PGA Segmenting and Clustering Neighborhoods in Toronto

March 1, 2020

1 Coursera Capstone Project: Segmenting and Clustering Neighborhoods in Toronto

Author: Kapil Kumar Nagwanshi

1.1 Assignmen Question 1

- **For this assignment, we have to explore and cluster the neighborhoods in Toronto.**

1.1.1 Part 1 of Question 1

1. Start by creating a new Notebook for this assignment. –This is the notebook

Before we get the data and start exploring it, let's download all the dependencies that we will need.

[2]:
```
#import necessary libraries
import numpy as np # library to handle data in a vectorized manner
import pandas as pd # library for data analsysis
import requests # Library for web scraping
import requests
from urllib.request import urlopen
from bs4 import BeautifulSoup
import ssl
import csv
print('Library import done...')
```

```
Library import done...
```

[3]:
```
# Ignore SSL certificate errors
ctx = ssl.create_default_context()
ctx.check_hostname = False
ctx.verify_mode = ssl.CERT_NONE
print('SSL certificate errors ignored...')
```

```
SSL certificate errors ignored...
```

1.1.2 Part 2 of Question 1

2. Use the Notebook to build the code to scrape the following Wikipedia page, https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M, in order to obtain the data that is in the table of postal codes and to transform the data into a pandas dataframe like the one shown in assignment.

This part shows raw data frame obtained from https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada withput cleaning

[4]:
```
#beautifulSoup instances
res_site = requests.get("https://en.wikipedia.org/wiki/
↪List_of_postal_codes_of_Canada:_M")
soup = BeautifulSoup(res_site.content,'lxml')
table = soup.find_all('table')[0]
#toronto_data = pd.read_html(str(table))[0]
##########################################
table_rows = table.tbody.find_all("tr")

res = []
for tr in table_rows:
    td = tr.find_all("td")
    row = [tr.text for tr in td]

    # Only process the cells that have an assigned borough. Ignore cells with a␣
↪borough that is Not assigned.
    if row != [] and row[1] != "Not assigned":
        # If a cell has a borough but a "Not assigned" neighborhood, then the␣
↪neighborhood will be the same as the borough.
        if "Not assigned" in row[2]:
            row[2] = row[1]
        res.append(row)

# Dataframe with 3 columns
df_toronto = pd.DataFrame(res, columns = ["PostalCode", "Borough",␣
↪"Neighborhood"])
df_toronto.head()
```

[4]:

	PostalCode	Borough	Neighborhood
0	M3A	North York	Parkwoods\n
1	M4A	North York	Victoria Village\n
2	M5A	Downtown Toronto	Harbourfront\n
3	M6A	North York	Lawrence Heights\n
4	M6A	North York	Lawrence Manor\n

[5]:
```
# Remove '\n' from Neighborhood
df_toronto["Neighborhood"] = df_toronto["Neighborhood"].str.replace("\n","")
```

```
df_toronto.head()
```

```
[5]:   PostalCode           Borough      Neighborhood
     0        M3A        North York         Parkwoods
     1        M4A        North York  Victoria Village
     2        M5A  Downtown Toronto      Harbourfront
     3        M6A        North York  Lawrence Heights
     4        M6A        North York    Lawrence Manor
```

```
[6]: df_toronto.shape
```

```
[6]: (210, 3)
```

1.1.3 Part 3 of Question 1

3. To create the above dataframe:

- The dataframe will consist of three columns: PostalCode, Borough, and Neighborhood
- Only process the cells that have an assigned borough. Ignore cells with a borough that is Not assigned.
- More than one neighborhood can exist in one postal code area. For example, in the table on the Wikipedia page, you will notice that M5A is listed twice and has two neighborhoods: Harbourfront and Regent Park. These two rows will be combined into one row with the neighborhoods separated with a comma as shown in row 11 in the above table.
- If a cell has a borough but a Not assigned neighborhood, then the neighborhood will be the same as the borough.
- Clean your Notebook and add Markdown cells to explain your work and any assumptions you are making.
- In the last cell of your notebook, use the .shape method to print the number of rows of your dataframe.

This part shows cleaned obtained dataframe https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_ with grouping of same postal codes

```
[7]: df_toronto = df_toronto.groupby(["PostalCode", "Borough"])["Neighborhood"].
      ↪apply(", ".join).reset_index()
     df_toronto.head()
```

```
[7]:   PostalCode      Borough                           Neighborhood
     0        M1B  Scarborough                         Rouge, Malvern
     1        M1C  Scarborough  Highland Creek, Rouge Hill, Port Union
     2        M1E  Scarborough       Guildwood, Morningside, West Hill
     3        M1G  Scarborough                                 Woburn
     4        M1H  Scarborough                              Cedarbrae
```

```
[8]: df_toronto.shape
```

[8]: (103, 3)

[9]:
```
df_toronto.head(15)
```

[9]:

	PostalCode	Borough	Neighborhood
0	M1B	Scarborough	Rouge, Malvern
1	M1C	Scarborough	Highland Creek, Rouge Hill, Port Union
2	M1E	Scarborough	Guildwood, Morningside, West Hill
3	M1G	Scarborough	Woburn
4	M1H	Scarborough	Cedarbrae
5	M1J	Scarborough	Scarborough Village
6	M1K	Scarborough	East Birchmount Park, Ionview, Kennedy Park
7	M1L	Scarborough	Clairlea, Golden Mile, Oakridge
8	M1M	Scarborough	Cliffcrest, Cliffside, Scarborough Village West
9	M1N	Scarborough	Birch Cliff, Cliffside West
10	M1P	Scarborough	Dorset Park, Scarborough Town Centre, Wexford ...
11	M1R	Scarborough	Maryvale, Wexford
12	M1S	Scarborough	Agincourt
13	M1T	Scarborough	Clarks Corners, Sullivan, Tam O'Shanter
14	M1V	Scarborough	Agincourt North, L'Amoreaux East, Milliken, St...

1.1.4 Part 4 of the Question 1

- Submit a link to your Notebook on your Github repository. (10 marks) #### End of Question 1

1.2 Assignment Question 2

Now that you have built a dataframe of the postal code of each neighborhood along with the borough name and neighborhood name, in order to utilize the Foursquare location data, we need to get the latitude and the longitude coordinates of each neighborhood.

