COVID19

Thomas Bohn

2023-04-20

Introduction

Data Science Process

The following report follows the Data Science Process from beginning to end, ensuring there is a discussion on the following areas in the flow:

- Import
- Tidy
- Transform
- Visualize
- Model
- Communicate

Overview of Report Structure

The following report will contain the following sections:

- **Background**: Why should I care?
- **Data Source**: Where is your data from?
- **Tidying and Transform the Data**: How has the data been cleaned and transformed?
- **Analysis and Visualizations**: What does it tell you?
- **Models & Conclusions**: What do you conclude?
- **Review of Bias**: How could you be wrong?

By including comprehensive details in a well structured document, the results and findings of this analysis should be reproducible for any user.

R Libraries Utilized

The analysis in this report will utilize the following libraries in R for Data Analysis:

```
library(tidyverse)
library(lubridate)
library(rvest)
library(xml2)
library(car)
```

Background

What is COVID-19

> "Coronavirus disease 2019 (COVID-19) is a contagious disease caused by a virus, the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). The first known case was identified in Wuhan, China, in December 2019. The disease quickly spread worldwide, resulting in the COVID-19 pandemic.
>
> The symptoms of COVID-19 are variable but often include fever, cough, headache, fatigue, breathing difficulties, loss of smell, and loss of taste. Symptoms may begin one to fourteen days after exposure to the virus. At least a third of people who are infected do not develop noticeable symptoms. Of those who develop symptoms noticeable enough to be classified as patients, most (81%) develop mild to moderate symptoms (up to mild pneumonia), while 14% develop severe symptoms (dyspnea, hypoxia, or more than 50% lung involvement on imaging), and 5% develop critical symptoms (respiratory failure, shock, or multiorgan dysfunction). Older people are at a higher risk of developing severe symptoms. Some people continue to experience a range of effects (long COVID) for months after recovery, and damage to organs has been observed. Multi-year studies are underway to further investigate the long-term effects of the disease.[13]
>
> COVID-19 transmits when infectious particles are breathed in or come into contact with the eyes, nose, or mouth. The risk is highest when people are in close proximity, but small airborne particles containing the virus can remain suspended in the air and travel over longer distances, particularly indoors. Transmission can also occur when people touch their eyes, nose or mouth after touching surfaces or objects that have been contaminated by the virus. People remain contagious for up to 20 days and can spread the virus even if they do not develop symptoms."

see the the COVID-19 article on Wikipedia for more details on this disease.

Data Source

Source of Data

The data used for the COVID-19 analysis is sourced from the **COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University**. It can be found the following the following Github URL: https://github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data

This is the data repository for the 2019 Novel Corona virus Visual Dashboard operated by the Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE). Also, Supported by ESRI Living Atlas Team and the Johns Hopkins University Applied Physics Lab (JHU APL).

Note: On March 10, 2023, the Johns Hopkins Corona virus Resource Center ceased its collecting and reporting of global COVID-19 data.

Import Core Data

```
#Build URLs to Access the Data on Github
url_base <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/"

url_in <- str_c(url_base, "csse_covid_19_data/csse_covid_19_time_series/")

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)

url_in_uid <- str_c(url_base, "csse_covid_19_data/")
file_names_uid <- "UID_ISO_FIPS_LookUp_Table.csv"
url_uid <- str_c(url_in_uid,file_names_uid)

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_
```

```
url_uid
```

```
## [1] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/UID_ISO_FIPS
```

```
#Read in the data to data sets in R
cases_us      <- read_csv(urls[1])
cases_global  <- read_csv(urls[2])
deaths_us     <- read_csv(urls[3])
deaths_global <- read_csv(urls[4])
uid           <- read_csv(url_uid)
```

```
#Preview the dataset
cases_us
```

```
## # A tibble: 3,342 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 3,332 more rows
## # 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>, ...
```

```
cases_global
```

```
## # A tibble: 289 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 <NA>              Afghanistan       33.9  67.7          0         0         0
##  2 <NA>              Albania           41.2  20.2          0         0         0
##  3 <NA>              Algeria           28.0   1.66         0         0         0
##  4 <NA>              Andorra           42.5   1.52         0         0         0
##  5 <NA>              Angola           -11.2  17.9          0         0         0
##  6 <NA>              Antarctica       -71.9  23.3          0         0         0
##  7 <NA>              Antigua and Bar~  17.1 -61.8          0         0         0
##  8 <NA>              Argentina        -38.4 -63.6          0         0         0
##  9 <NA>              Armenia           40.1  45.0          0         0         0
## 10 Australian Capit~ Australia        -35.5 149.           0         0         0
## # i 279 more rows
## # 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>, ...
```

```
deaths_us
```

```
## # A tibble: 3,342 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 3,332 more rows
## # 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>, ...
```

```
deaths_global
```

```
## # A tibble: 289 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 <NA>              Afghanistan       33.9  67.7          0         0         0
```

```
##  2 <NA>              Albania           41.2  20.2          0          0          0
##  3 <NA>              Algeria           28.0   1.66         0          0          0
##  4 <NA>              Andorra           42.5   1.52         0          0          0
##  5 <NA>              Angola           -11.2  17.9          0          0          0
##  6 <NA>              Antarctica       -71.9  23.3          0          0          0
##  7 <NA>              Antigua and Bar~  17.1 -61.8          0          0          0
##  8 <NA>              Argentina        -38.4 -63.6          0          0          0
##  9 <NA>              Armenia           40.1  45.0          0          0          0
## 10 Australian Capit~ Australia        -35.5 149.           0          0          0
## # i 279 more rows
## # 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>, ...
```

```
uid
```

```
## # A tibble: 4,321 x 12
##      UID iso2  iso3  code3 FIPS  Admin2 Province_State Country_Region        Lat
##    <dbl> <chr> <chr> <dbl> <chr> <chr>  <chr>          <chr>                <dbl>
##  1     4 AF    AFG       4 <NA>  <NA>   <NA>           Afghanistan           33.9
##  2     8 AL    ALB       8 <NA>  <NA>   <NA>           Albania               41.2
##  3    10 AQ    ATA      10 <NA>  <NA>   <NA>           Antarctica           -71.9
##  4    12 DZ    DZA      12 <NA>  <NA>   <NA>           Algeria               28.0
##  5    20 AD    AND      20 <NA>  <NA>   <NA>           Andorra               42.5
##  6    24 AO    AGO      24 <NA>  <NA>   <NA>           Angola               -11.2
##  7    28 AG    ATG      28 <NA>  <NA>   <NA>           Antigua and Barbuda   17.1
##  8    32 AR    ARG      32 <NA>  <NA>   <NA>           Argentina            -38.4
##  9    51 AM    ARM      51 <NA>  <NA>   <NA>           Armenia               40.1
## 10    40 AT    AUT      40 <NA>  <NA>   <NA>           Austria               47.5
## # i 4,311 more rows
## # i 3 more variables: Long_ <dbl>, Combined_Key <chr>, Population <dbl>
```

Import Party Affiliation Data

```
#Define the URL of the webpage that needs to be scraped for party data
url <- "https://www.pewresearch.org/religion/religious-landscape-study/compare/party-affiliation/by/sta
#head the html page and extract the table containing the data from the page
party_aff <- url %>%
  read_html() %>%
  html_nodes(xpath='//*[@id="page-23474"]/div[2]/section/div[3]/table') %>%
  html_table()
#strip the table around the table
party_aff <- party_aff[[1]]
#cast the data frame as a tibble and repair the table names
party_aff <- as_tibble(party_aff, .name_repair = make.names)
party_aff <- party_aff %>%
  #rename the columns to shorter names
  rename(
    state = "State",
```

```
    rep = "Republican.lean.Rep.",
    no_lean = "No.lean",
    dem = "Democrat.lean.Dem.",
    sample_size = "Sample.size"
  ) %>%
  #assign the numeric columns as integers
  mutate(
    rep = as.integer(parse_number(rep)),
    no_lean = as.integer(parse_number(no_lean)),
    dem = as.integer(parse_number(dem)),
    sample_size = as.integer(parse_number(sample_size))
  ) %>%
  #assign the state to a factor
  mutate(
    state = factor(state)
  )
#preview the table
party_aff
```

```
## # A tibble: 51 x 5
##    state                 rep no_lean   dem sample_size
##    <fct>               <int>   <int> <int>       <int>
##  1 Alabama                52      13    35         511
##  2 Alaska                 39      29    32         310
##  3 Arizona                40      21    39         653
##  4 Arkansas               46      16    38         311
##  5 California             30      21    49        3697
##  6 Colorado               41      17    42         504
##  7 Connecticut            32      18    50         377
##  8 Delaware               29      17    55         301
##  9 District of Columbia   11      15    73         303
## 10 Florida                37      19    44        2020
## # i 41 more rows
```

```
str(party_aff)
```

```
## tibble [51 x 5] (S3: tbl_df/tbl/data.frame)
##  $ state      : Factor w/ 51 levels "Alabama","Alaska",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ rep        : int [1:51] 52 39 40 46 30 41 32 29 11 37 ...
##  $ no_lean    : int [1:51] 13 29 21 16 21 17 18 17 15 19 ...
##  $ dem        : int [1:51] 35 32 39 38 49 42 50 55 73 44 ...
##  $ sample_size: int [1:51] 511 310 653 311 3697 504 377 301 303 2020 ...
```

Tidying the Data

The following outlines how the data was modified to be tidy and transformed to contain variables for further analysis. This section contains:

- A summary of the data
- Clean up of the dataset by changing appropriate variables to factors, updating date types, and getting rid of any columns not needed
- Transforming the data to add useful variables and derived elements
- Summary of the data to be sure there is no missing data

Data Summerization

Given the extremely wide nature of the data, no additional summarization results will be displayed here in the report. Commented out code is included for completeness, but outputs extremely long results.

```
#Show the structure of the datasets
table_names <- c(cases_us, cases_global, deaths_us, deaths_global, uid)
```

```
#Show the structure of the datasets
#str(cases_us)
#str(cases_global)
#str(deaths_us)
#str(deaths_global)
#str(uid)
```

```
#Summary of the dataset
#summary(cases_us)
#summary(cases_global)
#summary(deaths_us)
#summary(deaths_global)
#summary(uid)
```

```
#Show the column names of the columns in datasets
#str_to_lower(colnames(cases_us))
#str_to_lower(colnames(cases_global))
#str_to_lower(colnames(deaths_us))
#str_to_lower(colnames(deaths_global))
#str_to_lower(colnames(uid))
```

Scope for Initial Tidy & Transform

List of initial tidy adjustments to make:

- Shape data to vastly reduce the number of columns in each dataset and make the datasets longer (versus their current wide configuration)
- Drop columns that will not be needed for analysis
- Rename columns and update data types (especially for integers and dates)
- Join data to consolidate data elements into a table for US and Global

Tidy & Transform of the UID Data

```
#Remove columns not needed for analysis
tidy_uid <- uid %>%
  select(-c("Lat","Long_", "iso2", "iso3", "code3", "Admin2"))
head(tidy_uid, n=3)
```

```
## # A tibble: 3 x 6
##     UID FIPS  Province_State Country_Region Combined_Key Population
##   <dbl> <chr> <chr>          <chr>          <chr>             <dbl>
## 1     4 <NA>  <NA>           Afghanistan    Afghanistan    38928341
## 2     8 <NA>  <NA>           Albania        Albania         2877800
## 3    10 <NA>  <NA>           Antarctica     Antarctica           NA
```

