

## Description of the data and how it will be used to solve the problem

### Data Acquisition and Data Cleaning:

#### 1. Data Acquisition

The data acquired for the project is the combination data from three sources. The first source of the project uses [London Coronavirus Infected cases](#) in London.

The dataset contains following columns:

- **Borough:** Common Name of London Borough
- **Confirmed Cases:** No. of Coronavirus infected patients in a Borough

The second source of data is scraped from the Wikipedia page that contains the [list of London Boroughs](#). The page contains additional information about the borough. The dataset contains the following columns:

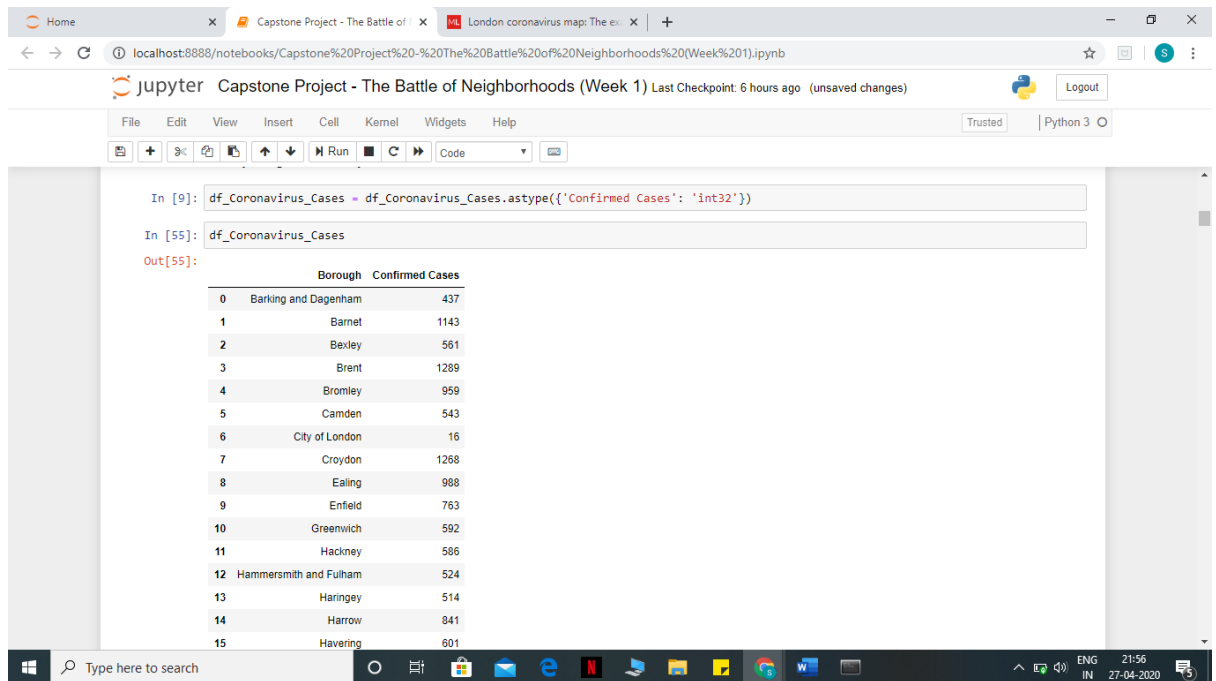
- **Borough:** The names of the 33 London boroughs.
- **Inner:** Categorizing the borough as an Inner London borough or an Outer London Borough.
- **Status:** Categorizing the borough as Royal, City or other borough.
- **Local authority:** The local authority assigned to the borough.
- **Political control:** The political party that control the borough.
- **Headquarters:** Headquarters of the Boroughs.
- **Area (sq mi):** Area of the borough in square miles.
- **Population (2013 est)[1]:** The population in the borough recorded during the year 2013.
- **Co-ordinates:** The latitude and longitude of the boroughs.
- **Nr. in map:** The number assigned to each borough to represent visually on a map.

The third data source is the list of Neighbourhoods in the Royal Borough of Brent as found on a Wikipedia page. This dataset is created from scratch using the [list of neighbourhoods](#) available on the site, the following are columns:

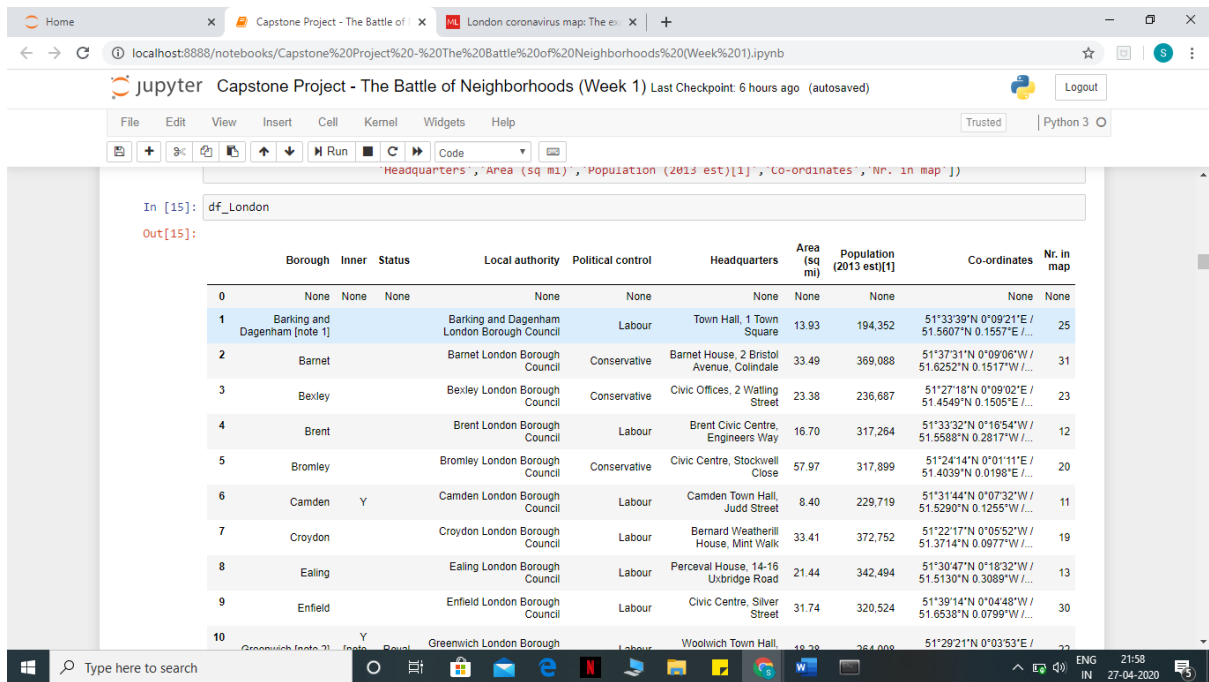
- **Neighbourhood:** Name of the neighbourhood in the Borough.
- **Borough:** Name of the Borough.
- **Latitude:** Latitude of the Borough.
- **Longitude:** Longitude of the Borough.

## 2. Data Cleaning

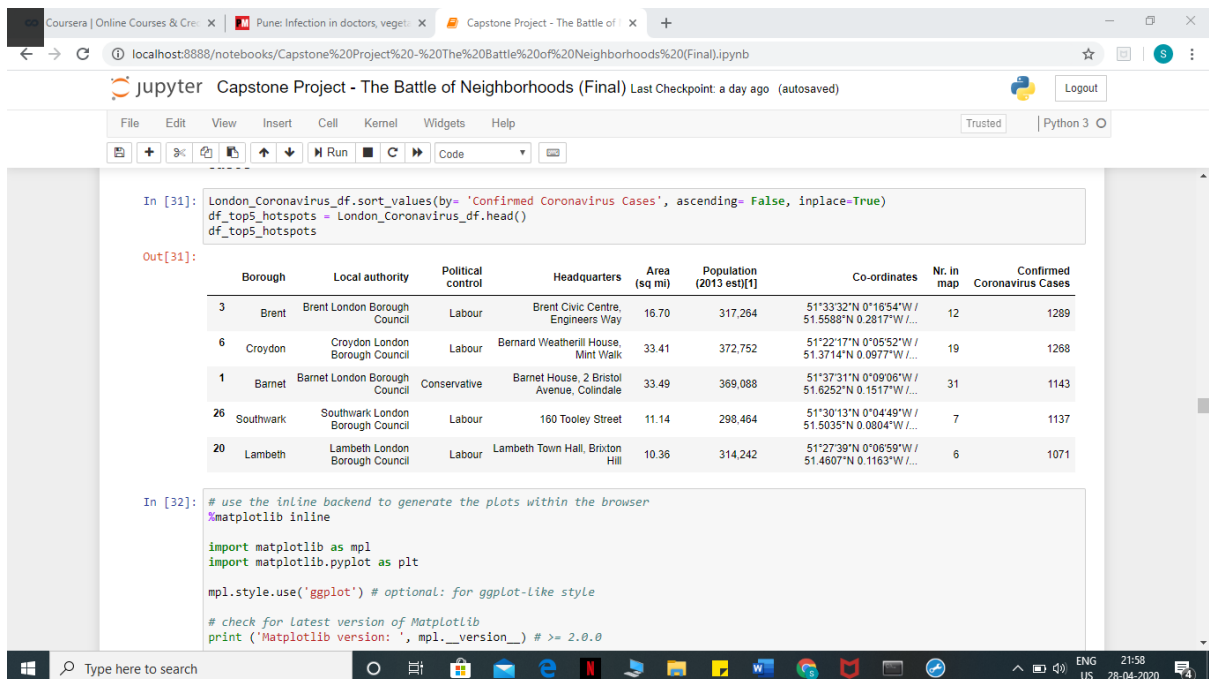
The data preparation for each of the three sources of data is done separately. Initially, we took the data from showing total number of confirmed infections in each borough as shown in the fig.



The second data is scraped from a Wikipedia page using the BeautifulSoup library in python. Using this library, we can extract the data in the tabular format as shown in the website. After the web scraping, string manipulation is required to get the names of the boroughs in the correct form. This is important because we will be merging the two datasets together using the Borough names.



The two datasets are merged on the Borough names to form a new dataset that combines the necessary information in one dataset. The purpose of this dataset is to visualize the coronavirus infection rates in each borough and identify the worst effected borough with the maximum number of coronavirus cases recorded so far.



After visualizing the coronavirus cases in each borough, we can find the borough with the maximum number of cases and hence tag that borough as the worst

effected borough. The third source of data is acquired from the list of neighbourhoods in the worst affected borough on Wikipedia. This dataset is created from scratch, the pandas data frame is created with the names of the neighbourhoods and the name of the borough with the latitude and longitude are obtained using Google Maps API geocoding to get the final dataset.

```
In [70]: df_neighbourhood = {'Neighborhood': Neighborhood, 'Borough':Borough, 'Latitude': Latitude, 'Longitude':Longitude}
Brent_neigh = pd.DataFrame(data=df_neighbourhood, columns=['Neighborhood', 'Borough', 'Latitude', 'Longitude'], index=None)
Brent_neigh

Out[70]:
```

	Neighborhood	Borough	Latitude	Longitude
0	Alperton	Brent	51.540804	-0.300096
1	Brent Park	Brent	51.499138	-0.399273
2	Brondesbury	Brent	51.545049	-0.202596
3	Brondesbury Park	Brent	51.540514	-0.210336
4	Church End	Brent	51.601117	-0.191890
5	Cricklewood	Brent	51.556699	-0.215751
6	Dollis Hill	Brent	51.551963	-0.239021
7	Harlesden	Brent	51.536357	-0.257833
8	Kensal Green	Brent	51.530606	-0.224445
9	Kilburn	Brent	51.541882	-0.197936
10	Kingsbury	Brent	51.584317	-0.278714

