# COVID-19 Case and Mortality Trends: Insights from Johns Hopkins Data

Anh Mai

2024-10-26

## Introduction

The COVID-19 pandemic has had a profound and far-reaching impact on global health, economies, and daily life. As the pandemic continues to evolve, analyzing trends in COVID-19 cases and mortality rates becomes crucial for understanding its trajectory and informing public health responses.

This report focuses on analyzing COVID-19 case and mortality trends both in the United States and globally using data provided by Johns Hopkins University. Johns Hopkins University's COVID-19 Dashboard has been a critical resource throughout the pandemic, offering comprehensive and real-time data on the spread of the virus.

### Objectives

The primary objectives of this analysis are:

1. **To Examine Global Trends:** Investigate the overall trends in COVID-19 cases and mortality rates on a global scale. This includes identifying patterns in the spread of the virus and understanding the impacts on different regions.
2. **To Analyze U.S. Trends:** Analyze the trends in COVID-19 cases and mortality rates specifically within the United States. This will include a closer look at how the situation has evolved over time and the impact of various public health interventions.
3. **To Compare Regional Differences:** Compare the trends between the U.S. and global data to identify any significant differences or similarities. This will help to contextualize the U.S. experience within the broader global picture.

### Data Source

The data for this analysis is sourced from the Johns Hopkins University COVID-19 Dashboard found on GitHub.

"https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/"

"time_series_covid19_confirmed_US.csv", "time_series_covid19_confirmed_global.csv", "time_series_covid19_deaths_U
"time_series_covid19_deaths_global.csv"

### Scope of Analysis

This analysis will cover:

- **Case Trends:** Examining the trajectory of COVID-19 case numbers, including daily new cases, cumulative cases, and growth rates.
- **Mortality Trends:** Analyzing the trends in COVID-19-related deaths, including daily mortality rates, cumulative deaths, and mortality rates per capita.
- **Comparative Insights:** Comparing the case and mortality trends between the U.S. and other countries to identify key differences and similarities.

By delving into these trends, we aim to provide valuable insights into the progression of the pandemic and its impacts, offering a foundation for informed decision-making and policy development.

## Data Wrangling

### Data Loading

Load and preprocess your data here.

```
# Wrap text
# Global options for code chunk
knitr::opts_chunk$set(
  tidy = TRUE,           # Automatically tidy code
  width = 80,            # Set output width
  collapse = TRUE,       # Collapse code and output together
  comment = "#>"         # Add comment prefix to output
)
options(width = 80)      # Set output width globally for code results

# Load necessary libraries
library(formatR)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- 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 errors
```

```
library(ggplot2)
library(dplyr)
library(knitr)
```

```r
# Set global options
opts_chunk$set(echo = TRUE)

# Load your dataset
url_in <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_cov

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")

# Concate url_in and file_names to make whole url for these 4 files of data
urls <- str_c(url_in, file_names)

# Read data/load data into tables
us_cases <- 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.
```

```r
global_cases <- 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.
```

```r
us_deaths <- 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.
```

```r
global_deaths <- 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.
```

## Data Cleanup

```
# Display the first few rows of the dataset
head(global_cases)
#> # A tibble: 6 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
#> # 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>, ...
head(global_deaths)
#> # A tibble: 6 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
#> # 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>, ...
head(us_cases)
#> # A tibble: 6 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
#> # 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>, ...
head(us_deaths)
#> # A tibble: 6 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
#> # 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>, ...

# Tidying up global_cases data
global_cases <- global_cases %>%
    pivot_longer(cols = -c(`Province/State`, `Country/Region`, Lat, Long), names_to = "Date",
        values_to = "Cases") %>%
    select(-c(Lat, Long))

# Tidying up global_deaths data
global_deaths <- global_deaths %>%
    pivot_longer(cols = -c(`Province/State`, `Country/Region`, Lat, Long), names_to = "Date",
        values_to = "Deaths") %>%
    select(-c(Lat, Long))

# Combing global_cases and global_deaths into 1 big table global
global <- global_cases %>%
    full_join(global_deaths) %>%
    rename(Country_Region = `Country/Region`, Province_State = `Province/State`) %>%
    mutate(Date = mdy(Date))
#> Joining with `by = join_by(`Province/State`, `Country/Region`, Date)`

# Summary Data
summary(global)
#>  Province_State     Country_Region          Date                Cases
#>  Length:330327      Length:330327      Min.   :2020-01-22   Min.   :      0
#>  Class :character   Class :character   1st Qu.:2020-11-02   1st Qu.:    680
#>  Mode  :character   Mode  :character   Median :2021-08-15   Median :  14429
#>                                        Mean   :2021-08-15   Mean   : 959384
```

```
#>                                          3rd Qu.:2022-05-28   3rd Qu.:   228517
#>                                          Max.   :2023-03-09   Max.   :103802702
#>      Deaths
#>  Min.   :      0
#>  1st Qu.:      3
#>  Median :    150
#>  Mean   :  13380
#>  3rd Qu.:   3032
#>  Max.   :1123836

# Filter the rows that have no cases
global <- global %>%
    filter(Cases > 0)
# In case we want to check if any issue with data such as duplicate... do
# filter the data and check
global %>%
    filter(Cases > 2)
#> # A tibble: 302,667 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
#>  4 <NA>           Afghanistan    2020-02-27     5      0
#>  5 <NA>           Afghanistan    2020-02-28     5      0
#>  6 <NA>           Afghanistan    2020-02-29     5      0
#>  7 <NA>           Afghanistan    2020-03-01     5      0
#>  8 <NA>           Afghanistan    2020-03-02     5      0
#>  9 <NA>           Afghanistan    2020-03-03     5      0
#> 10 <NA>           Afghanistan    2020-03-04     5      0
#> # i 302,657 more rows

# Tidying up us_cases data
us_cases <- us_cases %>%
    pivot_longer(cols = -(UID:Combined_Key), names_to = "Date", values_to = "Cases") %>%
    select(Admin2:Cases) %>%
    mutate(Date = mdy(Date)) %>%
    select(-c(Lat, Long_))

# Tidying up us_deaths data
us_deaths <- us_deaths %>%
    pivot_longer(cols = -(UID:Population), names_to = "Date", values_to = "Deaths") %>%
    select(Admin2:Deaths) %>%
    mutate(Date = mdy(Date)) %>%
    select(-c(Lat, Long_))

# Combine us_cases and us_deaths into 1 table US
US <- us_cases %>%
    full_join(us_deaths)
#> Joining with `by = join_by(Admin2, Province_State, Country_Region,
#> Combined_Key, Date)`

# Make 2 tables US and global identical fields Global doesn't have Combined_Key
```

```r
# field, create Combined_Key for global like US
global <- global %>%
    unite("Combined_Key", c(Province_State, Country_Region), sep = ", ", na.rm = TRUE,
        remove = FALSE)

# Add Population column into global
uid_lookup_url <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/U

uid <- read_csv(uid_lookup_url) %>%
    select(-c(Lat, Long_, Combined_Key, code3, iso2, iso3, Admin2))
#> Rows: 4321 Columns: 12
#> -- Column specification --------------------------------------------------------
#> Delimiter: ","
#> chr (7): iso2, iso3, FIPS, Admin2, Province_State, Country_Region, Combined_Key
#> dbl (5): UID, code3, Lat, Long_, Population
#>
#> 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 <- 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)
```

# Data Analysis

## Summary

```r
# Summarize the dataset
summary(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
summary(US)
#>     Admin2          Province_State     Country_Region     Combined_Key
#>  Length:3819906     Length:3819906     Length:3819906     Length:3819906
#>  Class :character   Class :character   Class :character   Class :character
#>  Mode  :character   Mode  :character   Mode  :character   Mode  :character
```

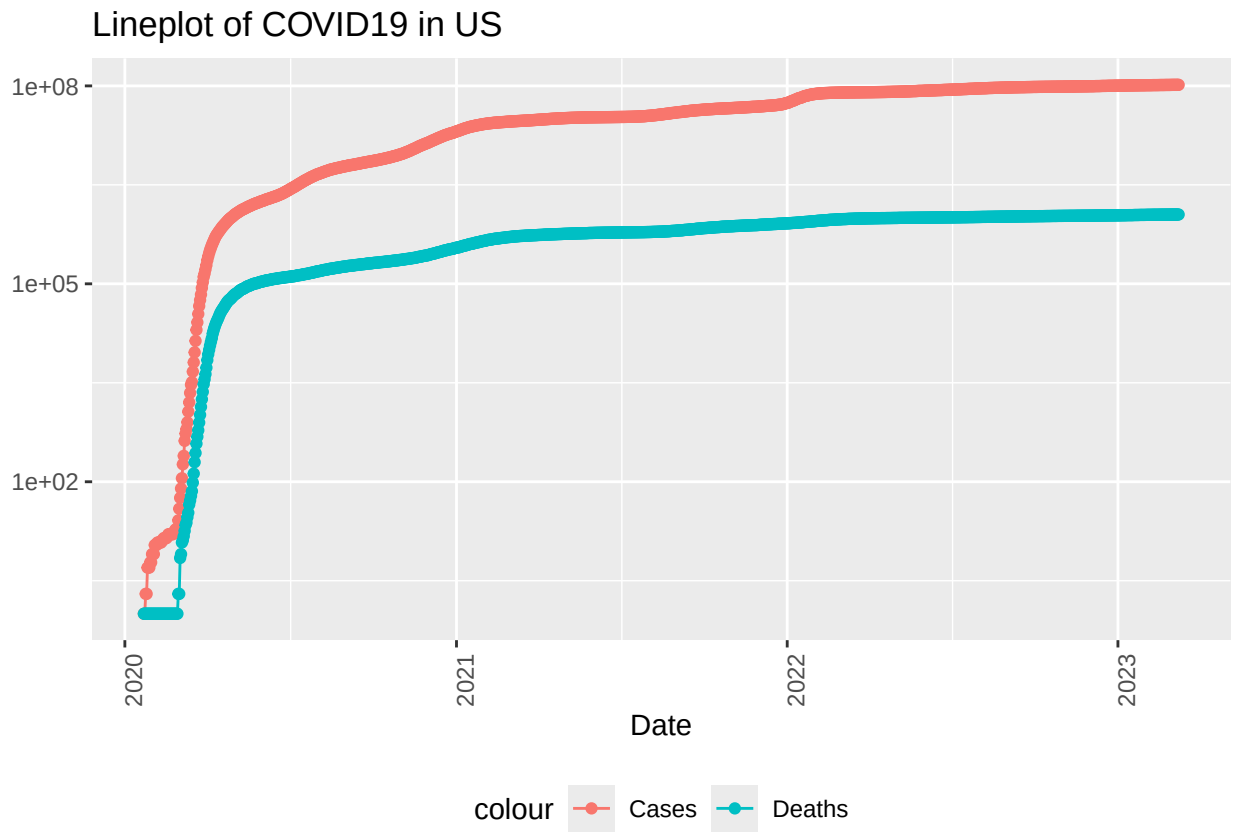
```
#>
#>
#>
#>       Date                Cases             Population           Deaths
#>  Min.   :2020-01-22   Min.   :  -3073   Min.   :       0   Min.   :  -82.0
#>  1st Qu.:2020-11-02   1st Qu.:    330   1st Qu.:    9917   1st Qu.:    4.0
#>  Median :2021-08-15   Median :   2272   Median :   24892   Median :   37.0
#>  Mean   :2021-08-15   Mean   :  14088   Mean   :   99604   Mean   :  186.9
#>  3rd Qu.:2022-05-28   3rd Qu.:   8159   3rd Qu.:   64979   3rd Qu.:  122.0
#>  Max.   :2023-03-09   Max.   :3710586   Max.   :10039107   Max.   :35545.0

US_by_State <- US %>%
    group_by(Province_State, Country_Region, Date) %>%
    summarize(Cases = sum(Cases), Deaths = sum(Deaths), Population = sum(Population)) %>%
    mutate(Deaths_per_million = Deaths * 1e+06/Population) %>%
    select(Province_State, Country_Region, Date, Cases, Deaths, Deaths_per_million,
        Population) %>%
    ungroup()
#> `summarise()` has grouped output by 'Province_State', 'Country_Region'. You can
#> override using the `.groups` argument.

US_Totals <- US %>%
    group_by(Country_Region, Date) %>%
    summarize(Cases = sum(Cases), Deaths = sum(Deaths), Population = sum(Population)) %>%
    mutate(Deaths_per_million = Deaths * 1e+06/Population) %>%
    select(Country_Region, Date, Cases, Deaths, Deaths_per_million, Population) %>%
    ungroup()
#> `summarise()` has grouped output by 'Country_Region'. You can override using
#> the `.groups` argument.

# Analyze more after general graphs have been drafted
summary(US_Totals)
#>  Country_Region          Date                Cases               Deaths
#>  Length:1143        Min.   :2020-01-22   Min.   :        1   Min.   :      1
#>  Class :character   1st Qu.:2020-11-02   1st Qu.:  9401880   1st Qu.: 232564
#>  Mode  :character   Median :2021-08-15   Median : 36845902   Median : 618029
#>                     Mean   :2021-08-15   Mean   : 47080794   Mean   : 624563
#>                     3rd Qu.:2022-05-27   3rd Qu.: 84083678   3rd Qu.:1006626
#>                     Max.   :2023-03-09   Max.   :103802702   Max.   :1123836
#>  Deaths_per_million   Population
#>  Min.   :   0.003   Min.   :332875137
#>  1st Qu.: 698.652   1st Qu.:332875137
#>  Median :1856.639   Median :332875137
#>  Mean   :1876.267   Mean   :332875137
#>  3rd Qu.:3024.033   3rd Qu.:332875137
#>  Max.   :3376.149   Max.   :332875137

# Add new_cases and new_deaths into US_by_State and US_Totals
US_by_State <- US_by_State %>%
    mutate(New_Cases = Cases - lag(Cases), New_Deaths = Deaths - lag(Deaths))
US_Totals <- US_Totals %>%
    mutate(New_Cases = Cases - lag(Cases), New_Deaths = Deaths - lag(Deaths))
```

```
# US_State_Totals
US_State_Totals <- US_by_State %>%
    group_by(Province_State) %>%
    summarize(Deaths = max(Deaths), Cases = max(Cases), Population = max(Population),
        Cases_per_thousand = Cases * 1000/Population, Deaths_per_thousand = Deaths *
            1000/Population) %>%
    filter(Cases > 0, Population > 0)

# Best 10 States
US_State_Totals %>%
    slice_min(Deaths_per_thousand, n = 10)
#> # A tibble: 10 x 6
#>    Province_State           Deaths   Cases Population Cases_per_thousand
#>    <chr>                     <dbl>   <dbl>      <dbl>              <dbl>
#>  1 American Samoa               34    8320      55641               150.
#>  2 Northern Mariana Islands     41   13666      55144               248.
#>  3 Virgin Islands              130   24813     107268               231.
#>  4 Hawaii                     1841  380608    1415872               269.
#>  5 Vermont                     929  152618     623989               245.
#>  6 Puerto Rico                5823 1101469    3754939               293.
#>  7 Utah                       5298 1090346    3205958               340.
#>  8 Alaska                     1486  307655     740995               415.
#>  9 District of Columbia       1432  177945     705749               252.
#> 10 Washington                15683 1928913    7614893               253.
#> # i 1 more variable: Deaths_per_thousand <dbl>

# Worst 10 States
US_State_Totals %>%
    slice_max(Deaths_per_thousand, n = 10)
#> # A tibble: 10 x 6
#>    Province_State Deaths   Cases Population Cases_per_thousand
#>    <chr>           <dbl>   <dbl>      <dbl>              <dbl>
#>  1 Arizona         33102 2443514    7278717               336.
#>  2 Oklahoma        17972 1290929    3956971               326.
#>  3 Mississippi     13370  990756    2976149               333.
#>  4 West Virginia    7960  642760    1792147               359.
#>  5 New Mexico       9061  670929    2096829               320.
#>  6 Arkansas        13020 1006883    3017804               334.
#>  7 Alabama         21032 1644533    4903185               335.
#>  8 Tennessee       29263 2515130    6829174               368.
#>  9 Michigan        42205 3064125    9986857               307.
#> 10 Kentucky        18130 1718471    4467673               385.
#> # i 1 more variable: Deaths_per_thousand <dbl>
```

