

# Capstone Project - Where to? Phase I

## IBM Datascience Professional Certificate Course

### Table of contents

- [Introduction: Business Problem and Interested Parties](#)
- [Data: Sources and Process](#)
- [Methodology: Algorithms and Analysis](#)
- [Results](#)
- [Conclusion](#)

### Introduction: Business Problem

The purpose of this project is to find the ideal location for a fusion restaurant in Manhattan NY. The project will find a location which is not too crowded with restaurants of that type. This project will identify a few suitable neighborhoods to setup shop. This project is executed for investors who are looking to invest money for this type of venture.

### Data

Following items will factor into the decision making:

- number of existing restaurants in the neighborhood
- number of fusion restaurants and their proximity

Below data sources are used for this:

- neighborhood clustering and segmentation data for Manhattan (pre-existing local file mnhttn\_nghbrhds.tsv)
- **Foursquare API** will be used to determine type and number of restaurants in each neighborhood

### Neighborhoods

List of Manhattan Neighborhoods

```
In [2]: import pandas as pd
mtn_data = pd.read_csv('mnhttn_nghbrhds.tsv', sep='\t')
print(mtn_data.size)
mtn_data.head()
```

200

Out[2]:

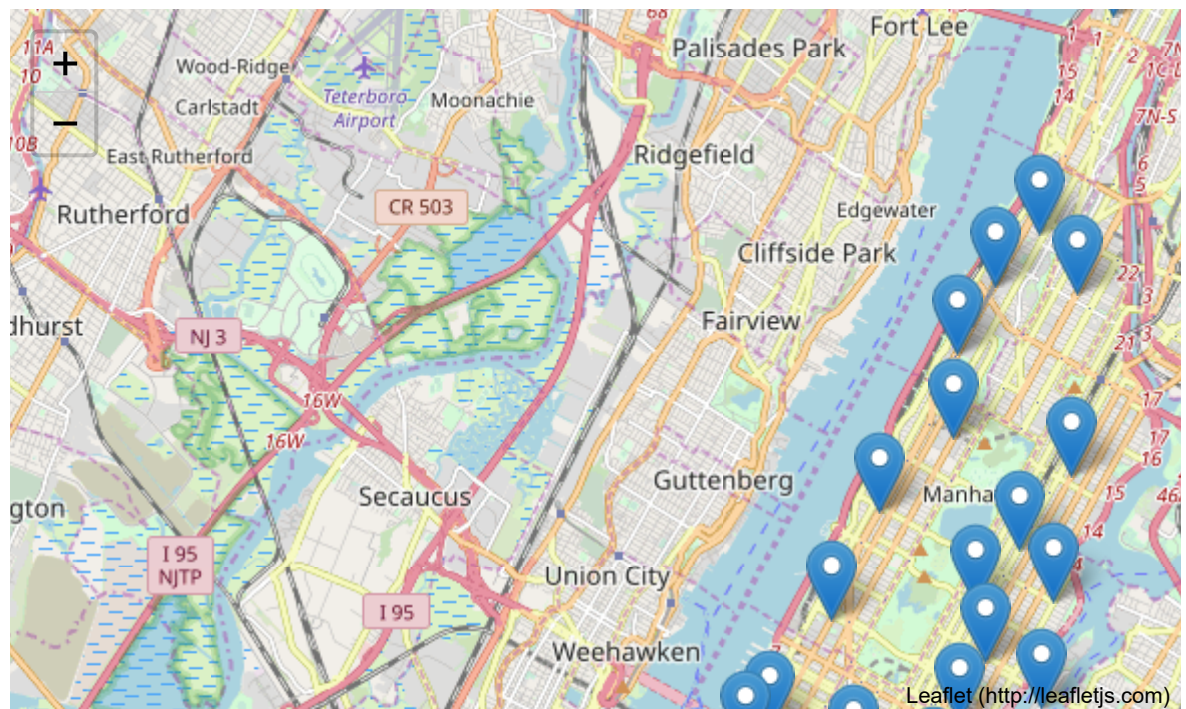
	Unnamed: 0	Borough	Neighborhood	Latitude	Longitude
0	0	Manhattan	Marble Hill	40.876551	-73.910660
1	1	Manhattan	Chinatown	40.715618	-73.994279
2	2	Manhattan	Washington Heights	40.851903	-73.936900
3	3	Manhattan	Inwood	40.867684	-73.921210
4	4	Manhattan	Hamilton Heights	40.823604	-73.949688

```
In [49]: ► import folium
map_mtn = folium.Map(location=[40.7900869, -73.9598295], zoom_start=12)

for lat, lng, label in zip(mtn_data['Latitude'], mtn_data['Longitude'], mtn_c
    folium.Marker(location=[ lat, lng ]).add_to( map_mtn )

map_mtn
```

Out[49]:



## Foursquare

Using foursquare API to get locations and count of restaurants. Since Foursquare doesn't have a fusion category so a google maps search result data was also used to get an idea of the fusion restaurants [URL](https://www.google.com/maps/search/fusion+restaurant/@40.7801874,-74.0368307,12z/data=!4m8!3m7!1s40.7801874,-74.0368307,12z)  
(<https://www.google.com/maps/search/fusion+restaurant/@40.7801874,-74.0368307,12z/data=!4m8!3m7!1s40.7801874,-74.0368307,12z>)

API call output data is **not uploaded** to Github (response json was processed and results are posted on this notebook)

Some sample URLs (the real url was limited to a specific lat,long & radius with api keys included)

- Search Resturants URL <https://api.foursquare.com/v2/venues/search?ll=40.7484,-73.9857&categoryId=4d4b7105d754a06374d81259>  
(<https://api.foursquare.com/v2/venues/search?ll=40.7484,-73.9857&categoryId=4d4b7105d754a06374d81259>)
- Search Fusion Restaurants URL <https://api.foursquare.com/v2/venues/search?ll=40.7484,-73.9857&categoryId=4d4b7105d754a06374d81259&query=fusion>  
(<https://api.foursquare.com/v2/venues/search?ll=40.7484,-73.9857&categoryId=4d4b7105d754a06374d81259&query=fusion>)
- Explore Resturants URL <https://api.foursquare.com/v2/venues/explore?ll=40.7484,-73.9857&categoryId=4d4b7105d754a06374d81259>  
(<https://api.foursquare.com/v2/venues/explore?ll=40.7484,-73.9857&categoryId=4d4b7105d754a06374d81259>)
- Explore Fusion Restaurants URL <https://api.foursquare.com/v2/venues/explore?ll=40.7484,-73.9857&categoryId=4d4b7105d754a06374d81259&query=fusion>  
(<https://api.foursquare.com/v2/venues/explore?ll=40.7484,-73.9857&categoryId=4d4b7105d754a06374d81259&query=fusion>)

```
In [ ]: #code to download foursquare data
#import codecs
#import json
#import urllib.request, json
#for index, row in mtn_data.iterrows():
#    with urllib.request.urlopen("https://api.foursquare.com/v2/venues/search?ll=40.7484,-73.9857&categoryId=4d4b7105d754a06374d81259&query=fusion") as url:
#        data = json.loads(url.read().decode())
#        with codecs.open("foursquare/fuse/"+str(index)+".txt", "w", "utf-8-sig") as text_file:
#            json.dump(data, text_file)
```

```

In [18]:  import json
import ast
from pandas.io.json import json_normalize

venue_map = []
fusion_venue_map = []
uq_ids = set()
uq_fids = set()

for index, row in mtn_data.iterrows():
    with open('foursquare/rest/'+str(index)+'.txt', encoding='utf-8-sig') as fh:
        venue_search_result = json.load(fh)
        for venue in venue_search_result['response']['venues']:
            category = 'U'
            uq_ids.add(venue.get("id"))
            venue_map.append({"neigh":index,"id":venue.get("id"),"name":venue.get("name"),
                              "lat":venue['location']['lat'],'lng':venue['location']['lng']})
    with open('foursquare/fuse/'+str(index)+'.txt', encoding='utf-8-sig') as fh:
        fusion_venue_search_result = json.load(fh)
        for venue in fusion_venue_search_result['response']['venues']:
            category = 'U'
            uq_fids.add(venue.get("id"))
            fusion_venue_map.append({"neigh":index,"id":venue.get("id"),"name":venue.get("name"),
                                     "lat":venue['location']['lat'],'lng':venue['location']['lng']})

all_eateries = pd.DataFrame(venue_map)
all_fusion_eateries = pd.DataFrame(fusion_venue_map)

```

```

In [4]:  import numpy as np

print('Total number of restaurants:',len(uq_ids))
print('Total number of fusion restaurants:', len(uq_fids))
print('Percentage of fusion restaurants: {:.2f}%'.format(len(uq_fids) / len(uq_ids)))

```

```

Total number of restaurants: 929
Total number of fusion restaurants: 6
Percentage of fusion restaurants: 0.65%

```

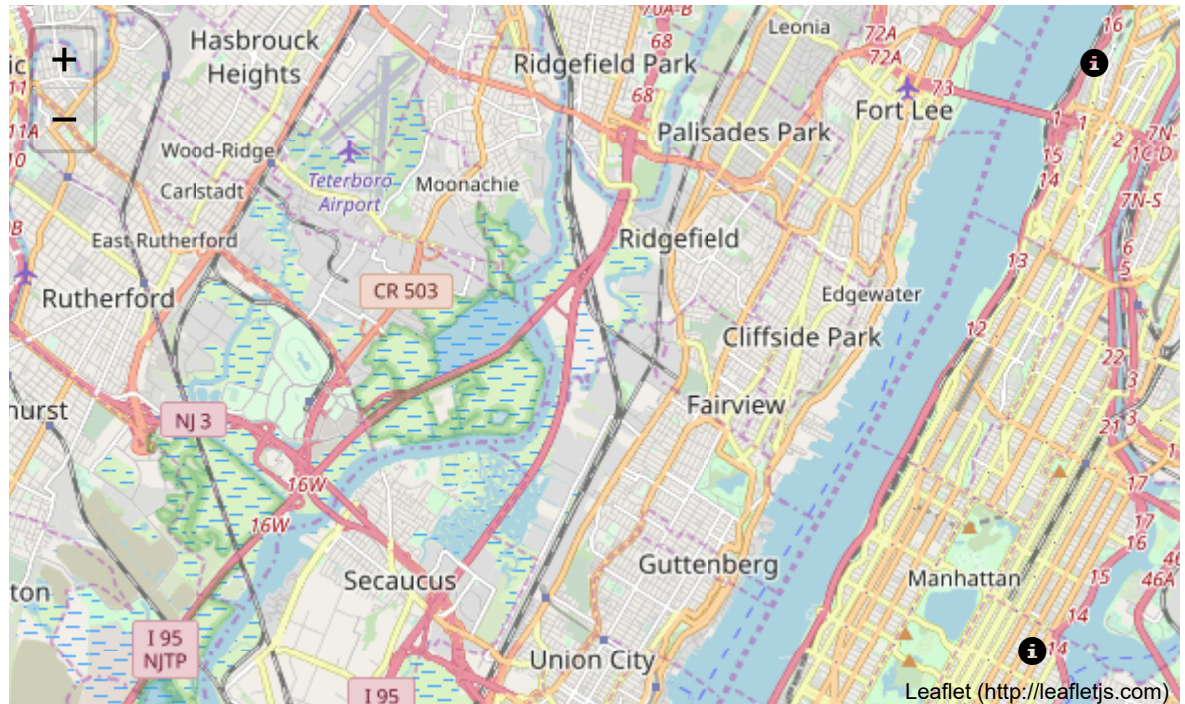
Show all collected data on the map with fusion restaurants in a different color

```
In [52]: ❏ import folium
map_mtn = folium.Map(location=[40.7900869, -73.9598295], zoom_start=12)

#for index, row in all_eateries.iterrows():
#    folium.Marker(location=[ row.Lat, row.Lng ]).add_to( map_mtn )
for index, row in all_fusion_eateries.iterrows():
    folium.Marker(location=[ row.lat, row.lng ],icon=folium.Icon(color='#ff00

map_mtn
```

Out[52]:



## Methodology

With data in hand we will look for desirable places with low restaurant density and with no fusion themed restaurants nearby. A heatmap will be generated to get a good picture of restaurant density. The promising areas will be further filtered using filter criteria.

## Analysis

A count of restaurants in every neighbourhood



```
In [48]: ▶ rest_counts_by_neigh = all_eateries.groupby('neigh').size()
for index, row in mtn_data.iterrows():
    print(str(index)+" "+row['Neighborhood']+"": "+str(rest_counts_by_neigh[index], row['neigh'])

0 Marble Hill: 7
1 Chinatown: 30
2 Washington Heights: 16
3 Inwood: 22
4 Hamilton Heights: 30
5 Manhattanville: 21
6 Central Harlem: 21
7 East Harlem: 30
8 Upper East Side: 17
9 Yorkville: 27
10 Lenox Hill: 24
11 Roosevelt Island: 8
12 Upper West Side: 27
13 Lincoln Square: 30
14 Clinton: 17
15 Midtown: 30
16 Murray Hill: 30
17 Chelsea: 25
18 Greenwich Village: 30
19 East Village: 30
20 Lower East Side: 21
21 Tribeca: 18
22 Little Italy: 30
23 Soho: 12
24 West Village: 30
25 Manhattan Valley: 18
26 Morningside Heights: 30
27 Gramercy: 15
28 Battery Park City: 30
29 Financial District: 30
30 Carnegie Hill: 15
31 Noho: 30
32 Civic Center: 30
33 Midtown South: 30
34 Sutton Place: 30
35 Turtle Bay: 22
36 Tudor City: 9
37 Stuyvesant Town: 4
38 Flatiron: 30
39 Hudson Yards: 23
```

List of neighborhoods with fusion eateries

```
In [47]: all_fusion_eateries.groupby('neigh').size()
```

```
Out[47]: neigh
          2      1
          9      1
         16      1
         18      1
         32      1
         34      1
          dtype: int64
```

At the start of the project the idea was to look for places where there aren't many restaurants but since it looks like there aren't many fusion restaurants we should look for places with quite a few restaurants indicating areas with good chunk of customers but away from existing fusion themed restaurants.

## Results and Discussion

The analysis has shown that even though there are many restaurants in Manhattan there aren't many fusion themed restaurants. Based on the count and cluster there are many promising neighborhoods like Midtown South. Detailed analysis and additional reporting is outside the scope of this project and will be taken up in phase 2 and beyond.

## Conclusion

Purpose of this project was to determine the viability of a fusion restaurant in Manhattan NY. The analysis has shown a few promising neighborhoods. The final decision will be made by investors based on additional factors like rental prices, crime, living vs work areas, walkability and other ROI factors. This concludes Phase I one of the 'Fusion Reactor' project.