5301 Week 3

2024-09-23

For my COVID-19 analysis, I followed along with the professor's models and graphs and added 2 of my own graphs at the end as she said that it was alright to use her models in the project instructions.

This is a data set from John Hopkins that tracked COVID cases and deaths throughout the world. It includes information about where and when each case and death occurred.

To start, I opened and read the data so that we could begin to clean it.

```
confirmed_us = read_csv('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/refs/heads/master/css
```

```
## Rows: 3342 Columns: 1154
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (6): iso2, iso3, Admin2, Province_State, Country_Region, Combined_Key
## dbl (1148): UID, code3, FIPS, Lat, Long_, 1/22/20, 1/23/20, 1/24/20, 1/25/20...
##
## 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.
```

```
confirmed_global = read_csv('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/refs/heads/maste
```

```
## Rows: 289 Columns: 1147
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (2): Province/State, Country/Region
## dbl (1145): Lat, Long, 1/22/20, 1/23/20, 1/24/20, 1/25/20, 1/26/20, 1/27/20,...
##
## 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.
```

```
deaths_us = read_csv('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/refs/heads/master/csse_
```

```
## Rows: 3342 Columns: 1155
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (6): iso2, iso3, Admin2, Province_State, Country_Region, Combined_Key
## dbl (1149): UID, code3, FIPS, Lat, Long_, Population, 1/22/20, 1/23/20, 1/24...
##
## 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.
```

```
deaths_global = read_csv('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/refs/heads/master/cs
```

```
## Rows: 289 Columns: 1147
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr    (2): Province/State, Country/Region
## dbl (1145): Lat, Long, 1/22/20, 1/23/20, 1/24/20, 1/25/20, 1/26/20, 1/27/20,...
##
## 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.
```

From here, we cleaned each set of data so that it would be easier to read and process in our analysis. We both moved some labels and data and omitted some values that we will not be looking at. By doing this, the data becomes easier to use and analysis as we do not get overwhelmed by everything all at once.

```
confirmed_global = confirmed_global %>%
  pivot_longer(cols = -c('Province/State', 'Country/Region', Lat, Long),
               names_to = "date",
               values_to = "cases") %>%
  select(-c(Lat, Long))
```

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

```
global = confirmed_global %>%
  full_join(deaths_global) %>%
  rename(Province_State = 'Province/State',
         Country_Region = 'Country/Region') %>%
  mutate(date = mdy(date))
```

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

```
global = global %>% filter(cases>0)
```

```
confirmed_us = confirmed_us %>%
  pivot_longer(cols = -(UID:Combined_Key),
               names_to = "date",
               values_to = "cases") %>%
  select(Admin2:cases) %>%
  mutate(date = mdy(date)) %>%
  select(-c(Lat, Long_))
```

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

```
us = confirmed_us %>%
  full_join(deaths_us)
```

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

We then renamed and and combined some elements in the data sets so that they would be easier to compare and combine if we wanted.

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

```
uid = read_csv('https://raw.githubusercontent.com/CSSEGISandData/COVID-19/refs/heads/master/csse_covid_
  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)) %>%
  select(Province_State, Country_Region, date, cases, deaths, Population, Combined_Key)
```

We then used the US data (both cases and deaths) to create one data frame which represents all of the information we are interested in looking at in the analysis.

```
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()
```

