Covid-19 Data Report

Meg McGrath

2025-11-15

This project will use data from John Hopkins, available on Github. It includes reported COVID cases and deaths worldwide. The database also includes some information that we will not be using for this report, for example latitude and longitude.
This project will also use population estimates from database.earth.
We will look at the data from the US compared to four other countries: Germany, Brazil, India, and Russia. While obviously not a complete worldview of the impacts of the pandemic, this selection of countries provides variety without overloading our analysis by looking at data from too many countries at once. By looking at the COVID data from these five countries, we will examine whether they had similar or different experiences with COVID cases and deaths, and whether any one country was possibly more successful in managing the pandemic.

```
# import data
global_cases <- 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.
```

```
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.
```

```
# take a look at data
head(global_cases)
```

```
## # A tibble: 6 x 1,147
##   `Province/State` `Country/Region`   Lat  Long `1/22/20` `1/23/20` `1/24/20`
##   <chr>            <chr>            <dbl> <dbl>     <dbl>     <dbl>     <dbl>
```

```
## 1 <NA>             Afghanistan       33.9 67.7          0         0         0
## 2 <NA>             Albania           41.2 20.2          0         0         0
## 3 <NA>             Algeria           28.0  1.66         0         0         0
## 4 <NA>             Andorra           42.5  1.52         0         0         0
## 5 <NA>             Angola           -11.2 17.9          0         0         0
## 6 <NA>             Antarctica       -71.9 23.3          0         0         0
## # i 1,140 more variables: `1/25/20` <dbl>, `1/26/20` <dbl>, `1/27/20` <dbl>,
## #   `1/28/20` <dbl>, `1/29/20` <dbl>, `1/30/20` <dbl>, `1/31/20` <dbl>,
## #   `2/1/20` <dbl>, `2/2/20` <dbl>, `2/3/20` <dbl>, `2/4/20` <dbl>,
## #   `2/5/20` <dbl>, `2/6/20` <dbl>, `2/7/20` <dbl>, `2/8/20` <dbl>,
## #   `2/9/20` <dbl>, `2/10/20` <dbl>, `2/11/20` <dbl>, `2/12/20` <dbl>,
## #   `2/13/20` <dbl>, `2/14/20` <dbl>, `2/15/20` <dbl>, `2/16/20` <dbl>,
## #   `2/17/20` <dbl>, `2/18/20` <dbl>, `2/19/20` <dbl>, `2/20/20` <dbl>, ...
```

```
# clean data, changing dates to rows

global_cases_long <- global_cases %>%
  pivot_longer(
    cols = matches("^\\d{1,2}/\\d{1,2}/\\d{2}$"),
    names_to = "date",
    values_to = "cases"
  ) %>%
  mutate(date = as.Date(date, format = "%m/%d/%y"))

global_deaths_long <- global_deaths %>%
    pivot_longer(
    cols = matches("^\\d{1,2}/\\d{1,2}/\\d{2}$"),
    names_to = "date",
    values_to = "deaths"
  ) %>%
  mutate(date = as.Date(date, format = "%m/%d/%y"))
```

```
# continue to clean data, want to view only date, location, and totals

global_cases_clean <- global_cases_long %>%
  mutate(country = `Country/Region`) %>%
  group_by(country) %>%
  select(date, country, cases
  )
global_deaths_clean <- global_deaths_long %>%
  mutate(country = `Country/Region`) %>%
  group_by(country) %>%
  select(date, country, deaths
  )
```

```
max(global_cases_clean$date)
```

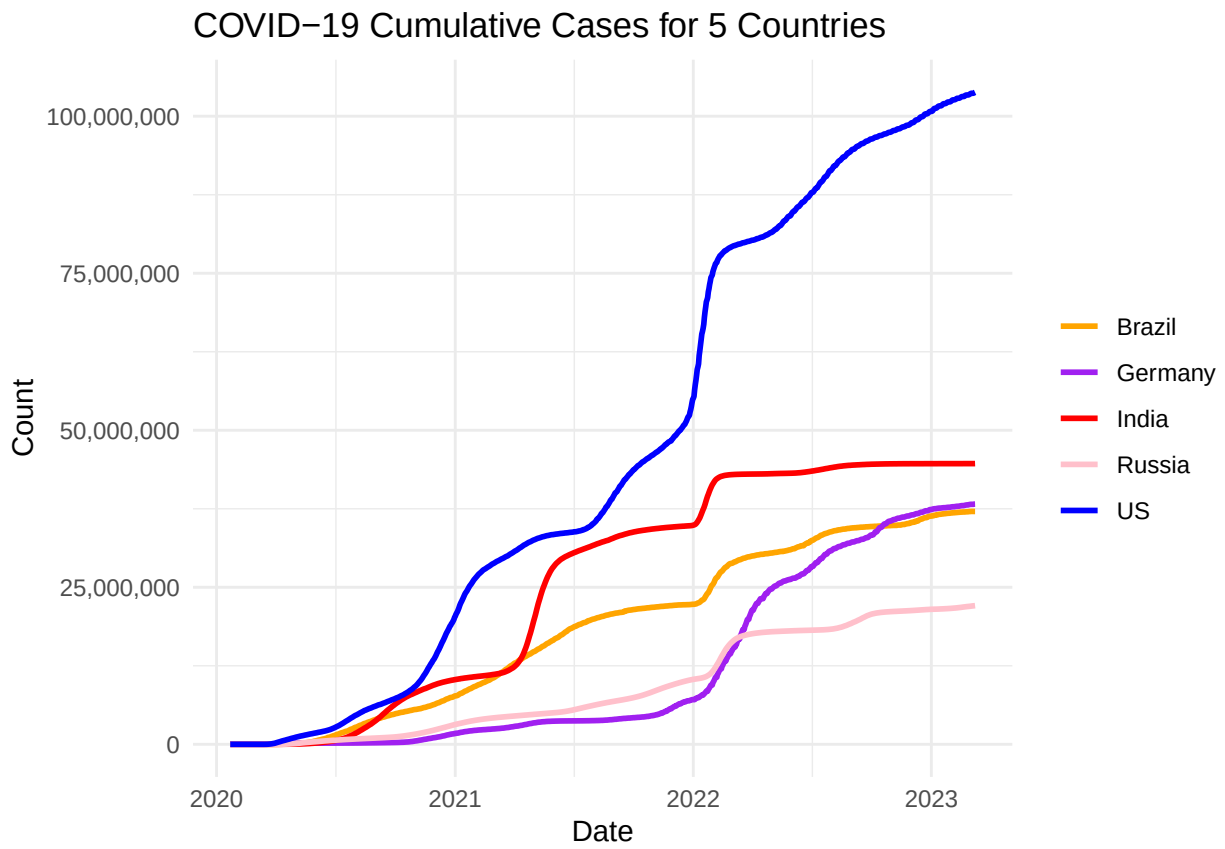
```
## [1] "2023-03-09"
```

```
# 2023-03-09

# will use estimates of populations from 2020 from the following database
# https://database.earth/population/by-country/2020
```

```r
# choosing 5 countries to compare
five_countries <- c("US", "India", "Russia", "Germany", "Brazil")

five_global_cases <- global_cases_clean %>%
  filter(country %in% five_countries)

five_global_deaths <- global_deaths_clean %>%
  filter(country %in% five_countries)
```

After cleaning all of the data, we will take a look at the cumulative cases for the five selected countries.

```r
ggplot(five_global_cases, aes(x = date, y = cases, color = country)) +
  geom_line(linewidth = 1) +
  labs(
    title = "COVID-19 Cumulative Cases for 5 Countries",
    x = "Date",
    y = "Count"
  ) +
  scale_color_manual(values = c(
    "US" = "blue",
    "India" = "red",
    "Russia" = "pink",
    "Germany" = "purple",
    "Brazil" = "orange"
    )) +
  scale_y_continuous(labels = scales::comma) +
  theme_minimal() +
  theme(legend.title = element_blank())
```



Looking at cumulative cases, we can see that the US has more than double of any of the other countries. But we also know that this data does not tell us much if we do not take into consideration the population size of each country.

Next, we will create two separate visuals for cumulative cases and cumulative deaths per 100,000 people, each based on the respective country's population. We will include linear regression models, to see if the data consistently follows a linear trend.

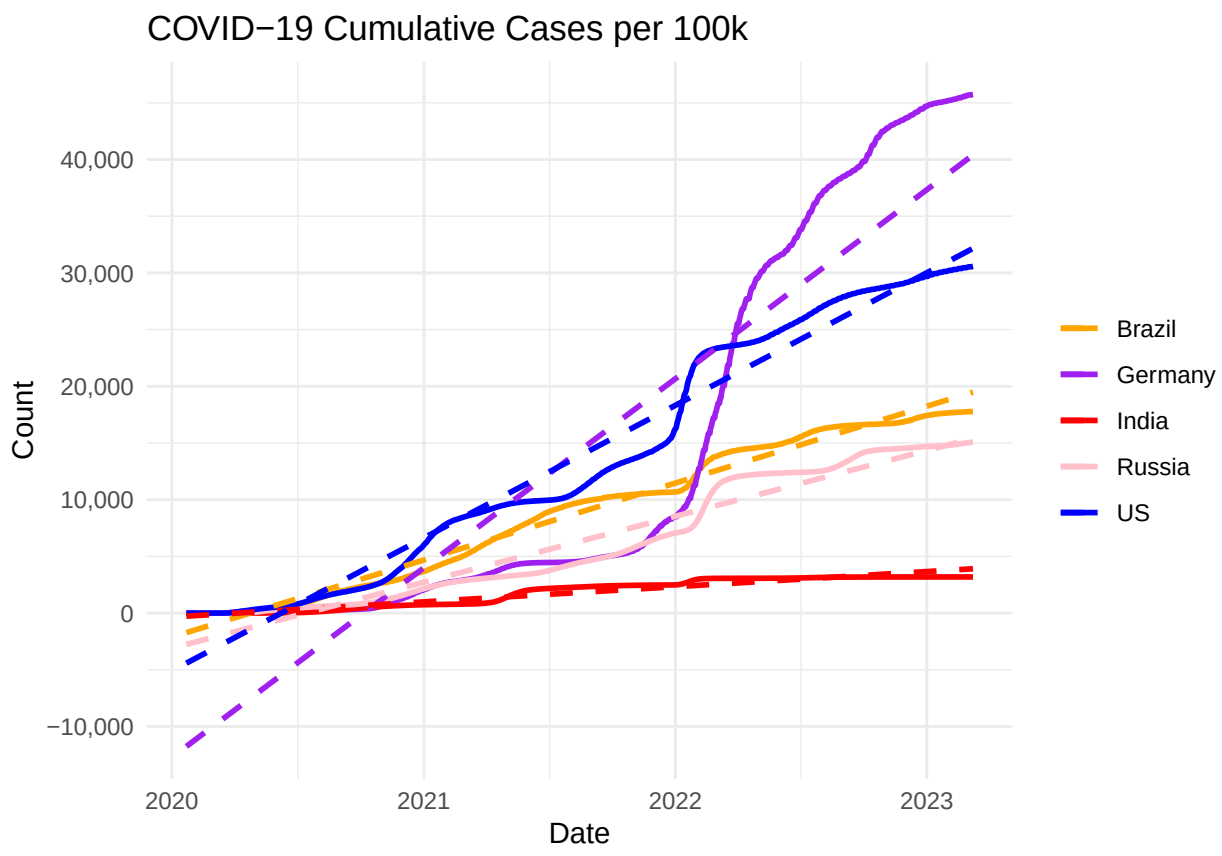
```
population <- tibble(
  country = c("US", "Germany", "India", "Brazil", "Russia"),
  population = c(339436159, 83628708, 1402617695, 208660842, 146371298)
)

five_global_cases <- five_global_cases %>%
  left_join(population, by = "country")

five_global_deaths <- five_global_deaths %>%
  left_join(population, by = "country")
```

```
# mutate data to look at cases and deaths per 100k based on countries population

cases_per_100k <-  five_global_cases %>%
  mutate(
    cases_per_100k = (cases * 100000 / population)
  )

deaths_per_100k <-  five_global_deaths %>%
```

```
  mutate(
    deaths_per_100k = (deaths * 100000 / population)
  )
```

```
ggplot(cases_per_100k, aes(x = date, y = cases_per_100k, color = country)) +
  geom_line(linewidth = 1) +
  geom_smooth(method = "lm", se = FALSE, linewidth = 1, linetype = "dashed") +
  labs(
    title = "COVID-19 Cumulative Cases per 100k",
    x = "Date",
    y = "Count"
  ) +
  scale_color_manual(values = c(
    "US" = "blue",
    "India" = "red",
    "Russia" = "pink",
    "Germany" = "purple",
    "Brazil" = "orange"
    )) +
  scale_y_continuous(labels = scales::comma) +
