Clustering Neighbourhoods to make Business Decisions

Clustering of Toronto Neighbourhoods

Shardul Kavale - June 11, 2020

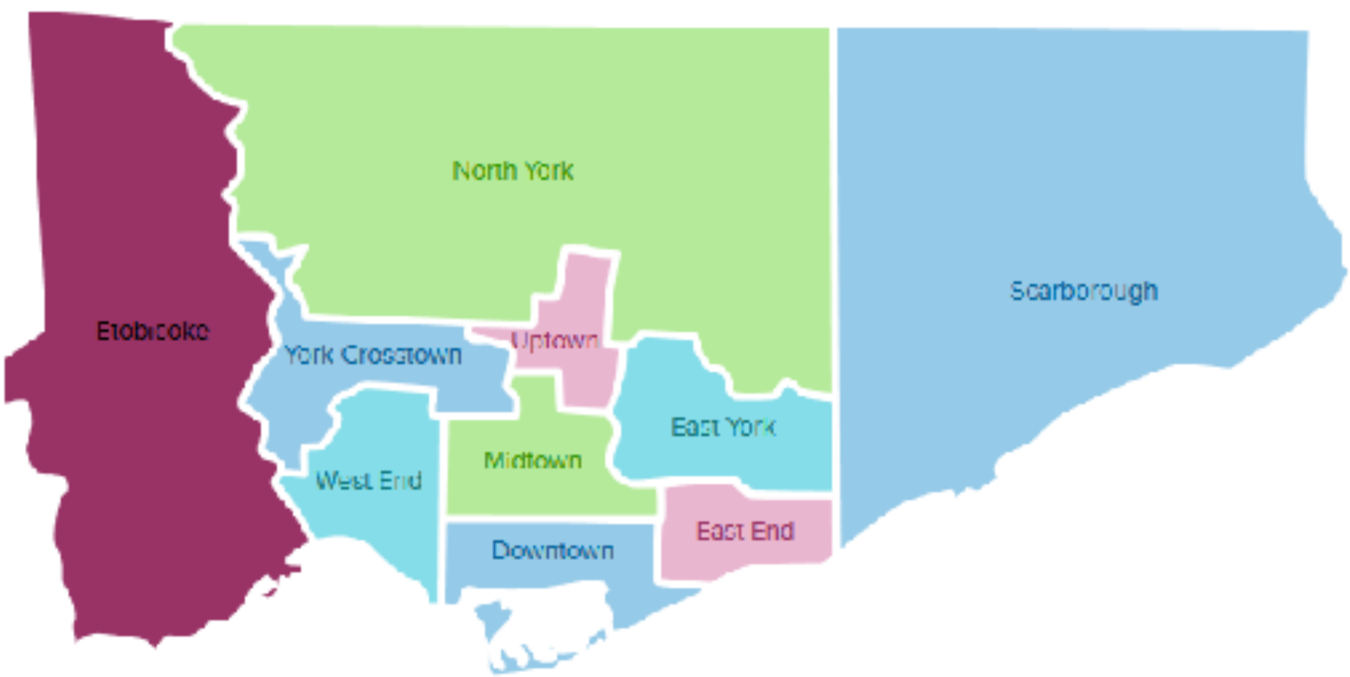
Introduction

Different neighbourhoods and different cities are a house to unique venues that define the culture of those places. Some cities and neighbourhoods might be similar in terms of the kind of venues they house. What if we could group these neighbourhoods based on the venues? How would it help investors/ Businesses to make decisions?

Problem

Clustering and Grouping similar neighbourhoods, which will provide insights into what kind of venues they have, thereby aiding potential business owners determine the most feasible location to open their business in.

Eg: *North York has fewer cafes, thus opening a cafe here would prove to be a wise investment due to minimal competition*



Data sources and cleaning

The project works with the data of Torontos's different Boroughs and their postal codes. This table can be found on wikipedia.

Link: https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M

This data table was web scraped using pandas library. After dropping not assigned values a data frame like below was obtained:

	PostalCode	Borough	Neighborhood
0	M5G	Downtown Toronto	Central Bay Street
1	M2H	North York	Hillcrest Village
2	M4B	East York	Parkview Hill, Woodbine Gardens
3	M1J	Scarborough	Scarborough Village
4	M4G	East York	Leaside
5	M4M	East Toronto	Studio District
6	M1R	Scarborough	Wexford, Maryvale
7	M9V	Etobicoke	South Steeles, Silverstone, Humbergate, Jamest...
8	M9L	North York	Humber Summit
9	M5V	Downtown Toronto	CN Tower, King and Spadina, Railway Lands, Har...
10	M1B	Scarborough	Malvern, Rouge
11	M5A	Downtown Toronto	Regent Park, Harbourfront

Nominatim Package from geopy.geocoders was used to find the approximate latitudes and longitudes of the neighbourhoods in the city. A table like below was obtained:

minutes.

