

Introduction

This study focuses on recommender systems that use location data to help tourists navigate the physical world. There are various recommendation problems in the literature: recommending new places, recommending the next place to visit, events to attend, and recommending neighbourhoods or large areas to explore further. In this paper, I recommend a tourist restaurants and hotels in the area of Manhattan.

Data

For this study, the Manhattan neighborhood data from the lab(https://geo.nyu.edu/catalog/nyu_2451_34572) is used. The data contained the Borough, the name and latitude, longitude of each neighborhood in Manhattan. The Foursquare API is utilised to get the location information for these neighborhoods.

Methodology

As a database, I used GitHub repository in my study. My master data which has the main components Borough, Average House Price, Latitude and Longitude informations of the city

Various Python libraries are used in this project as presented below.

In [1]:

```
import numpy as np # library to handle data in a vectorized manner

import pandas as pd # library for data analysis
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

import json # library to handle JSON files

!conda install -c conda-forge geopy --yes # uncomment this line if you haven't completed the Foursquare API lab
from geopy.geocoders import Nominatim # convert an address into latitude and longitude values

import requests # library to handle requests
from pandas.io.json import json_normalize # transform JSON file into a pandas dataframe

# 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 # map rendering library

print('Libraries imported.')
```

Libraries imported.

- Next, the data is transformed into a pandas dataframe and the data is downloaded. As mentioned, it contained information about Borough, Neighbourhood, Latitude, and Longitude in Manhattan. Please, see the table below for details.

In [2]:

```
!wget -q -O 'newyork_data.json' https://cocl.us/new_york_dataset
print('Data downloaded!')

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

neighborhoods_data = newyork_data['features']

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

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

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)

neighborhoods.head()
```

Data downloaded!

Out[2]:

	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

- The data gathered comprised of 4 columns and 306 rows.

In [3]:

```
neighborhoods.shape
```

Out[3]:

```
(306, 4)
```

- We retrieved the Neighborhoods in Manhattan

In [4]:

```
manhattan_data = neighborhoods[neighborhoods['Borough'] == 'Manhattan'].reset_index(drop=True)
manhattan_data.head()
```

Out[4]:

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

- geopy library is used to get the latitude and longitude values of Manhattan

In [5]:

```
address = 'Manhattan, NY'

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

The geograpical coordinate of Manhattan are 40.7900869, -73.9598295.

- The map of Manhattan is created using latitude and longitude values

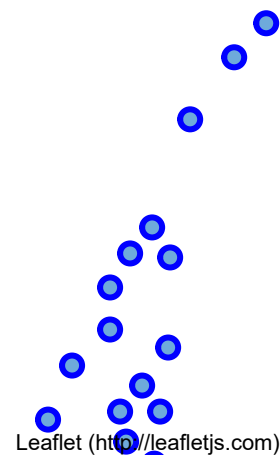
In [6]:

```
# create map of Manhattan using Latitude and Longitude values
map_manhattan = folium.Map(location=[latitude, longitude], zoom_start=11)

# add markers to map
for lat, lng, label in zip(manhattan_data['Latitude'], manhattan_data['Longitude'], manhattan_data['Neighborhood']):
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(map_manhattan)

map_manhattan
```

Out[6]:



- Now, we have the neighborhood data(name, Latitude, Longitude of the neighborhood in Manhattan).
- Next we will utilize the foursquare API to further explore/search the area.

1. Exploring Neighborhoods in Manhattan

In [7]:

```
# @hide cell
CLIENT_ID = 'YCZ2PFBDENOXZUX25S41HS3ZWHGMA4WDY3BTDEXFMDUFZJUB' # your Foursquare ID
CLIENT_SECRET = 'U1EUP1HXLFDAXUZVTNPTXDGTSTBZD3BPBLGFOY5WWPLBJSNF' # your Foursquare Secret
VERSION = '20180605' # Foursquare API version
```

In [8]:

```
radius=500
LIMIT=100

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)
```

1. Listing Venues in Manhattan

In [9]:

```
manhattan_venues = getNearbyVenues(names=manhattan_data['Neighborhood'],  
                                   latitudes=manhattan_data['Latitude'],  
                                   longitudes=manhattan_data['Longitude']  
                                   )
```

Marble Hill
Chinatown
Washington Heights
Inwood
Hamilton Heights
Manhattanville
Central Harlem
East Harlem
Upper East Side
Yorkville
Lenox Hill
Roosevelt Island
Upper West Side
Lincoln Square
Clinton
Midtown
Murray Hill
Chelsea
Greenwich Village
East Village
Lower East Side
Tribeca
Little Italy
Soho
West Village
Manhattan Valley
Morningside Heights
Gramercy
Battery Park City
Financial District
Carnegie Hill
Noho
Civic Center
Midtown South
Sutton Place
Turtle Bay
Tudor City
Stuyvesant Town
Flatiron
Hudson Yards

Results and Discussion

Our analysis shows that there is a great number of restaurants in Manhattan. The results of this study could inform the any tourist who would like to go to a hotels with a nice resataurants around there in Manhattan.

Conclusion

Purpose of this project was to identify the restourants around top hotels of Manhattan. Various restaurants with hotels are identified using the methodology mentioned above. The outcomes of this study could inform the any tourist who would like togo to a hotels with a nice resataurants around there in Manhattan.

In []: