Week 1 Part 3

This file consists of exercises for the course

HarvardX PH525.1x Statistics and R

Agnesh Panta

dplyr

Exercises #1

Read in the msleep_ggplot2.csv file with the function read.csv() and use the function class() to determine what type of object is returned.

```
msleep <- read.csv('https://raw.githubusercontent.com/genomicsclass/dagdata/master/inst/extdata/msleep_g

class(msleep)
```

```
## [1] "data.frame"
```

Exercises #2

Now use the filter() function to select only the primates.

How many animals in the table are primates? Hint: the nrow() function gives you the number of rows of a data frame or matrix.

```
f_primates <- dplyr::filter(msleep,order=='Primates')

nrow(f_primates)
```

```
## [1] 12
```

Exercises #3

What is the class of the object you obtain after subsetting the table to only include primates?

```
class(f_primates)
```

```
## [1] "data.frame"
```

Exercises #4

Now use the select() function to extract the sleep (total) for the primates.

What class is this object? Hint: use %>% to pipe the results of the filter() function to select().

```
library(magrittr)

s_primates <- dplyr::filter(msleep,order=='Primates') %>%
  dplyr::select(sleep_total)

class(s_primates)
```

```
## [1] "data.frame"
```

Exercises #5

Now we want to calculate the average amount of sleep for primates (the average of the numbers computed above). One challenge is that the mean() function requires a vector so, if we simply apply it to the output above, we get an error. Look at the help file for unlist() and use it to compute the desired average.

What is the average amount of sleep for primates?

```
s_primates <- unlist(s_primates)

mean(s_primates)
```

```
## [1] 10.5
```

Exercises #6

For the last exercise, we could also use the dplyr summarize() function. We have not introduced this function, but you can read the help file and repeat exercise 5, this time using just filter() and summarize() to get the answer.

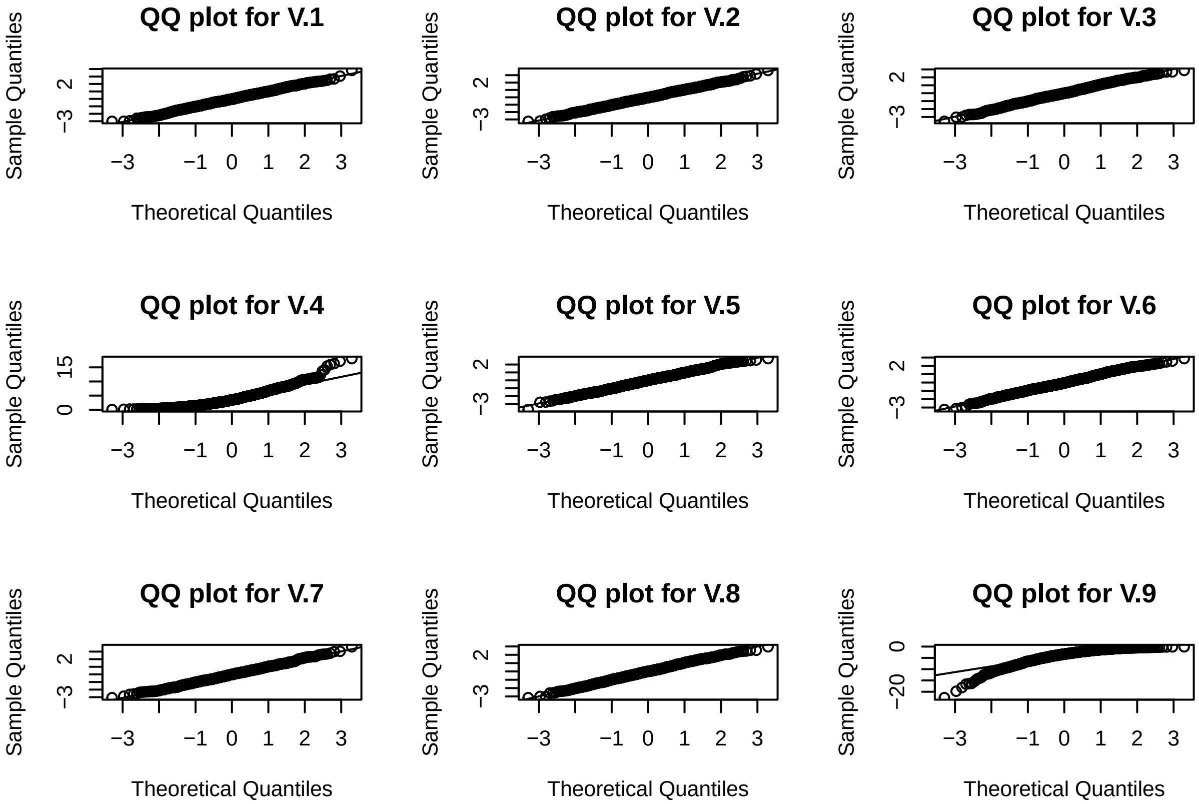
What is the average amount of sleep for primates calculated by summarize()

```
avg_sleep <- dplyr::filter(msleep,order=='Primates') %>%
  dplyr::summarise(avg_sleep=mean(sleep_total))
```

```
dat <- get(load('C:/Users/agnes/HESSENBOX-DA/R_Codes_and_Data/HarvardR/Project_1/skew.RData'))
dim(dat)
```

```
## [1] 1000    9
```

```
par(mfrow = c(3,3))
for (k in 1:9) {
  x <- dat[,k]
  qqnorm(x,  main=paste0('QQ plot for V.',k,sep=''))
  qqline(x)
}
```

