# Covid19Data

S. Scruggs

2024-03-05

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.0     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- 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)
```

## Covid 19 Data

I will start by reading in the data from the four main csv files.

```
##Get current Data in the four files.
# They all begin the same way.
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_series_covid19_confirmed_US.csv",
               "time_series_covid19_deaths_US.csv")
urls <- str_c(url_in, file_names)
```

Let's read in the data and see what we have.

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

After looking at global_cases and global_deaths, I would like to tidy those datasets and put each variable (data, cases, deaths) in their own column.

Also, I don't need Lat and Long for the analysis I am planning so I will get rid of those and rename Region and State to be more R friendly.

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

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

global <- global_cases  %>% full_join(global_deaths) %>% rename(Country_Region = 'Country/Region', Provi

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

global <- global %>% filter(cases>0)
#This allows us to look at cases that are positive

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 <- 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

US <- US_cases %>% full_join(US_deaths)
```

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

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

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))
```

## Visualizing

We are now going to start visualizing our data.

```
US_by_state <- US %>% group_by(Province_State, Country_Region, date) %>% summarize(cases = sum(cases), d
```

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

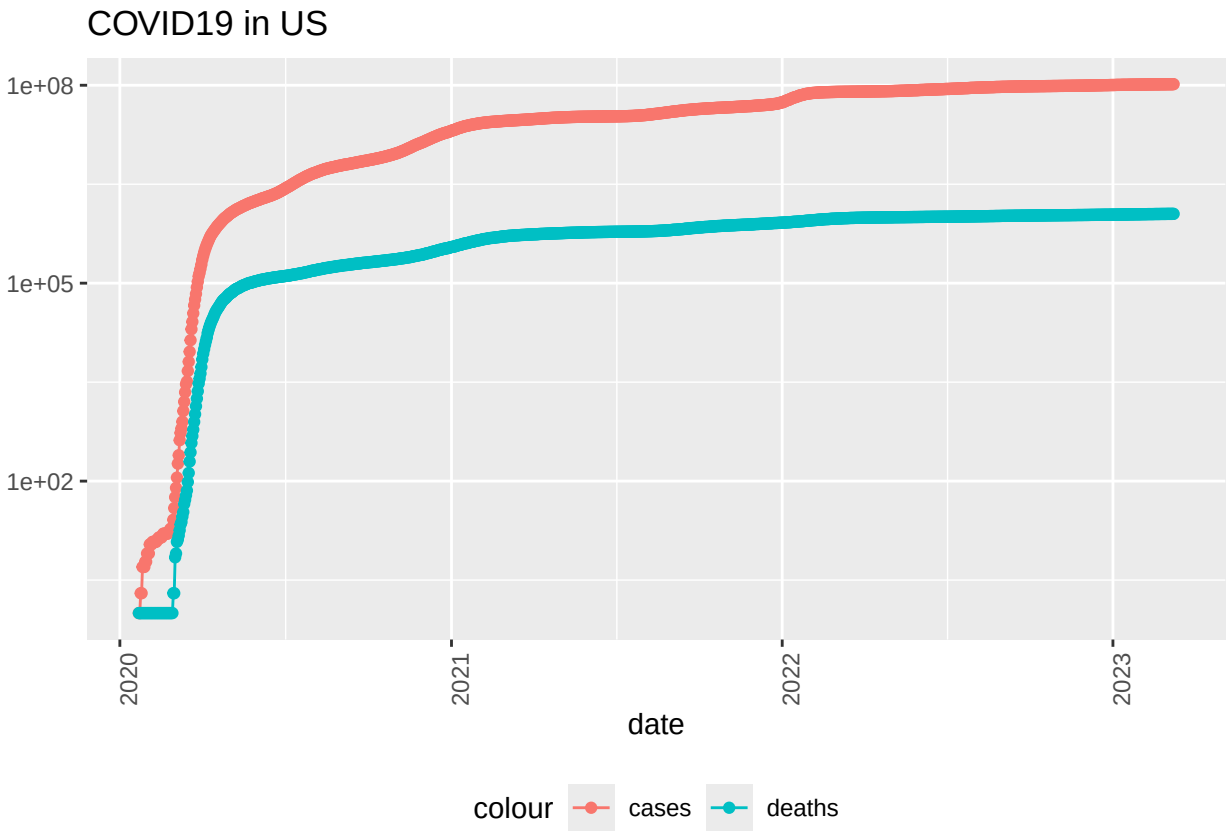
```
US_totals <- US_by_state %>% group_by(Country_Region, date) %>% summarize(cases = sum(cases), deaths = s
```

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

```
tail(US_totals)
```

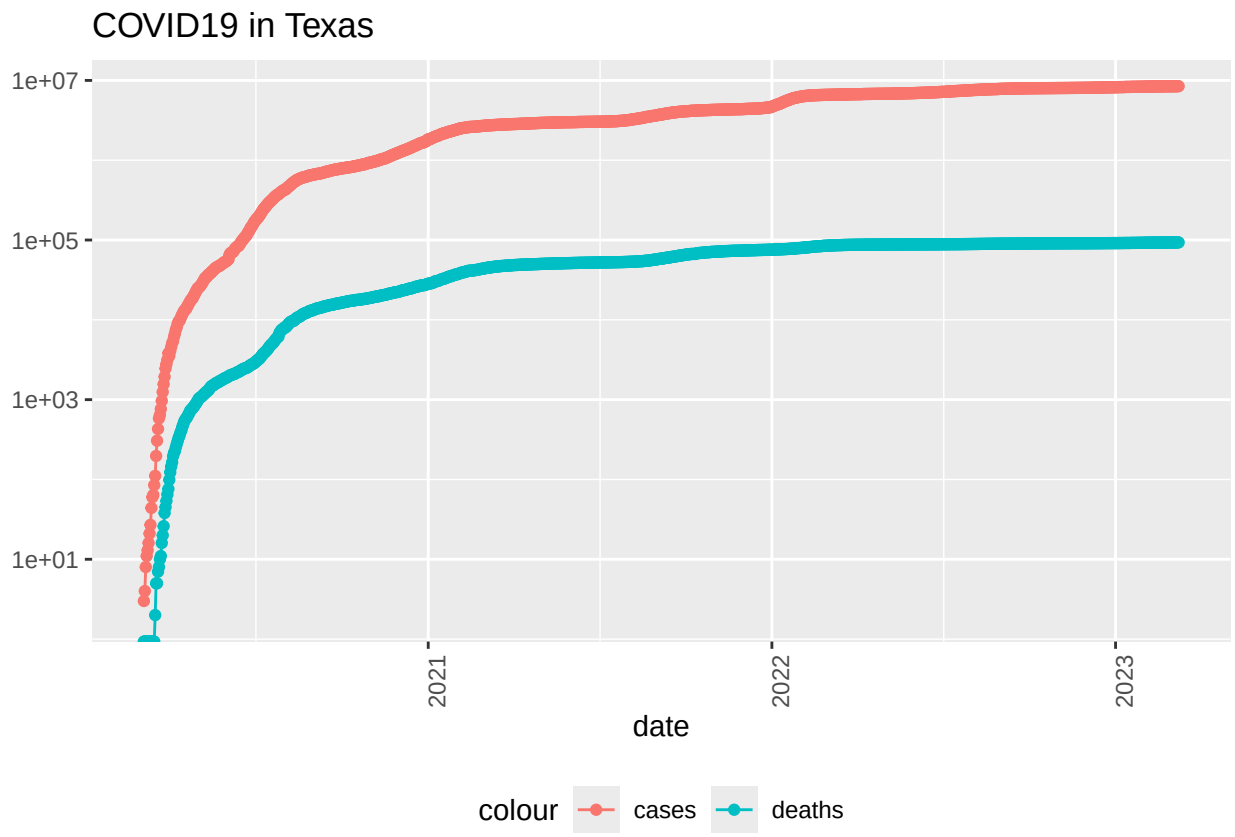
```
## # A tibble: 6 x 6
##   Country_Region date           cases  deaths deaths_per_mill Population
##   <chr>          <date>         <dbl>   <dbl>           <dbl>      <dbl>
## 1 US             2023-03-04 103650837 1122172           3371.  332875137
## 2 US             2023-03-05 103646975 1122134           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")) +
```




```
state <- "Texas"
US_by_state %>% filter(Province_State == state) %>% filter(cases > 0) %>% ggplot(aes(x = date, y = cases
```

```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
## log-10 transformation introduced infinite values.
```




```
max(US_totals$date)
```

```
## [1] "2023-03-09"
```

```
max(US_totals$deaths)
```

```
## [1] 1123836
```

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

```
## # A tibble: 6 x 8
##   new_cases new_deaths Country_Region date          cases deaths deaths_per_mill
##       <dbl>      <dbl> <chr>          <date>        <dbl>  <dbl>           <dbl>
## 1      2147          7 US             2023-03-04   1.04e8 1.12e6           3371.
## 2     -3862        -38 US             2023-03-05   1.04e8 1.12e6           3371.
## 3      8564         47 US             2023-03-06   1.04e8 1.12e6           3371.
## 4     35371        335 US             2023-03-07   1.04e8 1.12e6           3372.
## 5     64861        730 US             2023-03-08   1.04e8 1.12e6           3374.
## 6     46931        590 US             2023-03-09   1.04e8 1.12e6           3376.
## # i 1 more variable: Population <dbl>
```

```
US_totals %>% filter(cases > 0) %>% ggplot(aes(x = date, y = new_cases)) + geom_line(aes(color = "new_ca
```

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

```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
```

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

```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
```

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

```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
```

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

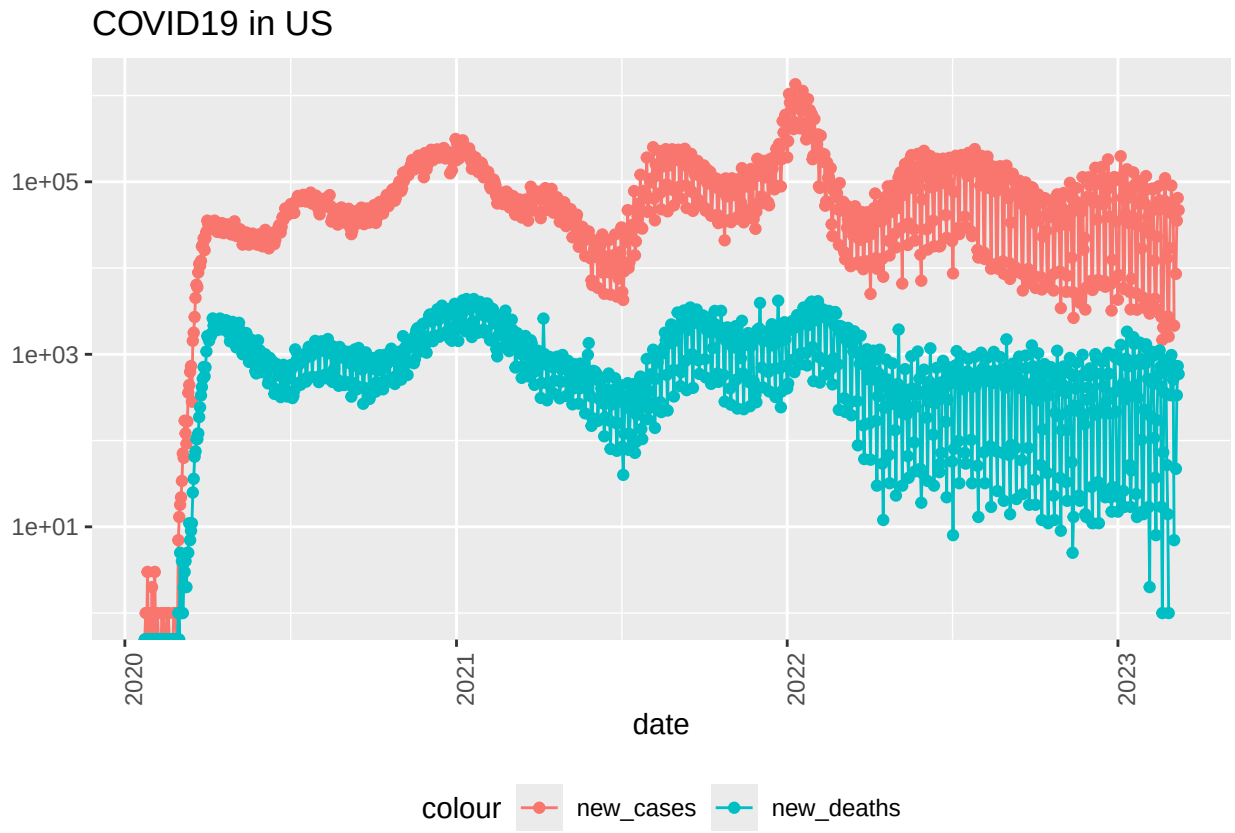
```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_line()`).
```

```
## Warning: Removed 2 rows containing missing values or values outside the scale range
## (`geom_point()`).
```

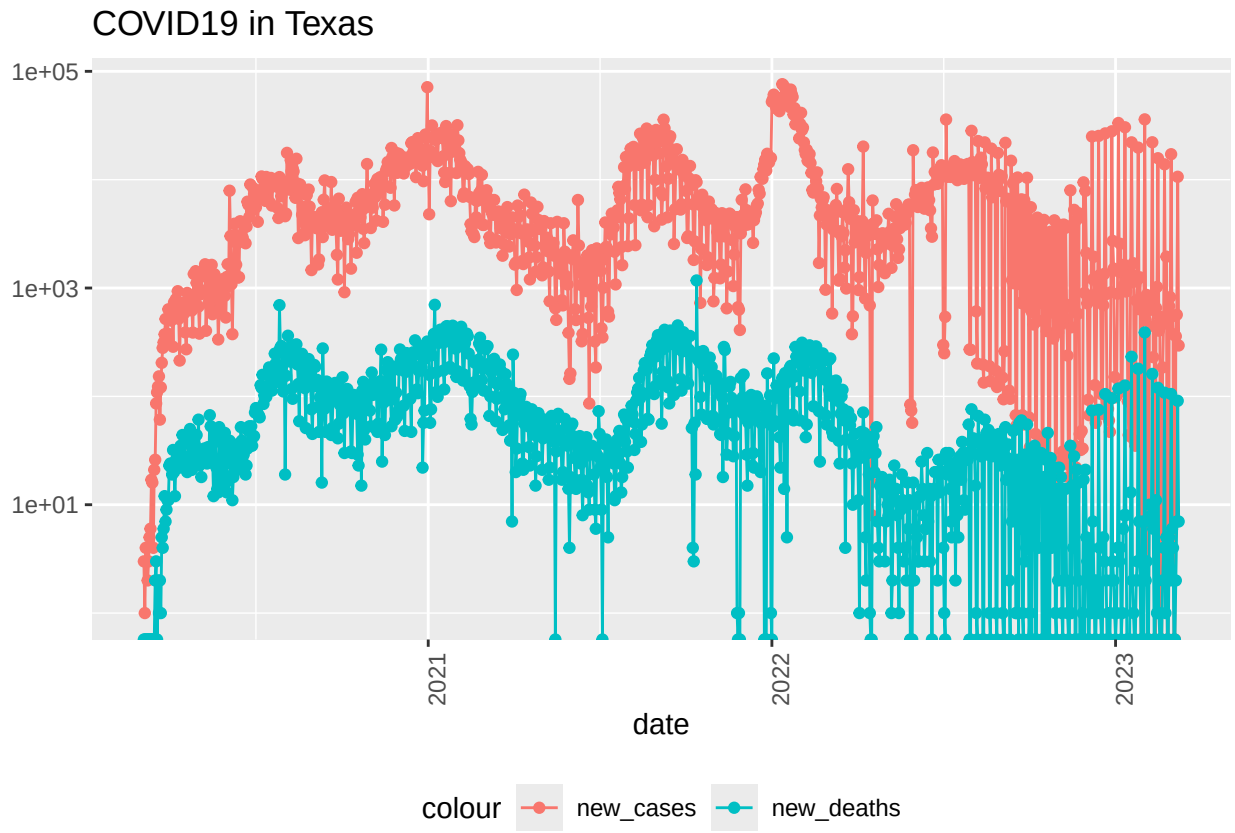
```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_line()`).
```

```
## Warning: Removed 4 rows containing missing values or values outside the scale range
## (`geom_point()`).
```




```
state <- "Texas"
US_by_state %>% filter(Province_State == state) %>% filter(cases > 0) %>% ggplot(aes(x = date, y = new_
```

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

## Warning in scale_y_log10(): log-10 transformation introduced infinite values.

## Warning in transformation$transform(x): NaNs produced

## Warning in scale_y_log10(): log-10 transformation introduced infinite values.

## Warning in transformation$transform(x): NaNs produced

## Warning in scale_y_log10(): log-10 transformation introduced infinite values.

## Warning in transformation$transform(x): NaNs produced

## Warning in scale_y_log10(): log-10 transformation introduced infinite values.

## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_point()`).

## Warning: Removed 3 rows containing missing values or values outside the scale range
## (`geom_point()`).
```




```
US_state_totals <- US_by_state %>%group_by(Province_State) %>% summarize(deaths = max(deaths), cases = m

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

```
## # 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, every
```

```
## # 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
```

## Modeling Data

We are now going to model our data.

```
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
```

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

```
## # A tibble: 1 x 6
##   Province_State deaths cases population cases_per_thou deaths_per_thou
##   <chr>           <dbl> <dbl>      <dbl>          <dbl>           <dbl>
## 1 American Samoa     34  8320      55641           150.           0.611
```

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

```
## # A tibble: 1 x 6
##   Province_State deaths  cases population cases_per_thou deaths_per_thou
##   <chr>           <dbl>  <dbl>      <dbl>          <dbl>           <dbl>
## 1 Rhode Island     3870 460697    1059361           435.            3.65
```

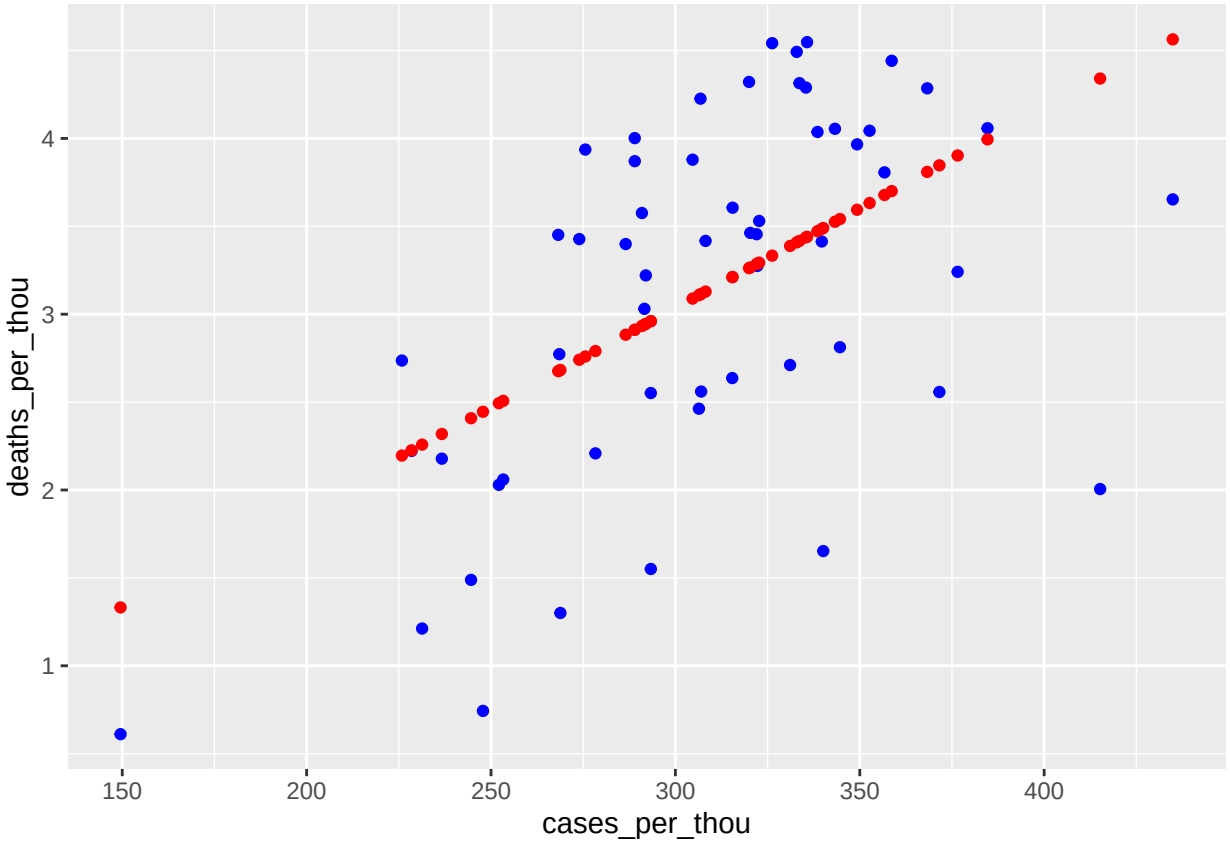
```
#x_grid <- seq(1,151)

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

US_total_w_pred
```

```
## # 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_total_w_pred %>% ggplot() + geom_point(aes(x = cases_per_thou, y = deaths_per_thou), color = "blue")
```




## Bias

It is important to always understand bias in data science, so we must understand where bias could come from here. Many people perhaps did not believe in COVID so they did not report it. This could drastically change our number of cases. Personal bias could also change the data and how we clean/organize it.