Final Project 2: Reproducible Report on COVID19 Data

WenhaoC

Introduction The COVID-19 pandemic has profoundly affected the world since its emergence in late 2019. Understanding the trends in COVID-19 cases, deaths, and recoveries is crucial for public health planning, resource allocation, and implementing effective interventions. This study aims to analyze the daily trends of COVID-19 in selected countries, providing insights into the progression of the pandemic and the effectiveness of response measures.

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

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

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr   1.1.4     v readr   2.1.5
## v forcats 1.0.0     v stringr 1.5.1
## v ggplot2 3.5.1     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
```

Data Collection: Collecting data on COVID-19 confirmed cases, deaths, and recoveries from the dataset

```
url_in <- "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covi
file_name = c("time_series_covid19_confirmed_US.csv",
"time_series_covid19_confirmed_global.csv",
"time_series_covid19_deaths_US.csv",
"time_series_covid19_deaths_global.csv",
"time_series_covid19_recovered_global.csv")
urls = str_c(url_in, file_name)
urls
```

```
## [1] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_1
## [2] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_1
## [3] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_1
## [4] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_1
## [5] "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_1
```

```
comfirm_us <- read.csv(urls[1]);
comfirm_global <- read_csv(urls[2])
```

```
## 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.
```

```
comfirm_deaths_us <- read.csv(urls[3])
comfirm_death_global <- read.csv(urls[4])
comfirm_recovered_global <- read.csv(urls[5])
```

Data Preprocessing: Cleaning and transforming the data to ensure accuracy and consistency.

```
comfirm_global1 <- comfirm_global %>%
  pivot_longer(cols = -c('Province/State', 'Country/Region', Lat, Long) , names_to = 'date', values_to =
  select(-c(Lat, Long)) %>%
  rename('Country_Region' = 'Country/Region', 'Province_State' = 'Province/State') %>%
  mutate(date = mdy(date)) %>%
  mutate(date = format(date, "%m/%d/%y"))

comfirm_death_global1 <- comfirm_death_global %>%
  pivot_longer(cols = -c('Province.State', 'Country.Region', Lat, Long) , names_to = 'date', values_to =
  select(-c(Lat, Long)) %>%
  rename('Country_Region' = 'Country.Region', 'Province_State' = 'Province.State') %>%
  mutate(date = mdy(gsub("^X", "", date))) %>%
  mutate(date = format(date, "%m/%d/%y"))

comfirm_recovered_global1 <- comfirm_recovered_global %>%
  pivot_longer(cols = -c('Province.State', 'Country.Region', Lat, Long) , names_to = 'date', values_to =
  select(-c(Lat, Long)) %>%
  rename('Country_Region' = 'Country.Region', 'Province_State' = 'Province.State') %>%
  mutate(date = mdy(gsub("^X", "", date))) %>%
  mutate(date = format(date, "%m/%d/%y"))


summaryglobal <- comfirm_recovered_global1 %>%
  full_join(comfirm_global1, by = c("Country_Region", "Province_State", "date")) %>%
  full_join(comfirm_death_global1, by = c("Country_Region", "Province_State", "date"))%>%
  mutate(date = mdy(date)) %>%
  filter(cases > 0)%>%
  unite ("combined_key", c('Province_State', 'Country_Region'),
         sep = ', ', na.rm = TRUE, remove = FALSE)


summary(summaryglobal)
```

```
##  combined_key       Province_State     Country_Region          date
##  Length:306827      Length:306827      Length:306827      Min.   :2020-01-22
##  Class :character   Class :character   Class :character   1st Qu.:2020-12-12
##  Mode  :character   Mode  :character   Mode  :character   Median :2021-09-16
##                                                           Mean   :2021-09-11
```

```
##                                                          3rd Qu.:2022-06-15
##                                                          Max.   :2023-03-09
##
##       Cure              cases               deaths
##  Min.   :   -1.0   Min.   :        1   Min.   :    0
##  1st Qu.:    0.0   1st Qu.:     1316   1st Qu.:    1
##  Median :    0.0   Median :    20365   Median :    6
##  Mean   :  952.4   Mean   :  1032863   Mean   :  549
##  3rd Qu.:  221.0   3rd Qu.:   271281   3rd Qu.:   69
##  Max.   :64435.0   Max.   :103802702   Max.   :82195
##  NA's   :226699                        NA's   :210689
```

```
summary_by_country <- summaryglobal %>%
  group_by(Province_State, Country_Region, date) %>%
  summarise(cases = sum(cases, na.rm = TRUE), Cure = sum(Cure, na.rm = TRUE),
            deaths = sum(deaths, na.rm = TRUE)) %>%
  ungroup()
```

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

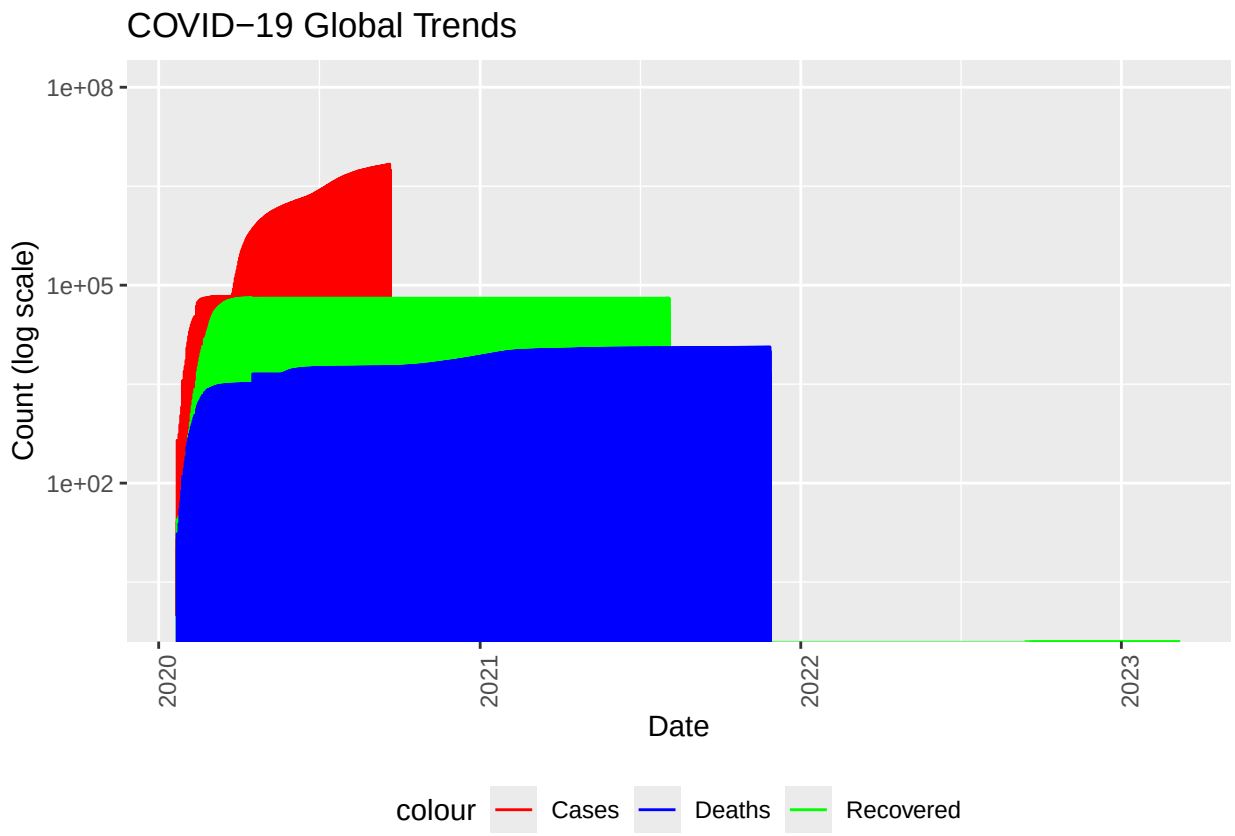
```
summary_by_country_new <- summary_by_country %>%
  mutate(new_cases = cases - lag(cases),
         new_deaths = deaths - lag(deaths),
         new_cure = Cure - lag(Cure)) %>%
   filter(!is.na(new_cases) & !is.na(new_deaths) & !is.na(new_cure))



groupsummary_by_country <- summary_by_country_new %>%
  summarise(new_cases = sum(new_cases, na.rm = TRUE),
            new_deaths = sum(new_deaths, na.rm = TRUE),
            new_cure = sum(new_cure, na.rm = TRUE)) %>%
  ungroup()
```

Data Analysis: Visualizing the data to identify initial trends and anomalies. This involves plotting the cumulative and daily counts of cases, deaths, and recoveries.

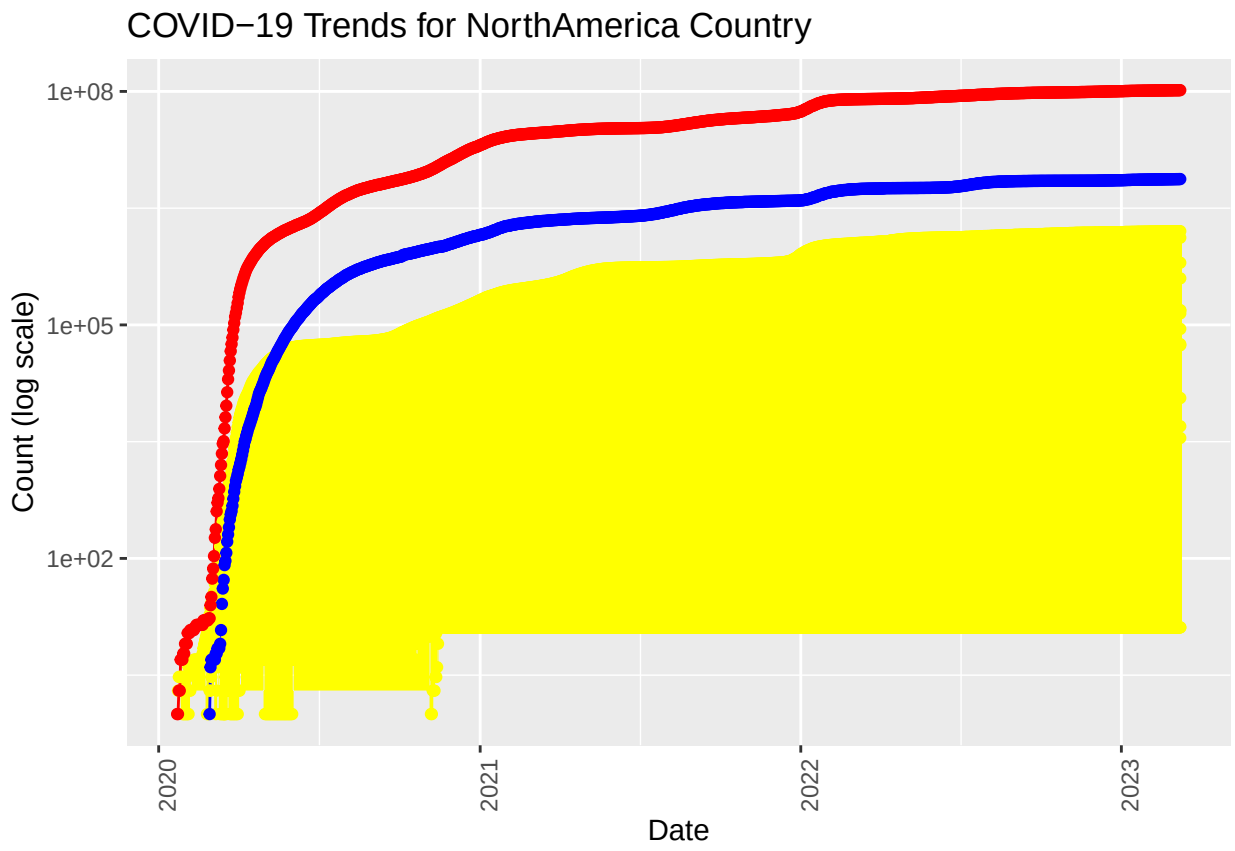
```
global_graph <- summary_by_country %>%
  filter(cases > 0) %>%
  ggplot(aes(x = date)) +
  geom_line(aes(y = cases, color = "Cases")) +
  geom_line(aes(y = Cure, color = "Recovered")) +
  geom_line(aes(y = deaths, color = "Deaths")) +
  scale_y_log10() +
  theme(legend.position = "bottom", axis.text.x = element_text(angle = 90)) +
  labs(title = "COVID-19 Global Trends", y = "Count (log scale)", x = "Date") +
  scale_color_manual(values = c("Cases" = "red", "Recovered" = "green", "Deaths" = "blue"))

northamerican <- c("Canada", "Mexico", "US")
selectedcountry_graph <- summary_by_country %>%
  filter(Country_Region %in% northamerican) %>%
  filter(cases>0) %>%
```

```
  ggplot(aes(x = date)) +
  geom_line(aes(y = cases, color = Country_Region)) +
  geom_point(aes(y = cases, color = Country_Region)) +
  scale_y_log10() +
  theme(legend.position = "buttom", axis.text.x = element_text(angle = 90)) +
  labs(title = str_c("COVID-19 Trends for NorthAmerica Country"), y = "Count (log scale)", x = "Date") +
  scale_color_manual(values = c("US" = "red", "Canada" = "yellow", "Mexico" = "blue"))

global_graph_new <- summary_by_country_new %>%
  ggplot(aes(x = date)) +
  geom_line(aes(y = new_cases, color = "New Cases")) +
  geom_line(aes(y = new_deaths, color = "New Deaths")) +
  geom_line(aes(y = new_cure, color = "New Recoveries")) +
  scale_y_log10() +
  labs(title = "Daily COVID-19 Trends for Selected Countries(US)", y = "Daily Count (log scale)", x = "D
  theme_minimal() +
  theme(legend.position = "bottom", axis.text.x = element_text(angle = 90)) +
  scale_color_manual(values = c("New Cases" = "blue", "New Deaths" = "red", "New Recoveries" = "yellow")

global_graph
```

```
## Warning in transformation$transform(x): NaNs produced

## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
## log-10 transformation introduced infinite values.
```

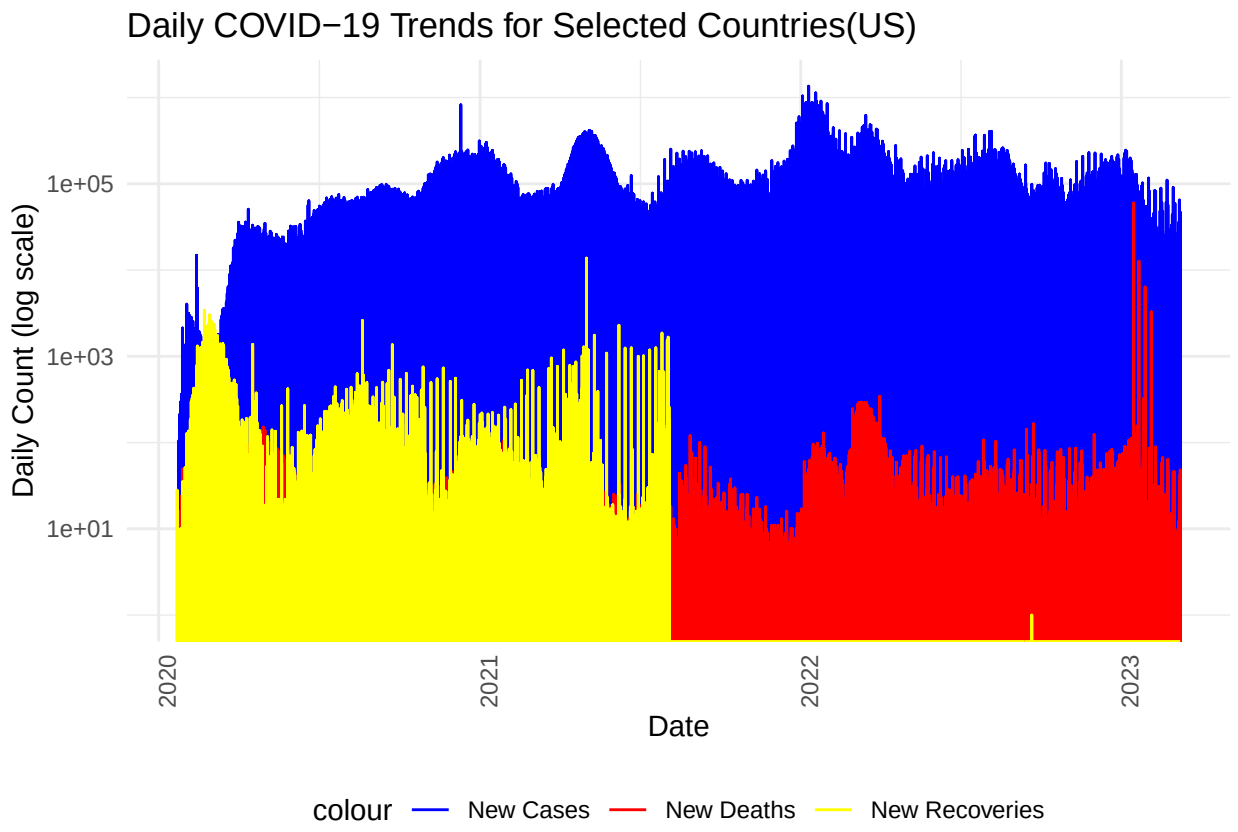


```
selectedcountry_graph
```




```
global_graph_new
```

```
## Warning in transformation$transform(x): NaNs produced
## Warning in transformation$transform(x): log-10 transformation introduced
## infinite values.

## Warning in transformation$transform(x): NaNs produced

## Warning in scale_y_log10(): log-10 transformation introduced infinite values.

## Warning in transformation$transform(x): NaNs produced

## Warning in scale_y_log10(): log-10 transformation introduced infinite values.

## Warning: Removed 25 rows containing missing values or values outside the scale range
## (`geom_line()`).

## Warning: Removed 13 rows containing missing values or values outside the scale range
## (`geom_line()`).
```



For this part of the study, I have generated three graphs: Global COVID-19 Trends, COVID-19 Trends for North American Countries, and Daily COVID-19 Trends for Selected Countries. The first graph visualizes global trends in COVID-19 cases, recoveries, and deaths over time. The second graph compares the data between some North American countries (Canada, Mexico, and the United States). The third graph shows the trends of newer data for specific countries, focusing on daily new cases, deaths, and recoveries. These graphs collectively provide a comprehensive view of the pandemic's progression globally, regionally, and daily.

Modeling: Applying linear regression to model the daily new deaths and forecast future trends.

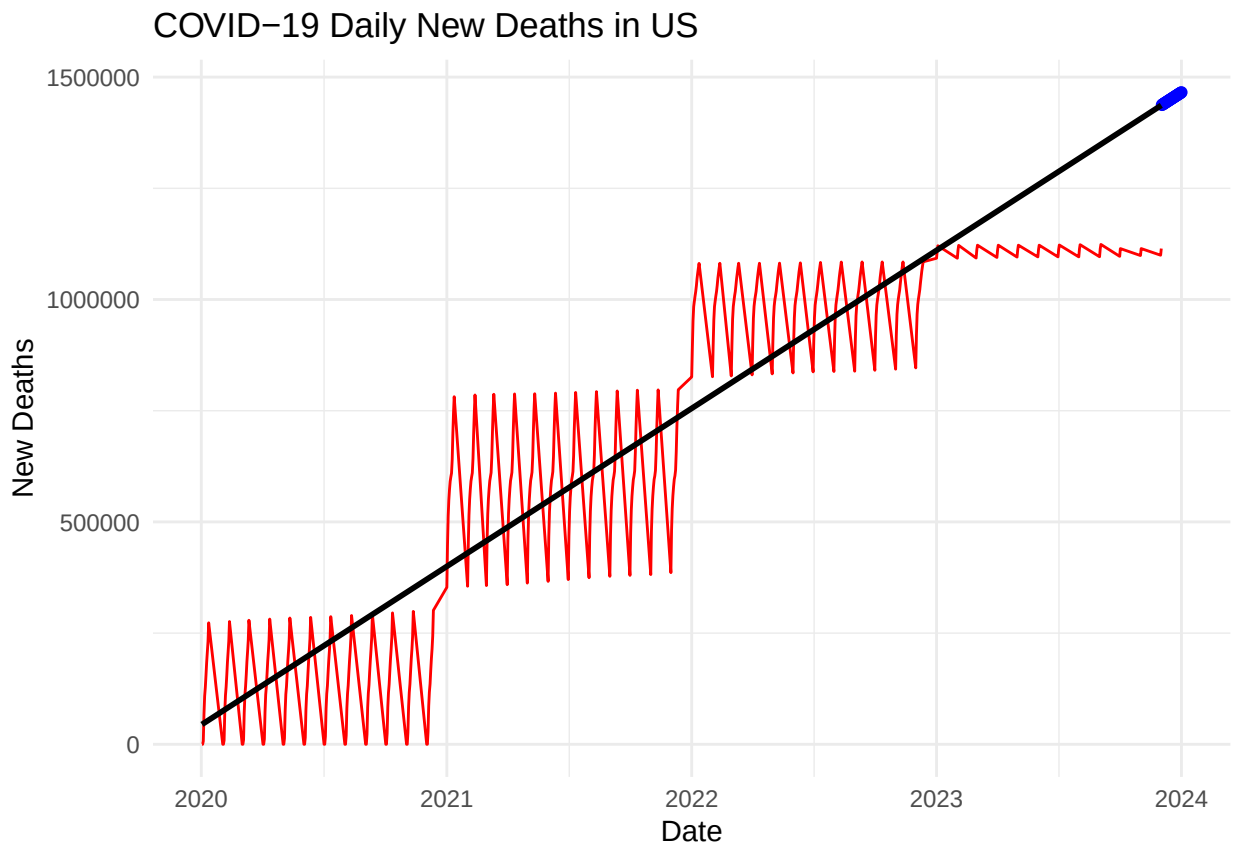
```
country <- "US"

deaths_country <- comfirm_death_global1 %>%
  filter(Country_Region == country) %>%
  group_by(date) %>%
  summarise(deaths = sum(deaths, na.rm = TRUE)) %>%
  mutate(date = as.Date(date, format = "%d/%m/%y")) %>%
  filter(!is.na(date)) %>%
  mutate(day_num = as.numeric(date - min(date)))

model <- lm(deaths ~ day_num, data = deaths_country)
summary(model)
```

```
##
## Call:
## lm(formula = deaths ~ day_num, data = deaths_country)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -371707 -105591     575  111681  370739
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 44714.70   14147.01   3.161  0.00168 **
## day_num       973.32      20.34  47.852  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 153200 on 451 degrees of freedom
## Multiple R-squared:  0.8355, Adjusted R-squared:  0.8351
## F-statistic:  2290 on 1 and 451 DF,  p-value: < 2.2e-16
```

```
future_days <- data.frame(day_num = (max(deaths_country$day_num) + 1):(max(deaths_country$day_num) + 30)

future_days$date <- seq.Date(from = max(deaths_country$date) + 1, by = "day", length.out = 30)

future_days$predicted_new_deaths <- predict(model, newdata = future_days)

ggplot(deaths_country, aes(x = date, y = deaths)) +
  geom_line(color = "red") +
  geom_point(data = future_days, aes(x = date, y = predicted_new_deaths), color = "blue") +
  labs(title = paste("COVID-19 Daily New Deaths in", country), x = "Date", y = "New Deaths") +
  theme_minimal() +
  geom_smooth(method = "lm", se = FALSE, color = "black")
```

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



The linear regression model analyzing daily new COVID-19 deaths indicates a significant upward trend, with a coefficient of 973.32 for day_num, suggesting an average increase of approximately 973 deaths per day. The model's intercept is 44,714.70, representing the estimated initial death count. The model is statistically significant ($p < 2e\text{-}16$) with an R-squared value of 0.8355, indicating that approximately 83.55% of the variance in daily deaths is explained by the model. Despite this, the residual standard error of 153,200 deaths highlights considerable variation not captured by the model, suggesting the presence of other influencing factors.

Bias Issue: In COVID-19 data, such as differences in reporting standards, data quality, and selection bias, can significantly impact the predictability and accuracy of results. Data lag in recovery reporting, caused by delays, inconsistent criteria, underreporting, and backlogs, can lead to misleading trends and reduced model accuracy.

```
## 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/Los_Angeles
## tzcode source: internal
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] forcats_1.0.0   stringr_1.5.1   dplyr_1.1.4     purrr_1.0.2
##  [5] readr_2.1.5     tidyr_1.3.1     tibble_3.2.1    ggplot2_3.5.1
##  [9] tidyverse_2.0.0 lubridate_1.9.3
##
## 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_0.6.35     magrittr_2.0.3    evaluate_0.24.0
##  [9] grid_4.4.0        timechange_0.3.0  fastmap_1.2.0     Matrix_1.7-0
## [13] mgcv_1.9-1        fansi_1.0.6       scales_1.3.0      cli_3.6.2
## [17] rlang_1.1.4       crayon_1.5.3      bit64_4.0.5       munsell_0.5.1
## [21] splines_4.4.0     withr_3.0.0       yaml_2.3.8        tools_4.4.0
## [25] parallel_4.4.0    tzdb_0.4.0        colorspace_2.1-0  curl_5.2.1
## [29] vctrs_0.6.5       R6_2.5.1          lifecycle_1.0.4   bit_4.0.5
## [33] vroom_1.6.5       pkgconfig_2.0.3   pillar_1.9.0      gtable_0.3.5
## [37] glue_1.7.0        xfun_0.45         tidyselect_1.2.1  highr_0.11
## [41] rstudioapi_0.16.0 knitr_1.47        farver_2.1.2      nlme_3.1-164
## [45] htmltools_0.5.8.1 labeling_0.4.3    rmarkdown_2.27    compiler_4.4.0
```

Thanks for your time.