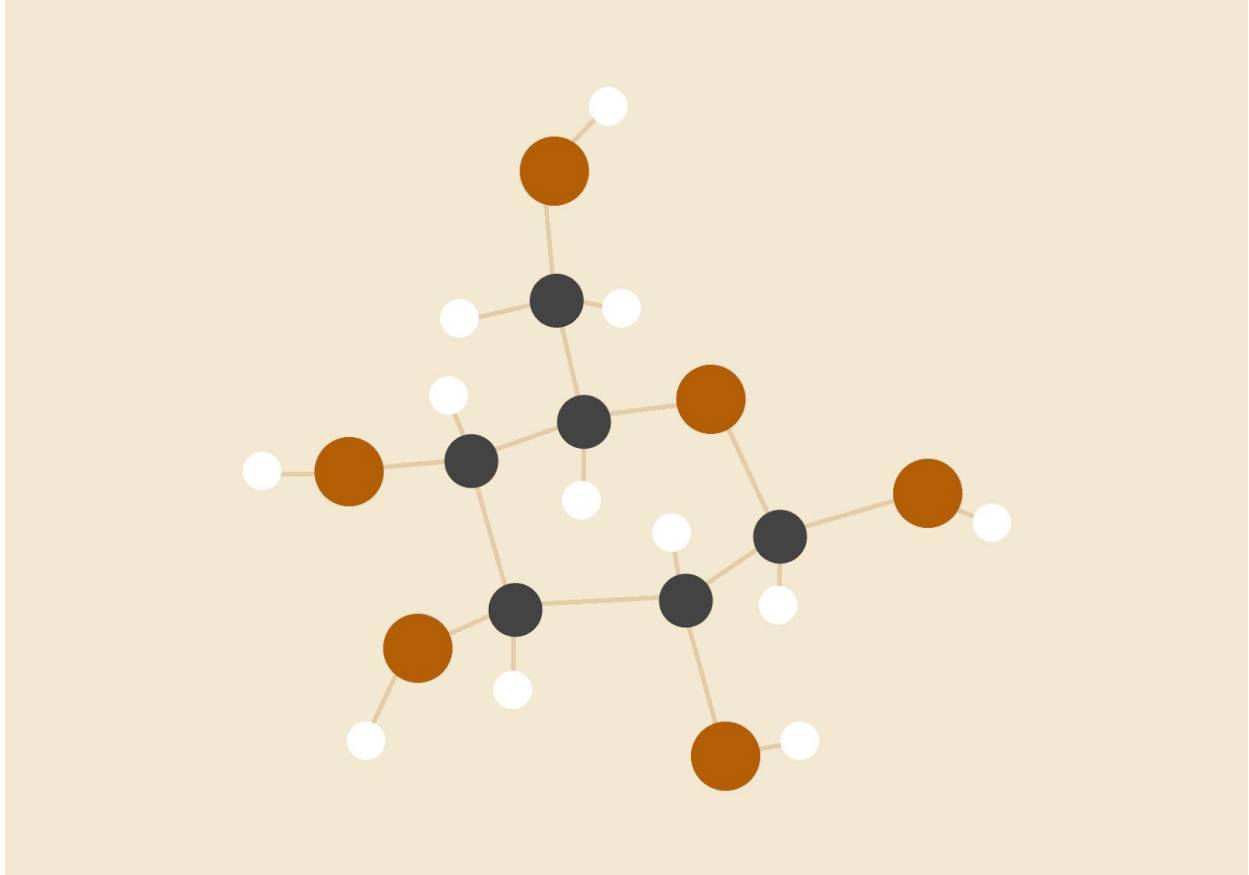


# Capstone Project

*The Battle of Neighborhoods*



**TAN WEI KHANG**

28.07.2020

## INTRODUCTION

As we known, New York city is main and important financial centre in the world and also described as the cultural, financial, and media capital of the world, significantly influencing commerce, entertainment, research, technology, education, politics, tourism, art, fashion, and sports. To survive in the competitive market in the New York city, ensure a correct location for starting own business is the key of success. I will use my exploratory data analysis to help on suggesting which location is suitable for restaurant opening.

## TARGET AUDIENCE

To help the stakeholder who is interested in opening his/ her restaurant in the New York city and suggest them on correct location for his/ her business.

