Restaurant Location Recommender – New York City

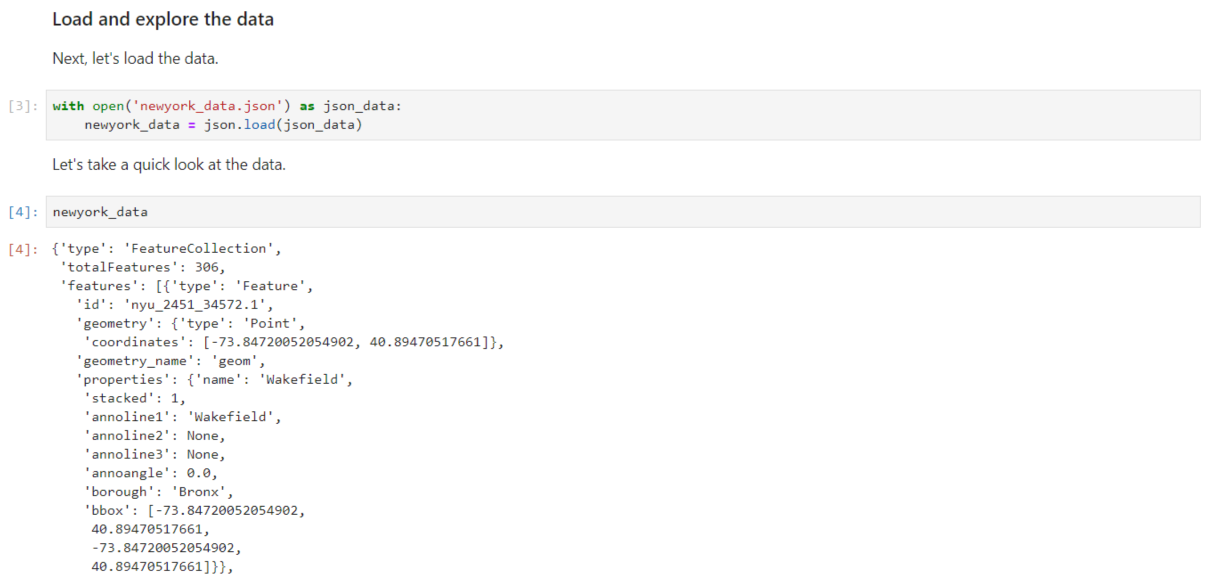
Introduction/Business Problem:

This capstone project is meant to build a restaurant location recommender in New York City. NYC is one of the world's most prosperous and diverse metropolis. There are a huge number of restaurants in NYC already, recorded at 26,697 in 2017, which is up from 25,305 in 2014 [1]. Therefore, it is indisputable that the competition in the NYC restaurant business is fierce. With the number of restaurants still increasing, it can be anticipated that the competition will become even stronger in the future. Therefore, for a prospective new restaurant owner who dreams of opening a restaurant anywhere in NYC, it is critical for them to understand where to open their restaurant based on his/her desired restaurant type (e.g. fast food stand/fine-dining restaurant, cuisine, etc.), to maximize their probability of survival and profitability in the sea of restaurants in NYC. Therefore, this project is targeted for anyone who is interested in opening a restaurant of any type in NYC.

Of course, determination of a restaurant's location is based on various factors, such as availability of retail space, population of the area, etc., and comprehensive coverage of all these factors is not within scope of this project. This project provides a straightforward location suitability score metric to help the restaurant owner narrow the prospective location down to which of the 306 neighborhoods in NYC should they pay special attention to in hunting for a prospective restaurant location. To maximize the profitability of opening their desired type of restaurant in NYC, this location recommender would be a great start for restaurant owners.

Data:

The first dataset needed for this project will be the json data of NYC, which includes 5 boroughs and 306 neighborhoods. This data is essential for obtaining a complete set of location data, including boroughs, neighborhood longitude, neighborhood latitude, etc. It is also the basis of the location suitability score we are trying to obtain as a final product of the project. This dataset has been used in the ungraded lab of Week 3, so it will be processed similarly in this project.



Load and explore the data

Next, let's load the data.

