Covid19

Mike Brozowski

2024-06-08

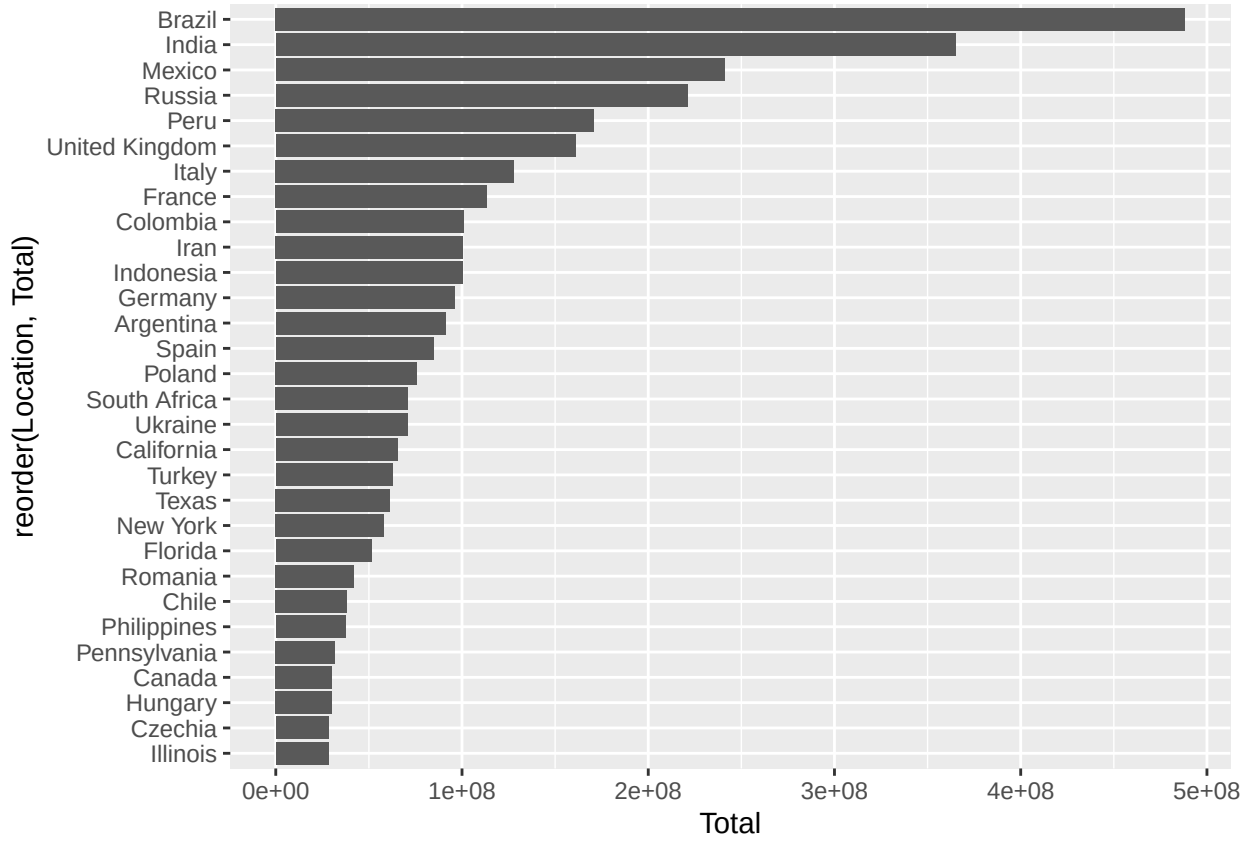
```
tidy_confirmed_us = raw_confirmed_us %>%
  pivot_longer(cols=names(raw_confirmed_us)[12:length(raw_confirmed_us)],
               names_to="Date",
               values_to="Confirmed") %>%
  select(c(Admin2,Province_State,Date,Confirmed)) %>%
  rename(County = Admin2, State = Province_State)
tidy_deaths_us = raw_deaths_us %>%
  pivot_longer(cols=names(raw_deaths_us)[13:length(raw_deaths_us)],
               names_to="Date",
               values_to="Deaths") %>%
  select(c(Admin2,Province_State,Population,Date,Deaths)) %>%
  rename(County = Admin2, State = Province_State)
tidy_confirmed_global = raw_confirmed_global %>%
  pivot_longer(cols=-c("Province/State","Country/Region",Lat,Long),
               names_to="Date",
               values_to="Confirmed") %>%
  select(-c(Lat,Long))
tidy_deaths_global = raw_deaths_global %>%
  pivot_longer(cols=-c("Province/State","Country/Region",Lat,Long),
               names_to="Date",
               values_to="Deaths") %>%
  select(-c(Lat,Long))
```

Visual 1: Deaths Overall - Top 30 US States Or Countries

The chart below shows the top 30 US States or Countries with the highest death count. With the United States included, it would be the highest. Breaking down the data in smaller subsets, such as states, may bring some insight into how each government or population handled the pandemic. In the effort of fairness, if the data was included, other countries should be broken down into similar groups: Canada into provinces, India into states, etc. to best represent a population.

```
# total deaths per state
total_deaths_per_state = tidy_deaths_us %>%
  group_by(State) %>%
  summarize(Total=sum(Deaths)) %>%
  rename(Location=State)
# total deaths per country (except US)
total_deaths_per_country = tidy_deaths_global %>%
  group_by(`Country/Region`) %>%
  filter(`Country/Region` != "US") %>%
```

```
  summarize(Total=sum(Deaths)) %>%
  rename(Location=`Country/Region`)
# union
total_deaths_per_location = union(total_deaths_per_state, total_deaths_per_country)
# get top 30
top_total_deaths_per_location = total_deaths_per_location %>%
  arrange(-Total) %>%
  top_n(30,Total)
# y axis deaths, x axis location
ggplot(top_total_deaths_per_location) +
  geom_col(aes(x=Total,y=reorder(Location,Total)))
```



Visual 2: Death Rate Overall - Top 30 US States or Countries

This chart takes the rate of death into account, comparing the death total to the confirmed case total to find a death or recovery rate. The chart below shows the top 30 highest death rates. The highest population on the chart isn't even a country, but rather a cruise ship. It isn't the only cruise ship to show up on this chart, the Grand Princess is another one with a high death rate. With the close proximity from person to person and lack of hospital medical staff, it can be inferred that is the reason for such a high death rate. Although the United States was excluded from the country-list in favor of showing states, no US state shows up on this list.

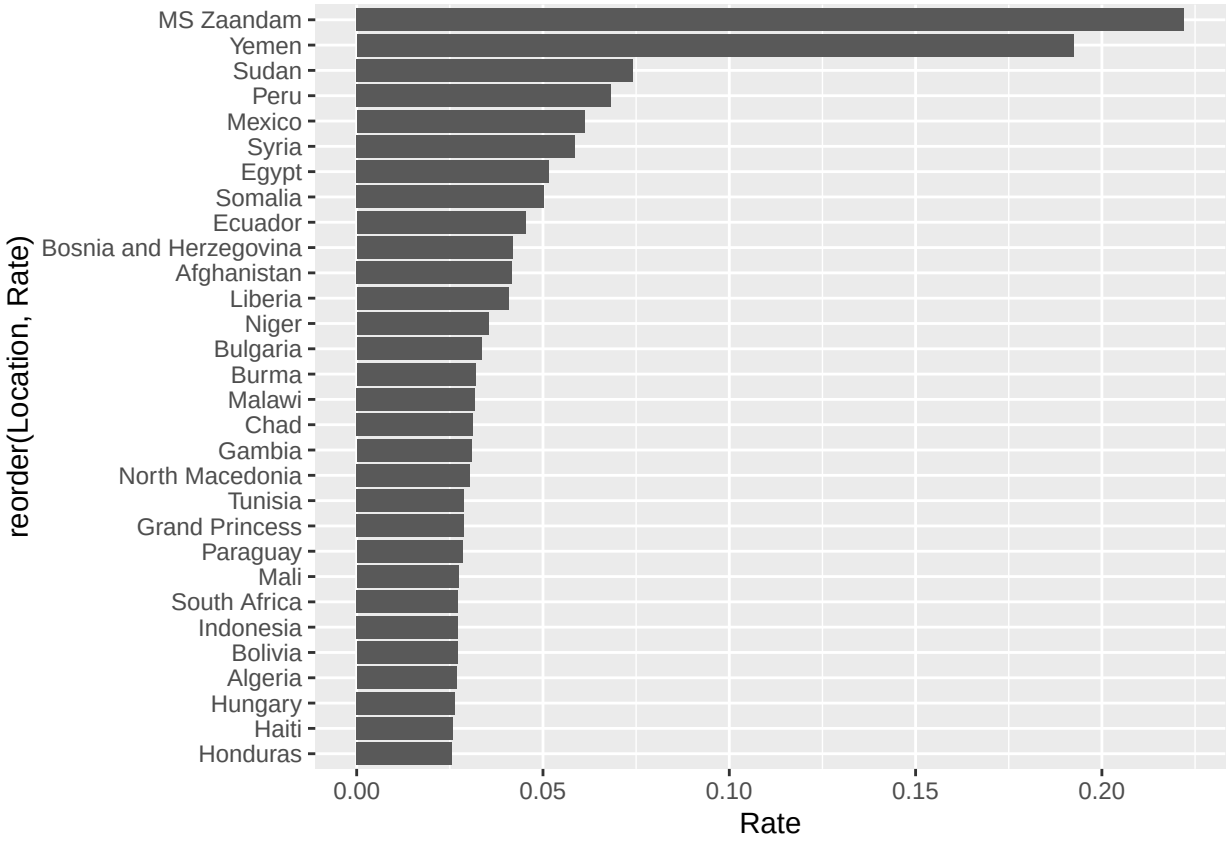
```
# total confirmed per state
total_confirmed_per_state = tidy_confirmed_us %>%
  group_by(State) %>%
  summarize(Confirmed=sum(Confirmed)) %>%
  rename(Location=State)
# total deaths per state
total_deaths_per_state = tidy_deaths_us %>%
  group_by(State) %>%
  summarize(Deaths=sum(Deaths)) %>%
  rename(Location=State)
# join - death rate per state
state_join = full_join(total_confirmed_per_state, total_deaths_per_state)
```

```
## Joining with `by = join_by(Location)`
```

```
# total confirmed per country (except US)
total_confirmed_per_country = tidy_confirmed_global %>%
  group_by(`Country/Region`) %>%
  filter(`Country/Region` != "US") %>%
  summarize(Confirmed=sum(Confirmed)) %>%
  rename(Location=`Country/Region`)
# total deaths per country (except US)
total_deaths_per_country = tidy_deaths_global %>%
  group_by(`Country/Region`) %>%
  filter(`Country/Region` != "US") %>%
  summarize(Deaths=sum(Deaths)) %>%
  rename(Location=`Country/Region`)
# join - death rate per country (except US)
country_join = full_join(total_confirmed_per_country, total_deaths_per_country)
```

```
## Joining with `by = join_by(Location)`
```

```
# union
total_join = union(state_join, country_join) %>%
  group_by(Location,Confirmed,Deaths) %>%
  filter(Confirmed>0) %>%
  reframe(Rate=(Deaths/Confirmed)) %>%
  select(c(Location,Rate,Confirmed,Deaths))
# get top 30
top_total_per_location = total_join %>%
  arrange(Rate) %>%
  filter(Location != "Korea, North") %>%
  ungroup() %>%
  top_n(30,Rate)
# y axis death rate, x axis location
ggplot(top_total_per_location) +
  geom_col(aes(x=Rate,y=reorder(Location,Rate)))
```



Model 1: Correlation Between County Population and Death Rate

The plot below shows all United States counties by population (y-axis) and death rate (x-axis) in order to see if there was a correlation between population and death rate. The red vertical line is the mean of all death rates and the black vertical line is the median of all death rates. Since the median and mean are very close, no strong correlation one way or another is justified. Strangely, there is a *slight* lean towards the lower population counties to have a higher death rate.

```
# total confirmed per county
total_confirmed_per_county = tidy_confirmed_us %>%
  group_by(State,County) %>%
  summarize(Confirmed=sum(Confirmed)) %>%
  rename(Location=County)
```

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

```
# total deaths per county
total_deaths_per_county = tidy_deaths_us %>%
  group_by(State,County) %>%
  summarize(Deaths=sum(Deaths),Population=mean(Population)) %>%
  rename(Location=County) %>%
  select(c(Location,Deaths,Population))
```

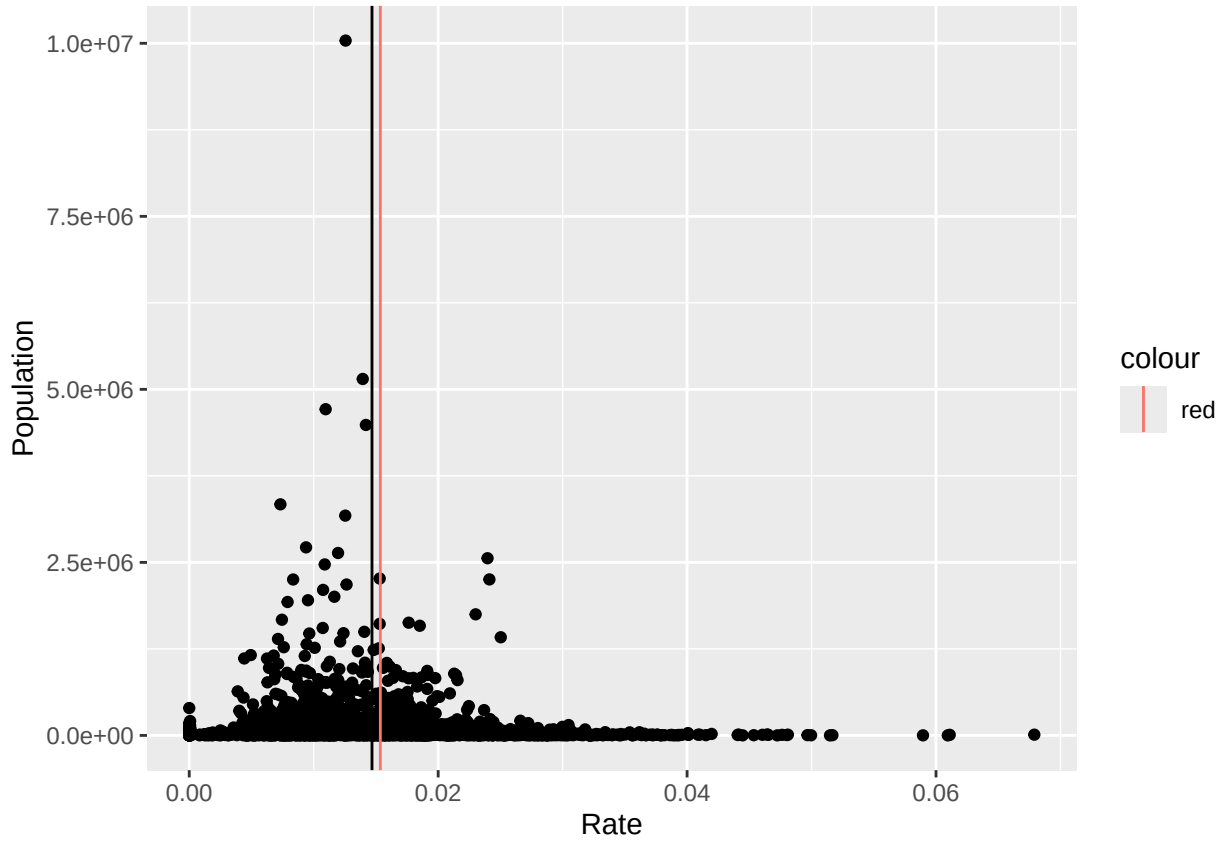
```
## `summarise()` has grouped output by 'State'. You can override using the
## `.groups` argument.
## Adding missing grouping variables: `State`
```

```
# join - death rate per county
county_join = full_join(total_confirmed_per_county, total_deaths_per_county) %>%
  group_by(State,Location,Confirmed,Deaths,Population) %>%
  reframe(Rate=(Deaths/Confirmed)) %>%
  filter(Location != "Unassigned") %>%
  filter(Rate != Inf) %>%
  select(c(Location,Rate,Confirmed,Deaths,Population))
```

```
## Joining with `by = join_by(State, Location)`
```

```
# y axis death rate, x axis population
ggplot(county_join, aes(x=Rate,y=Population)) +
  geom_point() +
  geom_vline(aes(xintercept=mean(county_join$Rate),color="red")) +
  geom_vline(aes(xintercept=median(county_join$Rate)))
```

```
## Warning: Use of `county_join$Rate` is discouraged.
## i Use `Rate` instead.
## Use of `county_join$Rate` is discouraged.
## i Use `Rate` instead.
```



```r
# model line for correlation between the two
model = lm(Population ~ Rate, data = county_join)
summary(model)
```

```
##
## Call:
## lm(formula = Population ~ Rate, data = county_join)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -167803  -90893  -64389  -24897 9924347
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   167803      13119  12.791  < 2e-16 ***
## Rate        -4221519     767167  -5.503 4.03e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 328000 on 3222 degrees of freedom
## Multiple R-squared:  0.00931,    Adjusted R-squared:  0.009003
## F-statistic: 30.28 on 1 and 3222 DF,  p-value: 4.032e-08
```

```r
sessionInfo()
```

```
## R version 4.4.0 (2024-04-24 ucrt)
## Platform: x86_64-w64-mingw32/x64
## Running under: Windows 11 x64 (build 22631)
##
## 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_4.0.5         gtable_0.3.5      highr_0.10        crayon_1.5.2
##  [5] compiler_4.4.0    tidyselect_1.2.1  parallel_4.4.0    scales_1.3.0
```

```
##  [9] yaml_2.3.8        fastmap_1.1.1     R6_2.5.1          labeling_0.4.3
## [13] generics_0.1.3    curl_5.2.1        knitr_1.46        munsell_0.5.1
## [17] pillar_1.9.0      tzdb_0.4.0        rlang_1.1.3       utf8_1.2.4
## [21] stringi_1.8.4     xfun_0.43         bit64_4.0.5       timechange_0.3.0
## [25] cli_3.6.2         withr_3.0.0       magrittr_2.0.3    digest_0.6.35
## [29] grid_4.4.0        vroom_1.6.5       rstudioapi_0.16.0 hms_1.1.3
## [33] lifecycle_1.0.4   vctrs_0.6.5       evaluate_0.23     glue_1.7.0
## [37] farver_2.1.2      fansi_1.0.6       colorspace_2.1-0  rmarkdown_2.26
## [41] tools_4.4.0       pkgconfig_2.0.3   htmltools_0.5.8.1
```