Tidy & Transform of the Cases Global and Deaths Global Data

```
#Tidy Cases Global
tidy_cases_global <- cases_global %>%
  pivot_longer(cols = -c('Province/State', 'Country/Region', 'Lat', 'Long'),
               names_to = "Date",
               values_to = "Cases") %>%
  select(-c("Lat","Long"))
head(tidy_cases_global, n=3)
```

```
## # A tibble: 3 x 4
##   `Province/State` `Country/Region` Date    Cases
##   <chr>            <chr>            <chr>   <dbl>
## 1 <NA>             Afghanistan      1/22/20     0
## 2 <NA>             Afghanistan      1/23/20     0
## 3 <NA>             Afghanistan      1/24/20     0
```

```
#Tidy Deaths Global
tidy_deaths_global <- deaths_global %>%
  pivot_longer(cols = -c('Province/State', 'Country/Region', 'Lat', 'Long'),
               names_to = "Date",
               values_to = "Deaths") %>%
  select(-c("Lat","Long"))
head(tidy_deaths_global, n=3)
```

```
## # A tibble: 3 x 4
##   `Province/State` `Country/Region` Date    Deaths
##   <chr>            <chr>            <chr>    <dbl>
## 1 <NA>             Afghanistan      1/22/20      0
## 2 <NA>             Afghanistan      1/23/20      0
## 3 <NA>             Afghanistan      1/24/20      0
```

```
#Combine Global Deaths and Cases
tidy_global <- tidy_cases_global %>%
  full_join(tidy_deaths_global) %>%
  rename(Country_Region = `Country/Region`,
         Province_State = `Province/State`) %>%
  mutate(Date = mdy(Date)) %>%
  filter(Cases > 0)
```

```
## Joining with `by = join_by(`Province/State`, `Country/Region`, Date)`
```

```
head(tidy_global, n=3)
```

```
## # A tibble: 3 x 5
##   Province_State Country_Region Date       Cases Deaths
##   <chr>          <chr>          <date>     <dbl>  <dbl>
## 1 <NA>           Afghanistan    2020-02-24     5      0
## 2 <NA>           Afghanistan    2020-02-25     5      0
## 3 <NA>           Afghanistan    2020-02-26     5      0
```

```
#Add Population and Combined Key to Global Data Set
tidy_global <- tidy_global %>%
  left_join(uid, by = c("Province_State", "Country_Region")) %>%
  select(-c(UID, FIPS)) %>%
  select(Province_State, Country_Region, Date, Cases,
         Deaths, Population, Combined_Key)
head(tidy_global, n=3)
```

```
## # A tibble: 3 x 7
##   Province_State Country_Region Date       Cases Deaths Population Combined_Key
##   <chr>          <chr>          <date>     <dbl>  <dbl>      <dbl> <chr>
## 1 <NA>           Afghanistan    2020-02-24     5      0   38928341 Afghanistan
## 2 <NA>           Afghanistan    2020-02-25     5      0   38928341 Afghanistan
## 3 <NA>           Afghanistan    2020-02-26     5      0   38928341 Afghanistan
```

Validate the Tidy & Transform for Global Cases and Deaths

```
#Display summary of global dataset
summary(tidy_global)
```

```
##  Province_State     Country_Region          Date                Cases
##  Length:306827      Length:306827      Min.   :2020-01-22   Min.   :        1
##  Class :character   Class :character   1st Qu.:2020-12-12   1st Qu.:     1316
##  Mode  :character   Mode  :character   Median :2021-09-16   Median :    20365
##                                        Mean   :2021-09-11   Mean   :  1032863
##                                        3rd Qu.:2022-06-15   3rd Qu.:   271281
##                                        Max.   :2023-03-09   Max.   :103802702
##
##      Deaths          Population        Combined_Key
##  Min.   :      0   Min.   :6.700e+01   Length:306827
##  1st Qu.:      7   1st Qu.:7.866e+05   Class :character
##  Median :    214   Median :6.948e+06   Mode  :character
##  Mean   :  14405   Mean   :2.890e+07
##  3rd Qu.:   3665   3rd Qu.:2.914e+07
##  Max.   :1123836   Max.   :1.380e+09
##                    NA's   :6729
```

```
#Display structure of global dataset
str(tidy_global)
```

```
## tibble [306,827 x 7] (S3: tbl_df/tbl/data.frame)
##  $ Province_State: chr [1:306827] NA NA NA NA ...
##  $ Country_Region: chr [1:306827] "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...
##  $ Date          : Date[1:306827], format: "2020-02-24" "2020-02-25" ...
##  $ Cases         : num [1:306827] 5 5 5 5 5 5 5 5 5 5 ...
##  $ Deaths        : num [1:306827] 0 0 0 0 0 0 0 0 0 0 ...
##  $ Population    : num [1:306827] 38928341 38928341 38928341 38928341 38928341 ...
##  $ Combined_Key  : chr [1:306827] "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...
```

```
#Display results of Global Data Set
colnames(tidy_global)
```

```
## [1] "Province_State" "Country_Region" "Date"           "Cases"
## [5] "Deaths"         "Population"     "Combined_Key"
```

Tidy & Transform for the US Cases and Deaths Data

```
#Tidy US Cases Data
tidy_cases_us <- cases_us %>%
  pivot_longer(cols = -c('UID', 'iso2', 'iso3', 'code3', 'FIPS', 'Admin2',
                         'Province_State', 'Country_Region', 'Lat',
                         'Long_', 'Combined_Key'),
               names_to = "Date",
               values_to = "Cases") %>%
  select(-c('UID', 'iso2', 'iso3', 'code3', 'FIPS')) %>%
  select(-c("Lat","Long_")) %>%
  mutate(Date = lubridate::mdy(Date))
head(tidy_cases_us, n=3)
```

```
## # A tibble: 3 x 6
##   Admin2  Province_State Country_Region Combined_Key         Date       Cases
##   <chr>   <chr>          <chr>          <chr>                <date>     <dbl>
## 1 Autauga Alabama        US             Autauga, Alabama, US 2020-01-22     0
## 2 Autauga Alabama        US             Autauga, Alabama, US 2020-01-23     0
## 3 Autauga Alabama        US             Autauga, Alabama, US 2020-01-24     0
```

```
#Tidy US Deaths Data)
tidy_deaths_us <- deaths_us %>%
  pivot_longer(cols = -c('UID', 'iso2', 'iso3', 'code3', 'FIPS', 'Admin2',
                         'Province_State', 'Country_Region', 'Lat',
                         'Long_', 'Combined_Key', 'Population'),
               names_to = "Date",
               values_to = "Deaths") %>%
  select(-c('UID', 'iso2', 'iso3', 'code3', 'FIPS')) %>%
  select(-c("Lat","Long_")) %>%
  mutate(Date = lubridate::mdy(Date))
head(tidy_deaths_us, n=3)
```

```
## # A tibble: 3 x 7
##   Admin2 Province_State Country_Region Combined_Key Population Date       Deaths
##   <chr>  <chr>          <chr>          <chr>             <dbl> <date>      <dbl>
## 1 Autau~ Alabama        US             Autauga, Al~      55869 2020-01-22      0
## 2 Autau~ Alabama        US             Autauga, Al~      55869 2020-01-23      0
## 3 Autau~ Alabama        US             Autauga, Al~      55869 2020-01-24      0
```

```
#Combine US cases and deaths data into one dataset
tidy_us <- tidy_cases_us %>%
  full_join(tidy_deaths_us) %>%
  rename(County = `Admin2`) %>%
  filter(Cases > 0)
```

```
## Joining with `by = join_by(Admin2, Province_State, Country_Region,
## Combined_Key, Date)`
```

```
head(tidy_us, n=3)
```

```
## # A tibble: 3 x 8
##   County  Province_State Country_Region Combined_Key Date       Cases Population
##   <chr>   <chr>          <chr>          <chr>        <date>     <dbl>      <dbl>
## 1 Autauga Alabama        US             Autauga, Al~ 2020-03-24     1      55869
## 2 Autauga Alabama        US             Autauga, Al~ 2020-03-25     5      55869
## 3 Autauga Alabama        US             Autauga, Al~ 2020-03-26     6      55869
## # i 1 more variable: Deaths <dbl>
```

Validate the Tidy & Transform for US Cases and Deaths

```
#Display summary of us dataset
summary(tidy_us)
```

```
##     County          Province_State     Country_Region     Combined_Key
##  Length:3474292     Length:3474292     Length:3474292     Length:3474292
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##       Date                Cases           Population          Deaths
##  Min.   :2020-01-22   Min.   :      1   Min.   :       0   Min.   :    0.0
##  1st Qu.:2020-12-27   1st Qu.:    687   1st Qu.:   10953   1st Qu.:   10.0
##  Median :2021-09-20   Median :   2849   Median :   26248   Median :   47.0
##  Mean   :2021-09-19   Mean   :  15489   Mean   :  104502   Mean   :  205.1
##  3rd Qu.:2022-06-15   3rd Qu.:   9345   3rd Qu.:   68098   3rd Qu.:  137.0
##  Max.   :2023-03-09   Max.   :3710586   Max.   :10039107   Max.   :35545.0
```

```
#Display structure of us dataset
str(tidy_us)
```

```
## tibble [3,474,292 x 8] (S3: tbl_df/tbl/data.frame)
##  $ County        : chr [1:3474292] "Autauga" "Autauga" "Autauga" "Autauga" ...
##  $ Province_State: chr [1:3474292] "Alabama" "Alabama" "Alabama" "Alabama" ...
##  $ Country_Region: chr [1:3474292] "US" "US" "US" "US" ...
##  $ Combined_Key  : chr [1:3474292] "Autauga, Alabama, US" "Autauga, Alabama, US" "Autauga, Alabama, U
##  $ Date          : Date[1:3474292], format: "2020-03-24" "2020-03-25" ...
##  $ Cases         : num [1:3474292] 1 5 6 6 6 6 8 8 10 12 ...
##  $ Population    : num [1:3474292] 55869 55869 55869 55869 55869 ...
##  $ Deaths        : num [1:3474292] 0 0 0 0 0 0 0 0 0 0 ...
```

```
#Display results of us Data Set
colnames(tidy_us)
```

```
## [1] "County"         "Province_State" "Country_Region" "Combined_Key"
## [5] "Date"           "Cases"          "Population"     "Deaths"
```

Analysis and Visualizations

Through analysis and visualization, I would like to look at factors and trends that influence COVID19 cases and deaths. In order to better understand the factors that contribute to the global pandemics. I'd like to do some analysis around the following areas:

- What Does the Trend of Cases and Deaths look like overall for the US?
- What Does the Trend of Cases and Deaths look like overall for Illinois?
- What is the Largest Total Deaths and Date in the covid19 in the US Plot?
- How do New Deaths and New Cases Trend Over Time in the US?
- How do New Deaths and New Cases Trend Over Time for the State of Illinois?
- Create a List of the Top 10 Best and Worst State for covid19 Deaths per Thousand People?

What Does the Trend of Cases and Deaths look like overall for the US?

```
#Create a US by State View
us_by_state <- tidy_us %>%
  group_by(Province_State, Country_Region, Date) %>%
  summarize(
    Cases = sum(Cases),
    Deaths = sum(Deaths),
    Population = sum(Population)
  ) %>%
  mutate(Deaths_Per_Mill = Deaths * 1000000 / Population) %>%
  select(Province_State, Country_Region, Date, Cases,
         Deaths, Deaths_Per_Mill, Population) %>%
  ungroup()
```

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

```
head(us_by_state, n = 3)
```

```
## # A tibble: 3 x 7
##   Province_State Country_Region Date       Cases Deaths Deaths_Per_Mill
##   <chr>          <chr>          <date>     <dbl>  <dbl>           <dbl>
## 1 Alabama        US             2020-03-11     3      0               0
## 2 Alabama        US             2020-03-12     4      0               0
## 3 Alabama        US             2020-03-13     8      0               0
## # i 1 more variable: Population <dbl>
```

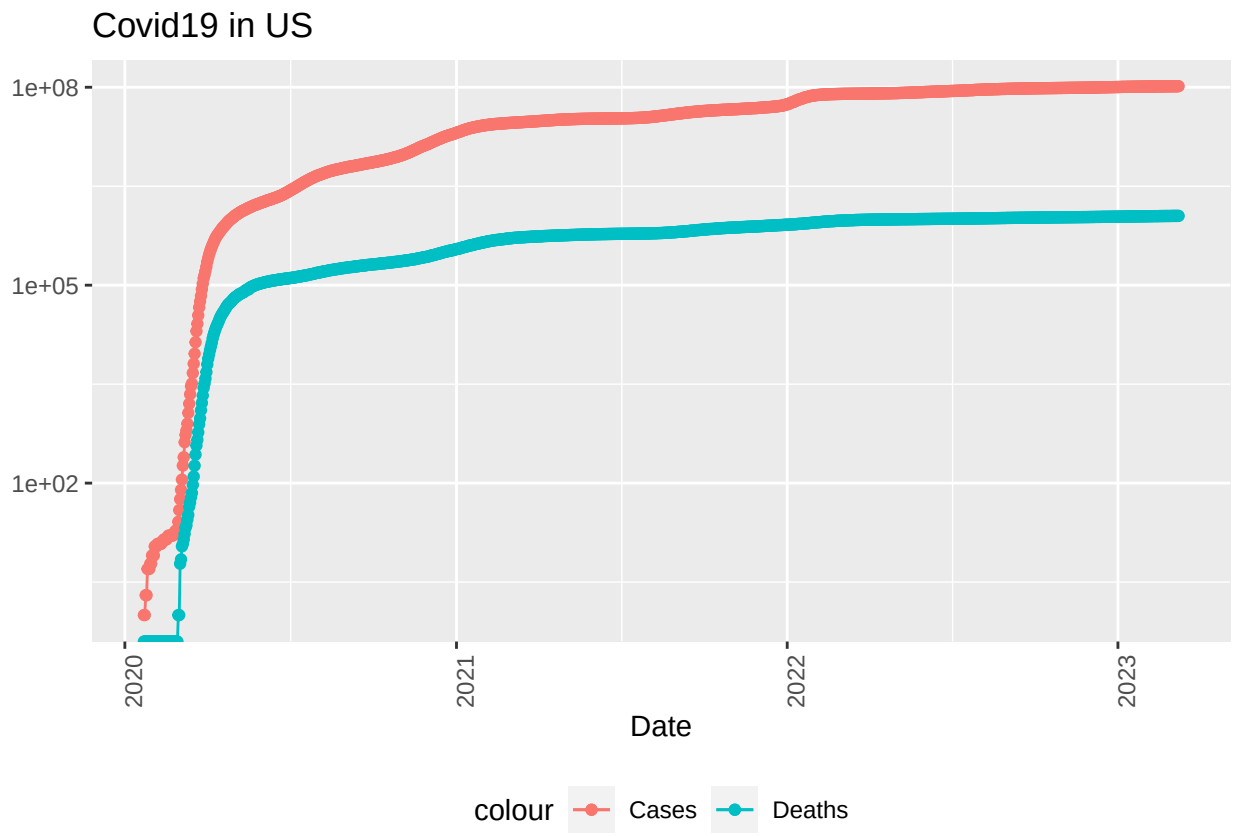
```
#Create a US Total View
us_totals <- us_by_state %>%
  group_by(Country_Region, Date) %>%
  summarize(Cases = sum(Cases), Deaths = sum(Deaths),
            Population = sum(Population)) %>%
  mutate(Deaths_Per_Mill = Deaths * 1000000 / Population) %>%
  select(Country_Region, Date, Cases, Deaths, Deaths_Per_Mill, Population) %>%
  ungroup()
```

