

COVID_19

A.Pavlenko

2022-06-06

Coronavirus Disease 2019 (COVID-19)

Coronavirus disease (COVID-19) is an infectious disease caused by the SARS-CoV-2 virus.

COVID-19 affects different people in different ways. Infected people have had a wide range of symptoms reported – from mild symptoms to severe illness.

```
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.1 --

## v ggplot2 3.3.6     v purrr    0.3.4
## v tibble  3.1.7     v dplyr    1.0.9
## v tidyr   1.2.0     v stringr  1.4.0
## v readr   2.1.2     vforcats  0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()   masks stats::lag()

## Download data in 4 files:
url_in <- "C:/Users/pavle/Desktop/COVID-19/"
file_names <- c("time_series_covid19_confirmed_global.csv",
              "time_series_covid19_deaths_global.csv",
              "time_series_covid19_confirmed_US.csv",
              "time_series_covid19_deaths_US.csv")
urls <- str_c(url_in, file_names)
urls

## [1] "C:/Users/pavle/Desktop/COVID-19/time_series_covid19_confirmed_global.csv"
## [2] "C:/Users/pavle/Desktop/COVID-19/time_series_covid19_deaths_global.csv"
## [3] "C:/Users/pavle/Desktop/COVID-19/time_series_covid19_confirmed_US.csv"
## [4] "C:/Users/pavle/Desktop/COVID-19/time_series_covid19_deaths_US.csv"
```

Reading Data:

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

## Rows: 285 Columns: 871
## -- Column specification -----
## Delimiter: ","
## chr (2): Province/State, Country/Region
## dbl (869): 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[2])
```

```
## Rows: 285 Columns: 871
## -- Column specification -----
## Delimiter: ","
## chr (2): Province/State, Country/Region
## dbl (869): 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.
```

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

```
## Rows: 3342 Columns: 878
## -- Column specification -----
## Delimiter: ","
## chr (6): iso2, iso3, Admin2, Province_State, Country_Region, Combined_Key
## dbl (872): 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.
```

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

```
## Rows: 3342 Columns: 879
## -- Column specification -----
## Delimiter: ","
## chr (6): iso2, iso3, Admin2, Province_State, Country_Region, Combined_Key
## dbl (873): UID, code3, FIPS, Lat, Long_, Population, 1/22/20, 1/23/20, 1/24/20, ...
##
## 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.
```

Cleanup 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))
global_cases
```

```

## # A tibble: 247,095 x 4
##   'Province/State' 'Country/Region' date     cases
##   <chr>           <chr>        <chr>    <dbl>
## 1 <NA>             Afghanistan  1/22/20    0
## 2 <NA>             Afghanistan  1/23/20    0
## 3 <NA>             Afghanistan  1/24/20    0
## 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
## # ... with 247,085 more rows

```

```

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_deaths

```

```

## # A tibble: 247,095 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
## # ... with 247,085 more rows

```

```
library(lubridate)
```

```

##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
## 
##   date, intersect, setdiff, union