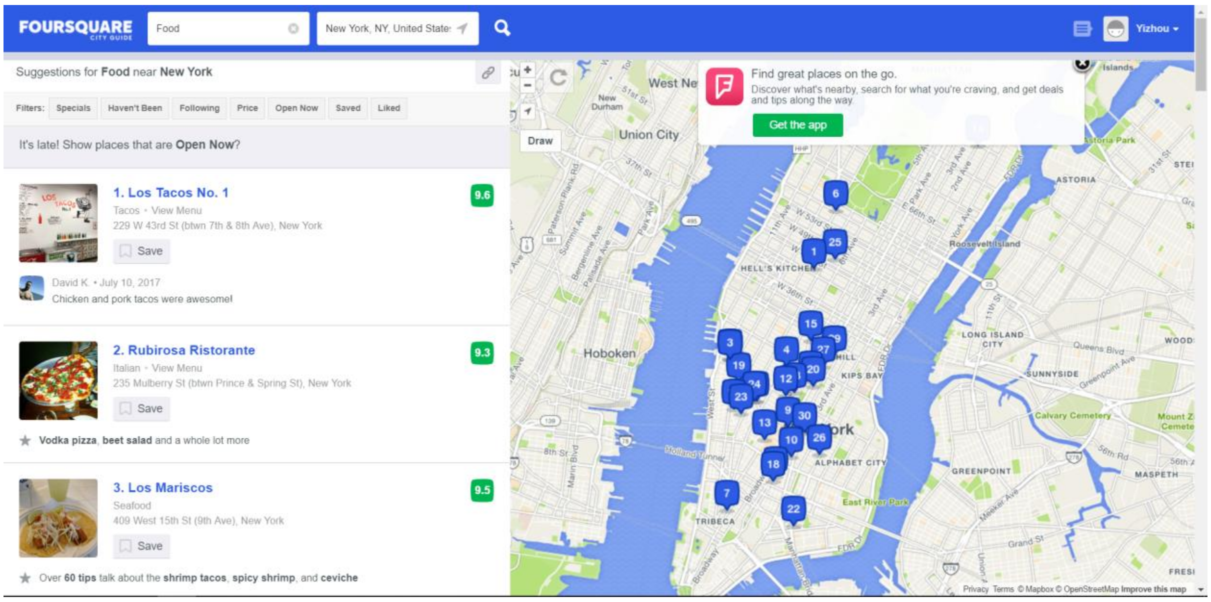
```
[3]: with open('newyork_data.json') as json_data:
         newyork_data = json.load(json_data)
```

Let's take a quick look at the data.

```
[4]: newyork_data
```

```
[4]: {'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]}},
```

Besides, there is of course the Foursquare location data, in which you can find hundreds of categories of restaurants in NYC with various types of information. Foursquare location data provides venues in which we can access details including ID, name, exact location (latitude, longitude), URL, contact information, ratings, tips and other statistics. These information can be particularly useful for a more refined restaurant locator than what the present project presents. However, some of these information requires premium calls that is limited to 500 each day for free developer accounts, therefore limiting the possibility of the location recommender. An example of Foursquare location data is provided below.



Suggestions for **Food** near **New York**

Filters: Specials Haven't Been Following Price Open Now Saved Liked

It's late! Show places that are **Open Now**?

1. Los Tacos No. 1 9.6
Tacos • View Menu
229 W 43rd St (btwn 7th & 8th Ave), New York
David K. • July 10, 2017
Chicken and pork tacos were awesome!

2. Rubirosa Ristorante 9.3
Italian • View Menu
235 Mulberry St (btwn Prince & Spring St), New York
Vodka pizza, **beet salad** and a whole lot more

3. Los Mariscos 9.5
Seafood
409 West 15th St (9th Ave), New York
Over **60 tips** talk about the **shrimp tacos**, **spicy shrimp**, and **ceviche**

Find great places on the go.
Discover what's nearby, search for what you're craving, and get deals and tips along the way.

```
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]},
 'venue': {'id': '4b79cc46f964a520c5122fe3',
  'name': 'Tibbett Diner',
  'location': {'address': '3033 Tibbett Ave',
   'crossStreet': 'btwn 230th & 231st',
   'lat': 40.8804044222466,
   'lng': -73.90893738006402,
   'labeledLatLngs': [{'label': 'display',
     'lat': 40.8804044222466,
     'lng': -73.90893738006402}],
   'distance': 452,
   'postalCode': '10463',
   'cc': 'US',
   'city': 'Bronx',
   'state': 'NY',
   'country': 'United States',
   'formattedAddress': ['3033 Tibbett Ave (btwn 230th & 231st)',
    'Bronx, NY 10463',
    'United States']},
  'categories': [{'id': '4bf58dd8d48988d147941735',
    'name': 'Diner',
    'pluralName': 'Diners',
    'shortName': 'Diner',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/diner_',
     'suffix': '.png'},
    'primary': True}],
  'photos': {'count': 0, 'groups': []}},
 'referralId': 'e-0-4b79cc46f964a520c5122fe3-2'},
```

Foursquare location data of restaurants will provide insights into the competitiveness of a certain type of restaurants (cuisine, etc.) in each neighborhood. The relevant data of the restaurants will be obtained through the search API, belonging to regular calls, with up to 99,500 calls per day. The most useful information extracted from the requested data of API calls would include the categories, latitude, longitude of the restaurant. We will count how many restaurants in total and by categories can be found in each neighborhood.

