DATA100 Group Project – 2020 Fall

Shengda Hu

02/11/2020

```
writeLines("Team Name: Covid Combatters")
```

```
## Team Name: Covid Combatters
```

```
tribble(
  ~"Last Name (Family)", ~"First Name (Given)", ~"Student ID",
  "Ashraf ", "Hafsah", 201911240,
  "Coonjobeeharry", "Rajveer", 170707330,
  "Lo", "Mariam", 200368340,
  "Palar", "Valencia Isabelle", 203302280,
  "Vangipuram", "Vivek", 200797670
)
```

```
## # A tibble: 5 x 3
##   `Last Name (Family)` `First Name (Given)` `Student ID`
##   <chr>                <chr>                       <dbl>
## 1 "Ashraf "            Hafsah                  201911240
## 2 "Coonjobeeharry"     Rajveer                 170707330
## 3 "Lo"                 Mariam                  200368340
## 4 "Palar"              Valencia Isabelle       203302280
## 5 "Vangipuram"         Vivek                   200797670
```

Project introduction

Many things have happened and are happening in this year of 2020. The longest lasting, most widespread and probably defining event of the year so far – aside from what happens tomorrow – looks to be the COVID-19 pandemic. This project will ask you to use the methods and techniques we learned in DATA100 to get some understanding of this recent historical event.

Since many events are still unfolding and whatever data are out there are constantly being updated, revised, debated and reinterpreted, the understanding that will come out from this project will inevitably be incomplete at the best, probably inconclusive and plainly unreasonable at worst. So the main goal of the project is not to come up with the most reasonable or objective interpretations of the data or events involved, as what look reasonable now might become way off the mark as more information comes to light. The main goal of this project is to understand as much as possible what the stories the data sets available might tell. Put it in the cliched language: "let the data talk", or in the more interesting phrase: "let the data ask questions".

The theme of the story that we would like to understand is the following:

What factors can be related to the level of observed infection / recovery / death by COVID-19 at a given time and given region.

A most simple minded answer would be *everything*, because COVID-19 has definitely touched upon all facets of life. Through out this course, we are learning tools for "torturing the data until it confesses", and the project is an attempt at teasing out some more detailed information. Note that the term **relationship** may be interpreted at least in the following three categories:

1) Causes higher / lower levels of COVID-19 infection / recovery / death

2) Caused by higher / lower levels of COVID-19 infection / recovery / death

3) Shows a correlation but causality unclear

There are more sophisticated methods that can provide more information to distinguish these three interpretations. For this project, it would be enough to give an intuitive interpretation in terms of one of these categories if you identified any relationship among the various factors. It is also completely reasonable that, from the data sets we have, it may appear that some factors do not correlate much to the COVID-19 – which is also knowledge gained.

In the following, besides the **3** online data sets on COVID-19, we provide **22** data sets concerning a number of potential factors of interest, such as *educational*, *political*, *economical*, *technological*, *employment*, *health*, *demographic*, *self perception* factors. As you can see, a number of them are not up-to-date, which is due to the availability of timely data – most of the interesting current data are not open data, or not easy to locate in more readily useful form to us. A number of COVID-19 related data sets are included, which by the collective work of many organizations, are updated real time. You are encouraged to track the most up-to-date version. We included the *WorldRegions.csv* data from World Regions Classification list on Wikipedia.

Also included are the data *WorldHappinessReport2020-Score.csv* from the World Happiness Report `2020`, which concerns the years `2017-2019`. It is computed based on the answers of people to the following question: "Please imagine a ladder, with steps numbered from 0 at the bottom to 10 at the top. The top of the ladder represents the best possible life for you and the bottom of the ladder represents the worst possible life for you. On which step of the ladder would you say you personally feel you stand at this time?" (Statistical Appendix for Chapter 2 of World Health Report 2020) Thus, the score can be seen as giving one interpretation of happiness.

Descriptions of the Online Datasets and Chosen Sets

Online Datasets

covid_cases: Gives an in depth run down of all the countries and the dates starting from the beginning of the year until realtime. Each variable provides a good statistic of deaths or cases by different time spans. The variables vary from the total overall cases and deaths by each timestamp, the new cases and deaths at that timestamp. and then for each biweekly, weekly, and daily time marker it gives an update on the amount of cases and deaths for that timestamp. Overall it is very useful in gauging the overall COVID-19 case increase and decrease on certain dates and how they span over various timeframes. The way we are using it is to compare the overall COVID cases and deaths by specific countries and regions. We are focusing on the dates leading up to september 1st as that is the timeframe we have chosen. We also decided to make graphs comparing total cases and total deaths in specific as we feel it is the best measure for this dataset and for the setup of our project to give us the conclusions we are looking for. The original data set had 10 columns and 59354 rows. The columns represented different information about Covid-19 in different countries like the number of new cases, weekly deaths, total cases in the country, etc. The variable types for all the columns were "double", other than the location which represented the date ("date" type), and the location columns ("character" type). This data had a lot of information that we could use for our project We used this and also picked specific countries like Canada, United States, and Italy, since those had high rates for Covid-19 cases, and made a new data set called covid_cases_filtered to use for our graphs. This resulted in a new data set of 10 columns and 245 rows. This made it much easier for our group to make several different graphs that we can analyze and make conclusions.

covid_tests: The covid tests data set provides information about Covid-19 tests performed in over 94 countries. It displays the country name, the number of tests performed, and the month it was performed. Similar to some of the other data sets, "covid_tests" was also untidy. It had many unknown values, and some columns and countries were just not necessary for our project. Therefore, we decided to filter out the data set and chose Canada, the United States, and Italy to analyze, since those countries had many covid tests performed. We also filtered out the date so the dataset shows the number of tests performed from January 1st, 2020, up until September 1st, 2020(the last date that we chose as seen in question 1). Finally, we decided to take out all the columns starting from "Source URL" up to "Notes", since those weren't going to be needed in our graphs. Our final dataset had 11 columns and 184 rows. The variable types were also correct, so we didn't need to change any(for example, Country was a character variable, Date was a date variable, Cumulative total was a double, etc).

covid_response: The Covid response data set gave us information about government policies in different countries around the world. The scale was according to the level of precautions enforced by government officials regarding school and work closures, travel plans, closing public events, etc. The function glimpse(), allowed us to learn more about the data set, and find the dimensions of the data. The unfiltered data consisted of 47 columns and 64925 rows. We did not need all of this data for our project, therefore we used to filter() to only display columns that were necessary. This includes Canada, the United States, and Italy. These countries had a high number of cases compared to other countries. We also ensured that the values under the columns "C1_School closing" and "C4_Restrictions on gatherings" were above zero, to make sure that they were relevant. In addition, we only selected a few columns to use in our final dataset to use while creating our graphs. Therefore, our final, organized dataset for Covid response, which is called "covid_response_gatherings_filtered", has only five columns, and 4,491 rows (which contained the countries that we selected earlier).

Chosen Datasets

lifeexpect: Explanation

population: The population dataset is very straightforward as it gives the overall population of a country since July of 2020. It gives the amount and lists the countries by rank, Number one being the most populated country which is China, and 238 being the Pitcairn islands. The main usage for this dataset is to use it for each country to compare individual statistics to the overall population. It is a very simple but very effective dataset that is applied in various scenarios. One to name is compared ot realtime datasets such as the covid cases to see the ratio and percentage of a country infected to compare it to other countries. It gives a measure and insight as to how well a country has stopped the spread, and dealt with the spread.

birthrate: Explanation

unemployment: The Unemployment rate dataset provided us with information on the unemployment rate from April to September in the year 2020. This data set allowed us to make conclusions regarding the different factors that may have been affected or further affected after the start of the COVID pandemic. We used the ggplot() function to create a boxplot that would display the data in a neat and organized manner. The

unemp_youth: Explanation

laborforce: The labor force data set is an up to date dataset that shows the labor force in each country from the year 1990 to 2020. The original data set was untidy, contained many missing values, and most column names were actual values. To make it easier to read and analyze, we decided to gather the years and put them into one variable (key="year"), and the population in another, value="population. The final dataset had 4 columns and 8184 rows, which represented the different countries in alphabetical order.

gdppp: Explanation

healthexp: Explanation

regionclassification: Explanation

The Map

The map below shows the `new cases` on Oct. 31, 2020 obtained from Our world in data.

```
COVID_cases <- read_csv("COVID-2020-10-31.csv",
                        col_types = cols(
                          location = col_character(),
                          new_cases = col_double(),
                          total_cases = col_double()
                        ))
COVID_cases
```

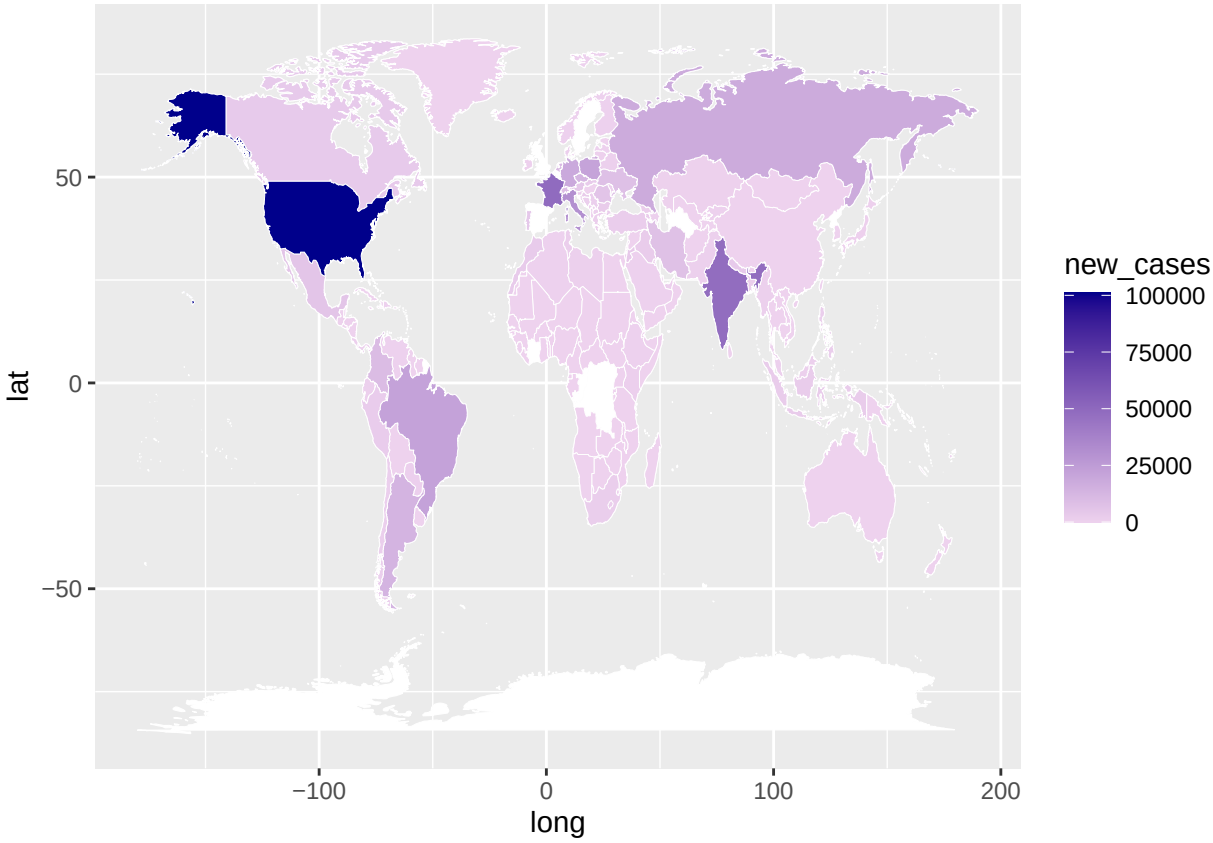
```
## # A tibble: 210 x 3
##    location            new_cases total_cases
##    <chr>                   <dbl>       <dbl>
##  1 Afghanistan               157       41425
##  2 Albania                   319       20634
##  3 Algeria                   319       57651
##  4 Andorra                    98        4665
##  5 Angola                    195       10269
##  6 Anguilla                    0           3
##  7 Antigua and Barbuda         3         127
##  8 Argentina               13955     1157166
##  9 Armenia                  2381       89813
## 10 Aruba                      17        4472
## # ... with 200 more rows
```

```
world <- map_data("world")

iu <- COVID_cases %>% rename (region = location)

iu$region[198] <- "USA" # to match world map data

iu <- semi_join(iu, world, by = "region") #only keep countries according to world map data

# code below is modified from
# https://stackoverflow.com/questions/29614972/ggplot-us-state-map-colors-are-fine-polygons-jagged-r
gg <- ggplot()

gg <- gg + geom_map(
  data = world,
  map = world,
  aes(x = long, y = lat, map_id = region),
  fill = "#ffffff",
  color = "#ffffff",
  size = 0.20
  )
```

```
## Warning: Ignoring unknown aesthetics: x, y
```

```
  gg <- gg + geom_map(
  data = iu,
```

```
map = world,
aes(fill = new_cases, map_id = region),
color = "#ffffff",
size = 0.15
)

gg <- gg + scale_fill_continuous(low = 'thistle2', high = 'darkblue',
guide = 'colorbar')
gg
```



The Setup:

- Choose two dates to do your analysis, which have to be in different periods in the year 2020 as outlined below:
  - Before April 01
  - Between April 01 and September 01
  - After September 01
- From the **22** data sets provided below in the `.csv` files, select a subset (of at least `8`), covering at least `3` of the factors mentioned above. Describe how the data sets selected measure the factors chosen. I expect different groups would choose different subsets to work with.
- You need to include all **3** real-time online data sets, i.e. `COVID-19 cases`, `COVID-19 government responses` and `COVID-19 testing`. Besides the `8` data sets mentioned above, the real-time online

data sets on `COVID-19 government responses` and `COVID-19 testing` must be included in your discussion.

How the data sets selected measure the factors chosen: (Group Answer)

The data sets we selected are death, life expectancy, population, population distribution, unemployment rate, unemployed youth, labor force, GDP per capita, and expenditure. The unemployment rate, unemployed youth, expenditure, and GDP per capita all provide information about the employment factor during the time of COVID, which then also affects the economical factor. The death rate, life expectancy, and population count all provide information on the health and demographic factor of each country. The number of unemployed youth may also provide information to make conclusions about the education of a certain country and provide some insights about economic conditions.

The Questions:

Q: Provide a brief justification of the choice your group makes about the dates. `Random choice` is an acceptable justification.

A: Based on the dates given, our group chose the dates from before April 1st and from April 1st to September 1st. We picked these dates because they seem to be the peak periods of the first wave of COVID-19. These dates cover a time period where the cases were increasing at a high exponential rate. We thought that the data from before April 1st to September 1st would be the most helpful when analyzing and answering questions for this project. For the most part, the period of time before April 1st was the time when COVID was most active. While from April 1st to September 1st, COVID was relatively milder and includes the "core" part of the first wave.

Q: Form your own opinion concerning which factors are most likely to affect / be affected by the COVID-19 infection / recovery / death of a region, on the dates you selected. Note that most of the data sets are for years prior to `2020`.

A:After discussing as a group, we concluded that in the time period we chose, the overall health and economy of every country was negatively affected the most. People were not as aware of the dangers of COVID and the importance of making efforts to prevent its spread at the time. During the peak of the COVID pandemic, the overall health of most countries depleted not only due to the drastic increase of COVID cases, but also because many people who needed treatment for diseases such as cancer, diabetes, etc., had not been receiving the medical attention they need to remain healthy. Health services in most countries had also been disrupted as many health workers at the time were reassigned to deal with severe COVID cases. The cancellation of planned treatment also terminated health services at the time.

Q: Based on the interpretation of your group, analyze how the factors affect / are affected by the COVID-19 infection / recovery / death of a region, on the dates you selected, as represented by the real-time online data sets on `COVID-19 cases`.

A:

Q: For the two chosen dates, for different regions, do you see the relationship you describe using the data sets change? What could be the potential reasons for such changes?

The Data sets

There are a total of **25** data sets, **3** of which are online real-time data sets that are regularly updated, while the remaining **22** can be obtained as `csv` files on MyLS. You may need to make the data tidy for some of

them. Please note that the data sets are from different sources, you may need to first make sure, for example, the country / region names provided indeed do correspond.

The sources of the data are contained in the hyperlink. They are the following:

- CIA World FactBook: from which we obtained a majority of the data sets as `.csv` files
- World Bank Data: from which we obtained the data sets on Access to Electricity, Internet Usage, and Labor Force data
- Our World in Data: from which we obtain the online data sets on COVID-19 cases and testing
- Economist Intelligence Unit: which developed the democracy index. The version we use is from the Wikipedia page.
- University of Oxford: from which we obtain the online data set on Government Response
- United Nations: from which we obtained the data set on Population Distribution by Age and Gender
- Wikipedia: from which we obtained the World Regions Classification data set, aside from the democracy index above

Real-time COVID-19 data sets:

These data are regularly updated, and they contain all the historical data, which include the periods that we are interested in. Once you choose and fixed the dates to work with, the updates should not affect your report.

COVID-19 cases (Our world in data)

```
covid_cases <- read_csv("https://covid.ourworldindata.org/data/ecdc/full_data.csv",
                        col_types = cols(
                          .default = col_double(),
                          date = col_date(format =""),
                          location = col_character()
                        ))%>%
  filter(between(date,as.Date("2020-01-01"),as.Date("2020-09-01")))
covid_cases
```

```
## # A tibble: 40,461 x 10
##    date       location new_cases new_deaths total_cases total_deaths
##    <date>     <chr>        <dbl>      <dbl>       <dbl>        <dbl>
##  1 2020-01-01 Afghani~         0          0          NA           NA
##  2 2020-01-02 Afghani~         0          0          NA           NA
##  3 2020-01-03 Afghani~         0          0          NA           NA
##  4 2020-01-04 Afghani~         0          0          NA           NA
##  5 2020-01-05 Afghani~         0          0          NA           NA
##  6 2020-01-06 Afghani~         0          0          NA           NA
##  7 2020-01-07 Afghani~         0          0          NA           NA
##  8 2020-01-08 Afghani~         0          0          NA           NA
##  9 2020-01-09 Afghani~         0          0          NA           NA
## 10 2020-01-10 Afghani~         0          0          NA           NA
## # ... with 40,451 more rows, and 4 more variables: weekly_cases <dbl>,
## #   weekly_deaths <dbl>, biweekly_cases <dbl>, biweekly_deaths <dbl>
```

```
covid_cases%>%filter(location == "Canada")
```

```
## # A tibble: 245 x 10
##    date       location new_cases new_deaths total_cases total_deaths
```

```
##    <date>     <chr>        <dbl>      <dbl>       <dbl>        <dbl>
##  1 2020-01-01 Canada           0          0          NA           NA
##  2 2020-01-02 Canada           0          0          NA           NA
##  3 2020-01-03 Canada           0          0          NA           NA
##  4 2020-01-04 Canada           0          0          NA           NA
##  5 2020-01-05 Canada           0          0          NA           NA
##  6 2020-01-06 Canada           0          0          NA           NA
##  7 2020-01-07 Canada           0          0          NA           NA
##  8 2020-01-08 Canada           0          0          NA           NA
##  9 2020-01-09 Canada           0          0          NA           NA
## 10 2020-01-10 Canada           0          0          NA           NA
## # ... with 235 more rows, and 4 more variables: weekly_cases <dbl>,
## #   weekly_deaths <dbl>, biweekly_cases <dbl>, biweekly_deaths <dbl>
```

```
covid_cases_filtered<-covid_cases%>%
  filter(location == c("Canada", "United States", "Italy", "Russia", "Australia", "Egypt","South Africa
```

```
## Warning in location == c("Canada", "United States", "Italy", "Russia",
## "Australia", : longer object length is not a multiple of shorter object length
```

```
covid_cases_filtered_CAN_USA_ITA<- covid_cases_filtered %>%
  filter(location == c("Canada", "United States", "Italy"))
```

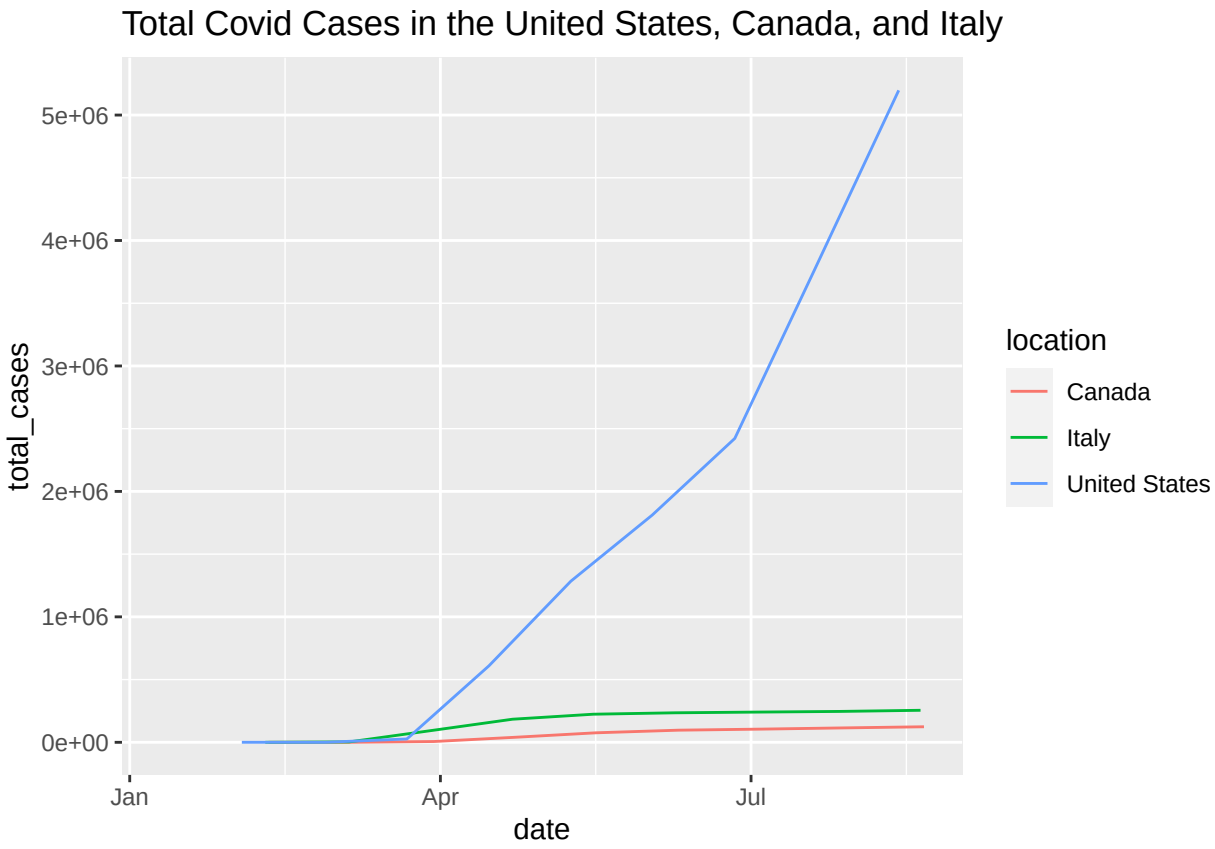
```
## Warning in location == c("Canada", "United States", "Italy"): longer object
## length is not a multiple of shorter object length
```

```
covid_cases_filtered_RUS_AUS_EGY<-covid_cases%>%
  filter(location == c("Russia", "Australia", "Egypt"))

covid_cases_filtered_ZEF_IND<-covid_cases%>%
  filter(location == c("South Africa", "India"))
```

```
## Warning in location == c("South Africa", "India"): longer object length is not a
## multiple of shorter object length
```

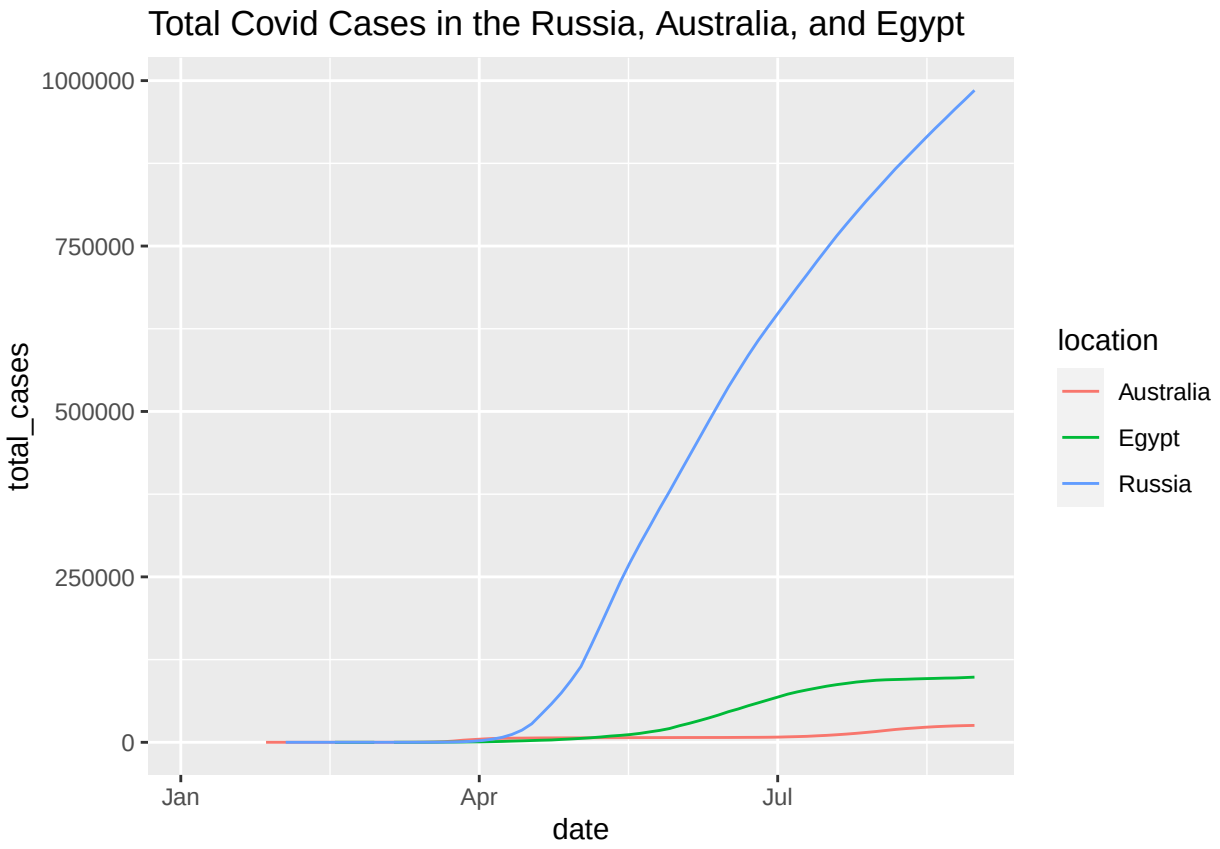
```
covid_cases_filtered_CAN_USA_ITA%>%
  ggplot()+ geom_line(mapping = aes(x = date, y = total_cases, color = location), position = "Jitter") +
  labs(title = "Total Covid Cases in the United States, Canada, and Italy")
```

```
## Warning: Removed 3 row(s) containing missing values (geom_path).
```



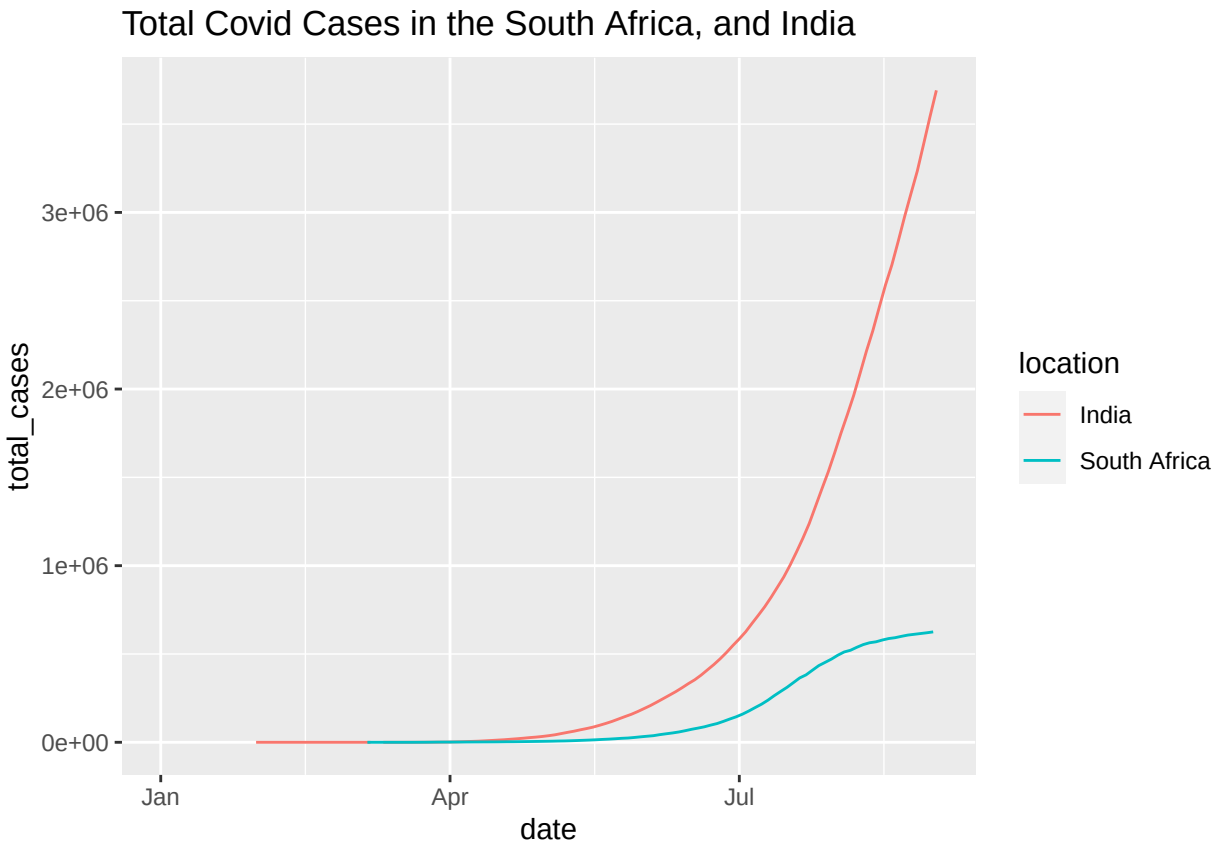

```
covid_cases_filtered_RUS_AUS_EGY%>%
  ggplot()+ geom_line(mapping = aes(x = date, y = total_cases, color = location))+
  labs(title = "Total Covid Cases in the Russia, Australia, and Egypt")
```

```
## Warning: Removed 33 row(s) containing missing values (geom_path).
```




```
covid_cases_filtered_ZEF_IND%>%
  ggplot() + geom_line(mapping = aes(x = date, y = total_cases, color = location))+
  labs(title = "Total Covid Cases in the South Africa, and India")
```

```
## Warning: Removed 15 row(s) containing missing values (geom_path).
```




```
covid_cases_filtered
```

```
## # A tibble: 238 x 10
##    date       location new_cases new_deaths total_cases total_deaths
##    <date>     <chr>        <dbl>      <dbl>       <dbl>        <dbl>
##  1 2020-01-05 Austral~         0          0          NA           NA
##  2 2020-01-13 Austral~         0          0          NA           NA
##  3 2020-01-21 Austral~         0          0          NA           NA
##  4 2020-01-29 Austral~         0          0           4           NA
##  5 2020-02-06 Austral~         1          0          13           NA
##  6 2020-02-14 Austral~         1          0          15           NA
##  7 2020-02-22 Austral~         4          0          21           NA
##  8 2020-03-01 Austral~         1          1          26            1
##  9 2020-03-09 Austral~         6          0          80            3
## 10 2020-03-17 Austral~        77          0         375            5
## # ... with 228 more rows, and 4 more variables: weekly_cases <dbl>,
## #   weekly_deaths <dbl>, biweekly_cases <dbl>, biweekly_deaths <dbl>
```

COVID-19 testing (Our world in data)

Hasell, J., Mathieu, E., Beltekian, D. et al. A cross-country database of COVID-19 testing. Sci Data 7,

```
covid_tests <- read_csv("https://covid.ourworldindata.org/data/testing/covid-testing-all-observations.cs
```

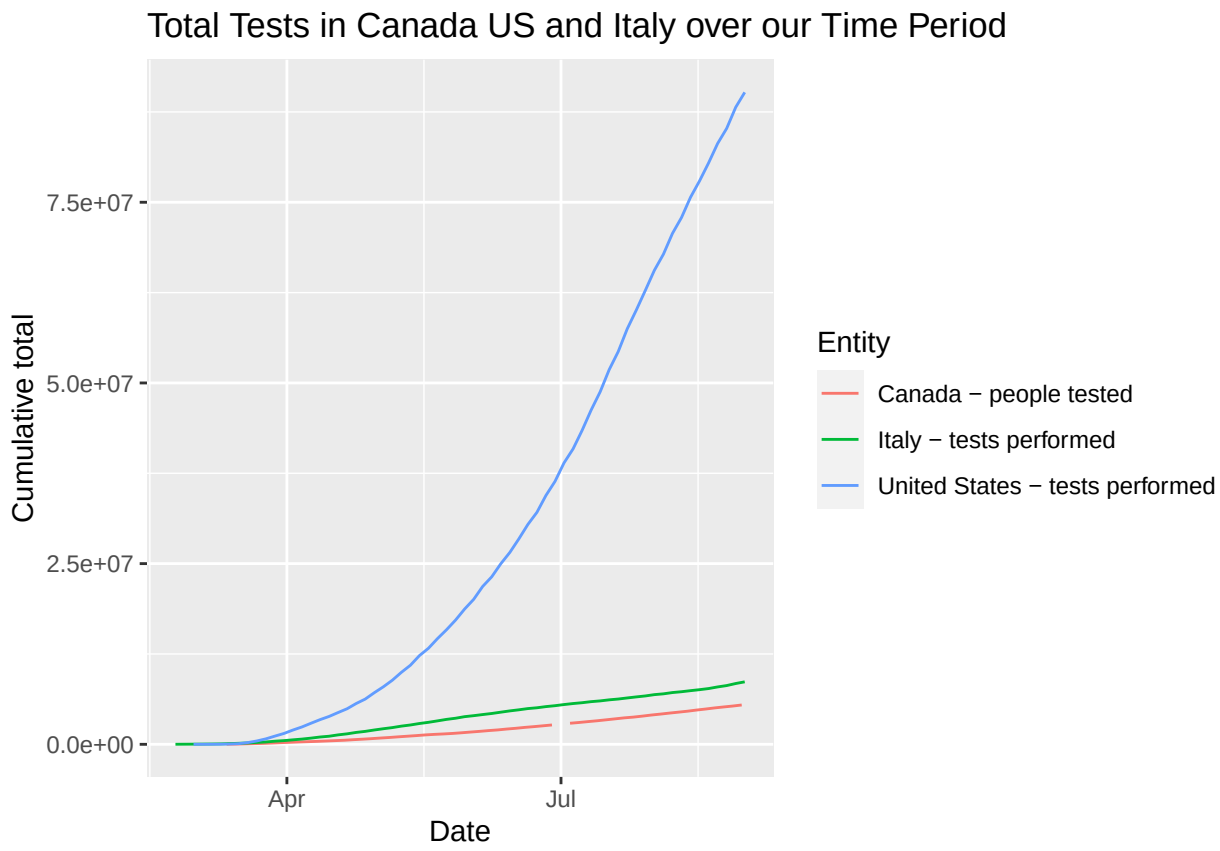
```
## Parsed with column specification:
## cols(
##   Entity = col_character(),
##   `ISO code` = col_character(),
##   Date = col_date(format = ""),
##   `Source URL` = col_character(),
##   `Source label` = col_character(),
##   Notes = col_character(),
##   `Daily change in cumulative total` = col_double(),
##   `Cumulative total` = col_double(),
##   `Cumulative total per thousand` = col_double(),
##   `Daily change in cumulative total per thousand` = col_double(),
##   `7-day smoothed daily change` = col_double(),
##   `7-day smoothed daily change per thousand` = col_double(),
##   `Short-term positive rate` = col_double(),
##   `Short-term tests per case` = col_double()
## )
```

```
covid_tests_filtered <- covid_tests %>%
  filter(between(Date,as.Date("2020-01-01"),as.Date("2020-09-01"))) %>%
  filter(Entity == c("United States - tests performed", "Canada - people tested", "Italy - tests perfor
  select(-(`Source URL`:`Notes`))
```

```
## Warning in Entity == c("United States - tests performed", "Canada - people
## tested", : longer object length is not a multiple of shorter object length
```

```
covid_tests_filtered%>% ggplot(aes(x = Date, y = `Cumulative total`)) +
  geom_line(aes(color = Entity), se = TRUE) + labs(title = "Total Tests in Canada US and Italy over our
```

```
## Warning: Ignoring unknown parameters: se
```




```
covid_tests_filtered
```

```
## # A tibble: 184 x 11
##    Entity `ISO code` Date       `Daily change i~ `Cumulative tot~
##    <chr>  <chr>      <date>                <dbl>            <dbl>
##  1 Canad~ CAN        2020-03-12             4162            15185
##  2 Canad~ CAN        2020-03-15             2568            24977
##  3 Canad~ CAN        2020-03-18             9370            53546
##  4 Canad~ CAN        2020-03-21            12069            88883
##  5 Canad~ CAN        2020-03-24            17915           125062
##  6 Canad~ CAN        2020-03-27             9041           170644
##  7 Canad~ CAN        2020-03-30            15270           225705
##  8 Canad~ CAN        2020-04-02            16733           273666
##  9 Canad~ CAN        2020-04-05            12820           324791
## 10 Canad~ CAN        2020-04-08            13864           361969
## # ... with 174 more rows, and 6 more variables: `Cumulative total per
## #   thousand` <dbl>, `Daily change in cumulative total per thousand` <dbl>,
## #   `7-day smoothed daily change` <dbl>, `7-day smoothed daily change per
## #   thousand` <dbl>, `Short-term positive rate` <dbl>, `Short-term tests per
## #   case` <dbl>
```

COVID-19 government responses

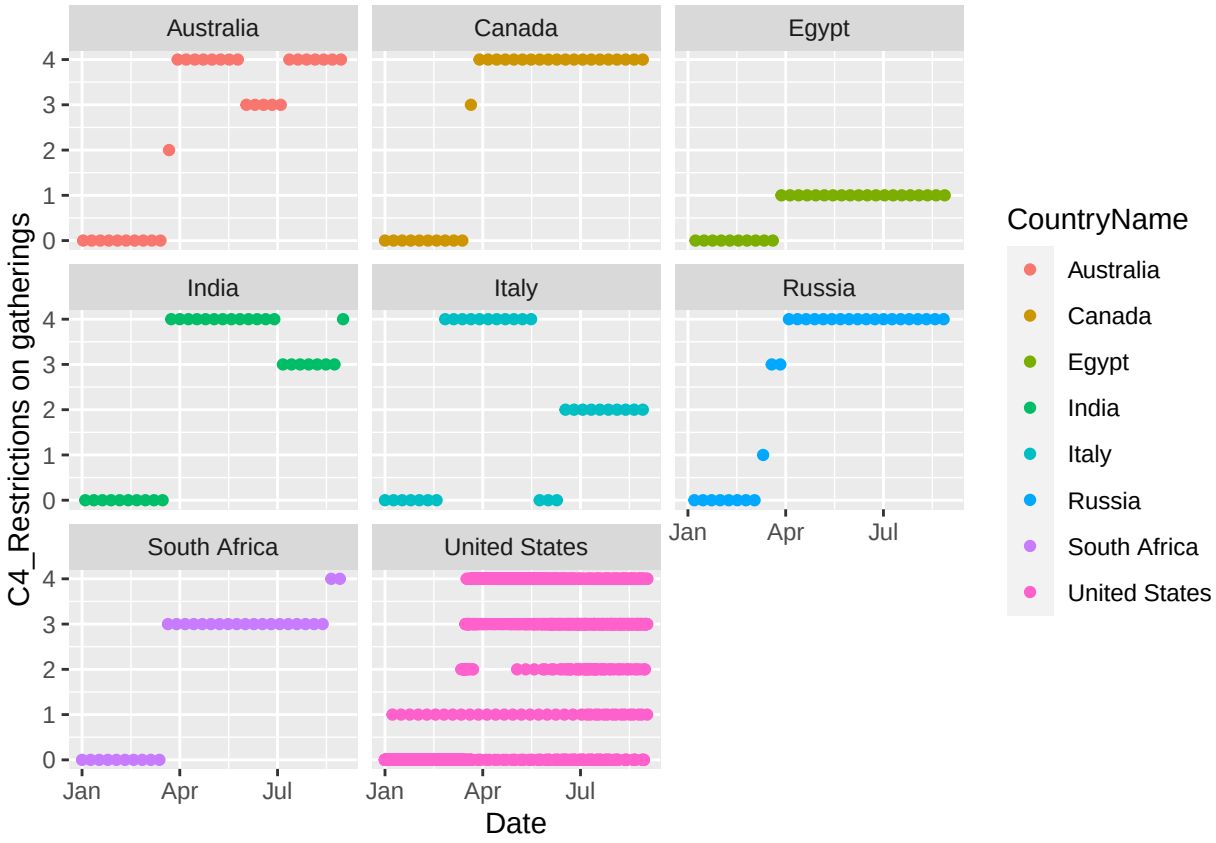
```
University of Oxford, Blavatnik School of Government, "Coronavirus government response tracker"
```

You need to use the codebook to understand the meanings of the values

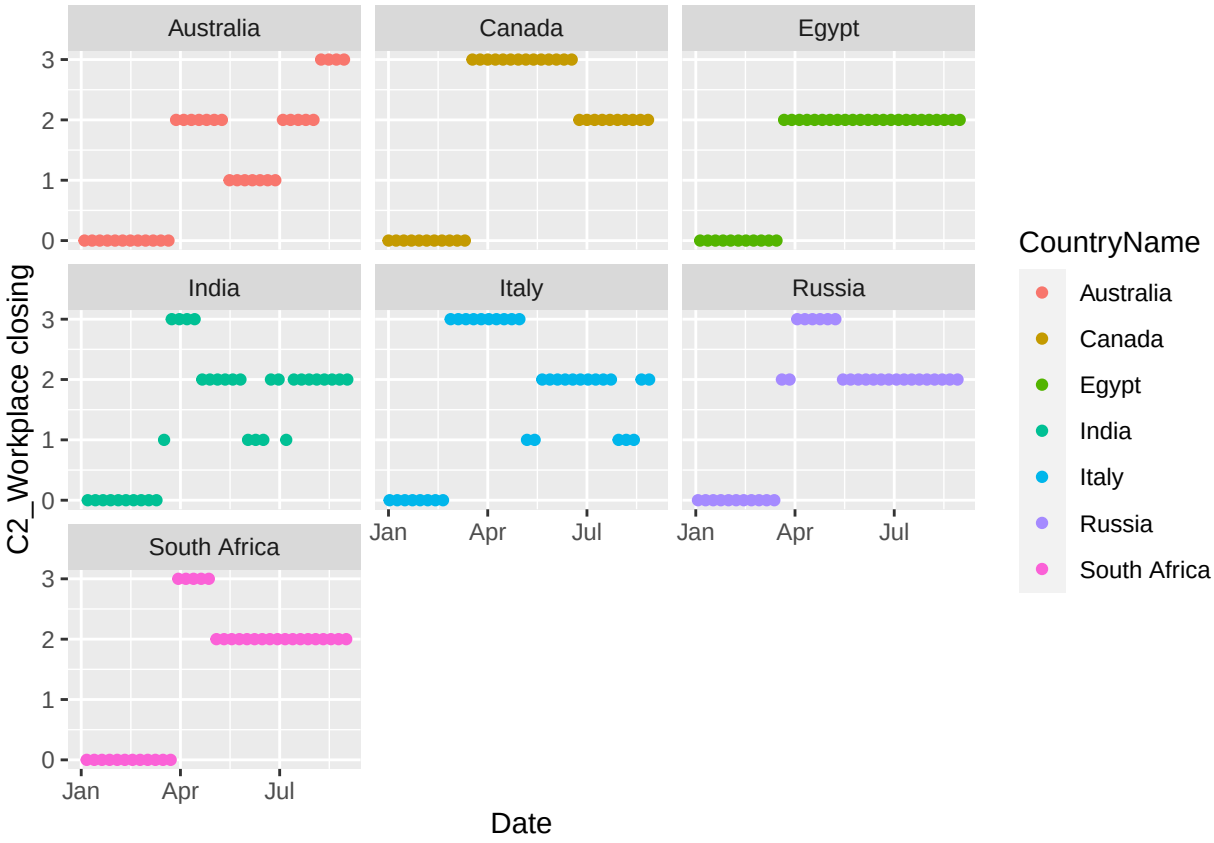
```
covid_response <- read_csv("https://raw.githubusercontent.com/OxCGRT/covid-policy-tracker/master/data/O
                           col_types = cols(
                             .default = col_character(),
                             Date = col_date(format = "%Y%m%d"),
                             `E3_Fiscal measures` = col_double(),
                             `E4_International support` = col_double(),
                             `H4_Emergency investment in healthcare` = col_double(),
                             `H5_Investment in vaccines` = col_double(),
                             `C1_School closing` = col_double(),
                             `C2_Workplace closing` = col_double(),
                             `C4_Restrictions on gatherings` = col_double(),
                             `C1_Flag` = col_logical(),
                             `C2_Flag` = col_logical(),
                             `C3_Flag` = col_logical(),
                             `C4_Flag` = col_logical(),
                             `C5_Flag` = col_logical(),
                             `C6_Flag` = col_logical(),
                             `C7_Flag` = col_logical(),
                             `E1_Flag` = col_logical(),
                             `H1_Flag` = col_logical()
                           )) %>% filter(between(Date,as.Date("2020-01-01"),as.Date("2020-09-01")))

covid_response_gatherings_filtered <- covid_response %>%
  filter(CountryCode == c("CAN", "USA", "ITA","RUS", "AUS", "EGY", "ZAF", "IND"), `C1_School closing` >=
  select(`CountryName`:`Date`,`C4_Restrictions on gatherings`, -(`RegionName`:`RegionCode`))
```

```
## Warning in CountryCode == c("CAN", "USA", "ITA", "RUS", "AUS", "EGY", "ZAF", :
## longer object length is not a multiple of shorter object length
```

```
covid_response_workplace_restrictions <- covid_response %>%
  filter(CountryCode == c("CAN", "ITA","RUS", "AUS", "EGY", "ZAF", "IND"), `C2_Workplace closing`>=0)%>%
  select(`CountryName`:`Date`,`C2_Workplace closing`, -(`RegionName`:`RegionCode`))


covid_response_gatherings_filtered %>%
  ggplot(mapping = aes(x = Date, y = `C4_Restrictions on gatherings`, color = CountryName)) + geom_poin
```




```
covid_response_workplace_restrictions%>%
  ggplot(mapping = aes(x = Date, y = `C2_Workplace closing`, color = CountryName))+geom_point()+facet_wr
```




```
#view(covid_response)
```

Datasets contained in .csv files democracy index developed by the Economist Intelligence Unit, which is contained in the table from the Wikipedia page.

```
democracyindex <- read_tsv("DEMOCRACYINDEX.csv")
```

```
## Parsed with column specification:
## cols(
##   Rank = col_double(),
##   Country = col_character(),
##   Score = col_double(),
##   `Electoral process and pluralism` = col_double(),
##   `Functioning of government` = col_double(),
##   `Political participation` = col_double(),
##   `Political culture` = col_double(),
##   `Civil liberties` = col_double(),
##   `Regime type` = col_character(),
##   Region = col_character(),
##   `Change from last year: Score` = col_character(),
##   `Change from last year: rank` = col_character()
## )
```

```
democracyindex %>% head()
```

```
## # A tibble: 6 x 12
##    Rank Country Score `Electoral proc~  `Functioning o~ `Political part~
##   <dbl> <chr>   <dbl>            <dbl>            <dbl>            <dbl>
## 1     1 Norway   9.87               10             9.64               10
## 2     2 Iceland  9.58               10             9.29                8.89
## 3     3 Sweden   9.39                9.58            9.64                8.33
## 4     4 New Ze~  9.26               10             9.29                8.89
## 5     5 Finland  9.25               10             8.93                8.89
## 6     6 Ireland  9.24               10             7.86                8.33
## # ... with 6 more variables: `Political culture` <dbl>, `Civil
## #   liberties` <dbl>, `Regime type` <chr>, Region <chr>, `Change from last
## #   year: Score` <chr>, `Change from last year: rank` <chr>
```

World Regions Classification

```
regionclassification <- read_tsv("WorldRegions.csv")
```

```
## Parsed with column specification:
## cols(
##   Country = col_character(),
##   Region = col_character(),
##   `Global South` = col_character()
## )
```

```
regionclassification %>% head()
```

```
## # A tibble: 6 x 3
##   Country              Region              `Global South`
##   <chr>                <chr>               <chr>
## 1 Andorra              Europe              Global North
## 2 United Arab Emirates Arab States         Global South
## 3 Afghanistan          Asia & Pacific      Global South
## 4 Antigua and Barbuda  South/Latin America Global South
## 5 Anguilla             South/Latin America Global South
## 6 Albania              Europe              Global North
```

World happiness report 2020, happiness score

```
happinessscore <- read_tsv("WorldHappinessReport2020-Score.csv")
```

```
## Parsed with column specification:
## cols(
##   Country = col_character(),
##   `Regional indicator` = col_character(),
##   `Ladder score` = col_double()
## )
```

```
happinessscore
```

```
## # A tibble: 153 x 3
##    Country     `Regional indicator`  `Ladder score`
##    <chr>       <chr>                          <dbl>
##  1 Finland     Western Europe                  7.81
##  2 Denmark     Western Europe                  7.65
##  3 Switzerland Western Europe                  7.56
##  4 Iceland     Western Europe                  7.50
##  5 Norway      Western Europe                  7.49
##  6 Netherlands Western Europe                  7.45
##  7 Sweden      Western Europe                  7.35
##  8 New Zealand North America and ANZ           7.30
##  9 Austria     Western Europe                  7.29
## 10 Luxembourg  Western Europe                  7.24
## # ... with 143 more rows
```

Area of the regions

```
area <- read_tsv("AREA.csv")
```

```
## Parsed with column specification:
## cols(
##   Rank = col_double(),
##   Country = col_character(),
##   `(sq km)` = col_double()
## )
```

```
area
```

```
## # A tibble: 258 x 3
##     Rank Country       `(sq km)`
##    <dbl> <chr>             <dbl>
##  1     1 Russia         17098242
##  2     2 Antarctica     14200000
##  3     3 Canada          9984670
##  4     4 United States   9833517
##  5     5 China           9596960
##  6     6 Brazil          8515770
##  7     7 Australia       7741220
##  8     8 India           3287263
##  9     9 Argentina       2780400
## 10    10 Kazakhstan      2724900
## # ... with 248 more rows
```

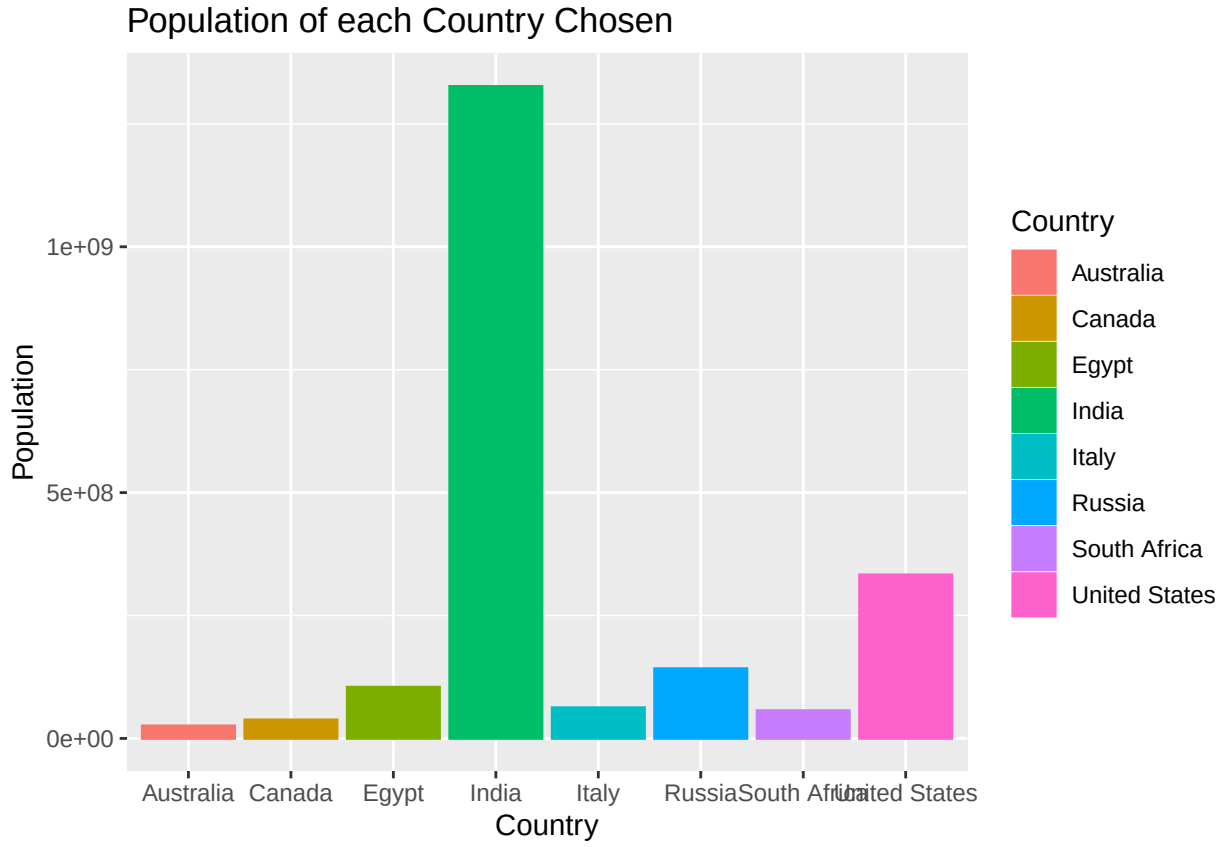
Population in the region

```
population <- read_tsv("POPULATION.csv")
```

```
## Parsed with column specification:
## cols(
```

```
##   Rank = col_double(),
##   Country = col_character(),
##   Population = col_double(),
##   `Date of Information` = col_character()
## )
```

```
population%>%
  filter(Country %in% c("Canada", "United States", "Italy", "Russia", "Australia", "Egypt","South Africa
  select(Country, Population)%>%
  ggplot() + geom_bar(mapping = aes(x = Country, y = Population, fill = Country, color = Country), stat
```

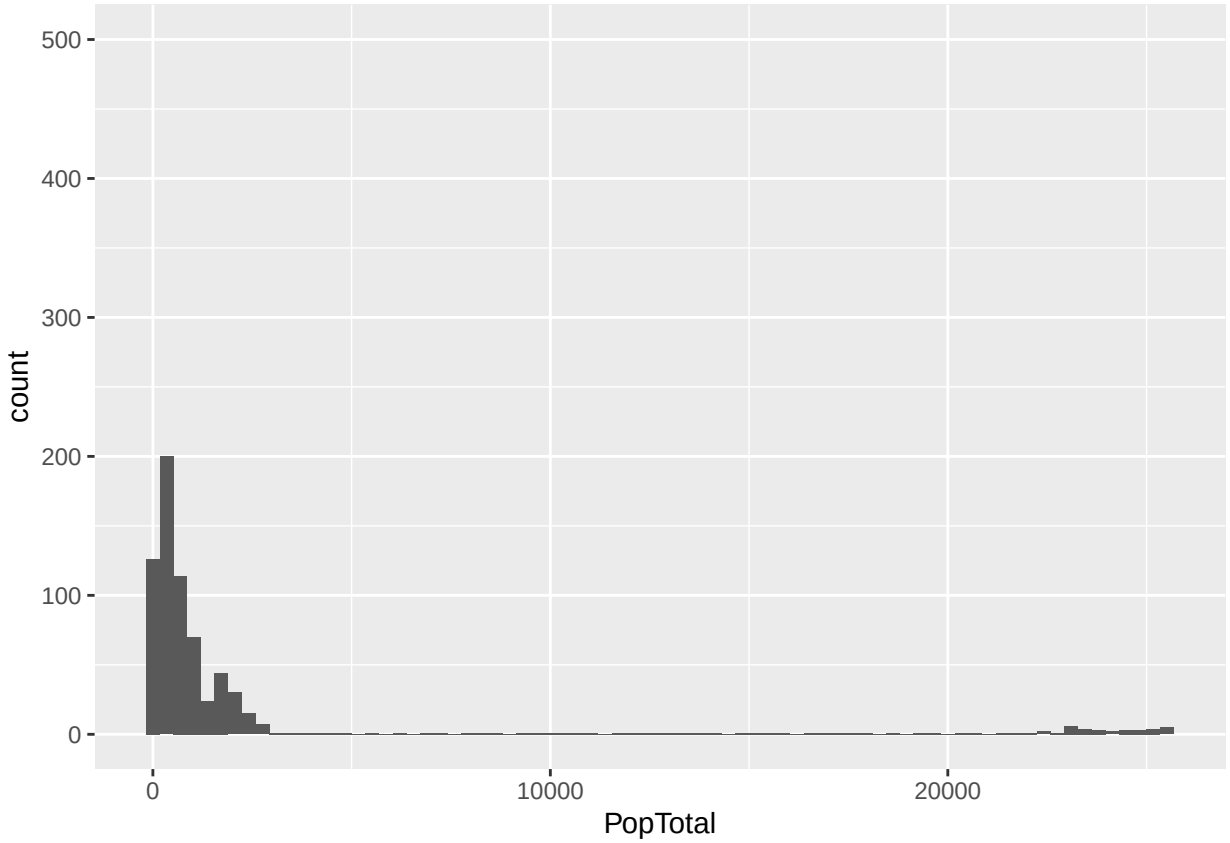


Population distribution The numeric values for the Pop columns are population in thousands.

```
populationdistribution <- read_tsv("POPULATIONDISTRIBUTION.csv")
```

```
## Parsed with column specification:
## cols(
##   Location = col_character(),
##   Time = col_double(),
##   Age = col_double(),
##   PopMale = col_double(),
##   PopFemale = col_double(),
##   PopTotal = col_double()
## )
```

```
populationdistribution%>%
  filter(Location %in% c("Canada", "United States", "Italy", "Russia", "Australia", "Egypt","South Afri
  arrange(Location)%>%
  ggplot() + geom_histogram(mapping = aes(x = PopTotal), bins = 75) + ylim(0,500)
```