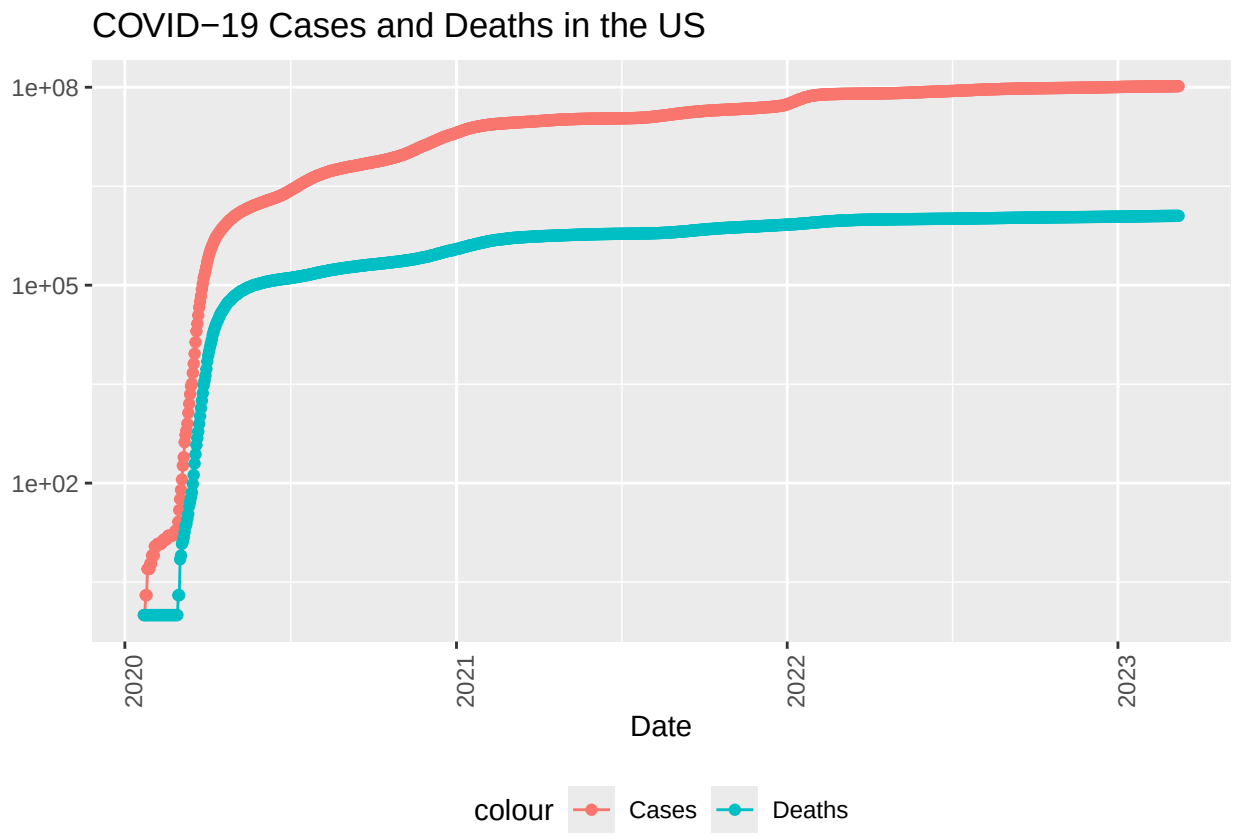
## Visualization

```
# Create a plot for US Totals in US Way1
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")) +
```

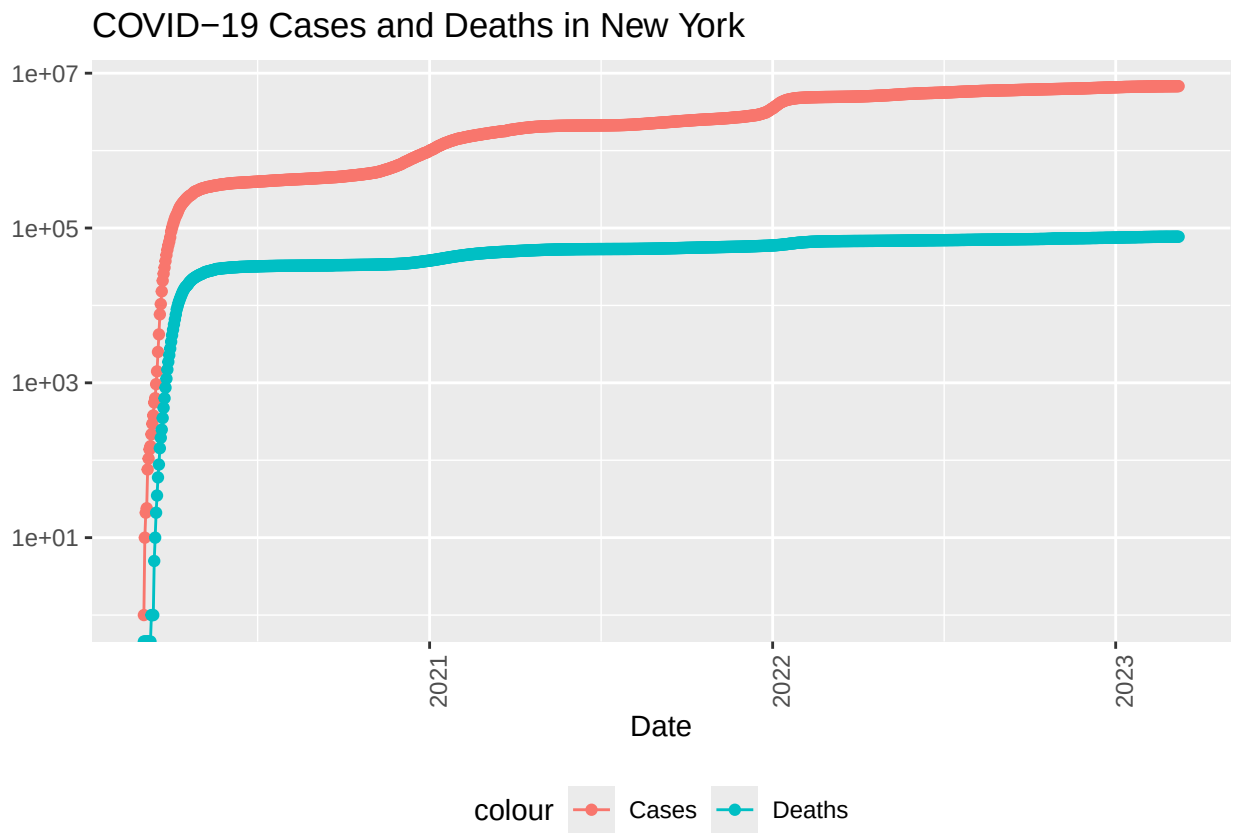
```
    scale_y_log10() + theme(legend.position = "bottom", axis.text.x = element_text(angle = 90)) +
    labs(title = "Lineplot of COVID19 in US", y = NULL)
```



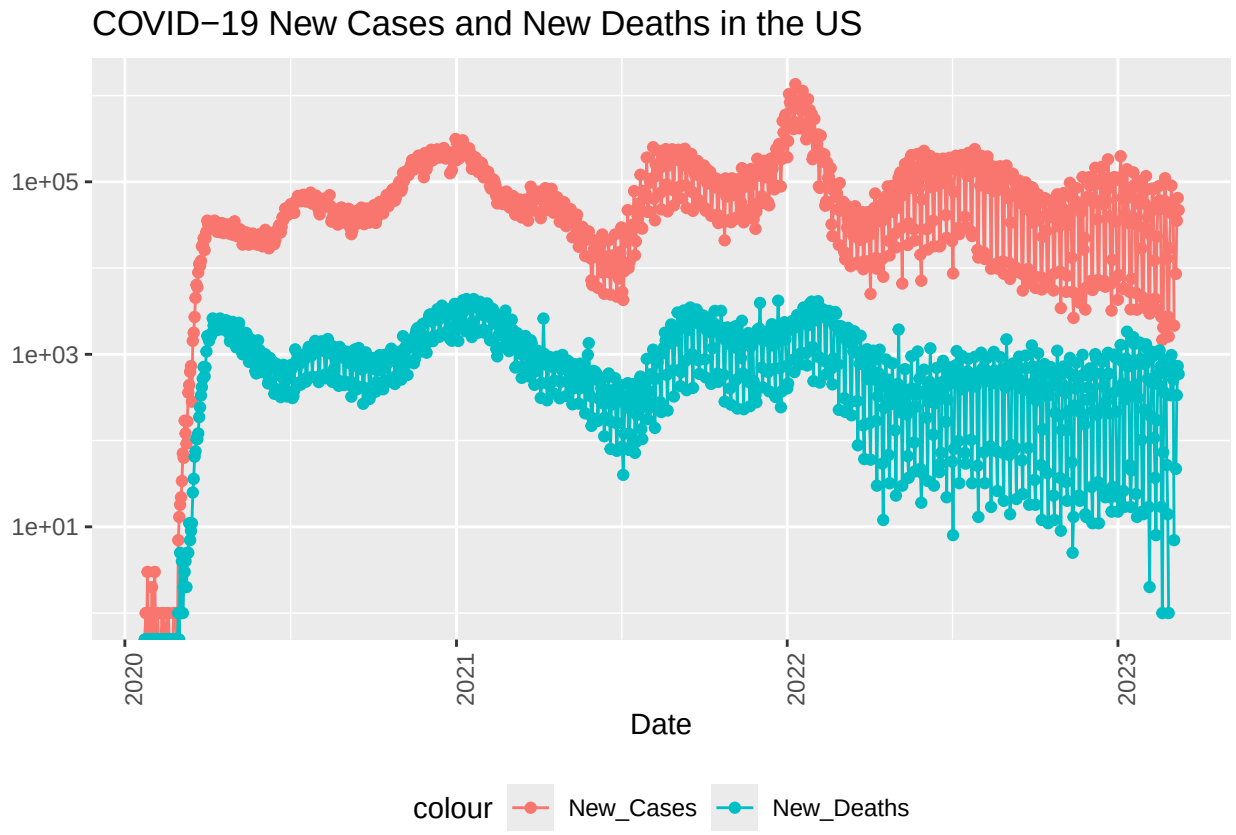

```
# Way2
US_Totals %>%
    filter(Cases > 0) %>%
    ggplot(aes(x = Date)) + geom_line(aes(y = Cases, color = "Cases")) + geom_point(aes(y = Cases,
    color = "Cases")) + geom_line(aes(y = Deaths, color = "Deaths")) + geom_point(aes(y = Deaths,
    color = "Deaths")) + scale_y_log10() + theme(legend.position = "bottom", axis.text.x = element_text
    labs(title = "COVID-19 Cases and Deaths in the US", y = NULL)
```



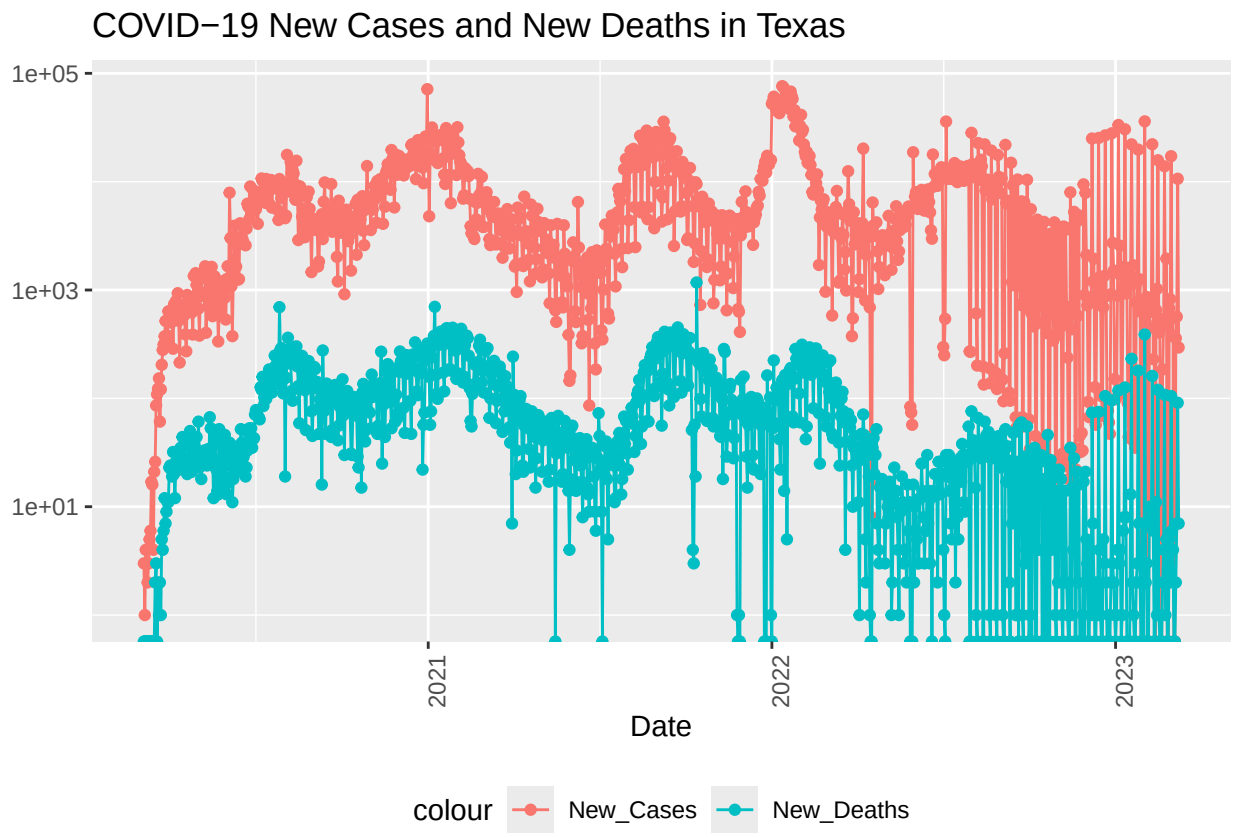

```
# Plot by State
state <- "New York"
US_by_State %>%
    filter(Province_State == state) %>%
    filter(Cases > 0) %>%
    ggplot(aes(x = Date)) + geom_line(aes(y = Cases, color = "Cases")) + geom_point(aes(y = Cases,
    color = "Cases")) + geom_line(aes(y = Deaths, color = "Deaths")) + geom_point(aes(y = Deaths,
    color = "Deaths")) + scale_y_log10() + theme(legend.position = "bottom", axis.text.x = element_text
    labs(title = str_c("COVID-19 Cases and Deaths in ", state), y = NULL)
#> Warning in scale_y_log10(): log-10 transformation introduced infinite values.
#> log-10 transformation introduced infinite values.
```




```
# Plot for New_Cases and New_Deaths
US_Totals %>%
    ggplot(aes(x = Date)) + geom_line(aes(y = New_Cases, color = "New_Cases")) +
    geom_point(aes(y = New_Cases, 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 = "COVID-19 New Cases and New Deaths in the US", y = NULL)
#> Warning in transformation$transform(x): NaNs produced
#> Warning in transformation$transform(x): log-10 transformation introduced
#> infinite values.
#> Warning in transformation$transform(x): NaNs produced
#> Warning in scale_y_log10(): log-10 transformation introduced infinite values.
#> Warning in transformation$transform(x): NaNs produced
#> Warning in scale_y_log10(): log-10 transformation introduced infinite values.
#> Warning in transformation$transform(x): NaNs produced
#> Warning in scale_y_log10(): log-10 transformation introduced infinite values.
#> Warning: Removed 1 row containing missing values or values outside the scale range
#> (`geom_line()`).
#> Warning: Removed 2 rows containing missing values or values outside the scale range
#> (`geom_point()`).
#> Warning: Removed 1 row containing missing values or values outside the scale range
#> (`geom_line()`).
#> Warning: Removed 4 rows containing missing values or values outside the scale range
#> (`geom_point()`).
```




```
# State Texas for New Cases and New Deaths
state <- "Texas"
US_by_State %>%
    filter(Province_State == state) %>%
    filter(Cases > 0) %>%
    ggplot(aes(x = Date)) + geom_line(aes(y = New_Cases, color = "New_Cases")) +
    geom_point(aes(y = New_Cases, 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 = str_c("COVID-19 New Cases and New Deaths in ", state), y = NULL)
#> Warning in transformation$transform(x): NaNs produced
#> Warning in scale_y_log10(): log-10 transformation introduced infinite values.
#> Warning in transformation$transform(x): NaNs produced
#> Warning in scale_y_log10(): log-10 transformation introduced infinite values.
#> Warning in transformation$transform(x): NaNs produced
#> Warning in scale_y_log10(): log-10 transformation introduced infinite values.
#> Warning in transformation$transform(x): NaNs produced
#> Warning in scale_y_log10(): log-10 transformation introduced infinite values.
#> Warning: Removed 1 row containing missing values or values outside the scale range
#> (`geom_point()`).
#> Warning: Removed 3 rows containing missing values or values outside the scale range
#> (`geom_point()`).
```




# Data Modeling

## Modeling

```
model <- lm(Deaths_per_thousand ~ Cases_per_thousand, data = US_State_Totals)

summary(model)
#>
#> Call:
#> lm(formula = Deaths_per_thousand ~ Cases_per_thousand, data = US_State_Totals)
#>
#> Residuals:
#>     Min      1Q  Median      3Q     Max
#> -2.3352 -0.5978  0.1491  0.6535  1.2086
#>
#> Coefficients:
#>                    Estimate Std. Error t value Pr(>|t|)
#> (Intercept)        -0.36167    0.72480  -0.499     0.62
#> Cases_per_thousand  0.01133    0.00232   4.881 9.76e-06 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Residual standard error: 0.8615 on 54 degrees of freedom
```

```
#> Multiple R-squared:  0.3061, Adjusted R-squared:  0.2933
#> F-statistic: 23.82 on 1 and 54 DF,  p-value: 9.763e-06

US_Totals_w_Pred <- US_State_Totals %>%
    mutate(Prediction = predict(model))
```

## Visualization with Model

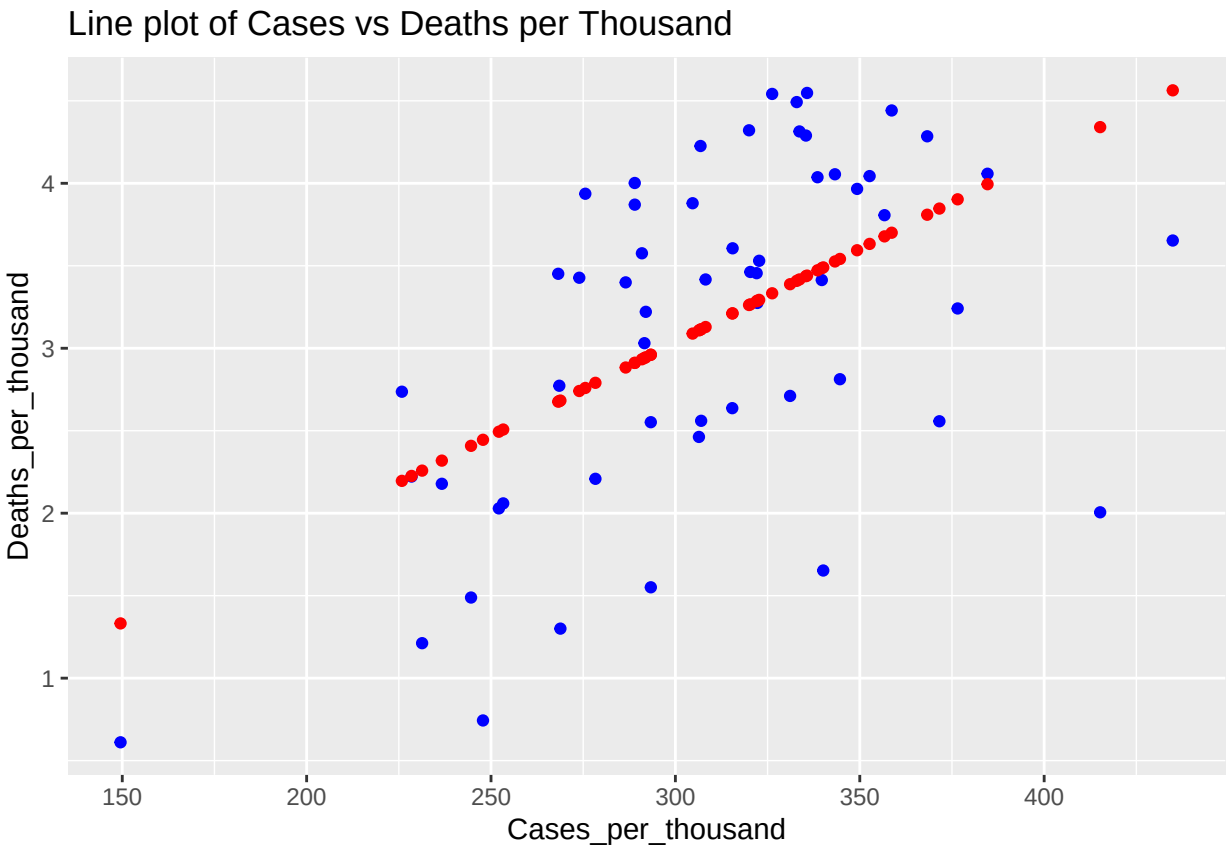
### Tables

```
# Create a table of results
kable(head(US_Totals_w_Pred))
```

