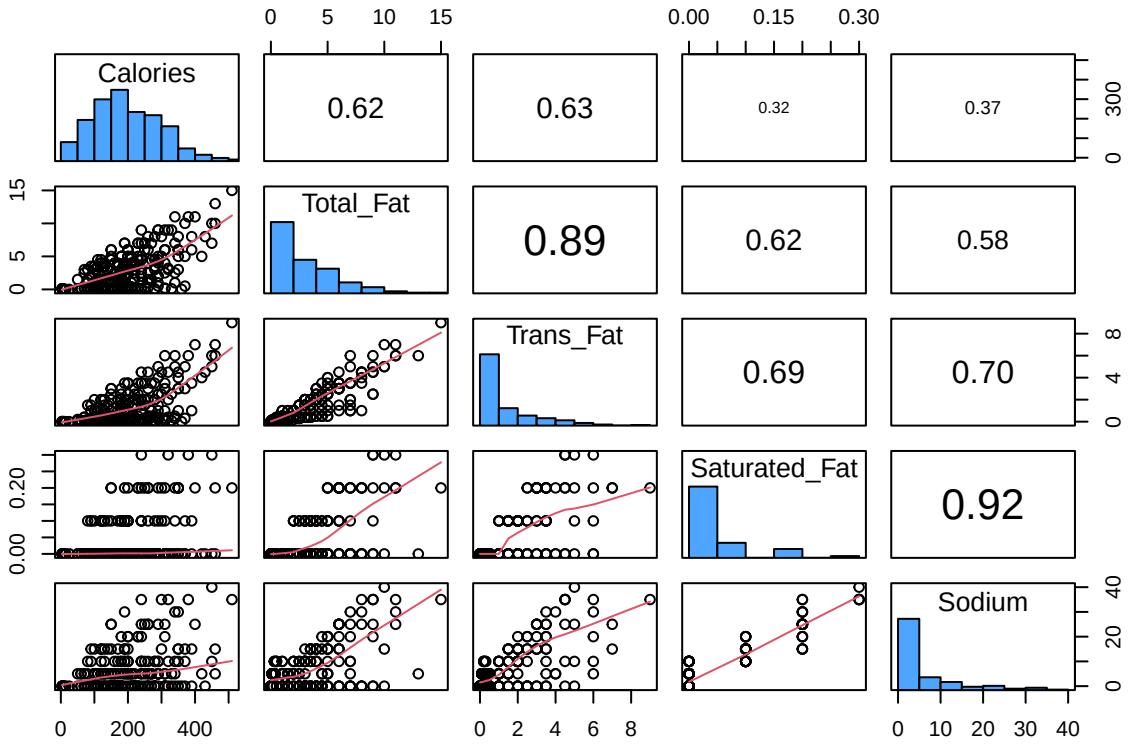
First of all we have to define the function for the pairplot.

```
# Histogram function
panel.hist <- function(x, colour, ...)
{
  usr <- par("usr")
  on.exit(par(usr))
  par(usr = c(usr[1:2], 0, 1.5))
  h <- hist(x, plot = FALSE)
  breaks <- h$breaks
  nB <- length(breaks)
  y <- h$counts
  y <- y / max(y)
  rect(breaks[-nB], 0, breaks[-1], y, col = colour, ...)
}

# Correlations function
panel.cor <- function(x, y, digits = 2, prefix = "", cex.cor, ...)
{
  usr <- par("usr")
  on.exit(par(usr))
  par(usr = c(0, 1, 0, 1))
  r <- abs(cor(x, y))
  txt <- format(c(r, 0.123456789), digits = digits)[1]
  txt <- paste0(prefix, txt)
  if(missing(cex.cor)) cex.cor <- 0.5/strwidth(txt)
  text(0.5, 0.5, txt, cex = cex.cor * r)
}
```