In an older version of this course, we were leveraging the Google Maps Geocoding API to get the latitude and the longitude coordinates of each neighborhood. However, recently Google started charging for their API: http://geoawesomeness.com/developers-up-in-arms-over-google-maps-api-insane-price-hike/, so we will use the Geocoder Python package instead: https://geocoder.readthedocs.io/index.html.

The problem with this Package is you have to be persistent sometimes in order to get the geographical coordinates of a given postal code. So you can make a call to get the latitude and longitude coordinates of a given postal code and the result would be None, and then make the call again and you would get the coordinates. So, in order to make sure that you get the coordinates for all of our neighborhoods, you can run a while loop for each postal code.

- Check for geopy and geocoder packages

```
[10]: import geopy
from  geopy.geocoders import Nominatim
nominatim_service = Nominatim(user_agent='X@yy.com') # Important line
geopy.geocoders.options.default_user_agent = "X@yy.com" # Important line
geolocator = Nominatim()
```

```
[11]: city ="Toronto"
country ="Canada"
loc = geolocator.geocode(city+','+ country)
print("latitude is :-" ,loc.latitude,"\nlongtitude is:-" ,loc.longitude)
```

```
latitude is :- 43.653963
longtitude is:- -79.387207
```

```
[12]: location = geolocator.geocode("Toronto, North York, Parkwoods")
print(location.address)
print('')
print((location.latitude, location.longitude))
print('')
print(location.raw)
```

```
Parkwoods Village Drive, Parkway East, Don Valley East, North York, Toronto,
Golden Horseshoe, Ontario, M3A 2X2, Canada

(43.7587999, -79.3201966)
```


```
{'place_id': 124974741, 'licence': 'Data Âľ OpenStreetMap contributors, ODbL 1.0.
https://osm.org/copyright', 'osm_type': 'way', 'osm_id': 160406961,
'boundingbox': ['43.7576231', '43.761106', '-79.3239088', '-79.316215'], 'lat':
'43.7587999', 'lon': '-79.3201966', 'display_name': 'Parkwoods Village Drive,
Parkway East, Don Valley East, North York, Toronto, Golden Horseshoe, Ontario,
M3A 2X2, Canada', 'class': 'highway', 'type': 'secondary', 'importance': 0.51}
```


1.2.1 Get the latitude and the longitude coordinates of each neighborhood

```
[13]: import geopy
from  geopy.geocoders import Nominatim
import pandas as pd
locator = Nominatim(user_agent="KapilsGeocoder")
location = locator.geocode("Toronto, Canada")
from geopy.extra.rate_limiter import RateLimiter
# PostalCode        Borough        Neighborhood
df_temp=df_toronto
# 1 - conveneint function to delay between geocoding calls
geocode = RateLimiter(locator.geocode, min_delay_seconds=1)
# 2- - create location column
```

```
df_temp['Address'] = df_temp['PostalCode'].astype(str) + ',' + ' Toronto'
df_temp['Location'] = df_temp['Address'].apply(geocode)
# 3 - create longitude, laatitude and altitude from location column (returns␣
↪tuple)
df_temp['Point'] = df_temp['Location'].apply(lambda loc: tuple(loc.point) if loc␣
↪else None)
# 4 - split point column into latitude, longitude and altitude columns
# df_temp[['latitude', 'longitude', 'altitude']] = pd.DataFrame(df_temp['Point'].
↪tolist(), index=df_temp.index)
```

[14]:
```
df_temp
```

[14]:
```
    PostalCode      Borough  \
0          M1B  Scarborough
1          M1C  Scarborough
2          M1E  Scarborough
3          M1G  Scarborough
4          M1H  Scarborough
..         ...          ...
98         M9N         York
99         M9P    Etobicoke
100        M9R    Etobicoke
101        M9V    Etobicoke
102        M9W    Etobicoke

                                          Neighborhood       Address  \
0                                       Rouge, Malvern  M1B, Toronto
1               Highland Creek, Rouge Hill, Port Union  M1C, Toronto
2                    Guildwood, Morningside, West Hill  M1E, Toronto
3                                               Woburn  M1G, Toronto
4                                            Cedarbrae  M1H, Toronto
..                                                 ...           ...
98                                              Weston  M9N, Toronto
99                                           Westmount  M9P, Toronto
100  Kingsview Village, Martin Grove Gardens, Richv...  M9R, Toronto
101  Albion Gardens, Beaumond Heights, Humbergate, ...  M9V, Toronto
102                                          Northwest  M9W, Toronto

                                              Location  \
0    (Toronto, Punta Gorda, Montevideo, 11403, Urug...
1    (Toronto, Punta Gorda, Montevideo, 11403, Urug...
2                                                 None
3    (ScarboroughâĂŤGuildwood, Scarborough, Toronto, ...
4                                                 None
..                                                 ...
98                                                None
99                                                None
```

```
100  (Etobicoke Centre, Etobicoke, Toronto, Golden ...
101                                              None
102                                              None

                                           Point
0                (-34.8899421, -56.0790982, 0.0)
1                (-34.8899421, -56.0790982, 0.0)
2                                           None
3    (43.76571676956549, -79.22189842824983, 0.0)
4                                           None
..                                           ...
98                                          None
99                                          None
100  (43.69516618990701, -79.55088985426742, 0.0)
101                                         None
102                                         None

[103 rows x 6 columns]
```

1.2.2 From above simple solution, we are not able to get the geohraphical coordinates of the neighborhoods using the Geocoder package, we use the given csv file instead.

[15]:
```python
df_geo_coor = pd.read_csv("./Geospatial_Coordinates.csv")
df_geo_coor.head()
```

[15]:

	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

[16]:
```python
df_toronto.head()
# df_toronto dataframe played through geocoder
# beacause of call limits it won't work for all see 'none' in point columns
```

[16]:
```
  PostalCode      Borough                                 Neighborhood  \
0        M1B  Scarborough                               Rouge, Malvern
1        M1C  Scarborough  Highland Creek, Rouge Hill, Port Union
2        M1E  Scarborough       Guildwood, Morningside, West Hill
3        M1G  Scarborough                                     Woburn
4        M1H  Scarborough                                  Cedarbrae

       Address                                          Location  \
0  M1B, Toronto  (Toronto, Punta Gorda, Montevideo, 11403, Urug...
```

```
1  M1C, Toronto  (Toronto, Punta Gorda, Montevideo, 11403, Urug...
2  M1E, Toronto                                               None
3  M1G, Toronto  (ScarboroughâĂŤGuildwood, Scarborough, Toronto, ...
4  M1H, Toronto                                               None

                                         Point
0              (-34.8899421, -56.0790982, 0.0)
1              (-34.8899421, -56.0790982, 0.0)
2                                         None
3  (43.76571676956549, -79.22189842824983, 0.0)
4                                         None
```

[17]:
```
# drop address location and point columns to get original dataframe
df_toronto.drop(['Address', 'Location', 'Point'], axis=1, inplace=True)
df_toronto.head()
```

[17]:

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

Now We need to couple 2 dataframes "df_toronto" and "df_geo_coor" into one dataframe.

[18]:
```
df_toronto2 = pd.merge(df_toronto, df_geo_coor, how='left', left_on =␣
↪'PostalCode', right_on = 'Postal Code')
# remove the "Postal Code" column
df_toronto2.drop("Postal Code", axis=1, inplace=True)
df_toronto2.head()
```

[18]:

	PostalCode	Borough	Neighborhood	Latitude \
0	M1B	Scarborough	Rouge, Malvern	43.806686
1	M1C	Scarborough	Highland Creek, Rouge Hill, Port Union	43.784535
2	M1E	Scarborough	Guildwood, Morningside, West Hill	43.763573
3	M1G	Scarborough	Woburn	43.770992
4	M1H	Scarborough	Cedarbrae	43.773136

	Longitude
0	-79.194353
1	-79.160497
2	-79.188711
3	-79.216917
4	-79.239476

1.3 Assignment Question 3 Explore and cluster the neighborhoods in Toronto

Explore and cluster the neighborhoods in Toronto. You can decide to work with only boroughs that contain the word Toronto and then replicate the same analysis we did to the New York City data. It is up to you.

Just make sure:

1. to add enough Markdown cells to explain what you decided to do and to report any observations you make.
2. to generate maps to visualize your neighborhoods and how they cluster together.

[19]:
```
address = "Toronto, ON"
geolocator = Nominatim(user_agent="toronto_explorer") # Changed user agent due␣
↪to collision
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geographical coordinate of Toronto city are {}, {}.'.format(latitude,␣
↪longitude))
```

```
The geographical coordinate of Toronto city are 43.653963, -79.387207.
```

1.3.1 Create a map of the whole Toronto City with neighborhoods superimposed on top

[20]:
```
import folium # map rendering library

# import k-means from clustering stage
from sklearn.cluster import KMeans

# Matplotlib and associated plotting modules
import matplotlib.cm as cm
import matplotlib.colors as colors
```

[21]:
```
# create map of Toronto using latitude and longitude values
map_toronto = folium.Map(location=[latitude, longitude], zoom_start=10)
map_toronto
```

[21]:
```
<folium.folium.Map at 0x25cb3559ef0>
```

1.3.2 Add markers to the map.

[22]:
```
for lat, lng, borough, neighborhood in zip(
        df_toronto2['Latitude'],
        df_toronto2['Longitude'],
        df_toronto2['Borough'],
```

```
        df_toronto2['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
```

[22]: <folium.folium.Map at 0x25cb3559ef0>

1.3.3 Map of a part of Toronto City

We are going to work with only the boroughs that contain the word "Toronto".

[23]:
```
# "denc" = [D]owntown Toronto, [E]ast Toronto, [N]orth Toronto, [C]entral Toronto
df_toronto_denc = df_toronto2[df_toronto['Borough'].str.contains("Toronto")].
 ↪reset_index(drop=True)
df_toronto_denc.head()
```

[23]:
```
  PostalCode          Borough                    Neighborhood   Latitude  \
0        M4E     East Toronto                     The Beaches  43.676357
1        M4K     East Toronto    The Danforth West, Riverdale  43.679557
2        M4L     East Toronto  The Beaches West, India Bazaar  43.668999
3        M4M     East Toronto                 Studio District  43.659526
4        M4N  Central Toronto                   Lawrence Park  43.728020

   Longitude
0 -79.293031
1 -79.352188
2 -79.315572
3 -79.340923
4 -79.388790
```

1.3.4 New marked map

[24]:
```
map_toronto_denc = folium.Map(location=[latitude, longitude], zoom_start=12)
for lat, lng, borough, neighborhood in zip(
        df_toronto_denc['Latitude'],
        df_toronto_denc['Longitude'],
        df_toronto_denc['Borough'],
        df_toronto_denc['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_denc)

map_toronto_denc
```

[24]:
```
<folium.folium.Map at 0x25cb400eb38>
```

1.3.5 Define Foursquare Credentials and Version

On the public repository on Github, I has removed this field for the privacy!

[25]:
```
CLIENT_ID = 'DDUUMMYYDDUUMMYYDDUUMMYYDDUUMMYYDDUUMMYY' # your Foursquare ID
CLIENT_SECRET = 'DDUUMMYYDDUUMMYYDDUUMMYYDDUUMMYY' # your Foursquare Secret
VERSION = '12345678'
LIMIT = 30
print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
```

```
Your credentails:
CLIENT_ID: DDUUMMYYDDUUMMYYDDUUMMYYDDUUMMYYDDUUMMYY
CLIENT_SECRET:DDUUMMYYDDUUMMYYDDUUMMYYDDUUMMYY
```

1.3.6 Explore the first neighborhood in our data frame "df_toronto_denc"

[28]:
```
neighborhood_name = df_toronto_denc.loc[0, 'Neighborhood']
print(f"The first neighborhood's name is '{neighborhood_name}'.")
```

```
The first neighborhood's name is 'The Beaches'.
```

Get the neighborhood's latitude and longitude values.

[29]:
```
neighborhood_latitude = df_toronto_denc.loc[0, 'Latitude'] # neighborhood␣
↪latitude value
neighborhood_longitude = df_toronto_denc.loc[0, 'Longitude'] # neighborhood␣
↪longitude value

print('Latitude and longitude values of {} are {}, {}.'.format(neighborhood_name,
                                                                ␣
↪neighborhood_latitude,
                                                                ␣
↪neighborhood_longitude))
```

```
Latitude and longitude values of The Beaches are 43.67635739999999, -79.2930312.
```

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

[31]:
```
#CLIENT_ID = 'DDUUMMYYDDUUMMYYDDUUMMYYDDUUMMYYDDUUMMYY' # your Foursquare ID
#CLIENT_SECRET = 'DDUUMMYYDDUUMMYYDDUUMMYYDDUUMMYY' # your Foursquare Secret
#VERSION = '12345678'
url = 'https://api.foursquare.com/v2/venues/explore?
↪&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
    CLIENT_ID,
    CLIENT_SECRET,
    VERSION,
    neighborhood_latitude,
    neighborhood_longitude,
    radius,
    LIMIT)
# get the result to a json file
results = requests.get(url).json()
```

Function that extracts the category of the venue

[32]:
```
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']
```

Now we are ready to clean the json and structure it into a pandas dataframe.

