Covid 19 DataSet

2023-02-02

Covid-19

Libraries

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.7     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

## Loading required package: timechange

##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

Dataset

About the Data

This dataset comes from the Johns Hopkins github site collecting information on COVID-19 cases by state and by country.

Collecting the Raw Data

The first step is to record & combine URLs for desired .csv files that need to be downloaded.

```
url_in <- "https://github.com/CSSEGISandData/COVID-19/raw/master/csse_covid_19_data/csse_covid_19_time_s
file_names <- c("time_series_covid19_confirmed_US.csv", "time_series_covid19_confirmed_global.csv", "tim
urls <- str_c(url_in, file_names)
```

Next, we can read in the data sets.

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

The datasets are tidied by putting each variable in its own column, and disregarding Lat & Lon. I also renamed region & state to be easier to use in R.

```
global_cases <- global_cases %>%
  pivot_longer(cols= -c('Province/State', 'Country/Region', Lat, Long), # All column headings except fo
                                              names_to = "date",        # will now go as observations u
                                              values_to = "cases") %>%  # and their respective values w
  select(-c(Lat,Long))    # Lat and Long will be excluded
global_deaths <- global_deaths %>%  # Same adjustment made to global_deaths data
  pivot_longer(cols= -c('Province/State', 'Country/Region', Lat, Long),
                                              names_to = "date",
                                              values_to = "deaths") %>%
  select(-c(Lat,Long))
```

Deaths and cases per region are consolidated into a single global set

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

```
## Joining, by = c("Province/State", "Country/Region", "date")
```

Data Exploration

As the first step in exploring the data, I looked at the summary values by column.

```
summary(global)
```

```
##  Province_State     Country_Region         date                 cases
##  Length:328882      Length:328882      Min.   :2020-01-22   Min.   :        0
##  Class :character   Class :character   1st Qu.:2020-11-01   1st Qu.:      671
##  Mode  :character   Mode  :character   Median :2021-08-12   Median :    14265
##                                        Mean   :2021-08-12   Mean   :   953317
##                                        3rd Qu.:2022-05-24   3rd Qu.:   227225
##                                        Max.   :2023-03-04   Max.   :103645674
##      deaths
##  Min.   :      0
##  1st Qu.:      3
##  Median :    148
##  Mean   :  13334
##  3rd Qu.:   3013
##  Max.   :1122164
```

There are observations with zero cases, which do not contribute to our analysis, so these were removed.

```
global <- global %>% filter(cases >0)
summary(global)
```

```
##  Province_State     Country_Region          date                cases
##  Length:305392      Length:305392      Min.   :2020-01-22   Min.   :        1
##  Class :character   Class :character   1st Qu.:2020-12-11   1st Qu.:     1305
##  Mode  :character   Mode  :character   Median :2021-09-13   Median :    20204
##                                        Mean   :2021-09-09   Mean   :  1026644
##                                        3rd Qu.:2022-06-11   3rd Qu.:   270231
##                                        Max.   :2023-03-04   Max.   :103645674
##      deaths
##  Min.   :      0
##  1st Qu.:      7
##  Median :    213
##  Mean   :  14360
##  3rd Qu.:   3637
##  Max.   :1122164
```

I then filtered for observations of over 100 million cases to check that upper observations are not due to typos

```
global %>% filter(cases>100000000)
```

```
## # A tibble: 75 x 5
##    Province_State Country_Region date           cases  deaths
##    <chr>          <chr>          <date>         <dbl>   <dbl>
##  1 <NA>           US             2022-12-20 100050707 1088434
##  2 <NA>           US             2022-12-21 100232826 1089476
##  3 <NA>           US             2022-12-22 100328970 1090072
##  4 <NA>           US             2022-12-23 100368198 1090279
##  5 <NA>           US             2022-12-24 100374720 1090301
##  6 <NA>           US             2022-12-25 100377934 1090316
##  7 <NA>           US             2022-12-26 100390363 1090345
##  8 <NA>           US             2022-12-27 100501295 1090701
##  9 <NA>           US             2022-12-28 100614639 1091691
## 10 <NA>           US             2022-12-29 100718741 1092615
## # ... with 65 more rows
```

As there are multiple values in this range, it is likely accurate.

Finally, tidying up US cases and US deaths as was already done with global data

```
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_))

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

```
## Warning: 3342 failed to parse.
```

I then combined US data sets while excluding entries lacking a date.

