IBM Data Science Capstone Project

This Project is to create a model which can recommend the nearest neighborhood in canada to move to using machine learning alogrithms and geo api from foursquare

---

Business Case:-

- **Help people to find the nearst neighbourhood to move to in canada**
  - which most people face an issue to find the best neigbourhood to move to
  - and this issue if sovled will make it easier for more people to move and find another work easly
  - create a new life in the new place without a problem and people lives will be better

Data:-

- **Postal code of Canada**
- **Avenues data from fouresquare api**
  - these data will help find the coordinates of the neighbourhood and get the top venues
  - which people can move to without and issue to face and see the map describtion for the avenue

Table of Content:

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from geopy.geocoders import Nominatim
import folium
from pandas.io.json import json_normalize
import requests
import matplotlib.cm as cm
import matplotlib.colors as colors

from sklearn.cluster import KMeans
```

In [2]:
```
data = pd.read_html("https://en.wikipedia.org/w/index.php?title=List_of_postal_codes
df = data[0]
df.head()
```

Out[2]:

	Postal Code	Borough	Neighbourhood
0	M1A	Not assigned	Not assigned
1	M2A	Not assigned	Not assigned

	Postal Code	Borough	Neighbourhood
2	M3A	North York	Parkwoods
3	M4A	North York	Victoria Village
4	M5A	Downtown Toronto	Regent Park, Harbourfront

Cleaning and Data Preparing

In [3]:
```
# cleaning
df = df[df['Borough'] != 'Not assigned']
df.head(10)
```

Out[3]:

	Postal Code	Borough	Neighbourhood
2	M3A	North York	Parkwoods
3	M4A	North York	Victoria Village
4	M5A	Downtown Toronto	Regent Park, Harbourfront
5	M6A	North York	Lawrence Manor, Lawrence Heights
6	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government
8	M9A	Etobicoke	Islington Avenue, Humber Valley Village
9	M1B	Scarborough	Malvern, Rouge
11	M3B	North York	Don Mills
12	M4B	East York	Parkview Hill, Woodbine Gardens
13	M5B	Downtown Toronto	Garden District, Ryerson

In [4]:
```
df[df['Neighbourhood'] == 'Not assigned']
```

Out[4]:

	Postal Code	Borough	Neighbourhood

In [5]:
```
df.shape
```

Out[5]:
```
(103, 3)
```

Import Data Cordinates

In [6]:
```
df_cor = pd.read_csv('https://cocl.us/Geospatial_data')
```

In [7]:
```
df_cor.head()
```

Out[7]:

	Postal Code	Latitude	Longitude
0	M1B	43.806686	-79.194353
1	M1C	43.784535	-79.160497

	Postal Code	Latitude	Longitude
2	M1E	43.763573	-79.188711
3	M1G	43.770992	-79.216917
4	M1H	43.773136	-79.239476

In [8]:
```
df_cor.set_index('Postal Code')
```

Out[8]:

Postal Code	Latitude	Longitude
M1B	43.806686	-79.194353
M1C	43.784535	-79.160497
M1E	43.763573	-79.188711
M1G	43.770992	-79.216917
M1H	43.773136	-79.239476
...	...	...
M9N	43.706876	-79.518188
M9P	43.696319	-79.532242
M9R	43.688905	-79.554724
M9V	43.739416	-79.588437
M9W	43.706748	-79.594054

103 rows × 2 columns

join postal codes data with cordinates data

In [9]:
```
p_data = df.join(df_cor.set_index('Postal Code'), on='Postal Code')
p_data.reset_index(drop = True, inplace=True)
p_data.head()
```

Out[9]:

	Postal Code	Borough	Neighbourhood	Latitude	Longitude
0	M3A	North York	Parkwoods	43.753259	-79.329656
1	M4A	North York	Victoria Village	43.725882	-79.315572
2	M5A	Downtown Toronto	Regent Park, Harbourfront	43.654260	-79.360636
3	M6A	North York	Lawrence Manor, Lawrence Heights	43.718518	-79.464763
4	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.662301	-79.389494

Define fouresquare cerdentials

In [10]:
```
CLIENT_ID = 'QMFYKU4KSWVFRMNFCPSMQR2Q3SBS55EGZNEQKKERK1G02530' # your Foursquare ID
```

