THE BATTLE OF NEIGHBORHOODS

Introduction

Description of the Problem

- The population of London grew considerably over the last decades and in consequence grew very diverse. In London, as you can get fresh food supplies from all over the world, one begins to wonder how efficient the supply mechanism is.
- The real deal is that as much as there are many fine restaurants in London – Asian, Middle Eastern, Latin and American restaurants, you can struggle to find a good place to dine in the finest of West African cuisine that has a combination of Nigerian, Ghanaian, Cameroonian, Senegalese and more.

Discussion of the Background

- Our client, a successful restaurant chain in Africa is looking to expand operation in Europe through London. They want to create a high-end restaurant that comes with organic mix and healthy. Their target is not only West Africans, but they are pro-organic and healthy eating. To them, the client should be served as a king.
- Since the London demography is so big, our client needs deeper insight from available data in order to decide where to establish the first Europe "royal" restaurant. This company spends a lot on research and provides customers with data insight into the ingredients used at restaurants.

Target Audience

- The target audience is broad, it ranges from Londoners, tourists and those who are passionate about organic food.

Data

Description of Data

- This project will rely on public data from Wikipedia and Foursquare.
- Considering the diversity of London, there is a high multicultural sense. London is a place where different shades live. As such, in the search for a high-end African-inclined restaurant, there is a high shortage.
- The London Area consists of 32 Boroughs and the "City of London". Our data will be from the link: List of areas of London

Description of Data

- The web scrapped of the Wikipedia page for the Greater London Area data is provided below:
- The ***BeautifulSoup*** package is used to scrap the needed data from Wikipedia.

```
# library for BeautifulSoup
from bs4 import BeautifulSoupwikipedia_link = 'https://en.wikipedia.org/wiki/List_of_areas_of_London'
wikipedia_page = requests.get(wikipedia_link)
```

Description of Data

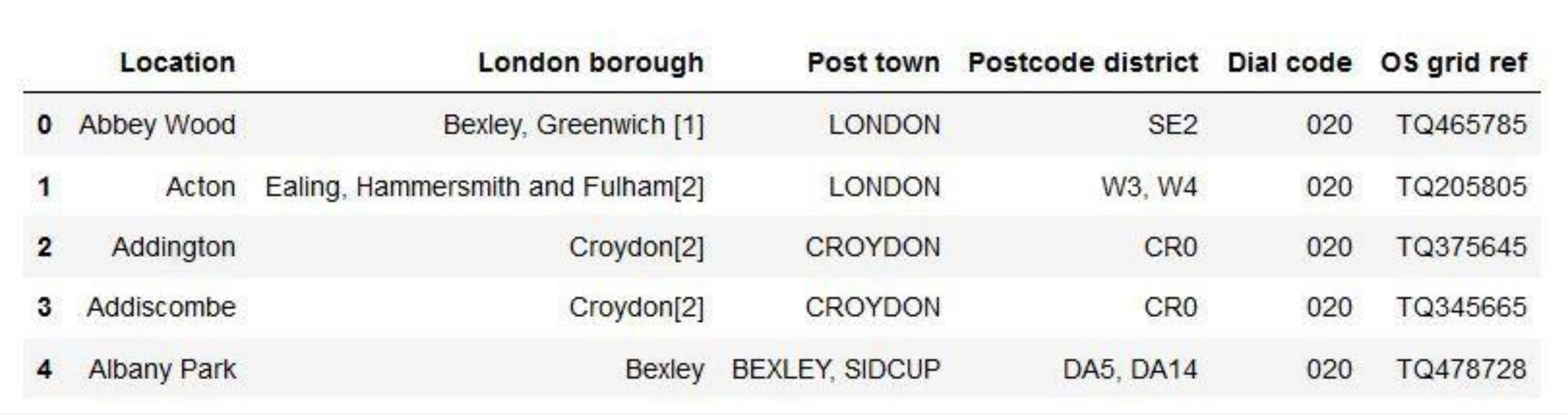
◦ Then the html is cleaned and parsed accordingly.

```
# Cleans html file
soup = BeautifulSoup(wikipedia_page.content, 'html.parser')# This extracts the "tbody" within the table where class is "wikitable sortable"
table = soup.find('table', {'class':'wikitable sortable'}).tbody# Extracts all "tr" (table rows) within the table above
rows = table.find_all('tr')# Extracts the column headers, removes and replaces possible '\n' with space for the "th" tag
columns = [i.text.replace('\n', '')
for i in rows[0].find_all('th')]# Converts columns to pd dataframe
df = pd.DataFrame(columns = columns)'''
Extracts every row with corresponding columns then appends the values to the create pd dataframe "df". The first row (row[0]) is skipped because it is already the header
'''for i in range(1, len(rows)):
tds = rows[i].find_all('td')
if len(tds) == 7:
values = [tds[0].text, tds[1].text, tds[2].text.replace('\n', ''.replace('\xa0','')), tds[3].text, tds[4].text.replace('\n', ''.replace('\xa0','')), tds[5].text.replace('\n', ''.replace('\xa0','')), tds[6].text.replace('\n', 
''.replace('\xa0',''))]
else:
values = [td.text.replace('\n', '').replace('\xa0','') for td in tds]

df = df.append(pd.Series(values, index = columns), ignore_index = True)
df
```

Description of Data

- The result of the df.head(5) is shown below:

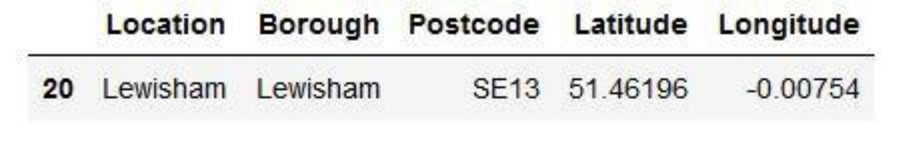


	Location	London borough	Post town	Postcode district	Dial code	OS grid ref
0	Abbey Wood	Bexley, Greenwich [1]	LONDON	SE2	020	TQ465785
1	Acton	Ealing, Hammersmith and Fulham[2]	LONDON	W3, W4	020	TQ205805
2	Addington	Croydon[2]	CROYDON	CR0	020	TQ375645
3	Addiscombe	Croydon[2]	CROYDON	CR0	020	TQ345665
4	Albany Park	Bexley	BEXLEY, SIDCUP	DA5, DA14	020	TQ478728

Methodology

Single Neighborhood

- Single Neighborhood — An initial exploration of a single Neighborhood within the London area was done to examine the Foursquare workability. The Lewisham Borough postcode SE13 and Location - Lewisham is used for this.



	Location	Borough	Postcode	Latitude	Longitude
20	Lewisham	Lewisham	SE13	51.46196	-0.00754

Single Neighborhood

- Let’s explore the top 100 venues that are within a 2000 metres radius of Lewisham. And then, let’s create the GET request URL, and then the url is named.

- Since there is a limit to Foursquare usage → https://developer.foursquare.com/docs/api/troubleshooting/rate-limits

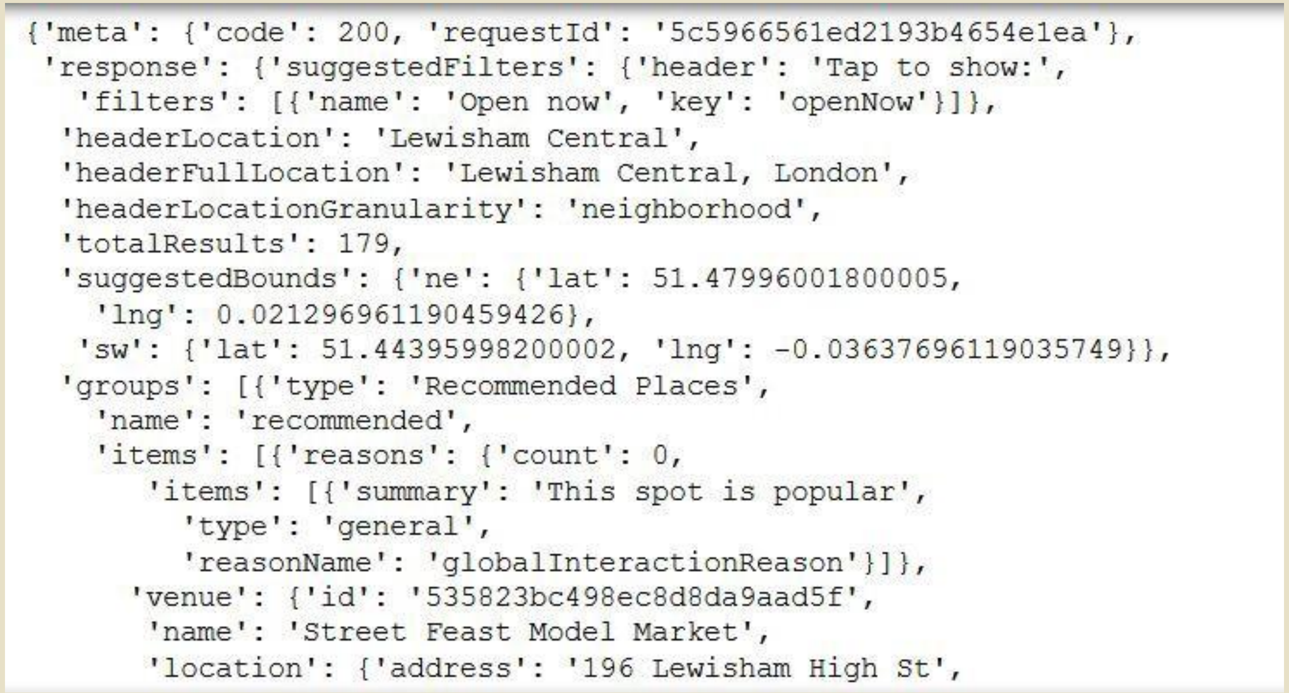
Single Neighborhood

- Let’s explore the top 100 venues that are within a 2000 meters radius of Lewisham. And then, let’s create the GET request URL, and then the URL is named.

```
LIMIT = 100 # limit of number of venues returned by Foursquare API
radius = 2000 # define radius
url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
    CLIENT_ID,
    CLIENT_SECRET,
    VERSION,
    lewisham_lat,
    lewisham_long,
    radius,
    LIMIT)
# displays URL
url
```

