The Battle of Neighborhoods – Hospitals & Covid19 in Rio de Janeiro, Brazil - Report

1. Introduction

1.1. Background

The current global pandemic caused by the Covid-19 virus has changed the way our world works. New challenges arise as mankind tries to understand and deal with this virus that has caused the loss of so many lives worldwide. The high contagion rate of this virus created chaos in most countries, as they tried to slow it down. One of the biggest issues that such an elevated infection rate causes is related to resources allocation. Most, if not all countries, do not have a medical infrastructure readily available for such an event. Thus, it becomes crucial to for a country to properly allocate its resources in order to have the most efficient counter measure to this crisis. This major problem becomes even more pressing in underdeveloped countries, where resources are scarcer and healthcare system is less developed.

Brazil, as one of the underdeveloped countries, is currently struggling in the fight against Covid-19, clocking over 100,000 fatalities with one of the highest numbers of confirmed cases. The country also goes through an economic crisis, which shorten the amount of resources that the government can utilize. Lastly, due to a political clash, there is a lack of coordination between federal government and the mayors of part of cities in the country, which increases the difficulty of the situation.

1.2. Business Problem

Our goal is to use the available data to improve the management of the resources dedicated to deal with this pandemic. By knowing the distribution of Covid-19 cases, it is possible to decide where medical resources, such as more testing, hospitals bed and workforce, and others, should be allocated to. For this project, we will use the city of Rio de Janeiro as a test case.

1.3. Target Audience

The target audience of this project is both the common people as well as government personnel. For the first, having knowledge of the hotspots of cases in the city can better their awareness of places that should be avoided, while for second group it can help in strategical choices which can optimize resource management while improving the results in the fight against Covid-19.

2. Data

For this project, two main sources of data were used. The first one is related to the total number of confirmed cases of Covid-19 in the city of Rio de Janeiro and its distribution across the neighborhoods. This was obtained from the website below. It is important to highlight that the information displayed is updated daily, so the actual numbers are in always changing. The data used was obtained on 04/08/2020.

https://experience.arcgis.com/experience/38efc69787a346959c931568bd9e2cc4

From this website, an excel file was downloaded and later, uploaded into IBM database for treatment. This excel file contains information on each case reported, such as date, neighborhood, postal code, and status. Figure 1 shows an example of the data.

NOTIFIC	INICIO	BAIRRO	RESID	EVOLUCAO	OBITO	CEP	ATUALIZACAO
01/04/2020	28/03/2020	PACIENCIA	5.3	�bito	06/04/2020	23585800	03/08/2020
01/04/2020	01/04/2020	COSTA BARROS	3.3	recuperado		21531991	03/08/2020
01/04/2020	28/03/2020	RAMOS	3.1	recuperado		21060330	03/08/2020
01/04/2020	28/03/2020	CAMPO GRANDE	5.2	recuperado		23013630	03/08/2020
01/04/2020	22/03/2020	LEBLON	2.1	recuperado		22441030	03/08/2020

Figure 1 - Data on Covid-19 cases in Rio de Janeiro uploaded into IBM database

More information on the treatment of this data will be described in the next section.

Lastly, Foursquare database was used to obtain the location of the main hospitals in the city. A query was used to find the location of the hospitals inside each neighborhood. Figure 2 below displays an example of the results. Additionally, more details will be discussed in the Methodology section.

	NEIGHBOURHOOD	NEIGHBOURHOOD LATITUDE	NEIGHBOURHOOD LONGITUDE	HOSPITAL	HOSPITAL LATITUDE	HOSPITAL LONGITUDE
0	abolicao	-22.886161	-43.299846	Hospital Psiquiatrico	-22.887598	-43.304885
1	andarai	-22.929084	-43.253486	Hospital Federal do Andaraí	-22.927836	-43.252607
2	andarai	-22.929084	-43.253486	Associação Dos Veteranos do Hospital Do Andarai	-22.927926	-43.253674
3	andarai	-22.929084	-43.253486	Centro Cirúrgico	-22.927771	-43.252309
4	andarai	-22.929084	-43.253486	Ambulatório	-22.927190	-43.252167

Figure 2 - Hospital information retrieved from Foursquare database

3. Methodology

The methodology of this work is divided into 3 main steps. First, it will include an analysis to find out the main hotspots of Covid-19 cases in the city. Then, the second part will investigate the distribution of hospitals inside each neighborhood. Lastly, we will try to optimize the hospitals that should be prioritized in terms of resources.

