HW5_Do_Quyen

Quyen Do

September 21, 2018

Problem 3

My thoughts on what makes a good figure It should give you some new information or insights about the data at hand

Problem 4

a. A function computing the proportion of successes in a vector

```
## TODO: run microbenchmark on different count_success function

count_success <- function (vect, value = 1) {
  # Compute the proportion of successes in a vector

  # Args:
  # vect: the vector on which the proportion of successes will be computed
  # value: the value presented "success" value in the vector. Default value is 1

  #Return:
  # A real number from 0 to 1

  length(vect[which(vect==value)])/length(vect)
}
```

b. Create a simuluated matrix

```
set.seed(12345)
P4b_data <- matrix(rbinom(10,1,prob =(30:40)/100),nrow = 10, ncol =10)
```

c. Checking the proportion of success

```
# Calculate the proportion of success across matrix row
prop_row <- apply(P4b_data,1,count_success)
prop_col <- apply(P4b_data,2,count_success)

prop_mat <- matrix(c(prop_row,prop_col),nrow=2,ncol=10,byrow = TRUE, dimnames = list(c("By Row","By Col
prop_mat
```

```
##          1   2   3   4   5   6   7   8   9  10
## By Row 1.0 1.0 1.0 1.0 0.0 0.0 0.0 0.0 1.0 1.0
## By Col 0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6
```

The matrix of the simulated binomial using the code in b didn't produce the data as the intentionn. Instead of applying different probabilities to draw success among 10 rows of the matrix, the function seems to apply $p = 1$ to all the rows instead.

d.

```
simulate_binom <- function(probability) {
  # Simulate 10 random binomial variables
```

```
  #  of n = 10 and given probability

  # Args:
  # probability: the probability for the binomial distribution

  #Return:
  # a vector containing 10 RVs drawn from binomial distribution

  print(probability)
  return(rbinom(10, 1, prob = probability))
}

# A vector of probability
prob_vect <- (30:40)/100

# apply simulate_binom on each element of prob_vect
correct_mat <- sapply(prob_vect,simulate_binom)
```

```
## [1] 0.3
## [1] 0.31
## [1] 0.32
## [1] 0.33
## [1] 0.34
## [1] 0.35
## [1] 0.36
## [1] 0.37
## [1] 0.38
## [1] 0.39
## [1] 0.4
```

```
# Calculate the proportion of success
# across rows and columns of correct_mat
prop_row2 <- apply(correct_mat,1,count_success)
prop_col2 <- apply(correct_mat,2,count_success)

prop_mat2 <- matrix(c(prop_row2,prop_col2),nrow=2,ncol=10,byrow = TRUE, dimnames = list(c("By Row","By C
```

```
## Warning in matrix(c(prop_row2, prop_col2), nrow = 2, ncol = 10, byrow =
## TRUE, : data length [21] is not a sub-multiple or multiple of the number of
## rows [2]
```

```
prop_mat2
```

```
##                1         2         3         4         5         6
## By Row 0.7272727 0.2727273 0.5454545 0.4545455 0.3636364 0.1818182
## By Col 0.2000000 0.3000000 0.4000000 0.3000000 0.4000000 0.6000000
##                7         8          9        10
## By Row 0.8181818 0.3636364 0.09090909 0.1818182
## By Col 0.3000000 0.3000000 0.50000000 0.6000000
```

Problem 5

```
#Import raw data from url
url <- "https://www2.isye.gatech.edu/~jeffwu/book/data/starch.dat"
```

