Extended Mapping Assignment

Fan Feng

2020/11/11

1. Background

Main Task:

The purpose of this assignment is to make sure that you know the basics of using R to produce documents, presentations, and shiny apps. These are basic skills that every data scientists must have. Use the mapping project you have just completed as a starting point. I have provided additional county level data from FEMA and FEC so that you can produce maps with more features.

Using this data, produce a document using rmarkdown, a presentation using revealjs, and a Shiny application that you publish on shinyapps.io. Your document should include structure elements such as headings, lists, tables. Your presentation should also include these elements. Your Shiny application should allow users to select of adjust the display of maps and data plots. All of these deliverables should contain maps and data plots. You don't need to go overboard. Do enough to demonstrate your skill.

2. Data Cleaning

Import the dataset & Select columns

```r
fema = read.csv("clean_data.csv")
```

```r
fema$state <- tolower(fema$state)
fema$county <- tolower(fema$county)
fema %<>% select(disasterNumber,date,damageCategory,projectSize,
                 state,county, projectAmount,federalShareObligated)
names(fema) <- c("hurricane_id","date","damage_category","project_size",
                 "region", "subregion", "project_amount", "federal_share")
```

Name each hurricane

Based on the information from the FEMA website, we

```r
#year18<-fema %>% filter(year(date)==2018)
#year18$hurricane_id %>% unique()
#2009
IDA <- fema %>% filter(hurricane_id == 1866) %>%
  mutate(hurr_name = ’IDA’)
```

```
#2010
EARL <- fema %>% filter(hurricane_id %in% c(1939,3314,3315))%>%
  mutate(hurr_name = ’EARL’)
ALEX <- fema %>% filter(hurricane_id == 1931) %>%
  mutate(hurr_name = ’ALEX’)
#2011
IRENE <- fema %>% filter(year(date) == 2011) %>%
  mutate(hurr_name = ’IRENE’)
#2012-2013
SANDY <- fema %>% filter((year(date) == 2012) | (year(date) == 2013)) %>%
  mutate(hurr_name = ’SANDY’)
#2016
MATTHEW <- fema %>%
  filter(hurricane_id %in% c(4291,4286,4285,4283,4284)) %>%
  mutate(hurr_name = ’MATTHEW’)
HERMINE <- fema %>% filter(hurricane_id == 4280) %>%
  mutate(hurr_name = ’HERMINE’)
#2017
HARVEY <- fema %>% filter(hurricane_id %in% c(4332,4345)) %>%
  mutate(hurr_name = ’HARVEY’)
NATE <- fema %>% filter(hurricane_id %in% c(3392,3395,4349,4350)) %>%
  mutate(hurr_name = ’NATE’)
IRMA <- fema %>%
  filter(hurricane_id %in% c(3384,3389,4336,4337,4338,4346,4335,4341)) %>%
  mutate(hurr_name = ’IRMA’)
MARIA <- fema %>% filter(hurricane_id %in% c(4340,4339)) %>%
  mutate(hurr_name = ’MARIA’)
#2018
MICHAEL <- fema %>% filter(hurricane_id %in% c(4411,4406,4400,4399)) %>%
  mutate(hurr_name = ’MICHAEL’)
FLORENCE <- fema %>% filter(hurricane_id %in% c(4393,4394,4401)) %>%
  mutate(hurr_name = ’FLORENCE’)
LANE_hawaii <- fema %>% filter(hurricane_id == 4395) %>%
  mutate(hurr_name = ’LANE’)
GITA <- fema %>% filter(hurricane_id == 4357) %>%
  mutate(hurr_name = ’GITA’)
```

Combine all the hurricanes

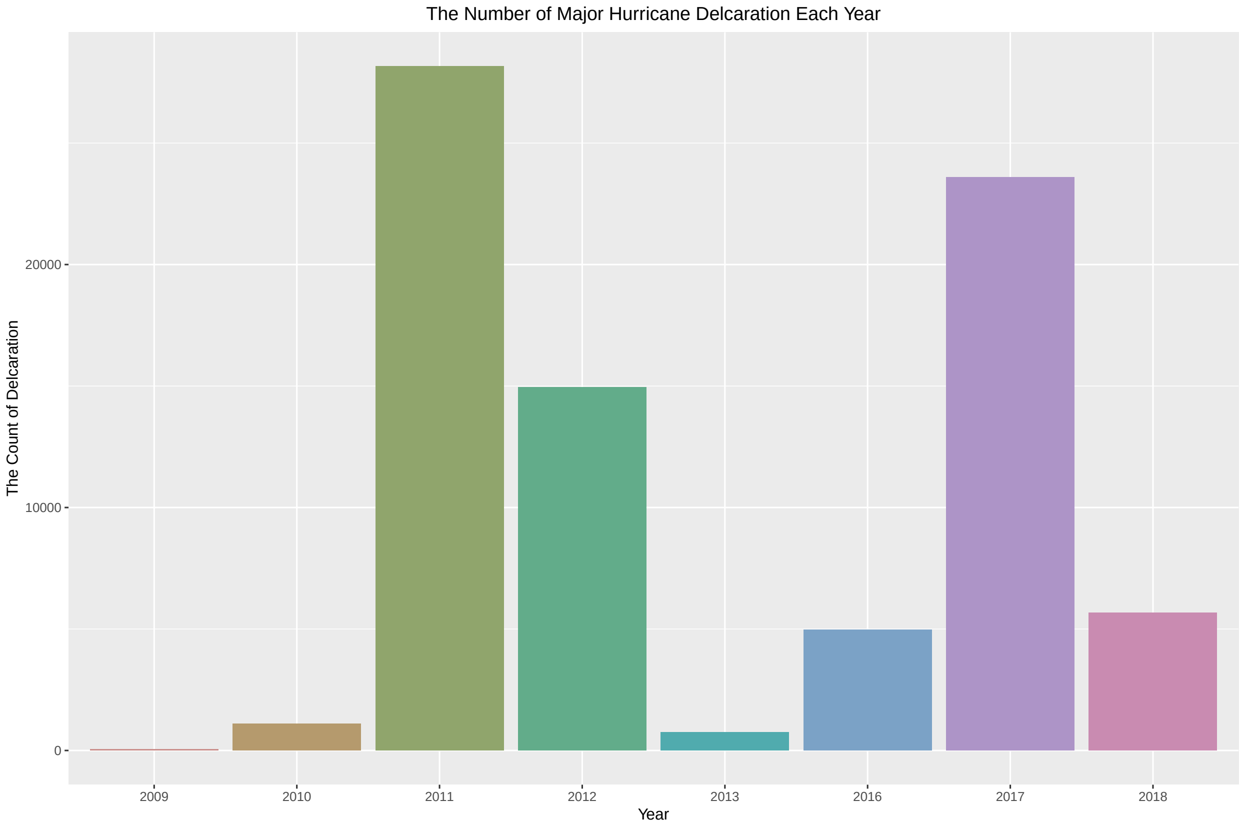
```
fema_new <- do.call("rbind", list(IDA,EARL,ALEX,IRENE,SANDY,MATTHEW,
                                  HERMINE,HARVEY,NATE,IRMA,MARIA,MICHAEL,
                                  FLORENCE,LANE_hawaii,GITA))
#write.csv(fema_new, "Shiny/shiny_data.csv")
#write.csv(fema_new,"slides_data.csv")
```

