Covid-19 United States Analysis

Marc Vucovich

2023-04-26

Import Libraries

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```
library(readr)
library(dplyr)
library(ggplot2)
```

Load in COVID-19 Data and US Census Data

Load the Covid-19 data sets from the URL provided in the code chunk below. Once the data has been loaded, drop the NA's from the data set and output the first 10 rows to ensure the data sets are correct.

```
url_in <-"https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covi
```

```
file_names <- c("time_series_covid19_confirmed_US.csv",
                "time_series_covid19_confirmed_global.csv",
                "time_series_covid19_deaths_US.csv",
                "time_series_covid19_deaths_global.csv")
```

```
urls <- str_c(url_in, file_names)
urls
```

```
## [1] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_
## [2] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_
## [3] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_
## [4] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_
```

```
us_cases1 <- read_csv(urls[1])
```

```
## Rows: 3342 Columns: 1154
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (6): iso2, iso3, Admin2, Province_State, Country_Region, Combined_Key
## dbl (1148): UID, code3, FIPS, Lat, Long_, 1/22/20, 1/23/20, 1/24/20, 1/25/20...
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
global_cases1 <- read_csv(urls[2])
```

```
## Rows: 289 Columns: 1147
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (2): Province/State, Country/Region
## dbl (1145): Lat, Long, 1/22/20, 1/23/20, 1/24/20, 1/25/20, 1/26/20, 1/27/20,...
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
us_deaths1 <- read_csv(urls[3])
```

```
## Rows: 3342 Columns: 1155
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (6): iso2, iso3, Admin2, Province_State, Country_Region, Combined_Key
## dbl (1149): UID, code3, FIPS, Lat, Long_, Population, 1/22/20, 1/23/20, 1/24...
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
global_deaths1 <- read_csv(urls[4])
```

```
## Rows: 289 Columns: 1147
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (2): Province/State, Country/Region
## dbl (1145): Lat, Long, 1/22/20, 1/23/20, 1/24/20, 1/25/20, 1/26/20, 1/27/20,...
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
us_cases1 <- drop_na(us_cases1)
head(us_cases1, n=10)
```

```
## # A tibble: 10 x 1,154
##         UID iso2  iso3  code3  FIPS Admin2   Province_State Country_Region   Lat
##       <dbl> <chr> <chr> <dbl> <dbl> <chr>    <chr>          <chr>          <dbl>
##  1 84001001 US    USA     840  1001 Autauga  Alabama        US              32.5
##  2 84001003 US    USA     840  1003 Baldwin  Alabama        US              30.7
##  3 84001005 US    USA     840  1005 Barbour  Alabama        US              31.9
##  4 84001007 US    USA     840  1007 Bibb     Alabama        US              33.0
##  5 84001009 US    USA     840  1009 Blount   Alabama        US              34.0
##  6 84001011 US    USA     840  1011 Bullock  Alabama        US              32.1
##  7 84001013 US    USA     840  1013 Butler   Alabama        US              31.8
##  8 84001015 US    USA     840  1015 Calhoun  Alabama        US              33.8
##  9 84001017 US    USA     840  1017 Chambers Alabama        US              32.9
## 10 84001019 US    USA     840  1019 Cherokee Alabama        US              34.2
## # i 1,145 more variables: Long_ <dbl>, Combined_Key <chr>, ‘1/22/20‘ <dbl>,
## #   ‘1/23/20‘ <dbl>, ‘1/24/20‘ <dbl>, ‘1/25/20‘ <dbl>, ‘1/26/20‘ <dbl>,
## #   ‘1/27/20‘ <dbl>, ‘1/28/20‘ <dbl>, ‘1/29/20‘ <dbl>, ‘1/30/20‘ <dbl>,
## #   ‘1/31/20‘ <dbl>, ‘2/1/20‘ <dbl>, ‘2/2/20‘ <dbl>, ‘2/3/20‘ <dbl>,
## #   ‘2/4/20‘ <dbl>, ‘2/5/20‘ <dbl>, ‘2/6/20‘ <dbl>, ‘2/7/20‘ <dbl>,
## #   ‘2/8/20‘ <dbl>, ‘2/9/20‘ <dbl>, ‘2/10/20‘ <dbl>, ‘2/11/20‘ <dbl>,
## #   ‘2/12/20‘ <dbl>, ‘2/13/20‘ <dbl>, ‘2/14/20‘ <dbl>, ‘2/15/20‘ <dbl>, ...
```

```
global_cases1 <- drop_na(global_cases1)
head(global_cases1, n=10)
```

```
## # A tibble: 10 x 1,147
##    ‘Province/State‘   ‘Country/Region‘   Lat  Long ‘1/22/20‘ ‘1/23/20‘ ‘1/24/20‘
##    <chr>              <chr>            <dbl> <dbl>     <dbl>     <dbl>     <dbl>
##  1 Australian Capita~ Australia        -35.5  149.         0         0         0
##  2 New South Wales    Australia        -33.9  151.         0         0         0
##  3 Northern Territory Australia        -12.5  131.         0         0         0
##  4 Queensland         Australia        -27.5  153.         0         0         0
##  5 South Australia    Australia        -34.9  139.         0         0         0
##  6 Tasmania           Australia        -42.9  147.         0         0         0
##  7 Victoria           Australia        -37.8  145.         0         0         0
##  8 Western Australia  Australia        -32.0  116.         0         0         0
##  9 Alberta            Canada            53.9 -117.         0         0         0
## 10 British Columbia   Canada            53.7 -128.         0         0         0
## # i 1,140 more variables: ‘1/25/20‘ <dbl>, ‘1/26/20‘ <dbl>, ‘1/27/20‘ <dbl>,
## #   ‘1/28/20‘ <dbl>, ‘1/29/20‘ <dbl>, ‘1/30/20‘ <dbl>, ‘1/31/20‘ <dbl>,
## #   ‘2/1/20‘ <dbl>, ‘2/2/20‘ <dbl>, ‘2/3/20‘ <dbl>, ‘2/4/20‘ <dbl>,
## #   ‘2/5/20‘ <dbl>, ‘2/6/20‘ <dbl>, ‘2/7/20‘ <dbl>, ‘2/8/20‘ <dbl>,
## #   ‘2/9/20‘ <dbl>, ‘2/10/20‘ <dbl>, ‘2/11/20‘ <dbl>, ‘2/12/20‘ <dbl>,
## #   ‘2/13/20‘ <dbl>, ‘2/14/20‘ <dbl>, ‘2/15/20‘ <dbl>, ‘2/16/20‘ <dbl>,
## #   ‘2/17/20‘ <dbl>, ‘2/18/20‘ <dbl>, ‘2/19/20‘ <dbl>, ‘2/20/20‘ <dbl>, ...
```

