615 Project

Xijia Luo

November 11, 2020

Input&Intro Data

The data is about the disaster distribution in the United State.

```r
r <- read_csv("C:/Users/Jack-/Desktop/MA615/project/r.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   disasterNumber = col_double(),
##   declarationDate = col_datetime(format = ""),
##   pwNumber = col_double(),
##   countyCode = col_double(),
##   stateNumberCode = col_double(),
##   projectAmount = col_double(),
##   federalShareObligated = col_double(),
##   totalObligated = col_double(),
##   obligatedDate = col_datetime(format = ""),
##   lastRefresh = col_datetime(format = "")
## )

## See spec(...) for full column specifications.
```

```r
head(r)
```

```
## # A tibble: 6 x 22
##   disasterNumber declarationDate     incidentType pwNumber applicationTitle
##            <dbl> <dttm>              <chr>           <dbl> <chr>
## 1           1239 1998-08-26 04:00:00 Severe Stor~       41 Not Provided
## 2           1239 1998-08-26 04:00:00 Severe Stor~       51 Not Provided
## 3           1239 1998-08-26 04:00:00 Severe Stor~       43 Not Provided
## 4           1239 1998-08-26 04:00:00 Severe Stor~        2 (L)
## 5           1239 1998-08-26 04:00:00 Severe Stor~       47 Not Provided
## 6           1239 1998-08-26 04:00:00 Severe Stor~       35 Not Provided
## # ... with 17 more variables: applicantId <chr>, damageCategoryCode <chr>,
## #   dcc <chr>, damageCategory <chr>, projectSize <chr>, county <chr>,
## #   countyCode <dbl>, state <chr>, stateCode <chr>, stateNumberCode <dbl>,
## #   projectAmount <dbl>, federalShareObligated <dbl>, totalObligated <dbl>,
## #   obligatedDate <dttm>, hash <chr>, lastRefresh <dttm>, id <chr>
```

Data Cleaning

First of all, we need to select the variable we need. They are disasterNumber which helps us identify which disaster we are looking, county and state which help us locate where these disaster happened, incident type

which helps us know what kinds of disaster it was, and state number code and county code which help us draw the map.

```r
data<-r%>%group_by(disasterNumber,county,state)%>%summarise(projectAmount=sum(projectAmount),countyCode=
data1<-data%>%group_by(county)%>%summarise(projectAmount=sum(projectAmount),countyCode=max(countyCode),
head(data1)
```

```
## # A tibble: 6 x 6
##   county    projectAmount countyCode stateNumberCode incidentType   state
##   <chr>             <dbl>      <dbl>           <dbl> <chr>          <chr>
## 1 Abbeville        588681.          1              45 Severe Storm(~ South Carol~
## 2 Acadia        16758715.          1              22 Severe Storm(~ Louisiana
## 3 Accomack       2613022.          1              51 Severe Storm(~ Virginia
## 4 Ada            3948655.          1              16 Flood          Idaho
## 5 Adair         14558158.          1              40 Snow           Oklahoma
## 6 Adams         58364474.          3              55 Tornado        Wisconsin
```

MAPPing

Here is examples for our data visualization on US Map.

