Capstone -Visualizing the rise of covid-19 in the world

Author: Andrew Chikunga (achikunga@iubh.de)

Date: 10 January 2020

ABSTRACT

With international spread of the coronavirus disease 2019 (COVID-19),this essay seeks to provide a direct visualization of the COVID-19 pandemic risk; infographics are provided in the figures. By showing visually confirmed cases, active cases, deaths and recovery cases of COVID-19 across the globe, visual graphs provide a relatively simple yet powerful way to explain the behavior of this infectious disease by displaying the most affected countries and least affected countries in graphs.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

List of figures

List of tables

GLOSSARY

COVID-19	An infectious disease caused by newly discovered coronavirus.
Plotly	Visualization tool available in Python.
Density line plot	Representation of the distribution of a numeric variable.
Tree map	Display of hierarchical data using nested rectangles.
Bar graph	Representation of categorical data with rectangular bars where heights are proportional to values, they represent.
Lollipops	Lollipop chart is an alternative to bar plots, when you have a large set of values to visualize.

SYMBOLS/ NOMENCLATURE

WHO	World Health Organization
R	Opensource programming language for statistical analysis
USA	United States of America

1.INTRODUCTION

1.1. Background

In December 2019, COVID-19 coronavirus was first identified in the Wuhan region of China. By March 11, 2020, the World Health Organization (WHO) categorized the COVID-19 outbreak as a pandemic. The pandemic has led to dramatic loss of life and unprecedented challenges to public health, food systems and workplaces worldwide. Severely affected countries include Iran, South Korea, Italy, France, United Kingdom, Brazil, USA and South Africa.

The coronavirus (COVID-19) has not gone away and continues to have a major impact on people's lives. Statistics allow us to keep track of the pandemic; not just through the number of deaths or infections, but also with measures of the impact on the economy and businesses, as well as people's well-being, working situation and their daily routine. It has been established that COVID-19 spreads via respiratory droplets which can be transmitted through coughing, sneezing, or speaking. It is necessary to understand how the virus spread across the globe and to analyse whether responses such as shutdowns were effective in flattening the curve.

1.2. Problem Statement:

Visualize and observe COVID-19 cases, deaths, recoveries and observe patterns between new cases, deaths, recoveries and draw conclusions.

1.3. Objectives

The main objective of this study is to visualize COVID-19 data from the first several weeks of the outbreak to determine the point when the outbreak became a global pandemic. In addition, the study will visualize and observe COVID-19 cases, deaths, and recoveries through Plotly, a visualization tool available in python, and R which supports a number of interactive, high-quality graphs and is a great tool for data scientists.

2. METHODOLOGY

2.1. Importing Essential Libraries

Required dependencies were first imported. The following dependencies are popularly used for data wrangling operations and visualizations:

```r
library(knitr),library(tidyverse),library(caret),library(dslabs),library(dplyr),library(ggplot2)
library(matrixStats),library(data.table),library(lubridate),library(lattice),library(reshape2)
library(mice),library(DataExplorer),library(VIM),library(readr),library(plotly)
library(scales)
```

2.2. Dataset:

Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE) accumulates, processes and makes data-set sources available on their github page. These data-sets were used in this study.

2.3. Loading the required datasets.

<mark>Terms of use: As stated by JHU CSSE in link. [Data to be used only for research purposes]. The dataset has been loaded using the code below:</mark>

```r
# Load the readr, ggplot2, and dplyr packages
devtools::install_github("RamiKrispin/coronavirus")

library(coronavirus)
update_dataset ()
```

I will start with using the *dim* function to print out the dimensionality of our dataframe.

```
274512      7
```

The dataset has 268128 rows and 7 columns.

The summary function is used to review the contents and shape of the data. As its name implies, the summarize function reduces a data frame to a summary of just one vector or value. This function is valuable because it often provides exactly what is needed in terms of summary statistics.

```r
summary(coronavirus)
```

Now that a general overview of the nature of the data as shown by the statistics above, an in-depth analysis of the data is now required. I have noticed that there are 668 missing values. For such a large dataset, it is highly expected.

The str method will allows us to know the data type of each variable and factor levels if any.

```
'data.frame'          : 274512 obs. of  7 variables:
date                  : Date, format: "2020-01-22" "2020-01-23" ...
province              : chr  "" "" "" "" ...
country               : chr  "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...
lat                   : num  33.9 33.9 33.9 33.9 33.9 ...
long                  : num  67.7 67.7 67.7 67.7 67.7 ...
type                  : chr  "confirmed" "confirmed" "confirmed" "confirmed" ...
cases                 : int  0 0 0 0 0 0 0 0 0 0 ...
```

This coronavirus dataset has the following fields:

• **date -** The date of the summary

• **province -** The province or state, when applicable

• **country -** The country or region name

• **lat -** Latitude point

• **long -** Longitude point

• **type -** the type of case (i.e., confirmed, death)

• **cases -** the number of daily cases (corresponding to the case type)

It is really hard to digest a huge dataset which have 7 columns and a thousand of rows. I am going to use the head() to display and view the first n rows present in the input data frame as below:

```
head(coronavirus)

date province    country       lat       long      type      cases
1. 2020-01-22   Afghanistan 33.93911 67.70995   confirmed     0
2. 2020-01-23   Afghanistan 33.93911 67.70995   confirmed     0
3. 2020-01-24   Afghanistan 33.93911 67.70995   confirmed     0
4. 2020-01-25   Afghanistan 33.93911 67.70995   confirmed     0
5. 2020-01-26   Afghanistan 33.93911 67.70995   confirmed     0
6. 2020-01-27   Afghanistan 33.93911 67.70995   confirmed     0
```

Another important aspect of showing your key findings is to use only a set of charts that infer key insight from the data rather than showing too many charts with redundant information.

3. THE GLOBAL IMPACT OF COVID-19: VISUAL ANALYSIS

Early on in the outbreak, the COVID-19 cases were primarily centered in China before it rapidy spread into Europe, USA and the rest of the world gradually. The visual analysis is structured into four parts for easy analysis as follows:

3.1 a tree map of the confirmed cases

3.2 Geographical distribution of confirmed, deaths and recovered cases

3.3 Density line plots of confirmed, deaths and recovered cases

3.4 Most affected countries in the world

3.5 Least affected countries in the world.

3.1. Summary of the total confirmed cases

Table 1 shows the cumulative confirmed cases of COVID-19 worldwide by country.

Table 1: Total confirmed cases by country

No.	Country	Total cases
1	US	19740468
2	India	10266674
3	Brazil	7619200
4	Russia	3100018
5	France	2657624
6	United Kingdom	2440202
7	Turkey	2194272
8	Italy	2083689
9	Spain	1910218
10	Germany	1741153
11	Colombia	1626461
12	Argentina	1613928
13	Mexico	1413935
14	Poland	1281414
15	Iran	1218753
16	Ukraine	1076880

Table 1 (continued): Total confirmed cases by country

17	South Africa	1039161
18	Peru	1010496
19	Netherlands	798592
20	Indonesia	735124

```r
# Summary of the total confrimed cases by country (top 20):
library(dplyr)
summary_df <- coronavirus %>%
 filter(type == "confirmed") %>%
 group_by(country) %>%
 summarise(total_cases = sum(cases)) %>%
 arrange(-total_cases)

summary_df %>% head(20)
```

Even though the outbreak was first identified in China, there is no country from East Asia in Table 1. Four of the listed countries (France, Germany, Italy, and Spain) are in Europe and share borders. USA, India, Brazil, Russia, France, United Kingdom, Turkey, Italy, Spain and Germany have the highest number of total cases in the world, however just reading numbers in a table makes it hard to get a sense of the scale and growth of the outbreak.

3.2. Treemap of confirmed cases.

```r
# Plot the confirmed cases distribution by counrty with treemap plot:
 conf_df <- coronavirus %>%
 filter(type == "confirmed") %>%
 group_by(country) %>%
 summarise(total_cases = sum(cases)) %>%
 arrange(-total_cases) %>%
 mutate(parents = "Confirmed") %>%
 ungroup()

plot_ly(data = conf_df,
     type= "treemap",
     values = ~total_cases,
     labels= ~ country,
     parents=  ~parents,
     domain = list(column=0),
```

```
    name = "Confirmed",
  textinfo="label+value+percent parent")
```
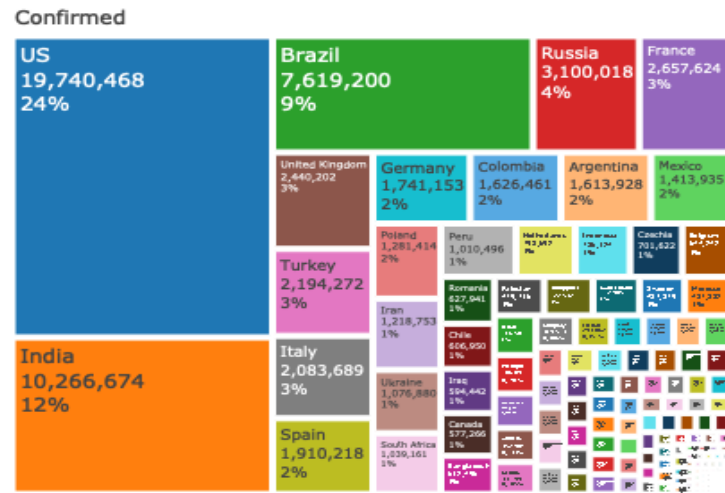

Figure 1: Tree map of confirmed cases

Observation: From Figure 1, it can be seen that the disease has infected a larger number of people in USA, India, Brazil and many European countries such as UK, France, Italy more than China where the virus was first discovered.

3.3. Bubble maps

To get more context on the impact of the virus on a geographical landscape we can plot the spread of confirmed, deaths and recovery using bubble maps. These bubble maps will give a visual general overview of the spread, death and recovery rates.

```
#Geographical distribution of confirmed, death and recovered cases.
world <- map_data("world")
confirmed_cases <- coronavirus %>%
filter(type == "confirmed")) %>%
group_by(country,long,lat) %>%
summarise(total_cases = sum(cases)) %>%
arrange(-total_cases) %>%
mutate(case = "Confirmed") %>%
ungroup()

p0<-ggplot() +
geom_polygon(data = world, aes(x=long, y=lat, group=group), fill="gray", colour = "darkgray", size=0.5, alpha=0.3) + #resizing to just CONUS
geom_point(data=confirmed_cases, aes(x=long, y=lat, color=total_cases, size=total_cases), alpha=0.5) + #everything inside aes appears in legend
coord_map() + #plotting with correct mercator projection (prior plot was cartesian coordinates)
```

```
scale_color_viridis_c()+
ggtitle("Confirmed COVID-19 Cases in the World.") +
guides( colour = guide_legend()) +
theme_void()  #gets rid of axes
p0
```
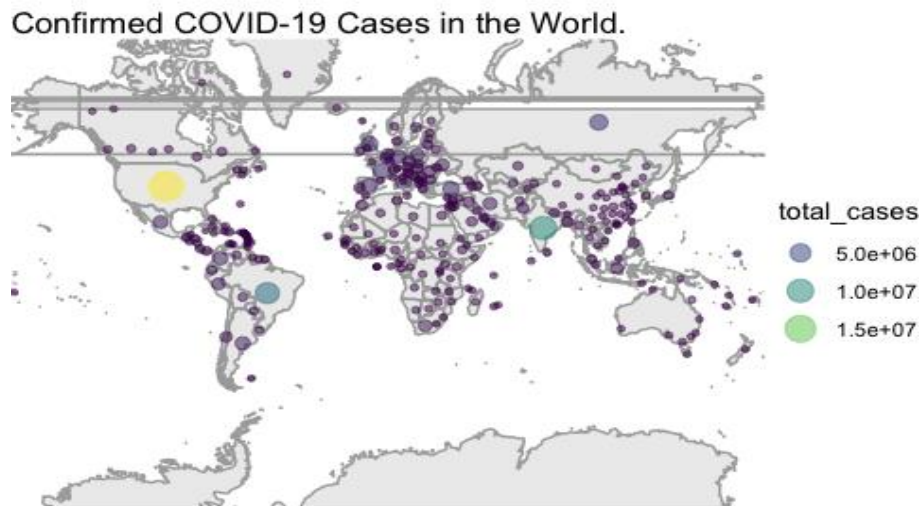

Figure 2: World map of confirmed COVID-19 cases.

```
#Geographical distribution of deaths cases
death_cases <- coronavirus %>%
filter(type == "death") %>%
group_by(country,long,lat) %>%
summarise(total_cases = sum(cases)) %>%
arrange(-total_cases) %>%
mutate(case = "death") %>%
ungroup()

p1<-ggplot() +
geom_polygon(data = world, aes(x=long, y=lat, group=group), fill="gray", colour = "darkgray", size=0.5, alpha=0.3) + #resizing to ju
st CONUS
geom_point(data=death_cases, aes(x=long, y=lat, color=total_cases, size=total_cases), alpha=0.5) + #everything inside aes appe
ars in legend
coord_map() + #plotting with correct mercator projection (prior plot was cartesian coordinates)
scale_color_viridis_c()+
ggtitle("Death COVID-19 Cases in the World.") +
guides( colour = guide_legend()) +
theme_void()  #gets rid of axes
p1
```

Death COVID-19 Cases in the World.
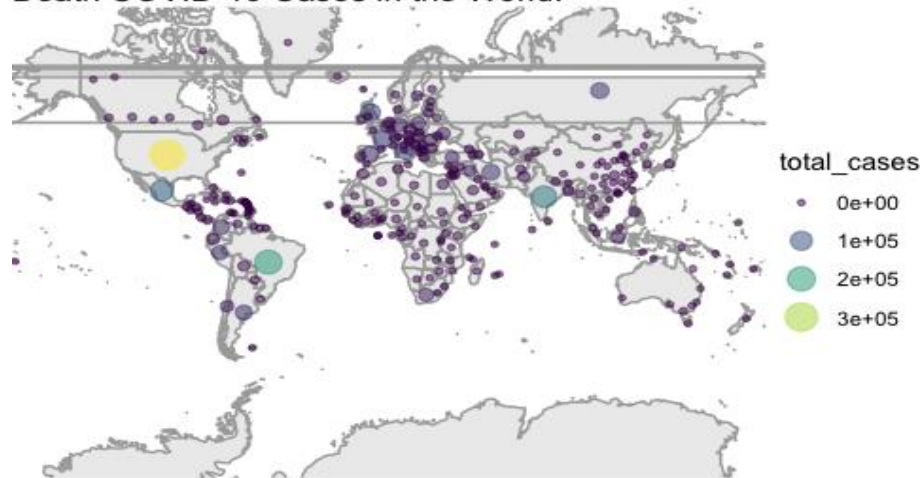
Figure 3: World map of COVID-19 deaths

```
#Geographical distribution of recovered cases
recovered_cases <- coronavirus %>%
filter(type == "recovered") %>%
group_by(country,long,lat) %>%
summarise(total_cases = sum(cases)) %>%
arrange(-total_cases) %>%
mutate(case = "recovered") %>%
ungroup()

p2<-ggplot() +
geom_polygon(data = world, aes(x=long, y=lat, group=group), fill="gray", colour = "darkgray", size=0.5, alpha=0.3) + #resizing to ju
st CONUS
geom_point(data=recovered_cases, aes(x=long, y=lat, color=total_cases, size=total_cases), alpha=0.5) + #everything inside aes a
ppears in legend
coord_map() + #plotting with correct mercator projection (prior plot was cartesian coordinates)
scale_color_viridis_c()+
ggtitle("Recovered COVID-19 Cases in the World.") +
guides( colour = guide_legend()) +
theme_void()  #gets rid of axes
p2
```
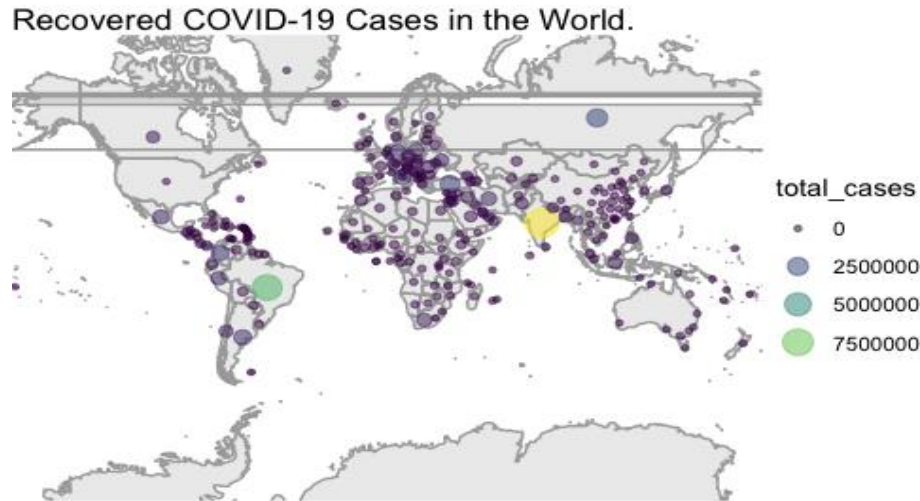
Figure 4: World map of Recovered COVID-19 Cases

The scatterplots shown in Figure 1 to Figure 4 give a general overview of the reported COVID-19 cases around the world. The markers with yellowish color reflect the relatively higher cases compared with those darker colors. Even though the disease originated in China, it is observed that the number of deaths in China are considerably low and a lot of patients recovered. It is further observed that the US hits the most confirmed death cases but seems to be staggering below with recovery cases compared to South American countries, India and other Asian countries. European countries such as UK, Russia, Spain, France and Italy also show a significantly higher number of all cases combined.

3.4. Density Line Plot of confirmed, deaths and recovered cases.

```
# Plotting the total cases by type worldwide:
library(plotly)
coronavirus %>%
  group_by(type, date) %>%
  summarise(total_cases = sum(cases)) %>%
  pivot_wider(names_from = type, values_from = total_cases) %>%
  arrange(date) %>%
  mutate(active = confirmed - death - recovered) %>%
  mutate(active_total = cumsum(active),
      recovered_total = cumsum(recovered),
      death_total = cumsum(death)) %>%
  plot_ly(x = ~ date,
        y = ~ active_total,
      name = 'Active',
      fillcolor = '#1f77b4',
      type = 'scatter',
      mode = 'none',
```

```
      stackgroup = 'one') %>%
 add_trace(y = ~ death_total,
      name = "Death",
      fillcolor = '#E41317') %>%
 add_trace(y = ~recovered_total,
      name = 'Recovered',
      fillcolor = 'forestgreen') %>%
layout(title = "Distribution of Covid19 Cases Worldwide",
      legend = list(x = 0.1, y = 0.9),
      yaxis = list(title = "Number of Cases"),
      xaxis = list(title = "Source: Johns Hopkins University Center for Systems Science and Engineering"))
```
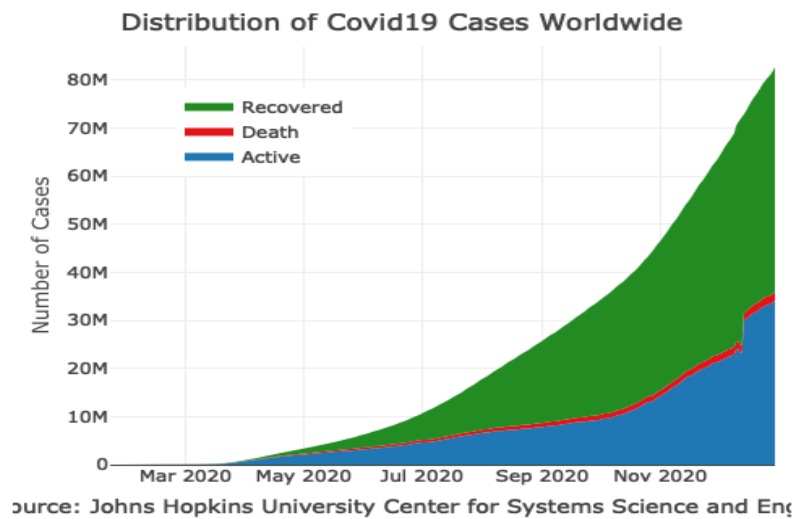

