Capstone Project - The Battle of the Neighborhoods (Week 2)

THU LE

Table of contents

Introduction: Business Problem

Toronto is the city of food lovers, offering a huge variety of cuisines with high standards. My client is interested in promoting Vietnamese cuisine in Toronto and is looking for a good neighborhood to start their business. They have not decided which borough in Toronto to take a deeper look for the neighborhood. They would like to be given a big picture of how dynamic these boroughs are compared to each other, based on some basic criteria of venue categories. Then after they pick the borough, they would like to look at how neighborhoods in the borough are doing in restaurant categories.

Methodology

Stage 1: List top 5 boroughs in Toronto with the consideration of their venue dynamic\

```
- number\
- variety\
- popularity in restaurant/bar/entertainment)\
```

Stage 2: After the desired borough is picked by the customer, present the venue analysis on each neighborhood

Data

- Toronto Neighborhoods with Geospatial Coordinates
- Foursquare to retrieve trending venues nearby

Importing libraries

In [ ]:
```
import pandas as pd
from pandas.io.json import json_normalize
import numpy as np
```

```
import json
import matplotlib.cm as cm
import matplotlib.colors as colors
from geopy.geocoders import Nominatim
from bs4 import BeautifulSoup
import requests
```

Analysis

Stage 1: List top 5 boroughs in Toronto with their venue dynamic

I. Prepare dataframe of boroughs and neigborhoods in Toronto with geospatial coordinates

In [ ]:

```
# Create soup object storing parsed data from web
url ='https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M'
data = requests.get(url).text
soup = BeautifulSoup(data,'html.parser')

# Data preparation and cleaning
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('
        table_contents.append(cell)
df=pd.DataFrame(table_contents)
df['Borough']=df['Borough'].replace({'Downtown TorontoStn A PO Boxes25 The Esplanade':'
                                      'East TorontoBusiness reply mail Processin
                                      'EtobicokeNorthwest':'Etobicoke Northwest'
                                      'MississaugaCanada Post Gateway Processing

# Add latitude and longitude to corresponding postal code
geo_coords = pd.read_csv(r'C:\Users\user\Downloads\Geospatial_Coordinates.csv')
geo_coords
df=df.merge(geo_coords, left_on ='PostalCode', right_on ='Postal Code').drop('Postal Co
df
```

In [ ]:

```
print('The dataframe has {} boroughs and {} neighborhoods'.format(len(df['Borough'].uni
```

II. Prepare dataframe of boroughs and their venues dynamic

In [ ]:

```
# Foursquare credentials
CLIENT_ID = 'E3ECED54X54UQ2KZPLNAKXJGRUYU0SAFICERMC1EG0O4OWIV'
CLIENT_SECRET = 'QMVFLY2DNZQO5HTOEURTVPDTUHY2TQMLX1J3KEEBGK3QYRAL'
VERSION = '20190505'
LIMIT = 100
```

In [ ]:

```
# Create get venues nearby function
def getNearbyVenues( neigborhood, latitudes, longitudes, radius=500):
```

```
    venues_list=[]
    for neigborhood, lat, lng in zip( neigborhood, latitudes, longitudes):
        #print(name)

        # create the API request URL
        url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret
            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([(
            neigborhood,
            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_
    nearby_venues.columns = ['Neighborhood',
                  'Neighborhood Latitude',
                  'Neighborhood Longitude',
                  'Venue',
                  'Venue Latitude',
                  'Venue Longitude',
                  'Venue Category']

    return(nearby_venues)
```

In [ ]:

```
#Run the function to have dataframe of venues in radius of 500 for each boroughs in dow
total_venues = getNearbyVenues(df['Neighborhood'], df['Latitude'], df['Longitude'])
total_venues.head()
```

In [ ]:

```
print(total_venues.shape)
total_venues.head()
```

In [ ]:

```
# Add borough back for further analysis on borough
df_to_merge = df[['Borough','Neighborhood']]
total_venues_borough = total_venues.merge(df_to_merge, on = 'Neighborhood')
total_venues_borough.head()
total_venues_borough.groupby('Borough').count()
```

In [197…

```
# one hot encoding
borough_onehot = pd.get_dummies(total_venues_borough[['Venue Category']], prefix="", pr
```

```python
# add borough column back to dataframe
borough_onehot['Borough'] = total_venues_borough['Borough']

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

borough_onehot.head()
```

