Final Project

Rachel Feingold

5/1/2017

Wildfires

Packages Used

These are the pachages required by my code to run

Loading Data

I decide to start with the first year in the Google Drive, 2007.

Cleaning the 2007 Data

I used two primary functions: table(factor()) and length(x[is.na(x)]) to pick and choose which variables I would keep to observe.

Then, I transformed the character variables that had coding for different factors using the codes.dbf file.

Then, I wanted to reorder columns into more sensible train of thought. First, I have variables that describe the incident, like incident number, incident ID, date of inicident, and information about the location of the event. Then, I have variables describing the severity and extent of the incident. After, I have variables concerning the factors contributing to cause of ignition. If a human started the incident, then I ordered variables that described the demographics of the person.

```
wild07 <- wild07[,c("INC_NO", "FDID", "INC_DATE", "STATE", "LATITUDE","LONGITUDE", "AREA_TYPE", "ACRES_
```

Similar Procedure to Load and Clean Data from the year 2015

I needed to use read table because the data was in txt files.

Cleaning Data for 2015

Analysis of Data

Finally, I have cleaned all the data. Although I will not be using all the variables for analysis, I saved the ones I found most useful. Elevation, spread rate, and flame length all describe the severity of the fire. With more time, I would look at these levels of severity and see if there was an increase over this 8 year period. Then, I would see if there was any correlation with this and acres burned.

For this project, the question I am trying to solve is: was there a major change in the frequency of wildfires in the US from 2005 to 2017. Do some states have less fires as a result of changes in national policies. The quantitative variables I am using to measure this change is amount of fires is ACRES_BURN, which calculates the total acres burned for each fire incident. I am going to see how this number changes over time for each incident different states.

Data Plots

I wanted to look at total acres burned per state and see how this changes over time. Do some states have more fires in 2007 than in 2015 and why?

```
# Plot Distribution of Acres Burned
# I needed to change to a logarithmic scale for Acres Burned because the range of values is so large (O
dat1 <- select(wild07, STATE, ACRES_BURN)
# I  wanted to ignore fires that were less than 1 acre of land burned.
dat1 <- filter(dat1, ACRES_BURN > 1)
dat1 <- mutate(dat1, LOG_ACRES = log10(dat1$ACRES_BURN))
# Plot distribution
dist1 <- ggplot(dat1) + geom_density(aes(x = LOG_ACRES)) + ggtitle("ggplot of ACRES BURNED distribution

# Compare this to 2015 data
dat2 <- select(wild15, STATE, ACRES_BURN)
dat2 <- filter(dat2, ACRES_BURN > 1)
dat2 <- mutate(dat2, LOG_ACRES = log10(dat2$ACRES_BURN))
dist2 <- ggplot(dat2) + geom_density(aes(x = LOG_ACRES)) + ggtitle("ggplot of ACRES BURNED distribution

# Use Plotly to mouse over the points in the graph
ggplotly(dist1)
```

```
ggplotly(dist2)
```

Looking at these plots, it makes sense that there are a lot more wildfires that cover a smaller acreage. Thankfully, wildfires that burn a larger area of land are less common. Just looking at these graphs, it is difficult to see whether there was a statistically significant change in the amount of acres burned during this time period. I decided to look at bar graphs showing the distribution of the total number of wildfires burned in each state. I was curious if there was a mjaor change. Normally, the climatic conditions as well as the geographic landscape of a state will determine whether there are many fires. I hypothesized that there should not be much change.

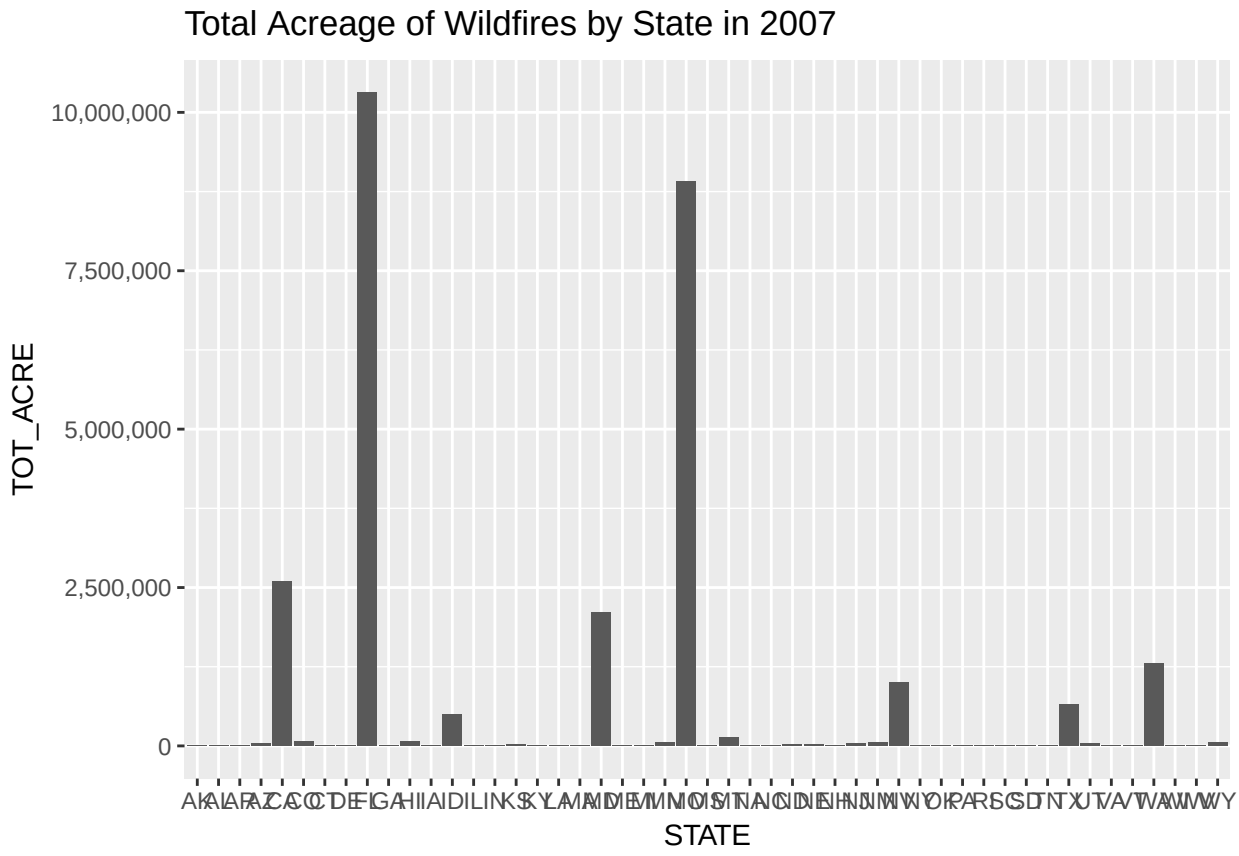
```
# Plot of Distribution of Wildfires Per State- 2007
ggplotly(ggplot(dat1) + geom_bar(aes(x = STATE)) + ggtitle("Counts of Wildfires in Each State in 2007"))
```

```
# Plot of Distribution of Wildfires Per State- 2015
ggplotly(ggplot(dat2) + geom_bar(aes(x = STATE)) + ggtitle("Counts of Wildfires in Each State in 2015"))
```

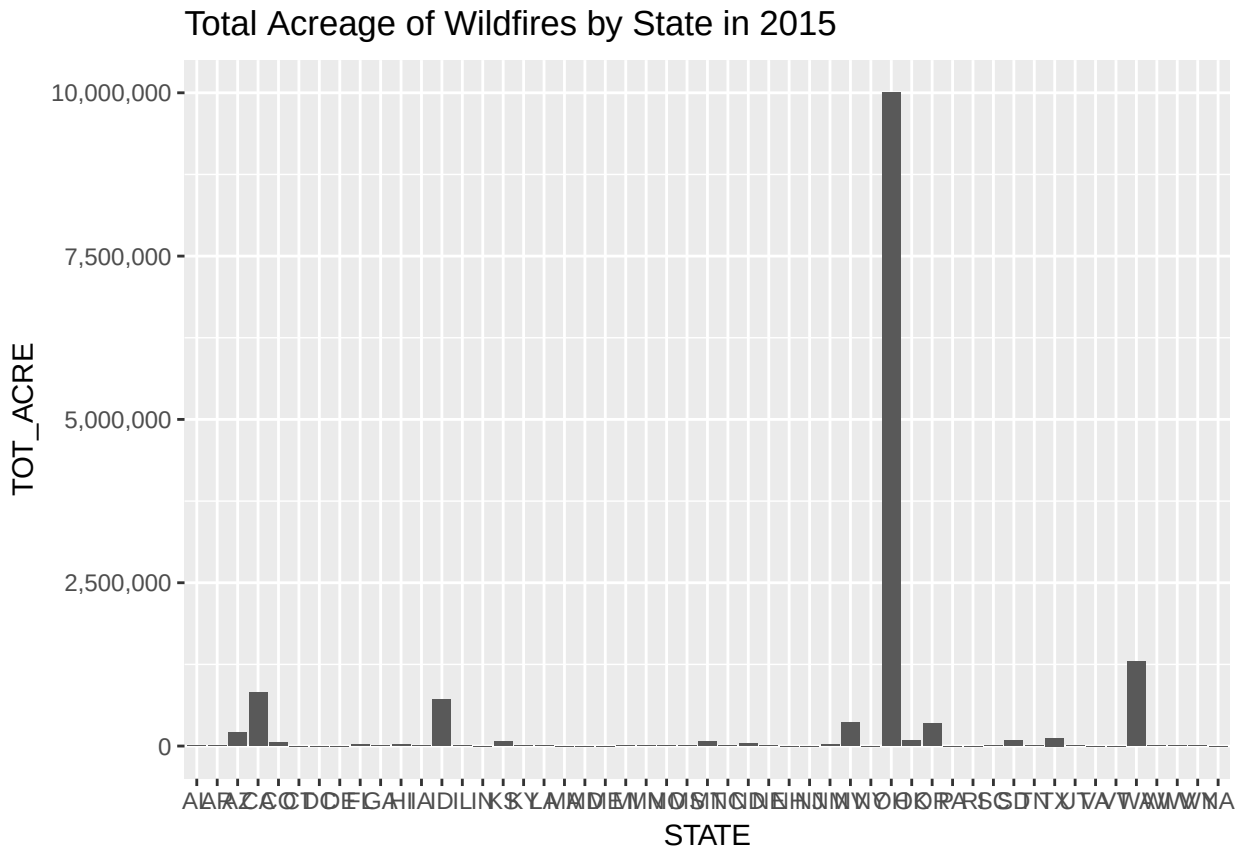
```
# Try plotting the total amount of acres burned per state.
# First need to clean dat1 and dat2 so that we have a cumulative sum of the total acreage of land burne

# 2007
tot_acres1 <- within(dat1, {TOT_ACRE <- ave(dat1$ACRES_BURN, dat1$STATE, FUN = cumsum)})
tot_acres1 <- select(tot_acres1, STATE, TOT_ACRE)
tot_acres1 <- tot_acres1 %>% group_by(STATE)  %>% slice(which.max(TOT_ACRE))

# 2015
tot_acres2 <- within(dat2, {TOT_ACRE <- ave(dat2$ACRES_BURN, dat2$STATE, FUN = cumsum)})
tot_acres2 <- select(tot_acres2, STATE, TOT_ACRE)
tot_acres2 <- tot_acres2 %>% group_by(STATE)  %>% slice(which.max(TOT_ACRE))

# Plot of Distribution of TOT_ACRES of Wildfires Per State
require(scales)
```

```
## Loading required package: scales
```

```
# 2007
ggplot(subset(tot_acres1, !is.na(TOT_ACRE)), aes(x= STATE, y = TOT_ACRE)) + geom_col() +  scale_y_contin
```




```
# 2015
ggplot(subset(tot_acres2, !is.na(TOT_ACRE)), aes(x= STATE, y = TOT_ACRE)) + geom_col() +  scale_y_contin
```



Data Tables These graphs are difficult to read, let's look at a table of the total acres burned for each state.

```
ta07 <- arrange(tot_acres1, desc(TOT_ACRE))
ta15 <- arrange(tot_acres2, desc(TOT_ACRE))

# Use table to get counts of wildfire incidents per state in 2007 and 2015
states07 <- table(wild07$STATE)
states07<- sort(states07,decreasing = T)

states15 <- table(wild15$STATE)
states15<- sort(states15,decreasing = T)

rm(ta07)
rm(ta15)
rm(states07)
rm(states15)
```

State	Total Wildlands Acreage Burned in 2007
FL	10311883.1
MO	8905000.5
CA	2594418.2
MD	2113027.5
WA	1300468.2
NV	995931.2
TX	659759.2
ID	495536.0

State	Total Wildlands Acreage Burned in 2007
MT	136638.7
HI	72208.1

State	Total Wildlands Acreage Burned in 2015
OH	10000680
WA	1291781
CA	822280
ID	711350
NV	359926
OR	341561
AZ	209769
TX	123791
OK	84736
SD	78592

State	CA	TX	GA	FL	WA	KS
Total Wildfire Incidents in 2007	11831	5192	4026	3772	3449	2437

State	CA	TX	FL	WA	KS	GA
Total Wildfire Incidents in 2015	9030	6216	3458	2941	2878	2070

There is a disparity between total Wildfires per state and total acres burned! One explanation is that I removed all the fires with an acerage < 1 from the data. We should use a statistical approach or method to see if there is really a difference in the amount of wildfires in different states from 2007 to 2015.

Hypothesis Test

I ran a two sample t-test, testing if there is a statisticaly significant difference in mean amount of total acres burned in 2007 and 2015. The null hypthesis is that the there is no difference in mean amount of acres burned. The p value of the test was 0.39. At a significance level, alpha= 0.05, we fail to reject the null. We do not have enough evidence to say that there is a difference in mean amount of total acres burned in 2007 and 2015.

```
t.test(tot_acres1$TOT_ACRE, tot_acres2$TOT_ACRE, mu = 0, var.equal = T)
```

```
##
##  Two Sample t-test
##
## data:  tot_acres1$TOT_ACRE and tot_acres2$TOT_ACRE
## t = 0.8558, df = 98, p-value = 0.3942
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -383825.1  965885.2
## sample estimates:
## mean of x mean of y
##  573870.5  282840.4
```

Playing with Maps

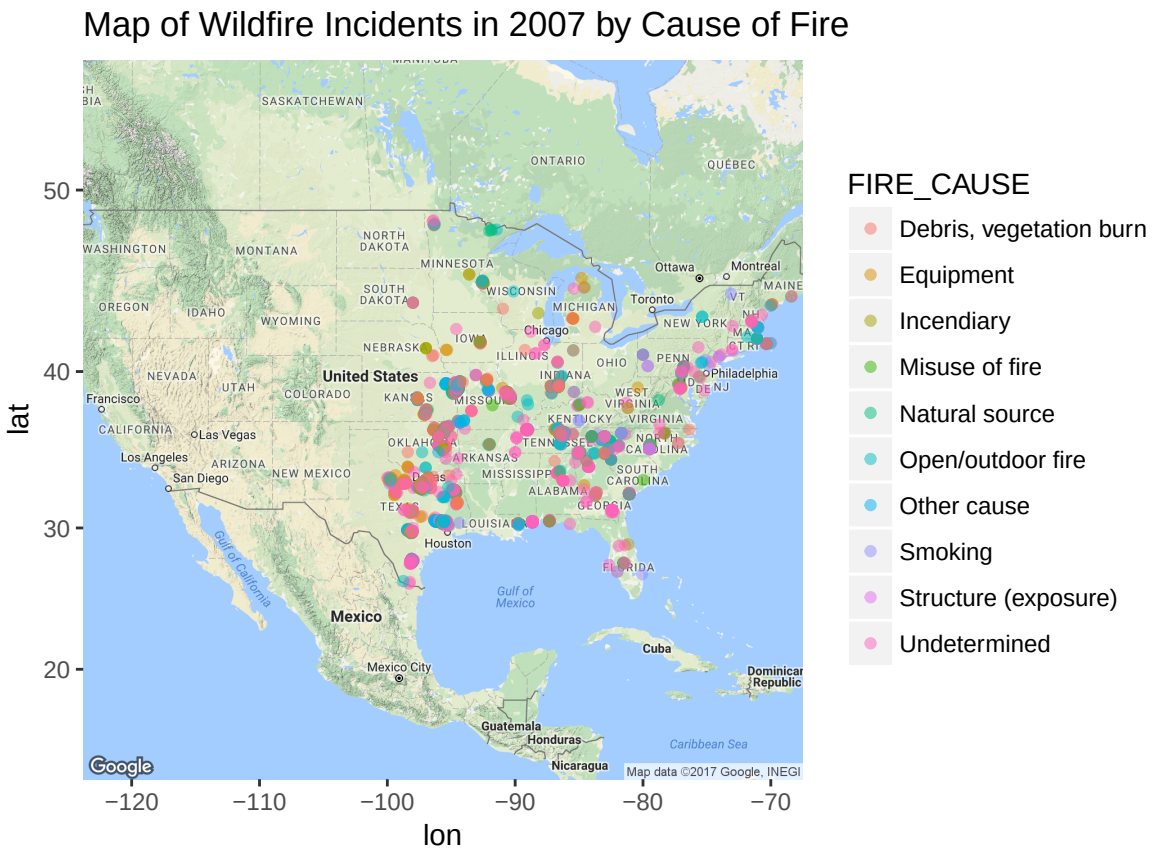
I'd like to map different causes of fires over the map of the US, to see if one part of the country has more fires than another.

```
map1 <- get_map(location = "united states", zoom = 4, maptype = "terrain", source = "google", color = "
```

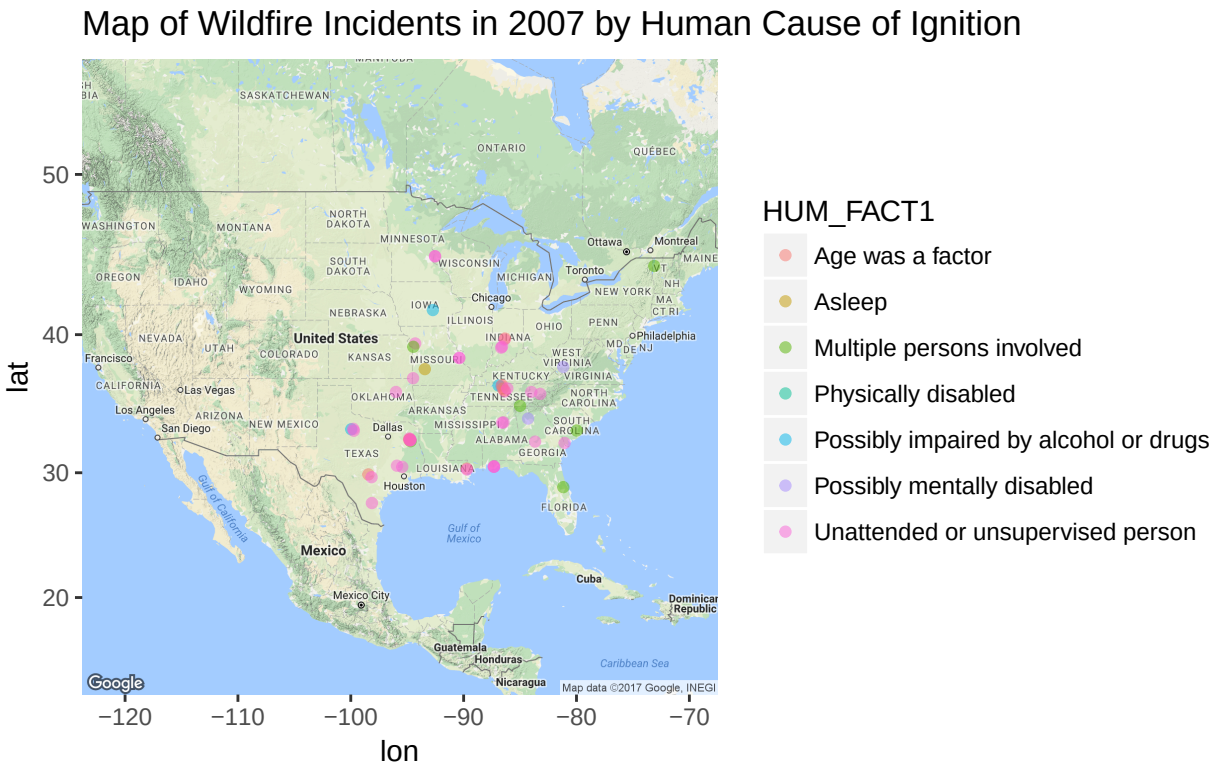
```
## Map from URL : http://maps.googleapis.com/maps/api/staticmap?center=united+states&zoom=4&size=640x640
```

```
## Information from URL : http://maps.googleapis.com/maps/api/geocode/json?address=united%20states&sensor
```

```
# This is just a map of the US
#ggmap(map1)

# Mapping the wildfire incidents, showing the different causes related to the fire, by colored points.
data <- select(wild07, LATITUDE, LONGITUDE, FIRE_CAUSE)
# If there were NA values for these character variables, the codes set their value as the name of the v
data <- filter(data, FIRE_CAUSE != "WILDLAND FIRE CAUSE")
# add points
m1 <- ggmap(map1) + geom_point(aes(x = LONGITUDE, y = LATITUDE, color = FIRE_CAUSE), data = data, alpha
m1
```




```
# Mapping the wildfire incidents, showing the different human factors related to fire ignition, by colo
data1 <- select(wild07, LATITUDE, LONGITUDE, HUM_FACT1)
data1 <- filter(data1, HUM_FACT1 != "HUMAN FACTORS CONTRIBUTING TO IGNITION 1")
data1 <- filter(data1, HUM_FACT1 != "None")
m2 <- ggmap(map1) + geom_point(aes(x = LONGITUDE, y = LATITUDE, color = HUM_FACT1), data = data1, alpha
m2
```



