report.Rmd

Valentina Anthonio

2023-06-19

COVID19 Data Analysis

This report presents an analysis of COVID19 data, specifically focusing on the number of cases and deaths since the outbreak. The data used for this analysis was sourced from the GitHub account of John Hopkins, spanning from 2020 to 2023. Five types of datasets were collected, including global cases, global deaths, US cases, US deaths, and population data which are;

```
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 errors
```

```
library(lubridate)
```

Import files

```
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_confirmed_global.csv", "t
urls <-str_c(url_in,file_names)

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

```
## Rows: 3342 Columns: 1154
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (6): iso2, iso3, Admin2, Province_State, Country_Region, Combined_Key
## dbl (1148): UID, code3, FIPS, Lat, Long_, 1/22/20, 1/23/20, 1/24/20, 1/25/20...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

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

```
## Rows: 289 Columns: 1147
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (2): Province/State, Country/Region
## dbl (1145): Lat, Long, 1/22/20, 1/23/20, 1/24/20, 1/25/20, 1/26/20, 1/27/20,...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

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

```
## Rows: 3342 Columns: 1155
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (6): iso2, iso3, Admin2, Province_State, Country_Region, Combined_Key
## dbl (1149): UID, code3, FIPS, Lat, Long_, Population, 1/22/20, 1/23/20, 1/24...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

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

```
## Rows: 289 Columns: 1147
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (2): Province/State, Country/Region
## dbl (1145): Lat, Long, 1/22/20, 1/23/20, 1/24/20, 1/25/20, 1/26/20, 1/27/20,...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Data cleaning

```
#data cleaning and dropping some columns
global_cases <- global_cases %>% pivot_longer(cols = -c('Province/State','Country/Region','Lat','Long')

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

```
#merge the two global data
global <- global_cases %>% full_join(global_deaths) %>% rename(Country_Region = 'Country/Region', Provi
```

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

```
# filter for cases greater than zero
global <- global %>% filter(cases >0)
```

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

```
# perform data cleaning for the US data
us_cases <- us_cases %>% pivot_longer(cols = -(UID:Combined_Key), names_to = "date",values_to = "cases"

us_deaths <- us_deaths %>% pivot_longer(cols = -(UID:Population), names_to = "date",values_to = "deaths

#merge the US datasets
US <- us_cases %>% full_join(us_deaths)
```

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

```
#get population data from same site
uid_lookup_url <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/U

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

```
## Rows: 4321 Columns: 12
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (7): iso2, iso3, FIPS, Admin2, Province_State, Country_Region, Combined_Key
## dbl (5): UID, code3, Lat, Long_, Population
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

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

Data Analysis

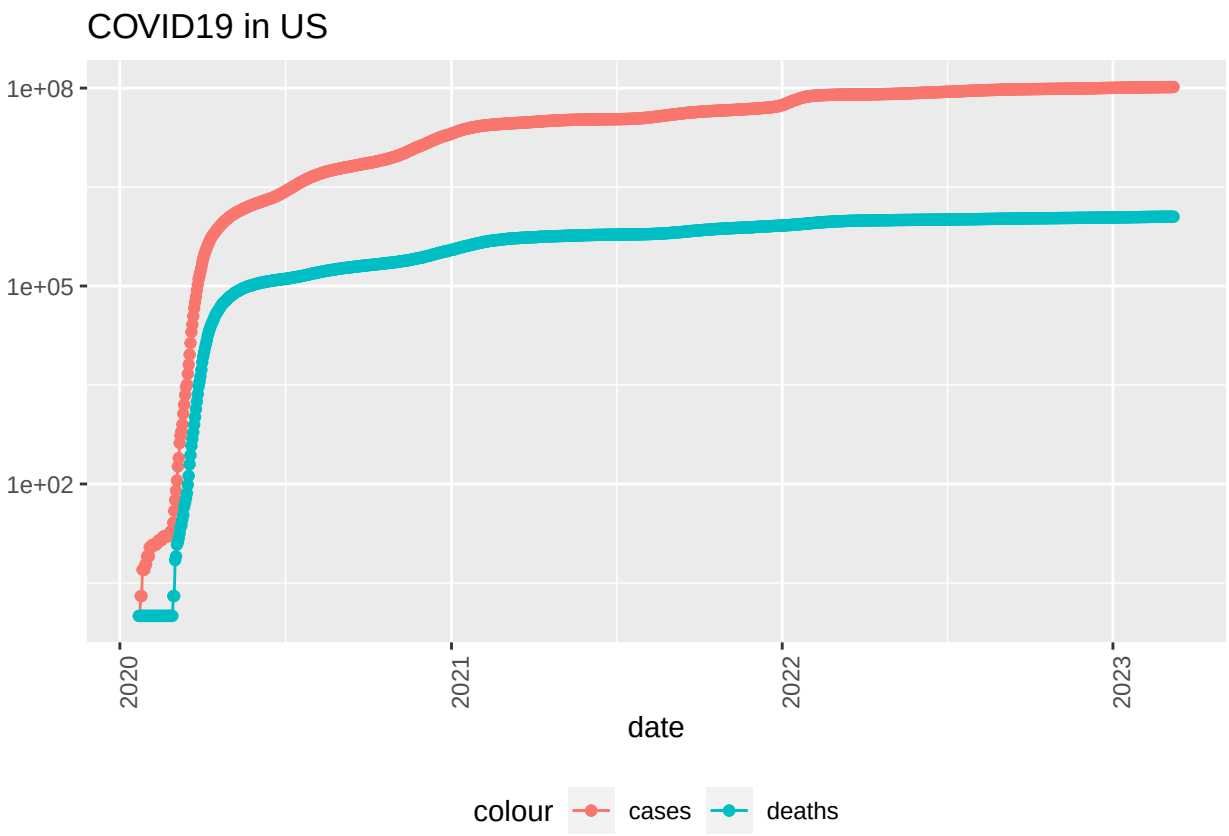
```
US_by_state <- US %>% group_by(Province_State,Country_Region,date) %>%
  summarize(cases=sum(cases), deaths=sum(deaths), Population= sum(Population)) %>%
  mutate(deaths_per_mill = deaths*1000000/Population) %>%
  select(Province_State,Country_Region,date,cases,deaths,deaths_per_mill,Population) %>% ungroup()
```

