VISUALIZING PANDEMIC DATA

COSC3000: VISUALIZATION, COMPUTER GRAPHICS & DATA ANALYSIS

William E. G. Kvaale
University of Queensland
w.kvaale@uq.net.au
s46301303

May 6, 2020

ABSTRACT

In this visualization project we will take a dive into *The 2019 Novel Coronavirus Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University*. This repository contains time series data describing the situation per country regarding amount of confirmed cases, deaths, and recovered cases. It is updated daily by the CSSE, and has become the main source of data for most analytics and visualization projects concering the Coronavirus disease 2019 (COVID-19).

The aim of this project is to undertake the challenge to make interesting visualizations and showcase a variety of aspects of the data, while exploring and investigating a large multi-dimensional dataset.

The dataset

The dataset is obtained from the Johns Hopkins University's Center for System Science and Engineering (CSSE) [1]. The CSSE daily updates a GitHub repository with time series with the current status for the date, and the preceding dates, for each country. It also contains data for status on number of confirmed, recovered and death cases, per country.

The *CSV-files* (Comma-separated values) is read using Python and Pandas, which is a open source tool for data analytics and data management. Pandas' data tables is named *DataFrames*. This concept is quite straightforward, when compared to tables in spreadsheet software such as LibreOffice Calc or Microsoft Excel. See example below.

```python
import pandas as pd
REPO = "https://raw.githubusercontent.com/CSSEGISandData/COVID-19/"
FILE = "web-data/data/cases_country.csv"
df = pd.read_csv(REPO+FILE)
```

Listing 1: Example for reading CSV-files using Python and Pandas

The code seen in Listing 1 yields the following result, as seen in Figure 1.

	Province/State	Country/Region	Lat	Long	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	...	4/24/20	4/25/20	4/26/20	4/27/20	4/28/20	4/29/20	4/30/20	5/1/20	5/2/20	5/3/20
0	NaN	Afghanistan	33.000000	65.000000	0	0	0	0	0	0	...	1351	1463	1531	1703	1828	1939	2171	2335	2469	2704
1	NaN	Albania	41.153300	20.168300	0	0	0	0	0	0	...	678	712	726	736	750	766	773	782	789	795
2	NaN	Algeria	28.033900	1.659600	0	0	0	0	0	0	...	3127	3256	3382	3517	3649	3848	4006	4154	4295	4474
3	NaN	Andorra	42.506300	1.521800	0	0	0	0	0	0	...	731	738	738	743	743	743	745	745	747	748
4	NaN	Angola	-11.202700	17.873900	0	0	0	0	0	0	...	25	25	26	27	27	27	27	30	35	35
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
261	NaN	Western Sahara	24.215500	-12.885800	0	0	0	0	0	0	...	6	6	6	6	6	6	6	6	6	6
262	NaN	Sao Tome and Principe	0.186360	6.613081	0	0	0	0	0	0	...	4	4	4	4	8	8	14	16	16	16
263	NaN	Yemen	15.552727	48.516388	0	0	0	0	0	0	...	1	1	1	1	1	6	6	7	10	10
264	NaN	Comoros	-11.645500	43.333300	0	0	0	0	0	0	...	0	0	0	0	0	0	1	1	3	3
265	NaN	Tajikistan	38.861034	71.276093	0	0	0	0	0	0	...	0	0	0	0	0	0	15	15	76	128

Figure 1: Dataframe for Confirmed Cases

Preprocessing the data

As for most datasets, it usually needs some prior cleaning up before being used for visualizations. This one is no exception.

