Delhi's Current Health Infrastructure Report (For COVID 19)

Made by: Akshay Sharma

Introduction:

The outbreak of COVID-19 in different parts of the world is a major concern for all the administrative units of the respective countries. Delhi is also facing this very tough task for controlling the virus outbreak and has managed its growth rate through some strict measures. Although Delhi has the best health infrastructure in India One problem remains intact in the state and i.e High population density and while with the help of Central govt and other central forces, Delhi govt made the biggest isolation center in the world. Centralizing the isolation centers creates some issues for the general public:

- Accessibility issues for the large chunk of the populations living at the extreme corners of the state.
- Transportation Issues: those who are transporting these infected people will need PPE and other protections and there is always a possibility of infection for them.
- Social Distancing Problem: Centralising the testing and isolation centers can cause huge problems for a state like Delhi.

So Decentralising the Testing and isolation centers will surely help to tackle this pandemic more efficiently. And This Work can be used by both Govt Agencies as well as private firms.

Data:

 I used the Wikipedia website to extract the information for the Delhi Region subdivisions and other demographics such as Area in square km and population in each of the region was also available on that page.

I then used web scraping library BeautifulSoup to extract the useful information from that page. And data was then fed to the coded functions to generate the Pandas Dataframe. I used the District level data from the Dataframe to obtain the longitude and latitude data from the Geopy Library function Nominatim. And then that data was merged into the original data frame.

Out[81]:

	District	Area SqKM	Population	Latitude	Longitude
0	Central Delhi	23	578671	28.698548	77.219391
1	North Delhi	59	883418	28.613895	77.209006
2	South Delhi	249	2733752	28.485169	77.196380
3	East Delhi	49	1707725	28.620477	77.309181
4	North East Delhi	56	2240749	28.723308	77.266857
5	South West Delhi	421	2292363	28.586448	76.979153
6	New Delhi	35	133713	28.613895	77.209006
7	North West Delhi	2344	3651261	28.613895	77.209006
8	West Delhi	131	2531583	28.647952	77.085565

Figure: District Dataframe with other demographics.

Exploratory Data Analysis:

```
In [82]: import matplotlib.pyplot as plt
         dfp_a=df["Population"]/df["Area SqKM"]
         dfp_a.index=df["District"]
         dfp_a.plot(kind="pie",figsize=(10,15))
         plt.xlabel("PIE CHART FOR POPULATION PER AREA KM^2")
Out[82]: Text(0.5, 0, 'PIE CHART FOR POPULATION PER AREA KM^2')
```
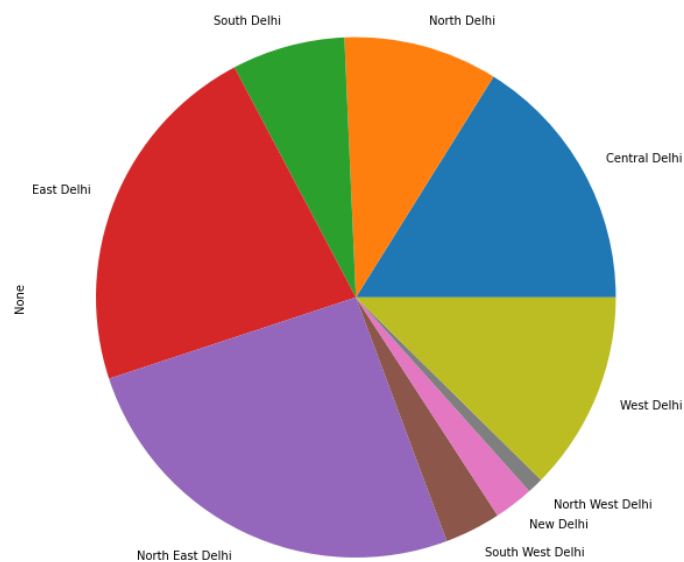

Figure: How Delhi's area is distributed with respect to the population numbers.

I then performed an exploratory data analysis on the above data frame to gain some small insights into the dataset. And as we can infer from the above pie chart, is that the area is not evenly distributed and this can be a problem while setting up the isolation and testing centers. I then also checked if the population has some correlation with the area.

```
In [83]: print("Correlation B/w Population and Area :",df[["Population","Area SqKM"]].corr())

          Correlation B/w Population and Area :            Population  Area SqKM
          Population    1.000000   0.666108
          Area SqKM     0.666108   1.000000
```

Figure: Correlation strength.

As the correlation strength is quite low so there is no need of performing the regression analysis for this data set.

Methodology and Results:

As we can see from the above exploratory data analysis, the data has very little correlation and is very random so there was no need of applying regression models to the problem will not give not us any fruitful results.

So I decided to generate the data for the nearby hospitals, medical suppliers, medical labs, and emergency centers and to see how they're clustered around the city. Local hospitals, labs can provide help to setup Isolation camps and test centers.

Out[84]:

	name	categories	lat	lng
0	Primus super speciality hospital	Hospital	28.592961	77.181430
1	Dr. Ram Manohar Lohia Hospital	Hospital	28.624654	77.200609
2	Sir Gangaram Hospital	सर गंगाराम अस्पताल	Hospital	28.638601	77.188884
3	Loknayak Hospital	Hospital	28.638303	77.238829
4	B.L kapoor memorial hospital	Hospital	28.643368	77.179899
5	BLK Super Specialty Hospital	Hospital	28.643392	77.179582
6	Safdarjung hospital	Hospital	28.566744	77.204870
7	St. Stephen's Hospital	Hospital	28.666454	77.214489
8	Aashlok Hospital	Hospital	28.566571	77.198902
9	AIIMS	Medical School	28.567922	77.209255
10	Moolchand Hospital	Hospital	28.564782	77.243863
11	Moolchand medcity	Hospital	28.566345	77.235344

Figure: Dataframe generated for Nearby Hospitals.

Out[85]:

	name	categories	lat	lng
0	Draeger India Private Limited	Medical Supply Store	28.565078	77.188793
1	Draeger India Private Limited	Medical Supply Store	28.565068	77.188774
2	Draeger India Private Limited	Medical Supply Store	28.565032	77.188851
3	Draeger India Private Limited	Medical Supply Store	28.565052	77.188774
4	Modern Times Helpline Pharma	Medical Supply Store	28.645798	77.243843
5	kirpashakti foundation	Medical Supply Store	28.615955	77.080628

Figure: Dataframe generated for Nearby Suppliers

nearby_medicallabs = json_normalize(venues5)

Out[86]:

	name	categories	lat	lng
0	Tomar Foundation Path Lab	Medical Lab	28.616119	77.213180
1	Max Lab	Medical Lab	28.632193	77.175816
2	DNA Forensics Laboratory	Medical Lab	28.580376	77.233345
3	CDH Fine Chemical	Medical Lab	28.654196	77.231569
4	Max Lab	Medical Lab	28.581893	77.251850
5	Max Lab	Medical Lab	28.564223	77.219682
6	Goyal MRI and Diagnostic Centre	Medical Lab	28.564569	77.189384
7	SRL Diagnostics	Medical Lab	28.569694	77.163410
8	Delhi MRI Scan	Medical Lab	28.553604	77.205550
9	Max Lab	Medical Lab	28.564937	77.163562
10	Max Lab	Medical Lab	28.556313	77.243687
11	Dr Lal Path Labs	Medical Lab	28.555286	77.242566
12	dhruv diagnostic centre	Medical Lab	28.674823	77.180210
13	Max Lab	Medical Lab	28.617948	77.284789
14	Max Lab	Medical Lab	28.551641	77.239966
15	Max Lab	Medical Lab	28.559393	77.256915
16	Max Lab	Medical Lab	28.642850	77.281485
17	sure path labs	Medical Lab	28.651361	77.278232
18	Mahajan Imaging Center	Medical Lab	28.546458	77.201400
19	Max Lab	Medical Lab	28.501313	77.182937
20	Clinical Research Lab, DIPSAR	Medical Lab	28.516902	77.221584
21	Max Lab	Medical Lab	28.511332	77.229310
22	Max Lab	Medical Lab	28.514187	77.165188
23	Max Lab	Medical Lab	28.525373	77.215576
24	Dr Mukul's DIAGNOSTIC CLINIC	Medical Lab	28.534832	77.207768
25	Dr Lal Path Labs	Medical Lab	28.518876	77.150334
26	Max Lab	Medical Lab	28.619596	77.304878
27	Max Lab	Medical Lab	28.633455	77.302515
28	Max Lab	Medical Lab	28.606562	77.295687
29	sure path labs p ltd	Medical Lab	28.604277	77.293494
30	dr lal pathlab	Medical Lab	28.639684	77.292705
31	Max Lab	Medical Lab	28.652062	77.314312
32	Pulse Medicare Centre Vaishali Sector 3	Medical Lab	28.640176	77.340085
33	Max Lab	Medical Lab	28.640563	77.340312
34	Max Lab	Medical Lab	28.619707	77.370275
35	SRL Diagnostics	Medical Lab	28.567719	77.324193

Figure: Dataframe generated for the Nearby Medical Labs.

Out[87]:

	name	categories	lat	lng
0	The Healthcare Institute	Emergency Room	28.608788	77.211421
1	Casualty	Emergency Room	28.626181	77.204719
2	Dharma Vira Heart Centre	Emergency Room	28.638523	77.190198
3	Innovarss	Emergency Room	28.656512	77.227321
4	J N Apex Trauma Centre	Emergency Room	28.569038	77.199800
5	Max Healthcare	Hospital	28.527784	77.211669
6	Apollo Clinic	Emergency Room	28.533932	77.148828
7	Yashoda Blood Bank	Emergency Room	28.646559	77.327202
8	BTRC	Emergency Room	28.569952	77.328951
9	Coronary Care Unit	Emergency Room	28.683558	77.310082
10	Surgery Em	Emergency Room	28.685528	77.310207
11	Cancer Healer Center	Emergency Room	28.564698	77.269292
12	Emergency OT, LNJP	Emergency Room	28.639434	77.237128
13	Emergency,LOKNAYAK HOSPITAL	Emergency Room	28.639359	77.236174
14	Pentamed Hospital	Hospital	28.699263	77.191737
15	MGS Hospital	Emergency Room	28.672672	77.136040

Figure: Dataframe generated for the Nearby Emergency Centre

So, the details for the Delhi health infrastructure were extracted using the Foursquare Developers API, and then data was fed into the data frame. I then merged all the data frames for Hospitals, Labs, Supply stores, and Emergency centers into one. and then Longitude and Latitude data from the data frame was then used for generating the city map and to see how all the entries are spread around the city.

Generating Map to Visualise all the Hospitals, Medical Supply Stores, Medical Labs and Emergency Centres in Delhi

```
In [88]: Df_Delhi_Medical_infra = pd.concat([nearby_hospitals, nearby_medicalsupply, nearby_medicallabs,nearby_emergencycentres], ignore_
```

```
In [89]: Df_Delhi_Medical_infra
```

Out[89]:

	name	categories	lat	lng
0	Primus super speciality hospital	Hospital	28.592961	77.181430
1	Dr. Ram Manohar Lohia Hospital	Hospital	28.624654	77.200609
2	Sir Gangaram Hospital	सर गंगाराम अस्पताल	Hospital	28.638601	77.188884
3	Loknayak Hospital	Hospital	28.638303	77.238829
4	B.L kapoor memorial hospital	Hospital	28.643368	77.179899
5	BLK Super Specialty Hospital	Hospital	28.643392	77.179582
6	Safdarjung hospital	Hospital	28.566744	77.204870
7	St. Stephen's Hospital	Hospital	28.666454	77.214489
8	Aashlok Hospital	Hospital	28.566571	77.198902
9	AIIMS	Medical School	28.567922	77.209255
10	Moolchand Hospital	Hospital	28.564782	77.243863
11	Moolchand medcity	Hospital	28.566345	77.235344

Figure: Merging all the Dataframe into one to see the spread of the data.

Then I used the Folium.map function and I passed the Longitude and Latitude Data to the function as the parameters.

```
for lat, lng, name, categories in zip(Df_Delhi_Medical_infra['lat'], Df_Delhi_Medical_infra['lng'],Df_Delhi_Medical_infra['name']
    label = '{}'.format(name)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
        fill_color='blue',
        fill_opacity=0.7,
        parse_html=False).add_to(Delhi_map)

Delhi_map
```
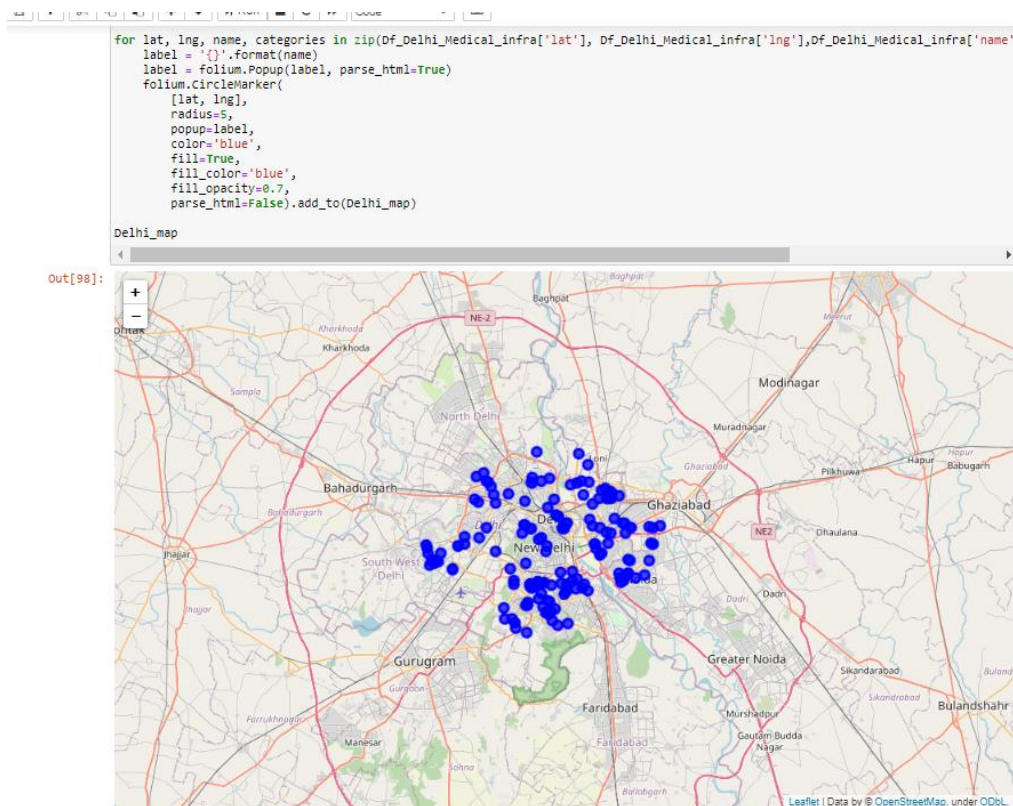
Out[98]:

Figure: Spread of Delhi's Hospitals, Labs, etc.

The same types of maps were generated for the individual data frame of Hospitals, Labs, Supply stores, and Emergency Centers.

```
for lat, lng, name, categories in zip(nearby_hospitals['lat'], nearby_hospitals['lng'],nearby_hospitals['name'], nearby_hospitals
    label = '{}'.format(name)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='red',
        fill=True,
        fill_color='red',
        fill_opacity=0.7,
        parse_html=False).add_to(Delhi_map)

Delhi_map
```
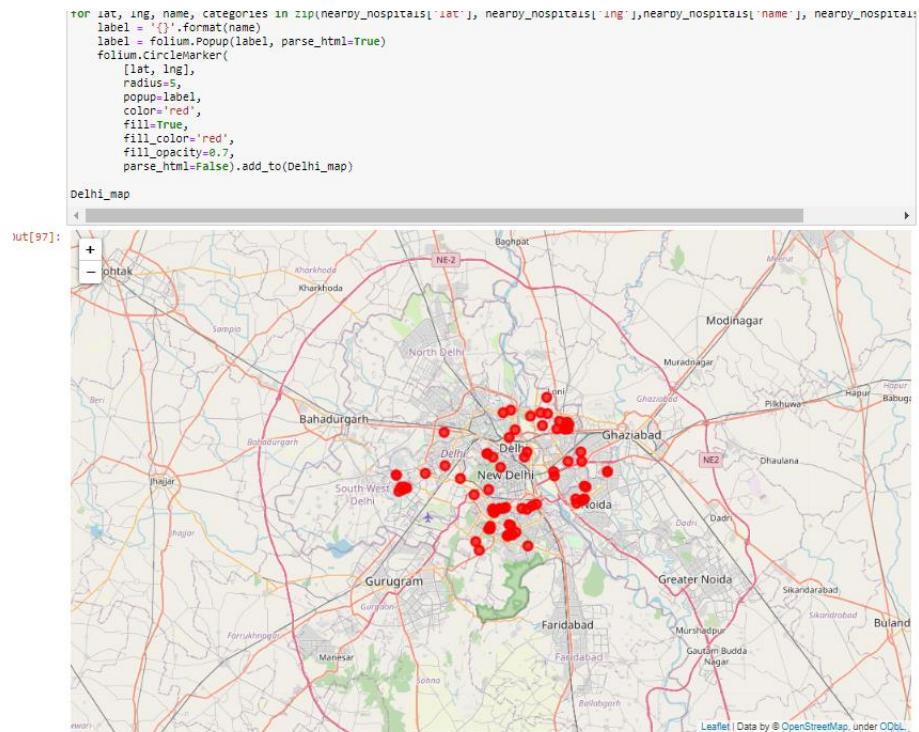

Figure: Spread of Hospitals.

```
In [96]:   Delhi_map = folium.Map(location=[28.7041, 77.1025], zoom_start=10)

           for lat, lng, name, categories in zip(nearby_medicalsupply['lat'], nearby_medicalsupply['lng'],nearby_medicalsupply['name'], nea
               label = '{}'.format(name)
               label = folium.Popup(label, parse_html=True)
               folium.CircleMarker(
                   [lat, lng],
                   radius=5,
                   popup=label,
                   color='green',
                   fill=True,
                   fill_color='green',
                   fill_opacity=0.7,
                   parse_html=False).add_to(Delhi_map)

           Delhi_map
```


Figure: Spread of Medical Suppliers.

```
In [95]:    Delhi_map = folium.Map(location=[28.7041, 77.1025], zoom_start=10)

            for lat, lng, name, categories in zip(nearby_medicallabs['lat'], nearby_medicallabs['lng'],nearby_medicallabs['name'], nearby_med
                label = '{}'.format(name)
                label = folium.Popup(label, parse_html=True)
                folium.CircleMarker(
                    [lat, lng],
                    radius=5,
                    popup=label,
                    color='purple',
                    fill=True,
                    fill_color='purple',
                    fill_opacity=0.7,
                    parse_html=False).add_to(Delhi_map)

            Delhi_map
```


Figure: Spread of Medical Labs

```
Delhi_map = folium.Map(location=[28.7041, 77.1025], zoom_start=10)

for lat, lng, name, categories in zip(nearby_emergencycentres['lat'], nearby_emergencycentres['lng'],nearby_emergencycentres['nar
    label = '{}'.format(name)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='black',
        fill=True,
        fill_color='black',
        fill_opacity=0.7,
        parse_html=False).add_to(Delhi_map)

Delhi_map
```
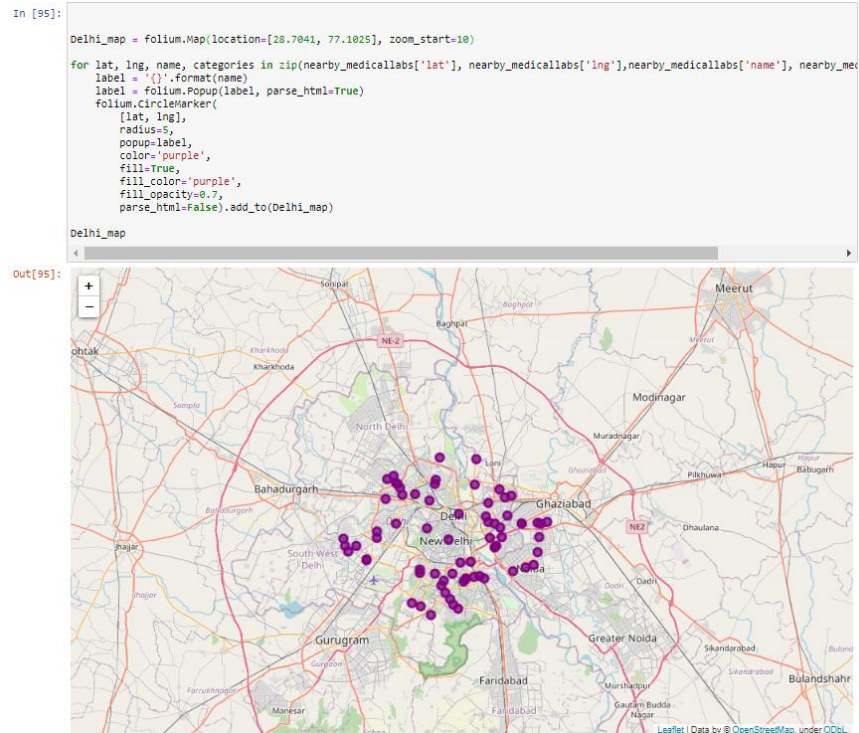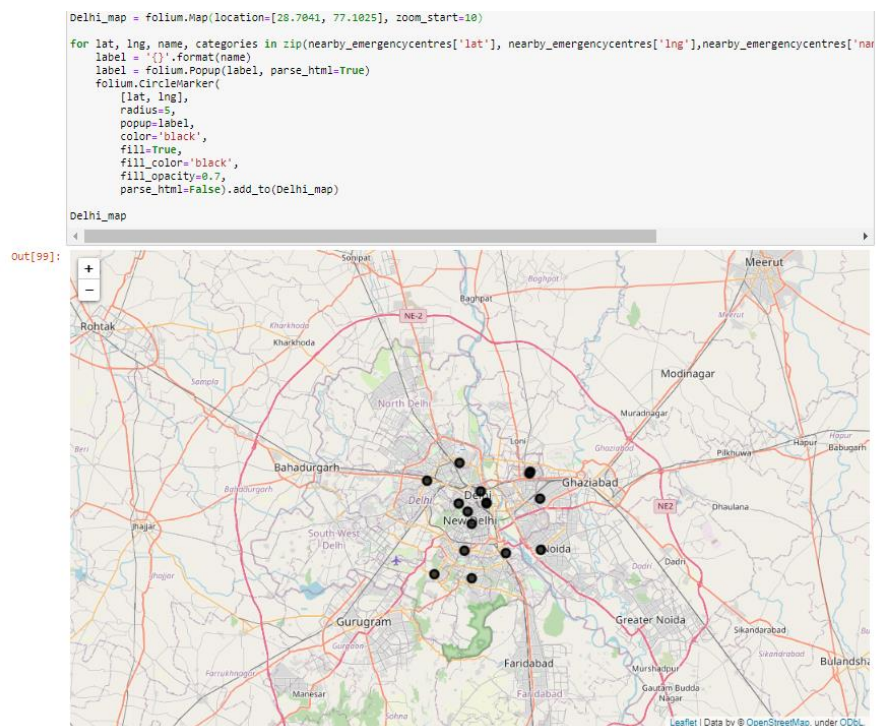

Figure: Spread of Emergency Centers.

As we can infer from the above spreads, the Facilities are not evenly spread throughout the city and are majorly clustered around the center of the city. Ideally, there should be even no hospitals of the district so that they can better serve the people but as we can see that from the above maps, this is not the case in the state.

So I used KMeans Clustering to find the actual no of clusters of Hospitals and others.

Using K-Means and Evaluation Measures to find how the Infra is actually clustered

```
In [103]: import matplotlib.pyplot as plt
          from sklearn.cluster import KMeans
          import seaborn as sns; sns.set()
          import csv
          df_kmeans=Df_Delhi_Medical_infra
```

```
In [104]: K_clusters = range(1,10)
          kmeans = [KMeans(n_clusters=i) for i in K_clusters]
          Y_axis = df_kmeans[['lat']]
          X_axis = df_kmeans[['lng']]
          score = [kmeans[i].fit(Y_axis).score(Y_axis) for i in range(len(kmeans))]
          # Visualize
          plt.plot(K_clusters, score)
          plt.xlabel('Number of Clusters')
          plt.ylabel('Score')
          plt.title('Elbow Curve')
          plt.show()
```
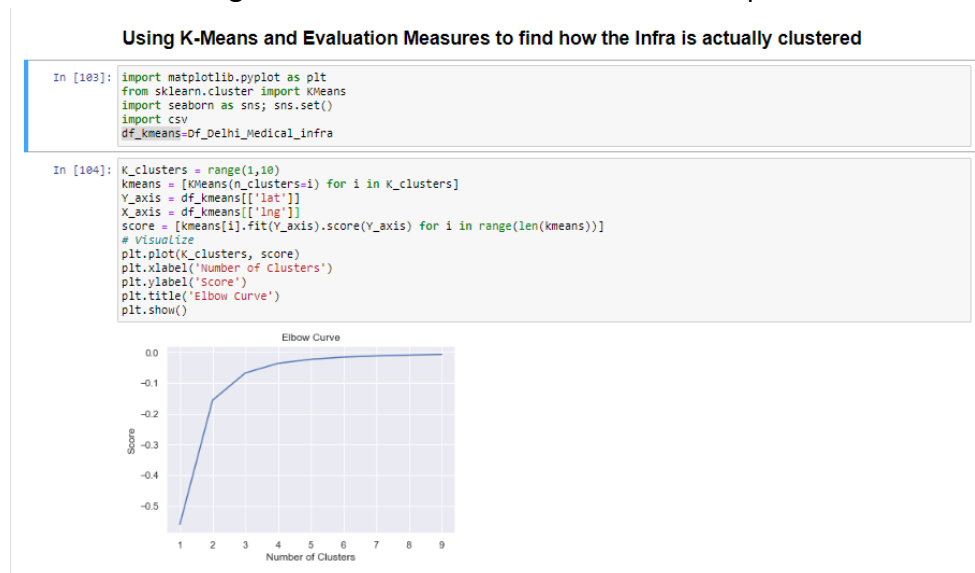
Figure: Kmeans and Elbow Evaluation Method.

Now ideally Delhi should have 9 clusters of hospitals and others but as we can infer from the elbow graph, it is only 3. So, as a result, the remaining districts of Delhi needs more no of camps and new hospitals to tackle this COVID 19 more effectively. I also assigned the label of clusters to the respective hospital, labs, etc.

```
In [106]: kmeans = KMeans(n_clusters = 3, init ='k-means++')
          kmeans.fit(df_kmeans[df_kmeans.columns[3:5]]) # Compute k-means clustering.
          df_kmeans['cluster_label'] = kmeans.fit_predict(df_kmeans[df_kmeans.columns[2:4]])
          centers = kmeans.cluster_centers_ # Coordinates of cluster centers.
          labels = kmeans.predict(df_kmeans[df_kmeans.columns[2:4]]) # Labels of each point
          df_kmeans
```

Out[106]:

	name	categories	lat	lng	cluster_label
0	Primus super speciality hospital	Hospital	28.592961	77.181430	2
1	Dr. Ram Manohar Lohia Hospital	Hospital	28.624654	77.200609	2
2	Sir Gangaram Hospital | सर गंगाराम अस्पताल	Hospital	28.638601	77.188884	2
3	Loknayak Hospital	Hospital	28.638303	77.238829	1
4	B.L kapoor memorial hospital	Hospital	28.643368	77.179899	2
5	BLK Super Specialty Hospital	Hospital	28.643392	77.179582	2
6	Safdarjung hospital	Hospital	28.566744	77.204870	2
7	St. Stephen's Hospital	Hospital	28.666454	77.214489	1
8	Aashlok Hospital	Hospital	28.566571	77.198902	2
9	AIIMS	Medical School	28.567922	77.209255	2
10	Moolchand Hospital	Hospital	28.564782	77.243863	2
11	Moolchand medcity	Hospital	28.566345	77.235344	2

Cluster Label is added to the corresponding Hospitals and Other facilities

Figure: Cluster Labels added to the respective hospitals.

Conclusion:

In this study, I analyzed the data for Delhi's Health Infrastructure and its demographics. And then used the data to view the spread of the hospital's facilities. Then I built the Kmean Clustering model to see the actual clustering of medical facilities in the state. And This model is very effective to see which of the state needs more attention in terms of new hospitals, testing centers, and isolation camps. For Example, I would highly recommend to setup more isolation centers and test on the outskirts of Delhi as there have relatively less no of medical facilities which are evident from the plots.

The above study can also be fine-tuned by adding by taking following into the consideration:

- Use of population data for clusterings.
- Taking into account the medical infrastructure of the individual facilities such as No of beds available and No of doctors, nurses available. Etc

These data are more difficult to extract and quantify, but if optimized, could bring significant improvements to the models.