Explore the Hot vs. Cold Cereals Data Set Using R and R Studio

Didem Bulut Aykurt

MIS510-1 – Data Mining and Visualization

Colorado State University-Global Campus

Dr. Emmanuel Tsukerman

Now 27,2022

Module 2-Option 1

This report is a data analysis of the Cereal dataset. I aim to write an R code for summary statistics such as mean, standard deviation, min, max, median, length, and the sum of missing values. Additionally, create a histogram and box plot.

The name of the dataset is Cereals dataset describes hot and cold cereals containing each cereal's nutrition and 77 different cereal products—the data sources from CSU-Global. The dataset has 77 observations and 16 variables or columns, including 13 numerical and three-character variables.

Data Description of Listings

- Name: product name
- Mfr: Manufacturers (A= American Home Food, G=General Mills, K=Kellogs, N=Nabisco, P=Post, Q=Quaker Oats, R=Ralston Purina)
- type: Cereal type (C=cold, H=Hot)
- calories: per serving calories
- protein: grams of protein
- fat: grams of fat
- sodium: milligrams of sodium
- fiber: grams of dietary fiber
- carbo: grams of carbohydrates
- sugar: grams of sugars
- potass: milligrams of potassium
- vitamins: vitamins and minerals (0, 25, or 100)
- shelf: display shelf (1,2,3)
- weight: weight in ounces of one serving
- cups: number of cups in one serving

- rating: a rating of the cereals

First, load the dataset, then view the statistical summary with the summary () function. This function shows each statistical result as a mean value to see an average data point for each variable. Sodium has the highest mean, 159 per serving. Deep end recommends limiting sodium to 1,500 mg daily. Following, calories mean at 106. Mean and median distance is essential for the distribution. The median is much bigger than the mean as the distribution is left-skewed(median>mean). The median is much smaller than the mean and the right-skewed distribution(median<mean). The median is equal to or close to 0.7, the normal distribution(median=mean). The left skew variables are calories (median at 110> mean at 106) and sodium (median at 180> mean at 159). The right skew of the variables is potassium (median at 90< mean at 98), vitamin (median at 25< mean at 28), and rating (median at 40< mean at 42). The standard distribution variables are protein, fat, fiber, sugar, shelf, weight, and cups.

A histogram is an excellent representation of category frequency distribution. The histogram plot presents a rectangle for each group of data. Figure 4 has all quantitative columns' results. The par() function help to show all variables in one view, makes easy to compare. I used two types of histogram charts. One is the hist() function that works with side-by-side plots, and another is the ggplot2 library. As we talked about distribution on statistical results, also histogram shows the distribution.

A Box plot is an excellent tool to see all the statistical points on graphs like min, median, mode, Q1, Q3, and outliers. Cold cereal Q1 at 100 per serving calorie, and the hot cereal has just one point data at 100. Cold cereal has more data points than hot cereal, and cold cereal has negative and positive side outliers.

Concern

All code is understandable and easy to figure out, but ggplot needs clarification. The function gives more errors like the carbon variable error message "Problem while computing aesthetics." I applied for many

numbers, binwidth, center, and the same error type. I need more detail for the ggplot on why I have this type of error.

Figure 1: *Import the Cereal dataset and summary statistics result into R Studio.*

