Covid

B. Leblanc

2024-12-07

This is an R Markdown report on Covid 19 data across the US and globally.

Initializing Data

We need to load all necessary libraries.

```
library(tidyverse)
library(lubridate)
library(ggplot2)
```

First I will read in the data from the csv files.

```
## Get current data from files

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

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

Now lets read in the data.

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

Tidy Data

After viewing the datasets, I would like to tidy global_cases and global_deaths and put each variable (date, cases, deaths) in their own column. Also I don't need position variables (Lat, Long) for analysis so I will get rid of those and rename columns to be more R friendly.

```
# pivot datset

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

```
  values_to = "cases") %>%
select(-c(Lat, Long))

# Repeat for deaths

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_deaths <- global_deaths %>%
  rename(Province_State = 'Province/State',
         Country_Region = 'Country/Region')

global_cases <- global_cases %>%
  rename(Province_State = 'Province/State',
         Country_Region = 'Country/Region')

# Combine global_cases and global_deaths

global <- global_cases %>%
  full_join(global_deaths) %>%
  mutate(date = mdy(date))
```

Let's view the global dataset and look for any issues.

```
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
```

Many rows have 0 cases. We will filter these out to only look at dates with reported cases.

```
# Filtering
global <- global %>% filter(cases > 0)
```

Now we will look at US_cases and US_deaths and tidy the dataset. We will join these two datasets together to make US dataset.

```r
# Checking pivot
US_cases %>%
  pivot_longer(cols = -(UID:Combined_Key),
               names_to = "date",
               values_to = "cases")
```

```
## # A tibble: 3,819,906 x 13
##         UID iso2  iso3  code3  FIPS Admin2  Province_State Country_Region   Lat
##       <dbl> <chr> <chr> <dbl> <dbl> <chr>   <chr>          <chr>          <dbl>
##  1 84001001 US    USA     840  1001 Autauga Alabama        US              32.5
##  2 84001001 US    USA     840  1001 Autauga Alabama        US              32.5
##  3 84001001 US    USA     840  1001 Autauga Alabama        US              32.5
##  4 84001001 US    USA     840  1001 Autauga Alabama        US              32.5
##  5 84001001 US    USA     840  1001 Autauga Alabama        US              32.5
##  6 84001001 US    USA     840  1001 Autauga Alabama        US              32.5
##  7 84001001 US    USA     840  1001 Autauga Alabama        US              32.5
##  8 84001001 US    USA     840  1001 Autauga Alabama        US              32.5
##  9 84001001 US    USA     840  1001 Autauga Alabama        US              32.5
## 10 84001001 US    USA     840  1001 Autauga Alabama        US              32.5
## # i 3,819,896 more rows
## # i 4 more variables: Long_ <dbl>, Combined_Key <chr>, date <chr>, cases <dbl>
```

```r
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_))


# Same process with US_deaths

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

To compare the US data to the global data, we want to have a population column and a combined key column. We will add those now.

```r
# Creating Combined Key with Province_State and Country_Region
global <- global %>%
  unite("Combined_Key",
        c(Province_State, Country_Region),
        sep = ", ",
```

```
        na.rm = TRUE,
        remove = FALSE)