Maps that best exemplifies my original research question:

```
# Mapping the total amount of Acres Burned by State for 2007
# Need to transform state abbreviations to full names first
#'x' is the column of a data.frame that holds 2 digit state codes
stateFromLower <-function(x) {
   #read 52 state codes into local variable [includes DC (Washington D.C. and PR (Puerto Rico)]
  st.codes<-data.frame(
                      state=as.factor(c("AK", "AL", "AR", "AZ", "CA", "CO", "CT", "DC", "DE", "FL", "GA"
                                        "HI", "IA", "ID", "IL", "IN", "KS", "KY", "LA", "MA", "MD", "M
                                        "MI", "MN", "MO", "MS",  "MT", "NC", "ND", "NE", "NH", "NJ", "
                                        "NV", "NY", "OH", "OK", "OR", "PA", "PR", "RI", "SC", "SD", "T
                                        "TX", "UT", "VA", "VT", "WA", "WI", "WV", "WY")),
                      full=as.factor(c("alaska","alabama","arkansas","arizona","california","colorado",
                                       "connecticut","district of columbia","delaware","florida","georg
                                       "hawaii","iowa","idaho","illinois","indiana","kansas","kentucky"
                                       "louisiana","massachusetts","maryland","maine","michigan","minne
                                       "missouri","mississippi","montana","north carolina","north dakot
                                       "nebraska","new hampshire","new jersey","new mexico","nevada",
                                       "new york","ohio","oklahoma","oregon","pennsylvania","puerto ric
                                       "rhode island","south carolina","south dakota","tennessee","texa
                                       "utah","virginia","vermont","washington","wisconsin",
                                       "west virginia","wyoming"))
                       )
     #create an nx1 data.frame of state codes from source column
  st.x<-data.frame(state=x)
     #match source codes with codes from 'st.codes' local variable and use to return the full state nam
  refac.x<-st.codes$full[match(st.x$state,st.codes$state)]
     #return the full state names in the same order in which they appeared in the original source
  return(refac.x)
}
```

```
TA1 <- select(tot_acres1, everything())
TA1$STATE <- stateFromLower(TA1$STATE)

#libraries needed
require(zipcode)
```

```
## Loading required package: zipcode
```

```
require("choroplethrMaps")
```

```
## Loading required package: choroplethrMaps
```

```
require("choroplethr")
```

```
## Loading required package: choroplethr
```

```
## Loading required package: acs
```

```
## Loading required package: stringr
```

```
## Loading required package: plyr
```

```
## ------------------------------------------------------------------------------
```

```
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
```

```
## ------------------------------------------------------------------------------
```

```
##
## Attaching package: 'plyr'
```

```
## The following objects are masked from 'package:plotly':
##
##     arrange, mutate, rename, summarise
```

```
## The following object is masked from 'package:lubridate':
##
##     here
```

```
## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
## Loading required package: XML
```

```
##
## Attaching package: 'acs'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
## The following object is masked from 'package:base':
##
##     apply
```

```
# Base Map
states_map <- map_data("state")
```

