STAT 5014 Homework 5

Chaoran Wang

2017-10-03

Problem 3

A good figure should shows clear information to readers in a second which means readers should derive the information we want them know quickly. It should includes clear title, axis labels, necessary values, and possible highlights. The plot should comes with appropriate type of graphs. For example, histogram is better to show the distribution of the data. If you want to focus on the relationship, especially the regression, scatter plot with a fitted line might be better. Box plot is better to show the basic statistics of the data like mean and medium. If you want to focus on the proportion of a set of categorical data with few categories, pie chart should be the best one. If it has many categories, bar chart should be better.

Problem 4

I first create the function to compute the proportion of successes in a vector. Then I use 'apply' function to compute the proportion of success in P4b_data by row and column. we can see the two results are different. The rows coming with all "1" always have 100% of success when we 'apply' by row. There is no column coming wih all "1", so the results are some proportions when we 'apply' by column. Therefore, using 'apply' by column seems make more sense. **However, both row and column seems to be incorrect because the codes fill the matrix just with the repeats of the first row/column it get, so we always get same result for each row/column which is not random.** Codes are in Appendix.

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

```
##  [1] 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5
```

Next, I create a function to simulate 10 flips of a coin when a probability is given. Then, I assign a vector of the desired probabilities to be (10, 20, . . . , 90). Using the simulation function, I get a matrix back. Using the 'Problem4_fun', I can prove the simulation function does work.

Table 1: Simulation when Probabilities are Given

prob	10.0	20.0	30.0	40.0	50.0	60.0	70.0	80.0	90.0
res_proof	0.3	0.1	0.5	0.4	0.5	0.7	0.6	0.8	0.8
Simulation									
	0.0	0.0	1.0	0.0	0.0	0.0	1.0	0.0	1.0
	1.0	1.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0
	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0
	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0
	1.0	0.0	1.0	1.0	1.0	1.0	0.0	1.0	1.0
	0.0	0.0	1.0	0.0	0.0	1.0	0.0	1.0	1.0
	0.0	0.0	0.0	1.0	0.0	1.0	1.0	1.0	1.0
	0.0	0.0	1.0	0.0	1.0	1.0	0.0	1.0	0.0
	1.0	0.0	0.0	0.0	1.0	0.0	1.0	1.0	0.0
	0.0	0.0	1.0	1.0	0.0	1.0	0.0	0.0	1.0

Problem 5

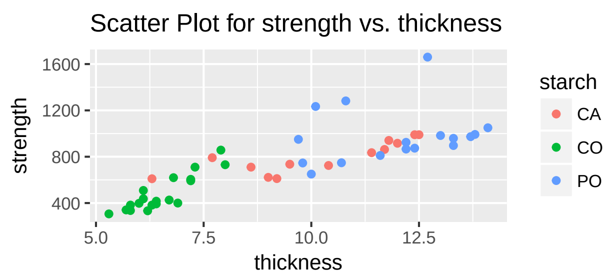
All codes of this problem are in Appendix.

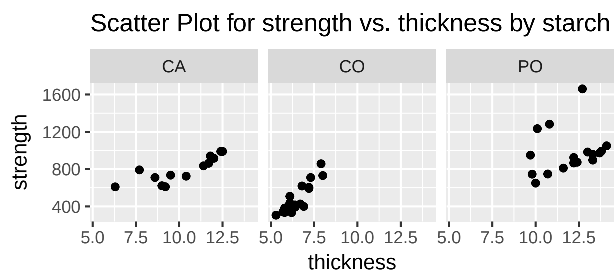
I first load and tidy the data, then print a basic summary table out by starch groups.

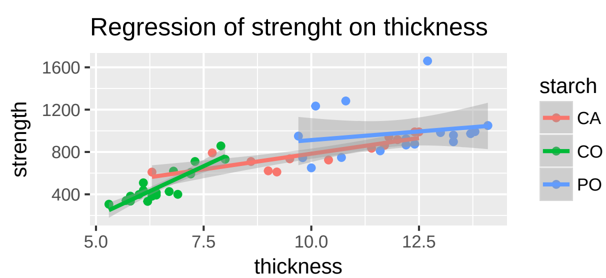
Table 2: Summary by starch Group

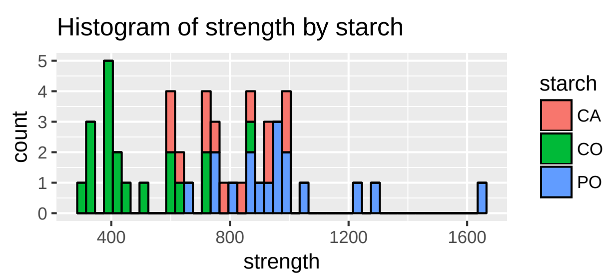
	group1	n	mean	sd	median	min	max	range	skew
strength1	CA	13	795.292308	139.0484775	791.7	610.0	990.0	380.0	0.0268670
strength2	CO	19	482.826316	157.5957693	416.0	306.4	857.3	550.9	0.8846916
strength3	PO	17	976.429412	237.7956207	950.0	650.0	1660.0	1010.0	1.3070692
thickness1	CA	13	10.192308	1.9708127	10.4	6.3	12.5	6.2	-0.4339276
thickness2	CO	19	6.531579	0.7409098	6.4	5.3	8.0	2.7	0.4222869
thickness3	PO	17	11.964706	1.5136633	12.2	9.7	14.1	4.4	-0.2028206

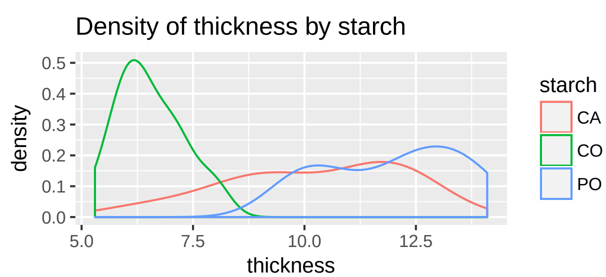
Then, I explore the data and print necessary figures out. All figures are based on starch groups.

