Guided Project: Investigating COVID-19 Virus Trends

Marcella Pedro

17/11/2021

Objetivo

Neste projeto iremos avaliar dados divulgados no Kaggle e trabalhados pela Dataquest que podem ser acessados através deste link. São dados sobre o ínicio da pandemia de COVID-19 (COrona VIrus Disease), sendo assim o dataset possui dados coletados entre 20/01/2020 e 01/06/2020 e nosso objetivo é ao final desse projeto conseguir responder à seguinte questão:

Quais países tiveram os maiores números de casos positivos comparado ao número de testes realizados?

Bibliotecas importadas ao longo do projeto

```
library(readr)  # para importação de arquivos csv
library(glue)   # para prints inteligentes
library(dplyr)  # manipulação e análise de dataframes
```

Conhecendo os dados

Importar a base

```
covid_df <- read_csv("~/Documentos/covid19.csv")
```

```
## Rows: 10903 Columns: 14
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): Continent_Name, Two_Letter_Country_Code, Country_Region, Province_...
## dbl  (9): positive, hospitalized, recovered, death, total_tested, active, ho...
## date (1): Date
```

```
##
## 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.
```

Tamanho da base

Apesar de a própria função de importação trazer essa informação temos algumas alternativas:

```
tamanho <- dim(covid_df)
glue("Tamanho da base: {tamanho[1]} linhas e {tamanho[2]} colunas")
```

```
## Tamanho da base: 10903 linhas e 14 colunas
```

ou

```
linhas  <- nrow(covid_df)
colunas <- ncol(covid_df)

glue("Tamanho da base: {linhas} linhas e {colunas} colunas")
```

```
## Tamanho da base: 10903 linhas e 14 colunas
```

ou (melhor ainda)

glimpse

```
glimpse(covid_df)
```

```
## Rows: 10,903
## Columns: 14
## $ Date                    <date> 2020-01-20, 2020-01-22, 2020-01-22, 2020-01-2~
## $ Continent_Name          <chr> "Asia", "North America", "North America", "Nor~
## $ Two_Letter_Country_Code <chr> "KR", "US", "US", "US", "US", "KR", "US", "US"~
## $ Country_Region          <chr> "South Korea", "United States", "United States~
## $ Province_State          <chr> "All States", "All States", "Washington", "All~
## $ positive                <dbl> 1, 1, 1, 1, 1, 2, 1, 1, 4, 0, 3, 0, 0, 0, 0, 1~
## $ hospitalized            <dbl> 0, 0, 0, 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, 0, 0, 0~
## $ death                   <dbl> 0, 0, 0, 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, 0, 0, ~
## $ active                  <dbl> 0, 0, 0, 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, 0, 0, 0~
## $ daily_tested            <dbl> 0, 0, 0, 0, 0, 5, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
## $ daily_positive          <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
```

Acima com a função `glimpse` foi possível em um único comando ver a quantidade de linhas, colunas, tipo de dados e exemplos dos dados de cada coluna.

No caso de utilizar os primeiros métodos de verificação do tamanho da base, poderia ser combinado com as funções `head` ou `tail` que apenas traz exemplo dos dados, e o `colnames` ou somente `names` para identificar o nome das colunas.

Dicionário de dados

Informação disponibilizada neste caso pelo Dataquest traduzida livremente, muito importante entender se de fato o nome das colunas tem um nome condizente com seu significado, ou tem um nome que pode levar à má interpretação.

Esse material servirá de apoio para os demais passos do projeto.

Campo	Significado
Date	Data da coleta
Continent_Name	Nome do continente da ocorrência
Two_Letter_Country_Code	Código do país da ocorrência
Province_State	Informção do estado ou província específica, ou então consolidada "All States"
positive	Casos positivos reportados acumulados
active	Casos ativos do dia
hospitalized	Número de hospitalizados reportados acumulados
hospitalizedCurr	Número de hospitalizados reportados do dia
recovered	Número de recuperados reportados acumulados
death	Número de mortes reportadas acumuladas
total_tested	Número de testes realizados reportados acumulados
daily_tested	Número de testes realizados reportados no dia (se não houver dados é a média)
daily_positive	Número de casos positivos reportados no dia (se não houver dados é a média)

Entendendo os dados a fundo

Filtrar dados a serem analisados

Um dataset costuma ter muito mais informação do que precisamos para responder a pergunta que levou à análise. Nesse caso vamos focar apenas no nível dos países e deixar os detalhes de cada província/estado para um outro momento.

