HW5_Wei_Yanran

Yanran Wei

October 2, 2017

Problem 3

In my opinion, there are three important points making a good figure.

1. Content. "A picture is worth a thousand words". A good figure should present information on datasets.
2. Show explanation of figure. For example, legend can help reader understand variables or lines on the graph.
3. Integrate evidence. A figure can show something not obvious of the dataset, like mean, sum, which needed to be calculated based on raw data.

Problem 4

c

Table 1: Proportion of Success

col_prop	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6
row_prop	1	1	1	1	0	0	0	0	1	1

From the table, we can see that the proportion of success is the same for each row which is 0.6. The proportion of success is 1 or 0 for each column. So the proportion does not follow the probability we determined in b.

d

Table 2: Proportion of Success

col_prop2	0.6	0.2	0.3	0.4	0.3	0.4	0.6	0.3	0.3	0.6
row_prop2	0.8	0.3	0.5	0.6	0.3	0	0.7	0.3	0.2	0.3

The proportion of success of each column and row is different.

Problem 5

Data has the highest density for PO.

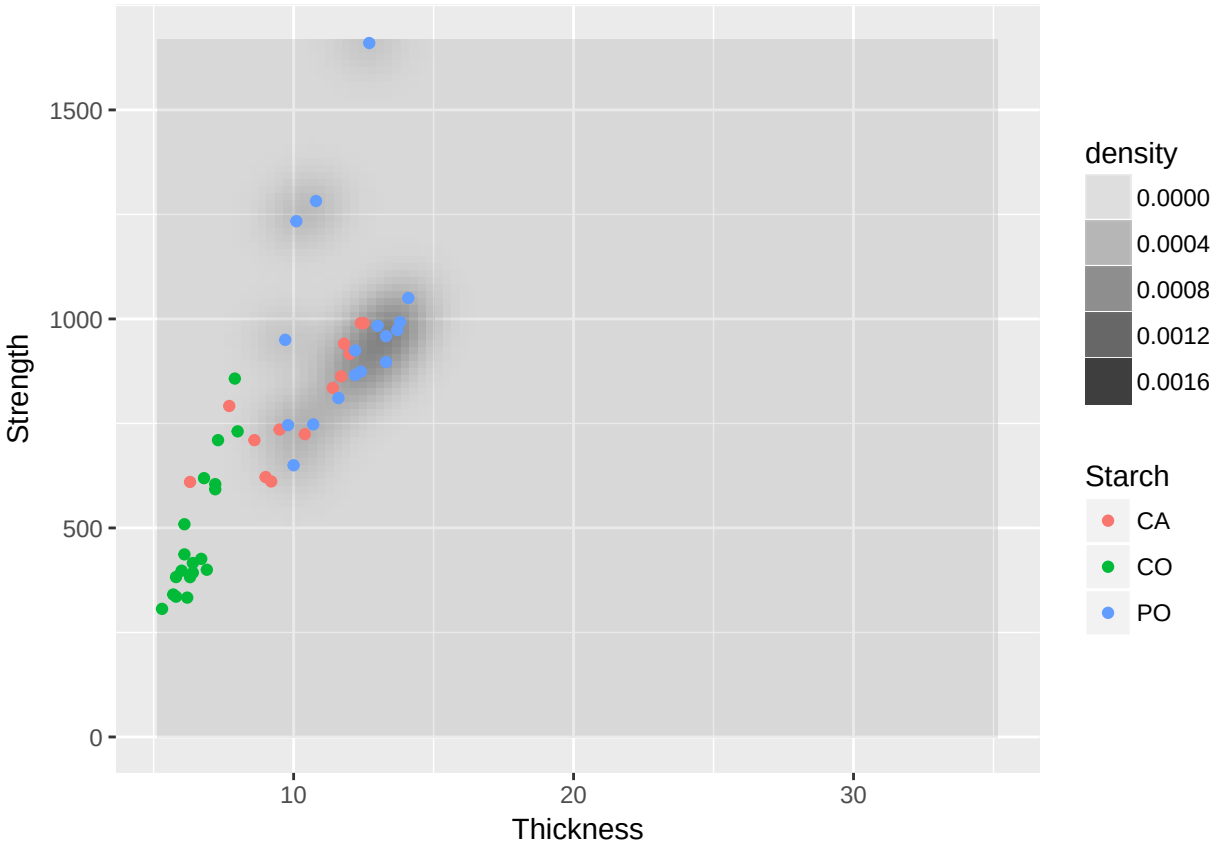


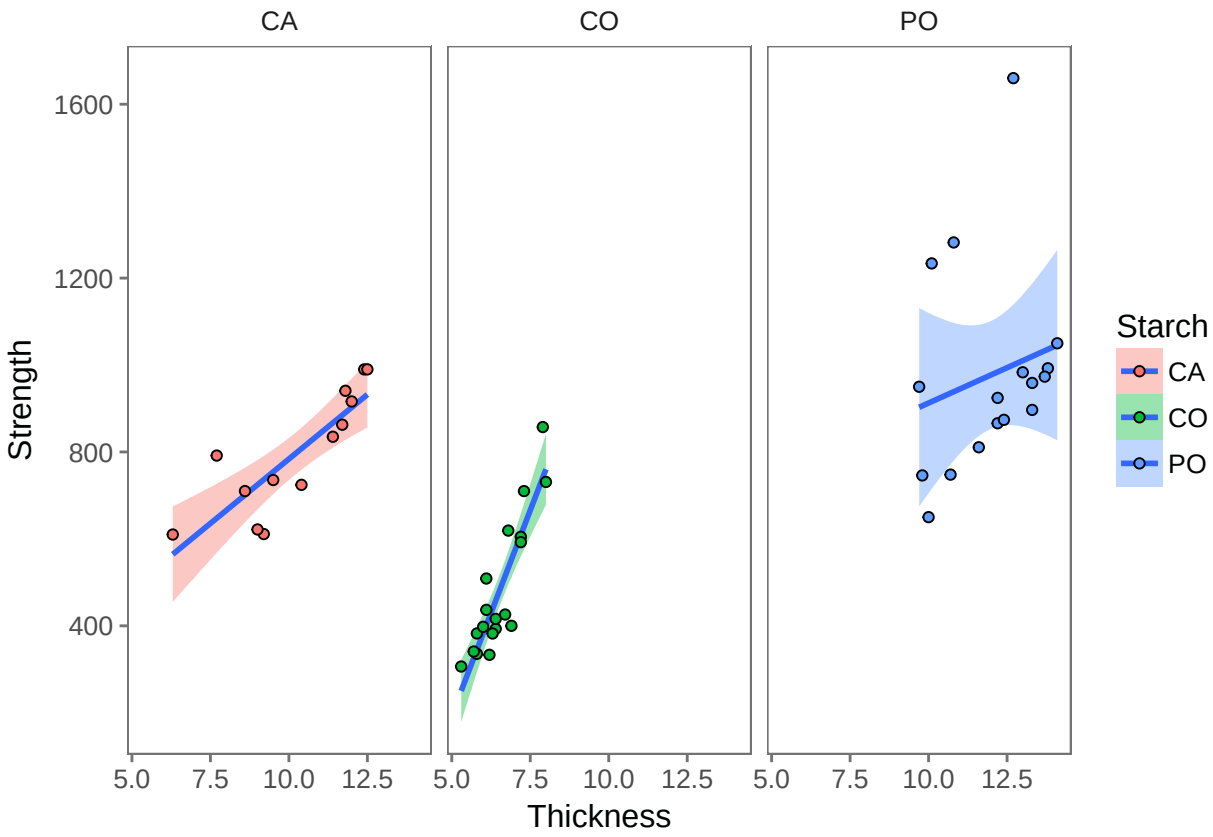
Table 3: Summary of Data

Starch	Strength	Thickness
Length:49	Min. : 306.4	Min. : 5.300
Class :character	1st Qu.: 508.8	1st Qu.: 6.700
Mode :character	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

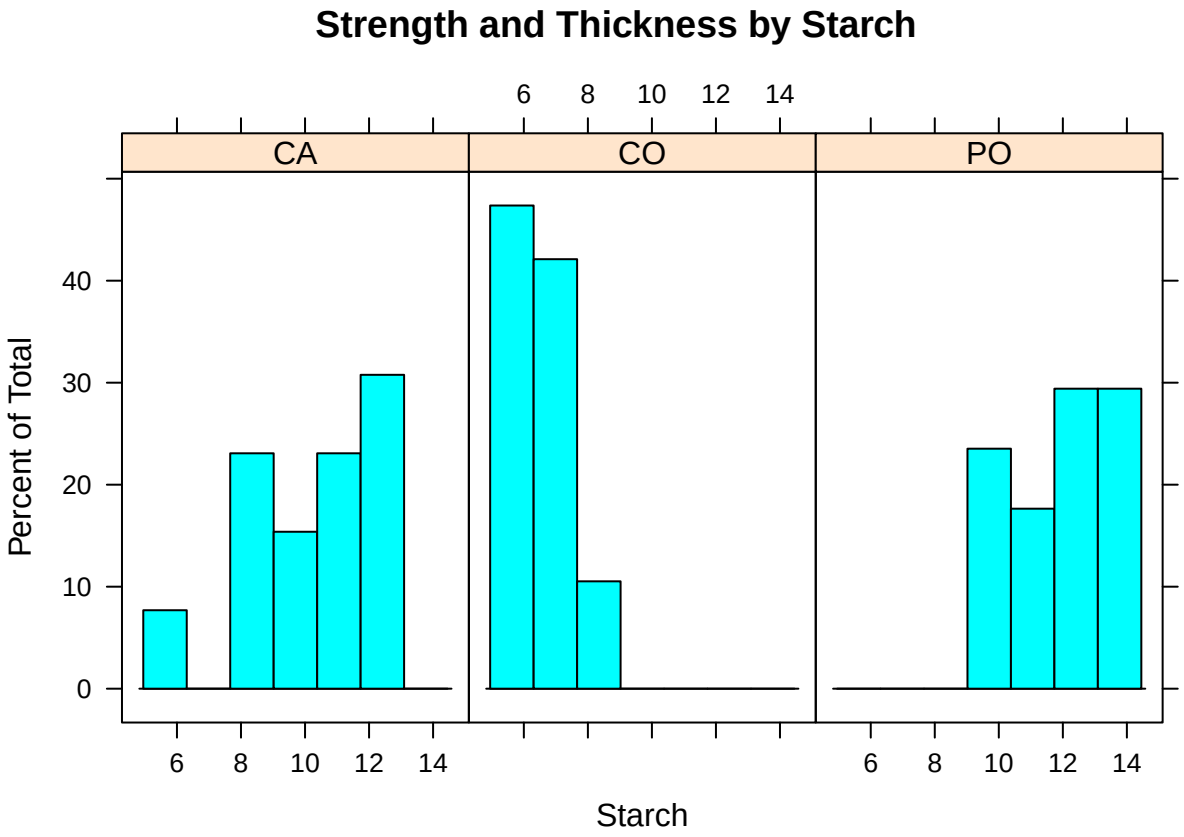
Table 4: Summary of Starch

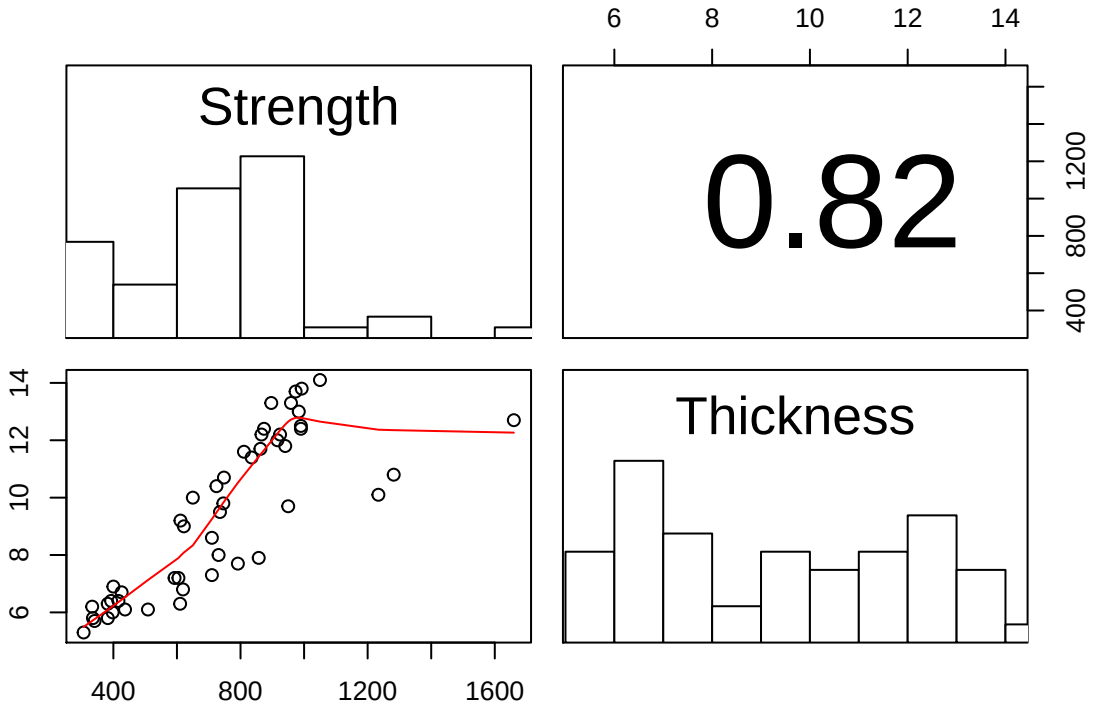
CA	CO	PO
13	19	17

The dataset contains 49 observations and three variables, starch, strength and thickness. For the categorical variable starch, three values are available, CA, CO and PO.



From the graph, there is obvious linear relationship for CA and CO. The linear relationship is not obvious for PO.


Strength and Thickness by Starch
CA
CO
PO
Percent of Total
0
10
20
30
40
6
8
10
12
14
Starch



The table created histograms of variables of Strength and Thickness. And from the lower-left graph, there is a linear relationship between strength and thickness. From the upper-right graph, the correlation between strength and thickness is 0.82.

Problem 6

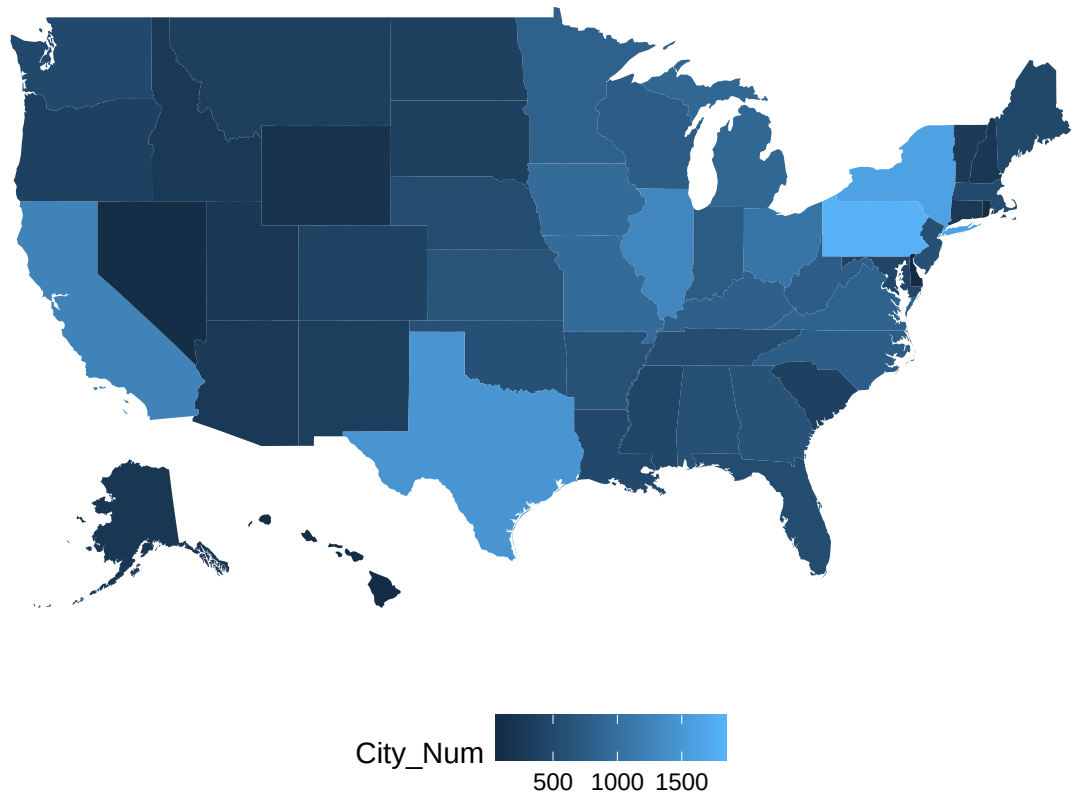
Part b

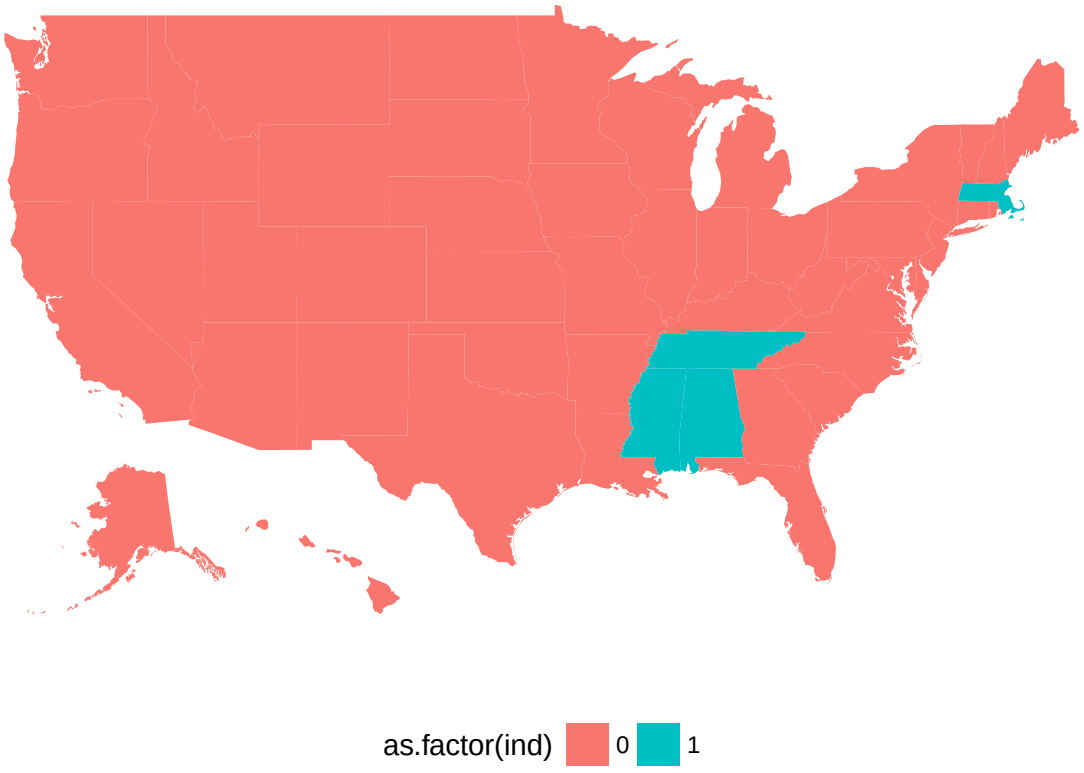
Table 5: Number of Cities included by State

Abbre	City_Num	State
AK	229	alaska
AL	579	alabama
AR	605	arkansas
AZ	264	arizona
CA	1239	california
CO	400	colorado
CT	269	connecticut
DE	57	delaware
FL	524	florida
GA	629	georgia
HI	92	hawaii
IA	937	iowa
ID	266	idaho
IL	1287	illinois

Abbre	City_Num	State
IN	738	indiana
KS	634	kansas
KY	803	kentucky
LA	479	louisiana
MA	511	massachusetts
MD	430	maryland
ME	461	maine
MI	885	michigan
MN	810	minnesota
MO	942	missouri
MS	440	mississippi
MT	360	montana
NC	762	north carolina
ND	373	north dakota
NE	528	nebraska
NH	255	new hampshire
NJ	579	new jersey
NM	346	new mexico
NV	99	nevada
NY	1612	new york
OH	1069	ohio
OK	585	oklahoma
OR	379	oregon
PA	1802	pennsylvania
RI	70	rhode island
SC	377	south carolina
SD	364	south dakota
TN	548	tennessee
TX	1466	texas
UT	250	utah
VA	839	virginia
VT	288	vermont
WA	493	washington
WI	753	wisconsin
WV	753	west virginia
WY	176	wyoming

Part d





Code for Problems 3-6

```
# Problem 4

# Question a
sucess_fun <- function(x){
  sum(x)/10
}

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

# Question c
col_prop <- apply(P4b_data, 2, sucess_fun)
row_prop <- apply(P4b_data, 1, sucess_fun)
pander(rbind(col_prop, row_prop), caption = "Proportion of Success")

# Question d
set.seed(12345)

# Function to create matrix
prob_func <- function(x){
  matrix(rbinom(10, 1, prob = x), nrow = 10, ncol = 1)
}
P4d_data <- mapply(prob_func, x = c(31:40)/100)

# Calcualte marginal success
col_prop2 <- apply(P4d_data, 2, sucess_fun)
row_prop2 <- apply(P4d_data, 1, sucess_fun)
pander(rbind(col_prop2, row_prop2), caption = "Proportion of Success")

**********
# Problem 5

#Load into data
url <- "http://www2.isye.gatech.edu/~jeffwu/book/data/starch.dat"
P5_raw<-read.table(url, header=F, skip=1, fill=T, stringsAsFactors = F)
colnames(P5_raw) <- c("Starch", "Strength", "Thickness")

require(MASS)
ggplot(P5_raw, aes(x = Thickness, y= Strength, colour = Starch)) +
  stat_density2d(aes(alpha = ..density..), geom = 'raster', contour = FALSE) +
  geom_point()+
  expand_limits(x = 35, yend = 6)

pander(summary(P5_raw), caption = "Summary of Data")
pander(table(P5_raw$Starch), caption = "Summary of Starch")

library(ggthemes)
ggplot(P5_raw, aes(x = Thickness, y = Strength, fill = Starch)) +
  geom_smooth(method = lm) +
  geom_point(shape =21) +
  facet_grid(.~Starch) +
```

```
  theme_few() +
  scale_colour_few()

library(lattice)
histogram(Strength ~ Thickness|as.factor(Starch), data = P5_raw,
          main = "Strength and Thickness by Starch",
          xlab = "Starch")

# Correlation and plot function
panel.cor <- function(x, y, digits = 2, prefix = " ", cex.cor, ...){
  usr <- par("usr")
  on.exit(par(usr))
  par(usr = c(0, 1, 0, 1))
  r <- abs(cor(x, y, use = "complete.obs"))
  txt <- format(c(r, 0.123456789), digits = digits)[1]
  txt <- paste(prefix, txt, sep = " ")
  if(missing(cex.cor)) cex.cor <- 0.8/strwidth(txt)
  text(0.5, 0.5, txt, cex = cex.cor * (1 + r)/2)
}

panel.hist <- function(x, ...){
  usr <- par("usr")
  on.exit(par(usr))
  par(usr = c(usr[1:2], 0, 1.5))
  h <- hist(x, plot = FALSE)
  breaks <- h$breaks
  nB <- length(breaks)
  y <- h$counts
  y <- y/max(y)
  rect(breaks[-nB], 0, breaks[-1], y, col = "white", ...)
}

pairs(P5_raw[, 2:3], upper.panel = panel.cor,
      diag.panel = panel.hist,
      lower.panel = panel.smooth)

**********************
# Problem 6
# Part a
  #we are grabbing a SQL set from here
    # http://www.farinspace.com/wp-content/uploads/us_cities_and_states.zip

    #read in data, looks like sql dump, blah
    library(data.table)
    states <- fread(input = "C:/Users/Echo/Desktop/states.sql", sep = "'", sep2 = ",", header = F, sele
    colnames(states) <- c("State", "Abbre")

    ### 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 = "C:/Users/Echo/Desktop/cities_extended.sql",sep = "'", sep2 = ",", header =
    colnames(cities) <- c("City", "Abbre")
```

```r
# Part b
# Create a summary table of the number of cities included by state
cities_count <- sqldf("
    select Abbre, count(city) as City_Num, State
    from cities_states
    group by Abbre"
    )
cities_count[, 3] <- tolower(cities_count[, 3])
pander(cities_count, caption = "Number of Cities included by State")

# Part c
 ##pseudo code
    letter_count <- data.frame(matrix(NA,nrow=50, ncol=27))
    letter_count[, 1] <- tolower(cities_count[, 3])
    colnames(letter_count) <- c("State", "a", "b", "c", "d", "e", "f", "g", "h", "i", "j", "k", "l", "m
    getCount <- function(l, s){
        count <- unlist(strsplit(s,""))
        return(sum(count == l))
    }

    for(i in 1:50){
        letter_count[i, 2:27 ] <- sapply(letters, getCount, s = cities_count[i, 3])
    }

# Part d
#https://cran.r-project.org/web/packages/fiftystater/vignettes/fiftystater.html
    library(ggplot2)
    library(fiftystater)
    library(mapproj)

    data("fifty_states") # this line is optional due to lazy data loading
    crimes <- data.frame(state = tolower(rownames(USArrests)), USArrests)
    # map_id creates the aesthetic mapping to the state name column in your data
    p <- ggplot(cities_count, aes(map_id = State)) +
      # map points to the fifty_states shape data
      geom_map(aes(fill = City_Num), 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())

    p
    #ggsave(plot = p, file = "HW5_Problem6_Plot_Settlage.pdf")

    # Second map
    r0 <- which(apply(letter_count[, 2:27], 1, max) <4)
    r1 <- which(apply(letter_count[, 2:27], 1, max) >3)
    ind <- matrix(nrow = 50, ncol = 1)
    ind[r0, 1] = 0
    ind[r1, 1] = 1
```

```
p2_dat <- cbind(cities_count, ind)
 p2 <- ggplot(p2_dat, aes(map_id = State)) +
  # map points to the fifty_states shape data
  geom_map(aes(fill = as.factor(ind)), 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())

p2
```