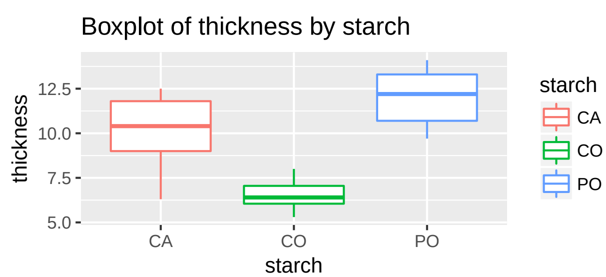


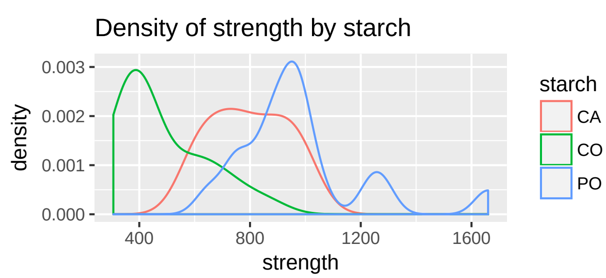


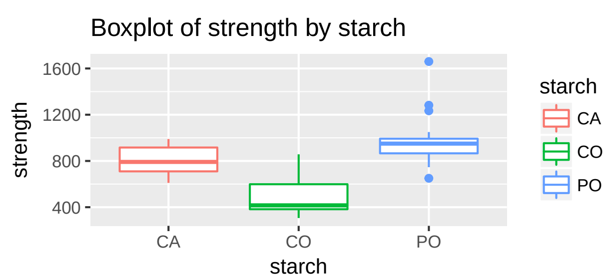




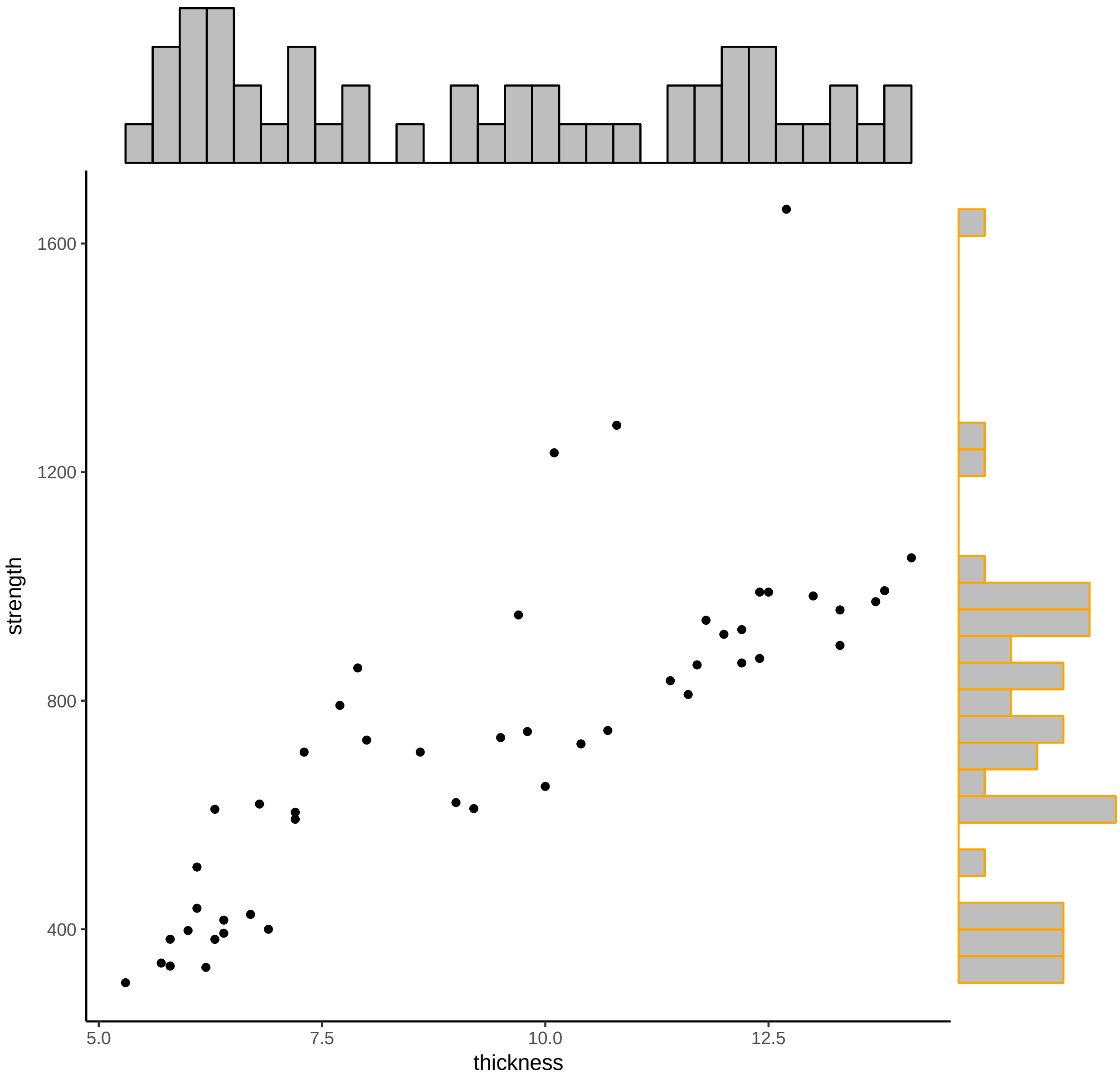




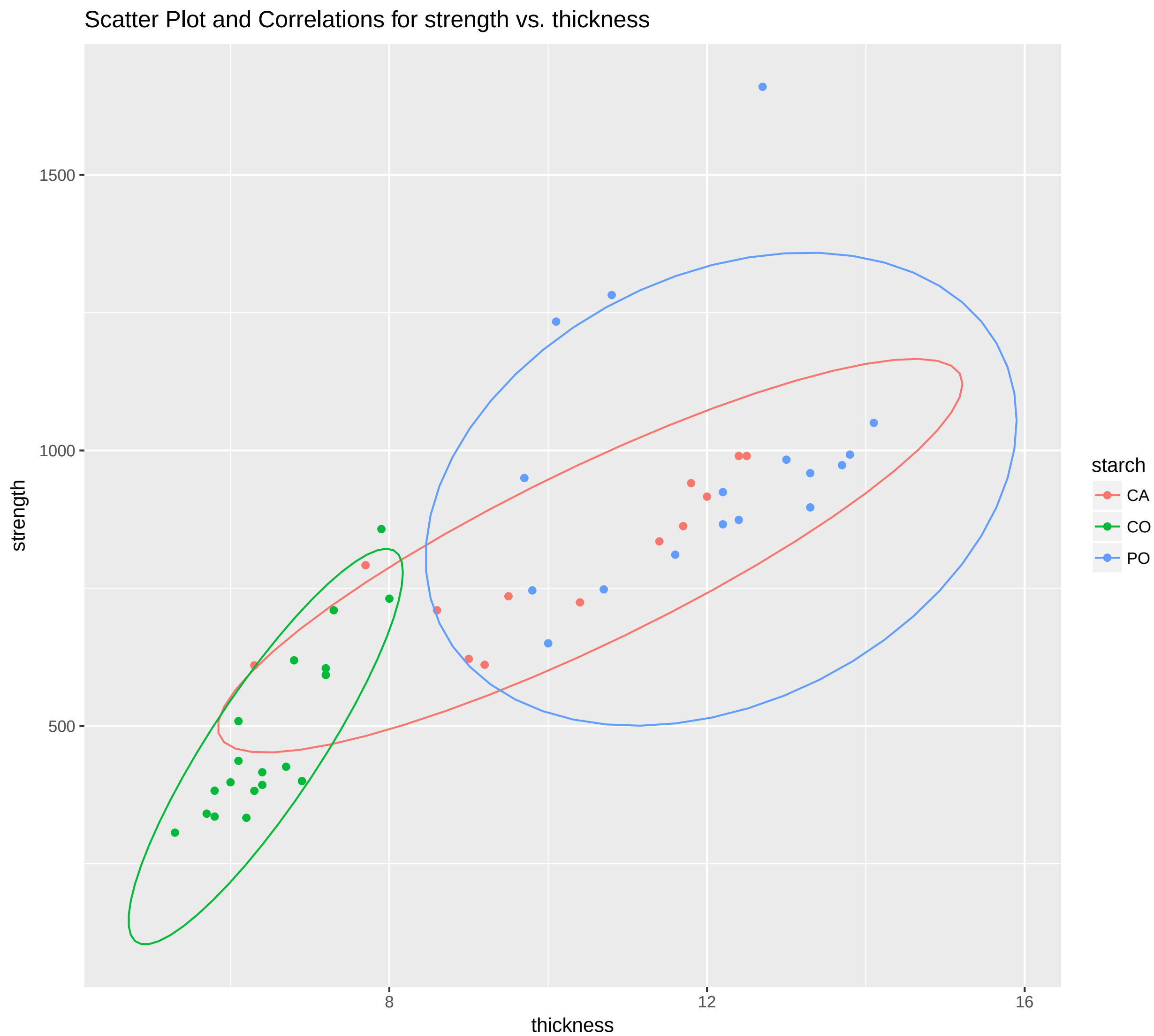




Density of a Scatter Marginal Plot



Finally, I print the plot to show the correlations between variables.



Problem 6

All codes used in this problem are in Appendix.

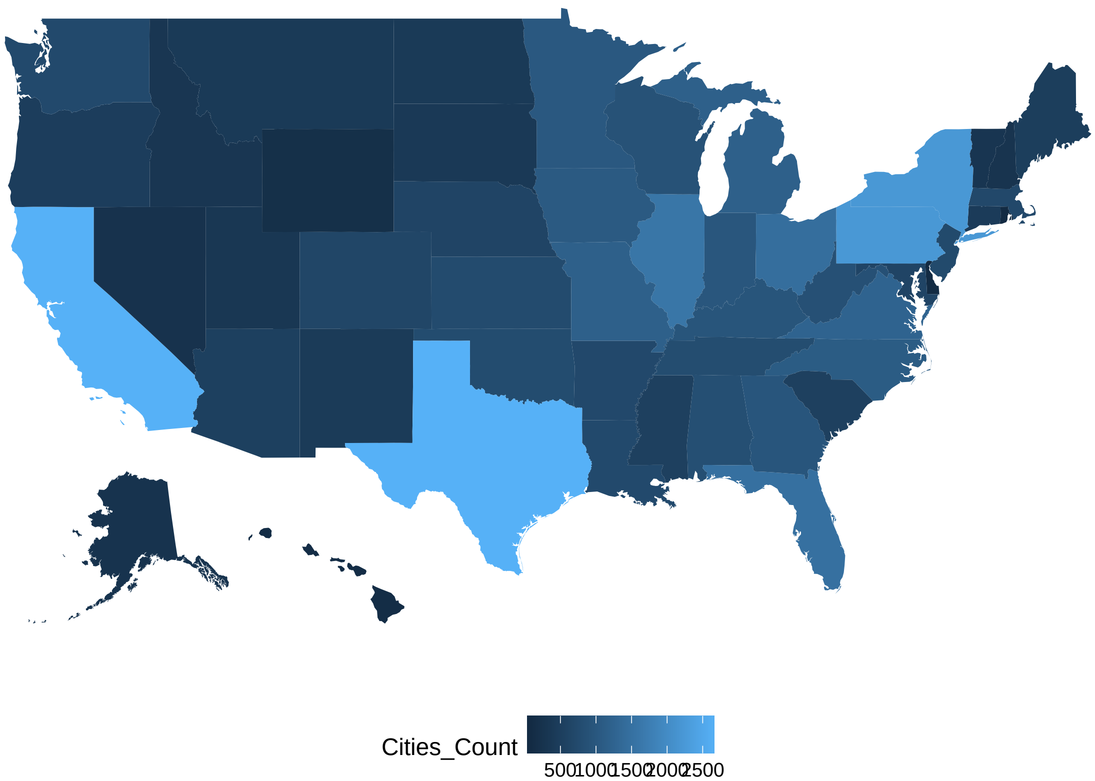
Table 3: The Number of Cities by State (first 5 rows)

Abbr	Cities_Count
AK	273
AL	838
AR	709
AZ	532
CA	2651

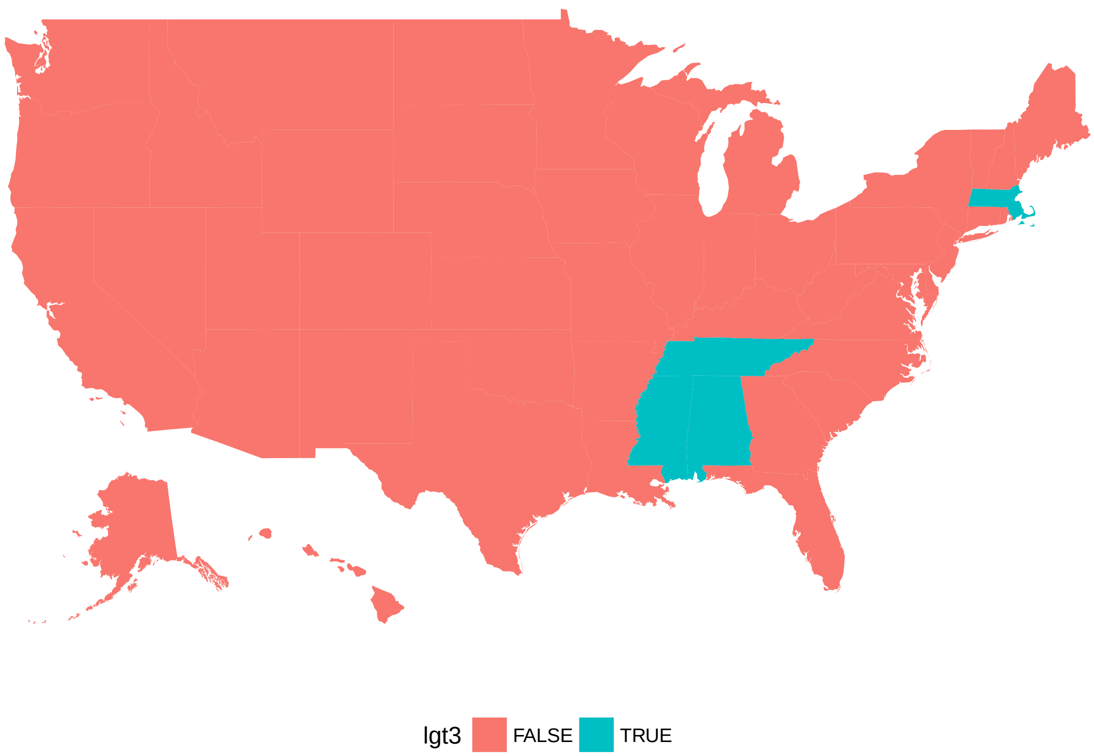
Table 4: Summary of the Number of Cities by State

Abbr	Cities_Count
AK : 1	Min. : 91.0
AL : 1	1st Qu.: 411.8
AR : 1	Median : 717.0
AZ : 1	Mean : 825.9
CA : 1	3rd Qu.:1020.5
CO : 1	Max. :2651.0
(Other):44	NA

Map 1: Colored by Count of Cities within States



Map 2: Highlights of States with 3 occurances of ANY letter



Appendix 1: R code

```r
###########################
#Problem4_fun
#function to compute the proportion of successes in a vector
###########################
compute_prop <- function(x){
  n <- length(x)
  num_S <- sum(x == 1)
  prop_S <- num_S/n
}
###########################
#Problem4_simulation
###########################
P4b_data <- matrix(rbinom(10, 1, prob = (30:40)/100), nrow = 10, ncol = 10)
prob_row <- apply(P4b_data, 1, function(x) compute_prop(x))
prob_row
prob_col <- apply(P4b_data, 2, function(x) compute_prop(x))
prob_col
###########################
#Problem4_fun_prob
#function to simulate 10 flips of a coin given a probability
###########################
simulate_prob <- function(x){
  simulate_data <- rbinom(10, 1, prob = x/100)
}
###########################
#Problem4_simu_prob
#function to simulate 10 flips of a coin given a probability
###########################
prob <- seq(10, 90, 10)
simu_res <- sapply(prob, function(x) simulate_prob(x))
res_proof <- apply(simu_res, 2, function(x) compute_prop(x))
knitr::kable(rbind(prob,res_proof,simu_res), format = "latex", booktabs=T,
             caption="Simulation when Probabilities are Given") %>%
  kable_styling(latex_options = "hold_position") %>%
  group_rows("Simulation", 2, 3, latex_gap_space = "2em")
###########################
#Problem5_starch_analysis
#get data
###########################
url <- "http://www2.isye.gatech.edu/~jeffwu/book/data/starch.dat"
starch_raw <- read.table(url, header = F, skip = 1, fill = T, stringsAsFactors = F)
colnames(starch_raw) <- c("starch", "strength", "thickness")
starch_sum <- describeBy(starch_raw[,2:3], starch_raw$starch, mat=TRUE)
knitr::kable(starch_sum[,c(2,4:7,10:13)],caption="Summary by starch Group")
###########################
# Problem5_multiplot
# Multiple plot function
# Credit to Cookbook for R
#
# ggplot objects can be passed in ..., or to plotlist (as a list of ggplot objects)
# - cols:   Number of columns in layout
```

```r
# - layout: A matrix specifying the layout. If present, 'cols' is ignored.
#
# If the layout is something like matrix(c(1,2,3,3), nrow=2, byrow=TRUE),
# then plot 1 will go in the upper left, 2 will go in the upper right, and
# 3 will go all the way across the bottom.
#
###########################
multiplot <- function(..., plotlist=NULL, file, cols=1, layout=NULL) {
  library(grid)

  # Make a list from the ... arguments and plotlist
  plots <- c(list(...), plotlist)

  numPlots = length(plots)

  # If layout is NULL, then use 'cols' to determine layout
  if (is.null(layout)) {
    # Make the panel
    # ncol: Number of columns of plots
    # nrow: Number of rows needed, calculated from # of cols
    layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
                    ncol = cols, nrow = ceiling(numPlots/cols))
  }

 if (numPlots==1) {
    print(plots[[1]])

  } else {
    # Set up the page
    grid.newpage()
    pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))

    # Make each plot, in the correct location
    for (i in 1:numPlots) {
      # Get the i,j matrix positions of the regions that contain this subplot
      matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))

      print(plots[[i]], vp = viewport(layout.pos.row = matchidx$row,
                                      layout.pos.col = matchidx$col))
    }
  }
}
###########################
#Problem5_plot
###########################
p1 <- ggplot(starch_raw, aes(x=thickness, y=strength, colour=starch, group=starch)) +
   geom_point() +
   ggtitle("Scatter Plot for strength vs. thickness")
p2 <- ggplot(starch_raw,  aes(x=thickness, y=strength, colour=starch, group=starch)) +
   geom_point() + stat_smooth(method = "lm") +
   ggtitle("Regression of strenght on thickness")
p3 <- ggplot(starch_raw, aes(x=thickness, colour=starch)) +
   geom_density() +
```

```r
   ggtitle("Density of thickness by starch")
p4 <- ggplot(starch_raw, aes(x=strength, colour=starch)) +
