COVID-19 Report

Anonymous

2025-04-16

R Markdown

This is an R Markdown document on the John Hopkins COVID19 dataset.Focusing on global cases and deaths, as well as US cases and deaths.

installing packages if necessary

Loading Required Packages

Loading Required Packages

```
# List of required packages
required_packages <- c("readr", "dplyr", "tidyr", "janitor", "stringr", "ggplot2", "scales", "sf", "rwo

# Install missing packages with a specified CRAN mirror
installed_packages <- required_packages %in% rownames(installed.packages())
if (any(!installed_packages)) install.packages(required_packages[!installed_packages], repos = "https:/

# Load the packages
lapply(required_packages, library, character.only = TRUE)
```

```
## [[1]]
## [1] "readr"     "stats"     "graphics"  "grDevices" "utils"     "datasets"
## [7] "methods"   "base"
##
## [[2]]
## [1] "dplyr"     "readr"     "stats"     "graphics"  "grDevices" "utils"
## [7] "datasets"  "methods"   "base"
##
## [[3]]
##  [1] "tidyr"     "dplyr"     "readr"     "stats"     "graphics"  "grDevices"
##  [7] "utils"     "datasets"  "methods"   "base"
##
## [[4]]
##  [1] "janitor"   "tidyr"     "dplyr"     "readr"     "stats"     "graphics"
##  [7] "grDevices" "utils"     "datasets"  "methods"   "base"
##
## [[5]]
##  [1] "stringr"   "janitor"   "tidyr"     "dplyr"     "readr"     "stats"
```

```
##  [7] "graphics"  "grDevices" "utils"     "datasets"  "methods"   "base"
##
## [[6]]
##  [1] "ggplot2"   "stringr"   "janitor"   "tidyr"     "dplyr"     "readr"
##  [7] "stats"     "graphics"  "grDevices" "utils"     "datasets"  "methods"
## [13] "base"
##
## [[7]]
##  [1] "scales"    "ggplot2"   "stringr"   "janitor"   "tidyr"     "dplyr"
##  [7] "readr"     "stats"     "graphics"  "grDevices" "utils"     "datasets"
## [13] "methods"   "base"
##
## [[8]]
##  [1] "sf"        "scales"    "ggplot2"   "stringr"   "janitor"   "tidyr"
##  [7] "dplyr"     "readr"     "stats"     "graphics"  "grDevices" "utils"
## [13] "datasets"  "methods"   "base"
##
## [[9]]
##  [1] "rworldmap" "sp"        "sf"        "scales"    "ggplot2"   "stringr"
##  [7] "janitor"   "tidyr"     "dplyr"     "readr"     "stats"     "graphics"
## [13] "grDevices" "utils"     "datasets"  "methods"   "base"
##
## [[10]]
##  [1] "kableExtra" "rworldmap"  "sp"         "sf"         "scales"
##  [6] "ggplot2"    "stringr"    "janitor"    "tidyr"      "dplyr"
## [11] "readr"      "stats"      "graphics"   "grDevices"  "utils"
## [16] "datasets"   "methods"    "base"
##
## [[11]]
##  [1] "cluster"    "kableExtra" "rworldmap"  "sp"         "sf"
##  [6] "scales"     "ggplot2"    "stringr"    "janitor"    "tidyr"
## [11] "dplyr"      "readr"      "stats"      "graphics"   "grDevices"
## [16] "utils"      "datasets"   "methods"    "base"
```

We will prepare to import the dataset

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

file_names <- c(
  "csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_confirmed_global.csv",
  "csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_deaths_global.csv",
  "csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_confirmed_US.csv",
  "csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_deaths_US.csv"
)

urls <- str_c(url_in, file_names)
```

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

We will now have to look at the data structure and decide what we can do to clean it up if necessary.

```
# Create a summary for each dataset
dataset_summary <- tibble(
  Dataset = c("Global Cases", "Global Deaths", "US Cases", "US Deaths"),
  Rows = c(nrow(global_cases), nrow(global_deaths), nrow(US_cases), nrow(US_deaths)),
  Columns = c(ncol(global_cases), ncol(global_deaths), ncol(US_cases), ncol(US_deaths)),
  First_5_Columns = c(
    paste(colnames(global_cases)[1:5], collapse = ", "),
    paste(colnames(global_deaths)[1:5], collapse = ", "),
    paste(colnames(US_cases)[1:5], collapse = ", "),
    paste(colnames(US_deaths)[1:5], collapse = ", ")
  )
)

# Display the summary table
kable(dataset_summary, caption = "Summary of Imported COVID-19 Datasets")
```

Table 1: Summary of Imported COVID-19 Datasets

Dataset	Rows	Columns	First_5_Columns
Global Cases	289	1147	Province/State, Country/Region, Lat, Long, 1/22/20
Global Deaths	289	1147	Province/State, Country/Region, Lat, Long, 1/22/20
US Cases	3342	1154	UID, iso2, iso3, code3, FIPS
US Deaths	3342	1155	UID, iso2, iso3, code3, FIPS

Cleaning the data

```
clean_covid_data <- function(df, region_col = "Country/Region", subregion_col = "Province/State") {

  df <- df %>%
    janitor::clean_names()

  region_col <- tolower(gsub("[^a-zA-Z0-9]", "_", region_col))
  subregion_col <- tolower(gsub("[^a-zA-Z0-9]", "_", subregion_col))

  df_long <- df %>%
    pivot_longer(
      cols = matches("^x?\\d{1,2}_\\d{1,2}_\\d{2}$"),
      names_to = "date",
      values_to = "cases"
    ) %>%
    mutate(
      date = gsub("^x", "", date),
      date = as.Date(date, format = "%m_%d_%y")
    )

  df_long <- df_long %>%
```

```
    mutate(
      location = ifelse(
        is.na(.data[[subregion_col]]) | .data[[subregion_col]] == "",
        .data[[region_col]],
        paste(.data[[subregion_col]], .data[[region_col]], sep = ", ")
      )
    ) %>%
    select(location, date, cases) %>%
    filter(!is.na(cases), cases >= 0)

  return(df_long)
}
```

Applying the cleaning function to our 4 datasets

```
# Clean all datasets using the reusable function
global_cases_clean <- clean_covid_data(global_cases, "Country/Region", "Province/State")
global_deaths_clean <- clean_covid_data(global_deaths, "Country/Region", "Province/State")
US_cases_clean <- clean_covid_data(US_cases, "Country_Region", "Province_State")
US_deaths_clean <- clean_covid_data(US_deaths, "Country_Region", "Province_State")
```

Looking at the clean data

```
# Create summary table for cleaned datasets
summary_table <- tibble(
  Dataset = c("Global Cases", "Global Deaths", "US Cases", "US Deaths"),
  Rows = c(nrow(global_cases_clean), nrow(global_deaths_clean), nrow(US_cases_clean), nrow(US_deaths_cl
  Date_Range = c(
    paste0(min(global_cases_clean$date), " to ", max(global_cases_clean$date)),
    paste0(min(global_deaths_clean$date), " to ", max(global_deaths_clean$date)),
    paste0(min(US_cases_clean$date), " to ", max(US_cases_clean$date)),
    paste0(min(US_deaths_clean$date), " to ", max(US_deaths_clean$date))
  ),
  Unique_Locations = c(
    n_distinct(global_cases_clean$location),
    n_distinct(global_deaths_clean$location),
    n_distinct(US_cases_clean$location),
    n_distinct(US_deaths_clean$location)
  )
)

# Display summary
kable(summary_table, caption = "Cleaned COVID-19 Dataset Summary")
```

Table 2: Cleaned COVID-19 Dataset Summary

Dataset	Rows	Date_Range	Unique_Locations
Global Cases	330327	2020-01-22 to 2023-03-09	289

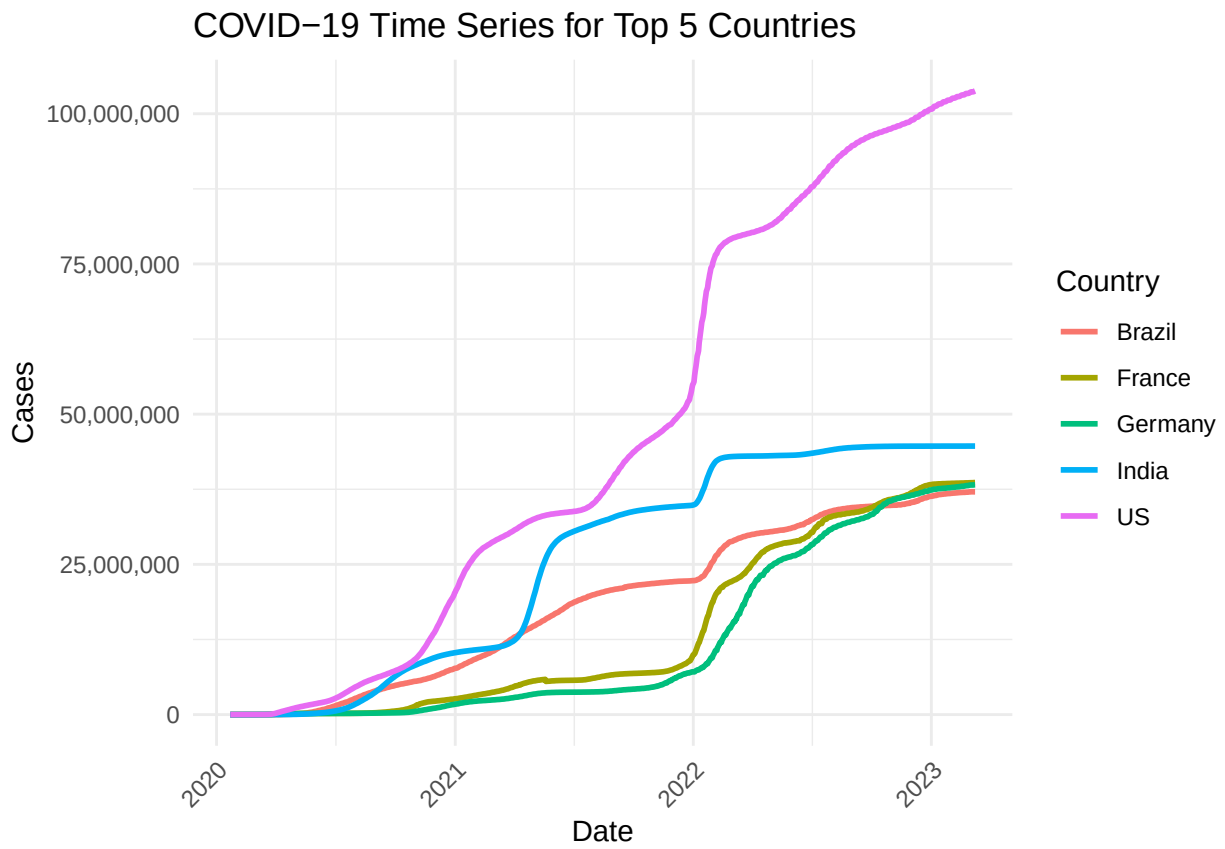
Dataset	Rows	Date_Range	Unique_Locations
Global Deaths	330327	2020-01-22 to 2023-03-09	289
US Cases	3819903	2020-01-22 to 2023-03-09	58
US Deaths	3819903	2020-01-22 to 2023-03-09	58

Looking at different data visualizations

Giving brief analysis along with images to what can be possibly gleaned from our clean dataset.

Time series for top 5 countries

```
library(ggplot2)

# Get the latest date from the cleaned global cases dataset
latest_date <- max(global_cases_clean$date)

# Filter for the top 5 countries based on the latest available case count
top_5_countries <- global_cases_clean %>%
  filter(date == latest_date) %>%
  arrange(desc(cases)) %>%
  slice_head(n = 5) %>%
  pull(location)

