Covid Data Analysis

Mitch Lewis

2023-07-19

Covid Data Analysis

This report will analyze data about cases and deaths of the Covid 19 virus, reported both in the US and globaly. The data set used was provided by Johns Hopkins and is available on github. First we need to read in the four data sets.

Cleaning Data

```r
library(tidyverse)
```

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

```r
library(lubridate)
cov19_confirmed_us <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_c
```

```
## 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
head(cov19_confirmed_us)
```

```
## # 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>, ...
```

```
cov19_deaths_us <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covi
```

```
## 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.
```

```
head(cov19_deaths_us)
```

```
## # 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>, ...
```

```
cov19_confirmed_global <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/css
```

```
## 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.
```

```
head(cov19_confirmed_global)
```

```
## # 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>, ...
```

```
cov19_deaths_global <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_
```

```
## 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.
```

```
head(cov19_deaths_global)
```

```
## # 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>, ...
```

We now need to do some cleaning of the data. We will start by doing some minor reformatting and dropping the Lat and Long columns that won't be used.

```
cov19_confirmed_global <- cov19_confirmed_global %>%
  pivot_longer(cols = -c('Province/State', 'Country/Region', Lat, Long), names_to = 'date', values_to =
  select(-c(Lat,Long))



cov19_deaths_global <- cov19_deaths_global %>%
   pivot_longer(cols = -c('Province/State', 'Country/Region', Lat, Long), names_to = 'date', values_to
  select(-c(Lat,Long))
```

Next we can combine tables to make a simpler object to work with. We will combine the global cases and global deaths data into a single global variable. We will then filter out where cases are 0.

```r
global <- cov19_confirmed_global %>%
  full_join(cov19_deaths_global) %>%
  rename(Country_Region = 'Country/Region', Province_State = 'Province/State') %>%
  mutate(date=mdy(date))
```

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

```r
  global <- global %>% filter(cases>0)
```

We also need to clean up some of the US data.

```r
cov19_confirmed_us <- cov19_confirmed_us %>%
  pivot_longer(cols = -(UID:Combined_Key), names_to = "date", values_to = 'cases') %>%
  select(Admin2:cases) %>%
  mutate(date = mdy(date)) %>%
  select(-c(Lat, Long_))

cov19_deaths_us <- cov19_deaths_us %>%
  pivot_longer(cols = -(UID:Population), names_to = 'date', values_to = 'deaths') %>%
  select(Admin2:deaths) %>%
  mutate(date = mdy(date)) %>%
  select(-c(Lat, Long_))
```

And now we can combine US cases and deaths into a single US variable

```r
US <- cov19_confirmed_us %>%
  full_join(cov19_deaths_us)
```

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

Missing from the global data is population data. We can read in additional country population data and join it with our global data set.

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

uid <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/UID
```

```
## 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.
```

```r
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.x)
```

Global Analysis

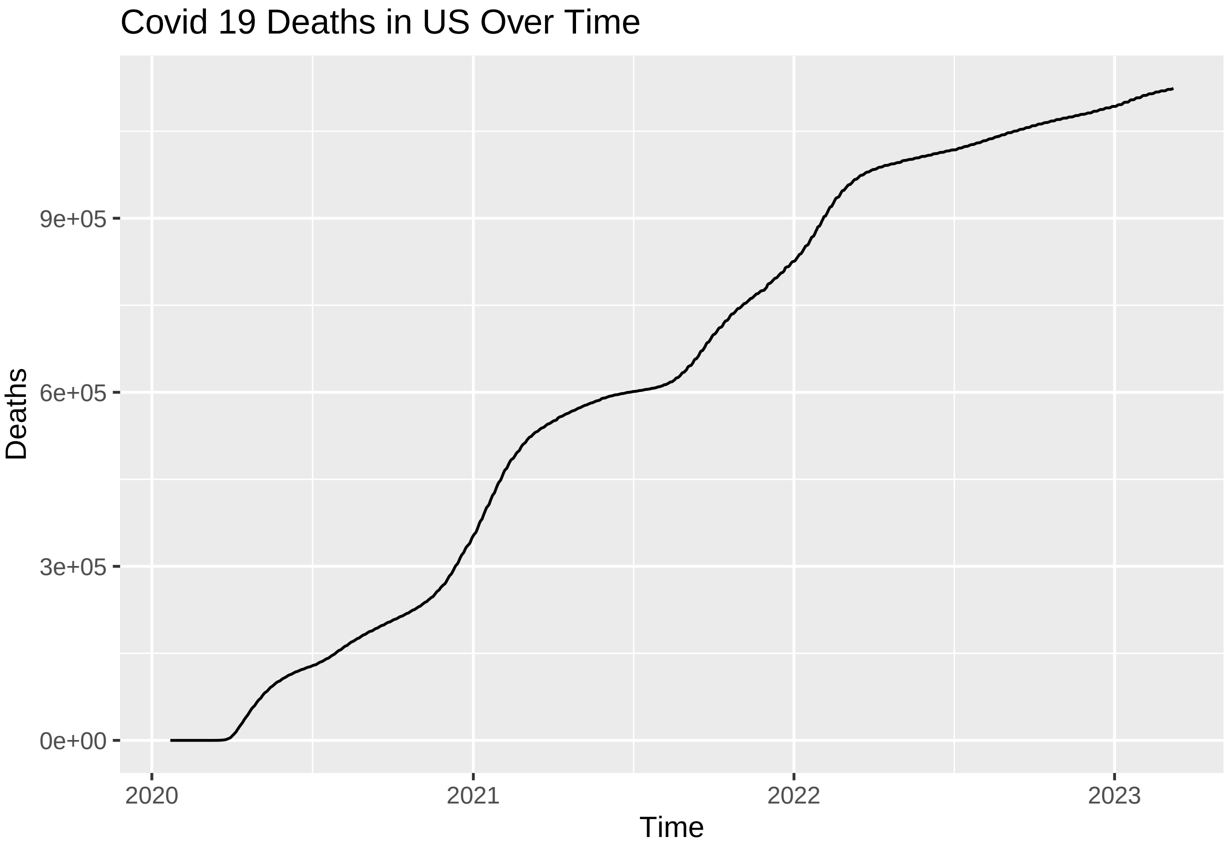
Now that the data has been cleaned up we can analyze the data. One thing that we can look at is to see the most deaths by country.

```
most_deaths <- global %>%
  group_by(Country_Region) %>%
  summarize(max_deaths = max(deaths, na.rm = TRUE)) %>%
  arrange(desc(max_deaths)) %>%
  head(5)

most_deaths
```

```
## # A tibble: 5 x 2
##   Country_Region max_deaths
##   <chr>               <dbl>
## 1 US                1123836
## 2 Brazil             699276
## 3 India              530779
## 4 Russia             388478
## 5 Mexico             333188
```

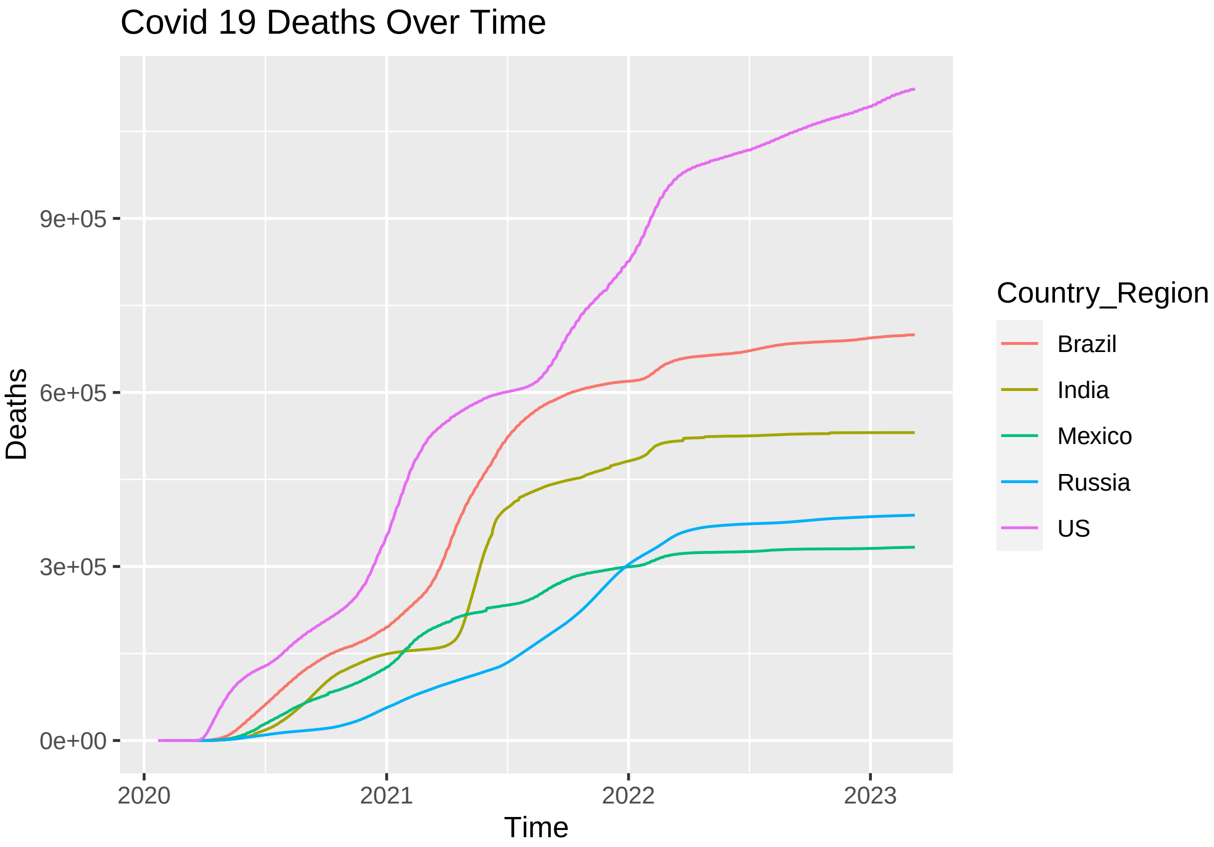
Since the US had the most deaths, we can visualize this over time.