[33]:
```
from pandas.io.json import json_normalize  # tranform JSON file into a pandas␣
 ↪dataframe
venues = results['response']['groups'][0]['items']
nearby_venues = json_normalize(venues) # flatten JSON

# filter columns
filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat',␣
 ↪'venue.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,␣
 ↪axis=1)

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

nearby_venues
```

```
C:\ProgramData\Anaconda3\lib\site-packages\ipykernel_launcher.py:3:
FutureWarning: pandas.io.json.json_normalize is deprecated, use
pandas.json_normalize instead
  This is separate from the ipykernel package so we can avoid doing imports
until
```

[33]:

	name	categories	lat	lng
0	Glen Manor Ravine	Trail	43.676821	-79.293942
1	The Big Carrot Natural Food Market	Health Food Store	43.678879	-79.297734
2	Grover Pub and Grub	Pub	43.679181	-79.297215
3	Domino's Pizza	Pizza Place	43.679058	-79.297382
4	Upper Beaches	Neighborhood	43.680563	-79.292869

1.3.8 Explore neighborhoods in a part of Toronto City

We are working on the data frame df_toronto_denc. Recall that, this region contain DENC of Toronto where,

"DENC" = [D]owntown Toronto, [E]ast Toronto, [N]orth Toronto, [C]entral Toronto

First, let's create a function to repeat the same process to all the neighborhoods in DENC of Toronto.

[36]:
```
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={},{}&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)
```

Now write the code to run the above function on each neighborhood and create a new dataframe called toronto_denc_venues

[35]:
```
toronto_denc_venues = getNearbyVenues(names=df_toronto_denc['Neighborhood'],
                                   latitudes=df_toronto_denc['Latitude'],
                                   longitudes=df_toronto_denc['Longitude']
                                  )
toronto_denc_venues.head()
```

[35]:
```
  Neighborhood  Neighborhood Latitude  Neighborhood Longitude  \
0  The Beaches              43.676357              -79.293031
1  The Beaches              43.676357              -79.293031
2  The Beaches              43.676357              -79.293031
3  The Beaches              43.676357              -79.293031
4  The Beaches              43.676357              -79.293031

                                Venue  Venue Latitude  Venue Longitude  \
0                    Glen Manor Ravine       43.676821       -79.293942
1  The Big Carrot Natural Food Market       43.678879       -79.297734
2                  Grover Pub and Grub       43.679181       -79.297215
3                       Domino's Pizza       43.679058       -79.297382
4                       Upper Beaches       43.680563       -79.292869

      Venue Category
0              Trail
1  Health Food Store
2                Pub
3        Pizza Place
4       Neighborhood
```

Let's check how many venues were returned for each neighborhood.

[37]:
```
toronto_denc_venues.groupby('Neighborhood').count()
```

[37]:
```
                                                    Neighborhood Latitude  \
Neighborhood
Adelaide, King, Richmond                                              100
Berczy Park                                                            57
Brockton, Exhibition Place, Parkdale Village                           22
Business Reply Mail Processing Centre 969 Eastern                      18
CN Tower, Bathurst Quay, Island airport, Harbou...                     16
Cabbagetown, St. James Town                                            44
Central Bay Street                                                     79
Chinatown, Grange Park, Kensington Market                              87
Christie                                                               17
Church and Wellesley                                                   85
Commerce Court, Victoria Hotel                                        100
Davisville                                                             38
Davisville North                                                        9
Deer Park, Forest Hill SE, Rathnelly, South Hil...                     15
```

```
Design Exchange, Toronto Dominion Centre                         100
Dovercourt Village, Dufferin                                      17
First Canadian Place, Underground city                           100
Forest Hill North, Forest Hill West                                4
Harbord, University of Toronto                                    39
Harbourfront                                                      50
Harbourfront East, Toronto Islands, Union Station                100
High Park, The Junction South                                     23
Lawrence Park                                                      3
Little Portugal, Trinity                                          57
Moore Park, Summerhill East                                        2
North Toronto West                                                20
Parkdale, Roncesvalles                                            15
Queen's Park                                                      41
Rosedale                                                           4
Roselawn                                                           3
Runnymede, Swansea                                                38
Ryerson, Garden District                                         100
St. James Town                                                   100
Stn A PO Boxes 25 The Esplanade                                   95
Studio District                                                   43
The Annex, North Midtown, Yorkville                               22
The Beaches                                                        5
The Beaches West, India Bazaar                                    19
The Danforth West, Riverdale                                      41

                                                    Neighborhood Longitude  \
Neighborhood
Adelaide, King, Richmond                                              100
Berczy Park                                                            57
Brockton, Exhibition Place, Parkdale Village                           22
Business Reply Mail Processing Centre 969 Eastern                      18
CN Tower, Bathurst Quay, Island airport, Harbou...                     16
Cabbagetown, St. James Town                                            44
Central Bay Street                                                     79
Chinatown, Grange Park, Kensington Market                              87
Christie                                                               17
Church and Wellesley                                                   85
Commerce Court, Victoria Hotel                                        100
Davisville                                                             38
Davisville North                                                        9
Deer Park, Forest Hill SE, Rathnelly, South Hil...                     15
Design Exchange, Toronto Dominion Centre                              100
Dovercourt Village, Dufferin                                           17
First Canadian Place, Underground city                                100
Forest Hill North, Forest Hill West                                     4
Harbord, University of Toronto                                         39
```

Harbourfront	50
Harbourfront East, Toronto Islands, Union Station	100
High Park, The Junction South	23
Lawrence Park	3
Little Portugal, Trinity	57
Moore Park, Summerhill East	2
North Toronto West	20
Parkdale, Roncesvalles	15
Queen's Park	41
Rosedale	4
Roselawn	3
Runnymede, Swansea	38
Ryerson, Garden District	100
St. James Town	100
Stn A PO Boxes 25 The Esplanade	95
Studio District	43
The Annex, North Midtown, Yorkville	22
The Beaches	5
The Beaches West, India Bazaar	19
The Danforth West, Riverdale	41

