

```
In [1]: ► import pandas as pd
import numpy as np # Library to handle data in a vectorized manner
import json # Library to handle JSON files
from geopy.geocoders import Nominatim # convert an address into latitude and
import requests # Library to handle requests
from pandas.io.json import json_normalize # transform JSON file into a pandas
# 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 yo
import folium # map rendering library
print('Libraries imported.')
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
from unicodedata import normalize
```

Libraries imported.

```
In [2]: ► tab = pd.read_html('https://en.wikipedia.org/wiki/List_of_postal_codes_of_Can
tabdf=tab[0]
coord=pd.read_csv('Geospatial_Coordinates.csv')
df=pd.merge(tabdf,coord,on='Postal Code',how='inner')
df.shape
```

Out[2]: (103, 5)

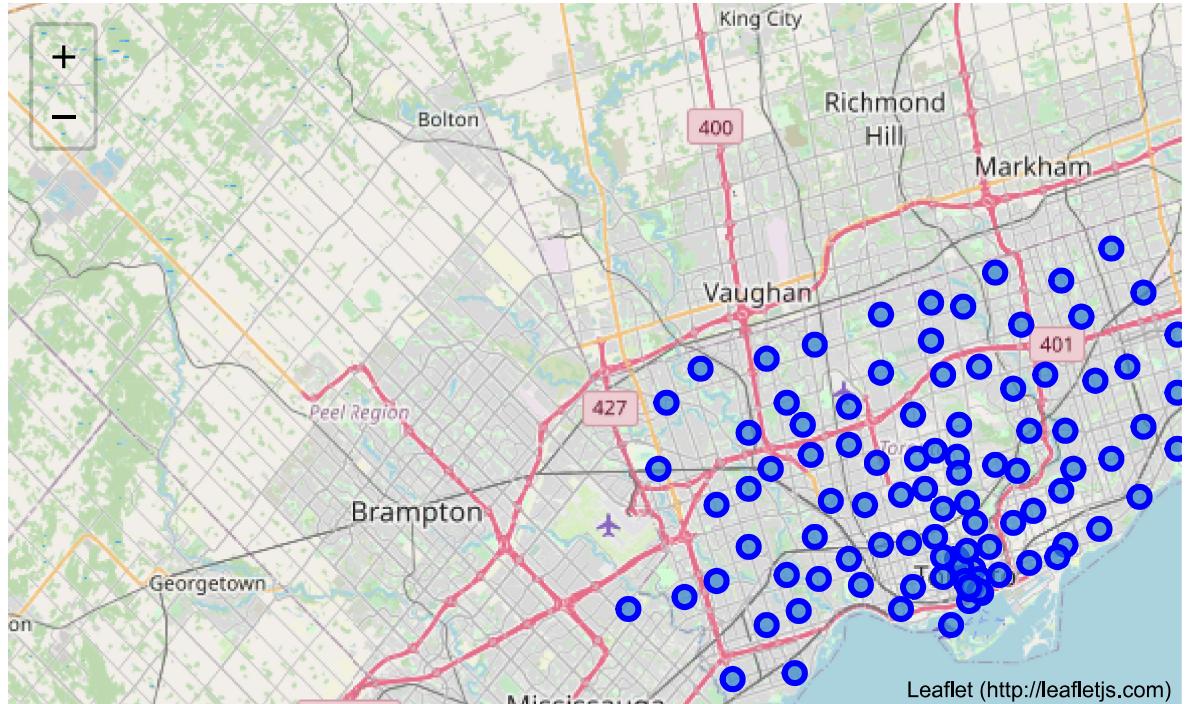
```
In [3]: ► address = 'Toronto'
geolocator = Nominatim(user_agent="to_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geographical coordinate of Toronto are {}, {}.'.format(latitude, lo
```

The geographical coordinate of Toronto are 43.6534817, -79.3839347.

```
In [4]: # create map of Toronto using latitude and longitude values
map_toronto = folium.Map(location=[latitude, longitude], zoom_start=10)
# add markers to map
for lat, lng, borough, neighborhood in zip(df['Latitude'], df['Longitude'], c
    label = '{}, {}'.format(neighborhood, borough)
    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_toronto)

map_toronto
```

Out[4]:



In [5]: ┶ toronto\_data = df[df['Borough'].str.contains('Toronto')].reset\_index(drop=True)  
toronto\_data

Out[5]:

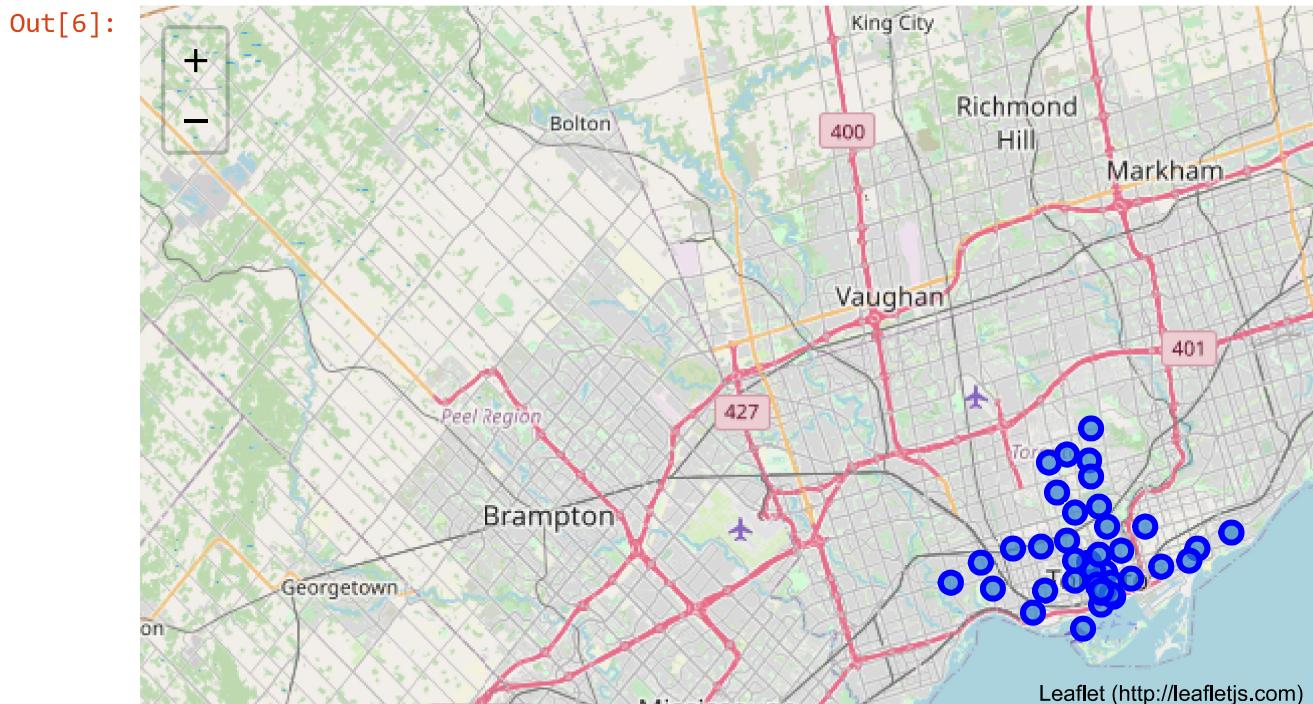
	Postal Code	Borough	Neighbourhood	Latitude	Longitude
0	M5A	Downtown Toronto	Regent Park, Harbourfront	43.654260	-79.360636
1	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.662301	-79.389494
2	M5B	Downtown Toronto	Garden District, Ryerson	43.657162	-79.378937
3	M5C	Downtown Toronto	St. James Town	43.651494	-79.375418
4	M4E	East Toronto	The Beaches	43.676357	-79.293031
5	M5E	Downtown Toronto	Berczy Park	43.644771	-79.373306
6	M5G	Downtown Toronto	Central Bay Street	43.657952	-79.387383
7	M6G	Downtown Toronto	Christie	43.669542	-79.422564
8	M5H	Downtown Toronto	Richmond, Adelaide, King	43.650571	-79.384568
9	M6H	West Toronto	Dufferin, Dovercourt Village	43.669005	-79.442259
10	M5J	Downtown Toronto	Harbourfront East, Union Station, Toronto Islands	43.640816	-79.381752
11	M6J	West Toronto	Little Portugal, Trinity	43.647927	-79.419750
12	M4K	East Toronto	The Danforth West, Riverdale	43.679557	-79.352188
13	M5K	Downtown Toronto	Toronto Dominion Centre, Design Exchange	43.647177	-79.381576
14	M6K	West Toronto	Brockton, Parkdale Village, Exhibition Place	43.636847	-79.428191
15	M4L	East Toronto	India Bazaar, The Beaches West	43.668999	-79.315572
16	M5L	Downtown Toronto	Commerce Court, Victoria Hotel	43.648198	-79.379817
17	M4M	East Toronto	Studio District	43.659526	-79.340923
18	M4N	Central Toronto	Lawrence Park	43.728020	-79.388790
19	M5N	Central Toronto	Roselawn	43.711695	-79.416936
20	M4P	Central Toronto	Davisville North	43.712751	-79.390197
21	M5P	Central Toronto	Forest Hill North & West, Forest Hill Road Park	43.696948	-79.411307
22	M6P	West Toronto	High Park, The Junction South	43.661608	-79.464763
23	M4R	Central Toronto	North Toronto West, Lawrence Park	43.715383	-79.405678
24	M5R	Central Toronto	The Annex, North Midtown, Yorkville	43.672710	-79.405678
25	M6R	West Toronto	Parkdale, Roncesvalles	43.648960	-79.456325
26	M4S	Central Toronto	Davisville	43.704324	-79.388790
27	M5S	Downtown Toronto	University of Toronto, Harbord	43.662696	-79.400049
28	M6S	West Toronto	Runnymede, Swansea	43.651571	-79.484450
29	M4T	Central Toronto	Moore Park, Summerhill East	43.689574	-79.383160
30	M5T	Downtown Toronto	Kensington Market, Chinatown, Grange Park	43.653206	-79.400049
31	M4V	Central Toronto	Summerhill West, Rathnelly, South Hill, Forest...	43.686412	-79.400049
32	M5V	Downtown Toronto	CN Tower, King and Spadina, Railway Lands, Har...	43.628947	-79.394420
33	M4W	Downtown Toronto	Rosedale	43.679563	-79.377529

Postal Code	Borough	Neighbourhood	Latitude	Longitude
34	M5W	Downtown Toronto	Stn A PO Boxes	43.646435 -79.374846
35	M4X	Downtown Toronto	St. James Town, Cabbagetown	43.667967 -79.367675
36	M5X	Downtown Toronto	First Canadian Place, Underground city	43.648429 -79.382280
37	M4Y	Downtown Toronto	Church and Wellesley	43.665860 -79.383160
38	M7Y	East Toronto	Business reply mail Processing Centre, South C...	43.662744 -79.321558

In [6]:

```
# create map of Toronto using Latitude and Longitude values
map_toronto = folium.Map(location=[latitude, longitude], zoom_start=10)
# add markers to map
for lat, lng, borough, neighborhood in zip(toonto_data['Latitude'], toonto_data['Longitude'],
                                             toonto_data['Borough'], toonto_data['Neighborhood']):
    label = '{}, {}'.format(neighborhood, borough)
    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_toronto)

map_toronto
```



In [7]:

```
CLIENT_ID = 'A30UPVRLHSPISMRAJ04G4VIRLN2AQOY3MRX4IVQTHCMG3AHK' # your Foursquare API client ID
CLIENT_SECRET = 'MAEPPWFABHM3URCERMEFDEP3ABMJ3KFRU214W1FQMMGESHKK' # your Foursquare API client secret
VERSION = '20201026' # Foursquare API version
LIMIT = 100 # A default Foursquare API limit value

print('Your credentials:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
```

Your credentials:  
 CLIENT\_ID: A30UPVRLHSPISMRAJ04G4VIRLN2AQOY3MRX4IVQTHCMG3AHK  
 CLIENT\_SECRET:MAEPPWFABHM3URCERMEFDEP3ABMJ3KFRU214W1FQMMGESHKK

```
In [8]: 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={}&cli-  
              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  
nearby_venues.columns = ['Neighborhood',  
                         'Neighborhood Latitude',  
                         'Neighborhood Longitude',  
                         'Venue',  
                         'Venue Latitude',  
                         'Venue Longitude',  
                         'Venue Category']  
  
    return(nearby_venues)
```

```
In [9]: ┆ toronto_venues = getNearbyVenues(names=toronto_data['Neighbourhood'],
                                         latitudes=toronto_data['Latitude'],
                                         longitudes=toronto_data['Longitude']
                                         )
```

Regent Park, Harbourfront  
Queen's Park, Ontario Provincial Government  
Garden District, Ryerson  
St. James Town  
The Beaches  
Berczy Park  
Central Bay Street  
Christie  
Richmond, Adelaide, King  
Dufferin, Dovercourt Village  
Harbourfront East, Union Station, Toronto Islands  
Little Portugal, Trinity  
The Danforth West, Riverdale  
Toronto Dominion Centre, Design Exchange  
Brockton, Parkdale Village, Exhibition Place  
India Bazaar, The Beaches West  
Commerce Court, Victoria Hotel  
Studio District  
Lawrence Park  
Roselawn  
Davisville North  
Forest Hill North & West, Forest Hill Road Park  
High Park, The Junction South  
North Toronto West, Lawrence Park  
The Annex, North Midtown, Yorkville  
Parkdale, Roncesvalles  
Davisville  
University of Toronto, Harbord  
Runnymede, Swansea  
Moore Park, Summerhill East  
Kensington Market, Chinatown, Grange Park  
Summerhill West, Rathnelly, South Hill, Forest Hill SE, Deer Park  
CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay, South Niagara, Island airport  
Rosedale  
Stn A PO Boxes  
St. James Town, Cabbagetown  
First Canadian Place, Underground city  
Church and Wellesley  
Business reply mail Processing Centre, South Central Letter Processing Plant Toronto

In [10]: ┌─ toronto\_venues.groupby('Neighborhood').count()

Out[10]:

Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
<b>Neighborhood</b>						
Berczy Park	55	55	55	55	55	55
Brockton, Parkdale Village, Exhibition Place	23	23	23	23	23	23
Business reply mail Processing Centre, South Central Letter Processing Plant Toronto	16	16	16	16	16	16
CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay, South Niagara, Island airport	16	16	16	16	16	16
Central Bay Street	68	68	68	68	68	68
Christie	16	16	16	16	16	16
Church and Wellesley	75	75	75	75	75	75
Commerce Court, Victoria Hotel	100	100	100	100	100	100
Davisville	33	33	33	33	33	33
Davisville North	9	9	9	9	9	9
Dufferin, Dovercourt Village	13	13	13	13	13	13
First Canadian Place, Underground city	100	100	100	100	100	100
Forest Hill North & West, Forest Hill Road Park	4	4	4	4	4	4
Garden District, Ryerson	100	100	100	100	100	100
Harbourfront East, Union Station, Toronto Islands	100	100	100	100	100	100
High Park, The Junction South	25	25	25	25	25	25
India Bazaar, The Beaches West	19	19	19	19	19	19
Kensington Market, Chinatown, Grange Park	74	74	74	74	74	74
Lawrence Park	3	3	3	3	3	3
Little Portugal, Trinity	45	45	45	45	45	45
Moore Park, Summerhill East	2	2	2	2	2	2
North Toronto West, Lawrence Park	18	18	18	18	18	18
Parkdale, Roncesvalles	14	14	14	14	14	14
Queen's Park, Ontario Provincial Government	33	33	33	33	33	33
Regent Park, Harbourfront	44	44	44	44	44	44
Richmond, Adelaide, King	100	100	100	100	100	100
Rosedale	4	4	4	4	4	4
Roselawn	2	2	2	2	2	2
Runnymede, Swansea	33	33	33	33	33	33
St. James Town	85	85	85	85	85	85
St. James Town, Cabbagetown	48	48	48	48	48	48

Neighborhood	Latitude	Neighborhood	Latitude	Venue	Venue	Venue	Venue
	Category		Longitude		Latitude	Longitude	
<b>Neighborhood</b>							
Stn A PO Boxes	96		96	96	96	96	96
Studio District	37		37	37	37	37	37
Summerhill West, Rathnelly, South Hill, Forest Hill SE, Deer Park	14		14	14	14	14	14
The Annex, North Midtown, Yorkville	19		19	19	19	19	19
The Beaches	4		4	4	4	4	4
The Danforth West, Riverdale	43		43	43	43	43	43
Toronto Dominion Centre, Design Exchange	100		100	100	100	100	100
University of Toronto, Harbord	34		34	34	34	34	34

In [11]: ► toronto\_venues.head()

Out[11]:

	Neighborhood	Neighborhood	Neighborhood	Venue	Venue	Venue	Venue
		Latitude	Longitude		Latitude	Longitude	
0	Regent Park, Harbourfront	43.65426	-79.360636	Roselle Desserts	43.653447	-79.362017	Bakery
1	Regent Park, Harbourfront	43.65426	-79.360636	Tandem Coffee	43.653559	-79.361809	Coffee Shop
2	Regent Park, Harbourfront	43.65426	-79.360636	Cooper Koo Family YMCA	43.653249	-79.358008	Distribution Center
3	Regent Park, Harbourfront	43.65426	-79.360636	Body Blitz Spa East	43.654735	-79.359874	Spa
4	Regent Park, Harbourfront	43.65426	-79.360636	Impact Kitchen	43.656369	-79.356980	Restaurant

In [12]: ► toronto\_venues.to\_csv('toronto\_venues.csv')

In [13]: ► print('There are {} uniques categories.'.format(len(toronto\_venues['Venue Category'])))  
There are 237 uniques categories.

In [14]: ► # one hot encoding  
toronto\_onehot = pd.get\_dummies(toronto\_venues[['Venue Category']], prefix="")  
# add neighborhood column back to dataframe  
toronto\_onehot['Neighborhood'] = toronto\_venues['Neighborhood']  
col\_name='Neighborhood'  
first\_col=toronto\_onehot.pop(col\_name)  
toronto\_onehot.insert(0, col\_name, first\_col)  
toronto\_onehot.head()  
toronto\_onehot.to\_csv('toronto\_onehot.csv')

In [15]: ► toronto\_grouped = toronto\_onehot.groupby('Neighborhood').mean().reset\_index()  
toronto\_grouped.to\_csv('toronto\_grouped.csv')

In [16]: ► 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 [17]: 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'] = toronto_grouped['Neighborhood']

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

neighborhoods_venues_sorted.head()
```

Out[17]:

	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
0	Berczy Park	Coffee Shop	Cocktail Bar	Seafood Restaurant	Bakery	Farmers Market	Beer Bar	Restaurant
1	Brockton, Parkdale Village, Exhibition Place	Café	Nightclub	Coffee Shop	Breakfast Spot	Bakery	Convenience Store	Performing Arts Venue
2	Business reply mail Processing Centre, South C...	Pizza Place	Garden Center	Farmers Market	Butcher	Brewery	Burrito Place	Recording Studio
3	CN Tower, King and Spadina, Railway Lands, Har...	Airport Lounge	Airport Service	Boutique	Plane	Airport	Airport Food Court	Airport Gate
4	Central Bay Street	Coffee Shop	Café	Italian Restaurant	Sandwich Place	Salad Place	Bubble Tea Shop	Department Store

In [18]: # set number of clusters

```
kclusters = 4
toronto_grouped_clustering = toronto_grouped.drop('Neighborhood', 1)

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

# check cluster Labels generated for each row in the dataframe
kmeans.labels_[0:100]
```

Out[18]: array([2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 1, 2, 0, 2, 2, 2, 2, 1, 3, 2, 2, 2, 2, 2, 2, 2, 0, 2, 2, 2])

In [19]: # add clustering labels

```
neighborhoods_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)
toronto_merged = toronto_data

# merge manhattan_grouped with manhattan_data to add Latitude/longitude for each neighborhood
toronto_merged = toronto_merged.join(neighborhoods_venues_sorted.set_index('Neighborhood'), on='Neighborhood')

toronto_merged.head() # check the last columns!
toronto_merged.to_csv('toronto_merged.csv')
```

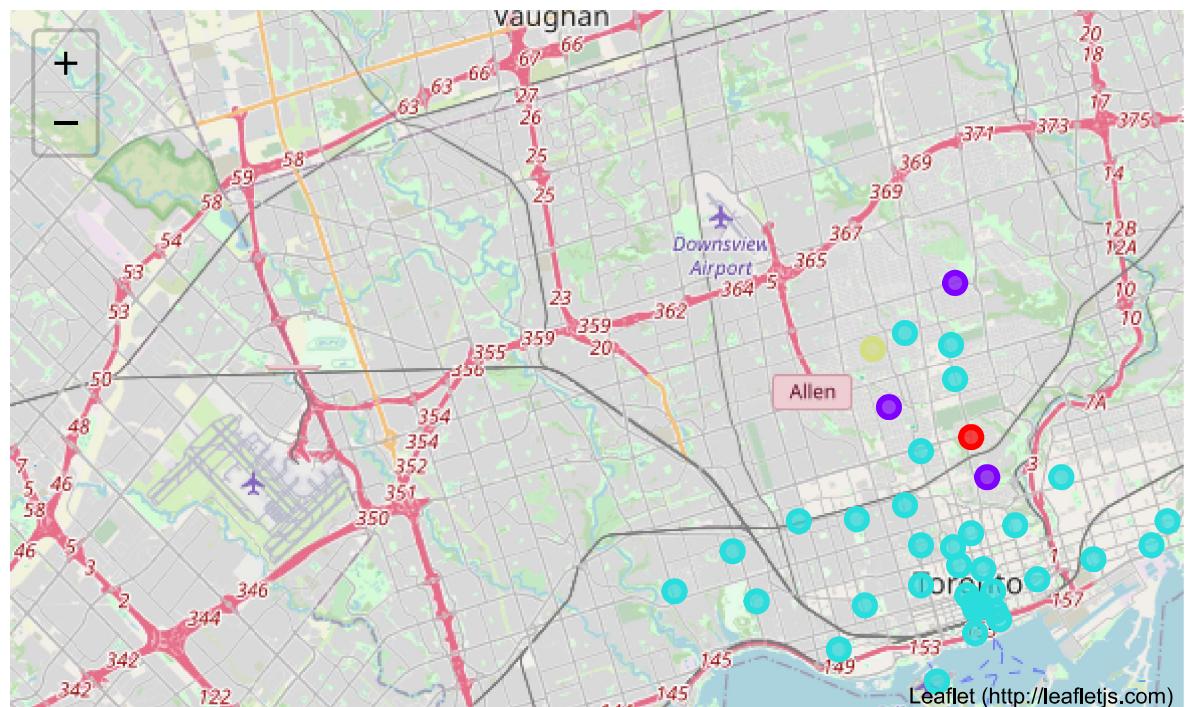
```
In [89]: # create map
map_clusters = folium.Map(location=[latitude, longitude], zoom_start=11)

# 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(toronto_merged['Latitude'], toronto_merged['Longitude'], toronto_merged['Neighborhood'], toronto_merged['Cluster']):
    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
```

Out[89]:



```
In [ ]: #4 clusters represented in red, green, blue, Light yellow colors
#which correspond to the labels assigned from k-means clustering.
```