CD 732: Data Visualization
Report for Datathon-3

Swasti Shreya Mishra (IMT2017043)
International Institute of
Information Technology Bangalore
Email: SwastiShreya.Mishra@iiitb.org

1. Introduction

We are given the tabular datasets published by the World Health Organization for COVID-19 cases. These include State/Country/Reign-wise time series data (over a period of 246 days starting from January 2020) for confirmed cases, recovered cases and deaths.
Our main aim is to effectively visualize the above data as a network to find underlying patterns or communities formed.

2. Methods

The data is mainly visualized using *plotly* and *networkx*, along with some additional matplotlib packages like *basemap* to visualize the world map.
For the warm-up, I experimented with Gephi as well as networkx. The reasons behind choosing networkx are:

- Gephi needs the data to be passed into the software in csv format. Whereas in networkx, we can update our graph using a few lines of code, without having to explicitly generate different files for visualizing different graphs.
- Gephi seems to be better for graph visualization but doesn't allow a lot of graph manipulation. Whereas, networkx might need supporting libraries to visualize the graph better, but it allows a lot more flexibility and can be integrated with other python packages.

2.1. Data Pre-processing

The raw data has the time series cases for different regions. This data is aggregated based on the countries they belong to. Now, we have a country-wise time series data which can be used to find correlation or similarity of progression of COVID-19 between different countries.

2.2. Correlation Network

We have a time series dataset for various countries of the world. For this data, we can directly derive a correlation matrix which will help us determine the correlation scores of progression of deaths/confirmed cases/recoveries between two different countries. The negatively correlated countries are ignored here. The *circular layout* is chosen so that all the countries(nodes) can be visualized effectively.

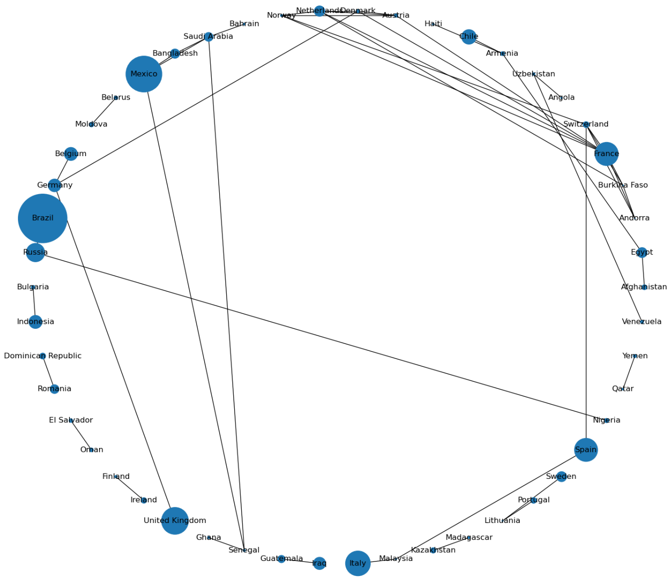

Figure 1. Correlation network of countries with correlation score > 0.998 for cumulative deaths (matplotlib)

2.2.1. Network using matplotlib. In the figure 1, the edges correspond to all countries that have a correlation score > 0.998. This means that the progression of deaths due to COVID-19 is highly correlated in these countries. The relative node sizes correspond to the average deaths per country. Therefore, the bigger the node, the greater the number of deaths in the country due to COVID-19.

