COVID Project

Anonymized for Project Submission

October 6, 2023

Introduction

In this document, I will show data cleaning and analysis, following the steps from the lecture videos.

Libraries

The libraries used for this assignment are: ggplot2, dplyr, knitr, rmarkdown, readr, tidyverse, and lubridate.

Dataset

Below are the datasets used within this document. Each of the links goes directly to a raw CSV file that can be downloaded from github. The more human-readable github is: https://github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data/csse_covid_19_time_series

```
global_cases <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19

global_deaths <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_

US_cases <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_da

US_deaths <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_da
```

Head

A brief inspection of the datasets. The variable inside of the head() function can be changed to inspect each dataset.

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

```
## # A tibble: 5 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
## # 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>, ...
```

Pre-cleaning the data in order to make it more readable and usable. Here, we are combining Province/State and Country/Region into one more readable combination. We are also dropping latitude and longitude as those are not necessary for this analysis due to how specific they get when country/state/city is enough.

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

global_deaths <- global_deaths %>%
  pivot_longer(cols = -c(`Province/State`,
                         `Country/Region`, Lat, Long),
               names_to = "date",
               values_to = "deaths") %>%
  select(-c(Lat, Long))
```

```
#global_cases
global_deaths
```

```
## # A tibble: 330,327 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
##  4 <NA>             Afghanistan      1/25/20      0
##  5 <NA>             Afghanistan      1/26/20      0
##  6 <NA>             Afghanistan      1/27/20      0
##  7 <NA>             Afghanistan      1/28/20      0
##  8 <NA>             Afghanistan      1/29/20      0
##  9 <NA>             Afghanistan      1/30/20      0
## 10 <NA>             Afghanistan      1/31/20      0
## # i 330,317 more rows
```

Renaming the country/region and Province/State columns to use an underscore. Also, here we are using the lubridate library to convert the date into a proper date format and data type.

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

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

A brief inspection of the variable we created in order to confirm that everything worked correctly.

```
global
```

```
## # A tibble: 330,327 x 5
##    Province_State Country_Region date       cases deaths
##    <chr>          <chr>          <date>     <dbl>  <dbl>
##  1 <NA>           Afghanistan    2020-01-22     0      0
##  2 <NA>           Afghanistan    2020-01-23     0      0
##  3 <NA>           Afghanistan    2020-01-24     0      0
##  4 <NA>           Afghanistan    2020-01-25     0      0
##  5 <NA>           Afghanistan    2020-01-26     0      0
##  6 <NA>           Afghanistan    2020-01-27     0      0
##  7 <NA>           Afghanistan    2020-01-28     0      0
##  8 <NA>           Afghanistan    2020-01-29     0      0
##  9 <NA>           Afghanistan    2020-01-30     0      0
## 10 <NA>           Afghanistan    2020-01-31     0      0
## # i 330,317 more rows
```

Filtering out the rows so that we only see rows where the cases is greater than 0, which means that COVID has been caught by someone there. We also look at the summary statistics of the dataset to confirm that there are no unusual values, such as a negative minimum in cases or an impossible maximum.

```
global <- global %>% filter(cases > 0)
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
##  Min.   :      0
##  1st Qu.:      7
##  Median :    214
##  Mean   :  14405
##  3rd Qu.:   3665
##  Max.   :1123836
```

Repeating the same steps for the US cases dataset: pivot, create a variable, join, and then inspect.

```
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_))
```

```
US_cases
```

```
## # A tibble: 3,819,906 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
##  4 Autauga Alabama        US             Autauga, Alabama, US 2020-01-25     0
##  5 Autauga Alabama        US             Autauga, Alabama, US 2020-01-26     0
##  6 Autauga Alabama        US             Autauga, Alabama, US 2020-01-27     0
##  7 Autauga Alabama        US             Autauga, Alabama, US 2020-01-28     0
##  8 Autauga Alabama        US             Autauga, Alabama, US 2020-01-29     0
##  9 Autauga Alabama        US             Autauga, Alabama, US 2020-01-30     0
## 10 Autauga Alabama        US             Autauga, Alabama, US 2020-01-31     0
## # i 3,819,896 more rows
```

Similar steps for the US deaths dataset.

```
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_))
```

```
US_deaths
```

```
## # A tibble: 3,819,906 x 7
##    Admin2  Province_State Country_Region Combined_Key      Population date
##    <chr>   <chr>          <chr>          <chr>                 <dbl> <date>
##  1 Autauga Alabama        US             Autauga, Alabama~      55869 2020-01-22
##  2 Autauga Alabama        US             Autauga, Alabama~      55869 2020-01-23
##  3 Autauga Alabama        US             Autauga, Alabama~      55869 2020-01-24
##  4 Autauga Alabama        US             Autauga, Alabama~      55869 2020-01-25
##  5 Autauga Alabama        US             Autauga, Alabama~      55869 2020-01-26
##  6 Autauga Alabama        US             Autauga, Alabama~      55869 2020-01-27
##  7 Autauga Alabama        US             Autauga, Alabama~      55869 2020-01-28
##  8 Autauga Alabama        US             Autauga, Alabama~      55869 2020-01-29
##  9 Autauga Alabama        US             Autauga, Alabama~      55869 2020-01-30
## 10 Autauga Alabama        US             Autauga, Alabama~      55869 2020-01-31
## # i 3,819,896 more rows
## # i 1 more variable: deaths <dbl>
```

This join will combine cases and deaths into one variable for the US. Its output will give us city, state, country (which should only be US or US territories), date, number of cases, total population of the city, and the number of deaths.