| Province_State | Deaths | Cases | Population | Cases_per_thousand | Deaths_per_thousand | Prediction |
|---|---|---|---|---|---|---|
| Alabama | 21032 | 1644533 | 4903185 | 335.4010 | 4.2894568 | 3.436947 |
| Alaska | 1486 | 307655 | 740995 | 415.1917 | 2.0054116 | 4.340625 |
| American Samoa | 34 | 8320 | 55641 | 149.5300 | 0.6110602 | 1.331850 |
| Arizona | 33102 | 2443514 | 7278717 | 335.7067 | 4.5477795 | 3.440410 |
| Arkansas | 13020 | 1006883 | 3017804 | 333.6476 | 4.3143955 | 3.417089 |
| California | 101159 | 12129699 | 39512223 | 306.9860 | 2.5601951 | 3.115131 |

### Figures

```
# Display a figure
US_Totals_w_Pred %>%
    ggplot() + geom_point(aes(x = Cases_per_thousand, y = Deaths_per_thousand), color = "blue") +
    geom_point(aes(x = Cases_per_thousand, y = Prediction), color = "red") + labs(title = "Line plot of
```




# Conclusion and Sources of Bias

## Conclusion

In this project, we have conducted a comprehensive analysis of COVID-19 data, focusing on case trends and mortality rates in both the United States and globally. Our analysis reveals several key findings:

**Significant Findings**: Our data indicates a clear upward trend in COVID-19 cases and mortality rates, with notable differences across various regions. The analysis of the United States shows significant variability in case rates, while global trends exhibit a general pattern of increasing cases and deaths, albeit with regional differences in the rate of increase.

**Model Effectiveness**: The linear regression model we employed demonstrated that the number of cases per thousand is a statistically significant predictor of mortality rates. Despite this, the model's explanatory power is moderate, as indicated by an R-squared value of 0.3061. This suggests that while our model captures some of the variability in mortality rates, other factors may also be influencing these outcomes.

**Insights and Recommendations**: Based on our findings, it is clear that addressing COVID-19 effectively requires a multifaceted approach, considering both regional differences and broader global trends. Recommendations include enhancing targeted interventions in high-risk areas and continuing to monitor and adjust strategies based on emerging data.

## Sources of Bias

### Sources of Bias

Several potential sources of bias may have influenced the results of this project:

1. **Data Collection Bias**: The dataset, sourced from publicly available repositories, may contain incomplete or inconsistent data across regions. Some countries may underreport cases or deaths due to limited testing or political reasons, leading to potentially skewed results when comparing countries with more transparent data.
2. **Sampling Bias**: The data may not fully represent all populations, particularly in regions with limited reporting or delays. This could result in an overrepresentation of areas with better reporting infrastructure, skewing global or regional trends.
3. **Model Bias**: The linear regression model used assumes a simple linear relationship between cases per thousand and mortality rates. This model does not account for complex factors like healthcare quality or public health responses, potentially oversimplifying the relationships.
4. **Confirmation Bias**: There's a risk of interpreting results in a way that supports pre-existing assumptions or expectations, such as emphasizing trends that fit widely accepted theories while downplaying outliers or unexpected findings.

### Personal Bias and Mitigation

As the analyst, I acknowledge potential personal biases that could influence the interpretation of results:

**Personal Bias**: My background and prior experiences may lead to a preference for certain explanations or models. For instance, I might be inclined to emphasize findings that align with widely accepted theories or recent studies.

**Mitigation Strategies**:

**Diverse Perspectives**: To mitigate personal bias, I have incorporated feedback from colleagues and experts in the field, ensuring a more balanced interpretation of the data. Transparent Reporting: I have been transparent about the assumptions and limitations of the models used, providing a clear account of the potential sources of bias and their impact on the findings.

**Multiple Models**: Utilizing various analytical approaches and models helps to cross-verify results and reduce the influence of any single model's limitations.

By acknowledging and addressing these biases, the analysis aims to provide a more accurate and objective assessment of the COVID-19 data, supporting informed decision-making and policy development.

# References

## Key Points

- **Metadata Section:** The YAML header at the top (`---`) includes the document title, author, date, and output format. You can change `html_document` to `pdf_document` or `word_document` depending on your needs.
- **Code Chunks:** The ```` ```{r} ... ``` ```` syntax is used to include R code. Code chunks are labeled with `{r}` and you can include additional options for controlling their behavior, such as `echo=FALSE` to hide the code.
- **Text Sections:** Regular Markdown syntax is used for text, headers, lists, and other formatting.