Covid 19 - Data Analysis

2024-04-03

```
knitr::opts_chunk$set(echo = TRUE)
```

Introduction

This report aims to provide an analysis of the Covid 19 data. Our goal is to understand and identify the possible trends, patterns, and any underlying issues within the data.

We will also do our best to acknowledge potential biases in data collection and analysis, aiming for an objective analysis.

Data Import and Description

```
# Load necessary libraries
library(reshape2)
library(readr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

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

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

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

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats 1.0.0     v stringr 1.5.1
## v ggplot2 3.5.0     v tibble  3.2.1
## v purrr   1.0.2     v tidyr   1.3.1

## -- 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
```

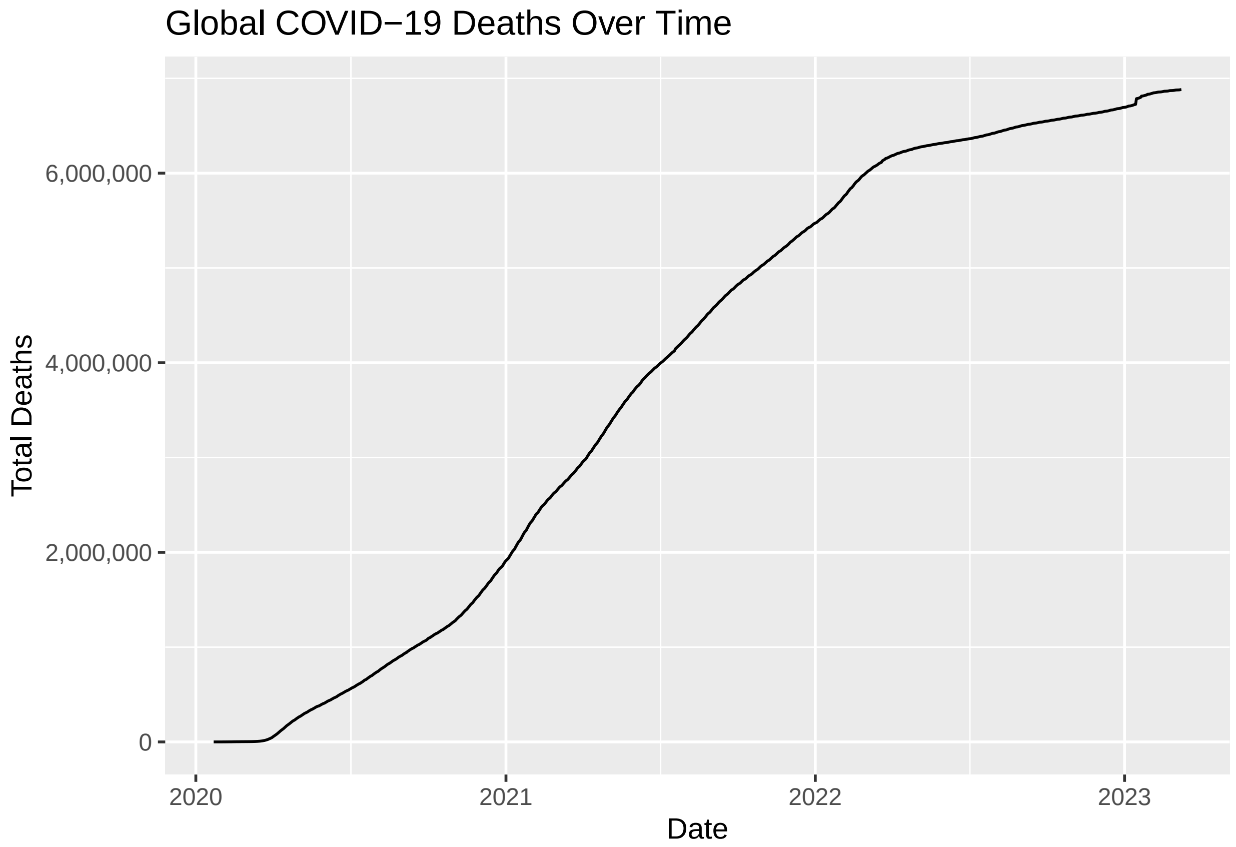
```
# Import dataset
url_COVID <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_
covid_death_data <- read.csv(url_COVID)

# Display the structure and summary of the dataset using the new variable name
str(covid_death_data)
```

```
## 'data.frame':    289 obs. of  1147 variables:
##  $ Province.State: chr  "" "" "" "" ...
##  $ Country.Region: chr  "Afghanistan" "Albania" "Algeria" "Andorra" ...
##  $ Lat           : num  33.9 41.2 28 42.5 -11.2 ...
##  $ Long          : num  67.71 20.17 1.66 1.52 17.87 ...
##  $ X1.22.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X1.23.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X1.24.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X1.25.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X1.26.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X1.27.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X1.28.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X1.29.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X1.30.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X1.31.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.1.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.2.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.3.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.4.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.5.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.6.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.7.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.8.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.9.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.10.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.11.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.12.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.13.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.14.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.15.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.16.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.17.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.18.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.19.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.20.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.21.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.22.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.23.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.24.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.25.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.26.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.27.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.28.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X2.29.20      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X3.1.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X3.2.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X3.3.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X3.4.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X3.5.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X3.6.