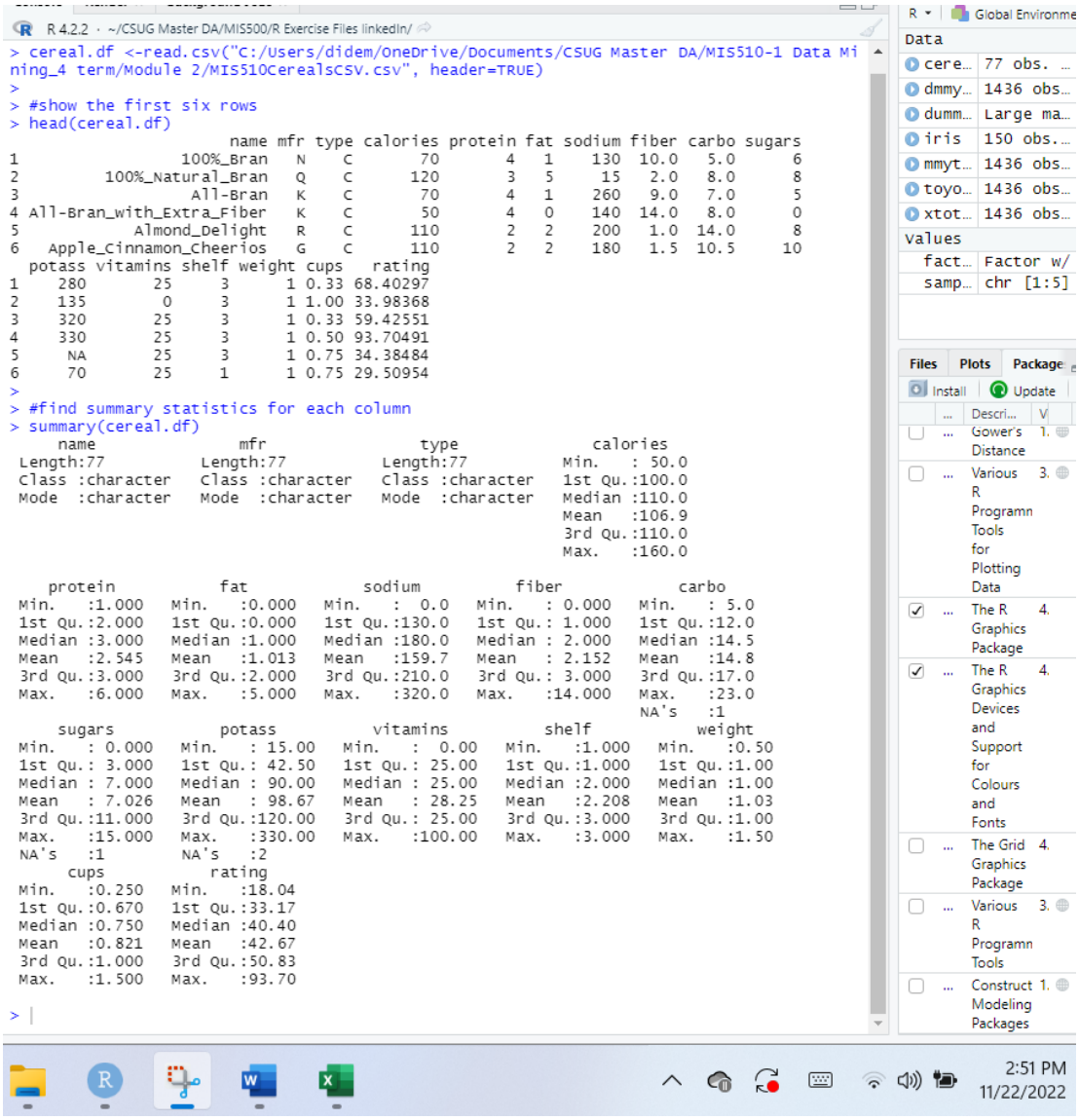
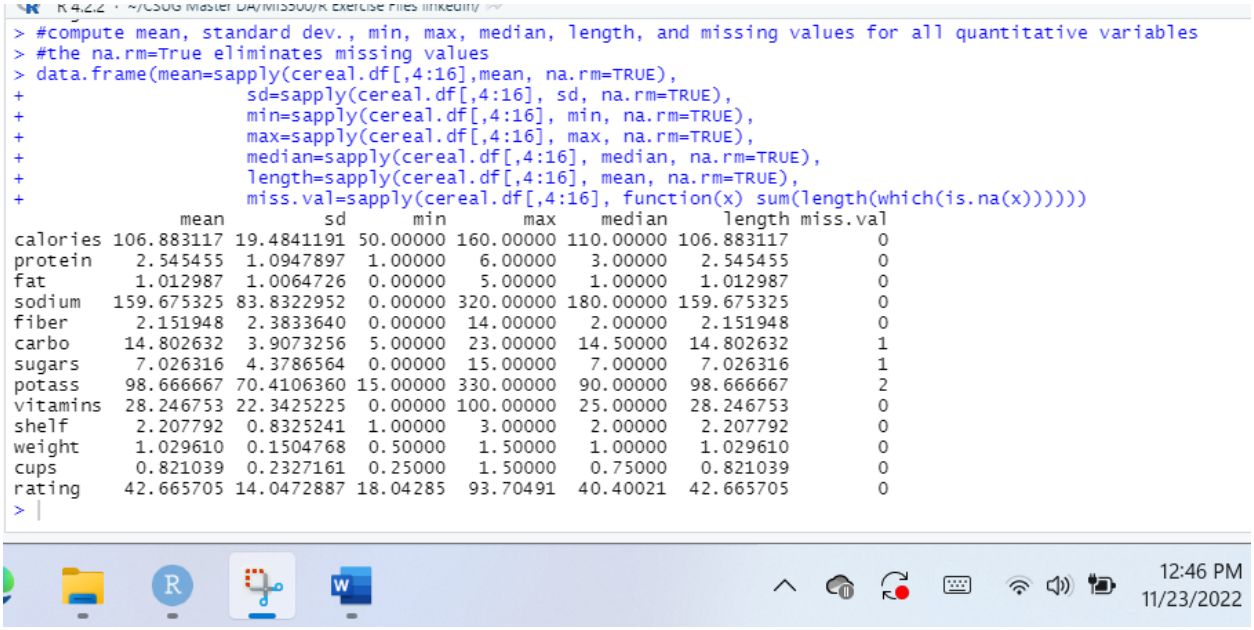
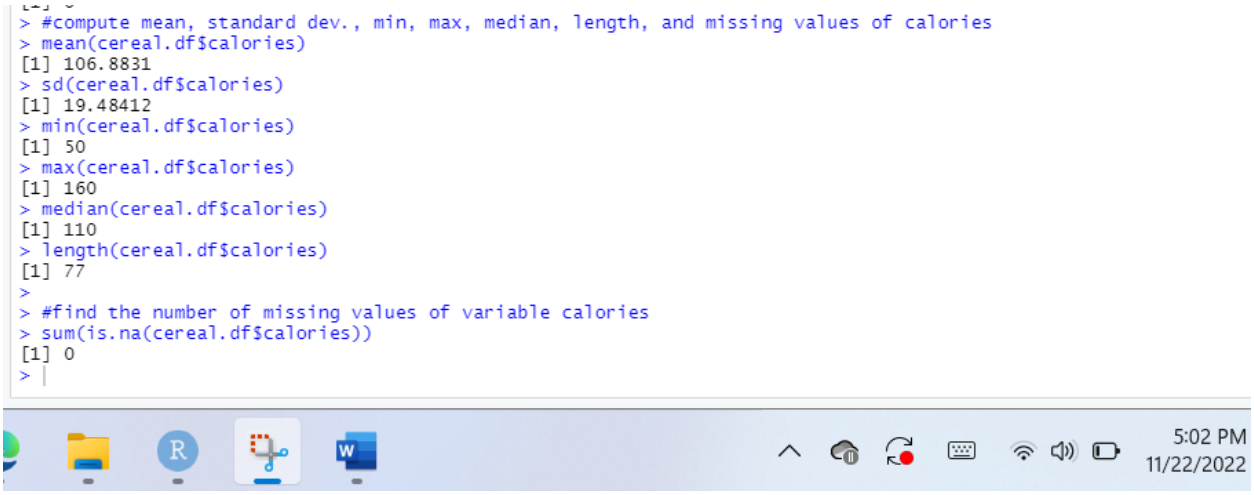


Figure 2: *Display all quantitative variables' statistical results with data.frame() function and use the na.rm=True function to eliminate missing values in R Studio.*



```
> #compute mean, standard dev., min, max, median, length, and missing values for all quantitative variables
> #the na.rm=True eliminates missing values
> data.frame(mean=sapply(cereal.df[,4:16],mean, na.rm=TRUE),
+                    sd=sapply(cereal.df[,4:16], sd, na.rm=TRUE),
+                    min=sapply(cereal.df[,4:16], min, na.rm=TRUE),
+                    max=sapply(cereal.df[,4:16], max, na.rm=TRUE),
+                    median=sapply(cereal.df[,4:16], median, na.rm=TRUE),
+                    length=sapply(cereal.df[,4:16], mean, na.rm=TRUE),
+                    miss.val=sapply(cereal.df[,4:16], function(x) sum(length(which(is.na(x))))))
               mean         sd      min       max    median     length miss.val
calories 106.883117 19.4841191 50.00000 160.00000 110.00000 106.883117        0
protein    2.545455  1.0947897  1.00000   6.00000   3.00000   2.545455        0
fat        1.012987  1.0064726  0.00000   5.00000   1.00000   1.012987        0
sodium   159.675325 83.8322952  0.00000 320.00000 180.00000 159.675325        0
fiber      2.151948  2.3833640  0.00000  14.00000   2.00000   2.151948        0
carbo     14.802632  3.9073256  5.00000  23.00000  14.50000  14.802632        1
sugars     7.026316  4.3786564  0.00000  15.00000   7.00000   7.026316        1
potass    98.666667 70.4106360 15.00000 330.00000  90.00000  98.666667        2
vitamins  28.246753 22.3425225  0.00000 100.00000  25.00000  28.246753        0
shelf      2.207792  0.8325241  1.00000   3.00000   2.00000   2.207792        0
weight     1.029610  0.1504768  0.50000   1.50000   1.00000   1.029610        0
cups       0.821039  0.2327161  0.25000   1.50000   0.75000   0.821039        0
rating    42.665705 14.0472887 18.04285  93.70491  40.40021  42.665705        0
>
```


