Coursera Capstone Project: Applied Data Science

Yader Rafael Carrillo Jaime

yaderrafa@gmail.es

National Autonomous University of Nicaragua, Managua

1) Introduction/Business Problem

Around the world, hundreds of people are trying every day to open small and medium businesses. No matter in what city they are planning to do it, they will look for the best place with the aim of increase their earnings. The present project, is directed to help future entrepreneurs in order to choose the best location to build their businesses in New York City, through providing data about neighborhoods' characteristics and common venues to set up the venture.

It should be noted that to reach this goal, we need to follow a particular structure to show the results. In this case, we were claimed to follow the typical Data science methodology. I hope to do the best of myself along the project.

2) Data

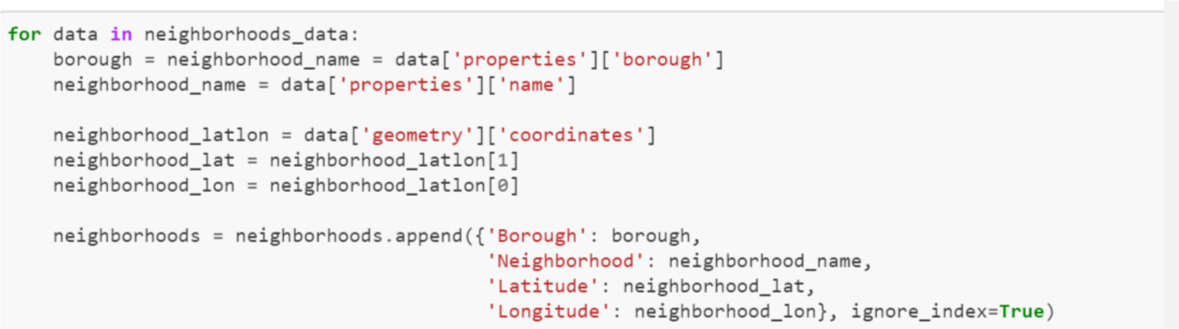
To reach the goal of this project and provide information to stakeholders, I'll be using New York data and Foursquare API to extract competitors on the same neighborhoods.
New York data can be found here https://geo.nyu.edu/catalog/nyu_2451_34572

2.1 Neighborhoods

The data of the neighborhoods in New York can be extracted from JSON file found in https://cocl.us/new_york_dataset.

2.2 Geopy library

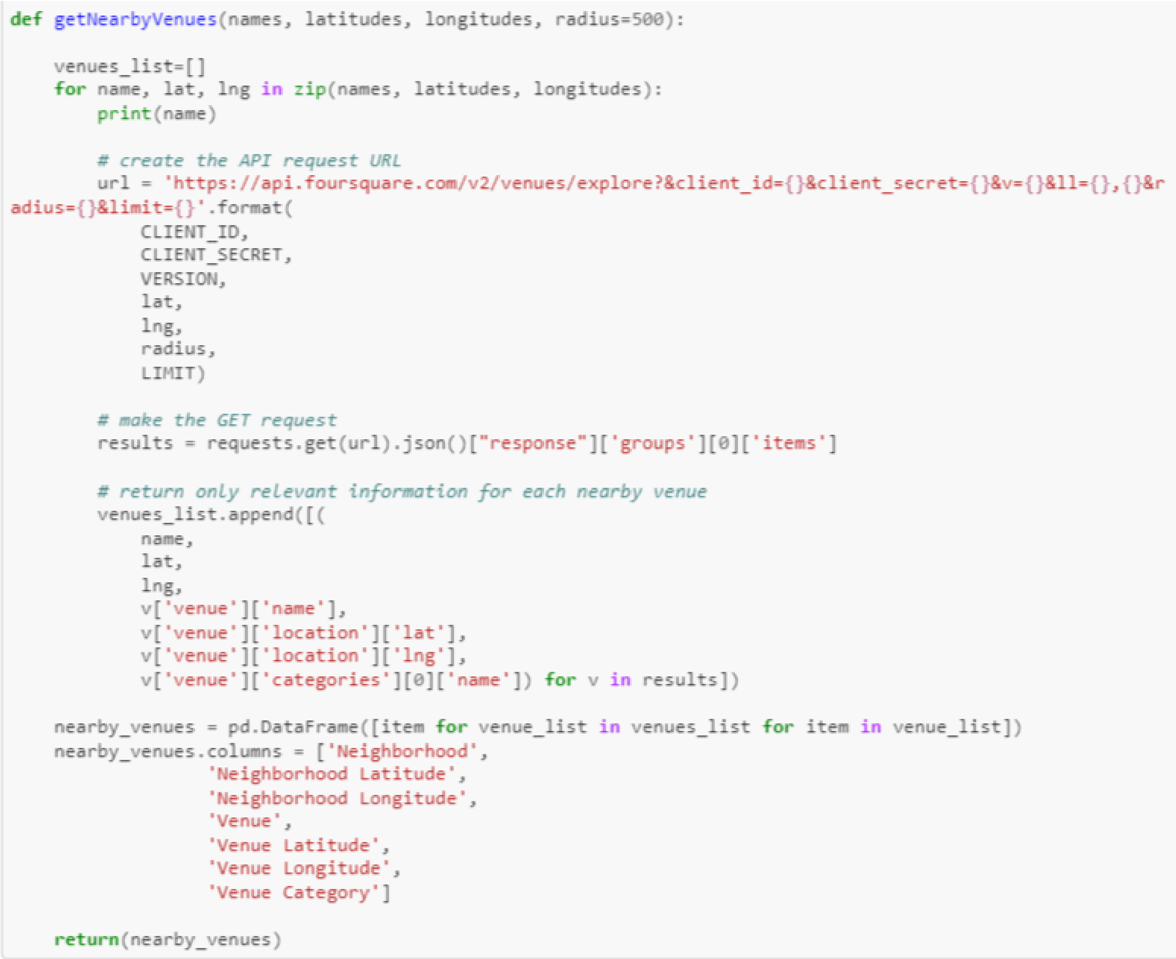
I used this library to get Bronx latitude and longitude



```
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)
```

2.3 Venue Data

From the location data obtained previously, the venue data is found out by passing in the required parameters to the FourSquare API, and creating another Data Frame to contain all the venue details along with the respective neighborhoods.



```
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={},{}&r
adius={}&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)
```

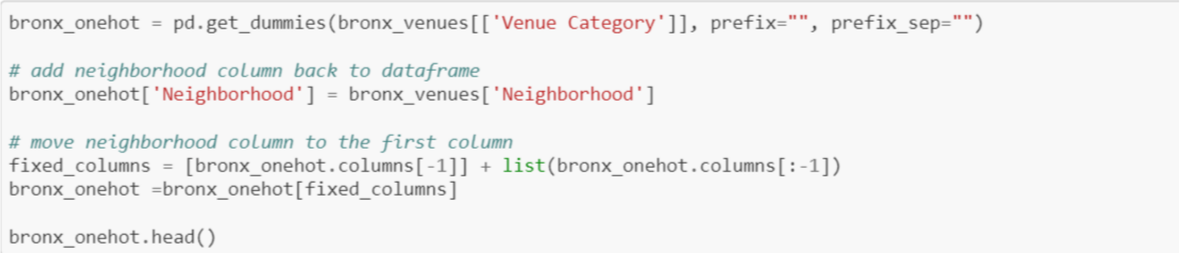
3. Methodology

3.1 Folium

Folium builds on the data wrangling strengths of the Python ecosystem and the mapping strengths of the leaflet.js library. All cluster visualization is done with help of Folium which in turn generates a Leaflet map made using OpenStreetMap technology.

3.2 One hot encoding

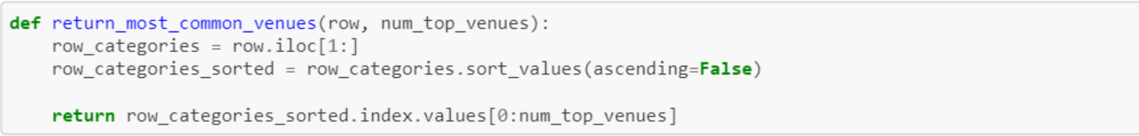
One hot encoding is a process by which categorical variables are converted into a form that could be provided to ML algorithms to do a better job in prediction. For the K-means Clustering Algorithm, all unique items under Venue Category are one-hot encoded.