In [17]:
```
for nd in df1['Neighborhood']:
    address=nd.split(',')[0]+',Toronto'
    geolocator = Nominatim(user_agent="toronto")
    location = geolocator.geocode(address)
    df1.loc[df1['Neighborhood'].str.contains(nd.split(',')[0]),'latitude']=location.latitud
    df1.loc[df1['Neighborhood'].str.contains(nd.split(',')[0]),'longitude']=location.longit
```

In [18]:
```
df1=df1.drop('Unnamed: 0',axis=1)
```

In [19]:
```
df1.tail()
```

Out[19]:

	Postal Code	Borough	Neighborhood	latitude	longitude
94	M1N	Scarborough	Birch Cliff	43.691805	-79.264484
95	M5R	Central Toronto	Davenport	43.671545	-79.448322
96	M6N	York	York South-Weston	43.694486	-79.498816
97	M3B	North York	Edwards Gardens	43.731442	-79.358380
98	M4C	East York	Toronto Danforth	43.678944	-79.344860

Foursquare API has been used to find venues in each Neighbourhood. The API call is being made in the function besides.

```
def getNearbyVenues(names, latitudes, longitudes, radius=1000):

    venues_list=[]
    for name, lat, lng in zip(names, latitudes, longitudes):

        url = 'https://api.foursquare.com/v2/venues/explore?&client_id
            CLIENT_ID,
            CLIENT_SECRET,
            VERSION,
            lat,
            lng,
            radius,
            Limit)

        results = requests.get(url).json()["response"]['groups'][0]['i
        for v in results:
            venues_list.append([
            name,
            lat,
            lng,
            v['venue']['name'],
            v['venue']['location']['lat'],
            v['venue']['location']['lng'],
            v['venue']['categories'][0]['name']])
    return venues_list
```

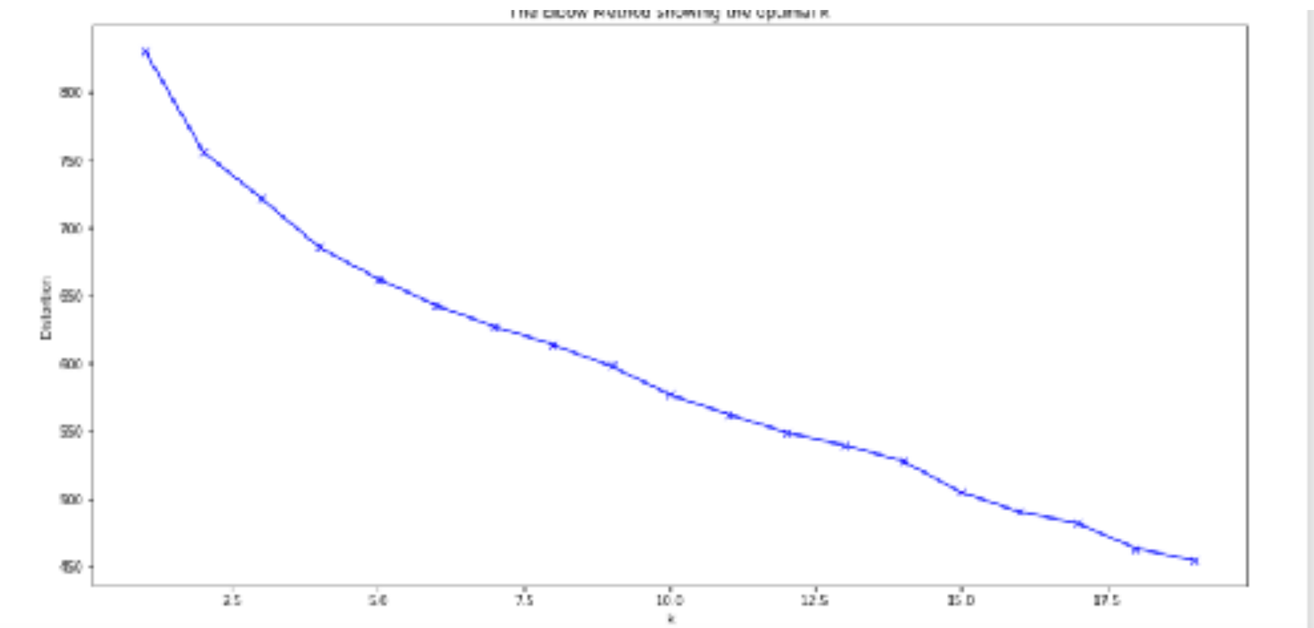
The list obtained from the above function is then converted into a Data frame, grouped by neighbourhood and consequently one-hot encoded. This resulting DataFrame is then used to find top 10 most popular venues and converted into a df like below.

