STAT 3280 Homework 3

Yingyi Zhu

September 30, 2022

.Rmd file can be found on Collab under Resources/Assigments

Q1: Using the `State_to_State_Migration.RData` file, plot the estimated migration out of each state for the year 2015. Include only the contiguous 48 US states. Ensure, colors, labels, and themes make the plot clear and easy to understand.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

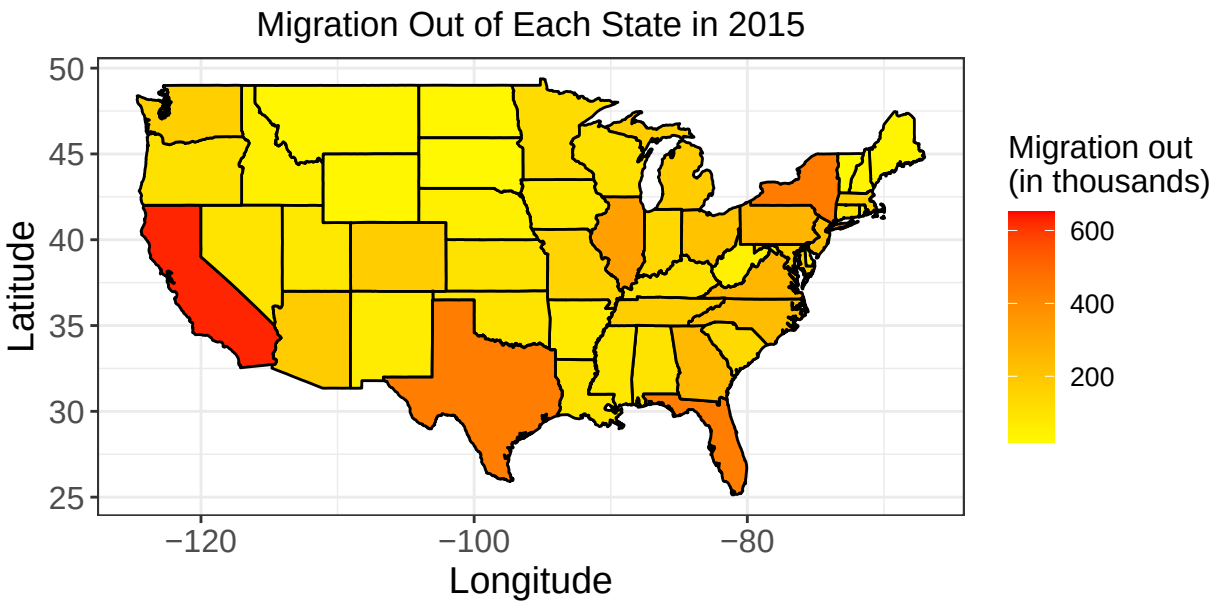
```
library(maps)
```

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

## The following object is masked from 'package:purrr':
##
##     map
```

```
library(dplyr)
library(ggplot2)
library(viridis)
```

```
## Loading required package: viridisLite

##
## Attaching package: 'viridis'

## The following object is masked from 'package:maps':
##
##     unemp
```

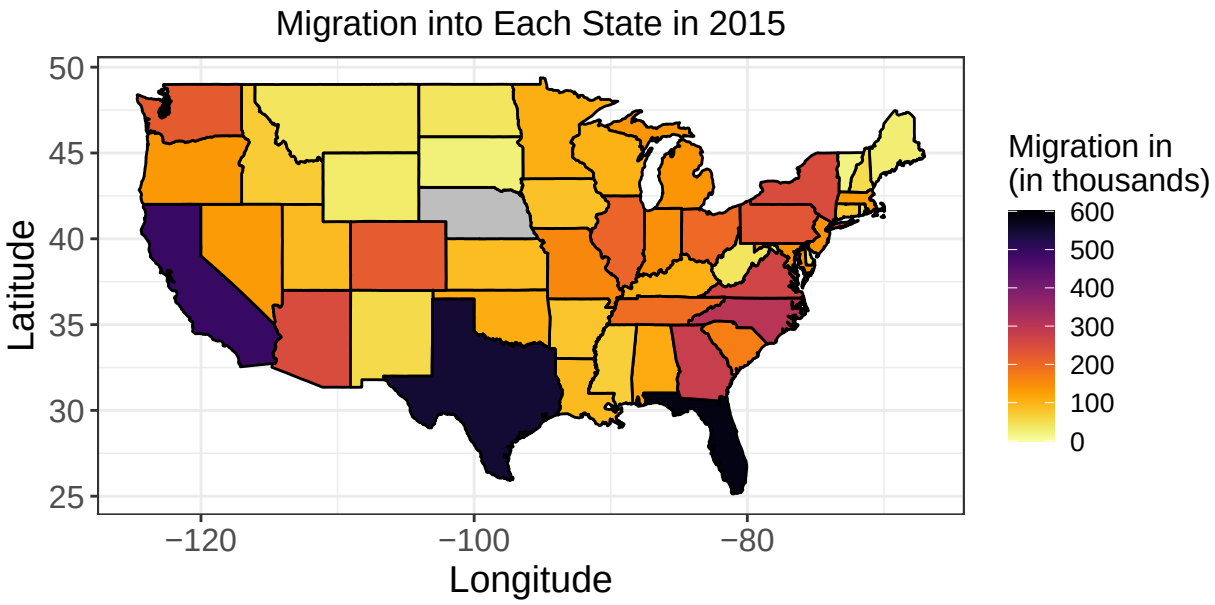
```r
setwd("/Users/zach0422/Desktop/STAT3280/data")
load("State_to_State_Migration.RData")

my_theme <- theme_bw() +
  theme(axis.text = element_text(size = 12),
        axis.title = element_text(size = 14),
        legend.text = element_text(size = 10),
        legend.title = element_text(size = 12)) +
  theme(plot.title = element_text(hjust = 0.5))

Migration1 <- Migration %>%
  filter(year == 2015 &
  state_to != "Alaska" &
  state_from != "Alaska" &
  state_to != "U.S. Island Area" &
  state_to != "Puerto Rico" &
  state_to != "Foreign Country" &