2.2.2. Network using plotly. The same network graph as above is visualized using *plotly*, as it gives more flexibility to navigate in the network space. An example is shown in figure 2. We can hover over the nodes to find out more information about the nodes. Also, the color of the nodes correspond to the average number of deaths. Compared to

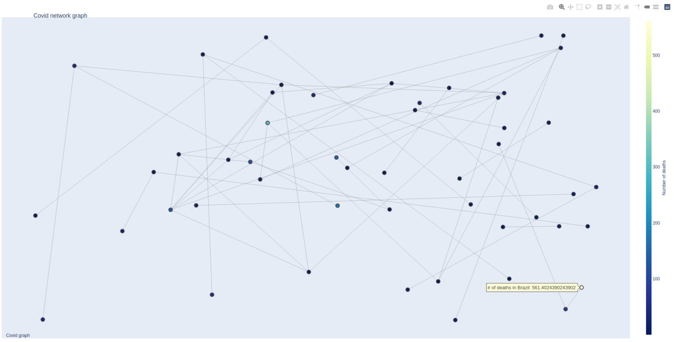

Figure 2. Correlation network of countries with correlation score > 0.998 for cumulative deaths (plotly)

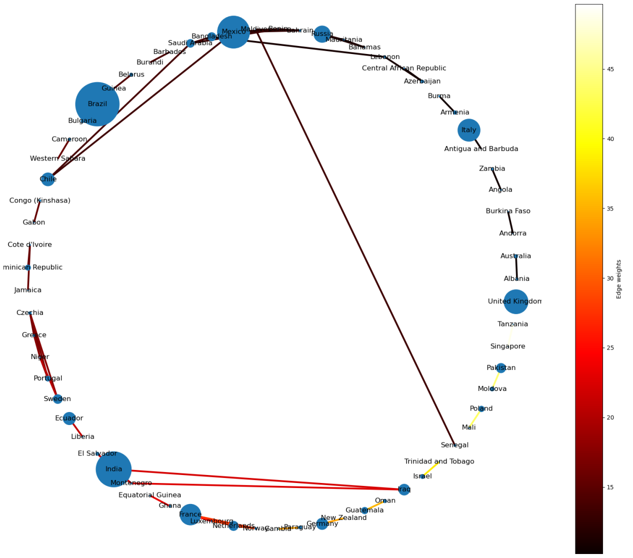

Figure 3. Average time of death network with edge weight between [10,50]

the previous visualization, here color captures the average number of deaths and we remove the notion of node size from the plot. This makes the plot neater as well as provides better optimization of the space compared to the circular layout.

2.3. Weighted Average Time Network

In order to transform a time series data into a graph, various aggregation techniques can be used. If we look at every node as a country, we can connect them on the basis of approximate time taken for the explosion of COVID-19 cases, this is as per the source [1].

We calculate a weighted average of time taken for a country to incur the total number of deaths till the last date in the dataset. The function used is as follows:

$$deaths(i) = \text{Cumulative death count of } i^{th} \text{ day}$$
$$average\_time(n) = \frac{\sum_{i=1}^{i=n} deaths(i)*i}{\sum_{i=1}^{i=n} deaths(i)}$$

Here, n is the n^{th} day until we are calculating the average time of cases.

Now, edge weights for country1 (say u) and country2 (say v) can be computed using:

weight(u,v) = 1/(u[average_time(n)] - v[average_time(n)])

The above edge gives us the information that, if the progression of COVID-19 was similar in two countries, they have higher weights.

Note: We add 0.001 in the denominator in the actual implementation so as to avoid division by zero.

The figure 3 refers to the average time of deaths network, computed using the above equations. The edge weights are displayed only if it lies between the range [10, 50]. The relative node size corresponds to the average number of deaths for the countries.

The reason behind choosing to visualize deaths is that, we cannot rely on the number of confirmed cases for every country. For many countries, testing the entire population is not possible. Also, we cannot determine exact the number of infected people since COVID-19 is characterized by a great number of asymptomatic people. When someone dies, the tests tend to be executed with a higher probability than for those who are infected with no symptoms. This idea is adopted as per source [1].

2.4. Geographically aligned Network

Even though we can infer similarity of COVID-19 progression between countries, it is very difficult to analyse the countries without them being in their correct geographical location. We have the latitude and longitude coordinates for every region in the data. For the sake of simplicity, we take one value of (latitude, longitude) corresponding to a country (even though we have the (latitude, longitude) pairs for all the regions of the country). Here, we are visualizing the same graphs as above along with *basemap*. The inspiration to do so has been drawn from this source [2].

The figure 4 corresponds to the average time of death network with edge weight between [10,100]. The relative node size corresponds to the average number of deaths for the countries. Here the country labels are only visible if the average number of deaths in 246 days exceed 5. This is done to avoid crowding of text in the visualization.

3. Inferences

3.1. Comparison of deaths and recoveries

The figure 5 and figure 6 can be used to compare the progression of COVID-19. Figure 5 corresponds to the top 10 countries with maximum deaths and figure 6 to top 10 with maximum number of recoveries respectively. The relative scaling of the node size is kept same across the plots. This is to show that, the number of recoveries largely exceed the number of deaths. The recovery rate is higher in India and US, even though they have a high death rate.

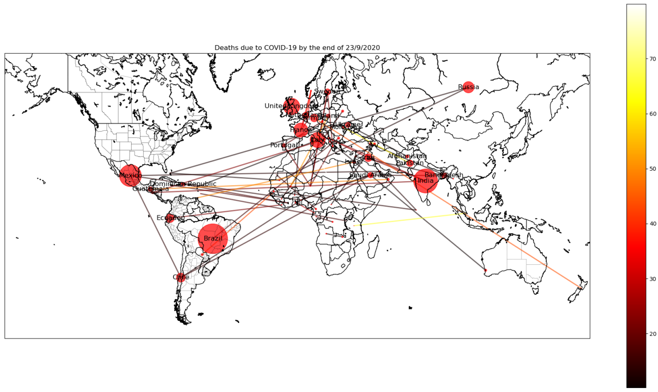

Figure 4. Average time of death network with edge weight between [10,100]

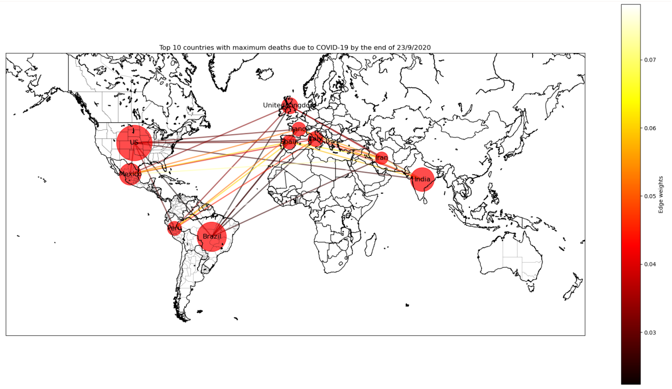
Top 10 countries with maximum deaths due to COVID-19 by the end of 23/9/2020
United Kingdom
US
Mexico
Iran
India
Peru
Brazil
Edge weights
0.07
0.06
0.05
0.04
0.03

Figure 5. Top 10 countries with maximum death (average time of death network)

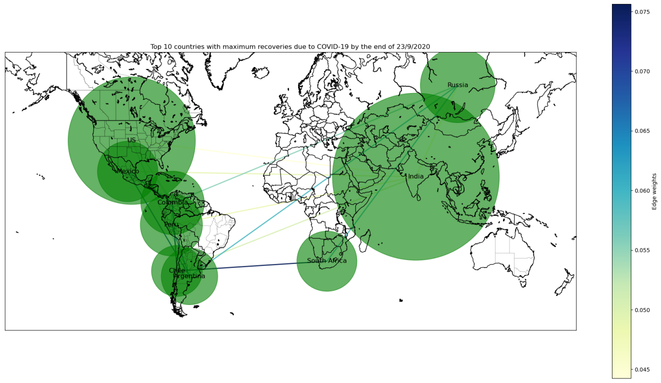

Figure 6. Top 10 countries with maximum recoveries (average time of recovery network)

Whereas, European nations like United Kingdom lag behind in the number of recoveries, even though the death rate due to COVID-19 is pretty high.

