Covid

David

2025-03-02

```
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   3.5.1     v tibble    3.2.1
## v lubridate 1.9.4     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

```
url_in <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/refs/heads/master/csse_covid_19_da

file_names <- c("time_series_covid19_confirmed_US.csv", "time_series_covid19_confirmed_global.csv", "ti
urls <- str_c(url_in, file_names)
```

```
global_cases <- read.csv(urls[2])
global_deaths <- read.csv(urls[4])
us_cases <- read.csv(urls[1])
us_deaths <- read.csv(urls[3])
global_recovered <- read.csv(urls[5])
```

Cleaning Data

```
#dropping Lat,Long columns, pivoting so dates are rows not columns
global_cases <- global_cases %>%
  pivot_longer(cols=
                 -c('Province.State', Lat, Long, 'Country.Region'),
               names_to = "date",
               values_to = "cases") %>%
```

```
  select(-c(Lat,Long))
#all dates start with the chr X need to remove that
global_cases$date <- global_cases$date %>%
  str_remove("X")

#same for global deaths
global_deaths <- global_deaths %>%
  pivot_longer(cols=
                 -c('Province.State', Lat, Long, 'Country.Region'),
               names_to = "date",
               values_to = "deaths") %>%
  select(-c(Lat,Long))

global_deaths$date <- global_deaths$date %>%
  str_remove("X")

#global recovered
global_recovered <- global_recovered %>%
  pivot_longer(cols=
                 -c('Province.State', Lat, Long, 'Country.Region'),
               names_to = "date",
               values_to = "cases") %>%
  select(-c(Lat,Long))

global_recovered$date <- global_recovered$date %>%
  str_remove("X")
#joining global deaths and cases
global <- global_cases %>%
  full_join(global_deaths) %>%
  rename(Country_Region = 'Country.Region',
         Province_State = 'Province.State') %>%
  mutate(date = mdy(date))
```

```
## Joining with `by = join_by(Province.State, Country.Region, date)`
```

```
global <- global %>% filter(cases>0)
global <- global %>%
  unite("Combined_Key",
        c(Province_State, Country_Region),
        sep = ", ",
        na.rm = TRUE,
        remove = FALSE)
uid_lookup_url <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/refs/heads/master/csse_cov
uid <- read.csv(uid_lookup_url) %>%
  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)
```

```
us_cases <- us_cases %>%
  pivot_longer(cols = -(UID:Combined_Key),
```

```
               names_to = "date",
               values_to ="cases")
us_cases$date <- us_cases$date %>%
  str_remove("X")

us_cases <- us_cases %>%
  select(Admin2:cases) %>%
  mutate(date = mdy(date)) %>%
  select(-c(Lat,Long_))

#us_deaths
us_deaths <- us_deaths %>%
  pivot_longer(cols = -(UID:Population),
               names_to = "date",
               values_to ="deaths")
us_deaths$date <- us_deaths$date %>%
  str_remove("X")

us_deaths <- us_deaths %>%
  select(Admin2:deaths) %>%
  mutate(date = mdy(date)) %>%
  select(-c(Lat,Long_))
#join
US <- us_cases %>%
  full_join(us_deaths)
```

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

```
US<- US %>% filter(cases>0)
```

```
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_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.
```