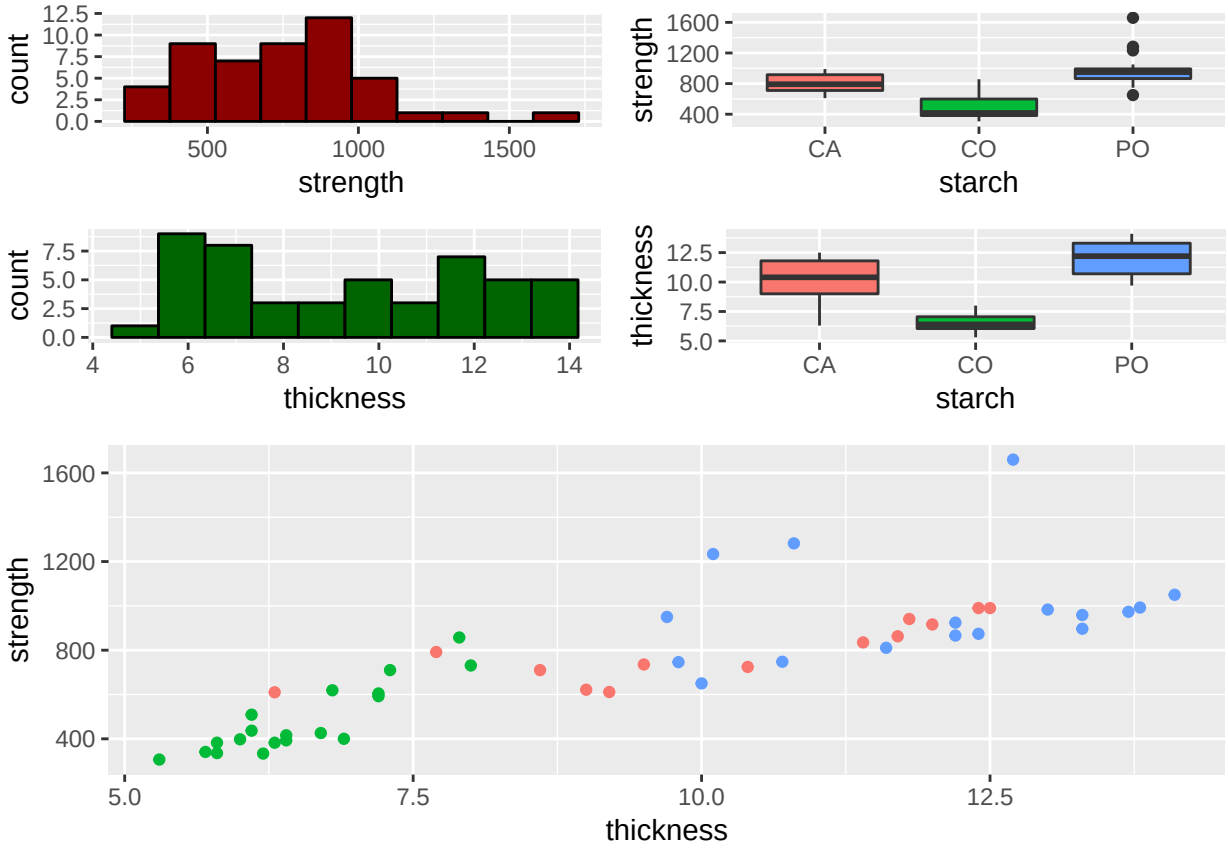
```
starch.dat <- read.csv(url, header=TRUE,sep="")

#Summary
str(starch.dat)
```

```
## 'data.frame':    49 obs. of  3 variables:
##  $ starch   : Factor w/ 3 levels "CA","CO","PO": 1 1 1 1 1 1 1 1 1 1 ...
##  $ strength : num  792 610 710 941 990 ...
##  $ thickness: num  7.7 6.3 8.6 11.8 12.4 12 11.4 10.4 9.2 9 ...
```