  theme_minimal() +
  theme(legend.title = element_blank())
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

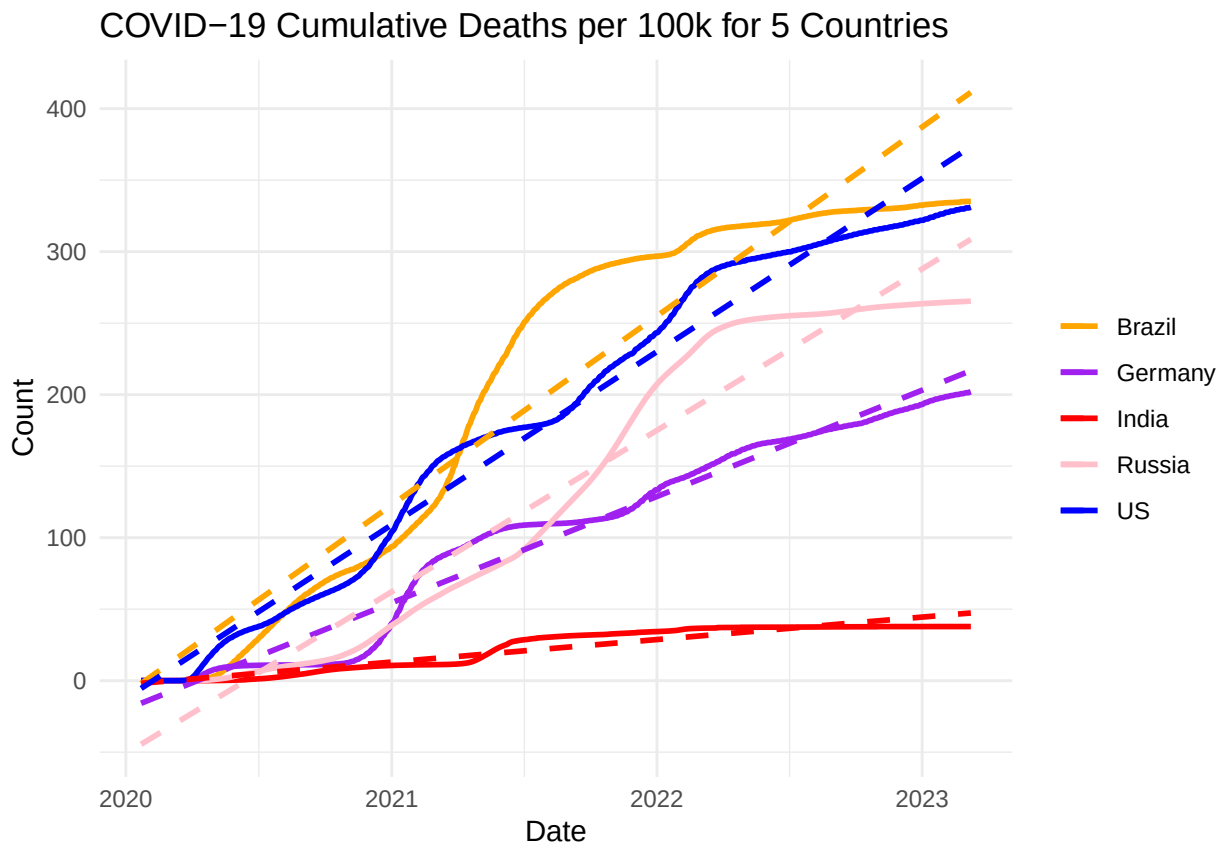


India is extremely low, almost non-existent. Brazil and Russia are relatively close to each other. The US is

double Russia, and Germany is triple Russia.
This data may tell us more about each country's data collection and COVID case reporting than it tells us about the spread of COVID in the country.
We could expect some variance in case totals from country to country due to factors like the prevalence of international travel, the efficiency of the government to communicate preventative measures, or the availability of vaccines. But this extreme difference between the five countries suggests biases in the data. Perhaps countries like India had much less availability of tests, therefore less case totals were reported or recorded.
The five countries do stay fairly close to the linear regression models, suggesting that reported case rates stayed relatively consistent. We can see some variations from these models, which could likely be attributed to causes like vaccine roll outs or new variants.

```
ggplot(deaths_per_100k, aes(x = date, y = deaths_per_100k, color = country)) +
  geom_line(linewidth = 1) +
  geom_smooth(method = "lm", se = FALSE, linewidth = 1, linetype = "dashed") +
  labs(
    title = "COVID-19 Cumulative Deaths per 100k for 5 Countries",
    x = "Date",
    y = "Count"
  ) +
  scale_color_manual(values = c(
    "US" = "blue",
    "India" = "red",
    "Russia" = "pink",
    "Germany" = "purple",
    "Brazil" = "orange"
    )) +
  scale_y_continuous(labels = scales::comma) +
  theme_minimal() +
  theme(legend.title = element_blank())
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



We would expect less bias in this data. While illness from COVID may be commonly unreported, deaths would likely still be recorded.But this data collection does still depend on how causes of death are attributed and reported in each country, so there is still a lot room for bias. Without testing, COVID tests could be attributed to another number of illnesses. Some countries may have also been motivated to keep their reported numbers low for political reasons.

All countries were somewhat close to their linear regression models. Each country at the end of the data set does appear to be slowing in cumulative deaths, each finishing with trends lower than their respective linear regression models, suggesting that new cases were slowing everywhere.

Once again, India is shockingly low. We could attribute this partially to a tangible reason, like a possible lack of travel between other countries. If less people are mobile, less disease would spread. The more likely explanation may be that many COVID deaths were missed. Deaths may have been attributed to other similar causes like the flu or pneumonia. This could make it seem as though India was less affected by COVID, when really they were just monitoring it in a different way.

Looking at the other four countries, Germany is significantly lower than Russia, the US and Brazil. It would be worth examining other factors like population age, availability of vaccines, and average distance to hospitals. There could still be problems with biases in data collection, but Germany may have had successful practices that would be worth studying, so that we could better prepare strategies for the next possible global pandemic.

```
sessionInfo()
```

```
## R version 4.4.2 (2024-10-31 ucrt)
## Platform: x86_64-w64-mingw32/x64
## Running under: Windows 11 x64 (build 26100)
##
## 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/Denver
## tzcode source: internal
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] broom_1.0.7     scales_1.3.0    lubridate_1.9.3 forcats_1.0.0
##  [5] stringr_1.5.1   dplyr_1.1.4     purrr_1.0.2     readr_2.1.5
##  [9] tidyr_1.3.1     tibble_3.2.1    ggplot2_3.5.1   tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
##  [1] utf8_1.2.4        generics_0.1.3    lattice_0.22-6    stringi_1.8.4
##  [5] hms_1.1.3         digest_0.6.37     magrittr_2.0.3    evaluate_1.0.1
##  [9] grid_4.4.2        timechange_0.3.0  fastmap_1.2.0     Matrix_1.7-1
## [13] backports_1.5.0   mgcv_1.9-1        cli_3.6.3         rlang_1.1.4
## [17] crayon_1.5.3      splines_4.4.2     bit64_4.5.2       munsell_0.5.1
## [21] withr_3.0.2       yaml_2.3.10       tools_4.4.2       parallel_4.4.2
## [25] tzdb_0.4.0        colorspace_2.1-1  curl_6.0.1        vctrs_0.6.5
## [29] R6_2.5.1          lifecycle_1.0.4   bit_4.5.0.1       vroom_1.6.5
## [33] pkgconfig_2.0.3   pillar_1.10.0     gtable_0.3.6      glue_1.8.0
## [37] xfun_0.49         tidyselect_1.2.1  rstudioapi_0.17.1 knitr_1.49
## [41] farver_2.1.2      nlme_3.1-166      htmltools_0.5.8.1 rmarkdown_2.29
## [45] labeling_0.4.3    compiler_4.4.2
```