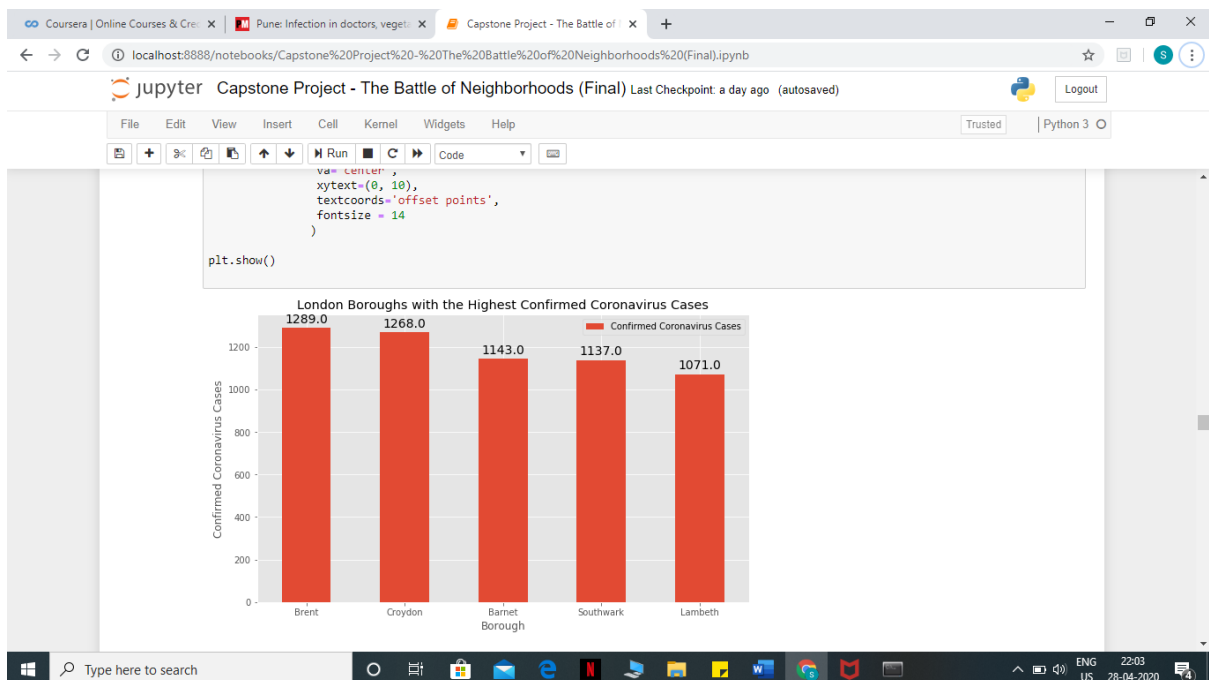
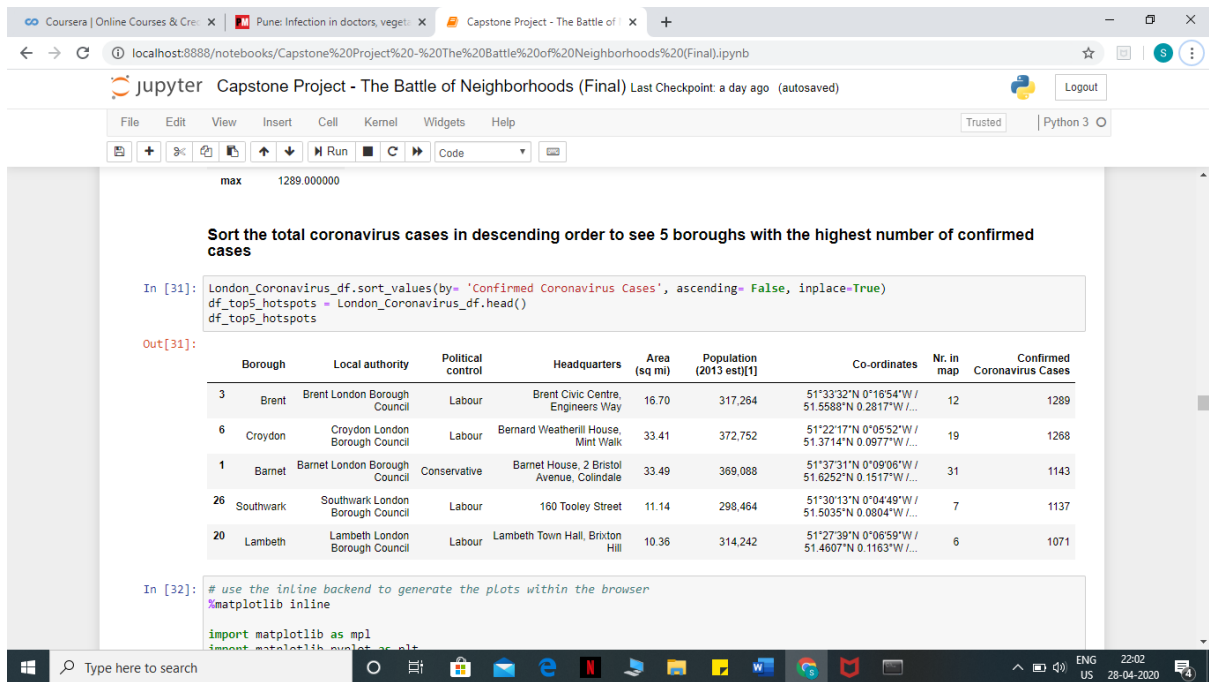
Get the co-ordinates of Central Location in London, United Kingdom (The center neighborhood of Brent)

```
In [71]: address = "Harlesden, London, United Kingdom"
```