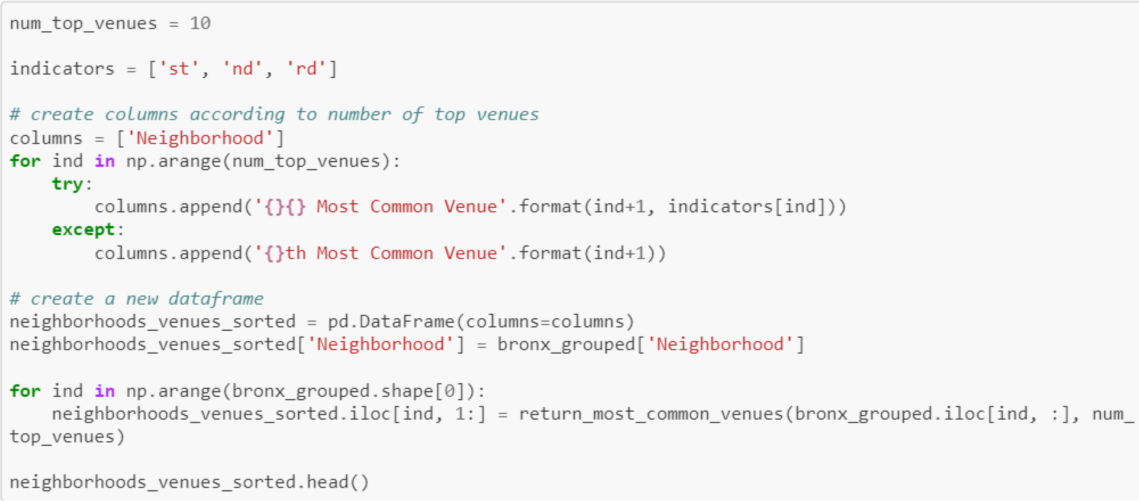
```
bronx_onehot = pd.get_dummies(bronx_venues[['Venue Category']], prefix="", prefix_sep="")

# add neighborhood column back to dataframe
bronx_onehot['Neighborhood'] = bronx_venues['Neighborhood']

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

bronx_onehot.head()
```

3.3 Top 10 most common venues

Due to high variety in the venues, only the top 10 common venues are selected and a new Data Frame is made, which is used to train the K-means Clustering Algorithm.



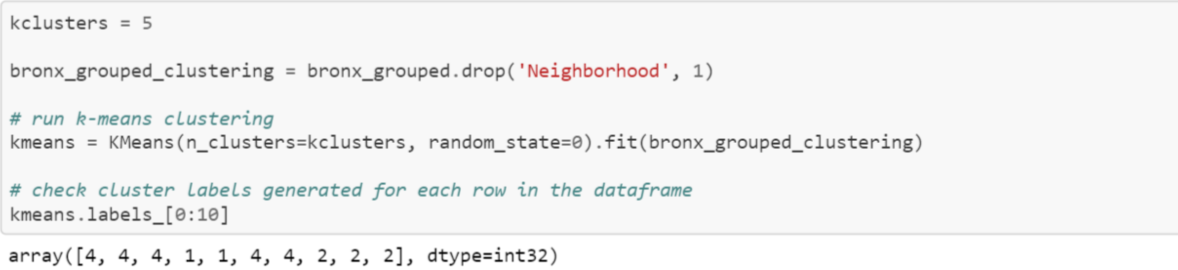
```
def return_most_common_venues(row, num_top_venues):
    row_categories = row.iloc[1:]
    row_categories_sorted = row_categories.sort_values(ascending=False)

    return row_categories_sorted.index.values[0:num_top_venues]
```



```
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'] = bronx_grouped['Neighborhood']

for ind in np.arange(bronx_grouped.shape[0]):
    neighborhoods_venues_sorted.iloc[ind, 1:] = return_most_common_venues(bronx_grouped.iloc[ind, :], num_
top_venues)

neighborhoods_venues_sorted.head()
```

3.4 K-means clustering

The venue data is then trained using K-means Clustering Algorithm to get the desired clusters to base the analysis on. K-means was chosen as the variables (Venue Categories) are huge, and in such situations K-means will be computationally faster than other clustering algorithms.

