STA 360/602 Homework 1: Data Wrangling in R

Chavez Cheong

1/5/2022

1. Working with data

a. Load the data set into R and make it a data frame called `rain.df`. What command did you use?

```
rain.df <- read.table('data/rnf6080.dat',header = FALSE)
```

I used the `read.table()` command.

b. How many rows and columns does `rain.df` have? How do you know? (If there are not 5070 rows and 27 columns, you did something wrong in the first part of the problem.)

```
dim(rain.df)
```

```
## [1] 5070   27
```

`rain.df` has 5070 rows and 27 columns. I know this because I can see the dimensions of the data frame with the `dim()` command.

c. What command would you use to get the names of the columns of `rain.df`? What are those names?

```
names(rain.df)
```

```
##  [1] "V1"  "V2"  "V3"  "V4"  "V5"  "V6"  "V7"  "V8"  "V9"  "V10" "V11" "V12"
## [13] "V13" "V14" "V15" "V16" "V17" "V18" "V19" "V20" "V21" "V22" "V23" "V24"
## [25] "V25" "V26" "V27"
```

I would use the `names` command to get the names of the coloumns of `rain.df`. These names are V1 through V27 (V followed by the corresponding column number).

d. What command would you use to get the value at row 2, column 4? What is the value?

```
rain.df[2,4]
```

```
## [1] 0
```

I would use the `[]` command, specifying the row number as the first element and the column number as the second element. The value is 0.

e. What command would you use to display the whole second row? What is the content of that row?

```
rain.df[2,]
```

```
##   V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21
## 2 60  4  2  0  0  0  0  0  0   0   0   0   0   0   0   0   0   0   0   0   0
##   V22 V23 V24 V25 V26 V27
## 2   0   0   0   0   0   0
```

I would use the `[]` command again, this time specifying 2, the first element as the row number, but leaving the column number blank as the second element, so that I display the whole second row. Aside from the first three columns, V1(60), V2(4) and V3(2), the rest of the columns all contain the value 0.

f. What does the following command do?

```
names(rain.df) <- c("year","month","day",seq(0,23))
```

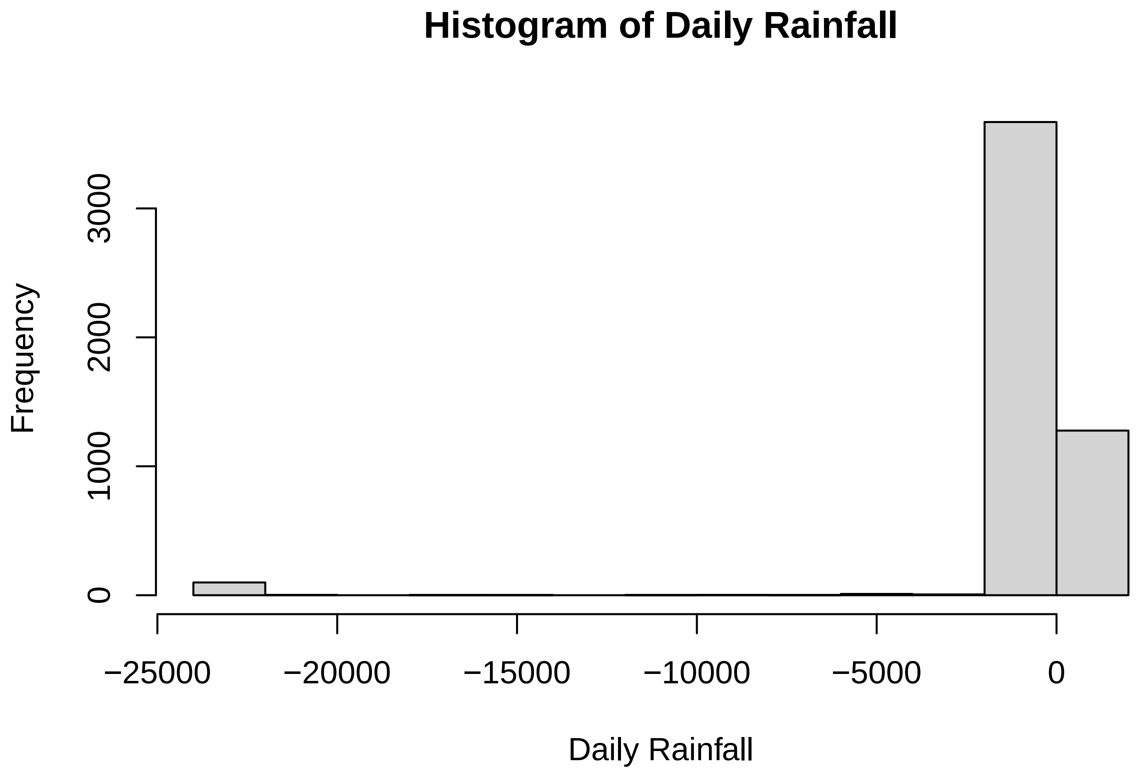
This command renames the first three columns as "year", "month" and "day", and then renames columns "V4" to "V27" as "0" through to "23".