# Adding population data
lookup_url <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/refs/heads/master/csse_covid_19
uid <- read_csv(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)
```

At this point, we are done with tidying and organizing our data. We will move on to visualizing the data and basic analysis.

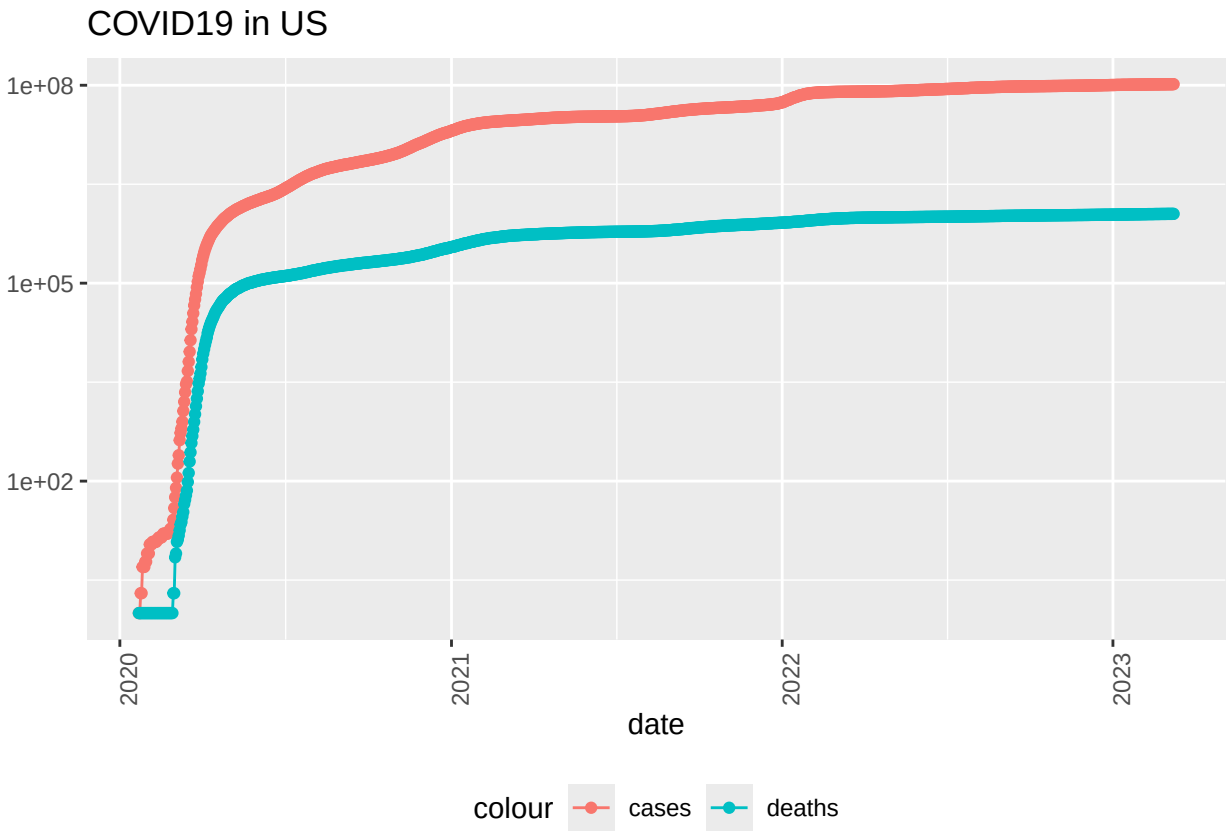
Visual Analysis

```
# Create new dataset of US by state
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()