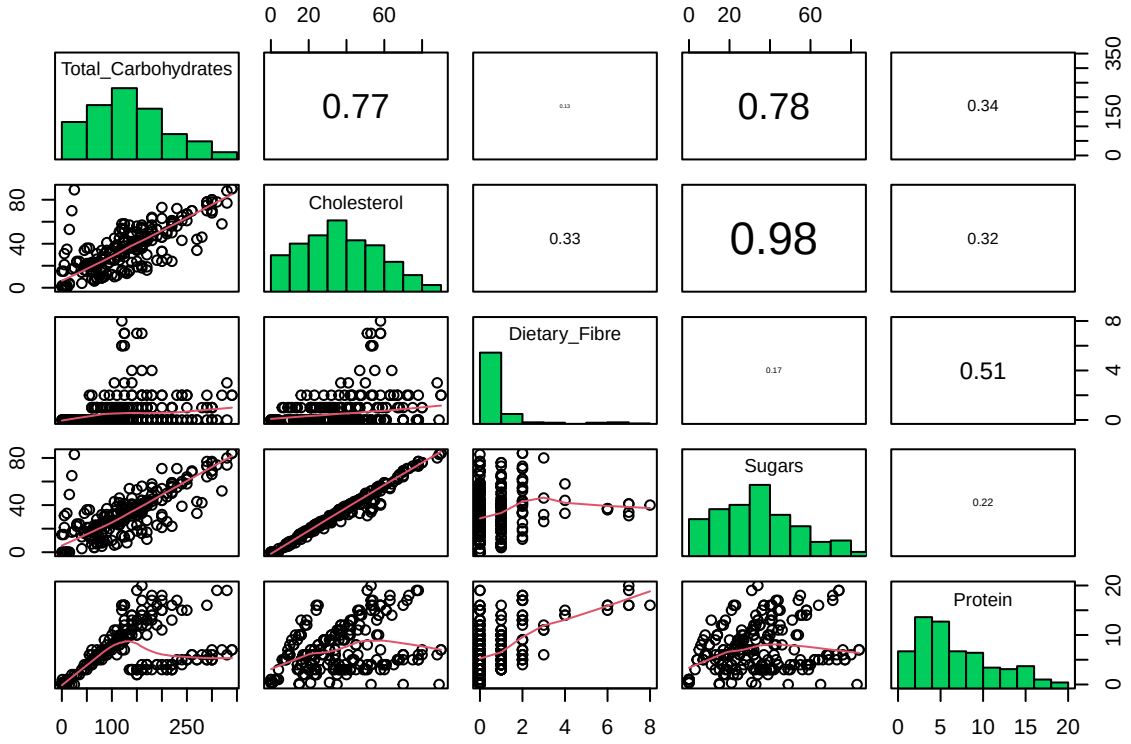
Now we can split the data into 3 groups in order to have a better visualization.

```
data_num1 <- data_num[, 1:5]
data_num2 <- data_num[, 6:10]
data_num3 <- data_num[, 11:15]

pairs(data_num1,
      diag.panel = panel.hist,
      upper.panel = panel.cor,
      lower.panel = panel.smooth,
```

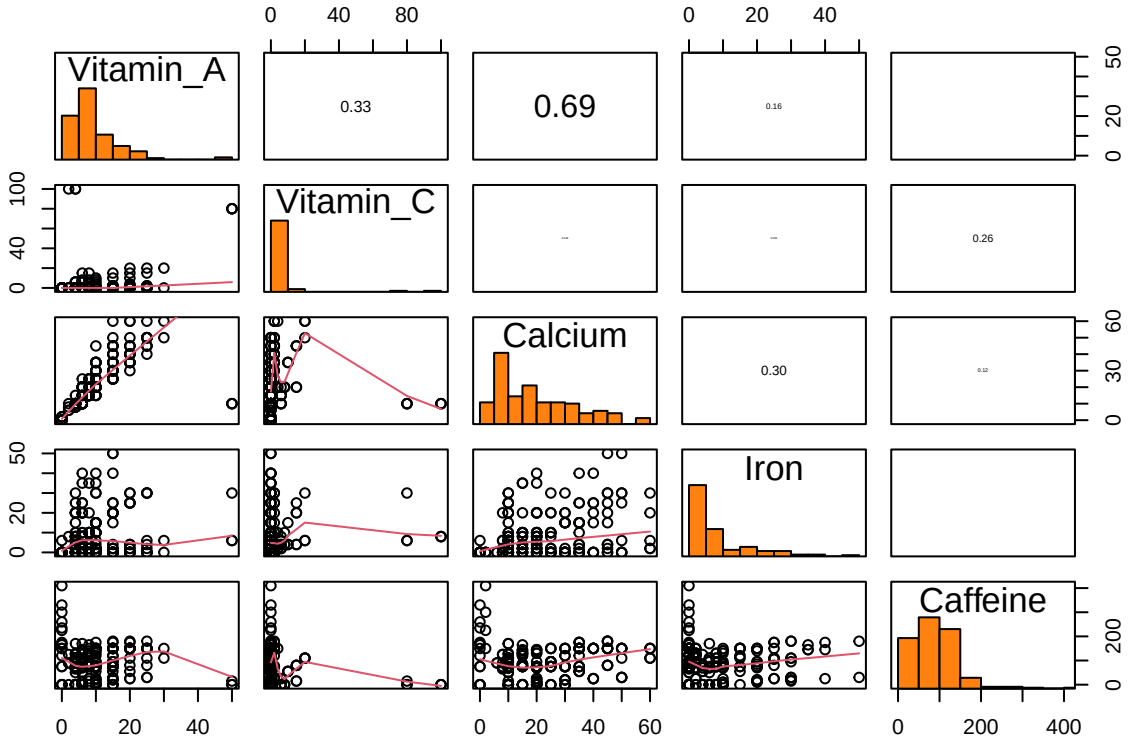
```
      colour = "#4ea5ff")
```




```
pairs(data_num2,
      diag.panel = panel.hist,
      upper.panel = panel.cor,
      lower.panel = panel.smooth,
      colour = "#00cd5c")
```



```r
pairs(data_num3,
      diag.panel = panel.hist,
      upper.panel = panel.cor,
      lower.panel = panel.smooth,
      colour = "#ff810f")
```

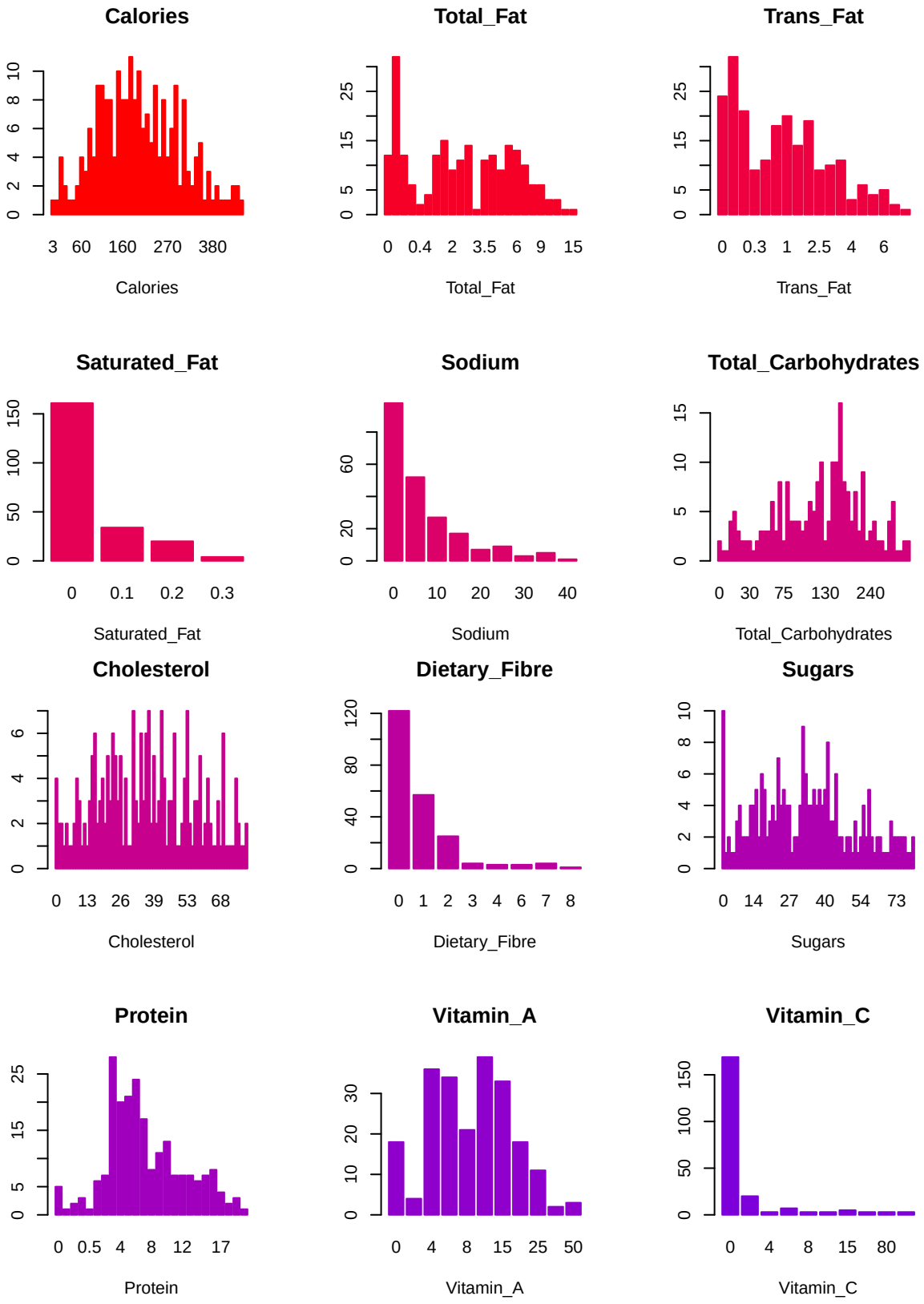


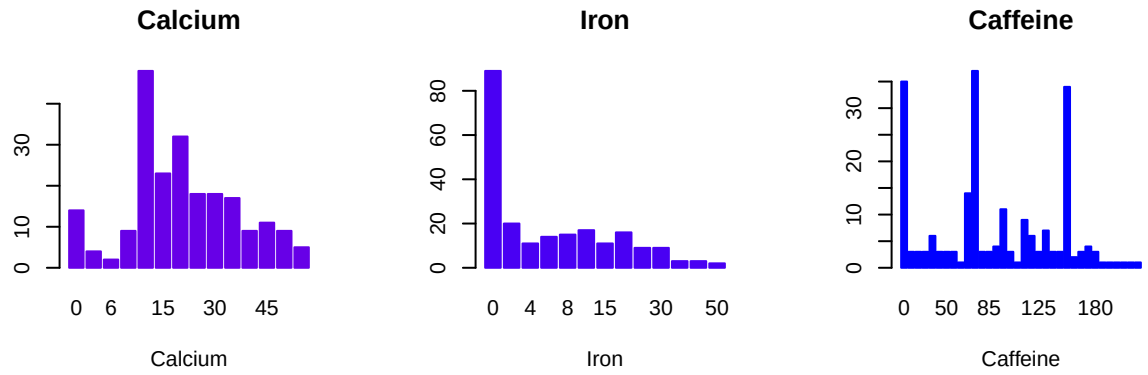
ADD COMMENTS ON THE GRAPH

5.3 Barplot

We will plot a barplot of the data.

```r
# Barplot of the data
par(mfrow = c(2, 3))
for (i in 1:ncol(data_num)) {
  barplot(table(data_num[, i]), main = colnames(data_num)[i],
          xlab = colnames(data_num)[i], col = col[i], border = col[i])
}
```


Calories
0 2 4 6 8 10
3 60 160 270 380
Calories
Total_Fat
0 5 15 25
0 0.4 2 3.5 6 9 15
Total_Fat
Trans_Fat
0 5 15 25
0 0.3 1 2.5 4 6
Trans_Fat
Saturated_Fat
0 50 100 150
0 0.1 0.2 0.3
Saturated_Fat
Sodium
0 20 60
0 10 20 30 40
Sodium
Total_Carbohydrates
0 5 10 15
0 30 75 130 240
Total_Carbohydrates
Cholesterol
0 2 4 6
0 13 26 39 53 68
Cholesterol
Dietary_Fibre
0 40 80 120
0 1 2 3 4 6 7 8
Dietary_Fibre
Sugars
0 2 4 6 8 10
0 14 27 40 54 73
Sugars
Protein
0 5 15 25
0 0.5 4 8 12 17
Protein
Vitamin_A
0 10 20 30
0 4 8 15 25 50
Vitamin_A
Vitamin_C
0 50 100 150
0 4 8 15 80
Vitamin_C

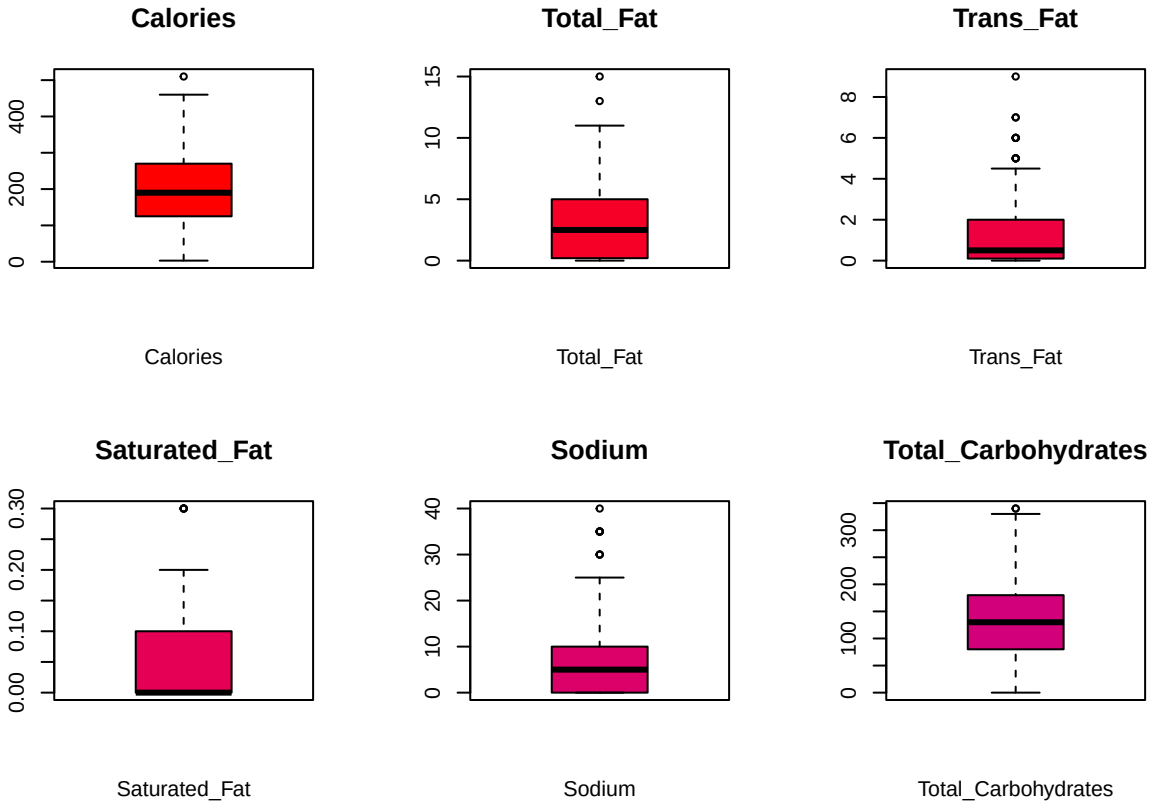


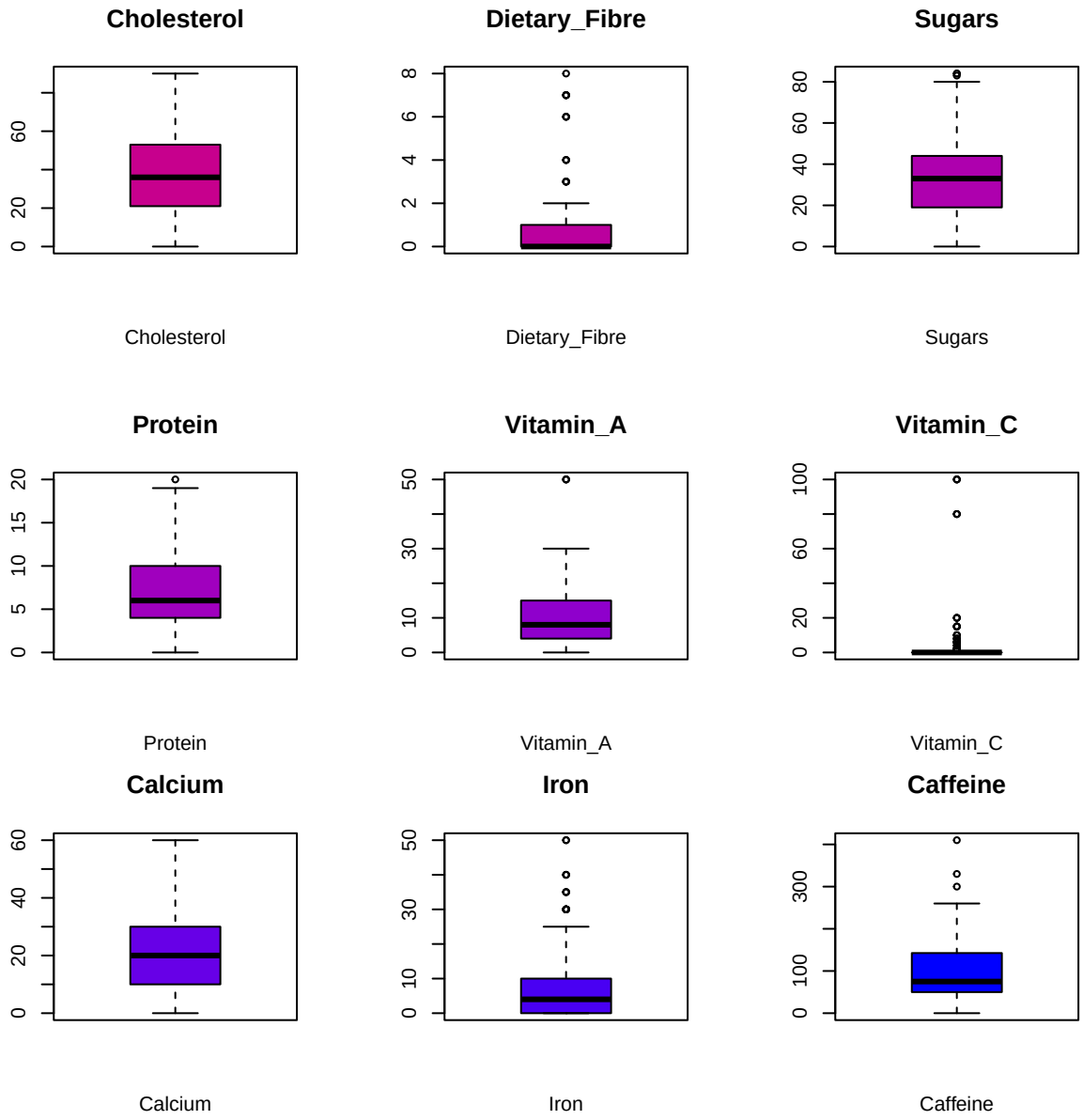
ADD COMMENTS ON THE GRAPH

5.4 Boxplot

We will plot a boxplot of the data.

```
# Boxplot of the data
par(mfrow = c(2, 3))
for (i in 1:ncol(data_num)) {
  boxplot(data_num[, i], main = colnames(data_num)[i],
          xlab = colnames(data_num)[i], col = col[i])
}
```





ADD COMMENTS ON THE GRAPH