   geom_density() +
   ggtitle("Density of strength by starch")
p5 <- ggplot(starch_raw, aes(x=thickness, y=strength)) +
     geom_point() + facet_wrap(~starch) +
     ggtitle("Scatter Plot for strength vs. thickness by starch")
p6 <- ggplot(starch_raw, aes(x=strength, fill=starch)) +
   geom_histogram(colour="black", binwidth=30) +
   ggtitle("Histogram of strength by starch")
p7 <- ggplot(starch_raw, aes(x=starch, y=thickness)) +
   geom_boxplot(stat="boxplot", aes(colour = starch)) +
   ggtitle("Boxplot of thickness by starch")
p8 <- ggplot(starch_raw, aes(x=starch, y=strength)) +
   geom_boxplot(stat="boxplot", aes(colour = starch)) +
   ggtitle("Boxplot of strength by starch")
multiplot(p1, p2, p3, p4, p5, p6, p7, p8, cols=2)
###########################
#Problem5_plot_mar
###########################
p <- ggplot(starch_raw, aes(thickness, strength)) + geom_point() + theme_classic() +
  ggtitle("Density of a Scatter Marginal Plot")
ggMarginal(p, starch_raw, type = "histogram", yparams=list(colour="orange"))
###########################
#Problem5_plot_cor
###########################
# Basic Scatterplot Matrix
ggpairs(starch_raw, aes(colour = starch, alpha = 0.4), title = "Simple Scatterplot Matrix",
        lower = list(combo = wrap("facethist", binwidth = 0.5)))
ggplot(starch_raw, aes(x=thickness, y=strength, colour=starch)) + geom_point() +
   stat_ellipse() + ggtitle("Scatter Plot and Correlations for strength vs. thickness")
###########################
#problem6_data
###########################
# 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_states.zip")
# unzip("us_cities_states.zip", exdir = "D:/STAT_5014/05_R_apply_family")

# read in data, looks like sql dump, blah
library(data.table)
states <- fread(input = "./us_cities_and_states/states.sql",
 sep = "'", sep2 = ",", header = F, select = c(2,
4))
colnames(states) <- c("State", "Abbr")
cities <- fread(input = "./us_cities_and_states/cities_extended.sql",
 sep = "'", sep2 = ",", header = F, select = c(2,
4))
colnames(cities) <- c("Cities", "State")
```

```r
### 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?
# delete DC
states_50 <- states[-8,]
states_50$State <- tolower(states_50$State)
# delete PR & DC
cities_50 <- cities[which(State != "PR" & State != "DC"),]
###########################
#problem6_letter_count
###########################
##pseudo code
letter_count <- data.frame(matrix(NA,nrow=50, ncol=26))
colnames(letter_count) <- c(letters)
rownames(letter_count) <- c(states_50$Abbr)
getCount <- function(letter,state_name){
    temp <- strsplit(state_name,"")
    count_letter <- data.frame(matrix(NA, nrow=1, ncol=26))
    colnames(count_letter) <- c(letters)
    for (i in 1:26){
      count_letter[,i]<-sapply(letter[i], function(x) x<-sum(x==unlist(temp)))
    }
    return(count_letter)
}

for(j in 1:50){
    letter_count[j,] <- getCount(letters,states_50[j,State])
}
###########################
#problem6_maps
###########################
##pseudo code
# https://cran.r-project.org/web/packages/fiftystater/vignettes/fiftystater.html
data("fifty_states") # this line is optional due to lazy data loading
# map_id creates the aesthetic mapping to the state name
# column in your data
p1 <- ggplot(count_cities, aes(map_id = State)) + # map points to the fifty_states shape data
geom_map(aes(fill = Cities_Count), 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()) +
  ggtitle("Map 1: Colored by Count of Cities within States")

p1

letter_count["lgt3"] <- ifelse(apply(letter_count, 1, function(x) any(x > 3)),"TRUE", "FALSE")
letter_count_states <- cbind(states_50$State,letter_count)
colnames(letter_count_states)[1] <- "State"

p2 <- ggplot(letter_count_states, aes(map_id = State)) + # map points to the fifty_states shape data
geom_map(aes(fill=lgt3), 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 = "") +
```

```r
  theme(legend.position = "bottom", panel.background = element_blank()) +
  ggtitle("Map 2: Highlights of States with 3 occurances of ANY letter")

p2

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