Corona-virus Analysis of Johns Hopkins Data set

James Greenwood

2024-06-24

Corona-virus Analysis of Johns Hopkins Data set

James Greenwood

2024-06-24

Description of the Data Resource

The Johns Hopkins Corona-virus Resource Center established a new standard for infectious disease tracking by publicly providing pandemic data in near real-time. It began on January 22, 2020, as the COVID-19 Dashboard, operated by the Center for Systems Science and Engineering (CSSE) and the Applied Physics Laboratory. This is the dataset that will be used for this analysis. The Corona-virus Resource Center ceased its data collection as of March 2022. Further information can be found at this *website.*

Objective

The objective of this analysis is to explore the COVID-19 dataset from Johns Hopkins University. The data set contains information on confirmed cases, deaths, and recoveries for countries around the world. The analysis will focus on visualizing the data, identifying trends, and gaining insights into the spread of the virus.

Data Collection

The dataset used in this analysis is sourced from the Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE) COVID-19 Data Repository. We are concerned with the time series data for confirmed cases, deaths, and recoveries at the global level and for the United States. This data can be found at the following GitHub *location.*

Data Description

The following is a list of the data sets that we downloaded from the Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE) COVID-19 Data Repository:

- global_confirmed_transformed: Time series data for confirmed COVID-19 cases at the global level.
- global_deaths: Time series data for COVID-19 deaths at the global level.
- global_recovered: Time series data for COVID-19 recoveries at the global level.
- us_confirmed: Time series data for confirmed COVID-19 cases in the United States.
- us_deaths: Time series data for COVID-19 deaths in the United States.

```
knitr::opts_chunk$set(echo = TRUE, tidy.opts = list(width.cutoff = 80), tidy
= TRUE)

# URL path for the permalink to the raw data on GitHub
raw_url <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-
19/4360e50239b4eb6b22f3a1759323748f36752177/csse_covid_19_data/csse_covid_19_
time_series/"

# File names
file_names <- c("time_series_covid19_confirmed_global.csv",
"time_series_covid19_deaths_global.csv",
    "time_series_covid19_recovered_global.csv",
"time_series_covid19_confirmed_US.csv",
    "time_series_covid19_deaths_US.csv")

# Function for reading files safely
read_csv_safe <- function(url) {
    tryCatch(read_csv(url), error = function(e) {
        warning(paste("Error reading file:", c(e, url)))
        return(NULL)
    })
}

# Concatenate URL and file name
urls <- str_c(raw_url, file_names)

# Read Files
global_confirmed <- read_csv_safe(urls[1])
global_deaths <- read_csv_safe(urls[2])
global_recovered <- read_csv_safe(urls[3])
us_confirmed <- read_csv_safe(urls[4])
us_deaths <- read_csv_safe(urls[5])
```

Data Overview

We have loaded the data into data frames for further analysis. Let’s examine the structure of the data sets to understand the columns and the first few rows of each data frame. The global data sets include global_confirmed, global_deaths, and global_recovered, while the US data sets include us_confirmed and us_deaths.

The title of each table displays the number of columns and rows in the dataset. The U.S. data sets contain case and death counts, with the us_deaths dataset also including the population of each county in the U.S. The extensive list of reporting dates, which contain the daily case counts, deaths, and recoveries (for the global data sets), and the daily case counts and deaths (for the U.S. data sets) are not displayed in their entirety.

```r
knitr::opts_chunk$set(echo = TRUE, tidy.opts = list(width.cutoff = 80), tidy
= TRUE)

# List of data frames with names converted to display the first 12 columns
and
# first few rows
global_confirmed_head <- head(global_confirmed[, 1:6])
global_deaths_head <- head(global_deaths[, 1:6])
global_recovered_head <- head(global_recovered[, 1:6])
us_confirmed_head <- head(us_confirmed[, 1:6])
us_deaths_head <- head(us_deaths[, 1:6])

# Display the column names and the first few rows of the datasets
kable(global_confirmed_head, caption = "Global Confirmed Cases")
```

