Covid-19 Final Report

Lillian Brown

2022-02-22

Contents

Introduction

This report is written as part of the class, 'Data Science as a Field' from the University of Colorado Boulder Masters in Data Science program.

The data analysed in this report is published by the Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE).

See additional information at: https://github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data/csse_covid_19_time_series

Additional data is sourced from the US Center for Disease Control and Prevention and can be found at: https://data.cdc.gov/Vaccinations/COVID-19-Vaccinations-in-the-United-States-Jurisdi/unsk-b7fc

Import Data

The following code was used to import the data and read in necessary librarires:

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr   0.3.5
## v tibble  3.1.6      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.0
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
```

```
##     date, intersect, setdiff, union
```

```
library(ggplot2)
library(dplyr)
library(ggstream)
library(hrbrthemes)
```

```
## NOTE: Either Arial Narrow or Roboto Condensed fonts are required to use these themes.
##       Please use hrbrthemes::import_roboto_condensed() to install Roboto Condensed and
##       if Arial Narrow is not on your system, please see https://bit.ly/arialnarrow
```

```
library(modelr)
library(geofacet)
library(wesanderson)


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

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

uid_lookup_url <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/U

usa_vaccination_url <- "https://data.cdc.gov/api/views/unsk-b7fc/rows.csv?accessType=DOWNLOAD"

urls <- str_c(url_in, file_names)

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

```
## Rows: 3342 Columns: 1140
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (6): iso2, iso3, Admin2, Province_State, Country_Region, Combined_Key
## dbl (1134): 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.
```

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

```
## Rows: 3342 Columns: 1141
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (6): iso2, iso3, Admin2, Province_State, Country_Region, Combined_Key
## dbl (1135): UID, code3, FIPS, Lat, Long_, Population, 1/22/20, 1/23/20, 1/24...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

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

```
## Rows: 289 Columns: 1133
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (2): Province/State, Country/Region
```

```
## dbl (1131): 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.
```

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

```
## Rows: 289 Columns: 1133
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (2): Province/State, Country/Region
## dbl (1131): 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.
```

```
uid <- read_csv(uid_lookup_url)
```

```
## 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.
```

```
usa_vaccination <- read_csv(usa_vaccination_url)
```

```
## Rows: 37784 Columns: 109
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (2): Date, Location
## dbl (107): MMWR_week, Distributed, Distributed_Janssen, Distributed_Moderna,...
##
## 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.
```

Tidy and Transform Data

In order to tidy and transform the data for analysis, I removed unused columns and added global population data so that the US and global data sets contained the same data with the same column names, changed date to a date object and Province_State and Country_Region to factors.

Noticing that there were three lines in the US data set where the cases are negative, (North Carolina on 2022-11-09 and South Carolina on 2022-05-05 and 2022-05-06,) I remove those lines.

Also noticing the first Ohio Covid-19 death was 2020-03-10 (https://governor.ohio.gov/media/news-and-media/ohio-records-first-covid19-death-senior-centers-adult-day-cares-to-close), but that there was a single death labeled in days prior despite there being no recorded Covid-19 cases, those rows were removed.

Below is the code to make those changes and a summary of the data to be analyzed:

```
usa_cases <- usa_cases %>%
  pivot_longer(cols = -(UID:Combined_Key),
               names_to = "date",
               values_to = "cases") %>%
  select(Admin2:cases) %>%
```

```
  mutate(date = mdy(date)) %>%
  select(-c(Lat,
            Long_))

usa_deaths <- usa_deaths %>%
  pivot_longer(cols = -(UID:Population),
               names_to = "date",
               values_to = "deaths") %>%
  select(Admin2:deaths) %>%
  mutate(date = mdy(date)) %>%
  select(-c(Lat,
            Long_))

usa <- usa_cases %>%
  full_join(usa_deaths)
```

```
## Joining, by = c("Admin2", "Province_State", "Country_Region", "Combined_Key",
## "date")
```

```
# Remove 3 Lines where Covid-19 cases are negative (1 in North Carolina and 2 in South Carolina).
usa <- usa %>%
  filter(cases>=0)

usa$Combined_Key = factor(usa$Combined_Key)
usa$Province_State = factor(usa$Province_State)
usa$Country_Region = factor(usa$Country_Region)

# First Ohio Covid-19 death was 2020-03-10, removing those rows listing a death but no cases.
usa$deaths <- ifelse((usa$cases == 0 &
                        usa$deaths == 1 &
                        usa$Province_State == 'Ohio' &
                        usa$Admin2 == 'Hamilton' &
                        usa$date < '2020-03-10'),
                     0,
                     usa$deaths)




global_cases <- global_cases %>%
  pivot_longer(cols = -c('Province/State',
                         'Country/Region',
                         'Lat',
                         'Long'),
               names_to = 'date',
               values_to = 'cases') %>%
  mutate(date = mdy(date)) %>%
  select(-c(Lat,
            Long))

global_deaths <- global_deaths %>%
  pivot_longer(cols = -c('Province/State',
                         'Country/Region',
                         'Lat',
```