Methodology:

The first step in the analysis is to import all the necessary modules and download the NYC neighborhoods JSON file ‘newyork_data.json’. Imported modules include numpy, pandas, json, requests, matplotlib and scipy, in which we will use distance_matrix to compute the distances between neighborhoods later.

Then we create the neighborhoods data frame by extracting the borough name, the neighborhood name and the exact location of the neighborhoods (latitude and longitude) from the JSON file. It is basically the same thing that we did in the ungraded lab in Week 3. The head of the data frame is shown below:

	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

Once the neighborhood information is obtained, Foursquare location data can be used to access the information of neighborhood restaurants. First, the Foursquare credentials and version are specified, then a search request URL with a query of ‘restaurant’ is established:

```
url = 'https://api.foursquare.com/v2/venues/search?client_id={}&client_secret={}&v={}&ll={},{}&radius={}&query=restaurant'.format(
    CLIENT_ID,
    CLIENT_SECRET,
    VERSION,
    lat,
    lng,
    radius)
```

The URL was sent to request all the restaurants within a 500m radius of the default center point of each neighborhood. Useful information (name, location and category) of the restaurants was extracted, and together with the neighborhood information obtained previously, fills a new data frame named ‘nearby_restaurants’. The head of this data frame is shown below:

[9]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Restaurant	Restaurant Latitude	Restaurant Longitude	Restaurant Category
0	Wakefield	40.894705	-73.847201	Cooler Runnings Jamaican Restaurant Inc	40.898276	-73.850381	Caribbean Restaurant
1	Wakefield	40.894705	-73.847201	Cool Running Restaurant	40.898130	-73.848516	Food
2	Wakefield	40.894705	-73.847201	Bay restaurant	40.890850	-73.848860	Not found
3	Wakefield	40.894705	-73.847201	Fort Negril Jamaican Restaurant	40.898318	-73.850442	Food
4	Wakefield	40.894705	-73.847201	Allure Restaurant And Bar	40.892543	-73.852265	Restaurant

A function was thereby created to integrate the above process associated with Foursquare location data. The function was named ‘NeighborhoodRestaurants’, to complete the following procedure:

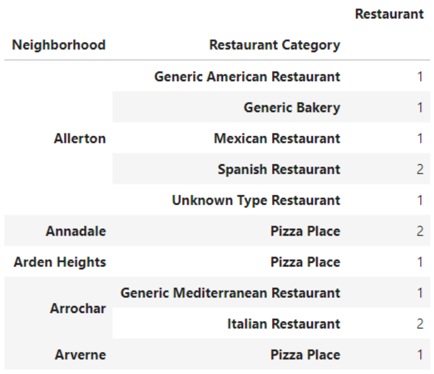
1. Create the request URL for all the restaurants in each neighborhood within a radius of 500 meters.
2. Make the GET request and acquire the results.
3. Extract relevant information for each nearby restaurant.
4. Create a data frame containing the relevant information for all the nearby restaurants.
5. Drop the duplicates, which can occur due to a restaurant being searched for more than one neighborhood.

After establishing the data frame of the useful information, the next step is data cleaning. This is a critical step in the project. In the Foursquare search results, some of the restaurants showed N/A in their categories. Besides, some of the listed search results were not actually restaurants of any types, including venues like Automotive Shop, Design Studio, etc. Therefore, it was necessary to rule out all the irrelevant venues. The idea was to group the ‘nearby_restaurants’ by restaurant categories. The venues with empty category name were categorized as ‘Not Found’. All the category names were manually examined to determine whether they can be considered a restaurant. Those that were not restaurants were ruled out, including all ‘Not Found’ venues. It is also an idea to manually examine the name of the ‘Not Found’ venues to see if they belong to a restaurant. However, some of the venue names could be hard to identify their categories, and in case there are thousands of ‘Not Found’ venues, manual examination can be a very tedious process.

Then we had all the venues that can be considered a restaurant. It was discovered that some of the restaurant categories may seem confusing because one category can be part of another category. For example, ‘Hunan Restaurant’, ‘Shanghai Restaurant’ and ‘Szechuan Restaurant’ can all belong to ‘Chinese Restaurant’, because these are all regional cuisines in China. Therefore, it was assumed that restaurants in category ‘Chinese Restaurant’ sells some dishes from all parts of China, or a mixture of different regions, while the regional cuisine restaurants feature specialties only from a region in China. Considering this, the category ‘Chinese Restaurant’ was renamed ‘Generic Chinese Restaurant’. The same thing happened to ‘Middle East Restaurant’, ‘African Restaurant’, ‘Latin American Restaurant’, etc. Also, for those categories that were very generic and vague, like ‘Food’ and ‘Restaurant’, we combined them into a single category named ‘Unknown Type Restaurant’. This was to remind the restaurant owner that Foursquare location data cannot identify the exact type of this restaurant, and the owner may need to visit that restaurant to understand the type if necessary.

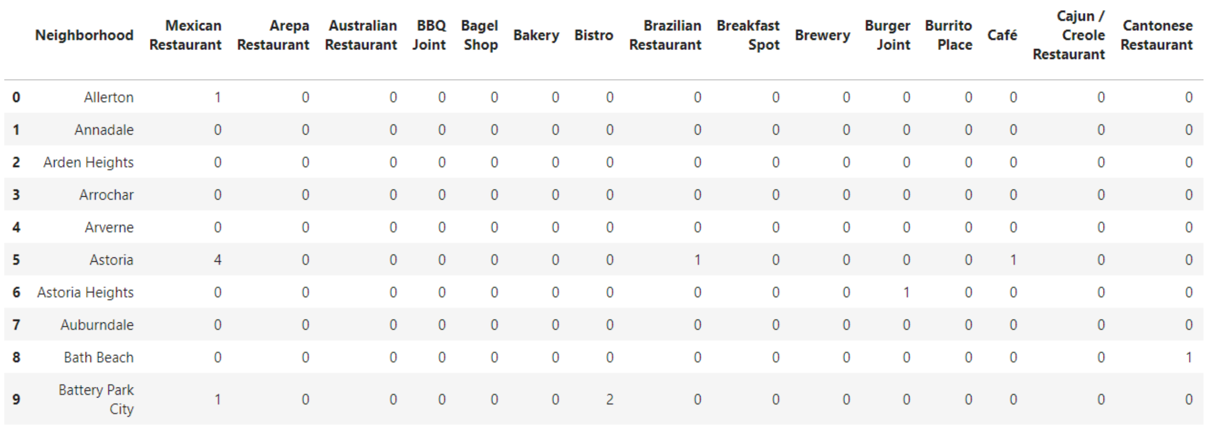
In addition, many of their categories symbolized the cuisines, such as African, Chinese, Italian, etc., while others showed their styles, especially bars. However, some of the categories could be combined. For example, a ‘Dim Sum Restaurant’ has the same cuisine and is basically the same type of restaurants as a ‘Cantonese Restaurant’, only perhaps featuring Dim Sum as their specialties. Therefore, they were combined into a single category ‘Cantonese Restaurant’. Similar examples included ‘Sushi Restaurant’ and ‘Japanese Restaurant’. Besides, some of the categories were actually synonyms, including ‘Coffee Shop’ and ‘Cafe’, ‘Food Court’ and ‘Fast Food Restaurant’, etc. They were combined similarly to a single type. By counting the unique values in ‘Restaurant Category’, it was discovered that the total number of restaurant types was cut down to 101 types. These types should already cover all the major types of restaurants out there, and should be enough for a restaurant owner to find his/her right choice.

After all the cleaning work, we used group by again jointly on neighborhoods and categories to count how many categories of restaurants and how many of each category are available around each neighborhood. The first 10 rows of the grouped data frame are shown as follows:



		Restaurant
Neighborhood	Restaurant Category	
Allerton	Generic American Restaurant	1
	Generic Bakery	1
	Mexican Restaurant	1
	Spanish Restaurant	2
	Unknown Type Restaurant	1
Annadale	Pizza Place	2
Arden Heights	Pizza Place	1
Arrochar	Generic Mediterranean Restaurant	1
	Italian Restaurant	2
Arverne	Pizza Place	1

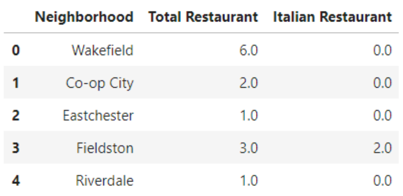
Although visually it gives a clear indication of the restaurants in each neighborhood, it is not friendly for statistical purposes, which will eventually be used to compute the location suitability score. Therefore, one-hot encoding was used. Then the encoded data frame was grouped by neighborhood and count was applied to all 101 categories, getting a more orderly frame more friendly for statistical analysis. The first 10 rows of the one-hot encoded and grouped data frame 'newyork_restaurants_grouped' are given below (only the first few categories in columns are shown):



	Neighborhood	Mexican Restaurant	Arepa Restaurant	Australian Restaurant	BBQ Joint	Bagel Shop	Bakery	Bistro	Brazilian Restaurant	Breakfast Spot	Brewery	Burger Joint	Burrito Place	Café	Cajun / Creole Restaurant	Cantonese Restaurant
0	Allerton	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	Annadale	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	Arden Heights	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	Arrochar	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	Arverne	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	Astoria	4	0	0	0	0	0	0	1	0	0	0	0	1	0	0
6	Astoria Heights	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
7	Auburndale	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	Bath Beach	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
9	Battery Park City	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0

From the shape of the 'newyork_restaurants_grouped' data frame, it was discovered that only 264 out of 306 neighborhoods actually has a venue that can be considered a restaurant.

Now we can start creating a new data frame called 'recommender' to hold features needed for final calculation of location suitability score. First, the total number of all restaurants and the specific category of restaurants that the person want to open was included in the analysis. It was assumed that if there are a ton of restaurants already in or near this neighborhood, then it could be slightly more difficult for a new restaurant to stand out, regardless of the type. Of course, the greater impact would come from restaurants of the same cuisine or type, so restaurants from the same categories would be something to avoid when opening new restaurants. Suppose someone wants to open a new Italian restaurant, the head of the initial 'recommender' data frame would look like this:



	Neighborhood	Total Restaurant	Italian Restaurant
0	Wakefield	6.0	0.0
1	Co-op City	2.0	0.0
2	Eastchester	1.0	0.0
3	Fieldston	3.0	2.0
4	Riverdale	1.0	0.0

The next step is to compute the distances between neighborhoods and correlate the neighborhoods based on their distances. It is generally assumed that the farther neighborhoods are from each other, the less impact the restaurants in one neighborhood would have on those in the other one. Since we have the latitudes and longitudes of each neighborhood, we can use distance_matrix in scipy to compute a 306*306 square distance matrix for all neighborhoods.

```
array([[0.        , 0.02673134, 0.02066992, ..., 0.29479646, 0.17000679,
        0.363258  ],
       [0.02673134, 0.        , 0.01343187, ..., 0.27064158, 0.16539791,
        0.35978343],
       [0.02066992, 0.01343187, 0.        , ..., 0.28307107, 0.17653736,
        0.37082949],
       ...,
       [0.29479646, 0.27064158, 0.28307107, ..., 0.        , 0.23073143,
        0.3158289 ],
       [0.17000679, 0.16539791, 0.17653736, ..., 0.23073143, 0.        ,
        0.19438557],
       [0.363258  , 0.35978343, 0.37082949, ..., 0.3158289 , 0.19438557,
        0.        ]])
```

Then we inverse all the distances values as the impact of restaurants in one neighborhood on another neighborhood is negatively correlated. For diagonal elements, which are 0 due to the distance between each neighborhood and themselves, were converted to a large number of 100 to replace infinity in inverse, to indicate their huge impact. The resultant matrix essentially becomes an impact matrix.

```
array([[100.        ,  37.40927129,  48.37949057, ...,   3.392171  ,
          5.88211802,   2.75286435],
       [ 37.40927129, 100.        ,  74.4497814 , ...,   3.69492379,
          6.04602548,   2.77944983],
       [ 48.37949057,  74.4497814 , 100.        , ...,   3.53268172,
          5.66452344,   2.69665719],
       ...,
       [  3.392171  ,   3.69492379,   3.53268172, ..., 100.        ,
          4.33404328,   3.16627137],
       [  5.88211802,   6.04602548,   5.66452344, ...,   4.33404328,
        100.        ,   5.14441492],
       [  2.75286435,   2.77944983,   2.69665719, ...,   3.16627137,
          5.14441492, 100.        ]])
```

Since the customer is interested in opening an Italian restaurant, we will need two versions of impact matrix: one for all categories of restaurants, and one specifically for Italian restaurants. Take Italian restaurants for example, the updated impact matrix E_i for Italian restaurants, containing impact scores of Italian restaurants in the other neighborhoods to one in the current neighborhoods were computed, by multiplying the old impact matrix E_o with the number of Italian restaurants in each neighborhood, recorded as a vector v_i, which is the series 'Italian Restaurant' in the 'newyork_restaurants_grouped' data frame

$$E_i = E_o v_i$$

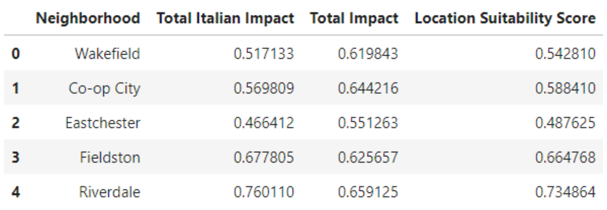
The updated matrix E_i was then transformed into a data frame and visualized as follows. As can be seen, each column represents the impact from a specific neighborhood to the current neighborhood in rows.