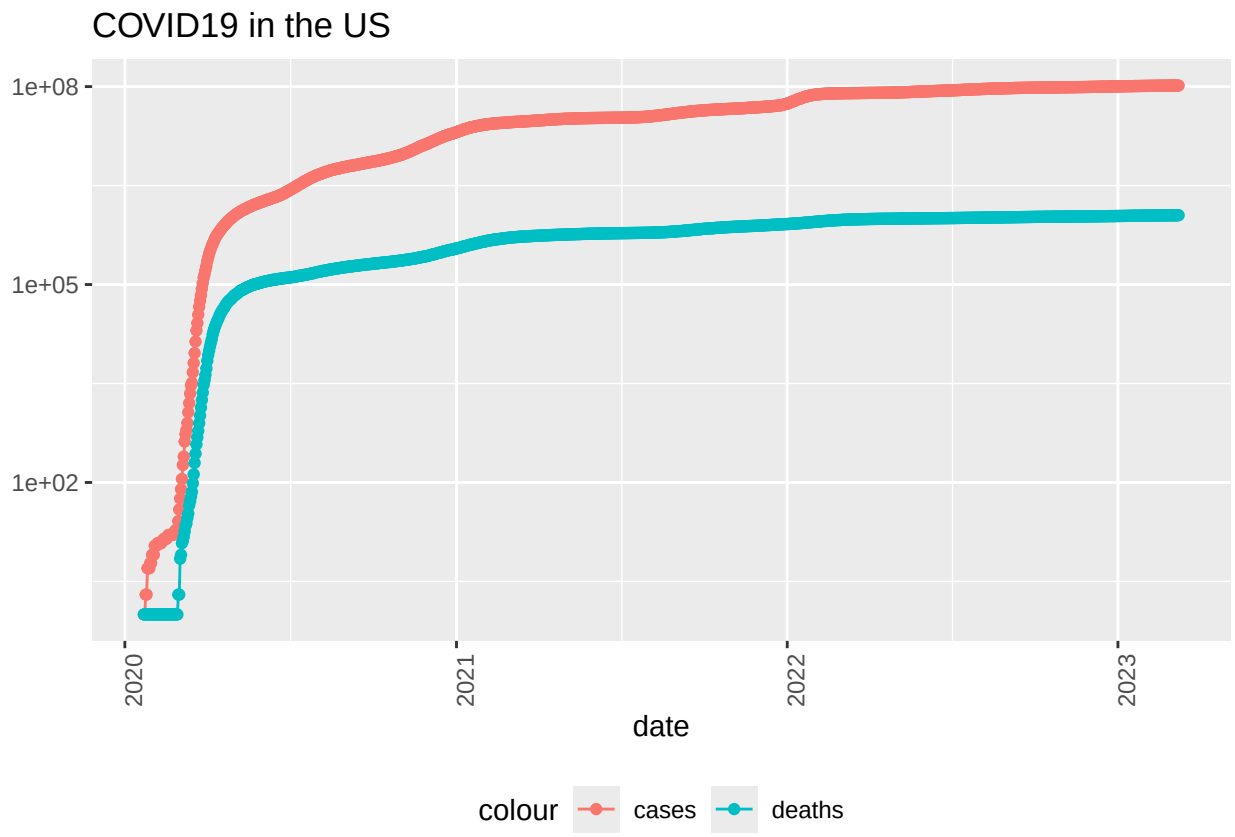
```
## `summarise()` has grouped output by 'Province_State', 'Country_Region'. You can
## override using the `.groups` argument.
```

```
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.
```

Using our combination of data sets, we can create a model of COVID-19 cases and deaths in the US. Looking at the data, which has been scaled by log10, we see a logarithmic shape. Because of this, we are able to hypothesis that COVID-19 had an exponential rate of spreading throughout the US. Furthermore, since the deaths have a similar shape, just with lower values, we can hypothesis that there is a linear relationship between cases and deaths as when cases increased, deaths followed the same progression.

