Coursera - Capstone Project for IBM Data Science Certificate

Segmenting and Clustering Neighborhoods in Toronto

Objective: explore, segment, and cluster the neighborhoods in the city of Toronto

Data Sources: (Wiki page and shared csv files via Google Drive)

> https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M (https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M)
>
> http://cocl.us/Geospatial_data (http://cocl.us/Geospatial_data)
>
> https://github.com/tookitooki12/Coursera_Capstone (https://github.com/tookitooki12/Coursera_Capstone)

Table of contents:

System & Data Setup

In [ ]:

In [1]:
```
import pandas as pd
import numpy as np
import requests
from bs4 import BeautifulSoup

#mapping tools
!pip install geopy
from geopy.geocoders import Nominatim # convert an address into latitude and l
ongitude values

!pip install folium
import folium # map rendering library

def warn(*args, **kwargs):
    pass
import warnings
warnings.warn = warn
```

```
Requirement already satisfied: geopy in /home/jupyterlab/conda/lib/python3.6/
site-packages (1.18.1)
Requirement already satisfied: geographiclib<2,>=1.49 in /home/jupyterlab/con
da/lib/python3.6/site-packages (from geopy) (1.49)
Requirement already satisfied: folium in /home/jupyterlab/conda/lib/python3.
6/site-packages (0.5.0)
Requirement already satisfied: branca in /home/jupyterlab/conda/lib/python3.
6/site-packages (from folium) (0.3.1)
Requirement already satisfied: jinja2 in /home/jupyterlab/conda/lib/python3.
6/site-packages (from folium) (2.10)
Requirement already satisfied: requests in /home/jupyterlab/conda/lib/python
3.6/site-packages (from folium) (2.21.0)
Requirement already satisfied: six in /home/jupyterlab/conda/lib/python3.6/si
te-packages (from folium) (1.12.0)
Requirement already satisfied: MarkupSafe>=0.23 in /home/jupyterlab/conda/li
b/python3.6/site-packages (from jinja2->folium) (1.1.0)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in /home/jupyterlab/cond
a/lib/python3.6/site-packages (from requests->folium) (3.0.4)
Requirement already satisfied: certifi>=2017.4.17 in /home/jupyterlab/conda/l
ib/python3.6/site-packages (from requests->folium) (2018.11.29)
Requirement already satisfied: urllib3<1.25,>=1.21.1 in /home/jupyterlab/cond
a/lib/python3.6/site-packages (from requests->folium) (1.24.1)
Requirement already satisfied: idna<2.9,>=2.5 in /home/jupyterlab/conda/lib/p
ython3.6/site-packages (from requests->folium) (2.8)
```

create pydrive for uploading of csv file 'Geospatial_Coordinates'

In [ ]:

In [ ]:

In [2]:
```
# read csv file
Geospacial_Coordinates = pd.read_csv('http://cocl.us/Geospatial_data', sep =
','')
# examine the shape of original input data
print(Geospacial_Coordinates.shape)
```

```
(103, 3)
```

Create table - step 1 use BeautifulSoup to scrape data from website:

In [3]:
```
#create an object with raw data from website
website_url = requests.get("https://en.wikipedia.org/wiki/List_of_postal_codes
_of_Canada:_M").text
```

In [4]:
```
#create an object with the data from the website
soup = BeautifulSoup(website_url)
```

Create table - step 2 based on the table in the website (search for table > extract to dict & create pd DataFrame)

In [5]:
```
#search for table
My_table = soup.find('table',{'class':'wikitable sortable'})
My_table; #remove ';' to view output
```

Create table - step 3 extract table data and create pd DataFrame

In [6]:
```
#extract row data to dict
row_data = []
for row in My_table.find_all("tr"):
    cols = row.find_all("td")
    cols = [ele.text.strip() for ele in cols]
    row_data.append(cols)

row_data; #remove ';' to view output
```

In [ ]:
```
```

Part 1 - Create initial table with 103 Postcodes ('Postcode', 'Borough', 'Neighborhood')

In [7]:
```
#create initial pd DataFrame
df_table = pd.DataFrame(row_data)
df_table = df_table.rename(columns={0:"Postcode",1:"Borough",2:"Neighborhood"
})
df_table.head()
```

Out[7]:

	Postcode	Borough	Neighborhood
0	None	None	None
1	M1A	Not assigned	Not assigned
2	M2A	Not assigned	Not assigned
3	M3A	North York	Parkwoods
4	M4A	North York	Victoria Village

In [8]:
```
#drop the first row (index = 0), and any row where 'Bourough' = 'Not assigned'
df_table2 = df_table.copy()
df_table2 = df_table.drop([0])
df_table2 = df_table2.drop(df_table2[df_table2['Borough']=='Not assigned'].ind
ex)
df_table2 = df_table2.reset_index(drop=True)
df_table2.head()
```

Out[8]:

	Postcode	Borough	Neighborhood
0	M3A	North York	Parkwoods
1	M4A	North York	Victoria Village
2	M5A	Downtown Toronto	Harbourfront
3	M5A	Downtown Toronto	Regent Park
4	M6A	North York	Lawrence Heights

Create table - step 4 - data transform - if 'Neighborhood' = 'Not Assigned', then use 'Borough'