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

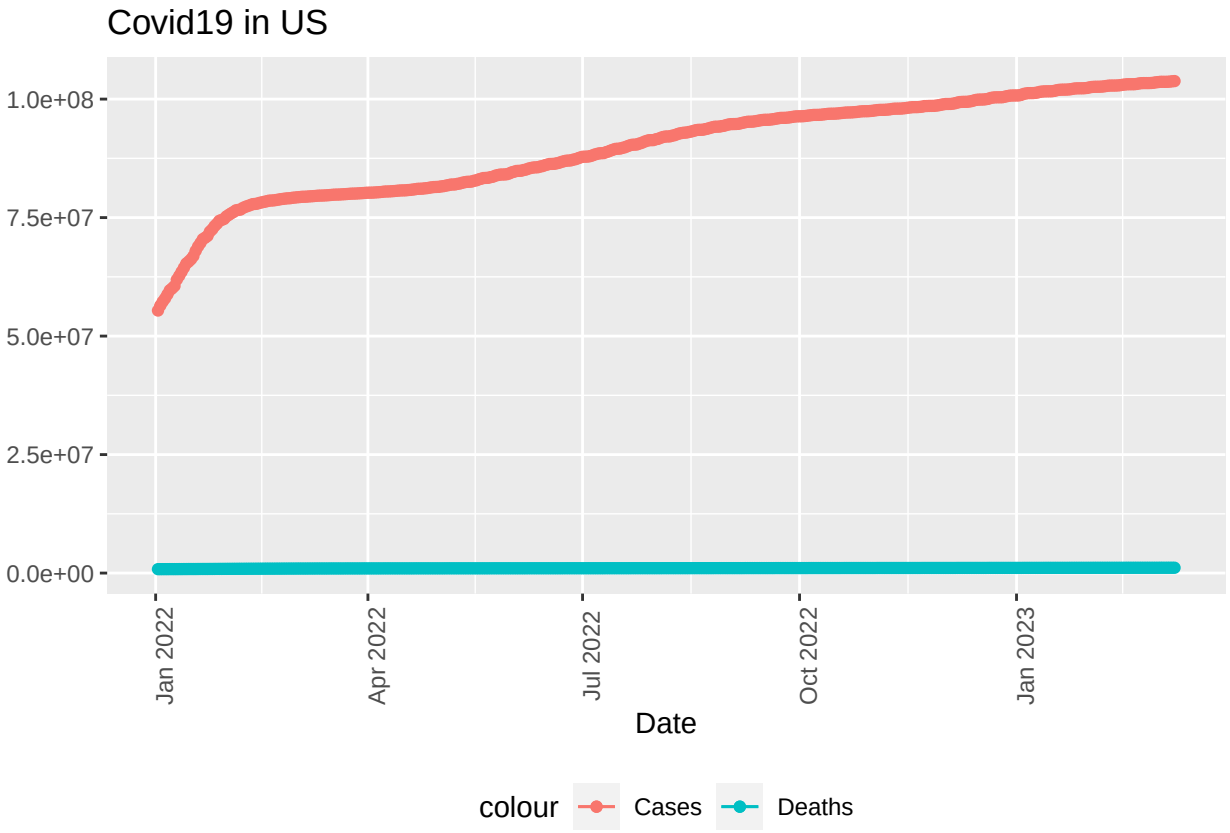
```
head(us_by_state, n = 3)
```

```
## # A tibble: 3 x 7
##   Province_State Country_Region Date       Cases Deaths Deaths_Per_Mill
##   <chr>          <chr>          <date>     <dbl>  <dbl>           <dbl>
## 1 Alabama        US             2020-03-11     3      0               0
## 2 Alabama        US             2020-03-12     4      0               0
## 3 Alabama        US             2020-03-13     8      0               0
## # i 1 more variable: Population <dbl>
```

```
#Create US Totals Visualization
us_totals %>%
  filter(Cases > 0) %>%
  ggplot(aes(x = Date, y = Cases)) +
  geom_line(aes(color = "Cases")) +
  geom_point(aes(color = "Cases")) +
  geom_line(aes(y = Deaths, color = "Deaths")) +
  geom_point(aes(y = Deaths, color = "Deaths")) +
  scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "Covid19 in US", y = NULL)
```



```r
#Create US Totals Visualization
us_totals %>%
  filter(Cases > 0 & Date > "2022-01-01") %>%
  ggplot(aes(x = Date, y = Cases)) +
  geom_line(aes(color = "Cases")) +
  geom_point(aes(color = "Cases")) +
  geom_line(aes(y = Deaths, color = "Deaths")) +
  geom_point(aes(y = Deaths, color = "Deaths")) +
  #scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "Covid19 in US", y = NULL)
```

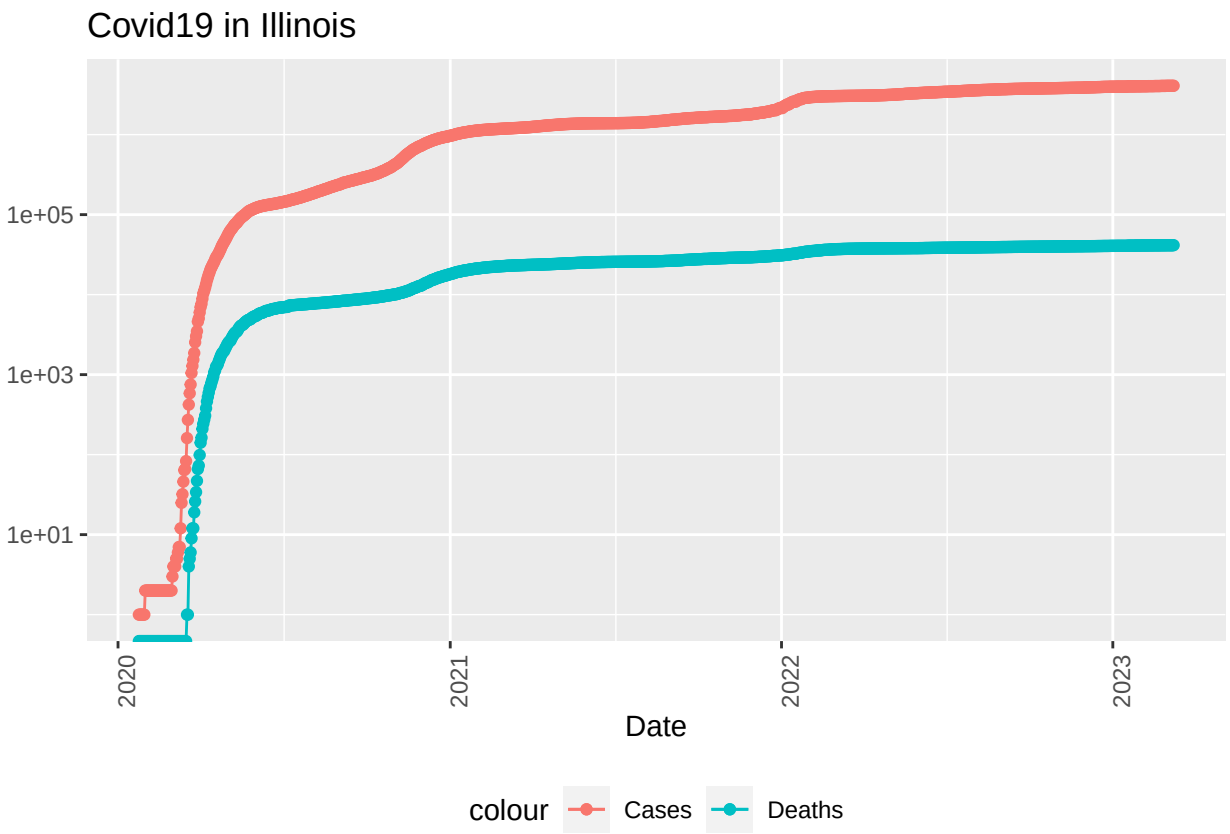


Conclusion: This plot displays the cumulative total for the US. Given the extremely large number of cases and the log scale, it is hard to tell for recent data how much the chart is increasing and if cases and deaths are going up on a daily basis. Overall, it displays that there was a sharp increase initially, but then cases began to taper off and grow slower than exponential. Looking at the zoomed in chart on 2022 (with the log scale removed, we see growth that looks more linear than exponential.

What Does the Trend of Cases and Deaths look like overall for Illinois?

```r
#Filter for New York and Create State Visualization
state <- "Illinois"
us_by_state %>%
```

```
  filter(Province_State == state) %>%
  filter(Cases > 0) %>%
  ggplot(aes(x = Date, y = Cases)) +
  geom_line(aes(color = "Cases")) +
  geom_point(aes(color = "Cases")) +
  geom_line(aes(y = Deaths, color = "Deaths")) +
  geom_point(aes(y = Deaths, color = "Deaths")) +
  scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = str_c("Covid19 in ", state), y = NULL)
```



Conclusion: Comparing Illinois to the US totals, we see a similar pattern. Extream growth of cases initially, then it tapers off on the log scale graph. Overall, the macro patterns look the same for both.

What is the Largest Total Deaths and Date in the Covid in the US Plot?

```
max(us_totals$Date)
```

```
## [1] "2023-03-09"
```

```
max(us_totals$Deaths)
```

```
## [1] 1122724
```

Conclusion: The largest data point for the US is 1,122,724 total cases and occurs on 2023-03-09.

How do New Deaths and New Cases Trend Over Time in the US?

```
#Add New Cases and New Deaths calculated field to the US state view
us_by_state <- us_by_state %>%
  mutate(New_Cases = Cases - lag(Cases),
       New_Deaths = Deaths - lag(Deaths))
#Add New Cases and New Deaths calculated field to the US totals view
us_totals <- us_totals %>%
  mutate(New_Cases = Cases - lag(Cases),
       New_Deaths = Deaths - lag(Deaths))
```

```
#Display the changes to the two tables
tail(us_by_state %>% select(New_Cases, New_Deaths, everything()))
```

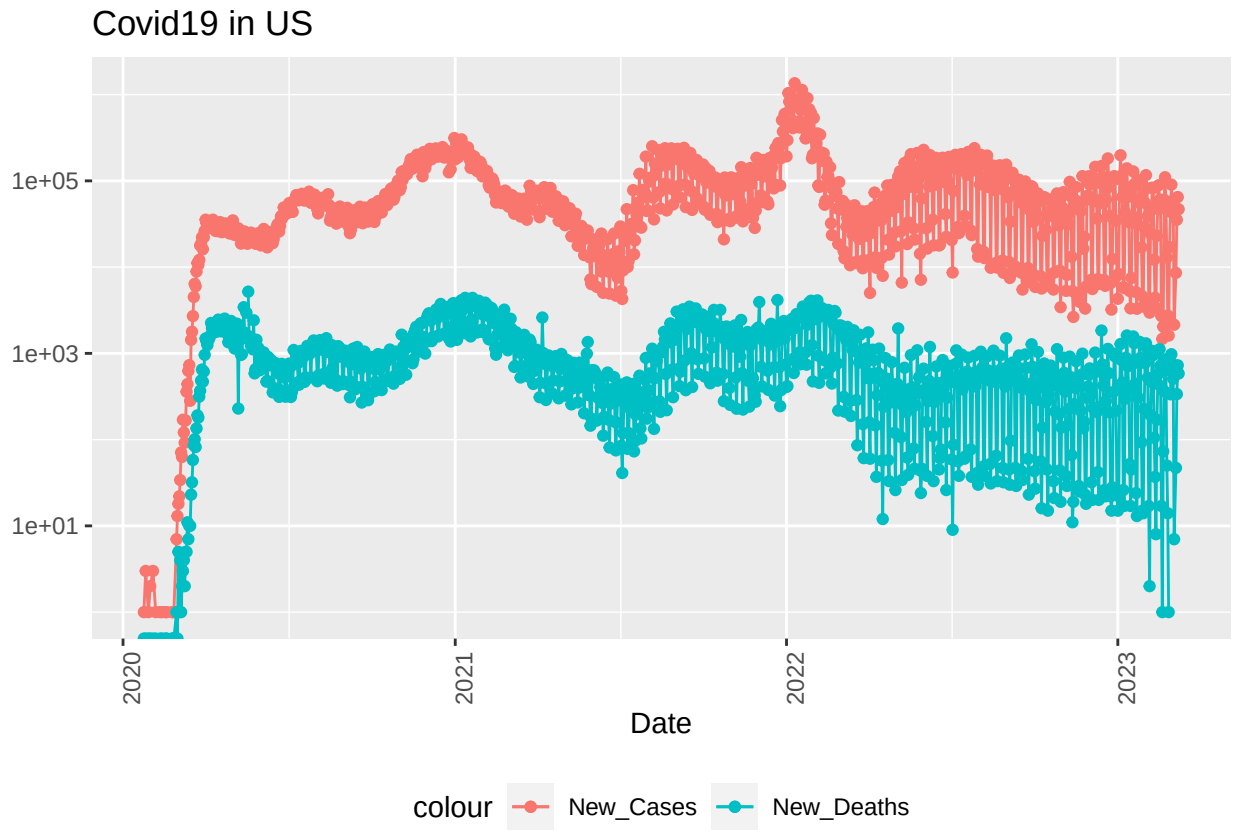
```
## # A tibble: 6 x 9
##   New_Cases New_Deaths Province_State Country_Region Date        Cases Deaths
##       <dbl>      <dbl> <chr>          <chr>          <date>      <dbl>  <dbl>
## 1         0          0 Wyoming        US             2023-03-04 185159   2002
## 2         0          0 Wyoming        US             2023-03-05 185159   2002
## 3         0          0 Wyoming        US             2023-03-06 185159   2002
## 4       226          2 Wyoming        US             2023-03-07 185385   2004
## 5         0          0 Wyoming        US             2023-03-08 185385   2004
## 6         0          0 Wyoming        US             2023-03-09 185385   2004
## # i 2 more variables: Deaths_Per_Mill <dbl>, Population <dbl>
```

```
tail(us_totals %>% select(New_Cases, New_Deaths, everything()))
```

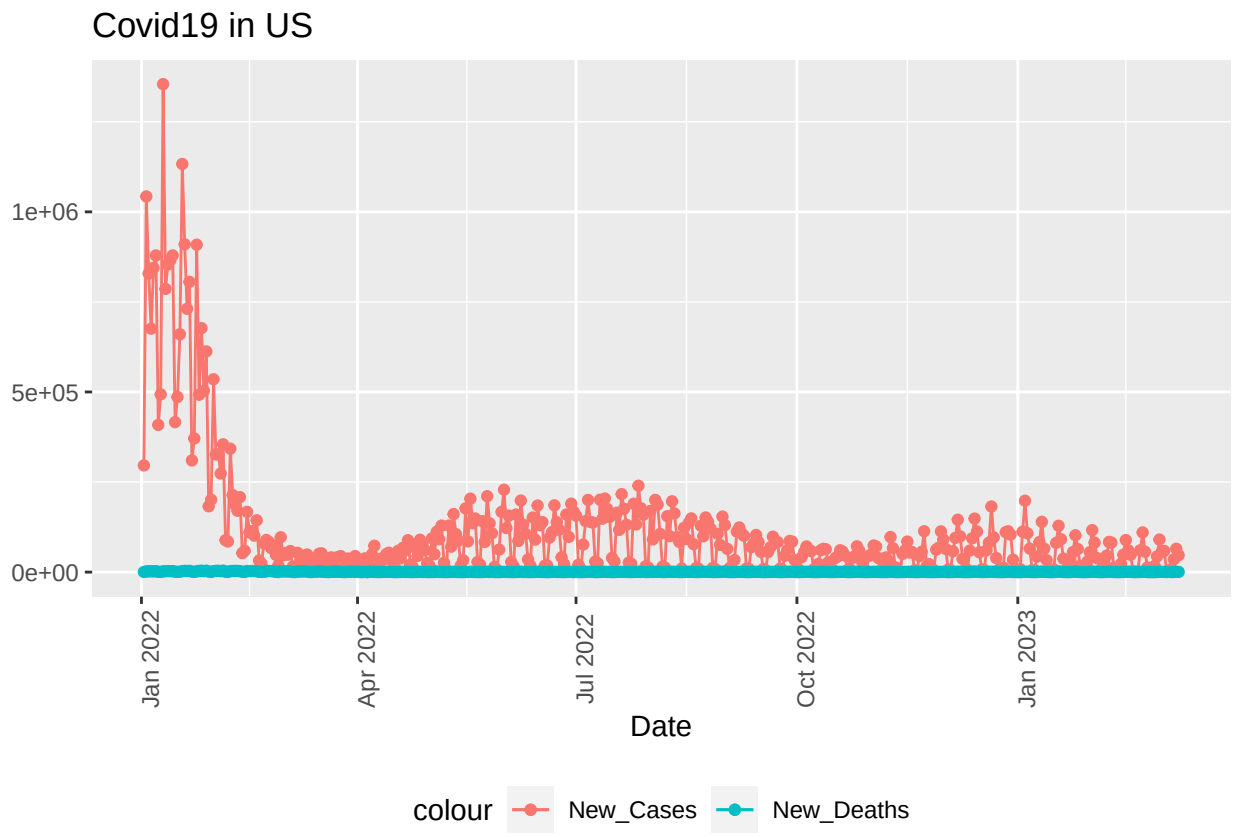
```
## # A tibble: 6 x 8
##   New_Cases New_Deaths Country_Region Date          Cases Deaths Deaths_Per_Mill
##       <dbl>      <dbl> <chr>          <date>        <dbl>  <dbl>           <dbl>
## 1      2147          7 US             2023-03-04   1.04e8 1.12e6           3377.
## 2     -3862        -38 US             2023-03-05   1.04e8 1.12e6           3377.
## 3      8564         47 US             2023-03-06   1.04e8 1.12e6           3377.
## 4     35371        337 US             2023-03-07   1.04e8 1.12e6           3378.
## 5     64861        727 US             2023-03-08   1.04e8 1.12e6           3381.
## 6     46931        584 US             2023-03-09   1.04e8 1.12e6           3382.
## # i 1 more variable: Population <dbl>
```

```
#Create US totals visualization for New Deaths and New Cases
us_totals %>%
  filter(New_Cases > 0) %>%
  ggplot(aes(x = Date, y = New_Cases)) +
  geom_line(aes(color = "New_Cases")) +
```

```
  geom_point(aes(color = "New_Cases")) +
  geom_line(aes(y = New_Deaths, color = "New_Deaths")) +
  geom_point(aes(y = New_Deaths, color = "New_Deaths")) +
  scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "Covid19 in US", y = NULL)
```



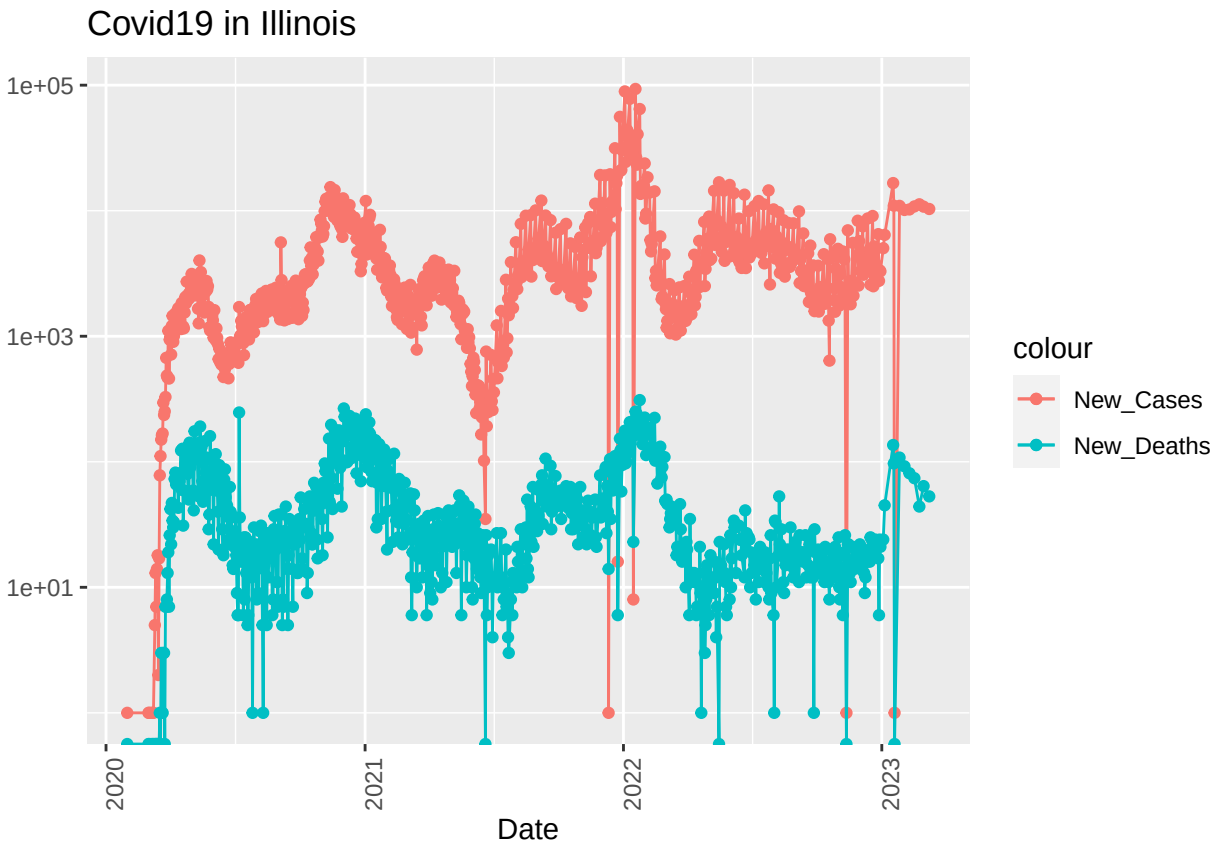

```
#Create US totals visualization for New Deaths and New Cases
us_totals %>%
  filter(New_Cases > 0 & Date > "2022-01-01") %>%
  ggplot(aes(x = Date, y = New_Cases)) +
  geom_line(aes(color = "New_Cases")) +
  geom_point(aes(color = "New_Cases")) +
  geom_line(aes(y = New_Deaths, color = "New_Deaths")) +
  geom_point(aes(y = New_Deaths, color = "New_Deaths")) +
  #scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "Covid19 in US", y = NULL)
```