Figure 5: Distribution of COVID-19 Cases Worldwide

Here I have only shown the increasing trends for recovered, death and active cases over time, the graph clearly shows that the percentage prpoprtion of people recovering from the deasease is higher than the active and death cases combined.

4. DESCRIPTIVE ANALYSIS ON INFECTED, MORTALITY AND RECOVERY RATES

4.1. Most Affected Countries in the world

I started by mining my data and split it into a data-frame that shows detailed data of country by country using the codes below.The table below shows the first 6 countries of the data frame.

```
total_cases_per_country <- data.frame(coronavirus$country,coronavirus$type,coronavirus$cases)
# renaming columns
colnames(total_cases_per_country)[1:3] <-c("country","type","cases")
total_cases_per_country_wide<- total_cases_per_country%>%
group_by(country, type) %>%
summarise(total_cases = sum(cases)) %>%
pivot_wider(names_from = type, values_from = total_cases) %>%
mutate(active = confirmed - death - recovered) %>%
mutate(active_total = cumsum(active),
recovered_total = cumsum(recovered),
death_total = cumsum(death))
```

I have used the code below to arrange the countries in order of the number of confirmed cases. The table shows the first 6 least affected countries.

```
sum_total_cases_per_country <- data.frame(total_cases_per_country_wide$country,total_cases_per_country_wide$confirmed, tot
al_cases_per_country_wide$active_total,total_cases_per_country_wide$recovered_total,total_cases_per_country_wide$death_tota
l)

# renaming columns

colnames(sum_total_cases_per_country)[1:5]<-c("country","confirmed","active_total","recovered_total","death_total")

countries_in_order <- sum_total_cases_per_country[order(sum_total_cases_per_country[, "confirmed"]), , drop = FALSE]
```

I am going to analyse the first 20 most affected countries and the least 20 affected countries using the codes below. Here I have used various charts to show how information can be mined from data sources.

4.2. Most affected Countries in the world:

```
most_affected_countries_20 <- tail(countries_in_order,n=20)
most_affected_countries_20
```

Table 2: Most affected countries in the world

No.	country	confirmed	active total	total recove red	death total
1.	US	19740468	19398156	0	342312
2.	India	10266674	257656	9860280	148738
3.	Brazil	7619200	611233	6814092	193875
4.	Russia	3100018	544861	2499465	55692
5.	France	2657624	2394150	198966	64508
6.	United Kingdom	2440202	2362131	5414	72657
7.	Turkey	2194272	95001	2078629	20642
8.	Italy	2083689	564395	1445690	73604
9.	Spain	1910218	1709153	150376	50689
10.	Germany	1741153	361971	1345952	33230
11.	Colombia	1626461	86372	1497180	42909
12.	Argentina	1613928	144089	1426676	43163
13.	Mexico	1413935	222267	1066771	124897
14.	Poland	1281414	227506	1025889	28019
15.	Iran	1218753	184944	978714	55095
16.	Ukraine	1076880	333679	724143	19058
17.	South Africa	1039161	143531	867597	28033
18.	Peru	1010496	21604	951318	37574
19.	Netherlands	798592	777786	9389	11417
20.	Indonesia	735124	109439	603741	21944

The cases reported are visualized in analytics dashboard to show the outbreak trend for confirmed, recovered and deaths cases for all regions and countries. This aligns with our objectives to show the outbreak progress over the period of time for each segment. From Table 2, it was found that the total number of confirmed cases for all countries and regions are increasing steadily, with India and Bulgaria having the highest numbers of people that have recovered from COVID-19.

4.2.1. Confirmed cases: Analysis using pie chart

```
lollipop_chart<-most_affected_countries_20 %>%
mutate(prop = confirmed / sum(countries_in_order$confirmed)*100) -> mydf
ggdotchart(lollipop_chart, x = "country", y = "confirmed",
color="country",
```

```
sorting = "ascending",              # Sort value in descending order
add = "segments",                   # Add segments from y = 0 to dots
dot.size = 6,                       # Large dot size
label = round(lollipop_chart$prop,2),
font.label = list(color = "black", size = 9,
vjust = 0.5),            # Adjust label parameters
ggtheme = theme_pubr()              # ggplot2 theme)
```
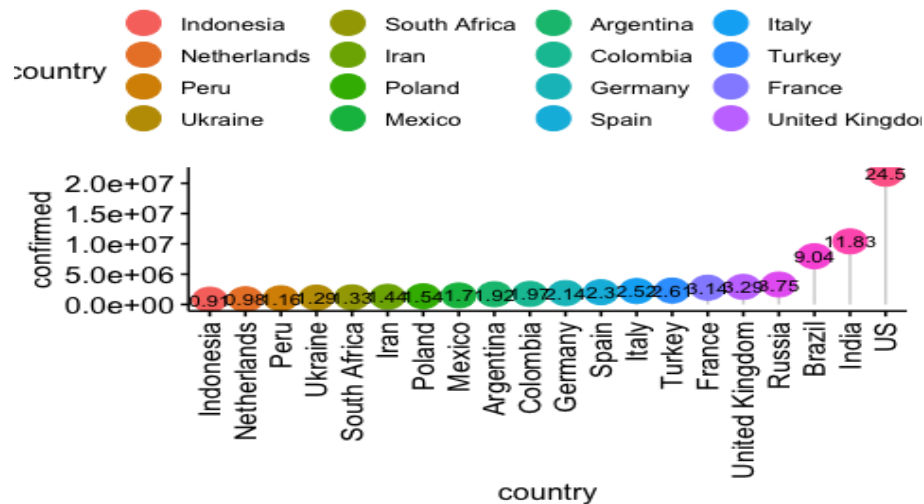

Figure 6: Confirmed cases of the 20 most affected countries

Figure 6 shows that USA has the highest number of confirmed cases compared to other regions. Next to USA is India, Brazil, Russia, France, UK, Ukraine, Turkey, Spain, South Africa, Mexico and Iran show high number of infected patients.

4.2.2. Recovery rates: Analysis using bar graph

```
bar_graph<-ggplot(data=most_affected_countries_20, aes(x=country, y=recovered_total)) +
geom_bar(stat="identity", fill="steelblue")+theme_minimal() + theme(axis.text.x = element_text(angle = 45, hjust = 1))
bar_graph
```
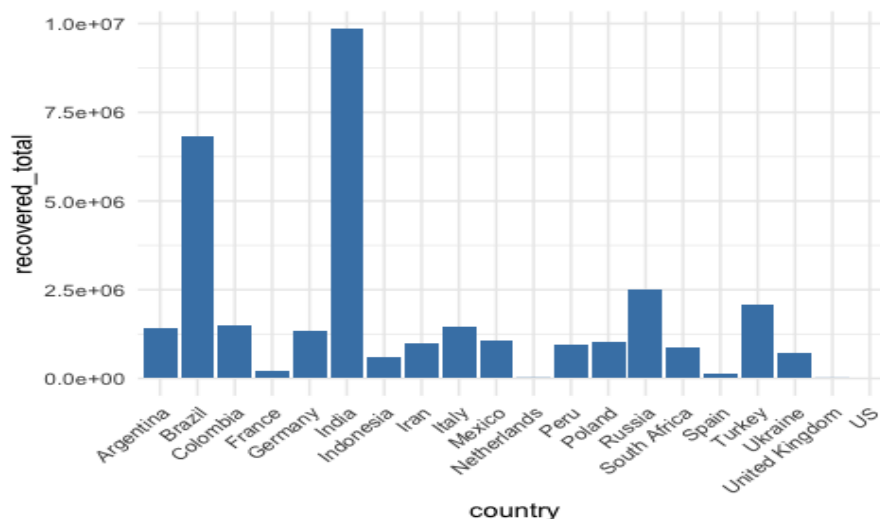
From the Figure 7 we can see that even though USA has the highest number of confirmed cases compared to India and Brazil, their recovery rate is higher compared to USA.from a medical point of view, this is an interesting fact to research further.

4.2.3. Death rates: Analysis using tree map

```
library(treemap)
treemap(most_affected_countries_20, #Your data frame object
index=c("country"), #A list of your categorical variables
vSize = "death_total", #This is your quantitative variable
type="index", #Type sets the organization and color scheme of your treemap
palette = "Reds", #Select your color palette from the RColorBrewer presets or make your own.
title="Death Totals", #Customize your title
fontsize.title = 14) #Change the font size of the title
```
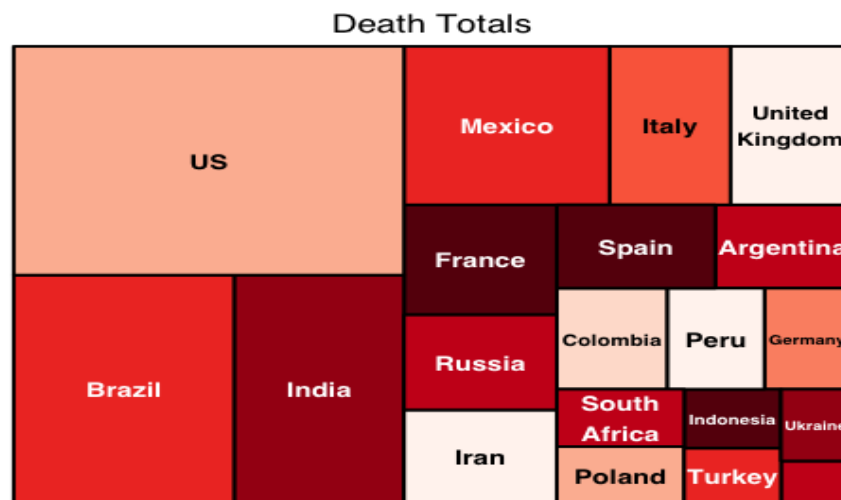

Figure 8: Death cases of the 20 most affected countries

The tree map portrayed in Figure 8 shows that countries with the highest confirmed rates do also have the highest mortality rates. An interesting point to note is that South Africa has a significant number of mortality rates although its confrimed cases are lower than Iran, Turkey and Ukraine.

4.3. Least affected Countries in the world

```
least_affected_countries_20 <- head(countries_in_order,n=20)
least_affected_countries_20
```

Table 3: Least affected countries in the world

No.	country	confirmed	active total	total recovered	death total
1.	Tanzania	**509**	305	183	21

14

2.	Barbados	**372**	53	312	7
3.	Cambodia	**366**	5	371	0
4.	Saint Lucia	**340**	57	278	5
5.	Seychelles	**256**	44	212	0
6.	Antigua and Barbuda	**158**	5	148	5
7.	Brunei	**157**	5	149	3
8.	Grenada	**127**	20	107	0
9.	Saint Vincent and the Grenadines	**115**	26	89	0
10.	Dominica	**88**	5	83	0
11.	Fiji	**49**	3	44	2
12.	Timor-Leste	**44**	12	32	0
13.	Laos	**41**	1	40	2
14.	Saint Kitts and Nevis	**32**	3	29	0
15.	Holy See	**27**	12	15	0
16.	Solomon Islands	**17**	12	5	0
17.	MS Zaandam	**9**	0	7	0
18.	Marshall Islands	**4**	0	4	0
19.	Samoa	**2**	0	2	2
20.	Vanuatu	**1**	0	1	0

4.3.1. Confirmed cases: Analysis using lollipop chart

```
lollipop_chart1<-least_affected_countries_20 %>%
mutate(prop = confirmed / sum(countries_in_order$confirmed)*100) -> mydf
ggdotchart(lollipop_chart1, x = "country", y = "confirmed",
color="country",
sorting = "ascending",            # Sort value in descending order
add = "segments",                 # Add segments from y = 0 to dots
dot.size = 6,                     # Large dot size
label = round(lollipop_chart1$prop,2),
font.label = list(color = "black", size = 9,
vjust = 0.5),          # Adjust label parameters
ggtheme = theme_pubr()            # ggplot2 theme)
```
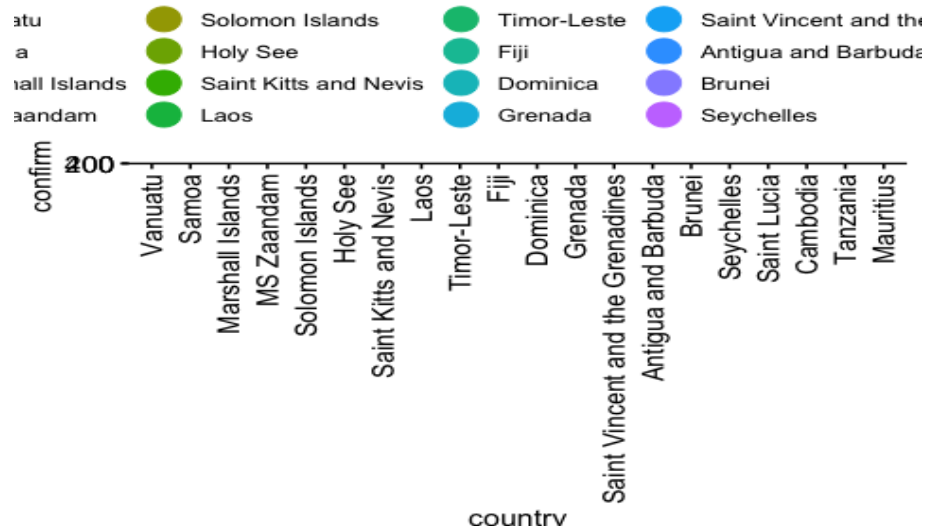
Figure 9: Confirmed case of the 20 least affected countries

Interestingly, all countries in this band show almost zero percentage of confirmed cases. A closer look at the countries shows that these countries have very low populations and are unlikely to be visited by international tourists. Moreover, they are mostly islands with the exception of Brunei and Bhutan which are tiny Kingdoms. This data shows the safest countries not impacted by the pandemic.

4.3.2. Recovery rates: Analysis using bar graph

```
bar_graph1<-ggplot(data=least_affected_countries_20, aes(x=country, y=recovered_total)) +
geom_bar(stat="identity", fill="steelblue")+theme_minimal() + theme(axis.text.x = element_text(angle = 45, hjust = 1))
bar_graph1
```
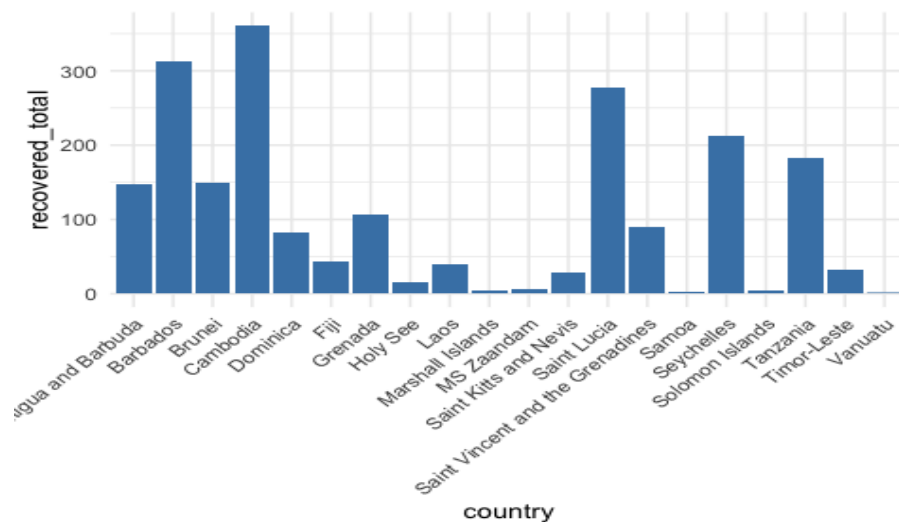

Figure 10: Recovery cases of the 20 least affected countries

Figure 10 shows that Bhutan, Cambodia, Saint Lucia, Seychelles' and Barbuda have high number of people recovering from the virus.

4.3.3. Death rates: Analysis using tree map

```
treemap(least_affected_countries_20, #Your data frame object
index=c("country"), #A list of your categorical variables
vSize = "death_total", #This is your quantitative variable
type="index", #Type sets the organization and color scheme of your treemap
palette = "Reds", #Select your color palette from the RColorBrewer presets or make your own.
title="Death Totals", #Customize your title
fontsize.title = 14 )#Change the font size of the title
```
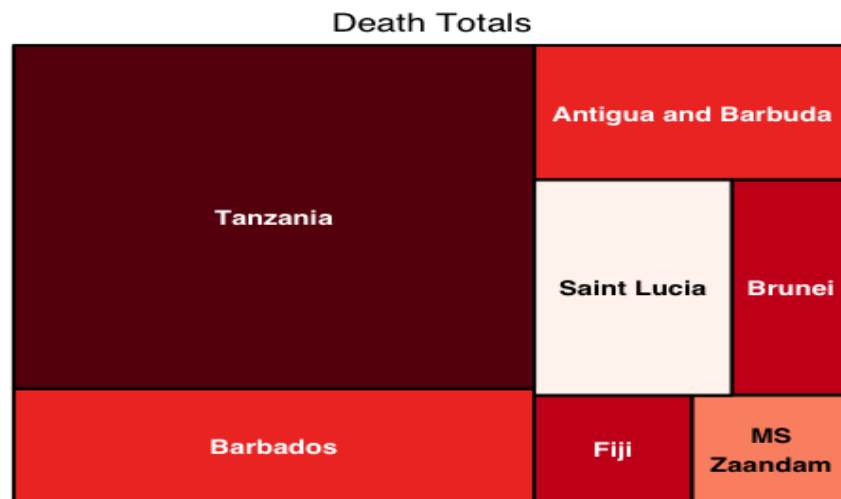

Figure 11: Death cases of the 20 least affected countries

The tree map shown in Figure 11 is consistent with graphs in Figure 10, in this band the countries with the highest number of confirmed cases also have the highest number of recovered and deaths respectively. Tanzania here records high deaths rates due to the fact that when other countries went into lock down, the country's political leaders refused to follow the WHO guidelines.

5. CONCLUSION

Understanding a bigger picture is an important step to take actions. One conclusion from this essay is that the Data Visualizations are giving a big picture of how COVID-19 is spreading across the globe, knowing of its impact to the people's lives in terms of deaths, effects on economy and how it affects our way of life helps us in making informed decisions in trying to stop the spread of the disease. It is clear from the visualizations above that the number of active cases is increasing with each day, however the recovery rates are also significantly higher with Brazil and India

recording over 90% recovery rates. With the exception of north Asian countries, the graphs show that North America, Brazil, India and Europe are the most affected regions in the world most probably due to their high population density.

6. BIBLIOGRAPHY

Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE):
https://systems.jhu.edu/

Beautiful graphics ggplot2
https://rstudio-pubs-static.s3.amazonaws.com/

WHO, 2020 WHO

Coronavirus disease (COVID-2019) situation reports-(2020)

https://www.who.int/emergencies/diseases/novel-coronavirus-2019/situation-reports

Google Scholar