The following steps were taken in order to prepare the dataset:

1. **Download raw datasets for**
   - Confirmed cases
   - Recovered cases
   - Death cases
2. **Load the datasets into dataframes**
   - Using Pandas' read_csv()
3. **Transpose dates in dataset**
   - Transforming dates, from columns to rows
   - This can be achieved using Pandas' melt()
4. **Merge the three dataframes into one**
   - Using Pandas' merge()
5. **Clean the data**
   - Convert date from string to datetime using Pandas' to_datetime()
   - Address entries with missing values
   - Extract the cruise ships reports
6. **Finally, we can aggregate the data**
   - Add column containing *Active Cases*
     - $Active = Confirmed - Recovered - Death$
     - *This being cumulative data*
   - Add a new column placeholder for day-by-day data
     - New Confirmed Cases
     - New Death Cases
     - New Recovered Cases

After a thoroughly preprocessing of the data, we are now left with a DataFrame which is ready to be visualized. The dataframe can be seen in Figure 15 in the Appendix.

Global Impact

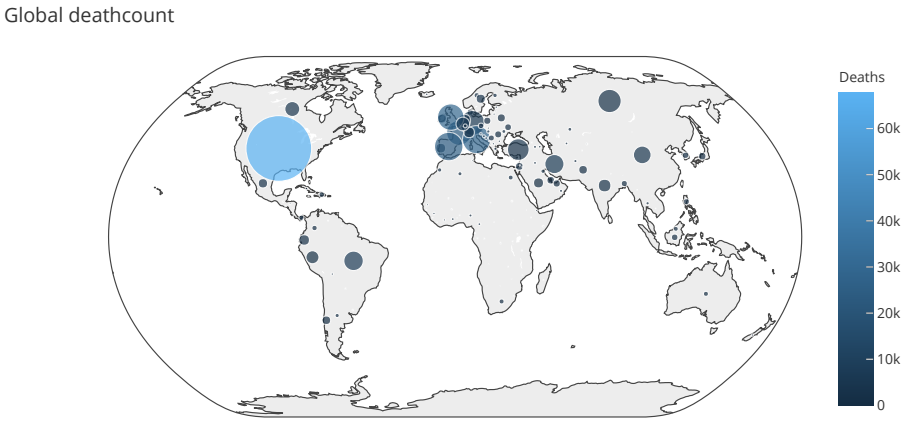

Figure 2: Scatter plot showing death cases globally

This project is a great opportunity to become familiarized with a great interactive visualization and graphing tool, namely Plotly.

Due to the writer's inquisitive nature, the initial visualization to be made was a scatter plot on the worldmap, as seen in Figure 2. Each *bubble* represent a country, and the bubble's size represents the amount of confirmed cases. Colorgrading schemes representing the number of death cases was also applied. One of the first observation made by looking at this visualization was how significantly much worse the situation seemed in the US and UK, in comparison to Australia. Another map representing the same situation was made, using a *choropleth map*. This can be seen in Figure 3.

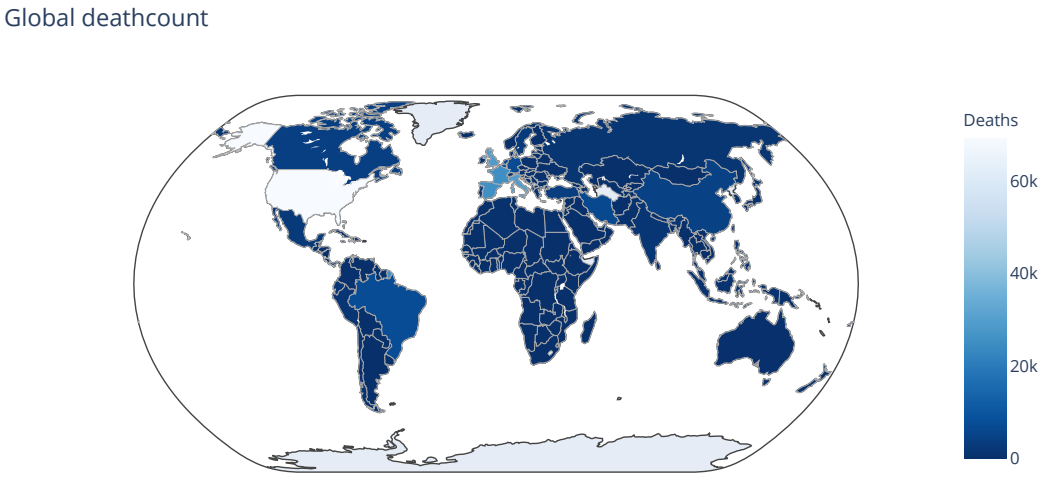

Figure 3: Choropleth map showing death cases globally[1]

These visualizations were made using the latitude and longitude for each country, as well as ISO 3166-1 alpha-3 .