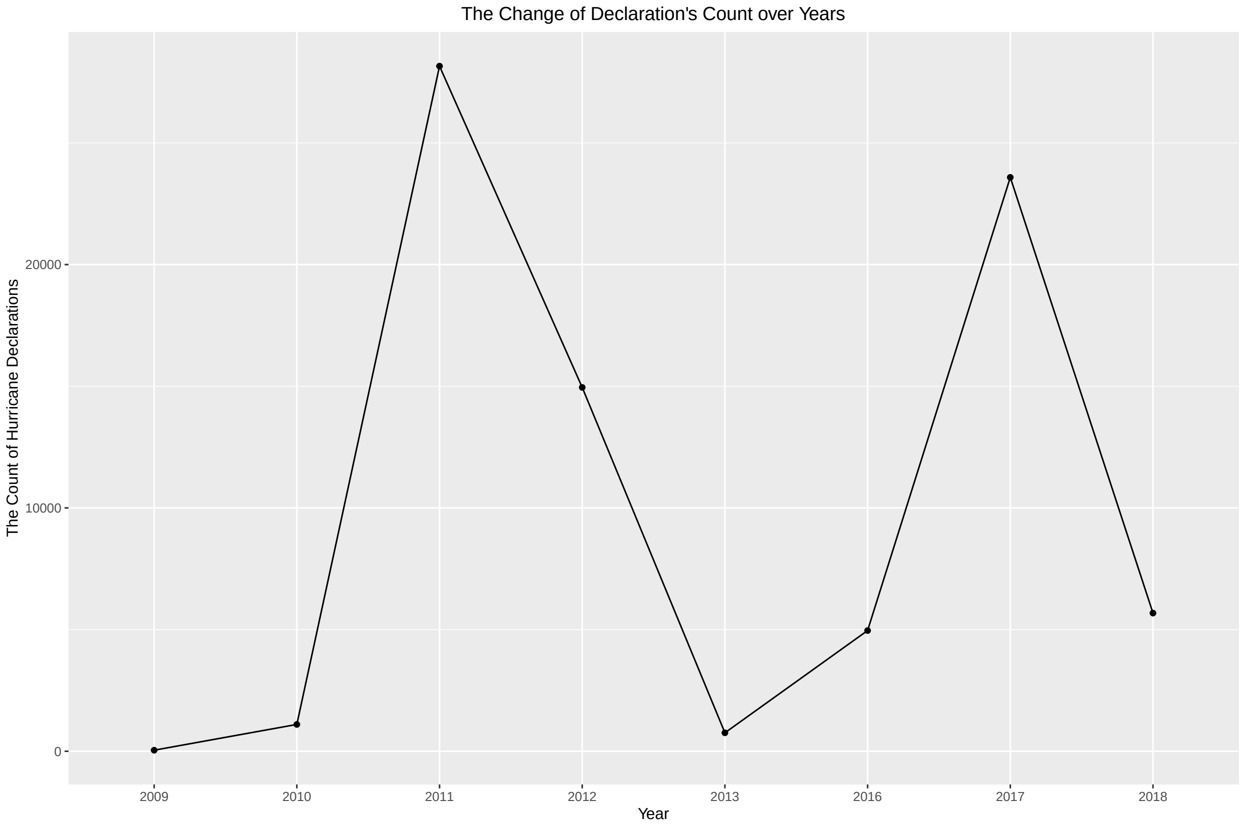
3. Exploratory Data Analysis

3.1 Temporal Aspect

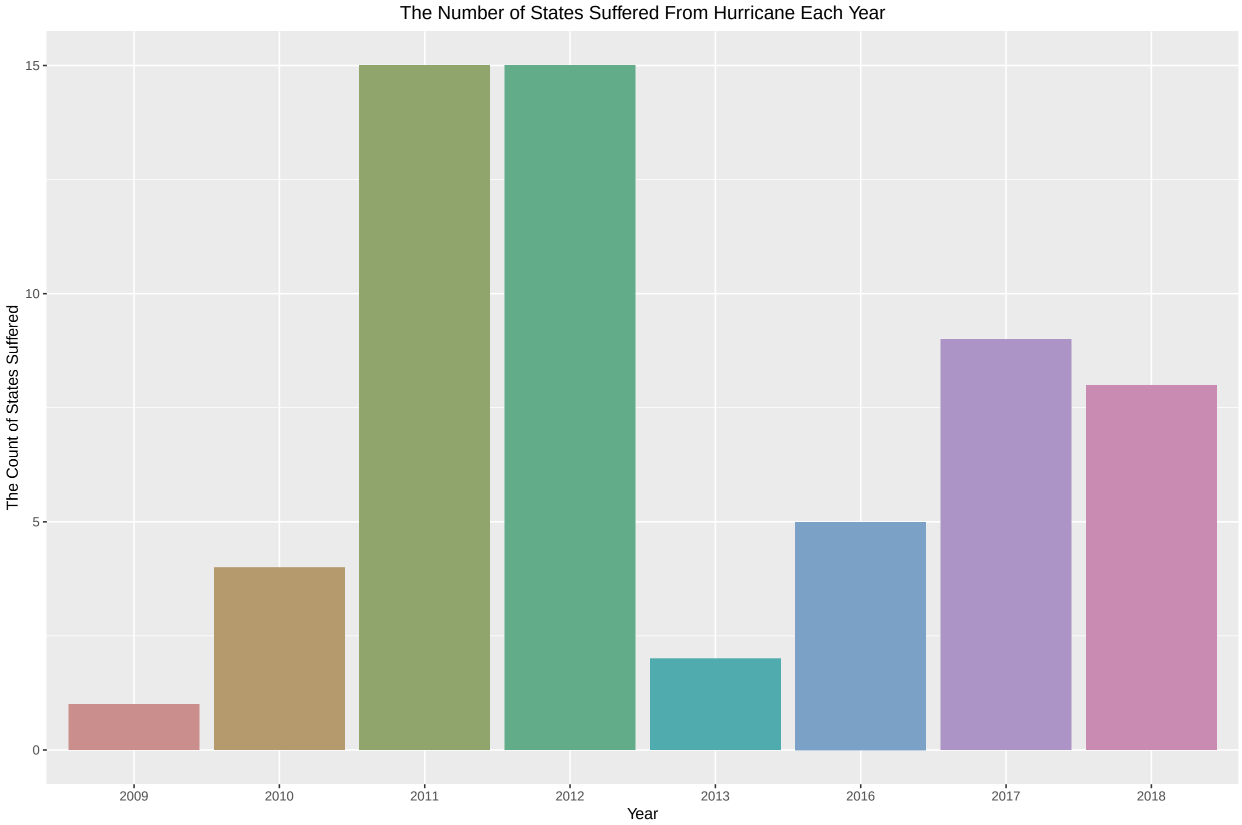
First, we explore our data from the aspect of time. We draw multiple plots based on the data of different years to reveal various information.

