

Reach of the Pandemic - Report

1.Introduction & Business Problem :

Problem Description:

2020 has been a nightmare to the entire due to an epidemic, Corona. There are hundreds of casualties daily due to this virus. Amidst this, everyone will be interested in knowing the statistics of their country, how other countries are handling the situation and many other details. So, the idea is to visualize each country's corona details in more appealing manner and try to cluster them using a Machine Learning Algorithm to better know about similarity in countries when it comes to handling the situation.

Success Criteria:

The success criteria of the project will be visualization of Corona tolls across the globe and clustering them according to their handling capability which in this project would be **more Recovery Rate and less Pending percentage**. The main idea behind these calculation is a nation which has very high Recovery rate and less cases pending to be handled eventually that country is better at handling the situation. Explanation about how Recovery Rate and Pending percentage are calculated will be provided later in the report.





















2. Data :

Country wise Corona details are required to map them into clusters and we need location for each country to visualize them.

We will be using the below datasets for analysis.

Data 1 : Fetching of information about corona for each country is done from Wikipedia which is available from the link below.

https://en.wikipedia.org/wiki/Template:2019%E2%80%9320_coronavirus_pandemic_data

[show all]					
2019–20 coronavirus pandemic by country and territory					
<div><ul style="list-style-type: none"> United States^[e] Spain^[f] Italy^[g] Germany^[h] France^[i] United Kingdom^[j] China^[k] Iran^[l] Turkey Belgium^[m]</div>					
Countries and territories ^[a]	Cases ^[b]	Deaths ^[c]	Recov. ^[d]	Ref.	
234	2,101,164	140,773	532,830	[2]	
 United States ^[e]	657,720	33,460	53,322	[11]	
 Spain ^[f]	182,816	19,130	74,797	[14]	
 Italy ^[g]	168,941	22,170	40,164	[17]	
 Germany ^[h]	135,549	3,850	66,269	[18] [19]	
 France ^[i]	108,847	17,920	32,812	[21] [22]	
 United Kingdom ^[j]	103,093	13,729	—	[24]	
 China ^[k]	82,341	3,342	77,892	[25]	
 Iran ^[l]	77,995	4,869	52,229	[31]	
 Turkey	74,193	1,643	7,089	[32]	
 Belgium ^[m]	34,809	4,857	7,526	[34]	

Data 2 : The other data location details of each nation which is available in the following.

https://developers.google.com/public-data/docs/canonical/countries_csv

countries.csv

country	latitude	longitude	name
AD	42.546245	1.601554	Andorra
AE	23.424076	53.847818	United Arab Emirates
AF	33.93911	67.709953	Afghanistan
AG	17.060816	-61.796428	Antigua and Barbuda
AI	18.220554	-63.068615	Anguilla
AL	41.153332	20.168331	Albania
AM	40.069099	45.038189	Armenia
AN	12.226079	-69.060087	Netherlands Antilles
AO	-11.202692	17.873887	Angola
AQ	-75.250973	-0.071389	Antarctica
AR	-38.416097	-63.616672	Argentina
AS	-14.270972	-170.132217	American Samoa
AT	47.516231	14.550072	Austria

3.Methodology :

Business Understanding .:

Our Main goal is visualize corona details for each nation.

Exploratory Data Analysis :

Data 1- Load Corona Details Data from Wikipedia

1. Make use of requests and get response in form of text from wiki table.
2. Using BeautifulSoup instance fetch table from response previously.
3. This dataframe contains many columns flag, Countries, References, Deaths, Confirmed Cases etc., Drop all unnecessary columns clean the data.
 - Remove Null value rows.
 - Remove some non-informative rows
 - Rename some countries so that it will be helpful to merge later in country name.

The columns left after cleaning would be “Country”, “Confirmed Cases”, “Deaths” and “Recovered” as shown below.

Corona Details Data After Cleaning

```
clean_df.head(10)
```

[11]:	Country	Confirmed_Cases	Deaths	Recovered
0	United States	684869	35585	56437
1	Spain	188093	19613	72963
2	Italy	172434	22745	42727
3	Germany	138369	4105	66500
4	France	109252	18681	34420
5	United Kingdom	108692	14576	0
6	China	82692	4632	76979
7	Iran	79494	4958	54064
8	Turkey	78546	1769	8631
9	Belgium	36138	5163	7961

We will be adding new columns for analytic purposes. The new columns are as follows

- 1.) Live Cases = Confirmed Cases – Deaths – Recovered.
- 2.) Verdict Cases = Deaths + Recovered
- 3.) Recovery Rate = Recovered*100/Verdict Cases
- 4.) Pending Percentage = (Live Cases *100)/(Confirmed Cases)
- 5.) Handling Value = Recovery Rate – Pending Percentage.

Using Handling value, if it is more then we can say that the particular country is handling Corona elegantly than others.

The Following picture is sorted in descending order of Handling Value and Countries that crossed more than 10,000 cases.

]:	Country	Confirmed_Cases	Deaths	Recovered	Live Cases	Verdict Cases	Recovery Rate	Pending percentage	Handling value
6	China	82692	4632	76979	1081	81611	94.324295	1.307261	93.017034
22	South Korea	10635	230	7829	2576	8059	97.146048	24.221909	72.924139
7	Iran	79494	4958	54064	20472	59022	91.599742	25.752887	65.846855
16	Austria	14553	431	9704	4418	10135	95.747410	30.358002	65.389408
14	Switzerland	26928	1325	16400	9203	17725	92.524683	34.176322	58.348361
3	Germany	138498	4194	66500	67804	70694	94.067389	48.956664	45.110725
19	Peru	13489	300	6120	7069	6420	95.327103	52.405664	42.921439
12	Brazil	30961	1956	14026	14979	15982	87.761231	48.380220	39.381011
1	Spain	188093	19613	72963	95517	92576	78.814163	50.781794	28.032369
11	Canada	31642	1310	10328	20004	11638	88.743770	63.219771	25.523999

To Confirm that our analysis is going in right direction all those countries which are actually handling it better than other countries are listed, like China, Switzerland, Iran, South Korea etc.,

Data 2- Load data from Countries.csv

1. Download the csv file locally as locations won't change.
2. Load it into a data frame using load_csv method of pandas library.
3. Remove un-used columns and rename required columns helpful for merging.
4. After cleaning, the data looks something like in following picture.

```
countries_csv.head()
```

1]:	latitude	longitude	Country
0	42.546245	1.601554	Andorra
1	23.424076	53.847818	United Arab Emirates
2	33.939110	67.709953	Afghanistan
3	17.060816	-61.796428	Antigua and Barbuda
4	18.220554	-63.068615	Anguilla

Merging the Data

Now, merging of the above both data frames is done to get latitude and longitude columns beside all the columns present in Data 1.

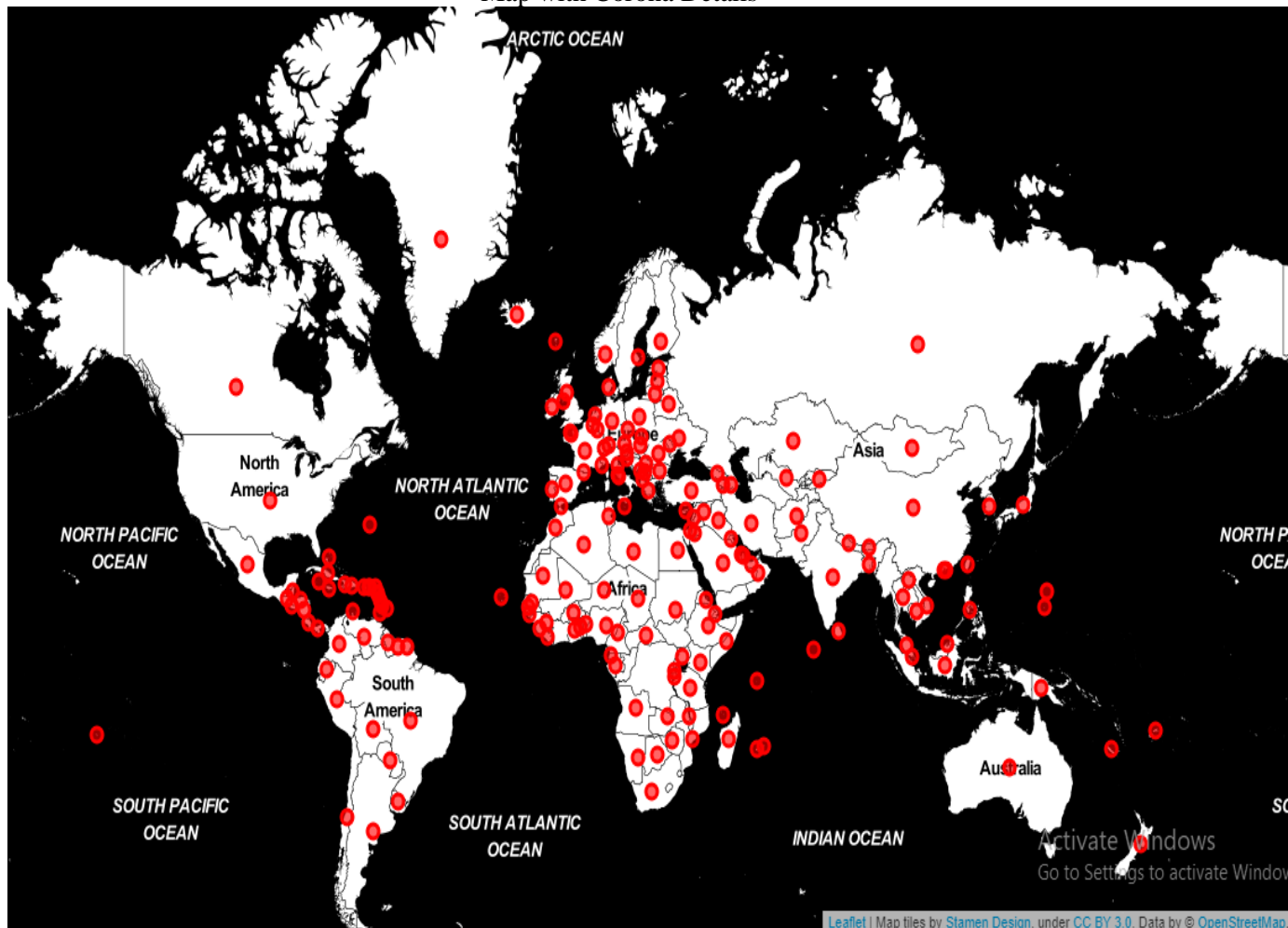
The following picture will give a better visual of the data we have.

	Country	Confirmed_Cases	Deaths	Recovered	Live Cases	Verdicted Cases	Recovery Rate	Pending percentage	Handling value	latitude	longitude
0	Greenland	11	0	11	0	11	100.000000	0.000000	100.000000	71.706936	-42.604303
1	China	82341	3342	77892	1107	81234	95.885959	1.344409	94.541550	35.861660	104.195397
2	Faroe Islands	184	0	166	18	166	100.000000	9.782609	90.217391	61.892635	-6.911806
3	U.S. Virgin Islands	51	1	43	7	44	97.727273	13.725490	84.001783	18.335765	-64.896335
4	Cambodia	122	0	98	24	98	100.000000	19.672131	80.327869	12.565679	104.990963
5	Gibraltar	131	0	105	26	105	100.000000	19.847328	80.152672	36.137741	-5.345374
6	Brunei	136	1	108	27	109	99.082569	19.852941	79.229628	4.535277	114.727669
7	South Korea	10613	229	7757	2627	7986	97.132482	24.752662	72.379820	35.907757	127.766922
8	New Zealand	1084	9	770	305	779	98.844673	28.136531	70.708141	-40.900557	174.885971
9	Maldives	23	0	16	7	16	100.000000	30.434783	69.565217	3.202778	73.220680

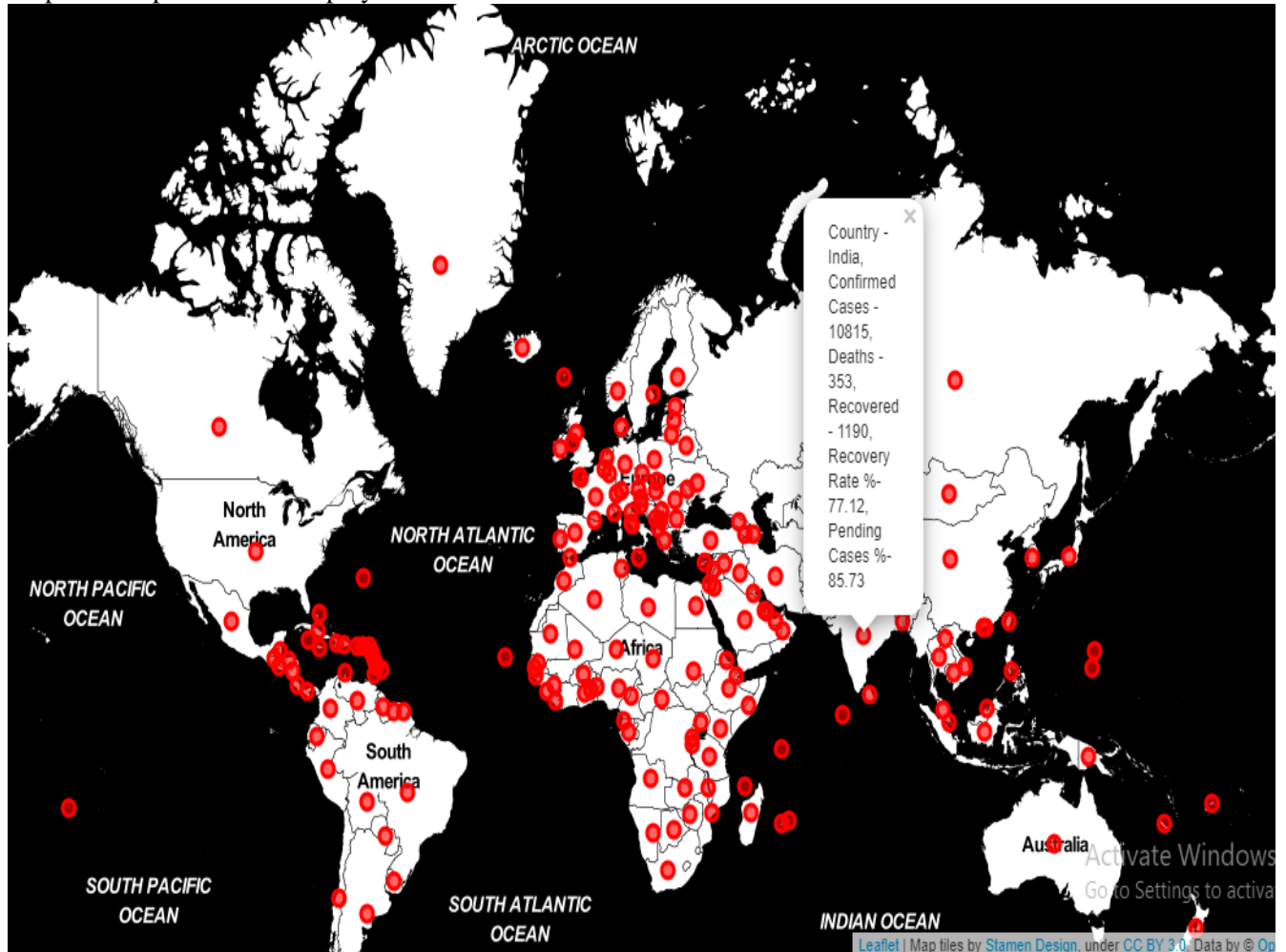
Visualizing of Data on Map

As this is nation wise data we can view this on a world map. To achieve that we use Folium and Circle Marker for each country. The map will also have details displayed once we click on a country as shown in the following figure.

Map with Corona Details



Map with Popout feature displayed once India is selected.

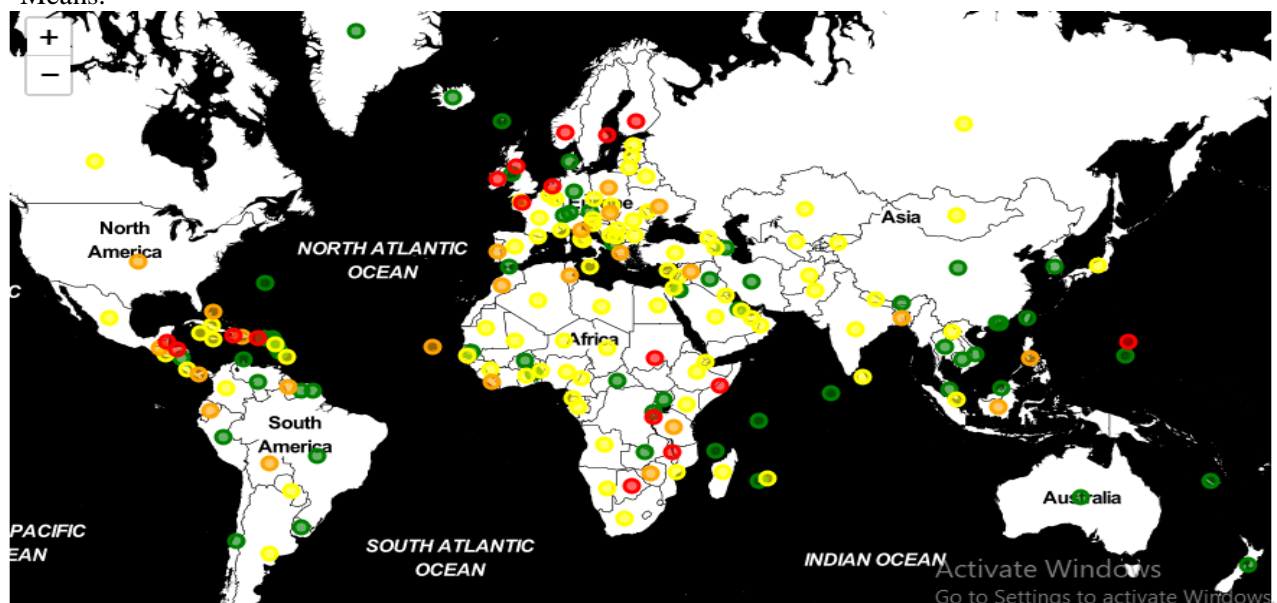


4.RESULTS :

K-Means clustering based on Handling Value Column:

To cluster the data into 4 clusters we used the K-Means clustering Algorithm. *K-means* clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean. It uses iterative refinement approach.

In the below Map Visualization, we can see the different types of clusters created by using K-Means.



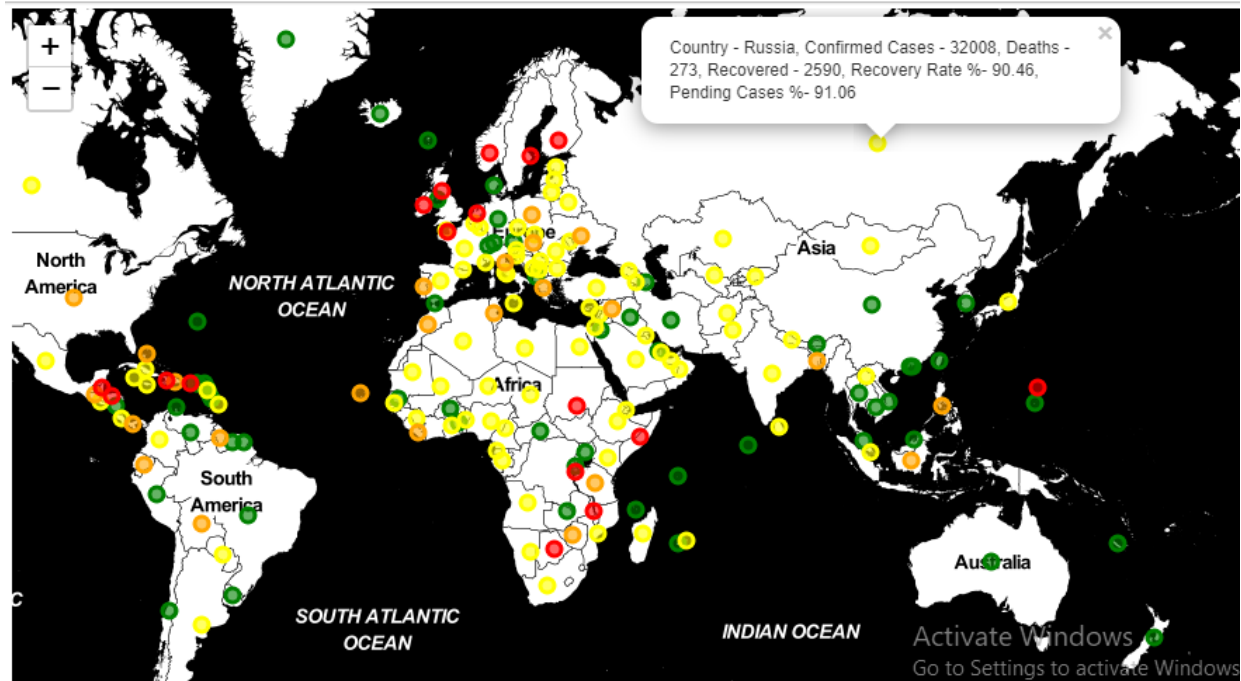
Green cluster - Recovery Rate very high and pending percentage low.

Yellow cluster- Recovery Rate high and pending percentage low.

Orange cluster- Recovery Rate high and pending percentage is high.

Red cluster - Almost no Recovery Rate and pending percentage is very high.

Map with Pop out feature after clustering



5. DISCUSSION:

1. The analysis can be extended to each state in a country and also to each city in a state which will give us a total and detailed view of what is happening in and around the world.
2. Dividing them into more clusters might also be one option

6. CONCLUSION:

This analysis is performed on limited data and could achieve some superficial result. This may be right or may be wrong. But if good amount of data is available there is scope to come up with better results.