

Dataquest Guided Project: Investigating COVID-19 Virus Trends

Cindy Zhang

Contents

Introduction	1
Loading the Data	1
Isolating the Necessary Rows	2
Isolating the Necessary Columns	3
Extracting Top Ten Tested Cases Countries	3
Identifying the Highest Positive Against Tested Cases	4
Keeping Relevant Information	4
Putting It All Together	4

Introduction

This is my solution to Dataquest's COVID-19 Guided Project from Course 2 (Data Structures in R). It answers the question: **Which countries have had the highest number of positive cases against the number of tests?**

More details, such as descriptions for variables, can be found in the "ReadMe" file of this project's repository in GitHub.

Loading the Data

```
covid_df <- data.frame(read.csv("covid19.csv"))
dim(covid_df)
```

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

```
vector_cols <- colnames(covid_df)
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"
```

```
head(covid_df)
```

##	Date	Continent_Name	Two_Letter_Country_Code	Country_Region			
## 1	2020-01-20	Asia	KR	South Korea			
## 2	2020-01-22	North America	US	United States			
## 3	2020-01-22	North America	US	United States			
## 4	2020-01-23	North America	US	United States			
## 5	2020-01-23	North America	US	United States			
## 6	2020-01-24	Asia	KR	South Korea			
##	Province_State	positive	hospitalized	recovered	death	total_tested	active
## 1	All States	1	0	0	0	4	0
## 2	All States	1	0	0	0	1	0
## 3	Washington	1	0	0	0	1	0
## 4	All States	1	0	0	0	1	0
## 5	Washington	1	0	0	0	1	0
## 6	All States	2	0	0	0	27	0
##	hospitalizedCurr	daily_tested	daily_positive				
## 1	0	0	0				
## 2	0	0	0				
## 3	0	0	0				
## 4	0	0	0				
## 5	0	0	0				
## 6	0	5	0				

```
glimpse(covid_df)
```

```
## Observations: 10,903
## Variables: 14
## $ Date <fct> 2020-01-20, 2020-01-22, 2020-01-22, 2020-01...
## $ Continent_Name <fct> Asia, North America, North America, North A...
## $ Two_Letter_Country_Code <fct> KR, US, US, US, US, KR, US, US, AU, AU, AU,...
## $ Country_Region <fct> South Korea, United States, United States, ...
## $ Province_State <fct> All States, All States, Washington, All Sta...
## $ positive <int> 1, 1, 1, 1, 1, 2, 1, 1, 4, 0, 3, 0, 0, 0, 0...
## $ hospitalized <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ recovered <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ death <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ total_tested <int> 4, 1, 1, 1, 1, 27, 1, 1, 0, 0, 0, 0, 0, 0, ...
## $ active <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ hospitalizedCurr <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ daily_tested <int> 0, 0, 0, 0, 0, 5, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ daily_positive <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
```

After downloading the `covid19.csv` file from Dataquest, I loaded the file and determined the dimension of the dataframe. I stored the column names as `vector_cols`, a character data structure. The `glimpse()` function is useful when exploring a new dataset because it makes it possible to see every column in a data frame and shows as much as it can so that you can visualize what the dataset looks like.

Isolating the Necessary Rows

As shown in the glimpse of the dataset above, the `Province_State` column mixes country and state/province level data. I was only interested in country-level data, so I filtered `covid_df` for rows that contain "All

States" in the Province_State column and stored it as covid_df_all_states. I then deleted Province_State from the new dataframe with the assurance that the new dataframe only contained country-level data.

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

Isolating the Necessary Columns

From covid_df_all_states, I extracted another subset, covid_df_all_states_daily, which contained only the dataset columns with daily information to avoid biasing the analysis by comparing daily data to cumulative data.

```
covid_df_all_states_daily <- subset(covid_df_all_states, select = c(Date, Country_Region, active, hospitali
```

Extracting Top Ten Tested Cases Countries

I calculated the sum of COVID-109 tested, positive, active, and hospitalized cases by country using the daily data; from the sum calculations, I extracted the top ten countries with the highest sums.

```
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), hospitali
  arrange(desc(tested))
covid_df_all_states_daily_sum
```

```
## # A tibble: 108 x 5
##   Country_Region tested positive active hospitalized
##   <fct>          <int>    <int>    <int>         <int>
## 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)
covid_top_10
```

```
## # A tibble: 10 x 5
##   Country_Region tested positive active hospitalized
##   <fct>          <int>    <int>    <int>         <int>
## 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
```

Identifying the Highest Positive Against Tested Cases

I extracted vectors from `covid_top_10` that allowed me to calculate the ratio of positive cases to tested cases and determine which three countries had the highest ratios.

```
countries <- covid_top_10$Country_Region
tested_cases <- covid_top_10$tested
positive_cases <- covid_top_10$positive
active_cases <- covid_top_10$active
hospitalized_cases <- covid_top_10$hospitalized
```

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

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

Keeping Relevant Information

I created a matrix that contained just the top three countries' COVID-19 information:

```
united_kingdom <- c(0.11, 1473672, 166909, 0, 0)
united_states <- c(0.10, 17282363, 1877179, 0, 0)
turkey <- c(0.08, 2031192, 163941, 2980960, 0)
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 0.11  1473672   166909         0          0
## united_states  0.10 17282363  1877179         0          0
## turkey         0.08  2031192   163941 2980960         0
```

Putting It All Together

Lastly, I stored all answers and datasets together in one list, `covid_analysis_list`. The second element of this list is printed below the code:

```

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)
dataframes <- c(covid_df, covid_df_all_states, covid_df_all_states_daily, covid_df_all_states_daily_sum)
matrices <- covid_mat
vectors <- c(active_cases, countries, hospitalized_cases, positive_cases, positive_tested_ratio, positive_tested_top_3)
data_structure_list <- c(dataframes, matrices, vectors)
covid_analysis_list <- c(question, answer, data_structure_list)
covid_analysis_list[2]

```

```

## $'Positive tested cases.United Kingdom'
## [1] 0.113

```