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

```
US_totals <- US_by_state %>%
  group_by(Country_Region,date) %>% summarize(cases=sum(cases),deaths=sum(deaths),Population = sum(Popul
```

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

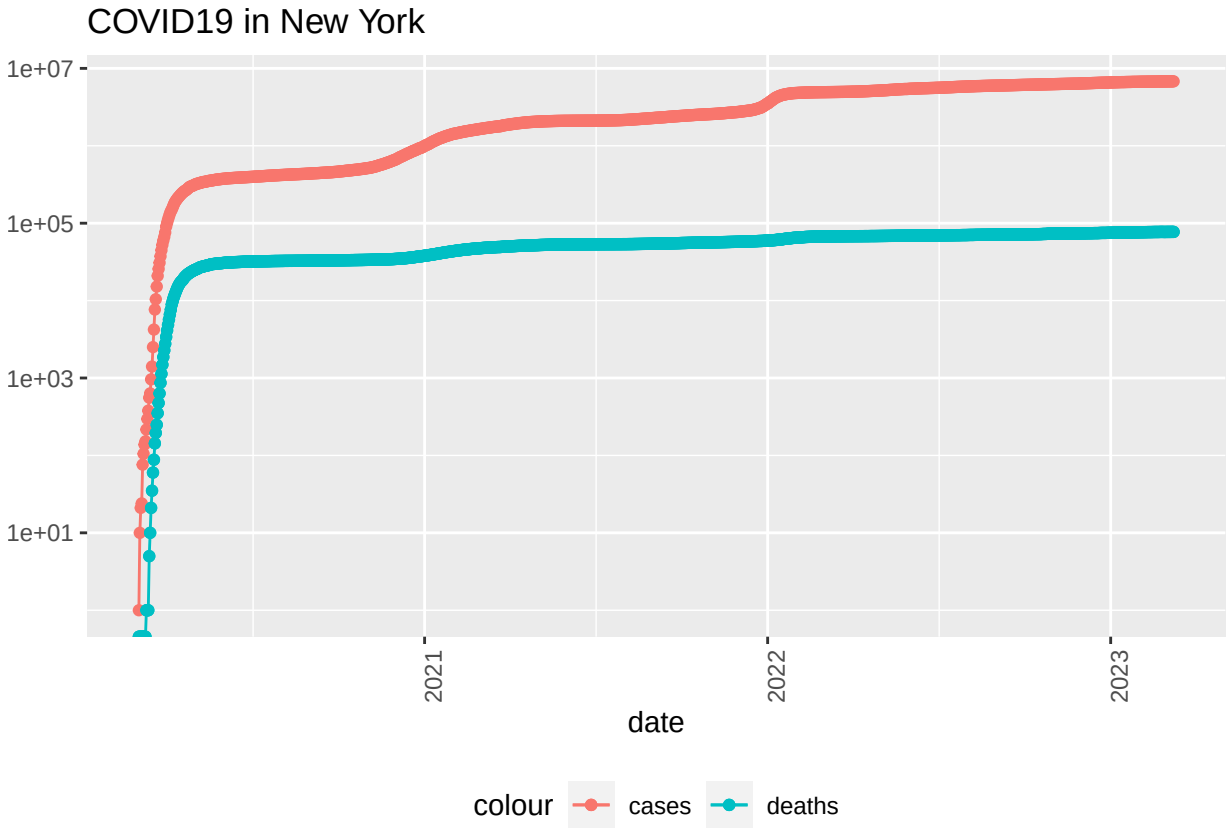
```
US_totals %>% filter(cases >0) %>%
  ggplot(aes(x=date,y=cases)) + geom_line(aes(color = "cases")) + geom_point(aes(color = "cases")) + ge
  theme(legend.position="bottom",axis.text.x = element_text(angle=90)) +
  labs(title="COVID19 in US", y= NULL)
```




```
#Filter for case 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")) +
  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
## Transformation introduced infinite values in continuous y-axis
```