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
0	Willowdale East	Coffee Shop	Bank	Japanese Restaurant	Grocery Store	Sandwich Place	Fast Food Restaurant	Ramen Restaurant	Pizza Place	Middle Eastern Restaurant	Sushi Restaurant
1	Agincourt	Chinese Restaurant	Shopping Mall	Cantonese Restaurant	Restaurant	Hotel	Noodle House	Korean Restaurant	Karaoke Bar	Gas Station	Bank
2	Ancaster	Park	Turkish Restaurant	Coffee Shop	Latin American Restaurant	Electronics Store	Italian Restaurant	Chinese Restaurant	Other Repair Shop	Intersection	Café
3	Bathurst Manor, Wilson Heights, Downsview North	Athletics & Sports	Gym / Fitness Center	Mediterranean Restaurant	Sandwich Place	Shoe Store	French Restaurant	Recreation Center	Basketball Court	Coffee Shop	Gas Station
4	Bayview Village	Furniture / Home Store	Café	Coffee Shop	Clothing Store	Bank	Moving Target	Chinese Restaurant	Liquor Store	Fast Food Restaurant	Spa

Clustering Methodology

Kmeans Clustering was used.

In order to determine the number of Clusters, several methods like Elbow method, Silhouette Score, Calinski Harabasz Score, Davies Bouldin Score were used:




```
For n_clusters = 2, silhouette score is 0.16583825540397104)

For n_clusters = 2, calinski harabasz score is 9.505683837331967)

For n_clusters = 2, DB score is 2.547694368883118)

For n_clusters = 3, silhouette score is 0.08359924375283861)

For n_clusters = 3, calinski harabasz score is 6.990768821395474)

For n_clusters = 3, DB score is 3.513492031304846)

For n_clusters = 4, silhouette score is 0.09755031084888124)

For n_clusters = 4, calinski harabasz score is 6.579046572144231 5)

For n_clusters = 4, DB score is 2.898054088737 9833)
```

Clustering Analysis

Some of the clusters were analyzed to find trends.

Eg Cluster 0 Analysis:

```
In [77]:  s=cluster0.apply(lambda x: x.mode()).iloc[0,6:11]
          s

Out[77]:  1st Most Common Venue              Coffee Shop
          2nd Most Common Venue                     Café
          3rd Most Common Venue               Restaurant
          4th Most Common Venue      Japanese Restaurant
          5th Most Common Venue               Restaurant
          Name: 0, dtype: object
```

Cluster 0 seems to be a home to a number of restarants

In [95]:
```
cluster0['3rd Most Common Venue'].value_counts()[:5]
```

Out[95]:
```
Restaurant              9
Café                    5
Fast Food Restaurant    3
Park                    2
Hotel                   2
Name: 3rd Most Common Venue, dtype: int64
```

In [98]:
```
cluster0['2nd Most Common Venue'].value_counts()[:5]
```

Out[98]:
```
Café              11
Coffee Shop        6
Bar                3
Hotel              3
Clothing Store     2
Name: 2nd Most Common Venue, dtype: int64
```

In [96]:
```
cluster0['4th Most Common Venue'].value_counts()[:5]
```

Out[96]:
```
Japanese Restaurant    5
Gastropub              3
Bar                    2
Park                   2
Baseball Field         2
Name: 4th Most Common Venue, dtype: int64
```

Cluster 0 Conclusion

Cluster 0 is defined by the amount of Restaurants, Coffee shops, Fast food and Bars. Almost all of Downtown Toronto Neighborhoods belong to this cluster, which makes sense.

Making Business Decisions

USER PERSONA:	
Bob Kenney Director, xyz	Bob owns a chain of Breweries and Bars. He wishes to expand his chain and open a few lounges in Toronto City. He wants to study the city neighborhoods and decide on the best locations to open his businesses in.

