BATTLE OF NEIGHBORHOODS - LONDON

Coursera Capstone Project

Contents

Business Problem:

- A successful Asian restaurant chain is looking to expand its operations through London. We were asked to identify and recommend the neighborhoods in London that will be good choice to start an Asian restaurant.

Target Audience:

- Companies that are looking to invest in food service industry of London.
- Individuals looking to relocate neighborhoods in London with particular venues.

Data Collection and Cleaning:

- London areas and neighborhoods data was scraped for the Wikipedia page- https://en.wikipedia.org/wiki/List_of_areas_of_London

- Demographics of London was also Scraped from the Wikipedia page - https://en.wikipedia.org/wiki/Demography_of_London

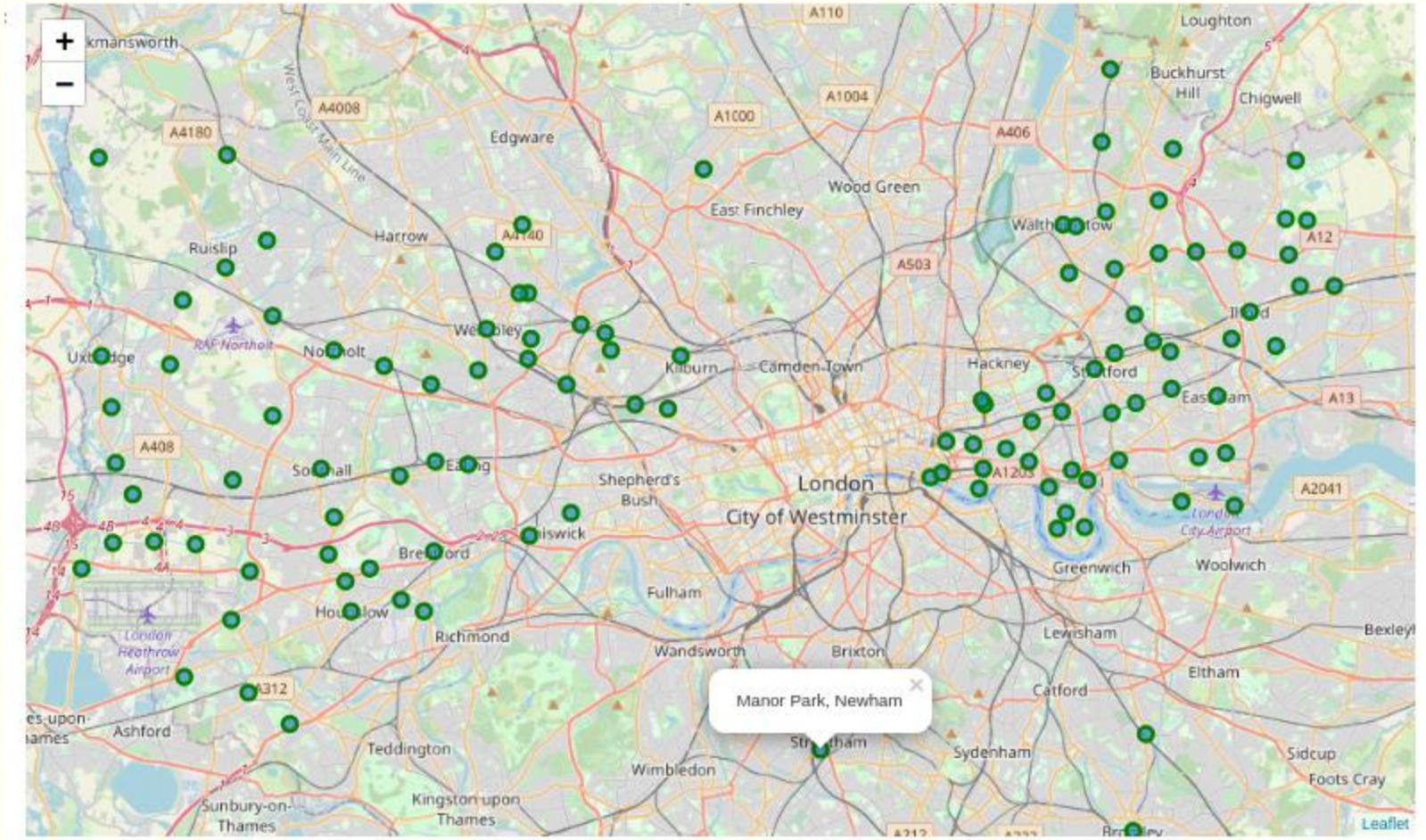
- The top-8 boroughs with highest Asian population are selected and neighborhood areas of those boroughs were only considered.

- Using Geopy library, latitudes and longitudes are added to each neighborhood.

- Cleaned data contain 6 features and 113 data points.

METHODOLOGY

Visualizing Neighborhoods:



Exploring Neighborhoods:

- URL is created to access the Foursquare API.
- Using GET method from requests library and the defined URL, nearby venues are fetched for each neighborhood.
- The json file that is obtained is cleaned and structered into pandas dataframe.
- The resulted data frame has 4711 rows and 7 features that include Neighborhood, Neighborhood Latitude, Neighborhood Longitude, Venue, Venue Latitude, Venue Longitude and Venue category.

One-hot Encoding:

- One-Hot encoding is applied to 'Venue Category' column in london_venues dataframe to convert categorical variables to integers.
- The dataframe is Grouped by neighborhood and by taking the mean of the frequency of occurrence of each category will result in the following dataframe.

```
In [56]: london_grouped = london_onehot.groupby('Neighborhood').mean().reset_index()
         london_grouped
```

	Neighborhood	Accessories Store	Afghan Restaurant	Airport	Airport Lounge	Airport Service	Airport Terminal	American Restaurant	Antique Shop	Arepa Restaurant	Argentinian Restaurant	Art Gallery	Arts & Crafts Store	Asian Restaurant
0	Aldborough Hatch	0.00	0.00	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.000000	0.000000	0.000000
1	Alperton	0.00	0.00	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.000000	0.000000	0.058824
2	Barkingside	0.00	0.00	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.000000	0.000000	0.000000
3	Beckton	0.00	0.00	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.000000	0.000000	0.000000
4	Bedford Park	0.00	0.00	0.000000	0.000000	0.000000	0.000000	0.000000	0.010000	0.00	0.010000	0.000000	0.000000	0.000000
5	Bethnal Green	0.00	0.00	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.01	0.010000	0.010000	0.000000	0.000000
6	Blackwall	0.00	0.00	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.000000	0.000000	0.010000
7	Bow	0.00	0.00	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.021739	0.000000	0.000000
8	Brent Park	0.00	0.00	0.000000	0.000000	0.000000	0.000000	0.026316	0.000000	0.00	0.000000	0.000000	0.000000	0.013158

Grouping and Statistics:

- Exploring the top 5 most common venues for each neighborhood

```
num_top_venues = 5

for hood in london_grouped['Neighborhood']:
    print("----"+hood+"----")
    temp = london_grouped[london_grouped['Neighborhood'] == hood].T.reset_index()
    temp.columns = ['venue','freq']
    temp = temp.iloc[1:]
    temp['freq'] = temp['freq'].astype(float)
    temp = temp.round({'freq': 2})
    print(temp.sort_values('freq', ascending=False).reset_index(drop=True).head(num_top_venues))
    print('\n')
```

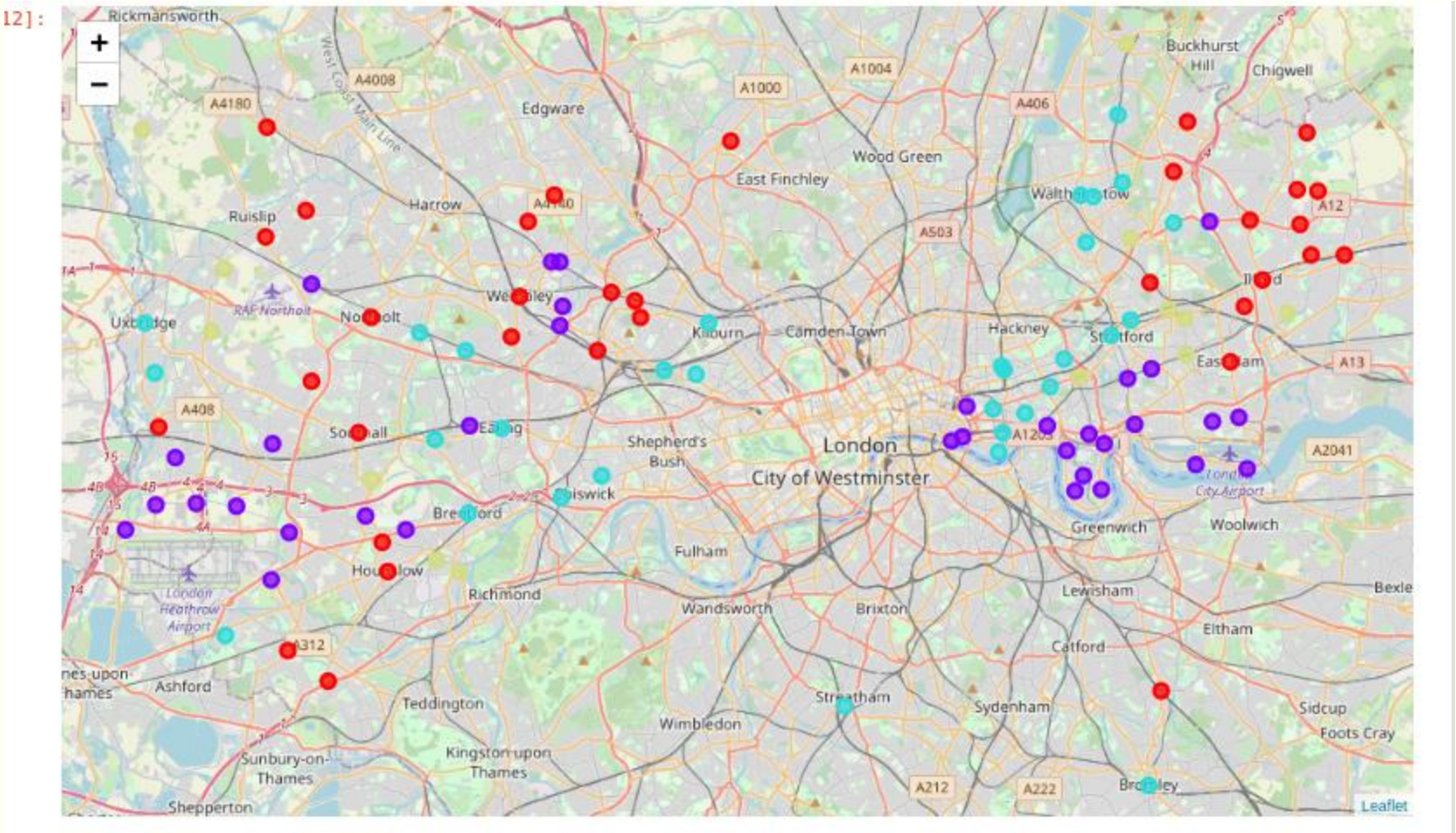
```
----Aldborough Hatch----
                venue  freq
0         Social Club  0.12
1  Sporting Goods Shop  0.12
2   Indian Restaurant  0.12
3        Soccer Field  0.12
4       Metro Station  0.12


----Alperton----
                venue  freq
0   Indian Restaurant  0.12
1         Supermarket  0.12
2  Gym / Fitness Center  0.12
3       Clothing Store  0.06
4                Café  0.06
```

Clustering:

- K-means Clustering is used to cluster the neighborhoods into four clusters depending the availability of venues nearby.
- The optimal value for clusters is decided using Elbow method. The algorithm has global optimum at k=4.
- The cluster labels are added back to the dataframe for easy identification.
- Now, each cluster is examined along with their top-10 most frequent venues and suitable cluster is identified for the opening the restaurant.

Visualizing Clusters:

12]:



Results:

- In Cluster-0, It is observed that Asian cuisines like Indian, Thai and Turkish restaurants are among the top-10 of most common venues in the almost every neighborhood. Hence, would not be an ideal place to start the business.

- In Cluster-1, coffee shops and Hotels are among the most common venues.

- Pubs, Cafes and Parks are the most common venues in the neighborhoods of Cluster-2.

- Pubs are the most common venue in almost all the neighborhoods in Cluster-3, with Asian restaurants appearing often.

Discussion and Conclusion:

- From this analysis, Neighborhoods in Cluster-1 and Cluster-2 are identified as viable to open a new Asian restaurant.
- However, only demographics data and venues data are used to cluster the neighborhoods.
- This model could be improved by considering other information like ease of public access, locations visibility, Competitors rating, etc.