The Battle of Neighbourhoods

Contents

INTRODUCTION/BUSINESS PROBLEM

This project aims to help business owners in exploring suitable areas to open Italian Restaurants in Toronto Area. With the purpose in mind, finding the location to open such a restaurant is one of the most important decisions for any business owner and I am designing this project to help him find the most suitable location.

In this project we will try to find the best locations to open this Italian restaurant. We will use our data science powers to find a few most promising neighbourhoods where there are not many Italian Restaurants yet.

Target Audience will be the business owners who are planning to establish or extend business in Toronto to determine a strategic location before deciding on a location that will attract more customers.

DATA SELECTION

Following data sources will be used to get the required information:

1. Wikipedia will be used scrap Toronto neighbourhoods.
2. Geospatial_Coordinates.csv will be used to get Latitude and Longitude information.
3. Foursquare API will be used to get restaurants data related to Toronto neighbourhoods.

Above data sources will be used to get venues and Italian restaurants information to identify in which area has the most Italian restaurants and, this way, select the area with the least number of restaurants.

Data flow

1. First, it is used data from get city open data to get city information as well as latitude and longitude coordinates.
2. Then, we created a data frame with borough and neighbourhood information. For Toronto, it is used Wikipedia to get the list of Postal Code of all Neighbourhoods in Toronto.

3. List of Restaurants will be gathered using Foursquare. With this information it is possible to come up with a total as well as draw the maps with Italian restaurants locations.

METHODOLOGY

The goal of this project is to come up with a study to identify area's in the city of Toronto, where Italian Restaurants are located. So, we can define areas of opportunities to invest/start a new Italian Restaurant.

And finally, in the last part of this study, it is showed a map showing the spots where these Italian restaurants are located and helps us to visualize the areas of opportunity for our restaurant.

Libraries used in this project:

BeautifulSoup - for web scraping
Geocoder - for retrieval of location data
Numpy – for working with arrays
Pandas - for dataframe creation and manipulation
Folium - for visualisation of geospatial data
Scikit-learn - for usage of k-means clustering algorithm
Matplotlib - for visualisation of data
Json - for handling json format

ANALYSIS

First, we extract the geographic data of Toronto by webscrapping "https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M" and save it in dataframe.

In [2]:

```
#We will use BeautifulSoup to get the zip code information of Canada from Wikipedia
page = requests.get("https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M")
soup = BeautifulSoup(page.content, 'html.parser')
```

In [3]:

```
table_contents=[]
table=soup.find('table')
for row in table.findAll('td'):
    cell = {}
    if row.span.text=='Not assigned':
        pass
    else:
        cell['PostalCode'] = row.p.text[:3]
        cell['Borough'] = (row.span.text).split('(')[0]
        cell['Neighborhood'] = (((((row.span.text).split('(')[1]).strip(')')).replace('
/',',')).replace(')',' ')).strip(' ')
        table_contents.append(cell)
```


In [4]:

```
#We save this to dataframe (df)
df=pd.DataFrame(table_contents)
df['Borough']=df['Borough'].replace({'Downtown TorontoStn A PO Boxes25 The Esplanade':
'Downtown Toronto Stn A',
                                      'East TorontoBusiness reply mail Processin
g Centre969 Eastern':'East Toronto Business',
                                      'EtobicokeNorthwest':'Etobicoke Northwest'
,'East YorkEast Toronto':'East York/East Toronto',
                                      'MississaugaCanada Post Gateway Processing
Centre':'Mississauga'})
df
```

Then, We downloaded the geospatial coordinates data from "https://cocl.us/Geospatial_data/Geospatial_Coordinates.csv" and put it in the datafprame

In [7]:

```
#download Geospatial_Coordinates and put it in dataframe (temp_df)
URL = "https://cocl.us/Geospatial_data/Geospatial_Coordinates.csv"
temp_df = pd.read_csv("https://cocl.us/Geospatial_data/Geospatial_Coordinates.csv")

# show the first 5 rows
temp_df.head ()
```

Out[7]:

	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

Then, we merged the two dataframes on "PostalCode" and put it into single dataframe.