Global Confirmed Cases

Province/State	Country/Region	Lat	Long	1/22/20	1/23/20
NA	Afghanistan	33.93911	67.70995	0	0
NA	Albania	41.15330	20.16830	0	0
NA	Algeria	28.03390	1.65960	0	0
NA	Andorra	42.50630	1.52180	0	0
NA	Angola	-11.20270	17.87390	0	0
NA	Antarctica	-71.94990	23.34700	0	0

```r
kable(global_deaths_head, caption = "Global Deaths Data")
```

Global Deaths Data

Province/State	Country/Region	Lat	Long	1/22/20	1/23/20
NA	Afghanistan	33.93911	67.70995	0	0
NA	Albania	41.15330	20.16830	0	0
NA	Algeria	28.03390	1.65960	0	0
NA	Andorra	42.50630	1.52180	0	0
NA	Angola	-11.20270	17.87390	0	0
NA	Antarctica	-71.94990	23.34700	0	0

```r
kable(global_recovered_head, caption = "Global Recovered Data")
```

Global Recovered Data

Province/State	Country/Region	Lat	Long	1/22/20	1/23/20
NA	Afghanistan	33.93911	67.70995	0	0
NA	Albania	41.15330	20.16830	0	0
NA	Algeria	28.03390	1.65960	0	0
NA	Andorra	42.50630	1.52180	0	0
NA	Angola	-11.20270	17.87390	0	0
NA	Antarctica	-71.94990	23.34700	0	0

```
kable(us_confirmed_head, caption = "U.S. Confirmed Cases Data")
```

U.S. Confirmed Cases Data

UID	iso2	iso3	code3	FIPS	Admin2
84001001	US	USA	840	1001	Autauga
84001003	US	USA	840	1003	Baldwin
84001005	US	USA	840	1005	Barbour
84001007	US	USA	840	1007	Bibb
84001009	US	USA	840	1009	Blount
84001011	US	USA	840	1011	Bullock

```
kable(us_deaths_head, caption = "U.S. Deaths Data")
```

U.S. Deaths Data

UID	iso2	iso3	code3	FIPS	Admin2
84001001	US	USA	840	1001	Autauga
84001003	US	USA	840	1003	Baldwin
84001005	US	USA	840	1005	Barbour
84001007	US	USA	840	1007	Bibb
84001009	US	USA	840	1009	Blount
84001011	US	USA	840	1011	Bullock

```
# Remove extra data frames
rm(global_confirmed_head, global_deaths_head, global_recovered_head,
us_confirmed_head,
    us_deaths_head)
```

Data Cleaning and Transformation

We will change the global data sets (confirmed, deaths, and recovered) to have the same columns: Province_State, Country_Region, date, cases, deaths, recovered, Lat, and Long. We will also change the US data sets (confirmed and deaths) by renaming Long_ to Long to match the global data sets The last change will be to rename Admin2 to County in the US data sets These changes will create a common structure for all data sets .

Data Transformation Summary

In this section, we describe the steps taken to transform the data for analysis. The process involves renaming columns, reshaping data, converting date formats, and aggregating data for both global and U.S. data sets Below is a detailed explanation of each transformation step:

Renaming Columns

To ensure consistency and clarity across the data sets, we checked and renamed certain columns. For the global data sets, we renamed “Province/State” to “Province_State” and “Country/Region” to “Country_Region”. For the U.S. data sets, we renamed “Admin2” to “County” and “Long_” to “Long”. This was achieved using a custom function that checks for the presence of old column names and renames them if they exist.

Reshaping Data

We reshaped the data sets from a wide format to a long format to facilitate time series analysis. Using the `pivot_longer` function, we transformed the global confirmed, deaths, and recovered data sets, as well as the U.S. confirmed and deaths data sets This restructuring allowed us to represent each date’s data as individual rows, which is more suitable for temporal analysis.

Date Conversion

We converted the date columns to the Date type to enable accurate date-based operations and analyses. This conversion was done using a custom function that applies the date conversion to each dataset.

Combining Data sets

For the global data, we combined the confirmed, deaths, and recovered data sets into a single dataset using full joins. This combined dataset contains information on cases, deaths, and recoveries for each location and date.

For the U.S. data, we combined the confirmed and deaths data sets We then aggregated the data at the county and state levels, as well as at the country level. We calculated additional metrics such as deaths per million population, and for the state and county data, we also computed new cases and new deaths by calculating the differences between consecutive dates.

Summary of Transformed Data

The following tables display samples of the transformed data sets The first table shows a sample of the combined global data, and the second table shows a sample of the aggregated U.S. data by county and state.

```
knitr::opts_chunk$set(echo = TRUE, tidy.opts = list(width.cutoff = 80), tidy
= TRUE,
    kinter.row = 30)

# Display the column names of the global_confirmed_deaths_joined data set in
a
# kable table
global_confirmed_head <- head(global_confirmed_deaths_joined)
kable(global_confirmed_head, caption = "Global confirmed cases data.")
```

Global confirmed cases data.

Province_State	Country_Region	date	cases	deaths	recovered	Lat	Long
NA	Afghanistan	2020-01-22	0	0	0	33.93911	67.70995
NA	Afghanistan	2020-01-23	0	0	0	33.93911	67.70995
NA	Afghanistan	2020-01-24	0	0	0	33.93911	67.70995
NA	Afghanistan	2020-01-25	0	0	0	33.93911	67.70995
NA	Afghanistan	2020-01-26	0	0	0	33.93911	67.70995
NA	Afghanistan	2020-01-27	0	0	0	33.93911	67.70995

```
us_by_county_state_head <- head(us_by_county_state)

# Display the performance metrics
kable(us_by_county_state_head, caption = "U.S. by County and State")
```

U.S. by County and State

County	Province_State	Country_Region	date	cases	deaths	Population	deaths_per_mill	new_cases	new_deaths
King	Washington	US	2020-02-29	6	1	2252782	0.4438956	6	1

County	Province_State	Country_Region	date	cases	deaths	Population	deaths_per_mill	new_cases	new_deaths
King	Washington	US	2020-03-01	9	1	2252782	0.4438956	3	0
King	Washington	US	2020-03-02	14	5	2252782	2.2194780	5	4
Snohomish	Washington	US	2020-03-02	4	1	822083	1.2164222	4	1
King	Washington	US	2020-03-03	21	6	2252782	2.6633736	7	1
Snohomish	Washington	US	2020-03-03	6	1	822083	1.2164222	2	0

Summary of Total Cases, Deaths, and Recoveries

This summary provides an overview of the total cases, deaths, and recoveries for both the global dataset and the U.S. dataset. The global dataset encompasses data from all countries, while the U.S. dataset includes data from all states and counties within the United States.

For the global dataset, the total number of recovered cases is counted up to December 13, 2020. The total recovered cases are determined by subtracting the total deaths from the total cases. The figures for total cases, deaths, and recoveries are presented in the table below.

```
knitr::opts_chunk$set(echo = TRUE, tidy.opts = list(width.cutoff = 80), tidy
= TRUE)

# Summarize global data
global_totals <- global_confirmed_deaths_joined %>%
    group_by(Country_Region) %>%
    filter(cases > 0, deaths > 0, recovered > 0) %>%
    filter(date == max(date, na.rm = TRUE)) %>%
    summarize(total_cases = sum(cases, na.rm = TRUE), total_deaths =
sum(deaths,
        na.rm = TRUE)) %>%
    ungroup() %>%
    summarize(total_cases = sum(total_cases, na.rm = TRUE), total_deaths =
sum(total_deaths,
        na.rm = TRUE), total_recovered = total_cases - total_deaths) %>%
    mutate(total_cases = format(total_cases, big.mark = ","), total_deaths =
format(total_deaths,
        big.mark = ","), total_recovered = format(total_recovered, big.mark =
","))

# Display the global totals
kable(global_totals, align = "c", caption = "Global Total Cases, Deaths, and
Recoveries") %>%
    kable_styling(latex_options = "striped", full_width = FALSE)
```

Global Total Cases, Deaths, and Recoveries

total_cases	total_deaths	total_recovered
170,014,199	3,740,079	166,274,120

```
# Summarize US data
total_us <- us_confirmed %>%
    left_join(us_by_country_region, by = c("Country_Region", "date", "cases",
"deaths")) %>%
    group_by(Country_Region) %>%
    filter(date == max(date, na.rm = TRUE)) %>%
    summarize(total_cases = sum(cases, na.rm = TRUE), total_deaths =
sum(deaths,
        na.rm = TRUE)) %>%
    ungroup() %>%
    mutate(total_recovered = total_cases - total_deaths, total_cases =
format(total_cases,
        big.mark = ","), total_deaths = format(total_deaths, big.mark = ","),
total_recovered = format(total_recovered,
        big.mark = ","))

# Display the US totals
kable(total_us, align = "c", caption = "U.S. Total Cases, Deaths, and
Recovered") %>%
    kable_styling(latex_options = "striped", full_width = FALSE)
```

U.S. Total Cases, Deaths, and Recovered

Country_Region	total_cases	total_deaths	total_recovered
US	103,802,702	1,123,836	102,678,866

Visualizing the Data

These graphs show the actual cases in blue, the linear regression line in red, and the smooth curve in green. The first graph displays the global COVID-19 cases over time, while the second graph shows the U.S. COVID-19 cases over time. The linear regression line and smooth curve provide insights into the trends and patterns of the data.

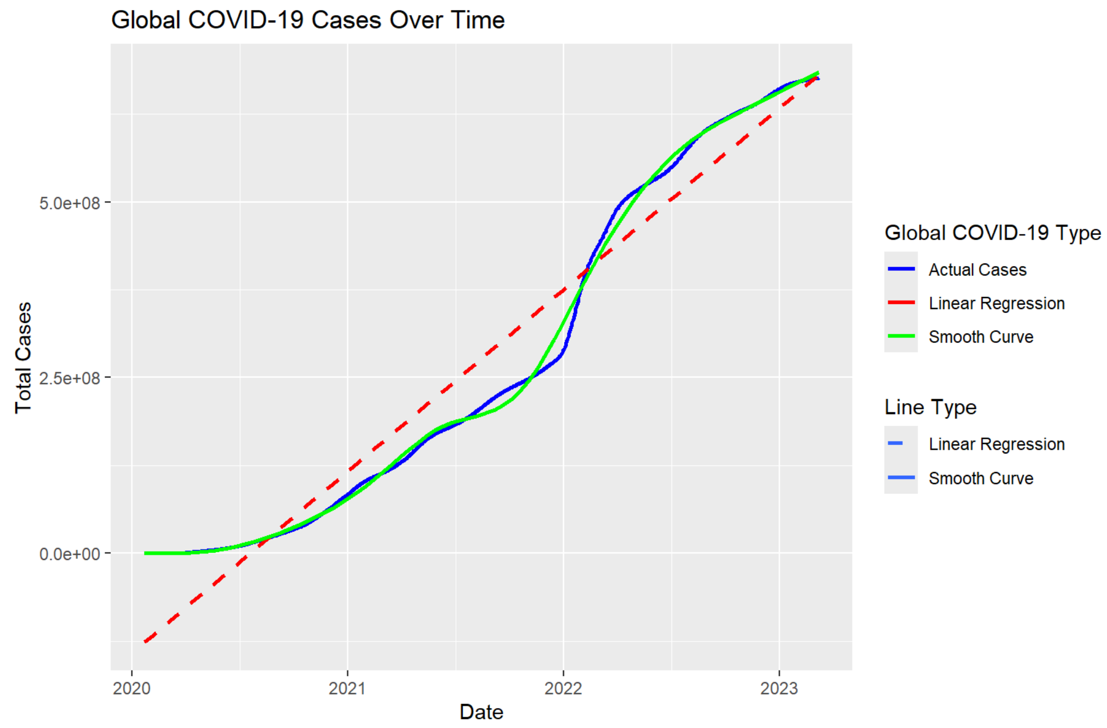
```
knitr::opts_chunk$set(echo = TRUE, tidy.opts = list(width.cutoff = 80), tidy
= TRUE)
```

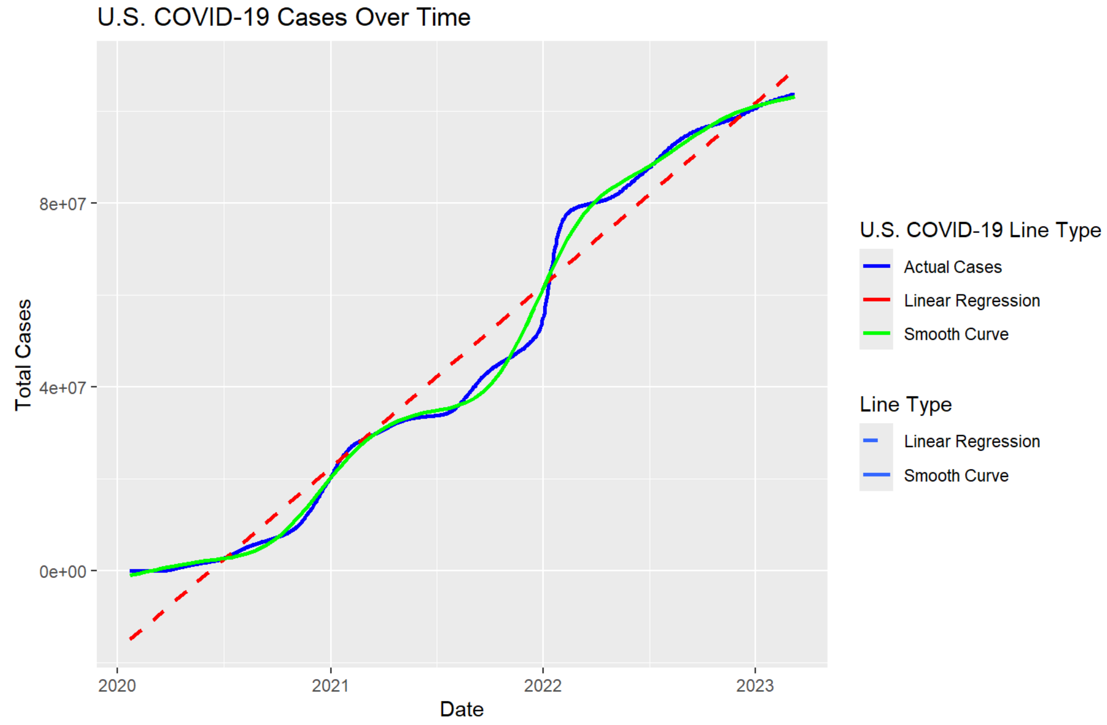
```
# Prepare Global Cases Data
global_cases_data <- global_confirmed %>%
    group_by(date) %>%
    summarize(total_cases = sum(cases, na.rm = TRUE))

# Prepare U.S. Cases Data
us_cases_data <- us_confirmed %>%
    group_by(date) %>%
    summarize(total_cases = sum(cases, na.rm = TRUE))

# Global Cases Over Time with Regression Line and Smooth Curve
global_cases_plot <- ggplot(global_cases_data, aes(x = date, y =
total_cases)) +
    geom_line(aes(color = "Actual Cases"), size = 1) + geom_smooth(method =
"lm",
    aes(color = "Linear Regression", linetype = "Linear Regression"), se =
FALSE,
    size = 1) + geom_smooth(aes(color = "Smooth Curve", linetype = "Smooth
Curve"),
    se = FALSE, size = 1) + labs(title = "Global COVID-19 Cases Over Time", x
= "Date",
    y = "Total Cases") + scale_color_manual(name = "Global COVID-19 Type",
values = c(`Actual Cases` = "blue",
    `Linear Regression` = "red", `Smooth Curve` = "green")) +
scale_linetype_manual(name = "Line Type",
    values = c(`Actual Cases` = "solid", `Linear Regression` = "dashed",
`Smooth Curve` = "solid")) +
    theme(legend.position = "right") + guides(color = guide_legend(order =
1), linetype = guide_legend(order = 2))
```

```
## 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.
```

```
# U.S. Cases Over Time with Regression Line and Smooth Curve
us_cases_plot <- ggplot(us_cases_data, aes(x = date, y = total_cases)) +
geom_line(aes(color = "Actual Cases"),
    size = 1) + geom_smooth(method = "lm", aes(color = "Linear Regression",
linetype = "Linear Regression"),
    se = FALSE, size = 1) + geom_smooth(aes(color = "Smooth Curve", linetype
= "Smooth Curve"),
    se = FALSE, size = 1) + labs(title = "U.S. COVID-19 Cases Over Time", x =
"Date",
    y = "Total Cases") + scale_color_manual(name = "U.S. COVID-19 Line Type",
values = c(`Actual Cases` = "blue",
    `Linear Regression` = "red", `Smooth Curve` = "green")) +
scale_linetype_manual(name = "Line Type",
    values = c(`Actual Cases` = "solid", `Linear Regression` = "dashed",
`Smooth Curve` = "solid")) +
    theme(legend.position = "right") + guides(color = guide_legend(order =
1), linetype = guide_legend(order = 2))

# Display plots
gridExtra::grid.arrange(global_cases_plot, us_cases_plot, ncol = 1)
```

```
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
## `geom_smooth()` using formula = 'y ~ x'
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

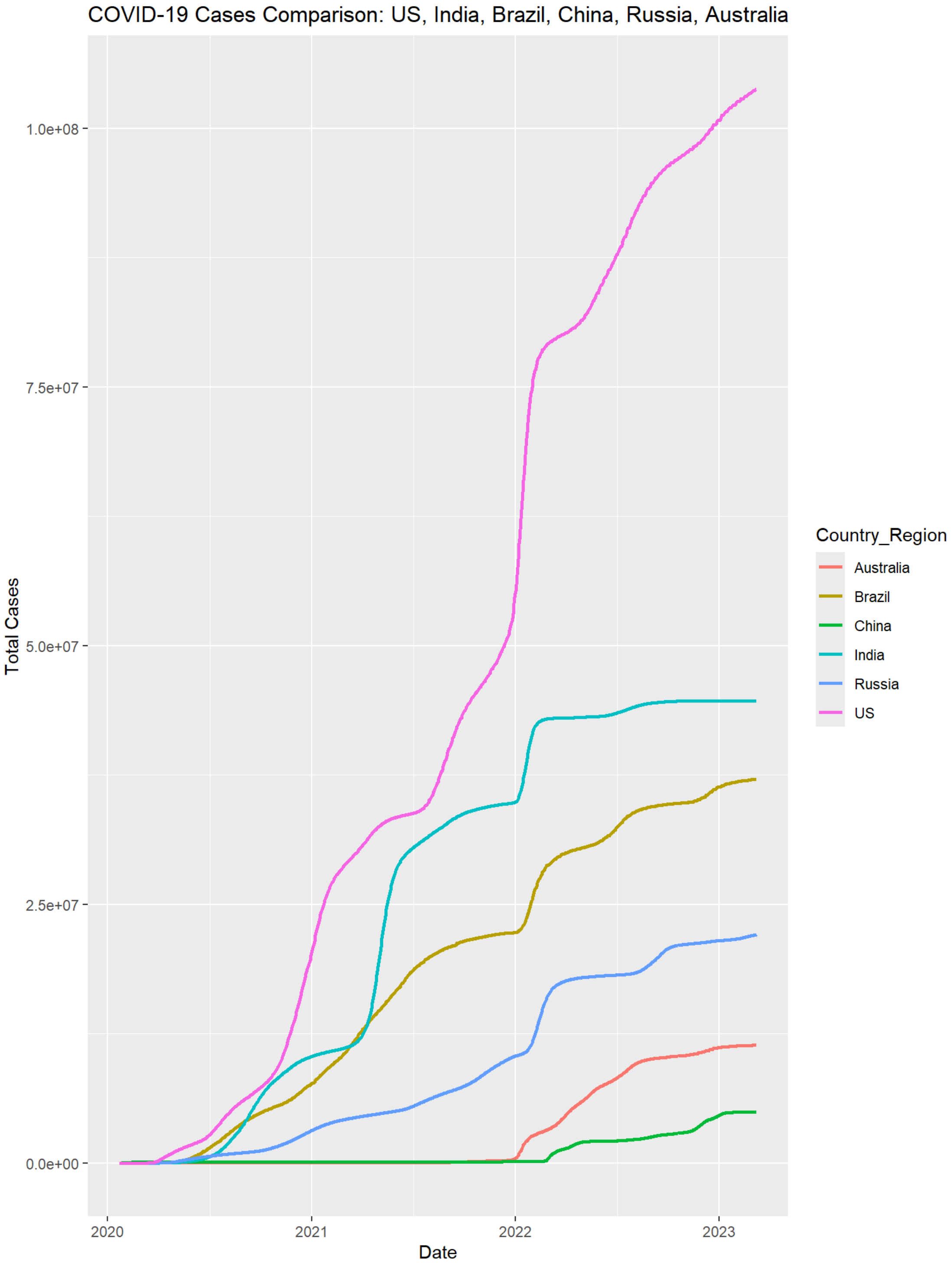




##

Comparative Analysis This graph shows the reported number of case for different countries over time. The countries included in the analysis are the United States, India, Brazil, China, Russia, and Australia. The graph provides a visual comparison of the COVID-19 cases in these countries and highlights the differences in the number of cases reported over time.