Out[197…

	Borough	Accessories Store	Adult Boutique	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airport Terminal	American Restaurant
0	North York	0	0	0	0	0	0	0	0	0
1	North York	0	0	0	0	0	0	0	0	0
2	North York	0	0	0	0	0	0	0	0	0
3	North York	0	0	0	0	0	0	0	0	0
4	North York	0	0	0	0	0	0	0	0	0

5 rows × 270 columns

In [198…

```python
borough_grouped = borough_onehot.groupby('Borough').mean().reset_index()
borough_grouped.head()
```

Out[198…

	Borough	Accessories Store	Adult Boutique	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airport Terminal	Amer Restau
0	Central Toronto	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.009
1	Downtown Toronto	0.0	0.000909	0.000909	0.000909	0.000909	0.001818	0.002727	0.001818	0.01
2	Downtown Toronto Stn A	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00
3	East Toronto	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.02
4	East Toronto Business	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.00

5 rows × 270 columns

In [207…

```
# Add number of venues to dataframe
borough_grouped_venues_sum = borough_onehot.groupby('Borough').sum().reset_index()
borough_grouped_venues_sum['Total venues'] =borough_grouped_venues_count.sum(axis=1)
borough_grouped_venues_sum = borough_grouped_venues_sum[['Borough','Total venues']]
borough_grouped_venues_sum.head()
borough_grouped= borough_grouped.merge(borough_grouped_venues_sum, on ='Borough')
```

In [209…

```
# Sort boroughs from the one with highest number of venues
borough_grouped= borough_grouped.sort_values('Total venues', ascending = False )
borough_grouped.head()
```

Out[209…

	Borough	Accessories Store	Adult Boutique	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airport Terminal	Ame Resta
1	Downtown Toronto	0.00000	0.000909	0.000909	0.000909	0.000909	0.001818	0.002727	0.001818	0.0
10	North York	0.00431	0.000000	0.004310	0.000000	0.000000	0.000000	0.000000	0.000000	0.0(
13	West Toronto	0.00000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0(
0	Central Toronto	0.00000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0(
3	East Toronto	0.00000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0:

