Homework 5

Ryan Kingery

10/4/2017

Problem 3

A good figure is one that the expected reader should be able to understand with minimal description. It doesn't have to be pretty enough to win any awards unless you're neurotic, but it should show the desired relationships in the data in the clearest possible way.

Problem 4

The following is a first attempt at computing proportions by column:

```
proportion <- function(vect) {
    return(sum(vect)/length(vect))
}
set.seed(12345)
P4b_data <- matrix(rbinom(10, 1, prob = (30:40)/100), nrow = 10, ncol = 10)
apply(P4b_data, 2, proportion)
```

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

Evidently we get the same proportion across every column. To alleviate this, we can apply the rbinom() function separately by defining a new function outcomes() and using apply() on that to produce a matrix with different columns.

```
outcomes <- function(p) {
    return(rbinom(10, 1, prob = p))
}
p <- as.matrix(30:40/100)
P4b_data_new <- as.matrix(apply(p, 1, outcomes), nrow = 10, ncol = 10)
apply(P4b_data_new, 2, proportion)
```

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

Note there also seems to be an issue with the new matrix having an extra observation, though I'm not quite sure where that is coming from.

Problem 5

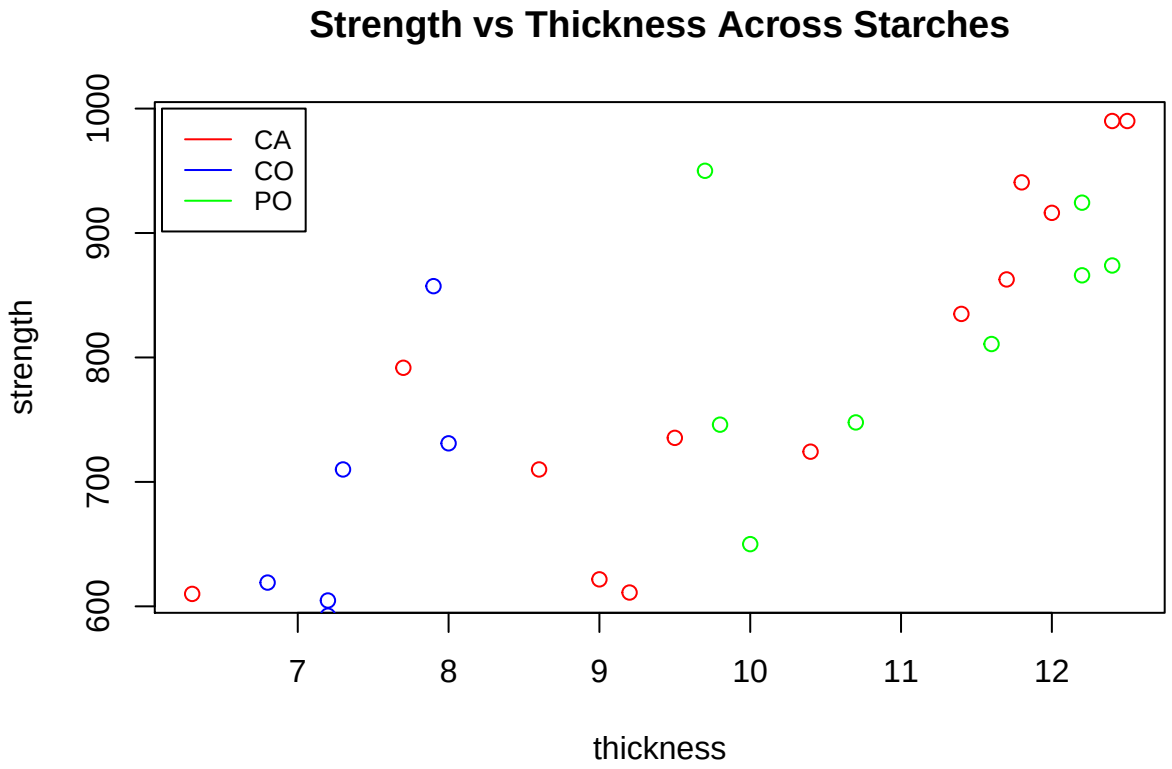
Since this was a .dat file, the easiest way to import it with minimal munging was as a table. Thankfully, the data already appears to be tidy, and there appear to be no missing values.

```
input <- read.table("http://www2.isye.gatech.edu/~jeffwu/book/data/starch.dat",
    header = TRUE, skip = 0)
input <- as_tibble(input)
str(input)
```

```
## Classes 'tbl_df', 'tbl' and '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 ...
```

Since there are only 3 starches and they appear to be categorical, a useful thing to do perhaps is to segment the dataset by starch. A reasonable question then to ask is whether strength and thickness depend on the type of starch used. We can, of course, examine this using a plot.

```
CA <- subset(input, input$starch == "CA")
CO <- subset(input, input$starch == "CO")
PO <- subset(input, input$starch == "PO")
plot(CA$thickness, CA$strength, col = "red", xlab = "thickness", ylab = "strength",
    main = "Strength vs Thickness Across Starches")
points(CO$thickness, CO$strength, col = "blue")
points(PO$thickness, PO$strength, col = "green")
legend(6.1, 1000, legend = c("CA", "CO", "PO"), col = c("red", "blue", "green"),
    lty = 1, cex = 0.8)
```



From the plot we can see that both thickness as well as strength depend on starch, and that in general there is a positive relationship between thickness and strength across all starches. From here, one could parhaps attempt to fit each of these with a curve, but I think the data over each subset are too few to justify making such a model assumption. Thus, from an initial exploratory standpoint I think this is sufficient for now.

Problem 6

The following code imports the database of US cities and states. Note that Washington DC and Puerto Rico have been removed from the cities list.

```
library(downloader)
library(stringr)
download("http://www.farinspace.com/wp-content/uploads/us_cities_and_states.zip",
    dest = "us_cities_states.zip")
unzip("us_cities_states.zip", exdir = "./")
library(data.table)
states <- fread(input = "./us_cities_and_states/states.sql", skip = 23, sep = "'",
    sep2 = ",", header = F, select = c(2, 4))
cities <- fread(input = "./us_cities_and_states/cities_extended.sql", skip = 23,
    sep = "'", sep2 = ",", header = F, select = c(2, 4))
cities <- subset(cities, V4 != "PR" & V4 != "DC")
```

A summary table of the number of cities by state is given below.

```
table(cities$V4)
```

```
##
##   AK   AL   AR   AZ   CA   CO   CT   DE   FL   GA   HI   IA   ID   IL   IN
##  273  838  709  532 2651  659  438   98 1487  972  139 1060  325 1587  989
##   KS   KY   LA   MA   MD   ME   MI   MN   MO   MS   MT   NC   ND   NE   NH
##  756  961  725  703  619  489 1170 1031 1170  533  405 1090  407  620  284
##   NJ   NM   NV   NY   OH   OK   OR   PA   RI   SC   SD   TN   TX   UT   VA
##  733  426  253 2205 1446  774  484 2208   91  539  394  795 2650  344 1238
##   VT   WA   WI   WV   WY
##  309  732  898  859  195
```

The following function uses the stringr package to count the number of occurances of a character in a string:

```
string_count <- function(letter, state) {
    return(str_count(state, letter))
}
```

Not exactly sure what's going on with the rest of this problem, but I'm out of time anyway.