Battle of Neighborhoods – Coursera Capstone

Introduction

This final project explores the best locations for opening an Italian restaurant in Brooklyn, New York. New York is the most densely populated and is a major city in the United States. New York City is composed of 5 boroughs and they are Brooklyn, Queens, Manhattan, Bronx, and Staten Island. Of these 5 boroughs, Brooklyn is the largest borough by population. New York has the largest population of Italians at 3.1 million people. People migrated from many parts of the world. There are many restaurants in New York City with different cuisines such as American, Italian, Chinese, Indian etc. Here the audience can be anyone who is looking to open or invest in a restaurant.

Data

For this project we need the following data:

1. New York City dataset that contains Borough, Neighborhoods along with their latitudes and longitudes
   Data Source: https://cocl.us/new_york_datase
   Description: This dataset contains the required information. And we will use this dataset to explore various neighborhoods of New York City.
2. Italian restaurants in Brooklyn neighborhood of New York City
   Data Source: Foursquare API
   Description: By using this API we will get all the venues in the Brooklyn neighborhood. We can filter these venues to get only Italian restaurants.

Problem Statement

1. What will be the best location for an Italian restaurant in Brooklyn, NY?
2. Which neighborhood is the best for one who is looking out to open an Italian restaurant in Brooklyn, NY?

Methodology

1. Collect the New York City data from the above-mentioned dataset.
2. We will get all the venues present in the Brooklyn Neighborhood using the Foursquare API.

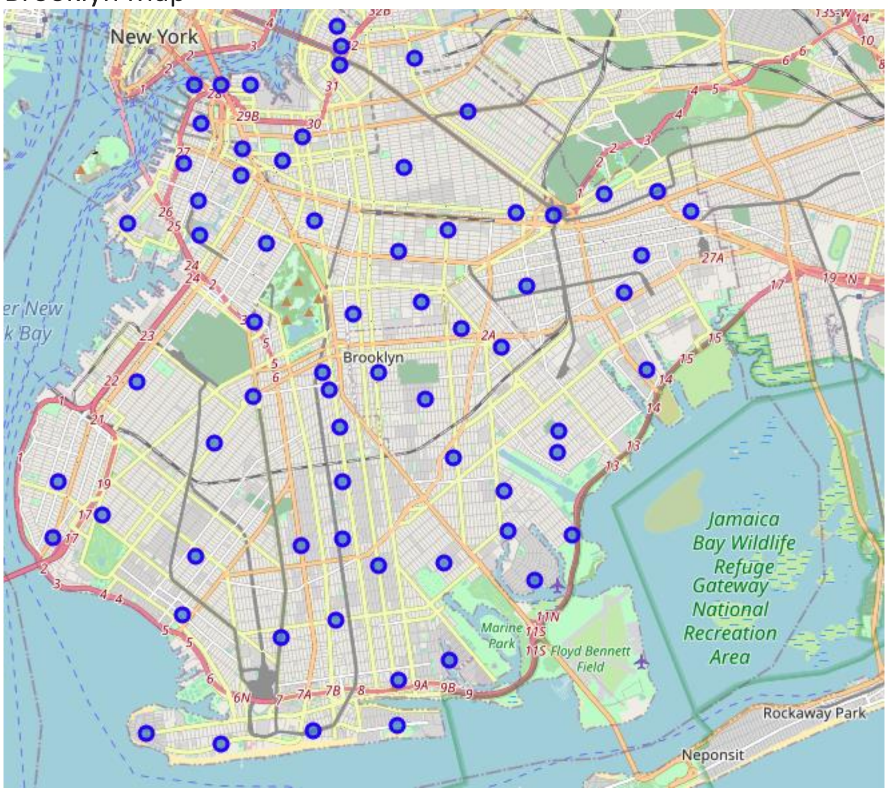
   First, we get the neighborhoods of the New York City by downloading the dataset and read the json data using the “json.load()” function. Then we pull the features attribute value from the data which consists of all the neighborhood related information. We then create a data frame and read the required columns into it using a for loop and those columns are ‘Borough’, ‘Neighbohood’, ‘Latitude’, and ‘Longitude’. Now we separate all the Brooklyn neighborhoods and will be using this data to get the venues.

```
[9] brooklyn_data = neighborhoods[neighborhoods['Borough'] == 'Brooklyn'].reset_index(drop=True)
    brooklyn_data.head()
```

	Borough	Neighborhood	Latitude	Longitude
0	Brooklyn	Bay Ridge	40.625801	-74.030621
1	Brooklyn	Bensonhurst	40.611009	-73.995180
2	Brooklyn	Sunset Park	40.645103	-74.010316
3	Brooklyn	Greenpoint	40.730201	-73.954241
4	Brooklyn	Gravesend	40.595260	-73.973471

Used geocoder to get the geographical coordinates of Brooklyn, New York. Created a map for Brooklyn using the folium library Map function

Brooklyn Map



Now the Foursquare API comes into picture. Defined a getNearbyVenues() function which takes in all the Brooklyn Neighborhoods, their latitude and longitude data as parameters and generates a data frame as output which consists of all the Venues present within each and every neighborhood of Brooklyn.

```
[22] print(brooklyn_venues.shape)
     brooklyn_venues.head()
```

(2742, 7)

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Bay Ridge	40.625801	-74.030621	Pilo Arts Day Spa and Salon	40.624748	-74.030591	Spa
1	Bay Ridge	40.625801	-74.030621	Bagel Boy	40.627896	-74.029335	Bagel Shop
2	Bay Ridge	40.625801	-74.030621	Leo's Casa Calamari	40.624200	-74.030931	Pizza Place
3	Bay Ridge	40.625801	-74.030621	Cocoa Grinder	40.623967	-74.030863	Juice Bar
4	Bay Ridge	40.625801	-74.030621	Pegasus Cafe	40.623168	-74.031186	Breakfast Spot

3. Filter out all the venues that are Italian restaurants.

   Italian restaurant is one of the 288 venues present in Brooklyn.

   We will do one hot encoding for getting dummies of the venue category. So that we can calculate the mean of all the venue groups by their neighborhoods.

```
[49] brooklyn_grouped = brooklyn_onehot.groupby('Neighborhood').mean().reset_index()
     print(brooklyn_grouped.shape)
     brooklyn_grouped.head()
```

(70, 288)

	Neighborhood	Accessories Store	American Restaurant	Antique Shop	Arepa Restaurant	Argentinian Restaurant	Art Gallery	Arts & Crafts Store	Arts & Entertainment	Asian Restaurant	Athletics & Sports	BBQ Joint	Bagel Shop	Bakery	Bank	Bar	Baseball Field	Baseball Stadium	Basketball Court	Beach	Beer Bar
0	Bath Beach	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.020833	0.000	0.0	0.020833	0.020833	0.020833	0.000000	0.000	0.0	0.0	0.0	0.0
1	Bay Ridge	0.0	0.037037	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.000	0.0	0.049383	0.000000	0.000000	0.037037	0.000	0.0	0.0	0.0	0.0
2	Bedford Stuyvesant	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.000	0.0	0.033333	0.000000	0.000000	0.066667	0.000	0.0	0.0	0.0	0.0
3	Bensonhurst	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.031250	0.000	0.0	0.031250	0.031250	0.000000	0.000000	0.000	0.0	0.0	0.0	0.0
4	Bergen Beach	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.125	0.0	0.000000	0.000000	0.000000	0.000000	0.125	0.0	0.0	0.0	0.0

   Now we will extract only the Neighborhood and Italian Restaurant column for further analysis.

```
[46] brooklyn_grouped_italian = brooklyn_grouped[['Neighborhood', 'Italian Restaurant']]
     brooklyn_grouped_italian.head()
```

	Neighborhood	Italian Restaurant
0	Bath Beach	0.041667
1	Bay Ridge	0.061728
2	Bedford Stuyvesant	0.033333
3	Bensonhurst	0.062500
4	Bergen Beach	0.000000

```
[47] brooklyn_grouped_clustering = brooklyn_grouped_italian.drop('Neighborhood', 1)
     brooklyn_grouped_clustering.head()
```

	Italian Restaurant
0	0.041667
1	0.061728
2	0.033333
3	0.062500
4	0.000000

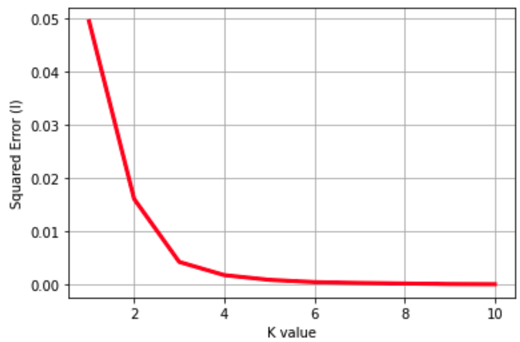
4. Analyzing the data using K-means Clustering and visualizing the neighborhoods with the number of Italian restaurants present.

   Calculated the best value of K in order to cluster the neighborhoods and then visualize them.

   As the K value is 3, we'll be clustering the neighborhoods of Brooklyn into 3 different clusters say Cluster 0 ,1, and 2.

   Later we'll examine each and every cluster and discuss the results accordingly.

```
[34] l = []
     for i in range(1, 11):
       k = KMeans(n_clusters = i, max_iter = 500)
       k.fit(brooklyn_grouped_clustering)
       l.append(k.inertia_)

     plt.plot(range(1, 11), l, color = 'r', linewidth = '3')
     plt.xlabel("K value")
     plt.ylabel("Squared Error (l)")
     plt.grid()
     plt.show()
```



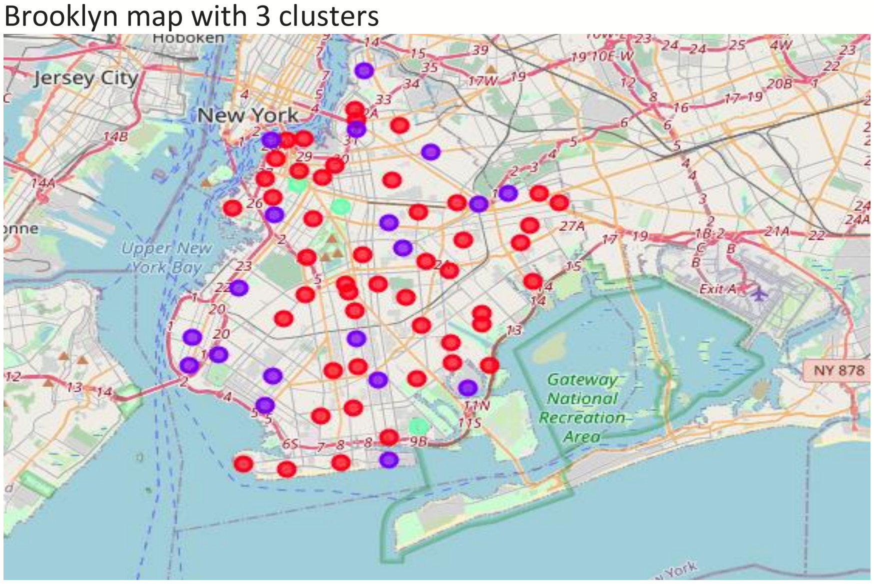
Merged the brooklyn_data and the brooklyn_grouped_italian on Neighborhood.

```
[ ] # add clustering labels
    brooklyn_data.insert(0, 'Cluster Labels', kmeans.labels_)

    brooklyn_merged = brooklyn_data

    # merge toronto_grouped with toronto_data to add latitude/longitude for each neighborhood
    brooklyn_merged = brooklyn_grouped_italian.join(brooklyn_merged.set_index('Neighborhood'), on='Neighborhood')

    brooklyn_merged.head() # check the last columns!
```

	Neighborhood	Italian Restaurant	Cluster Labels	Borough	Latitude	Longitude
0	Bath Beach	0.041667	2	Brooklyn	40.599519	-73.998752
1	Bay Ridge	0.061728	2	Brooklyn	40.625801	-74.030621
2	Bedford Stuyvesant	0.033333	0	Brooklyn	40.687232	-73.941785
3	Bensonhurst	0.062500	2	Brooklyn	40.611009	-73.995180
4	Bergen Beach	0.000000	0	Brooklyn	40.615150	-73.898556

Created a map for Brooklyn with 3 cluster of neighborhoods using the folium library Map function.