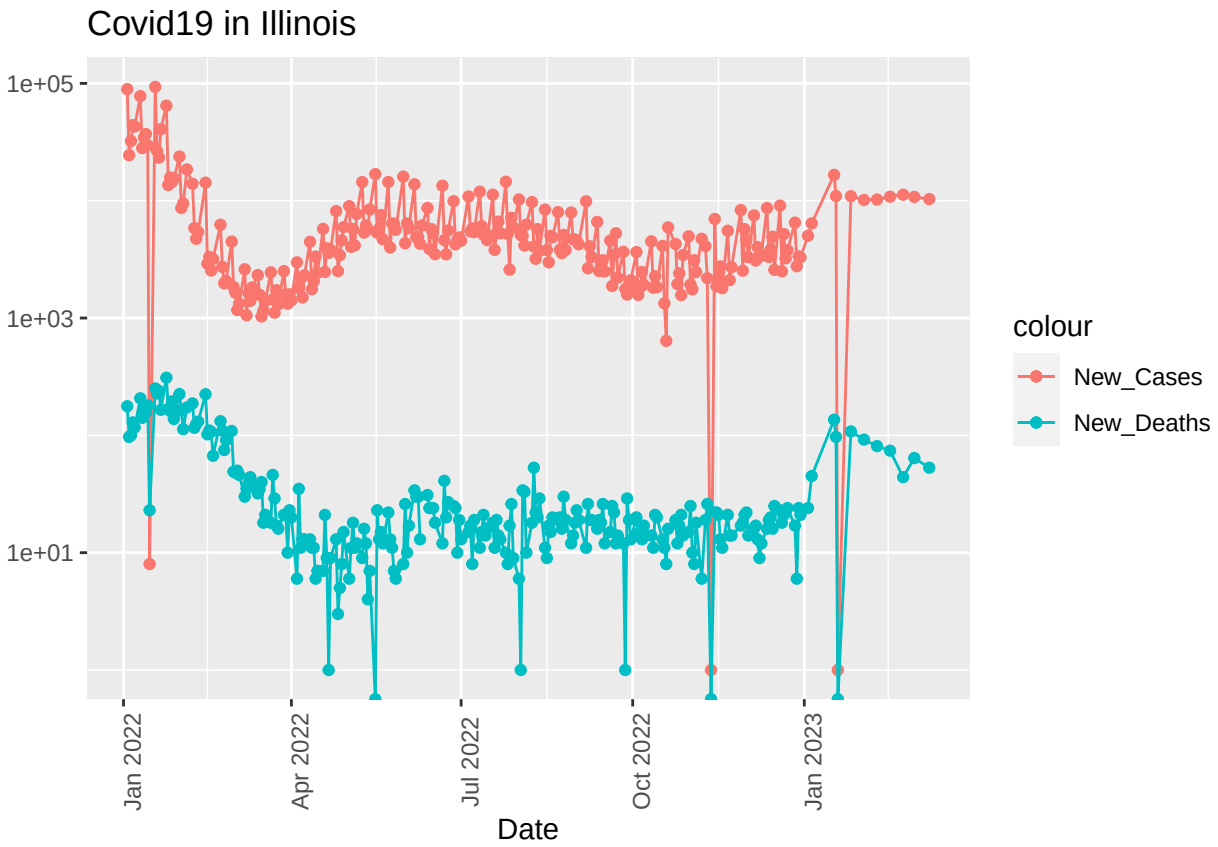
Conclusion: When we observe new cases and deaths, we see a peak for growth occurring around the beginning of the year in 2022. We then actually see the number begin to trend down. Zooming in on the data after 2022 and removing the log scale, we see some fluctuations in the data, but new cases and new deaths appear to be mostly flat, indicating linear growth.

How do New Deaths and New Cases Trend Over Time for the State of Illinois?

```
#Create Illinois totals visualization for New Deaths and New Cases
state <- "Illinois"
us_by_state %>%
  filter(Province_State == state & New_Cases > 0) %>%
  ggplot(aes(x = Date, y = New_Cases)) +
  geom_line(aes(color = "New_Cases")) +
  geom_point(aes(color = "New_Cases")) +
  geom_line(aes(y = New_Deaths, color = "New_Deaths")) +
  geom_point(aes(y = New_Deaths, color = "New_Deaths")) +
  scale_y_log10() +
  theme(legend.position = "right",
        axis.text.x = element_text(angle = 90)) +
  labs(title = str_c("Covid19 in ", state), y = NULL,
       fill = "Color")
```




```
#Create Illinois totals visualization for New Deaths and New Cases
state <- "Illinois"
us_by_state %>%
  filter(Province_State == state
         & New_Cases > 0
         & Date > "2022-01-01") %>%
  ggplot(aes(x = Date, y = New_Cases)) +
  geom_line(aes(color = "New_Cases")) +
  geom_point(aes(color = "New_Cases")) +
  geom_line(aes(y = New_Deaths, color = "New_Deaths")) +
  geom_point(aes(y = New_Deaths, color = "New_Deaths")) +
  scale_y_log10() +
  theme(legend.position = "right",
        axis.text.x = element_text(angle = 90)) +
  labs(title = str_c("Covid19 in ", state), y = NULL,
       fill = "Color")
```



Conclusion: When we observe new cases and deaths, we see a peak for growth occurring around the beginning of the year in 2022. We then actually see the number begin to trend down. Zooming in on the data after 2022 and removing the log scale, we see some fluctuations in the data, but new cases and new deaths appear to be mostly flat, indicating linear growth.

Create a List of the Top 10 Best and Worst State for Covid19 Deaths per Thousand People?

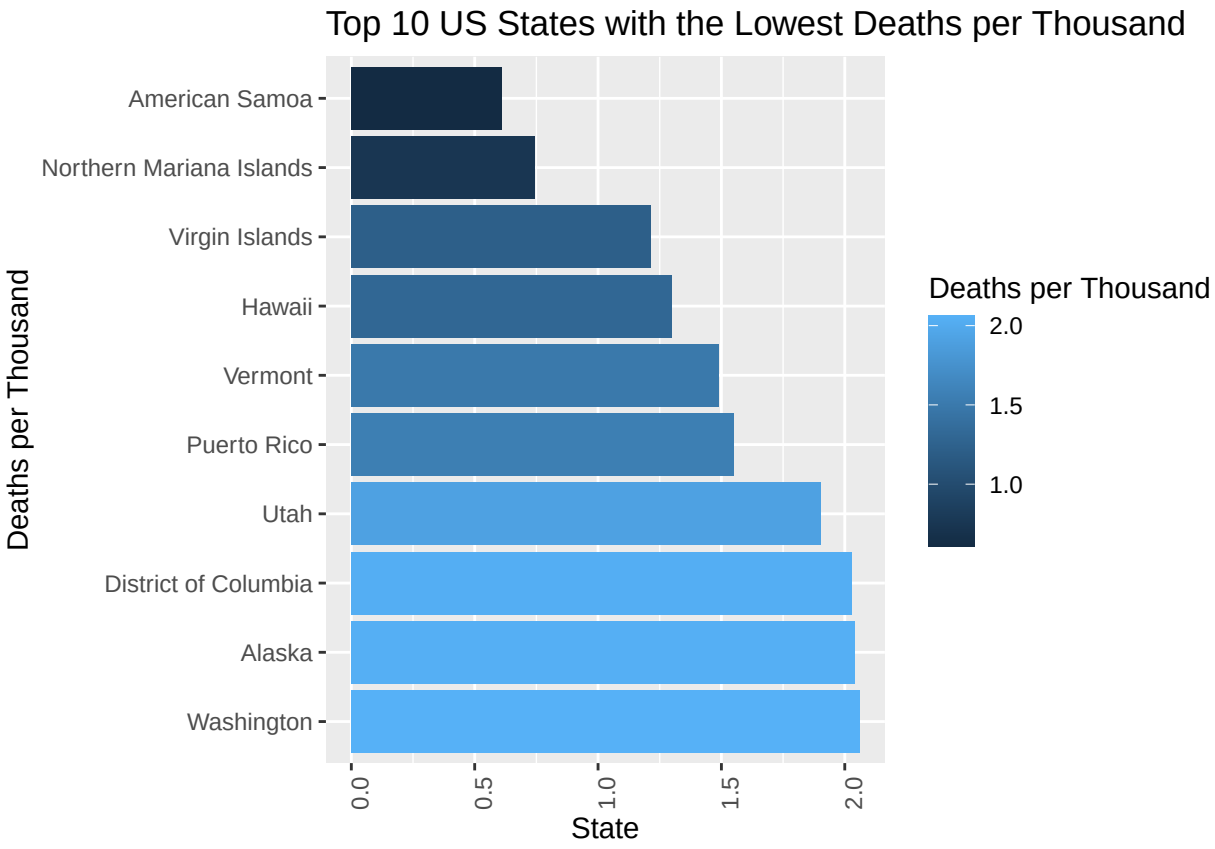
```
#Aggregate the table to remove the time element
#and create granularity at the state level
us_state_totals <- us_by_state %>%
  group_by(Province_State) %>%
  summarize(Cases = max(Cases), Deaths = max(Deaths),
            Population = max(Population),
            Cases_Per_Thou = 1000* Cases / Population,
            Deaths_Per_Thou = 1000* Deaths / Population) %>%
  filter(Cases > 0, Population > 0)
```

```
#Create a List of the Top 10 Best States
us_state_totals %>%
  slice_min(Deaths_Per_Thou, n = 10) %>%
  select(Deaths_Per_Thou, Cases_Per_Thou, everything())
```

```
## # A tibble: 10 x 6
##    Deaths_Per_Thou Cases_Per_Thou Province_State         Cases Deaths Population
```

```
##              <dbl>          <dbl> <chr>                  <dbl>  <dbl>      <dbl>
##  1           0.611           150. American Samoa        8.32e3     34      55641
##  2           0.744           248. Northern Mariana Isl~ 1.37e4     41      55144
##  3           1.21            231. Virgin Islands        2.48e4    130     107268
##  4           1.30            269. Hawaii                3.81e5   1841    1415872
##  5           1.49            245. Vermont               1.53e5    929     623989
##  6           1.55            293. Puerto Rico           1.10e6   5823    3754939
##  7           1.90            391. Utah                  1.09e6   5298    2785478
##  8           2.03            252. District of Columbia  1.78e5   1432     705749
##  9           2.04            422. Alaska                3.08e5   1486     728809
## 10           2.06            253. Washington            1.93e6  15683    7614893
```

```r
#plot the actual values and predictions
us_state_totals %>%
  slice_min(Deaths_Per_Thou, n = 10) %>%
  select(Deaths_Per_Thou, Cases_Per_Thou, everything()) %>%
  ggplot(aes(x = Deaths_Per_Thou, y = reorder(Province_State, -Deaths_Per_Thou),
             fill = Deaths_Per_Thou)) +
  geom_bar(stat = "identity") +
  theme(legend.position = "right",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "Top 10 US States with the Lowest Deaths per Thousand", y = NULL,
       fill = "Deaths per Thousand") +
  xlab("State") +
  ylab("Deaths per Thousand")
```