Figure 3: *Statistical results of calorie variable in R studio.*



```
> #compute mean, standard dev., min, max, median, length, and missing values of calories
> mean(cereal.df$calories)
[1] 106.8831
> sd(cereal.df$calories)
[1] 19.48412
> min(cereal.df$calories)
[1] 50
> max(cereal.df$calories)
[1] 160
> median(cereal.df$calories)
[1] 110
> length(cereal.df$calories)
[1] 77
>
> #find the number of missing values of variable calories
> sum(is.na(cereal.df$calories))
[1] 0
>
```

Figure 4: *Histogram plot for each quantitative variable side by side with hist() function into R Studio.*

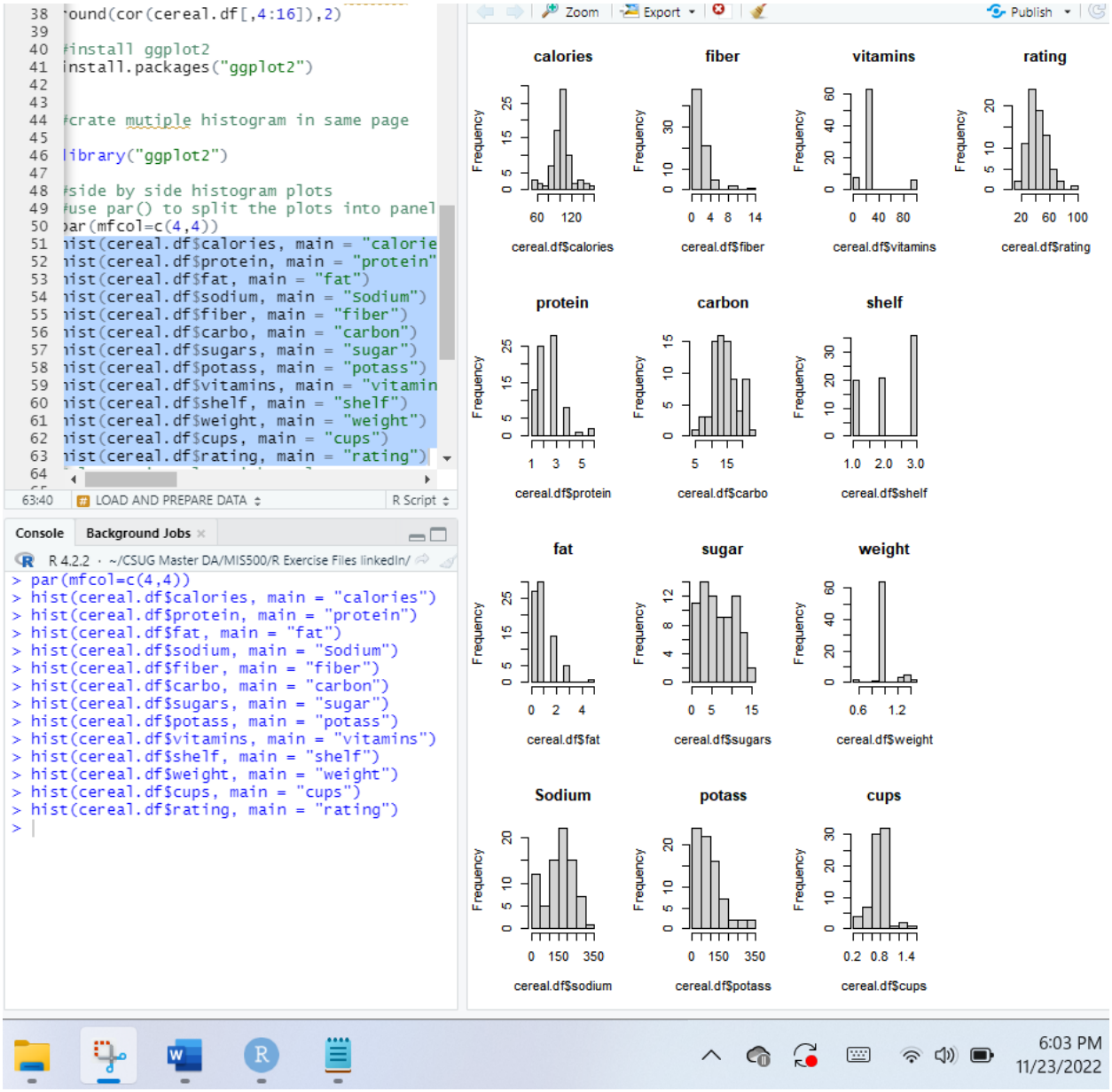


Figure 5: *Alternative way to create histogram plot with ggplot() function into R studio.*

Figure 6: *Boxplot for calorie of cereal type with boxplot() function result in R studio.*