```
knitr::opts_chunk$set(echo = TRUE, tidy.opts = list(width.cutoff = 80), tidy
= TRUE)
# Comparative Analysis Between Countries
country_comparison <- global_confirmed %>%
    filter(Country_Region %in% c("US", "India", "Brazil", "China", "Russia",
"Australia")) %>%
    group_by(Country_Region, date) %>%
    summarize(total_cases = sum(cases, na.rm = TRUE)) %>%
    ggplot(aes(x = date, y = total_cases, color = Country_Region)) +
geom_line(size = 1) +
    labs(title = "COVID-19 Cases Comparison: US, India, Brazil, China,
Russia, Australia",
        x = "Date", y = "Total Cases")

print(country_comparison)
```



Statistical Analysis

The linear regression model aims to predict the total number of deaths based on the total number of cases. The model's summary shows a very high correlation between the two variables, with an R-squared value of 0.8512, indicating that approximately 85.12% of the variability in total deaths can be explained by the total cases. The coefficient for total cases is 0.01343, meaning that for every additional case, there is an associated increase of approximately 0.01343 deaths. This coefficient is highly significant (p-value < 2e-16), suggesting a strong relationship between total cases and total deaths. The residuals indicate some variability, with a residual standard error of 28,240,000, showing the model's accuracy and the spread of residuals around the fitted values. The F-statistic of 1139 and its associated p-value (< 2.2e-16) further confirm the model's overall significance.

```
knitr::opts_chunk$set(echo = TRUE, tidy.opts = list(width.cutoff = 80), tidy
= TRUE)
# Correlation Analysis
cor_data <- global_confirmed_deaths_joined %>%
    group_by(Country_Region) %>%
    summarize(total_cases = sum(cases, na.rm = TRUE), total_deaths =
sum(deaths,
        na.rm = TRUE))

cor_result <- cor(cor_data$total_cases, cor_data$total_deaths, use =
"complete.obs")
cor_result
```

```
## [1] 0.9226252
```

```
# Linear Regression Model
lm_model <- lm(total_deaths ~ total_cases, data = cor_data)
summary(lm_model)
```

```
##
## Call:
## lm(formula = total_deaths ~ total_cases, data = cor_data)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -103692127   -2715751    -843647    -277593  202902409
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 8.160e+05  2.089e+06   0.391    0.696
## total_cases 1.343e-02  3.980e-04  33.745   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 28240000 on 199 degrees of freedom
## Multiple R-squared:  0.8512, Adjusted R-squared:  0.8505
## F-statistic:  1139 on 1 and 199 DF,  p-value: < 2.2e-16
```

Predictive Model

Using the data, we developed a predictive model to forecast future COVID-19 cases and deaths. This model will help anticipate the future burden on healthcare systems and inform public health interventions.

The predictive model is based on a linear regression approach, where the total cases are used to predict the total deaths. The model's performance is evaluated using the post-resampling method, which provides metrics such as RMSE, MAE, and R-squared to assess the model's accuracy.

```r
knitr::opts_chunk$set(echo = TRUE, tidy.opts = list(width.cutoff = 80), tidy
= TRUE)

set.seed(123)

# Data preparation for predictive modeling
us_confirmed_model <- us_cases_data %>%
    mutate(day_of_week = wday(date, label = TRUE), month = month(date),
day_of_year = yday(date)) %>%
    select(-date)

trainIndex <- createDataPartition(us_confirmed_model$total_cases, p = 0.8,
list = FALSE,
    times = 1)
trainData <- us_confirmed_model[trainIndex, ]
testData <- us_confirmed_model[-trainIndex, ]

# Fit a linear regression model
model <- train(total_cases ~ ., data = trainData, method = "lm")

# Make predictions
predictions <- predict(model, newdata = testData)

# Evaluate the model
postResample(predictions, testData$total_cases)
```

```
##         RMSE     Rsquared          MAE
## 3.680259e+07 7.259422e-03 3.300944e+07
```

Bias Considerations

It is important to consider potential biases in the data and analysis. The dataset may underreport cases and deaths due to limited testing and reporting inconsistencies across different regions. Additionally, the model's predictions are based on historical data, which may not fully capture future changes in virus transmission dynamics, public health interventions, or population behavior.

Limitations

This analysis has several limitations that should be considered when interpreting the results. The data is subject to reporting delays, inconsistencies, and inaccuracies, which can affect the accuracy of the analysis. The predictive model developed in this analysis is based on historical data and may not account for future changes in the pandemic. The model's assumptions, such as linearity and independence of variables, may not hold in practice, leading to potential errors in the predictions.

Conclusion

This analysis provides a comprehensive overview of the COVID-19 pandemic using the Johns Hopkins dataset. By transforming, visualizing, and analyzing the data, we gained insights into the global and U.S. spread of the virus, the impact on different countries, and the relationship between confirmed cases and deaths.

The predictive model developed in this analysis helps us understand the future burden of COVID-19 and inform public health interventions. However, it is essential to consider the limitations and biases in the data and analysis when interpreting the results. Further research and data collection are needed to improve the accuracy and reliability of COVID-19 analyses and predictions.

Future Considerations

Expand the analysis to include additional variables such as vaccination rates, testing capacity, and public health measures to provide a more comprehensive understanding of the pandemic. Incorporate machine learning algorithms to develop more sophisticated predictive models that can capture the complex dynamics of the virus spread. Collaborate with public health agencies and researchers to validate the findings and ensure the analysis is aligned with current scientific knowledge and best practices.

Reproducibility

To ensure the reproducibility of this analysis, the following steps can be followed:

1. Download the latest version of R and RStudio.
2. Install the required packages as listed in the "load libraries" section.
3. Download the dataset from the Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE) COVID-19 Data Repository.
4. Run the R script provided in this document.
5. Save the session information to verify the environment in which the analysis was conducted.

```
knitr::opts_chunk$set(echo = TRUE, tidy.opts = list(width.cutoff = 80), tidy
= TRUE)
```

```
# URL path for the permalink to the raw data on GitHub
raw_url <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-
19/4360e50239b4eb6b22f3a1759323748f36752177/csse_covid_19_data/csse_covid_19_
time_series/"

# File names
file_names <- c("time_series_covid19_confirmed_global.csv",
"time_series_covid19_deaths_global.csv",
    "time_series_covid19_recovered_global.csv",
"time_series_covid19_confirmed_US.csv",
    "time_series_covid19_deaths_US.csv")

# Save the session information to a text file
writeLines(capture.output(sessionInfo()), "session_info.txt")

# Write a file of all installed packages
write.csv(installed.packages()[, "Package"], file = "installed_packages.csv")
```