# create US totals
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()
```

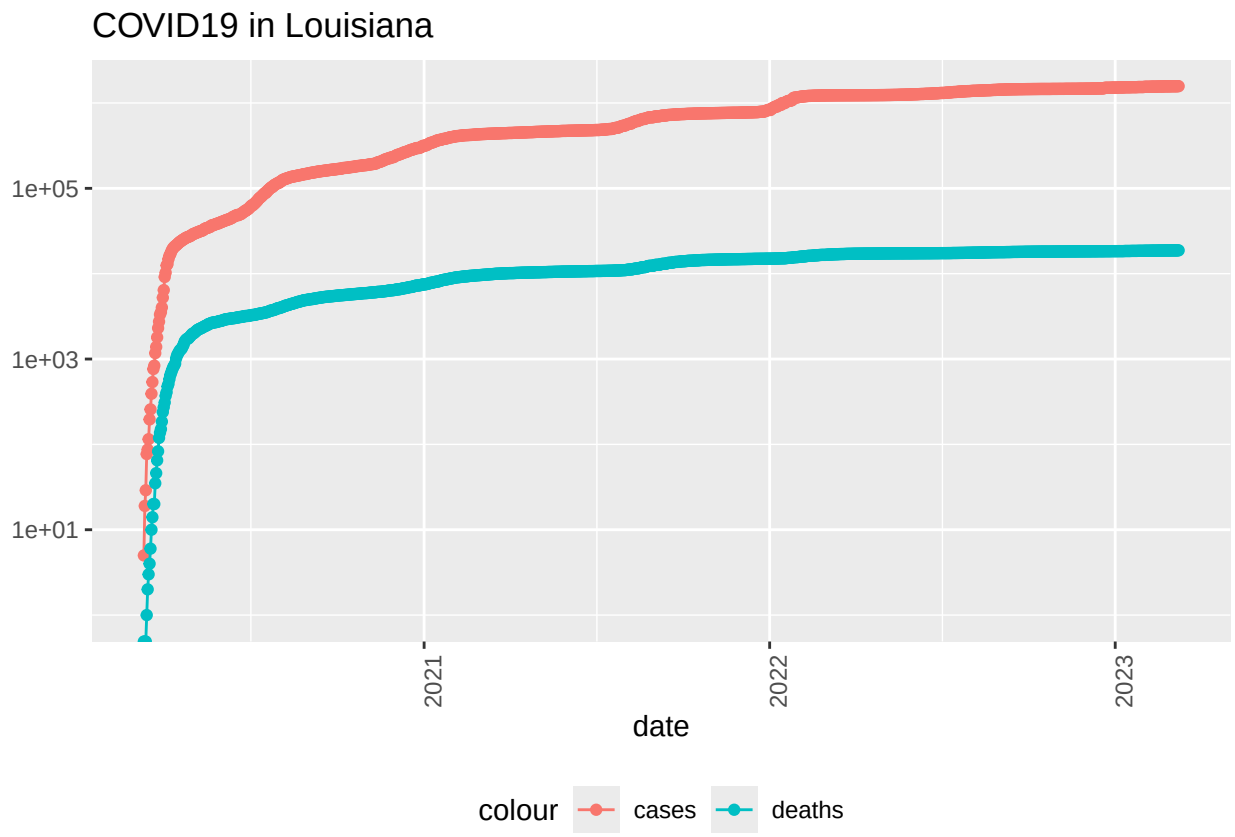
Create basic visual of US data comparing cases reported to the deaths reported.

```
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)
```



Lets narrow our search down to the state I reside in, Louisiana.

```
state <- "Louisiana"
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)
```




```
# Look at recent date
max(US_totals$date)
```

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