```
us_deaths1 <- drop_na(us_deaths1)
head(us_deaths1, n=10)
```

```
## # A tibble: 10 x 1,155
##         UID iso2  iso3  code3  FIPS Admin2   Province_State Country_Region   Lat
##       <dbl> <chr> <chr> <dbl> <dbl> <chr>    <chr>          <chr>          <dbl>
##  1 84001001 US    USA     840  1001 Autauga  Alabama        US              32.5
##  2 84001003 US    USA     840  1003 Baldwin  Alabama        US              30.7
##  3 84001005 US    USA     840  1005 Barbour  Alabama        US              31.9
##  4 84001007 US    USA     840  1007 Bibb     Alabama        US              33.0
##  5 84001009 US    USA     840  1009 Blount   Alabama        US              34.0
##  6 84001011 US    USA     840  1011 Bullock  Alabama        US              32.1
##  7 84001013 US    USA     840  1013 Butler   Alabama        US              31.8
##  8 84001015 US    USA     840  1015 Calhoun  Alabama        US              33.8
##  9 84001017 US    USA     840  1017 Chambers Alabama        US              32.9
## 10 84001019 US    USA     840  1019 Cherokee Alabama        US              34.2
## # i 1,146 more variables: Long_ <dbl>, Combined_Key <chr>, Population <dbl>,
## #   ‘1/22/20‘ <dbl>, ‘1/23/20‘ <dbl>, ‘1/24/20‘ <dbl>, ‘1/25/20‘ <dbl>,
## #   ‘1/26/20‘ <dbl>, ‘1/27/20‘ <dbl>, ‘1/28/20‘ <dbl>, ‘1/29/20‘ <dbl>,
## #   ‘1/30/20‘ <dbl>, ‘1/31/20‘ <dbl>, ‘2/1/20‘ <dbl>, ‘2/2/20‘ <dbl>,
## #   ‘2/3/20‘ <dbl>, ‘2/4/20‘ <dbl>, ‘2/5/20‘ <dbl>, ‘2/6/20‘ <dbl>,
## #   ‘2/7/20‘ <dbl>, ‘2/8/20‘ <dbl>, ‘2/9/20‘ <dbl>, ‘2/10/20‘ <dbl>,
## #   ‘2/11/20‘ <dbl>, ‘2/12/20‘ <dbl>, ‘2/13/20‘ <dbl>, ‘2/14/20‘ <dbl>, ...
```

```
global_deaths1 <- drop_na(global_deaths1)
head(global_deaths1, n=10)
```

```
## # A tibble: 10 x 1,147
##    ‘Province/State‘   ‘Country/Region‘   Lat  Long ‘1/22/20‘ ‘1/23/20‘ ‘1/24/20‘
##    <chr>              <chr>            <dbl> <dbl>     <dbl>     <dbl>     <dbl>
##  1 Australian Capita~ Australia        -35.5  149.         0         0         0
##  2 New South Wales    Australia        -33.9  151.         0         0         0
##  3 Northern Territory Australia        -12.5  131.         0         0         0
##  4 Queensland         Australia        -27.5  153.         0         0         0
##  5 South Australia    Australia        -34.9  139.         0         0         0
##  6 Tasmania           Australia        -42.9  147.         0         0         0
##  7 Victoria           Australia        -37.8  145.         0         0         0
##  8 Western Australia  Australia        -32.0  116.         0         0         0
##  9 Alberta            Canada            53.9 -117.         0         0         0
## 10 British Columbia   Canada            53.7 -128.         0         0         0
## # i 1,140 more variables: ‘1/25/20‘ <dbl>, ‘1/26/20‘ <dbl>, ‘1/27/20‘ <dbl>,
## #   ‘1/28/20‘ <dbl>, ‘1/29/20‘ <dbl>, ‘1/30/20‘ <dbl>, ‘1/31/20‘ <dbl>,
## #   ‘2/1/20‘ <dbl>, ‘2/2/20‘ <dbl>, ‘2/3/20‘ <dbl>, ‘2/4/20‘ <dbl>,
## #   ‘2/5/20‘ <dbl>, ‘2/6/20‘ <dbl>, ‘2/7/20‘ <dbl>, ‘2/8/20‘ <dbl>,
## #   ‘2/9/20‘ <dbl>, ‘2/10/20‘ <dbl>, ‘2/11/20‘ <dbl>, ‘2/12/20‘ <dbl>,
## #   ‘2/13/20‘ <dbl>, ‘2/14/20‘ <dbl>, ‘2/15/20‘ <dbl>, ‘2/16/20‘ <dbl>,
## #   ‘2/17/20‘ <dbl>, ‘2/18/20‘ <dbl>, ‘2/19/20‘ <dbl>, ‘2/20/20‘ <dbl>, ...
```

```
#Read in census population data gathered from the US Census

#IMPORTANT: MAY NEED TO ADJUST PATH
pop <- read_csv("Downloads/Census.csv")
```

```
## Rows: 76 Columns: 2
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): State
## num (1): Pop
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
#Change the Column "State" to match the covid dataset
pop$Province_State <- pop$State
```

```
#Check to make sure the datafram looks correct
head(pop, n = 5)
```

```
## # A tibble: 5 x 3
##   State           Pop Province_State
##   <chr>         <dbl> <chr>
## 1 Alabama     5044965 Alabama
## 2 Alaska       733517 Alaska
## 3 Arizona     7238881 Arizona
## 4 Arkansas    3024877 Arkansas
## 5 California 39303058 California
```

Manipulate COVID-19 and US Census Data Set for Confirmed US Cases

First, we are going to organize the data for our analysis. To do this we will drop the necessary columns and create a pivot table to set the data columns as rows.

```
#Remove the columns we don't need
us_cases <- us_cases1 %>%
  select(-c(UID,iso2,iso3,code3,FIPS,Admin2,Lat, Long_, Combined_Key))
```

Create a pivot table for the us confirmed cases to align the dates in rows instead of columns.

```
#Create a pivot to convert the date columns into rows
us_cases <- us_cases %>%
  pivot_longer(cols = -c(Province_State, Country_Region),
               names_to = "date",
               values_to = "cases")
```

Check the first 10 entries of the data set.

```
head(us_cases, n=10)
```

```
## # A tibble: 10 x 4
##    Province_State Country_Region date    cases
##    <chr>          <chr>          <chr>   <dbl>
##  1 Alabama        US             1/22/20     0
##  2 Alabama        US             1/23/20     0
##  3 Alabama        US             1/24/20     0
##  4 Alabama        US             1/25/20     0
##  5 Alabama        US             1/26/20     0
##  6 Alabama        US             1/27/20     0
##  7 Alabama        US             1/28/20     0
##  8 Alabama        US             1/29/20     0
##  9 Alabama        US             1/30/20     0
## 10 Alabama        US             1/31/20     0
```

Group the confirmed US cases data set by the State and the Year.

```
#Group the data by the States and the Dates and sum up the total case reported
us_states <- us_cases %>%
  mutate(yr = substr(date, nchar(date)-2+1, nchar(date))) %>%
  group_by(Province_State, yr) %>%
  summarise(total_cases=sum(cases),
            mean_cases=(mean(cases)))
```

```
## 'summarise()' has grouped output by 'Province_State'. You can override using
## the '.groups' argument.
```

Check the data set.

```
head(us_states, n=5)
```

```
## # A tibble: 5 x 4
## # Groups:   Province_State [2]
##   Province_State yr    total_cases mean_cases
##   <chr>          <chr>       <dbl>      <dbl>
## 1 Alabama        20       32296555      1357.
## 2 Alabama        21      227778353      9044.
## 3 Alabama        22      502789993     19964.
## 4 Alabama        23      109891172     23421.
## 5 Alaska         20        2875733       253.
```

Create the data frame of the total confirmed cases by state from the years 2020-2023.

```
tot_cases_states <- us_states %>%
  group_by(Province_State) %>%
  summarise(total_cases=sum(total_cases),
            mean_cases=(mean(total_cases)))
tot_cases_states <- merge(tot_cases_states,pop, by="Province_State", all.x=T)
```

```
tot_cases_states <- drop_na(tot_cases_states )
```

```
head(tot_cases_states, n =5)
```

```
##   Province_State total_cases mean_cases      State      Pop
## 1        Alabama   872756073  872756073    Alabama  5044965
## 2         Alaska   153011898  153011898     Alaska   733517
## 3        Arizona  1330372436 1330372436    Arizona  7238881
## 4       Arkansas   549955573  549955573   Arkansas  3024877
## 5     California  6166190335 6166190335 California 39303058
```

Use the census data to get the total cases by state divided by the states population.