	Impact from Wakefield	Impact from Co-op City	Impact from Eastchester	Impact from Fieldston	Impact from Riverdale	Impact from Kingsbridge	Impact from Marble Hill	Impact from Woodlawn	Impact from Norwood	Impact from Williamsbridge	Impact from Baychester	Impact from Pelham Parkway	Impact from City Island	Impact from Bedford Park	Impact from University Heights
0	0.0	0.0	0.0	34.219234	0.0	17.506811	0.0	0.0	54.599253	58.546006	99.696613	0.0	0.0	21.984583	0.
1	0.0	0.0	0.0	25.445098	0.0	13.651333	0.0	0.0	40.372758	35.308068	316.902295	0.0	0.0	17.945338	0.
2	0.0	0.0	0.0	25.564272	0.0	13.290673	0.0	0.0	38.016497	32.951170	135.216384	0.0	0.0	16.593798	0.
3	0.0	0.0	0.0	200.000000	0.0	71.239376	0.0	0.0	62.595335	19.880328	39.753057	0.0	0.0	30.965389	0.
4	0.0	0.0	0.0	240.093497	0.0	74.728190	0.0	0.0	55.746634	17.856383	37.293007	0.0	0.0	29.369151	0.

The impact scores were summed in columns to compute the total impact score of Italian restaurants in NYC to a new Italian restaurant in each neighborhood, and they were added to the 'recommender' data frame. The same procedure applied to the total number of restaurants of all categories as well. The head of the updated 'recommended' data frame is shown below:

	Neighborhood	Total Restaurant	Italian Restaurant	Total Italian Impact	Total Impact
0	Wakefield	6.0	0.0	1395.342676	25853.041878
1	Co-op City	2.0	0.0	1537.474271	26869.592109
2	Eastchester	1.0	0.0	1258.487703	22992.646620
3	Fieldston	3.0	2.0	1828.872576	26095.534581
4	Riverdale	1.0	0.0	2050.952045	27491.463797

Now that the impact scores were derived from the total number of restaurants, we may drop the first two columns. Then simple scaling normalization of the two series of impact scores were conducted as they were on different scales. For the final location suitability score, a weighted sum was adopted. By default, 0.25 was used as the weight for Total Impact, while 0.75 was used for Total Italian impact, since restaurants of the same type may not necessarily compete severely when they are close to each other, but Italian restaurants close to each other will almost surely compete hard against each other. The obtained final scores were added to the data frame. The head of the complete 'recommender' was shown below

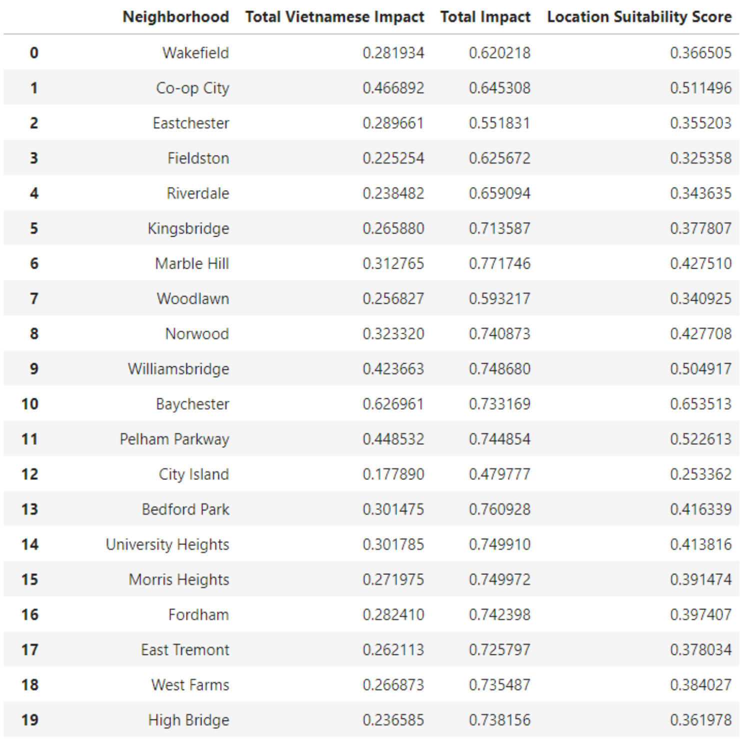


	Neighborhood	Total Italian Impact	Total Impact	Location Suitability Score
0	Wakefield	0.517133	0.619843	0.542810
1	Co-op City	0.569809	0.644216	0.588410
2	Eastchester	0.466412	0.551263	0.487625
3	Fieldston	0.677805	0.625657	0.664768
4	Riverdale	0.760110	0.659125	0.734864

Finally, the entire process of computing Location Suitability Score was integrated into a function named 'LocationSuitabilityScore', for ease of use in reality.