```
# maximum deaths
max(US_totals$deaths)
```

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

New Cases and New Deaths

After looking at the graph, I have some new questions. Has the number of cases and deaths leveled off? To answer this, we transform our data and add new variables.

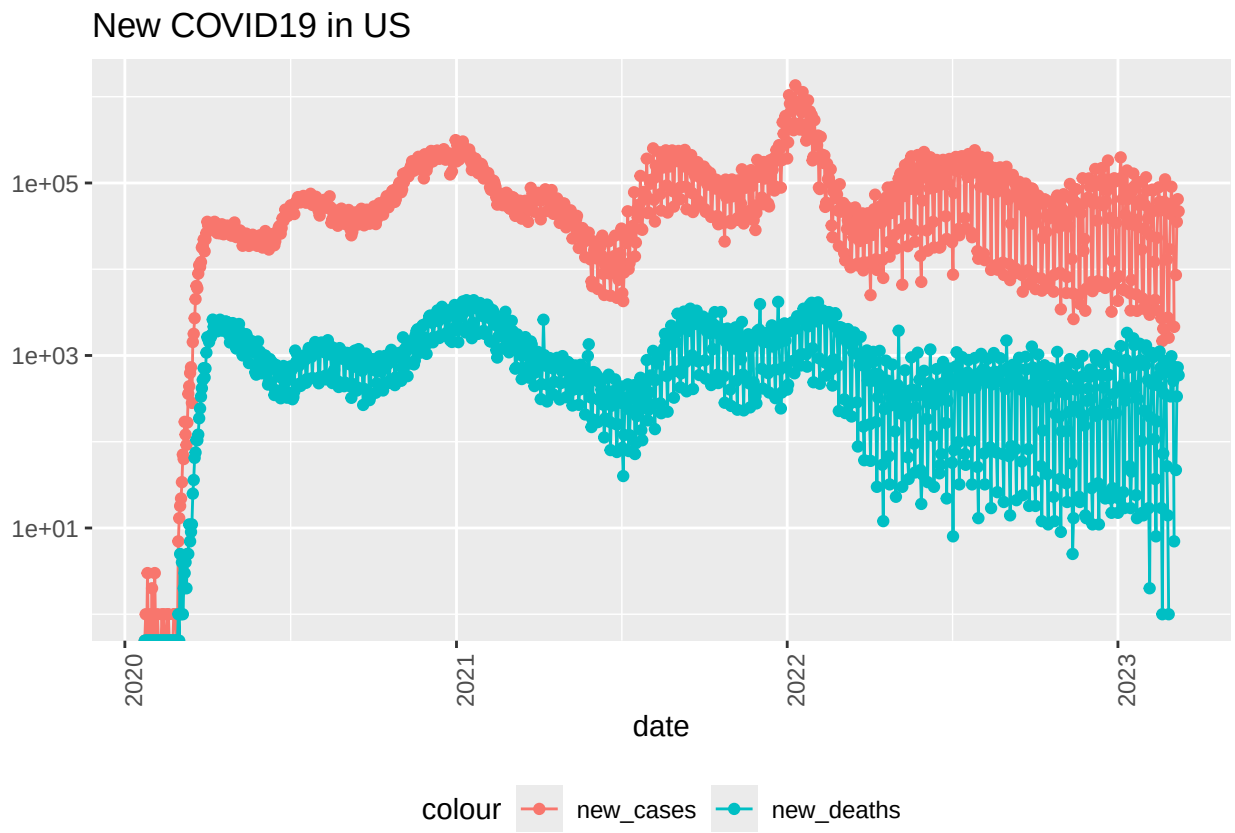
```
# Add new_cases and new_deaths columns to dataset
US_by_state <- US_by_state %>%
  mutate(new_cases = cases - lag(cases),
         new_deaths = deaths - lag(deaths))
US_totals <- US_totals %>%
  mutate(new_cases = cases - lag(cases),
         new_deaths = deaths - lag(deaths))

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>
```

```r
# Now lets graph the new cases and new deaths

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 = "New COVID19 in US", y = NULL)
```

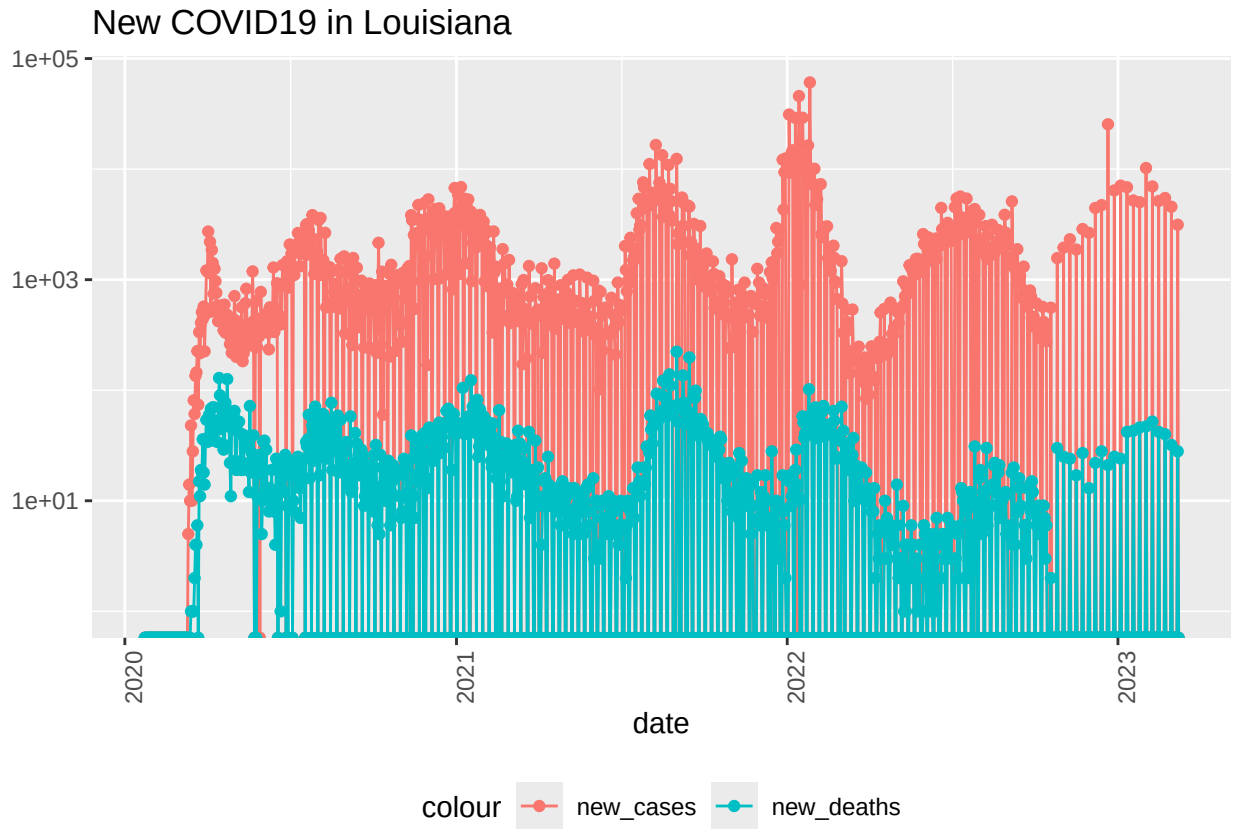


From the graph, we see that new covid 19 cases and deaths resulting from that have flatlined and stabilized

over time. Lets look at the state of Louisiana to see if this trend is similar to the US in total.

```
state <- "Louisiana"

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("New COVID19 in ", state), y = NULL)
```

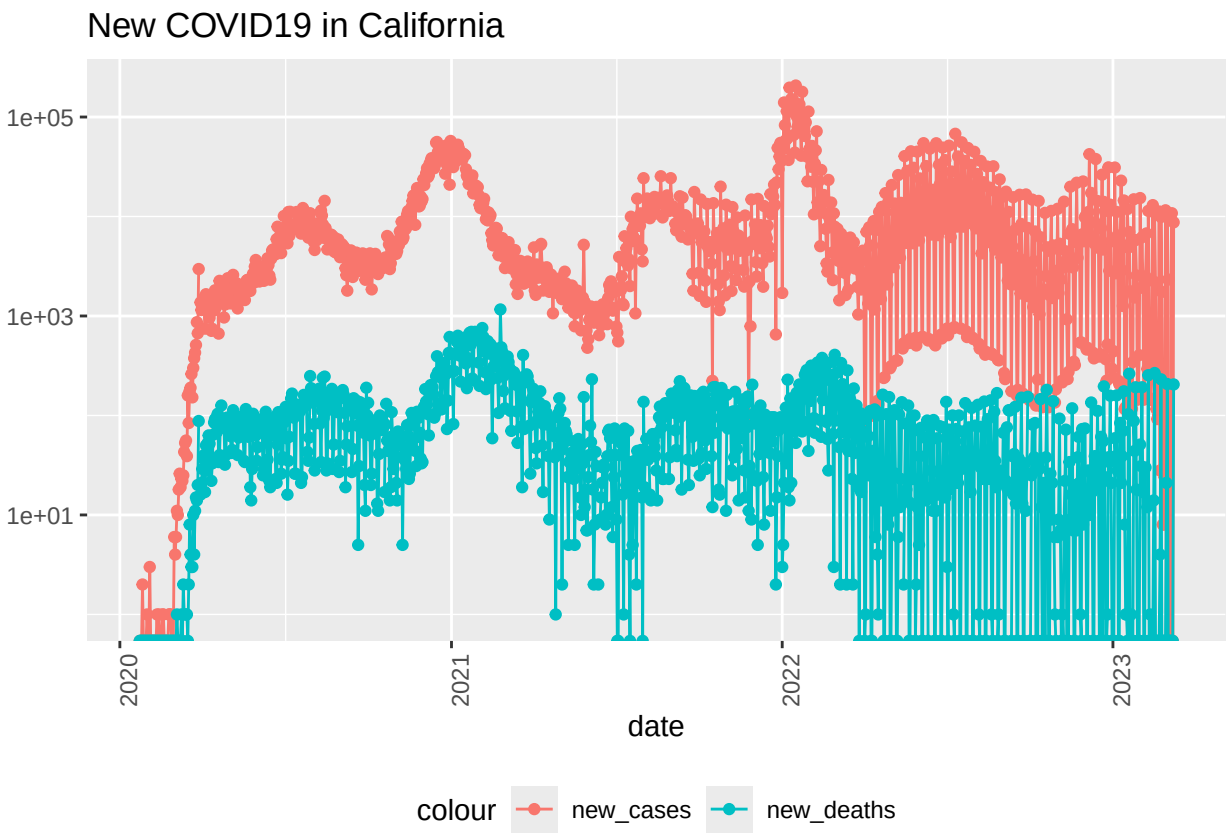


The Louisiana dataset is incredibly noisy. Does this have something to do with a lack of datapoints? Lets look at a state with higher population. I have chosen to look at California, the state with the highest population.