```

```

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

```

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

```
global
```

```
## # A tibble: 247,095 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
## # ... with 247,085 more rows
```

```
summary(global)
```

```
##  Province_State    Country_Region      date      cases
##  Length:247095    Length:247095    Min.   :2020-01-22  Min.   :    0
##  Class :character  Class :character  1st Qu.:2020-08-25  1st Qu.: 299
##  Mode  :character  Mode  :character  Median :2021-03-30  Median : 6912
##                                         Mean   :2021-03-30  Mean   : 588646
##                                         3rd Qu.:2021-11-02  3rd Qu.: 129326
##                                         Max.   :2022-06-06  Max.   :84882287
## 
##   deaths
##   Min.   :    0
##   1st Qu.:    2
##   Median :   86
##   Mean   : 10452
##   3rd Qu.: 1967
##   Max.   :1008857
```

```
global %>% filter(cases > 0)
```

```
global %>% filter(cases > 28000000)
```

```
## # A tibble: 990 x 5
##   Province_State Country_Region date      cases  deaths
##   <chr>          <chr>        <date>     <dbl>   <dbl>
## 1 <NA>           Brazil       2022-02-18 28072238 643340
## 2 <NA>           Brazil       2022-02-19 28177367 644195
## 3 <NA>           Brazil       2022-02-20 28218180 644592
## 4 <NA>           Brazil       2022-02-21 28258458 644918
## 5 <NA>           Brazil       2022-02-22 28361951 645735
## 6 <NA>           Brazil       2022-02-23 28493336 646714
## 7 <NA>           Brazil       2022-02-24 28589235 647703
## 8 <NA>           Brazil       2022-02-25 28679671 648496
## 9 <NA>           Brazil       2022-02-26 28749552 649184
```

```
## 10 <NA>          Brazil      2022-02-27 28776794 649437
## # ... with 980 more rows
```

```
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: 2,897,514 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
## # ... with 2,897,504 more rows
```

```
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: 2,897,514 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
## # ... with 2,897,504 more rows, and 1 more variable: deaths <dbl>
```

```

US <- US_cases %>%
  full_join(US_deaths)

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

US

## # A tibble: 2,897,514 x 8
##   Admin2 Province_State Country_Region Combined_Key date      cases Population
##   <chr>   <chr>        <chr>        <chr>      <date>    <dbl>     <dbl>
## 1 Autauga~ Alabama       US          Autauga, Al~ 2020-01-22    0     55869
## 2 Autauga~ Alabama       US          Autauga, Al~ 2020-01-23    0     55869
## 3 Autauga~ Alabama       US          Autauga, Al~ 2020-01-24    0     55869
## 4 Autauga~ Alabama       US          Autauga, Al~ 2020-01-25    0     55869
## 5 Autauga~ Alabama       US          Autauga, Al~ 2020-01-26    0     55869
## 6 Autauga~ Alabama       US          Autauga, Al~ 2020-01-27    0     55869
## 7 Autauga~ Alabama       US          Autauga, Al~ 2020-01-28    0     55869
## 8 Autauga~ Alabama       US          Autauga, Al~ 2020-01-29    0     55869
## 9 Autauga~ Alabama       US          Autauga, Al~ 2020-01-30    0     55869
## 10 Autauga~ Alabama      US          Autauga, Al~ 2020-01-31   0     55869
## # ... with 2,897,504 more rows, and 1 more variable: deaths <dbl>

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

## # A tibble: 227,505 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
## # ... with 227,495 more rows

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

uid <- read_csv(uid_lookup_url) %>%
  select(-c(Lat, Long_, Combined_Key, code3, iso2, iso3, Admin2))

## Rows: 4317 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.

url_in <- "https://github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data/csse_covid_19_time...
uid <- read_csv(uid_lookup_url) %>%
  select(-c(Lat, Long_, Combined_Key, code3, iso2, iso3, Admin2))

## Rows: 4317 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, Population,
         Combined_Key)
global

## # A tibble: 227,505 x 6
##   Province_State Country_Region date      cases Population Combined_Key
##   <chr>          <chr>       <date>     <dbl>      <dbl> <chr>
## 1 <NA>           Afghanistan 2020-02-24     5  38928341 Afghanistan
## 2 <NA>           Afghanistan 2020-02-25     5  38928341 Afghanistan
## 3 <NA>           Afghanistan 2020-02-26     5  38928341 Afghanistan
## 4 <NA>           Afghanistan 2020-02-27     5  38928341 Afghanistan
## 5 <NA>           Afghanistan 2020-02-28     5  38928341 Afghanistan
## 6 <NA>           Afghanistan 2020-02-29     5  38928341 Afghanistan
## 7 <NA>           Afghanistan 2020-03-01     5  38928341 Afghanistan
## 8 <NA>           Afghanistan 2020-03-02     5  38928341 Afghanistan
## 9 <NA>           Afghanistan 2020-03-03     5  38928341 Afghanistan
## 10 <NA>          Afghanistan 2020-03-04     5  38928341 Afghanistan
## # ... with 227,495 more rows

US_by_state <- US %>%
  group_by(Province_State, Country_Region, date) %>%
  summarise(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: 50,286 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
## # ... with 50,276 more rows, and 1 more variable: Population <dbl>
```

```
US_totals <- US_by_state %>%
  group_by(Country_Region, date) %>%
  summarise(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_totals
```

```
## # A tibble: 867 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
## # ... with 857 more rows
```

```
tail(US_totals)
```

```

## # A tibble: 6 x 6
##   Country_Region date      cases  deaths deaths_per_mill Population
##   <chr>        <date>    <dbl>   <dbl>       <dbl>     <dbl>
## 1 US          2022-06-01 84451901 1007714      3027. 332875137
## 2 US          2022-06-02 84570325 1008031      3028. 332875137
## 3 US          2022-06-03 84724329 1008422      3029. 332875137
## 4 US          2022-06-04 84748884 1008567      3030. 332875137
## 5 US          2022-06-05 84762022 1008585      3030. 332875137
## 6 US          2022-06-06 84882287 1008857      3031. 332875137

