Covid Analysis

duba1910

2024-01-27

Intro The COVID-19 pandemic has undeniably reshaped the landscape of global health, leaving an indelible mark on societies worldwide. In this report, I embark on a thorough analysis of COVID-19 deaths and cases, drawing insights from data meticulously sourced from Johns Hopkins University. My objective is to unravel patterns, discern trends, and uncover potential insights within this vast dataset. Through a meticulous examination of information provided by Johns Hopkins, I aim to shed light on key metrics, regional variations, and correlations between cases and fatalities.

Step 1 Install `tidyverse` and `lubridate` which are the packages I'll need to preform my analysis

```
library(tidyverse)
library(lubridate)
```

Step 2 Load in the data

```
url_in <-
  "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_ti

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",
    "UID_ISO_FIPS_LookUp_Table.csv"
  )

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

Step 3 Import the data using `read_csv`

```
us_cases <- read_csv(urls[1])
global_cases <- read_csv(urls[2])
us_deaths <- read_csv(urls[3])
global_deaths <- read_csv(urls[4])
uid <-
  read_csv(
    "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/UID_ISO_FIPS_L
  )
```

Step 4 Pivot the global data by date. Then join the cases and deaths tables together. Lastly, join with our UID table to get population.

```
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_deaths <- global_deaths %>%
  pivot_longer(
    cols = -c(`Province/State`, `Country/Region`, Lat, Long),
    names_to = "date",
    values_to = "deaths"
  ) %>%
  select(-c(Lat, Long))

global <- global_cases %>%
  full_join (global_deaths) %>%
  rename(Country_Region = `Country/Region`,
         Province_State = `Province/State`) %>%
  mutate(date = mdy(date)) %>%
  filter(cases > 0)

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)
```

Step 5 Take a peek at the new global table to make sure it looks ok

```
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          Population        Combined_Key
##  Min.   :      0   Min.   :6.700e+01   Length:306827
##  1st Qu.:      7   1st Qu.:7.866e+05   Class :character
##  Median :    214   Median :6.948e+06   Mode  :character
```

```
##  Mean   :  14405   Mean   :2.890e+07
##  3rd Qu.:   3665   3rd Qu.:2.914e+07
##  Max.   :1123836   Max.   :1.380e+09
##                    NA's   :6729
```

```
head(global)
```

```
## # A tibble: 6 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
```

Step 6 Pivot the US data. Then join them together

```
us_cases <- us_cases %>%
  pivot_longer(
    cols = -c(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 = -c(UID:Population),
    names_to = "date",
    values_to = "deaths"
  ) %>%
  select(Admin2:deaths) %>%
  mutate(date = mdy(date)) %>%
  select(-c(Lat, Long_))

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

Step 7 Take a peek at the new US table to make sure it looks ok

```
summary(us)
```

```
##     Admin2          Province_State     Country_Region     Combined_Key
##  Length:3819906     Length:3819906     Length:3819906     Length:3819906
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
```

```
##
##
##       date                cases             Population          deaths
##  Min.   :2020-01-22   Min.   : -3073   Min.   :       0   Min.   :  -82.0
##  1st Qu.:2020-11-02   1st Qu.:   330   1st Qu.:    9917   1st Qu.:    4.0
##  Median :2021-08-15   Median :  2272   Median :   24892   Median :   37.0
##  Mean   :2021-08-15   Mean   : 14088   Mean   :   99604   Mean   :  186.9
##  3rd Qu.:2022-05-28   3rd Qu.:  8159   3rd Qu.:   64979   3rd Qu.:  122.0
##  Max.   :2023-03-09   Max.   :3710586   Max.   :10039107   Max.   :35545.0
```

```
head(us)
```

```
## # A tibble: 6 x 8
##   Admin2  Province_State Country_Region Combin~1 date       cases Popul~2 deaths
##   <chr>   <chr>          <chr>          <chr>    <date>     <dbl>   <dbl>  <dbl>
## 1 Autauga Alabama        US             Autauga~ 2020-01-22     0   55869      0
## 2 Autauga Alabama        US             Autauga~ 2020-01-23     0   55869      0
## 3 Autauga Alabama        US             Autauga~ 2020-01-24     0   55869      0
## 4 Autauga Alabama        US             Autauga~ 2020-01-25     0   55869      0
## 5 Autauga Alabama        US             Autauga~ 2020-01-26     0   55869      0
## 6 Autauga Alabama        US             Autauga~ 2020-01-27     0   55869      0
## # ... with abbreviated variable names 1: Combined_Key, 2: Population
```

Step 8 Create a US by State table to get deaths per million

```
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()
```

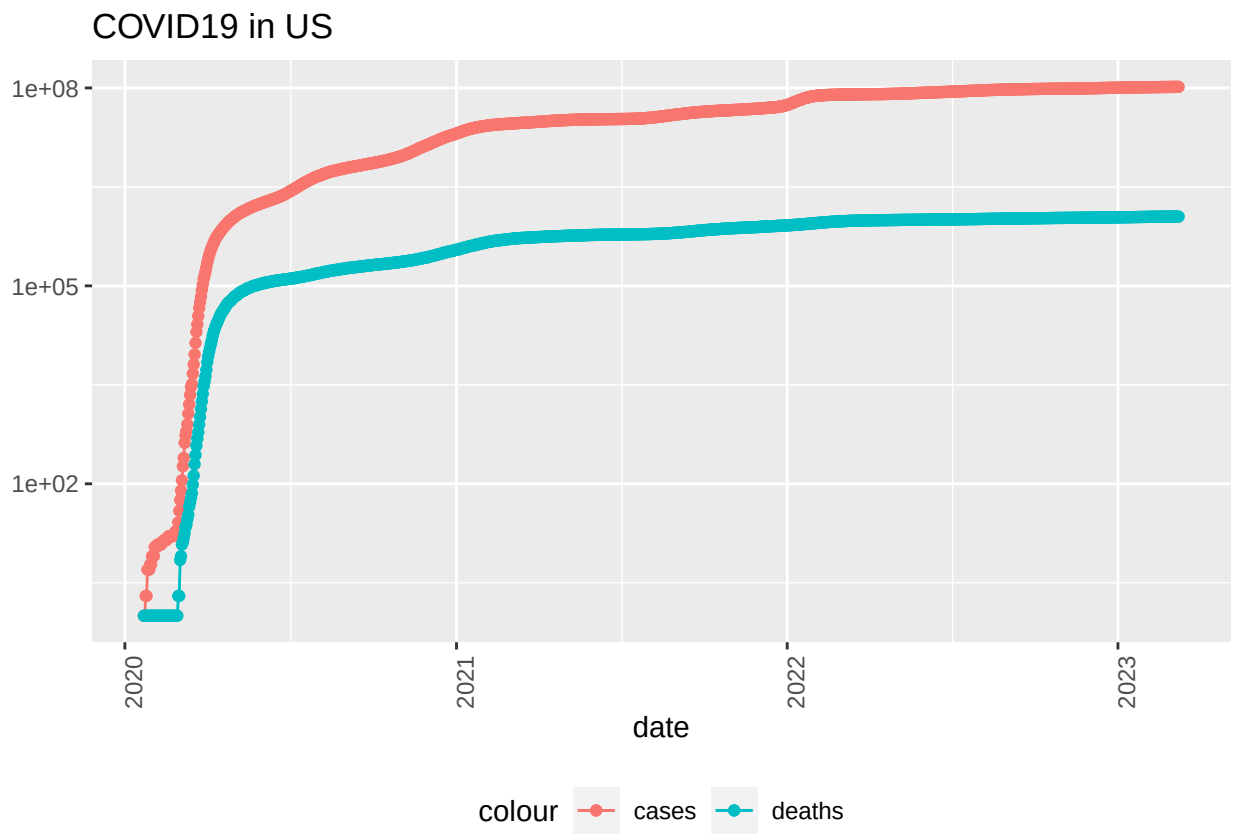
Step 9 Create a us_totals table to get deaths per million in the entire country

```
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()
```

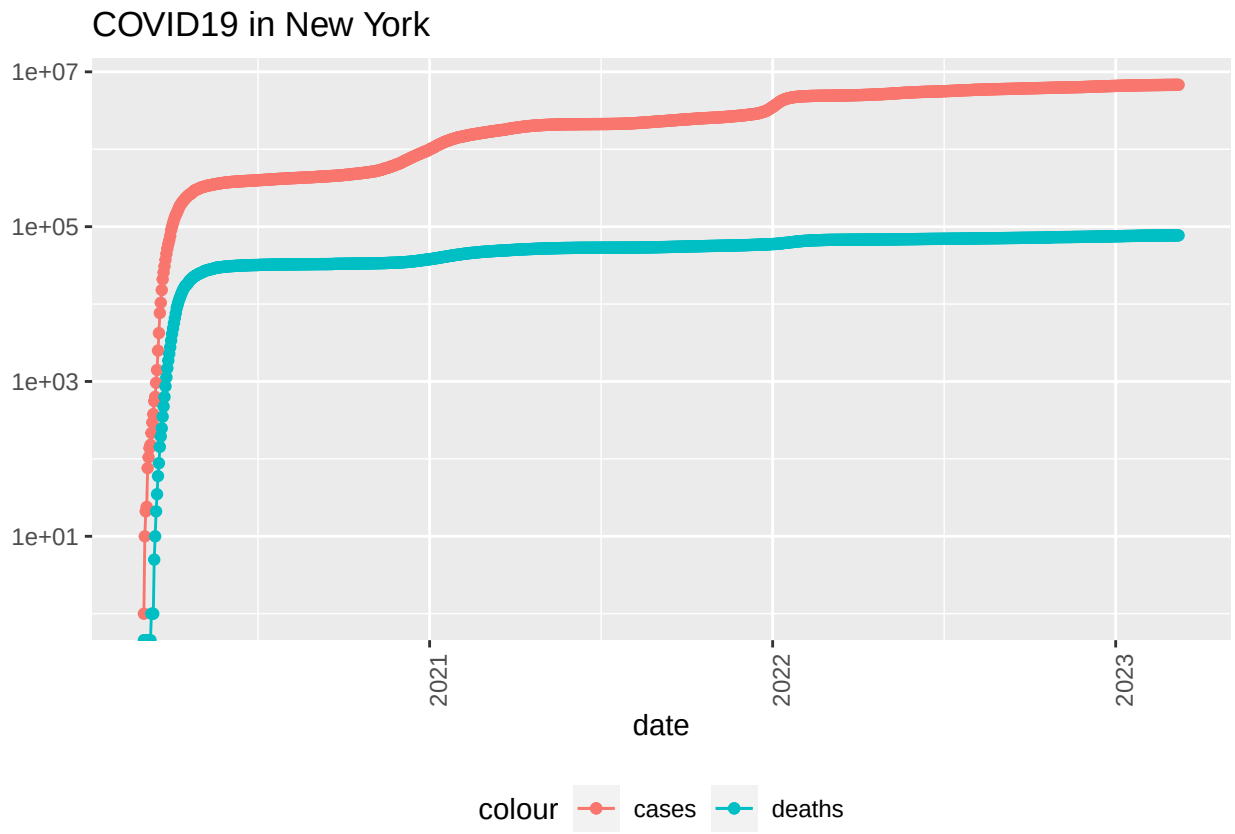
Step 10 Visualize the US totals to see how deaths and cases rose over time. We're using a log10 scale so the deaths and cases will not be too far apart. We find the deaths and cases rose very quickly through the first half of 2021, and seem to have leveled out since.

```
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)
```



Step 11 Visualize the NY state totals. Since NY is the 4th largest state by population, the graph looks very similar to the US graph. But, we do notice a pretty significant (even with the log scale) bump in cases in the beginning of 2022 which is probably due to the Omicron variant.

```
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)
```



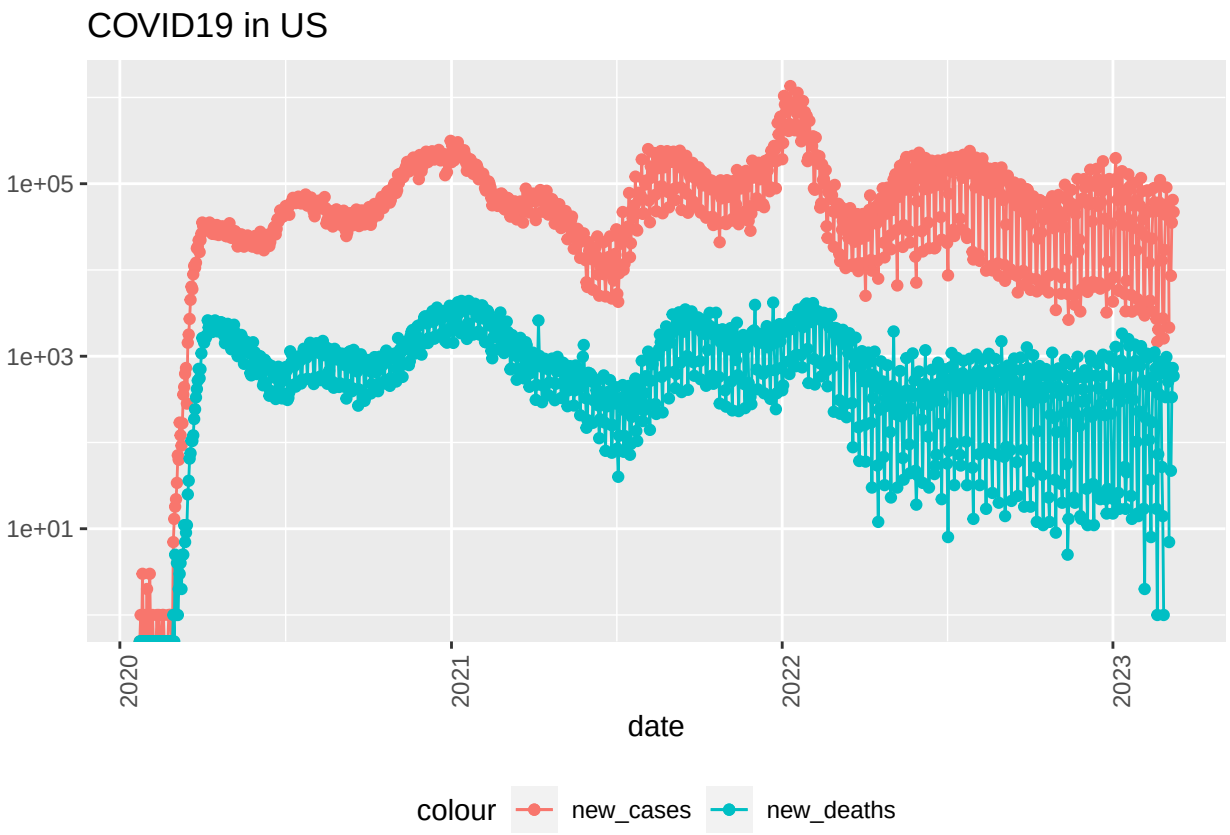
Step 12 After a while, when the cases and deaths keep increasing, it's hard to comprehend what is happening daily. We can add two new columns in our tables to show the changes in cases and deaths from the previous day.

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

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

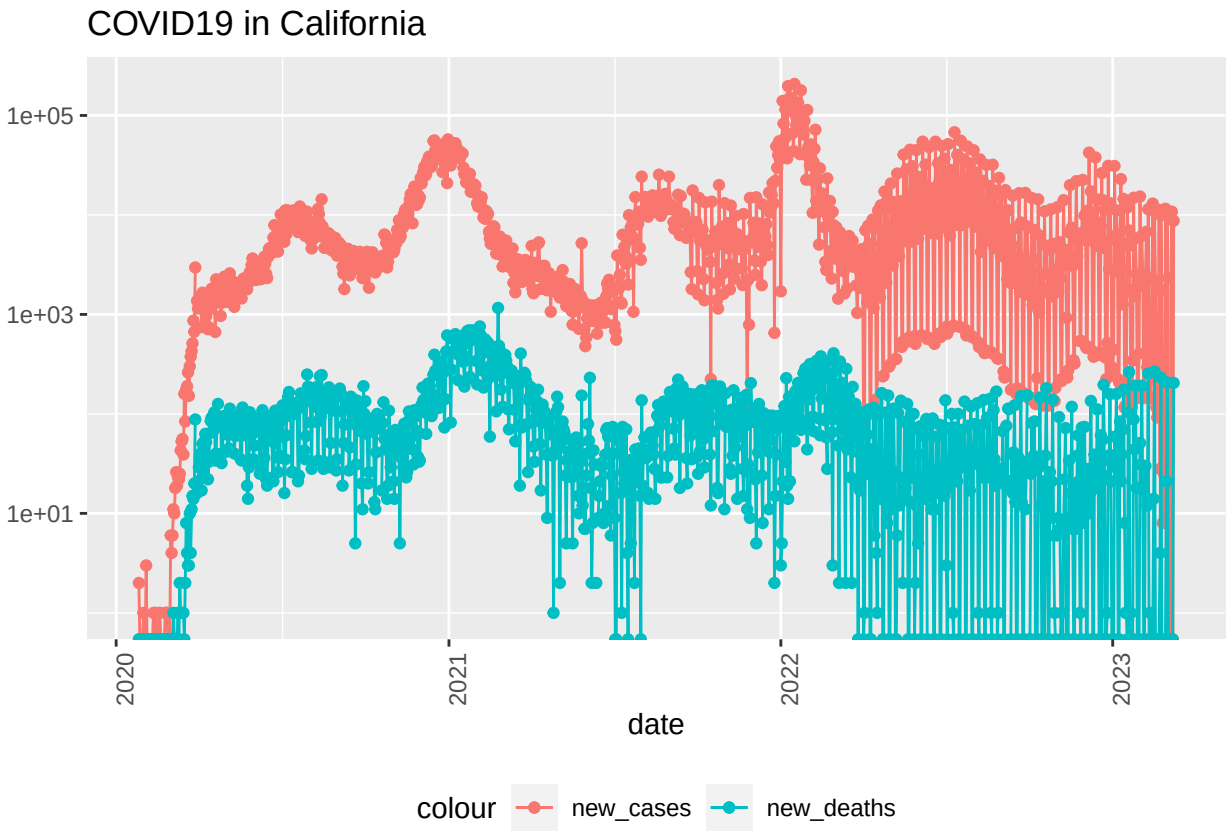
Step 13 Now we can visualize the change in cases and deaths in the US. This now clearly shows the increase in cases at the end of 2022 due to Omicron. We can also see how the Omicron variant did not cause a similar increase in deaths.

```r
us_totals %>%
  filter(cases > 0) %>%
  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)
```



Step 14 Let's now visualize the California change in cases and deaths. Since it's a single state, the changes are more prevalent. We also start to see where there are no new deaths in a given day as we get in to 2023

```r
state <- "California"
us_by_state %>%
  filter(Province_State == state) %>%
  filter(cases > 0) %>%
  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)
```



Step 15 Now we can create a us_state_totals table to see which states have the most cases and deaths overall. Then we can look at the cases and deaths per thousand people

```r
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)
```

Step 16 Just to make sure the data looks ok, we'll look at the states with the smallest deaths per thousand using `slice_min`

```
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 popul~1
##              <dbl>          <dbl> <chr>                     <dbl>  <dbl>   <dbl>
##  1           0.611           150. American Samoa               34 8.32e3   55641
##  2           0.744           248. Northern Mariana Islands     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
## # ... with abbreviated variable name 1: population
```

Step 17 We can also look at the largest states by deaths per thousand using `slice_max`

```
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
```

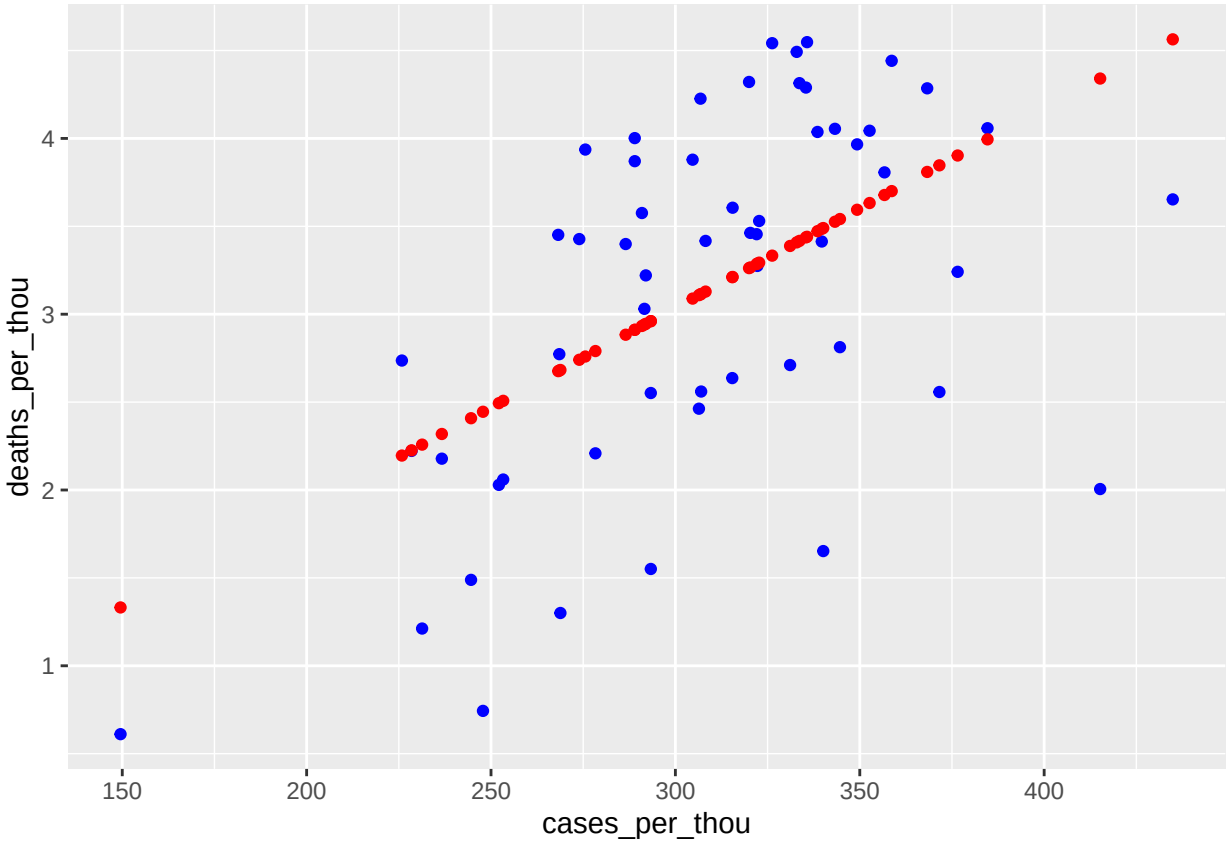
Step 18 Now we can model the data. I chose a linear model to see how the well my deaths per thousand are predicted by the cases per thousand. The P Value is super low which is great, but the rsquared is not too high.

```r
mod <- lm(deaths_per_thou ~ cases_per_thou, data = us_state_totals)
summary(mod)
```

```
##
## Call:
## lm(formula = deaths_per_thou ~ cases_per_thou, data = us_state_totals)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.3352 -0.5978  0.1491  0.6535  1.2086
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.36167    0.72480  -0.499     0.62
## cases_per_thou  0.01133    0.00232   4.881 9.76e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8615 on 54 degrees of freedom
## Multiple R-squared:  0.3061, Adjusted R-squared:  0.2933
## F-statistic: 23.82 on 1 and 54 DF,  p-value: 9.763e-06
```

Step 19 Lets graph it and see what it looks like. The blue dots (the actuals) do go up and to the right like the red dots (prediction), but there are a lot of variance and residuals. Comparing to the videos in class from early 2021, I would attribute this a lot to the prevalence of vaccinations and treatments. As cases went up early in the pandemic, deaths followed closely behind because there was no treatments and nobody was vaccinated. But now, vaccination rates are high and the treatments are readily available, so getting infected does not mean death.

```r
us_state_totals_w_pred <-
  us_state_totals %>% mutate(pred = predict(mod))
us_state_totals_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")
```



Final thoughts - bias reduction As I delve into the realm of COVID-19 data from Johns Hopkins, it is imperative to acknowledge the inherent biases that may influence my interpretations. The variables under scrutiny, such as total cases, deaths per thousand, and new cases, are not immune to personal perspectives. To address potential biases, I commit to injecting diversity into my analyses, exploring different angles, and challenging my initial assumptions. While biases are an inevitable aspect of data analysis, my commitment to awareness and a multi-faceted approach aims to mitigate their impact. By actively seeking alternative viewpoints, I strive for a more nuanced and balanced understanding of the intricate facets within the data. In the pursuit of unraveling the narrative hidden in the numbers, bias reduction becomes an integral part of my data analysis journey.s