```
CLIENT_SECRET = '2044ODGCFHSQ0WBXMIGYKXCOBG4FZIF13OAIAHJNUHK5Q00J' # your Foursquare
ACCESS_TOKEN = 'RHKVHHQNZPLMSTJ3IEY5Q1JVKWEW5QIHNBZUEVQ2CEVAWOXW' # your FourSquare
VERSION = '20180604'
LIMIT = 100
radius = 500
```

define a function which will be used to import venues data

In [11]:

```
def venues_(name, latitude, longitude, radius=500):

    venues_list = []

    for name, latitude, longitude in zip(name, latitude, longitude):

        url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_sec
            CLIENT_ID, CLIENT_SECRET,
            latitude, longitude,
            ACCESS_TOKEN, VERSION,
            radius, LIMIT)

        response = requests.get(url).json()['response']['groups'][0]['items']

        venues_list.append([(name, latitude, longitude, v['venue']['name'],
                             v['venue']['location']['lat'],
                             v['venue']['location']['lng'],
                             v['venue']['categories'][0]['name']) for v in respon

    df_venues = pd.DataFrame([item for venue_list in venues_list for item in venue_l
    df_venues.columns=['Neighbourhood', 'Neighbourhood_lat', 'Neighbourhood_lng',
                       'Venue', 'Venue_lat', 'Venue_lng', 'Venue_cat']

    return(df_venues)
```

In [12]:

```
combiend_df = venues_(p_data['Neighbourhood'], p_data['Latitude'], p_data['Longitude

combiend_df.head()
```

Out[12]:

	Neighbourhood	Neighbourhood_lat	Neighbourhood_lng	Venue	Venue_lat	Venue_lng	Ve
0	Parkwoods	43.753259	-79.329656	Brookbanks Park	43.751976	-79.332140	
1	Parkwoods	43.753259	-79.329656	Careful & Reliable Painting	43.752622	-79.331957	Con Lan
2	Parkwoods	43.753259	-79.329656	649 Variety	43.754513	-79.331942	Con
3	Parkwoods	43.753259	-79.329656	Towns On The Ravine	43.754754	-79.332552	
4	Parkwoods	43.753259	-79.329656	Sun Life	43.754760	-79.332783	Con Lan

In [13]:

```
combiend_df.groupby('Neighbourhood').count()
```

Out[13]:

Neighbourhood	Neighbourhood_lat	Neighbourhood_lng	Venue	Venue_lat	Venue_lng	Venue_cat

Neighbourhood	Neighbourhood_lat	Neighbourhood_lng	Venue	Venue_lat	Venue_lng	Venue_cat
Neighbourhood						
Agincourt	7	7	7	7	7	7
Alderwood, Long Branch	12	12	12	12	12	12
Bathurst Manor, Wilson Heights, Downsview North	34	34	34	34	34	34
Bayview Village	6	6	6	6	6	6
Bedford Park, Lawrence Manor East	53	53	53	53	53	53
...	...	...	...	...	...	...
Willowdale, Willowdale West	9	9	9	9	9	9
Woburn	4	4	4	4	4	4
Woodbine Heights	16	16	16	16	16	16
York Mills West	5	5	5	5	5	5
York Mills, Silver Hills	3	3	3	3	3	3

98 rows × 6 columns

In [14]:
```
len(combiend_df['Venue_cat'].unique())
```

Out[14]: 319

In [15]:
```
combiend_df_onehot = pd.get_dummies(combiend_df[['Venue_cat']], prefix="", prefix_se

combiend_df_onehot['Neighbourhood'] = combiend_df['Neighbourhood']