```
US <- US_cases %>%
  full_join(US_deaths)
```

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

Inspecting the US variable.

```
US
```

```
## # A tibble: 3,819,906 x 8
##    Admin2 Province_State Country_Region Combined_Key date       cases Population
##    <chr>  <chr>          <chr>          <chr>        <date>     <dbl>      <dbl>
##  1 Autau~ Alabama        US             Autauga, Al~ 2020-01-22     0      55869
##  2 Autau~ Alabama        US             Autauga, Al~ 2020-01-23     0      55869
##  3 Autau~ Alabama        US             Autauga, Al~ 2020-01-24     0      55869
##  4 Autau~ Alabama        US             Autauga, Al~ 2020-01-25     0      55869
##  5 Autau~ Alabama        US             Autauga, Al~ 2020-01-26     0      55869
##  6 Autau~ Alabama        US             Autauga, Al~ 2020-01-27     0      55869
##  7 Autau~ Alabama        US             Autauga, Al~ 2020-01-28     0      55869
##  8 Autau~ Alabama        US             Autauga, Al~ 2020-01-29     0      55869
##  9 Autau~ Alabama        US             Autauga, Al~ 2020-01-30     0      55869
## 10 Autau~ Alabama        US             Autauga, Al~ 2020-01-31     0      55869
## # i 3,819,896 more rows
## # i 1 more variable: deaths <dbl>
```

Using the unite() function, we create a new column called "Combined_Key", which will take data from the Province_State region. If there is no data, it will defer to the Country_Region column. If both of those columns have an NA value for that row, the row is removed from the dataset.

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

```
## # A tibble: 306,827 x 6
##    Combined_Key Province_State Country_Region date       cases deaths
##    <chr>        <chr>          <chr>          <date>     <dbl>  <dbl>
##  1 Afghanistan  <NA>           Afghanistan    2020-02-24     5      0
##  2 Afghanistan  <NA>           Afghanistan    2020-02-25     5      0
##  3 Afghanistan  <NA>           Afghanistan    2020-02-26     5      0
##  4 Afghanistan  <NA>           Afghanistan    2020-02-27     5      0
##  5 Afghanistan  <NA>           Afghanistan    2020-02-28     5      0
##  6 Afghanistan  <NA>           Afghanistan    2020-02-29     5      0
##  7 Afghanistan  <NA>           Afghanistan    2020-03-01     5      0
##  8 Afghanistan  <NA>           Afghanistan    2020-03-02     5      0
##  9 Afghanistan  <NA>           Afghanistan    2020-03-03     5      0
## 10 Afghanistan  <NA>           Afghanistan    2020-03-04     5      0
## # i 306,817 more rows
```

Here, we are using another dataset within the github to look up the meaning of ISO and FIPS meanings in order to make use of them. By using a join, we can cleanly join the UID to the Country and Province columns. This will make the data more readable and cross-comparable.

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

uid <- read_csv(UID_lookup) %>% 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)
global
```

```
## # A tibble: 306,827 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
##  4 <NA>           Afghanistan    2020-02-27     5      0   38928341 Afghanistan
##  5 <NA>           Afghanistan    2020-02-28     5      0   38928341 Afghanistan
##  6 <NA>           Afghanistan    2020-02-29     5      0   38928341 Afghanistan
##  7 <NA>           Afghanistan    2020-03-01     5      0   38928341 Afghanistan
##  8 <NA>           Afghanistan    2020-03-02     5      0   38928341 Afghanistan
##  9 <NA>           Afghanistan    2020-03-03     5      0   38928341 Afghanistan
## 10 <NA>           Afghanistan    2020-03-04     5      0   38928341 Afghanistan
## # i 306,817 more rows
```