5 rows × 271 columns

In [211…

```
num_top_venues = 10

for bo in borough_grouped['Borough']:
    print("----"+bo+"----")

    temp = borough_grouped[borough_grouped['Borough'] == bo].T.reset_index()
    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).head(num_top_
    print('\n')
```

```
----Downtown Toronto----
                venue     freq
0        Total venues  2200.00
1         Coffee Shop     0.10
2                Café     0.06
3               Hotel     0.03
4  Japanese Restaurant     0.03
5          Restaurant     0.03
6                Park     0.02
7       Clothing Store     0.02
8   Seafood Restaurant     0.02
9              Bakery     0.02
```

```
----North York----
                 venue    freq
0          Total venues  464.00
1           Coffee Shop    0.08
2        Clothing Store    0.05
3            Restaurant    0.04
4           Pizza Place    0.04
5  Fast Food Restaurant    0.03
6                  Café    0.03
7                  Bank    0.03
8                  Park    0.03
9         Grocery Store    0.03


----West Toronto----
                 venue    freq
0         Total venues  310.00
1                 Café    0.07
2                  Bar    0.06
3          Coffee Shop    0.05
4   Italian Restaurant    0.04
5           Restaurant    0.04
6       Breakfast Spot    0.03
7               Bakery    0.03
8        Grocery Store    0.02
9            Bookstore    0.02


----Central Toronto----
              venue    freq
0      Total venues  210.00
1       Coffee Shop    0.08
2    Sandwich Place    0.07
3              Park    0.06
4       Pizza Place    0.06
5              Café    0.06
6   Sushi Restaurant   0.04
7        Restaurant    0.04
8               Gym    0.03
9      Dessert Shop    0.03


----East Toronto----
                 venue    freq
0         Total venues  206.00
1     Greek Restaurant    0.08
2          Coffee Shop    0.07
3   Italian Restaurant    0.05
4       Ice Cream Shop    0.04
5                 Park    0.04
6              Brewery    0.04
7               Bakery    0.03
8                 Café    0.03
9                  Pub    0.03


----Downtown Toronto Stn A----
                 venue    freq
0         Total venues  200.00
1          Coffee Shop    0.12
2   Seafood Restaurant    0.04
3         Cocktail Bar    0.04
4               Bakery    0.03
5                 Café    0.03
```

```
6          Restaurant    0.03
7               Hotel    0.03
8  Japanese Restaurant   0.03
9   Italian Restaurant   0.03


----Scarborough----
                  venue    freq
0          Total venues  184.00
1  Fast Food Restaurant    0.05
2           Coffee Shop    0.05
3                Bakery    0.04
4                  Bank    0.04
5           Pizza Place    0.03
6          Intersection    0.03
7        Breakfast Spot    0.03
8    Chinese Restaurant    0.03
9   Fried Chicken Joint    0.02


----East York----
                    venue    freq
0            Total venues  140.00
1             Coffee Shop    0.06
2                    Bank    0.06
3            Intersection    0.04
4            Burger Joint    0.04
5  Furniture / Home Store    0.04
6             Pizza Place    0.04
7             Supermarket    0.03
8     Sporting Goods Shop    0.03
9          Sandwich Place    0.03


----Etobicoke----
                  venue    freq
0          Total venues  138.00
1           Pizza Place    0.12
2        Sandwich Place    0.07
3              Pharmacy    0.04
4                   Gym    0.04
5         Grocery Store    0.04
6                Bakery    0.04
7  Fast Food Restaurant    0.04
8           Coffee Shop    0.04
9                  Café    0.04


----Queen's Park----
                 venue   freq
0         Total venues  62.00
1          Coffee Shop   0.19
2     Sushi Restaurant   0.06
3       Sandwich Place   0.03
4             Beer Bar   0.03
5  Fried Chicken Joint   0.03
6   Mexican Restaurant   0.03
7        Smoothie Shop   0.03
8        Burrito Place   0.03
9                 Café   0.03


----East Toronto Business----
                  venue   freq
0          Total venues  34.00
```

```
1         Auto Workshop   0.06
2           Pizza Place   0.06
3            Comic Shop   0.06
4            Restaurant   0.06
5               Butcher   0.06
6        Farmers Market   0.06
7            Skate Park   0.06
8         Burrito Place   0.06
9   Fast Food Restaurant   0.06


----York----
               venue   freq
0       Total venues  32.00
1               Park   0.19
2  Convenience Store   0.12
3              Trail   0.06
4       Hockey Arena   0.06
5      Grocery Store   0.06
6              Field   0.06
7     Discount Store   0.06
8               Pool   0.06
9        Coffee Shop   0.06


----Mississauga----
                       venue   freq
0               Total venues  26.00
1                Coffee Shop   0.23
2                      Hotel   0.15
3             Sandwich Place   0.08
4        Fried Chicken Joint   0.08
5  Middle Eastern Restaurant   0.08
6                Gas Station   0.08
7   Mediterranean Restaurant   0.08
8        American Restaurant   0.08
9              Burrito Place   0.08


----Etobicoke Northwest----
                             venue  freq
0                     Total venues  10.0
1                              Bar   0.2
2              Rental Car Location   0.2
3                        Drugstore   0.2
4                       Truck Stop   0.2
5                    Garden Center   0.2
6       Modern European Restaurant   0.0
7              Moroccan Restaurant   0.0
8              Monument / Landmark   0.0
9  Molecular Gastronomy Restaurant   0.0


----East York/East Toronto----
                             venue  freq
0                     Total venues  6.00
1                     Intersection  0.33
2                             Park  0.33
3                Convenience Store  0.33
4               Miscellaneous Shop  0.00
5              Moroccan Restaurant  0.00
6              Monument / Landmark  0.00
7  Molecular Gastronomy Restaurant  0.00
8       Modern European Restaurant  0.00
9                Mobile Phone Shop  0.00
```

Stage 1 Conclusion

Client pick Downtown Toronto and West Toronto for deeper neighboorhood exploration as they has highest trafic of venues. Downtown Toronto has many Japanese restaurant, showing people's interest in Asian style cuisine Although West Toronto is not as crowded as in Downtown Toronto, top 5 most common venues are all in food/drink category. And, the cost of business could be lower if the client decide to go small with this restaurant later.