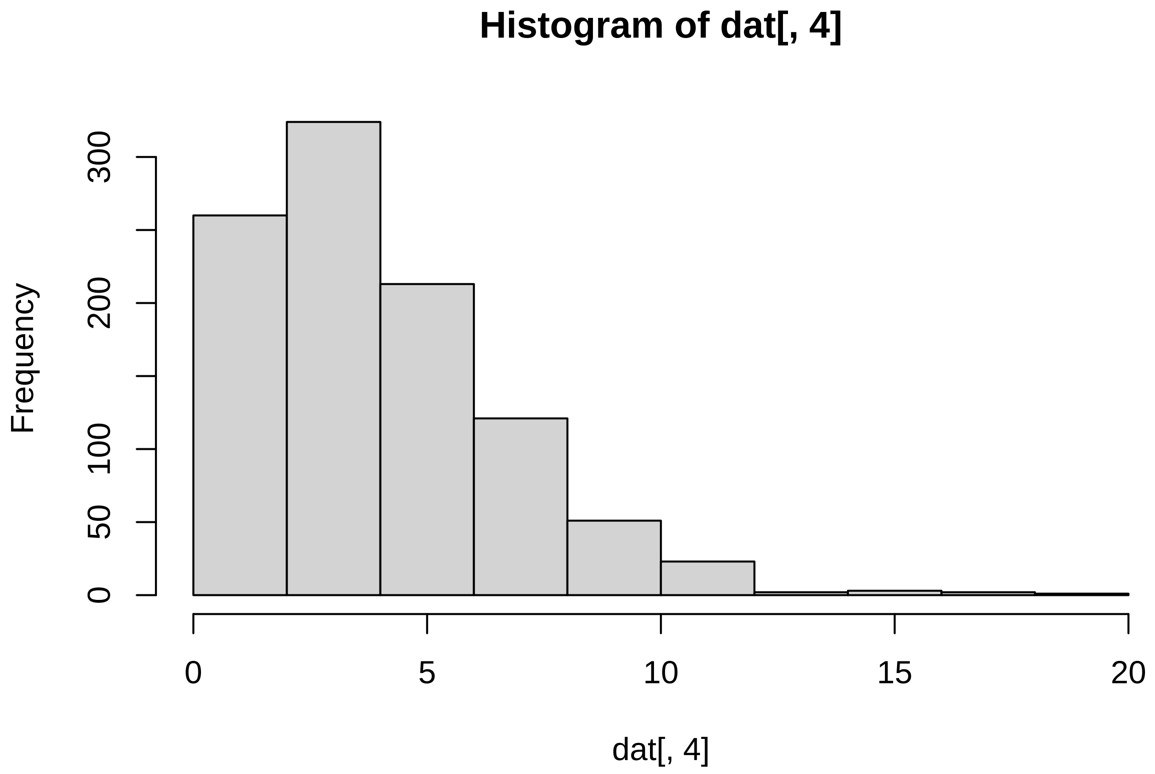


Identify the two columns which are skewed.

