5104 Homework 5

Ryan Christianson

September 26, 2018

```
# global options
knitr::opts_chunk$set(comment = NA)

# libraries
library("ggplot2")
library("GGally")
library("data.table")
library("downloader")
library("fiftystater")
```

Problem 1

Done.

Problem 2

Done.

Problem 3

They should be able to understand the true relationships in the data. For example uncover a dinosaur when plotting by a factor.

Problem 4

```
## part a - the mean function does this...
GetProp <- function(vec) {
  return(mean(vec))
}

## part b
set.seed(12345)
p4b.data <- matrix(rbinom(10, 1, prob = (30:40)/100), nrow = 10,
                   ncol = 10)

## part c
apply(p4b.data, 1, mean)
```

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

```
apply(p4b.data, 2, mean)
```

```
 [1] 0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6 0.6
```

```
## part d
set.seed(12345)
p4d.data <- matrix(rbinom(100, 1, prob = (30:40)/100), nrow = 10,
                   ncol = 10)

apply(p4d.data, 1, mean)
```

```
 [1] 0.8 0.3 0.5 0.4 0.3 0.1 0.7 0.2 0.2 0.3
```

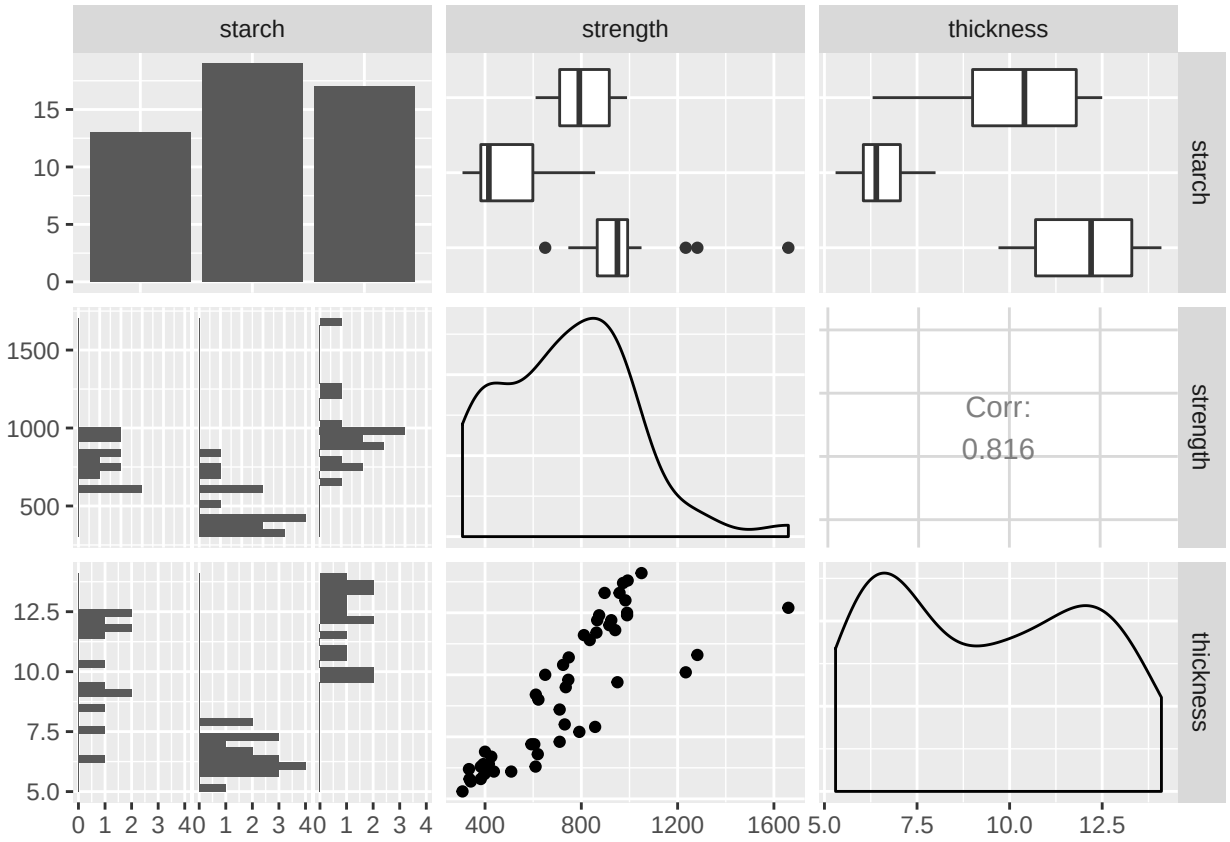
```
apply(p4d.data, 2, mean)
```

```
 [1] 0.6 0.2 0.3 0.3 0.3 0.4 0.6 0.3 0.3 0.5
```

c. It copied the same vector into each column of the data, so the row proportions are 1 or 0 and the column proportions are all the same.

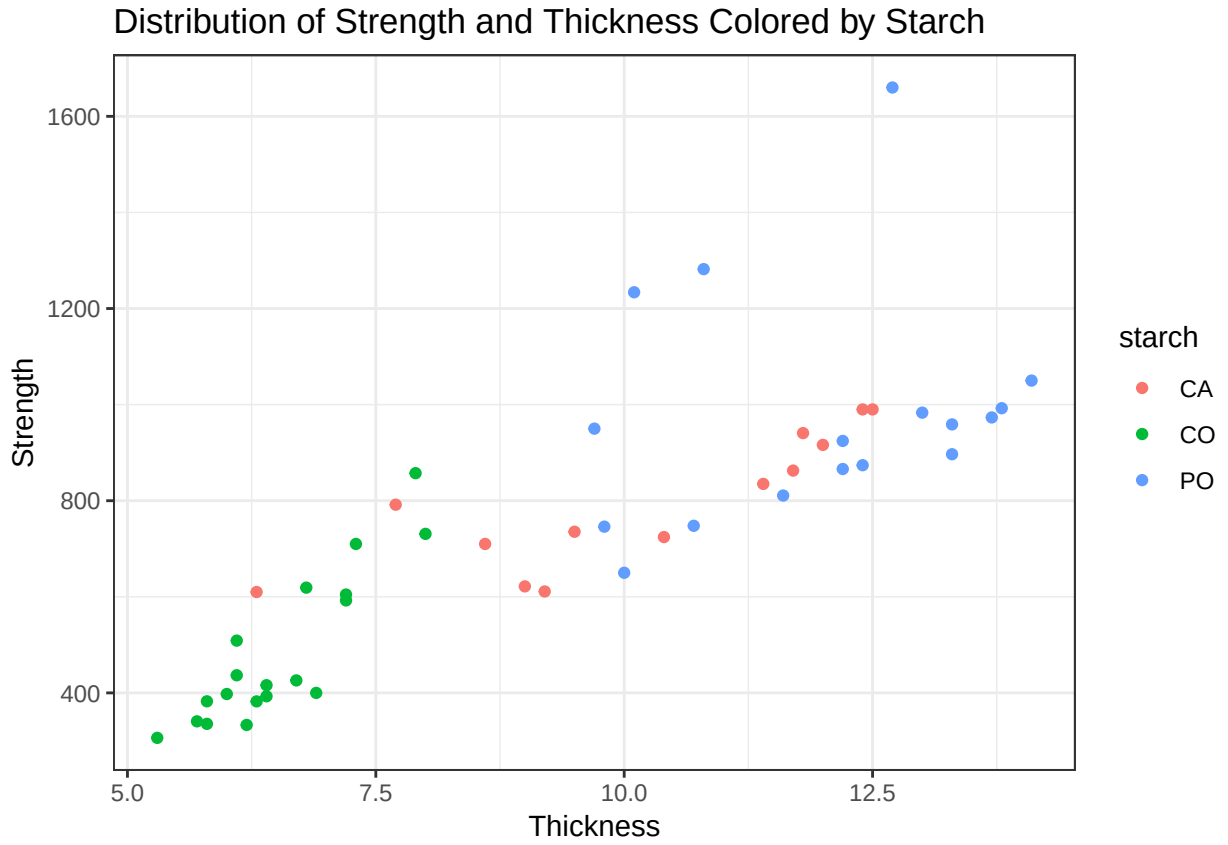
Problem 5

```
url5 <- "https://www2.isye.gatech.edu/~jeffwu/book/data/starch.dat"
p5.data <- read.table(url5, header = TRUE)
ggpairs(p5.data)
```




```
ggplot(data = p5.data, aes(x = thickness, y = strength, color = starch)) +
  geom_point() +
  labs(x = "Thickness", y = "Strength") +
  ggtitle("Distribution of Strength and Thickness Colored by Starch") +
```

```
  theme_bw()
```



I started off with a pairs plot to visualize the variables. Then I wanted to see a plot that included all variables, so I made a scatterplot colored by starch. I feel I can adequately see the structure of the data now.

Problem 6

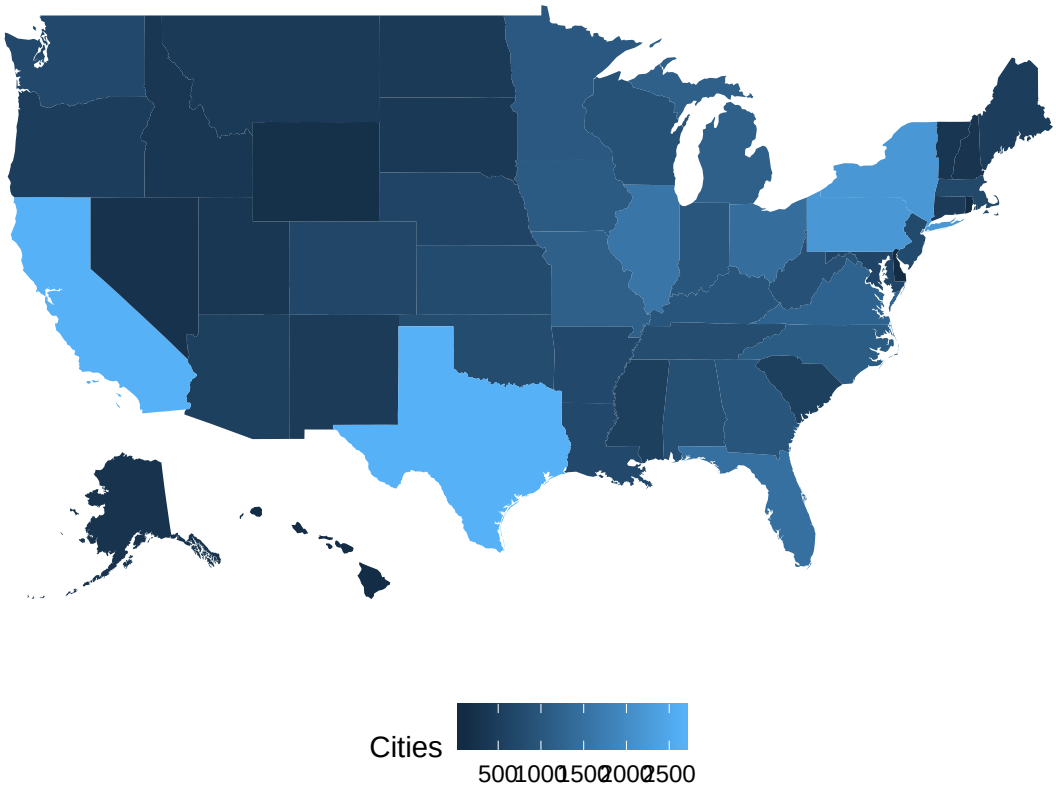
```
# part a

# 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
download("http://www.farinspace.com/wp-content/uploads/us_cities_and_states.zip",
         dest = "us_cities_states.zip")
# removed exdir = "./"
unzip("us_cities_states.zip")
# read in data, looks like sql dump, blah

# removed skip = 23 to read in the states that start with A
states <- fread(input = "./us_cities_and_states/states.sql",
                sep = "'", sep2 = ",",
                header = FALSE, select = c(2, 4))

colnames(states) <- c("State", "Code")
### YOU do the CITIES I suggest the cities_extended.sql
```

```
### may have everything you
cities <- fread(input = "./us_cities_and_states/cities_extended.sql",
                sep = "'", sep2 = ",",
                header = FALSE, select = c(2, 4))
colnames(cities) <- c("City", "State")

# part b
cities.table <- table(cities$State)
cities.table
```

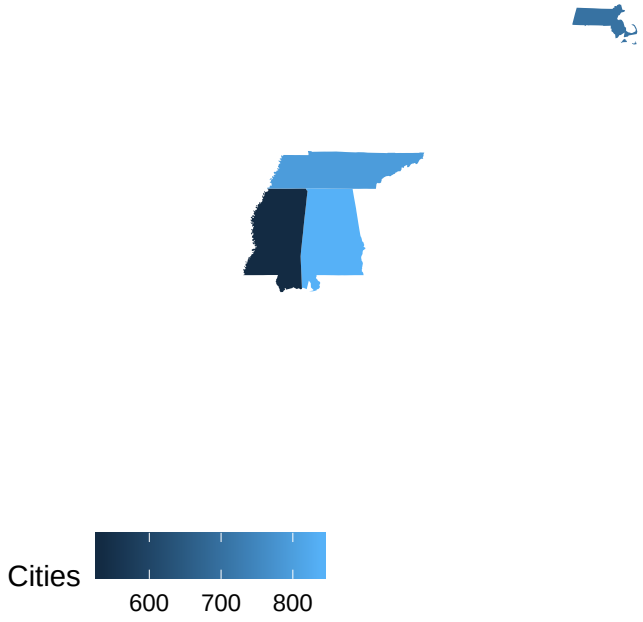
```
  AK   AL   AR   AZ   CA   CO   CT   DC   DE   FL   GA   HI   IA   ID   IL
 273  838  709  532 2651  659  438  284   98 1487  972  139 1060  325 1587
  IN   KS   KY   LA   MA   MD   ME   MI   MN   MO   MS   MT   NC   ND   NE
 989  756  961  725  703  619  489 1170 1031 1170  533  405 1090  407  620
  NH   NJ   NM   NV   NY   OH   OK   OR   PA   PR   RI   SC   SD   TN   TX
 284  733  426  253 2207 1446  774  484 2208  176   91  539  394  795 2650
  UT   VA   VT   WA   WI   WV   WY
 344 1238  309  732  898  859  195
```

```
# don't use PR and DC because they are not in crime dataset
cities.table <- cities.table[!(names(cities.table) %in%  c("PR", "DC"))]
states <- states[!(states$Code %in% c("PR", "DC")), ]
num.cities <- data.frame(Cities = as.numeric(cities.table),
                         Code = names(cities.table))
states <- merge(states, num.cities, by = "Code")

# part c
CountLetter <- function(letter, state.name) {
  state.name <- tolower(state.name)
  return(sum(strsplit(state.name, "")[[1]] == letter))
}

letter.count <- data.frame(matrix(NA, nrow = 50, ncol = 26))
colnames(letter.count) <- letters
for (i in 1:50) {
  letter.count[i, ] <- sapply(letters, CountLetter,
                              state.name = states$State[i])
}

# part d
# https://cran.r-project.org/web/packages/fiftystater/vignettes/fiftystater.html

data("fifty_states") # this line is optional due to lazy data loading
crimes <- data.frame(state = tolower(rownames(USArrests)),
                     USArrests)

states$State <- tolower(states$State)
crimes <- merge(crimes, states, by.x = "state", by.y = "State")
# map_id creates the aesthetic mapping to the state name
# column in your data
ggplot(crimes, aes(map_id = state)) +
  geom_map(aes(fill = Cities), map = fifty_states) +
  expand_limits(x = fifty_states$long, y = fifty_states$lat) +
```

```
  coord_map() +
  scale_x_continuous(breaks = NULL) +
  scale_y_continuous(breaks = NULL) +
  labs(x = "", y = "") +
  theme(legend.position = "bottom", panel.background = element_blank())
```




```
states.3letter <- states[apply(letter.count, 1, max) > 3, ]
ggplot(states.3letter, aes(map_id = State)) +
  geom_map(aes(fill = Cities), map = fifty_states) +
    expand_limits(x = fifty_states$long, y = fifty_states$lat) +
  coord_map() +
  scale_x_continuous(breaks = NULL) +
  scale_y_continuous(breaks = NULL) +
  labs(x = "", y = "") +
  theme(legend.position = "bottom", panel.background = element_blank())
```



Problem 7

Done.