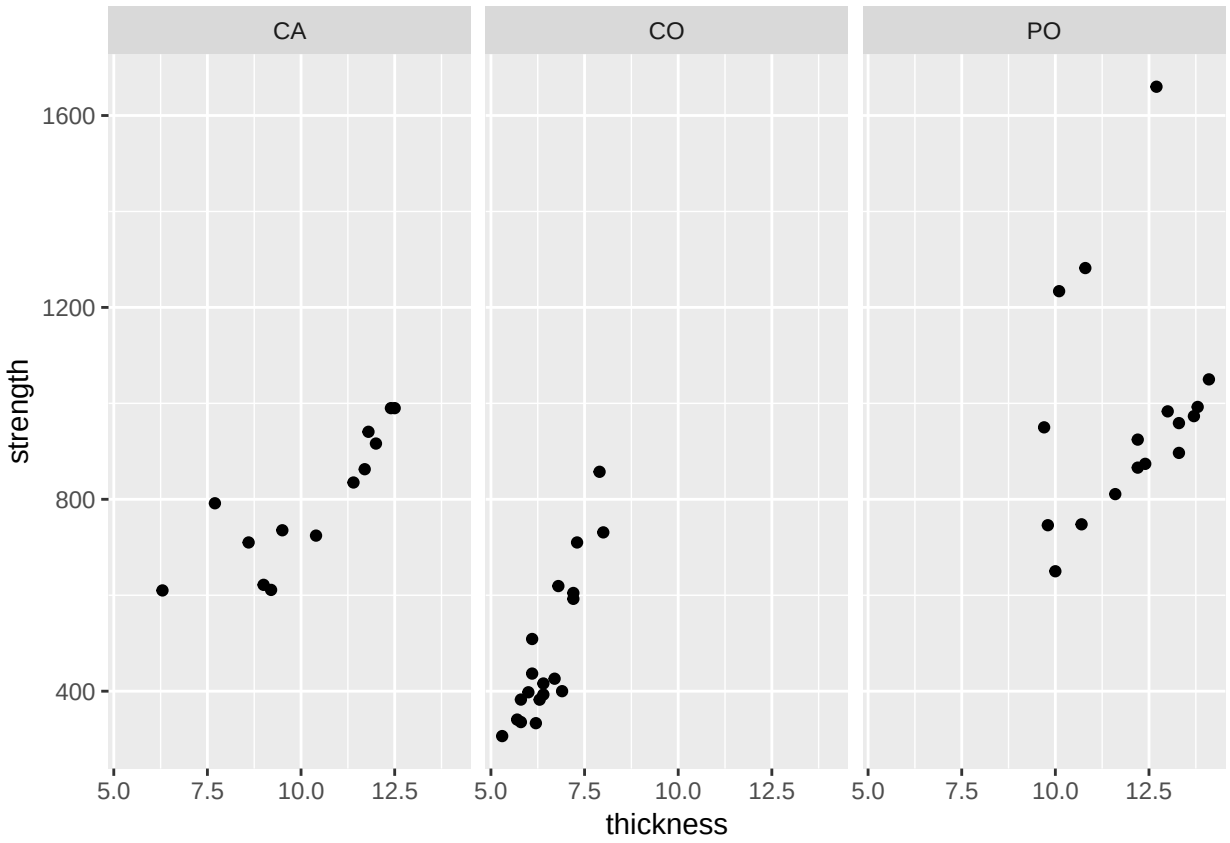
```
starch.dat$starch <- factor(starch.dat$starch)
knitr::kable(summary(starch.dat))
```

starch	strength	thickness
CA:13	Min. : 306.4	Min. : 5.300
CO:19	1st Qu.: 508.8	1st Qu.: 6.700
PO:17	Median : 735.4	Median : 9.500
NA	Mean : 737.0	Mean : 9.388
NA	3rd Qu.: 924.4	3rd Qu.:12.000
NA	Max. :1660.0	Max. :14.100

```
#Multipanel plot
#Multipane plot using ggplot and ggpubr
p1 <- ggplot(starch.dat,aes(x=strength)) + geom_histogram(colour= "black",bins=10,fill="darkred")

p2 <- ggplot(starch.dat,aes(x=starch,y=strength ,group=starch,fill=starch))
p2 <- p2 + geom_boxplot() + guides(fill=FALSE) + labs(x="starch")

p3 <- ggplot(starch.dat,aes(x=thickness)) + geom_histogram(colour= "black",bins=10,fill="darkgreen")

p4 <- ggplot(starch.dat,aes(x=starch,y=thickness ,group=starch,fill=starch))

p4 <- p4 + geom_boxplot() + guides(fill=FALSE) + labs(x="starch")

p5 <- ggplot(starch.dat,aes(thickness,strength,colour=starch)) + geom_point() + labs(x="thickness",y="s

ggarrange(ggarrange(p1,p2,p3,p4,ncol = 2,nrow=2), p5, nrow = 2)
```




```
ggplot(starch.dat,aes(thickness,strength)) + geom_point() + facet_wrap(~starch)
```



Problem 6

```
#we are grabbing a SQL set from here
# http://www.farinspace.com/wp-content/uploads/us_cities_and_states.zip

#download the files, looks like it is a .zip
library(downloader)
download("http://www.farinspace.com/wp-content/uploads/us_cities_and_states.zip",dest="us_cities_sta
unzip("us_cities_states.zip", exdir=".")

#read in data, looks like sql dump, blah
library(data.table)
states <- fread(input = "./us_cities_and_states/states.sql",skip = 23,sep = "'", sep2 = ",", header
### YOU do the CITIES
### I suggest the cities_extended.sql may have everything you need
### can you figure out how to limit this to the 50?
cities <- fread(input = "./us_cities_and_states/cities_extended.sql", skip =23, sep = "'",sep2 = ",
names(cities) <- c("city", "state_code", "zip", "latitude", "longitude", "county")



#The number of cities included by states
knitr::kable(t(table(cities$state_code)),caption = "Number of cities by state")
```