# move neighborhood file to the first column

neigh_col = [combiend_df_onehot.columns[-1]] + list(combiend_df_onehot.columns[:-1])
combiend_df_onehot = combiend_df_onehot[neigh_col]

combiend_df_onehot.head()
```

Out[15]:

	Neighbourhood	ATM	Accessories Store	Adult Boutique	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airp Termi
0	Parkwoods	0	0	0	0	0	0	0	0	
1	Parkwoods	0	0	0	0	0	0	0	0	
2	Parkwoods	0	0	0	0	0	0	0	0	
3	Parkwoods	0	0	0	0	0	0	0	0	

	Neighbourhood	ATM	Accessories Store	Adult Boutique	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airp Termi
4	Parkwoods	0	0	0	0	0	0	0	0	

5 rows × 320 columns

In [16]:
```
toronto_df = combiend_df_onehot.groupby('Neighbourhood').mean().reset_index()
toronto_df
```

Out[16]:

	Neighbourhood	ATM	Accessories Store	Adult Boutique	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Ai Ter
0	Agincourt	0.0000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
1	Alderwood, Long Branch	0.0000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
2	Bathurst Manor, Wilson Heights, Downsview North	0.0000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
3	Bayview Village	0.0000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
4	Bedford Park, Lawrence Manor East	0.0000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
...	...	...	...	...	...	...	...	...	...	
93	Willowdale, Willowdale West	0.0000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
94	Woburn	0.0000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
95	Woodbine Heights	0.0625	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
96	York Mills West	0.0000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
97	York Mills, Silver Hills	0.0000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

98 rows × 320 columns

Get The top venues data

In [17]:
```
def top_venues(data, num):
    df_venues = data.iloc[1:]
    df_venues_sorted = df_venues.sort_values(ascending=False)

    return df_venues_sorted.index.values[0: num]
```

In [18]:
```
num_venues = 10

col_ind = ['st', 'nd', 'rd']
```

```
cols = ['Neighbourhood']
for i in np.arange(num_venues):
    try:
        cols.append('{}{} Common Venues'.format(i+1, col_ind[i]))
    except:
        cols.append('{}th Common Venues'.format(i+1))

neigh_venues_sorted = pd.DataFrame(columns=cols)
neigh_venues_sorted['Neighbourhood'] = toronto_df['Neighbourhood']

for i in np.arange(toronto_df.shape[0]):
    neigh_venues_sorted.iloc[i, 1:] = top_venues(toronto_df.iloc[i, :], num_venues)


neigh_venues_sorted.head()
```

Out[18]:

	Neighbourhood	1st Common Venues	2nd Common Venues	3rd Common Venues	4th Common Venues	5th Common Venues	6th Common Venues	7th Common Venues
0	Agincourt	Latin American Restaurant	Hardware Store	Skating Rink	Lounge	Fireworks Store	Clothing Store	Breakfast Spo
1	Alderwood, Long Branch	Pizza Place	Playground	Gym	Skating Rink	Sandwich Place	Athletics & Sports	Coffee Shop
2	Bathurst Manor, Wilson Heights, Downsview North	Ice Cream Shop	Mobile Phone Shop	Pharmacy	Spa	Coffee Shop	Bank	Sush Restauran
3	Bayview Village	Chinese Restaurant	Gym	Bank	Spa	Café	Japanese Restaurant	Neighborhood
4	Bedford Park, Lawrence Manor East	Spa	Italian Restaurant	Pizza Place	Massage Studio	Sushi Restaurant	Business Service	Sandwich Place

Cluster the avenue data

In [19]:
```
toronto_df_clusters = toronto_df.drop('Neighbourhood', 1)

clusters = KMeans(n_clusters=5, random_state=0).fit(toronto_df_clusters)

clusters.labels_[0:20]
```

Out[19]:
```
array([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 1, 1, 1, 1, 1, 1, 1, 1, 1])
```

In [20]:
```
neigh_venues_sorted.insert(0, 'Cluster Labels', clusters.labels_)

toronto_data = p_data

toronto_data = toronto_data.join(neigh_venues_sorted.set_index('Neighbourhood'), on=

toronto_data.head()
```

Out[20]:

	Postal Code	Borough	Neighbourhood	Latitude	Longitude	Cluster Labels	1st Common Venues	2nd Common Venues	Co

	Postal Code	Borough	Neighbourhood	Latitude	Longitude	Cluster Labels	1st Common Venues	2nd Common Venues	Co
78	M1S	Scarborough	Agincourt	43.794200	-79.262029	1	Latin American Restaurant	Hardware Store	
93	M8W	Etobicoke	Alderwood, Long Branch	43.602414	-79.543484	1	Pizza Place	Playground	
28	M3H	North York	Bathurst Manor, Wilson Heights, Downsview North	43.754328	-79.442259	1	Ice Cream Shop	Mobile Phone Shop	Ph
39	M2K	North York	Bayview Village	43.786947	-79.385975	1	Chinese Restaurant	Gym	
55	M5M	North York	Bedford Park, Lawrence Manor East	43.733283	-79.419750	1	Spa	Italian Restaurant	

City Which to find the neighbourhood and nearest avenues

In [23]:

```
# address can be changed to the required city
address = 'Toronto'

geolocator = Nominatim(user_agent='explorer')

location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
```

Visualize The Data

In [22]:

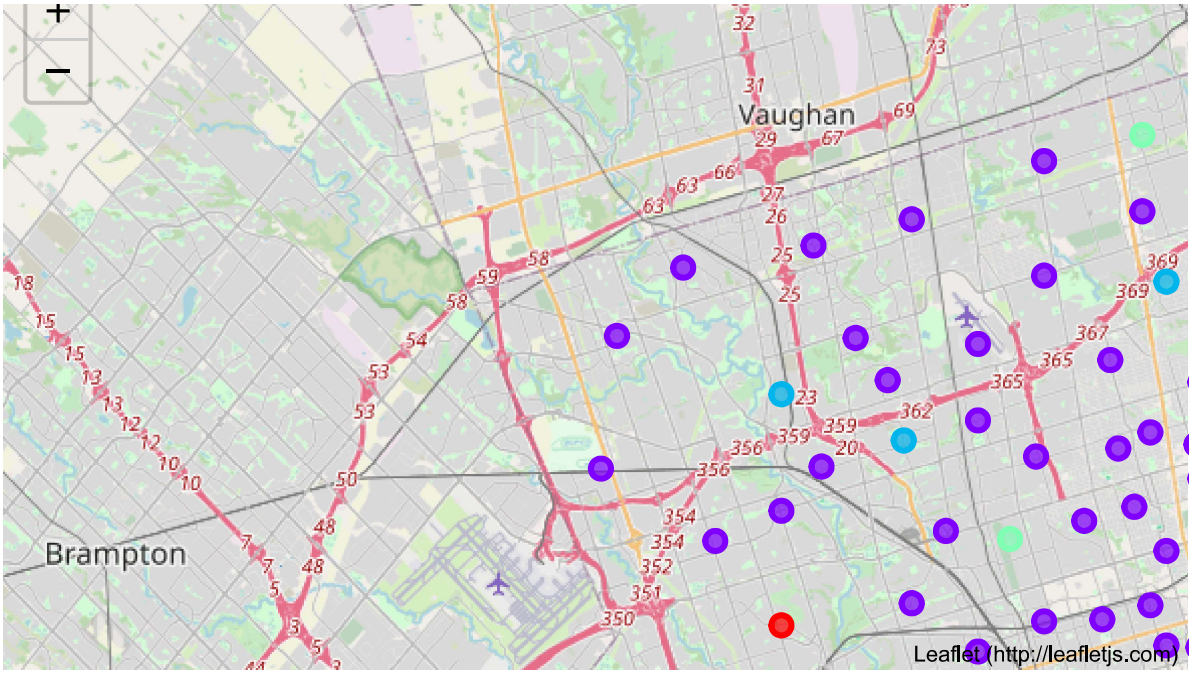
```
map_clusters = folium.Map(location=[latitude, longitude], zoom_start=11)

# set color scheme for the clusters
x = np.arange(5)
ys = [i + x + (i*x)**2 for i in range(5)]
colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
rainbow = [colors.rgb2hex(i) for i in colors_array]

# add markers to the map
markers_colors = []
for lat, lon, poi, cluster in zip(toronto_data['Latitude'], toronto_data['Longitude'
    label = folium.Popup(str(poi), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7).add_to(map_clusters)

map_clusters
```

Out[22]:


Vaughan
Brampton
Leaflet (http://leafletjs.com)