```
global_by_province <- global %>%
  group_by(Province_State, Country_Region, date) %>%
  summarize(cases = sum(cases), deaths=sum(deaths),
            Population = sum(Population)) %>%
  mutate(deaths_per_mil = deaths *1000000/Population) %>%
  select(Province_State, Country_Region, date, cases, deaths, deaths_per_mil, Population) %>%
  ungroup()
```

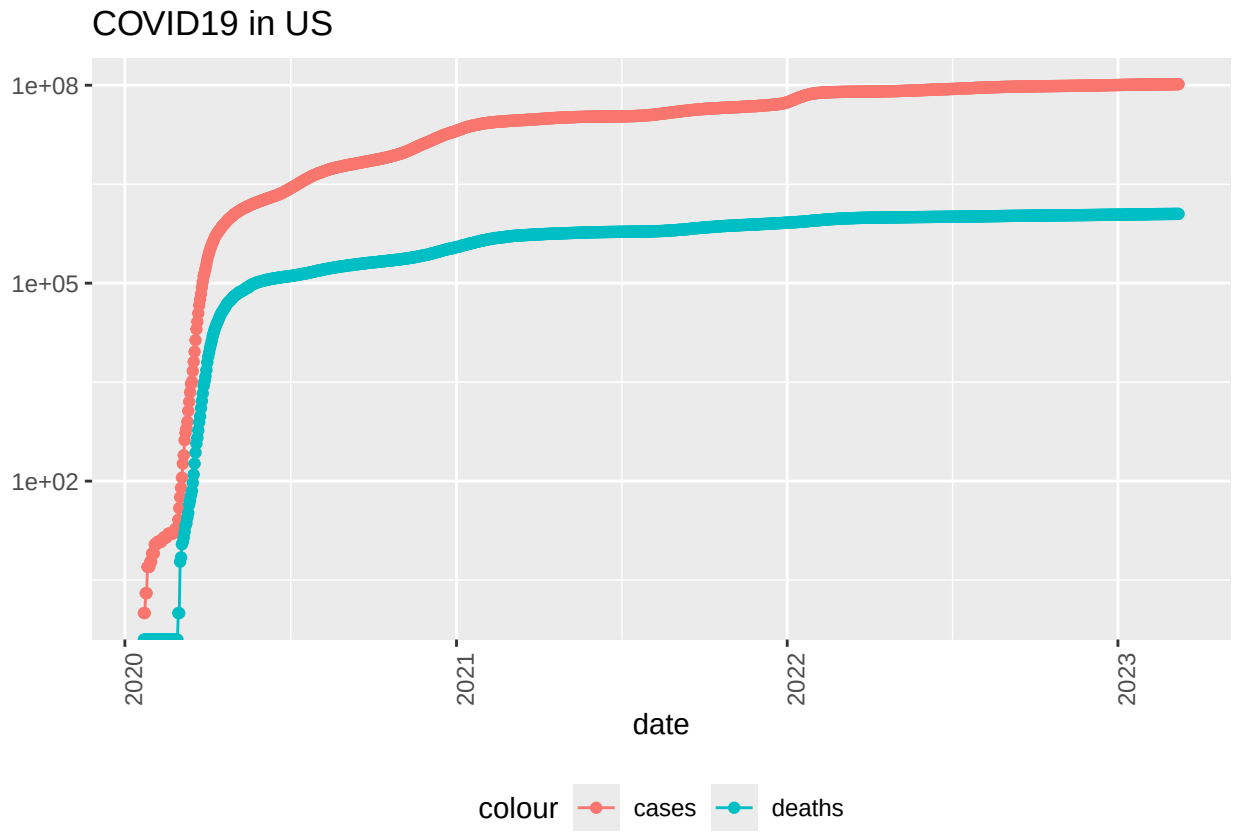
```
## `summarise()` has grouped output by 'Province_State', 'Country_Region'. You can
## override using the `.groups` argument.
```

```
global_totals <- global_by_province %>%
  group_by(Country_Region, date) %>%
  summarize(cases = sum(cases), deaths = sum(deaths),
            Population = sum(Population)) %>%
  mutate(deaths_per_mil = deaths * 1000000/Population) %>%
  select(Country_Region, date, cases, deaths, deaths_per_mil, Population) %>%
  ungroup()
```

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

```
#graph for us
US_totals_graph <- 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)
print(US_totals_graph)
```

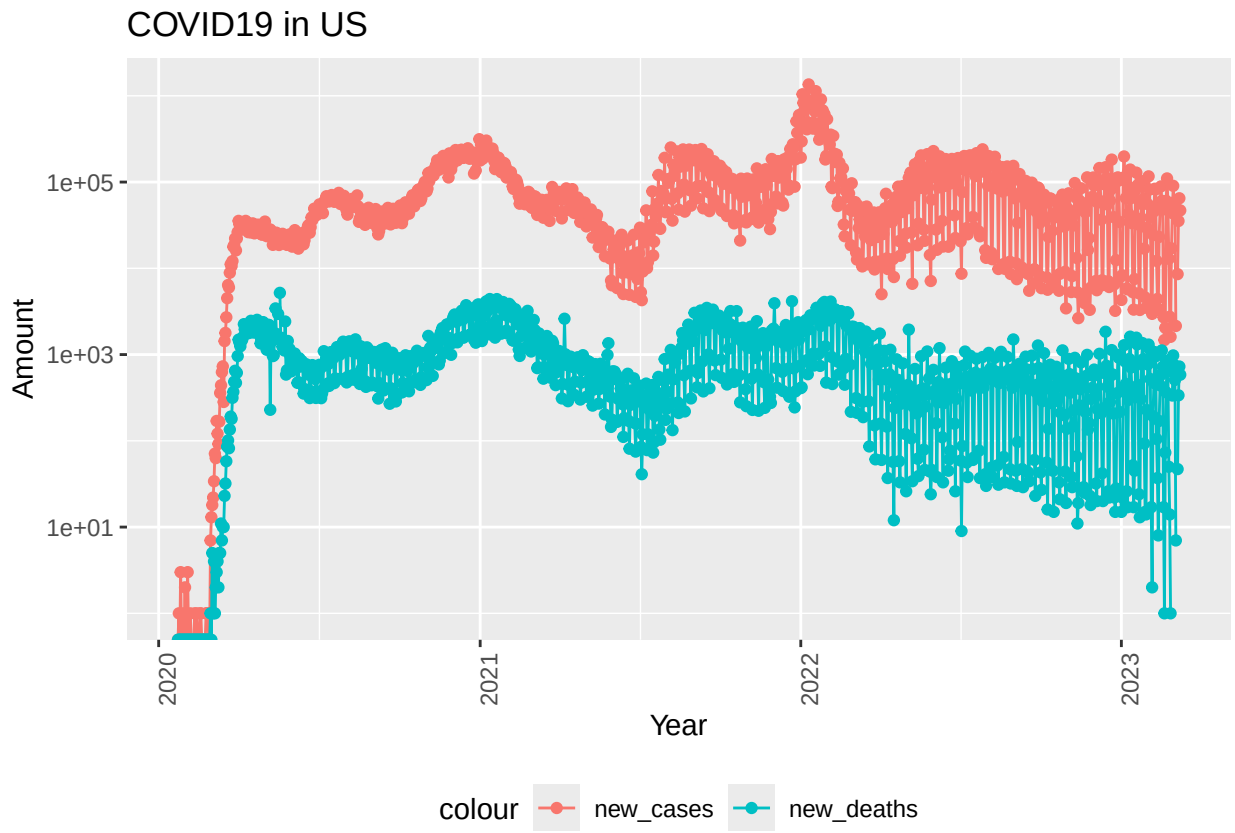
```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
## log-10 transformation introduced infinite values.
```




```
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))

US_new_totals_graph <- 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 = 'Amount', x = 'Year')
print(US_new_totals_graph)
```

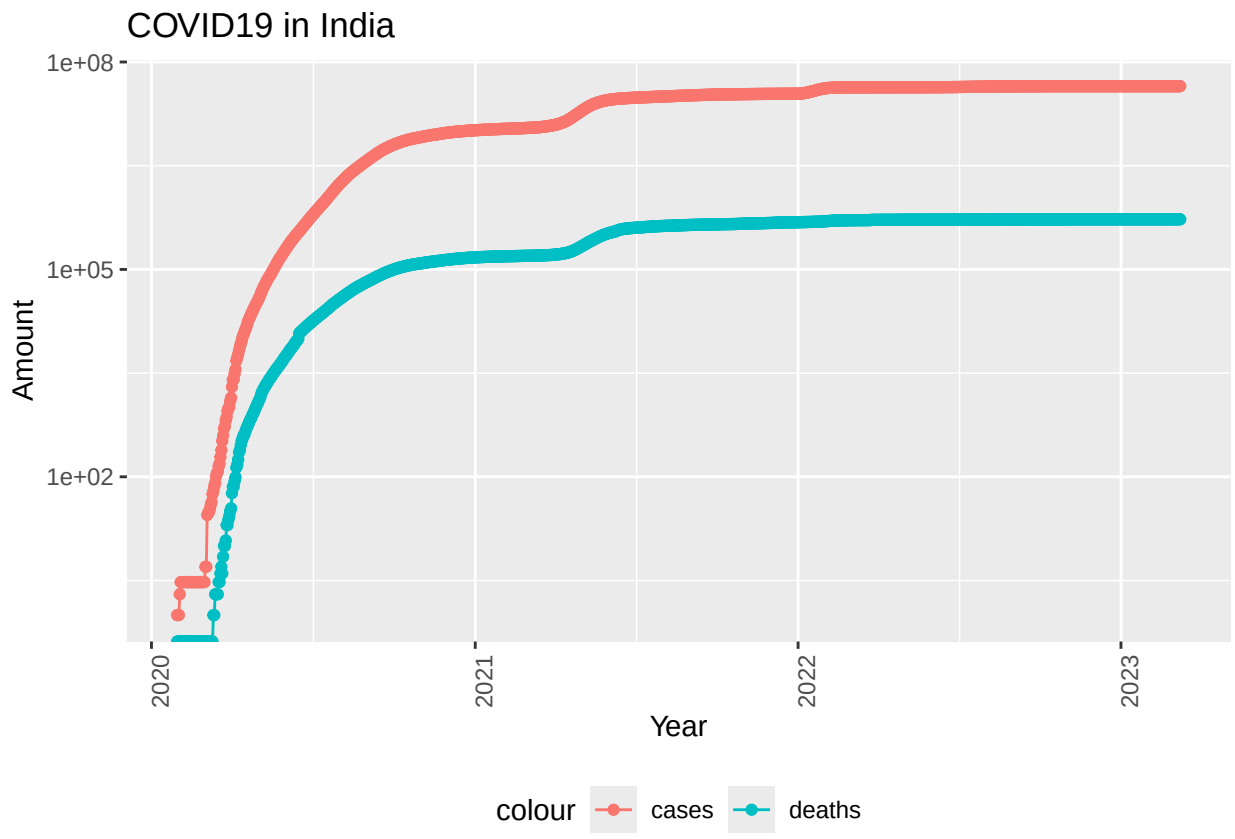
```
## Warning in transformation$transform(x): NaNs produced
## Warning in transformation$transform(x): log-10 transformation introduced
## infinite values.

## 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 7 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



```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)

#looking at india
india_graph <- global_totals %>%
  filter(cases > 0, Country_Region == c('India')) %>%
  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 India", y = "Amount", x = 'Year')
print(india_graph)
```

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

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

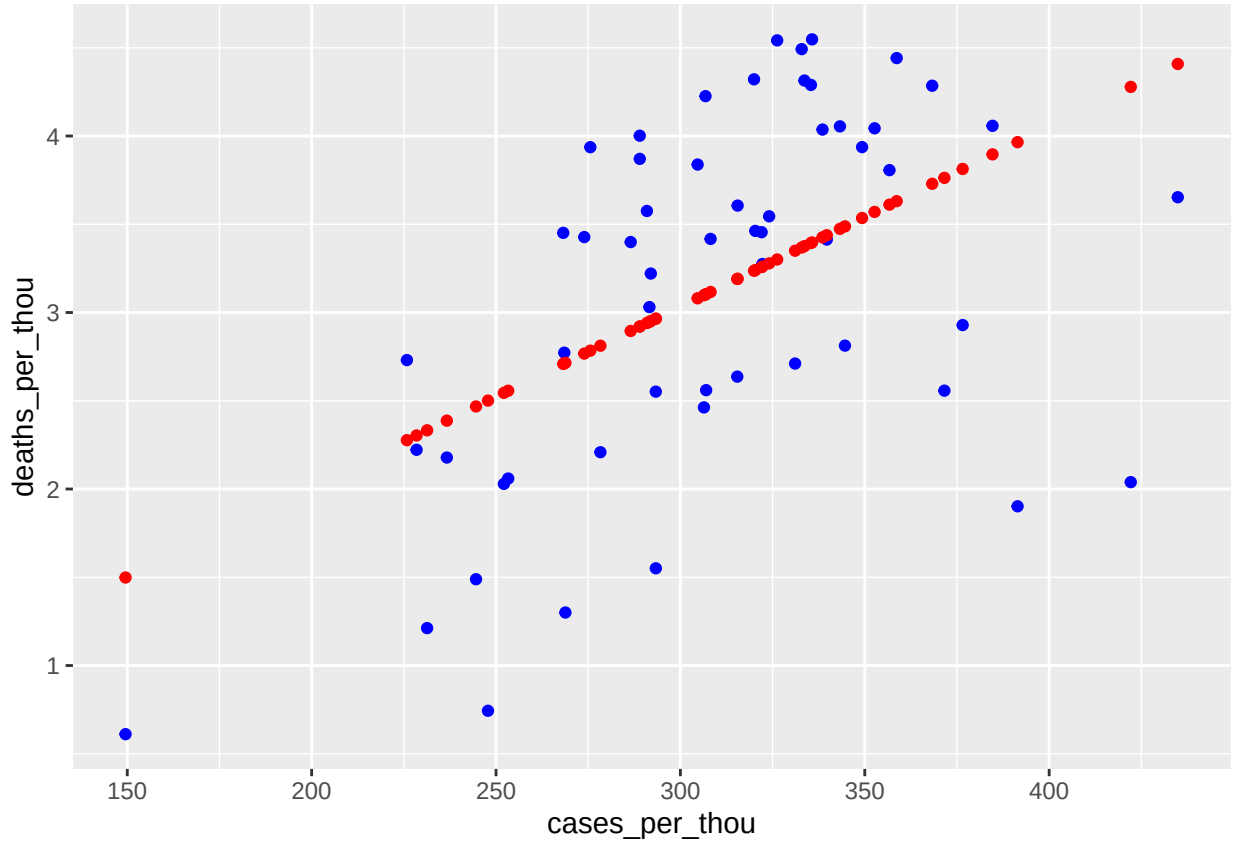


```r
#don't have good province data for india so will just look at all countries
#instead of looking at india by province
country_totals <- global_totals %>%
  filter(Country_Region != 'US') %>%
  group_by(Country_Region)%>%
  summarize(deaths = max(deaths), cases = max(cases),
            population = max(Population),
            cases_per_thou = 1000 *cases/population,
            deaths_per_thou = 1000 *deaths/population) %>%
  filter(cases > 0, population > 0)
```

```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
## -2.2394 -0.6114  0.1965  0.6413  1.2413
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.02599    0.72442  -0.036    0.972
## cases_per_thou  0.01020    0.00231   4.414 4.89e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8803 on 54 degrees of freedom
## Multiple R-squared:  0.2652, Adjusted R-squared:  0.2516
## F-statistic: 19.49 on 1 and 54 DF,  p-value: 4.894e-05
```

```r
x_grid <- seq(1,151)
new_df <- tibble(cases_per_thou=x_grid)
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), c
```

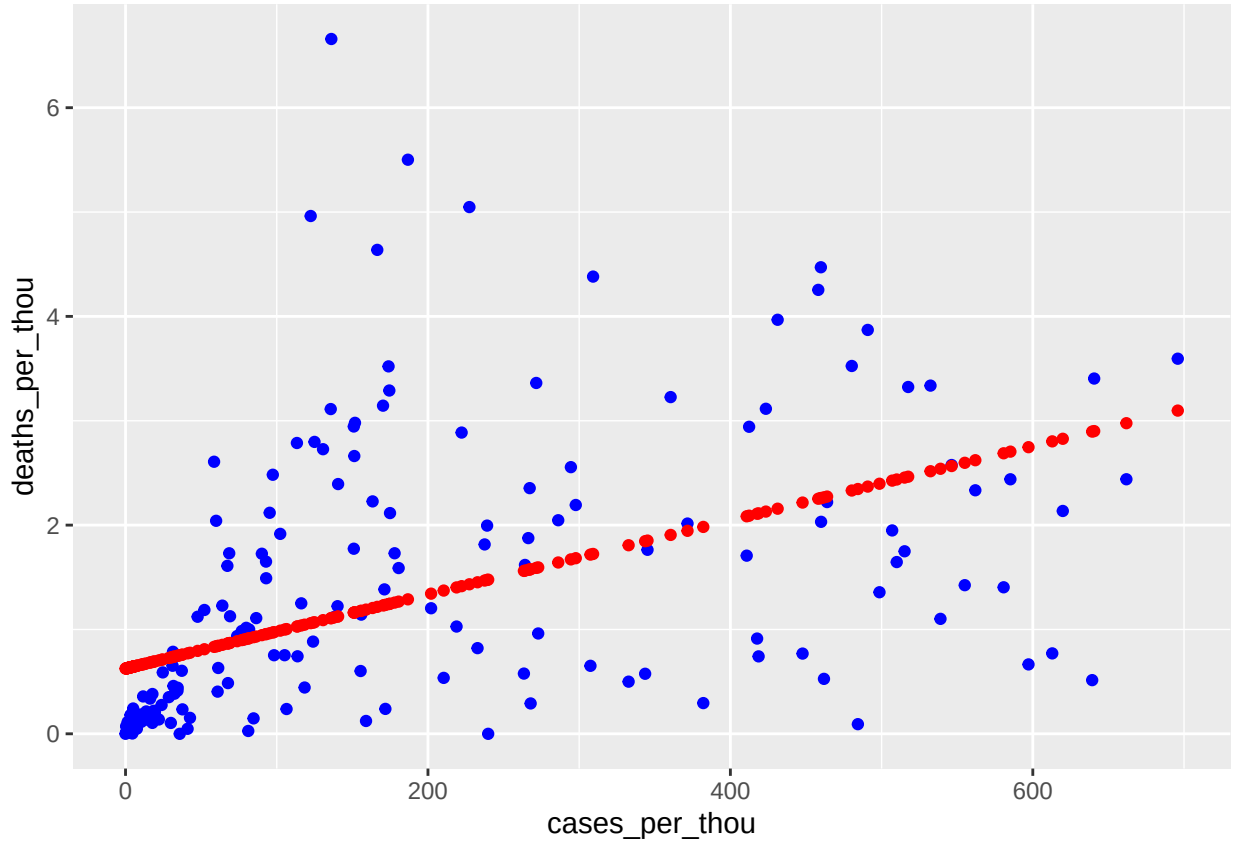



```
mod2 <- lm(deaths_per_thou ~ cases_per_thou, data= country_totals)
summary(mod)
```

```
##
## Call:
## lm(formula = deaths_per_thou ~ cases_per_thou, data = US_state_totals)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2394 -0.6114  0.1965  0.6413  1.2413
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.02599    0.72442  -0.036    0.972
## cases_per_thou  0.01020    0.00231   4.414 4.89e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8803 on 54 degrees of freedom
## Multiple R-squared:  0.2652, Adjusted R-squared:  0.2516
## F-statistic: 19.49 on 1 and 54 DF,  p-value: 4.894e-05
```

```
x_grid <- seq(1,151)
new_df <- tibble(cases_per_thou=x_grid)
Country_tot_w_pred <- country_totals %>% mutate(pred = predict(mod2))
```

```
Country_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), c
```



Conclusion

We can see that in India and the United States the largest growth in cases happened in 2020 through 2021. This is to be expected as that is when the virus first started to spread globally. We can see that there is a clear positive linear trend both in the US and globally between cases per thousand and deaths per thousand. Which is to be expected.

Possible sources of bias include inaccurate data reporting from the government agencies of each country. The COVID-19 pandemic was highly politicized and many governments did not accurately or efficiently collect and report data on the pandemic. This would lead to inaccuracies across the dataset which could skew data for certain countries.