Now we are grouping the US dataset down so that it goes state by state, which makes it easy to compare different sections of the US. We will also be adding a deaths per million statistic as well as a cases per million statistic. This will help compare across the states since population will vary. It also tracks how well a state might be handling the pandemic.

```
US_by_state <- 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.
```

```
US_by_state
```

```
## # A tibble: 66,294 x 7
##    Province_State Country_Region date       cases deaths deaths_per_mill
##    <chr>          <chr>          <date>     <dbl>  <dbl>           <dbl>
##  1 Alabama        US             2020-01-22     0      0               0
```

```
##  2 Alabama        US             2020-01-23     0      0                0
##  3 Alabama        US             2020-01-24     0      0                0
##  4 Alabama        US             2020-01-25     0      0                0
##  5 Alabama        US             2020-01-26     0      0                0
##  6 Alabama        US             2020-01-27     0      0                0
##  7 Alabama        US             2020-01-28     0      0                0
##  8 Alabama        US             2020-01-29     0      0                0
##  9 Alabama        US             2020-01-30     0      0                0
## 10 Alabama        US             2020-01-31     0      0                0
## # i 66,284 more rows
## # i 1 more variable: Population <dbl>
```

```
US_total <- 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.
```

```
US_total
```

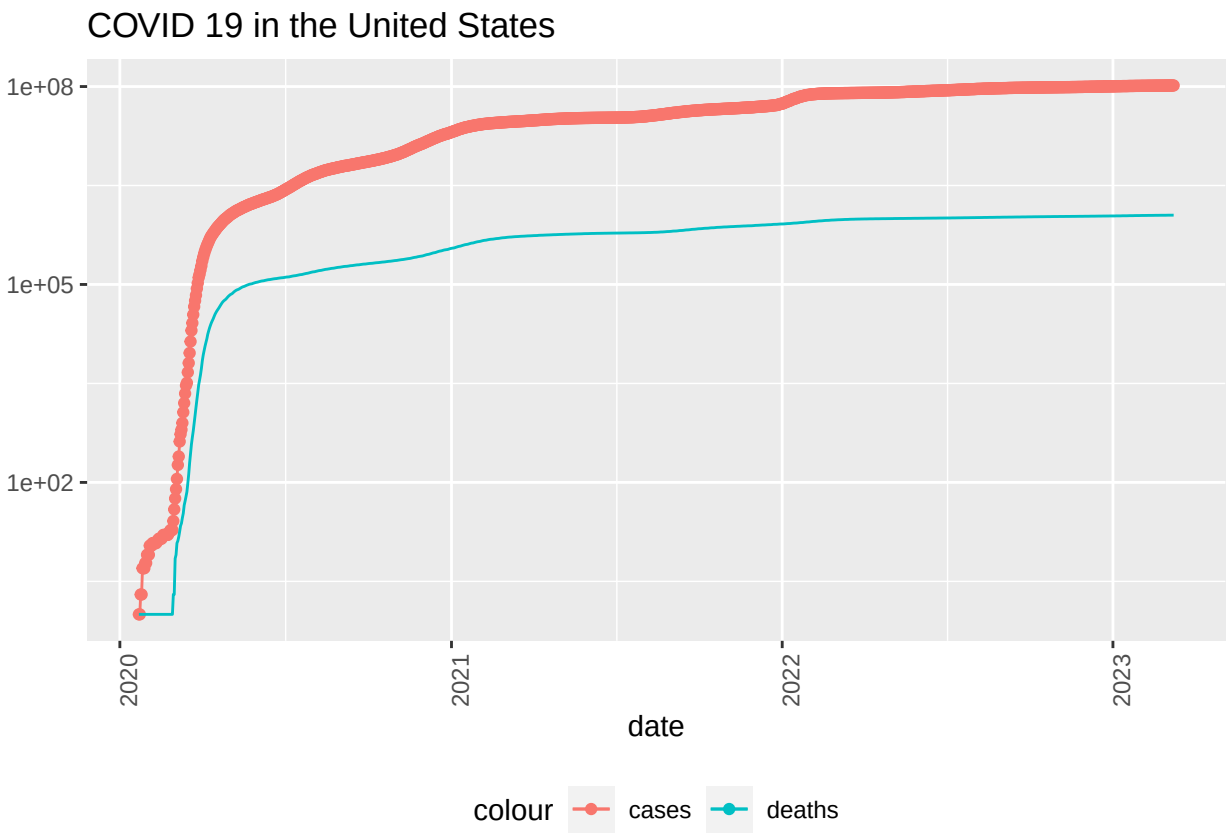
```
## # A tibble: 1,143 x 6
##    Country_Region date       cases deaths deaths_per_mill Population
##    <chr>          <date>     <dbl>  <dbl>           <dbl>      <dbl>
##  1 US             2020-01-22     1      1         0.00300  332875137
##  2 US             2020-01-23     1      1         0.00300  332875137
##  3 US             2020-01-24     2      1         0.00300  332875137
##  4 US             2020-01-25     2      1         0.00300  332875137
##  5 US             2020-01-26     5      1         0.00300  332875137
##  6 US             2020-01-27     5      1         0.00300  332875137
##  7 US             2020-01-28     5      1         0.00300  332875137
##  8 US             2020-01-29     6      1         0.00300  332875137
##  9 US             2020-01-30     6      1         0.00300  332875137
## 10 US             2020-01-31     8      1         0.00300  332875137
## # i 1,133 more rows
```