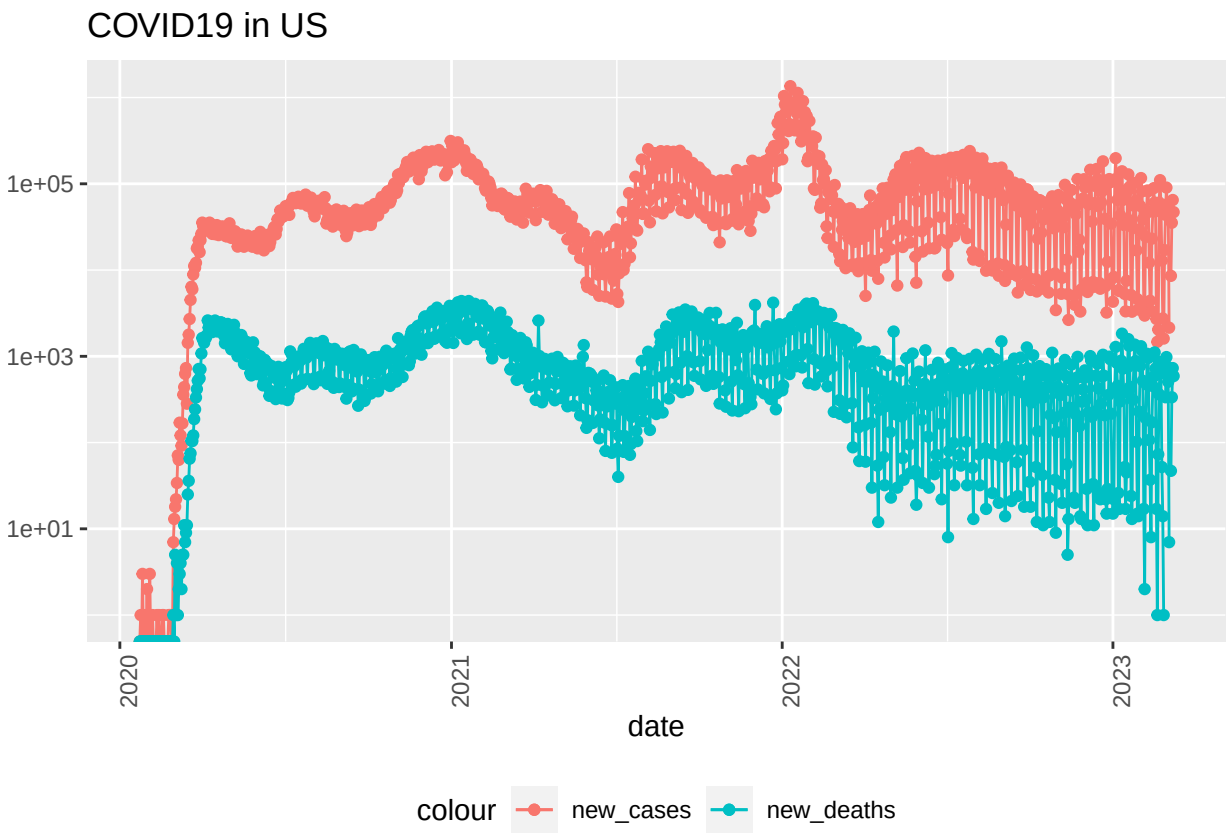



```
US_by_state <- US_by_state %>%
  mutate(new_cases = cases - lag(cases),
         new_deaths = deaths-lag(deaths))

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

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

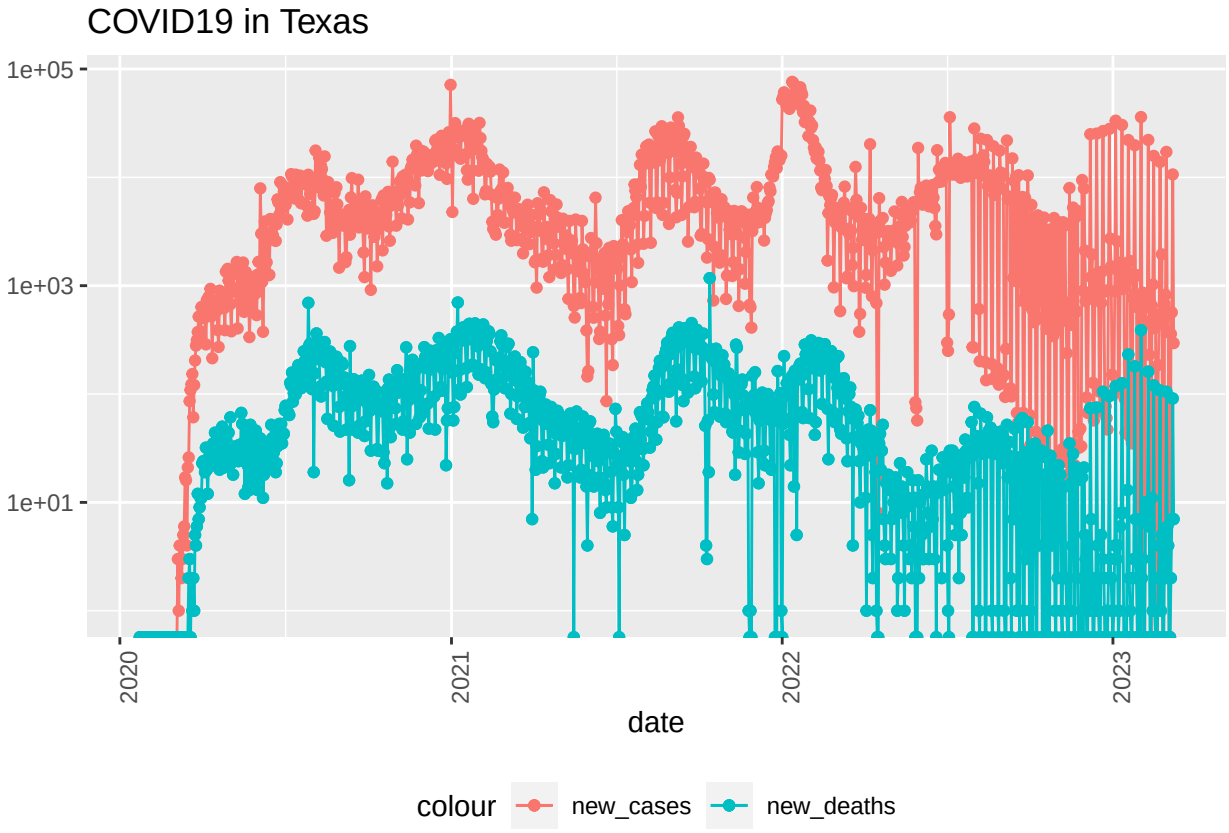
```
## Warning in self$trans$transform(x): NaNs produced

## 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()`).
```




```
state <- "Texas"

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")) +
  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 )
```

```
## 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 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()`).
## 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_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)
```

```
## # A tibble: 10 x 6
##    Province_State        deaths  cases population cases_per_thou deaths_per_thou
##    <chr>                  <dbl>  <dbl>      <dbl>          <dbl>           <dbl>
##  1 American Samoa            34 8.32e3      55641           150.           0.611
##  2 Northern Mariana Isl~     41 1.37e4      55144           248.           0.744
##  3 Virgin Islands           130 2.48e4     107268           231.           1.21
##  4 Hawaii                  1841 3.81e5    1415872           269.           1.30
##  5 Vermont                  929 1.53e5     623989           245.           1.49
##  6 Puerto Rico             5823 1.10e6    3754939           293.           1.55
##  7 Utah                    5298 1.09e6    3205958           340.           1.65
##  8 Alaska                  1486 3.08e5     740995           415.           2.01
##  9 District of Columbia    1432 1.78e5     705749           252.           2.03
## 10 Washington             15683 1.93e6    7614893           253.           2.06
```

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

```
## # A tibble: 10 x 6
##    deaths_per_thou cases_per_thou Province_State        deaths  cases population
##              <dbl>          <dbl> <chr>                 <dbl>  <dbl>      <dbl>
##  1           0.611           150. American Samoa           34 8.32e3      55641
##  2           0.744           248. Northern Mariana Isl~    41 1.37e4      55144
##  3           1.21            231. Virgin Islands          130 2.48e4     107268
##  4           1.30            269. Hawaii                 1841 3.81e5    1415872
##  5           1.49            245. Vermont                 929 1.53e5     623989
##  6           1.55            293. Puerto Rico            5823 1.10e6    3754939
##  7           1.65            340. Utah                   5298 1.09e6    3205958
##  8           2.01            415. Alaska                 1486 3.08e5     740995
##  9           2.03            252. District of Columbia   1432 1.78e5     705749
## 10           2.06            253. Washington            15683 1.93e6    7614893
```

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

```
## # A tibble: 10 x 6
##    deaths_per_thou cases_per_thou 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
```

Cases in France and it's territories

```
france <- global %>%
  filter(Country_Region =="France")
france
```

```
## # A tibble: 12,929 x 7
##    Province_State Country_Region date       cases deaths Population Combined_Key
##    <chr>          <chr>          <date>     <dbl>  <dbl>      <dbl> <chr>
##  1 French Guiana  France         2020-03-07     5      0     298682 French Guia~
##  2 French Guiana  France         2020-03-08     5      0     298682 French Guia~
##  3 French Guiana  France         2020-03-09     5      0     298682 French Guia~
##  4 French Guiana  France         2020-03-10     5      0     298682 French Guia~
##  5 French Guiana  France         2020-03-11     5      0     298682 French Guia~
##  6 French Guiana  France         2020-03-12     5      0     298682 French Guia~
##  7 French Guiana  France         2020-03-13     5      0     298682 French Guia~
##  8 French Guiana  France         2020-03-14     5      0     298682 French Guia~
##  9 French Guiana  France         2020-03-15     7      0     298682 French Guia~
## 10 French Guiana  France         2020-03-16    11      0     298682 French Guia~
## # i 12,919 more rows
```

```
summary(france)
```

```
##  Province_State     Country_Region          date                cases
##  Length:12929       Length:12929       Min.   :2020-01-24   Min.   :       1
##  Class :character   Class :character   1st Qu.:2020-12-21   1st Qu.:     542
##  Mode  :character   Mode  :character   Median :2021-09-17   Median :   10762
##                                        Mean   :2021-09-14   Mean   : 1245720
##                                        3rd Qu.:2022-06-13   3rd Qu.:   72898
##                                        Max.   :2023-03-09   Max.   :38618509
##      deaths          Population       Combined_Key
##  Min.   :     0   Min.   :    5795   Length:12929
##  1st Qu.:     6   1st Qu.:   38659   Class :character
##  Median :    63   Median :  285491   Mode  :character
##  Mean   :  8772   Mean   : 6001501
##  3rd Qu.:   413   3rd Qu.:  400127
##  Max.   :161512   Max.   :65249843
```

```
france_by_state <- france %>% 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.
```

```
france_by_state
```

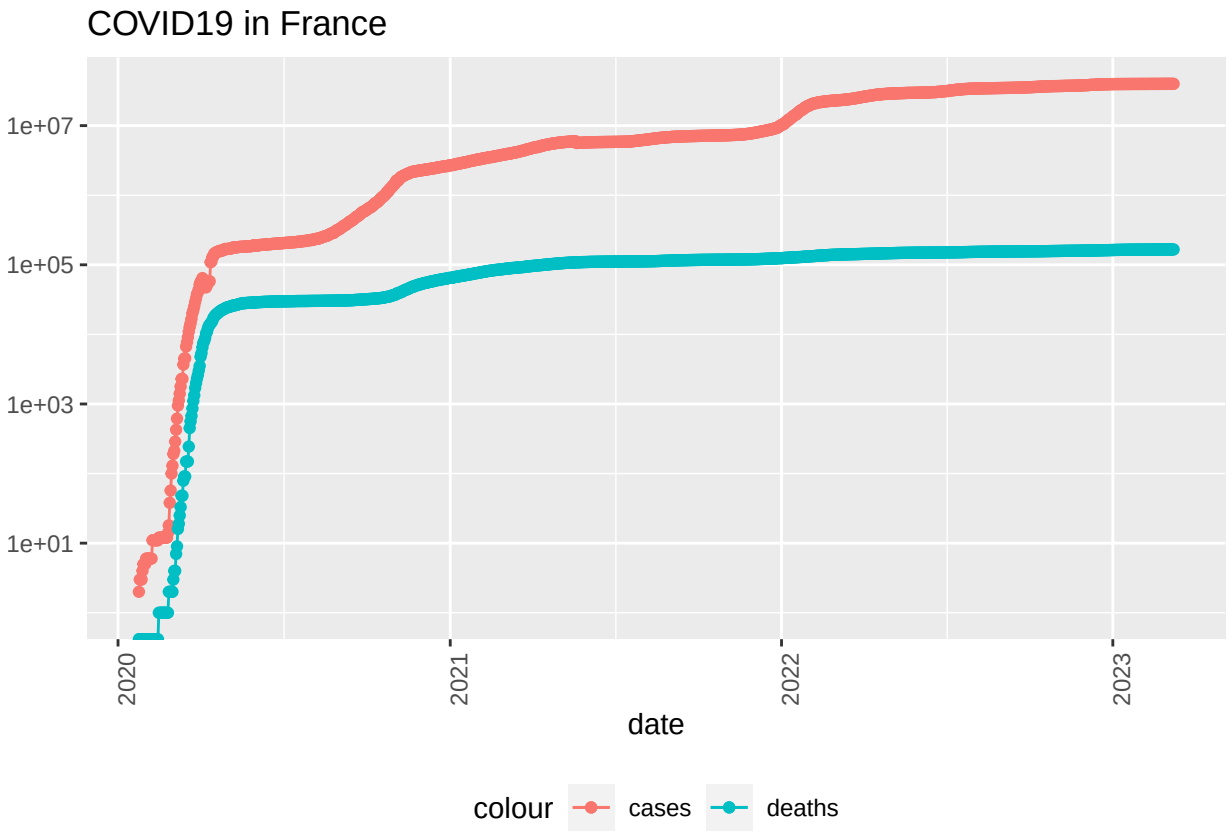
```
## # A tibble: 12,929 x 7
##    Province_State Country_Region date       cases deaths deaths_per_mill
##    <chr>          <chr>          <date>     <dbl>  <dbl>           <dbl>
##  1 French Guiana  France         2020-03-07     5      0               0
##  2 French Guiana  France         2020-03-08     5      0               0
##  3 French Guiana  France         2020-03-09     5      0               0
##  4 French Guiana  France         2020-03-10     5      0               0
##  5 French Guiana  France         2020-03-11     5      0               0
##  6 French Guiana  France         2020-03-12     5      0               0
##  7 French Guiana  France         2020-03-13     5      0               0
##  8 French Guiana  