Approach:

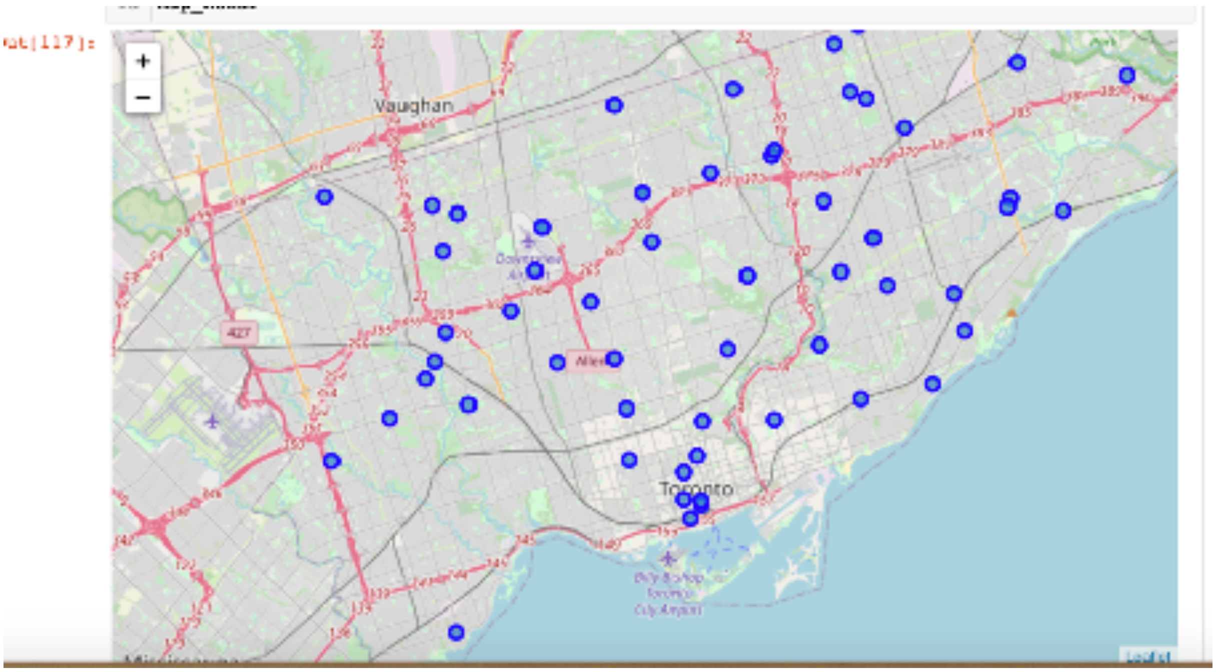
Defined a function and used it on Toronto Neighbourhoods Data and used it on toronto DataFrame.

```
[113]:
#function to return rows(neighborhoods) that have no bars/pubs/breweries near them
def noBar(d):

    noBarHere=d[~d['1st Most Common Venue'].isin(['Brewery','Pub','Bar','Gastropub','Beer Bar'])&
                ~d['2nd Most Common Venue'].isin(['Brewery','Pub','Bar','Gastropub','Beer Bar'])&
                ~d['3rd Most Common Venue'].isin(['Brewery','Pub','Bar','Gastropub','Beer Bar'])&
                ~d['4th Most Common Venue'].isin(['Brewery','Pub','Bar','Gastropub','Beer Bar'])&
                ~d['5th Most Common Venue'].isin(['Brewery','Pub','Bar','Gastropub','Beer Bar'])&
                ~d['6th Most Common Venue'].isin(['Brewery','Pub','Bar','Gastropub','Beer Bar'])&
                ~d['7th Most Common Venue'].isin(['Brewery','Pub','Bar','Gastropub','Beer Bar'])&
                ~d['8th Most Common Venue'].isin(['Brewery','Pub','Bar','Gastropub','Beer Bar'])&
                ~d['9th Most Common Venue'].isin(['Brewery','Pub','Bar','Gastropub','Beer Bar'])&
                ~d['10th Most Common Venue'].isin(['Brewery','Pub','Bar','Gastropub','Beer Bar'])]
    return noBarHere
```

Neighbourhoods with no Bars/Pubs in top 10 venues:-

Any of these locations will be a good location to open up a bar/Pub since there would be minimal competitors



vs

Map of Toronto Neighbourhoods: