Final Report

Capstone – The Battle of Neighborhoods

Finding a Suitable Place in Scarborough, Toronto

1. Introduction:

The purpose of this Project is to help people to explore better facilities around their neighborhood. It will also help people to make smart & informed decision on selecting good neighbourhoods in Scarborough, Toronto.

Lots of people are migrating to various provinces in Canada and need lot of exploration to find better housing prices and good schools for their children. This project would help those who are looking for better neighbourhoods to accommodate themselves.

This Project aim to create an analysis of features for people relocating to Scarborough to search for a better neighbourhood.

The features Include:

- Median Housing Price
- Better Schools
- Road Connectivity
- Water Resources
- Recreational Facilities

Data Section

Why Data?

Without leveraging data to make informed decisions about finding a better place to stay and settle down, one could spend countless hours of roaming around the neighborhood, consulting real estate agents with their own biased approach & exploitation and end up in a not so ideal location.

Data will provide better answers and better solutions to the task at hand.

Outcomes

The data enable someone to make an informed decision based on the proximity of schools, restaurants, shopping centres, grocery stores etc. and help to narrow down a better location.

Obtain the Data

- Research and find suitable sources for the neighbour data for Scarborough.
- Access and explore the data to determine if it can be manipulated for our purposes
- Data source link for Scarborough dataset: https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M

Initial Data Wrangling and Cleaning

- Clean the data and convert to a usable form as a dataframe

Data Analysis and Location Data

- Foursquare location data will be leveraged to explore neighborhood of Scarborough
- Gather information about venues (Name & Category) inside each neighbourhood by connecting to Foursquare API
- Data manipulation and analysis
- Identifying median house price and better schools

Visualization

- Analysis and plotting visualizations
- Data visualization using various mapping libraries
- Display of maps using Folium

Import and Install the Required Libraries

In [3]:
```
import pandas as pd
import requests
import numpy as np
import geocoder
import folium
import requests
import matplotlib.cm as cm
import matplotlib.colors as colors
import json
import xml
import matplotlib.pyplot as plt
%matplotlib inline
import warnings
warnings.filterwarnings("ignore")

from pandas.io.json import json_normalize
from sklearn.cluster import KMeans
from geopy.geocoders import Nominatim
from bs4 import BeautifulSoup

pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

print("All Required Libraries Imported!")
```

```
All Required Libraries Imported!
```

Data Extraction and Cleaning

In [4]:
```
url = "https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M"
extracting_data = requests.get(url).text
wiki_data = BeautifulSoup(extracting_data, 'lxml')
```

In [6]:
```
# clean dataframe
toronto = toronto[toronto.Borough!='Not assigned']
toronto = toronto[toronto.Borough!= 0]
toronto.reset_index(drop = True, inplace = True)
i = 0
for i in range(0,toronto.shape[0]):
    if toronto.iloc[i][2] == 'Not assigned':
        toronto.iloc[i][2] = toronto.iloc[i][1]
        i = i+1
```

In [7]:
```
df = toronto.groupby(['Postalcode','Borough'])['Neighborhood'].apply(', '.join).reset_index()
df.head()
```

Out[7]:

	Postalcode	Borough	Neighborhood
0	M1A\n	Not assigned\n	
1	M1B\n	Scarborough\n	Malvern / Rouge
2	M1C\n	Scarborough\n	Rouge Hill / Port Union / Highland Creek
3	M1E\n	Scarborough\n	Guildwood / Morningside / West Hill
4	M1G\n	Scarborough\n	Woburn

Use the Geocoder to get the latitude & longitude of Scarborough

In [20]:
```
address = 'Scarborough,Toronto'

geolocator = Nominatim()
location = geolocator.geocode(address)
latitude_x = location.latitude
longitude_y = location.longitude
print('The Geograpical Co-ordinate of Scarborough,Toronto are {}, {}.'.format(latitude_x, longitude_y))
```

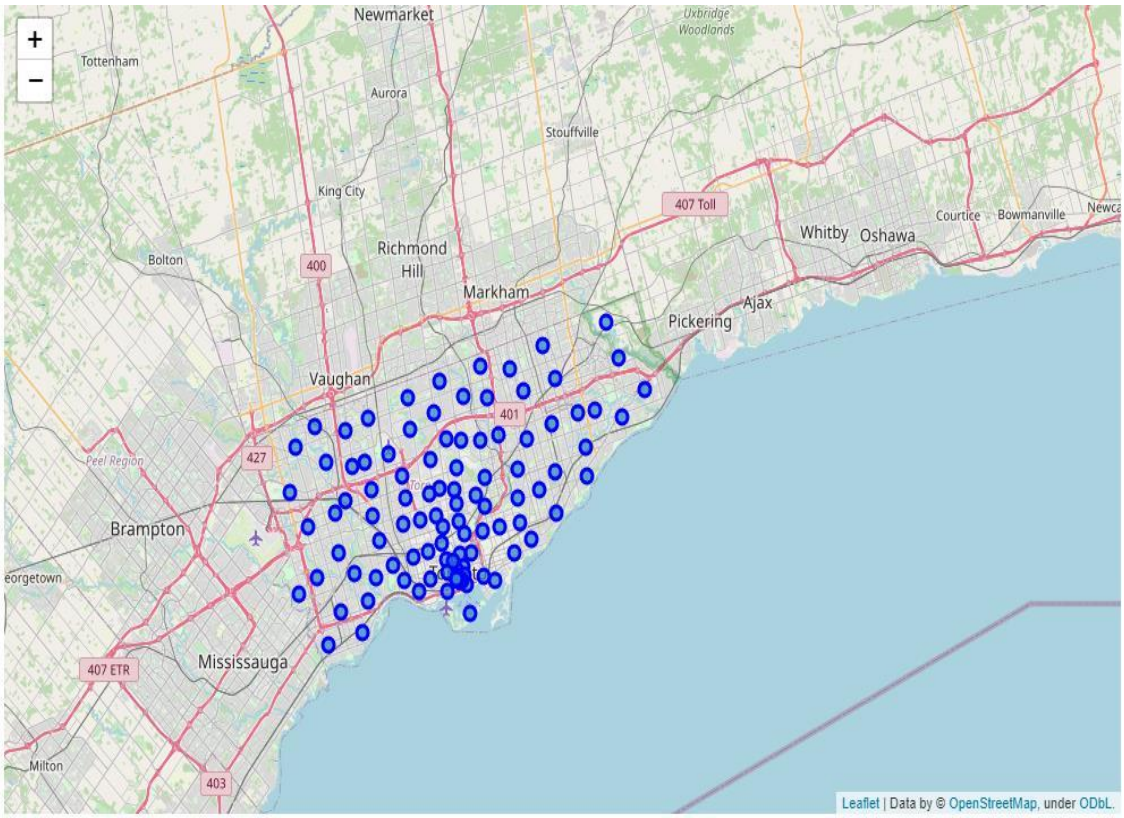
```
The Geograpical Co-ordinate of Scarborough,Toronto are 43.773077, -79.257774.
```

Create a Map of Scarborough using Folium

In [23]:
```
map_Scarborough = folium.Map(location=[latitude_x, longitude_y], zoom_start=10)

for lat, lng, nei in zip(df_2['Latitude'], df_2['Longitude'], df_2['Neighborhood']):

    label = '{}'.format(nei)
    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_Scarborough)

map_Scarborough
```

Map of Scarborough

Out[23]:



3. Methodology Section

Clustering Approach:

To compare the similarities within different neighborhoods, decided to explore neighbourhoods, segment them, and group them into clusters to find similar neighbourhoods in Scarborough. To be able to do that, we need to cluster data which is a form of unsupervised machine learning: k-means clustering algorithm.

Using K-Means Clustering Approach

In [41]:
```
neighborhoods_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)

Scarborough_merged =df_2.iloc[:16,:]

# merge toronto_grouped with toronto_data to add latitude/longitude for each neighborhood
Scarborough_merged = Scarborough_merged.join(neighborhoods_venues_sorted.set_index('Neighborhood'), on='Neighborhood')

Scarborough_merged.head()# check the last columns!
```

Out[41]:

	Postalcode	Borough	Neighborhood	Latitude	Longitude	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
0	M1A\n	Not assigned\n		43.648690	-79.385440	0	Coffee Shop	Café	Hotel	Japanese Restaurant	American Restaurant	Bookstore	Monument / Landmark	Tea Room	Theater	Deli / Bodega
1	M1B\n	Scarborough\n	Malvern / Rouge	43.808626	-79.189913	1	Park	Trail	Women's Store	Ethiopian Restaurant	Doner Restaurant	Donut Shop	Dumpling Restaurant	Eastern European Restaurant	Electronics Store	Elementary School
2	M1C\n	Scarborough\n	Rouge Hill / Port Union / Highland Creek	43.785779	-79.157368	2	Bar	Park	Doner Restaurant	Donut Shop	Dumpling Restaurant	Eastern European Restaurant	Electronics Store	Elementary School	Ethiopian Restaurant	Falafel Restaurant
3	M1E\n	Scarborough\n	Guildwood / Morningside / West Hill	43.765806	-79.185284	0	Pizza Place	Fast Food Restaurant	Park	Grocery Store	Coffee Shop	Bank	Pharmacy	Sports Bar	Discount Store	Fried Chicken Joint
4	M1G\n	Scarborough\n	Woburn	43.771545	-79.218135	0	Coffee Shop	Business Service	Park	Falafel Restaurant	Donut Shop	Dumpling Restaurant	Eastern European Restaurant	Electronics Store	Elementary School	Ethiopian Restaurant

Most Common venues near Neighbourhood

[39]:
```
import numpy as np
num_top_venues = 10

indicators = ['st', 'nd', 'rd']

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))

neighborhoods_venues_sorted = pd.DataFrame(columns=columns)
neighborhoods_venues_sorted['Neighborhood'] = Scarborough_grouped['Neighborhood']

for ind in np.arange(Scarborough_grouped.shape[0]):
    neighborhoods_venues_sorted.iloc[ind, 1:] = return_most_common_venues(Scarborough_grouped.iloc[ind, :], num_top_venues)

neighborhoods_venues_sorted.head()
```

Out[39]:

	Neighborhood	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
0		Coffee Shop	Café	Hotel	Japanese Restaurant	Bookstore	American Restaurant	Gym	Restaurant	Beer Bar	Seafood Restaurant
1	Agincourt	Shopping Mall	Chinese Restaurant	Bubble Tea Shop	Mediterranean Restaurant	Supermarket	Latin American Restaurant	Breakfast Spot	Lounge	Malay Restaurant	Skating Rink
2	Alderwood / Long Branch	Convenience Store	Coffee Shop	Pizza Place	Gas Station	Pharmacy	Pub	Dance Studio	Skating Rink	Sandwich Place	Athletics & Sports
3	Bathurst Manor / Wilson Heights / Downsview North	Bank	Coffee Shop	Ice Cream Shop	Restaurant	Fried Chicken Joint	Diner	Sandwich Place	Gas Station	Supermarket	Deli / Bodega
4	Bayview Village	Park	Construction & Landscaping	Trail	Farm	Donut Shop	Dumpling Restaurant	Eastern European Restaurant	Electronics Store	Ethiopian Restaurant	Falafel Restaurant

Nearby places of neighbourhoods are mined using credentials of Foursquare API features. Number of places/neighbourhood parameter is reasonably set to 100 and the radius parameter is set to 700.

In [26]:
```
radius = 700
LIMIT = 100
url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
    CLIENT_ID,
    CLIENT_SECRET,
    VERSION,
    latitude_n1,
   longitude_n1,
    radius,
   LIMIT)
results = requests.get(url).json()
```

Retrieving nearby venues & Categories

In [29]:
```
filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat', 'venue.location.lng']
nearby_venues =nearby_venues.loc[:, filtered_columns]
nearby_venues.head()
```

Out[29]:

	venue.name	venue.categories	venue.location.lat	venue.location.lng
0	Disney Store	[{'id': '4bf58dd8d48988d1f3941735', 'name': 'T...	43.775537	-79.256833
1	St. Andrews Fish & Chips	[{'id': '4edd64a0c7ddd24ca188df1a', 'name': 'F...	43.771865	-79.252645
2	SEPHORA	[{'id': '4bf58dd8d48988d10c951735', 'name': 'C...	43.775017	-79.258109
3	DAVIDsTEA	[{'id': '4bf58dd8d48988d1dc931735', 'name': 'T...	43.776320	-79.258688
4	American Eagle Outfitters	[{'id': '4bf58dd8d48988d103951735', 'name': 'C...	43.776012	-79.258334

Categories of Nearby Venues & Locations

In [30]:
```
nearby_venues['venue.categories'] = nearby_venues.apply(get_category_type, axis=1)

# clean columns
nearby_venues.columns = [col.split(".")[-1] for col in nearby_venues.columns]

nearby_venues.head(5)
```

Out[30]:

	name	categories	lat	lng
0	Disney Store	Toy / Game Store	43.775537	-79.256833
1	St. Andrews Fish & Chips	Fish & Chips Shop	43.771865	-79.252645
2	SEPHORA	Cosmetics Shop	43.775017	-79.258109
3	DAVIDsTEA	Tea Room	43.776320	-79.258688
4	American Eagle Outfitters	Clothing Store	43.776012	-79.258334

In [31]:
```
# Top 10 Categories
a=pd.Series(nearby_venues.categories)
a.value_counts()[:10]
```

Out[31]:
```
Clothing Store    7
Restaurant        4
Coffee Shop       4
Cosmetics Shop    3
Tea Room          2
Sandwich Place    2
Gas Station       2
Pharmacy          2
Grocery Store     1
Bar               1
Name: categories, dtype: int64
```

Calculate how many unique venue categories are there

In [34]:
```
print('There are {} Uniques Categories.'.format(len(Scarborough_venues['Venue Category'].unique())))
Scarborough_venues.groupby('Neighborhood').count().head()
```

```
There are 316 Uniques Categories.
```

Analyze each of the Neighborhoods

In [35]:
```
# one hot encoding
Scarborough_onehot = pd.get_dummies(Scarborough_venues[['Venue Category']], prefix="", prefix_sep="")

# add neighborhood column back to dataframe
Scarborough_onehot['Neighborhood'] = Scarborough_venues['Neighborhood']

# move neighborhood column to the first column
fixed_columns = [Scarborough_onehot.columns[-1]] + list(Scarborough_onehot.columns[:-1])
Scarborough_onehot = Scarborough_onehot[fixed_columns]
Scarborough_grouped = Scarborough_onehot.groupby('Neighborhood').mean().reset_index()
Scarborough_onehot.head(5)
```

Out[35]:

	Yoga Studio	Accessories Store	African Restaurant	Airport	American Restaurant	Antique Shop	Aquarium	Arcade	Art Gallery	Art Museum	Arts & Crafts Store	Asian Restaurant	Athletics & Sports	Auto Dealership	Auto Workshop	E J
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Print each neighborhood with its top 5 most common venues

In [36]:
```
num_top_venues = 5
for hood in Scarborough_grouped['Neighborhood']:
    print("---- "+hood+" ----")
    temp =Scarborough_grouped[Scarborough_grouped['Neighborhood'] == hood].T.reset_index()
    temp.columns = ['venue','freq']
    temp = temp.iloc[1:]
    temp['freq'] = temp['freq'].astype(float)
    temp = temp.round({'freq': 2})
    print(temp.sort_values('freq', ascending=False).reset_index(drop=True).head(num_top_venues))
    print('\n')
```

```
----  ----
                venue  freq
0          Coffee Shop  0.09
1                Café  0.07
2               Hotel  0.06
3  Japanese Restaurant  0.04
4  American Restaurant  0.03


---- Agincourt ----
               venue  freq
0       Shopping Mall  0.11
1  Chinese Restaurant  0.07
2       Grocery Store  0.04
3              Lounge  0.04
4    Malay Restaurant  0.04


---- Alderwood / Long Branch ----
              venue  freq
0  Convenience Store  0.11
1        Pizza Place  0.11
2     Sandwich Place  0.11
3        Gas Station  0.11
4        Coffee Shop  0.11


---- Bathurst Manor / Wilson Heights / Downsview North ----
            venue  freq
0     Coffee Shop  0.08
1            Bank  0.08
2  Sandwich Place  0.04
3     Supermarket  0.04
4     Gas Station  0.04
```

Retrieving most common venues near neighborhood

In [37]:
```
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]
```

In [39]:
```
import numpy as np
num_top_venues = 10

indicators = ['st', 'nd', 'rd']

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))

neighborhoods_venues_sorted = pd.DataFrame(columns=columns)
neighborhoods_venues_sorted['Neighborhood'] = Scarborough_grouped['Neighborhood']

for ind in np.arange(Scarborough_grouped.shape[0]):
    neighborhoods_venues_sorted.iloc[ind, 1:] = return_most_common_venues(Scarborough_grouped.iloc[ind, :], num_top_venues)

neighborhoods_venues_sorted.head()
```

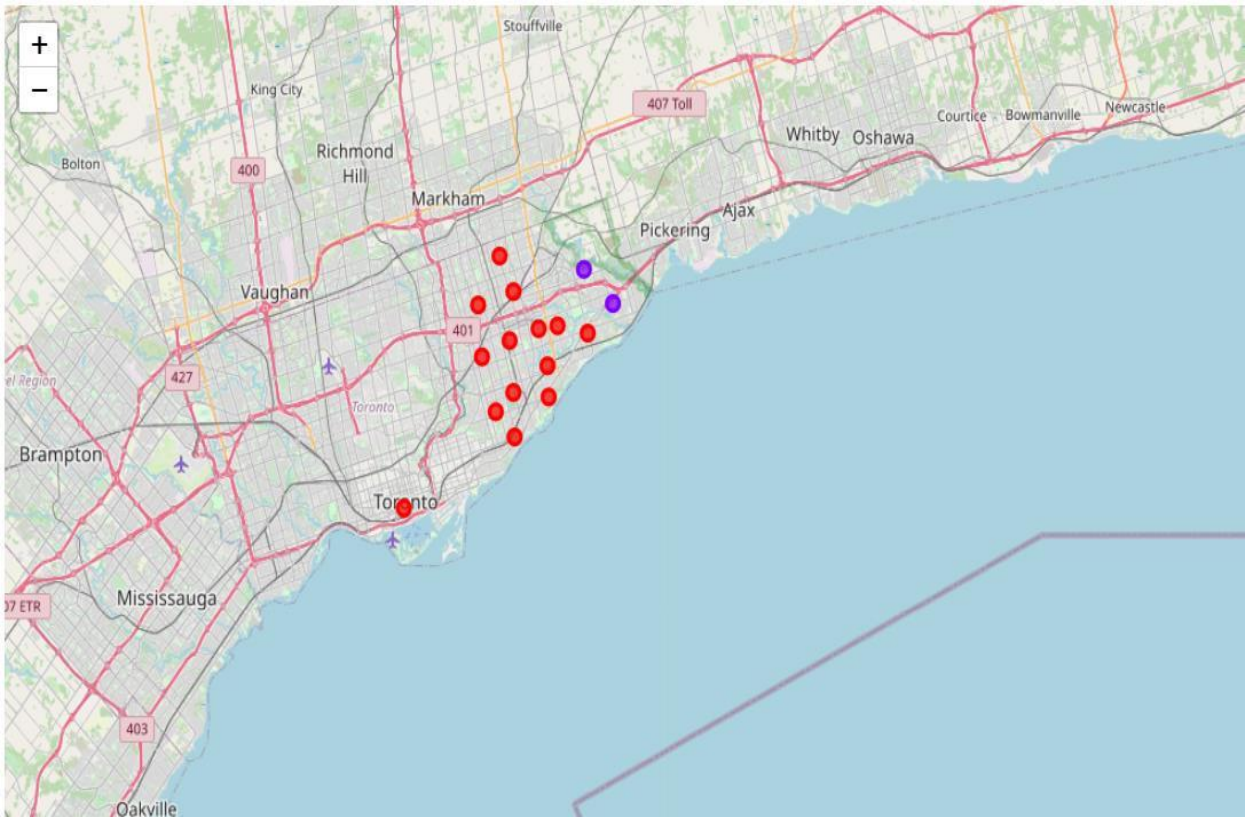
Out[39]:

	Neighborhood	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
0		Coffee Shop	Café	Hotel	Japanese Restaurant	American Restaurant	Bookstore	Monument / Landmark	Tea Room	Theater	Deli / Bodega
1	Agincourt	Shopping Mall	Chinese Restaurant	Hong Kong Restaurant	Mediterranean Restaurant	Sushi Restaurant	Supermarket	Latin American Restaurant	Breakfast Spot	Malay Restaurant	Skating Rink
2	Alderwood / Long Branch	Gas Station	Sandwich Place	Pizza Place	Convenience Store	Pub	Pharmacy	Gym	Skating Rink	Coffee Shop	Electronics Store
3	Bathurst Manor / Wilson Heights / Downsview North	Bank	Coffee Shop	Community Center	Deli / Bodega	Men's Store	Bridal Shop	Shopping Mall	Middle Eastern Restaurant	Sandwich Place	Supermarket
4	Bayview Village	Park	Construction & Landscaping	Trail	Women's Store	Falafel Restaurant	Donut Shop	Dumpling Restaurant	Eastern European Restaurant	Electronics Store	Elementary School

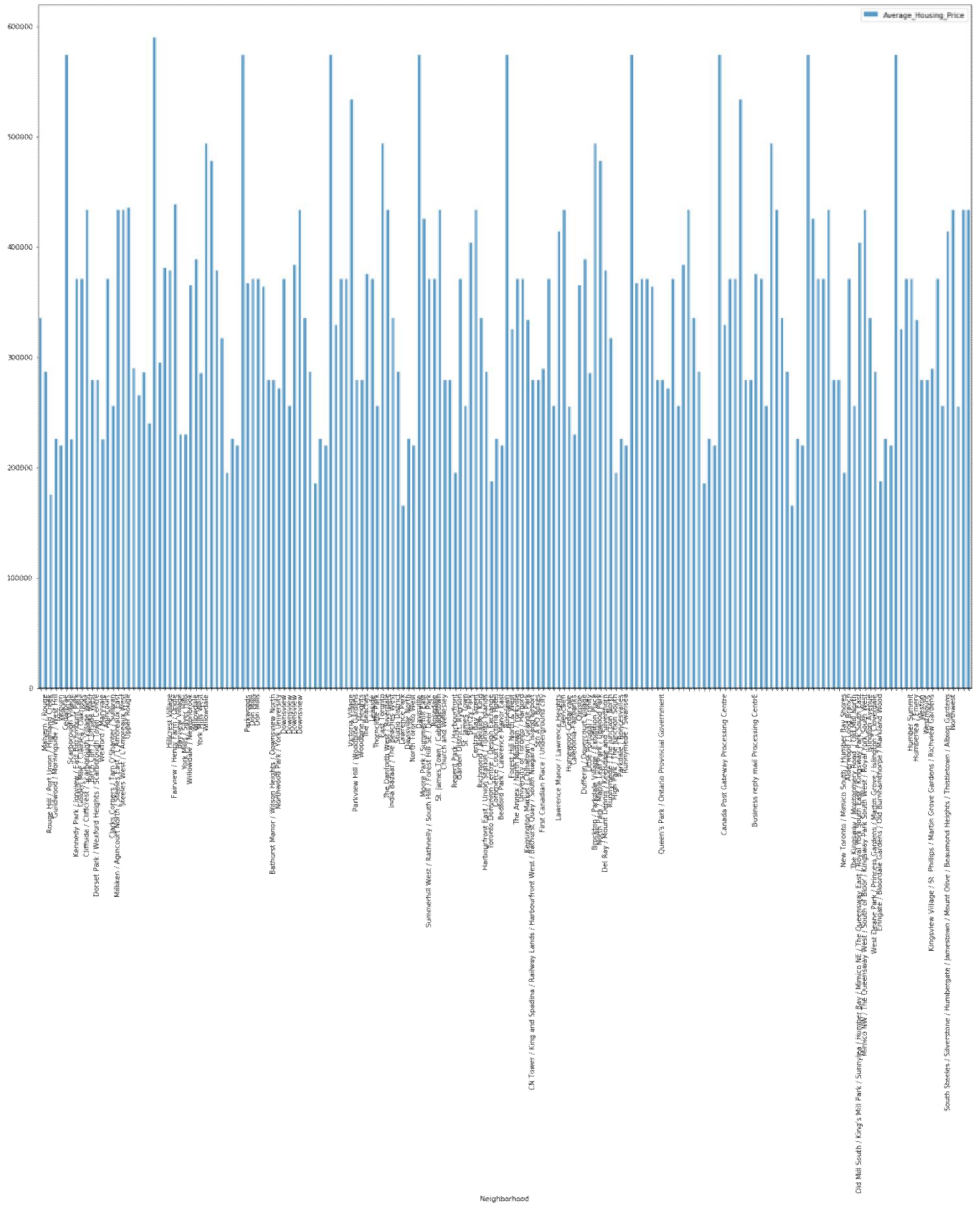
4. Results Section

Map of Clusters in Scarborough

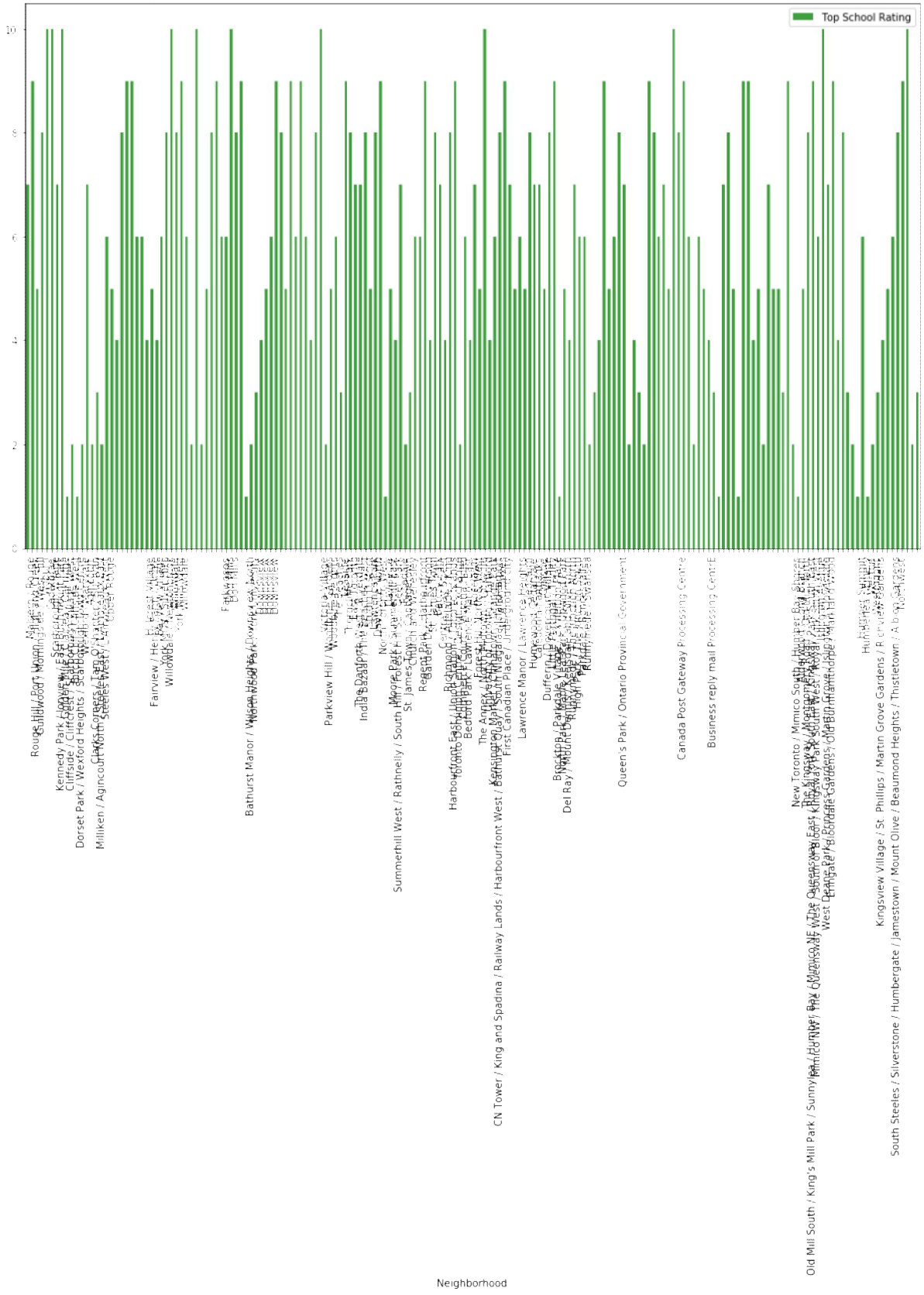
[43]:



Average Housing Price by Clusters in Scarborough



School Ratings by Clusters in Scarborough



5. Discussion Section

The main purpose of this project is to give details of housing rates & facilities such as schools, markets, grocery stores etc. and their proximity to the residence of the different neighborhoods in Scarborough. So that anyone wanting to settle down in Scarborough can make a viable and informed decision.

1. List of houses is sorted in terms of housing prices
2. List of schools sorted in terms of location, fees, rating and reviews

6. Conclusion

There are many ways this analysis could have been performed based on different methodology and perhaps different data sources. I chose the method I selected as it was a straight forward way to narrow down the options, not complicating what is actually simple in many ways – meeting the criteria for the surrounding venues in the neighborhood, and in my case, domain knowledge I have on the subject.

Libraries used:

Pandas: For creating and manipulating data frames.

Folium: Python visualization library would be used to visualize the neighbourhoods cluster distribution of using interactive leaflet map.

Scikit Learn: For importing k-means clustering.

JSON: Library to handle JSON files.

XML: To separate data from presentation and XML stores data in plain text format.

Geocoder: To retrieve Location Data.

Beautiful Soup and Requests: To scrap and library to handle http requests.

Matplotlib: Python Plotting Module.