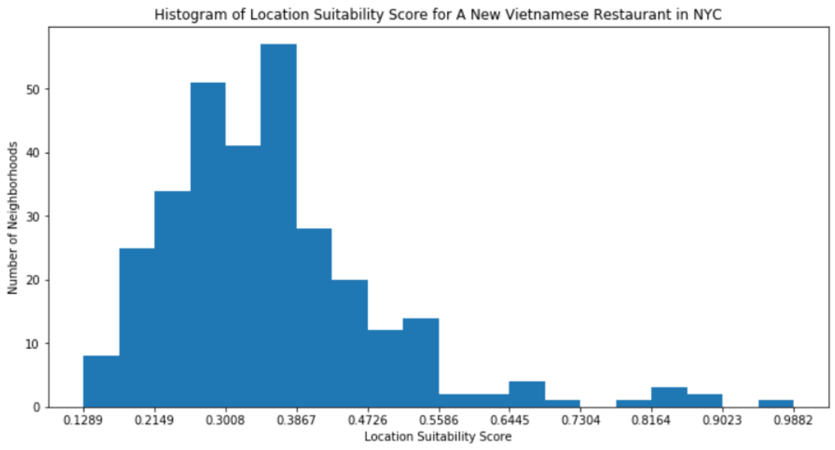
Results：

To test our restaurant location recommender, let's assume that another ambitious restaurant owner would like to open a new Vietnamese restaurant in NYC. He would like to make a preliminary choice of which neighborhood the new restaurant would be in. Using what was discussed in the previous section, notably the 'LocationSuitabilityScore' function, we can obtain a new 'recommender' data frame as follows (only the first 20 rows shown):



	Neighborhood	Total Vietnamese Impact	Total Impact	Location Suitability Score
0	Wakefield	0.281934	0.620218	0.366505
1	Co-op City	0.466892	0.645308	0.511496
2	Eastchester	0.289661	0.551831	0.355203
3	Fieldston	0.225254	0.625672	0.325358
4	Riverdale	0.238482	0.659094	0.343635
5	Kingsbridge	0.265880	0.713587	0.377807
6	Marble Hill	0.312765	0.771746	0.427510
7	Woodlawn	0.256827	0.593217	0.340925
8	Norwood	0.323320	0.740873	0.427708
9	Williamsbridge	0.423663	0.748680	0.504917
10	Baychester	0.626961	0.733169	0.653513
11	Pelham Parkway	0.448532	0.744854	0.522613
12	City Island	0.177890	0.479777	0.253362
13	Bedford Park	0.301475	0.760928	0.416339
14	University Heights	0.301785	0.749910	0.413816
15	Morris Heights	0.271975	0.749972	0.391474
16	Fordham	0.282410	0.742398	0.397407
17	East Tremont	0.262113	0.725797	0.378034
18	West Farms	0.266873	0.735487	0.384027
19	High Bridge	0.236585	0.738156	0.361978

To visualize the distribution of location suitability scores of all 306 NYC neighborhoods, histogram was implemented to the above results.



Evidently, most of the neighborhoods earned scores hanging around 0.2-0.4, while only a very small set of neighborhoods earned high scores of above 0.75 or so. Therefore, the choice of neighborhoods can be narrowed down only to those with very high scores, and the job of restaurant location hunting is therefore made much easier. Alternatively, other restaurant categories can also be tested.

Discussion：

The restaurant location recommender in this project is just a preliminary recommender, as it only provides a single metric based on the neighborhoods. It cannot yet be detailed to a specific street address or store location. However, if more detailed JSON data covering the available retail spaces within every neighborhood would be available, then it is possible to create a more refined recommender. Some parameters, such as the weight factors associated with the impact scores of all categories of restaurants and the specific category were also chosen empirically. Other assumptions include that every neighborhood is NYC is densely populated enough and are all easily accessible. In other words, opening a new restaurant where peer competition is minimal is always more likely to be profitable. This may not be always true in the real world. However, the biggest contribution of this recommender is that it is a handy, simple yet effective tool for narrowing the choice of a new restaurant location down from the entire NYC containing 306 neighborhoods to a much smaller number of neighborhoods. The restaurant owners only need to look at the available commercial spaces in the periphery of these neighborhoods to choose the final location.