Single Neighborhood

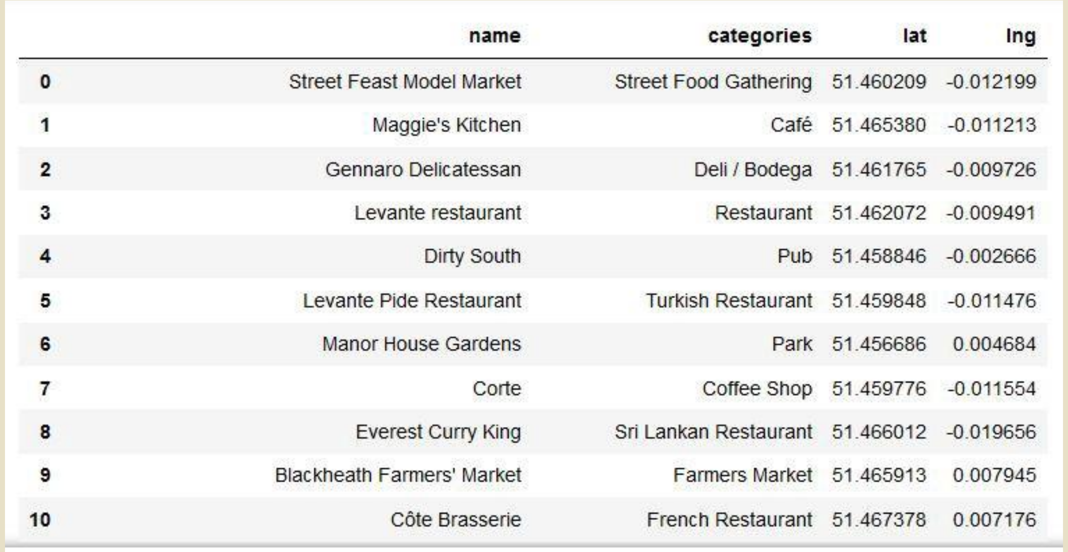
- Then, send the GET request and examine the results.



```
{'meta': {'code': 200, 'requestId': '5c5966561ed2193b4654e1ea'},
 'response': {'suggestedFilters': {'header': 'Tap to show:',
   'filters': [{'name': 'Open now', 'key': 'openNow'}]},
  'headerLocation': 'Lewisham Central',
  'headerFullLocation': 'Lewisham Central, London',
  'headerLocationGranularity': 'neighborhood',
  'totalResults': 179,
  'suggestedBounds': {'ne': {'lat': 51.47996001800005,
    'lng': 0.021296961190459426},
   'sw': {'lat': 51.44395998200002, 'lng': -0.03637696119035749}},
  'groups': [{'type': 'Recommended Places',
    'name': 'recommended',
    'items': [{'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '535823bc498ec8d8da9aad5f',
       'name': 'Street Feast Model Market',
       'location': {'address': '196 Lewisham High St',
```

Single Neighborhood

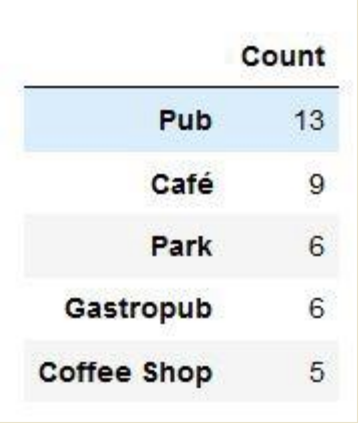
- From the results, the necessary information needs to be obtained from items key. To do this, the get_category_type function is used from the Foursquare lab.
- The result is then cleaned up from json to a structured pandas dataframe.



	name	categories	lat	lng
0	Street Feast Model Market	Street Food Gathering	51.460209	-0.012199
1	Maggie's Kitchen	Café	51.465380	-0.011213
2	Gennaro Delicatessan	Deli / Bodega	51.461765	-0.009726
3	Levante restaurant	Restaurant	51.462072	-0.009491
4	Dirty South	Pub	51.458846	-0.002666
5	Levante Pide Restaurant	Turkish Restaurant	51.459848	-0.011476
6	Manor House Gardens	Park	51.456686	0.004684
7	Corte	Coffee Shop	51.459776	-0.011554
8	Everest Curry King	Sri Lankan Restaurant	51.466012	-0.019656
9	Blackheath Farmers' Market	Farmers Market	51.465913	0.007945
10	Côte Brasserie	French Restaurant	51.467378	0.007176

Single Neighborhood

- Lewisham top 5 venues:

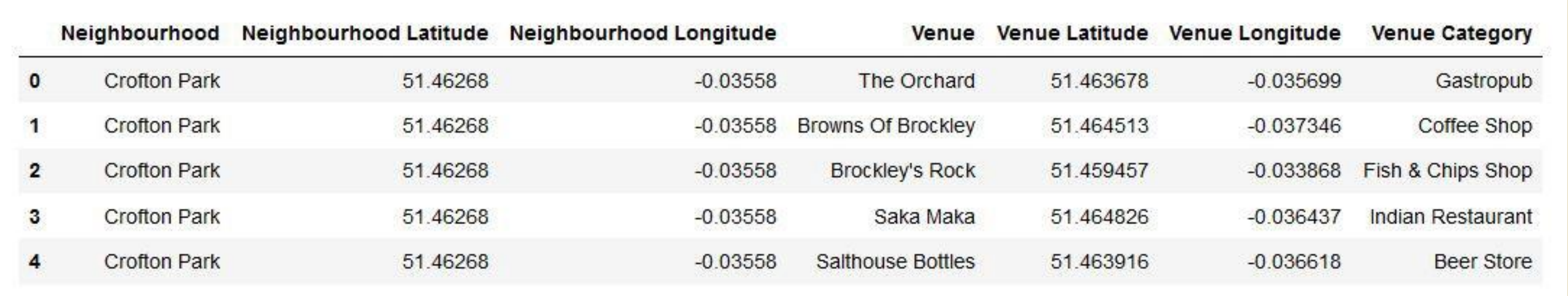


	Count
Pub	13
Café	9
Park	6
Gastropub	6
Coffee Shop	5

- Interestingly, even though there are restaurants are the Lewisham area, they are not even in the top 5 venues. It should be noted that since we are limited by data availability, our perspectives will be on what we have.

Multiple Neighborhoods

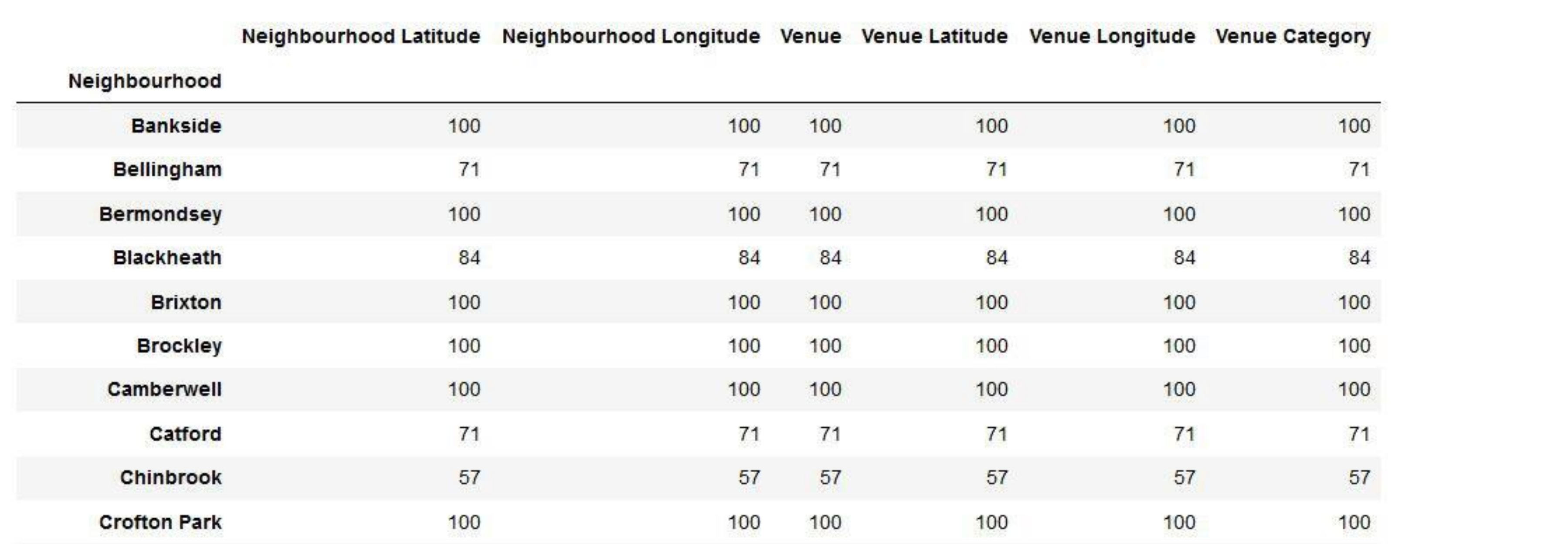
- Multiple Neighborhoods — Now let's explore (Multiple) Neighborhoods in the South East London area. To do this, the function getNearbyVenues is used and it's created to repeat the same process for all neighborhoods.
- The created function — getNearbyVenues is then used on each neighbourhoods. And creates a new dataframe called london_venues.
- A list of the venues is provided after running this. se_venues.shape is (4255, 7).



	Neighbourhood	Neighbourhood Latitude	Neighbourhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Crofton Park	51.46268	-0.03558	The Orchard	51.463678	-0.035699	Gastropub
1	Crofton Park	51.46268	-0.03558	Browns Of Brockley	51.464513	-0.037346	Coffee Shop
2	Crofton Park	51.46268	-0.03558	Brockley's Rock	51.459457	-0.033868	Fish & Chips Shop
3	Crofton Park	51.46268	-0.03558	Saka Maka	51.464826	-0.036437	Indian Restaurant
4	Crofton Park	51.46268	-0.03558	Salthouse Bottles	51.463916	-0.036618	Beer Store

Multiple Neighborhoods

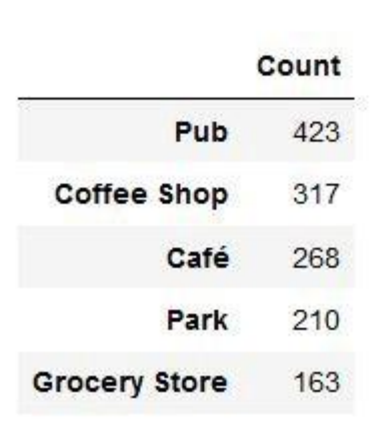
- The number of venues returned for each neighborhoods is then explored as follows:
- se_venues.groupby('Neighbourhood').count()



	Neighbourhood Latitude	Neighbourhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Neighbourhood						
Bankside	100	100	100	100	100	100
Bellingham	71	71	71	71	71	71
Bermondsey	100	100	100	100	100	100
Blackheath	84	84	84	84	84	84
Brixton	100	100	100	100	100	100
Brockley	100	100	100	100	100	100
Camberwell	100	100	100	100	100	100
Catford	71	71	71	71	71	71
Chinbrook	57	57	57	57	57	57
Crofton Park	100	100	100	100	100	100

Multiple Neighborhoods

- Then we discover that there are 186 unique categories and check the top 5 total venue categories