```
                        'Long'),
               names_to = 'date',
               values_to = 'deaths') %>%
  mutate(date = mdy(date)) %>%
  select(-c(Lat,
            Long))

global <- global_cases %>%
  full_join(global_deaths) %>%
  rename(Country_Region = 'Country/Region',
         Province_State = 'Province/State')
```

```
## Joining, by = c("Province/State", "Country/Region", "date")
```

```
global <- global %>%
  left_join(uid, by = c("Province_State",
                        "Country_Region")) %>%
  select(-c(UID, FIPS)) %>%
  select(Province_State,
         Country_Region,
         date,
         cases,
         deaths,
         Population)

global <- global %>%
  unite("Combined_Key",
        c(Province_State,
          Country_Region),
        sep = ", ",
        na.rm = TRUE,
        remove = FALSE)


global$Combined_Key = factor(global$Combined_Key)
global$Province_State = factor(global$Province_State)
global$Country_Region = factor(global$Country_Region)

summary(usa)
```

```
##     Admin2           Province_State    Country_Region
##  Length:3773115     Texas   : 289024   US:3773115
##  Class :character   Georgia : 181769
##  Mode  :character   Virginia: 152415
##                     Kentucky: 137738
##                     Missouri: 133222
##                     Kansas  : 120803
##                     (Other) :2758144
##                         Combined_Key          date                cases
##  Abbeville, South Carolina, US:   1129   Min.   :2020-01-22   Min.   :      0
##  Acadia, Louisiana, US        :   1129   1st Qu.:2020-10-30   1st Qu.:    320
##  Accomack, Virginia, US       :   1129   Median :2021-08-08   Median :   2230
##  Ada, Idaho, US               :   1129   Mean   :2021-08-07   Mean   :  13878
##  Adair, Iowa, US              :   1129   3rd Qu.:2022-05-17   3rd Qu.:   8019
```

```
##  Adair, Kentucky, US          :   1129   Max.   :2023-02-23   Max.   :3696875
##  (Other)                      :3766341
##    Population           deaths
##  Min.   :       0   Min.   :    0.0
##  1st Qu.:    9917   1st Qu.:    4.0
##  Median :   24909   Median :   37.0
##  Mean   :   99604   Mean   :  185.1
##  3rd Qu.:   64979   3rd Qu.:  121.0
##  Max.   :10039107   Max.   :35366.0
##
```

```
summary(global)
```

```
##           Combined_Key                           Province_State
##  Afghanistan    :  1129   Alberta                       :  1129
##  Albania        :  1129   Anguilla                      :  1129
##  Alberta, Canada:  1129   Anhui                         :  1129
##  Algeria        :  1129   Aruba                         :  1129
##  Andorra        :  1129   Australian Capital Territory:  1129
##  Angola         :  1129   (Other)                       : 97094
##  (Other)        :319507   NA's                          :223542
##         Country_Region        date                cases
##  China          : 38386   Min.   :2020-01-22   Min.   :        0
##  Canada         : 18064   1st Qu.:2020-10-30   1st Qu.:      657
##  United Kingdom: 16935   Median :2021-08-08   Median :    13865
##  France         : 13548   Mean   :2021-08-08   Mean   :   942287
##  Australia      :  9032   3rd Qu.:2022-05-17   3rd Qu.:   224699
##  Netherlands    :  5645   Max.   :2023-02-23   Max.   :103355824
##  (Other)        :224671
##      deaths           Population
##  Min.   :      0   Min.   :6.700e+01
##  1st Qu.:      3   1st Qu.:5.790e+05
##  Median :    146   Median :6.574e+06
##  Mean   :  13251   Mean   :2.769e+07
##  3rd Qu.:   2991   3rd Qu.:2.642e+07
##  Max.   :1119508   Max.   :1.380e+09
##                    NA's   :10161
```

Visualizations and Analysis

In order to begin visualizing and analyzing the data, the following code creates totals based on province/state and country/region along with new calculations: deaths_per_million, new_cases, and new_deaths :

```
usa_by_state  <- usa %>%
  group_by(Province_State,
           Country_Region,
           date) %>%
  summarize(cases = sum(cases),
            deaths = sum(deaths),
            Population = sum(Population)) %>%
  mutate(deaths_per_million = deaths * 1000000 / 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.
```