## DATA

1. Free and existing New York City dataset via [https://geo.nyu.edu/catalog/nyu\\_2451\\_34572](https://geo.nyu.edu/catalog/nyu_2451_34572)
2. Foursquare API

## METHODOLOGY

1. Import all dependencies and libraries

```
import numpy as np
import pandas as pd
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

#!conda install -c conda-forge geopy --yes
#convert an address into latitude and longitude values
from geopy.geocoders import Nominatim

import requests # Library to handle requests

import json
from pandas.io.json import json_normalize

# Matplotlib and associated plotting modules
import matplotlib.cm as cm
import matplotlib.colors as colors

# import k-means from clustering stage
from sklearn.cluster import KMeans

#!conda install -c conda-forge folium=0.5.0 --yes # uncomment this line if you haven't completed the Foursquare API Lab
import folium
print('Libraries imported.')

Libraries imported.
```

## 2. Download and Explore Dataset

### Load and explore the data

Load the data in json file

```
with open('newyork_data.json') as json_data:
    newyork_data = json.load(json_data)
```

Quick look on the dataset

```
newyork_data
{'type': 'FeatureCollection',
 'totalFeatures': 306,
 'features': [{'type': 'Feature',
  'id': 'nyu_2451_34572.1',
  'geometry': {'type': 'Point',
   'coordinates': [-73.84720052054902, 40.89470517661]},
  'geometry_name': 'geom',
  'properties': {'name': 'Wakefield',
   'stacked': 1,
   'annoline1': 'Wakefield',
   'annoline2': None,
   'annoline3': None,
   'annoangle': 0.0,
   'borough': 'Bronx',
   'bbox': [-73.84720052054902,
    40.89470517661,
    -73.84720052054902,
    40.89470517661]}},
  {'type': 'Feature',
   'id': 'nyu_2451_34572.2',
   'geometry': {'type': 'Point',
    'coordinates': [-73.82993910812398, 40.87429419303012]},
   'geometry_name': 'geom',
   'properties': {'name': 'Co-op City',
    'stacked': 2,
    'annoline1': 'Co-op',
    'annoline2': 'City',
    'annoline3': None,
```

## 3. Define and Transform the Dataset into DataFrame

### Transform the data into a *pandas* dataframe

```
# define the dataframe columns
column_names = ['Borough', 'Neighborhood', 'Latitude', 'Longitude']

# instantiate the dataframe
neighborhoods = pd.DataFrame(columns=column_names)
neighborhoods
```

	Borough	Neighborhood	Latitude	Longitude
--	---------	--------------	----------	-----------

Loop in the neighborhood data into the dataframe for each rows

```
for data in neighborhoods_data:
    borough = neighborhood_name = data['properties']['borough']
    neighborhood_name = data['properties']['name']

    neighborhood_latlon = data['geometry']['coordinates']
    neighborhood_lat = neighborhood_latlon[1]
    neighborhood_lon = neighborhood_latlon[0]

    neighborhoods = neighborhoods.append({'Borough': borough,
                                          'Neighborhood': neighborhood_name,
                                          'Latitude': neighborhood_lat,
                                          'Longitude': neighborhood_lon}, ignore_index=True)
```

```
#Quick Look the neighborhood dataframe
neighborhoods
```

	Borough	Neighborhood	Latitude	Longitude
0	Bronx	Wakefield	40.894705	-73.847201
1	Bronx	Co-op City	40.874294	-73.829939
2	Bronx	Eastchester	40.887556	-73.827806
3	Bronx	Fieldston	40.895437	-73.905643
4	Bronx	Riverdale	40.890834	-73.912585
5	Bronx	Kingsbridge	40.881687	-73.902818
6	Manhattan	Marble Hill	40.876551	-73.910660

4. Check the number of borough and neighborhoods in New York City

Check the number of borough and neighborhoods in New York

```
print('The dataframe has {} boroughs and {} neighborhoods.'.format(
    len(neighborhoods['Borough'].unique()),
    neighborhoods.shape[0]
))
```

The dataframe has 5 boroughs and 306 neighborhoods.

5. Use geopy to get the Latitude and Longitude of NYC

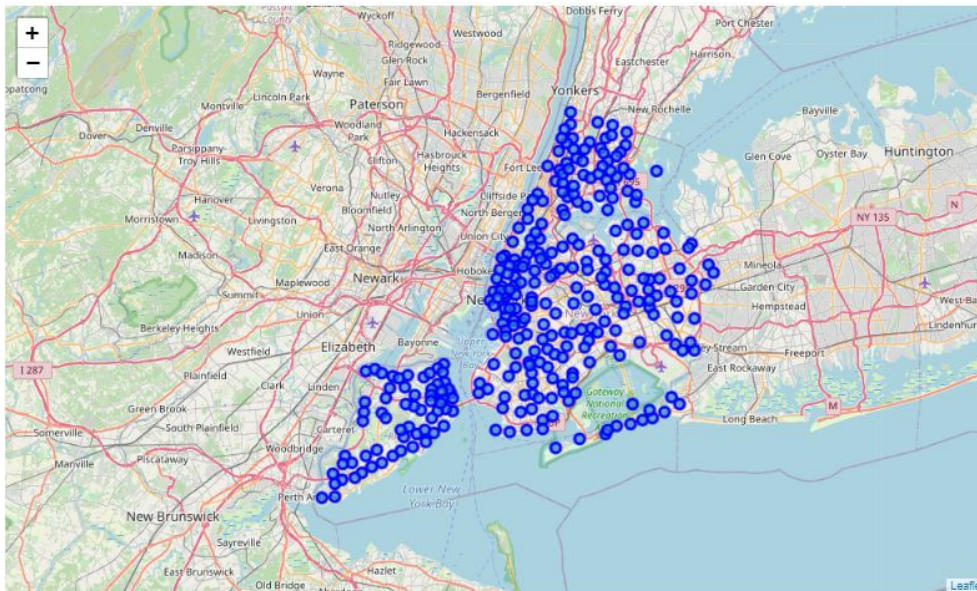
Use geopy to get the latitude and longitude values of New York.

```
address = 'New York City, NY'

geolocator = Nominatim(user_agent="ny_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geograpical coordinate of New York City are {}, {}'.format(latitude, longitude))
```