```
#tot_div_pop <- merge(tot_cases_states,pop, by="Province_State", all.x=T)
tot_div_pop <- transform(tot_cases_states, new = as.numeric(total_cases) / as.numeric(Pop))
```

```
head(tot_div_pop, n=5)
```

```
##   Province_State total_cases mean_cases      State      Pop      new
## 1        Alabama   872756073  872756073    Alabama  5044965 172.9955
## 2         Alaska   153011898  153011898     Alaska   733517 208.6003
## 3        Arizona  1330372436 1330372436    Arizona  7238881 183.7815
## 4       Arkansas   549955573  549955573   Arkansas  3024877 181.8109
## 5     California  6166190335 6166190335 California 39303058 156.8883
```

Break down the total confirmed US cases by year.

```
#Breakdown the data by year
covid_20 <- us_states %>%
  filter(yr =='20')
covid_21 <- us_states %>%
  filter(yr =='21')
covid_22 <- us_states %>%
  filter(yr =='22')
covid_23 <- us_states %>%
  filter(yr =='23')
```

Use the census again to get the total cases divided by the population.

```
#Create a new dataframe that is a combination of the covid and census data
#2020
tot_div_pop_20 <- merge(covid_20,pop, by="Province_State", all.x=T)
tot_div_pop_20 <- transform(tot_div_pop_20, new = as.numeric(total_cases) / as.numeric(Pop))

#2021
tot_div_pop_21 <- merge(covid_21,pop, by="Province_State", all.x=T)
tot_div_pop_21 <- transform(tot_div_pop_21, new = as.numeric(total_cases) / as.numeric(Pop))
#tot_div_pop_21 <- transform(tot_div_pop_21, new = as.integer(total_cases) / as.integer(Pop))

#2022
tot_div_pop_22 <- merge(covid_22,pop, by="Province_State", all.x=T)
tot_div_pop_22 <- transform(tot_div_pop_22, new = as.numeric(total_cases) / as.numeric(Pop))

#2023
tot_div_pop_23 <- merge(covid_23,pop, by="Province_State", all.x=T)
tot_div_pop_23 <- transform(tot_div_pop_23, new = as.numeric(total_cases) / as.numeric(Pop))
```

Manipulate COVID-19 and US Census Data Set for Confirmed US Deaths

We will now repeat the steps above for the US confirmed deaths data set.

```
#Remove the columns we don't need
us_deaths <- us_deaths1 %>%
  select(-c(UID,iso2,iso3,code3,FIPS,Admin2,Lat, Long_, Combined_Key))
```

Create a pivot table for the us confirmed cases to align the dates in rows instead of columns.

```
#Create a pivot to convert the date columns into rows
us_deaths <- us_deaths %>%
  pivot_longer(cols = -c(Province_State, Country_Region),
               names_to = "date",
               values_to = "deaths")
```

Check the output

```
head(us_deaths, n=10)
```

```
## # A tibble: 10 x 4
##    Province_State Country_Region date       deaths
##    <chr>          <chr>          <chr>       <dbl>
##  1 Alabama        US             Population  55869
##  2 Alabama        US             1/22/20         0
##  3 Alabama        US             1/23/20         0
##  4 Alabama        US             1/24/20         0
##  5 Alabama        US             1/25/20         0
##  6 Alabama        US             1/26/20         0
##  7 Alabama        US             1/27/20         0
##  8 Alabama        US             1/28/20         0
##  9 Alabama        US             1/29/20         0
## 10 Alabama        US             1/30/20         0
```

Group the confirmed US deaths data set by the State and the Year.

```
#Group the data by the States and the Dates and sum up the total case reported
us_states_deaths <- us_deaths %>%
  mutate(yr = substr(date, nchar(date)-2+1, nchar(date))) %>%
  group_by(Province_State, yr) %>%
  summarise(total_deaths=sum(deaths),
            mean_cases=(mean(deaths)))
```

```
## 'summarise()' has grouped output by 'Province_State'. You can override using
## the '.groups' argument.
```

Create the data frame of the total confirmed deaths by state from the years 2020-2023.

```
tot_deaths_states <- us_states_deaths %>%
  group_by(Province_State) %>%
  summarise(total_deaths=sum(total_deaths),
            mean_cases=(mean(total_deaths)))
tot_deaths_states <- merge(tot_deaths_states,pop, by="Province_State", all.x=T)
```

Use the census data to get the total deaths by state divided by the states population.

```
#tot_deaths_div_pop <- merge(tot_deaths_states,pop, by="Province_State", all.x=T)
tot_deaths_div_pop <- transform(tot_deaths_states, new = as.numeric(total_deaths) / as.numeric(Pop))
```

Break down the total confirmed US deaths by year.

```
#Breakdown the data by year
covid_20_deaths <- us_states_deaths%>%
  filter(yr =='20')
covid_21_deaths <- us_states_deaths %>%
  filter(yr =='21')
covid_22_deaths <- us_states_deaths %>%
  filter(yr =='22')
covid_23_deaths <- us_states_deaths %>%
  filter(yr =='23')
```

Use the census again to get the total cases divided by the population.

```
#Create a new dataframe that is a combination of the covid and census data
#2020
tot_deaths_div_pop_20 <- merge(covid_20_deaths,pop, by="Province_State", all.x=T)
tot_deaths_div_pop_20 <- transform(tot_deaths_div_pop_20, new = as.numeric(total_deaths) / as.numeric(Po

#2021
tot_deaths_div_pop_21 <- merge(covid_21_deaths,pop, by="Province_State", all.x=T)
tot_deaths_div_pop_21 <- transform(tot_deaths_div_pop_21, new = as.numeric(total_deaths) / as.numeric(Po

#2022
tot_deaths_div_pop_22 <- merge(covid_22_deaths,pop, by="Province_State", all.x=T)
tot_deaths_div_pop_22 <- transform(tot_deaths_div_pop_22, new = as.numeric(total_deaths) / as.numeric(Po