  state_to != "Hawaii" &
  state_from != "Hawaii" &
  state_from != "Puerto Rico" &
  state_from != "Foreign Country" &
  state_from != "U.S. Island Area" &
  state_from != "Hawaii" &
  state_from != "District of Columbia" &
  state_to != "District of Columbia")

Migration2 <- Migration1 %>%
  mutate(region = str_to_lower(state_from)) %>%
  group_by(state_from) %>%
  mutate(Total_Migration_out = sum(estimate))

state_map <- map_data("state")

state_map1 <- state_map %>%
  inner_join(Migration2, by = "region") %>%
  distinct(long, lat, group, Total_Migration_out)

plot1 <- ggplot(state_map1) +
  geom_polygon(aes(x = long, y = lat, group = group,
                   fill = Total_Migration_out / 1000), color = "black") +
      scale_fill_gradient2('Migration out \n(in thousands)',
                           low = "white",
                           mid = "yellow",
                           high = "red",
                           na.value = "gray",
                           limits = c(20, 650)) +
  coord_quickmap() +
  labs(x = "Longitude",
       y = "Latitude",
       title = "Migration Out of Each State in 2015") +
  my_theme

plot1
```



Q2: Using the `State_to_State_Migration.RData` file, plot the estimated migration into each state for the year 2015. Include only the contiguous 48 US states. Ensure, colors, labels, and themes make the plot clear and easy to understand. Color any `NA` values gray.

```
Migration3 <- Migration1 %>%
  mutate(region = str_to_lower(state_to)) %>%
  group_by(state_to) %>%
  mutate(Total_Migration_in = sum(estimate))

state_map2 <- state_map %>%
  left_join(Migration3, by = "region") %>%
  distinct(long, lat, group, Total_Migration_in)

plot2 <- ggplot(state_map2) +
  geom_polygon(aes(x = long,
                   y = lat,
                   group = group,
                   fill = Total_Migration_in / 1000),
               color = "black") +
      scale_fill_viridis(option = "inferno",
                         begin = 1,
                         end = 0,
                         na.value = "gray",
                         limits = c(0, 600)) +
  coord_quickmap() +
```

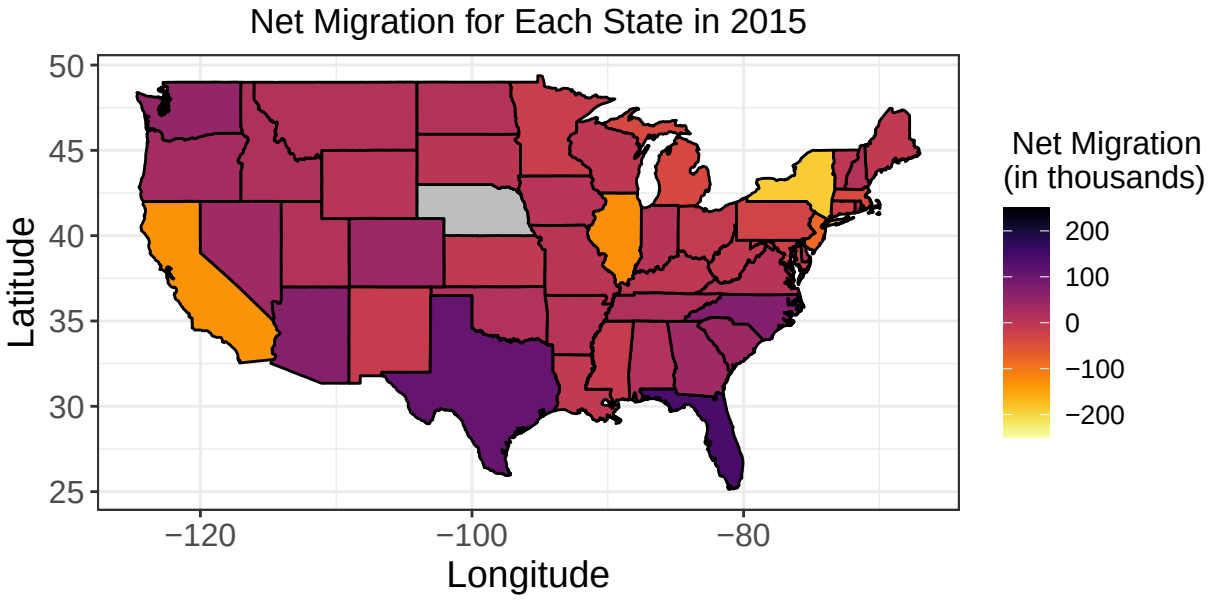
```
  labs(x = "Longitude",
       y = "Latitude",
       title = "Migration into Each State in 2015") +
  my_theme +
  labs(fill='Migration in \n(in thousands)')

plot2
```



Q3: Using the `State_to_State_Migration.RData` file, plot the estimated net migration for each state in the year 2015. Include only the contiguous 48 US states. Ensure, colors, labels, and themes make the plot clear and easy to understand. Color any `NA` values gray.

```
library(viridis)
Migration4 <- Migration3 %>%
  inner_join(Migration2, by = "region") %>%
  mutate(Net_Migration = Total_Migration_in - Total_Migration_out)

state_map3 <- state_map %>%
  left_join(Migration4, by = "region") %>%
  distinct(region, long, lat, Net_Migration, group)

plot3 <- ggplot(state_map3) +
  geom_polygon(aes(x = long,
                   y = lat,
```

```
                    group = group,
                    fill = Net_Migration / 1000),
                color = "black") +
    scale_fill_viridis(' Net Migration \n(in thousands)',
                       option = "inferno",
                       begin = 1,
                       end = 0,
                       na.value = "gray",
                       limits = c(-250, 250),
                       oob = scales::squish) +
  coord_quickmap() +
  labs(x = "Longitude",
       y = "Latitude",
       title = "Net Migration for Each State in 2015") +
  my_theme

plot3
```

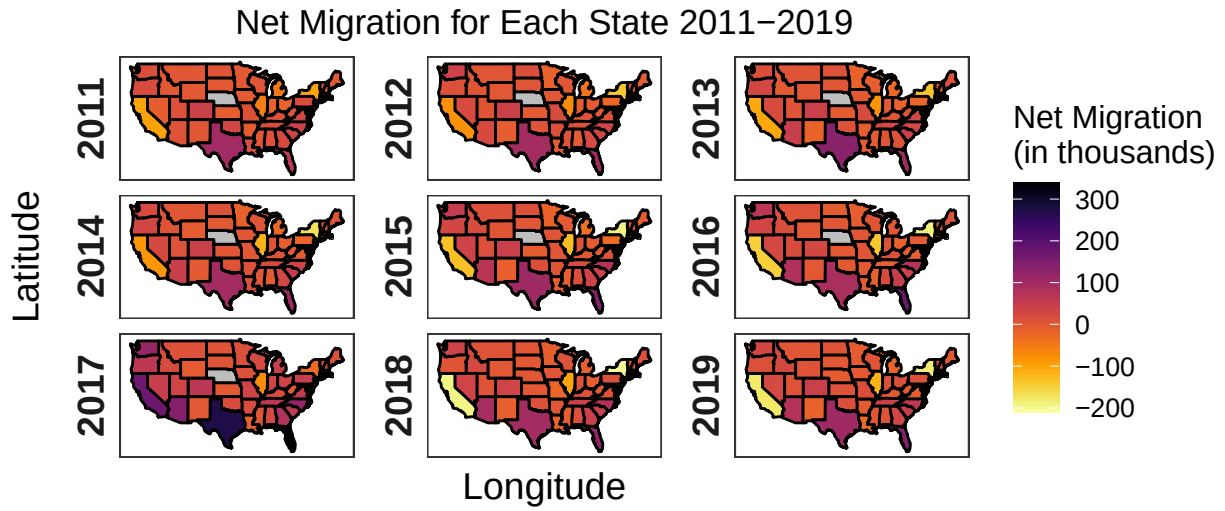


Migration to Nebrasaka data for 2015 does not exist.

Q4: Using the `State_to_State_Migration.RData` file, plot the estimated net migration for each state for all years 2011 to 2019. Include only the contiguous 48 US states. Ensure, colors, labels, and themes make the plot clear and easy to understand. Color any `NA` values gray.

```
Migration5 <- Migration %>%
  filter(state_to != "Alaska" &
  state_from != "Alaska" &
  state_to != "U.S. Island Area" &
  state_to != "Puerto Rico" &
  state_to != "Foreign Country" &
  state_to != "Hawaii" &
  state_from != "Hawaii" &
  state_from != "Puerto Rico" &
  state_from != "Foreign Country" &
  state_from != "U.S. Island Area" &
  state_from != "Hawaii" &
  state_from != "District of Columbia" &
  state_to != "District of Columbia") %>%
  mutate(region = str_to_lower(state_to)) %>%
  group_by(state_to, year) %>%
  mutate(Total_Migration_in = sum(estimate)) %>%
  distinct(region, Total_Migration_in)

Migration6 <- Migration %>%
  filter(state_to != "Alaska" &
  state_from != "Alaska" &
  state_to != "U.S. Island Area" &
  state_to != "Puerto Rico" &
  state_to != "Foreign Country" &
  state_to != "Hawaii" &
  state_from != "Hawaii" &
  state_from != "Puerto Rico" &
  state_from != "Foreign Country" &
  state_from != "U.S. Island Area" &
  state_from != "Hawaii" &
  state_from != "District of Columbia" &
  state_to != "District of Columbia") %>%
  mutate(region = str_to_lower(state_from)) %>%
  group_by(state_from, year) %>%
  mutate(Total_Migration_out = sum(estimate)) %>%
  distinct(region, Total_Migration_out)

Migration7 <- Migration6 %>%
  left_join(Migration5, by = c("year", "region")) %>%
  mutate(Net_Migration = Total_Migration_in - Total_Migration_out)

state_map4 <- state_map %>%
  inner_join(Migration7, by = "region") %>%
  distinct()

plot4 <- ggplot(state_map4) +
  geom_polygon(aes(x = long,
                   y = lat,
                   group = group,
                   fill = Net_Migration / 1000),
               color = "black") +
    scale_fill_viridis("Net Migration \n(in thousands)",
```

```
                        option = "inferno",
                        begin = 1,
                        end = 0,
                        na.value = "gray",
                        limits = c(-210, 340),
                        oob = scales::squish) +
  coord_quickmap() +
  labs(x = "Longitude",
       y = "Latitude",
       title = "Net Migration for Each State 2011-2019") +
  my_theme +
  facet_wrap(~ year, ncol = 3, strip.position = "left") +
  theme(strip.background = element_blank(),
        axis.text = element_blank(),
        axis.ticks = element_blank(), panel.grid = element_blank(),
        strip.text = element_text(size = 14, face = "bold"))

plot4
```



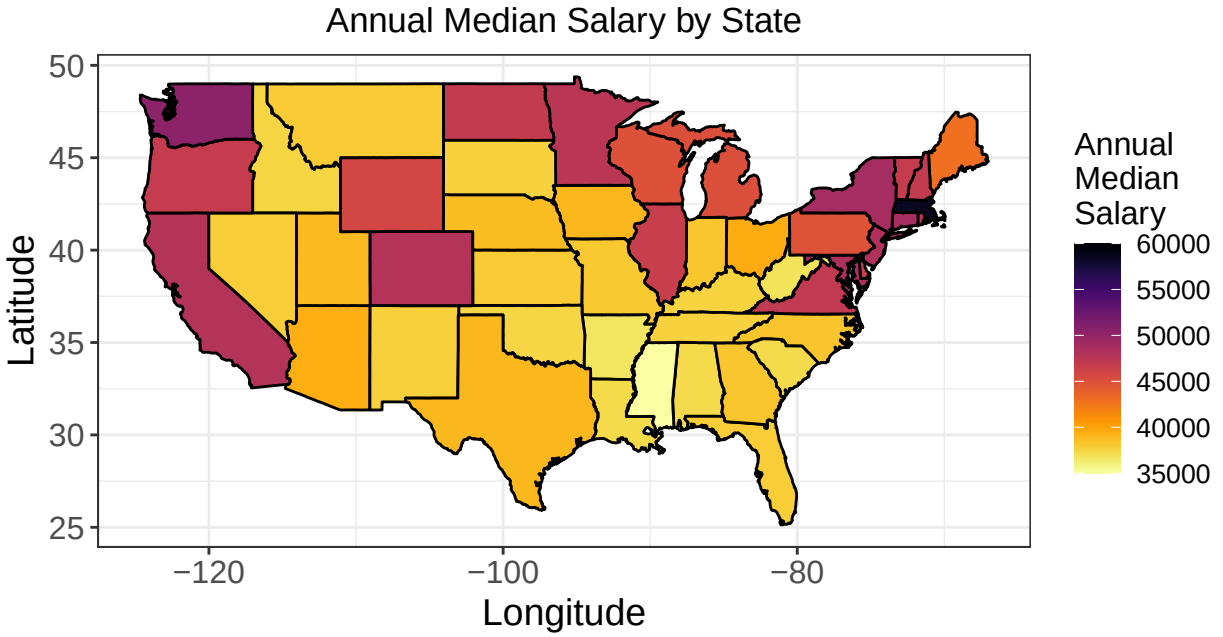
#data for Nebraska from 2011-2017 is missing. I removed the axis labels #to make the plot clear

Q5: Using the `Salary.RData` file, plot the annual median salary `A_MEDIAN` by state `AREA_TITLE`. To obtain this information, you must first filter the full dataset by `OCC_TITLE == "All Occupations"`.

```r
setwd("/Users/zach0422/Desktop/STAT3280/data")
load("Salary.RData")
Salary1 <- Salary %>%
  filter(OCC_TITLE == "All Occupations") %>%
  mutate(region = str_to_lower(AREA_TITLE))

state_map5 <- state_map %>%
  inner_join(Salary1) %>%
  distinct(long, lat, group, region, A_MEDIAN)
```

```
## Joining, by = "region"
```

```r
state_map5$A_MEDIAN <- as.numeric(state_map5$A_MEDIAN)

plot5 <- ggplot(state_map5) +
  geom_polygon(aes(x = long, y = lat, group = group,
                   fill = A_MEDIAN), color = "black") +
  coord_quickmap() +
      scale_fill_viridis("Annual \nMedian \nSalary",
                         option = "inferno",
                         begin = 1,
                         end = 0,
                         na.value = "gray",
                         limits = c(35000, 60000),
                         oob = scales::squish) +
  labs(x = "Longitude", y = "Latitude",
       title = "Annual Median Salary by State") +
  my_theme

plot5
```



#District of Columbia has the highest median annual salary (79,960) but since it is not showing up on the map when I adjust the limit to 80,000(maybe too small), I adopt a new limit (35,000, 60,000) that exludes it for better data visualization.

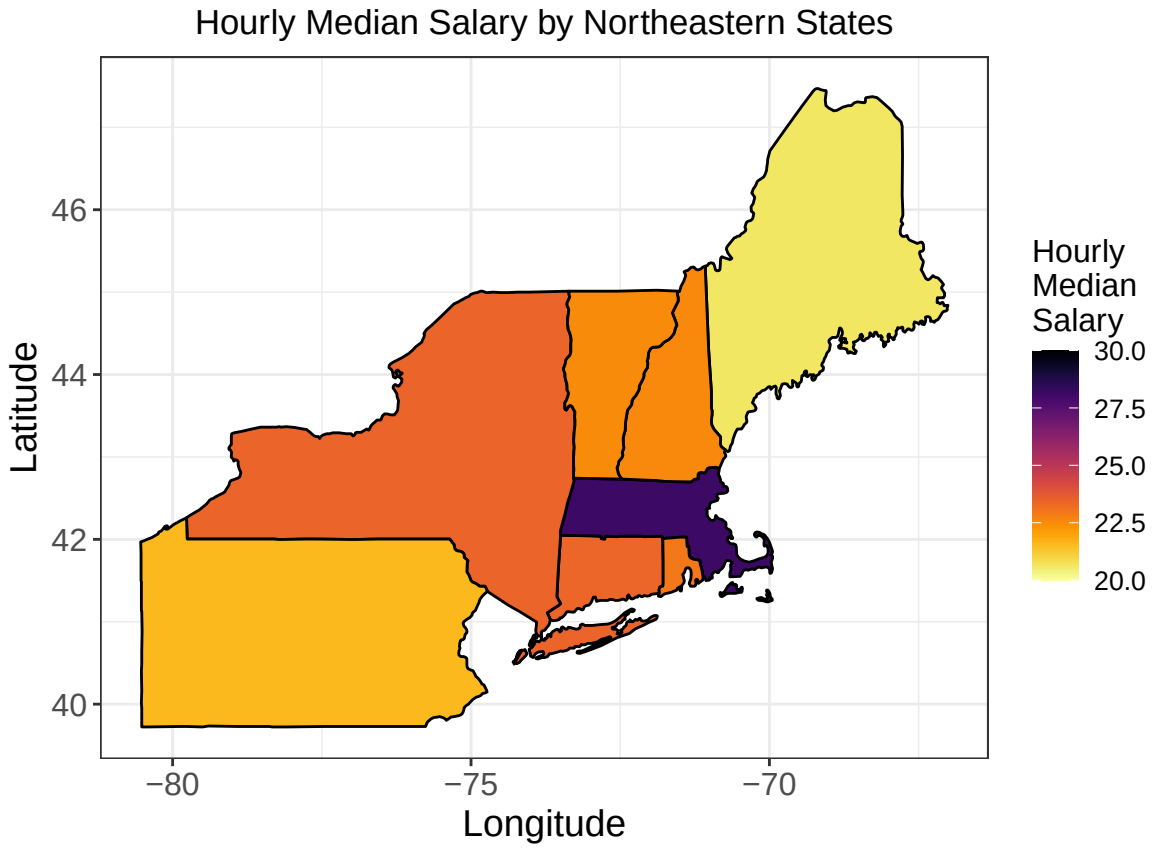
Q6: Plot the same information as in Q5, but instead use the hourly median salary, `H_MEDIAN`, as the response variable. Only include states in the Northeastern US (generally defined as New York or further NE).

```
state_map6 <- state_map %>%
  filter(region == "connecticut" |
           region == "maine" |
           region == "massachusetts" |
           region == "new hampshire" |
           region == "new jersy" |
           region == "new york" |
           region == "pennsylvania" |
           region == "rhode island" |
           region == "vermont") %>%
  inner_join(Salary1) %>%
  distinct(long, lat, group, region, H_MEDIAN)
```

```
## Joining, by = "region"
```

```
state_map6$H_MEDIAN <- as.numeric(state_map6$H_MEDIAN)
```

```
plot6 <- ggplot(state_map6) +
  geom_polygon(aes(x = long, y = lat, group = group,
                   fill = H_MEDIAN), color = "black") +
  coord_quickmap() +
      scale_fill_viridis("Hourly \nMedian \nSalary",
                         option = "inferno",
                         begin = 1,
                         end = 0,
                         na.value = "gray",
                         limits = c(20, 30),
                         oob = scales::squish) +
  labs(x = "Longitude", y = "Latitude",
       title = "Hourly Median Salary by Northeastern States") +
  my_theme

plot6
```



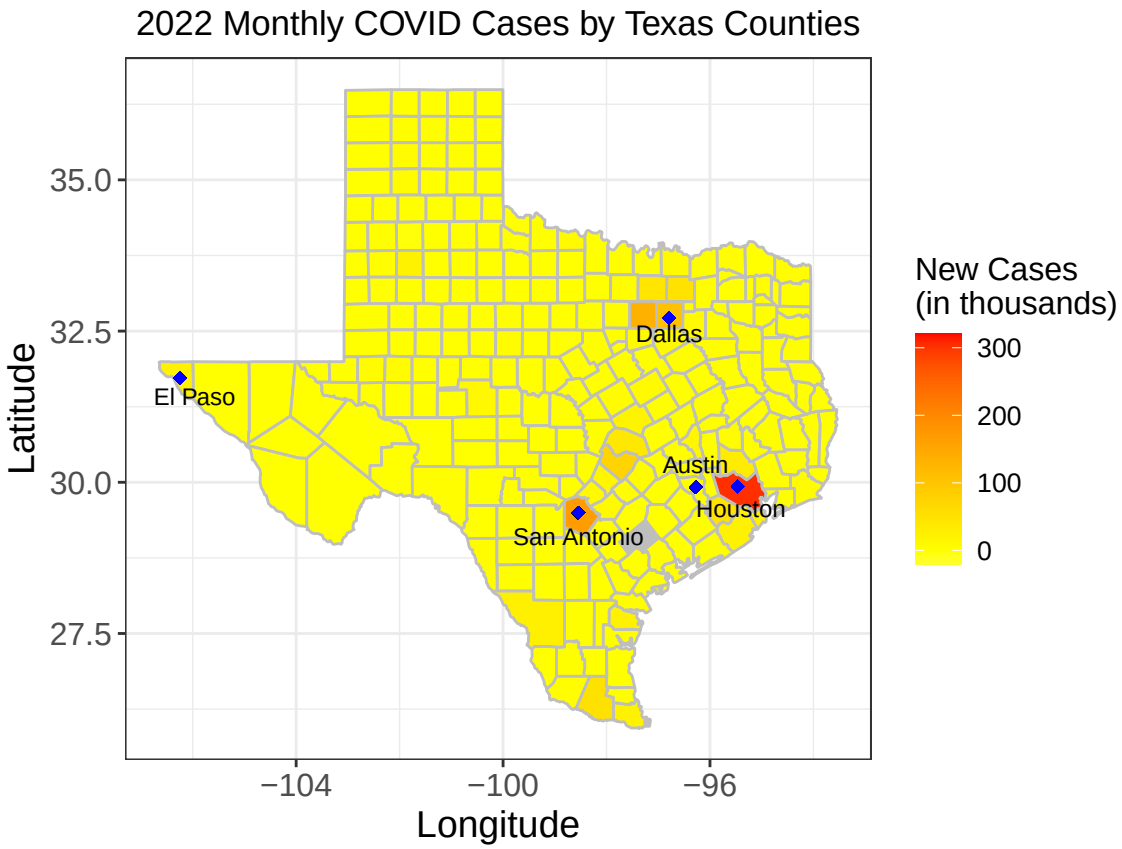
Q7: Using `COVID22.RData` plot the number of new COVID cases in January 2022 by county in the state of Texas. Add points and labels that correspond to the latitude and longitude of Houston, Dallas, Austin, San Antonio, and El Paso. Ensure colors, labels, and themes make the plot clear and easy to understand. Color any `NA` values gray.

```
setwd("/Users/zach0422/Desktop/STAT3280/data")
load("COVID22.RData")
```

```r
COVID <- COVID22 %>%
  filter(state == "Texas") %>%
  filter(date == "2022-01-01" | date == "2022-01-31") %>%
  group_by(county) %>%
  mutate(newcases = c(0, diff(cases))) %>%
  filter(date != "2022-01-01") %>%
  mutate(subregion = str_to_lower(county))

county_map <- map_data("county")

TX_county <- county_map %>%
  filter(region == "texas") %>%
  left_join(COVID)
```

```
## Joining, by = "subregion"
```

```r
plot7 <- ggplot(TX_county) +
  geom_polygon(aes(x = long, y = lat, group = group,
                   fill = newcases / 1000), color = "gray") +
  scale_fill_gradient2("New Cases \n(in thousands)",
                       low = "white",
                       mid = "yellow",
                       high = "red",
                       na.value = "gray",
                       limits = c(-20, 320)) +
  coord_quickmap() +
  geom_point(aes(x = -95.46360, y = 29.92870), color = "blue", shape = 18,
             size = 2) +
  geom_point(aes(x = -96.79132, y = 32.72091), color = "blue", shape = 18,
             size = 2) +
  geom_point(aes(x = -96.27690, y = 29.91921), color = "blue", shape = 18,
             size = 2) +
  geom_point(aes(x = -98.54301, y = 29.49516), color = "blue", shape = 18,
             size = 2) +
  geom_point(aes(x = -106.23972, y = 31.72396), color = "blue", shape = 18,
             size = 2) +
  annotate("text", x = -95.40, y = 29.57, label = "Houston",
           color = "black", size = 3) +
  annotate("text", x = -96.79, y = 32.47, label = "Dallas",
           color = "black", size = 3) +
  annotate("text", x = -96.27690, y = 30.30, label = "Austin",
           color = "black", size = 3) +
  annotate("text", x = -105.95972, y = 31.42, label = "El Paso",
           color = "black", size = 3) +
  annotate("text", x = -98.543, y = 29.1, label = "San Antonio",
           color = "black", size = 3) +
  labs(x = "Longitude", y = "Latitude",
       title = "2022 Monthly COVID Cases by Texas Counties") +
  my_theme

plot7
```



#Dewitt county does not exist in the county map.