In [9]:
```
#check a row where 'Neighborhood' = 'Not assigned'
df_table2.loc[6]
```

Out[9]:
```
Postcode                M7A
Borough         Queen's Park
Neighborhood    Not assigned
Name: 6, dtype: object
```

In [10]:
```
#create a new table and replace values if 'Neighborhood' = 'Not assigned' with
'Bourough'
df_table3 = df_table2.copy()

df_table3['Neighborhood'] = df_table3.apply(
    lambda row: row['Borough'] if row['Neighborhood'] == 'Not assigned' else r
ow['Neighborhood'],
    axis=1
)

#have a look at the transformed data
df_table3.loc[6]
```

Out[10]:
```
Postcode                M7A
Borough         Queen's Park
Neighborhood    Queen's Park
Name: 6, dtype: object
```

Create table - step 5 group the dataframe by Postcode & Borough and 'Join' values in 'Neighborhood'

In [11]:
```
df_table4 = df_table3.copy()

df_table4 = (df_table4.groupby(['Postcode','Borough'])['Neighborhood']
       .apply(lambda x: ','.join(set(x.dropna())))
       .reset_index())

df_table4 = pd.DataFrame(df_table4)
df_table4.head()
```

Out[11]:

	Postcode	Borough	Neighborhood
0	M1B	Scarborough	Malvern,Rouge
1	M1C	Scarborough	Highland Creek,Rouge Hill,Port Union
2	M1E	Scarborough	Morningside,Guildwood,West Hill
3	M1G	Scarborough	Woburn
4	M1H	Scarborough	Cedarbrae

In [12]:
```
df_table4.shape
```

Out[12]:
```
(103, 3)
```

df_table4 above is shape (103,3) - submission for part 1 of peer review

Part 2 - Concatinate initial table with Geospacial Coordinates

In [13]:
```
Geo = pd.DataFrame(Geospacial_Coordinates)
Geo.head()
```

Out[13]:

	Postal Code	Latitude	Longitude
0	M1B	43.806686	-79.194353
1	M1C	43.784535	-79.160497
2	M1E	43.763573	-79.188711
3	M1G	43.770992	-79.216917
4	M1H	43.773136	-79.239476

In [14]:
```
df_table_final = pd.concat([df_table4, Geo], axis=1)
df_table_final = df_table_final.drop(['Postal Code'], axis = 1)
df_table_final.head()
```

Out[14]:

	Postcode	Borough	Neighborhood	Latitude	Longitude
0	M1B	Scarborough	Malvern,Rouge	43.806686	-79.194353
1	M1C	Scarborough	Highland Creek,Rouge Hill,Port Union	43.784535	-79.160497
2	M1E	Scarborough	Morningside,Guildwood,West Hill	43.763573	-79.188711
3	M1G	Scarborough	Woburn	43.770992	-79.216917
4	M1H	Scarborough	Cedarbrae	43.773136	-79.239476

Part 3 - setup mapping

In [15]:
```
import json
import requests
import matplotlib.cm as cm
import matplotlib.colors as colors
from sklearn.cluster import KMeans
from pandas.io.json import json_normalize # tranform JSON file into a pandas d
ataframe
```

In [16]:
```
address = 'Toronto, Ontario'

geolocator = Nominatim()
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geograpical coordinate of Toronto, Canada are {}, {}.'.format(latit
ude, longitude))
```

```
The geograpical coordinate of Toronto, Canada are 43.653963, -79.387207.
```

In [17]:
```
df_toronto = df_table_final[df_table_final['Borough'].str.contains('Toronto')]
.reset_index(drop=True)
df_toronto.head()
```

Out[17]:

	Postcode	Borough	Neighborhood	Latitude	Longitude
0	M4E	East Toronto	The Beaches	43.676357	-79.293031
1	M4K	East Toronto	Riverdale,The Danforth West	43.679557	-79.352188
2	M4L	East Toronto	The Beaches West,India Bazaar	43.668999	-79.315572
3	M4M	East Toronto	Studio District	43.659526	-79.340923
4	M4N	Central Toronto	Lawrence Park	43.728020	-79.388790