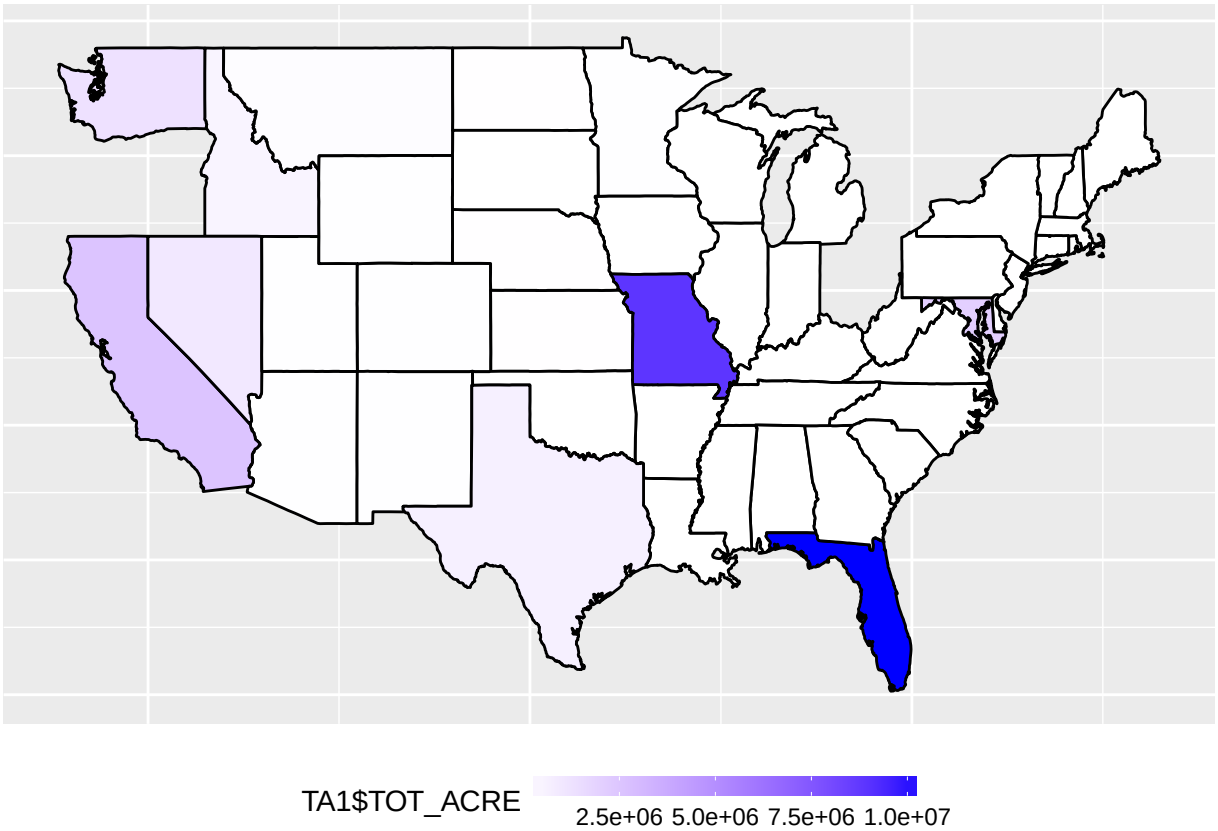
```
##
## Attaching package: 'maps'
```

```
## The following object is masked from 'package:plyr':
##
##     ozone
```