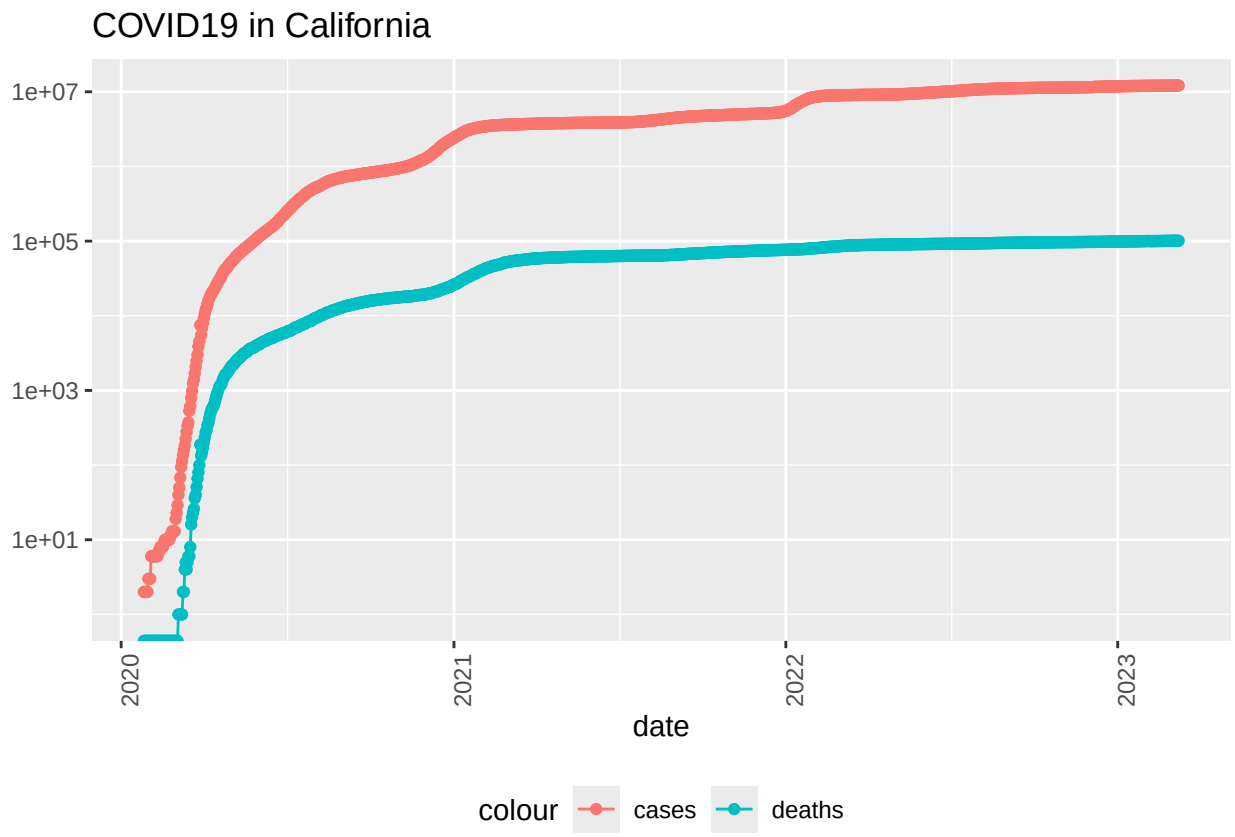
```
US_totals %>%
  filter(cases > 0) %>%
  ggplot(aes(x = date, y = cases)) +
  geom_line(aes(color = "cases")) +
  geom_point(aes(color = "cases")) +
  geom_line(aes(y = deaths, color = "deaths")) +
  geom_point(aes(y = deaths, color = "deaths")) +
  scale_y_log10() +
  theme(legend.position = "bottom", axis.text.x = element_text(angle = 90)) +
  labs(title = "COVID19 in the US", y = NULL)
```



Using the same model as before, we looked just at to see if it followed the same trends as the US population. Looking at the graphs, we can see that the shapes are similar, allowing us to hypothesize that COVID-19 spread was uniform between states and did not differ greatly in how it spread. However, we would need to look at all the states and their graphs to make this conclusion (which we did not do).

```
state = "California"
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)
```