## Solution of the Problem:

This is a critical problem that happened globally so the only solution suggested by the experts to maintain social distancing and to exhibit lockdown in the cities which are worst affected by the pandemic. This lockdown prohibits the unnecessary travel of the people as well as going out of the homes for the non-essential reasons. Also, building of hospitals along with quarantine facilities and employing the healthcare workers so that critical patients effected by the disease can be better taken care of.

One way is to analyse the worst regions of London and practice strict Lockdown there. Along with that the regions have to be critically monitored for the spread.



People should not be allowed to go out without masks and only one person per house should be permitted to go out for grocery items. No unnecessary travels should be available for citizens of that region. The infected people should be home quarantined and asked to stay away from other members. Senior citizens and children below age of 5 should be given extra precautions of that region. The complete region should be declared a red zone and strict lockdown should be enforced in the region.

The government should observe serious measures to handle the situation by maintaining required number of hospitals and medical facilities for the severely and critically ill patients infected by the disease. Currently, whatever number of hospitals and health facilities are there should be dedicated mostly to take care of above Covid-19 patients until unless there is an emergency of some other serious illness.

Hence, using the **Foursquare API** we can help locate common venues with hospitals in the neighbourhoods. Since hospitals and other healthcare facilities seems to be very less in the neighbourhoods of Brent which is worst effected, therefore plan should be to convert certain places like malls and restaurants which are closed for now to quarantine places or temporary hospitals where large number of patients can be taken care.

So that where home quarantine for some people is not possible their institutional quarantine should be provided so as to curb down the further spread of the coronavirus infections.

```

In [133]: print(Brent_venues.shape)
          Brent_venues.head()
(3, 7)

Out[133]:
   Neighborhood  Neighborhood Latitude  Neighborhood Longitude  Venue  Venue Latitude  Venue Longitude  Venue Category
0  Brondesbury      51.545049      -0.202596  Brondesbury medical center  51.543601      -0.200030      Hospital
1  Kilburn          51.541882      -0.197936  Brondesbury medical center  51.543601      -0.200030      Hospital
2  Willesden        51.546622      -0.235866  Willesden Centre for Health Care  51.542861      -0.235756      Hospital

In [134]: Brent_venues.groupby('Neighborhood').count()
Out[134]:
   Neighborhood Latitude  Neighborhood Longitude  Venue  Venue Latitude  Venue Longitude  Venue Category
Neighborhood
Brondesbury           1              1              1              1              1              1
Kilburn               1              1              1              1              1              1
Willesden             1              1              1              1              1              1

In [135]: print('There are {} uniques categories.'.format(len(Brent_venues['Venue Category'].unique())))
          There are 1 uniques categories.

In [136]: def addToMap(df, color, existingMap):
          for lat, lng, local, venue, venueCat in zip(df['Venue Latitude'], df['Venue Longitude'], df['Neighborhood'], df['Venue'], df[
            label = '{} ({} - {})'.format(venue, venueCat, local)
            label = folium.Popup(label, parse_html=True)
  