```
# Noticing infinity on deaths_per_million from special geographical regions with Population = 0, return

usa_by_state$deaths_per_million <- ifelse(usa_by_state$Population > 0, usa_by_state$deaths_per_million,

summary(usa_by_state)
```

```
##         Province_State  Country_Region      date                cases
##  Alabama        : 1129   US:65482       Min.   :2020-01-22   Min.   :       0
##  Alaska         : 1129                  1st Qu.:2020-10-30   1st Qu.:   29860
##  American Samoa: 1129                   Median :2021-08-08   Median :  286348
##  Arizona        : 1129                  Mean   :2021-08-08   Mean   :  799658
##  Arkansas       : 1129                  3rd Qu.:2022-05-17   3rd Qu.:  938864
##  California     : 1129                  Max.   :2023-02-23   Max.   :12080387
##  (Other)        :58708
##      deaths       deaths_per_million   Population
##  Min.   :     0   Min.   :   0.0     Min.   :       0
##  1st Qu.:   537   1st Qu.: 452.8     1st Qu.: 1068778
##  Median :  3770   Median :1619.1     Median : 3660113
##  Mean   : 10662   Mean   :1671.6     Mean   : 5739226
##  3rd Qu.: 13508   3rd Qu.:2675.1     3rd Qu.: 6892503
##  Max.   :100816   Max.   :4539.5     Max.   :39512223
##                   NA's   :2258
```

```
usa_totals <- usa_by_state %>%
  group_by(Country_Region,
           date) %>%
  summarize(cases = sum(cases),
            deaths = sum(deaths),
            Population = sum(Population)) %>%
  mutate(deaths_per_million = deaths * 1000000 / 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.
```

```
usa_totals <- usa_totals %>%
mutate(new_cases = cases - lag(cases),
       new_deaths = deaths - lag(deaths))

summary(usa_totals)
```

```
##  Country_Region      date                cases               deaths
```

```
##  US:1129        Min.   :2020-01-22   Min.   :        1   Min.   :      0
##                 1st Qu.:2020-10-30   1st Qu.:  9089579   1st Qu.: 229576
##                 Median :2021-08-08   Median : 35921986   Median : 613289
##                 Mean   :2021-08-08   Mean   : 46380143   Mean   : 618424
##                 3rd Qu.:2022-05-17   3rd Qu.: 82859682   3rd Qu.:1002288
##                 Max.   :2023-02-23   Max.   :103355824   Max.   :1119508
##
##  deaths_per_million   Population          new_cases         new_deaths
##  Min.   :   0.0     Min.   :332875137   Min.   :      0   Min.   :-254.0
##  1st Qu.: 689.7     1st Qu.:332875137   1st Qu.:  26306   1st Qu.: 327.8
##  Median :1842.4     Median :332875137   Median :  56247   Median : 710.5
##  Mean   :1857.8     Mean   :332875137   Mean   :  91628   Mean   : 992.5
##  3rd Qu.:3011.0     3rd Qu.:332875137   3rd Qu.: 112790   3rd Qu.:1422.8
##  Max.   :3363.1     Max.   :332875137   Max.   :1354500   Max.   :4377.0
##                                         NA's   :1         NA's   :1
```

```
global_by_provstate <- global %>%
  group_by(Province_State,
           Country_Region,
           date) %>%
  summarize(cases = sum(cases),
            deaths = sum(deaths),
            Population = sum(Population)) %>%
  mutate(deaths_per_million = deaths * 1000000 / 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.
```

```
summary(global_by_provstate)
```

```
##                       Province_State          Country_Region
##  Alberta                      :  1129   China         : 38386
##  Anguilla                     :  1129   Canada        : 18064
##  Anhui                        :  1129   United Kingdom: 16935
##  Aruba                        :  1129   France        : 13548
##  Australian Capital Territory:  1129   Australia     :  9032
##  (Other)                      : 97094   Netherlands   :  5645
##  NA's                         :223542   (Other)       :224671
##       date                cases              deaths        deaths_per_million
##  Min.   :2020-01-22   Min.   :        0   Min.   :      0   Min.   :   0.000
##  1st Qu.:2020-10-30   1st Qu.:      657   1st Qu.:      3   1st Qu.:   0.408
##  Median :2021-08-08   Median :    13865   Median :    146   Median :  79.360
##  Mean   :2021-08-08   Mean   :   942287   Mean   :  13251   Mean   : 552.181
##  3rd Qu.:2022-05-17   3rd Qu.:   224699   3rd Qu.:   2991   3rd Qu.: 734.721
##  Max.   :2023-02-23   Max.   :103355824   Max.   :1119508   Max.   :6653.768
##                                                             NA's   :10161
##    Population
```