#2023
tot_deaths_div_pop_23 <- merge(covid_23_deaths,pop, by="Province_State", all.x=T)
tot_deaths_div_pop_23 <- transform(tot_deaths_div_pop_23, new = as.numeric(total_deaths) / as.numeric(Po
```

Visualize Data

Here we will visualize all of the data we analyzed above. This will include the total confirmed cases by state from the year 2020 to the year 2023. It will then break down the confirmed covid cases by state and year, and finally it will show the year by year breakdown of confirmed cases in proprotion to each states population.

Heatmap of Total Confirmed Covid-19 Cases and Deaths 2020-2023

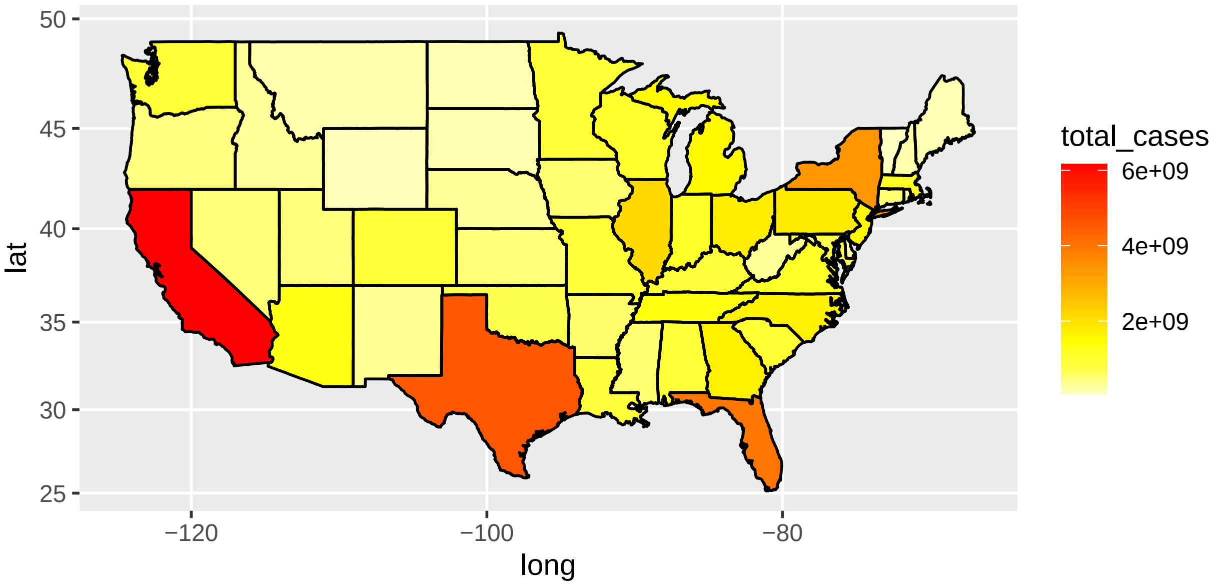
We will now plot a heat map of total confirmed cases in the US from 2020-2023

```
tot_cases_states$region <- tolower(tot_cases_states$Province_State)
library(ggplot2)
library(maps)
```

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

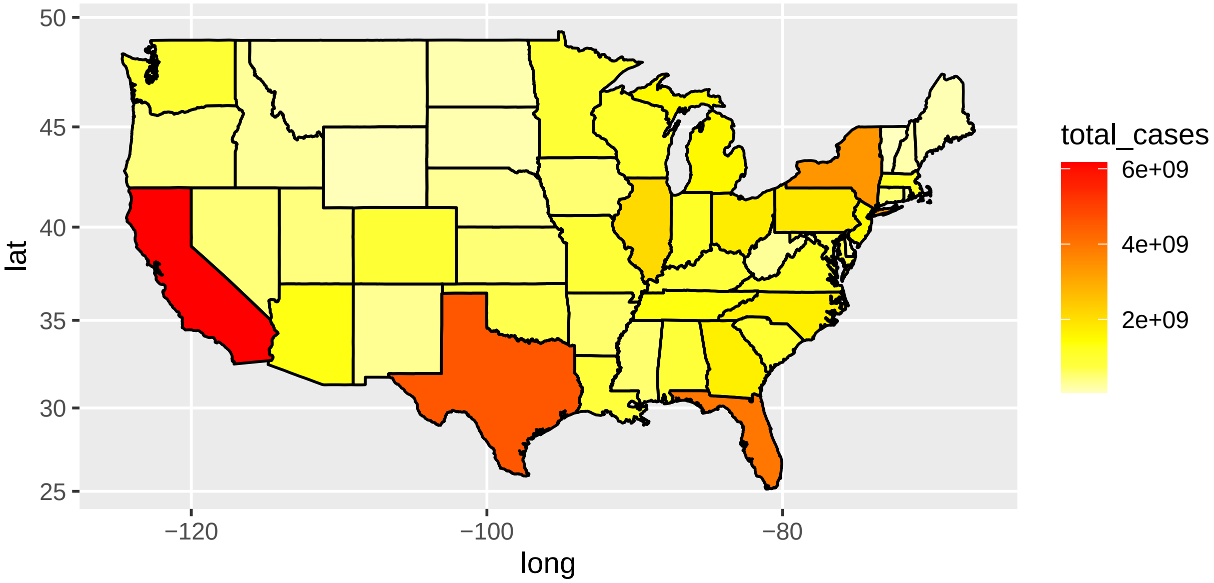
```
## The following object is masked from 'package:purrr':
##
##     map
```

```
states <- map_data("state")
map.df <- merge(states,tot_cases_states, by="region", all.x=T)
map.df <- map.df[order(map.df$order),]
ggplot(map.df, aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=total_cases))+
  geom_path()+
  scale_fill_gradientn(colours=rev(heat.colors(10)),na.value="grey90")+
  coord_map()
```



Break down total cases based on population.

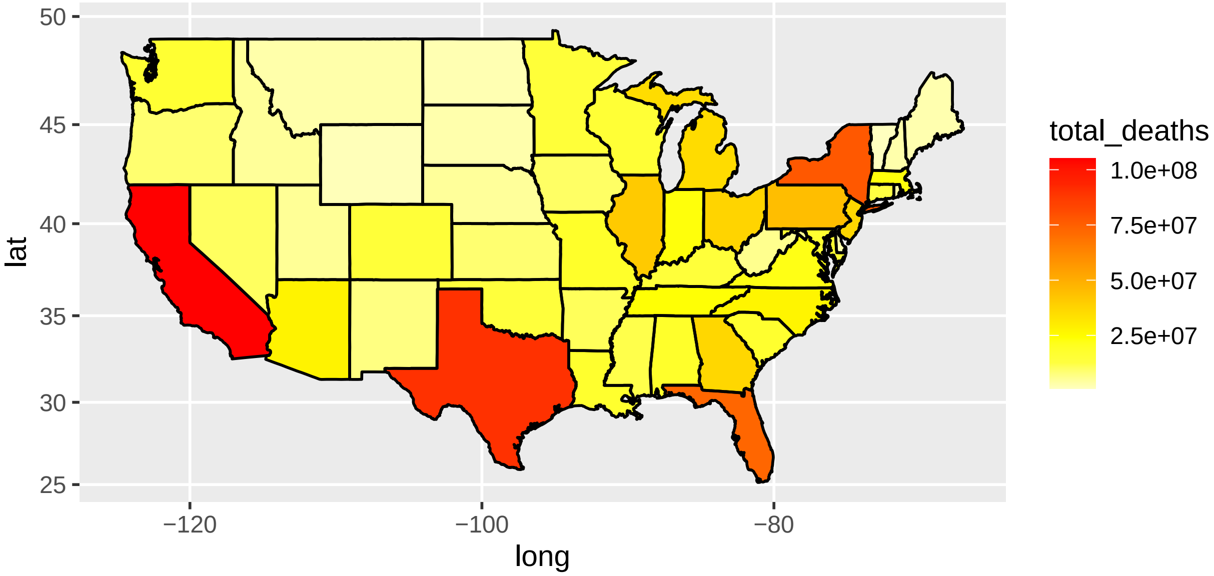
```
tot_div_pop$region <- tolower(tot_div_pop$Province_State)
library(ggplot2)
library(maps)
states <- map_data("state")
map.df <- merge(states,tot_div_pop, by="region", all.x=T)
map.df <- map.df[order(map.df$order),]
ggplot(map.df, aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=total_cases))+
  geom_path()+
  scale_fill_gradientn(colours=rev(heat.colors(10)),na.value="grey90")+
  coord_map()
```



As expected the states with the highest populations still had the most amount of cases in proportion to their population. Considering the virus spread by coming in contact with others, it makes sense the states with the most people would have the most cases.

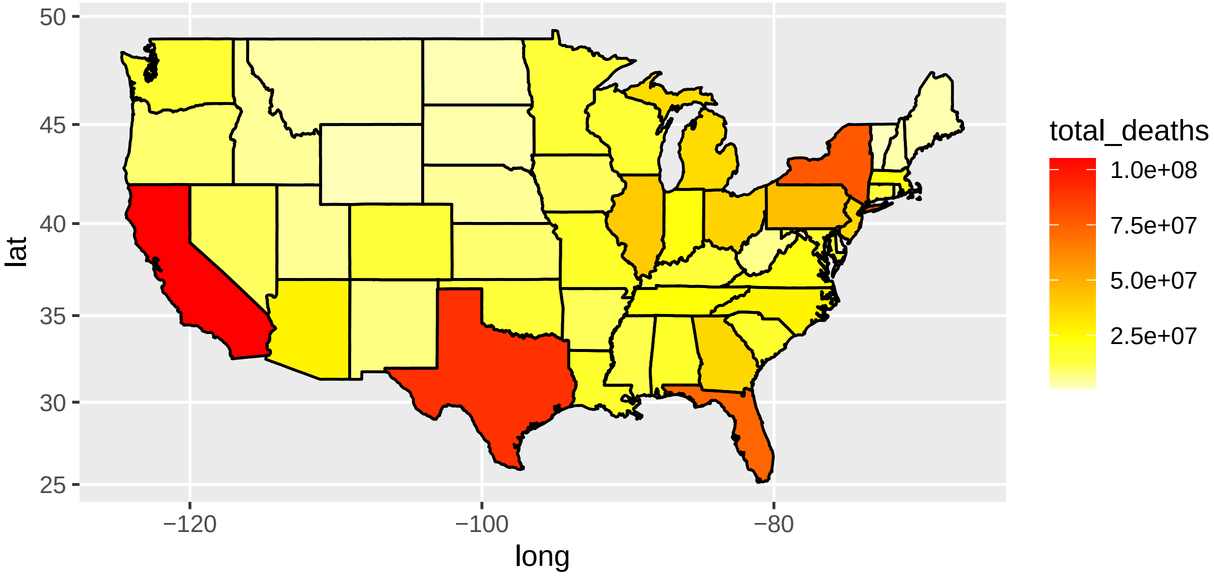
Breakdown total deaths from 2020-2023

```
tot_deaths_states$region <- tolower(tot_deaths_states$Province_State)
library(ggplot2)
library(maps)
states <- map_data("state")
map.df <- merge(states,tot_deaths_states, by="region", all.x=T)
map.df <- map.df[order(map.df$order),]
ggplot(map.df, aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=total_deaths))+
  geom_path()+
  scale_fill_gradientn(colours=rev(heat.colors(10)),na.value="grey90")+
  coord_map()
```



Heat map of total deaths based on population

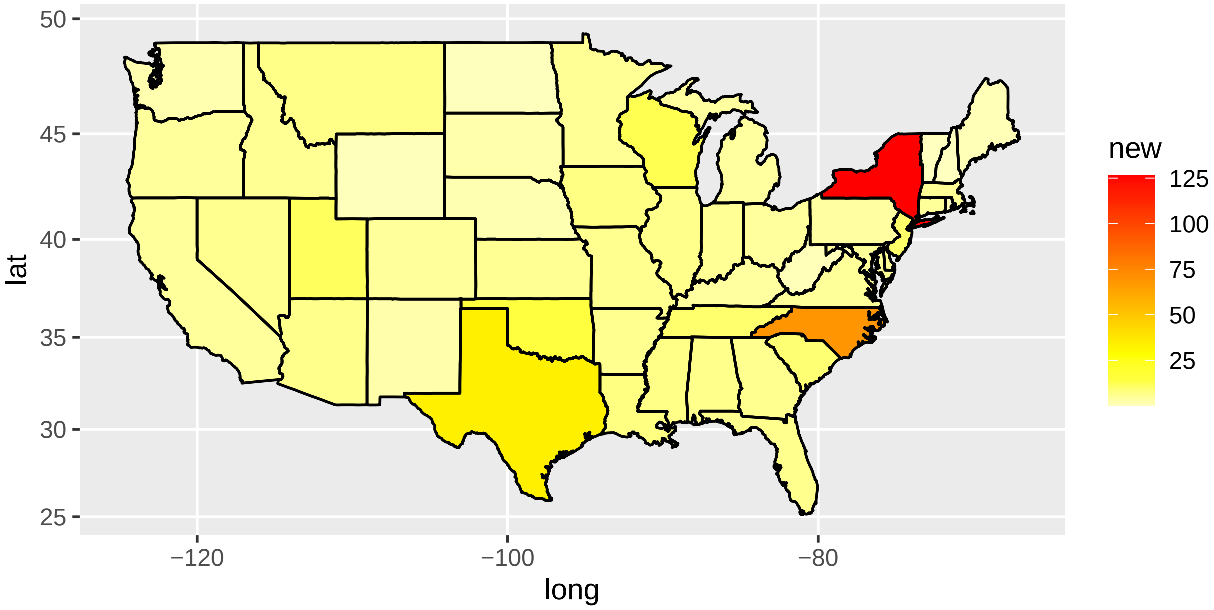
```
tot_deaths_div_pop$region <- tolower(tot_deaths_div_pop$Province_State)
library(ggplot2)
library(maps)
states <- map_data("state")
map.df <- merge(states,tot_deaths_div_pop, by="region", all.x=T)
map.df <- map.df[order(map.df$order),]
ggplot(map.df, aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=total_deaths))+
  geom_path()+
  scale_fill_gradientn(colours=rev(heat.colors(10)),na.value="grey90")+
  coord_map()
```



Similar to the analysis above, it makes sense that the states with the most cases would also have the most deaths.

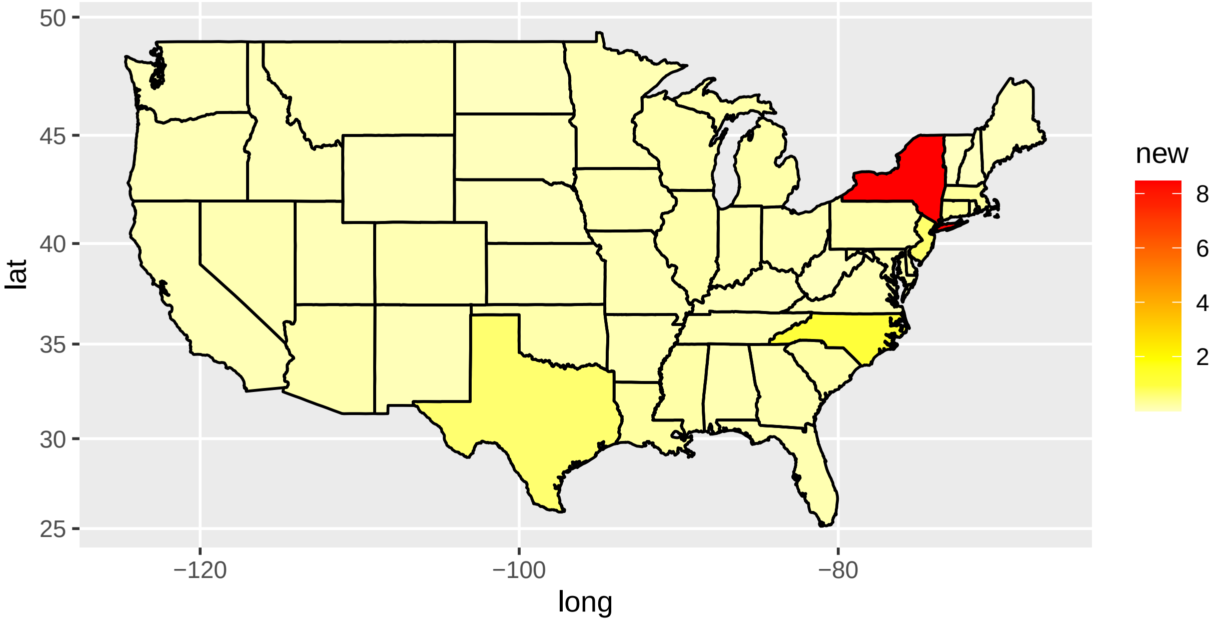
Heatmap of Total Confirmed Covid-19 Cases and Deaths in 2020 based on population

```
#Plot the graph for 2020
tot_div_pop_20$region <- tolower(tot_div_pop_20$Province_State)
library(ggplot2)
library(maps)
states <- map_data("state")
map.df <- merge(states,tot_div_pop_20, by="region", all.x=T)
map.df <- map.df[order(map.df$order),]
ggplot(map.df, aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=new))+
  geom_path()+
  scale_fill_gradientn(colours=rev(heat.colors(10)),na.value="grey90")+
  coord_map()
```



Deaths

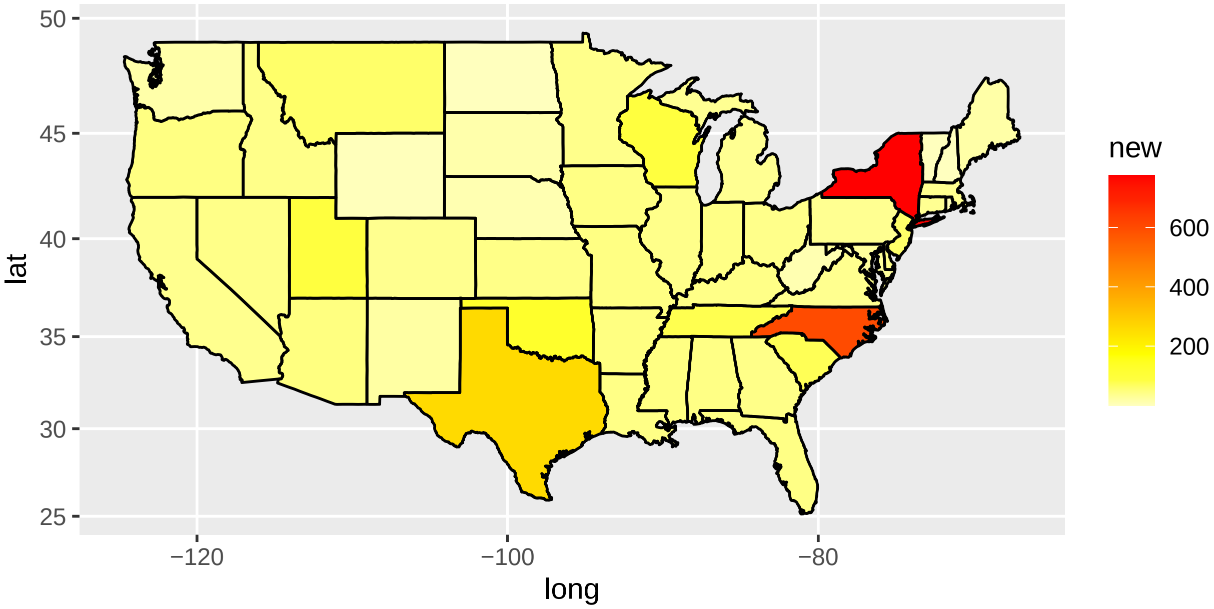
```
#Plot the graph for 2020
tot_deaths_div_pop_20$region <- tolower(tot_deaths_div_pop_20$Province_State)
library(ggplot2)
library(maps)
states <- map_data("state")
map.df <- merge(states,tot_deaths_div_pop_20, by="region", all.x=T)
map.df <- map.df[order(map.df$order),]
ggplot(map.df, aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=new))+
  geom_path()+
  scale_fill_gradientn(colours=rev(heat.colors(10)),na.value="grey90")+
  coord_map()
```



An interesting finding here is that North Carlolina had so many cases in proportion to their population. Even more fascinating, is that they seem to have less deaths than New York given the total number of cases.

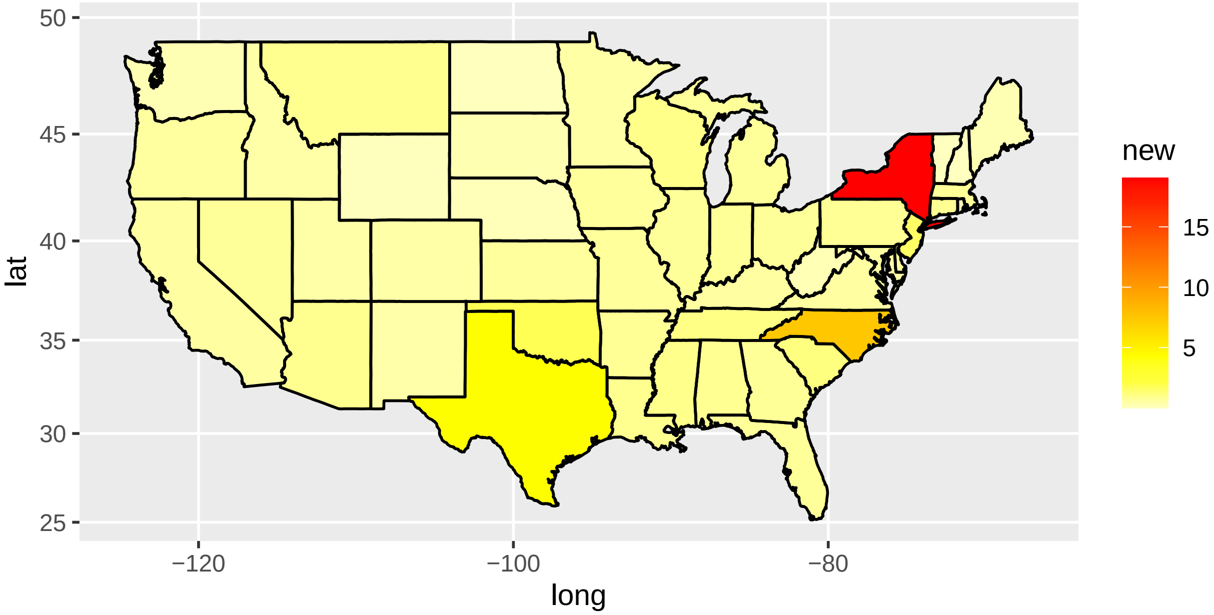
Heatmap of Total Confirmed Covid-19 Cases and Deaths in 2021 based on population

```r
#Plot the graph for 2021
tot_div_pop_21$region <- tolower(tot_div_pop_21$Province_State)
library(ggplot2)
library(maps)
states <- map_data("state")
map.df <- merge(states,tot_div_pop_21, by="region", all.x=T)
map.df <- map.df[order(map.df$order),]
ggplot(map.df, aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=new))+
  geom_path()+
  scale_fill_gradientn(colours=rev(heat.colors(10)),na.value="grey90")+
  coord_map()
```



Deaths

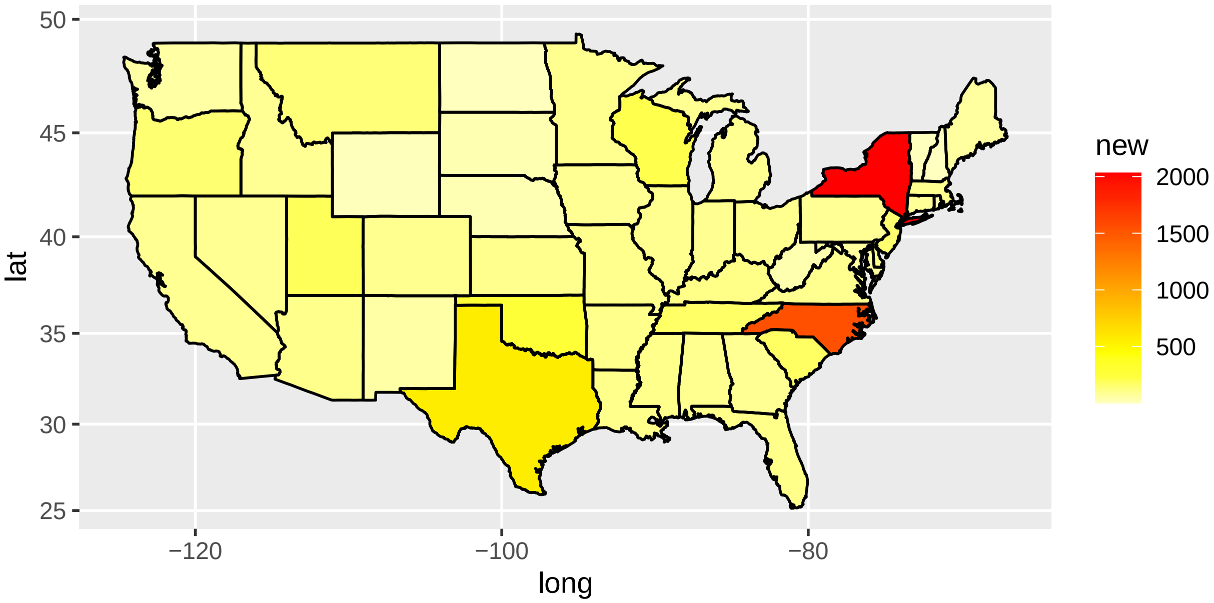
```r
#Plot the graph for 2020
tot_deaths_div_pop_21$region <- tolower(tot_deaths_div_pop_21$Province_State)
library(ggplot2)
library(maps)
states <- map_data("state")
map.df <- merge(states,tot_deaths_div_pop_21, by="region", all.x=T)
map.df <- map.df[order(map.df$order),]
ggplot(map.df, aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=new))+
  geom_path()+
  scale_fill_gradientn(colours=rev(heat.colors(10)),na.value="grey90")+
  coord_map()
```



Heat map looks very similar to 2020, but we can see some states starting to have more cases and deaths.

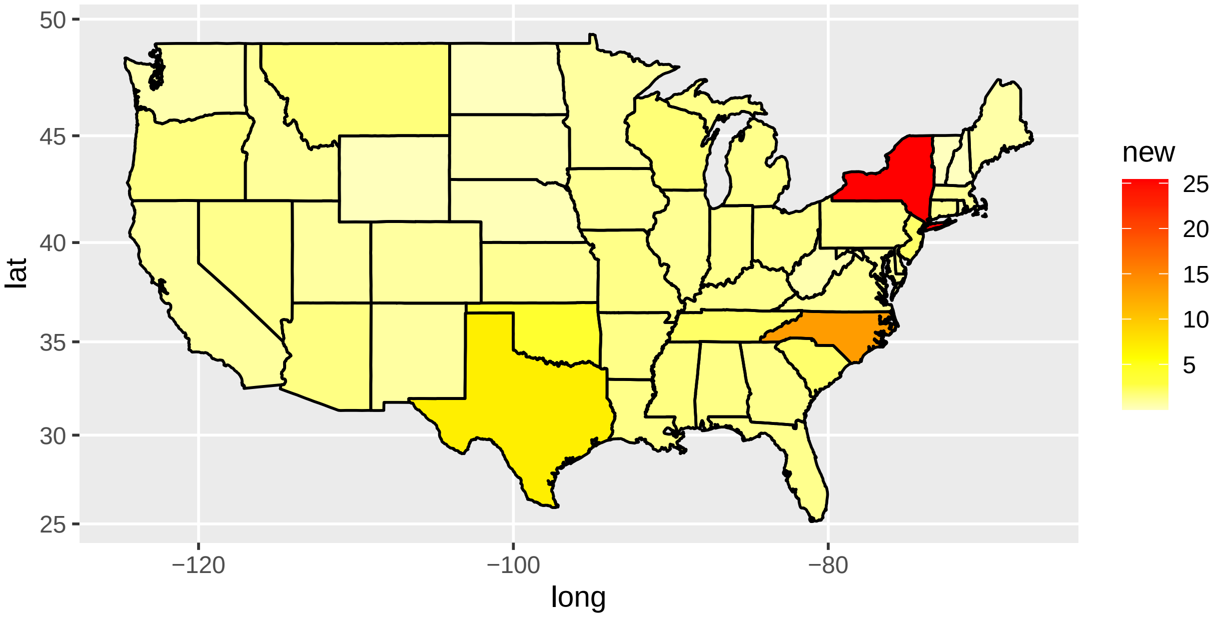
Heatmap of Total Confirmed Covid-19 Cases and Deaths in 2022 based on population

```
#Plot the graph for 2021
tot_div_pop_22$region <- tolower(tot_div_pop_22$Province_State)
library(ggplot2)
library(maps)
states <- map_data("state")
map.df <- merge(states,tot_div_pop_22, by="region", all.x=T)
map.df <- map.df[order(map.df$order),]
ggplot(map.df, aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=new))+
  geom_path()+
  scale_fill_gradientn(colours=rev(heat.colors(10)),na.value="grey90")+
  coord_map()
```



Deaths

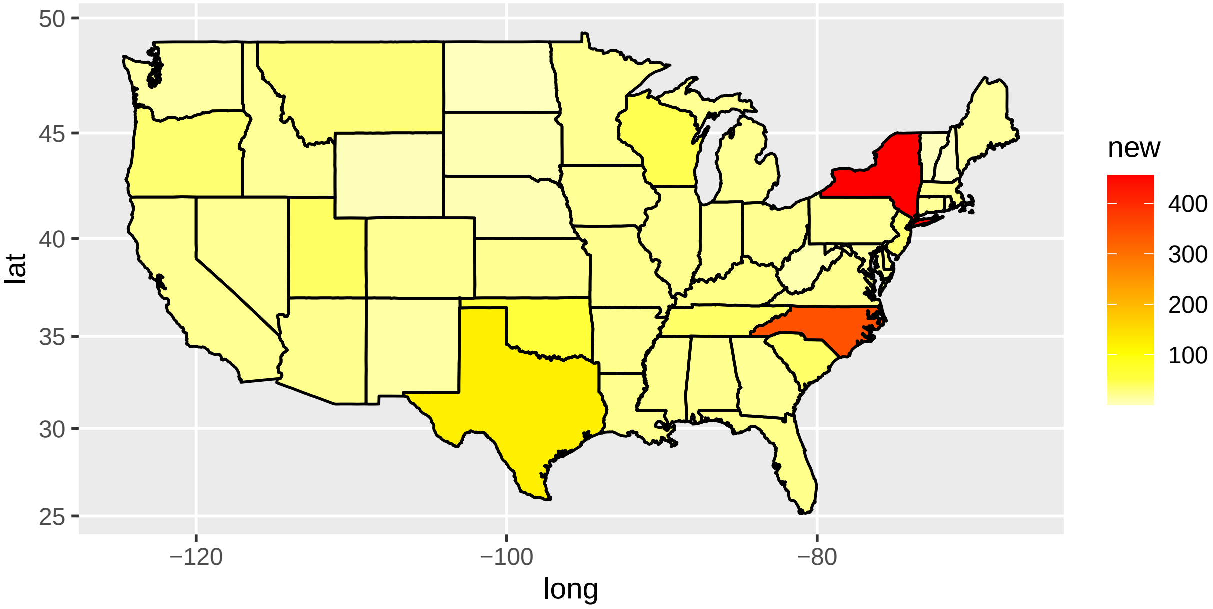
```
#Plot the graph for 2022
tot_deaths_div_pop_22$region <- tolower(tot_deaths_div_pop_22$Province_State)
library(ggplot2)
library(maps)
states <- map_data("state")
map.df <- merge(states,tot_deaths_div_pop_22, by="region", all.x=T)
map.df <- map.df[order(map.df$order),]
ggplot(map.df, aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=new))+
  geom_path()+
  scale_fill_gradientn(colours=rev(heat.colors(10)),na.value="grey90")+
  coord_map()
```



Very similar analysis as 2021.

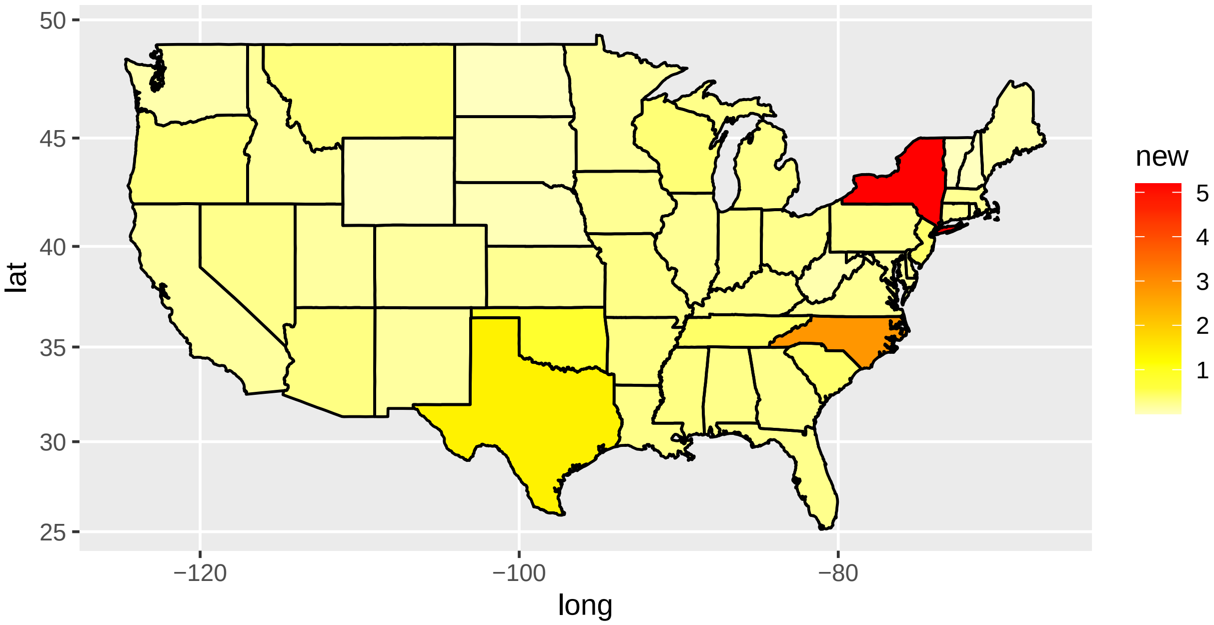
Heatmap of Total Confirmed Covid-19 Cases and Deaths in 2023 based on population

```
#Plot the graph for 2023
tot_div_pop_23$region <- tolower(tot_div_pop_23$Province_State)
library(ggplot2)
library(maps)
states <- map_data("state")
map.df <- merge(states,tot_div_pop_23, by="region", all.x=T)
map.df <- map.df[order(map.df$order),]
ggplot(map.df, aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=new))+
  geom_path()+
  scale_fill_gradientn(colours=rev(heat.colors(10)),na.value="grey90")+
  coord_map()
```



Deaths

```
#Plot the graph for 2023
tot_deaths_div_pop_23$region <- tolower(tot_deaths_div_pop_23$Province_State)
library(ggplot2)
library(maps)
states <- map_data("state")
map.df <- merge(states,tot_deaths_div_pop_23, by="region", all.x=T)
map.df <- map.df[order(map.df$order),]
ggplot(map.df, aes(x=long,y=lat,group=group))+
  geom_polygon(aes(fill=new))+
  geom_path()+
  scale_fill_gradientn(colours=rev(heat.colors(10)),na.value="grey90")+
  coord_map()
```



Although it looks very similar to the previous heat maps, it's important to look at the scale. The scale shows that the total cases and deaths have dropped.

Modeling

Create a linear model to get the correlation between the total cases divided by population an the total deaths divided by population.

```
case_deaths_pop <- merge(tot_deaths_div_pop,tot_div_pop, by="Province_State", all.x=T)
```

```
head(case_deaths_pop, n = 5)
```

```
##   Province_State total_deaths mean_cases.x    State.x    Pop.x    new.x
## 1        Alabama     18301446     18301446    Alabama  5044965 3.627666
## 2         Alaska      1492550      1492550     Alaska   733517 2.034786
## 3        Arizona     28068419     28068419    Arizona  7238881 3.877453
## 4       Arkansas     10739793     10739793   Arkansas  3024877 3.550489
## 5     California    105002525    105002525 California 39303058 2.671612
##     region.x total_cases mean_cases.y    State.y    Pop.y    new.y   region.y
## 1    alabama   872756073    872756073    Alabama  5044965 172.9955    alabama
## 2     alaska   153011898    153011898     Alaska   733517 208.6003     alaska
## 3    arizona  1330372436   1330372436    Arizona  7238881 183.7815    arizona
## 4   arkansas   549955573    549955573   Arkansas  3024877 181.8109   arkansas
## 5 california  6166190335   6166190335 California 39303058 156.8883 california
```

```
case_deaths_pop <- case_deaths_pop %>%
  select(c(new.x, new.y))
```

```
model <- lm(new.y ~ new.x, data = case_deaths_pop)
```

```
summary(model)
```

```
##
## Call:
## lm(formula = new.y ~ new.x, data = case_deaths_pop)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -320.16  -24.12  -14.51    6.08  615.91
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   23.465     16.261   1.443    0.155
## new.x         47.555      1.252  37.970   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 104.9 on 49 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.9671, Adjusted R-squared:  0.9665
## F-statistic:  1442 on 1 and 49 DF,  p-value: < 2.2e-16
```

By analyzing this data, we can see there is a strong correlation between deaths and covid cases when factoring in the population size of the states.

Bias & Conclusion

I had gone into this research expecting to see states with less restrictions during the pandemic to have more cases than states with more restrictions. However, even when factoring in the population size of the states I noticed that even states that had the most restriction (California and New York) still had a lot of Covid cases. Additionally, the one state that surprised me was North Carolina. Although it was hard to tell without the population being factored in, North Carolina had a lot of Covid cases in proportion to their population size.