```
state <- "California"

US_by_state %>%
  filter(Province_State == state) %>%
  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("New COVID19 in ", state), y = NULL)
```



From the California graph, we see a similar trend with less noise than Louisiana. I believe that it is true that the noise from the Louisiana graph was caused by a lack of data.

Best and Worst States

We want to know the worst and best states. How do we measure this? Lets do more analysis!

```
# Best state
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)

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

```
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 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
```

```
# Worst state
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
```

Modeling Data

Originally, we used a linear model to see if a variable is statistically significant. Alternatively, I have chosen to use a poisson regression model to predict deaths_per_thou.

```
library(MASS)

# Prepare data
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)
```

Once our data is prepared, we will create the poisson regression model.

```
poisson_mod <- glm(deaths ~ cases_per_thou + population,
                   data = US_state_totals,
                   family = poisson(link = "log"))

summary(poisson_mod)
```

```
##
## Call:
## glm(formula = deaths ~ cases_per_thou + population, family = poisson(link = "log"),
##     data = US_state_totals)
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)    8.133e+00  7.627e-03  1066.4   <2e-16 ***
## cases_per_thou 3.539e-03  2.354e-05   150.4   <2e-16 ***
## population     7.325e-08  6.927e-11  1057.5   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 1222656  on 55  degrees of freedom
## Residual deviance:  405601  on 53  degrees of freedom
## AIC: 406217
##
## Number of Fisher Scoring iterations: 5
```

From our summary, we can see that both cases_per_thou and population have statistical significant influence on deaths_per_thou. In particular, cases_per_thou is the most impactful variable.

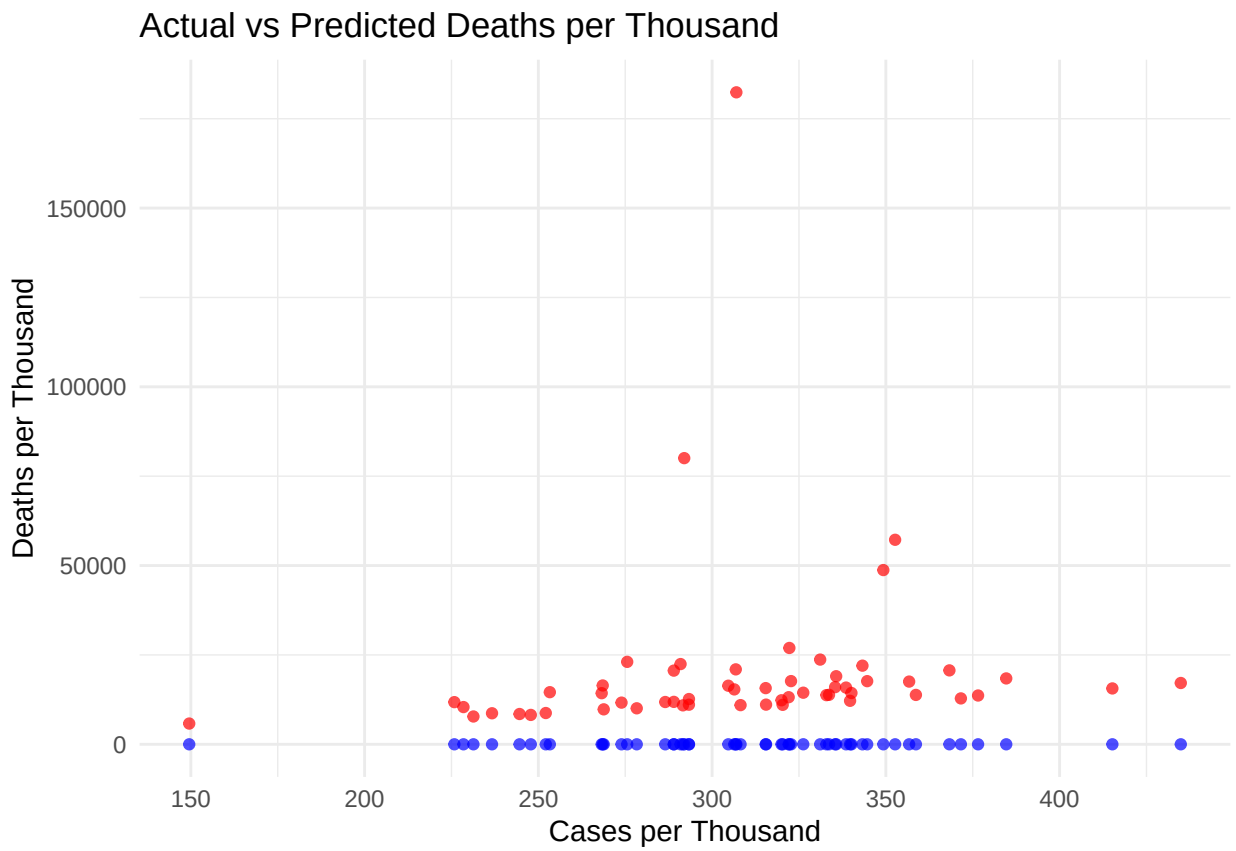
Using our model, we will add predictions to the dataset and visualize our actual deaths vs predicted deaths

```
# add predictions

US_tot_pred <- US_state_totals %>%
  mutate(pred_deaths = predict(poisson_mod, type = "response"))

# Visual of actual vs predicted

ggplot(US_tot_pred) +
  geom_point(aes(x = cases_per_thou, y = deaths_per_thou), color = "blue", alpha = 0.7) +
  geom_point(aes(x = cases_per_thou, y = pred_deaths), color = "red", alpha = 0.7) +
  labs(title = "Actual vs Predicted Deaths per Thousand",
       x = "Cases per Thousand",
       y = "Deaths per Thousand") +
  theme_minimal()
```



Looking at the graph, while the prediction follows the general trend, I wouldn't say the predictive model is accurate. Other factors seem to influence deaths_per_thou.

Bias

There is always bias involved with choosing a particular topic. In particular, my decision on what variables to focus on invovled bias. There is also bias in choosing to use poisson regression as opposed to other model types.

Conclusion

Our analysis of COVID 19 data revealed several insights into the pandemic's impact across the US. Initially, we found that the new cases and new deaths stabilized over time. Using our poisson regression model, we gained a better understanding of the factors that influenced death rates across the US. The model also highlighted the complexity of pandemic impact, showing that relying on cases and population alone is not sufficient to predict the number of deaths.