```
US <- US_cases %>%
  full_join(US_deaths) %>%
  filter(!is.na(date))
```

```
## Joining, by = c("Admin2", "Province_State", "Country_Region", "Combined_Key",
## "date")
```

I then created a combined key for global data to better reflect the location

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

As population data was missing, I obtained it from UID Look-up Table

```
uid_lookup_url <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/
uid <- read_csv(uid_lookup_url) %>%
  select(-c(Lat, Long_, Combined_Key, code3, iso2, iso3))
```

```
## 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.
```

And then joined population data from UID table with the global data set

```
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)
```

I then isolated US population data from UID table

```
US_uid <- uid %>%
  filter(Country_Region == "US") %>%
  select(-c(UID, FIPS))
```

And added population data from UID table to the US data set

```
US <- US %>%
  left_join(uid, by = c("Admin2", "Province_State", "Country_Region")) %>%
  select(-c(UID, FIPS)) %>%
  select(Province_State, Country_Region, date, cases,
         deaths, Population, Combined_Key)
```

Now the data is tidied and organized.

Visualizing, Analyzing & Modelling Data

Question of interest: How do cases and deaths relate to each other over time and by state?

In order to assess this, I grouped Covid cases by location & summarized total cases and deaths.

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

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

I then used this to aggregate totals for the US overall.

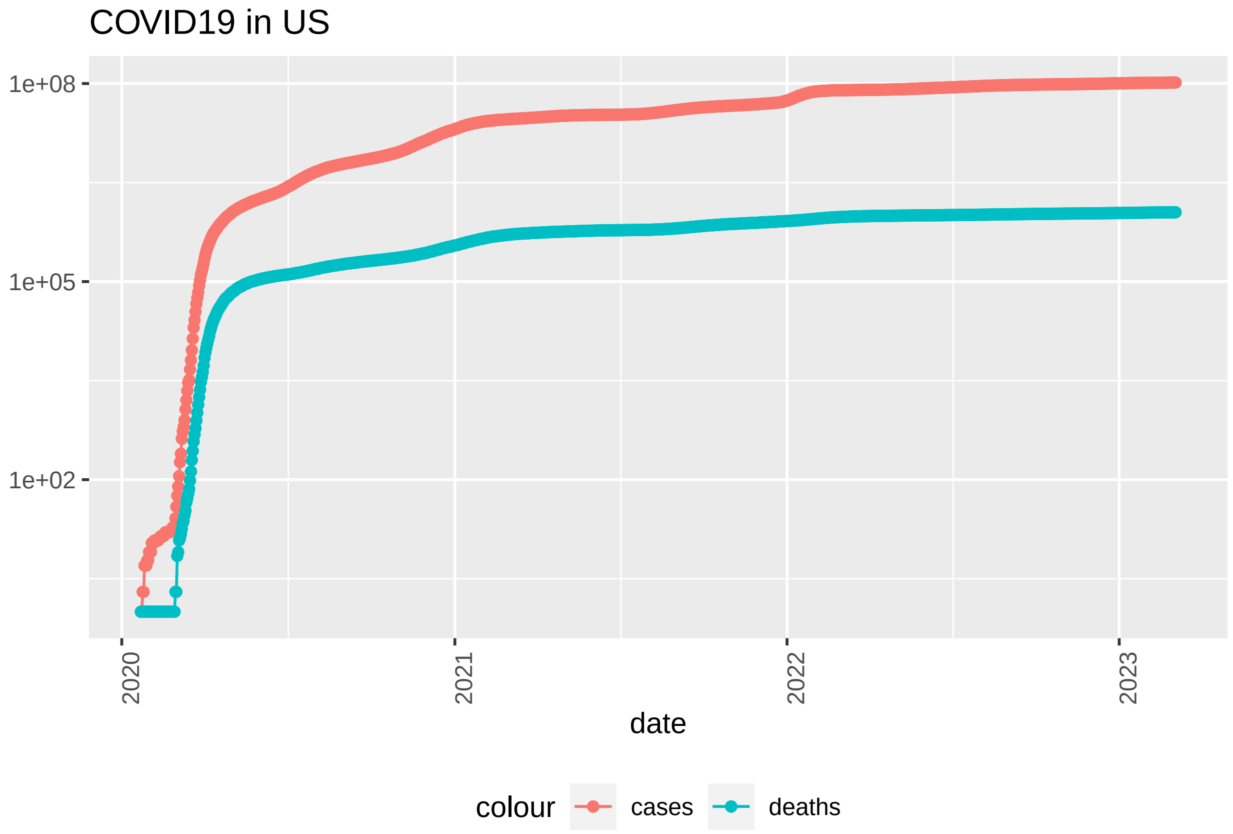
```
US_totals <- US_by_state %>%
  group_by(Country_Region, date) %>%
  summarize(cases = sum(cases), deaths = sum(deaths)) %>%
  select(Country_Region, date, cases, deaths) %>%
  ungroup()
```

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

Cases vs Deaths Over Time

The following graph displays results of cases and deaths over time (scaled logarithmically)

