Investigatng Covid-19 Virus Trends

Clifton Lee

Last modified: 13 June, 2021

Introduction

The objective of this project is to provide a data analysis methodology that utilizes R's different data structures in conjunction with the R Markdown style syntax in order to analyze the Covid-19 scenario over the period January 1, 2020 to June 30, 2020.

This analysis answers the question, "which nations had the largest number of positive cases relative to the total number of tests conducted over the given period?" Please visit Covid-19 Worldwide Testing Data for further information on the dataset utilized.

Data Understanding

Before I can begin properly analyzing the information, there are a few questions I need to address in order to have a thorough understanding of the data I'm analyzing:

- How much information is available? For instance, the number of rows and columns.
- What data do we truly have? For instance, the column's type and content.
- Is there anything "abnormal" that may obstruct my analysis? For instance, possibly some terms were misspelled or a column contains information at many levels (country/region/state).

Those were the initial questions I asked, and their answers saved me time and work later.

The dataset includes the daily and cumulative number of COVID-19 tests performed, as well as the number of positive, hospitalized, recovered, and death cases reported by nation. The columns in the dataset are as follows:

1. `Date`: Date
2. `Continent_Name`: Continent names
3. `Two_Letter_Country_Code`: Country codes
4. `Country_Region`: Country names
5. `Province_State`: States/province names; value is All States when state/provincial level data is not available
6. `positive`: Cumulative number of positive cases reported.
7. `active`: Number of actively cases on that day.
8. `hospitalized`: Cumulative number of hospitalized cases reported.
9. `hospitalizedCurr`: Number of actively hospitalized cases on that day.
10. `recovered`: Cumulative number of recovered cases reported.
11. `death`: Cumulative number of deaths reported.
12. `total_tested`: Cumulative number of tests conducted.
13. `daily_tested`: Number of tests conducted on the day; if daily data is unavailable, daily tested is averaged across number of days in between.
14. `daily_positive`: Number of positive cases reported on the day; if daily data is unavailable, daily positive is averaged across number of days in.

Importing the data:

```
covid_df <- read_csv("covid19.csv") # import the data
```

The dimension of the dataframe:

```
dim(covid_df)
```

```
## [1] 10903    14
```

The column names of the dataframe:

```
vector_cols <- colnames(covid_df) #vector variable storing the column names
print(vector_cols)
```

```
##  [1] "Date"                    "Continent_Name"
##  [3] "Two_Letter_Country_Code" "Country_Region"
##  [5] "Province_State"          "positive"
##  [7] "hospitalized"            "recovered"
##  [9] "death"                   "total_tested"
## [11] "active"                  "hospitalizedCurr"
## [13] "daily_tested"            "daily_positive"
```

The first 6 rows of the dataframe:

```
head(covid_df) # displays the first 6 rows
```

```
## # A tibble: 6 x 14
##   Date       Continent_Name Two_Letter_Coun~ Country_Region Province_State
##   <date>     <chr>          <chr>            <chr>          <chr>
## 1 2020-01-20 Asia           KR               South Korea    All States
## 2 2020-01-22 North America  US               United States  All States
## 3 2020-01-22 North America  US               United States  Washington
## 4 2020-01-23 North America  US               United States  All States
## 5 2020-01-23 North America  US               United States  Washington
## 6 2020-01-24 Asia           KR               South Korea    All States
## # ... with 9 more variables: positive <dbl>, hospitalized <dbl>,
## #   recovered <dbl>, death <dbl>, total_tested <dbl>, active <dbl>,
## #   hospitalizedCurr <dbl>, daily_tested <dbl>, daily_positive <dbl>
```

The summary of the dataframe

```
glimpse(covid_df) # summarizes and shows the structure of the dataframe
```

```
## Rows: 10,903
## Columns: 14
## $ Date                    <date> 2020-01-20, 2020-01-22, 2020-01-22, 2...
## $ Continent_Name          <chr> "Asia", "North America", "North Americ...
## $ Two_Letter_Country_Code <chr> "KR", "US", "US", "US", "US", "KR", "U...
## $ Country_Region          <chr> "South Korea", "United States", "Unite...
## $ Province_State          <chr> "All States", "All States", "Washingto...
## $ positive                <dbl> 1, 1, 1, 1, 1, 2, 1, 1, 4, 0, 3, 0, 0,...
## $ hospitalized            <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ recovered               <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ death                   <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ total_tested            <dbl> 4, 1, 1, 1, 1, 27, 1, 1, 0, 0, 0, 0, 0...
## $ active                  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ hospitalizedCurr        <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ daily_tested            <dbl> 0, 0, 0, 0, 0, 5, 0, 0, 0, 0, 0, 0, 0,...
## $ daily_positive          <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
```

Data Preparation

At this point, I'll keep working with the dataframe. Remember that my aim is to extract data that is relevant to answering the main question.

As I said in the previous stage, there may be some irregularities in the dataset columns. Looking at the first few lines of the dataset in the previous section, I saw that the `Province_State` column combines data from two levels: nation and state/province. I need to restrict what I'm interested in because I can't do an analysis on all of these levels at the same time.

As a result, in order to avoid bias in my analysis, I will only extract data at the national level. To do so, I filter the data such that only data pertaining to "All States" remains. "All States" is the value of the field `Province_State` to indicate that COVID-19 data is only accessible at the nation level.

```
covid_df_all_states <- covid_df %>%
                       filter(Province_State == "All States") %>%
                       select(-Province_State)

head(covid_df_all_states, 3) # observe the first 3 rows
```

```
## # A tibble: 3 x 13
##   Date       Continent_Name Two_Letter_Coun~ Country_Region positive
##   <date>     <chr>          <chr>            <chr>             <dbl>
## 1 2020-01-20 Asia           KR               South Korea           1
## 2 2020-01-22 North America  US               United States         1
## 3 2020-01-23 North America  US               United States         1
## # ... with 8 more variables: hospitalized <dbl>, recovered <dbl>,
## #   death <dbl>, total_tested <dbl>, active <dbl>, hospitalizedCurr <dbl>,
## #   daily_tested <dbl>, daily_positive <dbl>
```

At this point, I'm still working with dataframes. My purpose is to obtain the information that will help me answer the questions.

I realized that there are columns that offer daily information and others that offer cumulative information after reviewing the description of the dataset columns. Because I can't deal with both scenarios (columns with cumulative and daily information) at the same time, I have to manage them independently. Actually, my analysis would be skewed if I compared a column containing cumulative data to another having only one-day data. This is another another example of a circumstance that I would like to be aware of from the start of the project in order to better evaluate the dataset.

Following that, I will mostly deal with daily data. So, let us extract the columns pertaining to the daily measurements.

```
covid_df_all_states_daily <- covid_df_all_states %>%
                             select(Date, Country_Region,
                                    active, hospitalizedCurr,
                                    daily_tested, daily_positive)

head(covid_df_all_states_daily, 3) # observe the first 3 rows
```

```
## # A tibble: 3 x 6
##   Date       Country_Region active hospitalizedCurr daily_tested
##   <date>     <chr>           <dbl>            <dbl>        <dbl>
## 1 2020-01-20 South Korea         0                0            0
## 2 2020-01-22 United States       0                0            0
## 3 2020-01-23 United States       0                0            0
## # ... with 1 more variable: daily_positive <dbl>
```

Data Analysis

My purpose here is to get data from the top 10 affected nations. These are the questions I am asking myself as a data scientist at this stage:

- Given that I presently have daily data, how can I obtain the total number of COVID-19 tested, positive, active, and hospitalized cases by country?
- So, how can I get the top ten?

The first question obfuscates the concepts of grouping (by nation) and data aggregation (summing daily information). These two concepts are related to the methods `group by()` and `summarize()`. The second question, on the other hand, is about the concept of order. The secret is that if I sort the dataset by the number of tests run, the top 10 represent the first 10 rows of the sorted dataset. As a result, I may organize the dataset using the method `arrange()` and retrieve the top 10 rows using the function `head()`.

Hence, I will summarize the dataframe `covid_df_all_states_daily` by computing the total number of tested, positive, active, and hospitalized cases. The aggregated data can then be arranged by the total number of tested cases. Finally, the first 10 rows may be extracted as the top 10 tested case nations.

```
covid_df_all_states_daily_sum <- covid_df_all_states_daily %>%
                                 group_by(Country_Region) %>%
                                 summarize(tested = sum(daily_tested),
                                           positive = sum(daily_positive),
                                           active = sum(active),
                                           hospitalized = sum(hospitalizedCurr)) %>%
                                 arrange(desc(tested))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
covid_df_all_states_daily_sum
```

```
## # A tibble: 108 x 5
##    Country_Region   tested positive  active hospitalized
##    <chr>             <dbl>    <dbl>   <dbl>        <dbl>
##  1 United States  17282363  1877179       0            0
##  2 Russia         10542266   406368 6924890            0
##  3 Italy           4091291   251710 6202214      1699003
##  4 India           3692851    60959       0            0
##  5 Turkey          2031192   163941 2980960            0
##  6 Canada          1654779    90873   56454            0
##  7 United Kingdom  1473672   166909       0            0
##  8 Australia       1252900     7200  134586         6655
##  9 Peru             976790    59497       0            0
## 10 Poland           928256    23987  538203            0
## # ... with 98 more rows
```

```
covid_top_10 <- head(covid_df_all_states_daily_sum, 10) # Top 10 affected nations
```

Now I can continue to extract my solution using the tibble dataframe and some sophisticated 'dplyr' verbs, but I'll illustrate how it can be done using other data structures, since that was our original goal for the project:

To accomplish my goal, I will extract vectors from the `covid_top_10` dataframe that will allow me to do specified operations.

```
# Extracting the vectors from the tibble dataframe

countries <- covid_top_10$Country_Region
tested_cases <- covid_top_10$tested
```

```r
positive_cases <- covid_top_10$positive
active_cases <- covid_top_10$active
hospitalized_cases <- covid_top_10$hospitalized

# Naming the vectors

names(tested_cases) <- countries
names(positive_cases) <- countries
names(active_cases) <- countries
names(hospitalized_cases) <- countries

# Identifying the top 3 positive against tested cases

result <-  sort(positive_cases/tested_cases, decreasing = TRUE)
positive_tested_top_3 <- result[1:3]
positive_tested_top_3
```

```
## United Kingdom  United States         Turkey
##     0.11326062     0.10861819     0.08071172
```

My objective is to develop a means to store all of the information for the top three nations with the largest number of positive instances compared to the number of tests available.

To ensure that I do not lose any additional information about these nations, I may develop a matrix that includes the ratio as well as the total number of COVID-19 tested, positive, active, and hospitalized cases.

```r
united_kingdom <- c(positive_tested_top_3["United Kingdom"],
                    tested_cases["United Kingdom"],
                    positive_cases["United Kingdom"],
                    active_cases["United Kingdom"],
                    hospitalized_cases["United Kingdom"])

united_states <- c(positive_tested_top_3["United States"],
                    tested_cases["United States"],
                    positive_cases["United States"],
                    active_cases["United States"],
                    hospitalized_cases["United States"])

turkey <- c(positive_tested_top_3["Turkey"],
                    tested_cases["Turkey"],
                    positive_cases["Turkey"],
                    active_cases["Turkey"],
                    hospitalized_cases["Turkey"])


covid_mat <- rbind(united_kingdom,united_states,turkey)
colnames(covid_mat) <- c("Ratio","tested","positive","active","hospitalized")

covid_mat
```

```
##                    Ratio   tested positive  active hospitalized
## united_kingdom 0.11326062  1473672   166909       0            0
## united_states  0.10861819 17282363  1877179       0            0
## turkey         0.08071172  2031192   163941 2980960            0
```

I deal with lists in the next step. My goal is to compile all of my answers and datasets. Because a list may

include several sorts of objects, I can store all of my project's data in one place. This gives me a global picture from a single variable, as well as the ability to export my results for other use.

Throughout the project, I've constructed numerous data structures, including: * Dataframes: `covid_df`, `covid_df_all_states`, `covid_df_all_states_daily`, and `covid_top_10`. * Matrix: `covid_mat`. * Vectors: `vector_cols` and `countries`.

I'll now make a list to keep all of my work in the same variable.

```
question <- "Which countries have had the highest number of positive cases against the number of tests?"
answer <- c("Positive tested cases" = positive_tested_top_3)

data_structure_list <- list("Dataframes" = list(covid_df,covid_df_all_states,
                                                covid_df_all_states_daily,covid_top_10),
                            "Matrices" = list(covid_mat),
                            "Vectors" = list(vector_cols,countries))

covid_analysis_list <- list(question,answer,data_structure_list)
covid_analysis_list[[2]]
```

```
## Positive tested cases.United Kingdom  Positive tested cases.United States
##                           0.11326062                           0.10861819
##         Positive tested cases.Turkey
##                           0.08071172
```

Conclusion

To summarize, I was able to demonstrate an analysis of Kaggle's Covid-19 Worldwide testing data utilizing R's data structures (vectors, matrices, dataframes, and list) for the period January 1, 2020 to June 30, 2020. My investigation revealed the following answer to the research question: **"Which countries had the greatest number of positive cases in relation to the overall number of tests performed during the specified time period?"** The following is the answer:

```
covid_analysis_list[[2]]
```

```
## Positive tested cases.United Kingdom  Positive tested cases.United States
##                           0.11326062                           0.10861819
##         Positive tested cases.Turkey
##                           0.08071172
```