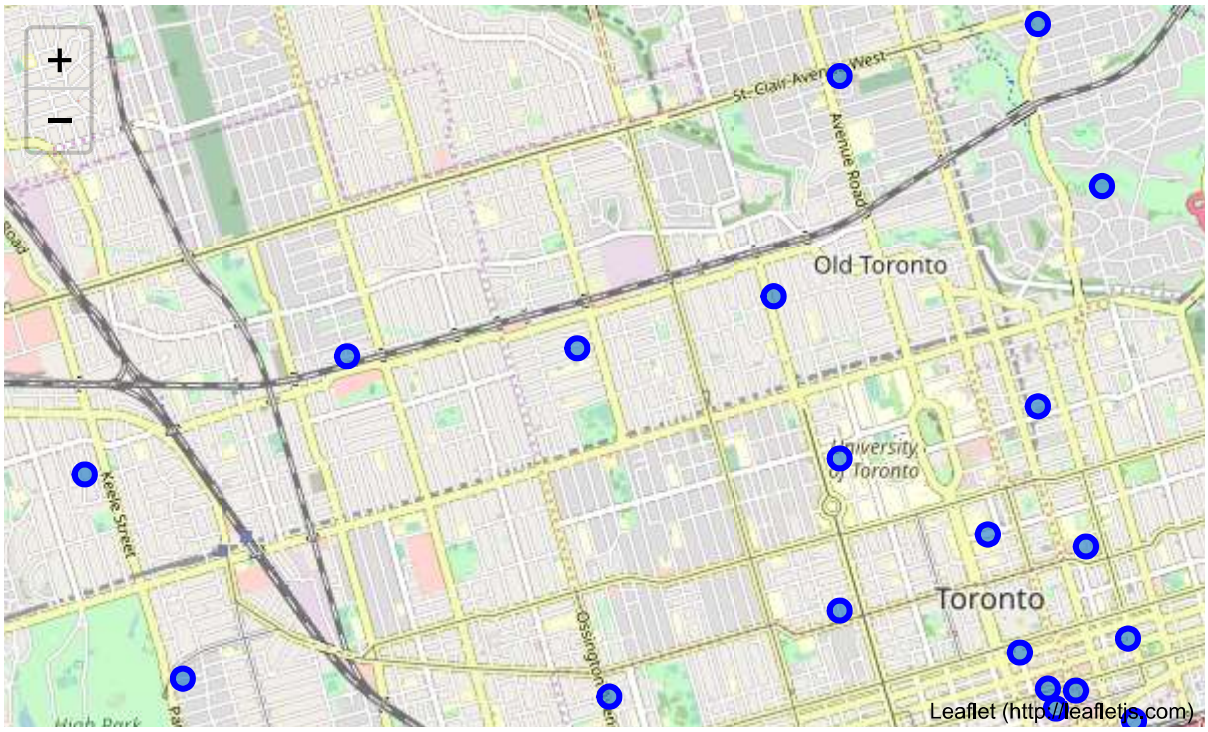
Part 3a - Create a map using Borough names that include 'Toronto' & highlight each Neighborhood

In [18]:
```
# create map of Toronto using latitude and longitude values
map_toronto = folium.Map(location=[latitude, longitude], zoom_start=13)

# add markers to map
for lat, lng, borough, neighborhood in zip(df_toronto['Latitude'], df_toronto[
'Longitude'], df_toronto['Borough'], df_toronto['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).add_to(map_toronto)

map_toronto
```

Out[18]:



Part 3b - explore a neighborhood using Foursquare API

In [19]:
```
CLIENT_ID = 'AR2WMOIEQ2H1CLUCD5FPYJLWOQOOEQ2SNYYUIJQS5NKFVDV1' # my Foursquare
ID
CLIENT_SECRET = 'P3PCXVFK5CLFBLE25KUXVHQ0YNUNTV4KGGBSSXQ4AEDQLQ3E' # your Four
square Secret
VERSION = '20180605' # Foursquare API version
radius = 500
LIMIT = 100

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

```
Your credentails:
CLIENT_ID: AR2WMOIEQ2H1CLUCD5FPYJLWOQOOEQ2SNYYUIJQS5NKFVDV1
CLIENT_SECRET:P3PCXVFK5CLFBLE25KUXVHQ0YNUNTV4KGGBSSXQ4AEDQLQ3E
```

Let's explore the 'studio district'.. that sounds like a cool spot

In [20]:
```
#define objects for 'Studio District' index [3] in df_toronto
neighborhood_latitude = df_toronto.loc[3, 'Latitude'] # neighborhood latitude
 value
neighborhood_longitude = df_toronto.loc[3, 'Longitude'] # neighborhood longitu
de value
neighborhood_name = df_toronto.loc[3, 'Neighborhood'] # neighborhood name

print('Latitude and longitude values of {} are {}, {}.'.format(neighborhood_na
me,
                                                               neighborhood_la
titude,
                                                               neighborhood_lo
ngitude))
```

```
Latitude and longitude values of Studio District are 43.6595255, -79.340923.
```

Now, let's get the top 100 venues that are in Studio District within a radius of 500 meters.

In [21]:
```
#step 1 - create the correct GET request URL
url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secre
t={}&v={}&ll={},{}&radius={}&limit={}'.format(
    CLIENT_ID,
    CLIENT_SECRET,
    VERSION,
    neighborhood_latitude,
    neighborhood_longitude,
    radius,
    LIMIT)
url # display GET request URL
```

Out[21]:
```
'https://api.foursquare.com/v2/venues/explore?&client_id=AR2WMOIEQ2H1CLUCD5FP
YJLWOQOOEQ2SNYYUIJQS5NKFVDV1&client_secret=P3PCXVFK5CLFBLE25KUXVHQ0YNUNTV4KGG
BSSXQ4AEDQLQ3E&v=20180605&ll=43.6595255,-79.340923&radius=500&limit=100'
```

In [22]:
```
results = requests.get(url).json()
results; # remove ';' to see json data
```

clean the json and structure it into a *pandas* dataframe.

In [23]:
```
# function that extracts the category of the venue
def get_category_type(row):
    try:
        categories_list = row['categories']
    except:
        categories_list = row['venue.categories']

    if len(categories_list) == 0:
        return None
    else:
        return categories_list[0]['name']
```

In [24]:
```
venues = results['response']['groups'][0]['items']

nearby_venues = json_normalize(venues) # flatten JSON

# filter columns
filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat', 'v
enue.location.lng']
nearby_venues =nearby_venues.loc[:, filtered_columns]