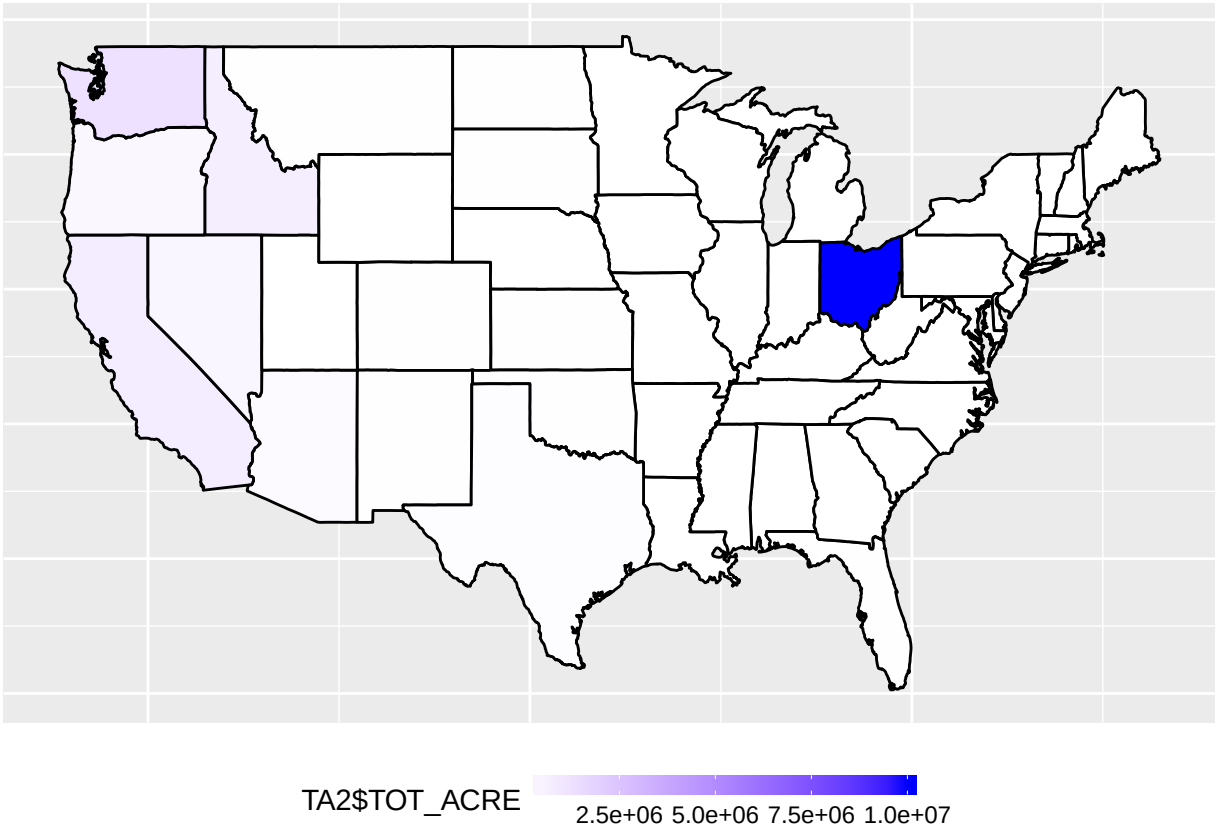
```
head(states_map)
```

```
##        long      lat group order  region subregion
## 1 -87.46201 30.38968     1     1 alabama      <NA>
## 2 -87.48493 30.37249     1     2 alabama      <NA>
## 3 -87.52503 30.37249     1     3 alabama      <NA>
## 4 -87.53076 30.33239     1     4 alabama      <NA>
## 5 -87.57087 30.32665     1     5 alabama      <NA>
## 6 -87.58806 30.32665     1     6 alabama      <NA>
```

```
# Maps the total acres burned, as density
ggplot(TA1, aes(map_id = TA1$STATE)) +
 geom_map(aes(fill = TA1$TOT_ACRE), map = states_map, color ="black") +
 expand_limits(x = states_map$long, y = states_map$lat) +
 theme(legend.position = "bottom",
 axis.ticks = element_blank(),
 axis.title = element_blank(),
 axis.text = element_blank()) +
 scale_fill_gradient(low="white", high="blue") +
 guides(fill = guide_colorbar(barwidth = 10, barheight = .5))
```



```
# 2017
TA2 <- select(tot_acres2, everything())
TA2$STATE <- stateFromLower(TA2$STATE)
TA2$STATE <- stateFromLower(tot_acres2$STATE)

# Maps the total acres burned, as density
ggplot(TA2, aes(map_id = TA2$STATE)) +
 geom_map(aes(fill = TA2$TOT_ACRE), map = states_map, color ="black") +
 expand_limits(x = states_map$long, y = states_map$lat) +
 theme(legend.position = "bottom",
 axis.ticks = element_blank(),
 axis.title = element_blank(),
 axis.text = element_blank()) +
 scale_fill_gradient(low="white", high="blue") +
 guides(fill = guide_colorbar(barwidth = 10, barheight = .5))
```



Saving Data to use in Shiny Apps

Link to Shiny App

Unfortunately, I was unable to publish my shiny app onto shinyapps.io. There was an error message with the package "tigris" which i didn't even use. It took very long to run and I could not trouble shoot this error. The error looked like:

- installing *source* package 'tigris' ... ** package 'tigris' successfully unpacked and MD5 sums checked ** R ** data *** moving datasets to lazyload DB ** preparing package for lazy loading Error in

dyn.load(file, DLLpath = DLLpath, ...) : unable to load shared object '/usr/local/lib/R/site-library/rgdal/libs/rgdal.so': libgdal.so.1: cannot open shared object file: No such file or directory ERROR: lazy loading failed for package 'tigris'

- removing '/usr/local/lib/R/site-library/tigris' ################################## End Task Log ################################## Error: Unhandled Exception: Child Task 386668264 failed: Error building image: Error building tigris (0.5). Build exited with non-zero status: 1 Execution halted

Here is the link to my shinyapp.io account anyways. I will also upload my app to github.

https://rfeingold.shinyapps.io/fpshiny2/