The geograpical coordinate of New York City are 40.7127281, -74.0060152.

6. Visualize and plot the neighborhoods in NYC





- Find the borough with the most neighborhoods in NYC, Queens is the borough with most neighborhoods, total number 81 .

Find the borough with the most neighborhood

```
neighborhoods.groupby('Borough').count()
```

	Neighborhood	Latitude	Longitude
Borough			
Bronx	52	52	52
Brooklyn	70	70	70
Manhattan	40	40	40
Queens	81	81	81
Staten Island	63	63	63

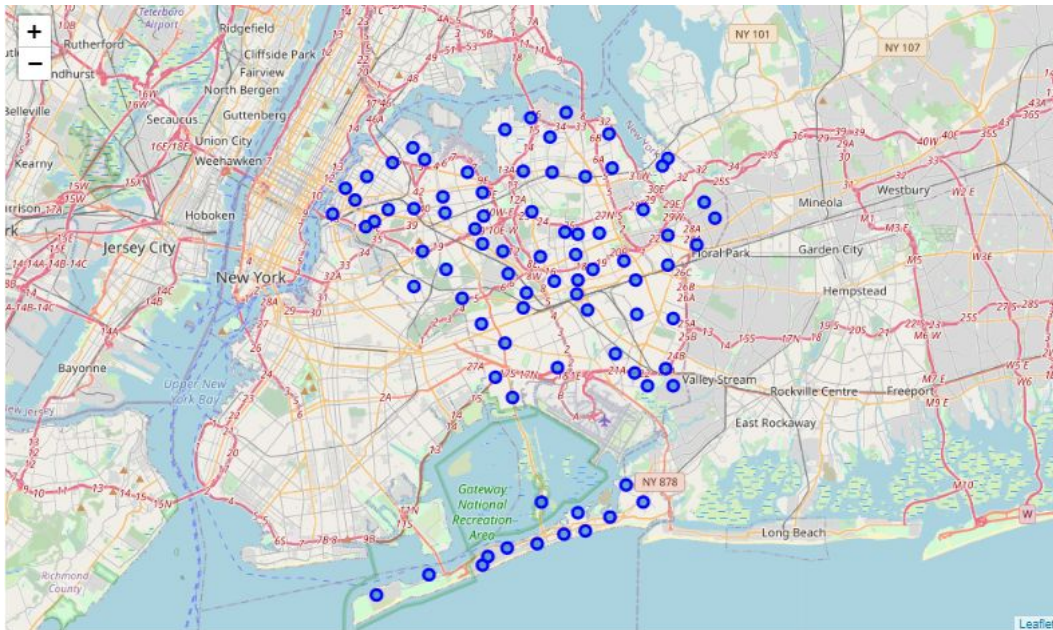
From the above table, we can see Queens borough has the highest number of neighborhood, hence I will focus and pick up Queens borough for my further segmenting and clustering.

Now, create a new DataFrame to capture the Queens's neighborhoods

```
queens_data = neighborhoods[neighborhoods['Borough'] == 'Queens'].reset_index(drop=True)
queens_data
```

	Borough	Neighborhood	Latitude	Longitude
0	Queens	Astoria	40.768509	-73.915654
1	Queens	Woodside	40.745349	-73.901842
2	Queens	Jackson Heights	40.751981	-73.882821
3	Queens	Elmhurst	40.744049	-73.881656
4	Queens	Howard Beach	40.654225	-73.838138
5	Queens	Corona	40.742382	-73.856825
6	Queens	Forest Hills	40.725284	-73.844475
7	Queens	Kew Gardens	40.705179	-73.829819
8	Queens	Richmond Hill	40.697947	-73.831633
9	Queens	Flushing	40.764454	-73.831773
10	Queens	Long Island City	40.750217	-73.939202