In [9]:

```
# Merge the 2 data sets (df and temp_df)
temp_df = pd.merge(df, temp_df, on='PostalCode')

#show the first 5 rows
temp_df.head(5)
```

Out[9]:

	PostalCode	Borough	Neighborhood	Latitude	Longitude
0	M3A	North York	Parkwoods	43.753259	-79.329656
1	M4A	North York	Victoria Village	43.725882	-79.315572
2	M5A	Downtown Toronto	Regent Park, Harbourfront	43.654260	-79.360636
3	M6A	North York	Lawrence Manor, Lawrence Heights	43.718518	-79.464763
4	M7A	Queen's Park	Ontario Provincial Government	43.662301	-79.389494

Now that we have our location candidates, let's use Foursquare API to get info on restaurants in each neighbourhood.
We're interested in venues in 'food' category, but only those that are proper restaurants - coffee shops, pizza places, bakeries etc. are not direct competitors so we don't care about those. So we will include in our list only venues that have 'restaurant' in category name, and we'll make sure to detect and include all the subcategories of specific 'Italian restaurant' category, as we need info on Italian restaurants in the neighbourhood.

In [12]:

```
# Lets get the venue data from foursquare
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)
```

Then, we filtered the venue to only Italian restaurants.

In [17]:

```
#create a new data frame with only the italian restaurants
Italian_Restaurants = to_grouped[["Neighborhoods","Italian Restaurant"]]

#show the first 5 rows
Italian_Restaurants.head()
```

Out[17]:

	Neighborhoods	Italian Restaurant
0	Agincourt	0.000000
1	Alderwood, Long Branch	0.000000
2	Bathurst Manor, Wilson Heights, Downsview North	0.000000
3	Bayview Village	0.000000
4	Bedford Park, Lawrence Manor East	0.090909

Now in this new dataset we want to determine clusters to see if we can find areas where there are not many restaurants yet. We will do this with a type of analysis called K-Means.

K-Means

K-means clustering is a type of unsupervised learning, which is used when you have unlabelled data (i.e., data without defined categories or groups). The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable K. The algorithm works iteratively to assign each data point to one of K groups based on the features that are provided. Data points are clustered into 3 clusters based on feature similarity.

In [18]:

```
# cluster the above dataset into 3 clusters.
toclusters = 3
to_clustering = Italian_Restaurants.drop(["Neighborhoods"], 1)
kmeans = KMeans(n_clusters=toclusters, random_state=1)
kmeans.fit_transform(to_clustering)
kmeans.labels_[0:20]
```

Out[18]:

```
array([1, 1, 1, 1, 0, 1, 1, 2, 1, 1, 1, 2, 0, 1, 1, 0, 1, 2, 0, 1])
```

In [19]:

```
#create dataset (to_merged)
to_merged = Italian_Restaurants.copy()

# add clustering labels
to_merged["Cluster Labels"] = kmeans.labels_
```

Then, we can see that clusters 0,1 and 2 are being created.

In [20]:

```
# Rename the columns
to_merged.rename(columns={"Neighborhoods": "Neighborhood"}, inplace=True)
to_merged.head(5)
```

Out[20]:

	Neighborhood	Italian Restaurant	Cluster Labels
0	Agincourt	0.000000	1
1	Alderwood, Long Branch	0.000000	1
2	Bathurst Manor, Wilson Heights, Downsview North	0.000000	1
3	Bayview Village	0.000000	1
4	Bedford Park, Lawrence Manor East	0.090909	0

In [21]:

Then, we combined this with the previous dataset to get one total data set.