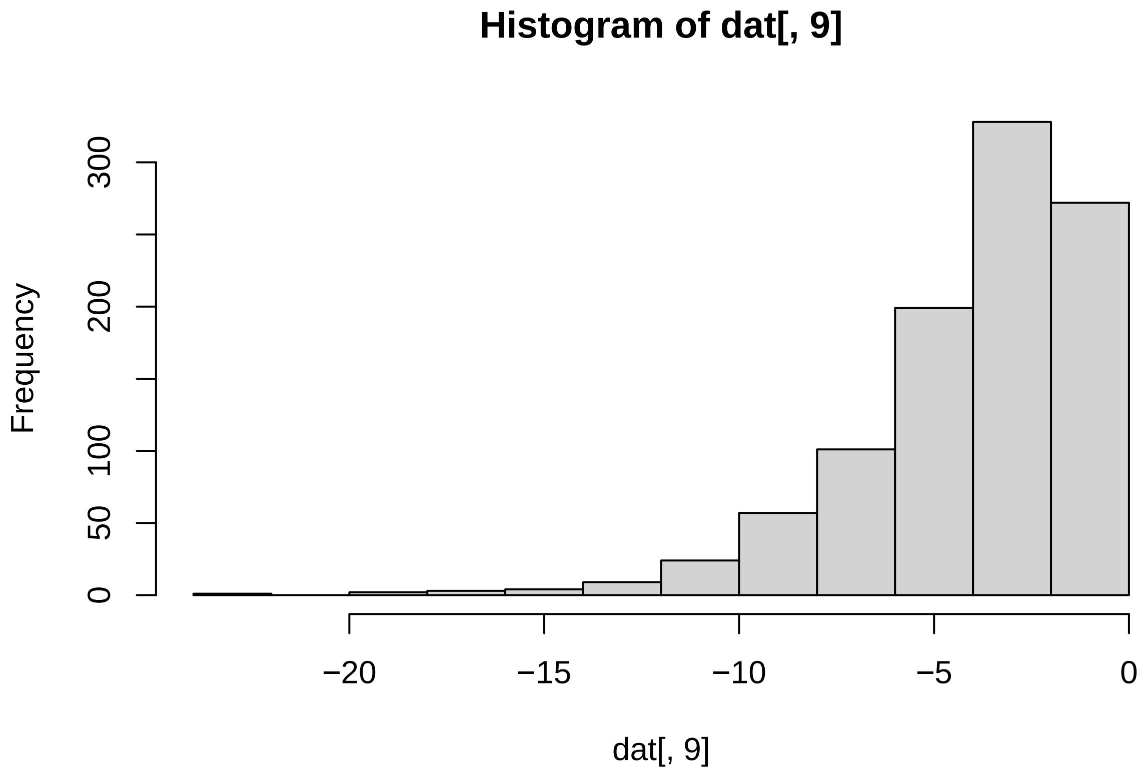
Examine each of these two columns using a histogram. Note which column has "positive skew", in other words the histogram shows a long tail to the right (toward larger values). Note which column has "negative skew", that is, a long tail to the left (toward smaller values). Note that positive skew looks like an up-shaping curve in a qqnorm() plot, while negative skew looks like a down-shaping curve.

You can use the following line to reset your graph to just show one at a time: par(mfrow=c(1,1))

```
hist(dat[,4])
```



```
hist(dat[,9])
```


Histogram of dat[, 9]
Frequency
0
50
100
200
300
−20
−15
−10
−5
0
dat[, 9]