

Reproducible Report on COVID19 Data

Dan Wesely

2025-02-22

```
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  1.9.3      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)
```

Introduction

COVID 19 data project, based heavily on examples provided in class.

Loading Data

Load global and US data.

```
global_deaths <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/refs/heads/master/
```

```
## 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.
```

```
us_cases <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/refs/heads/master/csse_
```

```
## 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.
```

```
global_cases <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/refs/heads/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_global_confirmed.csv")
```

```
## 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.
```

```
us_deaths <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/refs/heads/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_us_deaths.csv")
```

```
## 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.
```

```
#global_deaths <- read_csv("time_series_covid19_deaths_global.csv")
#us_cases <- read_csv("time_series_covid19_confirmed_US.csv")
#global_cases <- read_csv("time_series_covid19_confirmed_global.csv")
#us_deaths <- read_csv("time_series_covid19_deaths_US.csv")
```

```
uid <- read_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/refs/heads/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_us_confirmed.csv")
```

```
## 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.
```

```
#uid <- read_csv("UID_ISO_FIPS_LookUp_Table.csv") %>% select(-c(Lat, Long_, Combined_Key, code3, iso2,
```

Tidying Data

Create tables of global and US cases and deaths.

```
global_cases_pivot <- global_cases %>% pivot_longer(cols = -c(`Province/State`, `Country/Region`, Lat, Lon), values_to = "date")
global_deaths_pivot <- global_deaths %>% pivot_longer(cols = -c(`Province/State`, `Country/Region`, Lat, Lon), values_to = "date")
```

```
us_cases_pivot <- us_cases %>% pivot_longer(cols = -(UID:Combined_Key), names_to = "date", values_to = "cases")
us_deaths_pivot <- us_deaths %>% pivot_longer(cols = -(UID:Population), names_to = "date", values_to = "deaths")
us <- us_cases_pivot %>% full_join(us_deaths_pivot)
```

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

Merge Tables

Combine global deaths and cases counts with population of the area.

```
global <- global_cases_pivot %>% full_join(global_deaths_pivot) %>% rename(Country_Region = `Country/Region`)
```

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

```
global <- global %>% unite("Combined_Key", c(Province_State, Country_Region), sep = ", ", na.rm = TRUE)
global <- global %>% left_join(uid, by = c("Province_State", "Country_Region")) %>% select(-c(UID, FIPS))
```

Combine US deaths and cases counts with population of the area.

```
us_by_state <- us %>% group_by(Province_State, Country_Region, date) %>% summarize(cases = sum(cases), deaths = sum(deaths))
```

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

```
us_by_state <- us_by_state %>% mutate(new_cases = cases - lag(cases), new_deaths = deaths - lag(deaths))
```

```
us_totals <- us_by_state %>% group_by(Country_Region, date) %>% summarize(cases = sum(cases), deaths = sum(deaths))
```