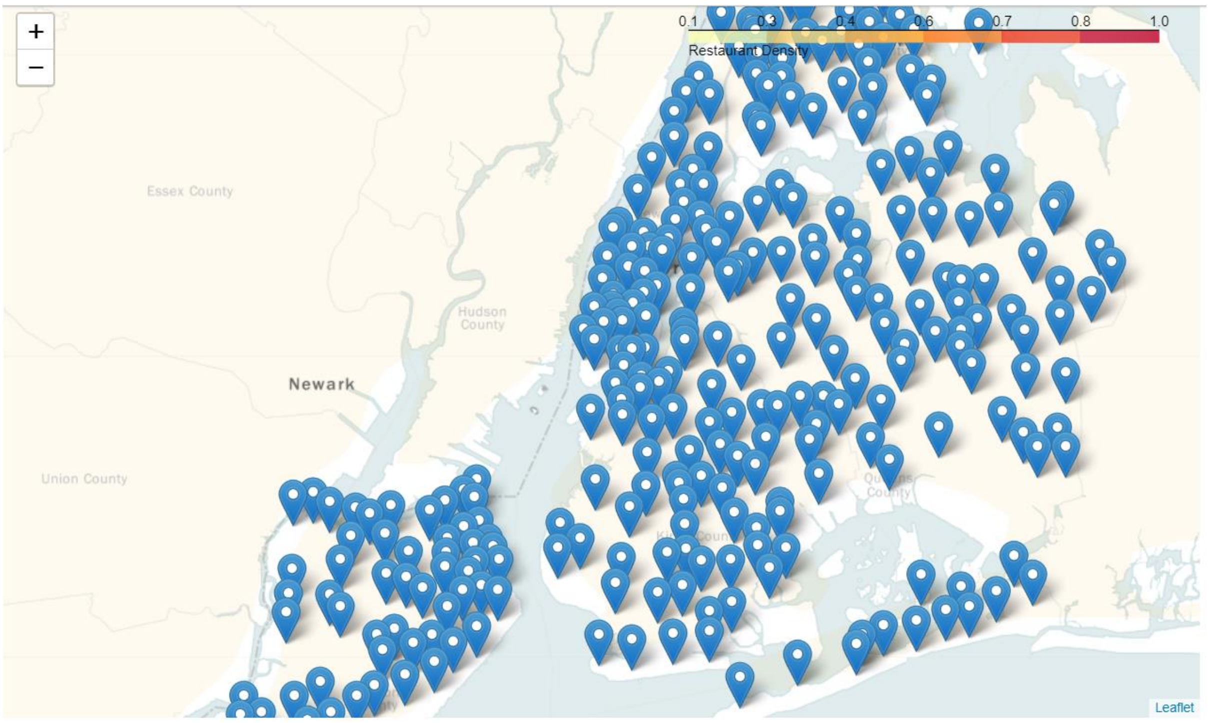
On the other hand, this project had meant to take some more factors from Foursquare location data, such as ratings of a restaurant, and the number of people that have visited a restaurant, which is a symbol of popularity of a restaurant. However, since accessing these information requires premium calls limited to only 500 every day, I eventually gave up on this idea. However, it would be very beneficial in reality to take the ratings and popularity of each restaurant into account. Avoiding being too close to a popular or highly rated restaurant of the same style or cuisine is very important to the sustainability of a new restaurant. Also, in terms of data visualization, I had meant to create a choropleth map for viewing the location suitability score. However, for some reasons, the choropleth map always looked as follows (I have no idea why these blue markers were out there), and I couldn’t find a way through this problem. Overall, the outcome could be even better, but it has already been a lot of efforts to go this far.

Create a choropleth map of Vietnamese Restaurants in New York City

```
vietrestaurant_choropleth_map = folium.Map(location=[latitude, longitude], zoom_start=11, tiles='Mapbox Bright')
newyork_geo = r'newyork_data.json'

desired_type = 'Vietnamese Restaurant'

vietrestaurant_choropleth_map.choropleth(
    geo_data=newyork_geo,
    data=recommender_vietnamese,
    columns=['Neighborhood','Location Suitability Score'],
    key_on='feature.properties.name',
    fill_color='YlOrRd',
    fill_opacity=0.7,
    line_opacity=0.2,
    legend_name='Restaurant Density',
    reset=True
)
vietrestaurant_choropleth_map
```



Conclusion:

All in all, this report addresses the development of a preliminary restaurant location recommender that aims to help future restaurant owner ambitious to open new restaurants in New York City, the greatest and most influential metropolis in the world, to preliminarily select which neighborhoods their new restaurants should be located in. NYC JSON data and Foursquare location data were used to extract information of over 100 categories of restaurants currently available in NYC. Extensive data cleaning and distance matrix were used to establish the algorithm of suitability of neighborhoods for the desired type of restaurants. The outcome includes a set of location suitability scores straightforward to understand, to help owners narrow their choices down to around a small number of neighborhoods.

References:

[1] Restaurant Statistics, NYC Department of Health, 2017