Homework 0

Yiming Gao (NetID: yimingg2)

2017/1/19

Exercise 1

For this exercise, we will use the `diabetes` dataset from the `faraway` package.

(a) Install and load the `faraway` package. **Do not** include the install command in your .Rmd file. (If you do it will install the package every time you knit your file.) **Do** include the command to load the package into your environment.

```
library(faraway)
library(readr)
```

(b) Coerce the data to be a tibble instead of a basic data frame. (You will need the `tibble` package to do so.) How many observations are in this dataset? How many variables?

```
library(tibble)
as_data_frame(diabetes)
```

```
## # A tibble: 403 <U+00D7> 19
##       id  chol stab.glu   hdl ratio glyhb   location   age gender height
## *  <int> <int>    <int> <int> <dbl> <dbl>     <fctr> <int> <fctr>  <int>
## 1   1000   203       82    56   3.6  4.31 Buckingham    46 female     62
## 2   1001   165       97    24   6.9  4.44 Buckingham    29 female     64
## 3   1002   228       92    37   6.2  4.64 Buckingham    58 female     61
## 4   1003    78       93    12   6.5  4.63 Buckingham    67   male     67
## 5   1005   249       90    28   8.9  7.72 Buckingham    64   male     68
## 6   1008   248       94    69   3.6  4.81 Buckingham    34   male     71
## 7   1011   195       92    41   4.8  4.84 Buckingham    30   male     69
## 8   1015   227       75    44   5.2  3.94 Buckingham    37   male     59
## 9   1016   177       87    49   3.6  4.84 Buckingham    45   male     69
## 10  1022   263       89    40   6.6  5.78 Buckingham    55 female     63
## # ... with 393 more rows, and 9 more variables: weight <int>,
## #   frame <fctr>, bp.1s <int>, bp.1d <int>, bp.2s <int>, bp.2d <int>,
## #   waist <int>, hip <int>, time.ppn <int>
```

There are 403 observations in and 19 vairables in `diabetes` dataset.

(c) Which variables are factor variables?

```r
sapply(diabetes, is.factor)
```

```
##       id     chol stab.glu      hdl    ratio    glyhb location      age
##    FALSE    FALSE    FALSE    FALSE    FALSE    FALSE     TRUE    FALSE
##   gender   height   weight    frame    bp.1s    bp.1d    bp.2s    bp.2d
##     TRUE    FALSE    FALSE     TRUE    FALSE    FALSE    FALSE    FALSE
##    waist      hip time.ppn
##    FALSE    FALSE    FALSE
```

We can conclude that location, gender and frame are factor variables.

(d) What is the mean HDL level (High Density Lipoprotein) of individuals in this sample?

```r
mean(diabetes$hdl, na.rm = TRUE)
```

```
## [1] 50.44527
```

The mean of HDL removing NA values is 50.445.

(e) What is the standard deviation of total cholesterol of individuals in this sample?

```r
sd(diabetes$chol,na.rm = TRUE)
```

```
## [1] 44.44556
```

The standard deviation of total choloesterol of individuals is 44.446.

(f) What is the range of ages of individuals in this sample?

```r
range(diabetes$age)
```

```
## [1] 19 92
```

The range of ages in this sample is 92-19 = 73.

(g) What is the mean HDL of females in this sample?