The number of major disaster declaration each year

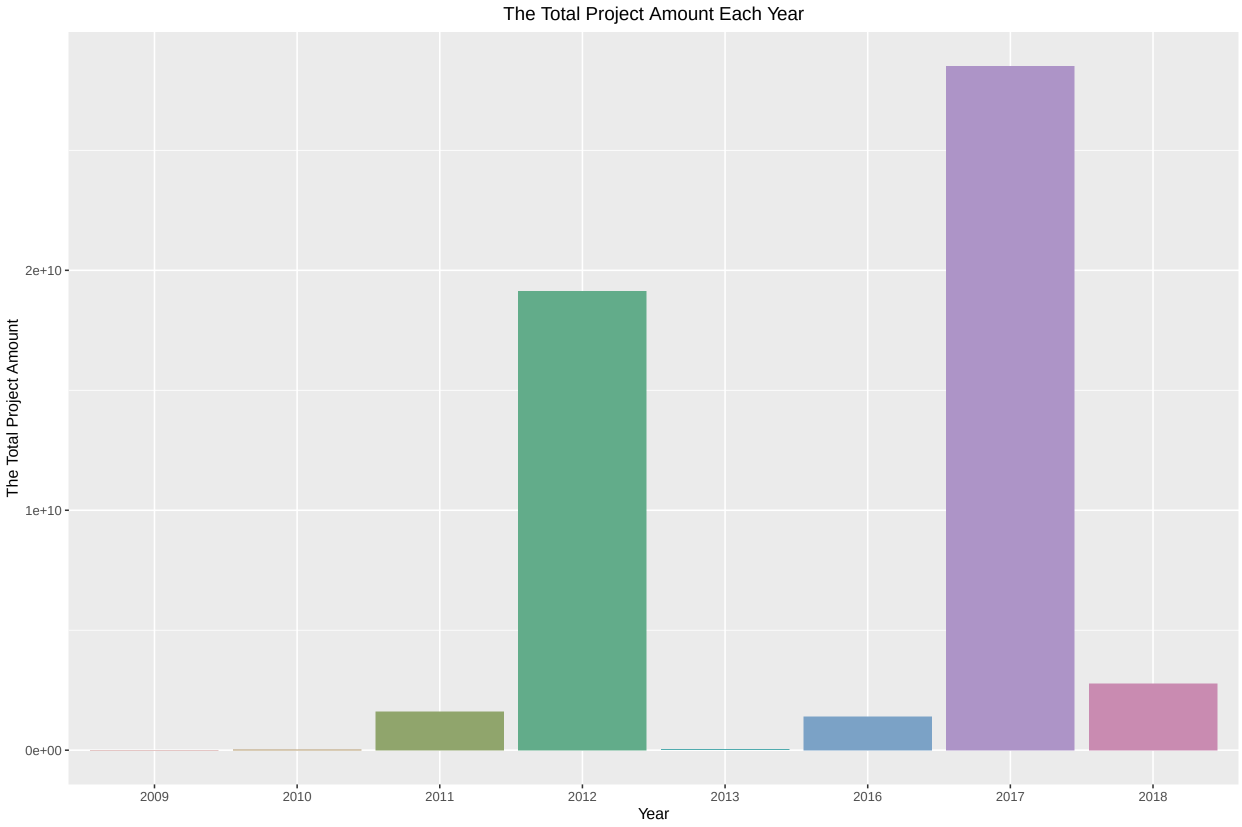


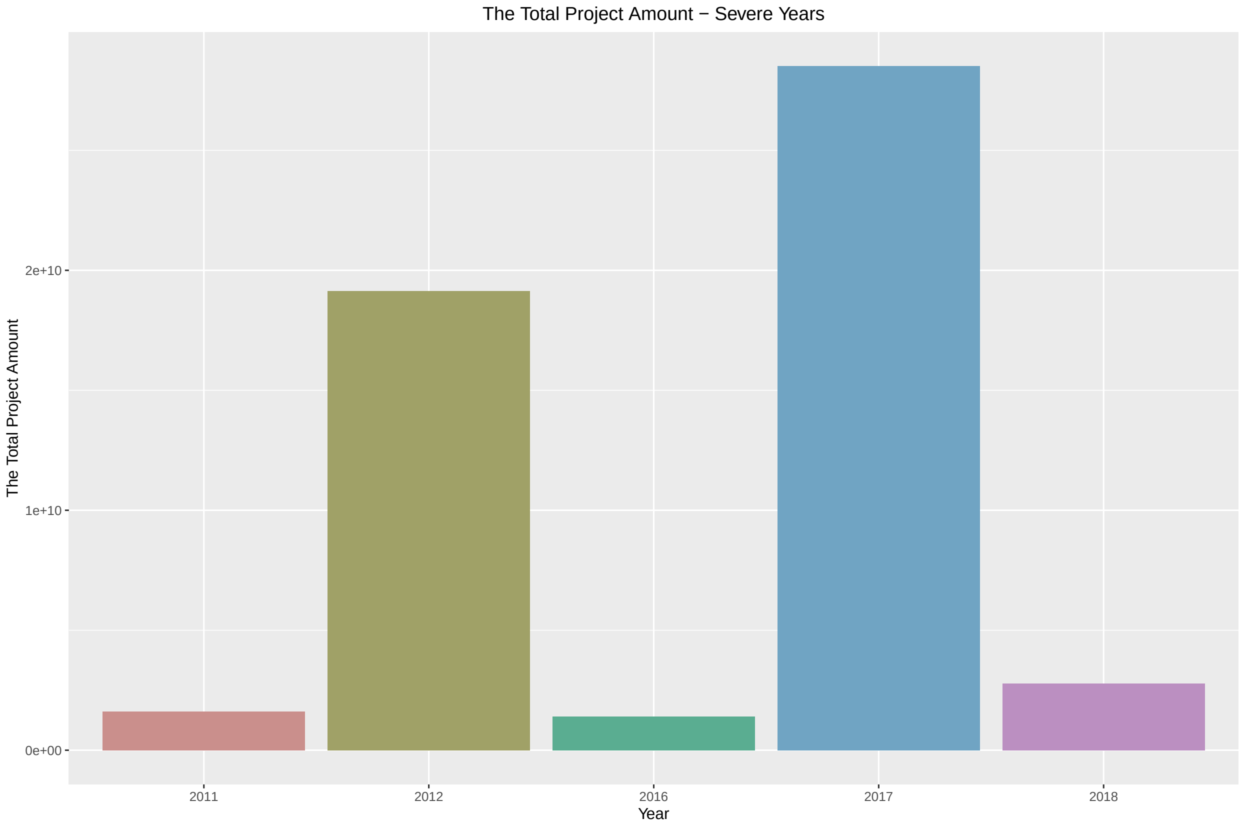

The Change of Declaration's Count over Years
The Count of Hurricane Declarations
20000
10000
0
2009
2010
2011
2012
2013
2016
2017
2018
Year

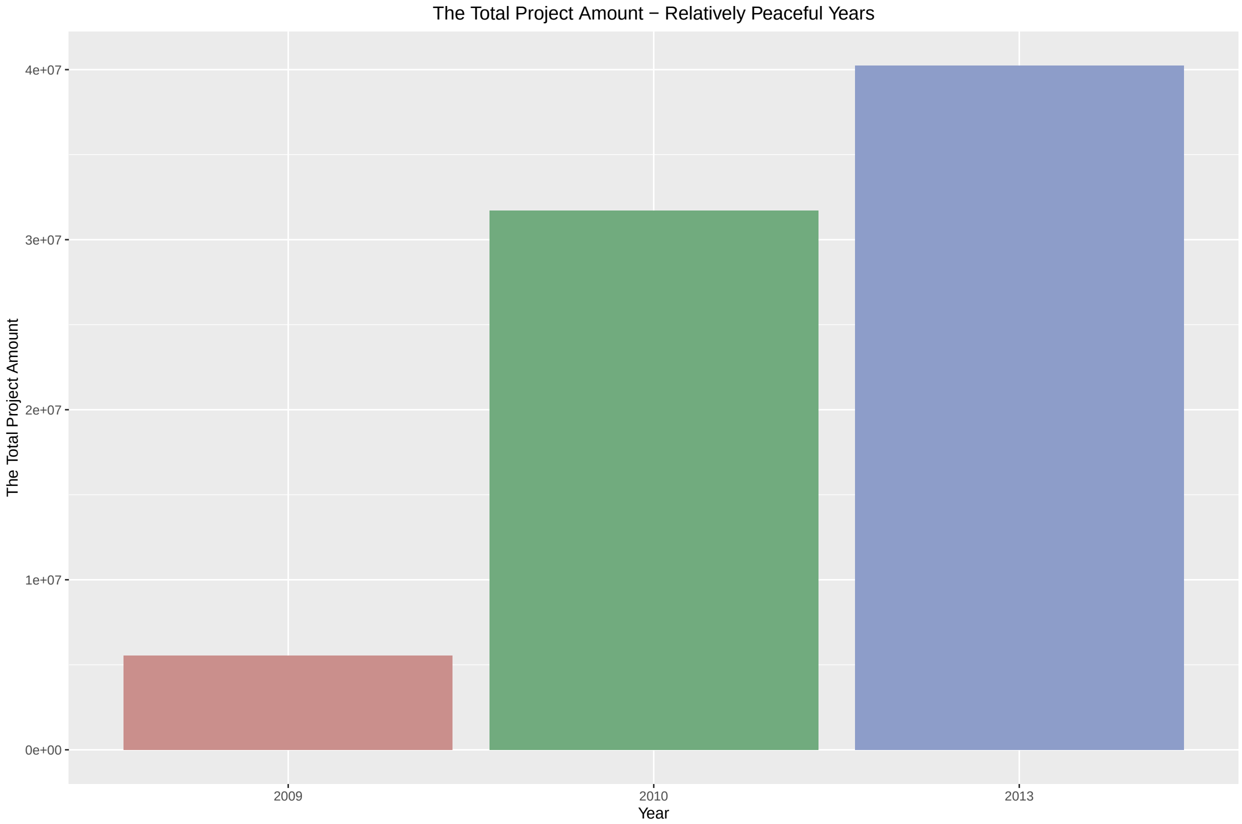
The Number of States Suffered From Hurricane Each Year

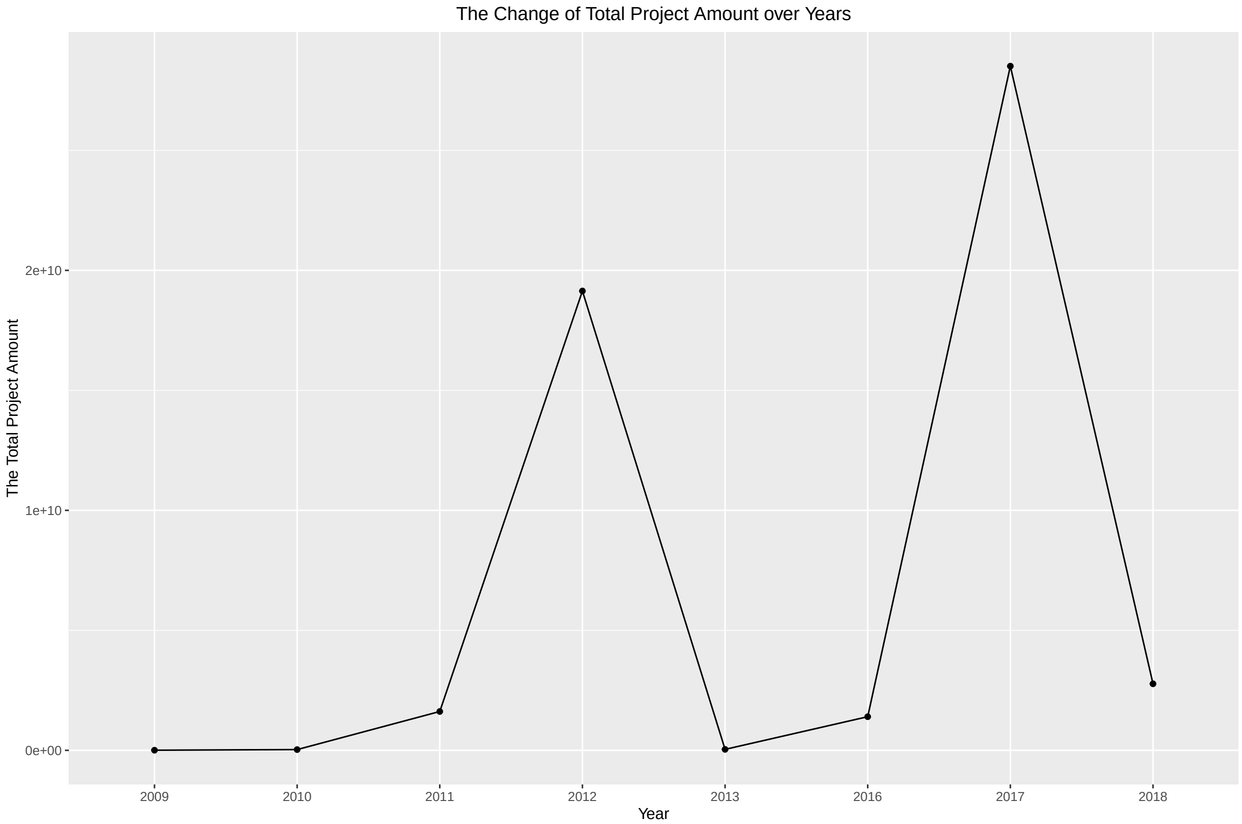


The Total Project Amount Each Year



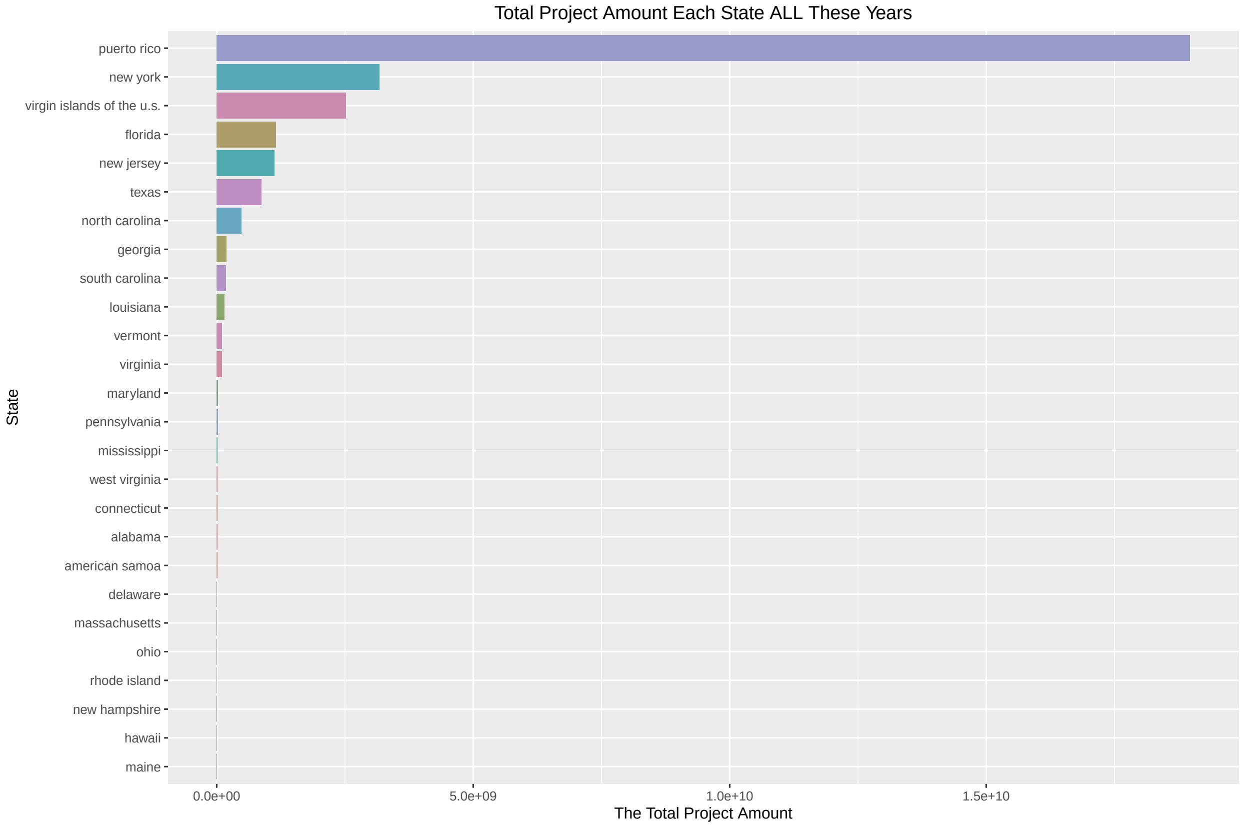

The Total Project Amount − Severe Years
The Total Project Amount
2e+10
1e+10
0e+00
2011
2012
2016
2017
2018
Year

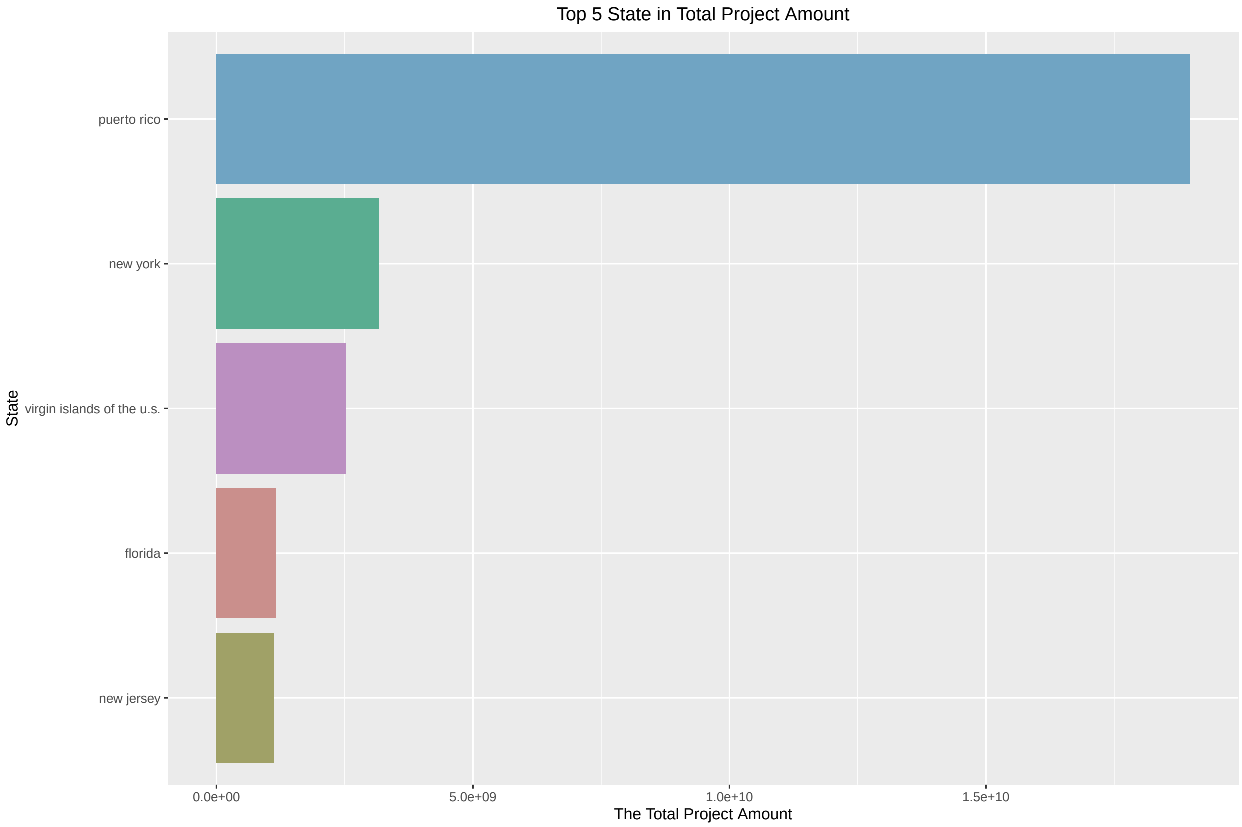

The Total Project Amount − Relatively Peaceful Years
The Total Project Amount
4e+07
3e+07
2e+07
1e+07
0e+00
2009
2010
2013
Year


The Change of Total Project Amount over Years
The Total Project Amount
2e+10
1e+10
0e+00
2009
2010
2011
2012
2013
2016
2017
2018
Year

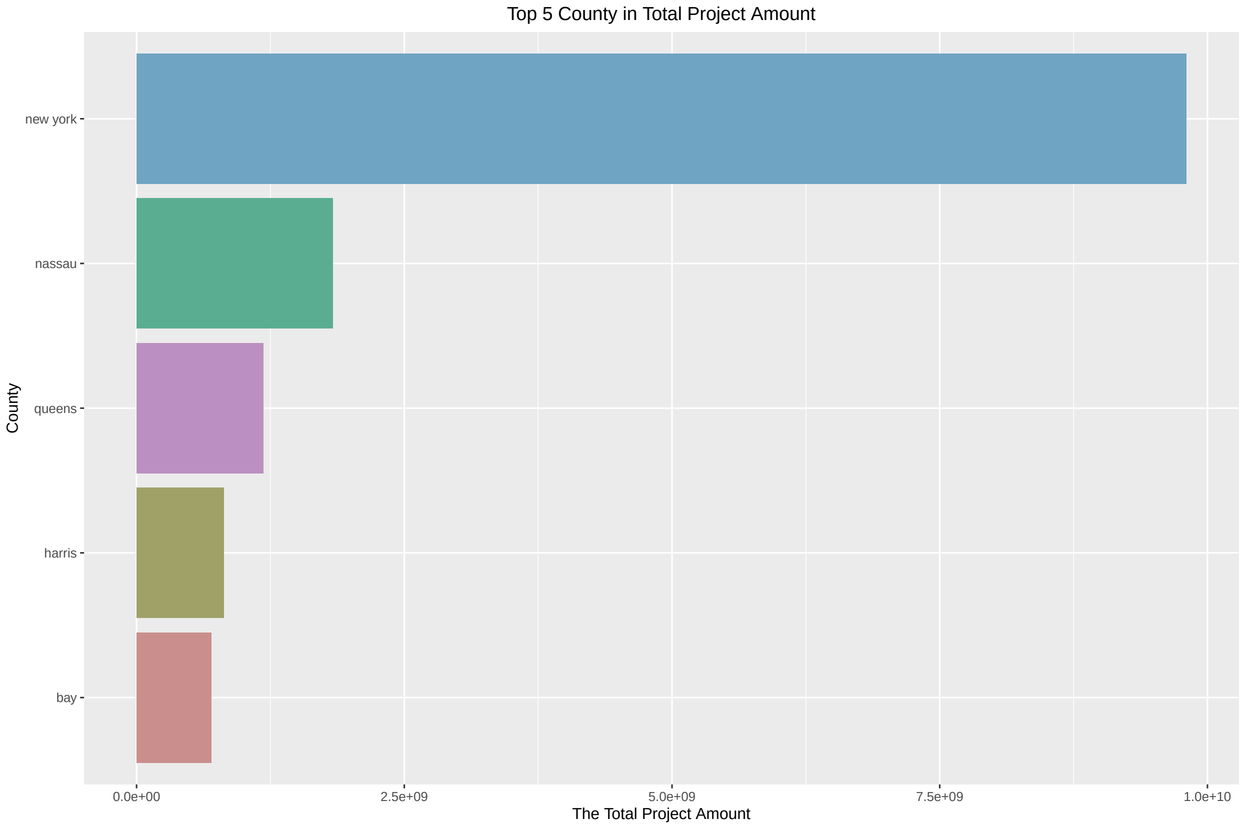
3.2 Spatial Aspect

The Total Project Amount Each State ALL These Years



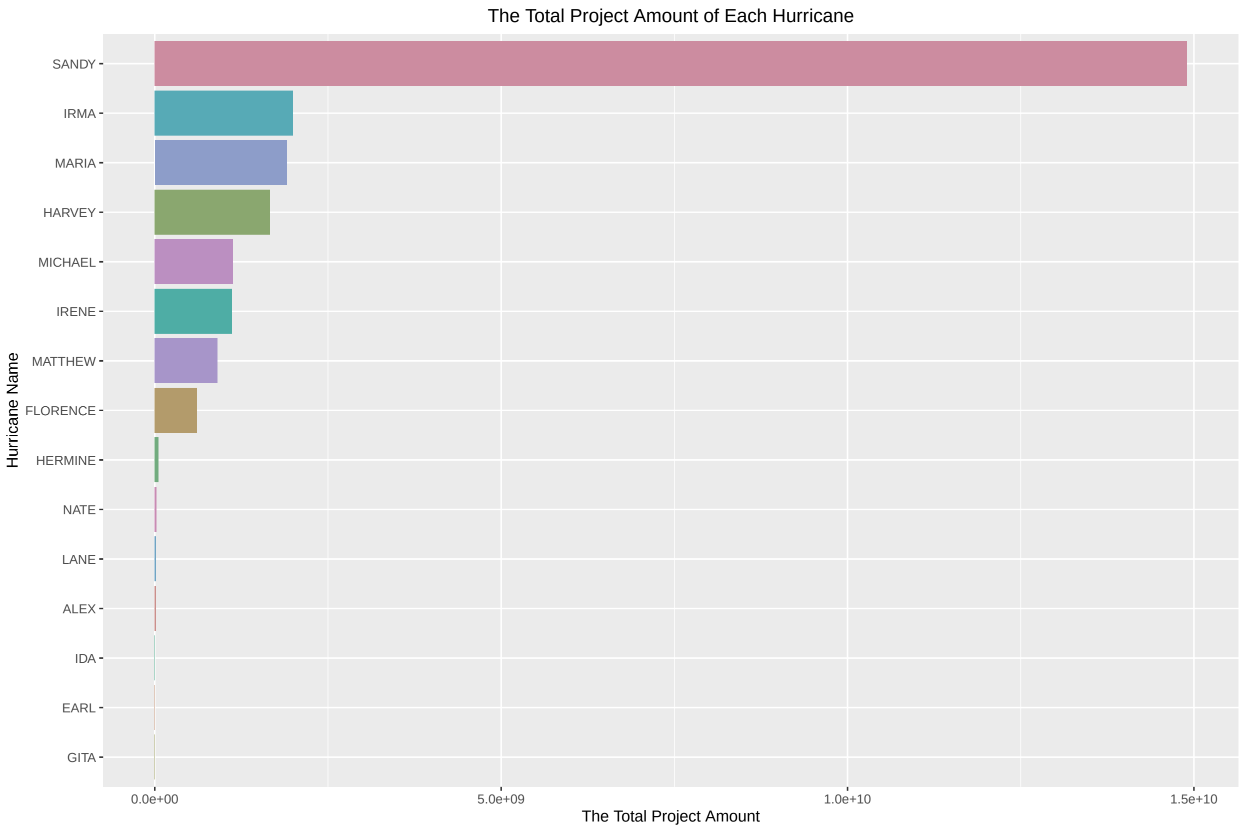

Top 5 State in Total Project Amount
State
puerto rico
new york
virgin islands of the u.s.
florida
new jersey
0.0e+00
5.0e+09
1.0e+10
1.5e+10
The Total Project Amount

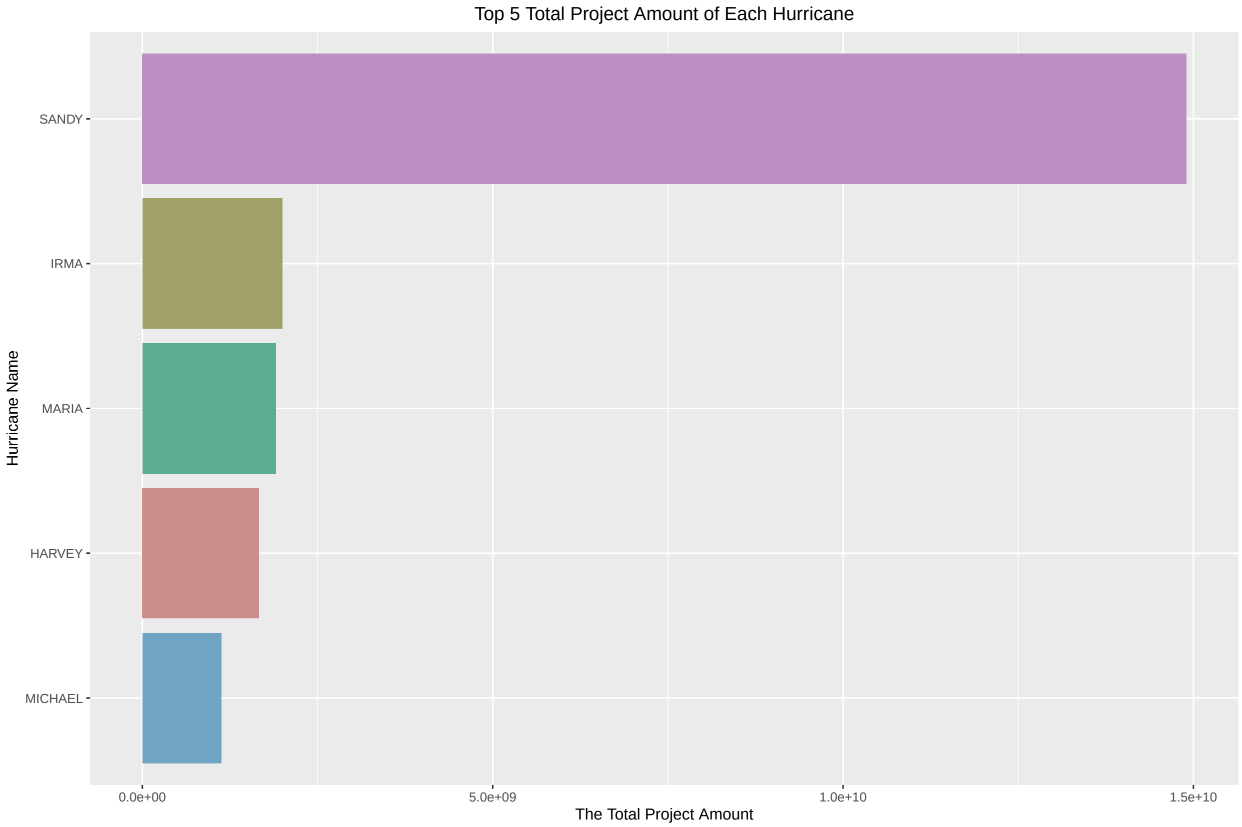
The Top 5 County in Total Project Amount



3.3 Hurriance Aspect

The Total Project Amount of Each Hurricane





3.4 project size aspect

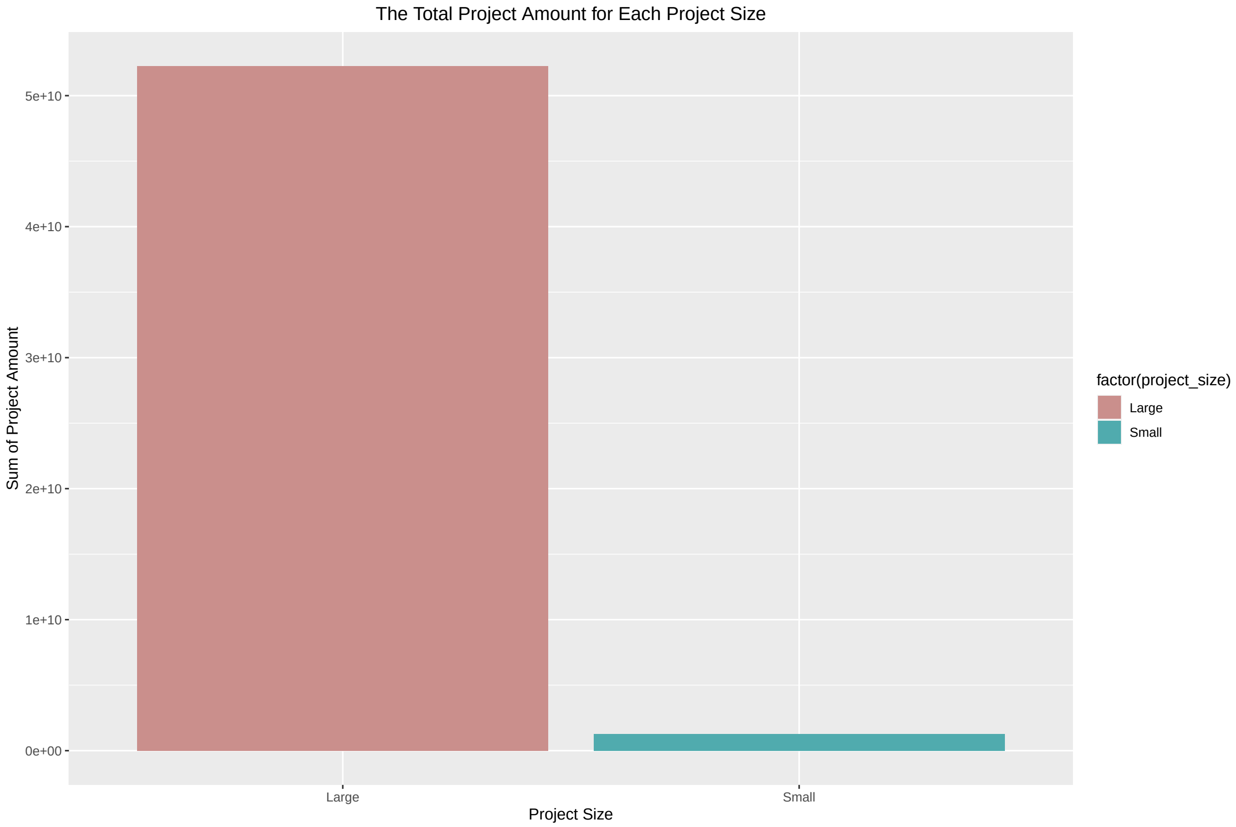
We have two categories of project size: Large and Small.

```
size_amount <- fema_new %>%
  group_by(project_size) %>%
  summarize(sum = sum(project_amount),.groups='drop')

size_count <- fema_new %>%
  group_by(project_size) %>%
  summarize(count = n(),.groups='drop')
```

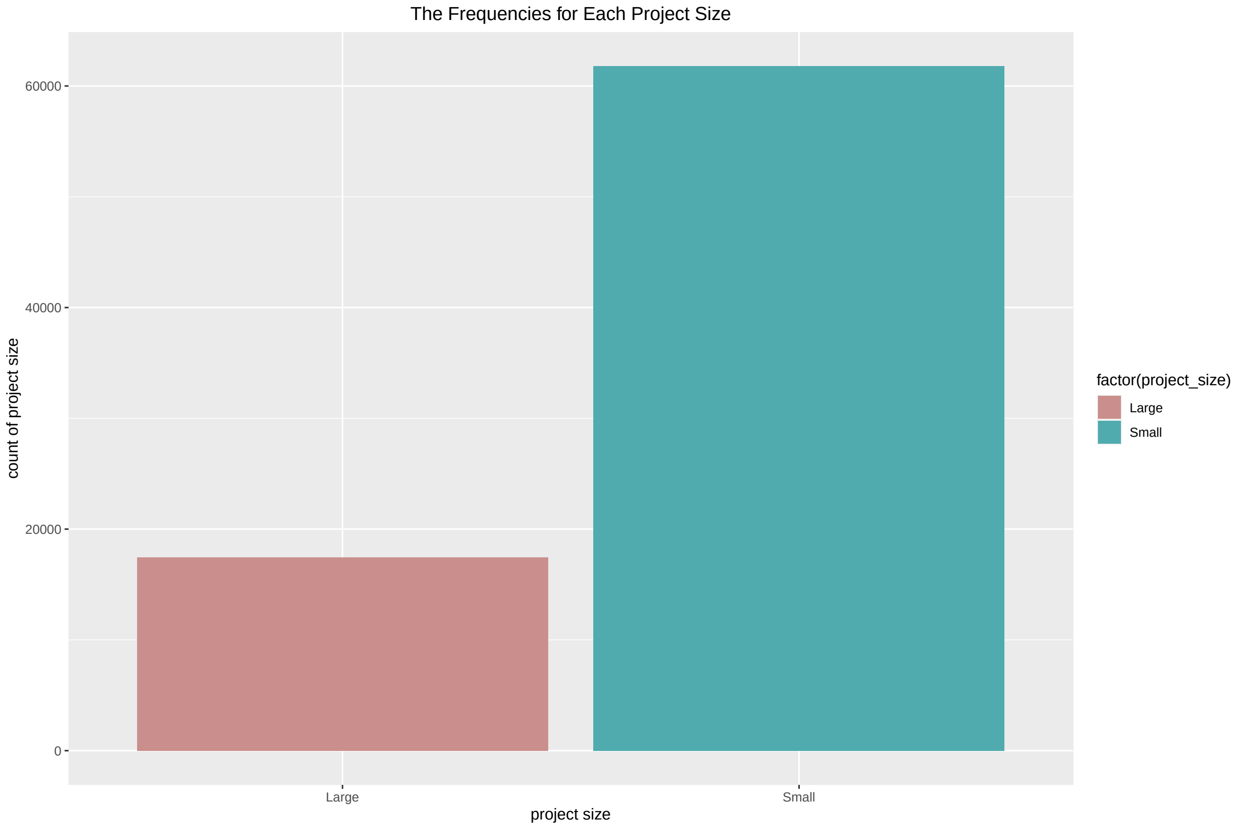
The Total Project Amount for Each Project Size

```
ggplot(data = size_amount,
       aes(x = factor(project_size), y = sum,
           fill = factor(project_size))) +
  geom_bar(stat = "identity") +
  scale_fill_hue(c = 40) +
  xlab("Project Size") +
  ylab("Sum of Project Amount") +
  ggtitle("The Total Project Amount for Each Project Size") +
  theme(plot.title = element_text(hjust = 0.5))
```



The Frequencies for Each Project Size

```
ggplot(data = size_count,
       aes(x = factor(project_size), y = count,
           fill = factor(project_size))) +
  geom_bar(stat = "identity") +
  scale_fill_hue(c = 40) +
  xlab("project size") +
  ylab("count of project size") +
  ggtitle("The Frequencies for Each Project Size") +
  theme(plot.title = element_text(hjust = 0.5))
```



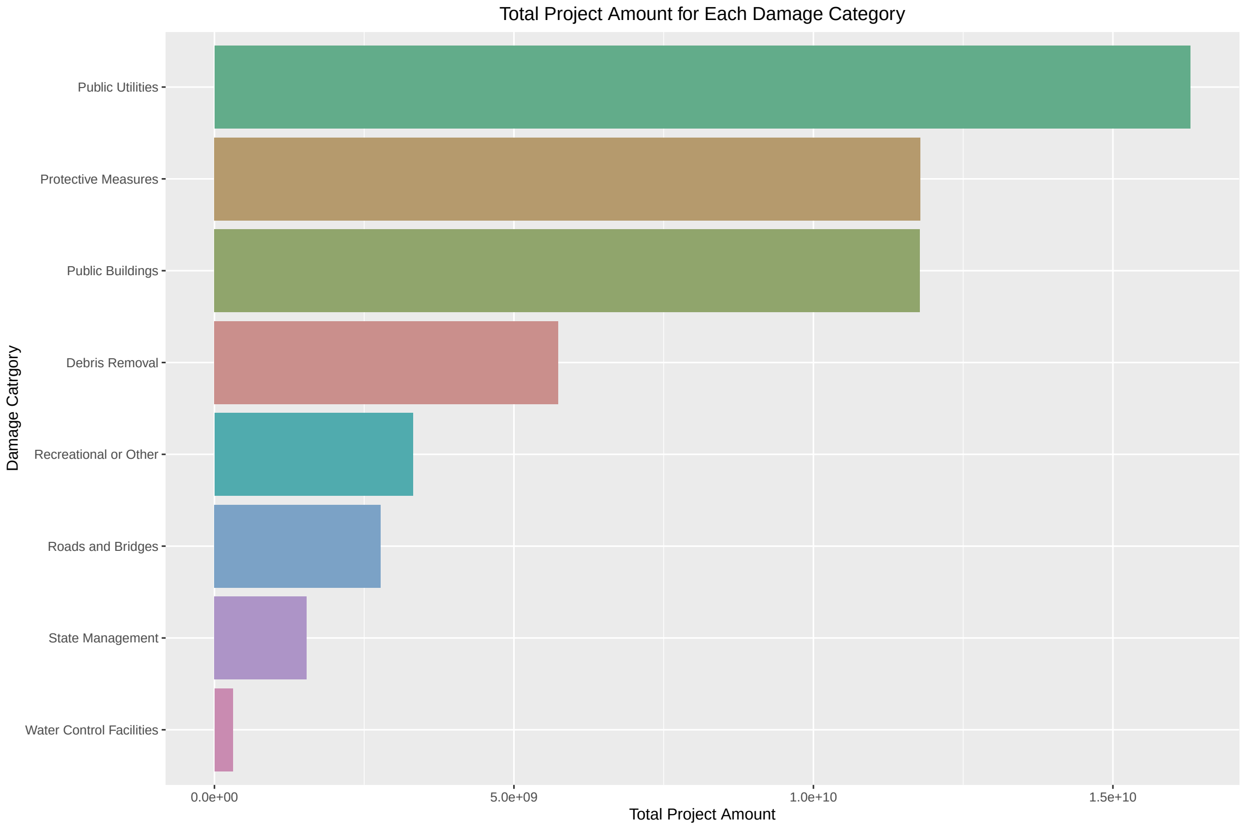
3.5 Damage Category Aspect

```
dcc_amount <- fema_new %>%
  group_by(damage_category) %>%
  summarize(sum = sum(project_amount),.groups=’drop’)

dcc_count <- fema_new %>%
  group_by(damage_category) %>%
  summarize(count = n(),.groups=’drop’)
```

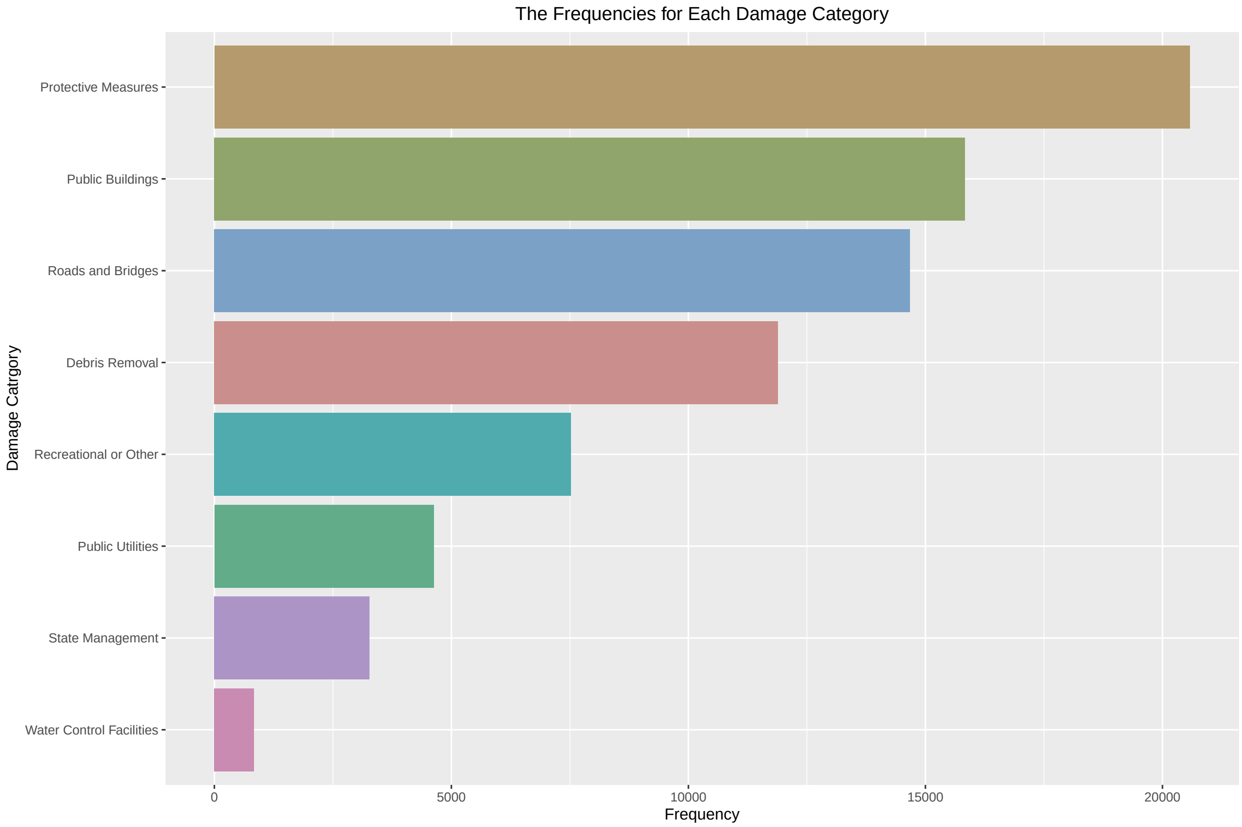
Total Project Amount for Each Damage Category

```
ggplot(data = dcc_amount,
       aes(x = reorder(damage_category,sum), y = sum,
           fill = as.factor(damage_category))) +
  geom_bar(stat = "identity") +
  scale_fill_hue(c = 40) +
  xlab("Damage Catrgory") +
  ylab("Total Project Amount") +
  ggtitle("Total Project Amount for Each Damage Category") +
  theme(plot.title = element_text(hjust = 0.5),
        legend.position = "none") +
  coord_flip()
```



The Frequencies for Each Damage Category

```
ggplot(data = dcc_count,
       aes(x = reorder(damage_category,count), y = count,
           fill = as.factor(damage_category))) +
  geom_bar(stat = "identity") +
  scale_fill_hue(c = 40) +
  xlab("Damage Catrgory") +
  ylab("Frequency") +
  ggtitle("The Frequencies for Each Damage Category") +
  theme(plot.title = element_text(hjust = 0.5),
        legend.position = "none") +
  coord_flip()
```



4. Ploting the maps

First, let's check whether the dataset we used to make the maps contains all our regions in the fema dataset.

```
all <- map_data("county")
a <- unique(all$region)
b <- unique(fema_new$region)
b[!(b %in% a)]
```

```
## [1] "virgin islands of the u.s." "puerto rico"
## [3] "hawaii"                     "american samoa"
```

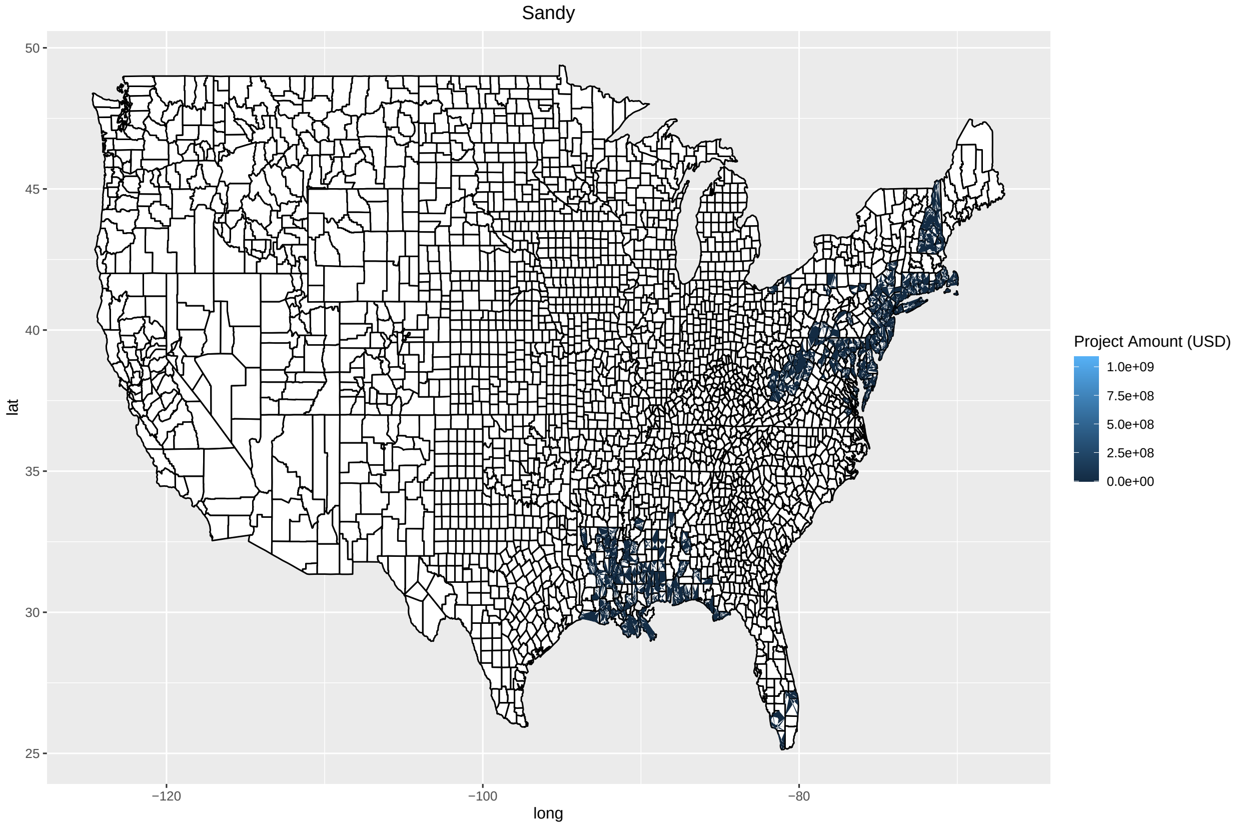
We notice that there are four regions not included in the map data.

Then, we load the data of longitude and latitude, and join them with our data

```
sandy_map <- merge(all,SANDY,by=c("region","subregion"))
irma_map <- merge(all,IRMA,by=c("region","subregion"))
#maria_map <- merge(all,MARIA,by=c("region","subregion"))
harvey_map <- merge(all,HARVEY,by=c("region","subregion"))
michael_map <- merge(all,MICHAEL,by=c("region","subregion"))
```

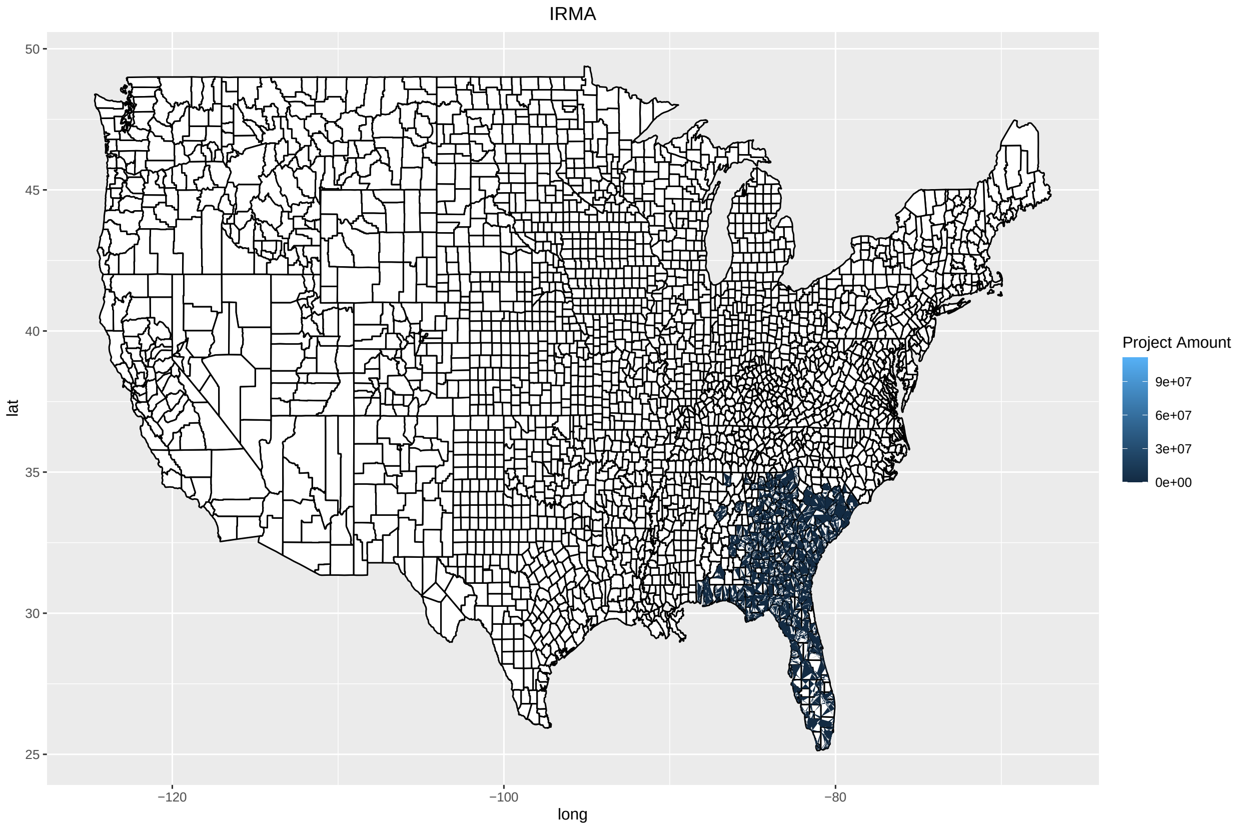
Make map plots of the project amount of Hurricane Sandy

```
sandyPlot <- ggplot() +
  geom_polygon(data=all,
               aes(x=long, y=lat, group=group),colour="black",fill="white")+
  geom_polygon(data=sandy_map,
               aes(x = long, y = lat, group = group, fill=project_amount))+
  labs(fill="Project Amount (USD)") +
  ggtitle("Sandy") +
  theme(plot.title = element_text(hjust = 0.5))
sandyPlot
```



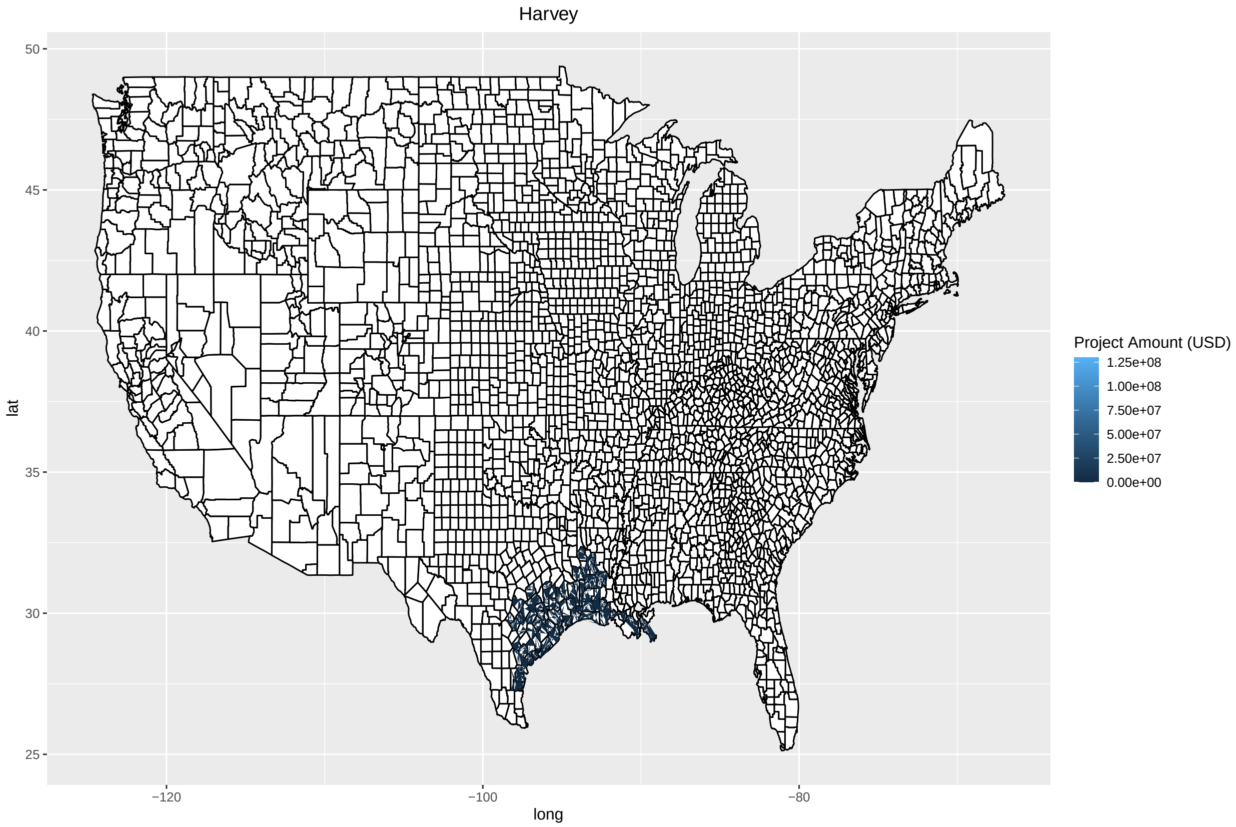
Make map plots of the project amount of Hurricane IRMA

```
irmaPlot <- ggplot() +
  geom_polygon(data=all,
               aes(x=long, y=lat, group=group),colour="black",fill="white")+
  geom_polygon(data=irma_map,
               aes(x = long, y = lat, group = group, fill=project_amount))+
  labs(fill="Project Amount") +
  ggtitle("IRMA") +
  theme(plot.title = element_text(hjust = 0.5))
irmaPlot
```



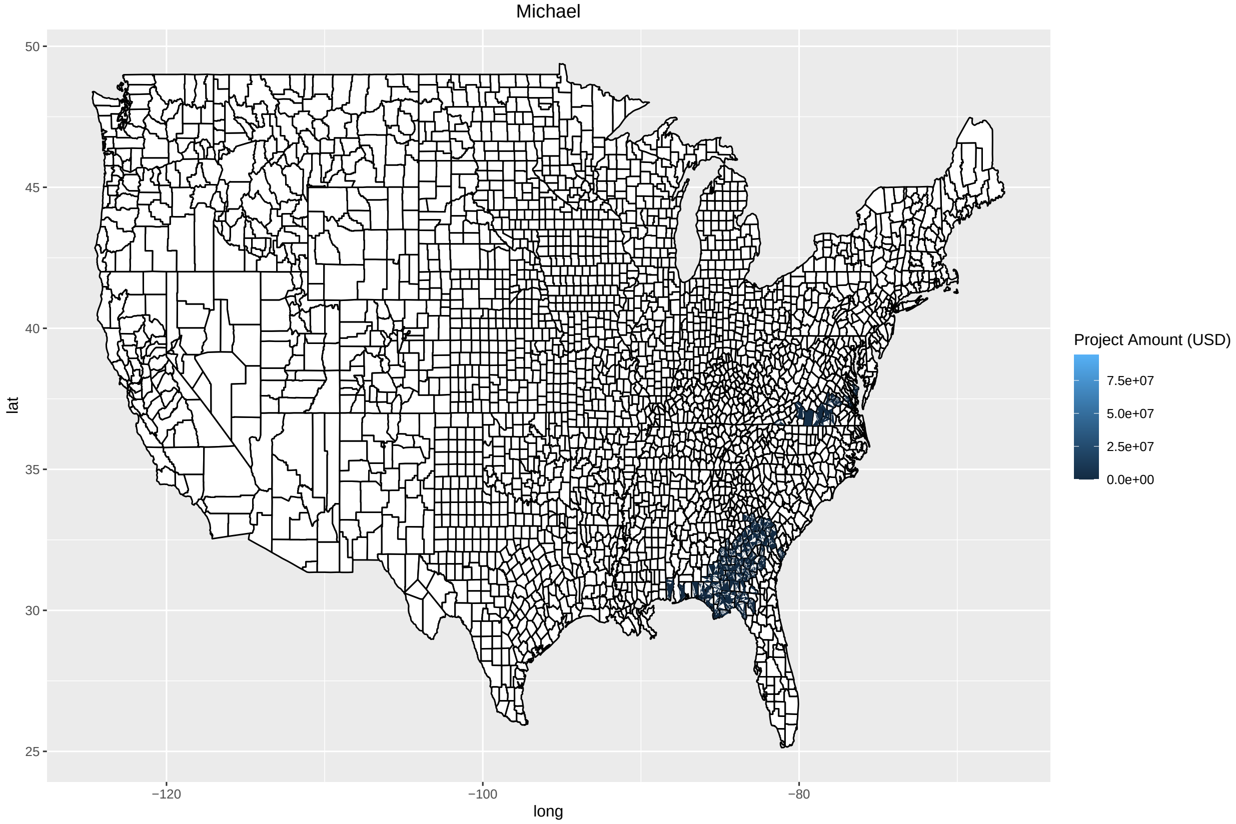
Make map plots of the project amount of Hurricane Harvey

```
harveyPlot <- ggplot() +
  geom_polygon(data=all,
               aes(x=long, y=lat, group=group),colour="black",fill="white")+
  geom_polygon(data=harvey_map,
               aes(x = long, y = lat, group = group, fill=project_amount))+
  labs(fill="Project Amount (USD)") +
  ggtitle("Harvey") +
  theme(plot.title = element_text(hjust = 0.5))
harveyPlot
```



Make map plots of the project amount of Hurricane Michael

```
michaelPlot <- ggplot() +
  geom_polygon(data=all,
               aes(x=long, y=lat, group=group),colour="black",fill="white")+
  geom_polygon(data=michael_map,
               aes(x = long, y = lat, group = group, fill=project_amount))+
  labs(fill="Project Amount (USD)") +
  ggtitle("Michael") +
  theme(plot.title = element_text(hjust = 0.5))
michaelPlot
```



5. Acknowledgements $ References

The data is collected from: https://www.fema.gov/openfema-data-page/public-assistance-funded-projects-details-v1

1. Yihui Xie (2020). knitr: A General-Purpose Package for Dynamic Report Generation in R. R package version 1.29. https://github.com/yihui/knitr
2. Hadley Wickham (2019). tidyverse: Easily Install and Load the 'Tidyverse'. R package version 1.3.0. https://cloud.r-project.org/package=tidyverse
3. Greg Freedman Ellis, Thomas Lumley, Tomasz Żółtak and Ben Schneider (2020). srvyr:'dplyr'-Like Syntax for Summary Statistics of SUrvey Data. R package version 0.4.0. https://cloud.r-project.org/package=srvyr