Filtrar dados a nível de país

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

```
## Rows: 3,781
## Columns: 13
## $ Date                   <date> 2020-01-20, 2020-01-22, 2020-01-23, 2020-01-2~
## $ Continent_Name         <chr> "Asia", "North America", "North America", "Asi~
## $ Two_Letter_Country_Code <chr> "KR", "US", "US", "KR", "US", "AU", "GB", "US"~
## $ Country_Region         <chr> "South Korea", "United States", "United States~
## $ positive               <dbl> 1, 1, 1, 2, 1, 4, 1, 1, 4, 0, 3, 1, 1, 5, 0, 0~
## $ hospitalized           <dbl> 0, 0, 0, 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, 0, 0, 0~
## $ death                  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
## $ total_tested           <dbl> 4, 1, 1, 27, 1, 0, 31, 1, 0, 3, 51, 52, 1, 0, ~
## $ active                 <dbl> 0, 0, 0, 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, 0, 0, 0~
## $ daily_tested           <dbl> 0, 0, 0, 5, 0, 0, 0, 0, 0, 0, 12, 21, 0, 0, 0,~
## $ daily_positive         <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0~
```

Já que filtramos um único valor do campo `Province_State` não precisamos desse campo para diferenciar os dados. Outro ponto é que de 10 mil registros aproximadamente agora vamos focar apenas e em torno de 3 mil deles.

Filtrar dados com informação diária

Não queremos confundir com os dados acumulados, então é uma boa ideia focar no que precisamos.

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

```
## Rows: 3,781
## Columns: 6
## $ Date             <date> 2020-01-20, 2020-01-22, 2020-01-23, 2020-01-24, 2020~
## $ Country_Region   <chr> "South Korea", "United States", "United States", "Sou~
## $ active           <dbl> 0, 0, 0, 0, 0, 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, 0, 0, 0, 0, 0,~
## $ daily_tested     <dbl> 0, 0, 0, 5, 0, 0, 0, 0, 0, 0, 12, 21, 0, 0, 0, 1, 10,~
## $ daily_positive   <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 1,~
```

Agregar dados por país

Uma vez que temos os dados necessários precisamos trabalha-los, realizar cálculos a fim de chegar no resultado que responde a pergunta inicial.

Aqui a ideia é consolidar os dados de todo o período numa única linha por país e os valores serão somados e ordenados.

```
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(-tested)
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
```

Como queremos os maiores número podemos continuar a análise com os 10 maiores.

```r
covid_top_10 <- head(covid_df_all_states_daily_sum, n=10)
covid_top_10
```

```
## # A tibble: 10 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
```

```r
covid_final <- covid_top_10 %>% mutate(ratio = positive/tested) %>%
                                arrange(-ratio)
covid_final
```

```
## # A tibble: 10 x 6
##    Country_Region   tested positive  active hospitalized   ratio
##    <chr>             <dbl>    <dbl>   <dbl>        <dbl>   <dbl>
##  1 United Kingdom  1473672   166909       0            0 0.113
##  2 United States  17282363  1877179       0            0 0.109
##  3 Turkey          2031192   163941 2980960            0 0.0807
##  4 Italy           4091291   251710 6202214      1699003 0.0615
##  5 Peru             976790    59497       0            0 0.0609
##  6 Canada          1654779    90873   56454            0 0.0549
##  7 Russia         10542266   406368 6924890            0 0.0385
##  8 Poland           928256    23987  538203            0 0.0258
##  9 India           3692851    60959       0            0 0.0165
## 10 Australia       1252900     7200  134586         6655 0.00575
```

Por fim conseguimos responder à pergunta inicial, os 3 países que mais tiveram casos positivos em relação aos testes realizados são:

```r
head(covid_final[,c("Country_Region","ratio")], n=3)
```

```
## # A tibble: 3 x 2
##   Country_Region  ratio
##   <chr>           <dbl>
## 1 United Kingdom 0.113
## 2 United States  0.109
## 3 Turkey         0.0807
```