France         2020-03-14     5      0               0
##  9 French Guiana  France         2020-03-15     7      0               0
## 10 French Guiana  France         2020-03-16    11      0               0
## # i 12,919 more rows
## # i 1 more variable: Population <dbl>
```

```
france_totals <- france_by_state %>% group_by(Country_Region,date) %>% summarize(cases=sum(cases),deaths
```

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

```
france_totals %>% filter(cases >0) %>%
  ggplot(aes(x=date,y=cases)) + geom_line(aes(color = "cases")) + geom_point(aes(color = "cases")) + ge
  theme(legend.position="bottom",axis.text.x = element_text(angle=90)) +
  labs(title="COVID19 in France", y= NULL)
```

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



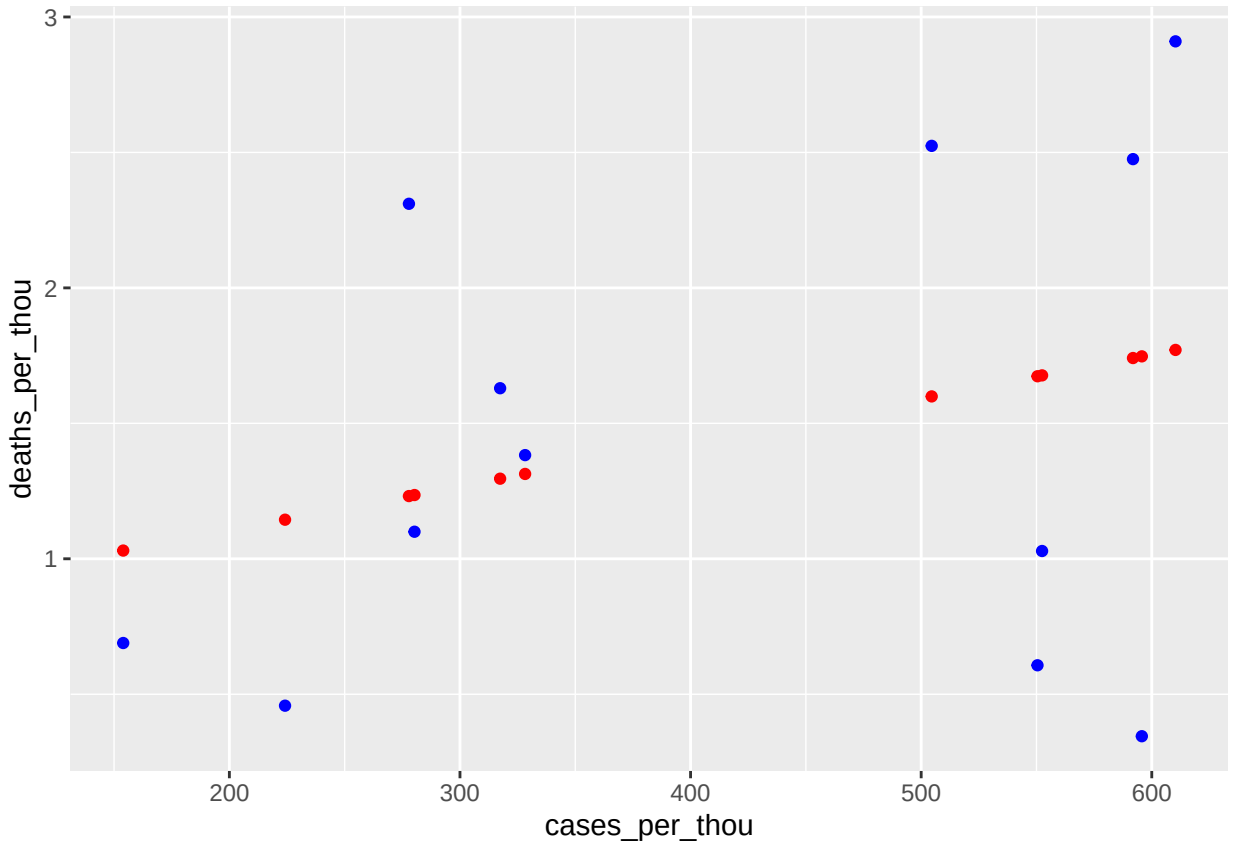

```
france_state_totals<- france_by_state %>%
  group_by(Province_State) %>%
  summarize(deaths = max(deaths),cases = max(cases), population= max(Population),
            cases_per_thou = 1000*cases/population,
            deaths_per_thou=1000*deaths/population) %>%
  filter(cases >0,population >0)
mod_fran <- lm(deaths_per_thou ~cases_per_thou,data=france_state_totals)
france_state_totals %>% mutate(pred=predict(mod_fran))
```

```
## # A tibble: 12 x 7
##    Province_State  deaths  cases population cases_per_thou deaths_per_thou  pred
##    <chr>            <dbl>  <dbl>      <dbl>          <dbl>           <dbl> <dbl>
##  1 French Guiana      413 9.80e4     298682           328.            1.38   1.31
##  2 French Polynes~    649 7.81e4     280904           278.            2.31   1.23
##  3 Guadeloupe        1010 2.02e5     400127           505.            2.52   1.60
##  4 Martinique        1092 2.29e5     375265           610.            2.91   1.77
##  5 Mayotte            188 4.20e4     272813           154.            0.689  1.03
##  6 New Caledonia      314 8.00e4     285491           280.            1.10   1.24
##  7 Reunion            921 4.95e5     895308           552.            1.03   1.68
##  8 Saint Barthele~      6 5.44e3       9885           550.            0.607  1.67
##  9 Saint Pierre a~      2 3.45e3       5795           596.            0.345  1.75
## 10 St Martin           63 1.23e4      38659           317.            1.63   1.30
## 11 Wallis and Fut~      7 3.43e3      15289           224.            0.458  1.14
## 12 <NA>            161512 3.86e7   65249843           592.            2.48   1.74
```

```
france_state_totals <- france_state_totals %>% replace_na(list(Province_State ="France"))
france_state_totals
```

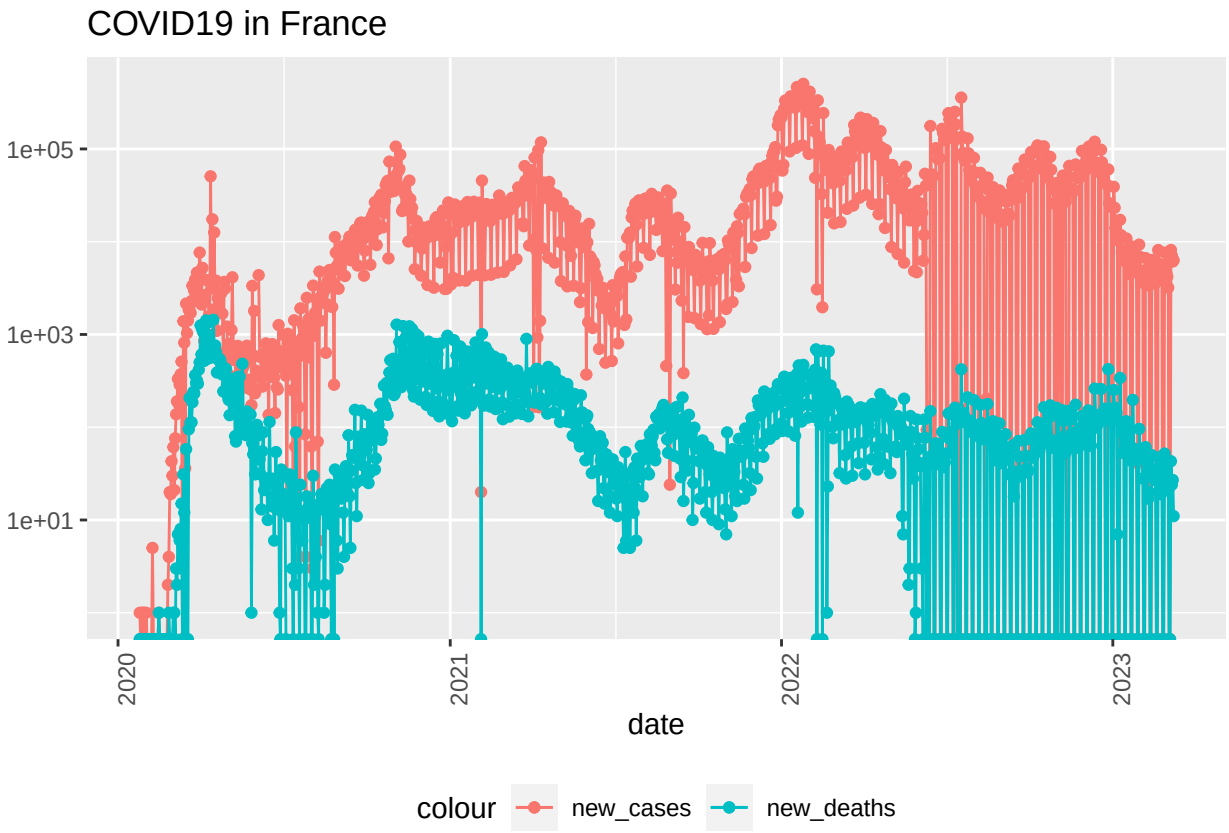
```
## # A tibble: 12 x 6
##    Province_State        deaths  cases population cases_per_thou deaths_per_thou
##    <chr>                  <dbl>  <dbl>      <dbl>          <dbl>           <dbl>
##  1 French Guiana            413 9.80e4     298682           328.           1.38
##  2 French Polynesia         649 7.81e4     280904           278.           2.31
##  3 Guadeloupe              1010 2.02e5     400127           505.           2.52
##  4 Martinique              1092 2.29e5     375265           610.           2.91
##  5 Mayotte                  188 4.20e4     272813           154.           0.689
##  6 New Caledonia            314 8.00e4     285491           280.           1.10
##  7 Reunion                  921 4.95e5     895308           552.           1.03
##  8 Saint Barthelemy           6 5.44e3       9885           550.           0.607
##  9 Saint Pierre and Miq~      2 3.45e3       5795           596.           0.345
## 10 St Martin                 63 1.23e4      38659           317.           1.63
## 11 Wallis and Futuna          7 3.43e3      15289           224.           0.458
## 12 France                161512 3.86e7   65249843           592.           2.48
```

```
france_tot_w_pred <- france_state_totals %>% mutate(pred=predict(mod_fran))
france_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")
```



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

france_totals %>%
  ggplot(aes(x=date,y=new_cases)) + geom_line(aes(color = "new_cases")) + geom_point(aes(color = "new_ca
  theme(legend.position="bottom",axis.text.x = element_text(angle=90)) +
  labs(title="COVID19 in France", 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 13 rows containing missing values (`geom_point()`).

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

## Warning: Removed 9 rows containing missing values (`geom_point()`).
```



The analysis reveals that the number of new cases and deaths in France and its territories are still increasing.

Conclusion

Based on the analysis of COVID19 data for France and its overseas territories, it is evident that there is a rise in the number of new cases and deaths. This indicates an ongoing spread and impact of the virus in the region. It is crucial to continue monitoring and implementing necessary measures to control and mitigate the further transmission of COVID19. It is important to note that this analysis is based on the available data from the John Hopkins University GitHub account. Any biases in the data collection or reporting methods could potentially affect the accuracy and reliability of the findings. Additionally, the analysis focuses specifically on France and may not capture the complete global picture of the COVID19 situation.In mitigating the my biases on the data, the analysis and report was void of any personal assumptions and interpretation of reasons for the increased number of cases and deaths.