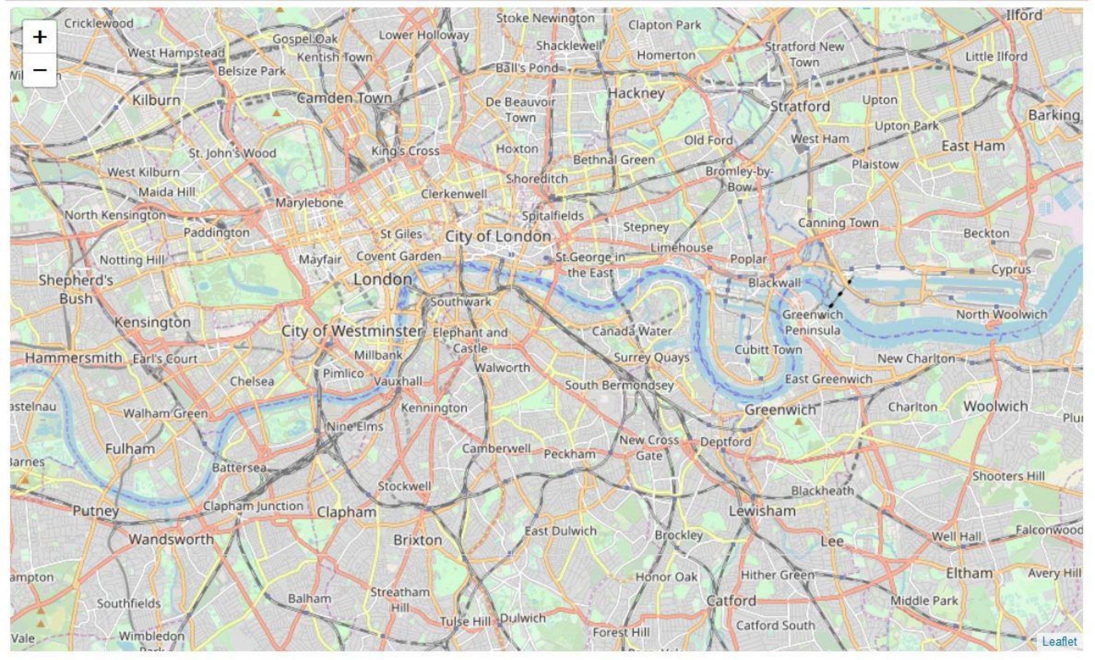
	Count
Pub	423
Coffee Shop	317
Café	268
Park	210
Grocery Store	163

Map Visualization

- For this section, the neighborhoods in South East London will be clustered based on the processed data obtained above.
- Using the geopy library, the latitude and longitude values of London is obtained: 51.5073219, -0.1276474.

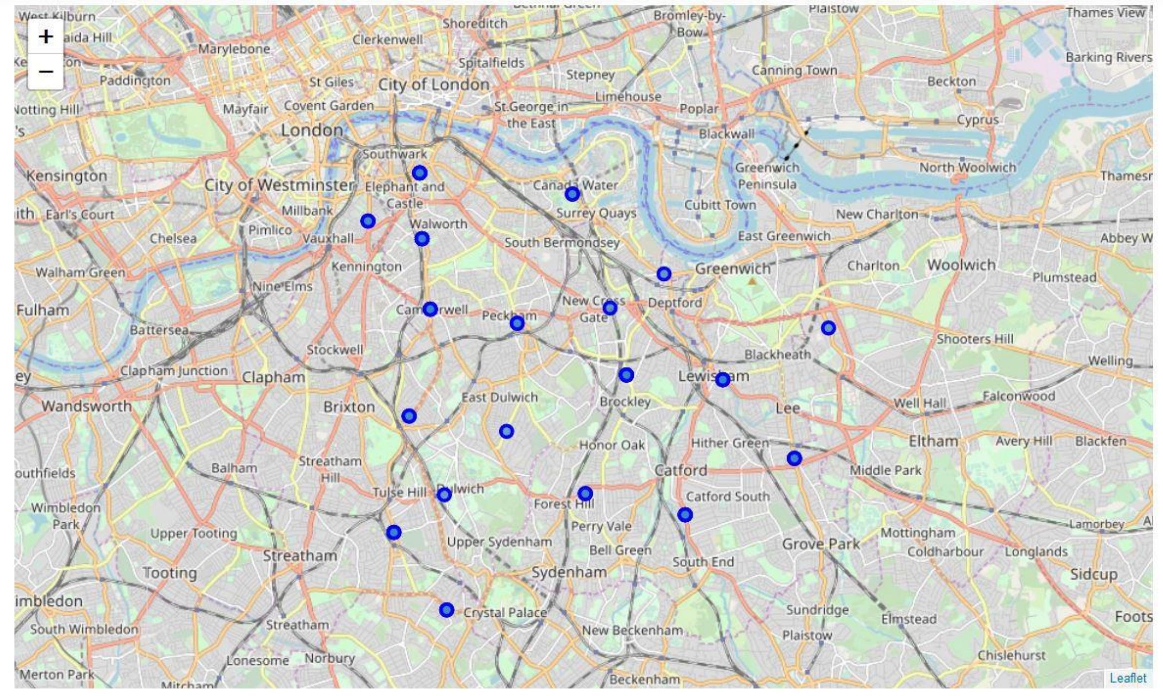
Map Visualization

- The folium library is then used to obtain the coordinates.