In [22]:

```
#Combine the sets and set index
to_merged = to_merged.join(toronto_venues.set_index("Neighborhood"), on="Neighborhood")

print(to_merged.shape)
to_merged.head()
```

```
(2104, 9)
```

Out[22]:

	Neighborhood	Italian Restaurant	Cluster Labels	Neighborhood Latitude	Neighborhood Longitude	Venue	Venu Latituc
84	The Danforth West, Riverdale	0.071429	0	43.679557	-79.352188	MenEssentials	43.67782
84	The Danforth West, Riverdale	0.071429	0	43.679557	-79.352188	Pantheon	43.67762
84	The Danforth West, Riverdale	0.071429	0	43.679557	-79.352188	Cafe Fiorentina	43.67774
84	The Danforth West, Riverdale	0.071429	0	43.679557	-79.352188	La Diperie	43.67770
84	The Danforth West, Riverdale	0.071429	0	43.679557	-79.352188	Dolce Gelato	43.67777

RESULT

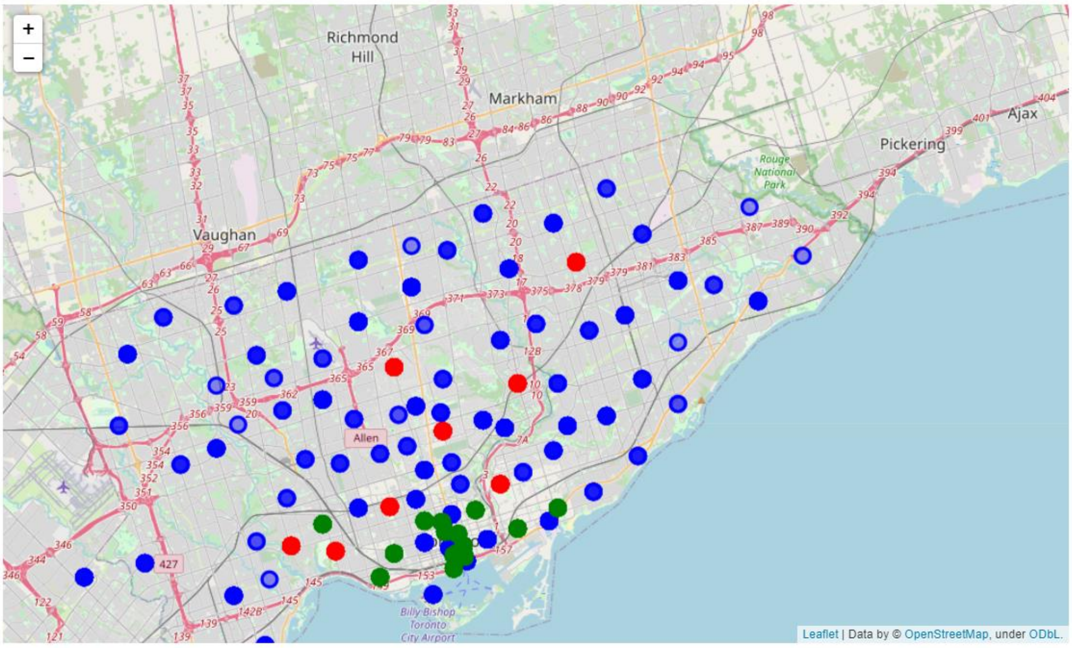
Now that we have create the clusters with K-means we want first find out in which cluster are the least number of Italian restaurants. So, we know where to invest. First let's visualize our findings

Cluster 0 = Red

Cluster 1 = Blue

Cluster 2 = Green

Out[24]:



CONCLUSION

Most of the Italian restaurants are in cluster 1 and are lowest in Cluster 0. Looking at nearby venues it seems cluster 0 might be a good location as there are not a lot of Italian restaurants in these areas. We therefore recommend the Business owners to open an authentic Italian restaurant in these locations. If we look to the total map of all the areas. We might want to explorer the areas close to the blue and green areas first because there are likely to be more downtown.