Based on DisasterNumber

Let's take #1264 disaster for example.
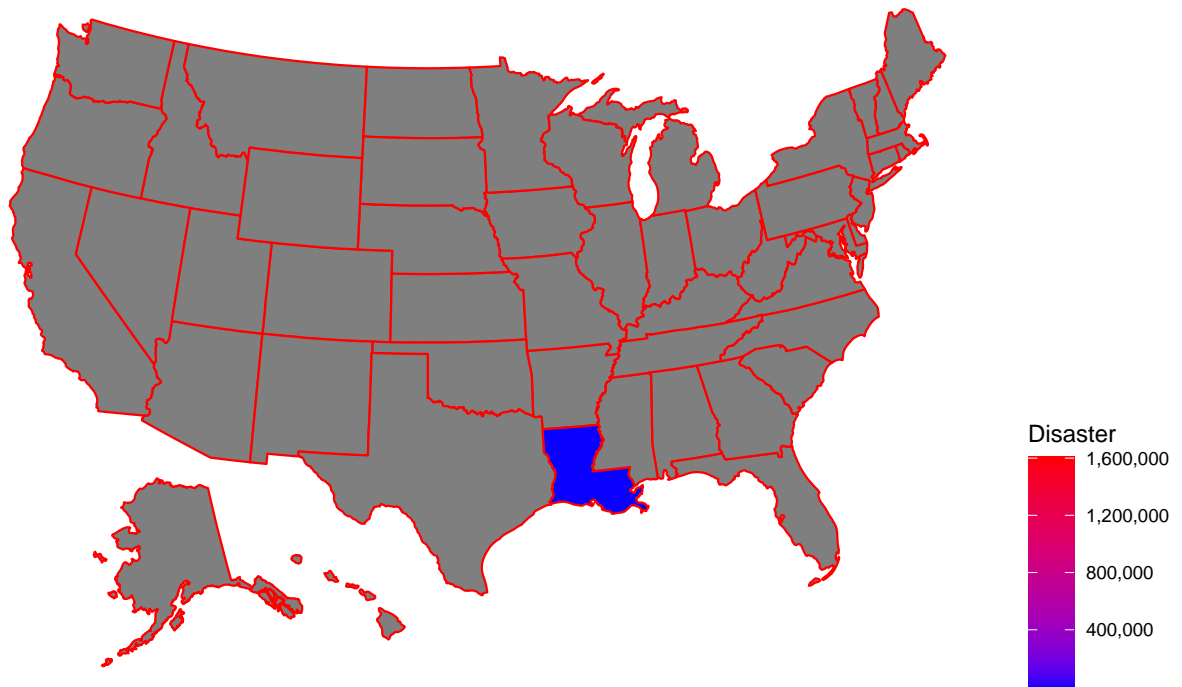
```r
data2<-data[data$disasterNumber == 1264,]

#based on us map
plot_usmap(data = data2, values = "projectAmount", color = "red") +
    scale_fill_continuous(
      low = "blue", high = "red", name = "Disaster", label = scales::comma
    ) + labs(title = "US Disaster") + theme(legend.position = "right")
```

```
## Warning: Use of `map_df$x` is discouraged. Use `x` instead.
```

```
## Warning: Use of `map_df$y` is discouraged. Use `y` instead.
```

```
## Warning: Use of `map_df$group` is discouraged. Use `group` instead.
```
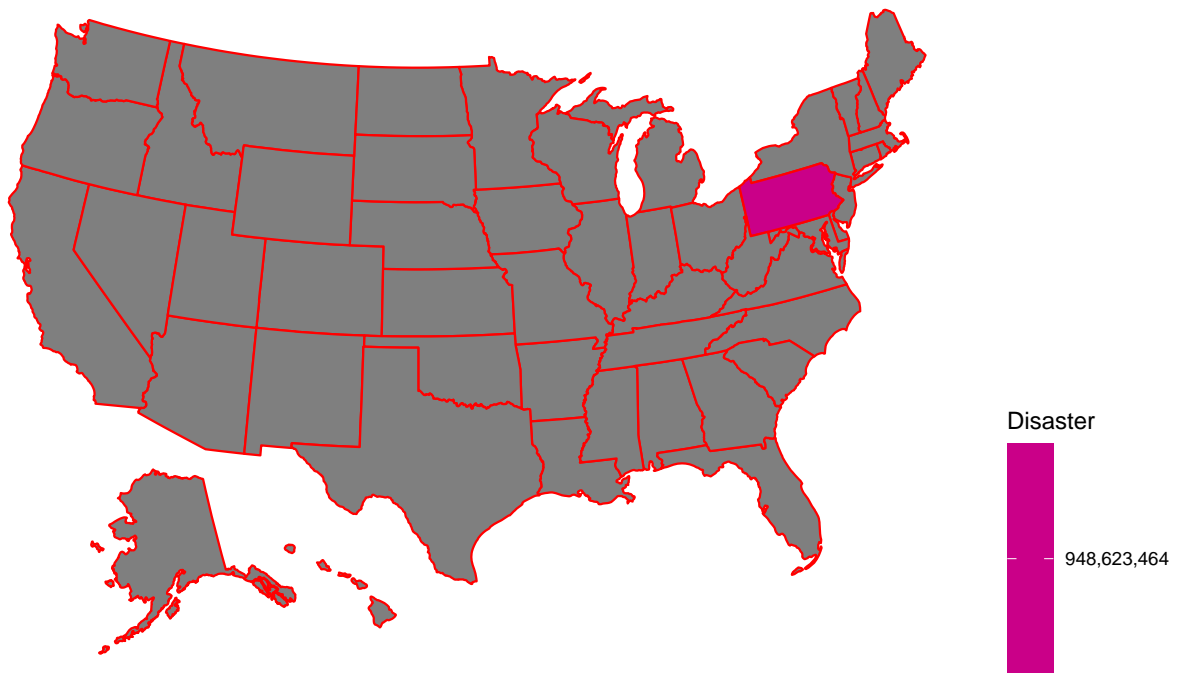
US Disaster


```r
#based on State map
a<-data2$state
data2<-data2%>%filter(county != "Statewide")
    data2<-data2 %>% rowwise %>% mutate(fips = 1000*stateNumberCode+countyCode)
plot_usmap(regions = "county", data = data2, values = "projectAmount",include = a ,color = "red") +
        scale_fill_continuous(
          low = "blue", high = "red", name = "Disaster", label = scales::comma
        ) + labs(title = "US Disaster") + theme(legend.position = "right")
```

```
## Warning: Use of `map_df$x` is discouraged. Use `x` instead.

## Warning: Use of `map_df$y` is discouraged. Use `y` instead.

## Warning: Use of `map_df$group` is discouraged. Use `group` instead.
```