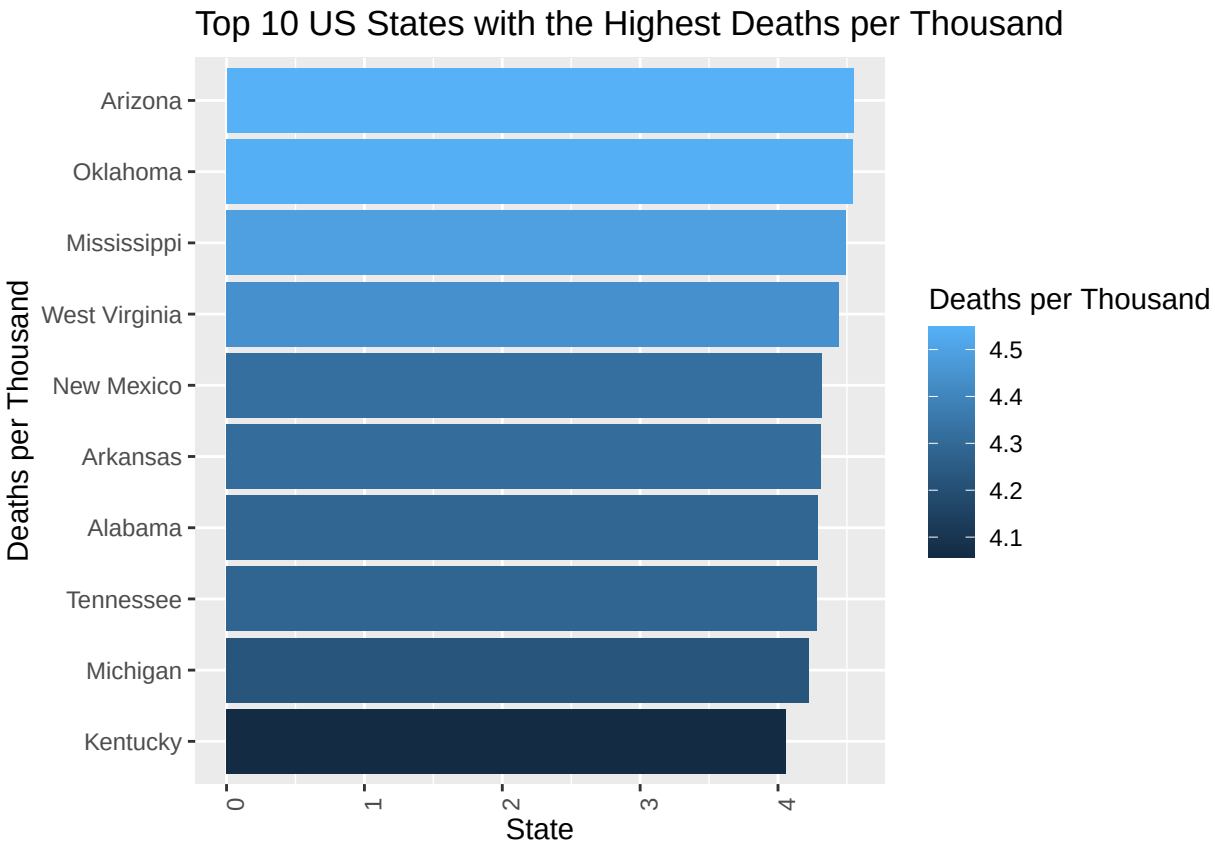


Conclusion: Looking at the states with the lowest deaths per thousand, we see that remote locations such as islands or low population locations seem to do better with deaths.

```
#Create a List of the Top 10 Worst States
us_state_totals %>%
  slice_max(Deaths_Per_Thou, n = 10) %>%
  select(Deaths_Per_Thou, Cases_Per_Thou, everything())
```

```
## # A tibble: 10 x 6
##    Deaths_Per_Thou Cases_Per_Thou Province_State   Cases Deaths Population
##              <dbl>          <dbl> <chr>             <dbl>  <dbl>      <dbl>
##  1            4.55           336. Arizona         2443514  33102    7278717
##  2            4.54           326. Oklahoma        1290929  17972    3956971
##  3            4.49           333. Mississippi      990756  13370    2976149
##  4            4.44           359. West Virginia    642760   7960    1792147
##  5            4.32           320. New Mexico       670929   9061    2096829
##  6            4.31           334. Arkansas        1006883  13020    3017804
##  7            4.29           335. Alabama         1644533  21032    4903185
##  8            4.28           368. Tennessee       2515130  29263    6829174
##  9            4.23           307. Michigan        3064125  42205    9986857
## 10            4.06           385. Kentucky        1718471  18130    4467673
```

```
#plot the actual values and predictions
us_state_totals %>%
  slice_max(Deaths_Per_Thou, n = 10) %>%
  select(Deaths_Per_Thou, Cases_Per_Thou, everything()) %>%
  ggplot(aes(x = Deaths_Per_Thou, y = reorder(Province_State, Deaths_Per_Thou),
             fill = Deaths_Per_Thou)) +
  geom_bar(stat = "identity") +
  theme(legend.position = "right",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "Top 10 US States with the Highest Deaths per Thousand", y = NULL,
       fill = "Deaths per Thousand") +
  xlab("State") +
  ylab("Deaths per Thousand")
```



Conclusion: There is less of an obvious pattern here for the top 10 states with the highest deaths per thousand people. If we had to categorize them based on properties, we might say they skew towards the southern US region and are more republican than democratic in their political views.

Additional Questions to Explore and Investigate

After completing some initial analysis and visualization, there is much more to explore and investigate. The following is a list of some potential questions to answer:

1. What are the results of the analysis when looking at the global data vs. the US data? Do we see more or less variance?
2. How does political affiliation effect covid19 cases and deaths?
3. How does average temperature for the date effect covid19 cases and deaths?
4. Is there a relationship between the population and the number of covid19 deaths?

Models & Conclusions

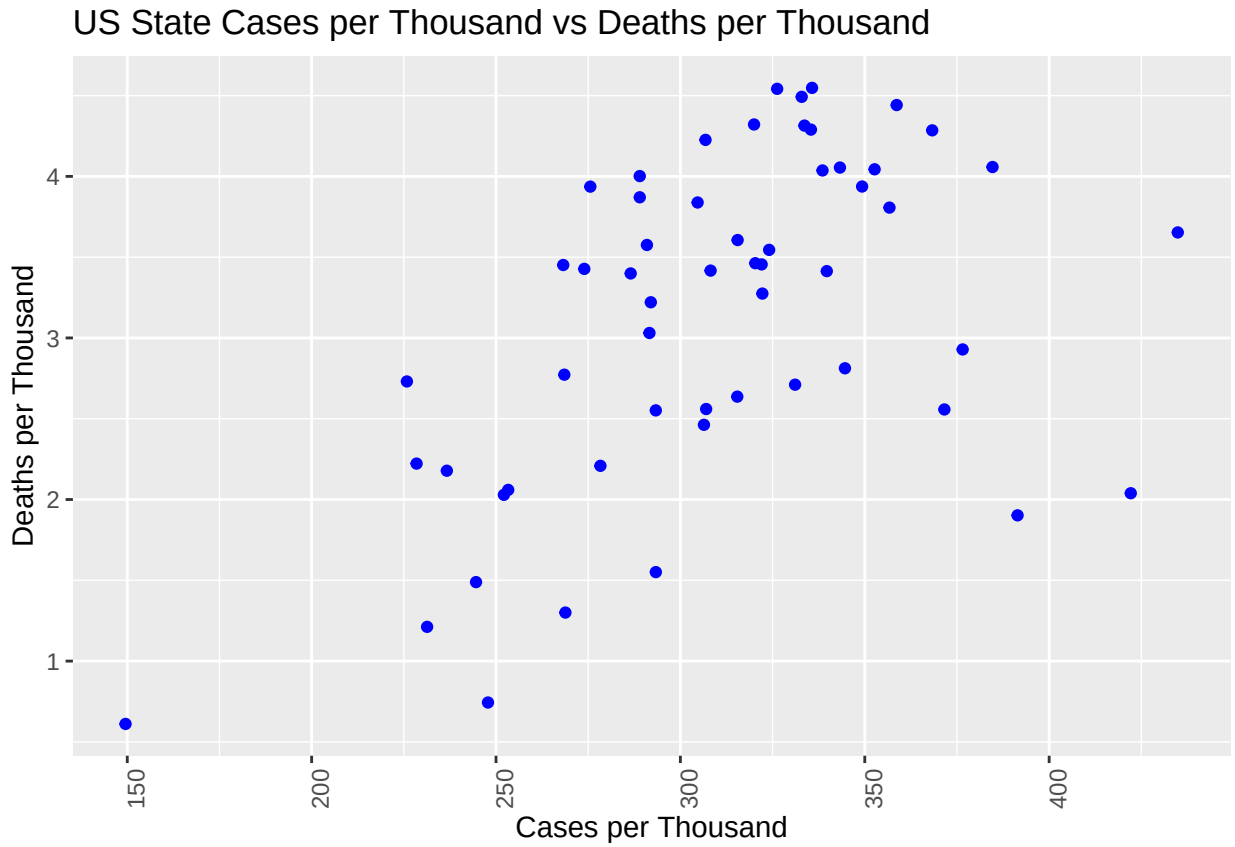
Using our dataset, we will investigate if there are enough factors available in the dataset to create a strong predictive model for covid19 deaths per thousand people.

Create a Basic Linear Prediction Models

```r
#show the data source to be used in the model
head(us_state_totals, n = 3)
```

```
## # A tibble: 3 x 6
##   Province_State   Cases Deaths Population Cases_Per_Thou Deaths_Per_Thou
##   <chr>            <dbl>  <dbl>      <dbl>          <dbl>           <dbl>
## 1 Alabama        1644533  21032    4903185           335.            4.29
## 2 Alaska          307655   1486     728809           422.            2.04
## 3 American Samoa    8320     34      55641           150.            0.611
```

```r
#build a Scatter Plot of cases per thousand vs. deaths per thousand
us_state_totals %>%
  ggplot(aes(x = Cases_Per_Thou, y = Deaths_Per_Thou)) +
  geom_point(color = "blue") +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "US State Cases per Thousand vs Deaths per Thousand", y = NULL,
       fill = "Delta") +
  xlab("Cases per Thousand") +
  ylab("Deaths per Thousand")
```



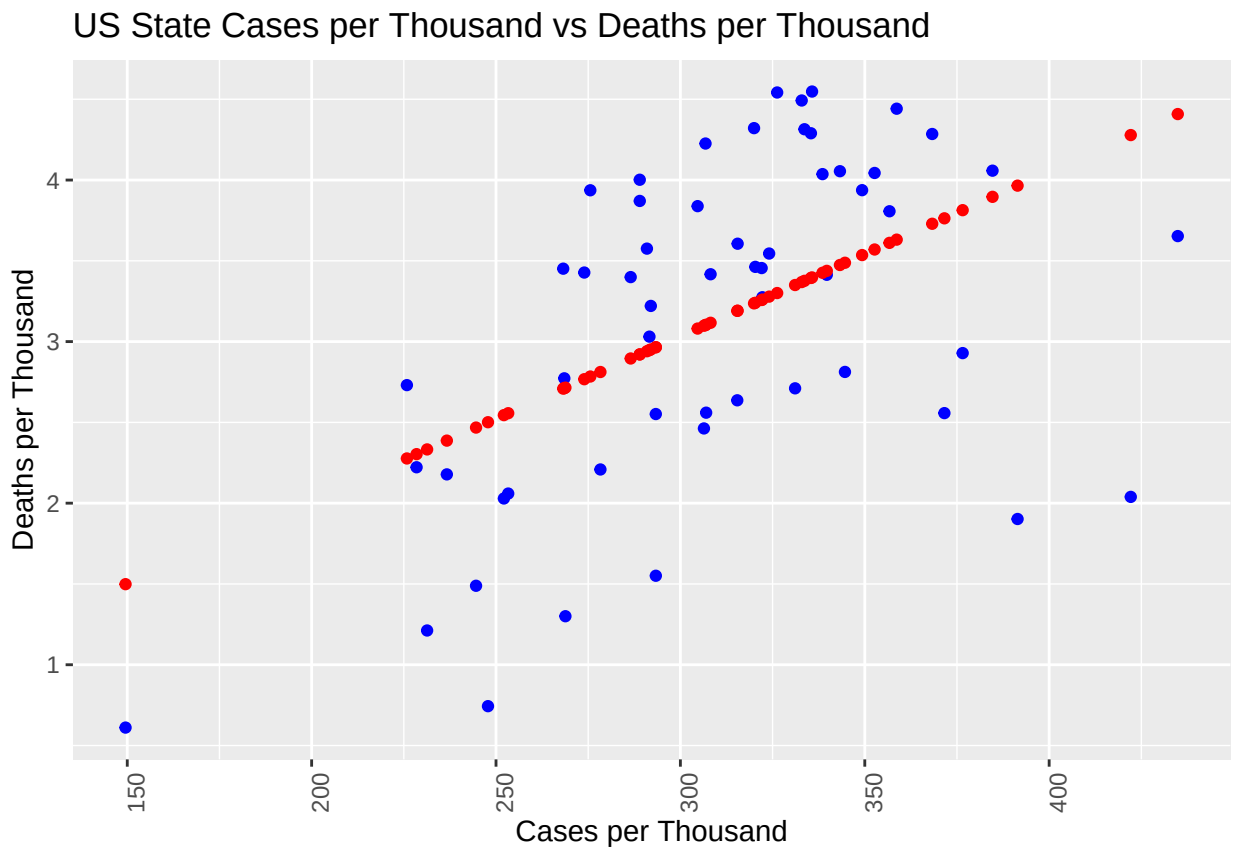

```r
#create the Prediction Model
mod <- lm(Deaths_Per_Thou ~ Cases_Per_Thou, data = us_state_totals)
summary(mod)
```

```
##
## Call:
## lm(formula = Deaths_Per_Thou ~ Cases_Per_Thou, data = us_state_totals)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2394 -0.6114  0.1965  0.6413  1.2413
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.02599    0.72442  -0.036    0.972
## Cases_Per_Thou  0.01020    0.00231   4.414 4.89e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8803 on 54 degrees of freedom
## Multiple R-squared:  0.2652, Adjusted R-squared:  0.2516
## F-statistic: 19.49 on 1 and 54 DF,  p-value: 4.894e-05
```

```
#Add the predictions to a data frame
us_state_totals_w_pred <- us_state_totals %>%
  modelr::add_predictions(mod)
us_state_totals_w_pred
```

```
## # A tibble: 56 x 7
##    Province_State   Cases Deaths Population Cases_Per_Thou Deaths_Per_Thou  pred
##    <chr>            <dbl>  <dbl>      <dbl>          <dbl>           <dbl> <dbl>
##  1 Alabama         1.64e6  21032    4903185           335.            4.29   3.39
##  2 Alaska          3.08e5   1486     728809           422.            2.04   4.28
##  3 American Samoa  8.32e3     34      55641           150.            0.611  1.50
##  4 Arizona         2.44e6  33102    7278717           336.            4.55   3.40
##  5 Arkansas        1.01e6  13020    3017804           334.            4.31   3.38
##  6 California      1.21e7 101159   39512223           307.            2.56   3.10
##  7 Colorado        1.76e6  14181    5758736           306.            2.46   3.10
##  8 Connecticut     9.77e5  12220    3565287           274.            3.43   2.77
##  9 Delaware        3.31e5   3324     973764           340.            3.41   3.44
## 10 District of Co~ 1.78e5   1432     705749           252.            2.03   2.54
## # i 46 more rows
```

```
#plot the actual values and predictions
us_state_totals_w_pred %>% ggplot() +
  geom_point(aes(x = Cases_Per_Thou, y = Deaths_Per_Thou), color = "blue") +
  geom_point(aes(x = Cases_Per_Thou, y = pred), color = "red") +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "US State Cases per Thousand vs Deaths per Thousand", y = NULL) +
  xlab("Cases per Thousand") +
  ylab("Deaths per Thousand")
```