3.2. Clusters in the middle phase of the spread

Another observation that can be drawn from the figure 7 is that, COVID-19 hadn't spread much in India by then. The European nations were very widely affected during that time. This figure 8 is a zoomed in version of the figure 7 in the Europe region. It also emerges as a cluster with rapid spread of the virus. It spread at nearly the same time which can be observed from its edge weights.

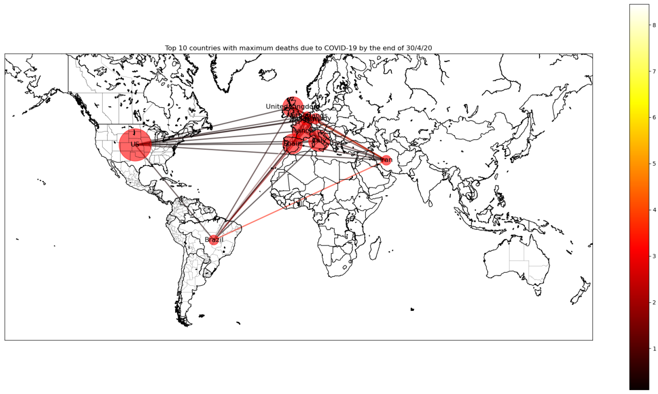
Top 10 countries with maximum deaths due to COVID-19 by the end of 30/4/20
United Kingdom
US
Iran
Brazil
Edge weights
8
7
6
5
4
3
2
1

Figure 7. Top 10 countries with maximum deaths (average time of recovery network) by the end of April

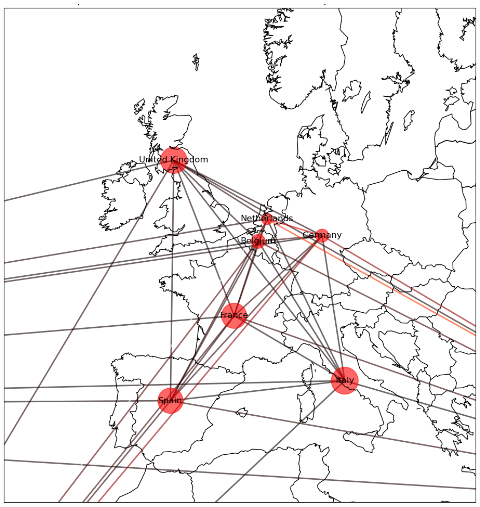

Figure 8. Zoomed European region of figure 7

3.3. Origin of the virus

From all these visualizations, one major country seems to be missing, i.e. China. All of us are well aware of the fact that this was the origin of virus. But, since we have the data from January to September, the cases per day in other countries such as US and India seem to dominate. This was because, by then, the COVID-19 cases curve had flattened out in China. So, we now look at the data only till 10 February 2020. When we visualize top 10 countries, only China is visible as a cluster and therefore, we know that it is the origin. This is shown in figure 9.

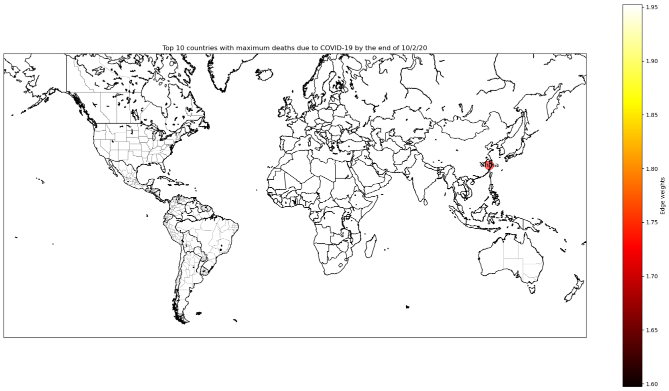
Top 10 countries with maximum deaths due to COVID-19 by the end of 10/2/20
China
1.95
1.90
1.85
1.80
1.75
1.70
1.65
1.60
Edge weights

Figure 9. Maximum deaths due to COVID-19 as of 10 February 2020

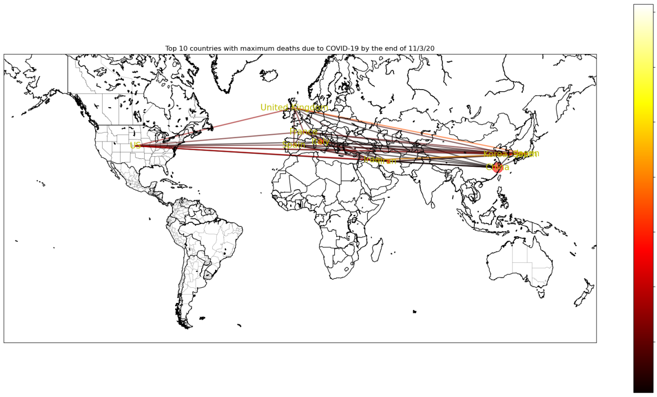
Top 10 countries with maximum deaths due to COVID-19 by the end of 11/3/20
United Kingdom
China
7
6
5
4
3
2
1
Edge weights

Figure 10. Top 10 countries with maximum deaths due to COVID-19 as of 11 March 2020

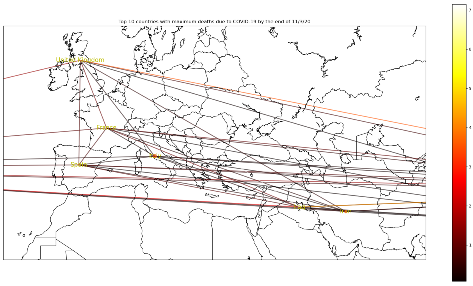
Top 10 countries with maximum deaths due to COVID-19 by the end of 11/3/20
Edge weights

Figure 11. Zoomed in European region of figure 10

3.4. The cluster of origin

This visualization of top 10 COVID-19 hit countries by the end of 11 March 2020 can be seen in figure 10. This shows the cluster of origin. The lighter edge colors show that the spread/transmission in these countries happened around the same time. We can see the virus being spread to US and European nations from China. The figure 11 is the zoomed in European region and figure 12 is the zoomed in Asian region respectively for the above mentioned plot.

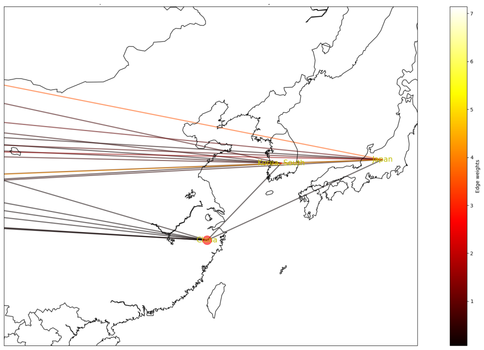

Figure 12. Zoomed in Asian region of figure 10

4. Conclusion

The possibilities of interpretations of the data we have visualized is very large. I have only been able to look at a few of them but they provide us with a lot of information. I did try out *greedy_modularity_communities* from *networkx.algorithms.community.modularity_max* but couldn't visualize the communities effectively. The cluster of origin seemed to be most intuitive and therefore, I stuck with that. Working on this datathon made me realise the importance of data visualization. During these times of pandemic, it is very crucial to effectively visualise data so that we can make meaningful inferences from them and take action accordingly.
Note: I have used *time_series_covid_19_deaths.csv* and *time_series_covid_19_recoveries.csv* for visualization of the above results.

References

[1] Graph theory: Covid-19 spreading clustering, Jun 2020.

[2] Tuan Doan Nguyen. Catching that flight: Visualizing social network with networkx and basemap, Jun 2018.