[1]No data for Antarctica, nor Greenland

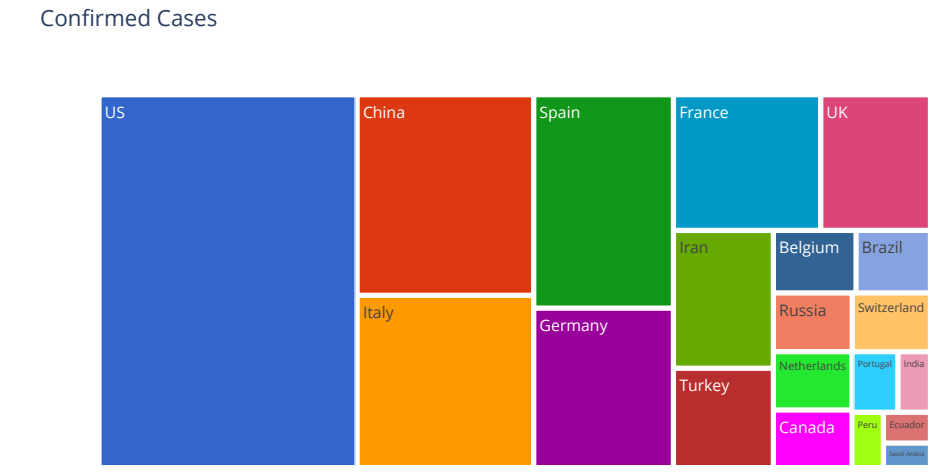

Figure 4: Tree Map for Confirmed Cases - Top 20 Countries

An additional way of displaying the global impact this virus has had, is by simple charts. *Tree Maps* are originally a way of showing hierarchical data, however, it is also a good alternative to the traditional pie chart. In Figure 4 we quickly get a feel of the distribution of the cases. US taking up the largest portion of this map, as we saw earlier in the choropleth and scatter maps. The most elementary interpretation from this is that the US has more cases than any other country. That being said, it might be due to massively testing their population - which might suggest that other countries could, to varying degree, have many unrecorded cases (hidden statistics).

Bar charts can be used to represent univariate and bivariate data quite concisely. Both the bar chart and tree map are highly effective ways of easily having bivariate data being easy to comprehend.

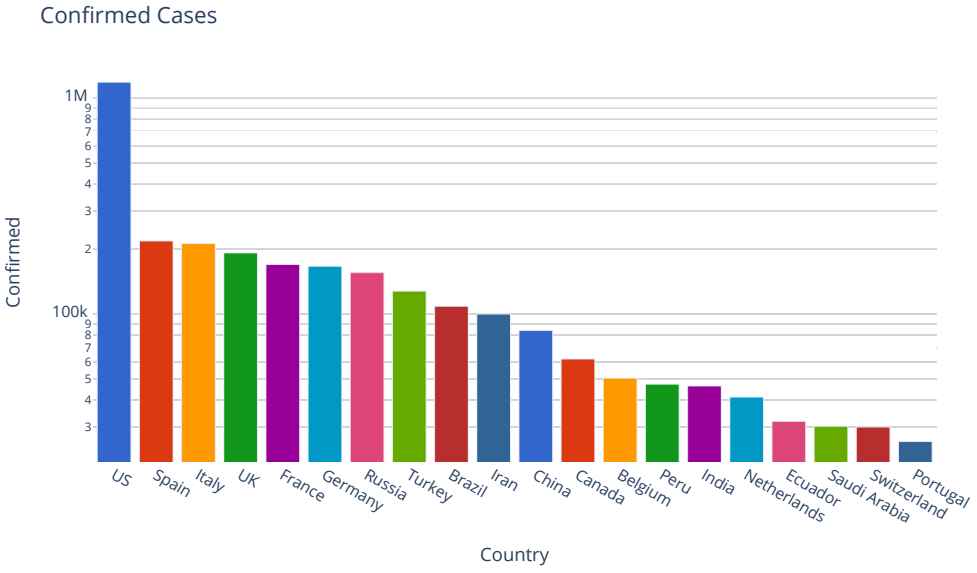

Figure 5: Bar Chart for Confirmed Cases - Top 20 Countries[2]

[2]Notice the logarithmic y-axis

Linear versus logarithmic, and the nature of epidemic growth

A important aspect to understand when looking at graphs representing epidemic data, is exponential growth. Viruses follow this growth regime. What causes new cases, is the existing cases. Therefore viruses grow exponentially. This can be formulated more precisely mathematically:

$$\begin{aligned} \Delta N_d &= EpN_d \\ N_{d+1} &= N_d + EpN_d \\ N_{d+1} &= N_d(1 + Ep) \end{aligned} \tag{1}$$

In this equation, N_d represents the daily number of cases. E represents the average number of people who is exposed to someone infected each day. And p is the probability for each of these exposures resulting in an infection.

By looking at the fact that N_{d+1} is simply equal to a factor $(1 + Ep) > 1$ multiplyed by N_d. The most common misconception in a pandemic, is seeing *Country X* having 5000 cases, and *Country Y* having 50, thinking that this must indeed imply that *Y* is doing so much better than *X*? Well, another way to look at it, is that *Y* is just a couple of weeks behind *X* in regards to number of cases. This is due to the nature of growth for viruses.