```
##  Min.   :6.700e+01
##  1st Qu.:5.790e+05
##  Median :6.574e+06
##  Mean   :2.769e+07
##  3rd Qu.:2.642e+07
##  Max.   :1.380e+09
##  NA's   :10161
```

```
global_totals <- global_by_provstate %>%
  group_by(Country_Region,
           date) %>%
  summarize(cases = sum(cases),
            deaths = sum(deaths),
            Population = sum(Population)) %>%
  mutate(deaths_per_million = deaths * 1000000 / 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.
```

```
summary(global_totals)
```

```
##      Country_Region        date                cases
##  Afghanistan:  1129   Min.   :2020-01-22   Min.   :        0
##  Albania    :  1129   1st Qu.:2020-10-30   1st Qu.:     3680
##  Algeria    :  1129   Median :2021-08-08   Median :    51002
##  Andorra    :  1129   Mean   :2021-08-08   Mean   :  1354830
##  Angola     :  1129   3rd Qu.:2022-05-17   3rd Qu.:   490533
##  Antarctica :  1129   Max.   :2023-02-23   Max.   :103355824
##  (Other)    :220155
##      deaths        deaths_per_million   Population
##  Min.   :      0   Min.   :   0.00    Min.   :8.090e+02
##  1st Qu.:     44   1st Qu.:  12.33    1st Qu.:1.886e+06
##  Median :    768   Median : 140.69    Median :8.696e+06
##  Mean   :  19052   Mean   : 663.36    Mean   :3.246e+07
##  3rd Qu.:   7118   3rd Qu.: 965.58    3rd Qu.:2.769e+07
##  Max.   :1119508   Max.   :6653.77    Max.   :1.380e+09
##                    NA's   :7903       NA's   :7903
```

Given that the smallest US state by population (Wyoming) is between 500,000 and 600,000 people and the largest state by population (California) is between 39,000,000 and 40,000,000, in order to account for this population discrepancy in an initial visualization, the following plot shows the US Covid-19 deaths per million people.

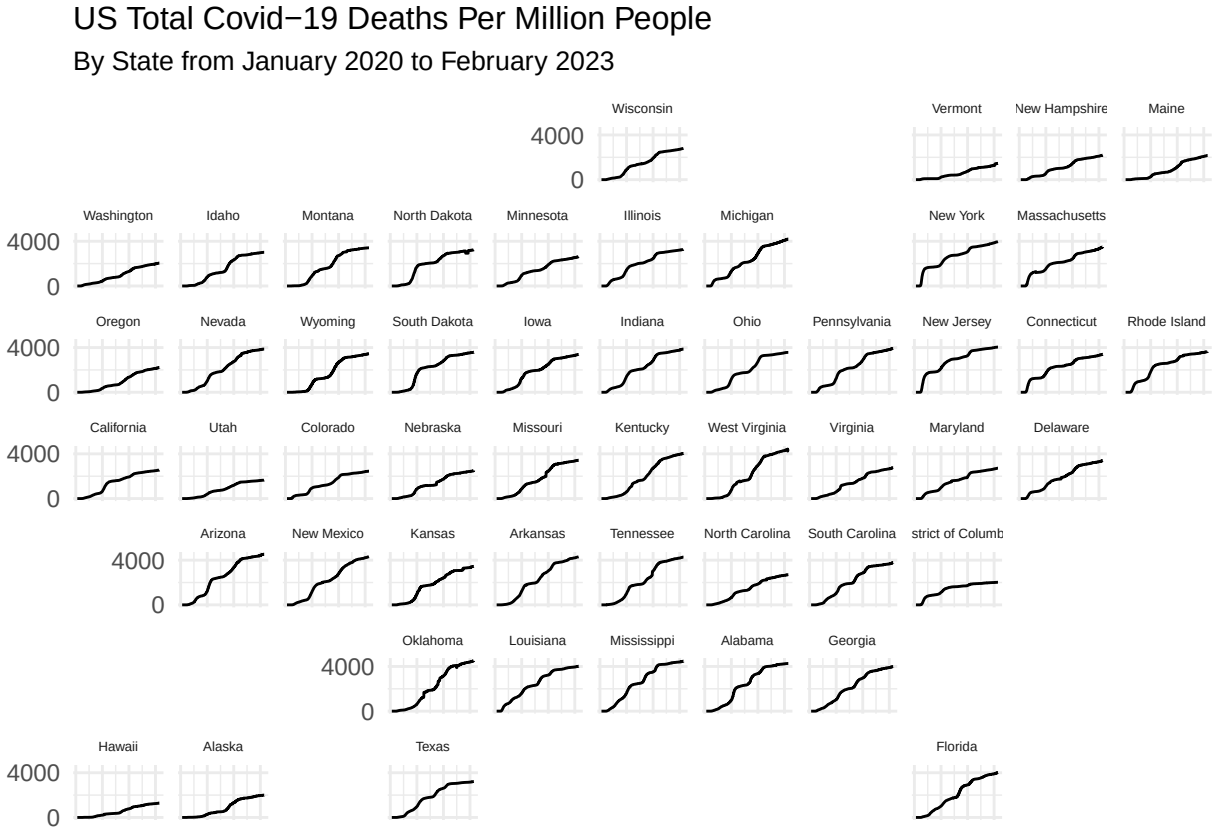
```
plot_state <- filter(usa_by_state, Province_State != "American Samoa" &
                       Province_State != "Diamond Princess" &
                       Province_State != "Grand Princess" &
                       Province_State != "Guam" &
                       Province_State != "Mariana Islands" &
                       Province_State != "Northern Mariana Islands" &
```