g. Create a new column called `daily`, which is the sum of the 24 hourly columns.

```
rain.df$daily <- rowSums(rain.df[(4:27)])
```

h. Give the command you would use to create a histogram of the daily rainfall amounts. Please make sure to attach your figures in your .pdf report.

```
hist(rain.df$daily, main = 'Histogram of Daily Rainfall', xlab = "Daily Rainfall")
```



i. Explain why that histogram above cannot possibly be right.

It is not possible that there are days with negative rainfall.

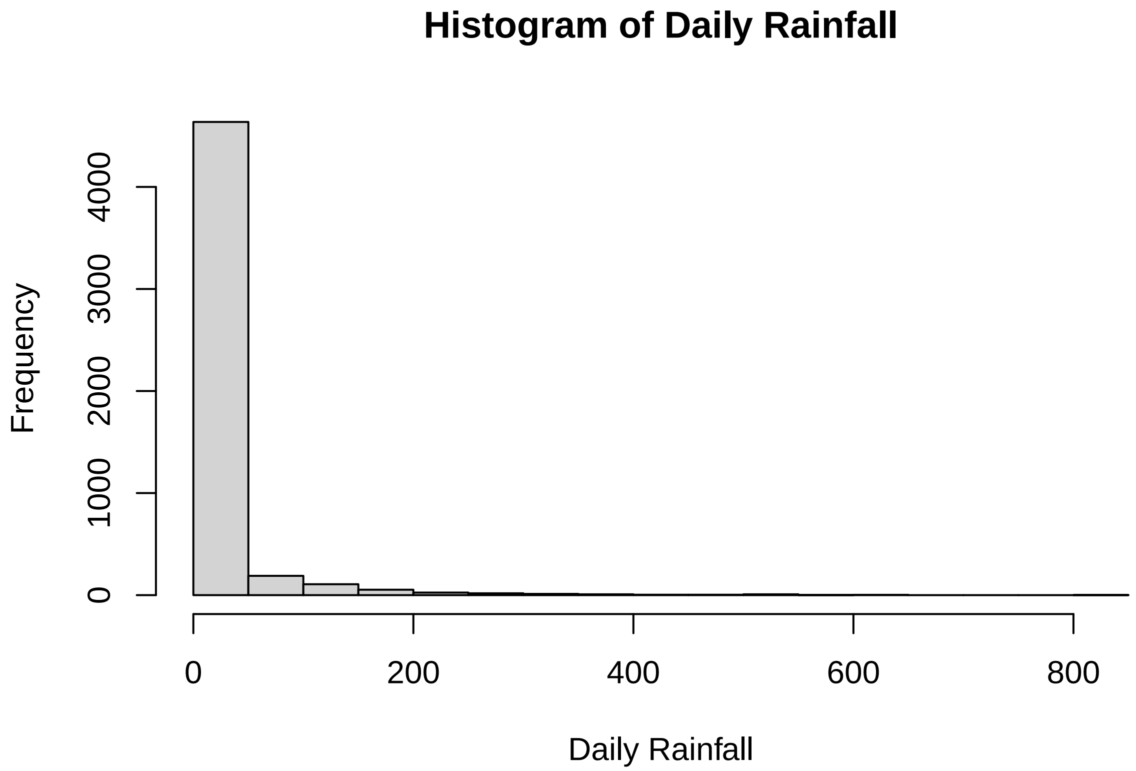
j. Give the command you would use to fix the data frame.

```
# Replace all negative values with 0
rain.df[rain.df < 0] <- 0
```

I would use the `[rain.df < 0]` command to replace all negative values in the dataframe with 0.

k. Create a corrected histogram and again include it as part of your submitted report. Explain why it is more reasonable than the previous histogram.

```
hist(rain.df$daily, main = 'Histogram of Daily Rainfall', xlab = "Daily Rainfall")
```



Now there are no days with negative total daily rainfall, and all values are positive values, which is more reasonable than the previous histogram which included negative values.

2 Data Types

a. For each of the following commands, either explain why they should be errors, or explain the non-erroneous result.

```
x <- c("5","12","7")
max(x)
```

```
## [1] "7"
```

```
sort(x)
```

```
## [1] "12" "5"  "7"
```

```
# sum(x)
```

max(x) would return “7”, as between all the three first characters of each element (“1” for “12”, “5” for “5” and “7” for “7”), “7” is the largest lexicographical value.

sort(x) would return the vector (“12”, “5”, “7”), as it sorts the elements by ascending lexicographic order of the first character of each element (“1” for “12”, “5” for “5” and “7” for “7”).

sum(x) would throw an error, as all elements in the vector x are characters, and the sum function only takes numeric arguments.

b. For the next two commands, either explain their results, or why they should produce errors.

```
y <- c("5",7,12)
# y[2] + y[3]
```

The command `y <- c("5",7,12)` assigns the vector ("5", "7", "12") to the variable `y`. This is because all elements in a vector must be of the same type, so the `c()` command automatically converted the integers 7 and 12 into characters.

`y[2] + y[3]` will return an error, as the `+` binary operator only accepts numeric arguments, but y[2] and y3 store "7" and "12", respectively, which are characters and non-numeric.

c. For the next two commands, either explain their results, or why they should produce errors.

```
z <- data.frame(z1="5",z2=7,z3=12)
z[1,2] + z[1,3]
```

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

`z <- data.frame(z1="5",z2=7,z3=12)` will assign a dataframe three columns and one row of data to the variable `z`. The first column, z1 contains the string "5", z2 and z3 contain the integers 7 and 12, respectively. The data types are preserved because it is possible, unlike in vectors, for different columns in data frames to have different data types.

`z[1,2] + z[1,3]` will return 19, as the sum of the elements in the first row, second column and first row, third column is 7+12=19. As explained earlier, the data types are preserved because it is possible, unlike in vectors, for different columns in data frames to have different data types.

3

a.) What is the point of reproducible code?

It is to ensure that other researchers can reproduce the results of your analysis and verify that your results are true. Additionally, reproducible code also allows different researchers to collaborate on the same project. Finally, it is time-saving as functions can be written and can be reused between projects.

b.) Given an example of why making your code reproducible is important for you to know in this class and moving forward.

If in this class I am working with a particular dataset and work out some analysis regarding the data, it is important for the course instructors to be able to reproduce my code and verify that my results are indeed accurate and that I am not falsifying my work. This is also relevant moving forward in the future as other independent researchers need to verify that my analysis is accurate.

c.) On a scale of 1 (easy) – 10 (hard), how hard was this assignment. If this assignment was hard (> 5), please state in one sentence what you struggled with.

This assignment is a 2 in terms of difficulty.