Brooklyn map with 3 clusters



Now let's separate each cluster data and look into it

Cluster 0

```
[40] brooklyn_merged.loc[brooklyn_merged['Cluster Labels'] == 0] # Cluster 0
```

	Neighborhood	Italian Restaurant	Cluster Labels	Borough	Latitude	Longitude
2	Bedford Stuyvesant	0.033333	0	Brooklyn	40.687232	-73.941785
4	Bergen Beach	0.000000	0	Brooklyn	40.615150	-73.898556
6	Borough Park	0.000000	0	Brooklyn	40.633131	-73.990498
7	Brighton Beach	0.000000	0	Brooklyn	40.576825	-73.965094
9	Brooklyn Heights	0.040000	0	Brooklyn	40.695864	-73.993782
10	Brownsville	0.000000	0	Brooklyn	40.663950	-73.910235
12	Canarsie	0.000000	0	Brooklyn	40.635564	-73.902093
13	Carroll Gardens	0.110000	0	Brooklyn	40.680540	-73.994654
14	City Line	0.000000	0	Brooklyn	40.678570	-73.867976
15	Clinton Hill	0.051546	0	Brooklyn	40.693229	-73.967843
16	Cobble Hill	0.031250	0	Brooklyn	40.687920	-73.998561
17	Coney Island	0.000000	0	Brooklyn	40.574293	-73.988683
19	Cypress Hills	0.000000	0	Brooklyn	40.682391	-73.876616
20	Ditmas Park	0.000000	0	Brooklyn	40.643675	-73.961013
21	Downtown	0.010000	0	Brooklyn	40.690844	-73.983463
22	Dumbo	0.033333	0	Brooklyn	40.703176	-73.988753
24	East Flatbush	0.000000	0	Brooklyn	40.641718	-73.936103
25	East New York	0.000000	0	Brooklyn	40.669926	-73.880699
26	East Williamsburg	0.000000	0	Brooklyn	40.708492	-73.938858
27	Erasmus	0.000000	0	Brooklyn	40.646926	-73.948177
28	Flatbush	0.000000	0	Brooklyn	40.636326	-73.958401
29	Flatlands	0.000000	0	Brooklyn	40.630446	-73.929113
30	Fort Greene	0.045455	0	Brooklyn	40.688527	-73.972906
33	Georgetown	0.034483	0	Brooklyn	40.623845	-73.916075
36	Gravesend	0.115385	0	Brooklyn	40.595260	-73.973471
39	Homecrest	0.000000	0	Brooklyn	40.598525	-73.959185
40	Kensington	0.000000	0	Brooklyn	40.642382	-73.980421

Cluster 2

```
[42] brooklyn_merged.loc[brooklyn_merged['Cluster Labels'] == 2] # Cluster 2
```

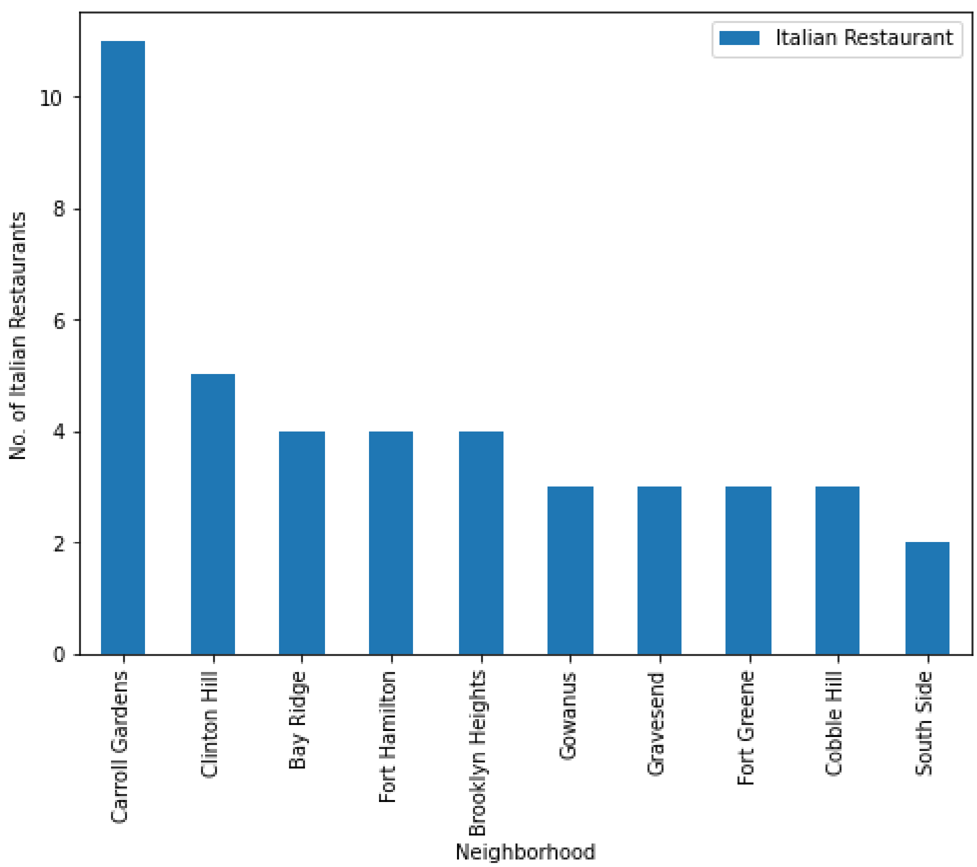
	Neighborhood	Italian Restaurant	Cluster Labels	Borough	Latitude	Longitude
5	Boerum Hill	0.011111	2	Brooklyn	40.685683	-73.983748
34	Gerritsen Beach	0.000000	2	Brooklyn	40.590848	-73.930102
54	Prospect Heights	0.000000	2	Brooklyn	40.676822	-73.964859

Cluster 1

```
[41] brooklyn_merged.loc[brooklyn_merged['Cluster Labels'] == 1] # Cluster 1
```

	Neighborhood	Italian Restaurant	Cluster Labels	Borough	Latitude	Longitude
0	Bath Beach	0.041667	1	Brooklyn	40.599519	-73.998752
1	Bay Ridge	0.049383	1	Brooklyn	40.625801	-74.030621
3	Bensonhurst	0.066667	1	Brooklyn	40.611009	-73.995180
8	Broadway Junction	0.000000	1	Brooklyn	40.677861	-73.903317
11	Bushwick	0.014493	1	Brooklyn	40.698116	-73.925258
18	Crown Heights	0.000000	1	Brooklyn	40.670829	-73.943291
23	Dyker Heights	0.000000	1	Brooklyn	40.619219	-74.019314
31	Fort Hamilton	0.064516	1	Brooklyn	40.614768	-74.031979
32	Fulton Ferry	0.016949	1	Brooklyn	40.703281	-73.995508
35	Gowanus	0.049180	1	Brooklyn	40.673931	-73.994441
37	Greenpoint	0.010000	1	Brooklyn	40.730201	-73.954241
38	Highland Park	0.000000	1	Brooklyn	40.681999	-73.890346
41	Madison	0.100000	1	Brooklyn	40.609378	-73.948415
42	Manhattan Beach	0.000000	1	Brooklyn	40.577914	-73.943537
45	Midwood	0.000000	1	Brooklyn	40.625596	-73.957595
47	Mill Island	0.000000	1	Brooklyn	40.606336	-73.908186
64	Sunset Park	0.028571	1	Brooklyn	40.645103	-74.010316
67	Williamsburg	0.030303	1	Brooklyn	40.707144	-73.958115
69	Wingate	0.000000	1	Brooklyn	40.660947	-73.937187

Bar graph to visualize the number of Italian restaurants located in each neighborhood of Brooklyn



Results

1. Carroll Gardens neighborhood has the highest number of Italian restaurants.
2. Bay Ridge neighborhood has a high density of Italian restaurants.
3. I will open the restaurant in Gerriston Beach. As it'll become a beachside restaurant and there is also a shopping outlet within a range of 1mi which leads to more profits sooner.

Discussion

According to the analysis, Gerriston Beach will provide the least competition for an upcoming Italian restaurant as there is a shopping mall close to this neighborhood. And it's going to be a beachside restaurant where people would like to explore, try something new, and would like to have more options handy. So, all this is the best place for all of those who are interested in getting a taste of the Italian Cuisine and also, the frequency of Italian restaurants is very low compared to the other neighborhoods. Carroll Gardens has the highest number of Italian restaurants and Bay Ridge is highly dense so, we will not open there. The analysis I did is completely relied on the data provided by Foursquare API and using K-means clustering. There are a number of factors such as the number of customers, land value, distance that play a major role in stating that this analysis is far from being conclusory. However, it definitely gives us some very important preliminary information on the possibilities of opening restaurants in the Brooklyn borough of New York City. And the results might vary if we had used some other clustering techniques like DBSCAN.

Conclusion

Finally, to conclude this project, we have got a small glimpse of how a real-life Data science project looks like. I have used some frequently used python libraries to handle loading the JSON file, plotting graphs, and performing other exploratory data analysis. Used Foursquare API to major boroughs of New York City and their neighborhoods. Potential for this kind of analysis in a real-life business problem is discussed in great detail. As a final note, all of the above analysis is based on the Foursquare data. A more comprehensive analysis and future work would need to incorporate data from other external resources.