Stage 2: Present venues dynamics for neighborhoods in Downtown Toronto and West Toronto

In [232…

```
downtown_venues = total_venues_borough[total_venues_borough.Borough.isin( ['Downtown To
west_venues = total_venues_borough[total_venues_borough.Borough.isin( ['West Toronto'])
```

Downtown Toronto

In [233…

```
downtown_onehot = pd.get_dummies(downtown_venues[['Venue Category']], prefix="", prefix
downtown_onehot['Neighborhood'] = downtown_venues['Neighborhood']
downtown_grouped = downtown_onehot.groupby('Neighborhood').mean().reset_index()
downtown_grouped
```

Out[233…

	Neighborhood	Adult Boutique	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airport Terminal	American Restaurant	Ar
0	Berczy Park	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.0
1	CN Tower, King and Spadina, Railway Lands, Har...	0.000000	0.066667	0.066667	0.066667	0.133333	0.2	0.133333	0.000000	0.0
2	Central Bay Street	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.0
3	Christie	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.0
4	Church and Wellesley	0.013158	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.013158	0.0
5	Commerce Court, Victoria Hotel	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.030000	0.0
6	First Canadian Place, Underground city	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.030000	0.0
7	Garden District, Ryerson	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.0

	Neighborhood	Adult Boutique	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airport Terminal	American Restaurant	Ar
8	Harbourfront East, Union Station, Toronto Islands	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.0
9	Kensington Market, Chinatown, Grange Park	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.0
10	Regent Park, Harbourfront	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.0
11	Richmond, Adelaide, King	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.020833	0.0
12	Rosedale	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.0
13	St. James Town	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.023810	0.0
14	St. James Town, Cabbagetown	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.0
15	Toronto Dominion Centre, Design Exchange	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.020000	0.0
16	University of Toronto, Harbord	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.0

17 rows × 200 columns

In [260…

```
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'] = downtown_grouped['Neighborhood']
```

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

neighborhoods_venues_sorted
```

Out[260…

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue
0	Berczy Park	Coffee Shop	Cocktail Bar	Bakery	Beer Bar	Restaurant	Pharmacy	Cheese Shop
1	CN Tower, King and Spadina, Railway Lands, Har...	Airport Service	Airport Lounge	Airport Terminal	Coffee Shop	Harbor / Marina	Rental Car Location	Sculpture Garden
2	Central Bay Street	Coffee Shop	Sandwich Place	Italian Restaurant	Café	Burger Joint	Bubble Tea Shop	Salad Place
3	Christie	Grocery Store	Café	Park	Athletics & Sports	Italian Restaurant	Baby Store	Candy Store
4	Church and Wellesley	Coffee Shop	Sushi Restaurant	Japanese Restaurant	Restaurant	Gay Bar	Fast Food Restaurant	Hotel
5	Commerce Court, Victoria Hotel	Coffee Shop	Restaurant	Café	Hotel	Gym	Italian Restaurant	Cocktail Bar
6	First Canadian Place, Underground city	Coffee Shop	Café	Hotel	Gym	Japanese Restaurant	Restaurant	Asian Restaurant
7	Garden District, Ryerson	Clothing Store	Coffee Shop	Cosmetics Shop	Bubble Tea Shop	Japanese Restaurant	Middle Eastern Restaurant	Café
8	Harbourfront East, Union Station, Toronto Islands	Coffee Shop	Aquarium	Café	Hotel	Brewery	Scenic Lookout	Sporting Goods Shop
9	Kensington Market, Chinatown, Grange Park	Café	Coffee Shop	Vietnamese Restaurant	Vegetarian / Vegan Restaurant	Bar	Gaming Cafe	Mexican Restaurant
10	Regent Park, Harbourfront	Coffee Shop	Park	Bakery	Pub	Café	Theater	Breakfast Spot
11	Richmond, Adelaide, King	Coffee Shop	Café	Restaurant	Thai Restaurant	Deli / Bodega	Clothing Store	Gym
12	Rosedale	Park	Playground	Trail	Adult Boutique	Museum	Mediterranean Restaurant	Men's Store
13	St. James Town	Coffee Shop	Café	Gastropub	Cosmetics Shop	Cocktail Bar	Lingerie Store	Park

	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
14	St. James Town, Cabbagetown	Coffee Shop	Café	Chinese Restaurant	Bakery	Italian Restaurant	Restaurant	Pub
15	Toronto Dominion Centre, Design Exchange	Coffee Shop	Hotel	Café	Japanese Restaurant	Italian Restaurant	Seafood Restaurant	Salad Place
16	University of Toronto, Harbord	Café	Bakery	Bar	Japanese Restaurant	Bookstore	College Arts Building	Beer Bar

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

In [265…