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



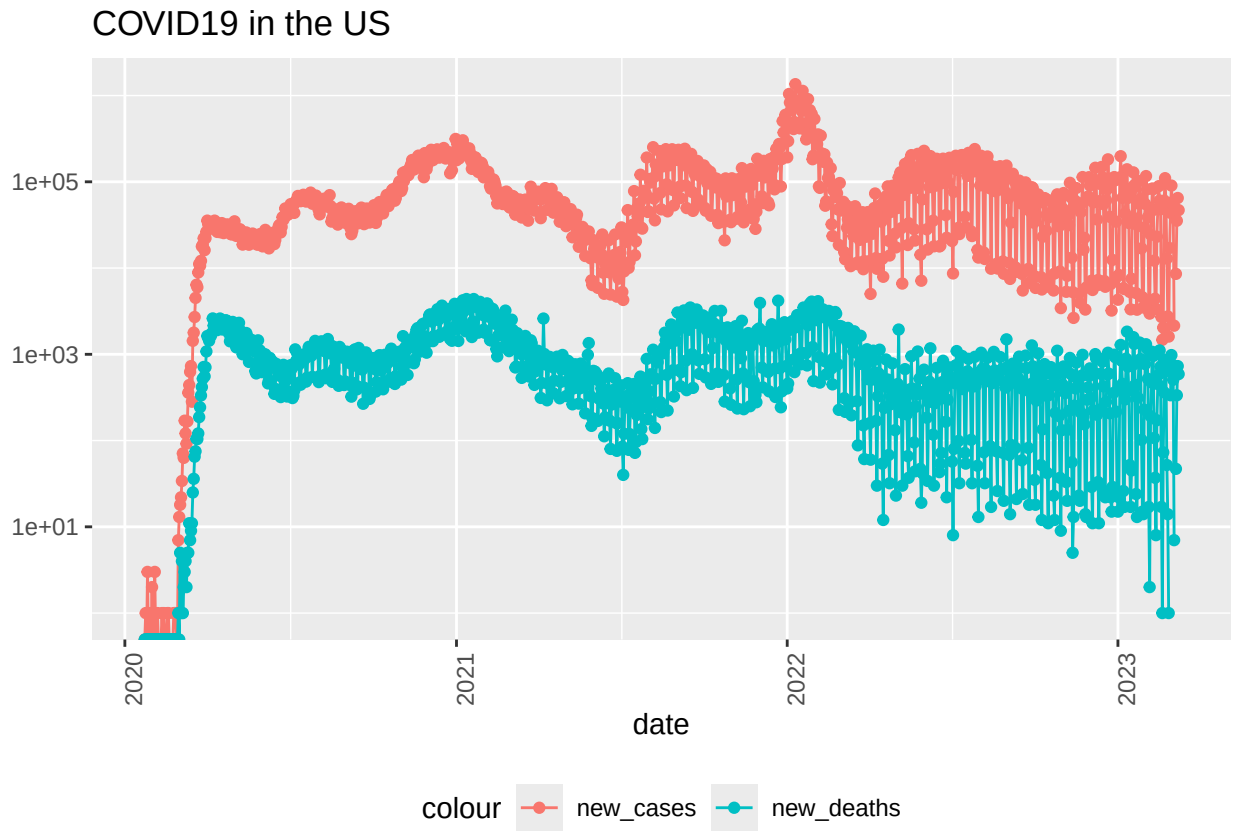

```
US_by_state = US_by_state %>%
  mutate(new_cases = cases - lag(cases),
         new_deaths = deaths - lag(deaths))
US_totals = US_totals %>%
  mutate(new_cases = cases - lag(cases),
         new_deaths = deaths - lag(deaths))
```

We then changed scale of cases and deaths by subtracting the lag (a shift of the time) to be able to look at new cases and new deaths rather than total deaths. This allowed us to see if there was changes in the

number of new deaths and new cases over time. At the beginning, we can once again see the logarithmic shape, but as time went on, the values began to fluctuate greatly, meaning that the model does not represent the data as COVID-19 continued to spread.

```
US_totals %>%
  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 the US", y = NULL)
```

```
## 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()`).
```




```
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)
```

We then looked at the number of deaths and cases per thousand, allowing us to compare the data between states without having to worry about how the difference in population might affect the number of cases or deaths.

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

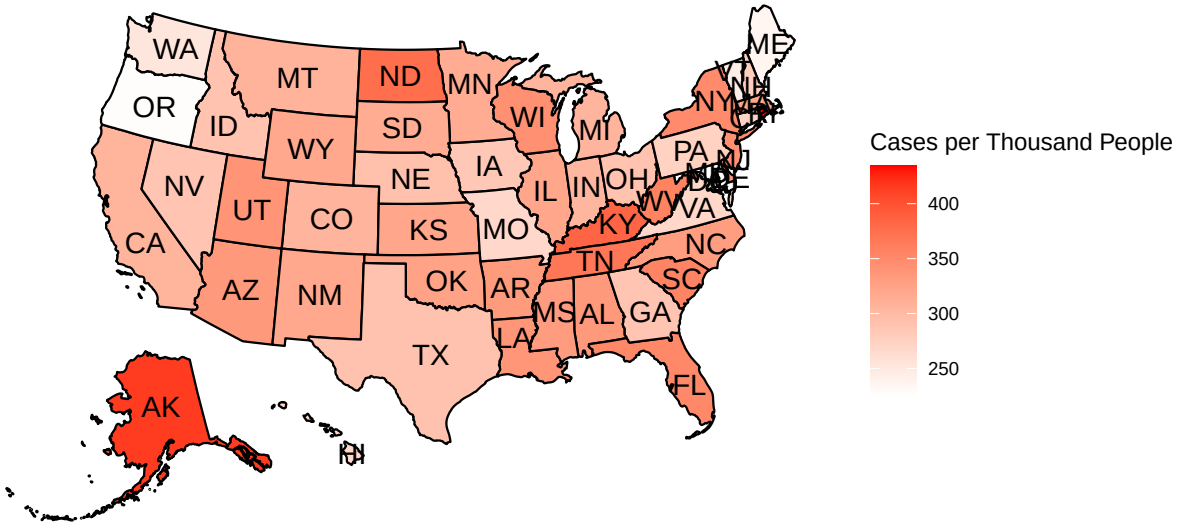
```
## # A tibble: 10 x 6
##    Province_State        deaths  cases population cases_per_thou deaths_per_thou
##    <chr>                  <dbl>  <dbl>      <dbl>          <dbl>           <dbl>
##  1 American Samoa            34 8.32e3      55641           150.           0.611
##  2 Northern Mariana 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
```