- Visualize and plot Queens's neighborhoods in the map



## 9. Explore neighborhoods in Queens by Foursquare API

```
def getNearbyVenues(names, latitudes, longitudes, radius=500):

    venues_list=[]
    for name, lat, lng in zip(names, latitudes, longitudes):
        print(name)

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

        # make the GET request
        results = requests.get(url).json()["response"]["groups"][0]["items"]

        # return only relevant information for each nearby venue
        venues_list.append([
            name,
            lat,
            lng,
            v['venue']['name'],
            v['venue']['location']['lat'],
            v['venue']['location']['lng'],
            v['venue']['categories'][0]['name'] for v in results])

    nearby_venues = pd.DataFrame([item for venue_list in venues_list for item in venue_list])
    nearby_venues.columns = ['Neighborhood',
                            'Neighborhood Latitude',
                            'Neighborhood Longitude',
                            'Venue',
                            'Venue Latitude',
                            'Venue Longitude',
                            'Venue Category']

    return(nearby_venues)
```

```
queens_venues = getNearbyVenues(names=queens_data['Neighborhood'],
                                latitudes=queens_data['Latitude'],
                                longitudes=queens_data['Longitude']
                                )
```

```
Astoria
Woodside
Jackson Heights
Elmhurst
Howard Beach
Corona
Forest Hills
Kew Gardens
Richmond Hill
Flushing
Long Island City
Sunnyside
East Elmhurst
Maspeth
Ridgewood
Glendale
Rego Park
Woodhaven
Ozone Park
South Ozone Park
College Point
Whitestone
Bayside
```

## 10. Check the unique venues

Check the unique categories for each venues

```
print('There are {} uniques categories.'.format(len(queens_venues['Venue Category'].unique())))
```

There are 273 uniques categories.

Check the unique categories' name, and looking for Japanese Restaurant

```
queens_venues['Venue Category'].unique()
```

```
array(['Brazilian Restaurant', 'Gourmet Shop', 'Gym',
      'Seafood Restaurant', 'Middle Eastern Restaurant',
      'Martial Arts Dojo', 'Pub', 'Dessert Shop', 'Restaurant',
      'Indian Restaurant', 'Falafel Restaurant', 'Greek Restaurant',
      'Bakery', 'Ice Cream Shop', 'Bagel Shop',
      'Health & Beauty Service', 'Salad Place',
      'Mediterranean Restaurant', 'Vegetarian / Vegan Restaurant',
      'Mexican Restaurant', 'Bar', 'American Restaurant',
      'Grocery Store', 'Pizza Place', 'Japanese Restaurant',
      'Bubble Tea Shop', 'Thai Restaurant', 'Food & Drink Shop',
      'Food Truck', 'Italian Restaurant', 'Poke Place', 'Food',
      'Korean Restaurant', 'Beer Garden', 'Cocktail Bar',
      'Gym / Fitness Center', 'Cajun / Creole Restaurant', 'BBQ Joint',
      'Soulwaki Shop', 'Comfort Food Restaurant', 'Optical Shop',
      'Hookah Bar', 'Burger Joint', 'Coffee Shop', 'Sandwich Place',
      'Spa', 'Liquor Store', 'Salon / Barbershop', 'Juice Bar',
      'Deli / Bodega', 'Latin American Restaurant', 'Café', 'Wine Shop',
      'Sushi Restaurant', 'Moroccan Restaurant', 'Intersection',
      'Bus Station', 'Fast Food Restaurant', 'Filipino Restaurant',
      'Chinese Restaurant', 'Diner', 'Arepa Restaurant',
      'Fruit & Vegetable Store', 'Tibetan Restaurant',
      'Bohile Phone Shop', 'Donut Shop', 'Bank', 'Himalayan Restaurant']
```

## 11. Analyze the neighborhoods with visit frequency

```
# one hot encoding
queens_onehot = pd.get_dummies(queens_venues[['Venue Category']], prefix="", prefix_sep="")

# add neighborhood column back to dataframe
queens_onehot['Neighborhood'] = queens_venues['Neighborhood']

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

queens_onehot.head()

# new dataframe size
print(queens_onehot.shape)

(2889, 273)
```

Group the neighborhood and take the mean of frequency of occurrence of each category

```
queens_grouped = queens_onehot.groupby('Neighborhood').mean().reset_index()
queens_grouped.head()
```

	Neighborhood	Yoga Studio	Accessories Store	Afghan Restaurant	American Restaurant	Arepa Restaurant	Argentinian Restaurant	Art Gallery	Art Museum	Arts & Crafts Store	Arts & Entertainment	Asian Restaurant	Athletics & Sports	Autom
0	Arverne	0.0	0.000000	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.00	
1	Astoria	0.0	0.000000	0.0	0.010000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.00	
2	Astoria Heights	0.0	0.000000	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.00	
3	Auburndale	0.0	0.000000	0.0	0.050000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.05	
4	Bay Terrace	0.0	0.028316	0.0	0.052632	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.00	

## 12. Get the restaurant, cafe and bars for further investigation

Get the columns with Restaurant, Cafe, Bar to investigate which neighborhood has highest frequency of visit on food and drink

```
grouped_columns = queens_grouped.columns
restaurant_columns = []
count = 0
for restaurant in grouped_columns:
    if 'Restaurant' in restaurant:
        restaurant_columns.append(restaurant)
        count+=1
    elif 'Food & Drink Shop' in restaurant:
        restaurant_columns.append(restaurant)
        count+=1
    elif 'Café' in restaurant:
        restaurant_columns.append(restaurant)
        count+=1
    elif 'BBQ' in restaurant:
        restaurant_columns.append(restaurant)
        count+=1
    elif 'Bar' in restaurant:
        restaurant_columns.append(restaurant)
        count+=1
    elif 'Cafe' in restaurant:
        restaurant_columns.append(restaurant)
        count+=1

print("Total {} of Restaurants, Cafe, Bars are added and available.".format(count))
restaurant_columns
```

Total 76 of Restaurants, Cafe, Bars are added and available.

```
['Afghan Restaurant',
 'American Restaurant',
 'Arepa Restaurant',
 'Argentinian Restaurant',
 'Asian Restaurant',
 'BBQ Joint',
 'Bar',
 'Beach Bar',
 'Brazilian Restaurant',
```

## 13. Find the Top 10 venues without any Japanese restaurant

Get Top 10 highest visited neighborhoods for food and drink by sorting dataframe in descending order

```
queens_grouped_sorted = queens_grouped.sort_values(ascending=False, by=['Total Visited Frequency']).reset_index(drop=True)
print(queens_grouped_sorted.shape)
queens_grouped_sorted.head(10)
```

(81, 274)

	Neighborhood	Yoga Studio	Accessories Store	Afghan Restaurant	American Restaurant	Arepa Restaurant	Argentinian Restaurant	Art Gallery	Art Museum	Arts & Crafts Store	Arts & Entertainment	Asian Restaurant	Athletics & Sports	Aut
0	Elmhurst	0.000000	0.0	0.000000	0.000000	0.000000	0.030303	0.0	0.0	0.0	0.0	0.000000	0.0	0
1	Murray Hill	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.0	0.0	0.0	0.0	0.021277	0.0	0
2	Ravenswood	0.000000	0.0	0.038462	0.000000	0.000000	0.000000	0.0	0.0	0.0	0.0	0.000000	0.0	0
3	Astoria	0.000000	0.0	0.000000	0.010000	0.000000	0.000000	0.0	0.0	0.0	0.0	0.000000	0.0	0
4	Jackson Heights	0.000000	0.0	0.000000	0.000000	0.000000	0.012500	0.0	0.0	0.0	0.0	0.012500	0.0	0
5	Laurelton	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.0	0.0	0.0	0.0	0.000000	0.0	0
6	Bayside	0.013158	0.0	0.000000	0.039474	0.000000	0.000000	0.0	0.0	0.0	0.0	0.013158	0.0	0
7	Rockaway Beach	0.000000	0.0	0.000000	0.000000	0.048512	0.000000	0.0	0.0	0.0	0.0	0.000000	0.0	0
8	Flushing	0.018667	0.0	0.000000	0.000000	0.000000	0.000000	0.0	0.0	0.0	0.0	0.018667	0.0	0
9	St Albans	0.000000	0.0	0.000000	0.000000	0.000000	0.000000	0.0	0.0	0.0	0.0	0.000000	0.0	0

Filter out the Neighborhoods without Japanese Restaurant opened before to explore out brand new place for Japanese restaurant opening without competitive with other existing japanese restaurants

Filter out the Neighborhoods without Japanese Restaurant opened before to explore out brand new place for Japanese restaurant opening without competitive with other existing japanese restaurants

```
no_jp_neighborhood = queens_grouped_sorted[queens_grouped_sorted['Japanese Restaurant']==0].reset_index(drop=True)
print(no_jp_neighborhood.shape)
no_jp_neighborhood
```

(67, 274)

	Neighborhood	Yoga Studio	Accessories Store	Afghan Restaurant	American Restaurant	Arepa Restaurant	Argentinian Restaurant	Art Gallery	Art Museum	Arts & Crafts Store	Arts & Entertainment	Asian Restaurant	Athletics & Sports	A
0	Elmhurst	0.000000	0.000000	0.000000	0.000000	0.000000	0.030303	0.000000	0.000000	0.000	0.000000	0.000000	0.00	
1	Jackson Heights	0.000000	0.000000	0.000000	0.000000	0.000000	0.012500	0.000000	0.000000	0.000	0.000000	0.012500	0.00	
2	Laurelton	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000	0.000000	0.000000	0.00	
3	Bayside	0.013158	0.000000	0.000000	0.039474	0.000000	0.000000	0.000000	0.000000	0.000	0.000000	0.013158	0.00	
4	Rockaway Beach	0.000000	0.000000	0.000000	0.000000	0.048512	0.000000	0.000000	0.000000	0.000	0.000000	0.000000	0.00	
5	Flushing	0.018667	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000	0.000000	0.018667	0.00	
6	St Albans	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000	0.000000	0.000000	0.00	
7	Sunnyside	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000	0.000000	0.000000	0.00	
8	Oakland Gardens	0.080000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000	0.000000	0.000000	0.00	
9	Jamaica Hills	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000	0.000000	0.000000	0.00	

## 14. Print top 5 visit venues for each of neighborhoods

Print each neighborhoods along with the top 5 most common venues

```
num_top_venues = 5

for hood in food_drink_neighborhoods['Neighborhood']:
    print("----"+hood+"----")
    temp = food_drink_neighborhoods[food_drink_neighborhoods['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')
```

----Elmhurst----

	venue	freq
0	Thai Restaurant	0.21
1	Mexican Restaurant	0.12
2	Chinese Restaurant	0.09
3	Bubble Tea Shop	0.06
4	South American Restaurant	0.06

----Jackson Heights----



## 15. Visualize the most visit venue with table

Create the new dataframe and display the top 10 venues for each neighborhood.

```
num_top_venues = 10

indicators = ['st', 'nd', 'rd']

# create columns according to number of top venues
columns = ['Neighborhood']
for ind in np.arange(num_top_venues):
    try:
        columns.append('{} {} Most Common Venue'.format(ind+1, indicators[ind]))
    except:
        columns.append('{}th Most Common Venue'.format(ind+1))

# create a new dataframe
neighborhoods_venues_sorted = pd.DataFrame(columns=columns)
neighborhoods_venues_sorted['Neighborhood'] = food_drink_neighborhoods['Neighborhood']

for ind in np.arange(food_drink_neighborhoods.shape[0]):
    neighborhood_venues_sorted.iloc[ind, 1:] = return_most_common_venues(food_drink_neighborhoods.iloc[ind, :], num_top_venues)

neighborhoods_venues_sorted.head()
```

	Neighborhood	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	Elmhurst	Thai Restaurant	Mexican Restaurant	Chinese Restaurant	Vietnamese Restaurant	South American Restaurant	Bubble Tea Shop	Pizza Place	Park	Salon / Barbershop	Malay Restaurant
1	Jackson Heights	Latin American Restaurant	Peruvian Restaurant	South American Restaurant	Mobile Phone Shop	Bakery	Grocery Store	Mexican Restaurant	Thai Restaurant	Clothing Store	Diner
2	Laurelton	Caribbean Restaurant	Park	Train Station	Women's Store	Event Space	Falafel Restaurant	Farm	Farmers Market	Fast Food Restaurant	Filipino Restaurant
3	Bayside	Bar	Pizza Place	American Restaurant	Sushi Restaurant	Indian Restaurant	Spa	Cosmetics Shop	Pharmacy	Bakery	Mediterranean Restaurant
4	Rockaway Beach	Beach	Ice Cream Shop	Arepa Restaurant	Seafood Restaurant	BBQ Joint	Fast Food Restaurant	Latin American Restaurant	Spanish Restaurant	New American Restaurant	Supermarket

## 16. Clustering the neighborhoods

Run k-means to cluster the neighborhood into 5 clusters.

```
# set number of clusters
kclusters = 5

queens_grouped_clustering = food_drink_neighborhoods.drop('Neighborhood', 1)

# run k-means clustering
kmeans = KMeans(n_clusters=kclusters, random_state=0).fit(queens_grouped_clustering)

# check cluster labels generated for each row in the dataframe
kmeans.labels_[0:10]

array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
```

Create a new dataframe that includes the cluster as well as the top 10 venues for each neighborhood.

```
# add clustering labels
neighborhoods_venues_sorted.insert(0, 'cluster_labels', kmeans.labels_)

queens_merged = queens_data

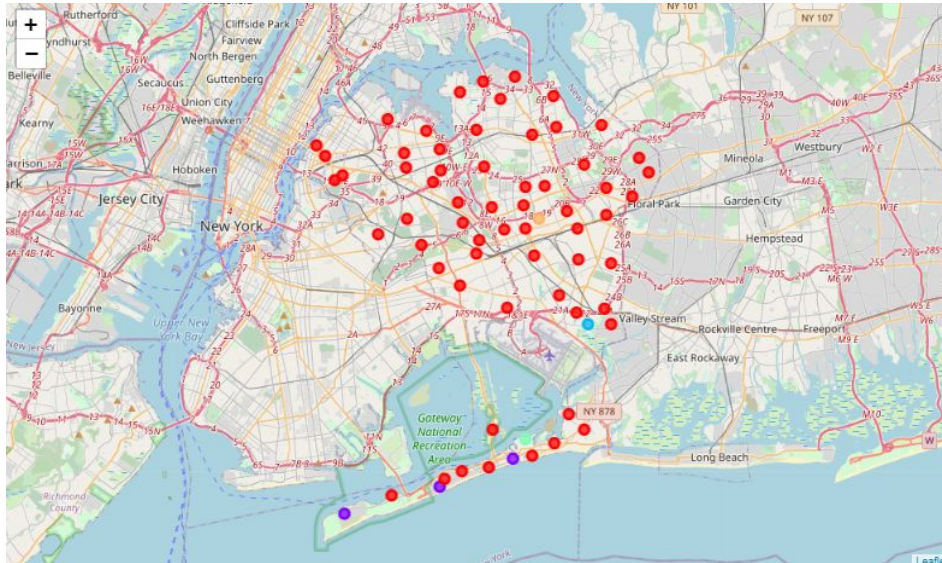
queens_merged = queens_merged.join(neighborhoods_venues_sorted.set_index('Neighborhood'), on='Neighborhood')

queens_merged = queens_merged.dropna().reset_index(drop=True)

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

	Borough	Neighborhood	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
0	Queens	Jackson Heights	40.751981	-73.882821	0.0	Latin American Restaurant	Peruvian Restaurant	South American Restaurant	Mobile Phone Shop	Bakery	Grocery Store	Mexican Restaurant	Thai Restaurant
1	Queens	Elmhurst	40.744049	-73.881656	0.0	Thai Restaurant	Mexican Restaurant	Chinese Restaurant	Vietnamese Restaurant	South American Restaurant	Bubble Tea Shop	Pizza Place	Park
2	Queens	Corona	40.742382	-73.856825	0.0	Mexican Restaurant	Supermarket	Convenience Store	Bakery	Pizza Place	Donut Shop	Restaurant	Park

## 17. Visualize the clustering, Cluster 1(Red), Cluster 2(Purple), Cluster 3(Light blue)



## 18. Examine each of the clusters

### Cluster 1

```
queens_merged.loc[queens_merged['Cluster Labels'] == 0, queens_merged.columns[[1] + list(range(5, queens_merged.shape[1]))]]
```

	Neighborhood	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	Jackson Heights	Latin American Restaurant	Peruvian Restaurant	South American Restaurant	Mobile Phone Shop	Bakery	Grocery Store	Mexican Restaurant	Thai Restaurant	Clothing Store	Diner
1	Elmhurst	Thai Restaurant	Mexican Restaurant	Chinese Restaurant	Vietnamese Restaurant	South American Restaurant	Bubble Tea Shop	Pizza Place	Park	Salon / Barbershop	Malay Restaurant
2	Corona	Mexican Restaurant	Supermarket	Convenience Store	Bakery	Pizza Place	Donut Shop	Restaurant	Park	Sandwich Place	Empanada Restaurant
3	Forest Hills	Gym	Gym / Fitness Center	Yoga Studio	Pizza Place	Pharmacy	Convenience Store	Park	Thai Restaurant	Asian Restaurant	Supermarket
4	Kew Gardens	Cosmetics Shop	Chinese Restaurant	Indian Restaurant	Bar	Bank	Park	Donut Shop	Deli / Bodega	Pet Store	Pizza Place
5	Richmond Hill	Latin American	Lounge	Pizza Place	Bank	Caribbean	Supermarket	Metro	Movino Target	Bus Station	Women's

### Cluster 2

```
queens_merged.loc[queens_merged['Cluster Labels'] == 1, queens_merged.columns[[1] + list(range(5, queens_merged.shape[1]))]]
```

	Neighborhood	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
36	Breezy Point	Beach	Monument / Landmark	Trail	Bus Stop	Women's Store	Fish & Chips Shop	Farm	Farmers Market	Fast Food Restaurant	Filipino Restaurant
42	Neponsit	Beach	Bar	Beach Bar	Frozen Yogurt Shop	Farm	Farmers Market	Fast Food Restaurant	Filipino Restaurant	Fish & Chips Shop	Fish Market
64	Hammels	Beach	Gym / Fitness Center	Dog Run	Fried Chicken Joint	Bus Station	Bus Stop	Fast Food Restaurant	Shoe Store	Food Truck	Diner

#### Cluster 3

```
queens_merged.loc[queens_merged['Cluster Labels'] == 2, queens_merged.columns[[1] + list(range(5, queens_merged.shape[1]))]]
```

	Neighborhood	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
53	Brookville	Deli / Bodega	Women's Store	Fish Market	Falafel Restaurant	Farm	Farmers Market	Fast Food Restaurant	Filipino Restaurant	Fish & Chips Shop	Flea Market

#### Cluster 4

```
queens_merged.loc[queens_merged['Cluster Labels'] == 3, queens_merged.columns[[1] + list(range(5, queens_merged.shape[1]))]]
```

	Neighborhood	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
52	Somerville	Park	Women's Store	Filipino Restaurant	Event Space	Falafel Restaurant	Farm	Farmers Market	Fast Food Restaurant	Fish & Chips Shop	Indie Movie Theater

#### Cluster 5

```
queens_merged.loc[queens_merged['Cluster Labels'] == 4, queens_merged.columns[[1] + list(range(5, queens_merged.shape[1]))]]
```

	Neighborhood	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
45	Jamaica Estates	Dog Run	Bus Station	Women's Store	Farm	Farmers Market	Fast Food Restaurant	Filipino Restaurant	Fish & Chips Shop	Fish Market	Flea Market

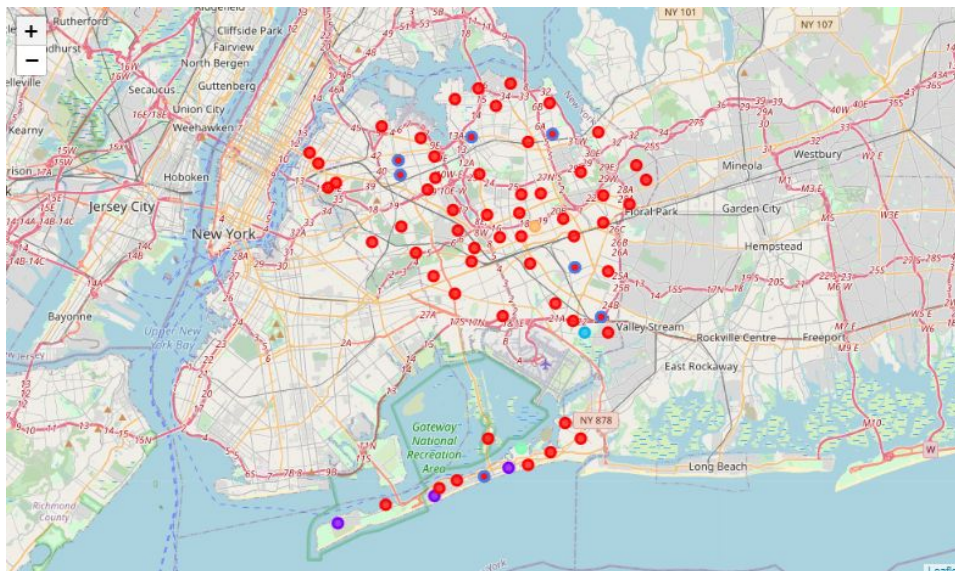
## 19. List out 10 of most visited venues for food and drinks

```
top_10_neigh = []
for neigh in queens_grouped_sorted['Neighborhood'].head(10):
    top_10_neigh.append(neigh)

top_10_neigh

['Elmhurst',
'Murray Hill',
'Ravenswood',
'Astoria',
'Jackson Heights',
'Laurelton',
'Bayside',
'Rockaway Beach',
'Flushing',
'St. Albans']
```

## 20. Visualize and plot the most common visit venues for food and drinks(dots which has red color covered with light blue color)





## RESULTS

- Top 10 Neighborhoods has been defined by using frequency of visit of each venues for food and drinks, the venues are:
  - Elmhurst *freq*=0.787879,
  - Murray Hill *freq*=0.595745,
  - Ravenswood *freq*=0.576923,
  - Astoria *freq*=0.550000,
  - Jackson Heights *freq*=0.525000,
  - Laurelton *freq*=0.500000,
  - Bayside *freq*=0.473684,
  - Rockaway Beach *freq*=0.465116,
  - Flushing *freq*=0.450000,
  - St. Albans *freq*=0.444444
- Top 10 Neighborhoods are clustered in Cluster 1(Red) and also they are covered with light blue color.
- Based on the above frequency, we can assume that Elmhurst has the highest frequency of visits, hence most of the people will travel to Elmhurst for food and drinks.

## CONCLUSION

In New York City, Queens has the most neighborhoods among other boroughs. Inside the Queens, there are a total 81 Neighborhoods with 273 Venues including 76 of Restaurants, Cafes and Bars. Among all of the neighborhoods, Elmhurst is the most common visit venue as there are lots of restaurants, cafes and bars and there is no Japanese restaurant opened yet. Hence, stakeholder may visit Elmhurst or another 9 venues for further investigation and consideration on Japanese Restaurant opening.