Life expectance at birth in the region

```
lifeexpect <- read_tsv("LIFEEXPECTANCYATBIRTH.csv")%>%
  spread(key, value)%>%
  filter(!is.na(female))%>%
  separate(female, into=c("female", "delete1", "delete2"), sep = " ",convert=TRUE)%>%
  separate(male, into=c("male", "delete"), sep = " ",convert=TRUE)%>%
  separate(`total population`, into=c("total population", "delete3"), sep = " ",convert=TRUE) %>%
  select(-delete1,-delete2,-delete3, -delete)
```

```
## Parsed with column specification:
## cols(
##   key = col_character(),
##   value = col_character(),
##   number = col_double()
## )
```

```
## Warning: Expected 3 pieces. Additional pieces discarded in 230 rows [1, 2, 3, 4,
## 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```
## Warning: Expected 3 pieces. Missing pieces filled with `NA` in 1 rows [69].

## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 1 rows [69].

## Warning: Expected 2 pieces. Additional pieces discarded in 1 rows [69].
```

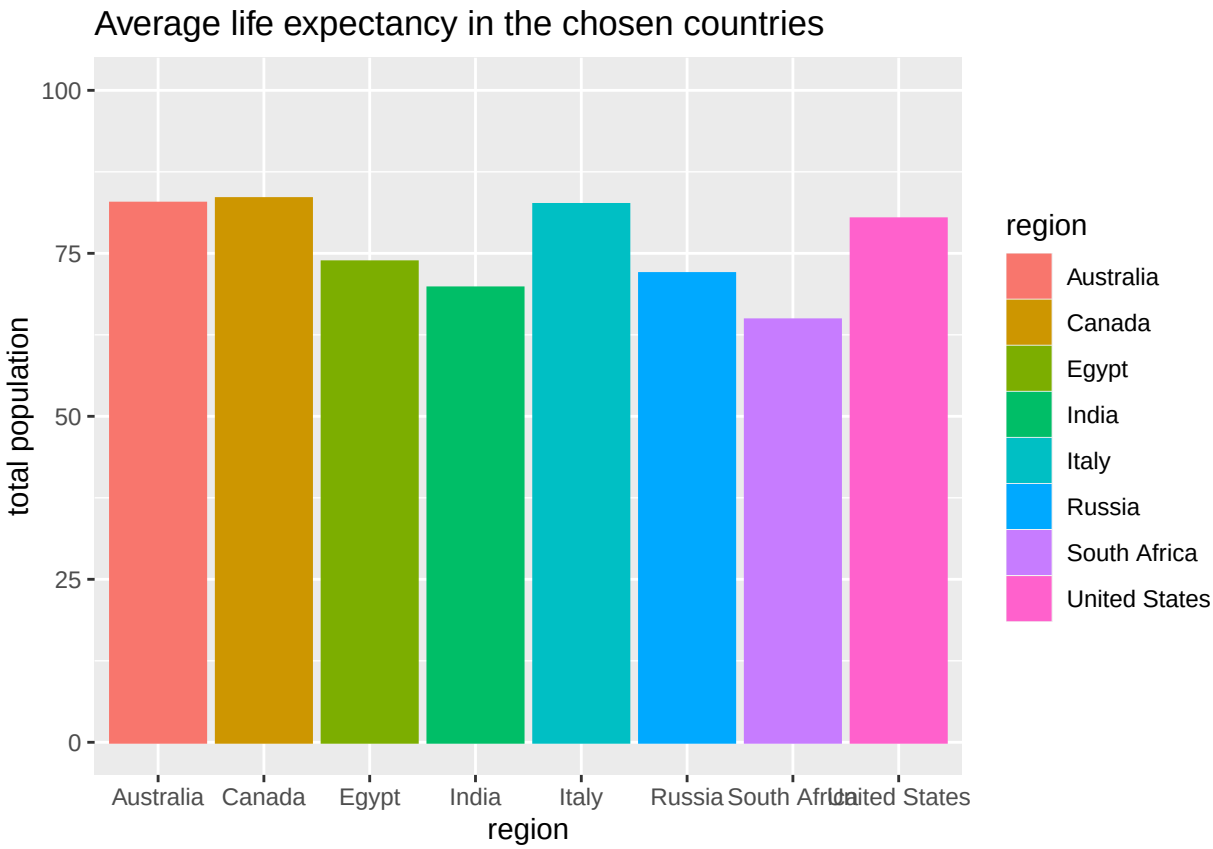
```
lifeexpect
```

```
## # A tibble: 231 x 5
##    number female  male region              `total population`
##     <dbl>  <dbl> <dbl> <chr>                            <dbl>
##  1      1   54.4  51.4 Afghanistan                       52.8
##  2      2   81.9  76.3 Albania                           79
##  3      3   79.1  76.1 Algeria                           77.5
##  4      4   77.5  72.3 American Samoa                    74.8
##  5      5   85.4  80.8 Andorra                           83
##  6      6   63.4  59.3 Angola                            61.3
##  7      7   84.5  79.2 Anguilla                          81.8
##  8      8   79.6  75.1 Antigua and Barbuda               77.3
##  9      9   81.1  74.7 Argentina                         77.8
## 10     10   79.2  72.3 Armenia                           75.6
## # ... with 221 more rows
```

```
#Descriptive Stats
lifeexpect2<-lifeexpect%>%
  filter(region %in% c("Canada", "United States", "Italy", "Russia", "Australia", "Egypt","South Africa
  mutate(Average_Expectancy = sum(`total population`)/n())%>%
  mutate(Expectancy_Difference = `total population` - Average_Expectancy)

lifeexpect2
```

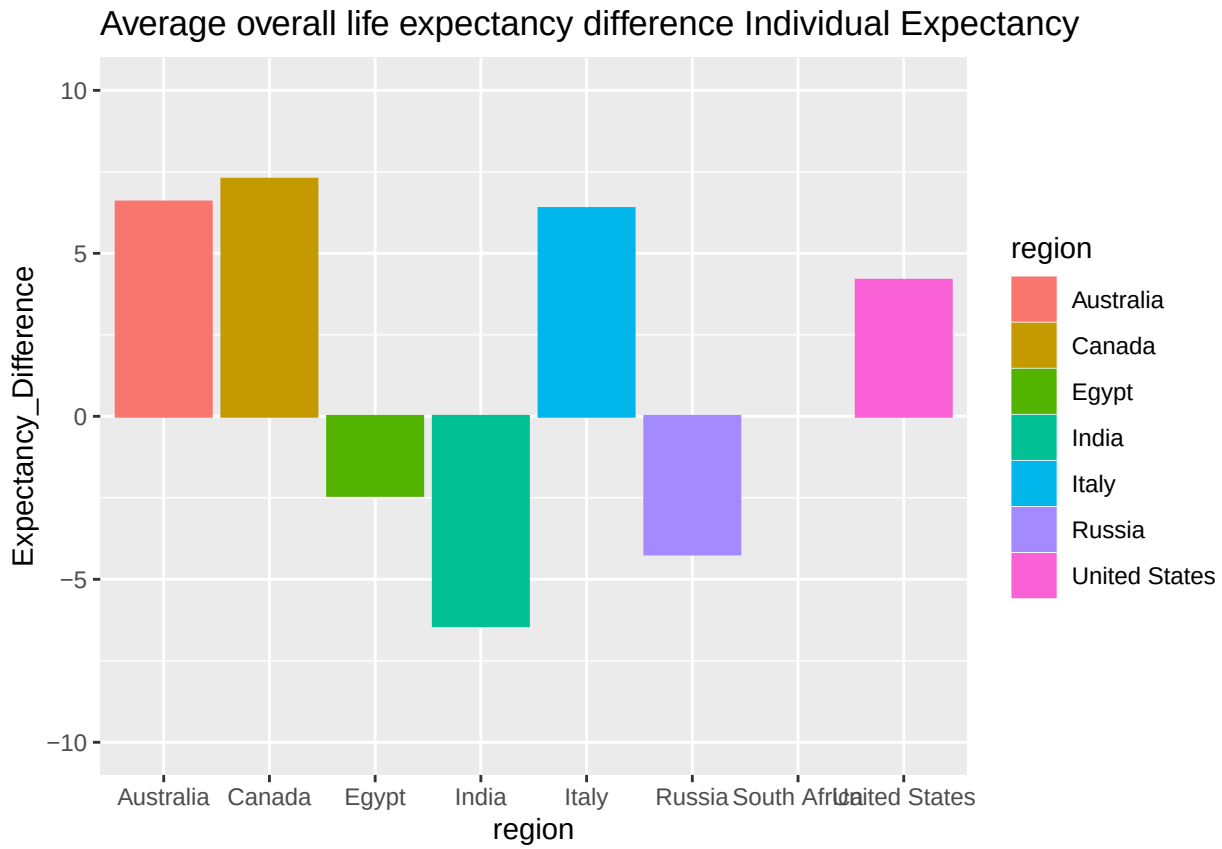
```
## # A tibble: 8 x 7
##   number female  male region  `total populati~ Average_Expecta~ Expectancy_Diff~
##    <dbl>  <dbl> <dbl> <chr>              <dbl>            <dbl>            <dbl>
## 1     12   85    80.5 Austra~             82.7             76.1             6.58
## 2     38   85.9  81.1 Canada              83.4             76.1             7.28
## 3     63   75.3  72.3 Egypt               73.7             76.1            -2.42
## 4     98   71.2  68.4 India               69.7             76.1            -6.42
## 5    105   85.3  79.8 Italy               82.5             76.1             6.38
## 6    174   77.8  66.3 Russia              71.9             76.1            -4.22
## 7    197   66.2  63.4 South ~             64.8             76.1           -11.3
## 8    225   82.5  78   United~             80.3             76.1             4.17
```

```
lifeexpect%>%
  filter(region %in% c("Canada", "United States", "Italy", "Russia", "Australia", "Egypt","South Africa
  ggplot() + geom_bar(mapping = aes(x = region, y = `total population`, fill = region, color = region),
```




```
lifeexpect2%>%
  filter(region %in% c("Canada", "United States", "Italy", "Russia", "Australia", "Egypt","South Africa
  ggplot() + geom_bar(mapping = aes(x = region, y = Expectancy_Difference, fill = region, color = regio
```

```
## Warning: Removed 1 rows containing missing values (position_stack).
```

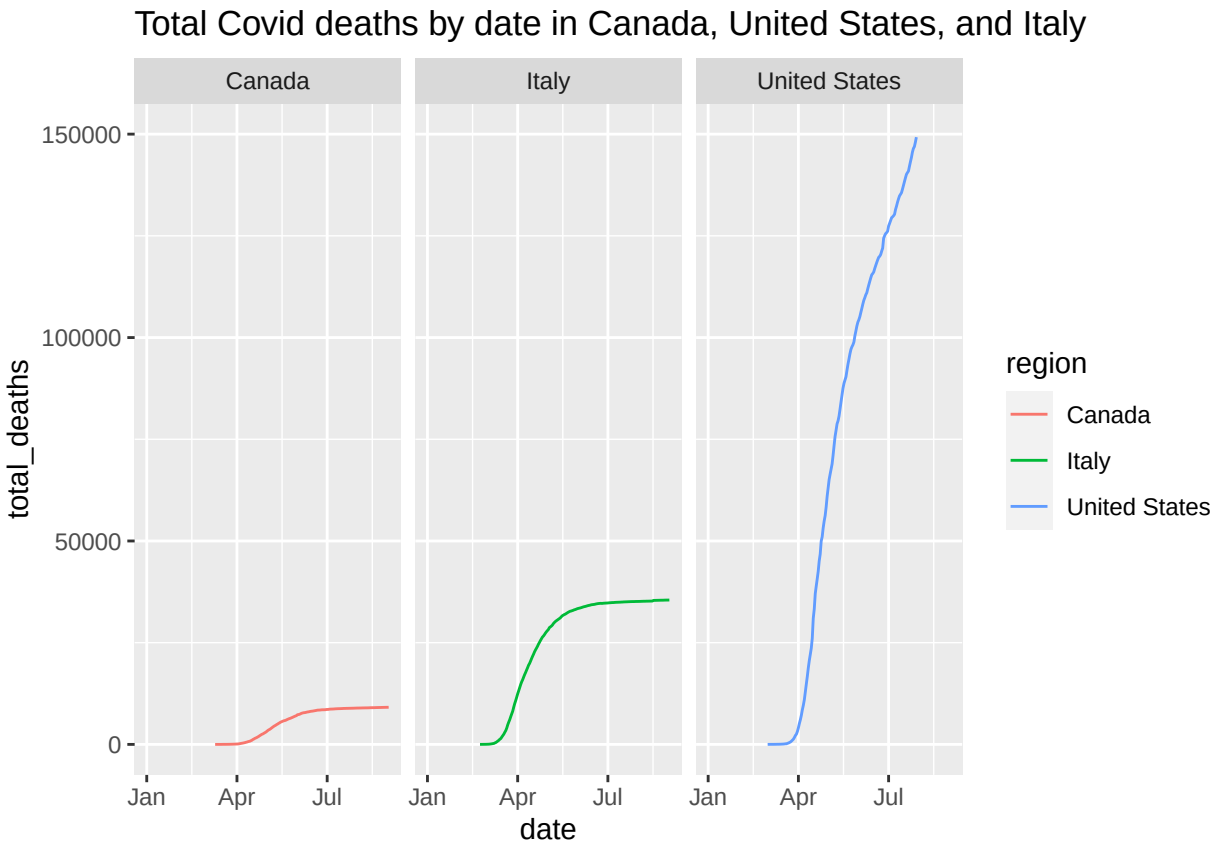



```
deaths_vs_lifeexpect<-full_join(lifeexpect, covid_cases, by = c("region" = "location"))
deaths_vs_lifeexpect
```

```
## # A tibble: 40,500 x 14
##    number female  male region `total populati~ date       new_cases new_deaths
##     <dbl>  <dbl> <dbl> <chr>             <dbl> <date>         <dbl>      <dbl>
##  1      1   54.4  51.4 Afgha~             52.8 2020-01-01         0          0
##  2      1   54.4  51.4 Afgha~             52.8 2020-01-02         0          0
##  3      1   54.4  51.4 Afgha~             52.8 2020-01-03         0          0
##  4      1   54.4  51.4 Afgha~             52.8 2020-01-04         0          0
##  5      1   54.4  51.4 Afgha~             52.8 2020-01-05         0          0
##  6      1   54.4  51.4 Afgha~             52.8 2020-01-06         0          0
##  7      1   54.4  51.4 Afgha~             52.8 2020-01-07         0          0
##  8      1   54.4  51.4 Afgha~             52.8 2020-01-08         0          0
##  9      1   54.4  51.4 Afgha~             52.8 2020-01-09         0          0
## 10      1   54.4  51.4 Afgha~             52.8 2020-01-10         0          0
## # ... with 40,490 more rows, and 6 more variables: total_cases <dbl>,
## #   total_deaths <dbl>, weekly_cases <dbl>, weekly_deaths <dbl>,
## #   biweekly_cases <dbl>, biweekly_deaths <dbl>
```

