

COVID19 Report

Konica Patait

2023-01-21

COVID19 Analysis

This report is based on COVID-19 data that is publicly available on Johns Hopkins GitHubsite. Link: <https://github.com/CSSEGISandData/COVID-19>

The agenda of this report is to analyze on below:

- Shows the cases and deaths due to COVID19 in all the countries since 2019.
- Show the number of cases and deaths due to COVID since 2019.
- Compare the fatality ratio in US verses all the countries in the world.

Case Fatality ratio is calculated as follows

$$\left(\frac{\text{Number of Cases reported in which patient died}}{\text{Number of Cases Reported}} \right) * 100$$

Load Data

Below set of lines load the COVID19 data available at Johns Hopkins github site. It is the data about the confirmed cases and deaths are available in 5 separate csv file.

To load the data, I am saving the data to my local store from the John Hopkins gitHub repository. If the file is not available in local store then i reload the file from the GitHub repository.

Each dataset contains the country/region, and the number of COVID-19 cases (or deaths or recoveries) reported daily.

```
url_in <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_data.csv"
file_names <- c("time_series_covid19_confirmed_US.csv",
                "time_series_covid19_confirmed_global.csv",
                "time_series_covid19_deaths_US.csv",
                "time_series_covid19_deaths_global.csv",
                "time_series_covid19_recovered_global.csv")
urls <- paste(url_in,file_names,sep = "")
urls
```

```
## [1] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_data.csv"
## [2] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_data.csv"
## [3] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_data.csv"
## [4] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_data.csv"
## [5] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_data.csv"
```

```

if(file.exists("global_confirmed_loaded.Rdata")) {
  print ('Loading Global Cases from cache')
  global_confirmed_loaded <- get(load("global_confirmed_loaded.Rdata"))
}else {
  print ('Loading Global Cases from URL')
  global_confirmed_loaded <- read_csv(urls[2])
  save(global_confirmed_loaded,file = "global_confirmed_loaded.Rdata")
}

```

```
## [1] "Loading Global Cases from cache"
```

```
global_confirmed_loaded
```

```

## # A tibble: 289 x 1,101
##   Provin~1 Count~2   Lat   Long 1/22/~3 1/23/~4 1/24/~5 1/25/~6 1/26/~7 1/27/~8
##   <chr>      <chr>   <dbl> <dbl> <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1 <NA>      Afghan~  33.9  67.7      0      0      0      0      0      0
## 2 <NA>      Albania  41.2  20.2      0      0      0      0      0      0
## 3 <NA>      Algeria  28.0   1.66      0      0      0      0      0      0
## 4 <NA>      Andorra  42.5   1.52      0      0      0      0      0      0
## 5 <NA>      Angola  -11.2  17.9      0      0      0      0      0      0
## 6 <NA>      Antarc~ -71.9  23.3      0      0      0      0      0      0
## 7 <NA>      Antigu~  17.1 -61.8      0      0      0      0      0      0
## 8 <NA>      Argent~ -38.4 -63.6      0      0      0      0      0      0
## 9 <NA>      Armenia  40.1  45.0      0      0      0      0      0      0
## 10 Austral~ Austr~ -35.5 149.      0      0      0      0      0      0
## # ... with 279 more rows, 1,091 more variables: '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>, '2/21/20' <dbl>, ...

```

```

if(file.exists("global_death_loaded.Rdata")) {
  print ('Loading Global Deaths from cache')
  global_death_loaded <- get(load("global_death_loaded.Rdata"))
} else {
  global_death_loaded <- read_csv(urls[4])
  save(global_death_loaded,file = "global_death_loaded.Rdata")
}

```

```
## [1] "Loading Global Deaths from cache"
```

```

if(file.exists("global_recovered_loaded.Rdata")) {
  global_recovered_loaded <- get(load("global_recovered_loaded.Rdata"))
} else {
  global_recovered_loaded <- read_csv(urls[5])
  save(global_recovered_loaded,file = "global_recovered_loaded.Rdata")
}

```

```

if(file.exists("us_confirmed_loaded.Rdata")) {
  print ('Loading US Confirmed from cache')
  us_confirmed_loaded <- get(load("us_confirmed_loaded.Rdata"))
} else {
  us_confirmed_loaded <- read_csv(urls[1])
  save(us_confirmed_loaded,file = "us_confirmed_loaded.Rdata")
}

```

```
## [1] "Loading US Confirmed from cache"
```

```

if(file.exists("us_death_cache.Rdata")) {
  print ('Loading US Deaths from cache')
  us_death_loaded <- get(load("us_death_cache.Rdata"))
} else {
  us_death_loaded <- read_csv(urls[3])
  save(us_death_loaded,file = "us_death_cache.Rdata")
}

```

```
## [1] "Loading US Deaths from cache"
```

Transforming Global COVID-19 data

The data containing the information about the COVID-19 cases from all the countries are available in different set such as confirmed COVID-19 cases per day per region and number of deaths due to COVID-19 per day per region.

```

global_confirmed <- global_confirmed_loaded %>%
  pivot_longer(
    cols = c('Province/State', 'Country/Region', 'Lat', 'Long'),
    names_to = "Date",
    values_to = "Cases") %>%
  select(-c('Lat', 'Long'))

global_death <- global_death_loaded %>%
  pivot_longer(
    cols = c('Province/State', 'Country/Region', 'Lat', 'Long'),
    names_to = "Date",
    values_to = "Deaths") %>%
  select(-c('Lat', 'Long'))

global_recovered <- global_recovered_loaded %>%
  pivot_longer(cols = c('Province/State', 'Country/Region', 'Lat', 'Long'),
    names_to = "Date",
    values_to = "Recovered") %>%
  select(-c('Lat', 'Long'))

global_covid_data <- global_confirmed %>%
  full_join(global_death) %>%
  full_join(global_recovered) %>%
  rename (
    Province_State = 'Province/State',

```

```
Country_Region = 'Country/Region') %>%
mutate(Date = mdy(Date)) %>%
select(-c(Province_State))
```

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

```
summary(global_covid_data )
```

```
## Country_Region      Date      Cases      Deaths
## Length:318130      Min.   :2020-01-22      Min.   :      0      Min.   :      0
## Class :character    1st Qu.:2020-10-22      1st Qu.:     591      1st Qu.:      3
## Mode  :character    Median :2021-07-23      Median :    12787      Median :     138
##                               Mean  :2021-07-23      Mean  :   901951      Mean  :   12945
##                               3rd Qu.:2022-04-23      3rd Qu.:  210940      3rd Qu.:   2848
##                               Max.   :2023-01-22      Max.   :102005805      Max.   :1104118
##                               NA's   :1097          NA's   :1097
## Recovered
## Min.   :      -1
## 1st Qu.:      0
## Median :      0
## Mean   :   78154
## 3rd Qu.:   1053
## Max.   :30974748
## NA's   :17552
```

```
global_covid_data
```

```
## # A tibble: 318,130 x 5
##   Country_Region Date      Cases Deaths Recovered
##   <chr>         <date>    <dbl>  <dbl>    <dbl>
## 1 Afghanistan 2020-01-22      0      0        0
## 2 Afghanistan 2020-01-23      0      0        0
## 3 Afghanistan 2020-01-24      0      0        0
## 4 Afghanistan 2020-01-25      0      0        0
## 5 Afghanistan 2020-01-26      0      0        0
## 6 Afghanistan 2020-01-27      0      0        0
## 7 Afghanistan 2020-01-28      0      0        0
## 8 Afghanistan 2020-01-29      0      0        0
## 9 Afghanistan 2020-01-30      0      0        0
## 10 Afghanistan 2020-01-31      0      0        0
## # ... with 318,120 more rows
```

```
global_cases_by_month <- global_covid_data %>%
  group_by( YearMonth = lubridate::floor_date(Date, 'month')) %>%
  drop_na() %>%
  summarize(Cases = sum(Cases),
            Deaths = sum(Deaths),
            Recovered = sum(Recovered))
```

```
global_cases_by_month_w_fr <- global_cases_by_month %>% mutate(FatalityRatio = (Deaths/Cases * 100))
global_cases_by_month_w_fr
```

```
## # A tibble: 37 x 5
##   YearMonth      Cases  Deaths Recovered FatalityRatio
##   <date>         <dbl>    <dbl>    <dbl>         <dbl>
## 1 2020-01-01     38527      891      869           2.31
## 2 2020-02-01    1671823    46976    380839         2.81
## 3 2020-03-01    8904936   414417   2701204         4.65
## 4 2020-04-01   62554158  4605222  16017554         7.36
## 5 2020-05-01  142784237 10236916  52953018         7.17
## 6 2020-06-01  243843190 14190509 117057627         5.82
## 7 2020-07-01  428473822 19539691 239581869         4.56
## 8 2020-08-01  668393458 25306146 420122385         3.79
## 9 2020-09-01  891340946 29751989 604919337         3.34
## 10 2020-10-01 1223567347 36071703 838117294         2.95
## # ... with 27 more rows
```

Transforming the US COVID-19 data

The dataset containing the information about the COVID-19 cases in US are transformed to get the number of confirmed cases and number of deaths due to COVID-19 per month. The data is also used to find the fatality rate of COVID-19 in US.

```
us_confirmed <- us_confirmed_loaded %>%
  pivot_longer(cols = -c("UID": 'Combined_Key'),
               names_to = "Date",
               values_to = "Cases") %>%
  select('Admin2': 'Cases') %>%
  mutate(Date = mdy(Date)) %>%
  select (-c('Lat', 'Long_')) %>%
  rename ( County = 'Admin2')

us_death <- us_death_loaded %>%
  pivot_longer(cols = -c("UID": 'Combined_Key'),
               names_to = "Date",
               values_to = "Deaths") %>%
  select('Admin2': 'Deaths') %>%
  mutate(Date = mdy(Date)) %>%
  select (-c('Lat', 'Long_')) %>%
  rename ( County = 'Admin2')

us_cases <- us_confirmed %>% full_join(us_death) %>% filter(Cases > 0)
```

```
## Joining, by = c("County", "Province_State", "Country_Region", "Combined_Key",
## "Date")
```

```
us_cases
```

```
## # A tibble: 3,324,940 x 7
##   County Province_State Country_Region Combined_Key   Date      Cases Deaths
##   <chr>    <chr>         <chr>         <chr>         <date>    <dbl>  <dbl>
## 1 Autauga Alabama      US           Autauga, Alaba~ 2020-03-24      1      0
## 2 Autauga Alabama      US           Autauga, Alaba~ 2020-03-25      5      0
## 3 Autauga Alabama      US           Autauga, Alaba~ 2020-03-26      6      0
```

```
## 4 Autauga Alabama US Autauga, Alaba~ 2020-03-27 6 0
## 5 Autauga Alabama US Autauga, Alaba~ 2020-03-28 6 0
## 6 Autauga Alabama US Autauga, Alaba~ 2020-03-29 6 0
## 7 Autauga Alabama US Autauga, Alaba~ 2020-03-30 8 0
## 8 Autauga Alabama US Autauga, Alaba~ 2020-03-31 8 0
## 9 Autauga Alabama US Autauga, Alaba~ 2020-04-01 10 0
## 10 Autauga Alabama US Autauga, Alaba~ 2020-04-02 12 0
## # ... with 3,324,930 more rows
```

```
us_cases_by_month <- us_cases %>%
  group_by(YearMonth = lubridate::floor_date(Date, 'month'), Country_Region) %>%
  summarize(Cases = sum(Cases),
            Deaths = sum(Deaths))
```

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

```
us_cases_by_month_w_fr <- us_cases_by_month %>% mutate(FatalityRatio = (Deaths/Cases * 100))
us_cases_by_month_w_fr
```

```
## # A tibble: 37 x 5
## # Groups:   YearMonth [37]
##   YearMonth Country_Region Cases Deaths FatalityRatio
##   <date>      <chr>      <dbl> <dbl>      <dbl>
## 1 2020-01-01 US           41      0          0
## 2 2020-02-01 US          420      1        0.238
## 3 2020-03-01 US       1121565  23973        2.14
## 4 2020-04-01 US       19977575  993919        4.98
## 5 2020-05-01 US      45414972 2700641        5.95
## 6 2020-06-01 US      64902874 3564131        5.49
## 7 2020-07-01 US     111253119 4318723        3.88
## 8 2020-08-01 US     166652074 5247535        3.15
## 9 2020-09-01 US     199762036 5850150        2.93
## 10 2020-10-01 US     251587325 6747877        2.68
## # ... with 27 more rows
```

```
us_cases_by_state <- us_cases %>%
  group_by(Country, Province_State, Country_Region, Date) %>%
  summarize(Total_Cases = sum(Cases), Total_Deaths = sum(Deaths)) %>%
  select('Province_State', 'Country_Region', 'Date', 'Total_Cases', 'Total_Deaths') %>%
  ungroup()
```

```
## 'summarise()' has grouped output by 'Country', 'Province_State',
## 'Country_Region'. You can override using the '.groups' argument.
## Adding missing grouping variables: 'Country'
```

```
tail(us_cases_by_state)
```

```
## # A tibble: 6 x 6
##   County Province_State Country_Region Date Total_Cases Total_Deaths
##   <chr> <chr>      <chr>      <date>      <dbl>      <dbl>
```

```
## 1 <NA>    Virgin Islands US          2023-01-17      24138      129
## 2 <NA>    Virgin Islands US          2023-01-18      24176      129
## 3 <NA>    Virgin Islands US          2023-01-19      24228      129
## 4 <NA>    Virgin Islands US          2023-01-20      24269      129
## 5 <NA>    Virgin Islands US          2023-01-21      24269      129
## 6 <NA>    Virgin Islands US          2023-01-22      24269      129
```

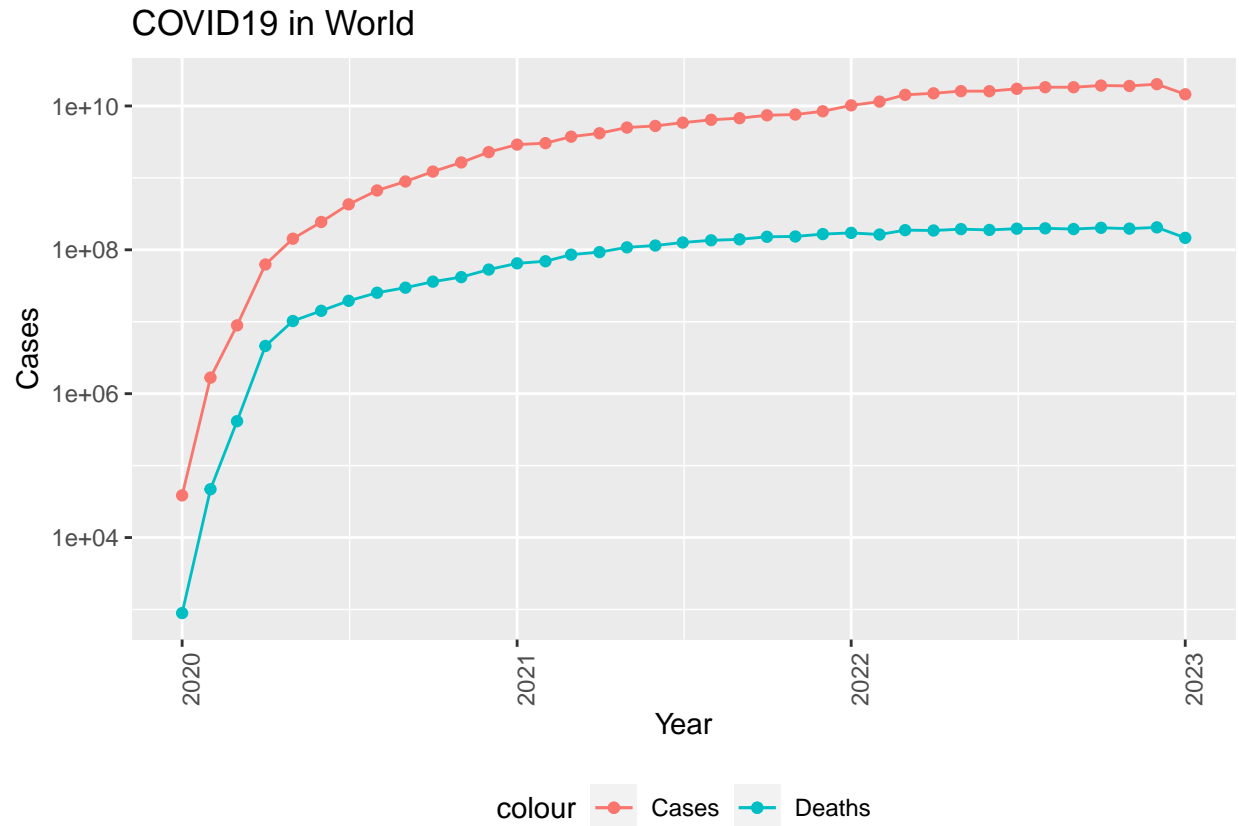
```
us_total_cases <- us_cases_by_state %>%
  group_by(Country_Region, Date) %>%
  summarize(
    Total_Cases = sum(Total_Cases),
    Total_Deaths = sum(Total_Deaths)
  ) %>%
  select(Country_Region, Date, Total_Cases, Total_Deaths) %>%
  ungroup()
```

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

Visualization

The below plot represents the COVID-19 trend for each year for the entire world. This visualization shows the number of COVID-19 cases and deaths due to COVID-19 happened yearly.

```
ggplot(data = global_cases_by_month, aes(x=YearMonth,y=Cases)) +
  geom_line(aes(color='Cases')) +
  geom_point(aes(color='Cases')) +
  geom_point(aes(y= Deaths, color='Deaths')) +
  geom_line(aes(y=Deaths, color = 'Deaths')) +
  scale_y_log10() +
  #scale_y_continuous(labels = scales::label_number_si()) +
  #scale_y_continuous(labels = scales::scale_y_log10()) +
  theme(legend.position = 'bottom', axis.text.x = element_text(angle=90)) +
  labs(title = "COVID19 in World", y = 'Cases', x = 'Year')
```



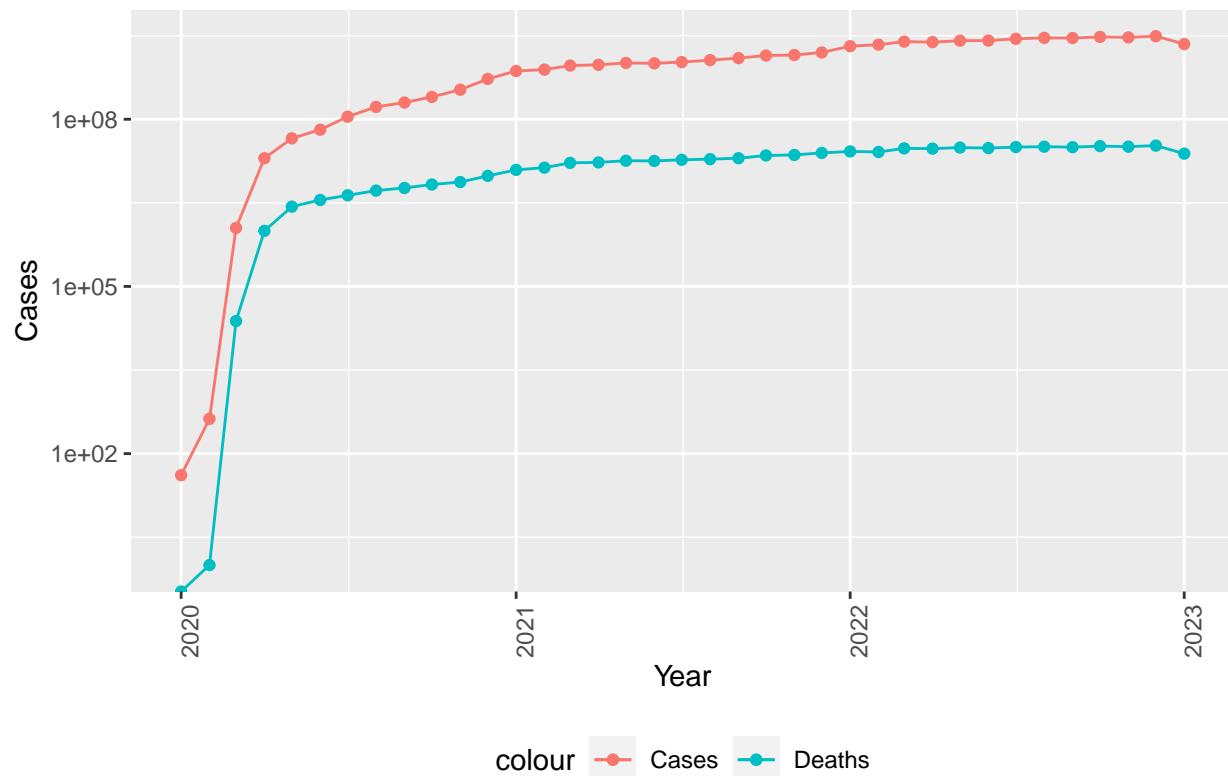
The below plot represents the COVID-19 trend for each year for United States. This visualization shows the number of COVID-19 cases and deaths due to COVID-19 yearly.

```
addUnits <- function(n) {
  labels <- ifelse(n < 1000, n, # less than thousands
    ifelse(n < 1e6, paste0(round(n/1e3), 'k'), # in thousands
      ifelse(n < 1e9, paste0(round(n/1e6), 'M'), # in millions
        ifelse(n < 1e12, paste0(round(n/1e9), 'B'), # in billions
          ifelse(n < 1e15, paste0(round(n/1e12), 'T'), # in trillions
            'too big!'
          )
        )
      )
    )
  return(labels)
}

ggplot(data = us_cases_by_month, aes(x=YearMonth,y=Cases)) +
  geom_line(aes(color='Cases')) +
  geom_point(aes(color='Cases')) +
  geom_point(aes(y= Deaths, color='Deaths')) +
  geom_line(aes(y=Deaths, color = 'Deaths')) +
  scale_y_log10() +
  #scale_y_continuous(labels = addUnits)
  theme(legend.position = 'bottom', axis.text.x = element_text(angle=90)) +
  labs(title = "COVID19 in US",y = 'Cases', x = 'Year')

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

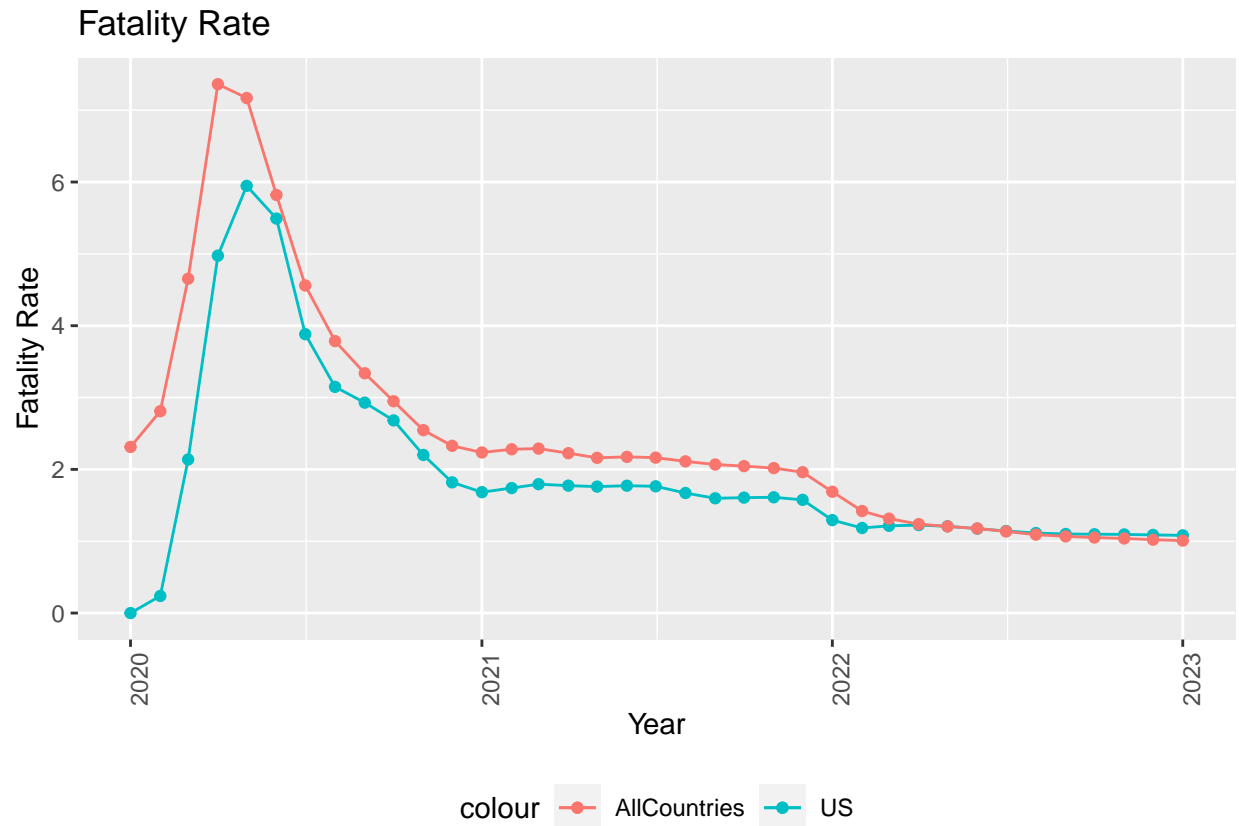

COVID19 in US



The below plot represents the comparison of the fatality rate globally verse the United States yearly. The fatality is the number of deaths in the reported COVID cases per number of COVID-19 cases reported.

```
all_fatality_rate <- us_cases_by_month_w_fr %>%
  rename(FatalityRatioUS = FatalityRatio) %>%
  right_join(global_cases_by_month_w_fr, by="YearMonth") %>%
  rename(FatalityRatioAllCountries = FatalityRatio) %>%
  select (c(YearMonth, FatalityRatioUS, FatalityRatioAllCountries))

ggplot(data = all_fatality_rate, aes(x=YearMonth, y=FatalityRatioUS)) +
  geom_point(aes(color='US')) +
  geom_line(aes(color='US')) +
  geom_point(aes(y= FatalityRatioAllCountries, color='AllCountries')) +
  geom_line(aes(y=FatalityRatioAllCountries, color = 'AllCountries')) +
  theme(legend.position = 'bottom', axis.text.x = element_text(angle=90)) +
  labs(title = "Fatality Rate", x='Year', y= 'Fatality Rate')
```



Modelling

The below model identifies the number of deaths based on the total number of cases reported.

Please note that it doesn't consider external factors like availability of vaccination, immunity gained in people who already had COVID in the past, etc. The Model uses all the available data to train the data, and the same data is used to plot the values to check how well the model is trained on the current data. Ideally, a different dataset should have been used to test it well.

```
mod <- lm(Total_Deaths ~ Total_Cases, data = us_total_cases)
summary(mod)
```

```
##
## Call:
## lm(formula = Total_Deaths ~ Total_Cases, data = us_total_cases)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -145840  -63973  -11730    89065   140048
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.458e+05  3.911e+03  37.29  <2e-16 ***
## Total_Cases 1.022e-02  6.860e-05  148.97  <2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 80300 on 1095 degrees of freedom
## Multiple R-squared:  0.953, Adjusted R-squared:  0.9529
## F-statistic: 2.219e+04 on 1 and 1095 DF, p-value: < 2.2e-16

us_total_cases_w_pred <- us_total_cases %>% mutate(PRED_DEATHS = predict(mod))

ggplot(data = us_total_cases_w_pred ) +
  geom_point(aes(x = Total_Cases, y = Total_Deaths ), color = "blue") +
  geom_point(aes(x = Total_Cases, y = PRED_DEATHS ), color = "red") +
  theme(legend.position = 'bottom', axis.text.x = element_text(angle=90)) +
  labs(title = "COVID-19 Actual verses Predicted", x='Cases', y= 'Deaths')
```