```
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)
```



This shows a steep rise in cases initially and then both cases and deaths in the early stages of the pandemic, followed by a gradual rise in both. However,it does not tell us whether or not new cases and deaths are leveling off. That is what we will assess next.

Instead of overall cases and deaths (above), examine NEW cases and deaths over time. First, add values for new cases and new deaths

```
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))
```

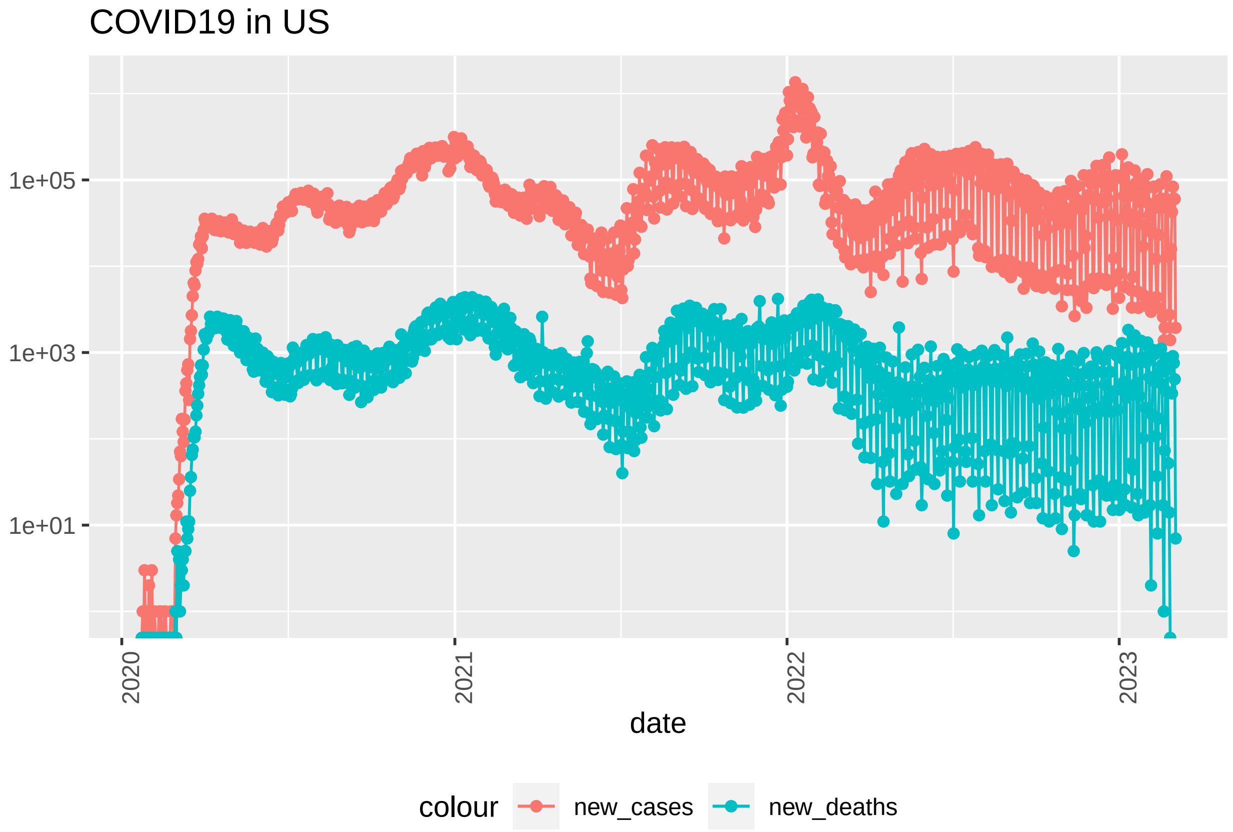
Look at most recent numbers, prioritizing new cases & new deaths

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

```
## # A tibble: 6 x 6
##   new_cases new_deaths Country_Region date           cases   deaths
##       <dbl>      <dbl> <chr>          <date>         <dbl>    <dbl>
## 1      15820        -10 US             2023-02-27 103396550 1119667
## 2      42553        335 US             2023-02-28 103439103 1120002
## 3      83376        909 US             2023-03-01 103522479 1120911
## 4      61160        756 US             2023-03-02 103583639 1121667
## 5      60103        490 US             2023-03-03 103643742 1122157
## 6       1932          7 US             2023-03-04 103645674 1122164
```

If we visualize new cases and new deaths, we see that it does taper off.

```
suppressWarnings(print(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 US", y = NULL)))
```

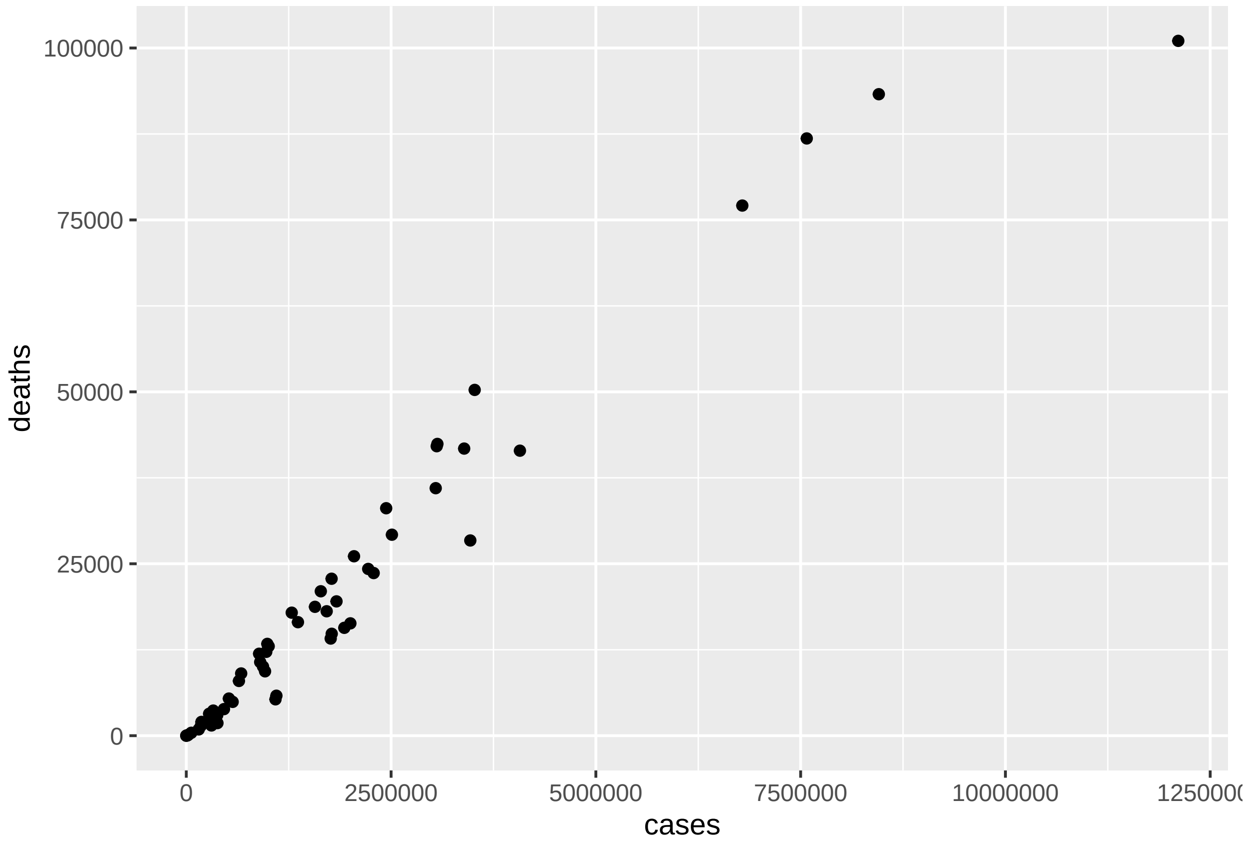


Based on this information, with some variation, new cases and deaths started to level off at the end of the first quarter of 2022, reaching an all-time high in early 2022, and remaining steady or decreasing slightly since then.

State-by-State Results

I would like to use the most recent data to assess the relative proportion of cases and deaths in each state.

```
US_by_state %>%
  filter(date == max(date)) %>%
  ggplot(aes(x=cases, y=deaths)) +
  geom_point()
```



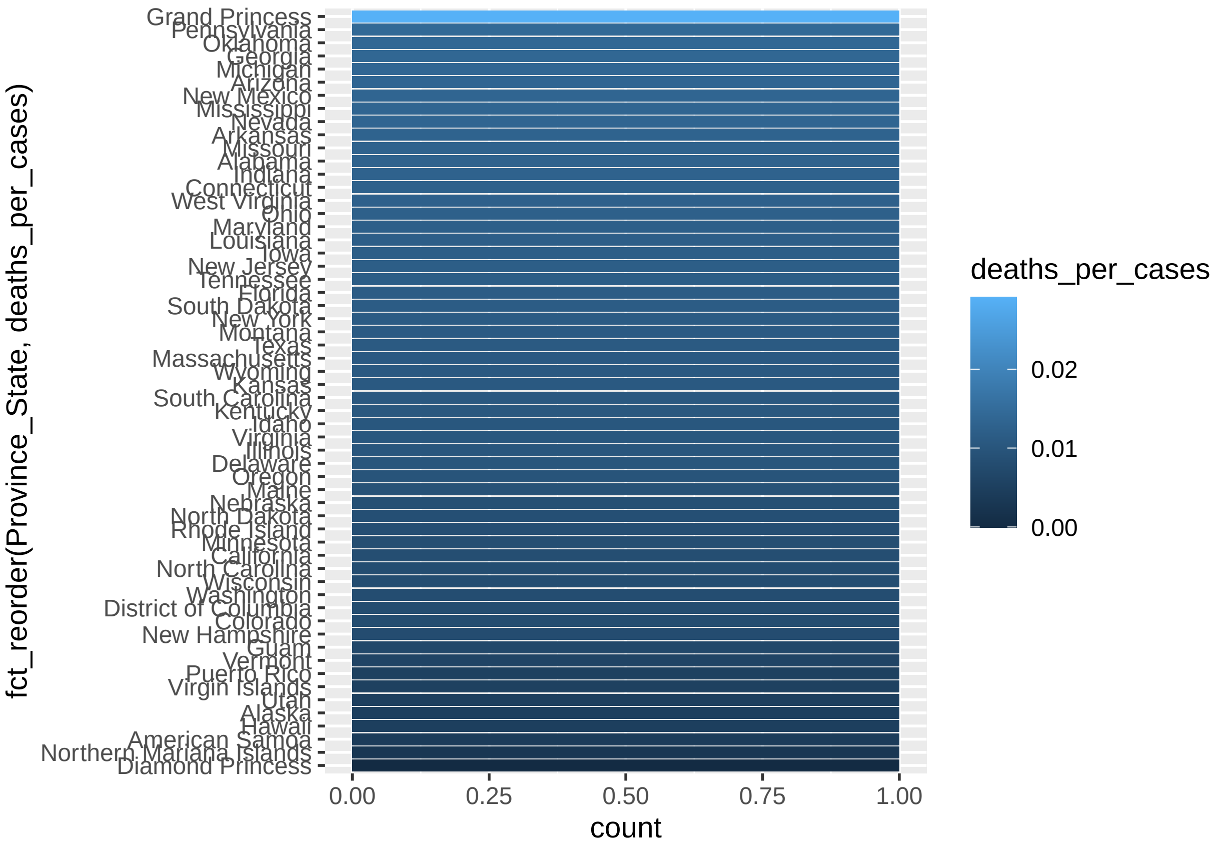
There appears to be clear correlation between cases and deaths, but it isn't perfectly linear, so I wanted to explore this further.

In order to better analyze the state-by-state relationship between cases and detahs, I created a new column for the rate of deaths per case for each state.

```
US_current <- US_by_state %>%
  filter(date == max(date)) %>%
  mutate(deaths_per_cases = deaths/cases)
```

The bar chart below shows the ratio of deaths to cases (from most recent data) sorted by highest proportion of deaths at the top to the lowest.

```
US_current %>%
  ggplot(aes(fill=deaths_per_cases, y=fct_reorder(Province_State, deaths_per_cases), lab=deaths_per_case
  geom_bar()
```



It seems that the Grand Princess (a cruise ship) had by far the largest number of deaths per cases, with the Diamond Princess having the lowest ratio. Among the states, Pennsylvania, Oklahoma, Georgia, Michigan, and Arizona had the highest five death rates while Alaska, Hawaii, Utah, Vermont, and New Hampshire had the lowest.

Conclusions

Death and Case counts appear to have leveled off in the US but are still nowhere near pre-pandemic levels, suggesting that we are in an endemic stage. There is some variation among states for the number of deaths per cases, though further analysis is necessary to explore the causes and contributing factors.

Bias Identification

For more recent COVID-19 data, the largest problem is a likely under-reporting of cases with the wide availability of home testing kits, as people who test positive at home may not be reporting to their local health agencies.