```
from sklearn.cluster import KMeans
# set number of clusters
kclusters = 5

downtown_grouped_clustering = downtown_grouped.drop('Neighborhood', 1)

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

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

Out[265…

```
array([0, 3, 0, 2, 0, 0, 0, 0, 0, 0])
```

In [266…

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

downtown_merged = downtown

# merge manhattan_grouped with manhattan_data to add latitude/longitude for each neighb
downtown_merged = downtown_merged.join(neighborhoods_venues_sorted.set_index('Neighborh
```

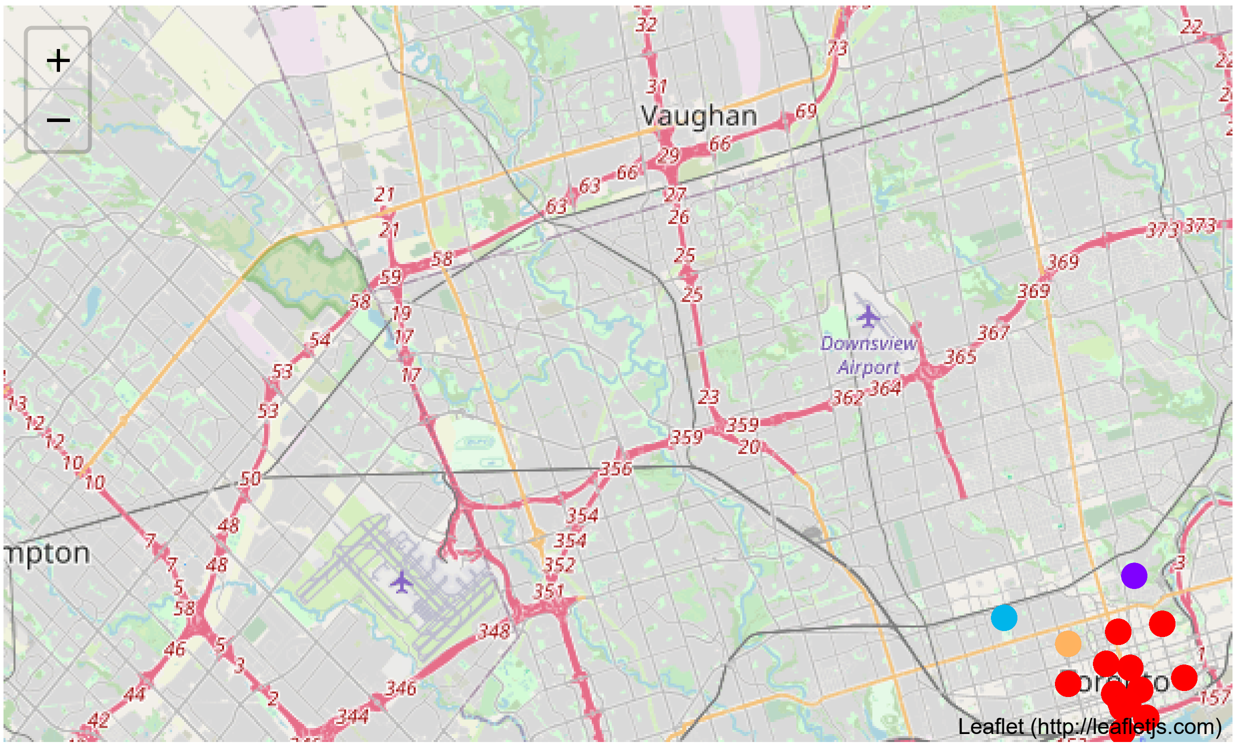
Visualize clusters

In [242…