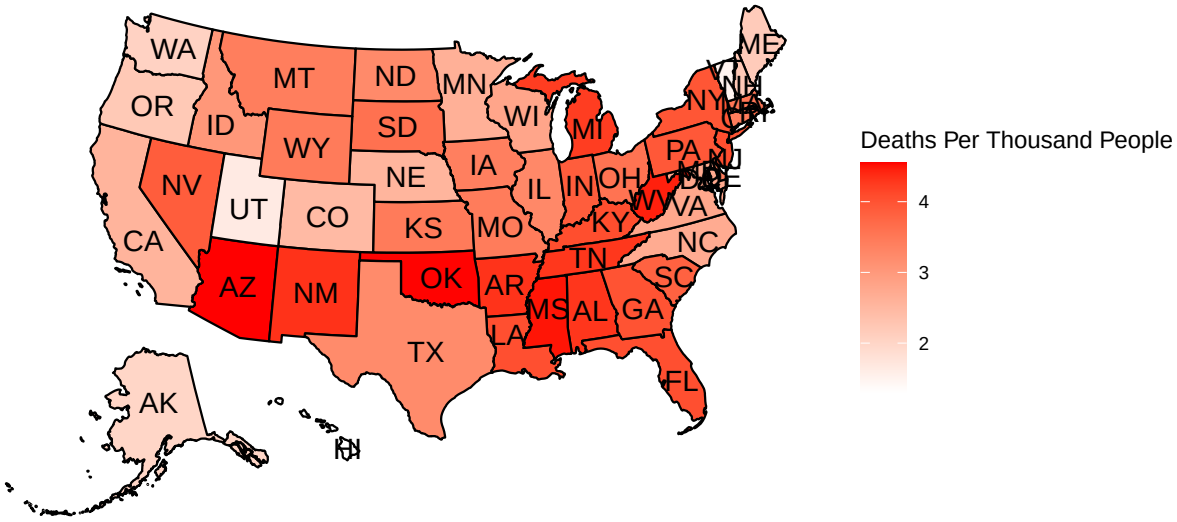
```
mod = lm(deaths_per_thou ~ cases_per_thou, data = US_state_totals)
```

Using these totals, I created a heat map of deaths and cases in each of the States. Looking at the map, we are able to see the states which had the greatest number of cases and deaths per thousand. Using this visualization, we can tell that southern states were more affected per thousand by COVID than northern and western states.

```
cases_map = data.frame(
  state = US_state_totals$Province_State,
  cases = US_state_totals$cases_per_thou
)
deaths_map = data.frame(
  state = US_state_totals$Province_State,
  deaths = US_state_totals$deaths_per_thou
)
plot_usmap(regions = "states", data = cases_map, values = "cases", labels = TRUE) +
  scale_fill_continuous(low = "white", high = "red", name = "Cases per Thousand People") +
  theme(legend.position = "right")
```



```
plot_usmap(regions = "states", data = deaths_map, values = "deaths", labels = TRUE) +
  scale_fill_continuous(low = "white", high = "red", name = "Deaths Per Thousand People") +
  theme(legend.position = "right")
```



Bias

A potential source of bias in the analysis has to do with the deaths and cases in the states. While we can somewhat normalize the data by finding the ratios of deaths and cases per thousand, we cannot say anything about what might causes the differences between the states. We would need to collect more data to understand why the differences exist.

Additionally, we applied the logarithmic function to the cases and deaths; however, we did not check how well this model fits the data. While it seems to fit, we would need to do additional analysis to ensure that this is the best model.

Conclusion

In conclusion, we can see that there is many models and graphs that can be created with this data set (and this is just the beginning). In the future, we could do more analysis with the global data and see which countries where the most affected by COVID.