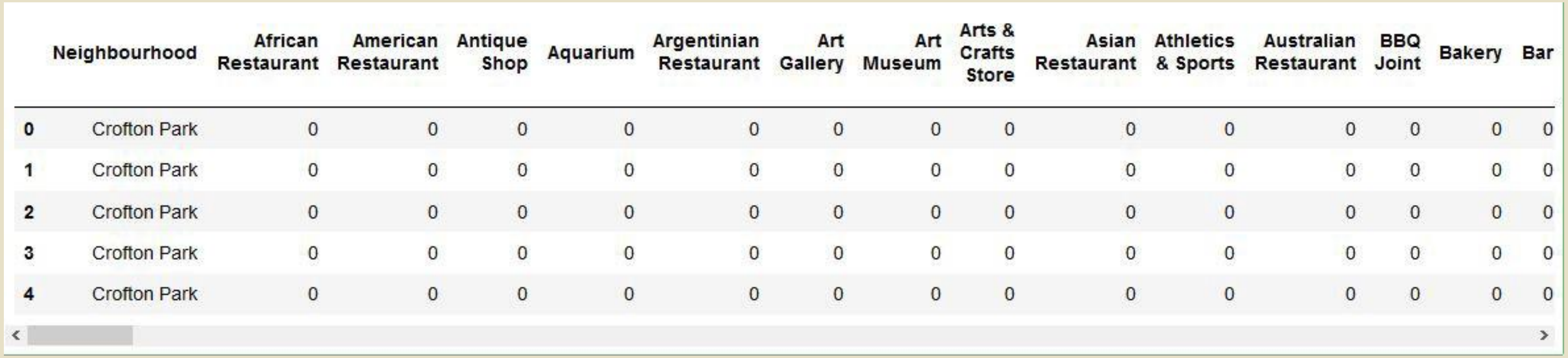
Map Visualization

- The South East London neighborhoods are then superimposed on top as shown below, still using the folium library.



Analyzing Each Neighborhood

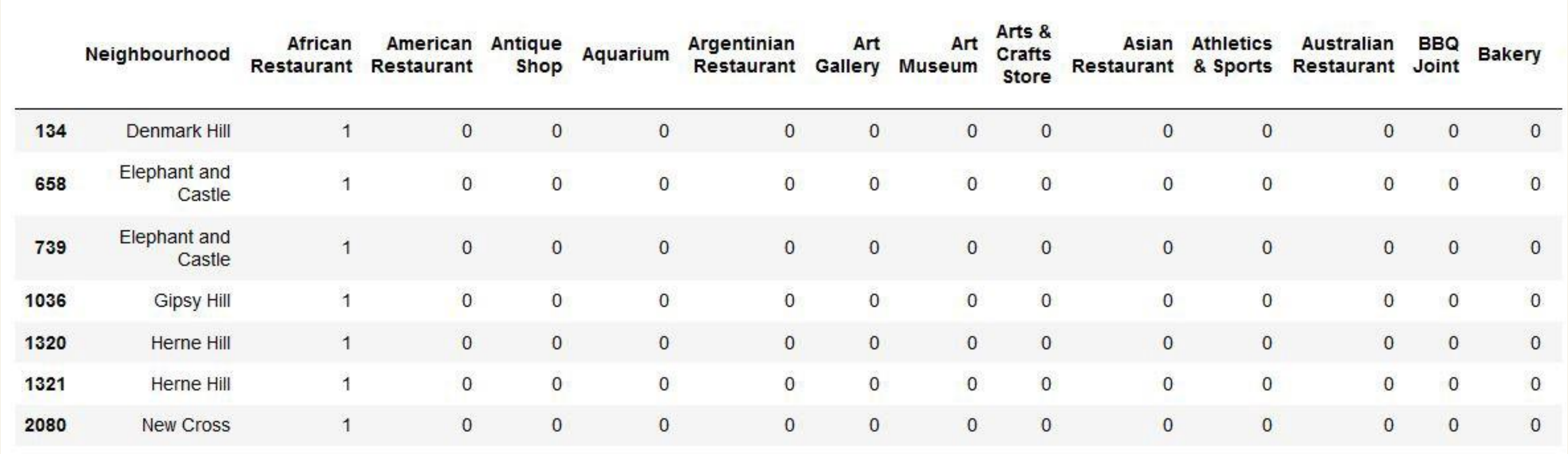
- In this section, the objective is to check and explore the venues in each neighborhood with One Hot Encoding.



	Neighbourhood	African Restaurant	American Restaurant	Antique Shop	Aquarium	Argentinian Restaurant	Art Gallery	Art Museum	Arts & Crafts Store	Asian Restaurant	Athletics & Sports	Australian Restaurant	BBQ Joint	Bakery	Bar
0	Crofton Park	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	Crofton Park	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	Crofton Park	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	Crofton Park	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	Crofton Park	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Analyzing Each Neighborhood

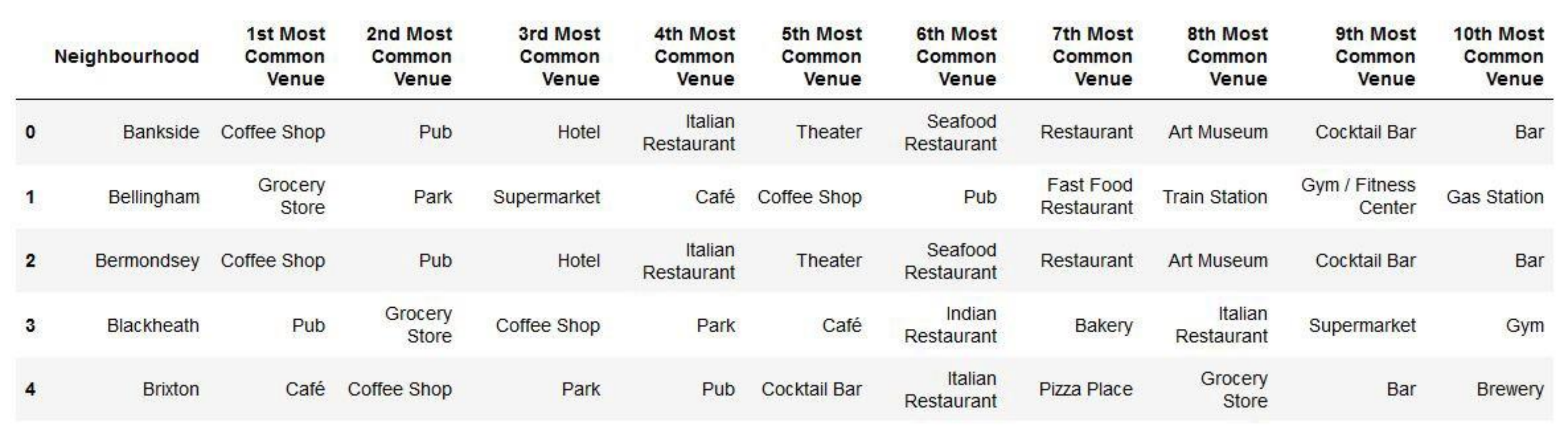
◦ Checking only the African Restaurants:



	Neighbourhood	African Restaurant	American Restaurant	Antique Shop	Aquarium	Argentinian Restaurant	Art Gallery	Art Museum	Arts & Crafts Store	Asian Restaurant	Athletics & Sports	Australian Restaurant	BBQ Joint	Bakery
134	Denmark Hill	1	0	0	0	0	0	0	0	0	0	0	0	0
658	Elephant and Castle	1	0	0	0	0	0	0	0	0	0	0	0	0
739	Elephant and Castle	1	0	0	0	0	0	0	0	0	0	0	0	0
1036	Gipsy Hill	1	0	0	0	0	0	0	0	0	0	0	0	0
1320	Herne Hill	1	0	0	0	0	0	0	0	0	0	0	0	0
1321	Herne Hill	1	0	0	0	0	0	0	0	0	0	0	0	0
2080	New Cross	1	0	0	0	0	0	0	0	0	0	0	0	0

Analyzing Each Neighborhood

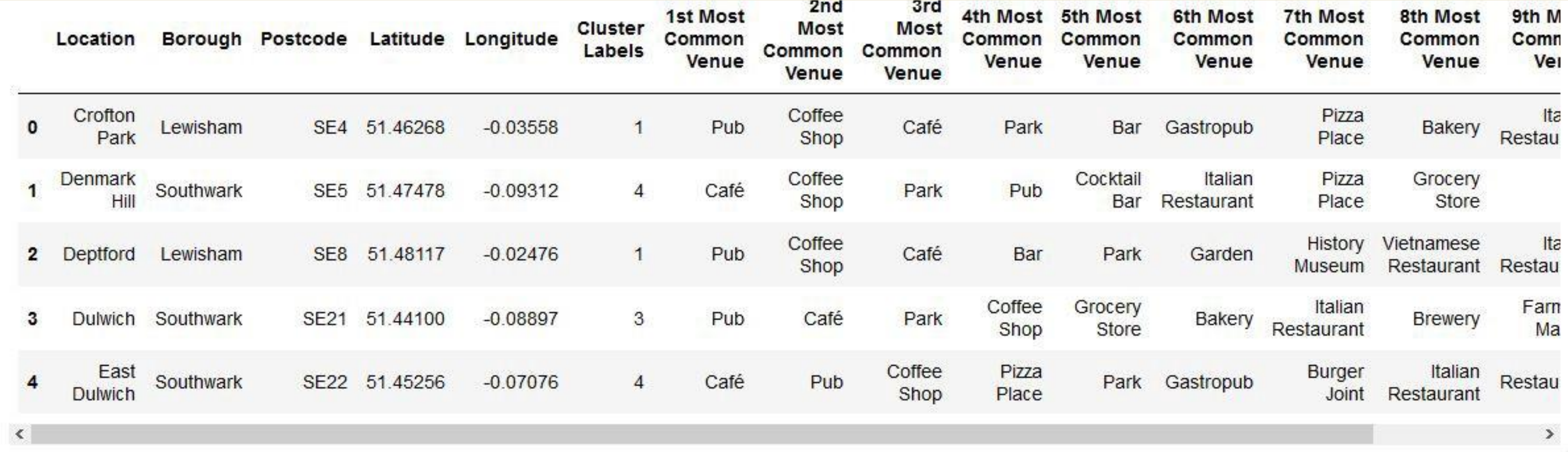
- Then we create a new panda dataframe with 10 most common venues as shown below:



	Neighbourhood	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	Bankside	Coffee Shop	Pub	Hotel	Italian Restaurant	Theater	Seafood Restaurant	Restaurant	Art Museum	Cocktail Bar	Bar
1	Bellingham	Grocery Store	Park	Supermarket	Café	Coffee Shop	Pub	Fast Food Restaurant	Train Station	Gym / Fitness Center	Gas Station
2	Bermondsey	Coffee Shop	Pub	Hotel	Italian Restaurant	Theater	Seafood Restaurant	Restaurant	Art Museum	Cocktail Bar	Bar
3	Blackheath	Pub	Grocery Store	Coffee Shop	Park	Café	Indian Restaurant	Bakery	Italian Restaurant	Supermarket	Gym
4	Brixton	Café	Coffee Shop	Park	Pub	Cocktail Bar	Italian Restaurant	Pizza Place	Grocery Store	Bar	Brewery

Analyzing Each Neighborhood

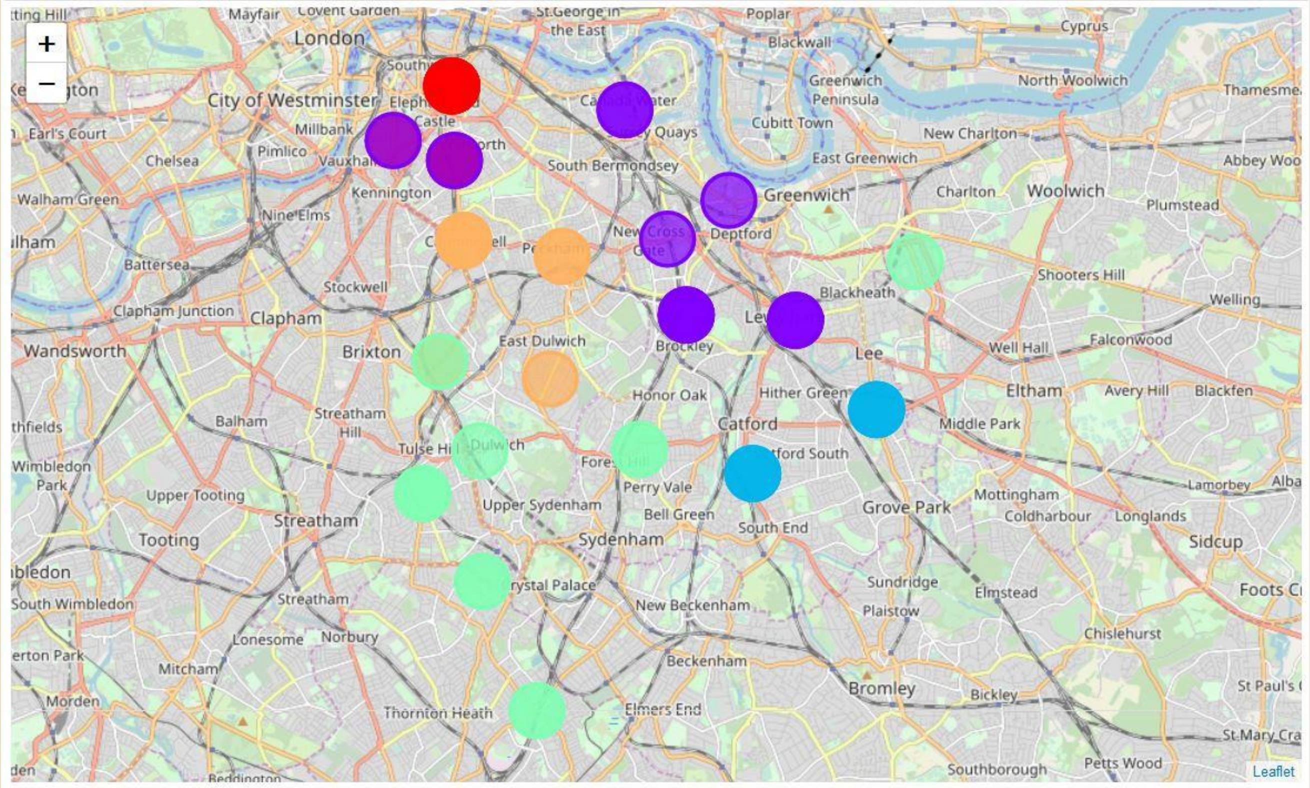
- We create the grouped clustering for the neighborhood and then create clusters of the neighborhood using the k-means to cluster the neighborhood into 5 clusters (this was decided after checking the Elbow Method and Silhouette Coefficients).



	Location	Borough	Postcode	Latitude	Longitude	Cluster Labels	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 M Comn Ve
0	Crofton Park	Lewisham	SE4	51.46268	-0.03558	1	Pub	Coffee Shop	Café	Park	Bar	Gastropub	Pizza Place	Bakery	Ita Restau
1	Denmark Hill	Southwark	SE5	51.47478	-0.09312	4	Café	Coffee Shop	Park	Pub	Cocktail Bar	Italian Restaurant	Pizza Place	Grocery Store	
2	Deptford	Lewisham	SE8	51.48117	-0.02476	1	Pub	Coffee Shop	Café	Bar	Park	Garden	History Museum	Vietnamese Restaurant	Ita Restau
3	Dulwich	Southwark	SE21	51.44100	-0.08897	3	Pub	Café	Park	Coffee Shop	Grocery Store	Bakery	Italian Restaurant	Brewery	Farm Ma
4	East Dulwich	Southwark	SE22	51.45256	-0.07076	4	Café	Pub	Coffee Shop	Pizza Place	Park	Gastropub	Burger Joint	Italian Restaurant	Restau

Analyzing Each Neighborhood

- We then create a folium map using the clustered information:



Results and Conclusion

Results

- The following are the highlights of the 5 clusters above:
  - Pubs, Cafe, Coffee Shops are popular in the South East London.
  - As for restaurants, the Italian Restaurants are very popular in the South East London area. Especially in Southwark and Lambeth areas.
  - With the Lewisham area being the most condensed area of Africans in the South East Area, it is surprising to see how in the top 10 venues, you can barely see restaurants in the top 5 venues.
  - Although, the Clusters have variations, a very visible presence is the predominance of pubs.

Discussion and Conclusion

- It is very important to note that Clusters 2 and 3 are the most viable clusters to create a brand African Restaurant. Their proximity to other amenities and accessibility to station are paramount. These 2 clusters do not have top restaurants that could rival their standards if they are created. And the proximity to resources needed is paramount as Lewisham and Lambeth are not far out from Peckham (under Southwark).

- In conclusion, this project would have had better results if there were more data in terms of crime data within the area, traffic access and allowance of more venues exploration with the Foursquare (limited venues for free calls).

- Also, getting the ratings and feedbacks of the current restaurants within the clusters would have helped in providing more insight into the best location.