# Filter data for the top 5 countries
top_5_data <- global_cases_clean %>%
  filter(location %in% top_5_countries)

# Create the time series plot
ggplot(top_5_data, aes(x = date, y = cases, color = location)) +
  geom_line(size = 1) +
  labs(title = "COVID-19 Time Series for Top 5 Countries",
       x = "Date", y = "Cases",
       color = "Country") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_y_continuous(labels = scales::comma)
```

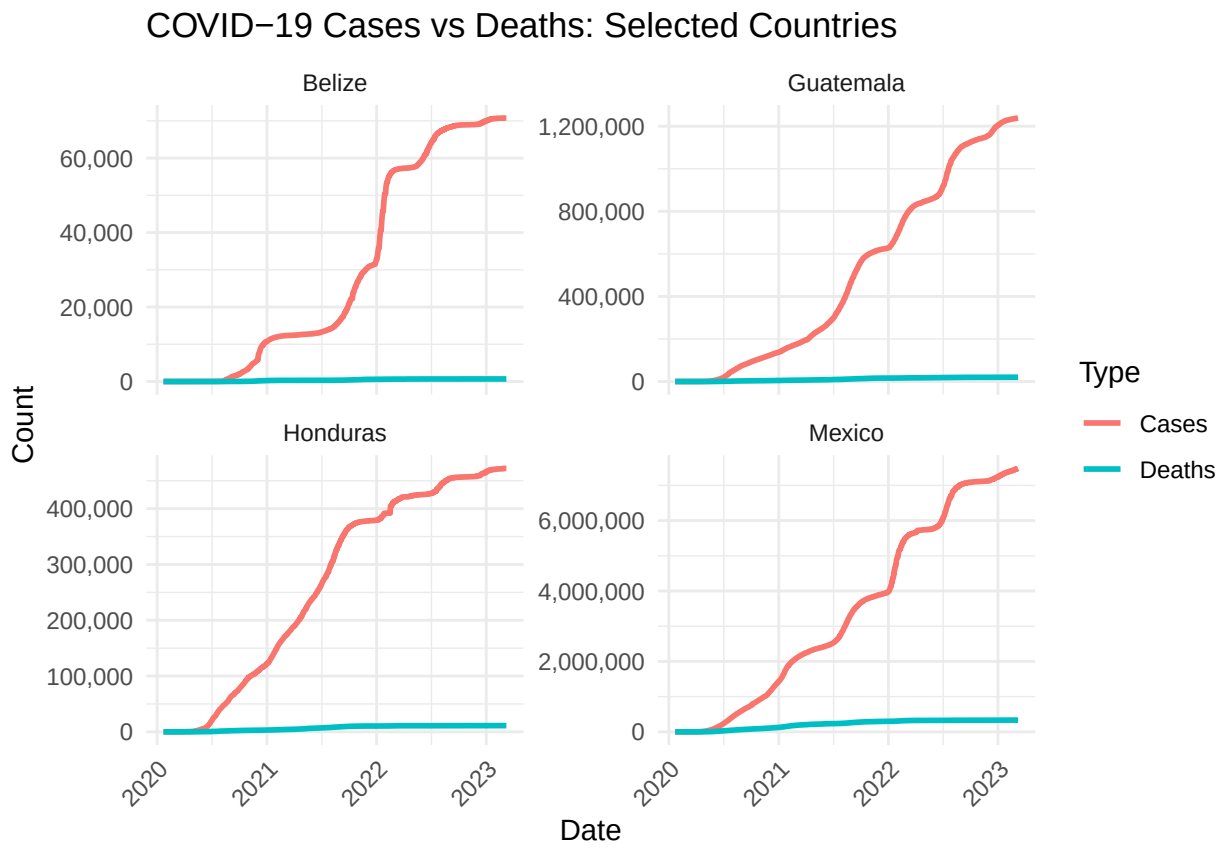


This visualization was just to quickly show us the countries that had the biggest impact when it came to infected individuals. We can see that the US had the largest number of cases over the period, followed by India. Brazil started out as the third largest but spikes in 2022 for Germany and France saw those two countries surpass it. However, due to the way the dataset was created, these could be cases of over-reporting , or simply areas with older populations and better health infrastructure so the number of cases will be high.

Time series for custom countries (cases vs deaths)

```r
library(ggplot2)
library(dplyr)
library(scales)

# Define your custom countries
my_countries <- c("Belize", "Guatemala", "Honduras", "Mexico")

# Filter and label global cases
cases_data <- global_cases_clean %>%
  filter(location %in% my_countries) %>%
  mutate(type = "Cases")

# Filter and label global deaths
deaths_data <- global_deaths_clean %>%
  filter(location %in% my_countries) %>%
  mutate(type = "Deaths")
```

```r
# Combine both
combined_data <- bind_rows(cases_data, deaths_data)

# Plot
ggplot(combined_data, aes(x = date, y = cases, color = type)) +
  geom_line(size = 1) +
  facet_wrap(~ location, scales = "free_y") +
  labs(
    title = "COVID-19 Cases vs Deaths: Selected Countries",
    x = "Date",
    y = "Count",
    color = "Type"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_y_continuous(labels = scales::comma)
```

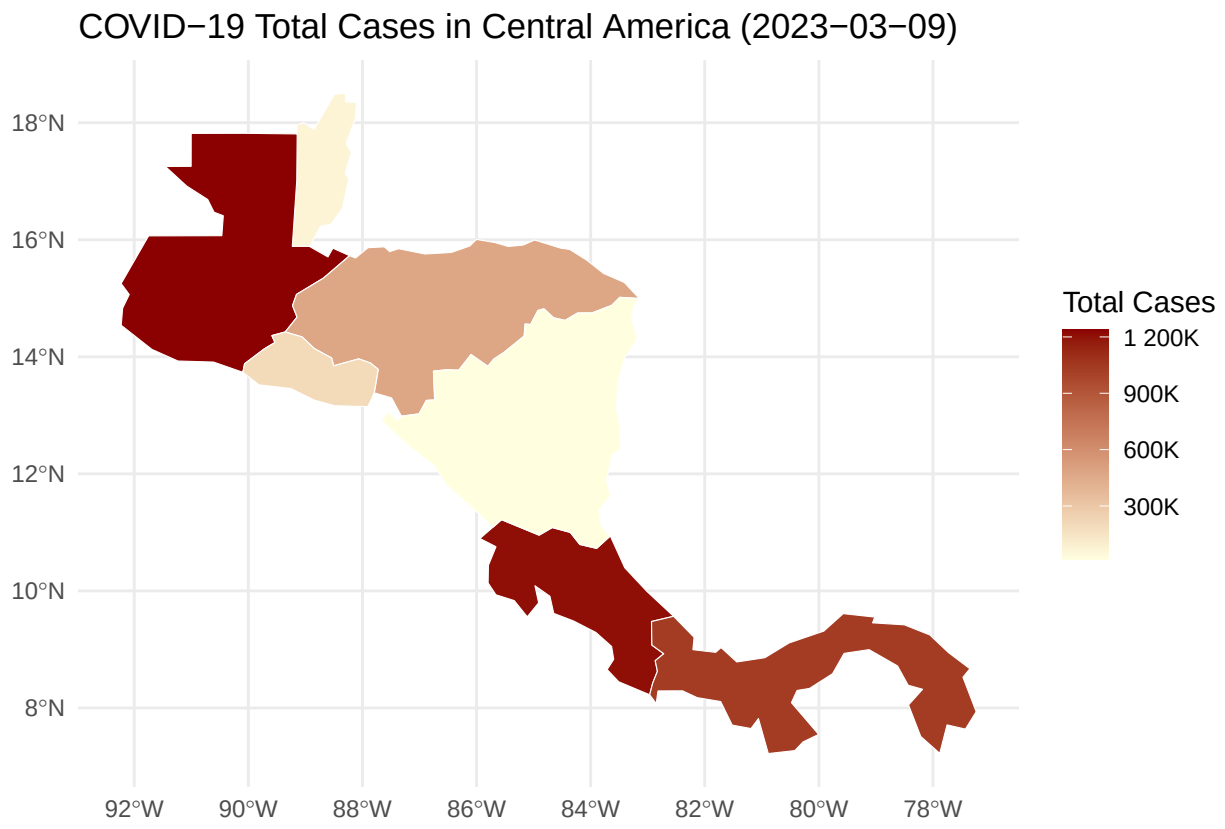



```r
##scale_y_continuous(labels = label_number(scale = 0.001, suffix = "K"))
```

I wanted to look at scenarios closer to home, I am from Belize so I wanted to look at how the immediate countries in the region were impacted by COVID-19. Vaccines were heavily pushed in my country (even though we would need data showing the strains and variants to properly surmise) so I did not expect the death rate to be too high, and the populations for the region are smaller except for Mexico.

Heat map of Central America

```
library(ggplot2)
library(dplyr)
library(sf)
library(rworldmap)
library(scales)

# List of Central American countries (add more if needed)
central_america_countries <- c("Belize", "Costa Rica", "El Salvador", "Guatemala", "Honduras", "Nicaragu

# Get shapefile data for world countries
world <- st_as_sf(rworldmap::getMap(resolution = 1))

# Filter world map data to include only Central America countries
central_america_map <- world %>%
  filter(NAME %in% central_america_countries)

# Get the latest date from global cases dataset
latest_date <- max(global_cases_clean$date)

# Summarize total cases by country for the latest date
central_america_cases <- global_cases_clean %>%
  filter(location %in% central_america_countries, date == latest_date) %>%
  group_by(location) %>%
  summarize(total_cases = sum(cases, na.rm = TRUE)) %>%
  rename(country = location)

# Merge the map data with COVID-19 cases data
central_america_map <- central_america_map %>%
  left_join(central_america_cases, by = c("NAME" = "country"))

# Plotting the heatmap
ggplot(central_america_map) +
  geom_sf(aes(fill = total_cases), color = "white") +
  scale_fill_continuous(low = "lightyellow", high = "darkred", name = "Total Cases",
                        labels = label_number(scale = 0.001, suffix = "K")) +
  labs(title = paste("COVID-19 Total Cases in Central America (", latest_date, ")", sep = "")) +
  theme_minimal() +
  theme(legend.position = "right")
```



Looking at the heat map of Central America, we can clearly see which countries had more cases. With Guatemala and Costa Rica represented by the darker colors, where as Belize and Nicaragua experienced the least cases. This can be due to many factors, such as vaccine acceptance, population and how long it took for the country to close down when COVID was first announced.

#Finding out the case fatality rate for Belize library(dplyr)

Get the latest date from the global cases dataset

latest_date <- max(global_cases_clean$date)

Filter data for Belize from both confirmed cases and deaths datasets

belize_cases_data <- global_cases_clean %>% filter(location == “Belize”, date == latest_date)

belize_deaths_data <- global_deaths_clean %>% filter(location == “Belize”, date == latest_date)

Summarize total cases and total deaths

belize_cases <- belize_cases_data %>% summarize(total_cases = sum(cases, na.rm = TRUE))

belize_deaths <- belize_deaths_data %>% summarize(total_deaths = sum(cases, na.rm = TRUE)) # “cases” is used for deaths

Calculate Case Fatality Rate (CFR)

belize_cfr <- (belize_deaths$total_deaths / belize_cases$total_cases) * 100

Output the Case Fatality Rate for Belize

The case fatality rate for Belize is 0.97%.

By looking at the sum of all deaths in the country over the sum of all cases, we can see an approximate fatality rate. This is just to show another type of information we can obtain with the data provided.

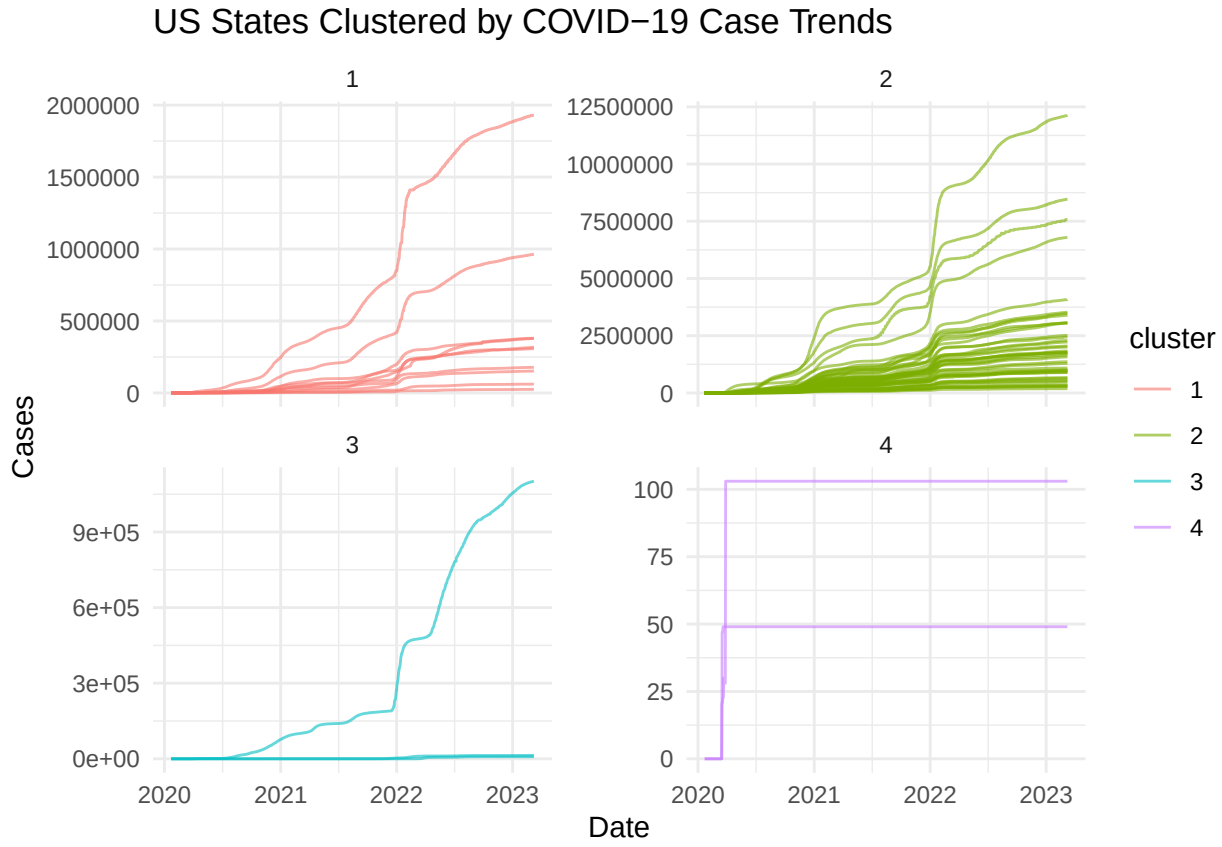
A model for the COVID19 database

library(dplyr) library(tidyr) library(ggplot2) library(cluster)

Step 1: Reshape and normalize time series data

```
library(dplyr)
library(tidyr)
library(ggplot2)
library(cluster)

# Step 1: Reshape and normalize time series data
us_cases_long <- US_cases %>%
  rename(state = `Province_State`) %>%
  select(state, matches("^\\d+/\\d+/\\d+$")) %>%
  pivot_longer(cols = -state, names_to = "date", values_to = "cases") %>%
  mutate(date = as.Date(date, format = "%m/%d/%y")) %>%
  group_by(state, date) %>%
  summarize(cases = sum(cases, na.rm = TRUE), .groups = 'drop')

us_cases_wide <- us_cases_long %>%
  pivot_wider(names_from = date, values_from = cases, values_fill = 0)

us_cases_matrix <- as.matrix(us_cases_wide[,-1])
us_cases_scaled <- t(scale(t(us_cases_matrix)))
rownames(us_cases_scaled) <- us_cases_wide$state

# Step 2: K-means clustering
set.seed(123)
k <- 4
clusters <- kmeans(us_cases_scaled, centers = k, nstart = 25)

# Step 3: Join cluster info back
cluster_df <- data.frame(state = rownames(us_cases_scaled),
                         cluster = as.factor(clusters$cluster))

us_cases_clustered <- us_cases_long %>%
  left_join(cluster_df, by = "state")
```

```r
# Step 4: Plot time series by cluster
ggplot(us_cases_clustered, aes(x = date, y = cases, group = state, color = cluster)) +
  geom_line(alpha = 0.6) +
  facet_wrap(~ cluster, scales = "free_y") +
  labs(title = "US States Clustered by COVID-19 Case Trends",
       x = "Date", y = "Cases") +
  theme_minimal()
```




```r
# Show states in each cluster as bullet points
cluster_df %>%
  arrange(cluster) %>%
  group_by(cluster) %>%
  summarize(states = list(state)) %>%
  rowwise() %>%
  mutate(output = paste0("### Cluster ", cluster, "\n",
                         paste0("- ", states, collapse = "\n"))) %>%
  pull(output) %>%
  cat(sep = "\n\n")
```

```
## ### Cluster 1
## - Alaska
## - District of Columbia
## - Guam
## - Hawaii
```

```
## - Maine
## - New Hampshire
## - Oregon
## - Vermont
## - Virgin Islands
## - Washington
##
## ### Cluster 2
## - Alabama
## - Arizona
## - Arkansas
## - California
## - Colorado
## - Connecticut
## - Delaware
## - Florida
## - Georgia
## - Idaho
## - Illinois
## - Indiana
## - Iowa
## - Kansas
## - Kentucky
## - Louisiana
## - Maryland
## - Massachusetts
## - Michigan
## - Minnesota
## - Mississippi
## - Missouri
## - Montana
## - Nebraska
## - Nevada
## - New Jersey
## - New Mexico
## - New York
## - North Carolina
## - North Dakota
## - Ohio
## - Oklahoma
## - Pennsylvania
## - Rhode Island
## - South Carolina
## - South Dakota
## - Tennessee
## - Texas
## - Utah
## - Virginia
## - West Virginia
## - Wisconsin
## - Wyoming
##
## ### Cluster 3
## - American Samoa
```

```
## - Northern Mariana Islands
## - Puerto Rico
##
## ### Cluster 4
## - Diamond Princess
## - Grand Princess
```

Looking at the models, we can group the different American states to see which ones had similar case trends, and after further research we can determine if these were based on similar policies, population size, vaccine rate etc.

Conclusion and bias statement

The John Hopkins COVID-19 dataset is a useful data for looking at how the world and the US was affected by the pandemic. It is vast, and for that reason majority of the visualizations used, I centered around the region I live since that is more relatable, however for the purposes of reproducibility, we can always adjust the countries chosen for future iterations of research. There are many biases that are to be expected when dealing with a global dataset. Firstly, different countries may have reporting bias as to under-reporting or over-reporting cases for political and economic reasons, even delayed reporting due to holidays can cause artificial dips or spikes. Limited testing may have been done in certain countries, or only symptomatic patients may have been tested.Each country would have their own demographic, so we know that populations with an older population was more at risk. The access to different types of vaccines and variant waves was not directly captured. The cases and deaths variables may not tell the a holistic picture of being directly affected by COVID-19. All these and many more can shape how our trends look , and can bias our models and plots.