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

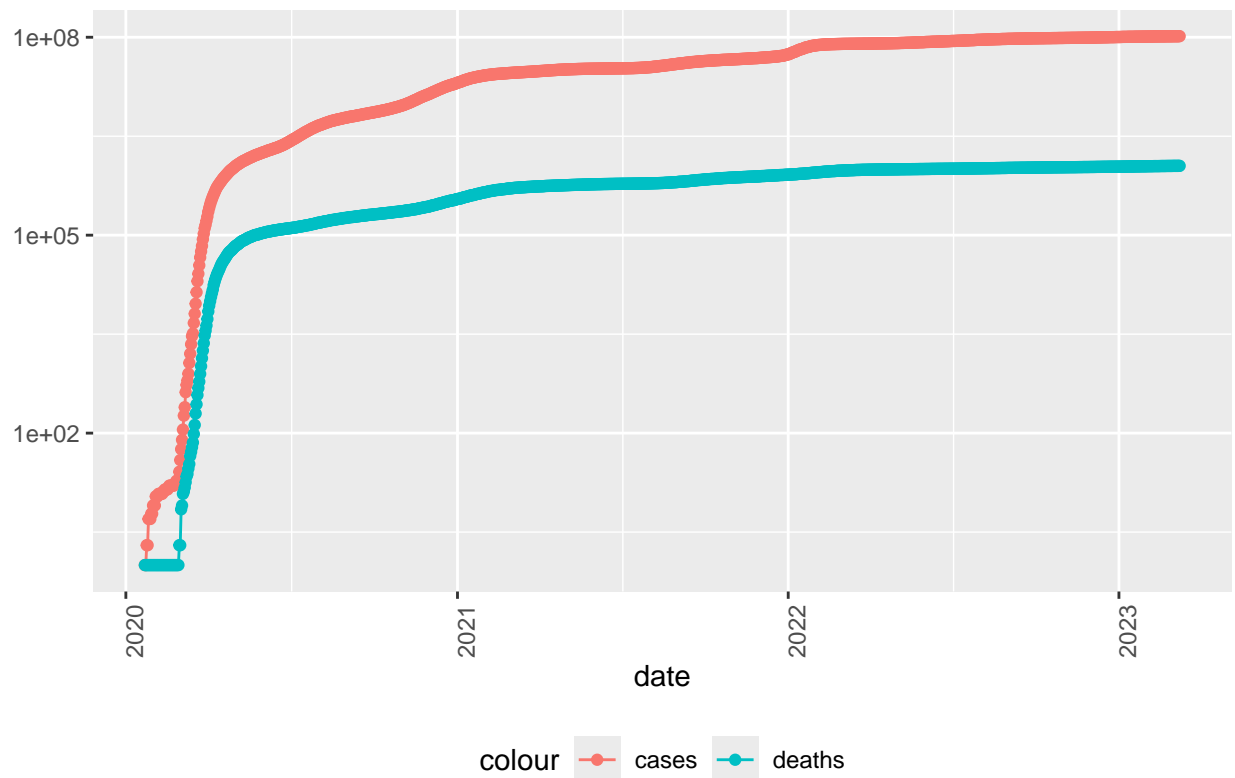
```
us_totals <- us_totals %>% mutate(new_cases = cases - lag(cases), new_deaths = deaths - lag(deaths))
```

Visualization

Plot cases and deaths over time, in US and in New York. Cumulative counts increase over time.

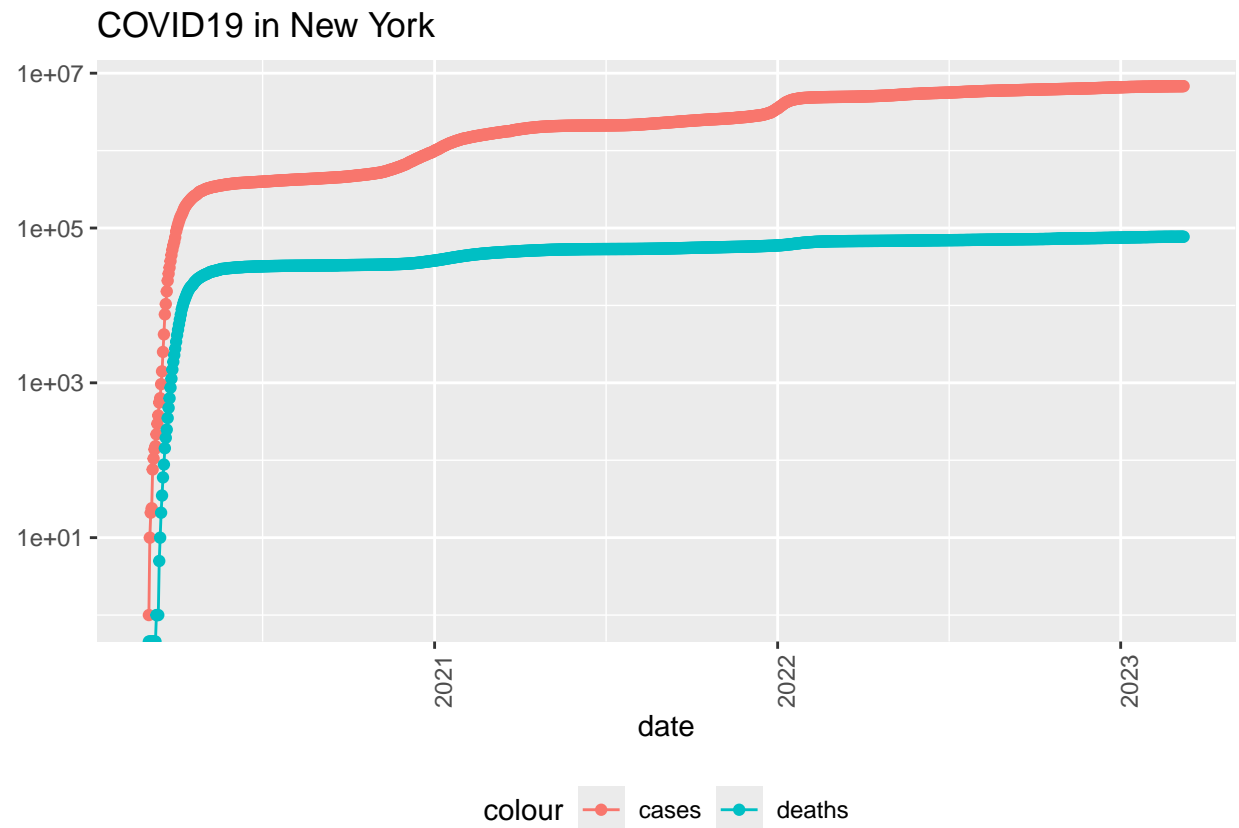
```
us_totals %>% filter(cases > 0) %>% ggplot(aes(x = date)) + geom_line(aes(y = cases, color = "cases")) +
```

COVID19 in US: Cases and Deaths



```
state <- "New York"
us_by_state %>% filter(Province_State == state) %>% filter(cases > 0) %>% ggplot(aes(x = date, y = ca
```

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



Delta

Plot new cases and new deaths, rather than cumulative total.

```
us_totals %>% ggplot(aes(x = date, y = new_cases)) + geom_line(aes(color = "new_cases")) + geom_point(aes(color = "new_deaths"))
```

```
## 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.
```

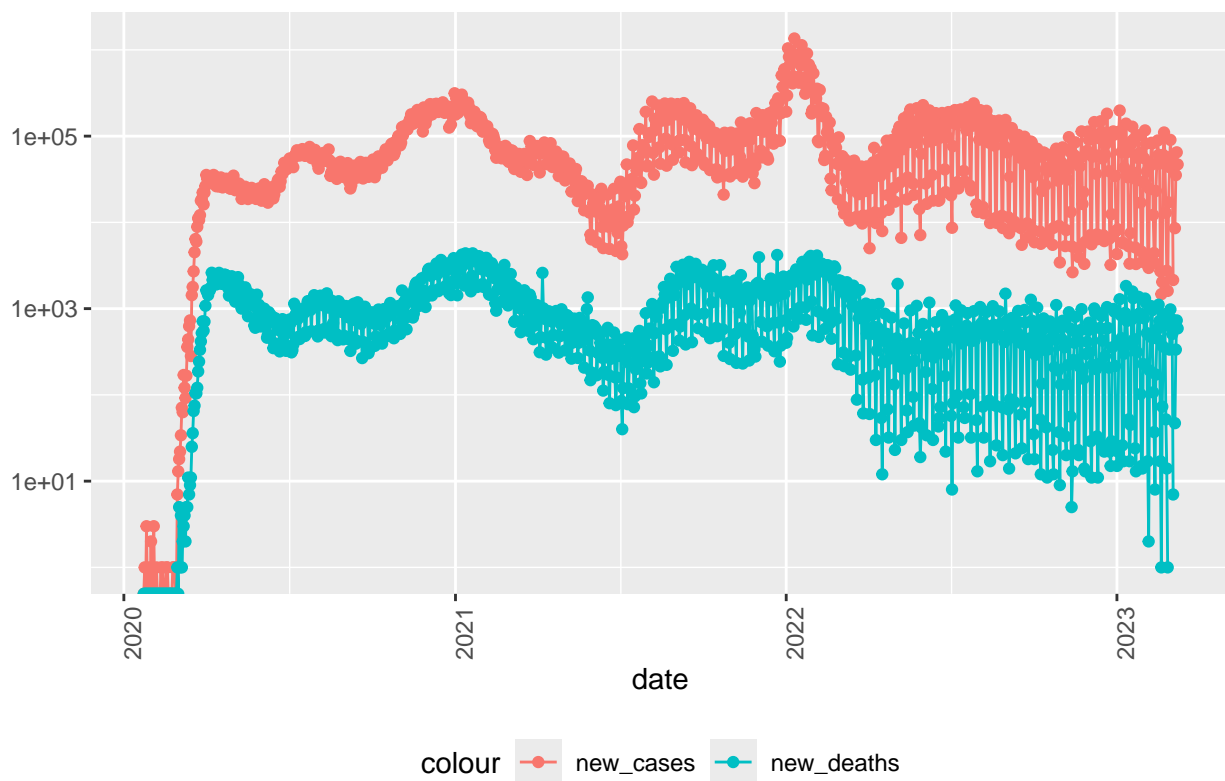
```
## Warning: Removed 1 row containing missing values or values outside the scale range
## ('geom_line()').

## 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()').
```

covid19 in US: New Cases



Rates

Convert counts to rates, using available population data.

```
us_state_totals <- us_by_state %>% group_by(Province_State) %>% summarize(deaths = max(deaths), cases =
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
```

```
us_state_totals %>% slice_max(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 Arizona        33102 2443514    7278717          336.            4.55
## 2 Oklahoma        17972 1290929    3956971          326.            4.54
## 3 Mississippi     13370 990756     2976149          333.            4.49
## 4 West Virginia   7960 642760     1792147          359.            4.44
## 5 New Mexico       9061 670929     2096829          320.            4.32
## 6 Arkansas        13020 1006883    3017804          334.            4.31
## 7 Alabama         21032 1644533    4903185          335.            4.29
## 8 Tennessee       29263 2515130    6829174          368.            4.28
## 9 Michigan        42205 3064125    9986857          307.            4.23
## 10 Kentucky       18130 1718471    4467673          385.            4.06
```

Model

Fit a linear model to the data to predict the number of deaths per thousand people, based on the number of cases per thousand people.

```
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(deaths_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(deaths_per_thou)
```

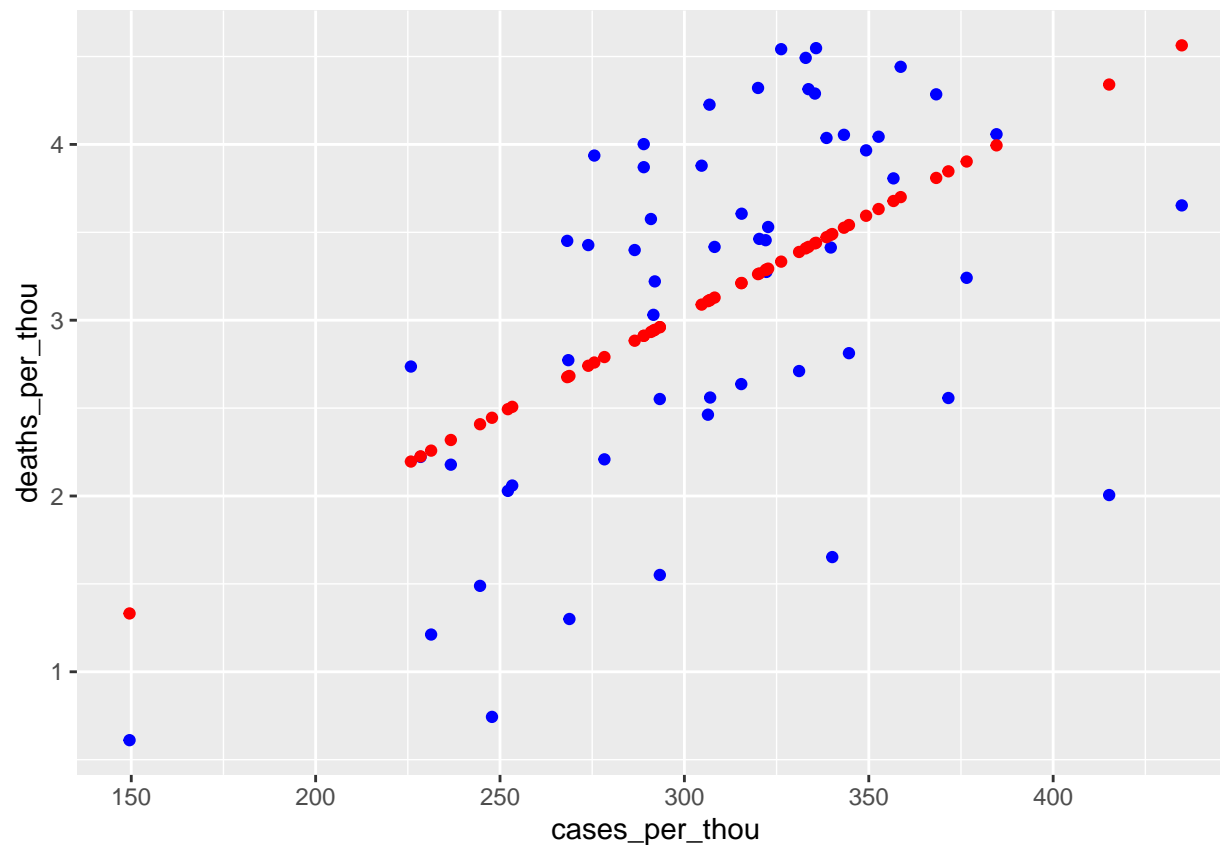
```
## # A tibble: 1 x 6
##   Province_State deaths cases population cases_per_thou deaths_per_thou
##   <chr>          <dbl> <dbl>         <dbl>         <dbl>         <dbl>
## 1 Arizona      33102 2443514      7278717          336.          4.55
```

```
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_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") +
```

Bias

Differences in reporting processes and infrastructure in the different states could have caused case and death rates to be skewed. Different state resources may have resulted in more or less accurate identification of COVID as well, which could again affect rates.

```
sessionInfo()
```

```
## R version 4.4.1 (2024-06-14 ucrt)
## Platform: x86_64-w64-mingw32/x64
## Running under: Windows 10 x64 (build 19045)
##
## Matrix products: default
##
##
## locale:
## [1] LC_COLLATE=English_United States.utf8
## [2] LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## time zone: America/New_York
## tzcode source: internal
##
```

```

## attached base packages:
## [1] stats      graphics  grDevices utils      datasets  methods   base
##
## other attached packages:
## [1] lubridate_1.9.3 forcats_1.0.0  stringr_1.5.1  dplyr_1.1.4
## [5] purrr_1.0.2     readr_2.1.5    tidyr_1.3.1    tibble_3.2.1
## [9] ggplot2_3.5.1   tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
## [1] bit_4.5.0      gtable_0.3.5    highr_0.11      crayon_1.5.3
## [5] compiler_4.4.1 tidyselect_1.2.1 tinytex_0.55     parallel_4.4.1
## [9] scales_1.3.0   yaml_2.3.10     fastmap_1.2.0   R6_2.5.1
## [13] labeling_0.4.3 generics_0.1.3   curl_5.2.3      knitr_1.48
## [17] munsell_0.5.1  pillar_1.9.0    tzdb_0.4.0      rlang_1.1.4
## [21] utf8_1.2.4     stringi_1.8.4   xfun_0.48       bit64_4.5.2
## [25] timechange_0.3.0 cli_3.6.3        withr_3.0.1     magrittr_2.0.3
## [29] digest_0.6.37  grid_4.4.1      vroom_1.6.5     rstudioapi_0.17.0
## [33] hms_1.1.3      lifecycle_1.0.4 vctrs_0.6.5     evaluate_1.0.1
## [37] glue_1.8.0     farver_2.1.2    fansi_1.0.6     colorspace_2.1-1
## [41] rmarkdown_2.28 tools_4.4.1      pkgconfig_2.0.3 htmltools_0.5.8.1

```