```
                        Province_State != "Puerto Rico" &
                        Province_State != "Virgin Islands")

plot_state <- droplevels(plot_state)

plot_state$State_Abbr = state.abb[match(plot_state$Province_State, state.name)]
plot_state$State_Abbr = factor(plot_state$State_Abbr)

ggplot(plot_state, aes(date, deaths_per_million)) +
  geom_line() +
  scale_y_continuous(breaks = c(0,4000)) +
  facet_geo(~Province_State) +
  labs(title = "US Total Covid-19 Deaths Per Million People",
       subtitle = "By State from January 2020 to February 2023",
       x = element_blank(),
       y = element_blank() ) +
  theme_minimal() +
  theme(
    strip.text = element_text(size = 5),
    axis.text.x = element_blank(),
    axis.ticks.x = element_blank(),
    axis.line.x = element_blank(),
    axis.line.y = element_blank(),
    plot.title = element_text(size = 12),
    plot.subtitle = element_text(size = 10),
    strip.background = element_rect(color = "white"))
```

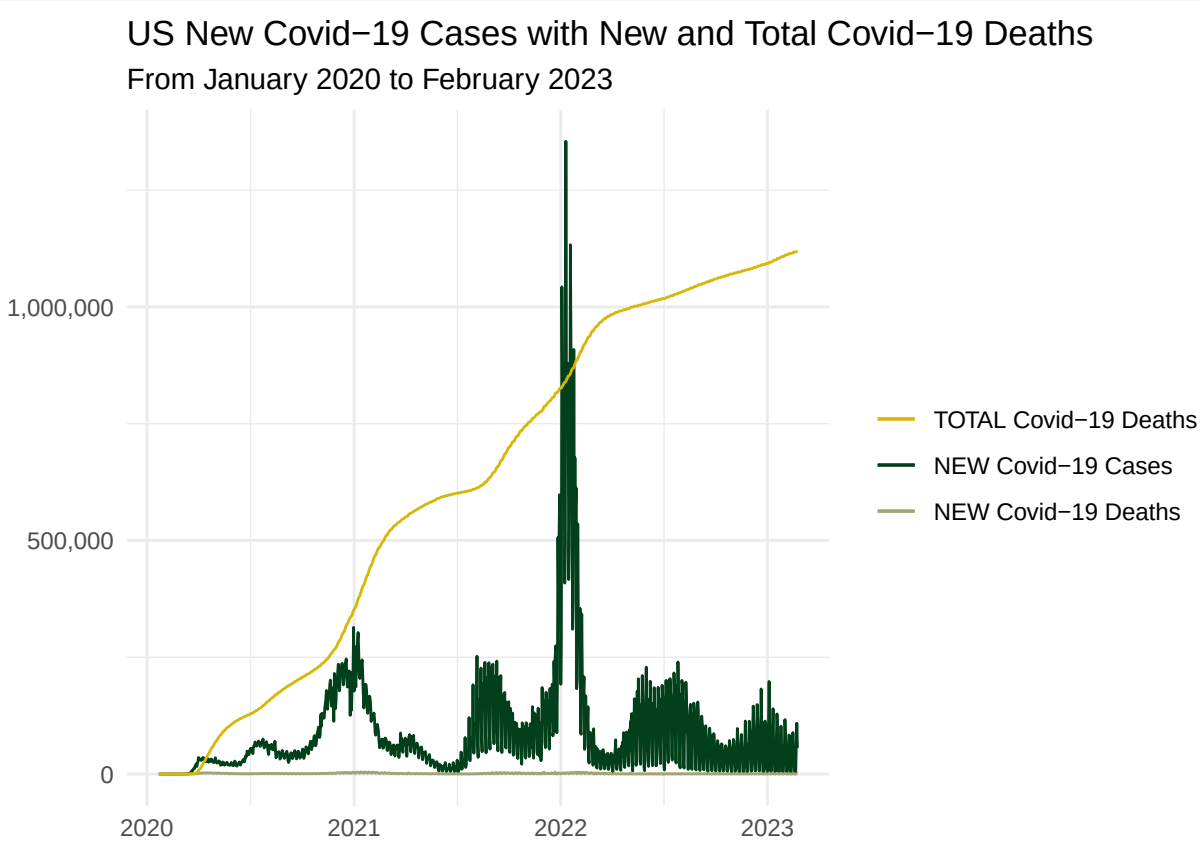


Given that over the course of the pandemic, a number of factors varied substantially, such as the virus

variants, ability to treat Covid-19 patients, lockdowns and other precautions, availability of testing, and availability of vaccines, in order to see what impact these may have had over Covid-19 deaths, the following plots show the new Covid-19 cases against new and total Covid-19 deaths.