US Disaster
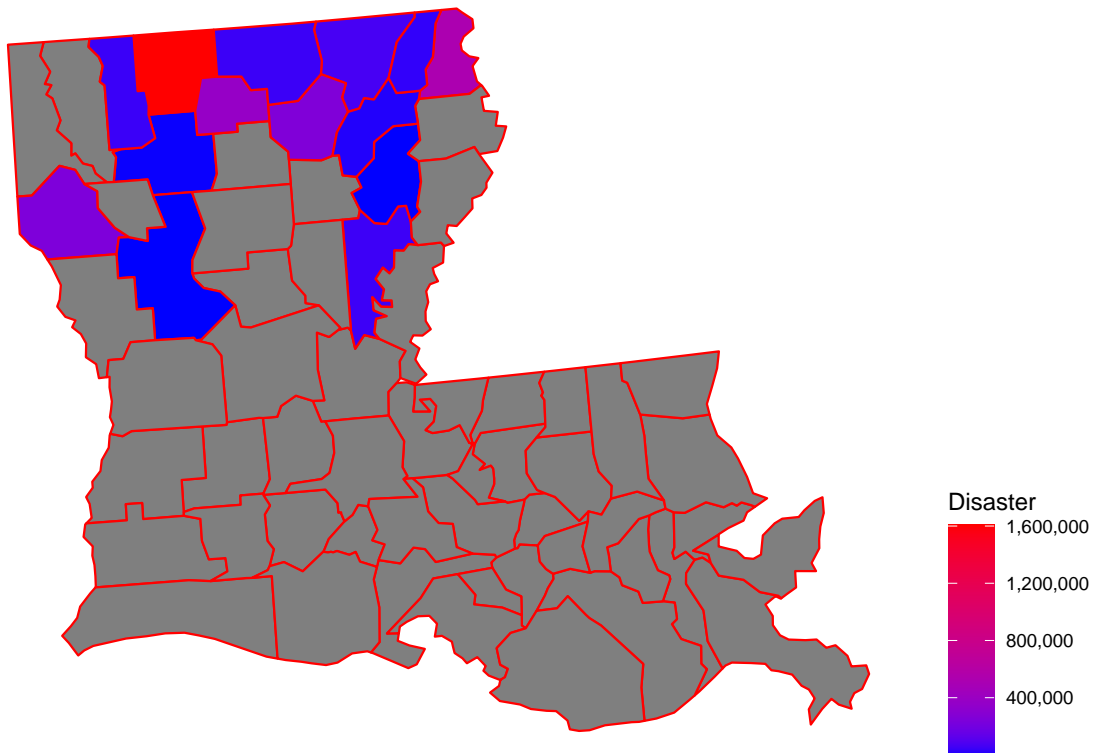
Based on State

Let's take PA as an example

```
data3<-data1[data1$state == "Pennsylvania",]

#Based on the US MaP
data3_1<-data3%>%group_by(state)%>%summarise(projectAmount=sum(projectAmount),countyCode=max(countyCode)

plot_usmap(data = data3_1, values = "projectAmount", color = "red") +
    scale_fill_continuous(
      low = "blue", high = "red", name = "Disaster", label = scales::comma
    ) + labs(title = "US Disaster") + theme(legend.position = "right")
```

```
## Warning: Use of `map_df$x` is discouraged. Use `x` instead.

## Warning: Use of `map_df$y` is discouraged. Use `y` instead.

## Warning: Use of `map_df$group` is discouraged. Use `group` instead.
```

US Disaster

Disaster

948,623,464

```r
#Based on the state Map
a<-data3$state
data3_2<-data3%>%filter(county != "Statewide")
    data3_2<-data3_2 %>% rowwise %>% mutate(fips = 1000*stateNumberCode+countyCode)
plot_usmap(regions = "county", data = data3_2, values = "projectAmount",include = a ,color = "red") +
    scale_fill_continuous(
        low = "blue", high = "red", name = "Disaster", label = scales::comma
    ) + labs(title = "US Disaster") + theme(legend.position = "right")
```

```
## Warning: Use of `map_df$x` is discouraged. Use `x` instead.

## Warning: Use of `map_df$y` is discouraged. Use `y` instead.

## Warning: Use of `map_df$group` is discouraged. Use `group` instead.
```

US Disaster
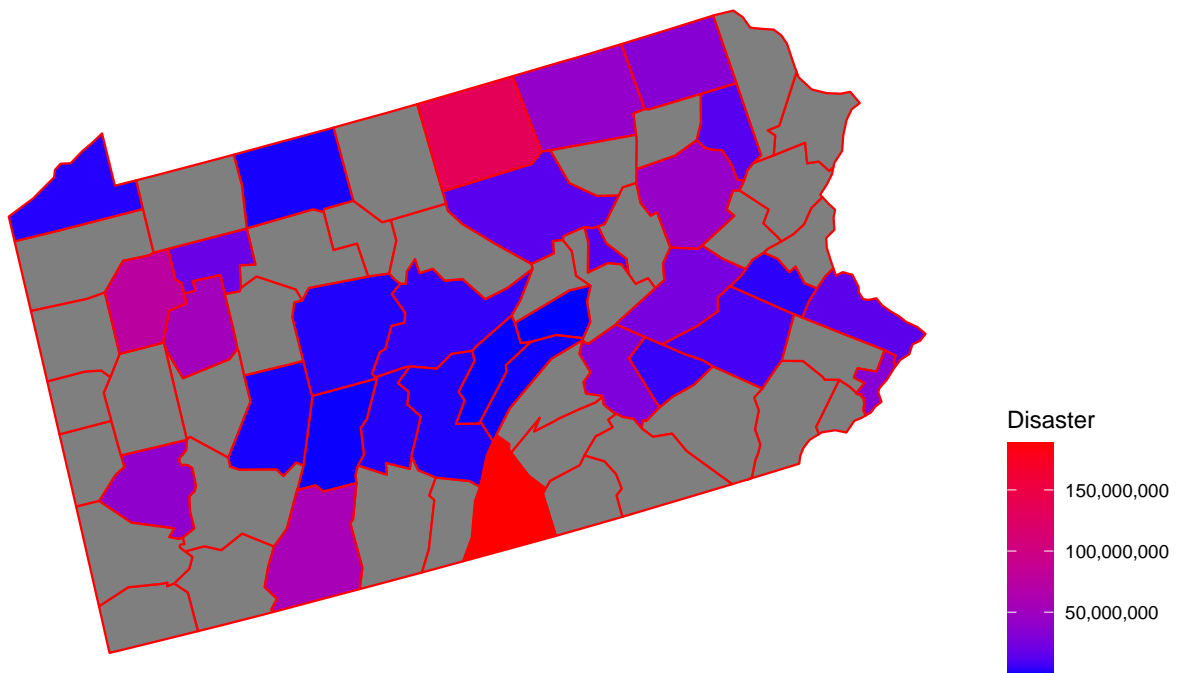
Based on Incident Type

Let's take Biological as an example

```
data4<-data1[data1$incidentType == "Biological",]

#Based on the US MaP
data4_1<-data4%>%group_by(state)%>%summarise(projectAmount=sum(projectAmount),countyCode=max(countyCode))
plot_usmap(data = data4_1, values = "projectAmount", color = "red") +
    scale_fill_continuous(
      low = "blue", high = "red", name = "Disaster", label = scales::comma
    ) + labs(title = "US Disaster") + theme(legend.position = "right")
```

```
## Warning: Use of `map_df$x` is discouraged. Use `x` instead.

## Warning: Use of `map_df$y` is discouraged. Use `y` instead.

## Warning: Use of `map_df$group` is discouraged. Use `group` instead.
```

US Disaster
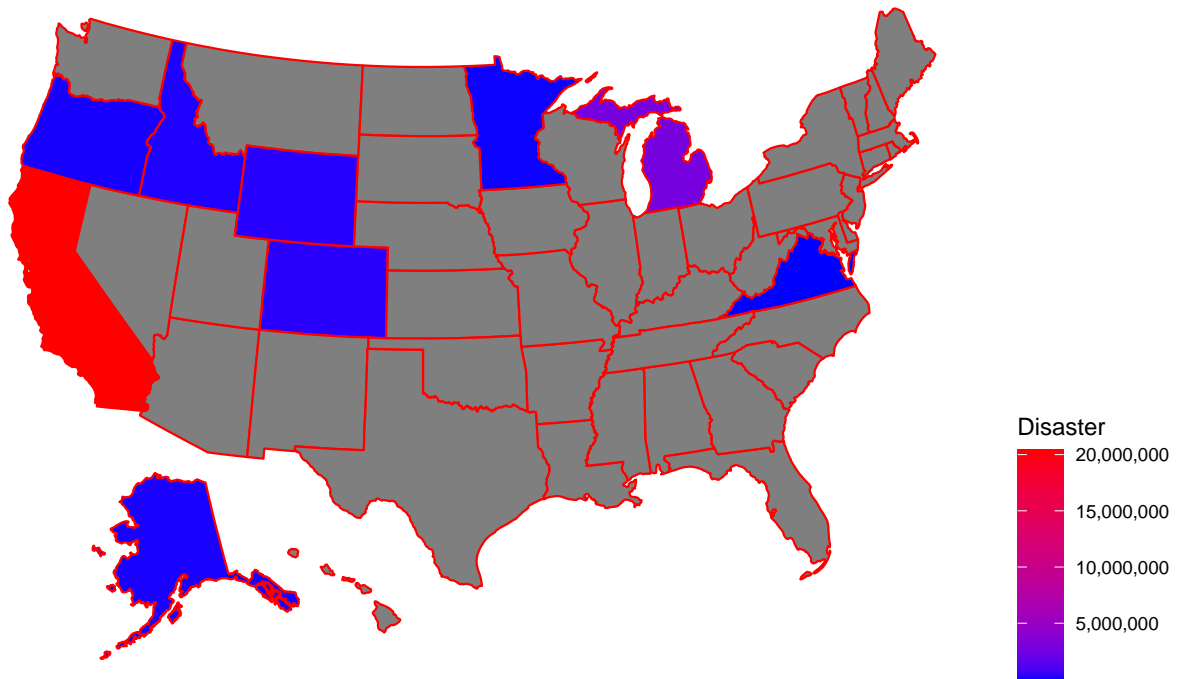

```r
#Based on the state Map
a<-data4$state
data4_2<-data4%>%filter(county != "Statewide")
    data4_2<-data4_2 %>% rowwise %>% mutate(fips = 1000*stateNumberCode+countyCode)
plot_usmap(regions = "county", data = data4_2, values = "projectAmount",include = a ,color = "red") +
    scale_fill_continuous(
        low = "blue", high = "red", name = "Disaster", label = scales::comma
    ) + labs(title = "US Disaster") + theme(legend.position = "right")
```

```
## Warning: Use of `map_df$x` is discouraged. Use `x` instead.
```

```
## Warning: Use of `map_df$y` is discouraged. Use `y` instead.
```

```
## Warning: Use of `map_df$group` is discouraged. Use `group` instead.
```

US Disaster

Disaster

20,000,000

15,000,000

10,000,000

5,000,000