# filter the category for each row
nearby_venues['venue.categories'] = nearby_venues.apply(get_category_type, axi
s=1)

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

nearby_venues.head()
```

Out[24]:

	name	categories	lat	lng
0	Ed's Real Scoop	Ice Cream Shop	43.660656	-79.342019
1	Leslieville Pumps	Sandwich Place	43.660892	-79.340626
2	Te Aro	Coffee Shop	43.661373	-79.338577
3	Queen Books	Bookstore	43.660651	-79.342267
4	Hooked	Fish Market	43.660407	-79.343257

In [25]:
```
print('{} venues were returned by Foursquare.'.format(nearby_venues.shape[0]))
```

```
39 venues were returned by Foursquare.
```

**create a map of the studio district and highlight nearby venues

In [26]:
```
map_studio = folium.Map(location=[neighborhood_latitude, neighborhood_longitude], zoom_start=17)

# add markers to map
for lat, lng, name, categories in zip(nearby_venues['lat'], nearby_venues['lng'], nearby_venues['name'], nearby_venues['categories']):
  label = '{},{}'.format(categories,name)
  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).add_to(map_studio)

map_studio
```

Out[26]:

Part 3c - Cluster Analysis of Venues across all neighborhoods

In [27]:
```
# create a function to get all venues for each neighborhood
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={}&clie
nt_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)
```

In [28]:
```
#run function for all toronto neighborhoods and create df 'toronto_venues'
toronto_venues = getNearbyVenues(names=df_toronto['Neighborhood'],
                                 latitudes=df_toronto['Latitude'],
                                 longitudes=df_toronto['Longitude']
                                )
```

```
The Beaches
Riverdale,The Danforth West
The Beaches West,India Bazaar
Studio District
Lawrence Park
Davisville North
North Toronto West
Davisville
Summerhill East,Moore Park
Summerhill West,Forest Hill SE,Deer Park,South Hill,Rathnelly
Rosedale
St. James Town,Cabbagetown
Church and Wellesley
Harbourfront,Regent Park
Ryerson,Garden District
St. James Town
Berczy Park
Central Bay Street
Adelaide,Richmond,King
Harbourfront East,Toronto Islands,Union Station
Design Exchange,Toronto Dominion Centre
Commerce Court,Victoria Hotel
Roselawn
Forest Hill West,Forest Hill North
Yorkville,The Annex,North Midtown
University of Toronto,Harbord
Kensington Market,Grange Park,Chinatown
South Niagara,Bathurst Quay,King and Spadina,Railway Lands,CN Tower,Harbourfr
ont West,Island airport
Stn A PO Boxes 25 The Esplanade
Underground city,First Canadian Place
Christie
Dovercourt Village,Dufferin
Little Portugal,Trinity
Exhibition Place,Parkdale Village,Brockton
High Park,The Junction South
Parkdale,Roncesvalles
Swansea,Runnymede
Business Reply Mail Processing Centre 969 Eastern
```

In [29]:
```
print(toronto_venues.shape)
toronto_venues.head()
```

```
(1695, 7)
```

Out[29]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	The Beaches	43.676357	-79.293031	The Big Carrot Natural Food Market	43.678879	-79.297734	Health Food Store
1	The Beaches	43.676357	-79.293031	Grover Pub and Grub	43.679181	-79.297215	Pub
2	The Beaches	43.676357	-79.293031	Starbucks	43.678798	-79.298045	Coffee Shop
3	The Beaches	43.676357	-79.293031	Upper Beaches	43.680563	-79.292869	Neighborhood
4	Riverdale,The Danforth West	43.679557	-79.352188	Pantheon	43.677621	-79.351434	Greek Restaurant

In [30]:
```
toronto_venues.groupby('Neighborhood').count().head()
```

Out[30]:

Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Adelaide,Richmond,King	100	100	100	100	100	100
Berczy Park	56	56	56	56	56	56
Business Reply Mail Processing Centre 969 Eastern	17	17	17	17	17	17
Central Bay Street	83	83	83	83	83	83
Christie	15	15	15	15	15	15

In [31]:
```
print('There are {} unique categories.'.format(len(toronto_venues['Venue Category'].unique())))
```

```
There are 236 unique categories.
```

create 'one hot' file with dummy values by venue category

In [32]:
```
# one hot encoding
toronto_onehot = pd.get_dummies(toronto_venues[['Venue Category']], prefix="",
prefix_sep="")

# add neighborhood column back to dataframe
toronto_onehot['Neighborhood'] = toronto_venues['Neighborhood']

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

toronto_onehot.head()
```

Out[32]:

	Yoga Studio	Adult Boutique	Afghan Restaurant	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airport Terminal	American Restaurant
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0

5 rows × 236 columns

Next, let's group rows by neighborhood and by taking the mean of the frequency of occurrence of each category

In [33]:
```
toronto_grouped = toronto_onehot.groupby('Neighborhood').mean().reset_index()
toronto_grouped
```

Out[33]:

	Neighborhood	Yoga Studio	Adult Boutique	Afghan Restaurant	Airport	Airport Food Court	Airport Gate	Airp Loun
0	Adelaide,Richmond,King	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
1	Berczy Park	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
2	Business Reply Mail Processing Centre 969 Eastern	0.058824	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
3	Central Bay Street	0.012048	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
4	Christie	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
5	Church and Wellesley	0.011364	0.011364	0.011364	0.000000	0.000000	0.000000	0.0000
6	Commerce Court,Victoria Hotel	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
7	Davisville	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
8	Davisville North	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
9	Design Exchange,Toronto Dominion Centre	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
10	Dovercourt Village,Dufferin	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
11	Exhibition Place,Parkdale Village,Brockton	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
12	Forest Hill West,Forest Hill North	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
13	Harbourfront East,Toronto Islands,Union Station	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
14	Harbourfront,Regent Park	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
15	High Park,The Junction South	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
16	Kensington Market,Grange Park,Chinatown	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
17	Lawrence Park	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
18	Little Portugal,Trinity	0.015873	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
19	North Toronto West	0.052632	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
20	Parkdale,Roncesvalles	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
21	Riverdale,The Danforth West	0.023810	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
22	Rosedale	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
23	Roselawn	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
24	Ryerson,Garden District	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000

	Neighborhood	Yoga Studio	Adult Boutique	Afghan Restaurant	Airport	Airport Food Court	Airport Gate	Airpo Loun
25	South Niagara,Bathurst Quay,King and Spadina,R...	0.000000	0.000000	0.000000	0.071429	0.071429	0.071429	0.1428
26	St. James Town	0.010000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
27	St. James Town,Cabbagetown	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
28	Stn A PO Boxes 25 The Esplanade	0.010526	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
29	Studio District	0.025641	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
30	Summerhill East,Moore Park	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
31	Summerhill West,Forest Hill SE,Deer Park,South...	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
32	Swansea,Runnymede	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
33	The Beaches	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
34	The Beaches West,India Bazaar	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
35	Underground city,First Canadian Place	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
36	University of Toronto,Harbord	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000
37	Yorkville,The Annex,North Midtown	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0000

38 rows × 236 columns

In [34]:
```
toronto_grouped.shape
```

Out[34]: (38, 236)

Let's print each neighborhood along with the top 5 most common venues

In [35]:
```
num_top_venues = 5

for hood in toronto_grouped['Neighborhood']:
    print("----"+hood+"----")
    temp = toronto_grouped[toronto_grouped['Neighborhood'] == hood].T.reset_in
dex()
    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).hea
d(num_top_venues))
    print('\n')
```

```
----Adelaide,Richmond,King----
                venue  freq
0         Coffee Shop  0.06
1               Café  0.05
2  American Restaurant  0.04
3      Thai Restaurant  0.04
4           Steakhouse  0.04


----Berczy Park----
          venue  freq
0   Coffee Shop  0.07
1  Cocktail Bar  0.05
2    Restaurant  0.05
3        Bakery  0.04
4   Cheese Shop  0.04


----Business Reply Mail Processing Centre 969 Eastern----
              venue  freq
0       Yoga Studio  0.06
1     Auto Workshop  0.06
2              Park  0.06
3        Comic Shop  0.06
4  Recording Studio  0.06


----Central Bay Street----
                venue  freq
0         Coffee Shop  0.16
1               Café  0.06
2  Italian Restaurant  0.05
3        Burger Joint  0.04
4                 Bar  0.04


----Christie----
               venue  freq
0              Café  0.20
1      Grocery Store  0.20
2               Park  0.13
3  Convenience Store  0.07
4         Baby Store  0.07


----Church and Wellesley----
                venue  freq
0  Japanese Restaurant  0.07
1     Sushi Restaurant  0.06
2          Coffee Shop  0.06
3              Gay Bar  0.05
4         Burger Joint  0.03


----Commerce Court,Victoria Hotel----
                venue  freq
0         Coffee Shop  0.10
```

```
1                Café  0.07
2          Restaurant  0.06
3               Hotel  0.06
4  American Restaurant  0.04


----Davisville----
            venue  freq
0     Pizza Place  0.11
1  Sandwich Place  0.08
2    Dessert Shop  0.08
3        Pharmacy  0.06
4     Coffee Shop  0.06


----Davisville North----
               venue  freq
0  Food & Drink Shop  0.12
1               Park  0.12
2       Burger Joint  0.12
3      Grocery Store  0.12
4               Gym  0.12


----Design Exchange,Toronto Dominion Centre----
                 venue  freq
0          Coffee Shop  0.15
1                Café  0.09
2               Hotel  0.08
3          Restaurant  0.04
4  American Restaurant  0.04


----Dovercourt Village,Dufferin----
            venue  freq
0          Bakery  0.10
1        Pharmacy  0.10
2     Supermarket  0.10
3  Discount Store  0.10
4            Bank  0.05


----Exhibition Place,Parkdale Village,Brockton----
                   venue  freq
0         Breakfast Spot  0.11
1            Coffee Shop  0.11
2                  Café  0.11
3  Performing Arts Venue  0.05
4           Climbing Gym  0.05


----Forest Hill West,Forest Hill North----
              venue  freq
0     Jewelry Store  0.25
1             Trail  0.25
2              Park  0.25
3  Sushi Restaurant  0.25
```

```
4       Music Venue  0.00
```

```
----Harbourfront East,Toronto Islands,Union Station----
         venue  freq
0  Coffee Shop  0.14
1        Hotel  0.05
2     Aquarium  0.05
3  Pizza Place  0.04
4         Café  0.04
```

```
----Harbourfront,Regent Park----
         venue  freq
0  Coffee Shop  0.17
1         Park  0.06
2         Café  0.06
3       Bakery  0.06
4          Pub  0.04
```

```
----High Park,The Junction South----
                    venue  freq
0      Mexican Restaurant  0.09
1                   Café  0.09
2  Furniture / Home Store  0.04
3             Flea Market  0.04
4                     Bar  0.04
```

```
----Kensington Market,Grange Park,Chinatown----
                           venue  freq
0                           Café  0.07
1                            Bar  0.06
2          Vietnamese Restaurant  0.05
3  Vegetarian / Vegan Restaurant  0.05
4            Dumpling Restaurant  0.04
```

```
----Lawrence Park----
                venue  freq
0                Park  0.25
1         Swim School  0.25
2  Dim Sum Restaurant  0.25
3            Bus Line  0.25
4         Yoga Studio  0.00
```

```
----Little Portugal,Trinity----
              venue  freq
0               Bar  0.13
1       Men's Store  0.06
2  Asian Restaurant  0.05
3       Coffee Shop  0.05
4            Bakery  0.03
```

```
----North Toronto West----
                venue  freq
0          Coffee Shop  0.11
1  Sporting Goods Shop  0.11
2          Yoga Studio  0.05
3            Gift Shop  0.05
4         Dessert Shop  0.05


----Parkdale,Roncesvalles----
            venue  freq
0       Gift Shop  0.12
1  Breakfast Spot  0.12
2       Bookstore  0.06
3    Dessert Shop  0.06
4         Dog Run  0.06


----Riverdale,The Danforth West----
                venue  freq
0    Greek Restaurant  0.24
1         Coffee Shop  0.10
2      Ice Cream Shop  0.07
3  Italian Restaurant  0.05
4           Bookstore  0.05


----Rosedale----
           venue  freq
0           Park  0.50
1     Playground  0.25
2          Trail  0.25
3      Nightclub  0.00
4  Metro Station  0.00


----Roselawn----
                      venue  freq
0              Home Service   0.5
1                    Garden   0.5
2  Mediterranean Restaurant   0.0
3             Metro Station   0.0
4        Mexican Restaurant   0.0


----Ryerson,Garden District----
                        venue  freq
0             Clothing Store  0.09
1                Coffee Shop  0.09
2                       Café  0.04
3             Cosmetics Shop  0.03
4  Middle Eastern Restaurant  0.03


----South Niagara,Bathurst Quay,King and Spadina,Railway Lands,CN Tower,Harbo
urfront West,Island airport----
             venue  freq
```

```
0    Airport Lounge  0.14
1   Airport Service  0.14
2  Airport Terminal  0.14
3          Boutique  0.07
4  Sculpture Garden  0.07


----St. James Town----
            venue  freq
0     Coffee Shop  0.07
1      Restaurant  0.06
2            Café  0.05
3           Hotel  0.05
4  Clothing Store  0.04


----St. James Town,Cabbagetown----
         venue  freq
0  Coffee Shop  0.11
1   Restaurant  0.07
2          Pub  0.05
3       Market  0.05
4         Café  0.05


----Stn A PO Boxes 25 The Esplanade----
         venue  freq
0  Coffee Shop  0.11
1   Restaurant  0.05
2         Café  0.04
3     Beer Bar  0.03
4          Pub  0.03


----Studio District----
                venue  freq
0                Café  0.10
1         Coffee Shop  0.08
2   Italian Restaurant  0.05
3              Bakery  0.05
4  American Restaurant  0.05


----Summerhill East,Moore Park----
                     venue  freq
0             Tennis Court  0.25
1               Playground  0.25
2                     Park  0.25
3                    Trail  0.25
4  New American Restaurant  0.00


----Summerhill West,Forest Hill SE,Deer Park,South Hill,Rathnelly----
                venue  freq
0                 Pub  0.14
1        Coffee Shop  0.14
2   Convenience Store  0.07
```

```
3  Light Rail Station  0.07
4    Sushi Restaurant  0.07


----Swansea,Runnymede----
                venue  freq
0         Coffee Shop  0.11
1                Café  0.08
2         Pizza Place  0.08
3    Sushi Restaurant  0.05
4  Italian Restaurant  0.05


----The Beaches----
               venue  freq
0  Health Food Store  0.25
1                Pub  0.25
2        Coffee Shop  0.25
3        Yoga Studio  0.00
4       Noodle House  0.00


----The Beaches West,India Bazaar----
           venue  freq
0           Park  0.11
1    Pizza Place  0.05
2    Coffee Shop  0.05
3   Burger Joint  0.05
4  Burrito Place  0.05


----Underground city,First Canadian Place----
                venue  freq
0                Café  0.08
1         Coffee Shop  0.08
2               Hotel  0.06
3          Restaurant  0.05
4  American Restaurant  0.04


----University of Toronto,Harbord----
                venue  freq
0                Café  0.12
1  Japanese Restaurant  0.06
2            Bookstore  0.06
3         Coffee Shop  0.06
4                 Bar  0.06


----Yorkville,The Annex,North Midtown----
            venue  freq
0  Sandwich Place  0.12
1           Café  0.12
2     Coffee Shop  0.12
3     Pizza Place  0.08
4  History Museum  0.04
```

First, let's write a function to sort the venues in descending order.

In [36]:
```
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 [37]:
```
num_top_venues = 5

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(toro
nto_grouped.iloc[ind, :], num_top_venues)

neighborhoods_venues_sorted
```

Out[37]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
0	Adelaide,Richmond,King	Coffee Shop	Café	American Restaurant	Thai Restaurant	Steakhouse
1	Berczy Park	Coffee Shop	Restaurant	Cocktail Bar	Bakery	Pub
2	Business Reply Mail Processing Centre 969 Eastern	Yoga Studio	Auto Workshop	Pizza Place	Recording Studio	Restaurant
3	Central Bay Street	Coffee Shop	Café	Italian Restaurant	Burger Joint	Bar
4	Christie	Café	Grocery Store	Park	Nightclub	Convenience Store
5	Church and Wellesley	Japanese Restaurant	Sushi Restaurant	Coffee Shop	Gay Bar	Burger Joint
6	Commerce Court,Victoria Hotel	Coffee Shop	Café	Restaurant	Hotel	American Restaurant
7	Davisville	Pizza Place	Sandwich Place	Dessert Shop	Coffee Shop	Sushi Restaurant
8	Davisville North	Burger Joint	Sandwich Place	Breakfast Spot	Gym	Grocery Store
9	Design Exchange,Toronto Dominion Centre	Coffee Shop	Café	Hotel	Restaurant	American Restaurant
10	Dovercourt Village,Dufferin	Discount Store	Pharmacy	Supermarket	Bakery	Music Venue
11	Exhibition Place,Parkdale Village,Brockton	Breakfast Spot	Coffee Shop	Café	Performing Arts Venue	Burrito Place
12	Forest Hill West,Forest Hill North	Park	Trail	Sushi Restaurant	Jewelry Store	Women's Store
13	Harbourfront East,Toronto Islands,Union Station	Coffee Shop	Aquarium	Hotel	Café	Pizza Place
14	Harbourfront,Regent Park	Coffee Shop	Park	Café	Bakery	Pub
15	High Park,The Junction South	Mexican Restaurant	Café	Bar	Bakery	Fried Chicken Joint
16	Kensington Market,Grange Park,Chinatown	Café	Bar	Vietnamese Restaurant	Vegetarian / Vegan Restaurant	Coffee Shop
17	Lawrence Park	Park	Swim School	Dim Sum Restaurant	Bus Line	Women's Store
18	Little Portugal,Trinity	Bar	Men's Store	Coffee Shop	Asian Restaurant	Cocktail Bar
19	North Toronto West	Sporting Goods Shop	Coffee Shop	Yoga Studio	Gym / Fitness Center	Clothing Store
20	Parkdale,Roncesvalles	Breakfast Spot	Gift Shop	Restaurant	Dog Run	Burger Joint
21	Riverdale,The Danforth West	Greek Restaurant	Coffee Shop	Ice Cream Shop	Bookstore	Italian Restaurant

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
22	Rosedale	Park	Playground	Trail	Diner	Fast Food Restaurant
23	Roselawn	Home Service	Garden	Women's Store	Dog Run	Filipino Restaurant
24	Ryerson,Garden District	Coffee Shop	Clothing Store	Café	Cosmetics Shop	Middle Eastern Restaurant
25	South Niagara,Bathurst Quay,King and Spadina,R...	Airport Terminal	Airport Service	Airport Lounge	Boat or Ferry	Airport Gate
26	St. James Town	Coffee Shop	Restaurant	Café	Hotel	Clothing Store
27	St. James Town,Cabbagetown	Coffee Shop	Restaurant	Market	Park	Italian Restaurant
28	Stn A PO Boxes 25 The Esplanade	Coffee Shop	Restaurant	Café	Cocktail Bar	Seafood Restaurant
29	Studio District	Café	Coffee Shop	Italian Restaurant	Bakery	American Restaurant
30	Summerhill East,Moore Park	Playground	Park	Tennis Court	Trail	Falafel Restaurant
31	Summerhill West,Forest Hill SE,Deer Park,South...	Pub	Coffee Shop	Light Rail Station	Sushi Restaurant	Supermarket
32	Swansea,Runnymede	Coffee Shop	Café	Pizza Place	Sushi Restaurant	Italian Restaurant
33	The Beaches	Health Food Store	Coffee Shop	Pub	Discount Store	Filipino Restaurant
34	The Beaches West,India Bazaar	Park	Italian Restaurant	Pet Store	Gym	Coffee Shop
35	Underground city,First Canadian Place	Café	Coffee Shop	Hotel	Restaurant	American Restaurant
36	University of Toronto,Harbord	Café	Bakery	Bar	Japanese Restaurant	Bookstore
37	Yorkville,The Annex,North Midtown	Coffee Shop	Café	Sandwich Place	Pizza Place	BBQ Joint

cluster neighborhoods

In [38]:
```
# set number of clusters
kclusters = 5

toronto_grouped_clustering = toronto_grouped.drop('Neighborhood', 1)

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

# check cluster labels generated for each row in the dataframe
kmeans.labels_[0:5]
```

Out[38]:
```
array([0, 0, 0, 0, 4], dtype=int32)
```

In [39]:
```
toronto_merged = df_toronto

# add clustering labels
toronto_merged['Cluster Labels'] = kmeans.labels_

# merge toronto_grouped with toronto_data to add latitude/longitude for each n
eighborhood
toronto_merged = toronto_merged.join(neighborhoods_venues_sorted.set_index('Ne
ighborhood'), on='Neighborhood')

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

Out[39]:

	Postcode	Borough	Neighborhood	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3r Co
0	M4E	East Toronto	The Beaches	43.676357	-79.293031	0	Health Food Store	Coffee Shop	
1	M4K	East Toronto	Riverdale,The Danforth West	43.679557	-79.352188	0	Greek Restaurant	Coffee Shop	Ice
2	M4L	East Toronto	The Beaches West,India Bazaar	43.668999	-79.315572	0	Park	Italian Restaurant	P
3	M4M	East Toronto	Studio District	43.659526	-79.340923	0	Café	Coffee Shop	Res
4	M4N	Central Toronto	Lawrence Park	43.728020	-79.388790	4	Park	Swim School	D Res

In [40]:
```
toronto_merged.tail()
```

Out[40]:

	Postcode	Borough	Neighborhood	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Mo Comme Ven
33	M6K	West Toronto	Exhibition Place,Parkdale Village,Brockton	43.636847	-79.428191	0	Breakfast Spot	Coff Sh
34	M6P	West Toronto	High Park,The Junction South	43.661608	-79.464763	0	Mexican Restaurant	Ca
35	M6R	West Toronto	Parkdale,Roncesvalles	43.648960	-79.456325	0	Breakfast Spot	Gift Sh
36	M6S	West Toronto	Swansea,Runnymede	43.651571	-79.484450	0	Coffee Shop	Ca
37	M7Y	East Toronto	Business Reply Mail Processing Centre 969 Eastern	43.662744	-79.321558	0	Yoga Studio	Au Worksh

Final output - cluster map by venue

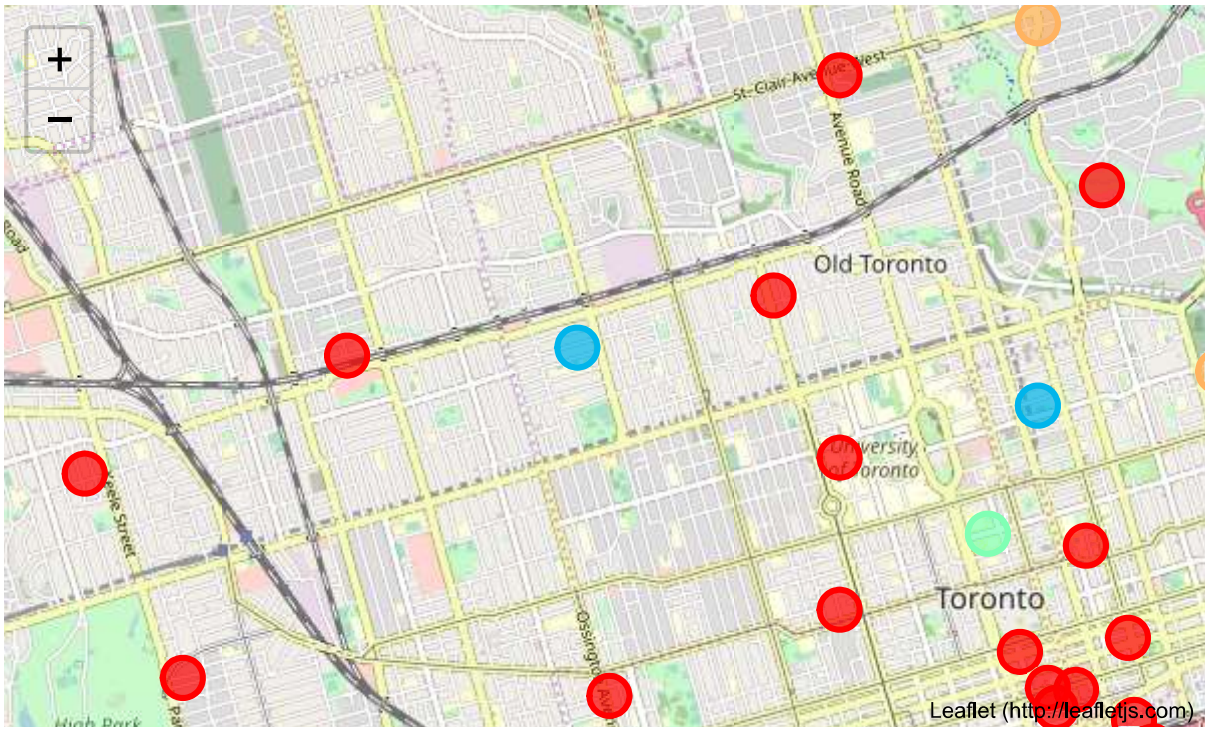
In [41]:
```
# create map
map_clusters = folium.Map(location=[latitude, longitude], zoom_start=13)

# 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 Labels'
]):
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), parse_html=Tru
e)
    folium.CircleMarker(
        [lat, lon],
        radius=10,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7).add_to(map_clusters)

map_clusters
```

Out[41]:



In [ ]: