COVID-19 Testing Data Scraper Report

Nasir Umar

2025-04-29

Project Overview

This project scrapes COVID-19 testing data from Wikipedia, cleans it, and performs basic analysis on COVID-19 testing metrics.

Load Libraries

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


# --- Step 1: Get Wikipedia Page via HTTP Request ---
wiki_base_url <- "https://en.wikipedia.org/w/index.php"
table_page <- list(title = "Template:COVID-19_testing_by_country")

response <- GET(url = wiki_base_url, query = table_page)

# Check if page loaded successfully
if (response$status_code == 200) {
  cat("Page loaded successfully!\n")
} else {
  stop("Failed to load page, status code:", response$status_code)
}
```

```
## Page loaded successfully!
```

```r
# --- Step 2: Parse and Extract Tables ---
page_html <- content(response, "text")
page <- read_html(page_html)

table_node <- html_nodes(page, "table")
```

```r
# Convert to data frames
tables_data_frame <- html_table(table_node, fill = TRUE)

# Select the second table which has the COVID testing data
covid_table_raw_df <- tables_data_frame[[2]]
covid_table_raw_df
```

```
## # A tibble: 173 x 9
##    `Country or region` `Date[a]`   Tested      `Units[b]` `Confirmed(cases)`
##    <chr>               <chr>       <chr>       <chr>      <chr>
##  1 Afghanistan         17 Dec 2020 154,767     samples    49,621
##  2 Albania             18 Feb 2021 428,654     samples    96,838
##  3 Algeria             2 Nov 2020  230,553     samples    58,574
##  4 Andorra             23 Feb 2022 300,307     samples    37,958
##  5 Angola              2 Feb 2021  399,228     samples    20,981
##  6 Antigua and Barbuda 6 Mar 2021  15,268      samples    832
##  7 Argentina           16 Apr 2022 35,716,069  samples    9,060,495
##  8 Armenia             29 May 2022 3,099,602   samples    422,963
##  9 Australia           9 Sep 2022  78,548,492  samples    10,112,229
## 10 Austria             1 Feb 2023  205,817,752 samples    5,789,991
## # i 163 more rows
## # i 4 more variables: `Confirmed/tested,%` <chr>,
## #   `Tested/population,%` <chr>, `Confirmed/population,%` <chr>, Ref. <chr>
```

```r
# --- Step 3: Pre-process the Extracted Data Frame ---
# View column names (optional)
names(covid_table_raw_df)
```

```
## [1] "Country or region"        "Date[a]"
## [3] "Tested"                   "Units[b]"
## [5] "Confirmed(cases)"         "Confirmed/tested,%"
## [7] "Tested/population,%"     "Confirmed/population,%"
## [9] "Ref."
```

```r
# Remove the World row
covid_table_raw_df <- covid_table_raw_df[!(covid_table_raw_df$`Country or region`=="World"),]
# Remove the last row
covid_table_raw_df <- covid_table_raw_df[1:172, ]
# Remove the Units and Ref columns
covid_table_raw_df["Ref."] <- NULL
covid_table_raw_df["Units[b]"] <- NULL
# Renaming the columns
names(covid_table_raw_df) <- c(
  "Country", "Date", "Tested", "Confirmed",
  "Confirmed.tested.ratio", "Tested.population.ratio",
  "Confirmed.population.ratio"
)

# Convert column data types
covid_table_raw_df$Country <- as.factor(covid_table_raw_df$Country)
covid_table_raw_df$Date <- as.factor(covid_table_raw_df$Date)
covid_table_raw_df$Tested <- as.numeric(gsub(",","",covid_table_raw_df$Tested))
```

```
covid_table_raw_df$Confirmed <- as.numeric(gsub(",","",covid_table_raw_df$Confirmed))
covid_table_raw_df$Confirmed.tested.ratio <- as.numeric(gsub(",","",covid_table_raw_df$Confirmed.tested
covid_table_raw_df$Tested.population.ratio <- as.numeric(gsub(",","",covid_table_raw_df$Tested.populatio
covid_table_raw_df$Confirmed.population.ratio <- as.numeric(gsub(",","",covid_table_raw_df$Confirmed.pop

# --- Step 4: Export the Cleaned Data Frame to CSV ---
write.csv(covid_table_raw_df, "global_covid_testing_data_clean.csv", row.names = FALSE)

cat("Data cleaned and saved successfully as 'global_covid_testing_data_clean.csv'!\n")
```

```
## Data cleaned and saved successfully as 'global_covid_testing_data_clean.csv'!
```

```
#---
# Get the summary of the processed data frame again
head(covid_table_raw_df)
```

```
## # A tibble: 6 x 7
##   Country   Date  Tested Confirmed Confirmed.tested.ratio Tested.population.ra~1
##   <fct>     <fct>  <dbl>     <dbl>                  <dbl>                  <dbl>
## 1 Afghanis~ 17 D~ 154767     49621                   32.1                    0.4
## 2 Albania   18 F~ 428654     96838                   22.6                   15
## 3 Algeria   2 No~ 230553     58574                   25.4                    0.53
## 4 Andorra   23 F~ 300307     37958                   12.6                  387
## 5 Angola    2 Fe~ 399228     20981                    5.3                    1.3
## 6 Antigua ~ 6 Ma~  15268       832                    5.4                   15.9
## # i abbreviated name: 1: Tested.population.ratio
## # i 1 more variable: Confirmed.population.ratio <dbl>
```

```
#summary(covid_table_raw_df)


# Load cleaned data
global_covid_testdata <- read.csv(
  "global_covid_testing_data_clean.csv",
  stringsAsFactors = FALSE,
  na.strings = c("NA", "", "N/A")
)

# View first few rows
head(global_covid_testdata)
```

```
##               Country        Date Tested Confirmed Confirmed.tested.ratio
## 1         Afghanistan 17 Dec 2020 154767     49621                   32.1
## 2             Albania 18 Feb 2021 428654     96838                   22.6
## 3             Algeria  2 Nov 2020 230553     58574                   25.4
## 4             Andorra 23 Feb 2022 300307     37958                   12.6
## 5              Angola  2 Feb 2021 399228     20981                    5.3
## 6 Antigua and Barbuda  6 Mar 2021  15268       832                    5.4
##   Tested.population.ratio Confirmed.population.ratio
## 1                    0.40                      0.130
## 2                   15.00                      3.400
## 3                    0.53                      0.130
## 4                  387.00                        49.000
## 5                    1.30                         0.067
## 6                   15.90                         0.860
```

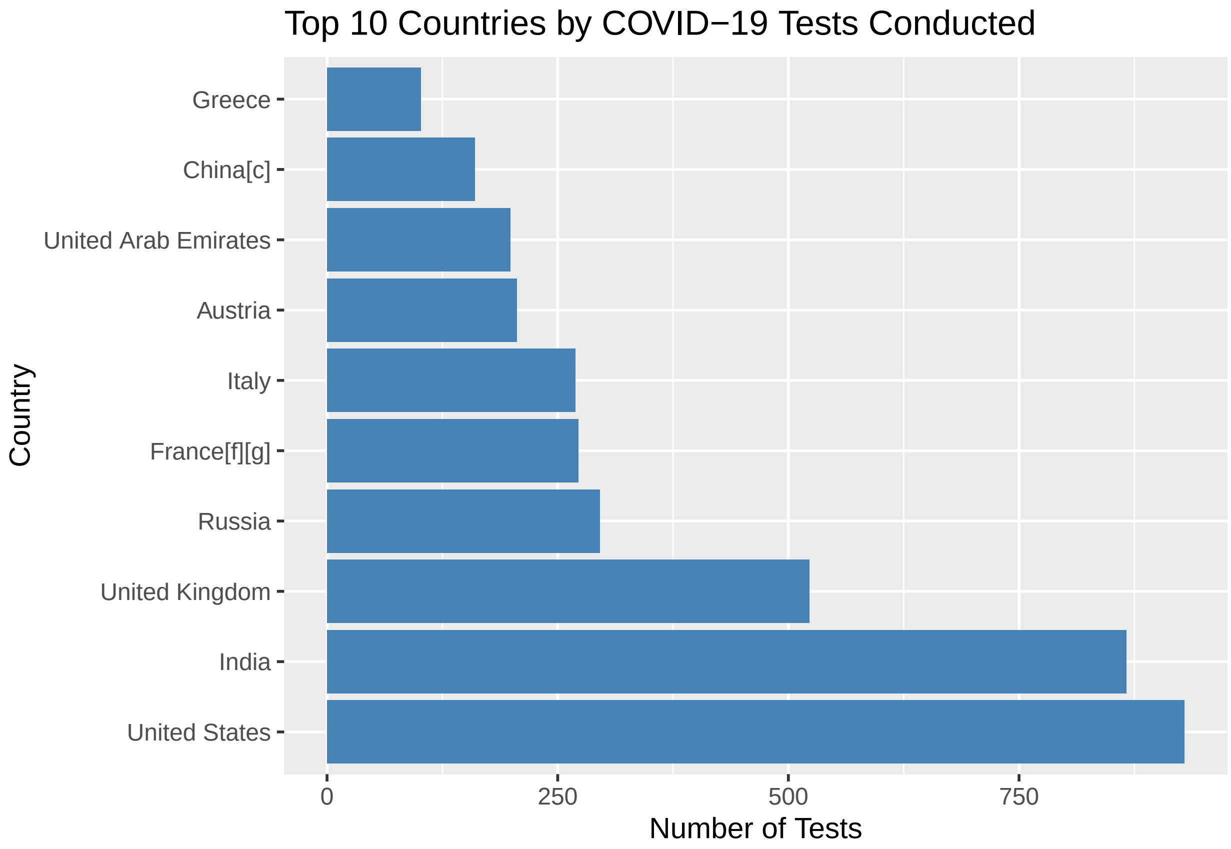
```
#Summary
summary(global_covid_testdata)
```

```
##    Country              Date               Tested            Confirmed
##  Length:172         Length:172         Min.   :     3880   Min.   :       0
##  Class :character   Class :character   1st Qu.:   512037   1st Qu.:   37839
##  Mode  :character   Mode  :character   Median :  3029859   Median :  281196
##                                        Mean   : 31377219   Mean   : 2508340
##                                        3rd Qu.: 12386725   3rd Qu.: 1278105
##                                        Max.   :929349291   Max.   :90749469
##  Confirmed.tested.ratio Tested.population.ratio Confirmed.population.ratio
##  Min.   : 0.00          Min.   :   0.006        Min.   : 0.000
##  1st Qu.: 5.00          1st Qu.:   9.475        1st Qu.: 0.425
##  Median :10.05          Median :  46.950        Median : 6.100
##  Mean   :11.25          Mean   : 175.504        Mean   :12.769
##  3rd Qu.:15.25          3rd Qu.: 156.500        3rd Qu.:16.250
##  Max.   :46.80          Max.   :3223.000        Max.   :74.400
```

```
# Top 10 Countries by Number of Tests Conducted
top_10_tests <- global_covid_testdata %>%
  arrange(desc(Tested)) %>%
  slice_head(n = 10)

top_10_tests
```

```
##                   Country        Date    Tested Confirmed Confirmed.tested.ratio
## 1           United States 29 Jul 2022 929349291  90749469                  9.800
## 2                   India  8 Jul 2022 866177937  43585554                  5.000
## 3          United Kingdom 19 May 2022 522526476  22232377                  4.300
## 4                  Russia  6 Jun 2022 295542733  18358459                  6.200
## 5            France[f][g] 15 May 2022 272417258  29183646                 10.700
## 6                   Italy 16 Mar 2023 269127054  25651205                  9.500
## 7                 Austria  1 Feb 2023 205817752   5789991                  2.800
## 8    United Arab Emirates  1 Feb 2023 198685717   1049537                  0.530
## 9                China[c] 31 Jul 2020 160000000     87655                  0.055
## 10                 Greece 18 Dec 2022 101576831   5548487                  5.500
##    Tested.population.ratio Confirmed.population.ratio
## 1                    281.0                    27.4000
## 2                     63.0                    31.7000
## 3                    774.0                    32.9000
## 4                    201.0                    12.5000
## 5                    417.0                    44.7000
## 6                    446.0                    42.5000
## 7                   2312.0                    65.0000
## 8                   2070.0                    10.9000
## 9                     11.1                     0.0061
## 10                   943.0                    51.5000
```

```r
# Install from CRAN
#install.packages("ggplot2")
# Load the package
library(ggplot2)

# Plot: Top 10 Countries by Number of Tests
library(scales)

ggplot(top_10_tests, aes(x = reorder(Country, -Tested), y = Tested/1e6)) +
  geom_col(fill = "steelblue") +
  coord_flip() +
  scale_y_continuous(labels = comma) +
  #scale_y_continuous(labels = scales::label_comma()) +
  labs(title = "Top 10 Countries by COVID-19 Tests Conducted",
       x = "Country",
       y = "Number of Tests")
```



```r
# Calculate mean positive ratio
mean_positive_ratio <- mean(global_covid_testdata$Confirmed.population.ratio, na.rm = TRUE)
mean_positive_ratio
```

```
## [1] 12.76858
```

```r
# Calculate worldwide COVID testing positive ratio¶
# Get the total confirmed cases worldwide
total_confirmed <- sum(global_covid_testdata$Confirmed, na.rm = TRUE)
# Get the total tested cases worldwide
total_tested <- sum(global_covid_testdata$Tested, na.rm = TRUE)
# Get the positive ratio (confirmed / tested)
positive_ratio <- round(total_confirmed / total_tested, 4)

print(positive_ratio)
```

```
## [1] 0.0799
```

```r
# Countries with confirmed to population ratio rate less than a 5% threshold
# Define threshold
threshold <- 5.0
# Subset countries below the threshold
low_ratio_countries <- global_covid_testdata[
  global_covid_testdata$Confirmed.population.ratio < threshold,
  c("Country", "Confirmed.population.ratio")
]

# Print results
print(low_ratio_countries)
```

```
##                   Country Confirmed.population.ratio
## 1          Afghanistan                    0.13000
## 2              Albania                    3.40000
## 3              Algeria                    0.13000
## 5               Angola                    0.06700
## 6   Antigua and Barbuda                    0.86000
## 14          Bangladesh                    0.70000
## 19               Benin                    0.06700
## 20              Bhutan                    1.71000
## 24              Brazil                    4.80000
## 25              Brunei                    0.07400
## 27        Burkina Faso                    0.05800
## 28             Burundi                    0.00740
## 29            Cambodia                    0.48000
## 30            Cameroon                    0.12000
## 32                Chad                    0.02900
## 34            China[c]                    0.00610
## 42            Djibouti                    1.70000
## 45            DR Congo                    0.02900
## 46             Ecuador                    2.80000
## 47               Egypt                    0.28000
## 48         El Salvador                    2.50000
## 49   Equatorial Guinea                    1.30000
## 51            Eswatini                    4.30000
## 52            Ethiopia                    0.24000
## 57               Gabon                    0.08200
## 58              Gambia                    0.21000
## 60             Germany                    4.50000
## 61                  Ghana                    0.31000
## 64                Grenada                    0.14000
## 66                 Guinea                    0.19000
## 67          Guinea-Bissau                    0.45000
## 69                  Haiti                    0.30000
## 70               Honduras                    3.90000
## 74              Indonesia                    2.50000
## 80            Ivory Coast                    0.13000
## 82                  Japan                    0.34000
## 84             Kazakhstan                    2.10000
## 85                  Kenya                    0.23000
## 88             Kyrgyzstan                    1.30000
## 89                   Laos                    0.00063
## 92                Lesotho                    1.60000
## 93                Liberia                    0.11000
## 97             Madagascar                    0.07600
## 98                 Malawi                    0.46000
## 101                  Mali                    0.07100
## 103            Mauritania                    0.41000
## 104             Mauritius                    0.03900
## 105                Mexico                    2.90000
## 107              Mongolia                    4.10000
## 109               Morocco                    3.40000
## 110            Mozambique                    0.34000
## 111               Myanmar                    0.81000
## 113                 Nepal                    3.50000
## 115         New Caledonia                    0.05000
## 117                 Niger                    0.02100
## 118               Nigeria                    0.07600
## 119           North Korea                    0.00000
## 123                  Oman                    2.50000
## 124              Pakistan                    0.27000
## 127      Papua New Guinea                    0.01100
## 130           Philippines                    4.00000
## 134               Romania                    3.70000
## 136                Rwanda                    0.76000
## 137 Saint Kitts and Nevis                    1.90000
## 141          Saudi Arabia                    2.20000
## 142               Senegal                    0.29000
## 144             Singapore                    1.10000
## 147          South Africa                    2.80000
## 148           South Korea                    0.17000
## 149           South Sudan                    0.08400
## 151             Sri Lanka                    0.43000
## 152                 Sudan                    0.05300
## 156              Tanzania                    0.00085
## 157              Thailand                    0.03800
## 158                  Togo                    0.46000
## 162                Uganda                    0.08700
## 168            Uzbekistan                    0.13000
## 169             Venezuela                    0.55000
## 171                Zambia                    1.80000
## 172              Zimbabwe                    1.70000
```