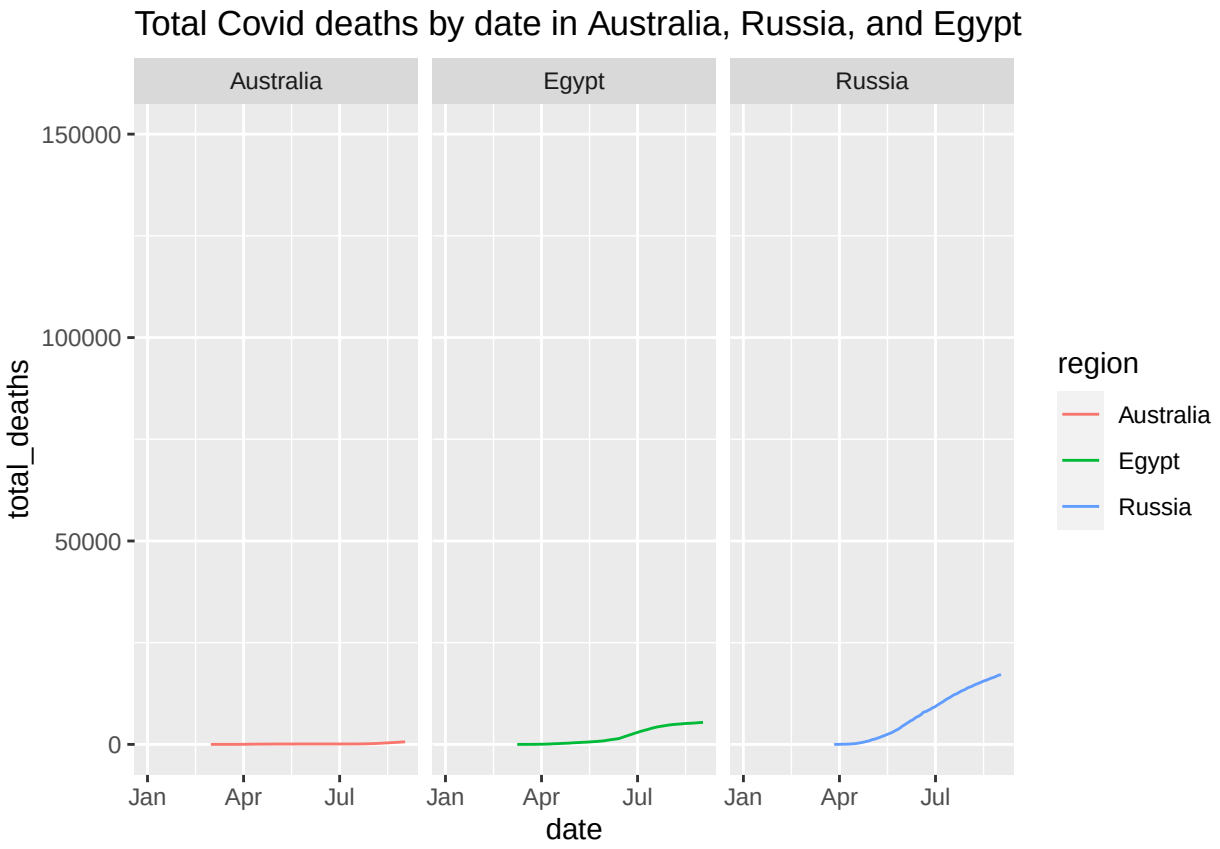
```
deaths_vs_lifeexpect%>%
  filter(region %in% c("Canada", "United States", "Italy"))%>%
  ggplot()+geom_line(mapping = aes(x = date, y = total_deaths, color = region)) + labs(title = "Total C
```

```
## Warning: Removed 216 row(s) containing missing values (geom_path).
```



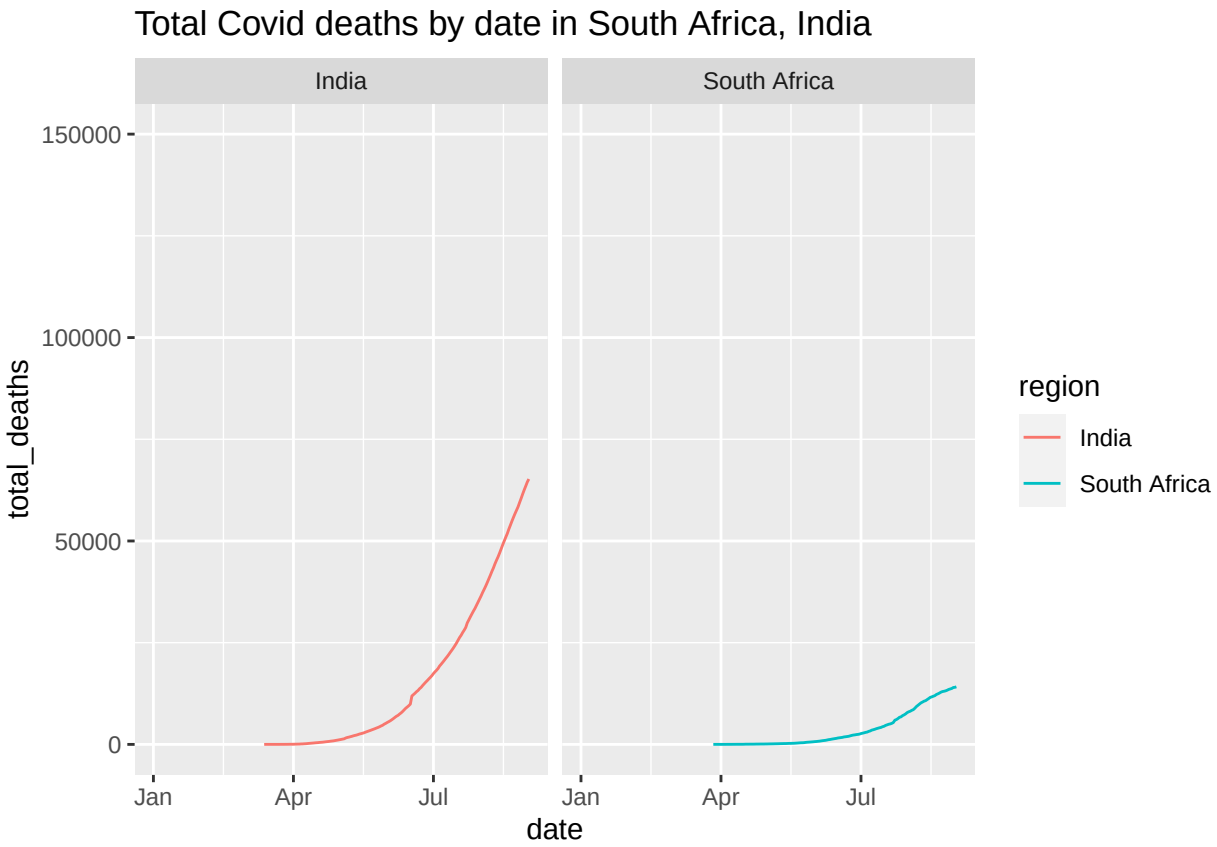

```
deaths_vs_lifeexpect%>%
  filter(region %in% c("Australia", "Russia", "Egypt"))%>%
  ggplot()+geom_line(mapping = aes(x = date, y = total_deaths, color = region)) + labs(title = "Total C
```

```
## Warning: Removed 214 row(s) containing missing values (geom_path).
```




```
deaths_vs_lifeexpect%>%
  filter(region %in% c("South Africa", "India"))%>%
  ggplot() + geom_line(mapping = aes(x = date, y = total_deaths, color = region)) + labs(title = "Total
```

```
## Warning: Removed 91 row(s) containing missing values (geom_path).
```



Birthrate in the regions

```
birthrate <- read_tsv("BIRTHRATE.csv")
```

```
## Parsed with column specification:
## cols(
##   Rank = col_double(),
##   Country = col_character(),
##   `(births/1,000 population)` = col_double(),
##   `Date of Information` = col_character()
## )
```

```
birthrate%>%
  filter(Country %in% c("Canada", "United States", "Italy", "Russia", "Australia", "Egypt","South Africa
```

```
## # A tibble: 8 x 4
##    Rank Country       `(births/1,000 population)` `Date of Information`
##   <dbl> <chr>                               <dbl> <chr>
## 1    42 Egypt                                27.2 2020 est.
## 2    78 South Africa                         19.2 2020 est.
## 3    87 India                                18.2 2020 est.
## 4   156 Australia                            12.4 2020 est.
## 5   157 United States                        12.4 2020 est.
## 6   190 Canada                               10.2 2020 est.
## 7   191 Russia                               10   2020 est.
## 8   218 Italy                                 8.4 2020 est.
```

Deathrate in the region

```
deathrate <- read_tsv("DEATHRATE.csv")
```

```
## Parsed with column specification:
## cols(
##   Rank = col_double(),
##   Country = col_character(),
##   `(deaths/1,000 population)` = col_double(),
##   `Date of Information` = col_character()
## )
```

```
deathrate%>%
  arrange((Rank))
```

```
## # A tibble: 229 x 4
##     Rank Country   `(deaths/1,000 population)` `Date of Information`
##    <dbl> <chr>                           <dbl> <chr>
##  1     1 Lesotho                          15.4 2020 est.
##  2     2 Lithuania                        15   2020 est.
##  3     3 Bulgaria                         14.6 2020 est.
##  4     4 Latvia                           14.6 2020 est.
##  5     5 Ukraine                          14   2020 est.
##  6     6 Serbia                           13.5 2020 est.
##  7     7 Russia                           13.4 2020 est.
##  8     8 Belarus                          13.1 2020 est.
##  9     9 Estonia                          12.9 2020 est.
## 10    10 Hungary                          12.9 2020 est.
## # ... with 219 more rows
```

Labor force data in the region: the up-to-date data we use. CIA has a slightly outdated data set.

```
laborforce <- read_tsv("LABORFORCE.csv",
                       col_types = cols(
                         .default = col_double(),
                         `Country Name` = col_character(),
                         `Country Code` = col_character()
                       )) %>%
  gather(`1960`:`2020`, key = "years", value = "population")%>%
  filter(years >= 1990)
laborforce
```

```
## # A tibble: 8,184 x 4
##    `Country Name`       `Country Code` years population
##    <chr>                <chr>          <chr>      <dbl>
##  1 Aruba                ABW            1990          NA
##  2 Afghanistan          AFG            1990     3049464
##  3 Angola               AGO            1990     4844454
##  4 Albania              ALB            1990     1404177
##  5 Andorra              AND            1990          NA
##  6 Arab World           ARB            1990    61300691
##  7 United Arab Emirates ARE            1990      917445
```

```
##  8 Argentina            ARG            1990    13580769
##  9 Armenia              ARM            1990     1422950
## 10 American Samoa       ASM            1990          NA
## # ... with 8,174 more rows
```

Unemployment in the region

```
unemployment <- read_tsv("UNEMP.csv")
```

```
## Parsed with column specification:
## cols(
##   Rank = col_double(),
##   Country = col_character(),
##   `(%)` = col_double(),
##   `Date of Information` = col_character()
## )
```

```
unemployment %>%
  filter(Country %in% c("Canada", "United States", "Italy", "Russia", "Australia", "Egypt","South Africa
```

```
## # A tibble: 8 x 4
##    Rank Country       `(%)` `Date of Information`
##   <dbl> <chr>         <dbl> <chr>
## 1    62 United States   4.4 2017 est.
## 2    77 Russia          5.2 2017 est.
## 3    81 Australia       5.6 2017 est.
## 4    94 Canada          6.3 2017 est.
## 5   122 India           8.5 2017 est.
## 6   153 Italy          11.3 2017 est.
## 7   161 Egypt          12.2 2017 est.
## 8   200 South Africa   27.5 2017 est.
```

Unemployment of youth in the region, ages 15-24

```
unemp_youth <- read_tsv("UNEMPYOUTH.csv")
```

```
## Parsed with column specification:
## cols(
##   Rank = col_double(),
##   Country = col_character(),
##   `(%)` = col_double()
## )
```

```
unemp_youth
```

```
## # A tibble: 181 x 3
##     Rank Country               `(%)`
##    <dbl> <chr>                 <dbl>
##  1     1 French Polynesia       56.7
##  2     2 South Africa           53.4
##  3     3 Kosovo                 52.4
```

```
##  4     4 Libya                   48.7
##  5     5 Eswatini                47.1
##  6     6 Gaza Strip              46.9
##  7     7 West Bank               46.9
##  8     8 Macedonia               46.7
##  9     9 Saint Lucia             46.2
## 10    10 Bosnia and Herzegovina  45.8
## # ... with 171 more rows
```

Degree of urbanization in the region

```
urbanization <- read_tsv("URBANIZATION.csv")
```

```
## Parsed with column specification:
## cols(
##   Region = col_character(),
##   Afghanistan = col_character()
## )
```

```
urbanization %>% head()
```

```
## # A tibble: 6 x 2
##   Region                      Afghanistan
##   <chr>                       <chr>
## 1 urban population            26% of total population
## 2 Data Year                   2020
## 3 annual rate of urbanization 3.37%
## 4 Rate est. period            2015-20
## 5 Region                      Albania
## 6 urban population            62.1% of total population
```

School (primary to tertiary education) life expectancy in the region

```
schooling <- read_tsv("SCHOOLINGEXPECTANCY.csv")
```

```
## Parsed with column specification:
## cols(
##   key = col_character(),
##   value = col_character(),
##   number = col_double()
## )
```

```
schooling %>% head()
```

```
## # A tibble: 6 x 3
##   key       value       number
##   <chr>     <chr>        <dbl>
## 1 Country   Afghanistan      1
## 2 total     10 years         1
## 3 male      13 years         1
## 4 female    8 years          1
## 5 data year 2018             1
## 6 Country   Albania          2
```

Health expenditure in the region

```
healthexp <- read_tsv("HEALTHEXP.csv")
```

```
## Parsed with column specification:
## cols(
##   Country = col_character(),
##   `Current Health Expenditure` = col_character(),
##   Year = col_double()
## )
```

```
healthexp
```

```
## # A tibble: 190 x 3
##    Country             `Current Health Expenditure`  Year
##    <chr>               <chr>                        <dbl>
##  1 Afghanistan         10.3%                         2015
##  2 Albania             6.8%                          2015
##  3 Algeria             7.1%                          2015
##  4 Andorra             12%                           2015
##  5 Angola              2.9%                          2015
##  6 Antigua and Barbuda 4.8%                          2015
##  7 Argentina           6.8%                          2015
##  8 Armenia             10.1%                         2015
##  9 Australia           9.4%                          2015
## 10 Austria             10.3%                         2015
## # ... with 180 more rows
```

Education expenditure in the region

```
educationexp <- read_tsv("EDUEXP.csv")
```

```
## Parsed with column specification:
## cols(
##   Rank = col_double(),
##   Country = col_character(),
##   `(% OF GDP)` = col_double(),
##   `Date of Information` = col_double()
## )
```

```
educationexp %>% head()
```

```
## # A tibble: 6 x 4
##    Rank Country               `(% OF GDP)` `Date of Information`
##   <dbl> <chr>                        <dbl>                 <dbl>
## 1     1 Lesotho                       13                    2008
## 2     2 Cuba                          12.8                  2010
## 3     3 Marshall Islands              12.2                  2003
## 4     4 Kiribati                      12                    2001
## 5     5 Botswana                       9.5                  2009
## 6     6 Sao Tome and Principe          9.5                  2010
```

GDP per capital in the region

```
gdppp <- read_tsv("GDPPP.csv")%>%
  mutate(`GDP - PER CAPITA ($)(PPP)` = parse_number(`GDP - PER CAPITA (PPP)`))%>%
  select(-`GDP - PER CAPITA (PPP)`)
```

```
## Parsed with column specification:
## cols(
##   Rank = col_double(),
##   Country = col_character(),
##   `GDP - PER CAPITA (PPP)` = col_character(),
##   `Date of Information` = col_character()
## )
```

```
gdppp
```

```
## # A tibble: 229 x 4
##     Rank Country       `Date of Information` `GDP - PER CAPITA ($)(PPP)`
##    <dbl> <chr>         <chr>                                       <dbl>
##  1     1 Liechtenstein 2009 est.                                  139100
##  2     2 Qatar         2017 est.                                  124500
##  3     3 Monaco        2015 est.                                  115700
##  4     4 Macau         2017 est.                                  111600
##  5     5 Luxembourg    2017 est.                                  106300
##  6     6 Bermuda       2016 est.                                   99400
##  7     7 Singapore     2017 est.                                   93900
##  8     8 Isle of Man   2014 est.                                   84600
##  9     9 Brunei        2017 est.                                   78200
## 10    10 Ireland       2017 est.                                   75500
## # ... with 219 more rows
```

Public debt in the region

```
publicdebt <- read_tsv("PUBLICDEBT.csv")
```

```
## Parsed with column specification:
## cols(
##   Rank = col_double(),
##   Country = col_character(),
##   `(% of GDP)` = col_double(),
##   `Date of Information` = col_character()
## )
```

```
publicdebt %>% head()
```

```
## # A tibble: 6 x 4
##    Rank Country  `(% of GDP)` `Date of Information`
##   <dbl> <chr>           <dbl> <chr>
## 1     1 Japan            238. 2017 est.
## 2     2 Greece           182. 2017 est.
## 3     3 Barbados         157. 2017 est.
## 4     4 Lebanon          147. 2017 est.
## 5     5 Italy            132. 2017 est.
## 6     6 Eritrea          131. 2017 est.
```

GDP composition by sector of origin in the region

```
gdpcomp <- read_tsv("GDPCOMPOSITION.csv")
```

```
## Parsed with column specification:
## cols(
##   Country = col_character(),
##   algriculture = col_character(),
##   industry = col_character(),
##   services = col_character(),
##   year = col_character(),
##   notes = col_character()
## )
```

```
gdpcomp
```

```
## # A tibble: 231 x 6
##    Country          algriculture industry services year    notes
##    <chr>            <chr>        <chr>    <chr>    <chr>   <chr>
##  1 Afghanistan      23%          21.1%    55.9%    2016 e~ data exclude opium p~
##  2 Albania          21.7%        24.2%    54.1%    2017 e~ <NA>
##  3 Algeria          13.3%        39.3%    47.4%    2017 e~ <NA>
##  4 American Samoa   27.4%        12.4%    60.2%    2012    <NA>
##  5 Andorra          11.9%        33.6%    54.5%    2015 e~ <NA>
##  6 Angola           10.2%        61.4%    28.4%    2011 e~ <NA>
##  7 Anguilla         3%           10.5%    86.4%    2017 e~ <NA>
##  8 Antigua and Bar~ 1.8%         20.8%    77.3%    2017 e~ <NA>
##  9 Argentina        10.8%        28.1%    61.1%    2017 e~ <NA>
## 10 Armenia          16.7%        28.2%    54.8%    2017 e~ <NA>
## # ... with 221 more rows
```

GINI index in the region

```
gini <- read_tsv("GINI.csv")
```

```
## Parsed with column specification:
## cols(
##   Rank = col_double(),
##   Country = col_character(),
##   `Distribution of family income - Gini index` = col_double(),
##   `Date of Information` = col_character()
## )
```

```
gini %>% head()
```

```
## # A tibble: 6 x 4
##    Rank Country               `Distribution of family income~ `Date of Informat~
##   <dbl> <chr>                                           <dbl> <chr>
## 1     1 Lesotho                                          63.2 1995
## 2     2 South Africa                                     62.5 2013 est.
## 3     3 Micronesia, Federate~                            61.1 2013 est.
```

```
## 4     4 Haiti                                                60.8 2012
## 5     5 Botswana                                             60.5 2009
## 6     6 Namibia                                              59.7 2010
```

Access to electricity (as percentage of population)

```
accesstoelectricity <- read_tsv("ACCESSTOELECTRICITY.csv")
```

```
## Parsed with column specification:
## cols(
##   `Country Name` = col_character(),
##   `Country Code` = col_character(),
##   `2015` = col_double(),
##   `2016` = col_double(),
##   `2017` = col_double(),
##   `2018` = col_double(),
##   `2019` = col_logical(),
##   `2020` = col_logical()
## )
```

```
accesstoelectricity %>% head()
```

```
## # A tibble: 6 x 8
##   `Country Name` `Country Code` `2015` `2016` `2017` `2018` `2019` `2020`
##   <chr>          <chr>           <dbl>  <dbl>  <dbl>  <dbl> <lgl>  <lgl>
## 1 Aruba          ABW             100    100    100    100   NA     NA
## 2 Afghanistan    AFG              71.5   97.7   97.7   98.7 NA     NA
## 3 Angola         AGO              42     40.7   42.0   43.3 NA     NA
## 4 Albania        ALB             100    100    100    100   NA     NA
## 5 Andorra        AND             100    100    100    100   NA     NA
## 6 Arab World     ARB              88.7   89.3   90.3   89.3 NA     NA
```

Individuals using the Internet (as percentage of population)

```
internetuser <- read_tsv("INTERNETUSER.csv")
```

```
## Parsed with column specification:
## cols(
##   Country = col_character(),
##   `Country Code` = col_character(),
##   `2015` = col_double(),
##   `2016` = col_double(),
##   `2017` = col_double(),
##   `2018` = col_double(),
##   `2019` = col_double(),
##   `2020` = col_logical()
## )
```

```
internetuser %>% head()
```

```
## # A tibble: 6 x 8
##   Country     `Country Code` `2015` `2016` `2017` `2018` `2019` `2020`
##   <chr>       <chr>          <dbl>  <dbl>  <dbl>  <dbl>  <dbl> <lgl>
## 1 Aruba       ABW             88.7   93.5   97.2   NA     NA   NA
## 2 Afghanistan AFG              8.26  NA     11.4   NA     NA   NA
## 3 Angola      AGO             12.4   13     14.3   NA     NA   NA
## 4 Albania     ALB             63.3   66.4   71.8   NA     69.6 NA
## 5 Andorra     AND             96.9   97.9   91.6   NA     NA   NA
## 6 Arab World  ARB             43.7   41.5   50.0   63.2   NA   NA
```

Use the above data for your project. You do not need to look for extra data sets for this project. You may look at other data sets for ideas and inspirations, but in the analysis and report, only use the data sets provided above.

========== Sanity check ==========

```
democracyindex
```

```
## # A tibble: 167 x 12
##     Rank Country Score `Electoral proc~  `Functioning o~ `Political part~
##    <dbl> <chr>   <dbl>            <dbl>            <dbl>            <dbl>
##  1     1 Norway   9.87            10                9.64            10
##  2     2 Iceland  9.58            10                9.29             8.89
##  3     3 Sweden   9.39             9.58             9.64             8.33
##  4     4 New Ze~  9.26            10                9.29             8.89
##  5     5 Finland  9.25            10                8.93             8.89
##  6     6 Ireland  9.24            10                7.86             8.33
##  7     7 Denmark  9.22            10                9.29             8.33
##  8     7 Canada   9.22             9.58             9.64             7.78
##  9     9 Austra~  9.09            10                8.93             7.78
## 10    10 Switze~  9.03             9.58             9.29             7.78
## # ... with 157 more rows, and 6 more variables: `Political culture` <dbl>,
## #   `Civil liberties` <dbl>, `Regime type` <chr>, Region <chr>, `Change from
## #   last year: Score` <chr>, `Change from last year: rank` <chr>
```

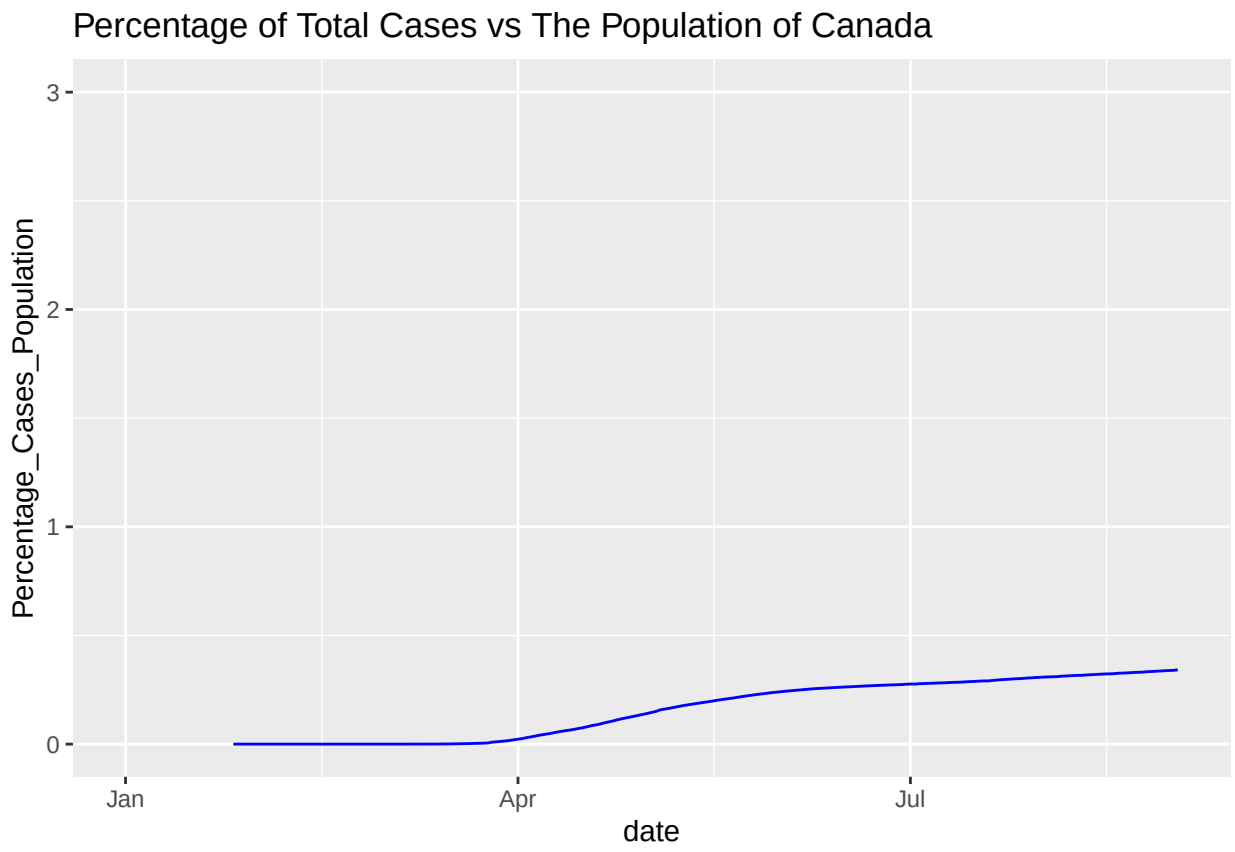
```
left_join(democracyindex, happinessscore, by = "Country")
```

```
## # A tibble: 167 x 14
##     Rank Country Score `Electoral proc~  `Functioning o~ `Political part~
##    <dbl> <chr>   <dbl>            <dbl>            <dbl>            <dbl>
##  1     1 Norway   9.87            10                9.64            10
##  2     2 Iceland  9.58            10                9.29             8.89
##  3     3 Sweden   9.39             9.58             9.64             8.33
##  4     4 New Ze~  9.26            10                9.29             8.89
##  5     5 Finland  9.25            10                8.93             8.89
##  6     6 Ireland  9.24            10                7.86             8.33
##  7     7 Denmark  9.22            10                9.29             8.33
##  8     7 Canada   9.22             9.58             9.64             7.78
##  9     9 Austra~  9.09            10                8.93             7.78
## 10    10 Switze~  9.03             9.58             9.29             7.78
## # ... with 157 more rows, and 8 more variables: `Political culture` <dbl>,
## #   `Civil liberties` <dbl>, `Regime type` <chr>, Region <chr>, `Change from
## #   last year: Score` <chr>, `Change from last year: rank` <chr>, `Regional
## #   indicator` <chr>, `Ladder score` <dbl>
```

```
#population dataset
population_vs_cases <- full_join(covid_cases, population, by = c("location" = "Country"))
population_vs_cases%>%
  filter(location == "Canada")%>%
  mutate(Percentage_Cases_Population = (100*total_cases)/(Population))%>%
  select(date, location, total_cases, total_deaths,Population, Percentage_Cases_Population)%>%
  ggplot(mapping = aes(x = date, y = `Percentage_Cases_Population`)) + geom_line(color = "blue", positi
  labs(title  = "Percentage of Total Cases vs The Population of Canada")
```

```
## Warning: Width not defined. Set with `position_dodge(width = ?)`
```

```
## Warning: Removed 25 row(s) containing missing values (geom_path).
```




```
population_vs_cases
```

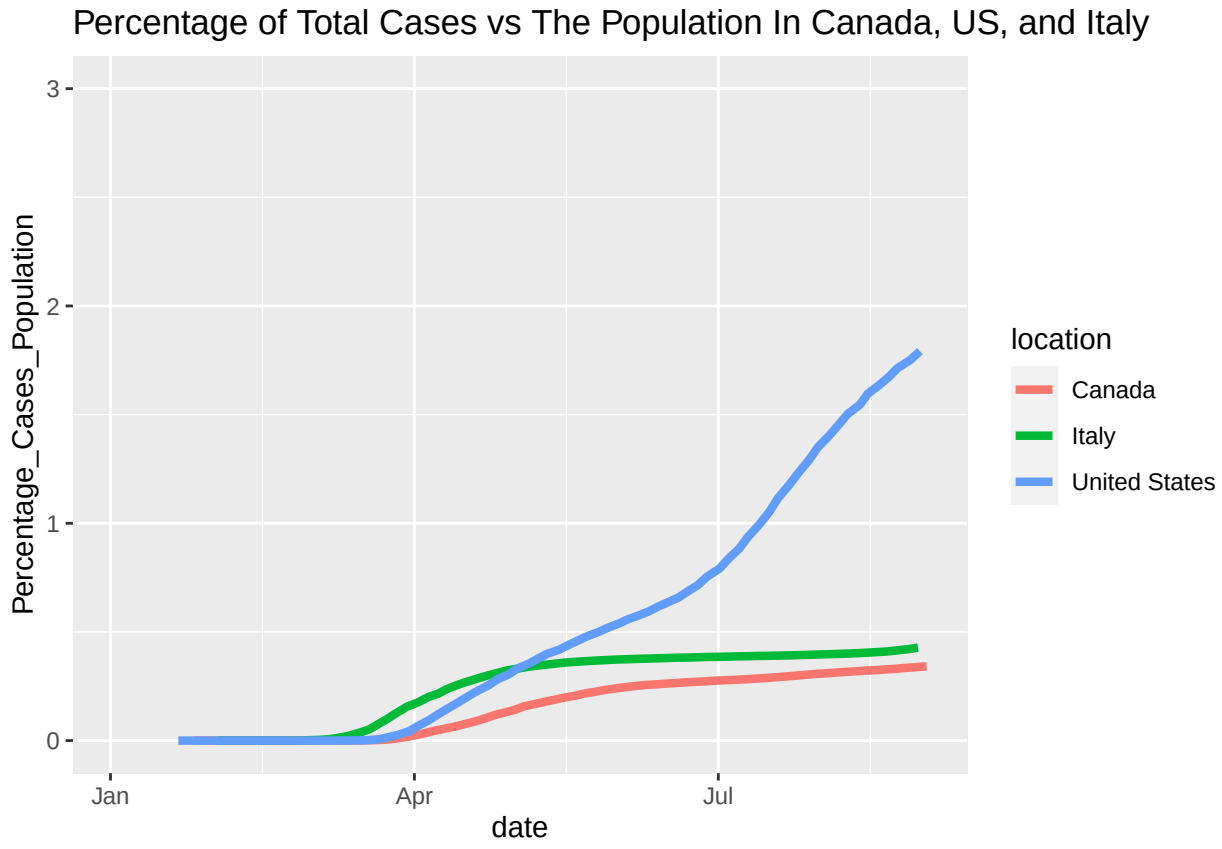
```
## # A tibble: 40,507 x 13
##    date       location new_cases new_deaths total_cases total_deaths
##    <date>     <chr>        <dbl>      <dbl>       <dbl>        <dbl>
##  1 2020-01-01 Afghani~         0          0          NA           NA
##  2 2020-01-02 Afghani~         0          0          NA           NA
##  3 2020-01-03 Afghani~         0          0          NA           NA
##  4 2020-01-04 Afghani~         0          0          NA           NA
##  5 2020-01-05 Afghani~         0          0          NA           NA
##  6 2020-01-06 Afghani~         0          0          NA           NA
```

```
##  7 2020-01-07 Afghani~         0          0          NA          NA
##  8 2020-01-08 Afghani~         0          0          NA          NA
##  9 2020-01-09 Afghani~         0          0          NA          NA
## 10 2020-01-10 Afghani~         0          0          NA          NA
## # ... with 40,497 more rows, and 7 more variables: weekly_cases <dbl>,
## #   weekly_deaths <dbl>, biweekly_cases <dbl>, biweekly_deaths <dbl>,
## #   Rank <dbl>, Population <dbl>, `Date of Information` <chr>
```

```
population_vs_cases%>%
  filter(location == c("United States", "Canada", "Italy"))%>%
  mutate(Percentage_Cases_Population = (100*total_cases)/(Population))%>%
  select(date, location, total_cases, total_deaths,Population, Percentage_Cases_Population)%>%
  ggplot() + geom_line(mapping = aes(x = date, y = `Percentage_Cases_Population`, color = location),size
```

```
## Warning in location == c("United States", "Canada", "Italy"): longer object
## length is not a multiple of shorter object length
```

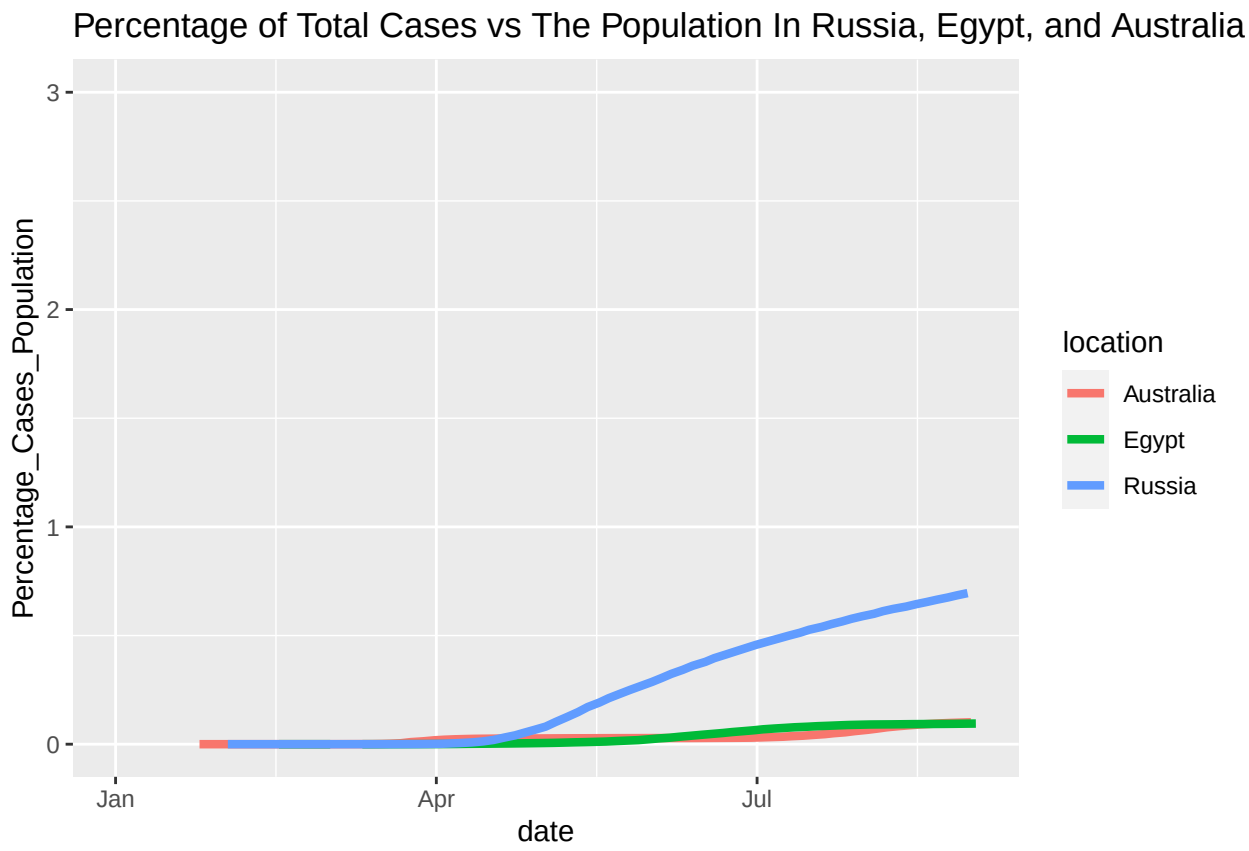
```
## Warning: Removed 24 row(s) containing missing values (geom_path).
```




```
population_vs_cases%>%
  filter(location == c("Russia", "Egypt", "Australia"))%>%
  mutate(Percentage_Cases_Population = (100*total_cases)/(Population))%>%
  select(date, location, total_cases, total_deaths,Population, Percentage_Cases_Population)%>%
  ggplot() + geom_line(mapping = aes(x = date, y = `Percentage_Cases_Population`, color = location),size
```

```
## Warning in location == c("Russia", "Egypt", "Australia"): longer object length
## is not a multiple of shorter object length
```

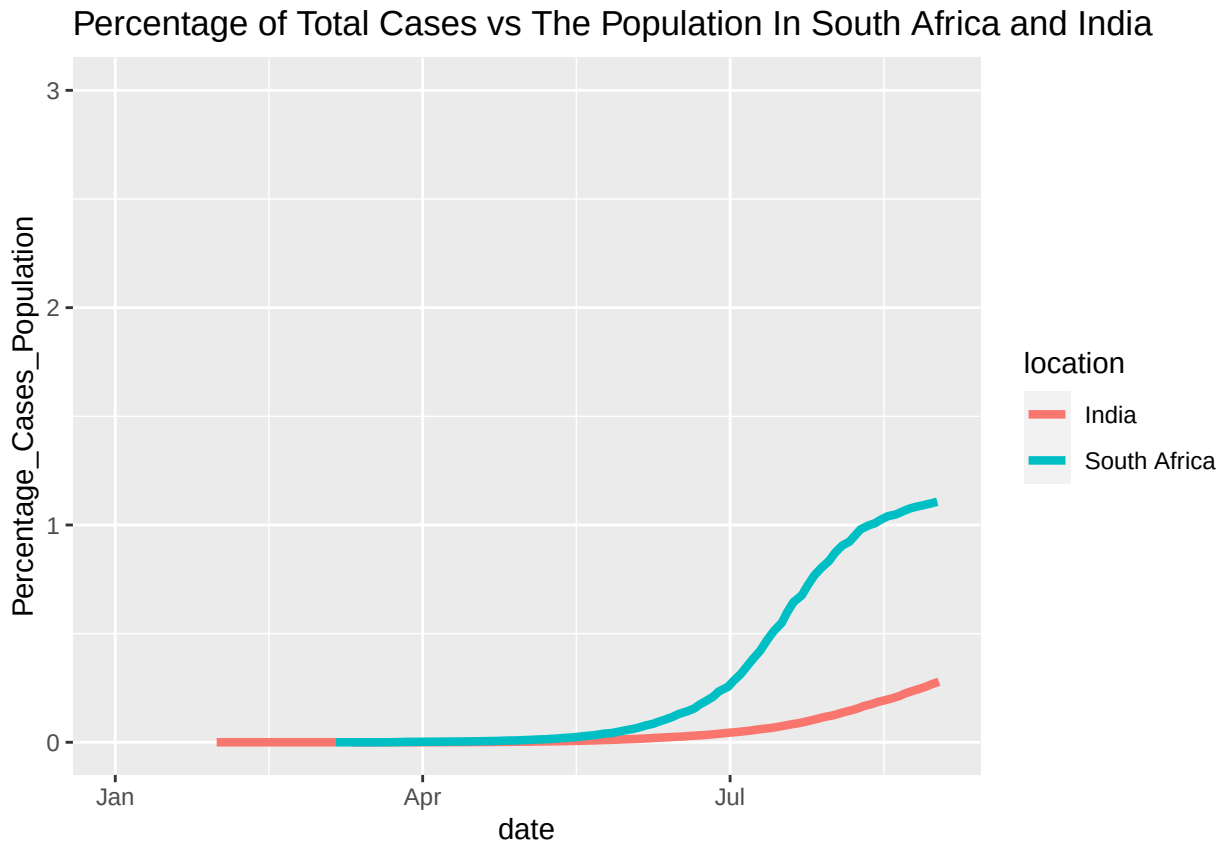
```
## Warning: Removed 33 row(s) containing missing values (geom_path).
```




```
population_vs_cases%>%
  filter(location == c("South Africa", "India"))%>%
  mutate(Percentage_Cases_Population = (100*total_cases)/(Population))%>%
  select(date, location, total_cases, total_deaths,Population, Percentage_Cases_Population)%>%
  ggplot() + geom_line(mapping = aes(x = date, y = `Percentage_Cases_Population`, color = location),size
```

```
## Warning in location == c("South Africa", "India"): longer object length is not a
## multiple of shorter object length
```

```
## Warning: Removed 15 row(s) containing missing values (geom_path).
```




```
population_vs_cases
```

```
## # A tibble: 40,507 x 13
##    date       location new_cases new_deaths total_cases total_deaths
##    <date>     <chr>        <dbl>      <dbl>       <dbl>        <dbl>
##  1 2020-01-01 Afghani~         0          0          NA           NA
##  2 2020-01-02 Afghani~         0          0          NA           NA
##  3 2020-01-03 Afghani~         0          0          NA           NA
##  4 2020-01-04 Afghani~         0          0          NA           NA
##  5 2020-01-05 Afghani~         0          0          NA           NA
##  6 2020-01-06 Afghani~         0          0          NA           NA
##  7 2020-01-07 Afghani~         0          0          NA           NA
##  8 2020-01-08 Afghani~         0          0          NA           NA
##  9 2020-01-09 Afghani~         0          0          NA           NA
## 10 2020-01-10 Afghani~         0          0          NA           NA
## # ... with 40,497 more rows, and 7 more variables: weekly_cases <dbl>,
## #   weekly_deaths <dbl>, biweekly_cases <dbl>, biweekly_deaths <dbl>,
## #   Rank <dbl>, Population <dbl>, 'Date of Information' <chr>
```