```

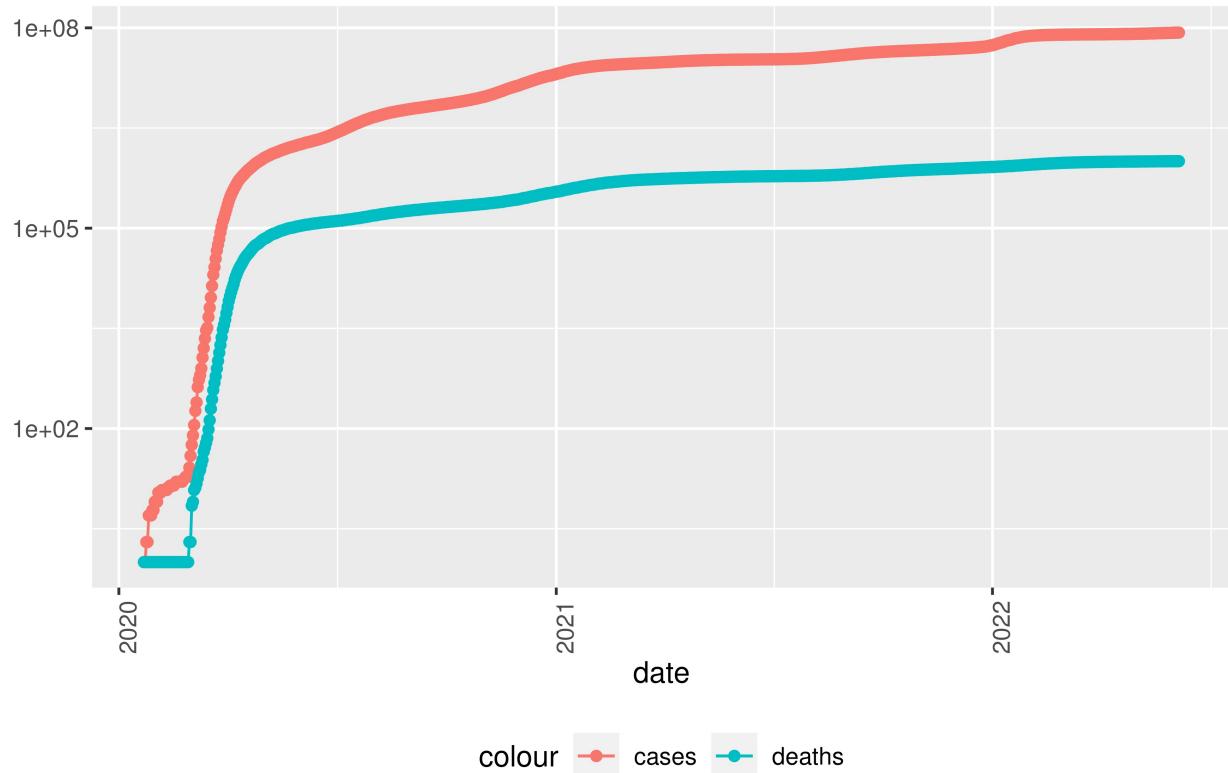
```
## Include Plots:
```

```

###COVID19 in US
US_totals %>%
  filter(cases > 0) %>%
  ggplot(aes(x = date, y = cases)) +
  geom_line(aes(color = "cases")) +
  geom_point(aes(color = "cases")) +
  geom_line(aes(y = deaths, color = "deaths")) +
  geom_point(aes(y = deaths, color = "deaths")) +
  scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = "COVID19 in US", y = NULL)

```

COVID19 in US

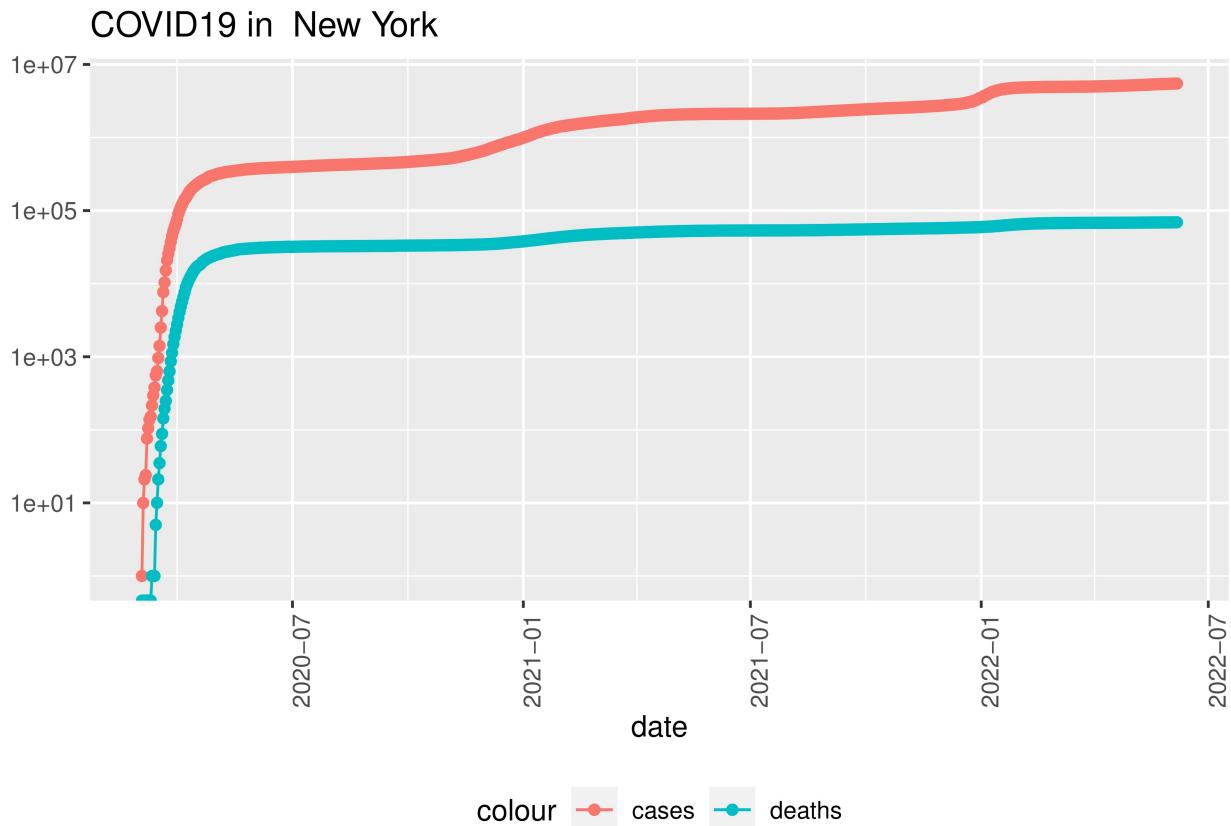


```

###COVID19 in New-York
state <- "New York"
US_by_state %>%
  filter(Province_State == state) %>%
  filter(cases > 0) %>%
  ggplot(aes(x = date, y = cases)) +
  geom_line(aes(color = "cases")) +
  geom_point(aes(color = "cases")) +
  geom_line(aes(y = deaths, color = "deaths")) +
  geom_point(aes(y = deaths, color = "deaths")) +
  scale_y_log10() +
  theme(legend.position = "bottom",
        axis.text.x = element_text(angle = 90)) +
  labs(title = str_c("COVID19 in ", state), y = NULL)

```

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



```

### Analysing Data: max(US_totals$date)
max(US_totals$deaths)

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))

tail(US_totals)

```

```

tail(US_totals %>% select(new_cases, new_deaths, everything()))

US_totals %>% ggplot(aes(x = date, y = new_cases)) + geom_line(aes(color = "new_cases")) +
  geom_point(aes(color = "new_cases")) + geom_line(aes(y = new_deaths, color = "new_deaths")) +
  geom_point(aes(y = new_deaths, color = "new_deaths")) + scale_y_log10() + theme(legend.position =
  "bottom", axis.text.x = element_text(angle = 90)) + labs(title = "COVID19 in US", y = NULL)

state <- "New York" US_by_state %>% filter(Province_State == state) %>% 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 = str_c("COVID19 in", state), y = NULL)

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

US_state_totals %>% slice_min(deaths_per_thou, n = 10)

US_state_totals %>% slice_min(deaths_per_thou, n = 10) %>% select(deaths_per_thou, cases_per_thou,
everything())

US_state_totals %>% slice_max(deaths_per_thou, n = 10) %>% select(deaths_per_thou, cases_per_thou,
everything())

```

Modeling Data:

```

mod <- lm(deaths_per_thou ~ cases_per_thou, data = US_state_totals) summary(mod)

US_state_totals %>% slice_min(cases_per_thou) US_state_totals %>% slice_max(cases_per_thou)

x_grid <- seq(1, 151) new_df <- tibble(cases_per_thou = x_grid) US_state_totals %>% mutate(pred =
predict(mod))

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

US_tot_w_pred %>% ggplot() + geom_point(aes(x = cases_per_thou, y = deaths_per_thou), color =
"blue") + geom_point(aes(x = cases_per_thou, y = pred), color = "red")

```