STAT 5014 HW5

Max McGill

2017-10-03

Problem 3

A good figure should be visually appealing, relevant, informative, well structured, and easy to follow.

Problem 4

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

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

```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11]
##  [1,]    0    0    1    1    1    1    1    1    1     0     1
##  [2,]    0    0    0    0    1    0    0    0    1     1     0
##  [3,]    1    1    0    1    0    1    0    0    0     1     1
##  [4,]    0    1    1    1    0    1    0    0    0     1     0
##  [5,]    0    0    0    0    1    1    0    0    1     0     1
##  [6,]    0    0    0    0    0    0    0    0    0     1     1
##  [7,]    0    1    1    0    1    1    1    1    1     1     1
##  [8,]    0    0    1    0    0    0    1    0    1     1     0
##  [9,]    0    0    0    0    0    0    0    1    0     0     0
## [10,]    1    0    0    0    0    1    0    0    0     0     0
```

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

```
##  [1] 0.72727273 0.27272727 0.54545455 0.45454545 0.36363636 0.18181818
##  [7] 0.81818182 0.36363636 0.09090909 0.18181818
```

First by column and then by row, it is seen that the provided matrix has columns with identical success proportions and rows with homogeneous compositions. The same vector was used ten times to form the columns of the matrix. Using the eleven proportions suggested in the previous matrix's creation, the new matrix provides individualized marginal proportions as desired, shown first by column, then by row.

Problem 5

Table 1: Starch Data Summary

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

Table 2: Summary Table for CA

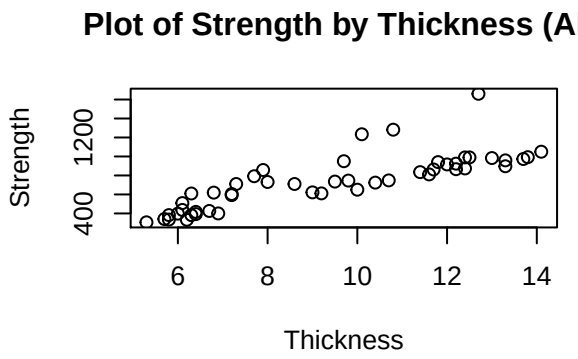
starch	strength	thickness
CA:13	Min. :610.0	Min. : 6.30
CO: 0	1st Qu.:710.0	1st Qu.: 9.00
PO: 0	Median :791.7	Median :10.40
NA	Mean :795.3	Mean :10.19
NA	3rd Qu.:916.2	3rd Qu.:11.80
NA	Max. :990.0	Max. :12.50

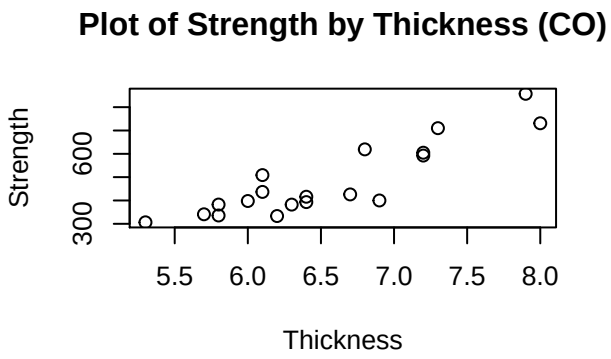
Table 3: Summary Table for CO

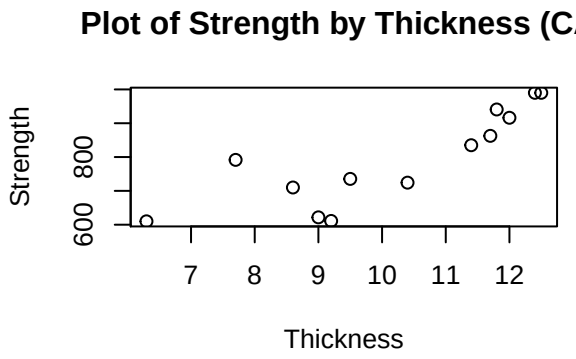
starch	strength	thickness
CA: 0	Min. :306.4	Min. :5.300
CO:19	1st Qu.:382.4	1st Qu.:6.050
PO: 0	Median :416.0	Median :6.400
NA	Mean :482.8	Mean :6.532
NA	3rd Qu.:598.6	3rd Qu.:7.050
NA	Max. :857.3	Max. :8.000

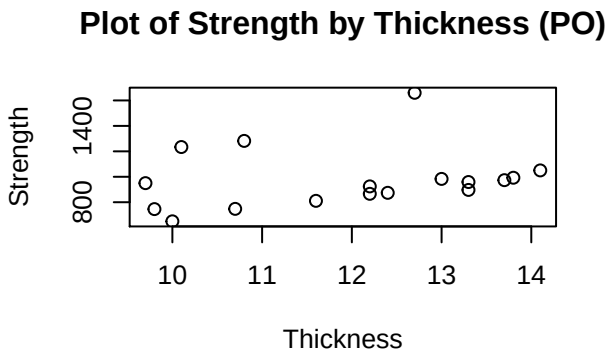
Table 4: Summary Table for PO

starch	strength	thickness
CA: 0	Min. : 650.0	Min. : 9.70
CO: 0	1st Qu.: 866.0	1st Qu.:10.70
PO:17	Median : 950.0	Median :12.20
NA	Mean : 976.4	Mean :11.96
NA	3rd Qu.: 992.5	3rd Qu.:13.30
NA	Max. :1660.0	Max. :14.10









Problem 6

Table 5: Number of Cities by State

statecode	citycount
NY	2207
PR	176
MA	703
RI	91
NH	284
ME	489
VT	309
CT	438
NJ	733
PA	2208
DE	98
DC	284
VA	1238
MD	619
WV	859
NC	1090
SC	539
GA	972
FL	1487

statecode	citycount
AL	838
TN	795
MS	533
KY	961
OH	1446
IN	989
MI	1170
IA	1060
WI	898
MN	1031
SD	394
ND	407
MT	405
IL	1587
MO	1170
KS	756
NE	620
LA	725
AR	709
OK	774
TX	2650
CO	659
WY	195
ID	325
UT	344
AZ	532
NM	426
NV	253
CA	2651
HI	139
OR	484
WA	732
AK	273

Appendix 1: R Code

Problem 4

```
# function to compute proportion of successes in a
# binomial vector (v) with values 1 and 0
succ <- function(v) {
    sum(v)/length(v)
}

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

# apply function by column
```

```r
apply(P4b_data, 2, succ)
# apply function by row
apply(P4b_data, 1, succ)
# we see that each column has the same proportion of
# success and that each row is homogeneous

# function to create vector of ten simulated coinflips
# from probability (p)
flips <- function(p) {
    rbinom(10, 1, prob = p)
}
# vector of desired probabilities wasn't sure if you
# still wanted only ten, since length(30:40)/100 = 11
despr <- c(0.3, 0.31, 0.32, 0.33, 0.34, 0.35, 0.36, 0.37,
    0.38, 0.39, 0.4)
# matrix truly desired above
coinmatr <- matrix(sapply(despr, flips), nrow = 10, ncol = 11)
coinmatr

# apply function by column
apply(coinmatr, 2, succ)
# apply column by row the eleven values cause a bit of
# uncleanliness in these proportions
apply(coinmatr, 1, succ)
```

Problem 5

```r
# get data
starch <- read.table("http://www2.isye.gatech.edu/~jeffwu/book/data/starch.dat",
    header = T)
# account for header, search for other sources of
# untidyness, begin exploration

# create basic summary for the data set as a whole
knitr::kable(summary(starch), caption = "Starch Data Summary")
# repeat for individual factors
knitr::kable(summary(subset(starch, starch == "CA")), caption = "Summary Table for CA")
knitr::kable(summary(subset(starch, starch == "CO")), caption = "Summary Table for CO")
knitr::kable(summary(subset(starch, starch == "PO")), caption = "Summary Table for PO")

# plot entire data set, treating strength as response
# and thickness as independent
par(mfcol = c(2, 2))
plot(y = starch$strength, x = starch$thickness, ylab = "Strength",
    xlab = "Thickness", main = "Plot of Strength by Thickness (All)")
# repeat for individual factors
plot(y = subset(starch$strength, starch$starch == "CA"),
    x = subset(starch$thickness, starch$starch == "CA"),
    ylab = "Strength", xlab = "Thickness", main = "Plot of Strength by Thickness (CA)")
plot(y = subset(starch$strength, starch$starch == "CO"),
    x = subset(starch$thickness, starch$starch == "CO"),
    ylab = "Strength", xlab = "Thickness", main = "Plot of Strength by Thickness (CO)")
```

```
plot(y = subset(starch$strength, starch$starch == "PO"),
    x = subset(starch$thickness, starch$starch == "PO"),
    ylab = "Strength", xlab = "Thickness", main = "Plot of Strength by Thickness (PO)")
# from this exploratory starting point, it can be seen
# that: the response and independent variables appear
# positively linearly correlated as a whole they seem
# less so when viewed independently of the other
# starches, excluding CA each starch operates in
# different ranges of thickness and strength, which
# increase together
```

Problem 6

```
# from assignment 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 = "./05_R_apply_family")

# read in data, looks like sql dump, blah
library(data.table)
states <- fread(input = "./us_cities_and_states/states.sql",
    skip = 19, sep = "'", sep2 = ",", header = F, select = c(2,
        4))
### 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?

# modified skip values to include all states and cities

# rename states columns
colnames(states) <- c("state", "state_code")
# read in cities
cities <- fread(input = "./us_cities_and_states/cities_extended.sql",
    skip = 19, sep = "'", sep2 = ",", header = F, select = c(2,
        4, 6, 8, 10, 12))
# rename cities columns
colnames(cities) <- c("city", "state_code", "zip", "latitude",
    "longitude", "county")

# create vector of states
statecode <- as.vector(unique(cities$state_code))
# create vector of city counts by state
citycount <- c()
for (i in 1:52) {
    citycount[i] <- length(subset(cities$state_code, cities$state_code ==
        statecode[i]))
}
```

```r
# table of number of cities by state
knitr::kable(cbind.data.frame(statecode, citycount), caption = "Number of Cities by State")

# function to count occurrences of a specific letter in
# a sring from assignment pseudo code letter_count <-
# data.frame(matrix(NA,nrow=52, ncol=26)) getCount <-
# function(letter,state_name){ temp <-
# strsplit(state_name,split=NULL) count <- table(temp)
# return(count) } for(i in 1:52){ letter_count[i,] <-
# xx-apply(args) }

# from assignment
# https://cran.r-project.org/web/packages/fiftystater/vignettes/fiftystater.html
# library(ggplot2) library(fiftystater)

# 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(crimes, aes(map_id = state))
# + map points to the fifty_states shape data
# geom_map(aes(fill = Assault), 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')
```