```
US_deaths <- global %>%
  filter(Country_Region == 'US') %>%
  arrange(date)

US_deaths_plot <- ggplot(US_deaths, aes(x = date, y = deaths))+
  geom_line() +
   labs(title = 'Covid 19 Deaths in US Over Time', x = 'Time', y = 'Deaths')

US_deaths_plot
```



We can also plot it alongside the other top 5 countries as a comparison.

```
Combined_deaths <- global %>%
  filter(Country_Region == 'US'| Country_Region == 'Brazil' | Country_Region == 'India' | Country_Regio
  arrange(date)


Combined_deaths_plot <- ggplot(Combined_deaths, aes(x=date, y=deaths, group = Country_Region))+
  geom_line(aes(color = Country_Region))+
   labs(title = 'Covid 19 Deaths Over Time', x = 'Time', y = 'Deaths')

Combined_deaths_plot
```



We can also look at which countries had the least deaths on any given day. Since there are some regions (like Antarctica), that will have 0, or some countries that had 0 deaths in a day, we will filter those out.

```
least_deaths <- global %>%
  group_by(Country_Region) %>%
  summarize(max_deaths = max(deaths, na.rm = TRUE)) %>%
  filter(max_deaths >0) %>%
  arrange(desc(max_deaths)) %>%
  tail(5)

least_deaths
```

```
## # A tibble: 5 x 2
##   Country_Region max_deaths
##   <chr>               <dbl>
## 1 Tonga                  13
## 2 Palau                  9
## 3 Korea, North           6
## 4 MS Zaandam             2
## 5 Nauru                  1
```

State Analysis

Just as we viewed the countries with the most covid deaths, we can view the top 5 US states by covid deaths.

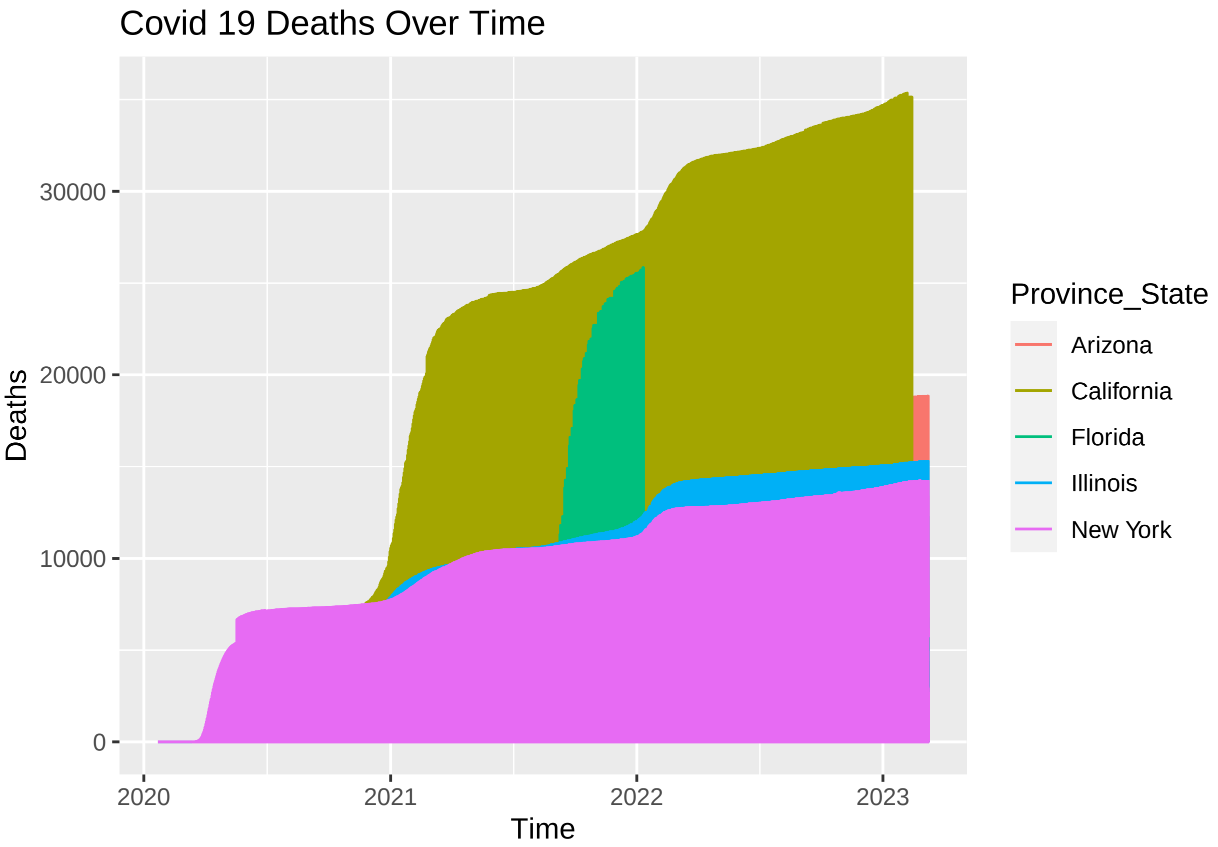
```
most_deaths_US <- US %>%
  group_by(Province_State) %>%
  summarize(max_deaths = max(deaths, na.rm = TRUE)) %>%
  arrange(desc(max_deaths)) %>%
  head(5)

most_deaths_US
```

```
## # A tibble: 5 x 2
##   Province_State max_deaths
##   <chr>               <dbl>
## 1 California          35545
## 2 Florida             25840
## 3 Arizona             18846
## 4 Illinois            15289
## 5 New York            14219
```

```
least_deaths_US <- US %>%
  group_by(Province_State) %>%
  summarize(max_deaths = max(deaths, na.rm = TRUE)) %>%
  filter(max_deaths >0) %>%
  arrange(desc(max_deaths)) %>%
  tail(5)

least_deaths_US
```

```
## # A tibble: 5 x 2
##   Province_State           max_deaths
##   <chr>                         <dbl>
## 1 Vermont                         230
## 2 Virgin Islands                  130
## 3 Northern Mariana Islands         41
## 4 American Samoa                   34
## 5 Grand Princess                    3
```

```
State_deaths <- US %>%
  filter(Province_State == 'California'| Province_State == 'Florida' | Province_State == 'Arizona' | Pro
  arrange(date)


State_deaths_plot <- ggplot(State_deaths, aes(x=date, y=deaths, group = Province_State))+
  geom_line(aes(color = Province_State))+
   labs(title = 'Covid 19 Deaths Over Time', x = 'Time', y = 'Deaths')

State_deaths_plot
```



The states with the most deaths in the US appear to be states with high populations. This intuitively makes sense, as with a higher population there can be higher death totals. Population density can also effect the ability for viruses to be transfered. We can take a look at the correlation between covid deaths and population by state.

```
US_deaths_population <- US %>%
  group_by(Province_State, Population) %>%
  summarize(max_deaths = max(deaths, na.rm = TRUE)) %>%
  arrange(desc(max_deaths))
```

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

```
cor.test(US_deaths_population$Population, US_deaths_population$max_deaths, method = "pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  US_deaths_population$Population and US_deaths_population$max_deaths
## t = 102.28, df = 3277, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.8642037 0.8805483
## sample estimates:
##       cor
## 0.8726201
```

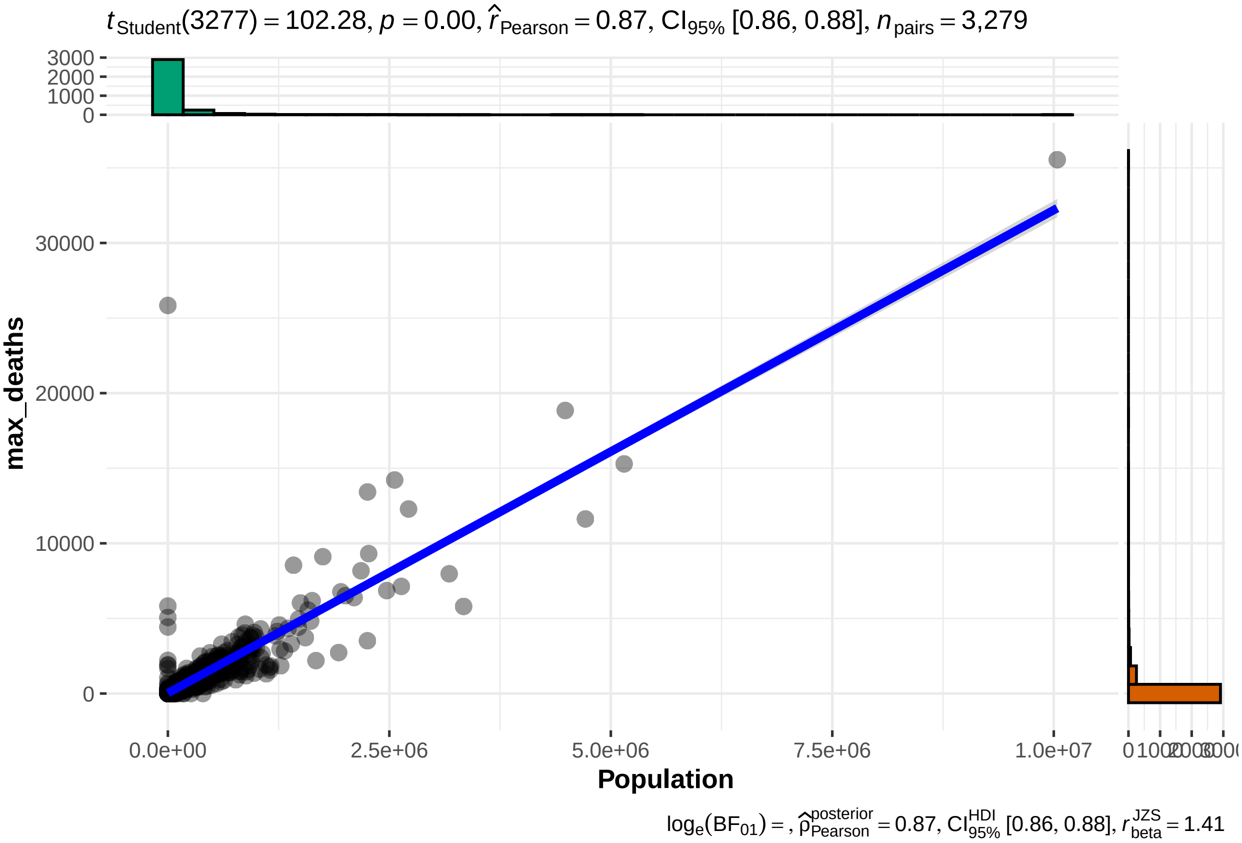
```
library(ggstatsplot)
```

```
## You can cite this package as:
##      Patil, I. (2021). Visualizations with statistical details: The 'ggstatsplot' approach.
##      Journal of Open Source Software, 6(61), 3167, doi:10.21105/joss.03167
```

```
ggscatterstats(data = US_deaths_population, x = Population, y = max_deaths)
```

```
## Registered S3 method overwritten by 'ggside':
##   method from
##   +.gg   ggplot2
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



This shows that there does seem to be a strong correlation between population of states and their covid deaths.

Conclusions and Bias

This report looked at covid deaths both by country and by states in the US. Unsurprisingly, it was found that both the countrys and states with the highest covid deaths were areas with lsarge populations. This generally makes sense, however there are other factors that should also be considered. Reporting of deaths may have varied, especially globally where different policies for reporting were put in place, and some countries may not have reported at all. There are also serveral other factors that could have impacted covid deaths beyond population. Such factors include varying healthcare systems, variation in regulations such as lockdown and mask policies, access to vaccines, and many others.