Visualizations

This plot will show the number of cases versus the number of deaths. The y axis will use a logarithmic scale in order to prevent the two subgraphs from being too far apart and creating a visual bias towards either side. There is a clear trend shown that COVID is being managed well after the original outbreak in early 2020.

```
US_total %>%
  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")) +
  scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "COVID 19 in the United States", y = NULL)
```

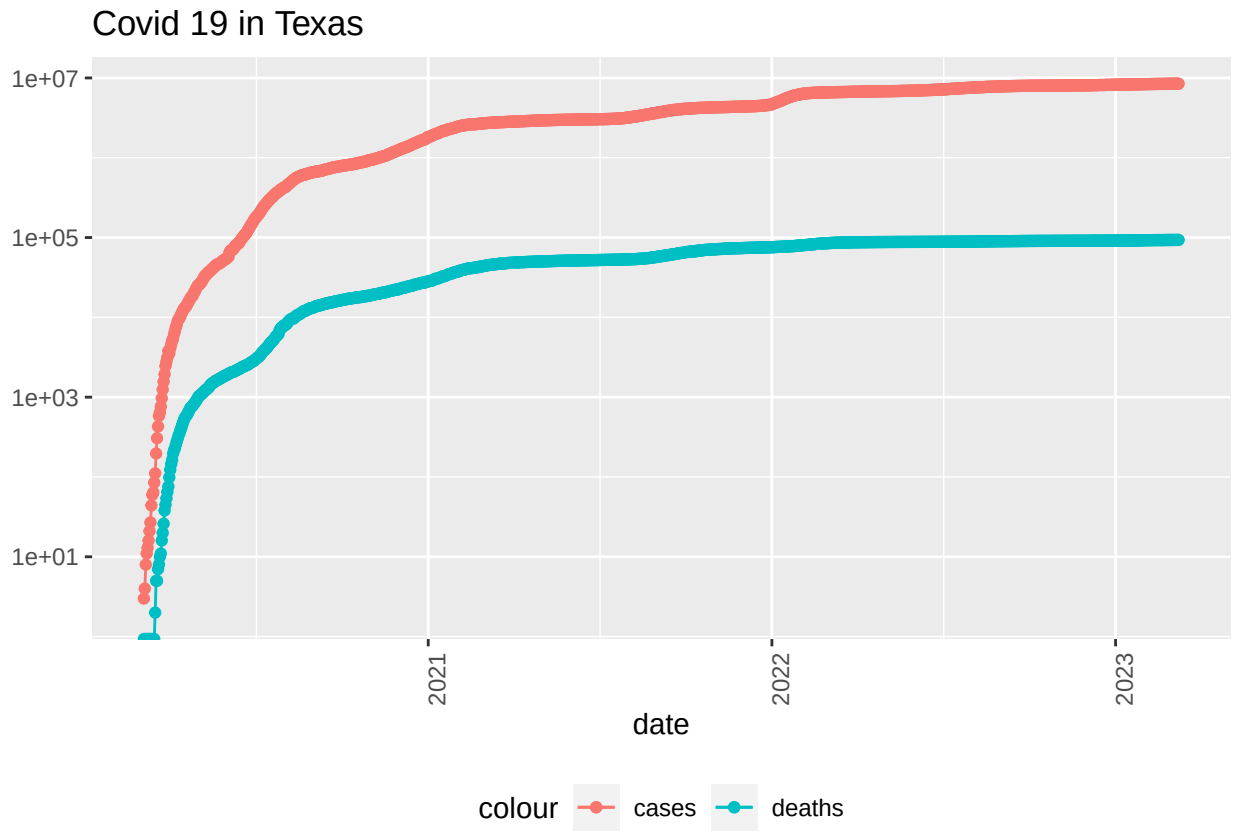


Let's take a look at Texas and compare it with California. These are the two largest states that heavily vote for Democrats (California) or Republicans (Texas) with no real sign of changing.

Texas shows that there were more cases and less consistent flattening of the deaths. This tracks as some people in Texas might not have believed in the vaccine and chose to listen to the incorrect Presidential recommendations of horse dewormer and excess UV light.

