

Covid-19 Data - Analyze / Visualize / Model

Kothandaraman Sikamani

12/02/2022

Covid 19 Global/US data analyzing and modeling

Purpose

In this document, I will be explaining/focusing on Analytics, Visualization, and Model building using Covid-19 Data Dataset from John Hopkins University. At a high level, I will be addressing the following topics.

1. Covid Cases from Global/US.

- Summarizing global covid-19 cases
- Summarizing global covid-19 deaths
- Summarizing US covid-19 cases
- Summarizing US covid-19 deaths

2. Tidying and Transforming Data

- Summarizing US covid-19 cases
- Summarizing US covid-19 deaths

3. Visualization and Analysis for Covid-19 cases from Global/US.

- Visualization of state wise cases/deaths using plots.
- Analyzing state wise data
- Analyzing state wise maximum cases and deaths

4. Model.

- Data preparation for Model
- LM model building
- Summarizing and analyzing model.
- Understanding model predictions and plot the model prediction visually.

5. Bias.

- Describe if any bias situations that can help improve model performance.

Data Source

I am using the data source from John Hopkins github for US/Global cases/deaths as csv format https://github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data/csse_covid_19_time_series/

From file URL's

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

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
```

Import the global data

I will read the .csv file using read.csv()

```
# Reading global cases and deaths raw csv data

global_cases <- read.csv(urls[1])
global_deaths <- read.csv(urls[2])

#global_cases.
#head(global_cases, 5)
#global_deaths
#head(global_deaths, 5)
```

1. Covid Cases from Global/US Raw data

2. Tidying and Transforming Data:

```
## Import the US data
# I will read the .csv file using read.csv()

us_cases <- read.csv(urls[3])
us_deaths <- read.csv(urls[4])

#us_cases
#us_deaths
#summary(us_cases)
#summary(us_deaths)
#data.table(us_cases)
#data.table(us_deaths)

# Data Cleaup
us_cases <- us_cases %>%
```

```

pivot_longer(cols = -c('Province_State',
                        'Country_Region', UID, iso2, iso3, code3, FIPS,
                        Combined_Key, Admin2, Lat, Long_),
             names_to = "date",
             values_to = "cases") %>%
select(-c(UID, iso2, iso3, code3, FIPS, Lat, Long_))

us_deaths <- us_deaths %>%
  pivot_longer(cols = -c('Province_State',
                          'Country_Region', UID, iso2, iso3, code3, FIPS,
                          Combined_Key, Population, Admin2, Lat, Long_),
               names_to = "date",
               values_to = "deaths") %>%
  select(-c(UID, iso2, iso3, code3, FIPS, Lat, Long_))

us_cases$date <- gsub("\\.", "-", us_cases$date)
us_cases$date <- gsub("\\X", "", us_cases$date)
us_deaths$date <- gsub("\\.", "-", us_deaths$date)
us_deaths$date <- gsub("\\X", "", us_deaths$date)

us_cases <- us_cases %>%
  mutate(date = mdy(date))
us_deaths <- us_deaths %>%
  mutate(date = mdy(date))

us_cases

```

```

## # A tibble: 2,516,526 x 6
##   Admin2 Province_State Country_Region Combined_Key      date      cases
##   <chr>   <chr>          <chr>         <chr>      <date>    <int>
## 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 2,516,516 more rows

```

```

us_deaths

## # A tibble: 2,516,526 x 7
##   Admin2 Province_State Country_Region Combined_Key Population date
##   <chr>   <chr>          <chr>         <chr>      <int> <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 2,516,516 more rows, and 1 more variable: deaths <int>
```

```
# Summarizing US cases:
summary(us_cases)
```

```
##      Admin2      Province_State      Country_Region      Combined_Key
## Length:2516526 Length:2516526 Length:2516526 Length:2516526
## Class :character Class :character Class :character Class :character
## Mode  :character Mode  :character Mode  :character Mode  :character
##
##
##
##      date      cases
## Min.   :2020-01-22 Min.   :      0
## 1st Qu.:2020-07-28 1st Qu.:     76
## Median :2021-02-01 Median :    1017
## Mean   :2021-02-01 Mean   :    7133
## 3rd Qu.:2021-08-08 3rd Qu.:    4016
## Max.   :2022-02-12 Max.   :   2757058
```

```
# Summarizing US Deaths:
summary(us_deaths)
```

```
##      Admin2      Province_State      Country_Region      Combined_Key
## Length:2516526 Length:2516526 Length:2516526 Length:2516526
## Class :character Class :character Class :character Class :character
## Mode  :character Mode  :character Mode  :character Mode  :character
##
##
##
##      Population      date      deaths
## Min.   :      0 Min.   :2020-01-22 Min.   :      0.0
## 1st Qu.:    9917 1st Qu.:2020-07-28 1st Qu.:      1.0
## Median :   24892 Median :2021-02-01 Median :     18.0
## Mean   :   99604 Mean   :2021-02-01 Mean   :    122.7
## 3rd Qu.:   64979 3rd Qu.:2021-08-08 3rd Qu.:     72.0
## Max.   :  10039107 Max.   :2022-02-12 Max.   :  29846.0
```

```
# data.table(us_cases)
# data.table(us_deaths)

# Join us_cases data with us_deaths dataset

US <- us_cases %>%
  full_join(us_deaths)
```

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

```

# Create Combined_Key column in global dataframe using Province_State, Country_Region)
# Column rename
global <- global %>%
  unite("Combined_Key",
        c(Province_State, Country_Region),
        sep = ", ",
        na.rm = TRUE,
        remove = FALSE)

```

```

# Import Lookup data for population details and additional features

```

```

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

```

```

uid <- read_csv(uid_lookup_url)

```

```

## Rows: 4218 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.

```

```

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

spec(uid)

```

```

## cols(
##   UID = col_double(),
##   iso2 = col_character(),
##   iso3 = col_character(),
##   code3 = col_double(),
##   FIPS = col_character(),
##   Admin2 = col_character(),
##   Province_State = col_character(),
##   Country_Region = col_character(),
##   Lat = col_double(),
##   Long_ = col_double(),
##   Combined_Key = col_character(),
##   Population = col_double()
## )

```

```

# Join global data with UID dataset

```

```

global <- global %>%
  left_join(uid, by = c("Province_State", "Country_Region")) %>%
  select(-c(UID, FIPS))

```

```

global <- global %>%
  select(Province_State, Country_Region, date, cases, deaths,

```

```

      Population, Combined_Key.x)

global <- global %>%
  rename(Combined_Key = `Combined_Key.x`)

#global

```

3. Visualization and Analysis for Covid cases from Global/US

```

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
summary(US_by_state)

```

```

## Province_State      Country_Region      date      cases
## Length:43674      Length:43674      Min.   :2020-01-22      Min.   :      0
## Class :character   Class :character   1st Qu.:2020-07-28      1st Qu.:    8232
## Mode  :character   Mode  :character   Median :2021-02-01      Median :   121871
##                                     Mean  :2021-02-01      Mean   :   410987
##                                     3rd Qu.:2021-08-08      3rd Qu.:  497558
##                                     Max.   :2022-02-12      Max.   : 8804417
##
##      deaths      deaths_per_mill      Population
## Min.   :      0      Min.   :    0.0      Min.   :      0
## 1st Qu.:   176      1st Qu.:  167.4      1st Qu.: 1068778
## Median :   2114      Median :   874.5      Median : 3660113
## Mean   :   7070      Mean   :    Inf      Mean   : 5739226
## 3rd Qu.:   8451      3rd Qu.: 1851.4      3rd Qu.: 6892503
## Max.   :  82500      Max.   :    Inf      Max.   :39512223
##                                     NA's   :821

```

```

US_totals <- 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.
```

```
summary(US_totals)
```

```
## Country_Region      date      cases      deaths
## Length:753      Min.   :2020-01-22      Min.   :      1      Min.   :      1
## Class :character  1st Qu.:2020-07-28      1st Qu.: 4346567      1st Qu.:149916
## Mode  :character  Median :2021-02-01      Median :26470178      Median :450185
##                      Mean   :2021-02-01      Mean   :23837231      Mean   :410066
##                      3rd Qu.:2021-08-08      3rd Qu.:35905164      3rd Qu.:616674
##                      Max.   :2022-02-12      Max.   :77707349      Max.   :919253
## deaths_per_mill      Population
## Min.   :      0.003      Min.   :332875137
## 1st Qu.: 450.367      1st Qu.:332875137
## Median :1352.414      Median :332875137
## Mean   :1231.891      Mean   :332875137
## 3rd Qu.:1852.569      3rd Qu.:332875137
## Max.   :2761.555      Max.   :332875137
```

```
#US_totals
```

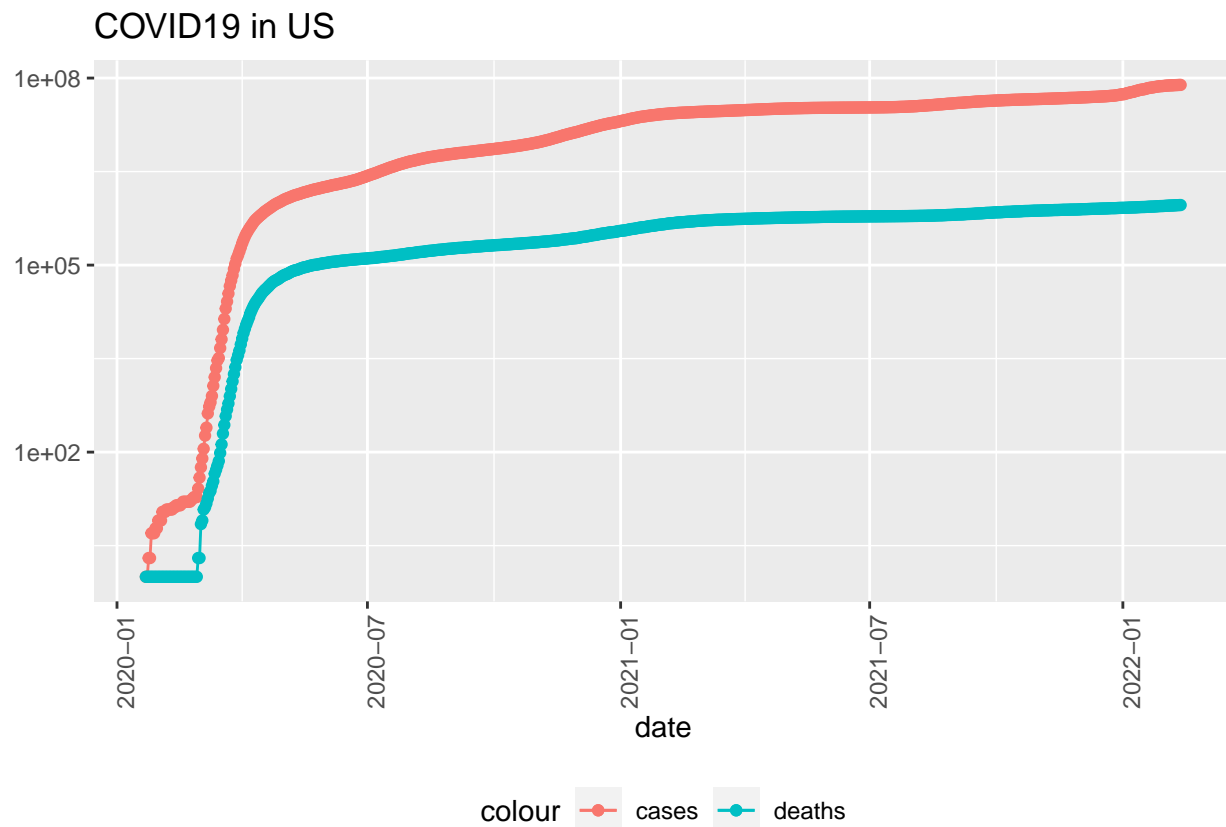
```
summary(US_by_state)
```

```
## Province_State      Country_Region      date      cases
## Length:43674      Length:43674      Min.   :2020-01-22      Min.   :      0
## Class :character    Class :character  1st Qu.:2020-07-28      1st Qu.: 8232
## Mode  :character    Mode  :character  Median :2021-02-01      Median : 121871
##                      Mean   :2021-02-01      Mean   : 410987
##                      3rd Qu.:2021-08-08      3rd Qu.: 497558
##                      Max.   :2022-02-12      Max.   :8804417
##
##      deaths      deaths_per_mill      Population
## Min.   :      0      Min.   :      0.0      Min.   :      0
## 1st Qu.: 176      1st Qu.: 167.4      1st Qu.: 1068778
## Median : 2114      Median : 874.5      Median : 3660113
## Mean   : 7070      Mean   :      Inf      Mean   : 5739226
## 3rd Qu.: 8451      3rd Qu.:1851.4      3rd Qu.: 6892503
## Max.   :82500      Max.   :      Inf      Max.   :39512223
##                      NA's      :821
```

```
tail(US_totals)
```

```
## # A tibble: 6 x 6
##   Country_Region date      cases deaths deaths_per_mill Population
##   <chr>      <date>      <int> <int>      <dbl>      <int>
## 1 US      2022-02-07 76861658 906330      2723.    332875137
## 2 US      2022-02-08 77083020 909233      2731.    332875137
## 3 US      2022-02-09 77270319 912668      2742.    332875137
## 4 US      2022-02-10 77441181 915847      2751.    332875137
## 5 US      2022-02-11 77650446 918451      2759.    332875137
## 6 US      2022-02-12 77707349 919253      2762.    332875137
```

```
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)
```



Top 5 cases count happened by date wise / Maximum cases in a day :

```
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)
```

```
## # A tibble: 6 x 8
```



```
## Country_Region date cases deaths deaths_per_mill Population new_cases
## <chr> <date> <int> <int> <dbl> <int> <int>
## 1 US 2022-02-07 76861658 906330 2723. 332875137 338779
## 2 US 2022-02-08 77083020 909233 2731. 332875137 221362
## 3 US 2022-02-09 77270319 912668 2742. 332875137 187299
## 4 US 2022-02-10 77441181 915847 2751. 332875137 170862
## 5 US 2022-02-11 77650446 918451 2759. 332875137 209265
## 6 US 2022-02-12 77707349 919253 2762. 332875137 56903
## # ... with 1 more variable: new_deaths <int>
```

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

```
## # A tibble: 6 x 8
## new_cases new_deaths Country_Region date cases deaths deaths_per_mill
## <int> <int> <chr> <date> <int> <int> <dbl>
## 1 338779 2920 US 2022-02-07 76861658 906330 2723.
## 2 221362 2903 US 2022-02-08 77083020 909233 2731.
## 3 187299 3435 US 2022-02-09 77270319 912668 2742.
## 4 170862 3179 US 2022-02-10 77441181 915847 2751.
## 5 209265 2604 US 2022-02-11 77650446 918451 2759.
## 6 56903 802 US 2022-02-12 77707349 919253 2762.
## # ... with 1 more variable: Population <int>
```

```
state <- "NEW YORK"

# sort dataframe by column in r
# select top N results

#spec(US_totals)
#US_totals <- US_totals %>% filter(!is.na(new_cases))
#spec(US_totals)

US_totals_By_Date_TOP_5_Cases_Count <- US_totals[order(-US_totals$new_cases),][1:5,]

US_totals_By_Date_TOP_5_Cases_Count
```

```
## # A tibble: 5 x 8
## Country_Region date cases deaths deaths_per_mill Population new_cases
## <chr> <date> <int> <int> <dbl> <int> <int>
## 1 US 2022-01-10 61690604 842422 2531. 332875137 1368563
## 2 US 2022-01-18 67705084 857674 2577. 332875137 1113068
## 3 US 2022-01-03 56337901 830371 2495. 332875137 1076439
## 4 US 2022-01-19 68703526 861595 2588. 332875137 998442
## 5 US 2022-01-12 63368974 847725 2547. 332875137 909036
## # ... with 1 more variable: new_deaths <int>
```

```
US_totals_By_Date_TOP_Cases <- US_totals_By_Date_TOP_5_Cases_Count[1,]

#US_totals_By_Date_TOP_Cases

# Maximum cases in a day
max(US_totals_By_Date_TOP_Cases$new_cases)
```

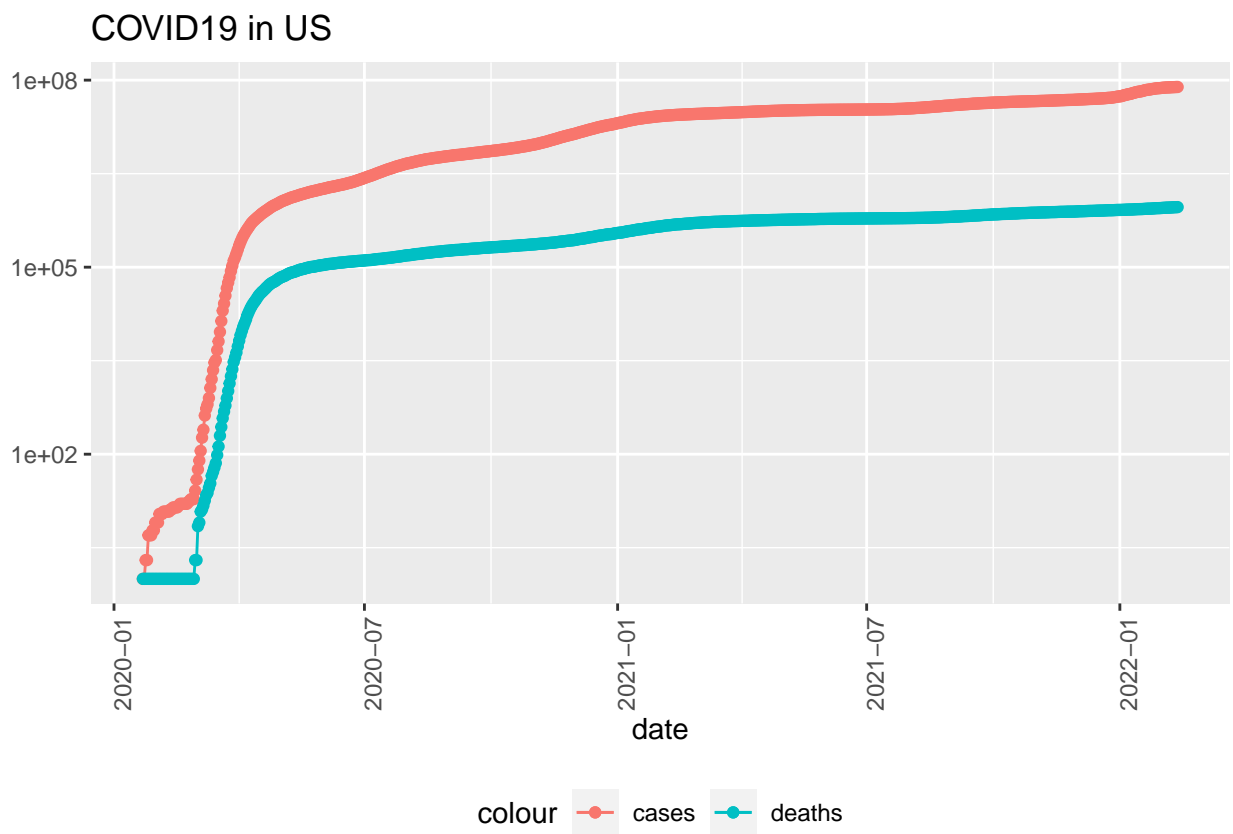
```
## [1] 1368563
```

```
# Maximum cases day - date  
max(US_totals_By_Date_TOP_Cases$date)
```

```
## [1] "2022-01-10"
```

Analizing Data:

```
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)
```



```
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>          <int>  <int>    <int>         <dbl>         <dbl>
## 1 American Samoa      0 1.8 e1    55641         0.324         0
## 2 Northern Mariana Isl~ 23 7.35e3    55144        133.         0.417
## 3 Hawaii            1258 2.31e5   1415872       163.         0.888
## 4 Vermont           566 1.09e5    623989       175.         0.907
## 5 Virgin Islands     105 1.52e4    107268       142.         0.979
## 6 Puerto Rico        4025 4.69e5   3754939       125.         1.07
## 7 Utah              4261 9.11e5   3205958       284.         1.33
## 8 Maine              1828 1.88e5   1344212       140.         1.36
## 9 Washington        11316 1.41e6   7614893       185.         1.49
## 10 Alaska            1114 2.32e5    740995       313.         1.50
```

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

US_state_totals
```

```
## # A tibble: 10 x 6
##   deaths_per_thou cases_per_thou Province_State      deaths  cases population
##   <dbl>          <dbl> <chr>          <int>  <int>    <int>
## 1 0 0.324 American Samoa      0 1.8 e1    55641
## 2 0.417 133. Northern Mariana Isl~ 23 7.35e3    55144
## 3 0.888 163. Hawaii            1258 2.31e5   1415872
## 4 0.907 175. Vermont           566 1.09e5    623989
## 5 0.979 142. Virgin Islands     105 1.52e4    107268
## 6 1.07 125. Puerto Rico        4025 4.69e5   3754939
## 7 1.33 284. Utah              4261 9.11e5   3205958
## 8 1.36 140. Maine              1828 1.88e5   1344212
## 9 1.49 185. Washington        11316 1.41e6   7614893
## 10 1.50 313. Alaska            1114 2.32e5    740995
```

```
summary(US_state_totals)
```

```
## deaths_per_thou cases_per_thou Province_State      deaths
## Min. :0.0000 Min. : 0.3235 Length:10 Min. : 0.0
## 1st Qu.:0.8931 1st Qu.:134.9147 Class :character 1st Qu.: 220.2
## Median :1.0254 Median :152.3341 Mode :character Median : 1186.0
## Mean :0.9942 Mean :165.9996 Mean : 2449.6
```

```
## 3rd Qu.:1.3522 3rd Qu.:182.5895 3rd Qu.: 3475.8
## Max. :1.5034 Max. :313.2302 Max. :11316.0
## cases population
## Min. : 18 Min. : 55144
## 1st Qu.: 38629 1st Qu.: 236448
## Median : 209338 Median :1042604
## Mean : 357240 Mean :1891891
## 3rd Qu.: 409479 3rd Qu.:2758436
## Max. :1410596 Max. :7614893
```

4. Model Building - Linear Regression model

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

```
# Sumamry - Model
```

```
summary(mod)
```

```
##
## Call:
## lm(formula = deaths_per_thou ~ cases_per_thou, data = US_state_totals)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.4347 -0.1682 -0.1090  0.2158  0.4807
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.270574   0.226618   1.194  0.26669
## cases_per_thou 0.004359   0.001223   3.563  0.00737 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.318 on 8 degrees of freedom
## Multiple R-squared:  0.6135, Adjusted R-squared:  0.5651
## F-statistic: 12.7 on 1 and 8 DF, p-value: 0.007366
```

```
US_state_totals %>% slice_min(cases_per_thou)
```

```
## # A tibble: 1 x 6
##   deaths_per_thou cases_per_thou Province_State deaths cases population
##           <dbl>         <dbl> <chr>           <int> <int>      <int>
## 1             0           0.324 American Samoa     0    18      55641
```

```
US_state_totals %>% slice_max(cases_per_thou)
```

```
## # A tibble: 1 x 6
##   deaths_per_thou cases_per_thou Province_State deaths cases population
##           <dbl>         <dbl> <chr>           <int> <int>      <int>
## 1           1.50          313. Alaska          1114 232102    740995
```

```
x_grid <- seq(1, 151)

new_df <- tibble(cases_per_thou = x_grid)

# Create Prediction column from model

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

US_tot_w_pred
```

```
## # A tibble: 10 x 7
##   deaths_per_thou cases_per_thou Province_State deaths cases population pred
##           <dbl>         <dbl> <chr>           <int> <int>      <int> <dbl>
## 1             0           0.324 American Samoa      0 1.8 e1      55641 0.272
## 2          0.417          133. Northern Maria~    23 7.35e3      55144 0.852
## 3          0.888          163. Hawaii             1258 2.31e5     1415872 0.982
## 4          0.907          175. Vermont             566 1.09e5      623989 1.03
## 5          0.979          142. Virgin Islands     105 1.52e4      107268 0.888
## 6          1.07          125. Puerto Rico       4025 4.69e5     3754939 0.815
## 7          1.33          284. Utah             4261 9.11e5     3205958 1.51
## 8          1.36          140. Maine             1828 1.88e5     1344212 0.879
## 9          1.49          185. Washington      11316 1.41e6     7614893 1.08
## 10         1.50          313. Alaska             1114 2.32e5      740995 1.64
```

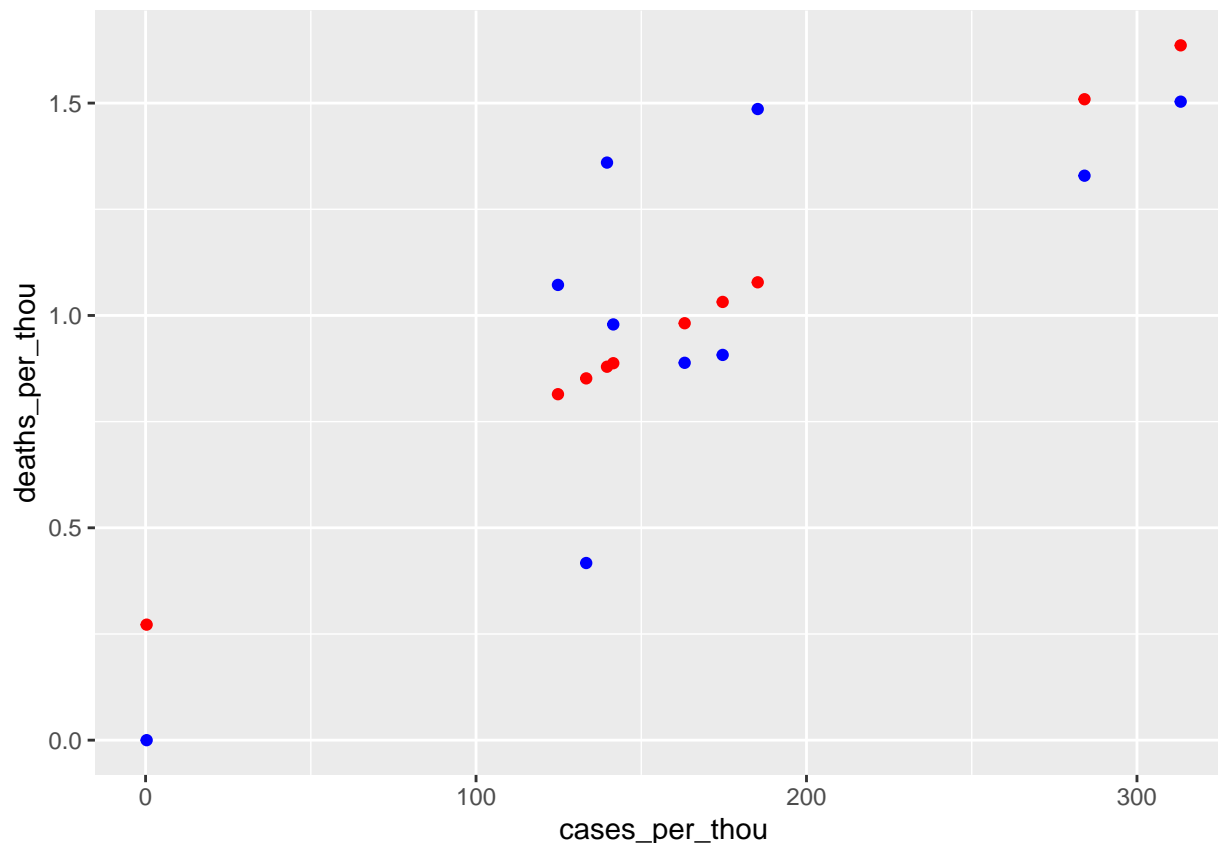
```
summary(US_tot_w_pred)
```

```
## deaths_per_thou cases_per_thou Province_State deaths
## Min. :0.0000 Min. : 0.3235 Length:10 Min. : 0.0
## 1st Qu.:0.8931 1st Qu.:134.9147 Class :character 1st Qu.: 220.2
## Median :1.0254 Median :152.3341 Mode :character Median : 1186.0
## Mean :0.9942 Mean :165.9996 Mean : 2449.6
## 3rd Qu.:1.3522 3rd Qu.:182.5895 3rd Qu.: 3475.8
## Max. :1.5034 Max. :313.2302 Max. :11316.0
## cases population pred
## Min. : 18 Min. : 55144 Min. :0.2720
## 1st Qu.: 38629 1st Qu.: 236448 1st Qu.:0.8587
## Median : 209338 Median :1042604 Median :0.9346
## Mean : 357240 Mean :1891891 Mean :0.9942
## 3rd Qu.: 409479 3rd Qu.:2758436 3rd Qu.:1.0665
## Max. :1410596 Max. :7614893 Max. :1.6360
```

```
# global_cases <- global_cases %>%
#   mutate(date = mdy(date))

# Model plot - Visualization

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")
```



Model Performance and Coefficients: – a and b values vary

From the model performance above, we can see the values of the intercept (“a” value) and the slope (“b” value) for the year. These “a” and “b” values plot a line between all the points of the data. So in this case, If there is a cases_per_thou that is 250 , a is 0.270799 and b is 0.004362, the model predicts (on average) that its death count is around $0.270799 + ((0.004362) * 250) = 1.36 = \sim 1$.

It might be possible to get better model performance by considering other features like infectious disease spread model, non-pharmaceutical interventions, authority policies, vaccine, health-related info, and lifestyle information.

5. Bias:

The above model currently used only samples of the time-series data to predict the future number of cases. A potential future direction to improve the estimation accuracy is to incorporate constraints such as infectious disease spread model, non-pharmaceutical interventions, authority policies, vaccine, health-related info, and lifestyle information. There is a possibility of some types of biases in the COVID-19 dataset. Then we started looking at deaths per 1000 or deaths per million.

It’s different depending on the variables that we are measuring. By reducing noise and adding more features it’s highly possible to predict better test results close to training data and the model can eventually perform better. With that said, it is important to monitor the data preparation processes closely to make sure the datasets are as bias-free as possible before they are used in the training phase.

Selection Bias: This seems like not an issue as this data is from John Hopkins github.

Overfitting and Underfitting: When a model gets trained with large amounts of data, it also starts learning from the noise and inaccurate data entries in the dataset. Consequently, the model does not categorize the data correctly, because of too many details and noise. In this data set, lat lang or many other features can cause noise but can be reduced.

Exclusion Bias: It's possible excluding some features can cause higher bias and this can be reduced including some features that can reduce bias like climate and economic situations and political situations, and inflation and seasons can be included to get more accurate model performance.

Conclusion:

To conclude, I have done the Visualizations, Model, and Bias from the above. The answers are as follows:

1. Summarized Global/US cases and deaths separately.
2. Visualized state-wise cases/deaths using plots, Analyzed state-wise data, Analyzed state-wise maximum cases and deaths.
3. Prepared data for the Model
 - a. LM model building
 - b. Summarized and analyzed model.
 - c. Understanding model predictions and plotting the model prediction visually.