Map Assignment

Group 10

Chenghao Meng, Fan Feng, Yinfeng Zhou

2020/10/25

1 Data Cleaning

To begin with, we will load the dataset from the package `hurricaneexposuredata`.

```
# Extract data for the mapping
addRepo("geanders")
data("hurr_tracks")
data("rain")
```

Then we extract the track data of **Floyd-1999** and **Allison-2001**.

```
TrackFloyd <- hurr_tracks %>% filter(storm_id=="Floyd-1999")
TrackAllison <- hurr_tracks %>% filter(storm_id=="Allison-2001")
```

After that, we will use the `filter` , `group_by` and `summarise` functions to extract rainfall data of **Floyd-1999** and **Allison-2001**.

```
RainFloydRaw <- rain %>%
  filter(storm_id=="Floyd-1999") %>%
  group_by(fips,storm_id) %>%
  summarise(sum_precip=sum(precip))

RainAllisonRaw <- rain %>%
  filter(storm_id=="Allison-2001") %>%
  group_by(fips,storm_id) %>%
  summarise(sum_precip=sum(precip))
```

Since the column of fips is not that standard, we will make some transformations on that to make all the fips numbers in a 5-digit format by using `str_pad` function in `stringr` package.

```
# Make adjustments on fips
CountyFips <- county.fips
CountyFips$fips <- as.character(CountyFips$fips)
# if fips looks like "1001" rather than "01001", transformation is needed
CountyFips$fips <- str_pad(CountyFips$fips,5,side = "left",pad = "0")
```

After that, we will use `merge` function to join the **CountyFips** data frame into the data frames of rainfall data.

```
RainFloydPre <- merge(RainFloydRaw,CountyFips,by="fips") %>% separate(polyname, into= c("region","subreg

RainAllisonPre <- merge(RainAllisonRaw,CountyFips,by="fips") %>% separate(polyname, into= c("region","su
```

To get the target map, we will use the `map` package to extract necessary dataset.

```
StatesInt <- c("texas","oklahoma","kansas","louisiana","arkansas","missouri","iowa","wisconsin","michig

MainStates <- map_data("county",StatesInt)
```

Then, we will create RainFloyd and RainAllison data frames for further analysis.

```
# Create RainFloyd
RainFloyd <- merge(MainStates,RainFloydPre,by=c("region","subregion"))
RainFloyd[RainFloyd==0.0] <- 0

# Make cut
RainFloyd$rain_cut <- cut(RainFloyd$sum_precip,breaks=c(0,25,50,75,100,125,150,175,200,225),ordered_resu
```

```
# Create RainAllison
RainAllison <- merge(MainStates,RainAllisonPre,by=c("region","subregion"))
RainAllison[RainAllison==0.0] <- 0

# Make the cut
RainAllison$rain_cut <- ifelse(RainAllison$sum_precip>175,"Exposed","Unexposed")
```

Before moving on, we should test if the created colums contain NAs for the convenience of mapping.

```
# NA test
anyNA(RainFloyd$rain_cut)
```

```
## [1] FALSE
```

```
anyNA(RainAllison$rain_cut)
```

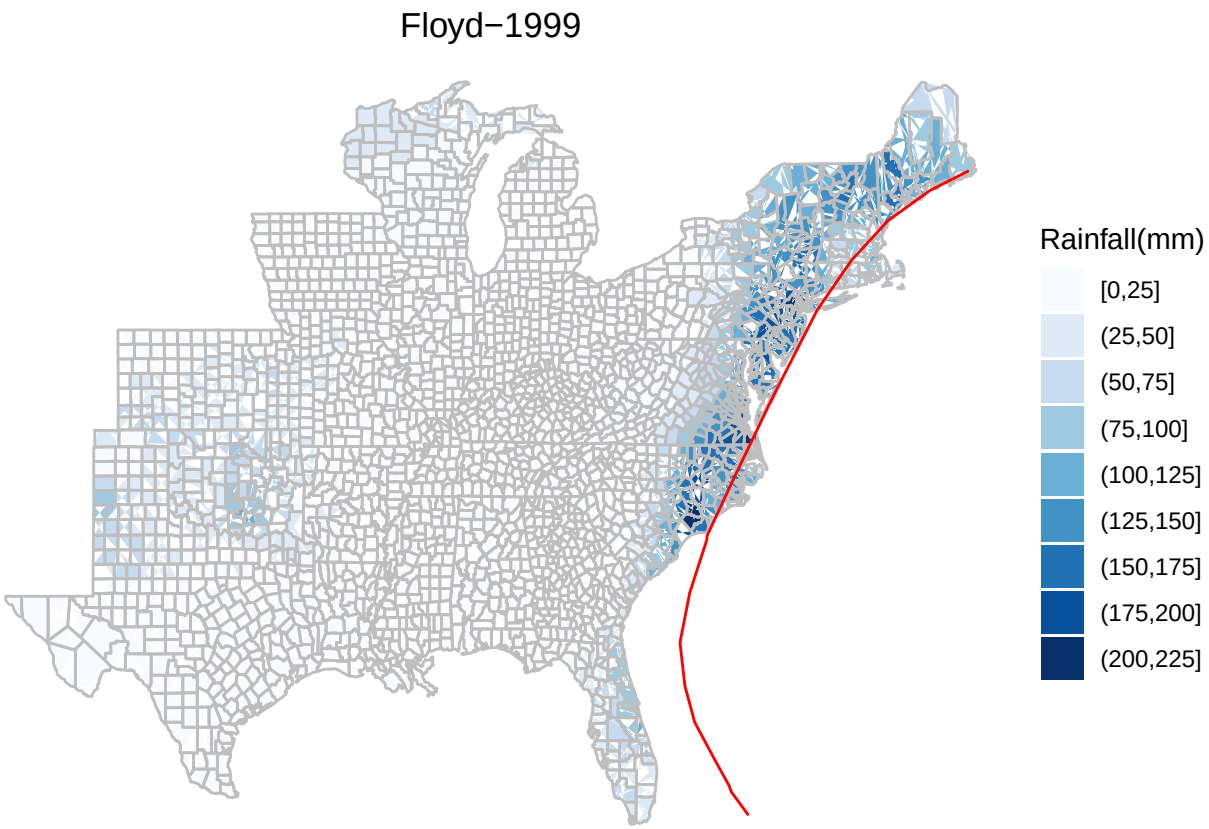
```
## [1] FALSE
```

2 Map: Using ggplot

2.1 ggplot: Floyd-1999

```
RainFloydPlot <- ggplot() +
  geom_polygon(data=RainFloyd,aes(x=long,y=lat,group=group,fill=rain_cut))+

  geom_path(data=MainStates,mapping=aes(long,lat,group=group),color="grey")+

  geom_path(data=TrackFloyd,aes(longitude, latitude),color="red")+

  xlim(min(MainStates$long),max(MainStates$long)) +
  ylim(min(MainStates$lat),max(MainStates$lat))
```

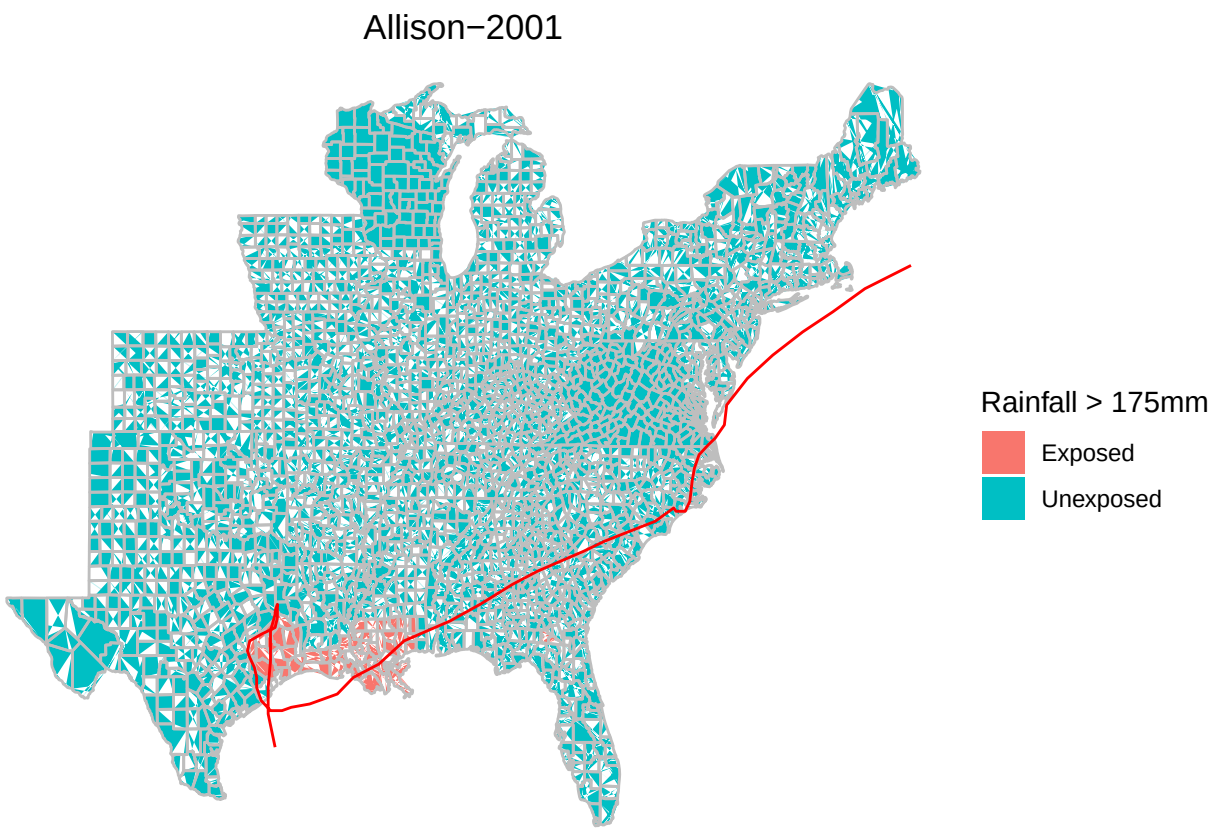
```
RainFloydPlot +
  # Change the title of the legend
  labs(fill="Rainfall(mm)") +
  xlab("Longitude") + ylab("Latitude") +
  ggtitle("Floyd-1999") +
  scale_fill_brewer(palette="Blues") +
  theme_void() +
  theme(plot.title = element_text(hjust = 0.5))
```



2.2 ggplot: Allison-2001

```
RainAllisonPlot <- ggplot() +
  #geom_polygon(data=MainStates, aes(x=long, y=lat, group=group),colour="black",fill="white") +

  geom_polygon(data=RainAllison,aes(x=long,y=lat,group=group,fill=rain_cut),colour="transparent")+

  geom_path(data=MainStates,mapping=aes(long,lat,group=group),color="grey")+

  geom_path(data=TrackAllison,aes(longitude, latitude),color="red")+

  xlim(min(MainStates$long),max(MainStates$long)) +
  ylim(min(MainStates$lat),max(MainStates$lat))
```

```
RainAllisonPlot +
  # Change the title of the legend
  labs(fill="Rainfall > 175mm") +
  xlab("Longitude") + ylab("Latitude") +
  ggtitle("Allison-2001") +
  theme_void() +
  theme(plot.title = element_text(hjust = 0.5))
```



3 Map: Using tmap

3.1 tmap: Floyd-1999

To use `tmap` package, we must transform the data into spatial version by using `sf :: st_as_sf` function.

```
# Spatial transformation
tMap <- st_as_sf(map("county",StatesInt,plot=F,fill=T))
```

In order to visualize the rainfall with `tmap`, we need to join the **RainFloyd** data frame to **tMap**, so that we can have a data frame in spatial format.

```
# RainFloyd in spatial foramt
tRainFloydPre <- RainFloyd %>%
  select(region,subregion,rain_cut) %>%
  mutate(ID=str_c(region,subregion,sep = ",")) %>%
```

```
  select(ID,rain_cut) %>%
  rename(`Rainfall(mm)`=rain_cut)

tRainFloyd <- left_join(tMap,tRainFloydPre,by="ID")
```

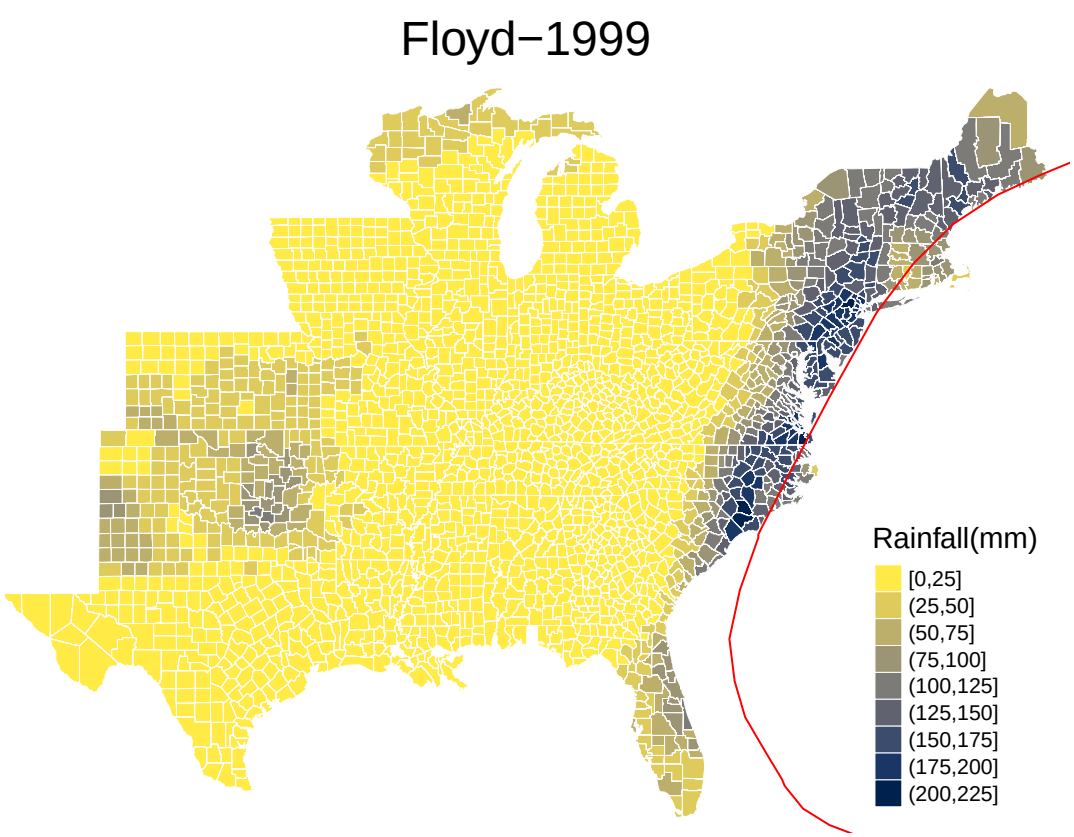
Similarly, **TrackFloyd** is also needed to be transformed into spatial lines.

```
# TrackFloyd in spatial format
tTrackFloyd=cbind(TrackFloyd$longitude,TrackFloyd$latitude)%>%
  Line() %>% Lines(ID='Floyd-1999') %>%
  list() %>% SpatialLines()
```

Now, we can have the rainfall status map of Floyd-1999 by using `tmap` package.

```
tRainFloydPlot <-
tm_shape(tRainFloyd)+
  tm_polygons(border.col="white",lwd=0.1,colorNA=NULL,
              col='Rainfall(mm)',style="cont",
              title="Rainfall(mm)",
              palette=cividis(n=7,direction=-1))+
  tm_shape(tTrackFloyd) +
  tm_lines(col='red')
```

```
# Add title
tRainFloydPlot +
  tm_layout(main.title='Floyd-1999',
            main.title.position="center",
            frame = FALSE)
```



3.2 tmap: Allison-2001

We will do the similar steps to get the rainfall data of Allison-2001 in spatial format.

```
# RainAllison in spatial format
tRainAllisonPre <- RainAllison %>%
  select(region,subregion,rain_cut) %>%
  mutate(ID=str_c(region,subregion,sep = ",")) %>%
  select(ID,rain_cut) %>%
  rename(`Rainfall > 175mm`=rain_cut)

tRainAllison <- left_join(tMap,tRainAllisonPre,by="ID")
```

```
# TrackFloyd in spatial format
tTrackAllison=cbind(TrackAllison$longitude,TrackAllison$latitude) %>%
  Line() %>% Lines(ID='Floyd-1999') %>%
  list() %>% SpatialLines()
```

Then, we can have the rainfall status map of Allison-2001.

```
tRainAllisonPlot <-
tm_shape(tRainAllison)+
  tm_polygons(border.col="white",lwd=0.1,colorNA=NULL,
              col='Rainfall > 175mm',style="cont",
```

```
              title="Rainfall > 175mm",
              palette=plasma(n=2,direction=-1))+
  tm_shape(tTrackAllison) +
  tm_lines(col='red')
```

```
# Add title
tRainAllisonPlot +
  tm_layout(main.title='Allison-2001',
            main.title.position="center",
            frame = FALSE)
```