```
address = 'Downtown Toronto, Toronto'

geolocator = Nominatim(user_agent="ny_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude

import folium
# 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(downtown_merged['Neighborhood Latitude'], downtown_me
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7).add_to(map_clusters)

map_clusters
```

Out[242…



West Toronto

In [254…

```
west_onehot = pd.get_dummies(west_venues[['Venue Category']], prefix="", prefix_sep="")
west_onehot['Neighborhood'] = west_venues['Neighborhood']
west_grouped = west_onehot.groupby('Neighborhood').mean().reset_index()
west_grouped
```

Out[254…

	Neighborhood	Antique Shop	Art Gallery	Arts & Crafts Store	Asian Restaurant	Bakery	Bank	Bar	Beer Store	Bookst
0	Brockton, Parkdale Village, Exhibition Place	0.00	0.000000	0.00	0.000000	0.086957	0.000000	0.043478	0.000000	0.000

	Neighborhood	Antique Shop	Art Gallery	Arts & Crafts Store	Asian Restaurant	Bakery	Bank	Bar	Beer Store	Bookst
1	Dufferin, Dovercourt Village	0.00	0.000000	0.00	0.000000	0.133333	0.066667	0.066667	0.000000	0.000
2	High Park, The Junction South	0.04	0.000000	0.04	0.000000	0.040000	0.000000	0.040000	0.000000	0.040
3	Little Portugal, Trinity	0.00	0.023256	0.00	0.046512	0.000000	0.000000	0.093023	0.023256	0.000
4	Parkdale, Roncesvalles	0.00	0.000000	0.00	0.000000	0.000000	0.000000	0.071429	0.000000	0.071
5	Runnymede, Swansea	0.00	0.000000	0.00	0.000000	0.000000	0.028571	0.028571	0.000000	0.028

6 rows × 82 columns

In [259…

```
# 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'] = west_grouped['Neighborhood']

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

neighborhoods_venues_sorted
```

Out[259…

	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 M Comr Ve
0	Brockton, Parkdale Village, Exhibition Place	Café	Bakery	Breakfast Spot	Coffee Shop	Gym	Intersection	Grocery Store	Furn / H S
1	Dufferin, Dovercourt Village	Pharmacy	Bakery	Liquor Store	Park	Music Venue	Middle Eastern Restaurant	Furniture / Home Store	
2	High Park, The Junction South	Mexican Restaurant	Thai Restaurant	Café	Antique Shop	Speakeasy	Italian Restaurant	Grocery Store	M Ve

	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 M Comi Ve
3	Little Portugal, Trinity	Bar	Restaurant	Café	Men's Store	Asian Restaurant	Vietnamese Restaurant	Vegetarian / Vegan Restaurant	Co S
4	Parkdale, Roncesvalles	Gift Shop	Breakfast Spot	Dog Run	Cuban Restaurant	Coffee Shop	Movie Theater	Eastern European Restaurant	Books
5	Runnymede, Swansea	Café	Sushi Restaurant	Coffee Shop	Pub	Pizza Place	Italian Restaurant	Gym	He F S

In [256…

```
from sklearn.cluster import KMeans
# set number of clusters
kclusters = 3

west_grouped_clustering = west_grouped.drop('Neighborhood', 1)

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

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

Out[256…

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

In [257…

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

west_merged = west_venues

# merge manhattan_grouped with manhattan_data to add latitude/longitude for each neighb
west_merged = west_merged.join(neighborhoods_venues_sorted.set_index('Neighborhood'), o
west_merged
```

Out[257…

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	Borou
685	Dufferin, Dovercourt Village	43.669005	-79.442259	The Greater Good Bar	43.669409	-79.439267	Bar	W Toro
686	Dufferin, Dovercourt Village	43.669005	-79.442259	Parallel	43.669516	-79.438728	Middle Eastern Restaurant	W Toro
687	Dufferin, Dovercourt Village	43.669005	-79.442259	Happy Bakery & Pastries	43.667050	-79.441791	Bakery	W Toro

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	Borou
688	Dufferin, Dovercourt Village	43.669005	-79.442259	Blood Brothers Brewing	43.669944	-79.436533	Brewery	W Toro
689	Dufferin, Dovercourt Village	43.669005	-79.442259	FreshCo	43.667918	-79.440754	Grocery Store	W Toro
...	...	...	...	...	...	...	...	
1592	Runnymede, Swansea	43.651571	-79.484450	Cards 'N' Such	43.650497	-79.480778	Post Office	W Toro
1593	Runnymede, Swansea	43.651571	-79.484450	West End Mamas	43.648703	-79.484919	Health Food Store	W Toro
1594	Runnymede, Swansea	43.651571	-79.484450	Kingsway Meat Products & Deli	43.650299	-79.480827	Butcher	W Toro
1595	Runnymede, Swansea	43.651571	-79.484450	(The New) Moksha Yoga Bloor West	43.648658	-79.485242	Yoga Studio	W Toro
1596	Runnymede, Swansea	43.651571	-79.484450	Think Fitness	43.647966	-79.486462	Gym	W Toro

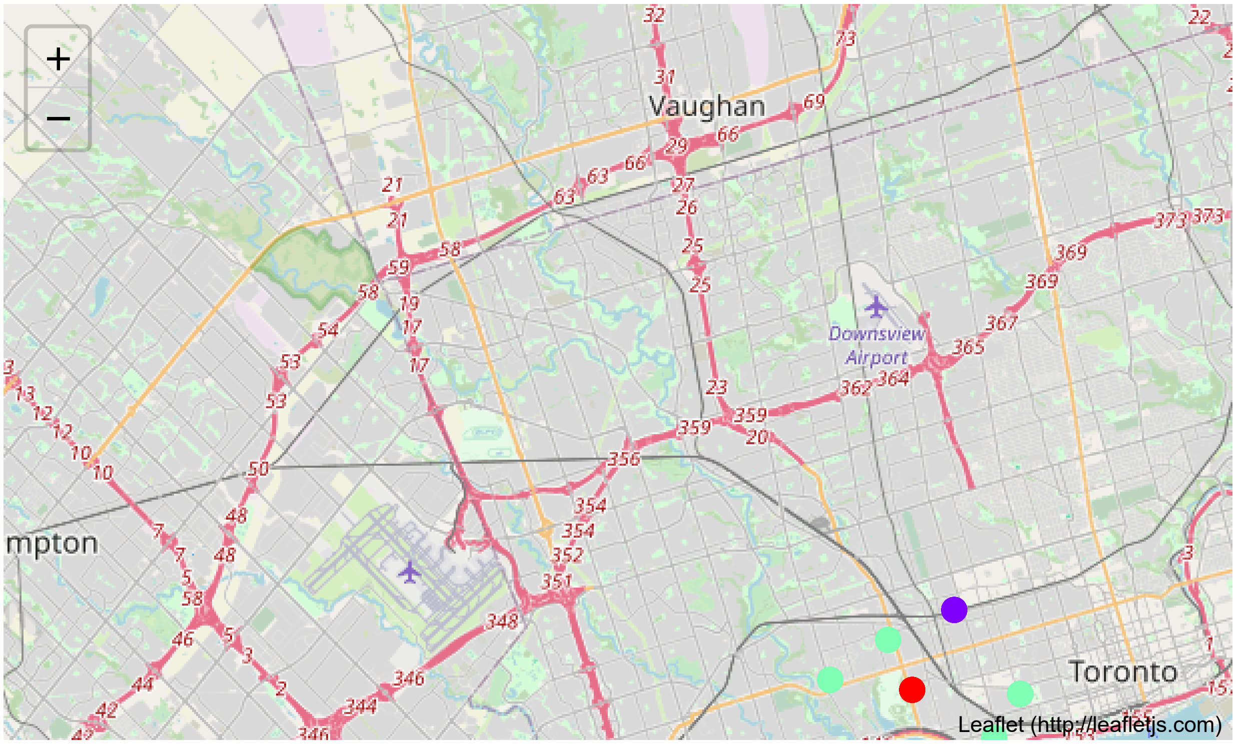
155 rows × 19 columns

In [258…

```
address = 'West Toronto, Toronto'

geolocator = Nominatim(user_agent="ny_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude

import folium
# 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(west_merged['Neighborhood Latitude'], west_merged['Ne
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7).add_to(map_clusters)

map_clusters
```

Out[258…



Stage 2 Conclusion

For Downtown Toronto, the client could pick neigborhoods in cluster 0 as they have the similar character, which is popular with dining/drink venues. For West Toronto, the cluster for restaurant venue is cluster 2 In top 10 most common venues table, there are neighborhoods that has restaurants listed at 2 in 3 most common venues, which are good location to start a restaurant business. Also, the table the information where Vietnamese restaurants are common. The client can pick these neighborhood to leverage existing customer traffice or choose another location that has had Vietnamese competior.

In [ ]: