CMTH 642 Data Analytics: Advanced Methods

Assignment 1 (10%)

[Ilhak Park]

[DJ0 & 501072432]

```r
micro <- read.csv(file= "USDA_Micronutrients.csv", sep= ",")
macro <- read.csv(file="USDA_Macronutrients.csv", sep =",")
```

1. Read the csv files in the folder. (3 points)

```r
USDA <- merge(macro,micro)
summary(USDA)
```

2. Merge the data frames using the variable “ID”. Name the Merged Data Frame “USDA”. (6 points)

```
##        ID        Description           Calories        Protein
##  Min.   : 1001   Length:7057        Min.   :  0.0   Min.   : 0.00
##  1st Qu.: 8387   Class :character   1st Qu.: 85.0   1st Qu.: 2.29
##  Median :13293   Mode  :character   Median :181.0   Median : 8.20
##  Mean   :14258                      Mean   :219.7   Mean   :11.71
##  3rd Qu.:18336                      3rd Qu.:331.0   3rd Qu.:20.43
##  Max.   :93600                      Max.   :902.0   Max.   :88.32
##
##     TotalFat       Carbohydrate        Sodium          Cholesterol
##  Min.   :  0.00   Min.   :  0.00   Length:7057        Min.   :   0.00
##  1st Qu.:  0.72   1st Qu.:  0.00   Class :character   1st Qu.:   0.00
##  Median :  4.37   Median :  7.13   Mode  :character   Median :   3.00
##  Mean   : 10.32   Mean   : 20.70                      Mean   :  41.55
##  3rd Qu.: 12.70   3rd Qu.: 28.17                      3rd Qu.:  69.00
##  Max.   :100.00   Max.   :100.00                      Max.   :3100.00
##                                                       NA's   :287
##      Sugar           Calcium             Iron          Potassium
##  Min.   : 0.000   Min.   :   0.00   Min.   :  0.000   Length:7057
##  1st Qu.: 0.000   1st Qu.:   9.00   1st Qu.:  0.520   Class :character
##  Median : 1.395   Median :  19.00   Median :  1.330   Mode  :character
##  Mean   : 8.257   Mean   :  73.53   Mean   :  2.828
##  3rd Qu.: 7.875   3rd Qu.:  56.00   3rd Qu.:  2.620
##  Max.   :99.800   Max.   :7364.00   Max.   :123.600
##  NA's   :1909     NA's   :135       NA's   :122
##     VitaminC          VitaminE          VitaminD
##  Min.   :   0.000   Min.   :  0.000   Min.   :  0.0000
##  1st Qu.:   0.000   1st Qu.:  0.120   1st Qu.:  0.0000
##  Median :   0.000   Median :  0.270   Median :  0.0000
##  Mean   :   9.436   Mean   :  1.488   Mean   :  0.5769
##  3rd Qu.:   3.100   3rd Qu.:  0.710   3rd Qu.:  0.1000
##  Max.   :2400.000   Max.   :149.400   Max.   :250.0000
##  NA's   :331        NA's   :2719      NA's   :2833
```

```
sapply (USDA, class)
```

3. Check the datatypes of the attributes. Delete the commas in the Sodium and Potasium records. Assign Sodium and Potasium as numeric data types. (6 points)

```
##          ID  Description     Calories      Protein     TotalFat Carbohydrate
##   "integer"  "character"    "integer"    "numeric"    "numeric"    "numeric"
##      Sodium  Cholesterol        Sugar      Calcium         Iron    Potassium
## "character"    "integer"    "numeric"    "integer"    "numeric"  "character"
##    VitaminC     VitaminE     VitaminD
##   "numeric"    "numeric"    "numeric"
```

```
USDA$Sodium <- gsub(",", "", USDA$Sodium)
USDA$Potassium <-gsub(",", "", USDA$Potassium)
USDA$Sodium <- as.numeric(USDA$Sodium)
USDA$Potassium <- as.numeric(USDA$Potassium)
```

```
USDA <- USDA[(apply (is.na(USDA),1,sum)) <= 4,]
nrow(USDA)
```

4. Remove records (rows) with missing values in more than 4 attributes (columns). How many records remain in the data frame? (6 points)

```
## [1] 6887
```

```
#The remaining records are 6,887.
```

```
USDA$Sugar[is.na(USDA$Sugar)] = mean(USDA$Sugar[!is.na(USDA$Sugar)])

USDA$VitaminE[is.na(USDA$VitaminE)] = mean(USDA$VitaminE[!is.na(USDA$VitaminE)])
```

```
USDA$VitaminD[is.na(USDA$VitaminD)] = mean(USDA$VitaminD[!is.na(USDA$VitaminD)])

#checking if 0
#USDA$Sugar[is.na(USDA$Sugar)]
#USDA$VitaminE[is.na(USDA$VitaminE)]
#USDA$VitaminD[is.na(USDA$VitaminD)]
```

5. For records with missing values for Sugar, Vitamin E and Vitamin D, replace missing values with mean value for the respective variable. (6 points)

```
USDAclean <- USDA[complete.cases(USDA), ]
nrow(USDAclean)
```

6. With a single line of code, remove all remaining records with missing values. Name the new Data Frame "USDAclean". How many records remain in the data frame? (6 points)

```
## [1] 6310
```

```
#The remaining records are 6,310.
```

```
max(USDAclean$Sodium)
```

7. Which food has the highest sodium level? (6 points)

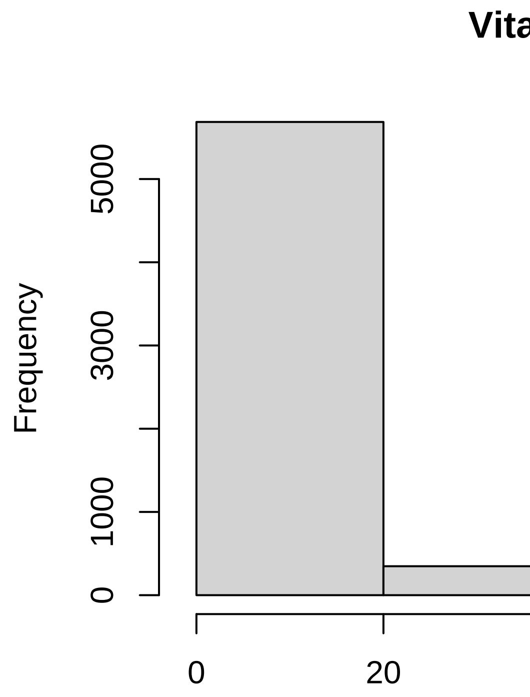
```
## [1] 38758
```

```
USDAclean$Description[USDAclean$Sodium ==max(USDAclean$Sodium)]
```

```
## [1] "SALT,TABLE"
```

```
#SALT,TABLE has the highest sodium level that is 38,758.
```

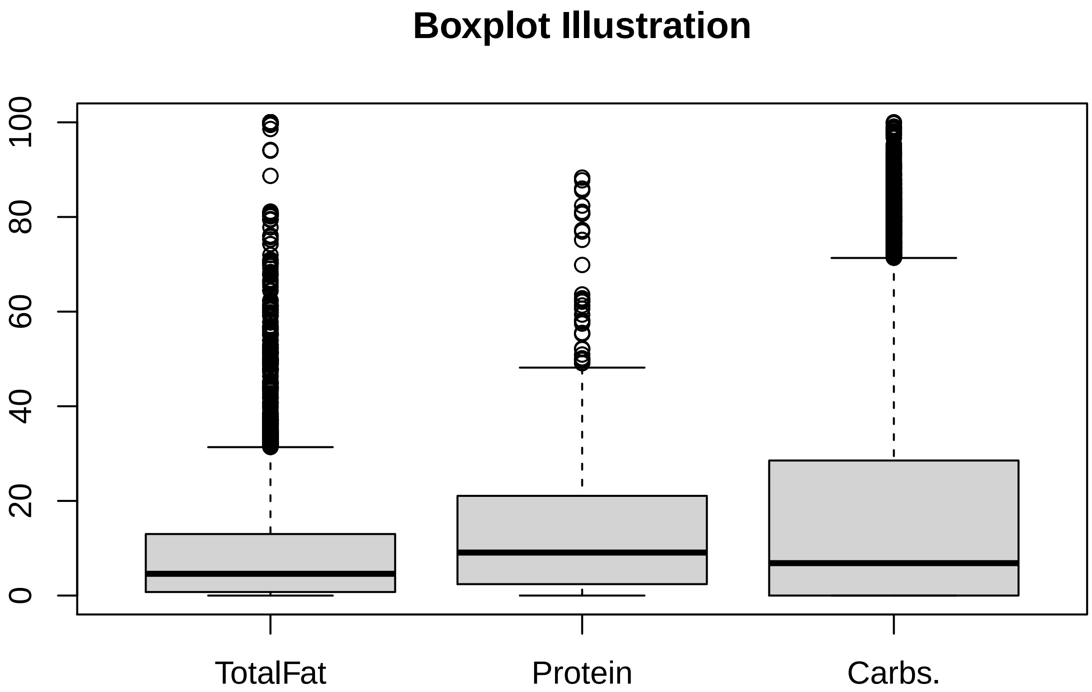
```
hist(USDAclean$VitaminC , xlim=c(0,100), breaks= 100, xlab = "Vitamin C", main ="Vitamin C Distribution
```



8. Create a histogram of Vitamin C distribution in foods. (6 points)

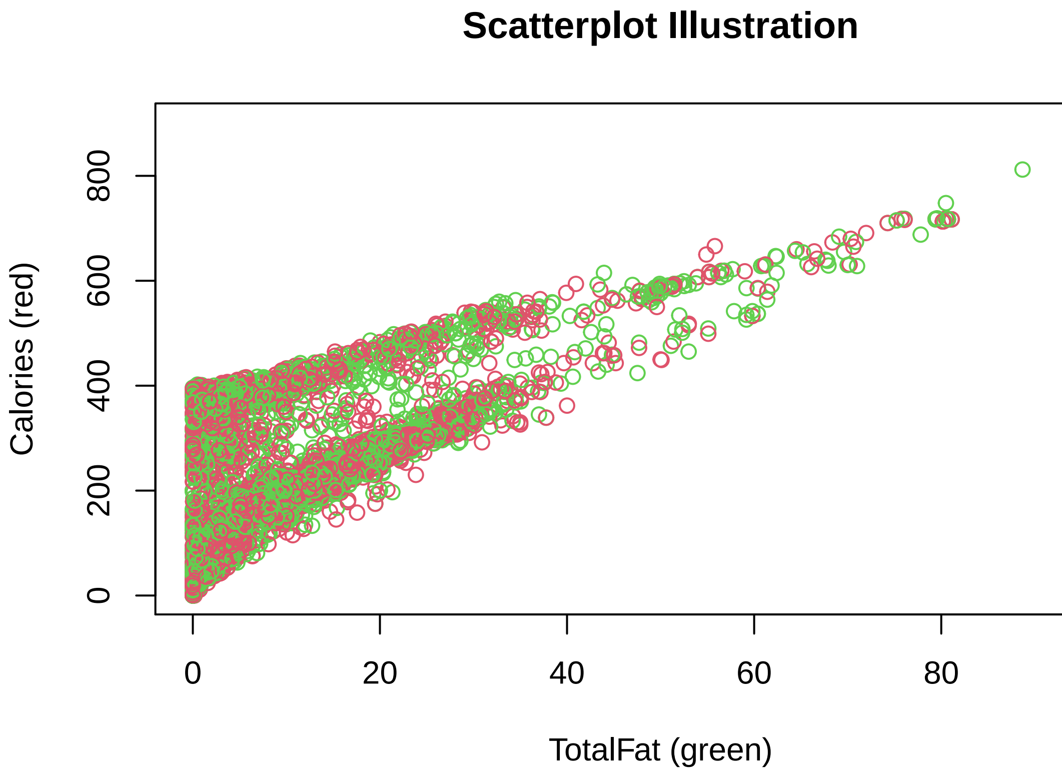
```
boxplot(USDAclean$TotalFat, USDAclean$Protein, USDAclean$Carbohydrate,main="Boxplot Illustration", name
```

9. Create a boxplot to illustrate the distribution of values for TotalFat, Protein and Carbohydrate. (6 points)



```
plot(USDAclean$TotalFat, USDAclean$Calories, main ="Scatterplot Illustration",xlab="TotalFat (green)",yl
```

10. Create a scatterplot to illustrate the relationship between a food's TotalFat content and its Calorie content. (6 points)



```
#High Sodium
USDAclean$HighSodium[USDAclean$Sodium > mean(USDAclean$Sodium)] <- 1
USDAclean$HighSodium[USDAclean$Sodium <= mean(USDAclean$Sodium)] <- 0

#High Calories
USDAclean$HighCalories[USDAclean$Calories > mean(USDAclean$Calories)] <- 1
USDAclean$HighCalories[USDAclean$Calories <= mean(USDAclean$Calories)] <- 0

#High Protein
USDAclean$HighProtein[USDAclean$Protein > mean(USDAclean$Protein)] <- 1
USDAclean$HighProtein[USDAclean$Protein <= mean(USDAclean$Protein)] <- 0

#High Sugar
USDAclean$HighSugar[USDAclean$Sugar > mean(USDAclean$Sugar)] <- 1
USDAclean$HighSugar[USDAclean$Sugar <= mean(USDAclean$Sugar)] <- 0

#High Fat
USDAclean$HighFat[USDAclean$TotalFat > mean(USDAclean$TotalFat)] <- 1
USDAclean$HighFat[USDAclean$TotalFat <= mean(USDAclean$TotalFat)] <- 0

High<-apply(USDAclean[c("HighSodium", "HighFat")], 1, sum)
table(High)
```

11. Add a variable to the data frame that takes value 1 if the food has higher sodium than average, 0 otherwise. Call this variable HighSodium. Do the same for High Calories, High Protein, High Sugar, and High Fat. How many foods have both high sodium and high fat? (8 points)

```
## High
##    0    1    2
## 3233 2433  644
```

```
#644 foods have both high sodium and high fat.
```

```
tapply(USDAclean$Iron, USDAclean$HighProtein, mean)
```

12. Calculate the average amount of iron, for high and low protein foods. (8 points)

```
##        0        1
## 2.696634 3.069541
```

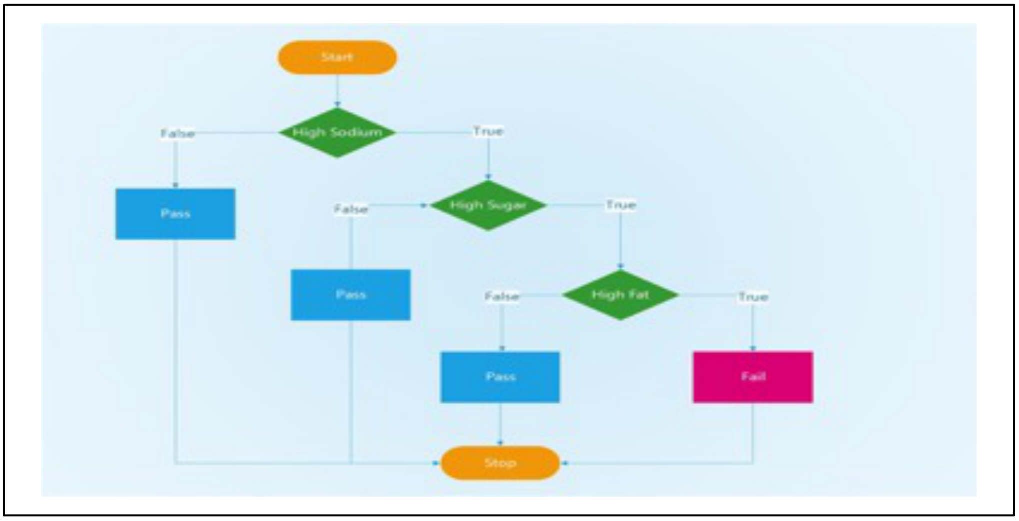
```
#The average amount of iron for high protein food is 3.069541.
#The average amount of iron for low protein food is 2.696634.
```

```
require(jpeg)
```

13. Create a script for a "HealthCheck" program to detect unhealthy foods. Use the algorithm flowchart below as a basis for this script. (8 points)

```
## Loading required package: jpeg
```

```
img<-readJPEG("HealthCheck.jpg")
plot(1:4, ty = 'n', ann = F, xaxt = 'n', yaxt = 'n')
rasterImage(img,1,1,4,4)
```



```r
HealthCheck <- function(food){if (food$HighSodium ==0) return ("Pass") else if (food$HighSugar ==0) retu
```

```r
for (index in 1:nrow(USDAclean)) {USDAclean$HealthCheck[index] = HealthCheck(USDAclean[index,])}
```

14. Add a new variable called HealthCheck to the data frame using the output of the function. (8 points)

```r
table(USDAclean$HealthCheck)
```

15. How many foods in the USDAclean data frame fail the HealthCheck? (8 points)

```
##
## Fail Pass
##  237 6073
```

```r
#237 foods fail the HealthCheck.
```

```r
write.csv(USDAclean, "USDAclean_Park")
```

16. Save your final data frame as “USDAclean_ [your last name].” (3 points) This is the end of Assignment 1

Ceni Babaoglu, PhD