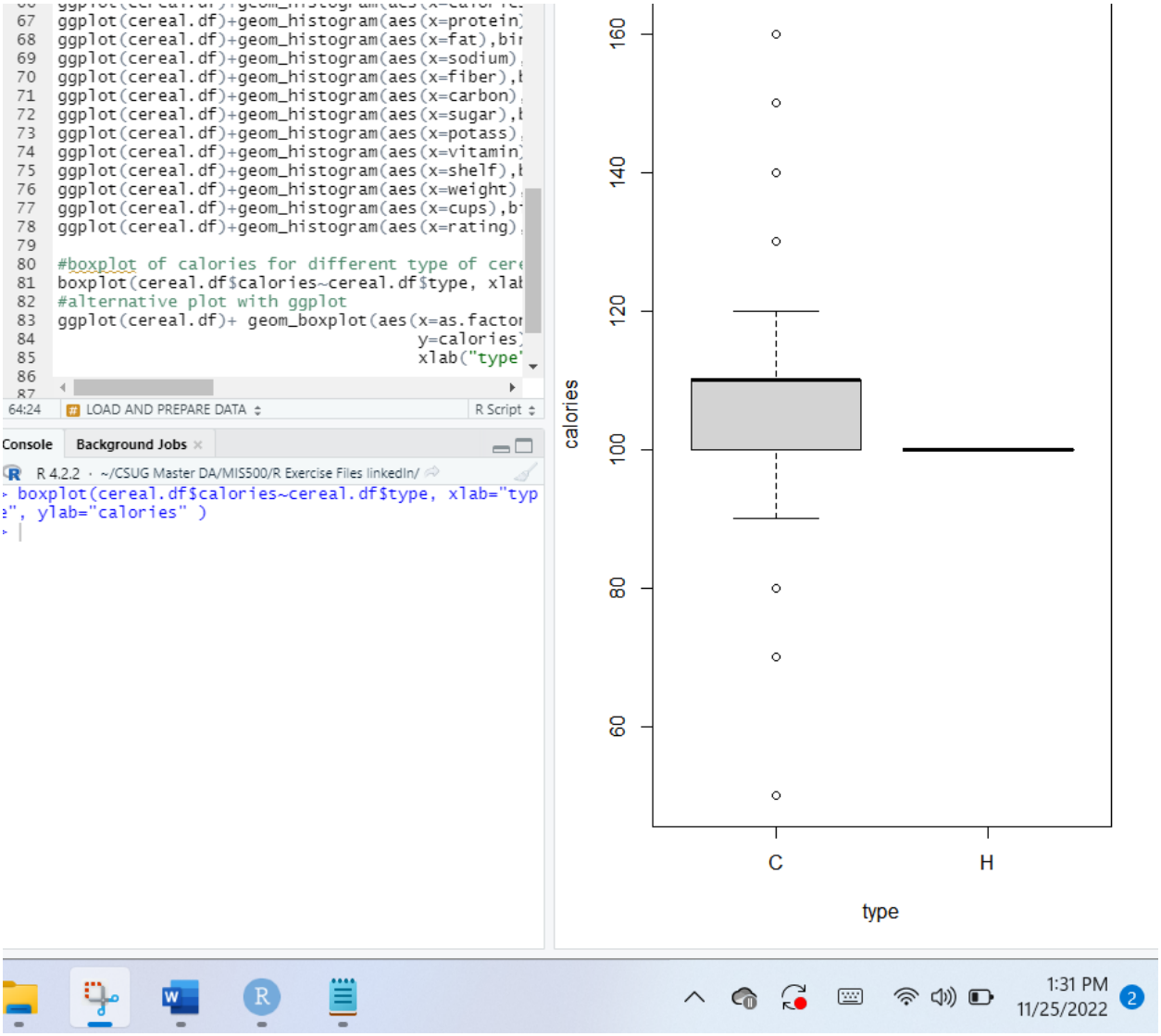
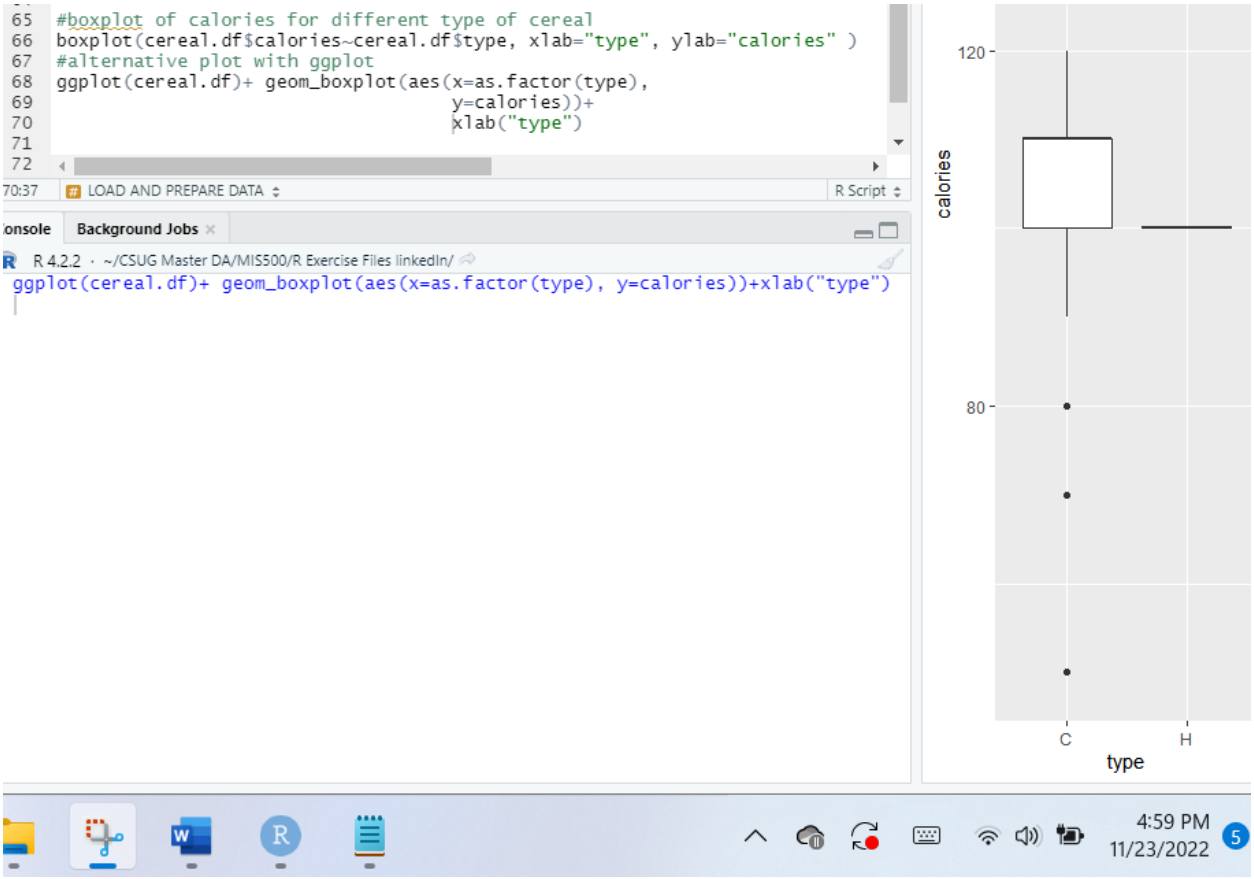


Figure 7: *Alternative way to create boxplot with ggplot() function result in R studio.*



Reference

Nutrition data on 80 cereal products, data content by Chris Crawford, 2017
https://www.heart.org/en/healthy-living/healthy-eating/eat-smart/sodium/how-much-sodium-should-i-eat-per-day
Written by American Heart Association editorial staff and reviewed by science and medicine advisers. November 1, 2021. https://www.heart.org/en/healthy-living/healthy-eating/eat-smart/sodium/how-much-sodium-should-i-eat-per-day

Data Mining for Business Analytics… concepts, Techniques, and Applications in R by Galit Shmueli: Peter C. Bruce: Inbal Yahav: Nitin R. Patel: Kenneth C. Lichtendahl, Jr. Page 56, chapter 3, Figures 3.2 and 3.3. Page 94, chapter 4, Figures 4.3 and 4.4.