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X3.7.20       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ X3.8.20       : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ X3.9.20       : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ X3.10.20      : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ X3.11.20      : int  0 1 0 0 0 0 0 1 0 0 ...
##  $ X3.12.20      : int  0 1 1 0 0 0 0 1 0 0 ...
##  $ X3.13.20      : int  0 1 2 0 0 0 0 2 0 0 ...
##  $ X3.14.20      : int  0 1 3 0 0 0 0 2 0 0 ...
##  $ X3.15.20      : int  0 1 4 0 0 0 0 2 0 0 ...
##  $ X3.16.20      : int  0 1 4 0 0 0 0 2 0 0 ...
##  $ X3.17.20      : int  0 1 4 0 0 0 0 2 0 0 ...
##  $ X3.18.20      : int  0 2 7 0 0 0 0 2 0 0 ...
##  $ X3.19.20      : int  0 2 9 0 0 0 0 3 0 0 ...
##  $ X3.20.20      : int  0 2 11 0 0 0 0 3 0 0 ...
##  $ X3.21.20      : int  0 2 15 0 0 0 0 4 0 0 ...
##  $ X3.22.20      : int  0 2 17 1 0 0 0 4 0 0 ...
##  $ X3.23.20      : int  1 4 17 1 0 0 0 4 0 0 ...
##  $ X3.24.20      : int  1 5 19 1 0 0 0 6 0 0 ...
##  $ X3.25.20      : int  1 5 21 1 0 0 0 8 0 0 ...
##  $ X3.26.20      : int  2 6 25 3 0 0 0 9 1 0 ...
##  $ X3.27.20      : int  2 8 26 3 0 0 0 13 1 0 ...
##  $ X3.28.20      : int  2 10 29 3 0 0 0 18 1 0 ...
##  $ X3.29.20      : int  4 10 31 6 2 0 0 19 3 0 ...
##  $ X3.30.20      : int  4 11 35 8 2 0 0 23 3 1 ...
##  $ X3.31.20      : int  4 15 44 12 2 0 0 27 3 1 ...
##  $ X4.1.20       : int  4 15 58 14 2 0 0 28 4 1 ...
##  $ X4.2.20       : int  4 16 86 15 2 0 0 36 7 1 ...
##  $ X4.3.20       : int  5 17 105 16 2 0 0 39 7 1 ...
##  $ X4.4.20       : int  5 20 130 17 2 0 0 43 7 2 ...
##  $ X4.5.20       : int  7 20 152 18 2 0 0 44 7 2 ...
##  $ X4.6.20       : int  7 21 173 21 2 0 0 48 8 2 ...
##  $ X4.7.20       : int  11 22 193 22 2 0 1 56 8 2 ...
##  $ X4.8.20       : int  14 22 205 23 2 0 2 63 9 2 ...
##  $ X4.9.20       : int  15 23 235 25 2 0 2 72 10 2 ...
##  $ X4.10.20      : int  15 23 256 26 2 0 2 82 12 2 ...
##  $ X4.11.20      : int  15 23 275 26 2 0 2 83 13 2 ...
##  $ X4.12.20      : int  18 23 293 29 2 0 2 90 13 2 ...
##  $ X4.13.20      : int  19 23 313 29 2 0 2 97 14 2 ...
##  $ X4.14.20      : int  22 24 326 31 2 0 2 102 16 2 ...
##  $ X4.15.20      : int  25 25 336 33 2 0 2 111 17 3 ...
##  $ X4.16.20      : int  29 26 348 33 2 0 3 115 18 3 ...
##  $ X4.17.20      : int  30 26 364 35 2 0 3 123 19 3 ...
##  $ X4.18.20      : int  30 26 367 35 2 0 3 129 20 3 ...
##  $ X4.19.20      : int  30 26 375 36 2 0 3 132 20 3 ...
##  $ X4.20.20      : int  33 26 384 37 2 0 3 136 22 3 ...
##  $ X4.21.20      : int  36 26 392 37 2 0 3 147 24 3 ...
##  $ X4.22.20      : int  36 27 402 37 2 0 3 152 24 3 ...
##  $ X4.23.20      : int  40 27 407 37 2 0 3 165 24 3 ...
##  $ X4.24.20      : int  40 27 415 40 2 0 3 176 27 3 ...
##  $ X4.25.20      : int  43 27 419 40 2 0 3 185 28 3 ...
##   [list output truncated]
```

```
covid_death_data_long <- pivot_longer(covid_death_data,
                                      cols = starts_with("X"),
                                      names_to = "Date",
                                      values_to = "Deaths")

# Convert the Date from its current format (e.g., X1.22.20) to a proper Date format

covid_death_data_long$Date <- gsub("X", "",
                                   covid_death_data_long$Date)
covid_death_data_long$Date <- gsub("\\.", "-",
                                   covid_death_data_long$Date)
covid_death_data_long$Date <- as.Date(covid_death_data_long$Date,
                                      format = "%m-%d-%y")

# View the transformed data
head(covid_death_data_long)
```

```
## # A tibble: 6 x 6
##   Province.State Country.Region   Lat  Long Date       Deaths
##   <chr>          <chr>          <dbl> <dbl> <date>      <int>
## 1 ""             Afghanistan     33.9  67.7 2020-01-22      0
## 2 ""             Afghanistan     33.9  67.7 2020-01-23      0
## 3 ""             Afghanistan     33.9  67.7 2020-01-24      0
## 4 ""             Afghanistan     33.9  67.7 2020-01-25      0
## 5 ""             Afghanistan     33.9  67.7 2020-01-26      0
## 6 ""             Afghanistan     33.9  67.7 2020-01-27      0
```

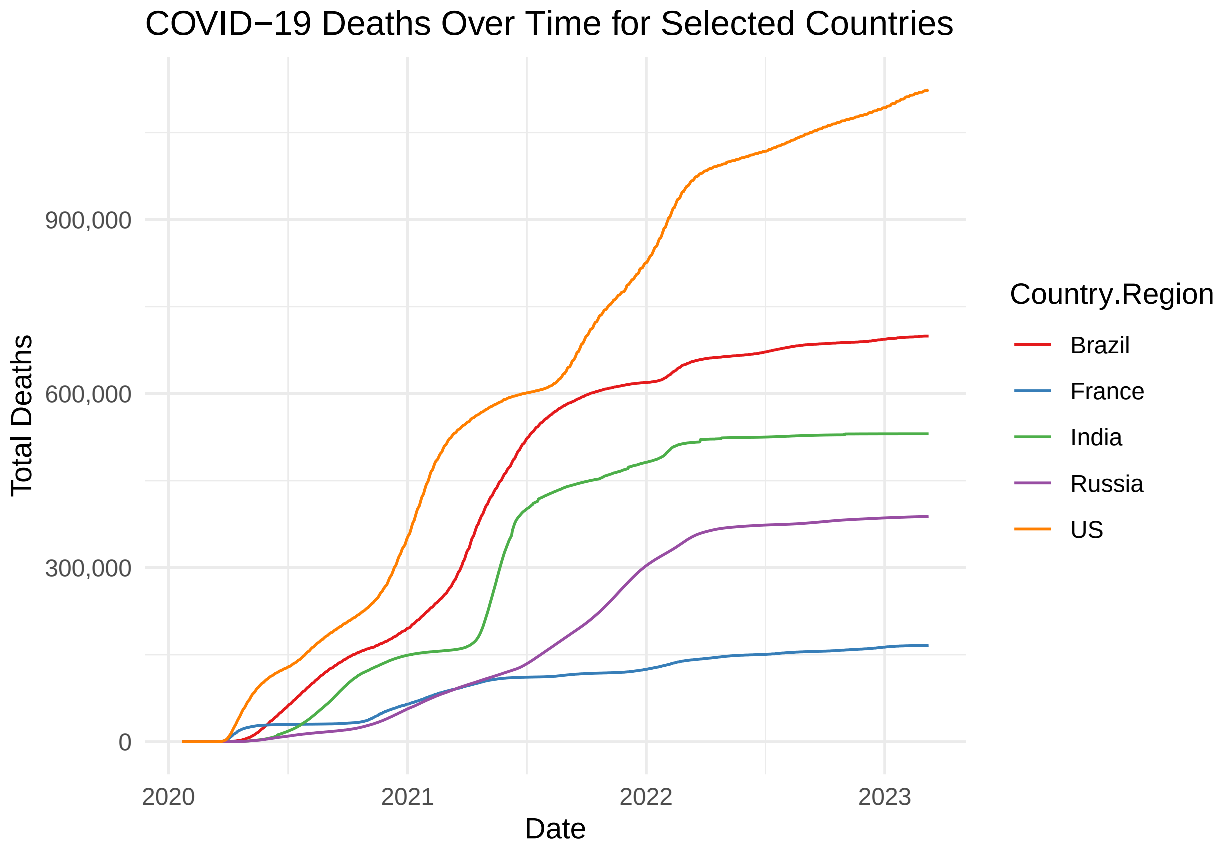
```
covid_death_data_long %>%
  group_by(Date) %>%
  summarise(TotalDeaths = sum(Deaths, na.rm = TRUE)) %>%
  ggplot(aes(x = Date, y = TotalDeaths)) +
  geom_line() +
  scale_y_continuous(labels = function(x) format(x, big.mark = ",",
                                                 scientific = FALSE)) +
  labs(title = "Global COVID-19 Deaths Over Time", x = "Date",
       y = "Total Deaths")
```



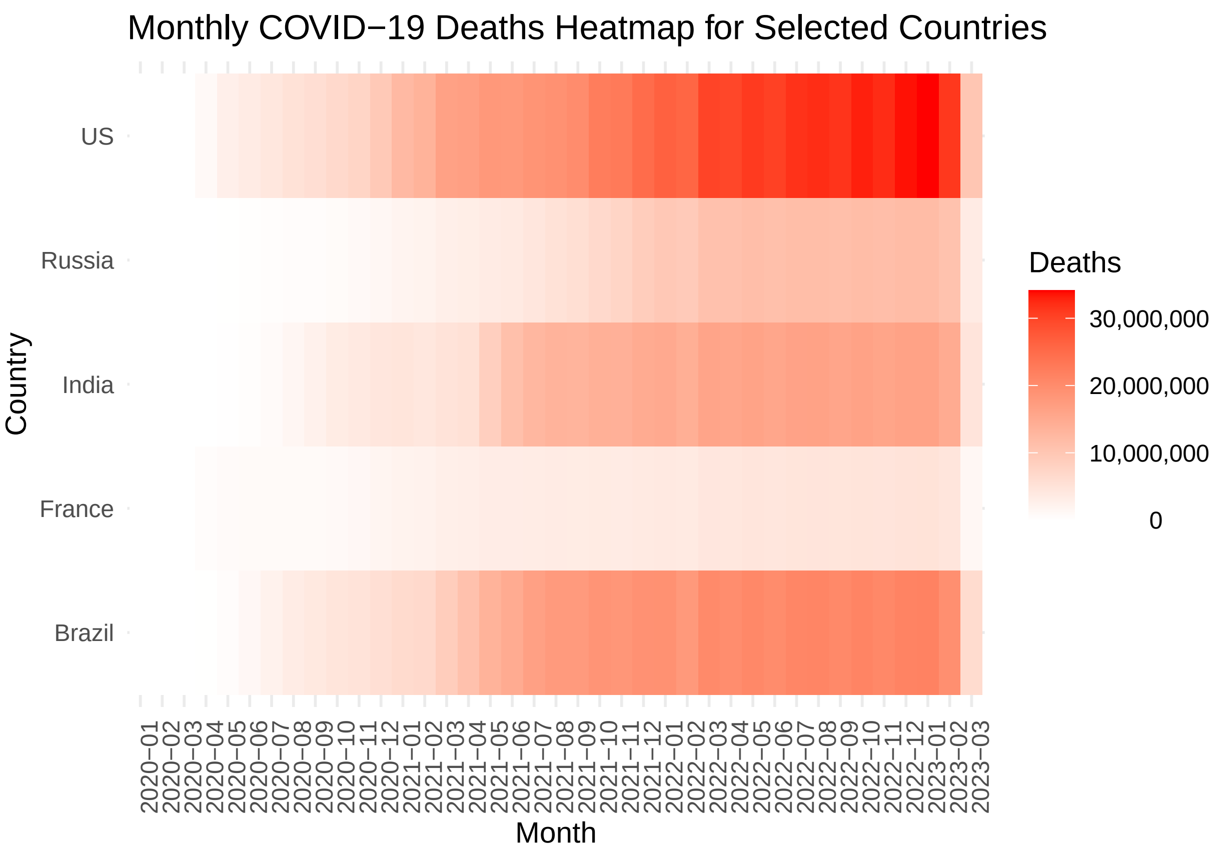
```r
selected_countries <- covid_death_data_long %>%
  filter(`Country.Region` %in% c("US", "France", "Brazil",
                                 "India", "Russia")) %>%

  group_by(Date, `Country.Region`) %>%
  summarise(TotalDeaths = sum(Deaths, na.rm = TRUE))
```

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

```r
ggplot(selected_countries, aes(x = Date, y = TotalDeaths,
                               color = `Country.Region`)) +
  geom_line() +
  scale_y_continuous(labels = function(x) format(x, big.mark = ",",
                                                 scientific = FALSE)) +
  labs(title = "COVID-19 Deaths Over Time for Selected Countries",
       x = "Date",
       y = "Total Deaths") +
  theme_minimal() +
  scale_color_brewer(palette = "Set1")
```



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

# Filter for selected countries
selected_deaths <- covid_death_data_long %>%
  filter(`Country.Region` %in% c("US", "France", "Brazil",
                                 "India", "Russia")) %>%
  mutate(Month = format(Date, "%Y-%m")) %>%
  group_by(`Country.Region`, Month) %>%
  summarise(TotalDeaths = sum(Deaths, na.rm = TRUE), .groups = 'drop')

# Pivot to wide format specifically for heatmap visualization
monthly_deaths_wide <- selected_deaths %>%
  pivot_wider(names_from = Month, values_from = TotalDeaths,
              values_fill = list(TotalDeaths = 0))

heatmap_data <- melt(monthly_deaths_wide, id.vars = 'Country.Region')

ggplot(heatmap_data, aes(x = variable, y = `Country.Region`, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "white", high = "red",
                      labels = function(x) format(x, big.mark = ",",
                                                  scientific = FALSE)) +
  labs(title = "Monthly COVID-19 Deaths Heatmap for Selected Countries",
       x = "Month", y = "Country", fill = "Deaths") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

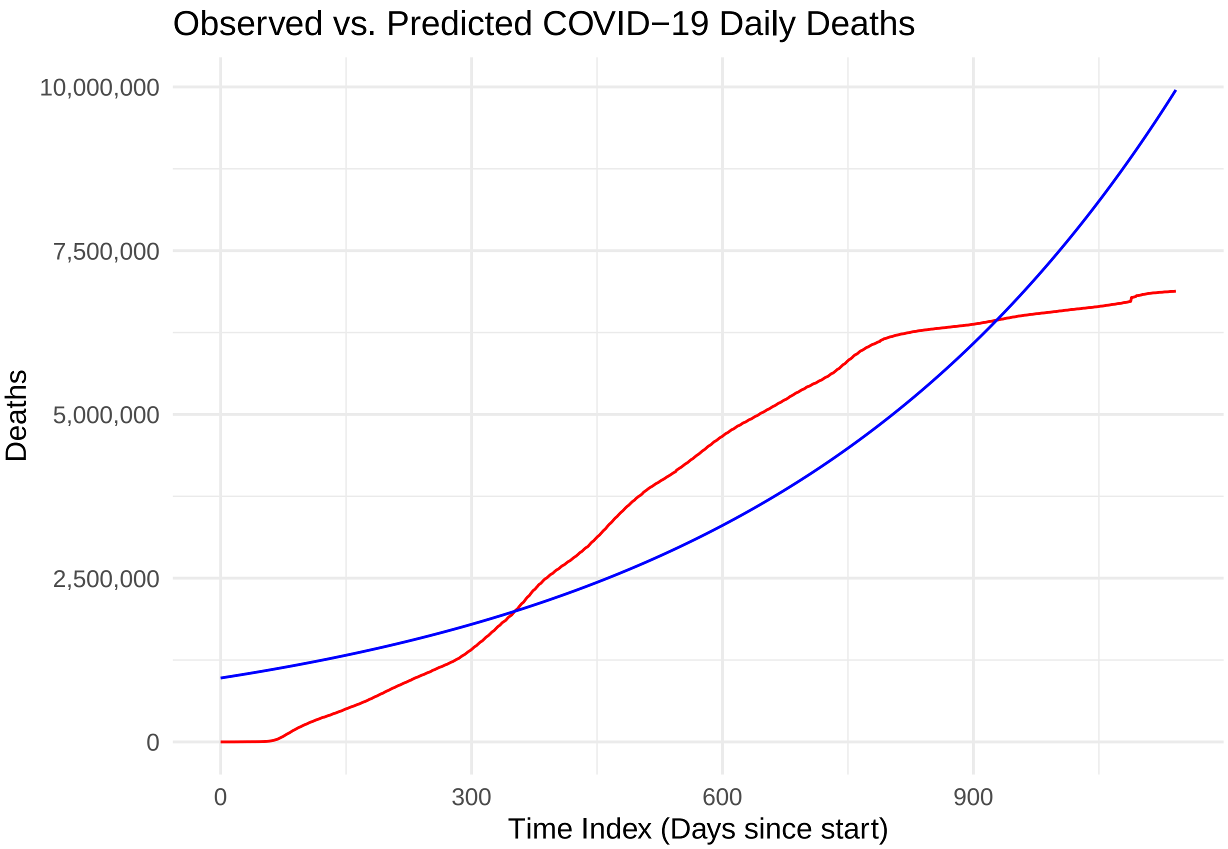


Monthly COVID−19 Deaths Heatmap for Selected Countries

```r
global_daily_deaths <- covid_death_data_long %>%
  group_by(Date) %>%
  summarise(TotalDeaths = sum(Deaths, na.rm = TRUE), .groups = 'drop')

global_daily_deaths$TimeIndex <- as.numeric(global_daily_deaths$Date - min(global_daily_deaths$Date))

# Fitting a Poisson GLM
glm_model <- glm(TotalDeaths ~ TimeIndex, family = poisson,
                 data = global_daily_deaths)

summary(glm_model)
```

```
##
## Call:
## glm(formula = TotalDeaths ~ TimeIndex, family = poisson, data = global_daily_deaths)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 1.379e+01  4.276e-05  322477   <2e-16 ***
## TimeIndex   2.034e-03  5.167e-08   39365   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 2264726437  on 1142  degrees of freedom
## Residual deviance:  509259551  on 1141  degrees of freedom
## AIC: 509278192
##
## Number of Fisher Scoring iterations: 5
```

```
global_daily_deaths$PredictedDeaths <- predict(glm_model, type = "response")

ggplot(global_daily_deaths, aes(x = TimeIndex)) +
  geom_line(aes(y = TotalDeaths), colour = "red") +
  geom_line(aes(y = PredictedDeaths), colour = "blue") +
  scale_y_continuous(labels = function(x) format(x,big.mark = ",",
                                                  scientific = FALSE)) +
  labs(title = "Observed vs. Predicted COVID-19 Daily Deaths",
       x = "Time Index (Days since start)", y = "Deaths") +
  theme_minimal()
```

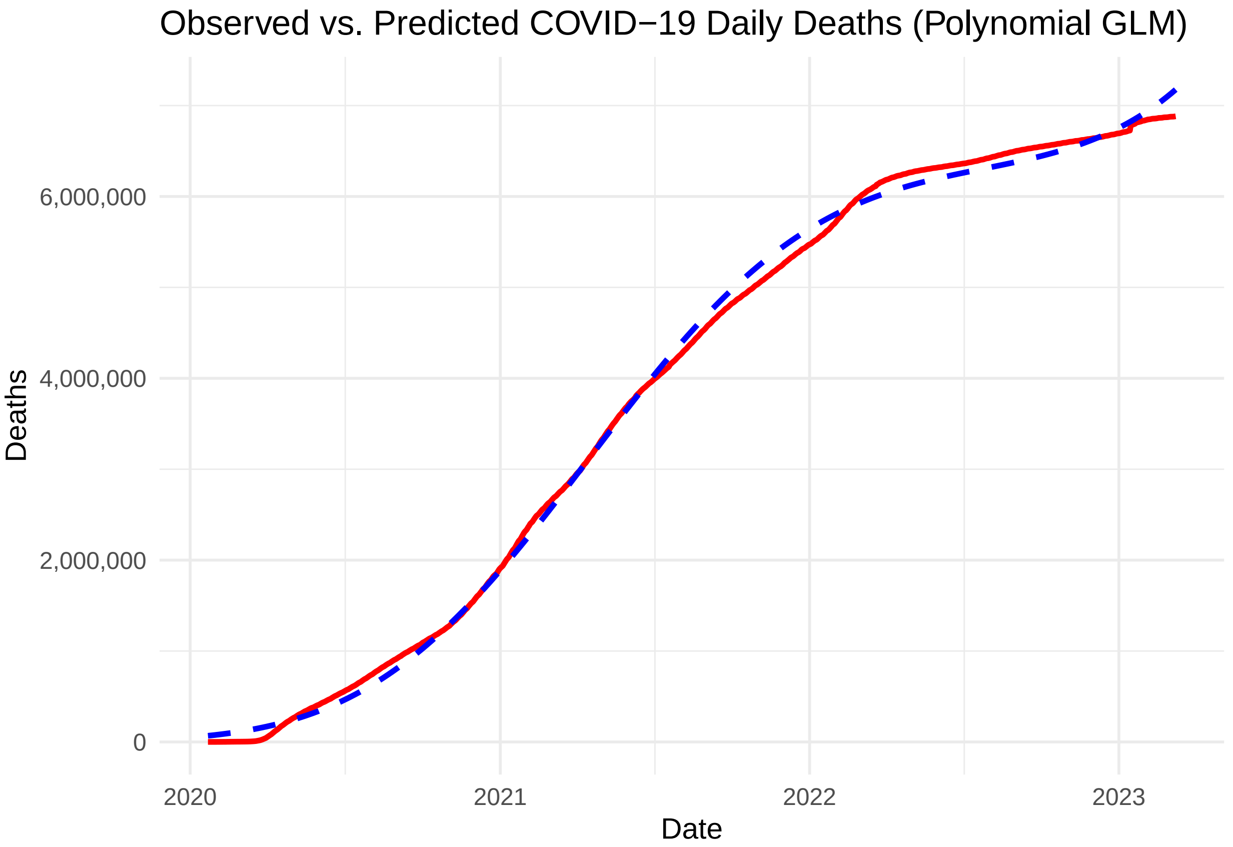


```
glm_poly_model <- glm(TotalDeaths ~ poly(TimeIndex, 3),
                      family = poisson, data = global_daily_deaths)

summary(glm_poly_model)
```

```
##
## Call:
## glm(formula = TotalDeaths ~ poly(TimeIndex, 3), family = poisson,
##     data = global_daily_deaths)
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)           1.470e+01  2.910e-05  505136   <2e-16 ***
## poly(TimeIndex, 3)1  3.754e+01  1.290e-03   29109   <2e-16 ***
## poly(TimeIndex, 3)2 -1.886e+01  1.120e-03  -16848   <2e-16 ***
## poly(TimeIndex, 3)3  5.248e+00  7.672e-04    6841   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 2264726437  on 1142  degrees of freedom
## Residual deviance:   17747311  on 1139  degrees of freedom
## AIC: 17765956
##
## Number of Fisher Scoring iterations: 5
```

```
global_daily_deaths$PredictedDeathsPoly <- predict(glm_poly_model,
                                                    type = "response")

# Plotting observed vs. predicted deaths
ggplot(global_daily_deaths, aes(x = Date)) +
  geom_line(aes(y = TotalDeaths), colour = "red", size = 1) +
  geom_line(aes(y = PredictedDeathsPoly), colour = "blue",
            linetype = "dashed", size = 1) +
  scale_y_continuous(labels = function(x) format(x, big.mark = ",",
                                                  scientific = FALSE)) +
  labs(title = "Observed vs. Predicted COVID-19 Daily Deaths (Polynomial GLM)",
       x = "Date", y = "Deaths") +
  theme_minimal()
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



Analysis Conclusion

The polynomial GLM demonstrates a statistically significant relationship between time (represented as TimeIndex) and the total number of daily deaths due to COVID-19 globally. The model includes polynomial terms up to the third degree, which allows for capturing non-linear trends in the data.

Intercept and Polynomial Terms: All coefficients are highly significant, as indicated by their p-values (<2e-16). This suggests a strong non-linear relationship between time and the number of deaths.

Model Fit: The inclusion of polynomial terms significantly improves the model fit compared to a simple linear model, as indicated by the large coefficients and their statistical significance. The residual deviance of 17,747,311 on 1139 degrees of freedom, while still substantial, represents a marked improvement over the null deviance of 2,264,726,437 on 1142 degrees of freedom.

AIC Value: The Akaike Information Criterion (AIC) of 17,765,956 provides a measure for comparing the model with alternative models, where a lower AIC indicates a better fit given the number of parameters.

Additional Questions

Model Generalization: How well does this model generalize to new data, especially given potential changes in the pandemic's dynamics over time?

Further Non-linear Trends: Are there other non-linear trends or patterns (e.g., periodicity, spikes) not captured by the current model?

Impact of Interventions: How do non-pharmaceutical interventions (lockdowns, mask mandates) and vaccination rollouts impact the trend and magnitude of daily deaths?

Potential Bias

Data Reporting: Inconsistencies in how deaths are reported across different countries and regions may introduce bias. For instance, some countries might have underreported deaths, especially early in the pandemic due to limited testing capabilities.

Changes in Testing and Diagnosis: Over time, the criteria for diagnosing COVID-19, as well as testing capabilities, have evolved, which could affect the number of reported deaths.

Overfitting: While polynomial models can capture complex relationships, there's a risk of overfitting, especially if higher-degree terms are used without sufficient justification from the data. Overfitting would make the model less generalizable to future data.

Simplicity of the Model: The model focuses solely on time as a predictor and does not account for other potentially influential factors such as population density, healthcare system strength, public health policies, and social behavior changes over time. This simplification might lead to biased or oversimplified interpretations of the pandemic's dynamics.