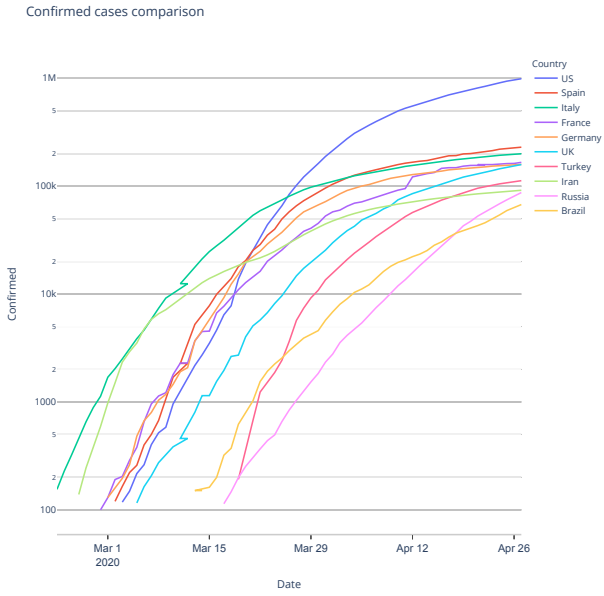

Figure 6: Logarithmic

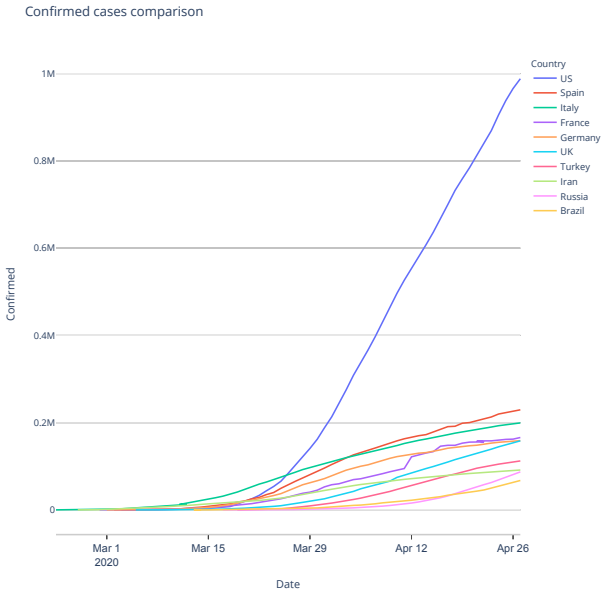

Figure 7: Linear

The situation for the US in Figure 7 is looking way worse, than in Figure 6. This is due to the y-axis being logarithmic, instead of linear. In a logarithmic axis', each "step" in y direction corresponds to multiplying by a factor, which in this case is 10, since we are using log_{10}. What seems to be the trend for most of the countries, disregarding the US for a moment, is that the the growth is coming to a halt.

Comparing COVID-19 with SARS and Ebola

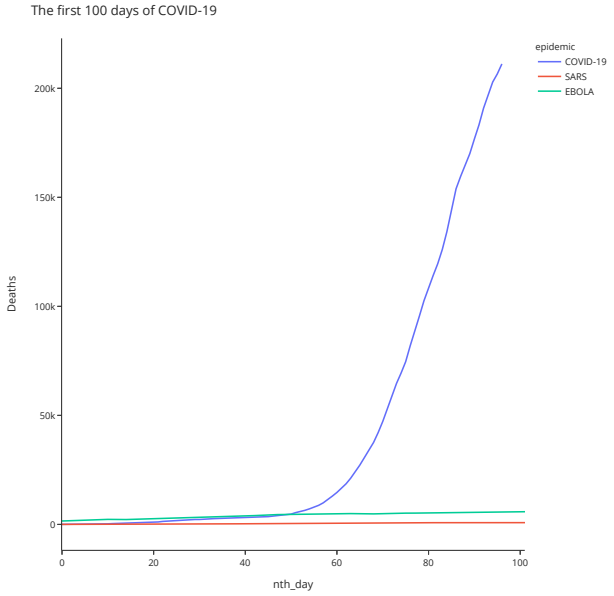

Figure 8: Linear y-axis

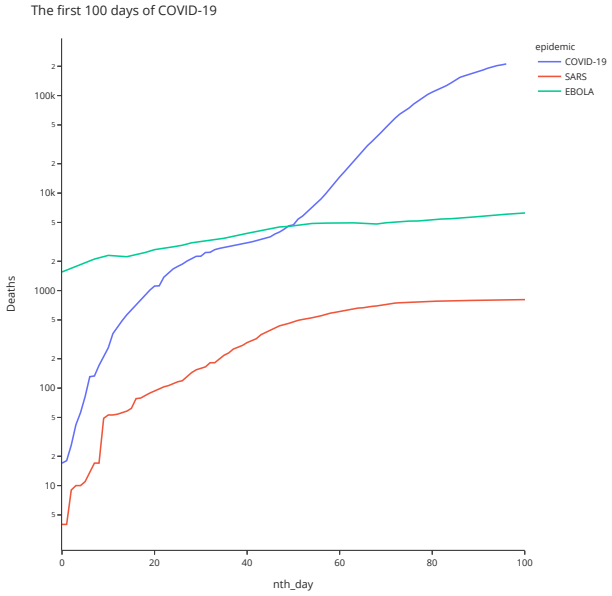

Figure 9: Logarithmic y-axis

For the vast majority of people, the COVID-19 pandemic is beyond any prior crisis experienced. In the period November 2002 - July 2003, the world was faced with Severe Acute Respiratory Syndrome Coronavirus, also known as *SARS*. COVID-19 (or SARS-COV-2) is it's successor. The worst hit countries of SARS in the early 2000's, was China and Hong Kong with respectively 5327 cases (349 deaths) and 1755 cases (299 deaths) [2]. Ebola Virus Disease (EVD) was a viral hemorrhagic fever with over 28 652 cases[3] and 11 325 deaths [3].

Another recent coronaviurs worth mentioning is the Middle East Respiratory Syndrome, *MERS*. The worst hit country by MERS is Saudi Arabia with 1029 cases and 452 deaths, which implies a 44% fatality rate. As for the novel COVID-19, US is worst hit with over 1.2 million cases and 71 148 deaths (≈6 % fatality).

As we can see in Figure 8 and Figure 9, which visualizes the growth of deaths the first 100 days, the graph with linear y-axis is quite worrying, and is what many news companies used early on to create what is known as clickbait.

Disclaimer:
It should be noted that the data for SARS and Ebola used in this particular comparison was found in a Kaggle kernel, and according to the author it was gathered from the World Health Organization's website.

[3]Total Cases (Suspected, Probable, Confirmed)

Working with multidimensional data

In the preprocessed dataset, a column was made to hold day-wise records for confirmed, death and recovered cases. A plot that can reveal data in a straightforward way by simply being somewhat intuitive, is a bubble line chart. And an aspect of the data that could be insightful to display, is the *new cases*, day by day.

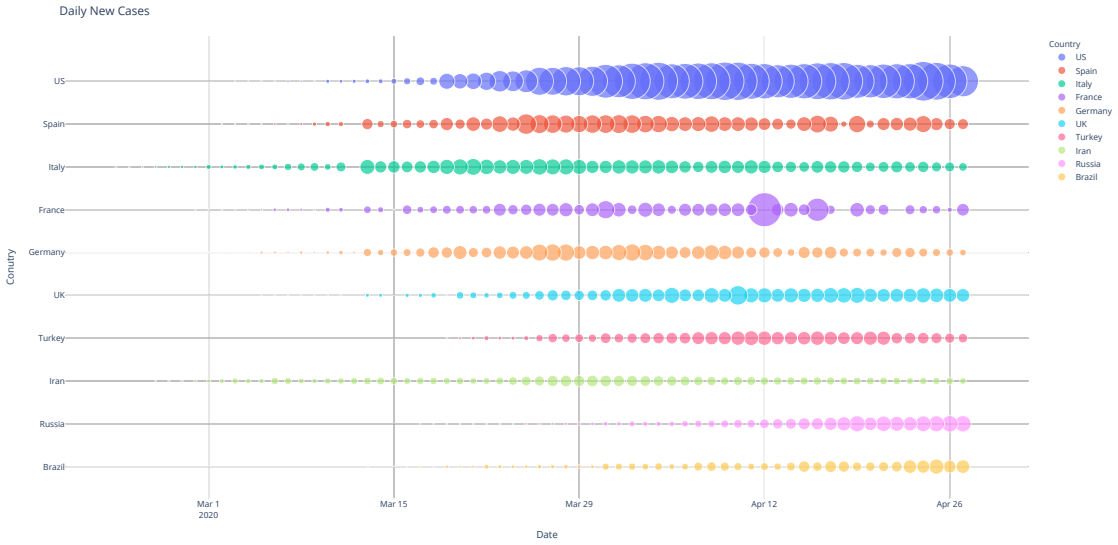

Figure 10: Daily New Cases

In this bubble chart, color is used to distinguish the countries, and size of the bubble is used to represent the number of daily new cases. As with the other plots, a recurring theme is US dominating the situation. A peculiar observation can be made on the 12^{th} and 16^{th} of April for France.

Displaying 3D visualizations on a 2D format

Having multiple variables to work with in this dataset, is for many a call for *3D plots*. However, displaying these interactive and fancy plots made in Ploty, on paper, is somewhat non-functional. By accessing the GitHub repository for this project, one could download the visualization notebooks, and entertain oneself with the interactive plots using Jupyter Notebook, or Visual Studio Code.

Nevertheless, the next page will include two plots made to reveal several aspects of the data. See Figure 11 to view the worst hit countries, with respect to number of death cases. In this plot, size of the spheres is representing the number of total cases for the respective country. Color is representing number of deaths. The y- and x-axis is aiding the size and color of the spheres, respectively.

In this output one could state that there are two clear clusters, and the recurring theme - US - being in a global maximum in this 3D room of death and confirmed cases.

As for in Figure 10, a 3D representation of the same situation, were made using a 3D Scatterplot and can be seen in Figure 12. Note that instead of top ten countries, here only five are showing due to the limiting aspect of showing 3D on a 2D medium (paper). In this 3D plot, both size and color is representing daily new cases, while the x-axis holds the dates and logarithmic y-axis represents the number of Active Cases. By observation, the $12th^{t=1\ th}$ of April can be considered as an outlier when compared to the rest of the time series.

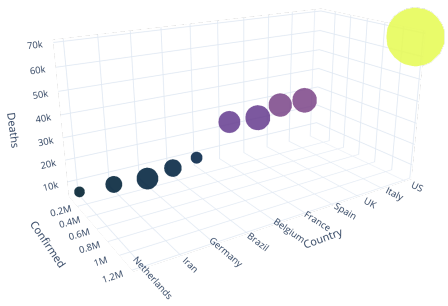

Figure 11: Worst hit countries in 3D

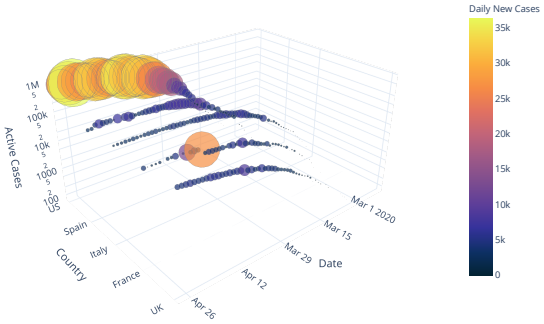

Figure 12: Daily New Cases in 3D

Reflections

As the reader might be aware of, most of the deepest insight and greatest visualizations are already created at the time of writing this paper. ABC News and Reuters Graphics have already created what most would declare as eye candy, yet informative and functional.

Throughout this visualization project, there has been challenge of feeling inferior when comparing my results and findings to the endless amount of great visualizations already made by others. Even though I many times considered changing my project to another topic, I am glad i saw it through to the end. The dataset provided by Johns Hopkins University [1], was very giving to be working with. Having it being uploaded daily, was greatly motivating.

As for design aspects, in the figures applicable, I have attempted to address colorblindness. More specifically, the red-green and blue-yellow blindness has been avoided in the choropleth map, scatter map and 3D plots. An attempt to address this completely were made, but I found it infeasible given the range of data and the aspects that I tried to communicate. A greater effort here could be made, and a completely thought through color palette for the whole project would be nothing less than neat.

For the data, I was quite happy with my main source of data. Regardless, looking into other types of data such as GDP data for countries, pollution, airplane flights or other complex data sources that could unravel some interesting insight and correlations could be something left for *future work*.

Familiarizing myself with Plotly has been quite rewarding, yet it is somewhat limited in modularity, and in many cases Pyplot from Matplotlib would be a much more suitable tool. Another interesting tool to be investigated is AltAir, espescially having the chance to add sizing of bubbles to the legend (as seen here).

With all being said and done, I am more than satisfied with my results, and I am under the impression that many aspects of the data has been shown throughout this paper. Comparisons, timeseries, geolocalization and different scales to display numerical data is some of the key points for this project.

Once again, I would like to refer to the GitHub Repository for this project, where all the code used to create these visualizations can be found.

References

[1] John Hopkins University. *COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University*. URL: https://github.com/CSSEGISandData/COVID-19 (visited on 05/01/2020).

[2] Wikipedia. *2002-2004 SARS outbreak*. URL: https://en.wikipedia.org/wiki/2002-2004_SARS_outbreak (visited on 05/05/2020).

[3] Centers for Disease Control and Prevention. *20014-20016 Ebola Outbreak*. URL: https://www.cdc.gov/vhf/ebola/history/2014-2016-outbreak/index.html (visited on 05/05/2020).

Appendices

A Figures

	Province/State	Country/Region	Lat	Long	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	...	4/24/20	4/25/20	4/26/20	4/27/20	4/28/20	4/29/20	4/30/20	5/1/20	5/2/20	5/3/20
0	NaN	Afghanistan	33.000000	65.000000	0	0	0	0	0	0	...	188	188	207	220	228	252	260	310	331	345
1	NaN	Albania	41.153300	20.168300	0	0	0	0	0	0	...	394	403	410	422	431	455	470	488	519	531
2	NaN	Algeria	28.033900	1.659600	0	0	0	0	0	0	...	1408	1479	1508	1558	1651	1702	1779	1821	1872	1936
3	NaN	Andorra	42.506300	1.521800	0	0	0	0	0	0	...	344	344	344	385	398	423	468	468	472	493
4	NaN	Angola	-11.202700	17.873900	0	0	0	0	0	0	...	6	6	6	6	6	7	7	11	11	11
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
247	NaN	Western Sahara	24.215500	-12.885800	0	0	0	0	0	0	...	5	5	5	5	5	5	5	5	5	5
248	NaN	Sao Tome and Principe	0.186360	6.613081	0	0	0	0	0	0	...	0	0	0	0	4	4	4	4	4	4
249	NaN	Yemen	15.552727	48.516388	0	0	0	0	0	0	...	1	1	1	1	1	1	1	1	1	1
250	NaN	Comoros	-11.645500	43.333300	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
251	NaN	Tajikistan	38.861034	71.276093	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

Figure 13: Dataframe for Recovered Cases

	Province/State	Country/Region	Lat	Long	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20	...	4/24/20	4/25/20	4/26/20	4/27/20	4/28/20	4/29/20	4/30/20	5/1/20	5/2/20	5/3/20
0	NaN	Afghanistan	33.000000	65.000000	0	0	0	0	0	0	...	43	47	50	57	58	60	64	68	72	85
1	NaN	Albania	41.153300	20.168300	0	0	0	0	0	0	...	27	27	28	28	30	30	31	31	31	31
2	NaN	Algeria	28.033900	1.659600	0	0	0	0	0	0	...	415	419	425	432	437	444	450	453	459	463
3	NaN	Andorra	42.506300	1.521800	0	0	0	0	0	0	...	40	40	40	40	41	42	42	43	44	45
4	NaN	Angola	-11.202700	17.873900	0	0	0	0	0	0	...	2	2	2	2	2	2	2	2	2	2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
261	NaN	Western Sahara	24.215500	-12.885800	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
262	NaN	Sao Tome and Principe	0.186360	6.613081	0	0	0	0	0	0	...	0	0	0	0	0	0	0	1	1	1
263	NaN	Yemen	15.552727	48.516388	0	0	0	0	0	0	...	0	0	0	0	0	0	2	2	2	2
264	NaN	Comoros	-11.645500	43.333300	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
265	NaN	Tajikistan	38.861034	71.276093	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	2	2

Figure 14: Dataframe for Death Cases

	Date	Country	Confirmed	Recovered	Deaths	Active	New Cases	New Recovered	New Deaths
0	2020-01-22	Afghanistan	0	0.0	0	0.0	0	0	0
187	2020-01-23	Afghanistan	0	0.0	0	0.0	0	0	0
374	2020-01-24	Afghanistan	0	0.0	0	0.0	0	0	0
561	2020-01-25	Afghanistan	0	0.0	0	0.0	0	0	0
748	2020-01-26	Afghanistan	0	0.0	0	0.0	0	0	0
...	...	...	...	...	...	...	...	...	...
18699	2020-04-30	Zimbabwe	40	5.0	4	31.0	8	0	0
18886	2020-05-01	Zimbabwe	40	5.0	4	31.0	0	0	0
19073	2020-05-02	Zimbabwe	34	5.0	4	25.0	0	0	0
19260	2020-05-03	Zimbabwe	34	5.0	4	25.0	0	0	0
19447	2020-05-04	Zimbabwe	34	5.0	4	25.0	0	0	0

Figure 15: Resulting Dataframe