```
plot_usa <- usa_totals %>%
  mutate(new_cases = ifelse(is.na(new_cases), 0, new_cases)) %>%
  mutate(new_deaths = ifelse(is.na(new_deaths), 0, new_deaths)) %>%
  ggplot(aes(x=date, y=new_cases)) +
  geom_line(aes(color = "new_cases")) +
  geom_line(aes(y= new_deaths, color = "new_deaths")) +
  geom_line(aes(y= deaths, color = "deaths")) +
  theme_minimal() +
  labs(title = "US New Covid-19 Cases with New and Total Covid-19 Deaths",
       subtitle = "From January 2020 to February 2023",
       x = element_blank(),
       y = element_blank() ) +
  scale_color_manual(name = NULL,
                     values = wes_palette(name="Cavalcanti1", n=3),
                     labels=c("TOTAL Covid-19 Deaths", "NEW Covid-19 Cases", "NEW Covid-19 Deaths"))+
  scale_y_continuous(labels = scales::comma_format())

plot_usa
```



Comparing the US Covid-19 deaths per million people to the global Covid-19 deaths per million people:

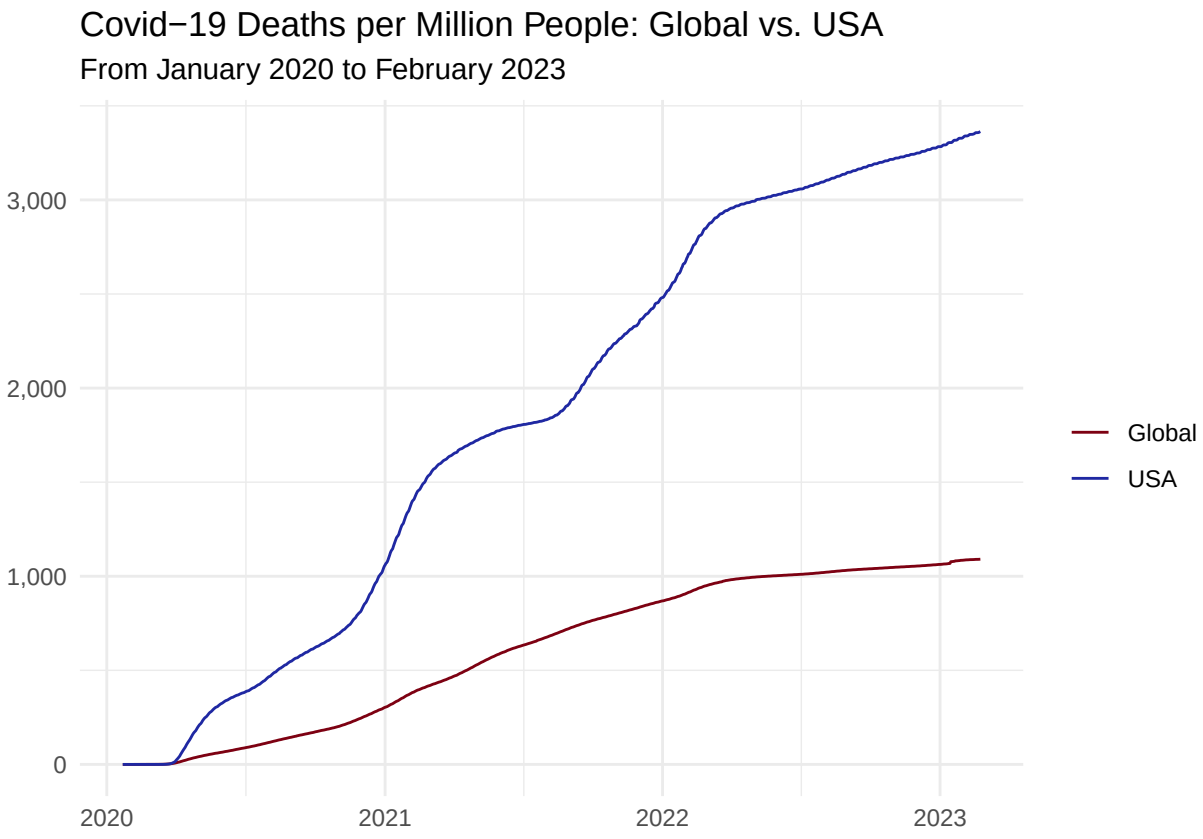
```
plot_global <- global_totals %>%
  mutate(Population = ifelse(is.na(Population), 0, Population)) %>%
  group_by(date) %>%
  summarize(cases = sum(cases),
```

```
            deaths = sum(deaths),
            Population = sum(Population)) %>%
  mutate(deaths_per_million = deaths * 1000000 / Population) %>%
  select(date,
         cases,
         deaths,
         deaths_per_million,
         Population) %>%
  ungroup()

df_usa_global <- bind_rows(
  usa_totals %>% mutate(df = "USA"),
  plot_global %>% mutate(df = "Global")
)

plot_usa_global <- ggplot(df_usa_global, aes(x = date, y = deaths_per_million, group = df)) +
  geom_line(aes(color = df)) +
  theme_minimal() +
  labs(title = "Covid-19 Deaths per Million People: Global vs. USA",
       subtitle = "From January 2020 to February 2023",
       x = element_blank(),
       y = element_blank() ) +
  scale_color_manual(name = NULL,
                     values = hcl.colors(n=2, palette = "Roma")) +
  scale_y_continuous(labels = scales::comma_format())

plot_usa_global
```



From "US Total Covid-19 Deaths Per Million People", it is evident that there is a substantial range of Covid-19 deaths per million people by state. While no state has yet (by February 2023) exceeded approximately 4,600 deaths per million people, and no state has fewer that 1,200 deaths per million people, the range between is quite substantial. The most populous state, California, has 2546 Covid-19 deaths per million people (as of February 22, 2023), while the least populous state, Wyoming, has 3452 Covid-19 deaths per million people (as of February 22, 2023). While not every largely populated state has fewer Covid-19 deaths than the less populated sates, it does suggest that population is not the only factor in Covid-19 deaths.

The plot "US New Covid-19 Cases with New and Total Covid-19 Death" illustrates the gap between new Covid-19 cases and new Covid-19 deaths. As of February 22, 2022, the maximum of new cases is 1354503 while the maximum of new deaths is 4377. In order to more clearly see the impact of new Covid-19 cases over the course of the pandemic on Covid-19 deaths, the total Covid-19 deaths is shown. From that it is apparent that the massive increase in new Covid-19 cases at the beginning of 2022, did not result in an increase of Covid-19 deaths proportional to the number of deaths from new Covid-19 cases earlier in the pandemic.

"Covid-19 Deaths per Million People: Global vs. USA" shows that the US Covid-19 deaths per million people is substantially higher than the global total.

Model

The following model incorporates data from the US CDC on doses of Covid-19 vaccination administered. The model predicts US Covid-19 deaths per million people from cases, population, and vaccine doses administered.

```
# US Vaccination Data
usa_vaccination <- usa_vaccination %>%
  mutate(Date = mdy(Date)) %>%
  select(c(Date,
           Location,
           Administered)) %>%
  rename(date = Date)

usa_vaccination$Location <- as.factor(usa_vaccination$Location)

usa_vaccination <- filter(usa_vaccination, Location == "US") %>%
  rename(Country_Region = Location)
usa_vaccination <- droplevels(usa_vaccination)

usa_totals_with_vac <- usa_totals %>%
  left_join(usa_vaccination, by = c("date",
                                    "Country_Region"))

first_vac <- as.Date("2020-12-13")

usa_totals_with_vac <-  usa_totals_with_vac %>%
  mutate(Administered = ifelse(is.na(Administered) & date< first_vac, 0, Administered))

# Model

lm_usa <- lm(deaths_per_million ~ 0 + cases + Population + Administered, data = usa_totals_with_vac)
lm_usa
```