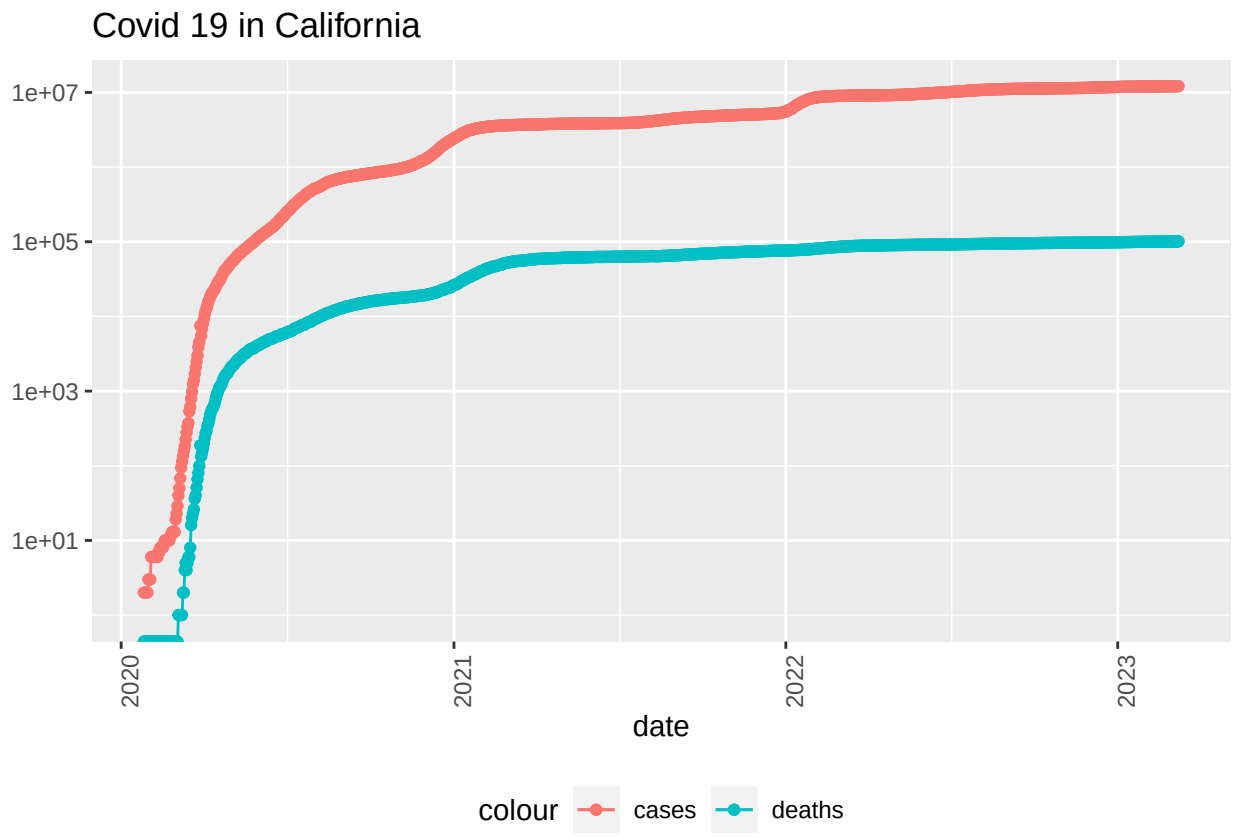
```
state <- "Texas"
US_by_state %>%
  filter(Province_State == state) %>%
  filter(cases > 0) %>%
  ggplot(aes(x = date, y = cases)) +
  geom_line(aes(color = "cases")) +
  geom_point(aes(color = "cases")) +
  geom_line(aes(y = deaths, color = "deaths")) +
  geom_point(aes(y = deaths, color = "deaths")) +
  scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = str_c("Covid 19 in ", state), y = NULL)
```

```
## Warning: Transformation introduced infinite values in continuous y-axis
## Transformation introduced infinite values in continuous y-axis
```




```
state <- "California"
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("Covid 19 in ", state), y = NULL)
```

```
## Warning: Transformation introduced infinite values in continuous y-axis
## Transformation introduced infinite values in continuous y-axis
```



Creating columns to track new deaths and new cases allows for more accurate tracking of how many cases are breaking out per day. Someone who contracted COVID yesterday reporting that they still have COVID is not surprising, however seeing a wave of new cases should be some amount of surprising.

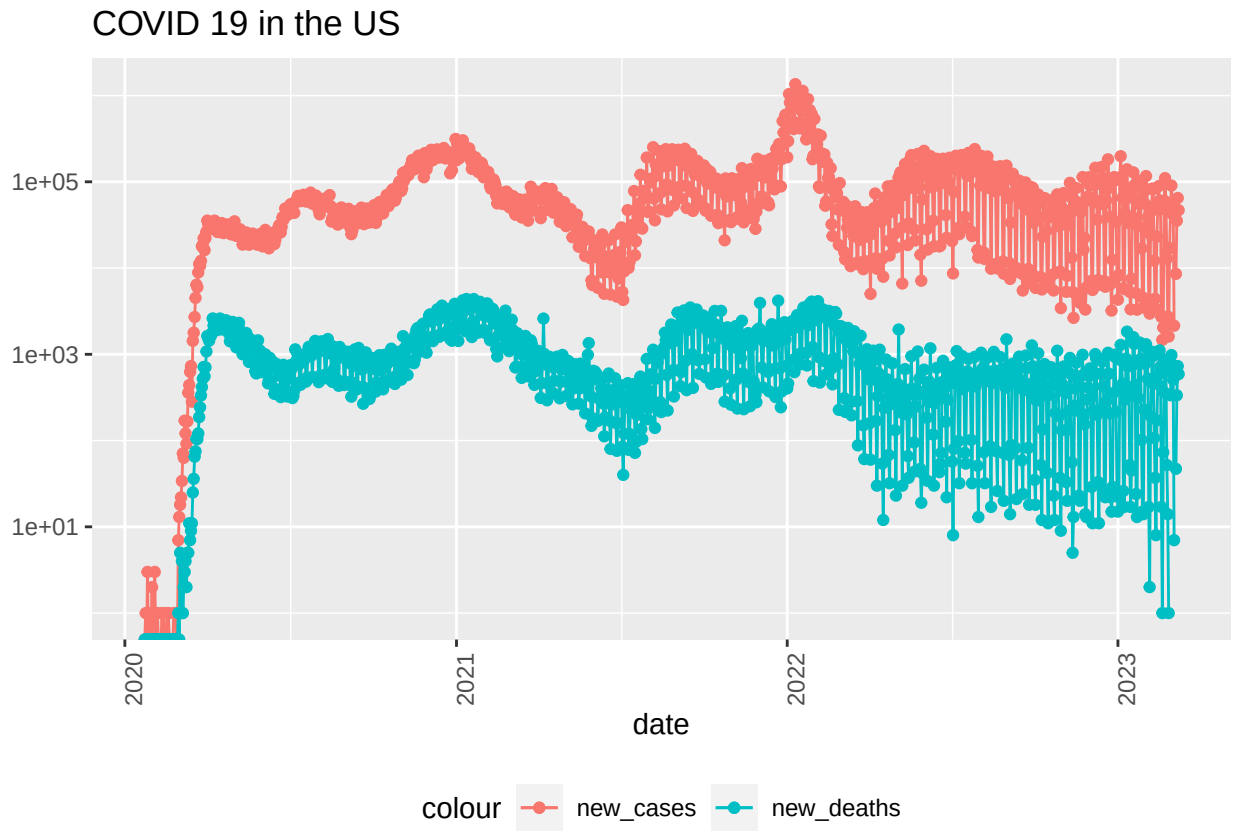
```
US_by_state <- US_by_state %>%
  mutate(new_cases = cases - lag(cases),
         new_deaths = deaths - lag(deaths))
US_total <- US_total %>%
  mutate(new_cases = cases - lag(cases),
         new_deaths = deaths - lag(deaths))

tail(US_total %>% 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           3371.
## 2     -3862        -38 US             2023-03-05   1.04e8 1.12e6           3371.
## 3      8564         47 US             2023-03-06   1.04e8 1.12e6           3371.
## 4     35371        335 US             2023-03-07   1.04e8 1.12e6           3372.
## 5     64861        730 US             2023-03-08   1.04e8 1.12e6           3374.
## 6     46931        590 US             2023-03-09   1.04e8 1.12e6           3376.
## # i 1 more variable: Population <dbl>
```

Tracking the overall count of new cases and new deaths over time in the US. While it might seem obvious that every death is "new" since you can only die once, it is still important to see the trend.

```
US_total %>%
  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 = "COVID 19 in the US", y = NULL)
```

```
## Warning in self$trans$transform(x): NaNs produced

## Warning: Transformation introduced infinite values in continuous y-axis

## Warning in self$trans$transform(x): NaNs produced

## Warning: Transformation introduced infinite values in continuous y-axis

## Warning in self$trans$transform(x): NaNs produced

## Warning: Transformation introduced infinite values in continuous y-axis

## Warning in self$trans$transform(x): NaNs produced

## Warning: Transformation introduced infinite values in continuous y-axis

## Warning: Removed 1 row containing missing values (`geom_line()`).

## Warning: Removed 2 rows containing missing values (`geom_point()`).

## Warning: Removed 1 row containing missing values (`geom_line()`).

## Warning: Removed 4 rows containing missing values (`geom_point()`).
```




```
US_state_totals <- US_by_state %>%
  group_by(Province_State) %>%
  summarize(deaths = max(deaths), cases = max(cases),
            population = max(Population),
            cases_per_1k = 1000 * cases / population,
            deaths_per_1k = 1000 * deaths / population) %>%
  filter(cases > 0, population > 0)

US_state_totals %>%
  slice_min(deaths_per_1k, n = 10) %>%
  select(deaths_per_1k, cases_per_1k, everything())
```

```
## # A tibble: 10 x 6
##    deaths_per_1k cases_per_1k Province_State          deaths   cases population
##            <dbl>        <dbl> <chr>                   <dbl>   <dbl>      <dbl>
##  1         0.611         150. American Samoa             34    8320      55641
##  2         0.744         248. Northern Mariana Islands   41   13666      55144
##  3         1.21          231. Virgin Islands            130   24813     107268
##  4         1.30          269. Hawaii                   1841  380608    1415872
##  5         1.49          245. Vermont                   929  152618     623989
##  6         1.55          293. Puerto Rico              5823 1101469    3754939
##  7         1.65          340. Utah                     5298 1090346    3205958
##  8         2.01          415. Alaska                   1486  307655     740995
##  9         2.03          252. District of Columbia     1432  177945     705749
## 10         2.06          253. Washington              15683 1928913    7614893
```

```
US_state_totals %>%
  slice_max(deaths_per_1k, n = 10) %>%
  select(deaths_per_1k, cases_per_1k, everything())
```

```
## # A tibble: 10 x 6
##    deaths_per_1k cases_per_1k Province_State deaths   cases population
##            <dbl>        <dbl> <chr>           <dbl>   <dbl>      <dbl>
##  1          4.55         336. Arizona         33102 2443514    7278717
##  2          4.54         326. Oklahoma        17972 1290929    3956971
##  3          4.49         333. Mississippi     13370  990756    2976149
##  4          4.44         359. West Virginia    7960  642760    1792147
##  5          4.32         320. New Mexico       9061  670929    2096829
##  6          4.31         334. Arkansas        13020 1006883    3017804
##  7          4.29         335. Alabama         21032 1644533    4903185
##  8          4.28         368. Tennessee       29263 2515130    6829174
##  9          4.23         307. Michigan        42205 3064125    9986857
## 10          4.06         385. Kentucky        18130 1718471    4467673
```

```
US_state_totals
```

```
## # A tibble: 56 x 6
##    Province_State       deaths    cases population cases_per_1k deaths_per_1k
##    <chr>                 <dbl>    <dbl>      <dbl>        <dbl>         <dbl>
##  1 Alabama               21032  1644533    4903185         335.         4.29
##  2 Alaska                 1486   307655     740995         415.         2.01
##  3 American Samoa           34     8320      55641         150.         0.611
##  4 Arizona               33102  2443514    7278717         336.         4.55
##  5 Arkansas              13020  1006883    3017804         334.         4.31
##  6 California           101159 12129699   39512223         307.         2.56
##  7 Colorado              14181  1764401    5758736         306.         2.46
##  8 Connecticut           12220   976657    3565287         274.         3.43
##  9 Delaware               3324   330793     973764         340.         3.41
## 10 District of Columbia   1432   177945     705749         252.         2.03
## # i 46 more rows
```

Model

A basic linear model that is meant to predict the deaths per 1000, dependent on the cases per 1000. This only predicts for the United States and United States territories. The r-squared value is 0.2933, which shows that there is some relationship between cases –> deaths, but not enough to declare it a clear and bound relationship. The p-value is under 0.05 which shows that it is statistically significant rather and that the cases per thousand impacts the deaths per thousand. On the grand scale, this is an ok model that is acceptable, but there is more that can be done to fine tune it.

```
mod <- lm(deaths_per_1k ~ cases_per_1k, data = US_state_totals)
summary(mod)
```

```
##
## Call:
## lm(formula = deaths_per_1k ~ cases_per_1k, 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_1k  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_state_totals %>% mutate(pred = predict(mod))
```

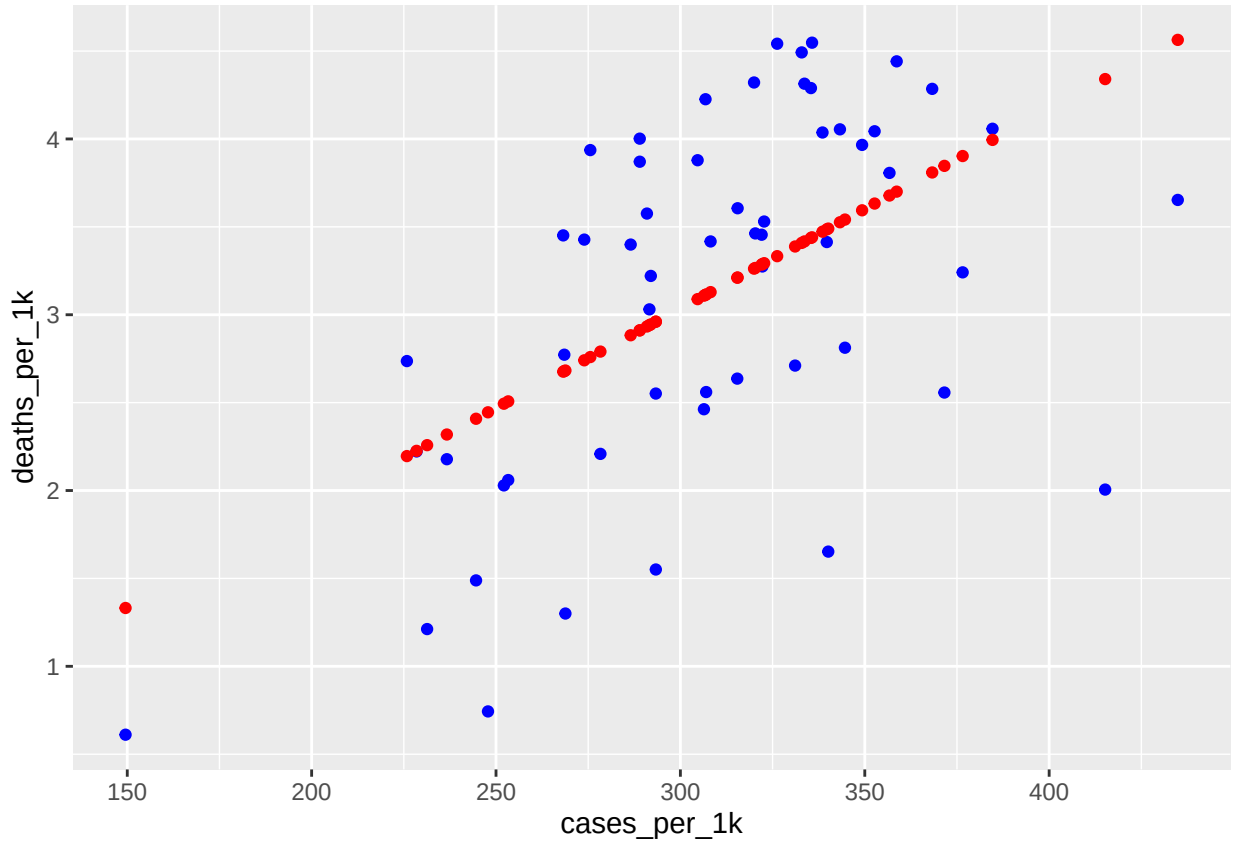
```
## # A tibble: 56 x 7
##    Province_State      deaths  cases population cases_per_1k deaths_per_1k  pred
##    <chr>                <dbl>  <dbl>      <dbl>        <dbl>         <dbl> <dbl>
##  1 Alabama              21032 1.64e6    4903185         335.          4.29   3.44
##  2 Alaska                1486 3.08e5     740995         415.          2.01   4.34
##  3 American Samoa          34 8.32e3      55641         150.          0.611  1.33
##  4 Arizona              33102 2.44e6    7278717         336.          4.55   3.44
##  5 Arkansas             13020 1.01e6    3017804         334.          4.31   3.42
##  6 California          101159 1.21e7   39512223         307.          2.56   3.12
##  7 Colorado             14181 1.76e6    5758736         306.          2.46   3.11
##  8 Connecticut          12220 9.77e5    3565287         274.          3.43   2.74
##  9 Delaware              3324 3.31e5     973764         340.          3.41   3.49
## 10 District of Columb~   1432 1.78e5     705749         252.          2.03   2.49
## # i 46 more rows
```

```
US_tot_with_pred <- US_state_totals %>% mutate(pred = predict(mod))
US_tot_with_pred
```

```
## # A tibble: 56 x 7
##    Province_State      deaths  cases population cases_per_1k deaths_per_1k  pred
##    <chr>                <dbl>  <dbl>      <dbl>        <dbl>         <dbl> <dbl>
##  1 Alabama              21032 1.64e6    4903185         335.          4.29   3.44
##  2 Alaska                1486 3.08e5     740995         415.          2.01   4.34
##  3 American Samoa          34 8.32e3      55641         150.          0.611  1.33
##  4 Arizona              33102 2.44e6    7278717         336.          4.55   3.44
##  5 Arkansas             13020 1.01e6    3017804         334.          4.31   3.42
##  6 California          101159 1.21e7   39512223         307.          2.56   3.12
##  7 Colorado             14181 1.76e6    5758736         306.          2.46   3.11
##  8 Connecticut          12220 9.77e5    3565287         274.          3.43   2.74
##  9 Delaware              3324 3.31e5     973764         340.          3.41   3.49
## 10 District of Columb~   1432 1.78e5     705749         252.          2.03   2.49
## # i 46 more rows
```

The graph below shows the prediction against the known data. The prediction graph is in blue and is much more scattered than the red. The red represents the actual data pulled from the US_tot_with_pred dataset.

```
US_tot_with_pred %>% ggplot() +
  geom_point(aes(x = cases_per_1k, y = deaths_per_1k), color = "blue") +
  geom_point(aes(x = cases_per_1k, y = pred), color = "red")
```



Biases

In this project, I focused mostly on the United States as that is both where I grew up and live. I made a relatively loose assumption that developing countries would struggle more than developed nations. I also had political biases in regards to the US data: I expected to see higher deaths in Republican voting states and lower deaths in Democrat leaning states. This stems from how the government handled the pandemic versus the president going out of his way to spread misinformation and promote an anti-mask sentiment despite masks being a step in the right direction.