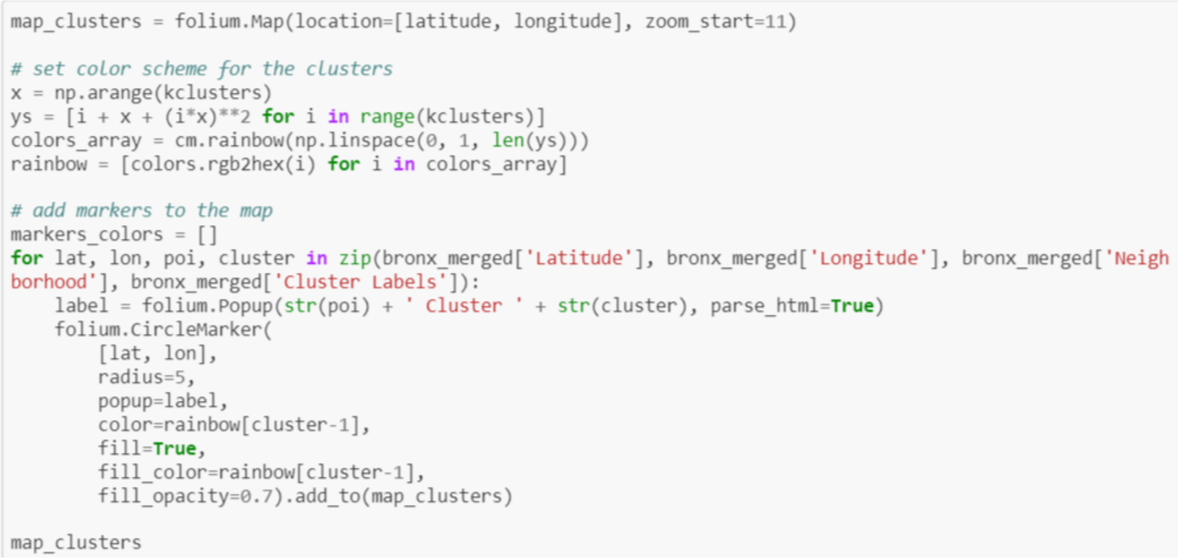


```
kclusters = 5

bronx_grouped_clustering = bronx_grouped.drop('Neighborhood', 1)

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

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

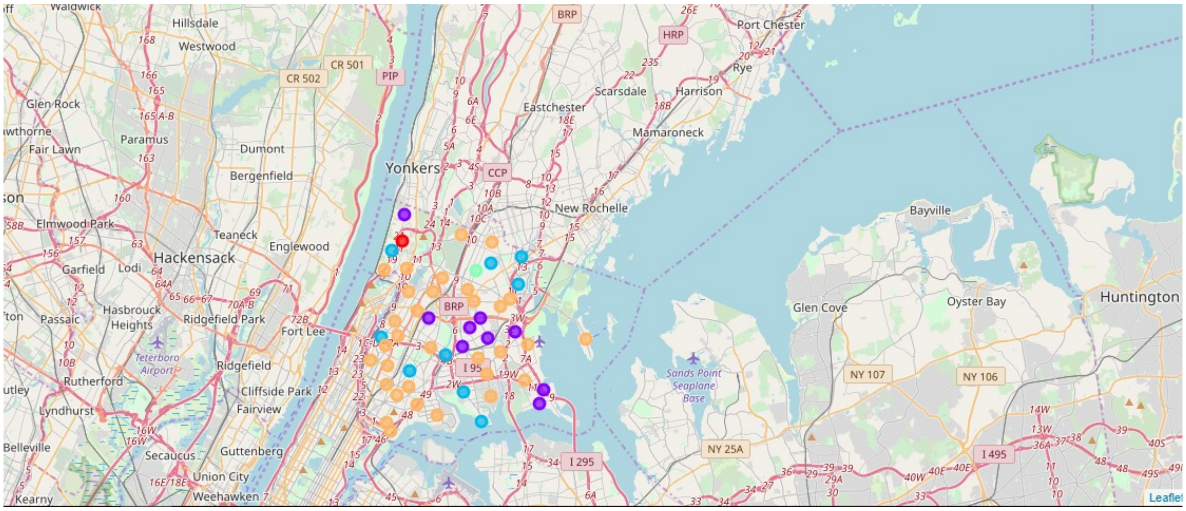
```
array([4, 4, 4, 1, 1, 4, 4, 2, 2, 2], dtype=int32)
```

4) Results

The neighborhoods are divided into n clusters where n is the number of clusters found using the optimal approach. The clustered neighborhoods are visualized using different colors so as to make them distinguishable

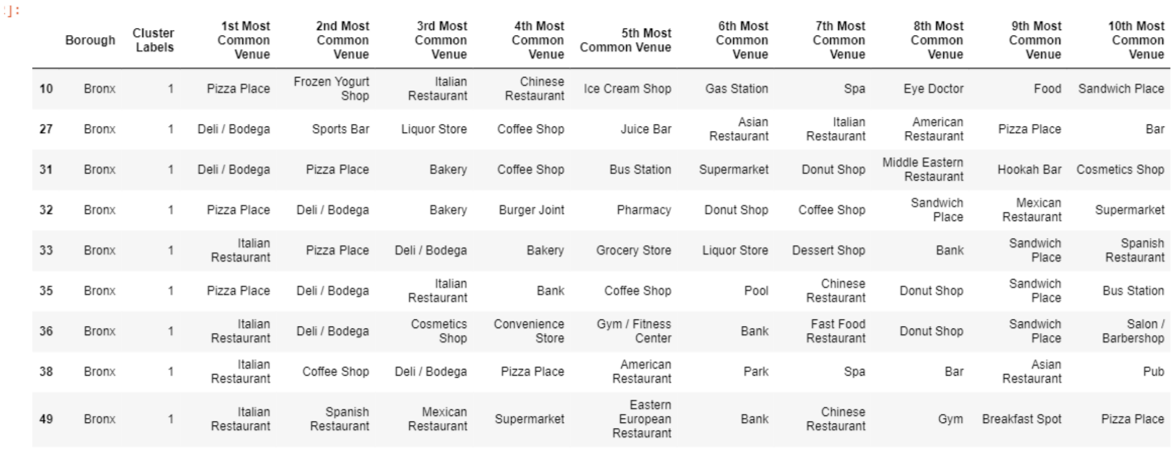


```
map_clusters = folium.Map(location=[latitude, longitude], zoom_start=11)

# set color scheme for the clusters
x = np.arange(kclusters)
ys = [i + x + (i*x)**2 for i in range(kclusters)]
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(bronx_merged['Latitude'], bronx_merged['Longitude'], bronx_merged['Neigh
borhood'], bronx_merged['Cluster Labels']):
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), 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
```



6 Discussion

After analyzing the various clusters produced by the Machine learning algorithm, cluster no 4, is a prime fit to solving the problem of finding a cluster with common venue as a restaurant mentioned before.



	Borough	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	9th Most Common Venue	10th Most Common Venue
10	Bronx	1	Pizza Place	Frozen Yogurt Shop	Italian Restaurant	Chinese Restaurant	Ice Cream Shop	Gas Station	Spa	Eye Doctor	Food	Sandwich Place
27	Bronx	1	Deli / Bodega	Sports Bar	Liquor Store	Coffee Shop	Juice Bar	Asian Restaurant	Italian Restaurant	American Restaurant	Pizza Place	Bar
31	Bronx	1	Deli / Bodega	Pizza Place	Bakery	Coffee Shop	Bus Station	Supermarket	Donut Shop	Middle Eastern Restaurant	Hookah Bar	Cosmetics Shop
32	Bronx	1	Pizza Place	Deli / Bodega	Bakery	Burger Joint	Pharmacy	Donut Shop	Coffee Shop	Sandwich Place	Mexican Restaurant	Supermarket
33	Bronx	1	Italian Restaurant	Pizza Place	Deli / Bodega	Bakery	Grocery Store	Liquor Store	Dessert Shop	Bank	Sandwich Place	Spanish Restaurant
35	Bronx	1	Pizza Place	Deli / Bodega	Italian Restaurant	Bank	Coffee Shop	Pool	Chinese Restaurant	Donut Shop	Sandwich Place	Bus Station
36	Bronx	1	Italian Restaurant	Deli / Bodega	Cosmetics Shop	Convenience Store	Gym / Fitness Center	Bank	Fast Food Restaurant	Donut Shop	Sandwich Place	Salon / Barbershop
38	Bronx	1	Italian Restaurant	Coffee Shop	Deli / Bodega	Pizza Place	American Restaurant	Park	Spa	Bar	Asian Restaurant	Pub
49	Bronx	1	Italian Restaurant	Spanish Restaurant	Mexican Restaurant	Supermarket	Eastern European Restaurant	Bank	Chinese Restaurant	Gym	Breakfast Spot	Pizza Place

Nine neighborhoods called: Pelham Parkway, Morris Park, Van Nest, Throgs Neck, Belmont, North Riverdale, Pelham Bay, Edgewater Park, Bronxdale are the best places to set up the venture in Bronx, New York City.