```

```

Brent_onehot = pd.get_dummies(Brent_venues[['Venue Category']], prefix="", prefix_sep="")

# add neighborhood column back to dataframe
Brent_onehot['Neighborhood'] = Brent_venues['Neighborhood']

# move neighborhood column to the first column
fixed_columns = [Brent_onehot.columns[-1]] + list(Brent_onehot.columns[:-1])
Brent_onehot = Brent_onehot[fixed_columns]

Brent_onehot.head()

Out[138]:
   Neighborhood  Hospital
0  Brondesbury         1
1  Kilburn            1
2  Willesden          1

In [139]: Brent_grouped = Brent_onehot.groupby('Neighborhood').mean().reset_index()
          Brent_grouped

Out[139]:
   Neighborhood  Hospital
0  Brondesbury         1
1  Kilburn            1
2  Willesden          1

In [140]: Brent_grouped.columns
  
```

Below, screenshot provide venues with restaurants and malls located with help of **Foursquare API** which can be converted to institutional quarantine places.

We can select the cluster with maximum no. of restaurants and studios available which can be utilized to fight the coronavirus pandemic in the region.

The screenshot shows a Jupyter Notebook interface with a table of venues in the Brent neighborhood. The table is titled 'Borough' and lists 10 clusters (0-9). Each cluster contains 10 venues, ranked by commonality. The venues include Supermarket, Café, Gym / Fitness Center, Asian Restaurant, Park, Indian Restaurant, Sandwich Place, Food & Drink Shop, Bus Stop, and Yoga Studio.

	Borough	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
0	Brent	Supermarket	Café	Gym / Fitness Center	Asian Restaurant	Park	Indian Restaurant	Sandwich Place	Food & Drink Shop	Bus Stop	Yoga Studio
1	Brent	Pub	Coffee Shop	Grocery Store	Indian Restaurant	Italian Restaurant	Middle Eastern Restaurant	Park	Pizza Place	Bus Stop	Café
2	Brent	Park	Farmers Market	Gym / Fitness Center	Pub	Grocery Store	Coffee Shop	Train Station	Japanese Restaurant	Yoga Studio	Discount Store
3	Brent	Turkish Restaurant	Indian Restaurant	Café	Coffee Shop	Japanese Restaurant	Pizza Place	Pub	Supermarket	Sandwich Place	Middle Eastern Restaurant
4	Brent	Coffee Shop	Hotel	Bus Stop	Fast Food Restaurant	Grocery Store	Breakfast Spot	Gym / Fitness Center	Pub	Sandwich Place	Bagel Shop
5	Brent	Hostel	Pet Store	Restaurant	Bakery	Grocery Store	Fast Food Restaurant	Flower Shop	Discount Store	Doner Restaurant	Entertainment Service
6	Brent	Pub	Auto Garage	Rental Car Location	Train Station	Movie Theater	Middle Eastern Restaurant	Yoga Studio	Fast Food Restaurant	Discount Store	Doner Restaurant
7	Brent	Pub	Portuguese Restaurant	Wine Shop	Grocery Store	Train Station	Indian Restaurant	Bakery	Restaurant	Farmers Market	Deli / Bodega
8	Brent	Indian Restaurant	Pub	Brazilian Restaurant	Supermarket	Café	Bus Stop	Middle Eastern Restaurant	Doner Restaurant	Coffee Shop	Clothing Store
9	Brent	Grocery Store	Herbs & Spices Store	Gym / Fitness Center	Coffee Shop	Indian Restaurant	Portuguese Restaurant	Fruit & Vegetable Store	Pub	Sandwich Place	Supermarket