Create a Basic Linear Prediction Models Against Additional Factors

In this section we will be additional models against the factors of % Republican Affiliation and Population of the state to see if we can improve our model of prediction.

```
#show the data source to be used in the model
head(us_state_totals, n = 3)
```

```
## # A tibble: 3 x 6
##   Province_State   Cases Deaths Population Cases_Per_Thou Deaths_Per_Thou
##   <chr>            <dbl>  <dbl>      <dbl>          <dbl>           <dbl>
## 1 Alabama        1644533  21032    4903185           335.            4.29
## 2 Alaska          307655   1486     728809           422.            2.04
## 3 American Samoa    8320     34      55641           150.            0.611
```

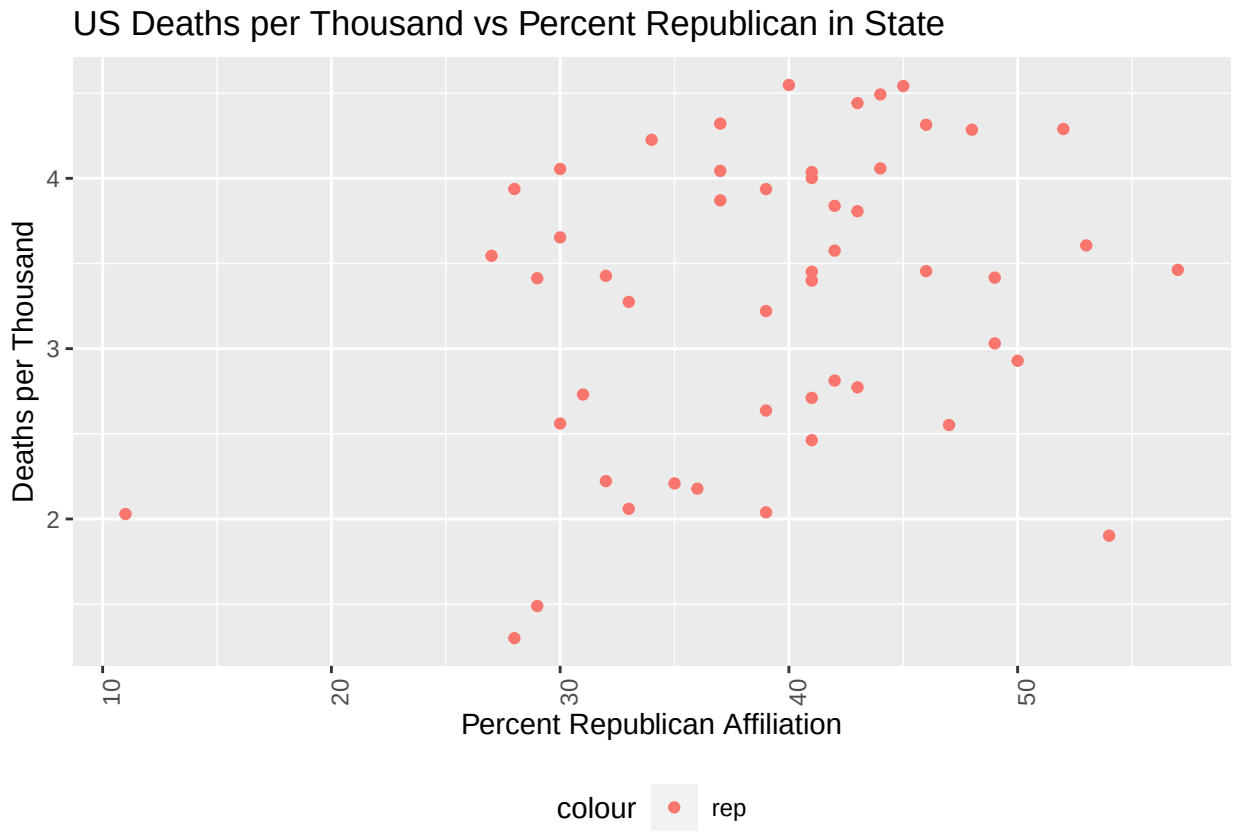
```
head(party_aff, n = 3)
```

```
## # A tibble: 3 x 5
##   state     rep no_lean   dem sample_size
##   <fct>   <int>   <int> <int>       <int>
## 1 Alabama    52      13    35         511
## 2 Alaska     39      29    32         310
## 3 Arizona    40      21    39         653
```

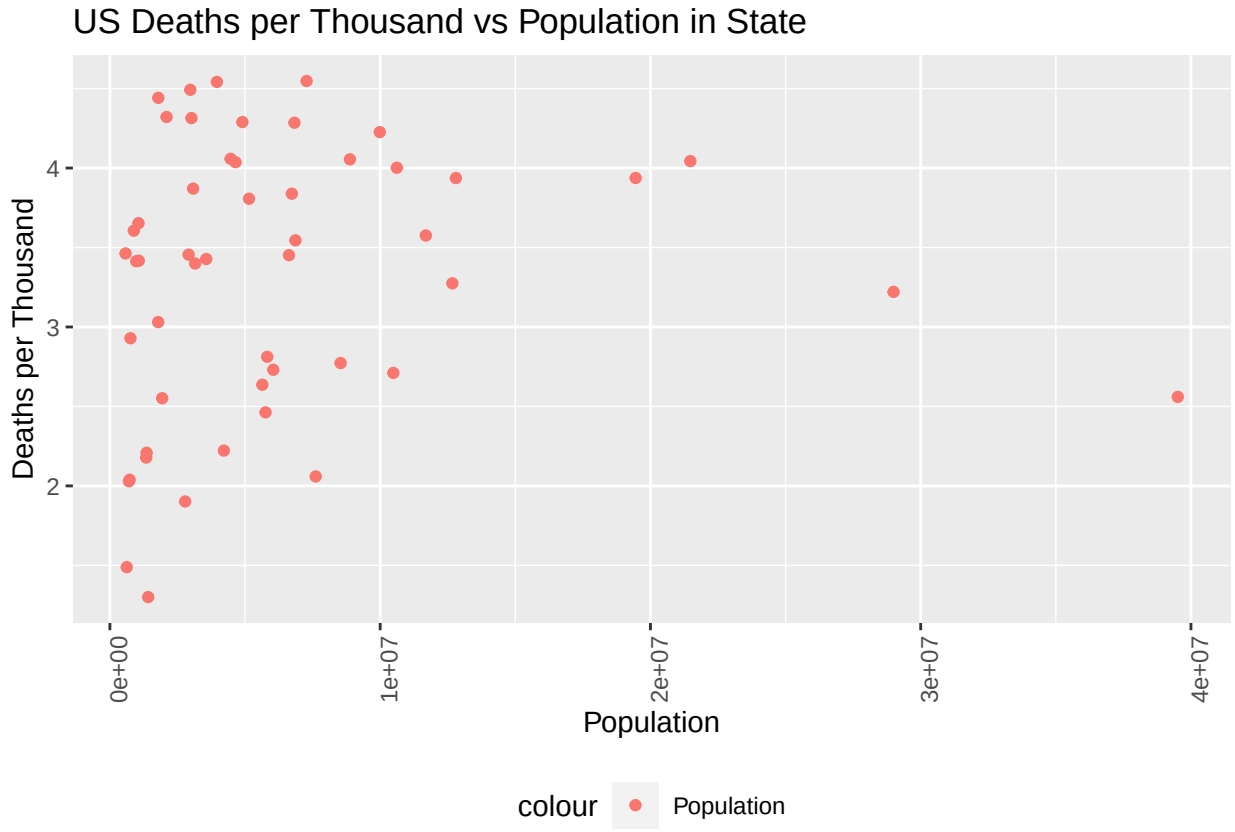
```r
#add the party affiliation data to the US data
#drop any rows that don't have party date (should get 51 states)
us_state_totals_with_party <- us_state_totals %>%
  inner_join(party_aff, by = join_by(Province_State == state))
head(us_state_totals_with_party, n = 3)
```

```
## # A tibble: 3 x 10
##   Province_State   Cases Deaths Population Cases_Per_Thou Deaths_Per_Thou   rep
##   <chr>            <dbl>  <dbl>      <dbl>          <dbl>           <dbl> <int>
## 1 Alabama        1644533  21032    4903185           335.            4.29    52
## 2 Alaska          307655   1486     728809           422.            2.04    39
## 3 Arizona        2443514  33102    7278717           336.            4.55    40
## # i 3 more variables: no_lean <int>, dem <int>, sample_size <int>
```

```r
#build a Scatter Plot of Deaths per Thousand vs. Percent Republican
us_state_totals_with_party %>%
  ggplot(aes(x = rep, y = Deaths_Per_Thou)) +
  geom_point(aes(color = "rep")) +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "US Deaths per Thousand vs Percent Republican in State",
       y = NULL) +
  xlab("Percent Republican Affiliation") +
  ylab("Deaths per Thousand")
```



```r
#build a Scatter Plot of Deaths per Thousand vs. Population
us_state_totals_with_party %>%
  ggplot(aes(x = Population, y = Deaths_Per_Thou)) +
  geom_point(aes(color = "Population")) +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "US Deaths per Thousand vs Population in State",
       y = NULL) +
  xlab("Population") +
  ylab("Deaths per Thousand")
```




```r
#create the Prediction Model
mod_party <- lm(Deaths_Per_Thou ~ rep, data = us_state_totals_with_party)
summary(mod_party)
```

```
##
## Call:
## lm(formula = Deaths_Per_Thou ~ rep, data = us_state_totals_with_party)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.84829 -0.64735  0.09234  0.64035  1.21886
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  2.12500    0.55573   3.824 0.000372 ***
## rep          0.03010    0.01386   2.172 0.034747 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8282 on 49 degrees of freedom
## Multiple R-squared:  0.0878, Adjusted R-squared:  0.06918
## F-statistic: 4.716 on 1 and 49 DF,  p-value: 0.03475
```

```
#create the Prediction Model
mod_pop <- lm(Deaths_Per_Thou ~ Population, data = us_state_totals_with_party)
summary(mod_pop)
```

```
##
## Call:
## lm(formula = Deaths_Per_Thou ~ Population, data = us_state_totals_with_party)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9473 -0.6502  0.1735  0.6261  1.2651
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.231e+00  1.612e-01  20.049   <2e-16 ***
## Population  1.151e-08  1.657e-08   0.695    0.491
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8629 on 49 degrees of freedom
## Multiple R-squared:  0.009753,   Adjusted R-squared:  -0.01046
## F-statistic: 0.4826 on 1 and 49 DF,  p-value: 0.4905
```

```
#create the Prediction Model
mod_cases_party_pop <- lm(Deaths_Per_Thou ~ rep + Cases_Per_Thou
                          + Population, data = us_state_totals_with_party)
summary(mod_cases_party_pop)
```

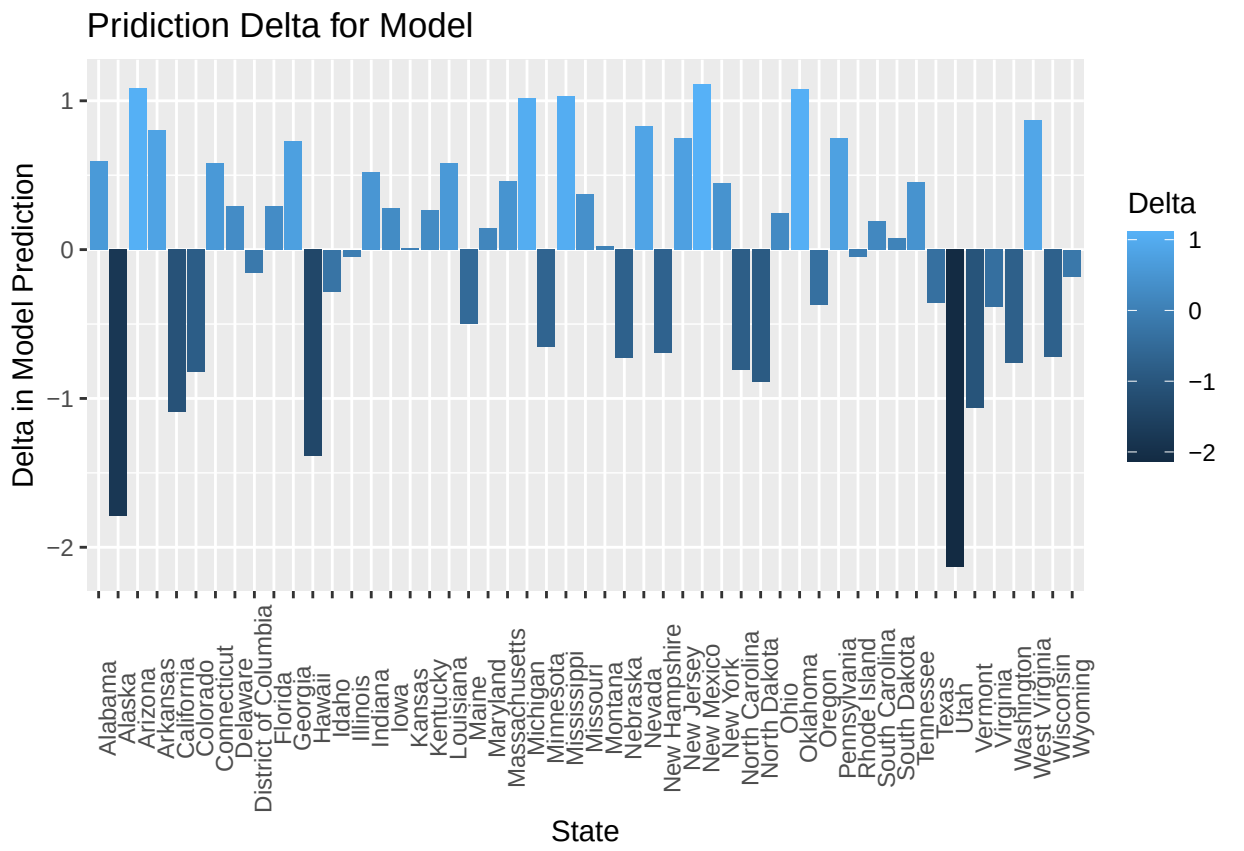
```
##
## Call:
## lm(formula = Deaths_Per_Thou ~ rep + Cases_Per_Thou + Population,
##     data = us_state_totals_with_party)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.1316 -0.5784  0.1441  0.5802  1.1136
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    4.303e-01  8.498e-01   0.506   0.6150
## rep            2.308e-02  1.427e-02   1.617   0.1125
## Cases_Per_Thou 5.892e-03  2.586e-03   2.278   0.0273 *
## Population     1.824e-08  1.552e-08   1.176   0.2455
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7908 on 47 degrees of freedom
## Multiple R-squared:  0.2022, Adjusted R-squared:  0.1513
## F-statistic: 3.971 on 3 and 47 DF,  p-value: 0.01329
```

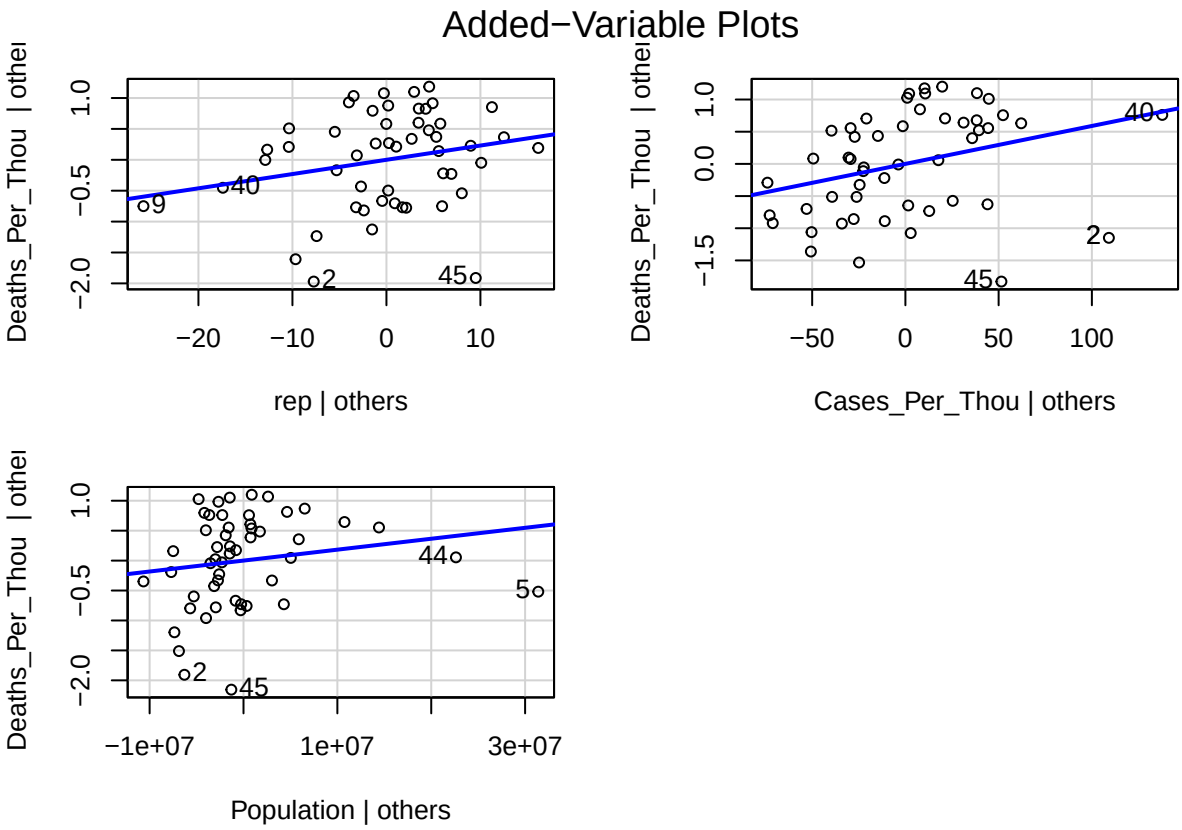
```
#Add the predictions to a data frame
us_state_totals_with_party <- us_state_totals_with_party %>%
  modelr::add_predictions(mod_party, var = "pred_party")  %>%
  modelr::add_predictions(mod_pop, var = "pred_pop")  %>%
  modelr::add_predictions(mod_cases_party_pop, var = "pred_cases_party_pop")  %>%
  mutate(delta_mod_cases_party_pop = Deaths_Per_Thou - pred_cases_party_pop)
us_state_totals_with_party
```

```
## # A tibble: 51 x 14
##    Province_State   Cases Deaths Population Cases_Per_Thou Deaths_Per_Thou   rep
##    <chr>            <dbl>  <dbl>      <dbl>          <dbl>           <dbl> <int>
##  1 Alabama         1.64e6  21032    4903185           335.            4.29    52
##  2 Alaska          3.08e5   1486     728809           422.            2.04    39
##  3 Arizona         2.44e6  33102    7278717           336.            4.55    40
##  4 Arkansas        1.01e6  13020    3017804           334.            4.31    46
##  5 California      1.21e7 101159   39512223           307.            2.56    30
##  6 Colorado        1.76e6  14181    5758736           306.            2.46    41
##  7 Connecticut     9.77e5  12220    3565287           274.            3.43    32
##  8 Delaware        3.31e5   3324     973764           340.            3.41    29
##  9 District of Co~ 1.78e5   1432     705749           252.            2.03    11
## 10 Florida         7.57e6  86850   21477737           353.            4.04    37
## # i 41 more rows
## # i 7 more variables: no_lean <int>, dem <int>, sample_size <int>,
## #   pred_party <dbl>, pred_pop <dbl>, pred_cases_party_pop <dbl>,
## #   delta_mod_cases_party_pop <dbl>
```

```
#plot the actual values and predictions
us_state_totals_with_party %>% ggplot(aes(x = Province_State,
                                y = delta_mod_cases_party_pop,
                                fill = delta_mod_cases_party_pop)) +
  geom_bar(stat = "identity") +
  theme(legend.position = "right",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "Pridiction Delta for Model", y = NULL,
       fill = "Delta") +
  xlab("State") +
  ylab("Delta in Model Prediction")
```




```
#plot the actual values and predictions
avPlots(mod_cases_party_pop)
```



Conclusion: Overall, there is some potential for the fact model using the cases per thousand, percent republican, affiliation, and population, but the predictive model is not very strong. Additional variables are likely needed to improve the accuracy of the prediction. Through this process these current variables explored might be replaced by more predictive variables. However, with a low confidence, we can say that an increase in all 3 of these factors for your state may make you more vulnerable to deaths from covid19.

Conclusions

After completing the analysis of data, visualization, and modeling, we can conclude the following:

Question	Conclusion
What Does the Trend of Cases and Deaths look like overall for the US?	This plot displays the cumulative total for the US. Given the extremely large number of cases and the log scale, it is hard to tell for recent data how much the chart is increasing and if cases and deaths are going up on a daily basis. Overall, it displays that there was a sharp increase initially, but then cases began to taper off and grow slower than exponential. Looking at the zoomed in chart on 2022 (with the log scale removed, we see growth that looks more linear than exponential.
What Does the Trend of Cases and Deaths look like overall for Illinois?	Comparing Illinois to the US totals, we see a similar pattern. Extreme growth of cases initially, then it tapers off on the log scale graph. Overall, the macro patterns look the same for both.

Question	Conclusion
What is the Largest Total Deaths and Date in the covid19 in the US Plot?	The largest data point for the US is 1,122,724 total cases and occurs on 2023-03-09.
How do New Deaths and New Cases Trend Over Time in the US?	When we observe new cases and deaths, we see a peak for growth occurring around the beginning of the year in 2022. We then actually see the number begin to trend down. Zooming in on the data after 2022 and removing the log scale, we see some fluctuations in the data, but new cases and new deaths appear to be mostly flat, indicating linear growth.
How do New Deaths and New Cases Trend Over Time for the State of Illinois?	When we observe new cases and deaths, we see a peak for growth occurring around the beginning of the year in 2022. We then actually see the number begin to trend down. Zooming in on the data after 2022 and removing the log scale, we see some fluctuations in the data, but new cases and new deaths appear to be mostly flat, indicating linear growth.
Create a List of the Top 10 Best and Worst State for covid19 Deaths per Thousand People?	Looking at the states with the lowest deaths per thousand, we see that remote locations such as islands or low population locations seem to do better with deaths. There is less of an obvious pattern here for the top 10 states with the highest deaths per thousand people. If we had to categorize them based on properties, we might say they skew towards the southern US region and are more republican than democratic in their political views.
Can we create a usable predictive model for covid19 deaths per thousand people using our US dataset?	In short, No. A significant predictive model cannot be created with the 3 state level variables in our dataset. Additional variables will need to be explored to develop a usable model.

Review of Bias

Considering Bias, I would place it into 3 categories:

1. Who is providing the data
2. Who is collecting the data
3. Who is analyzing the data

Provider: In the United States, data was provide by individuals and the hospital systems. Some people may have not reported positive cases when they were doing at home tests. Each hospital system likely had its own methodology of collecting the data to provide.

Collector: In the United States, each state was responsible for collecting covid19 data about cases and deaths. Through news media, we saw that there was some effort within state governments to manipulate the case numbers (or possible death numbers). How the state felt about covid19 may have influenced the outcome of the collection process.

Analyzer: As the analyst, I bring my own biases to the data. I am an urban resident in Illinois, so my covid19 experience is shaped by experience in this state. I'm also a Democrat and at scale the bias that political affiliation brought to covid19 shaped opinions across the political spectrum, I also not a subject matter expert in epidemiology and may not interpret the details or factors or trends correctly.

Session Summary

```
sessionInfo()
```

```
## R version 4.2.3 (2023-03-15 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 22621)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=English_United States.utf8
## [2] LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] car_3.1-2       carData_3.0-5   xml2_1.3.3      rvest_1.0.3
##  [5] lubridate_1.9.2 forcats_1.0.0   stringr_1.5.0   dplyr_1.1.1
##  [9] purrr_1.0.1     readr_2.1.4     tidyr_1.3.0     tibble_3.2.1
## [13] ggplot2_3.4.2   tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
##  [1] tidyselect_1.2.0 xfun_0.38        colorspace_2.1-0 vctrs_0.6.1
##  [5] generics_0.1.3   htmltools_0.5.5  yaml_2.3.7       utf8_1.2.3
##  [9] rlang_1.1.0      pillar_1.9.0     glue_1.6.2       withr_2.5.0
## [13] bit64_4.0.5      modelr_0.1.11    lifecycle_1.0.3  munsell_0.5.0
## [17] gtable_0.3.3     evaluate_0.20    labeling_0.4.2   knitr_1.42
## [21] tzdb_0.3.0       fastmap_1.1.1    parallel_4.2.3   curl_5.0.0
## [25] fansi_1.0.4      highr_0.10       broom_1.0.4      backports_1.4.1
## [29] scales_1.2.1     vroom_1.6.1      abind_1.4-5      farver_2.1.1
## [33] bit_4.0.5        hms_1.1.3        digest_0.6.31    stringi_1.7.12
## [37] grid_4.2.3       cli_3.6.1        tools_4.2.3      magrittr_2.0.3
## [41] crayon_1.5.2     pkgconfig_2.0.3  timechange_0.2.0 rmarkdown_2.21
## [45] httr_1.4.5       rstudioapi_0.14  R6_2.5.1         compiler_4.2.3
```