```
##
## Call:
## lm(formula = deaths_per_million ~ 0 + cases + Population + Administered,
##     data = usa_totals_with_vac)
##
```

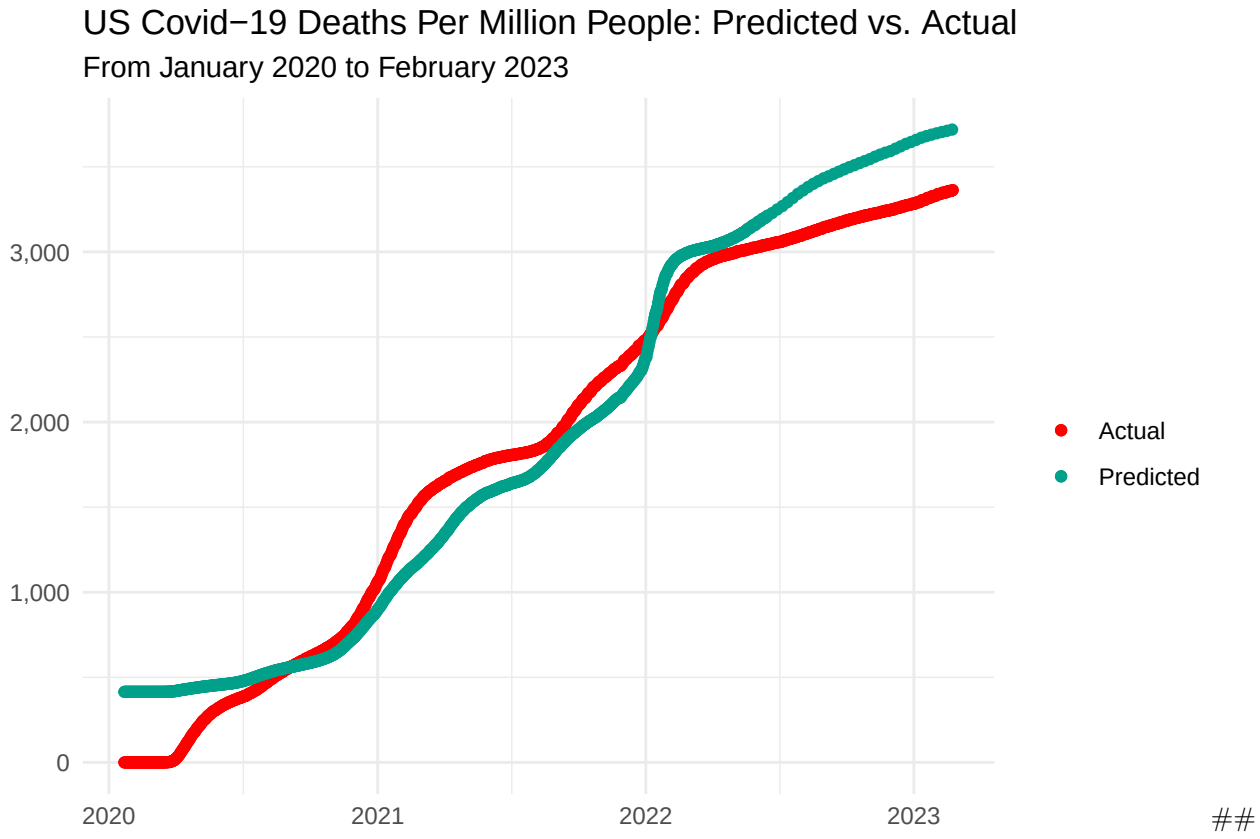
```
## Coefficients:
##        cases    Population  Administered
##    2.343e-05     1.245e-06     1.316e-06
```

```
predicted_deathspermill <- data.frame(PREDICTED = predict(lm_usa,
                                                          usa_totals_with_vac),
                                      cases = usa_totals_with_vac$cases,
                                      Population = usa_totals_with_vac$Population,
                                      Administered = usa_totals_with_vac$Administered)

predicted_vs_actual <- usa_totals
predicted_vs_actual$PREDICTED <- predicted_deathspermill$PREDICTED

plot_lm_usa <- predicted_vs_actual %>%
  ggplot(aes(x=date, y=deaths_per_million)) +
  geom_point(aes(color = "deaths_per_million"))+
  geom_point(aes(y= PREDICTED, color = "PREDICTED"), na.rm = TRUE) +
  theme_minimal() +
  labs(title = "US Covid-19 Deaths Per Million People: Predicted vs. Actual",
       subtitle = "From January 2020 to February 2023",
       x = element_blank(),
       y = element_blank() ) +
  scale_color_manual(name = NULL,
                     values = wes_palette(name="Darjeeling1", n=2),
                     labels=c("Actual", "Predicted"))+
  scale_y_continuous(labels = scales::comma_format())

plot_lm_usa
```



##

Further Analysis

While there are some general observations made from this report, there majority of Covid-19 factors that have yet to be explored. Most of the global data is left unexplored and age of population, population density, reporting quality, access to vaccine, and government policy are just a few of the additional unexplored factors.

Bias Identification

As an American, the majority of this report was focused on US data. I've also received 4 total doses of Covid-19 vaccine and am pro-vaccination. I also choose to primarily to focus on US data as the data set had shared external and reporting characteristics (such as federal government, reporting body, and vaccine types,) and wanted to see whether population size had an impact on the rate of Covid-19 deaths.