	Venue	Venue Latitude \
Neighborhood		
Adelaide, King, Richmond	100	100
Berczy Park	57	57
Brockton, Exhibition Place, Parkdale Village	22	22
Business Reply Mail Processing Centre 969 Eastern	18	18
CN Tower, Bathurst Quay, Island airport, Harbou...	16	16
Cabbagetown, St. James Town	44	44
Central Bay Street	79	79
Chinatown, Grange Park, Kensington Market	87	87
Christie	17	17
Church and Wellesley	85	85
Commerce Court, Victoria Hotel	100	100
Davisville	38	38
Davisville North	9	9
Deer Park, Forest Hill SE, Rathnelly, South Hil...	15	15
Design Exchange, Toronto Dominion Centre	100	100
Dovercourt Village, Dufferin	17	17
First Canadian Place, Underground city	100	100
Forest Hill North, Forest Hill West	4	4
Harbord, University of Toronto	39	39
Harbourfront	50	50
Harbourfront East, Toronto Islands, Union Station	100	100
High Park, The Junction South	23	23
Lawrence Park	3	3
Little Portugal, Trinity	57	57

Moore Park, Summerhill East	2	2
North Toronto West	20	20
Parkdale, Roncesvalles	15	15
Queen's Park	41	41
Rosedale	4	4
Roselawn	3	3
Runnymede, Swansea	38	38
Ryerson, Garden District	100	100
St. James Town	100	100
Stn A PO Boxes 25 The Esplanade	95	95
Studio District	43	43
The Annex, North Midtown, Yorkville	22	22
The Beaches	5	5
The Beaches West, India Bazaar	19	19
The Danforth West, Riverdale	41	41

	Venue Longitude \
Neighborhood	
Adelaide, King, Richmond	100
Berczy Park	57
Brockton, Exhibition Place, Parkdale Village	22
Business Reply Mail Processing Centre 969 Eastern	18
CN Tower, Bathurst Quay, Island airport, Harbou...	16
Cabbagetown, St. James Town	44
Central Bay Street	79
Chinatown, Grange Park, Kensington Market	87
Christie	17
Church and Wellesley	85
Commerce Court, Victoria Hotel	100
Davisville	38
Davisville North	9
Deer Park, Forest Hill SE, Rathnelly, South Hil...	15
Design Exchange, Toronto Dominion Centre	100
Dovercourt Village, Dufferin	17
First Canadian Place, Underground city	100
Forest Hill North, Forest Hill West	4
Harbord, University of Toronto	39
Harbourfront	50
Harbourfront East, Toronto Islands, Union Station	100
High Park, The Junction South	23
Lawrence Park	3
Little Portugal, Trinity	57
Moore Park, Summerhill East	2
North Toronto West	20
Parkdale, Roncesvalles	15
Queen's Park	41
Rosedale	4

Roselawn	3
Runnymede, Swansea	38
Ryerson, Garden District	100
St. James Town	100
Stn A PO Boxes 25 The Esplanade	95
Studio District	43
The Annex, North Midtown, Yorkville	22
The Beaches	5
The Beaches West, India Bazaar	19
The Danforth West, Riverdale	41

	Venue Category
Neighborhood	
Adelaide, King, Richmond	100
Berczy Park	57
Brockton, Exhibition Place, Parkdale Village	22
Business Reply Mail Processing Centre 969 Eastern	18
CN Tower, Bathurst Quay, Island airport, Harbou...	16
Cabbagetown, St. James Town	44
Central Bay Street	79
Chinatown, Grange Park, Kensington Market	87
Christie	17
Church and Wellesley	85
Commerce Court, Victoria Hotel	100
Davisville	38
Davisville North	9
Deer Park, Forest Hill SE, Rathnelly, South Hil...	15
Design Exchange, Toronto Dominion Centre	100
Dovercourt Village, Dufferin	17
First Canadian Place, Underground city	100
Forest Hill North, Forest Hill West	4
Harbord, University of Toronto	39
Harbourfront	50
Harbourfront East, Toronto Islands, Union Station	100
High Park, The Junction South	23
Lawrence Park	3
Little Portugal, Trinity	57
Moore Park, Summerhill East	2
North Toronto West	20
Parkdale, Roncesvalles	15
Queen's Park	41
Rosedale	4
Roselawn	3
Runnymede, Swansea	38
Ryerson, Garden District	100
St. James Town	100
Stn A PO Boxes 25 The Esplanade	95

```
Studio District                                                    43
The Annex, North Midtown, Yorkville                                22
The Beaches                                                         5
The Beaches West, India Bazaar                                     19
The Danforth West, Riverdale                                       41
```

Let's find out how many unique categories can be curated from all the returned venues

[38]:
```
print('There are {} uniques categories.'.format(len(toronto_denc_venues['Venue␣
↪Category'].unique())))
```

```
There are 236 uniques categories.
```

1.3.9 Analyze Each Neighborhood

[39]:
```
# one hot encoding
toronto_denc_onehot = pd.get_dummies(toronto_denc_venues[['Venue Category']],␣
↪prefix="", prefix_sep="")

# add neighborhood column back to dataframe
toronto_denc_onehot['Neighborhood'] = toronto_denc_venues['Neighborhood']

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

toronto_denc_onehot.head()
```

[39]:
```
   Yoga Studio  Afghan Restaurant  Airport  Airport Food Court  Airport Gate  \
0            0                  0        0                   0             0
1            0                  0        0                   0             0
2            0                  0        0                   0             0
3            0                  0        0                   0             0
4            0                  0        0                   0             0

   Airport Lounge  Airport Service  Airport Terminal  American Restaurant  \
0               0                0                 0                    0
1               0                0                 0                    0
2               0                0                 0                    0
3               0                0                 0                    0
4               0                0                 0                    0

   Antique Shop  ...  Toy / Game Store  Trail  Train Station  \
0             0  ...                 0      1              0
1             0  ...                 0      0              0
```

```
2             0  ...                  0      0              0
3             0  ...                  0      0              0
4             0  ...                  0      0              0

   Vegetarian / Vegan Restaurant  Video Game Store  Vietnamese Restaurant  \
0                              0                 0                      0
1                              0                 0                      0
2                              0                 0                      0
3                              0                 0                      0
4                              0                 0                      0

   Wine Bar  Wine Shop  Wings Joint  Women's Store
0         0          0            0              0
1         0          0            0              0
2         0          0            0              0
3         0          0            0              0
4         0          0            0              0

[5 rows x 236 columns]
```

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

[40]:

```
toronto_denc_grouped = toronto_denc_onehot.groupby('Neighborhood').mean().
 ↪reset_index()
toronto_denc_grouped.head()
```

[40]:

```
                                         Neighborhood  Yoga Studio  \
0                            Adelaide, King, Richmond     0.000000
1                                         Berczy Park     0.000000
2        Brockton, Exhibition Place, Parkdale Village     0.000000
3  Business Reply Mail Processing Centre 969 Eastern     0.055556
4  CN Tower, Bathurst Quay, Island airport, Harbo...     0.000000

   Afghan Restaurant  Airport  Airport Food Court  Airport Gate  \
0                0.0   0.0000              0.0000        0.0000
1                0.0   0.0000              0.0000        0.0000
2                0.0   0.0000              0.0000        0.0000
3                0.0   0.0000              0.0000        0.0000
4                0.0   0.0625              0.0625        0.0625

   Airport Lounge  Airport Service  Airport Terminal  American Restaurant  \
0           0.000           0.0000             0.000                 0.02
1           0.000           0.0000             0.000                 0.00
2           0.000           0.0000             0.000                 0.00
3           0.000           0.0000             0.000                 0.00
4           0.125           0.1875             0.125                 0.00
```

```
   ...  Toy / Game Store  Trail  Train Station  Vegetarian / Vegan Restaurant  \
0  ...               0.0    0.0            0.0                       0.020000
1  ...               0.0    0.0            0.0                       0.017544
2  ...               0.0    0.0            0.0                       0.000000
3  ...               0.0    0.0            0.0                       0.000000
4  ...               0.0    0.0            0.0                       0.000000

   Video Game Store  Vietnamese Restaurant  Wine Bar  Wine Shop  Wings Joint  \
0               0.0                    0.0      0.01        0.0          0.0
1               0.0                    0.0      0.00        0.0          0.0
2               0.0                    0.0      0.00        0.0          0.0
3               0.0                    0.0      0.00        0.0          0.0
4               0.0                    0.0      0.00        0.0          0.0

   Women's Store
0           0.01
1           0.00
2           0.00
3           0.00
4           0.00

[5 rows x 236 columns]
```

Check the 10 most common venues in each neighborhood.

[41]:
```
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'] =␣
 ↪toronto_denc_grouped['Neighborhood']
```

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

neighborhoods_venues_sorted.head()
```

[41]:

```
                                        Neighborhood 1st Most Common Venue  \
0                           Adelaide, King, Richmond           Coffee Shop
1                                        Berczy Park           Coffee Shop
2        Brockton, Exhibition Place, Parkdale Village        Breakfast Spot
3  Business Reply Mail Processing Centre 969 Eastern           Yoga Studio
4  CN Tower, Bathurst Quay, Island airport, Harbo...       Airport Service

  2nd Most Common Venue 3rd Most Common Venue 4th Most Common Venue  \
0       Thai Restaurant            Restaurant                   Bar
1    Seafood Restaurant     French Restaurant        Farmers Market
2                 CafÃľ           Coffee Shop                   Gym
3         Auto Workshop                  Park           Pizza Place
4      Airport Terminal        Airport Lounge              Boutique

  5th Most Common Venue 6th Most Common Venue 7th Most Common Venue  \
0                 CafÃľ            Steakhouse      Sushi Restaurant
1                Bakery            Restaurant           Cheese Shop
2                Bakery               Stadium         Burrito Place
3            Restaurant               Butcher         Burrito Place
4   Rental Car Location         Boat or Ferry       Harbor / Marina

  8th Most Common Venue 9th Most Common Venue 10th Most Common Venue
0    Seafood Restaurant             Gastropub         Cosmetics Shop
1                 CafÃľ          Cocktail Bar               Beer Bar
2            Restaurant          Climbing Gym              Pet Store
3               Brewery            Skate Park             Smoke Shop
4      Sculpture Garden                   Bar           Airport Gate
```

1.3.10 Cluster neighborhoods

Run k-means to cluster the neighborhood into 5 clusters.

[42]:

```
# set number of clusters to 5
kclusters = 5

toronto_denc_grouped_clustering = toronto_denc_grouped.drop('Neighborhood', 1)

# run k-means clustering
```

```
kmeans = KMeans(n_clusters=kclusters, random_state=0).
 ↪fit(toronto_denc_grouped_clustering)

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

[42]:
```
array([1, 1, 1, 1, 1, 1, 1, 1, 1, 1])
```

Let's create a new dataframe that includes the cluster as well as the top 10 venues for each neighborhood

[43]:
```
# add clustering labels
neighborhoods_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)

toronto_denc_merged = df_toronto_denc

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

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

[43]:
```
  PostalCode          Borough                    Neighborhood   Latitude  \
0        M4E     East Toronto                     The Beaches  43.676357
1        M4K     East Toronto    The Danforth West, Riverdale  43.679557
2        M4L     East Toronto  The Beaches West, India Bazaar  43.668999
3        M4M     East Toronto                 Studio District  43.659526
4        M4N  Central Toronto                   Lawrence Park  43.728020

   Longitude  Cluster Labels 1st Most Common Venue 2nd Most Common Venue  \
0 -79.293031               2                 Trail           Pizza Place
1 -79.352188               1      Greek Restaurant           Coffee Shop
2 -79.315572               2        Sandwich Place                  Park
3 -79.340923               1                  CafÃľ           Coffee Shop
4 -79.388790               2                  Park              Bus Line

  3rd Most Common Venue 4th Most Common Venue 5th Most Common Venue  \
0     Health Food Store                   Pub               Dog Run
1    Italian Restaurant        Ice Cream Shop             Bookstore
2                   Gym                   Pub        Burrito Place
3             Gastropub                Bakery               Brewery
4           Swim School         Women's Store    Dim Sum Restaurant

    6th Most Common Venue 7th Most Common Venue        8th Most Common Venue  \
0      Dim Sum Restaurant                 Diner               Discount Store
1  Furniture / Home Store                Lounge                          Spa
```

```
2    Fast Food Restaurant    Italian Restaurant            Fish & Chips Shop
3      Italian Restaurant   American Restaurant                  Yoga Studio
4    Ethiopian Restaurant     Electronics Store  Eastern European Restaurant

     9th Most Common Venue 10th Most Common Venue
0      Distribution Center          Women's Store
1                  Brewery        Bubble Tea Shop
2               Steakhouse       Sushi Restaurant
3  Comfort Food Restaurant     Seafood Restaurant
4      Dumpling Restaurant             Donut Shop
```

[44]:
```python
# 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_denc_merged['Latitude'],
        toronto_denc_merged['Longitude'],
        toronto_denc_merged['Neighborhood'],
        toronto_denc_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
```

[44]: <folium.folium.Map at 0x25cb4252f98>

1.3.11 Examine Clusters

Now, you can examine each cluster and determine the discriminating venue categories that distinguish each cluster.

Cluster 0

```
[45]: toronto_denc_merged.loc[toronto_denc_merged['Cluster Labels'] == 0,␣
      ↪toronto_denc_merged.columns[[1] + list(range(5, toronto_denc_merged.
      ↪shape[1]))]]
```

```
[45]:             Borough  Cluster Labels 1st Most Common Venue  \
      10  Downtown Toronto               0                  Park
      23   Central Toronto               0                  Park

         2nd Most Common Venue 3rd Most Common Venue 4th Most Common Venue  \
      10            Playground                 Trail      Department Store
      23         Jewelry Store                 Trail      Sushi Restaurant

         5th Most Common Venue 6th Most Common Venue        7th Most Common Venue  \
      10  Ethiopian Restaurant     Electronics Store  Eastern European Restaurant
      23          Dessert Shop  Ethiopian Restaurant            Electronics Store

                8th Most Common Venue 9th Most Common Venue 10th Most Common Venue
      10           Dumpling Restaurant            Donut Shop       Doner Restaurant
      23  Eastern European Restaurant   Dumpling Restaurant             Donut Shop
```

Cluster 1

```
[46]: toronto_denc_merged.loc[toronto_denc_merged['Cluster Labels'] == 1,␣
      ↪toronto_denc_merged.columns[[1] + list(range(5, toronto_denc_merged.
      ↪shape[1]))]]
```

```
[46]:             Borough  Cluster Labels 1st Most Common Venue  \
      1       East Toronto               1      Greek Restaurant
      3       East Toronto               1                  CafÃľ
      6    Central Toronto               1        Clothing Store
      7    Central Toronto               1          Dessert Shop
      9    Central Toronto               1           Coffee Shop
      11  Downtown Toronto               1           Coffee Shop
      12  Downtown Toronto               1           Coffee Shop
      13  Downtown Toronto               1           Coffee Shop
      14  Downtown Toronto               1           Coffee Shop
      15  Downtown Toronto               1           Coffee Shop
      16  Downtown Toronto               1           Coffee Shop
      17  Downtown Toronto               1           Coffee Shop
      18  Downtown Toronto               1           Coffee Shop
      19  Downtown Toronto               1           Coffee Shop
      20  Downtown Toronto               1           Coffee Shop
      21  Downtown Toronto               1           Coffee Shop
      24   Central Toronto               1                  CafÃľ
      25  Downtown Toronto               1                  CafÃľ
      26  Downtown Toronto               1                   Bar
      27  Downtown Toronto               1       Airport Service
      28  Downtown Toronto               1           Coffee Shop
```

```
29  Downtown Toronto             1          Coffee Shop
30  Downtown Toronto             1        Grocery Store
31      West Toronto             1               Bakery
32      West Toronto             1                  Bar
33      West Toronto             1       Breakfast Spot
34      West Toronto             1   Mexican Restaurant
35      West Toronto             1       Breakfast Spot
36      West Toronto             1          Coffee Shop
37  Downtown Toronto             1          Coffee Shop
38      East Toronto             1          Yoga Studio

   2nd Most Common Venue  3rd Most Common Venue  \
1            Coffee Shop    Italian Restaurant
3            Coffee Shop             Gastropub
6            Coffee Shop                 CafÃľ
7         Sandwich Place           Pizza Place
9                    Pub           Pizza Place
11                 CafÃľ                Market
12   Japanese Restaurant               Gay Bar
13                  Park                Bakery
14        Clothing Store       Bubble Tea Shop
15                 CafÃľ            Restaurant
16    Seafood Restaurant     French Restaurant
17    Italian Restaurant   Japanese Restaurant
18       Thai Restaurant            Restaurant
19              Aquarium                 Hotel
20                 CafÃľ            Restaurant
21                 CafÃľ            Restaurant
24        Sandwich Place           Coffee Shop
25             Bookstore            Restaurant
26                 CafÃľ  Vietnamese Restaurant
27      Airport Terminal        Airport Lounge
28            Restaurant                 CafÃľ
29                 CafÃľ            Restaurant
30                 CafÃľ                  Park
31              Pharmacy           Music Venue
32           Coffee Shop            Restaurant
33                 CafÃľ           Coffee Shop
34       Thai Restaurant                   Bar
35             Gift Shop         Movie Theater
36                 CafÃľ       Sushi Restaurant
37                   Gym                  Park
38         Auto Workshop                  Park

         4th Most Common Venue      5th Most Common Venue  \
1                Ice Cream Shop                  Bookstore
3                        Bakery                    Brewery
```

6	Restaurant	Dessert Shop
7	Italian Restaurant	Sushi Restaurant
9	Bagel Shop	Restaurant
11	Italian Restaurant	Pizza Place
12	Restaurant	Sushi Restaurant
13	Pub	Mexican Restaurant
14	Middle Eastern Restaurant	CafÃľ
15	Clothing Store	Hotel
16	Farmers Market	Bakery
17	Juice Bar	Sandwich Place
18	Bar	CafÃľ
19	CafÃľ	Italian Restaurant
20	Hotel	Bakery
21	Hotel	Gym
24	American Restaurant	Middle Eastern Restaurant
25	Bakery	Bar
26	Vegetarian / Vegan Restaurant	Bakery
27	Boutique	Rental Car Location
28	Japanese Restaurant	Seafood Restaurant
29	American Restaurant	Hotel
30	Restaurant	Baby Store
31	Bank	Brewery
32	Asian Restaurant	Men's Store
33	Gym	Bakery
34	CafÃľ	Diner
35	Eastern European Restaurant	Italian Restaurant
36	Italian Restaurant	Pizza Place
37	Burger Joint	Fast Food Restaurant
38	Pizza Place	Restaurant

	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue \
1	Furniture / Home Store	Lounge	Spa
3	Italian Restaurant	American Restaurant	Yoga Studio
6	Miscellaneous Shop	Salon / Barbershop	Chinese Restaurant
7	CafÃľ	Coffee Shop	Gym
9	Fried Chicken Joint	Sports Bar	Supermarket
11	Bakery	Pub	Restaurant
12	Pub	Men's Store	Mediterranean Restaurant
13	Breakfast Spot	CafÃľ	Restaurant
14	Japanese Restaurant	Ramen Restaurant	Italian Restaurant
15	Breakfast Spot	American Restaurant	Cosmetics Shop
16	Restaurant	Cheese Shop	CafÃľ
17	Burger Joint	Chinese Restaurant	Bar
18	Steakhouse	Sushi Restaurant	Seafood Restaurant
19	Scenic Lookout	Brewery	Fried Chicken Joint
20	Seafood Restaurant	American Restaurant	Italian Restaurant
21	American Restaurant	Seafood Restaurant	Japanese Restaurant

24	Pub	BBQ Joint	History Museum
25	Japanese Restaurant	Italian Restaurant	Flower Shop
26	Coffee Shop	Chinese Restaurant	Mexican Restaurant
27	Boat or Ferry	Harbor / Marina	Sculpture Garden
28	Beer Bar	Hotel	Italian Restaurant
29	Seafood Restaurant	Bar	Steakhouse
30	Candy Store	Diner	Italian Restaurant
31	CafÃľ	Art Gallery	Middle Eastern Restaurant
32	CafÃľ	Pizza Place	Bakery
33	Stadium	Burrito Place	Restaurant
34	Italian Restaurant	Bakery	Flea Market
35	Dog Run	Bar	Bank
36	Gym	Bookstore	Scenic Lookout
37	Portuguese Restaurant	Nightclub	Music Venue
38	Butcher	Burrito Place	Brewery

	9th Most Common Venue	10th Most Common Venue
1	Brewery	Bubble Tea Shop
3	Comfort Food Restaurant	Seafood Restaurant
6	Fast Food Restaurant	Diner
7	Indoor Play Area	Japanese Restaurant
9	American Restaurant	Liquor Store
11	Gastropub	Indian Restaurant
12	Hotel	Gastropub
13	Theater	Shoe Store
14	Bookstore	Electronics Store
15	Italian Restaurant	Bakery
16	Cocktail Bar	Beer Bar
17	Department Store	Salad Place
18	Gastropub	Cosmetics Shop
19	Restaurant	Bakery
20	Japanese Restaurant	Bar
21	Deli / Bodega	Italian Restaurant
24	Metro Station	Pizza Place
25	Pub	Poutine Place
26	Dumpling Restaurant	Grocery Store
27	Bar	Airport Gate
28	Pub	Cheese Shop
29	Japanese Restaurant	Gym
30	Coffee Shop	Gas Station
31	Pool	Gym / Fitness Center
32	Vietnamese Restaurant	Wine Bar
33	Climbing Gym	Pet Store
34	Speakeasy	Fried Chicken Joint
35	Dessert Shop	Bookstore
36	Sandwich Place	Restaurant
37	Mexican Restaurant	Juice Bar

```
38              Skate Park              Smoke Shop
```

Cluster 2

[47]:
```
toronto_denc_merged.loc[toronto_denc_merged['Cluster Labels'] == 2,␣
↪toronto_denc_merged.columns[[1] + list(range(5, toronto_denc_merged.
↪shape[1]))]]
```

[47]:
```
           Borough  Cluster Labels 1st Most Common Venue  \
0     East Toronto               2                 Trail
2     East Toronto               2        Sandwich Place
4  Central Toronto               2                  Park
5  Central Toronto               2                   Gym

  2nd Most Common Venue 3rd Most Common Venue 4th Most Common Venue  \
0           Pizza Place     Health Food Store                   Pub
2                  Park                   Gym                   Pub
4              Bus Line           Swim School         Women's Store
5                 Hotel     Convenience Store      Department Store

  5th Most Common Venue 6th Most Common Venue 7th Most Common Venue  \
0               Dog Run    Dim Sum Restaurant                 Diner
2         Burrito Place  Fast Food Restaurant    Italian Restaurant
4    Dim Sum Restaurant  Ethiopian Restaurant     Electronics Store
5        Sandwich Place               Dog Run        Breakfast Spot

         8th Most Common Venue 9th Most Common Venue 10th Most Common Venue
0               Discount Store   Distribution Center          Women's Store
2            Fish & Chips Shop            Steakhouse      Sushi Restaurant
4  Eastern European Restaurant  Dumpling Restaurant             Donut Shop
5            Food & Drink Shop                  Park            Gas Station
```

Cluster 3

[48]:
```
toronto_denc_merged.loc[toronto_denc_merged['Cluster Labels'] == 3,␣
↪toronto_denc_merged.columns[[1] + list(range(5, toronto_denc_merged.
↪shape[1]))]]
```

[48]:
```
            Borough  Cluster Labels 1st Most Common Venue  \
22  Central Toronto               3                  Pool

   2nd Most Common Venue 3rd Most Common Venue 4th Most Common Venue  \
22                Garden        Ice Cream Shop         Women's Store

   5th Most Common Venue 6th Most Common Venue 7th Most Common Venue  \
22          Dessert Shop  Ethiopian Restaurant     Electronics Store

   8th Most Common Venue 9th Most Common Venue 10th Most Common Venue
```

```
22  Eastern European Restaurant   Dumpling Restaurant             Donut Shop
```

Cluster 4

[49]:
```
toronto_denc_merged.loc[toronto_denc_merged['Cluster Labels'] == 4,␣
↪toronto_denc_merged.columns[[1] + list(range(5, toronto_denc_merged.
↪shape[1]))]]
```

[49]:
```
           Borough  Cluster Labels 1st Most Common Venue  \
8  Central Toronto               4            Playground

  2nd Most Common Venue 3rd Most Common Venue 4th Most Common Venue  \
8          Tennis Court         Women's Store      Department Store

  5th Most Common Venue 6th Most Common Venue        7th Most Common Venue  \
8  Ethiopian Restaurant     Electronics Store  Eastern European Restaurant

  8th Most Common Venue 9th Most Common Venue 10th Most Common Venue
8   Dumpling Restaurant            Donut Shop       Doner Restaurant
```

1.4 End of Coursera Capstone Project: Segmenting and Clustering Neighborhoods in Toronto

1.5 Thank you ! Evaluator

Kapil Kumar Nagwanshi