Final Project 2 Covid19 Data

William Vernon

2024-02-29

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

file_names <- c("time_series_covid19_confirmed_global.csv","time_series_covid19_deaths_global.csv","time

urls <- str_c(url_in,file_names)
```

```
global_cases <- read_csv(urls[1])
global_deaths <- read_csv(urls[2])
US_cases <- read_csv(urls[3])
US_deaths <- read_csv(urls[4])
```

```
# Reshape global_cases from wide to long format, removing Lat and Long columns
global_cases_long <- global_cases %>%
  pivot_longer(
    cols = -c("Province/State", "Country/Region", Lat, Long),
    names_to = "date",
    values_to = "cases"
  ) %>%
  select(-c(Lat, Long))

# Reshape global_deaths from wide to long format, removing Lat and Long columns
global_deaths_long <- global_deaths %>%
  pivot_longer(
    cols = -c("Province/State", "Country/Region", Lat, Long),
    names_to = "date",
    values_to = "deaths"
  ) %>%
  select(-c(Lat, Long))

# Join the cases and deaths data frames, rename columns, and convert date format
global <- global_cases_long %>%
  full_join(global_deaths_long, by = c("Province/State", "Country/Region", "date")) %>%
  rename(
    Country_Region = "Country/Region",
    Province_State = "Province/State"
  ) %>%
  mutate(date = mdy(date))
```

```
summary(global)
```

```
##  Province_State     Country_Region          date                  cases
##  Length:330327      Length:330327      Min.   :2020-01-22   Min.   :        0  
##  Class :character   Class :character   1st Qu.:2020-11-02   1st Qu.:      680  
##  Mode  :character   Mode  :character   Median :2021-08-15   Median :    14429  
##                                        Mean   :2021-08-15   Mean   :   959384  
##                                        3rd Qu.:2022-05-28   3rd Qu.:   228517  
##                                        Max.   :2023-03-09   Max.   :103802702  
##      deaths       
##  Min.   :      0  
##  1st Qu.:      3  
##  Median :    150  
##  Mean   :  13380  
##  3rd Qu.:   3032  
##  Max.   :1123836
```

```
# Remove any cases that are 0
global <- global %>% filter(cases > 0)

summary(global)
```

```
##  Province_State     Country_Region          date                cases          
##  Length:306827      Length:306827      Min.   :2020-01-22   Min.   :        1  
##  Class :character   Class :character   1st Qu.:2020-12-12   1st Qu.:     1316  
##  Mode  :character   Mode  :character   Median :2021-09-16   Median :    20365  
##                                        Mean   :2021-09-11   Mean   :  1032863  
##                                        3rd Qu.:2022-06-15   3rd Qu.:   271281  
##                                        Max.   :2023-03-09   Max.   :103802702  
##      deaths       
##  Min.   :      0  
##  1st Qu.:      7  
##  Median :    214  
##  Mean   :  14405  
##  3rd Qu.:   3665  
##  Max.   :1123836
```

US Cases

```
# Cleaning US_cases
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_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_))

# Joining US_cases and US_deaths
US <- US_cases %>%
    full_join(US_deaths)
```

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

```
# Combined Keys so both data sets have the same keys
global <- global %>%
    unite("Combined_Key",
          c(Province_State, Country_Region),
          sep = ", ",
          na.rm = TRUE,
          remove = FALSE)
global
```

```
## # A tibble: 306,827 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
## # i 306,817 more rows
```

```
uid_lookup_url <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/
uid <- read_csv(uid_lookup_url, show_col_types = FALSE) %>%
  select(-c(Lat, Long_, Combined_Key, code3, iso2, iso3, Admin2))
```

```
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)
global
```

```
## # A tibble: 306,827 x 7
##    Province_State Country_Region date       cases deaths Population Combined_Key
##    <chr>          <chr>          <date>     <dbl>  <dbl>      <dbl> <chr>
##  1 <NA>           Afghanistan    2020-02-24     5      0   38928341 Afghanistan
##  2 <NA>           Afghanistan    2020-02-25     5      0   38928341 Afghanistan
##  3 <NA>           Afghanistan    2020-02-26     5      0   38928341 Afghanistan
##  4 <NA>           Afghanistan    2020-02-27     5      0   38928341 Afghanistan
##  5 <NA>           Afghanistan    2020-02-28     5      0   38928341 Afghanistan
##  6 <NA>           Afghanistan    2020-02-29     5      0   38928341 Afghanistan
##  7 <NA>           Afghanistan    2020-03-01     5      0   38928341 Afghanistan
##  8 <NA>           Afghanistan    2020-03-02     5      0   38928341 Afghanistan
##  9 <NA>           Afghanistan    2020-03-03     5      0   38928341 Afghanistan
## 10 <NA>           Afghanistan    2020-03-04     5      0   38928341 Afghanistan
## # i 306,817 more rows
```

Visualizing Data

```r
# Visualizing the Data

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.
```

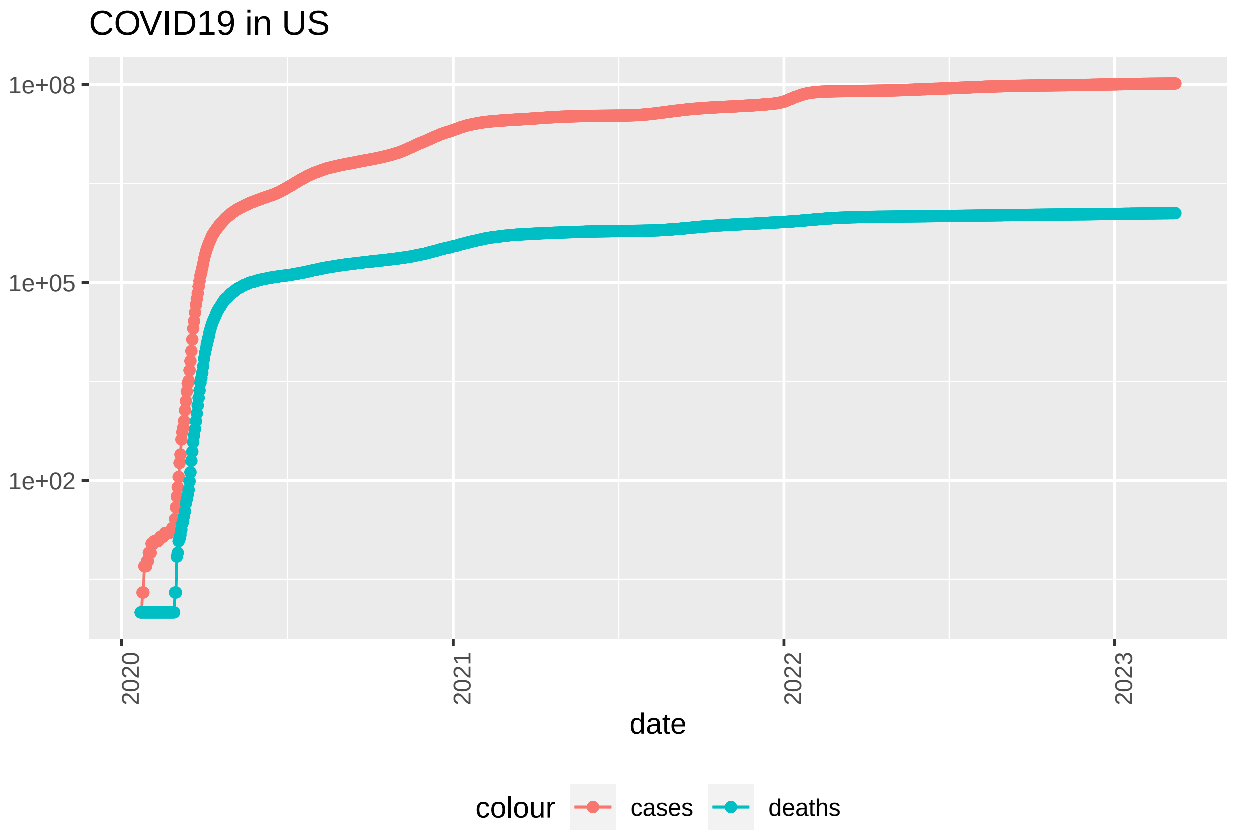
```r
tail(US_by_state)
```

```
## # A tibble: 6 x 7
##   Province_State Country_Region date        cases deaths deaths_per_mill
##   <chr>          <chr>          <date>      <dbl>  <dbl>           <dbl>
## 1 Wyoming        US             2023-03-04 185159   2002           3459.
## 2 Wyoming        US             2023-03-05 185159   2002           3459.
## 3 Wyoming        US             2023-03-06 185159   2002           3459.
## 4 Wyoming        US             2023-03-07 185385   2004           3463.
## 5 Wyoming        US             2023-03-08 185385   2004           3463.
## 6 Wyoming        US             2023-03-09 185385   2004           3463.
## # i 1 more variable: Population <dbl>
```

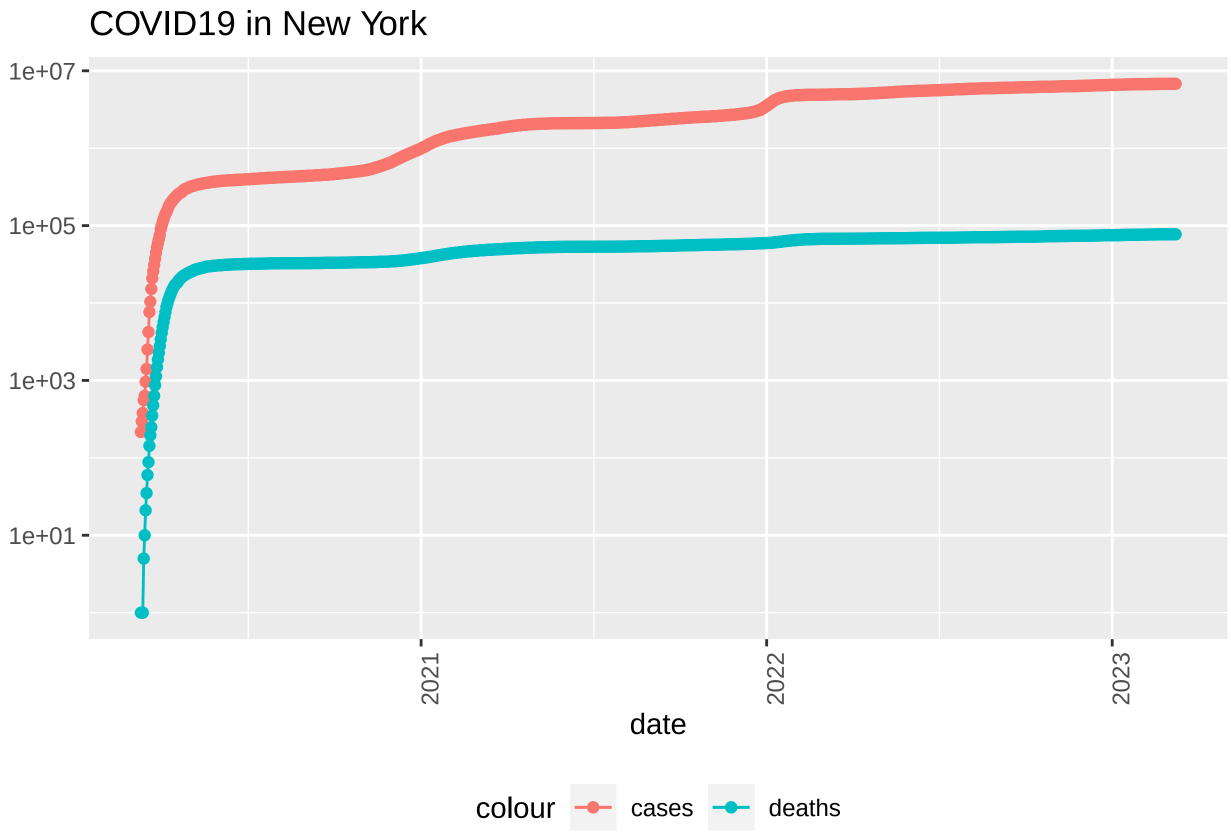
```r
US_totals <- US_by_state %>%
    group_by(Country_Region, date) %>%
    summarize(cases = sum(cases), deaths = sum(deaths),
              Population = sum(Population), .groups = "drop") %>%
    mutate(deaths_per_mill = deaths * 1000000 / Population) %>%
    select(Country_Region, date, cases, deaths, deaths_per_mill, Population) %>%
    ungroup()

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             2023-03-04 103650837 1122172          3371.  332875137
## 2 US             2023-03-05 103646975 1122134          3371.  332875137
## 3 US             2023-03-06 103655539 1122181          3371.  332875137
## 4 US             2023-03-07 103690910 1122516          3372.  332875137
## 5 US             2023-03-08 103755771 1123246          3374.  332875137
## 6 US             2023-03-09 103802702 1123836          3376.  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)
```



```
state <- "New York"
US_by_state %>%
    filter(Province_State == state) %>%
    filter(cases > 0, deaths > 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 = paste("COVID19 in", state), y = NULL)
```



```
US_state_totals <- US_by_state %>%
    group_by(Province_State) %>%
    summarise(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
```

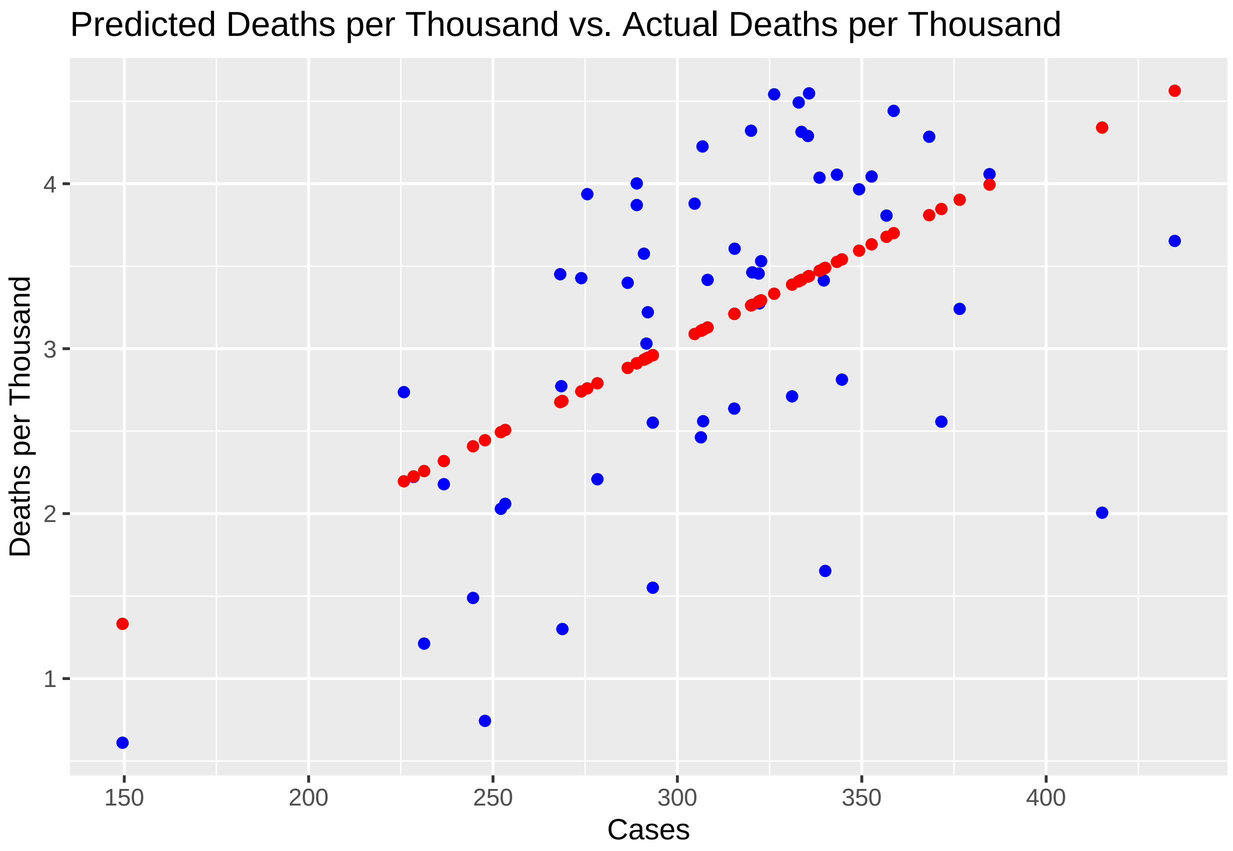
```
US_state_totals %>%
    slice_max(deaths_per_thou, n = 10)
```

```
## # A tibble: 10 x 6
##    Province_State deaths   cases population cases_per_thou deaths_per_thou
##    <chr>           <dbl>   <dbl>      <dbl>          <dbl>           <dbl>
##  1 Arizona         33102 2443514    7278717           336.            4.55
##  2 Oklahoma        17972 1290929    3956971           326.            4.54
##  3 Mississippi     13370  990756    2976149           333.            4.49
##  4 West Virginia    7960  642760    1792147           359.            4.44
##  5 New Mexico       9061  670929    2096829           320.            4.32
##  6 Arkansas        13020 1006883    3017804           334.            4.31
##  7 Alabama         21032 1644533    4903185           335.            4.29
##  8 Tennessee       29263 2515130    6829174           368.            4.28
##  9 Michigan        42205 3064125    9986857           307.            4.23
## 10 Kentucky        18130 1718471    4467673           385.            4.06
```

```
# Fit a linear regression model
mod <- lm(deaths_per_thou ~ cases_per_thou, data = US_state_totals)

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

```
## # A tibble: 56 x 7
##    Province_State  deaths   cases population cases_per_thou deaths_per_thou  pred
##    <chr>            <dbl>   <dbl>      <dbl>          <dbl>           <dbl> <dbl>
##  1 Alabama          21032 1.64e6     4903185           335.            4.29   3.44
##  2 Alaska            1486 3.08e5      740995           415.            2.01   4.34
##  3 American Samoa      34 8.32e3       55641           150.            0.611  1.33
##  4 Arizona          33102 2.44e6     7278717           336.            4.55   3.44
##  5 Arkansas         13020 1.01e6     3017804           334.            4.31   3.42
##  6 California      101159 1.21e7    39512223           307.            2.56   3.12
##  7 Colorado         14181 1.76e6     5758736           306.            2.46   3.11
##  8 Connecticut      12220 9.77e5     3565287           274.            3.43   2.74
##  9 Delaware          3324 3.31e5      973764           340.            3.41   3.49
## 10 District of Co~   1432 1.78e5      705749           252.            2.03   2.49
## # i 46 more rows
```

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

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") +
  labs(title = "Predicted Deaths per Thousand vs. Actual Deaths per Thousand", x = "Cases", y = "Deaths
```


Predicted Deaths per Thousand vs. Actual Deaths per Thousand
Deaths per Thousand
Cases