HurricanMaps

Bruce Mallory

11/2/2020

Read in all the data

```
addRepo("geanders")
data("hurr_tracks")
data("rain")
county.longlat <- map_data("county")
data(county.fips)

floyd_track <- hurr_tracks %>% filter(storm_id=="Floyd-1999") %>% filter(longitude <= -65) %>% filter(la
floyd_rain <- rain %>% filter(storm_id=="Floyd-1999")

allison_track <- hurr_tracks %>% filter(storm_id=="Allison-2001")
allison_rain <- rain %>% filter(storm_id=="Allison-2001")

mp_states <- c("texas", "oklahoma", "kansas",
               "iowa", "missouri", "arkansas", "louisiana",
               "alabama", "mississippi", "georgia", "florida",
               "tennessee","kentucky", "indiana",
               "wisconsin", "michigan", "illinois",
               "ohio", "west virginia", "pennsylvania",
               "south carolina", "north carolina",
               "virginia", "delaware", "maryland",
               "new jersey", "district of columbia",
               "new york", "connecticut", "rhode island",
               "massachusetts", "vermont", "new hampshire",
               "maine")

states <- map_data('state', region = mp_states)
```

Calculate total rain/fip, then make rain variable categorical (9 levels for FLoyd, 2 levels for Allison)

```
floyd_rain %<>% group_by(fips) %>% summarise(rain = sum(precip), .groups="drop")
floyd_rain %<>% as.data.frame()
for (i in 1:dim(floyd_rain)[1]){
  floyd_rain$rain[i] <- floyd_rain$rain[i]%/%25
}
floyd_rain$rain <- ordered(floyd_rain$rain,labels = c("[0,25]","(25,50]","(50,75]",
```

```
                                                "(75,100]","(100,125]","(125,150]",
                                                "(150,175]","(175,200]","(200,220]"))

allison_rain %<>% group_by(fips) %>% summarise(rain = sum(precip), .groups="drop")
allison_rain %<>% as.data.frame()
for (i in 1:dim(allison_rain)[1]){
  if (allison_rain$rain[i] < 175){
    allison_rain$rain[i] <- 0
  }
  else {allison_rain$rain[i] <- 1}
}
allison_rain$rain <- ordered(allison_rain$rain, labels = c("Unexposed","Exposed"))
```

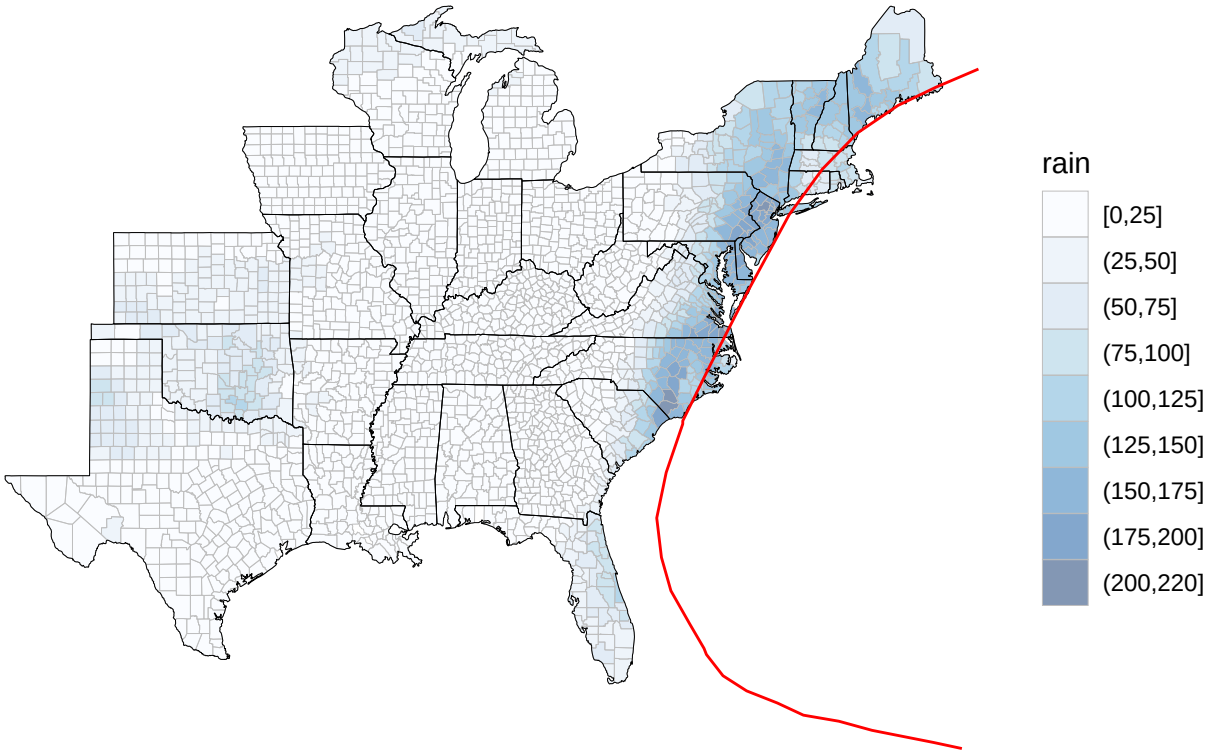
Join files to get a single file with long, lat, rain (which will be used to plot the data)

```
#first split "polyname" into "region" and "subregion" so: county.fips and county.longlat can be joined
county.fips %<>%
  separate(polyname, c("region", "subregion"), "," )
county <- left_join(county.longlat,county.fips, by = c("region", "subregion"))
#second make "fips" numeric so: "county" and "xxx_rain" can be joined
floyd_rain$fips <- as.numeric(floyd_rain$fips)
floyd_rain <- left_join(floyd_rain, county, by = "fips")
floyd_rain %<>% na.omit()
allison_rain$fips <- as.numeric(allison_rain$fips)
allison_rain <- left_join(allison_rain, county, by = "fips")
```

Now plot the data (first the rain, then the states, then the track) and format the graph

```
ggplot() +
  geom_polygon(data=floyd_rain, aes(x=long, y=lat, group=group, fill=rain), color="grey", size=0.05, al
  scale_fill_brewer(palette="Blues") +
  geom_polygon(data=states, aes(x=long, y=lat, group=group), fill="transparent", color="black", size=0.
  labs(title="Floyd-1999") +
  theme_void() +
  geom_path(data=floyd_track, aes(x=longitude, y=latitude), color="red", size=0.5)
```

Floyd−1999




```
ggplot() +
  geom_polygon(data=allison_rain, aes(x=long, y=lat, group=group, fill=rain), color="grey", size=0.05, a
  scale_fill_brewer(palette="Blues") +
  geom_polygon(data=states, aes(x=long, y=lat, group=group), fill="transparent", color="black", size=0.
  labs(title="Allison-1999") +
  theme_void() +
  geom_path(data=allison_track, aes(x=longitude, y=latitude), color="red", size=0.5)
```

Allison−1999