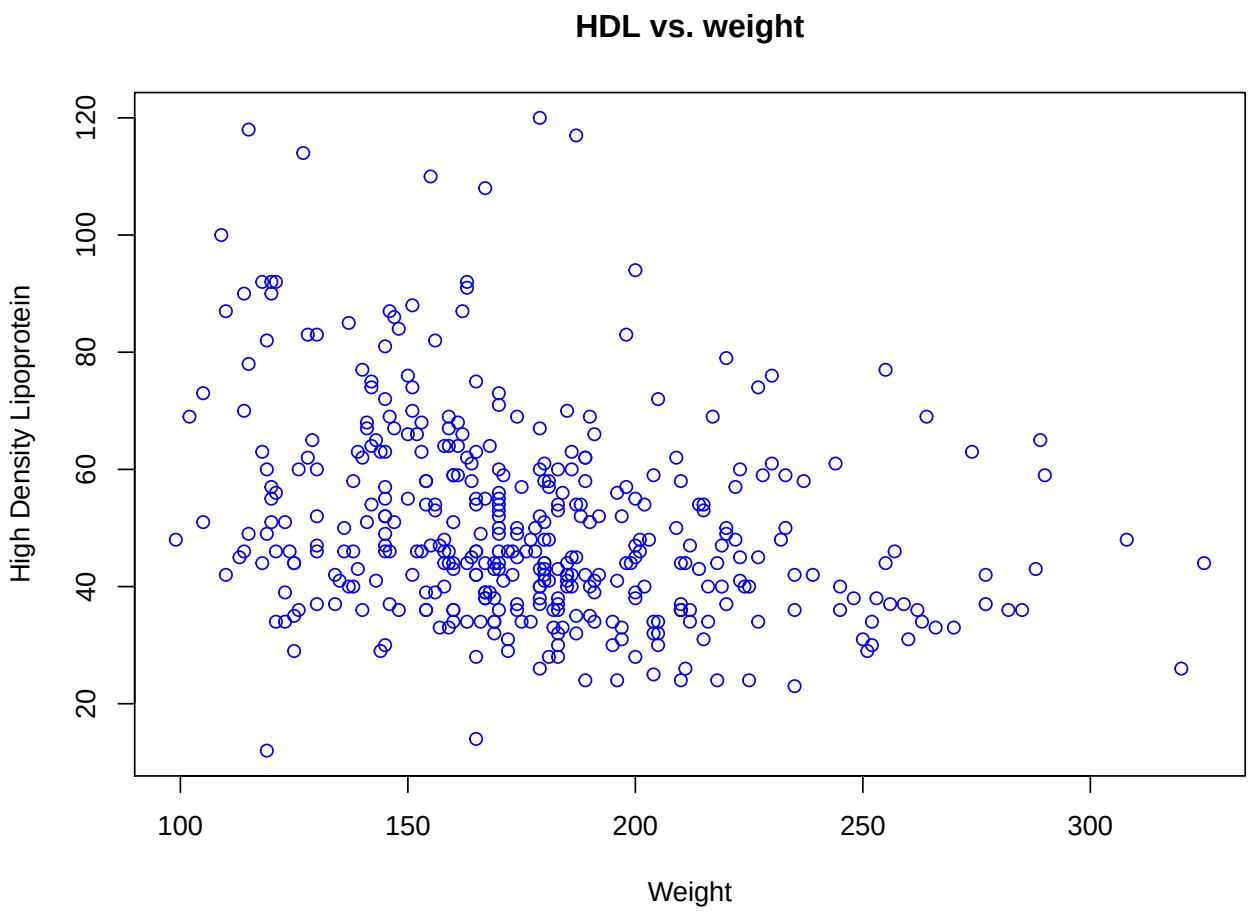
```r
mean(diabetes$hdl[which(diabetes$gender == "female")], na.rm = TRUE)
```

```
## [1] 52.11111
```

The mean hdl of females is 52.111.

(h) Create a scatterplot of HDL (y-axis) vs weight (x-axis). Use a non-default color for the points. (Also, be sure to give the plot a title and label the axes appropriately.) Based on the scatterplot, does there seem to be a relationship between the two variables? Briefly explain.

```
plot(diabetes$weight, diabetes$hdl, main = "HDL vs. weight",
     xlab = "Weight", ylab = "High Density Lipoprotein", col = "blue")
```



There seems to be a negative relationship between two variables. As weight increases, HDL decreases.

(i) Create a scatterplot of total cholesterol (y-axis) vs weight (x-axis). Use a non-default color for the points. (Also, be sure to give the plot a title and label the axes appropriately.) Based on the scatterplot, does there seem to be a relationship between the two variables? Briefly explain.

```
plot(diabetes$weight, diabetes$chol, main = "chol vs. weight",
     xlab = "Weight", ylab = "Total Cholesterol", col = "purple")
```

There seems to be no trend between two variables.

Exercise 2

For this exercise we will use the data stored in `nutrition.csv`. It contains the nutritional values per serving size for a large variety of foods as calculated by the USDA. It is a cleaned version totaling 5138 observations and is current as of September 2015.

The variables in the dataset are:

- `ID`
- `Desc` - Short description of food
- `Water` - in grams
- `Calories` - in kcal
- `Protein` - in grams
- `Fat` - in grams
- `Carbs` - Carbohydrates, in grams
- `Fiber` - in grams
- `Sugar` - in grams
- `Calcium` - in milligrams

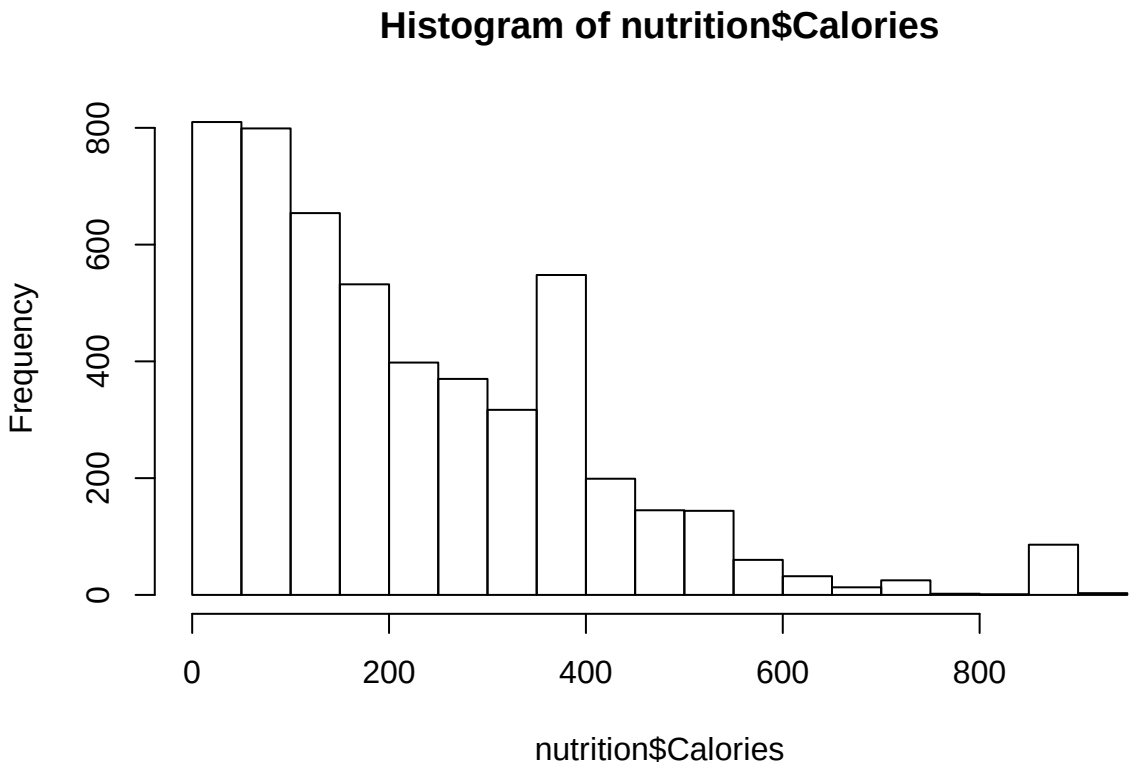
- `Potassium` - in milligrams
- `Sodium` - in milligrams
- `VitaminC` - Vitamin C, in milligrams
- `Chol` - Cholesterol, in milligrams
- `Portion` - Description of standard serving size used in analysis

(a) Create a histogram of `Calories`. Do not modify `R`'s default bin selection. Make the plot presentable. Describe the shape of the histogram. Do you notice anything unusual?

```
nutrition = read_csv("nutrition.csv")
```

```
## Parsed with column specification:
## cols(
##   ID = col_integer(),
##   Desc = col_character(),
##   Water = col_double(),
##   Calories = col_integer(),
##   Protein = col_double(),
##   Fat = col_double(),
##   Carbs = col_double(),
##   Fiber = col_double(),
##   Sugar = col_double(),
##   Calcium = col_integer(),
##   Potassium = col_integer(),
##   Sodium = col_integer(),
##   VitaminC = col_double(),
##   Chol = col_integer(),
##   Portion = col_character()
## )
```

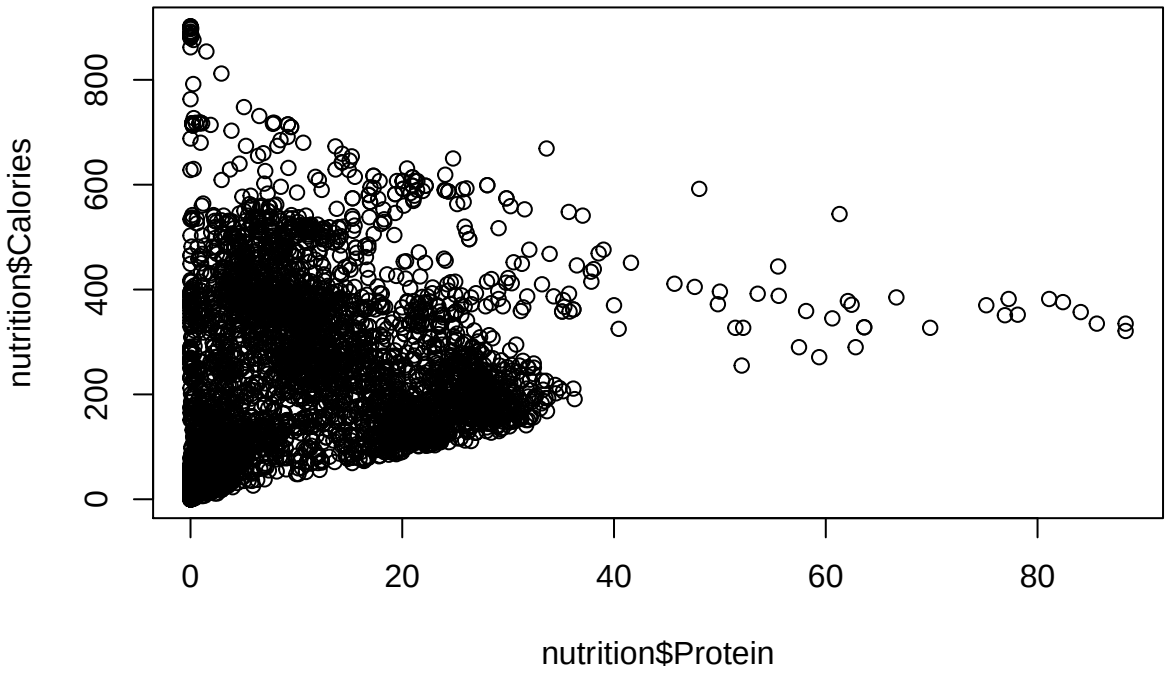
```
hist(nutrition$Calories)
```



We can see a clearly linear decreasing trend from the histogram of `Calories`.

(b) Create a scatterplot of calories (y-axis) vs protein (x-axis). Make the plot presentable. Do you notice any trends? Do you think that knowing only the protein content of a food, you could make a good prediction of the calories in the food?

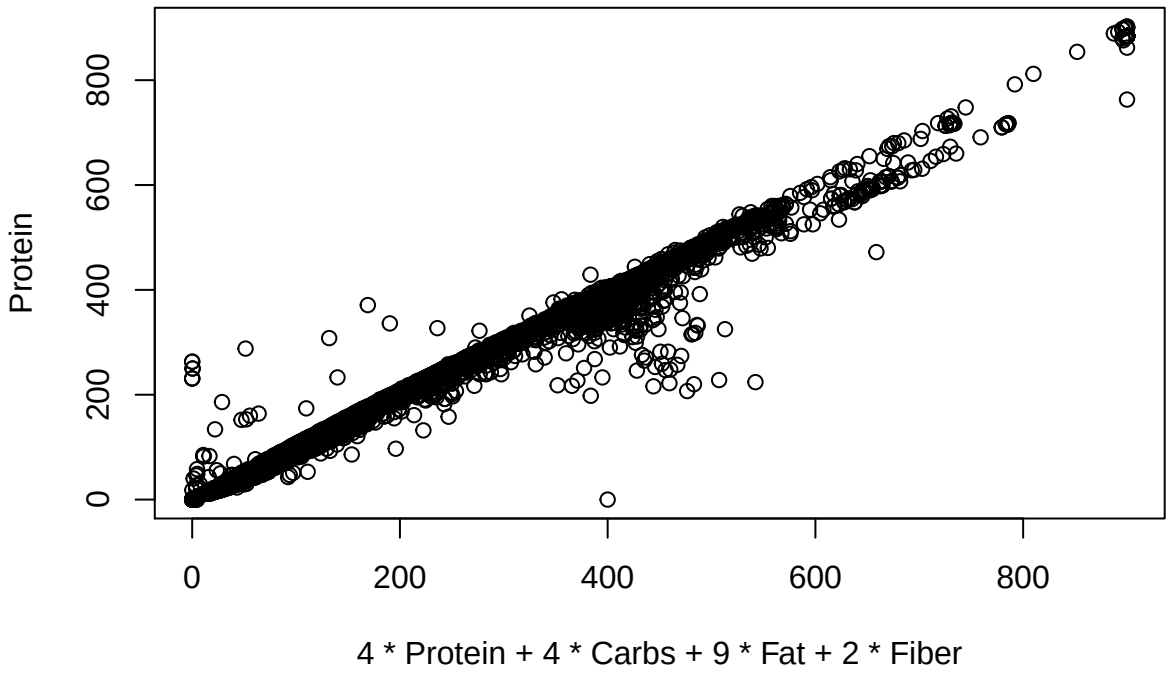
```
plot(nutrition$Protein, nutrition$Calories, main = "Protein vs Calories")
```



We can see that as protein increases, Calories gradually converges to 400. However, we cannot make a good prediction based on the given `Protein`, because one `Protein` value may correspond to two different `Calories` values.

(c) Create a scatterplot of `Calories` (y-axis) vs `4 * Protein + 4 * Carbs + 9 * Fat + 2 * Fiber` (x-axis). Make the plot presentable. You will either need to add a new variable to the data frame, or, use the `I()` function in your formula in the call to `plot()`. If you are at all familiar with nutrition, you may realize that this formula calculates the calorie count based on the protein, carbohydrate, and fat values. You'd expect then that the result here is a straight line. Is it? If not, can you think of any reasons why it is not?

```
nutrition$newx = 4*nutrition$Protein + 4* nutrition$Carbs+9*nutrition$Fat+2*nutrition$Fiber
plot(nutrition$newx, nutrition$Calories, xlab = "4 * Protein + 4 * Carbs + 9 * Fat + 2 * Fiber", ylab =
```



We can see a almost straight line in the plot. Maybe it involves some biology knowledge.

Exercise 3

For each of the following parts, use the following vectors:

```
a <- 1:10
b <- 10:1
c <- rep(1, times = 10)
d <- 2 ^ (1:10)
```

(a) Write a function called `sum_of_squares`.

- Arguments:
  - A vector of numeric data `x`.
- Output:
  - The sum of the squares of the elements of the vector. $\sum_{i=1}^{n} x_i^2$

Provide your function, as well as the result of running the following code:

```
sum_of_squares <- function(x){
  s = sum(x^2)
  return(s)
}

sum_of_squares(x = a)
```

```
## [1] 385
```

```
sum_of_squares(x = c(c, d))
```

```
## [1] 1398110
```

(b) Write a function called `rms_diff`.

- Arguments:
  - A vector of numeric data `x`.
  - A vector of numeric data `y`.
- Output:
  - $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2}$

Provide your function, as well as the result of running the following code:

```
rms_diff <- function(x, y){
  n = length(x)
  output = - sqrt((sum((x - y)^2))/n)
  return(output)
}

rms_diff(x = a, y = b)
```

```
## [1] -5.744563
```

```
rms_diff(x = d, y = c)
```

```
## [1] -373.3655
```

```
rms_diff(x = d, y = 1)
```

```
## [1] -373.3655
```

```
rms_diff(x = a, y = 0) ^ 2 * length(a)
```

```
## [1] 385
```