We start by installing and importing a few important libraries (Figure 3).

```python
# Importing necessary libraries
import pandas as pd
import numpy as np
import ibm_db
import ibm_db_dbi
import geopy
from geopy.geocoders import Nominatim
import geopy.distance
import folium
from folium import plugins
from folium.plugins import HeatMap
import requests
import urllib3
import json
from pandas.io.json import json_normalize
import seaborn as sns
import matplotlib.pyplot as plt
from matplotlib import cm
import matplotlib.colors as colors
from sklearn.cluster import KMeans
```

Figure 3 - Libraries imported for the project

3.1. Analysis of Covid-19 Cases in Rio de Janeiro

After creating a connection to the IBM database, we created a data frame containing both neighborhood names and postal codes. A total of 61,407 cases were in the data base when it was exported from the web site mentioned at chapter 2 (Figure 4).

```
In [4]: #Getting neighbourhood and postal code for all reported cases in the city and creating a dataframe with this information.
        sql = 'select BAIRRO,CEP from COVID_RJ'
        df_cases = pd.read_sql(sql, hdbi)
        df_cases.columns = ['NEIGHBOURHOOD', 'POSTAL_CODE']
        df_cases.head()
Out[4]:
```

	NEIGHBOURHOOD	POSTAL_CODE
0	CACHAMBI	20771330
1	TIJUCA	20261120
2	JACAREPAGUA	22753211
3	BENFICA	20910250
4	SANTISSIMO	23094140

```
In [5]: # Checking the total number of cases reported in the city
        df_cases.shape
Out[5]: (61407, 2)
```

Figure 4 - Initial data frame with Covid-19 cases

Following that, we created a new data frame combining the cases in the same neighborhoods. Figure 5 shows both the function used and a snip of the outcome.

In [6]:
```python
# Creating a dataframe with the number of cases in each neighbourhood
df_cases['TOTAL_CASES'] = 1
df_neighbourhood = df_cases.groupby('NEIGHBOURHOOD').count()[['TOTAL_CASES']].reset_index()
df_neighbourhood['NEIGHBOURHOOD'] = df_neighbourhood['NEIGHBOURHOOD'].str.lower()
df_neighbourhood.sort_values('TOTAL_CASES', ascending=False).head()
```

Out[6]:

	NEIGHBOURHOOD	TOTAL_CASES
35	copacabana	2768
9	barra da tijuca	2713
138	tijuca	2580
23	campo grande	2507
8	bangu	1708

Figure 5 - Data frame with cases combined by neighborhood

Additionally, we used the geolocator function of get the coordinates of each neighborhood and add it to the data frame. Figure 6 shows a snip of the result.

Out[7]:

	NEIGHBOURHOOD	TOTAL_CASES	LATITUDE	LONGITUDE
0	abolicao	129	-22.886161	-43.299846
1	acari	105	-22.822153	-43.340674
2	agua santa	68	-22.911143	-43.312126
3	alto da boa vista	109	-22.962113	-43.253582
4	anchieta	497	-22.823190	-43.399107

Figure 6 - Data frame with coordinates for each neighborhood

Next, we checked for any possible NaN value and used the same geolocator function to get the coordinates of the city of Rio de Janeiro. This was done to properly create a graph, using the folium function, to illustrate the distribution of cases across the city. Figure 7 shows the map of the city, in which the bigger the red circles, the higher the number of cases in that neighborhood.

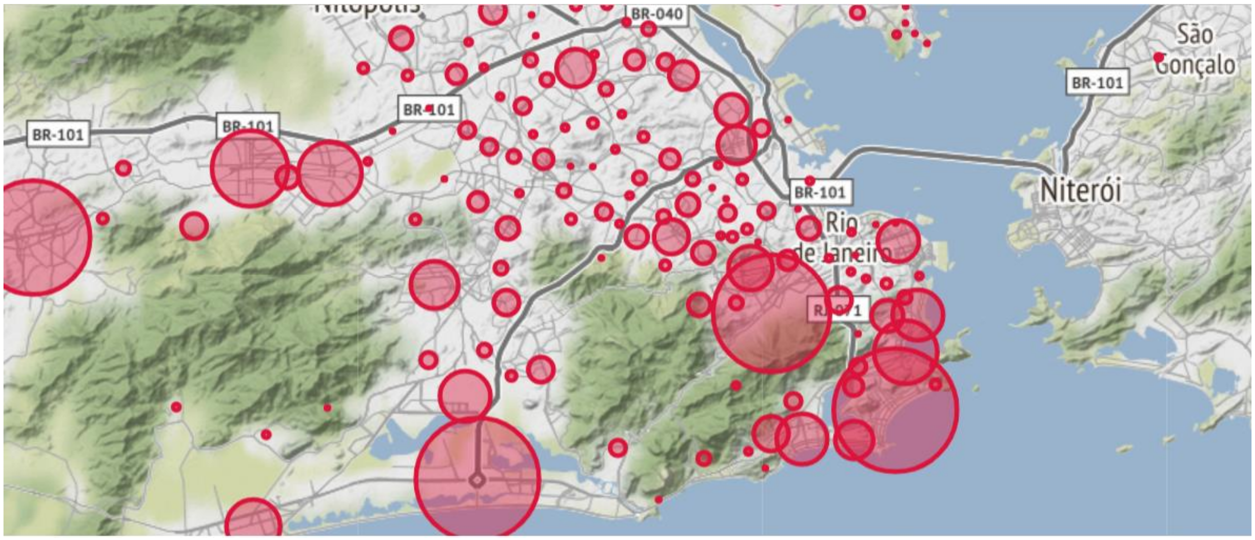

Figure 7 - Hotspots of Covid-19 cases in Rio de Janeiro

3.2. Hospitals Analysis

For this stage, we used the database from Foursquare. we started by passing our user information and creating a function to specifically query for hospitals inside each neighborhood in the city. Figure 8 below shows the function created, while Figure 9 displays a snip of the outcome.

```
def getNearbyHospitals(names, latitudes, longitudes):
    radius=500
    LIMIT=100
    cat_Id ='4bf58dd8d48988d196941735' #Hospital category in Foursquare
    query = 'Hospital'
    hospitals_list=[]
    for name, lat, lng in zip(names, latitudes, longitudes):
        print(name)

    # create the API request URL
        url = 'https://api.foursquare.com/v2/venues/search?client_id={}&client_secret={}&ll={},{}&v={}&radius={}&limit={}&query={}'.format(CLIENT_ID, CLIENT_SECRET, lat, lng, VERSION, radius, LIMIT,


    # make the GET request
        results = requests.get(url).json()["response"]["venues"]

    # return only relevant information for each nearby venue
        hospitals_list.append([(
            name,
            lat,
            lng,
            h['name'],
            h['location']['lat'],
            h['location']['lng']) for h in results])

    nearby_hospitals = pd.DataFrame([item for hospitals_list in hospitals_list for item in hospitals_list])
    nearby_hospitals.columns = ['NEIGHBOURHOOD',
                  'NEIGHBOURHOOD LATITUDE',
                  'NEIGHBOURHOOD LONGITUDE',
                  'HOSPITAL',
                  'HOSPITAL LATITUDE',
                  'HOSPITAL LONGITUDE']

    return(nearby_hospitals)
```

Figure 8 -Function to get hospitals from the Foursquare database

	NEIGHBOURHOOD	NEIGHBOURHOOD LATITUDE	NEIGHBOURHOOD LONGITUDE	HOSPITAL	HOSPITAL LATITUDE	HOSPITAL LONGITUDE
0	abolicao	-22.886161	-43.299846	Hospital Psiquiatrico	-22.887598	-43.304885
1	andarai	-22.929084	-43.253486	Hospital Federal do Andaraí	-22.927836	-43.252607
2	andarai	-22.929084	-43.253486	Associação Dos Veteranos do Hospital Do Andarai	-22.927926	-43.253674
3	andarai	-22.929084	-43.253486	Centro Cirúrgico	-22.927771	-43.252309
4	andarai	-22.929084	-43.253486	Ambulatório	-22.927190	-43.252167

Figure 9 - Data frame with hospitals per neighborhood

Additionally, we also plotted the hospitals in a map for a better visualization (Figure 10).

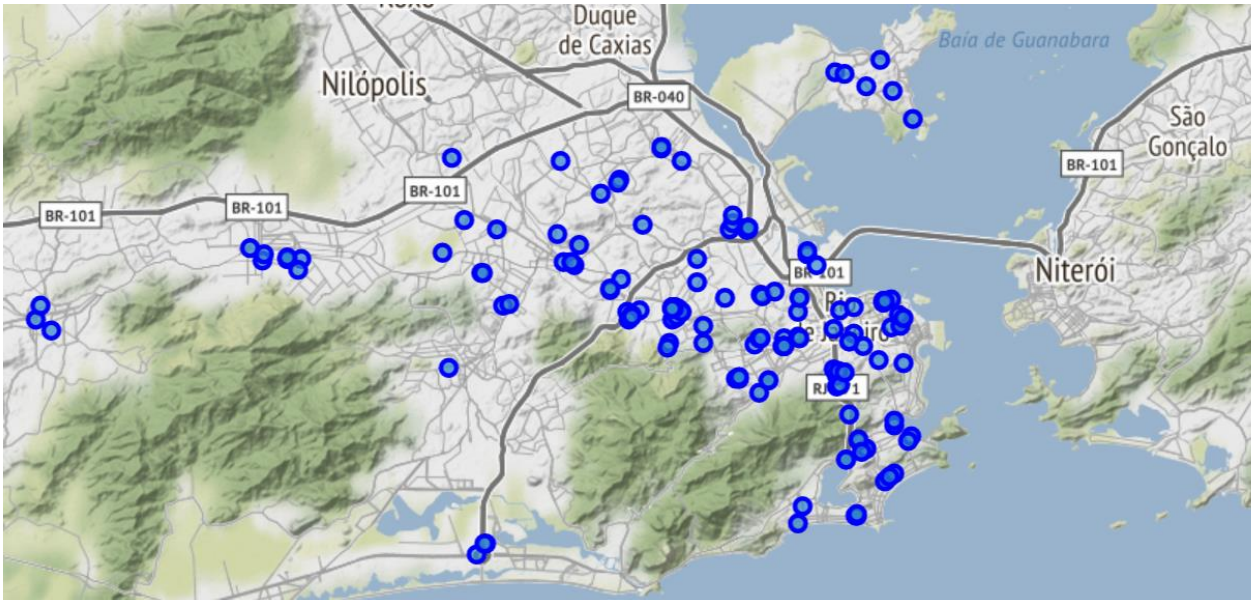

Figure 10 - Hospitals in the city of Rio de Janeiro

3.3. Analysis of Hospitals to be Prioritized

After getting all the needed data, we started to check the neighborhoods with the higher amount of cases. Figure 11 shows a data frame with the neighborhoods ranked from highest to lowest and a new column showing the percentage of cases in comparison to the total.

In [16]:
```
#Getting the neighbourhoods with higher number of cases and calculating the percentage of cases to define threshold
df_high_cases = df_neighbourhood.sort_values(by=['TOTAL_CASES'], ascending=False)
df_high_cases['PERCENTAGE'] = (df_high_cases['TOTAL_CASES'] / df_high_cases['TOTAL_CASES'].sum())*100
df_high_cases.head()
```

Out[16]:

	NEIGHBOURHOOD	TOTAL_CASES	LATITUDE	LONGITUDE	PERCENTAGE
35	copacabana	2768	-22.971964	-43.184343	4.557804
9	barra da tijuca	2713	-22.999740	-43.365993	4.467241
138	tijuca	2580	-22.933216	-43.238145	4.248242
23	campo grande	2507	-22.902953	-43.559129	4.128040
8	bangu	1708	-22.875305	-43.464880	2.812402

Figure 11 - Data frame with highest to lowest number of cases per neighborhood

We also plotted a bar plot to help in the decision-making process (Figure 12).

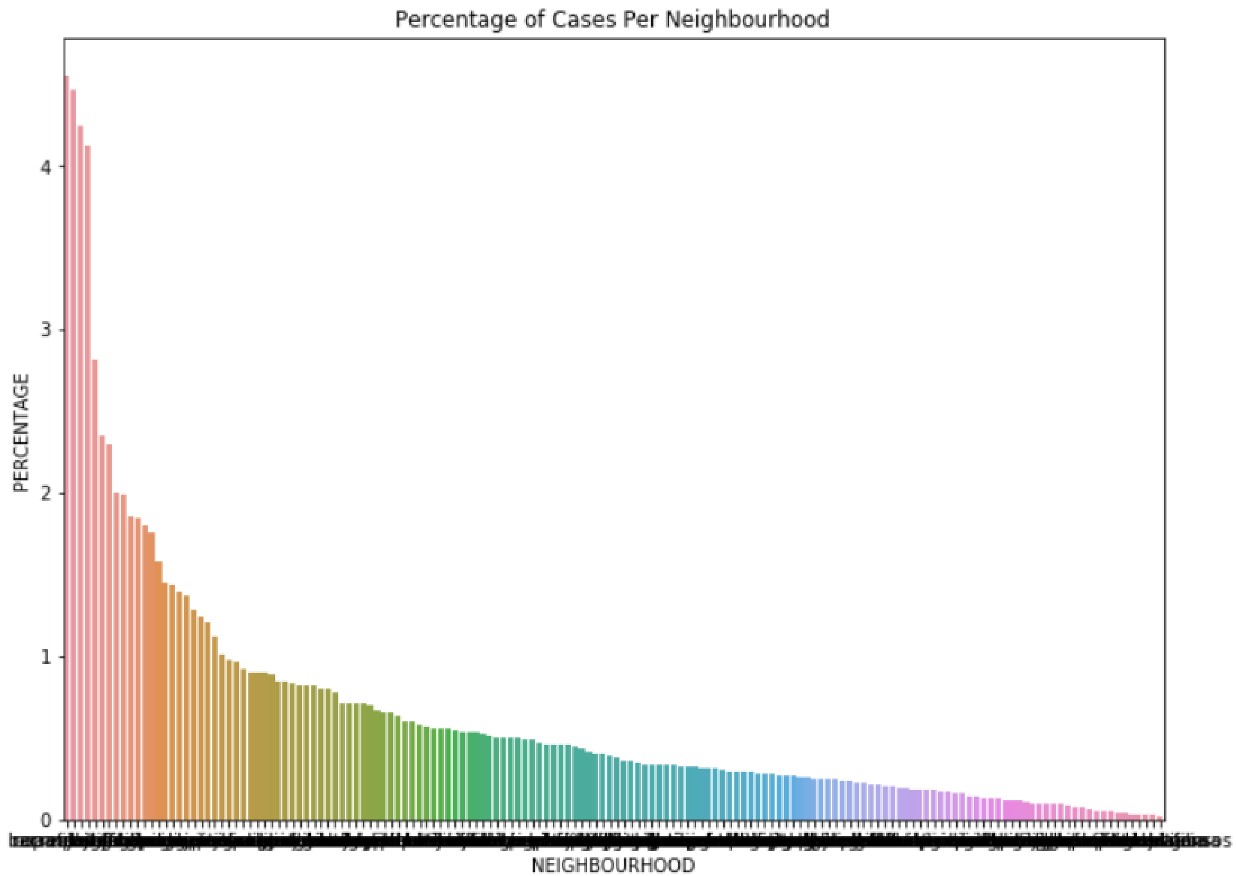

Figure 12 - Bar plot of percentage of cases per neighborhood

As we can see, there are a few neighborhoods with a slightly higher percentage of cases. We decided to put a cutoff at 2%, since these neighborhoods represent near 25% of all cases in the city. Figure 13 below shows the neighborhoods selected.

	NEIGHBOURHOOD	TOTAL_CASES	LATITUDE	LONGITUDE	PERCENTAGE
35	copacabana	2768	-22.971964	-43.184343	4.557804
9	barra da tijuca	2713	-22.999740	-43.365993	4.467241
138	tijuca	2580	-22.933216	-43.238145	4.248242
23	campo grande	2507	-22.902953	-43.559129	4.128040
8	bangu	1708	-22.875305	-43.464880	2.812402
114	realengo	1432	-22.877274	-43.430103	2.357939
15	botafogo	1397	-22.948845	-43.179829	2.300308

Figure 13 - Neighborhoods with cases over 2% of the total

Following that, we created a new data frame containing the hospitals inside these neighborhoods. This will be further discussed in the Results chapter. Nonetheless, we came to a total of 20 hospitals. Trying to narrow down a bit more, since resources are scarce, we used a k-means clustering algorithm to extract more information of the data. We started by choosing 4 as the ideal number of clusters (Figure 14).

```
#Finding out the ideal number of K
temp_df = df_neighbourhood.drop('NEIGHBOURHOOD', 1)
sse = {}
for k in range(1, 10):
    kmeans = KMeans(n_clusters=k, max_iter=1000).fit(temp_df)
    temp_df["clusters"] = kmeans.labels_
    sse[k] = kmeans.inertia_ # Inertia: Sum of distances of samples to their closest cluster center
plt.figure()
plt.plot(list(sse.keys()), list(sse.values()))
plt.xlabel("Number of cluster")
plt.ylabel("SSE")
plt.show()
```

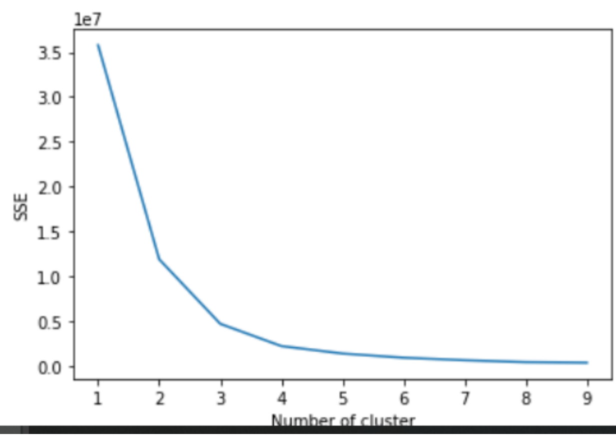

Figure 14 - Identifying the ideal number of clusters

Next, we applied the algorithm in the data. This allowed us to split the neighborhoods accordingly to their number of cases, and then associate the hospitals with this clusters. Figure 15 shows all the hospitals divided into different colors, representing the different clusters that they are related to. Moreover, Figure 16 shows the centroids of each cluster as well.

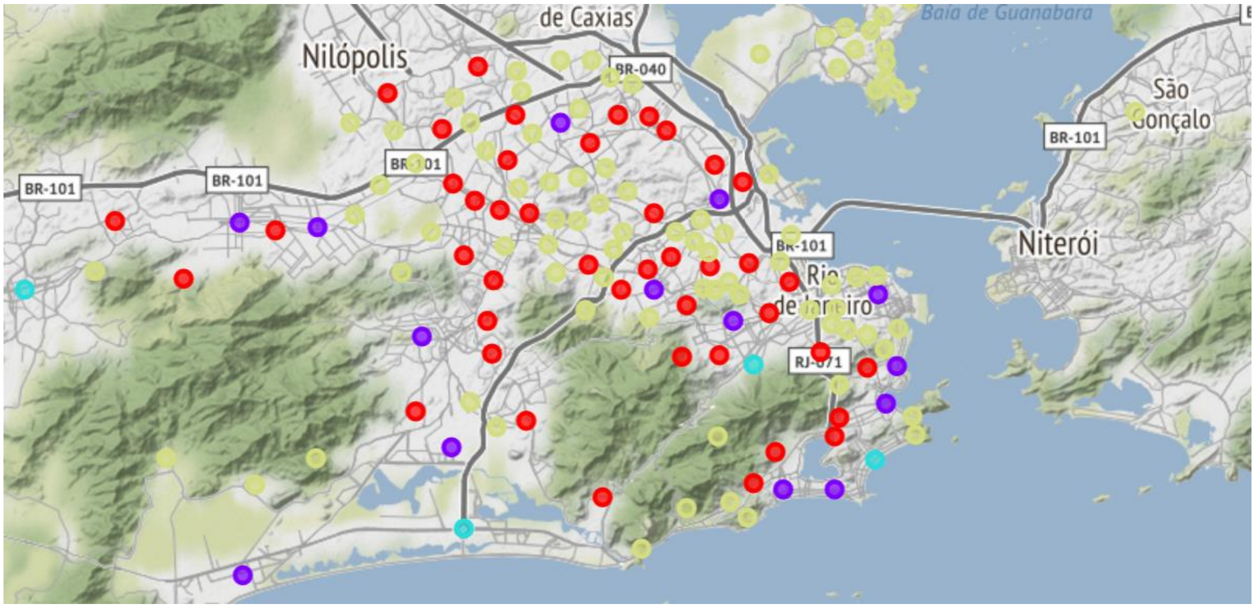

Figure 15 - Hospitals divided into all 4 clusters

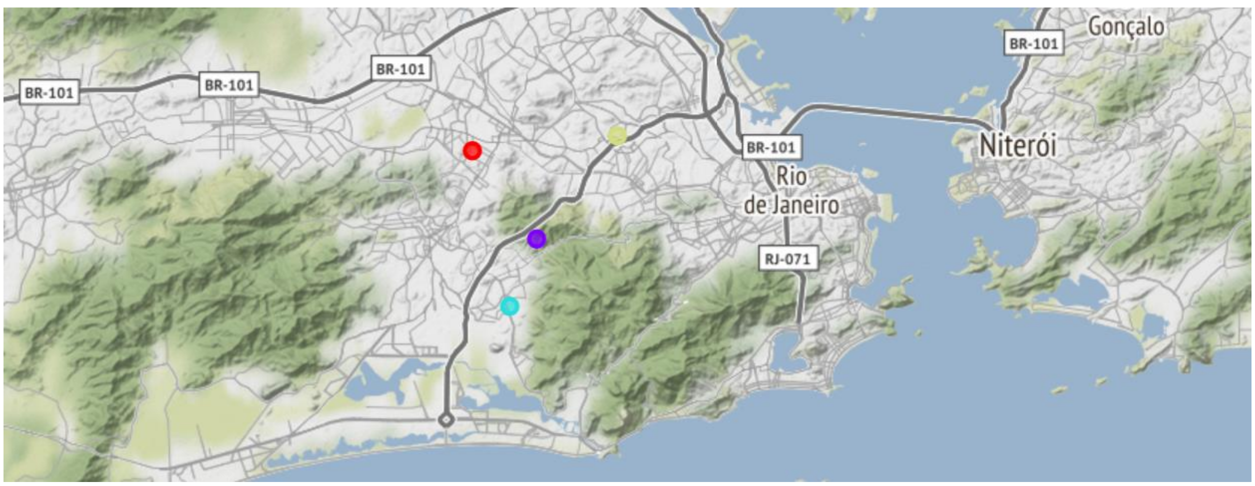

Figure 16 - Centroids of all 4 clusters

Lastly, we used the coordinates of the cluster representing the neighborhoods with the highest amount of cases (light blue) and ordered the hospitals accordingly to their distance to this centroid. Figure 17 shows the function used to calculate this distance, while Figure 18 shows a snip with the closest hospitals.

```python
def DistHosp(lat_h, long_h, names, lat_c, long_c):
    list_dist = []

    coords_2 = (lat_c, long_c)
    for lat, long, name in zip(lat_h, long_h, names):
        coords_1 = (lat, long)
        result = geopy.distance.distance(coords_1, coords_2).km

        list_dist.append([
            name,
            result])

        #list_dist.append(name)
        #list_dist.append(result)

    dist_hospitals= pd.DataFrame(list_dist)

    return(dist_hospitals)
```

Figure 17 - Function to calculate distance from hospitals to chosen centroid

	HOSPITAL	DISTANCE(KM)
147	Hospital Cardoso Fontes	5.509080
9	Hospital Municipal Lourenço Jorge	5.577774
11	BikeRio - Estação 108 Hospital Lourenço Jorge	5.620739
10	Hospital Lourenço Jorge	6.197632
129	Biolife Hospitalar	6.275018

Figure 18 - Example of closest hospitals from chosen centroid

The main goal was to cross check both results, from the initial chosen list of hospitals and the list generated by the k-means analysis and see if any hospitals would be chosen.

4. Results

From the initial analysis, considering the neighborhoods with most cases of Covid-19, we came to a total of 20 hospitals chosen. Figure 19 shows the data frame with all chosen hospitals, while Figure 20 displays them in a map, alongside with the hotspots of cases.

	NEIGHBOURHOOD	HOSPITAL	HOSPITAL LATITUDE	HOSPITAL LONGITUDE
0	copacabana	Hospital San Magno	-22.969902	-43.187972
1	copacabana	Hospital Copa D'or	-22.968952	-43.187092
2	copacabana	Hospital Miguel Couto - Serviço de Neurocirurgia	-22.966908	-43.183690
3	copacabana	Pronto Atendimento Unimed Rio	-22.967771	-43.185715
4	barra da tijuca	Hospital Municipal Lourenço Jorge	-22.995239	-43.364740
5	barra da tijuca	Hospital Lourenço Jorge	-22.999743	-43.368382
6	barra da tijuca	BikeRio - Estação 108 Hospital Lourenço Jorge	-22.995319	-43.365410
7	tijuca	Hospital Veterinário Canne & Gatto	-22.933902	-43.243249
8	tijuca	Hospital Tijutrauma	-22.928568	-43.239529
9	tijuca	Hospital	-22.928660	-43.239456
10	campo grande	Hospital WestDor	-22.903854	-43.563569
11	campo grande	hospital oeste dor	-22.908304	-43.556354
12	campo grande	Casa de Saúde Nossa Senhora do Carmo	-22.898326	-43.561291
13	bangu	Prosaúde Hospital de Clínicas	-22.879820	-43.463307
14	bangu	Hospital São Lourenço	-22.877684	-43.462823
15	bangu	CMS Waldyr Franco	-22.874973	-43.468639
16	botafogo	Hospital Amiu Botafogo	-22.946703	-43.183936
17	botafogo	Hospital Geral De Fortaleza do Exército	-22.951468	-43.176000
18	botafogo	Hospital Memorial Infantil Botafogo Amiu	-22.945337	-43.184080
19	botafogo	Laboratório Hospital Rocha Maia	-22.953232	-43.177586

Figure 19 - Hospitals chosen accordingly to neighborhood

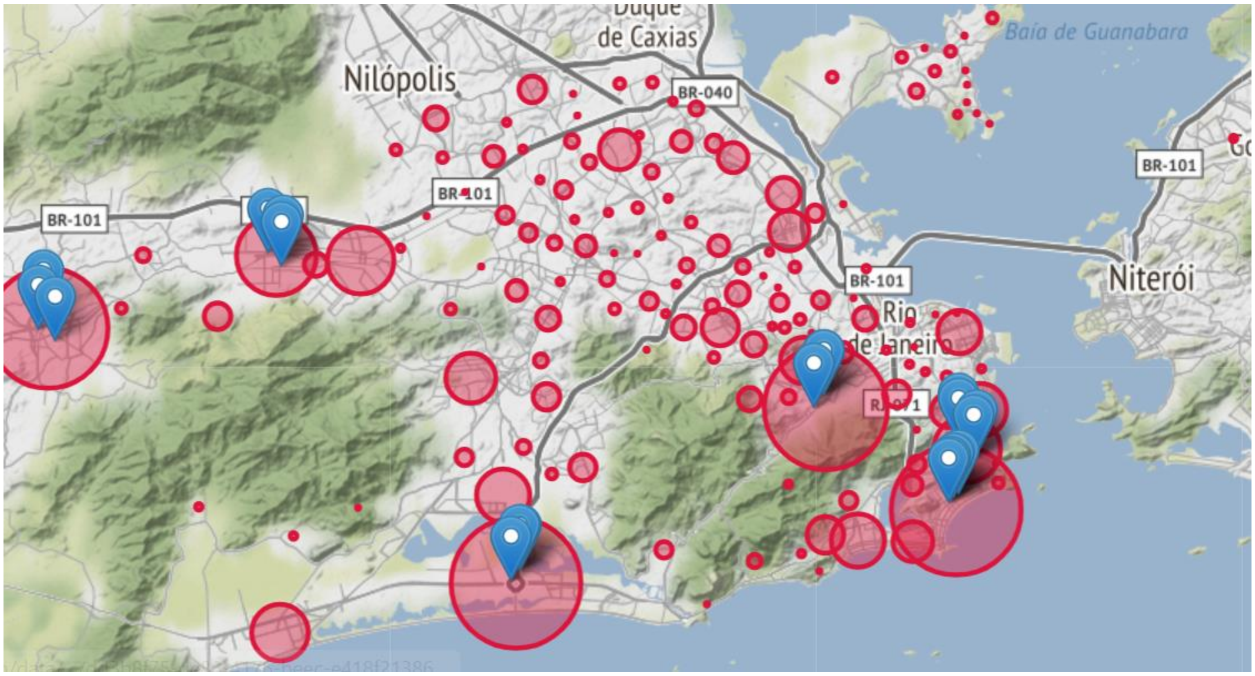

Figure 20 - Map of hospitals chosen in the city

As we previously discussed, to further narrow down our results, we used a k-means algorithm. For this, we also took the closest 20 hospitals to the centroid for the neighborhoods with the greatest number of cases. Figure 21 displays the hospitals chosen.

	HOSPITAL	DISTANCE(KM)
147	Hospital Cardoso Fontes	5.509080
9	Hospital Municipal Lourenço Jorge	5.577774
11	BikeRio - Estação 108 Hospital Lourenço Jorge	5.620739
10	Hospital Lourenço Jorge	6.197632
129	Biolife Hospitalar	6.275018
128	Hospital Beneficência Portuguesa	6.275289
67	Pista do Instituto Nise da Silveira	6.462215
68	Ab Med Hospitalar	6.642900
62	Hospital Pedro II	6.686138
66	Hospital Psiquiátrico Nise da Silveira	6.701888
92	UTQ - Hospital Naval Marcílio Dias	6.735405
91	Hospital Naval Marcílio Dias	6.937256
65	AmericanCor Hospital	7.104640
123	Hospital Municipal da Piedade	7.232810
124	Hospital Municipal Da Piedade	7.251772
107	Hospital dos Óculos	7.732544
44	Hospital Pasteur	7.833094
0	Hospital Psiquiatrico	7.848777
106	Hospital Paster	7.863191
43	Hospital Norte D'Or	7.914414

Figure 21 - Hospitals chosen accordingly to proximity of centroid

By comparing both data frames, we saw that 3 hospitals appeared in both cases (Figure 22).

In [41]:
```
c1_final = df_chosen_hospitals.merge(c1_hospitals, on='HOSPITAL')
c1_final
```

Out[41]:

	NEIGHBOURHOOD	HOSPITAL	HOSPITAL LATITUDE	HOSPITAL LONGITUDE	DISTANCE(KM)
0	barra da tijuca	Hospital Municipal Lourenço Jorge	-22.995239	-43.364740	5.577774
1	barra da tijuca	Hospital Lourenço Jorge	-22.999743	-43.368382	6.197632
2	barra da tijuca	BikeRio - Estação 108 Hospital Lourenço Jorge	-22.995319	-43.365410	5.620739

Figure 22 - Hospitals verified by cross-checking

It is important, nonetheless, to point out that by doing a quick google research, we can see that these hospitals are the same, so in reality, we only had 1 match by doing the cross-checking.

This hospital becomes our first choice to resource allocation. However, since we are talking about a large city and from Figure 20 we can see that the neighborhoods of highest cases are quite spread apart, thus a bit more consideration can be done regarding other possible choices.

5. Discussion

Since the cross-checking only resulted in a single hospital, we can investigate the previous result to more efficiently allocate our resources. By doing some research, there are a few hospitals that would not be suitable to handle tests or other treatments that are needed for this pandemic. Entries such laboratories or veterinaries, should not be taken into consideration.

Moreover, by looking at the hot spots, we can see 5 big clusters are initially made, which reside in places spread across the city. So, to better address any mobility issues that might be caused by this, we can choose 5 hospitals, 1 in each cluster, as our suggestion to focus the investment of the resources.

Figure 23 below shows the hospitals chosen, as Figure 24 displays them in the city map.

	NEIGHBOURHOOD	HOSPITAL	HOSPITAL LATITUDE	HOSPITAL LONGITUDE
0	copacabana	Hospital San Magno	-22.969902	-43.187972
4	barra da tijuca	Hospital Municipal Lourenço Jorge	-22.995239	-43.364740
8	tijuca	Hospital Tijutrauma	-22.928568	-43.239529
10	campo grande	Hospital WestDor	-22.903854	-43.563569
13	bangu	Prosaúde Hospital de Clínicas	-22.879820	-43.463307

Figure 23 - Final hospitals chosen

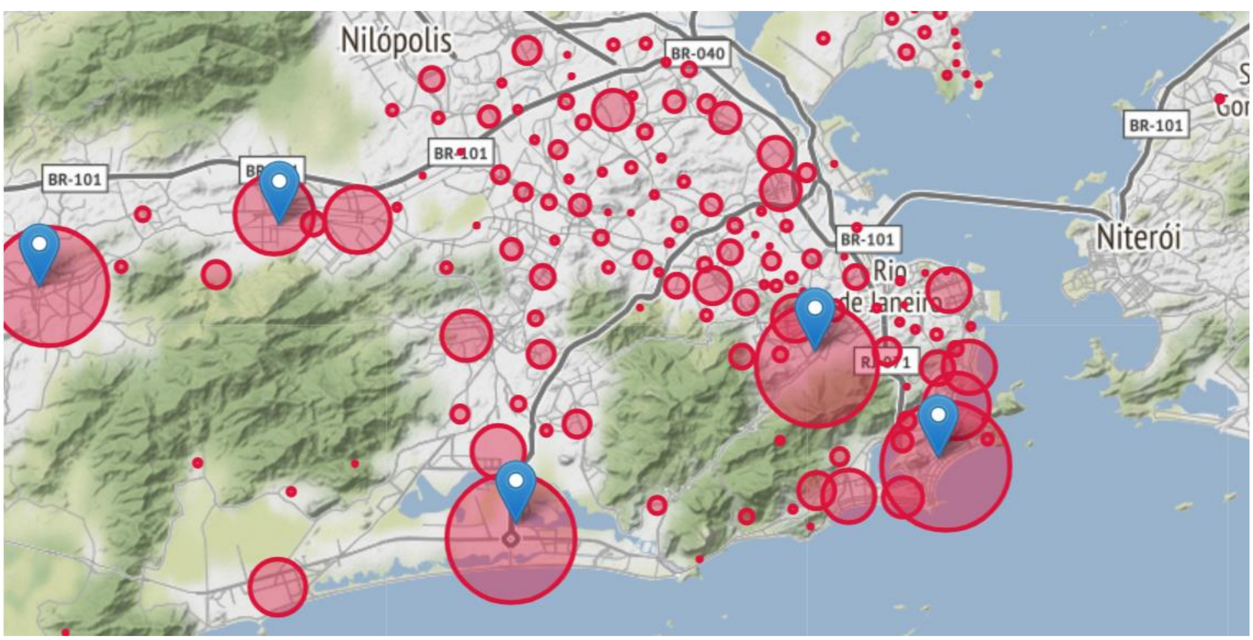

Figure 24 - Map of Rio de Janeiro with the chosen hospitals

6. Conclusion

After analyzing the data, we suggested 5 hospitals to be the focus of resources to help fight he Covid-19 pandemic in the city of Rio de Janeiro. Even though there are other details that would impact such decisions, this can work as a first step to help decision makers in the government.