ADS Capstone Project-Asgt4

June 19, 2020

1 Comparing NYC and Toronto for ideal hotel location

1.1 Introduction / Business Problem:

This project addresses the need to identify the location for a new hotel in the center of one of the two most important financial districts in North America: New York City & Toronto. The problem is to figure out which of the two locations would be preferable to attract people and justify the high cost of buying the property for the hotel. This project will attempt to compare the neighborhoods of downtowns of New York and Toronto and showcase the similarities and dissimilarities between the cities and justify the reasons for preferring one or the other location for starting a new hotel business.

[1]:
```
#Install all modules for this notebook
!pip install bs4
!pip install lxml
!pip install geocoder
!conda install -c conda-forge geopy --yes
!conda install -c conda-forge folium=0.5.0 --yes
```

```
Collecting bs4
  Downloading https://files.pythonhosted.org/packages/10/ed/7e8b97591f6f45617413
9ec089c769f89a94a1a4025fe967691de971f314/bs4-0.0.1.tar.gz
Collecting beautifulsoup4 (from bs4)
  Downloading https://files.pythonhosted.org/packages/66/25/ff030e24372656
16a1e9b25ccc864e0371a0bc3adb7c5a404fd661c6f4f6/beautifulsoup4-4.9.1-py3-none-
any.whl (115kB)
     |                  | 122kB 2.4MB/s eta 0:00:01
Collecting soupsieve>1.2 (from beautifulsoup4->bs4)
  Downloading https://files.pythonhosted.org/packages/6f/8f/457f4a5390eeae1cc3ae
ab89deb7724c965be841ffca6cfca9197482e470/soupsieve-2.0.1-py3-none-any.whl
Building wheels for collected packages: bs4
  Building wheel for bs4 (setup.py) … done
  Stored in directory: /home/jupyterlab/.cache/pip/wheels/a0/b0/b2/4f80b94
56b87abedbc0bf2d52235414c3467d8889be38dd472
Successfully built bs4
Installing collected packages: soupsieve, beautifulsoup4, bs4
Successfully installed beautifulsoup4-4.9.1 bs4-0.0.1 soupsieve-2.0.1
Collecting lxml
  Downloading https://files.pythonhosted.org/packages/55/6f/c87dffdd88a54d
```

```
d26a3a9fef1d14b6384a9933c455c54ce3ca7d64a84c88/lxml-4.5.1-cp36-cp36m-manylinux1_
x86_64.whl (5.5MB)
     |                  | 5.5MB 4.9MB/s eta 0:00:01
Installing collected packages: lxml
Successfully installed lxml-4.5.1
Collecting geocoder
  Downloading https://files.pythonhosted.org/packages/4f/6b/13166c909ad2f2
d76b929a4227c952630ebaf0d729f6317eb09cbceccbab/geocoder-1.38.1-py2.py3-none-
any.whl (98kB)
     |                  | 102kB 4.6MB/s ta 0:00:011
Collecting click (from geocoder)
  Downloading https://files.pythonhosted.org/packages/d2/3d/fa76db83bf75c4
f8d338c2fd15c8d33fdd7ad23a9b5e57eb6c5de26b430e/click-7.1.2-py2.py3-none-any.whl
(82kB)
     |                  | 92kB 1.5MB/s eta 0:00:011
Requirement already satisfied: six in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geocoder)
(1.15.0)
Requirement already satisfied: requests in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geocoder)
(2.23.0)
Collecting ratelim (from geocoder)
  Downloading https://files.pythonhosted.org/packages/f2/98/7e6d147fd16a10a5f821
db6e25f192265d6ecca3d82957a4fdd592cad49c/ratelim-0.1.6-py2.py3-none-any.whl
Collecting future (from geocoder)
  Downloading https://files.pythonhosted.org/packages/45/0b/38b06fd9b92dc2
b68d58b75f900e97884c45bedd2ff83203d933cf5851c9/future-0.18.2.tar.gz (829kB)
     |                  | 829kB 24.4MB/s eta 0:00:01
|                     | 583kB 24.4MB/s eta 0:00:01
Requirement already satisfied: certifi>=2017.4.17 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
requests->geocoder) (2020.4.5.2)
Requirement already satisfied: chardet<4,>=3.0.2 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
requests->geocoder) (3.0.4)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
requests->geocoder) (1.25.9)
Requirement already satisfied: idna<3,>=2.5 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
requests->geocoder) (2.9)
Requirement already satisfied: decorator in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
ratelim->geocoder) (4.4.2)
Building wheels for collected packages: future
  Building wheel for future (setup.py) … done
  Stored in directory: /home/jupyterlab/.cache/pip/wheels/8b/99/a0/81daf51
dcd359a9377b110a8a886b3895921802d2fc1b2397e
```

```
Successfully built future
Installing collected packages: click, ratelim, future, geocoder
Successfully installed click-7.1.2 future-0.18.2 geocoder-1.38.1 ratelim-0.1.6
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/jupyterlab/conda/envs/python

  added / updated specs:
    - geopy


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    geographiclib-1.50         |             py_0          34 KB  conda-forge
    geopy-1.22.0               |     pyh9f0ad1d_0          63 KB  conda-forge
    ------------------------------------------------------------
                                           Total:          97 KB

The following NEW packages will be INSTALLED:

  geographiclib      conda-forge/noarch::geographiclib-1.50-py_0
  geopy              conda-forge/noarch::geopy-1.22.0-pyh9f0ad1d_0



Downloading and Extracting Packages
geopy-1.22.0         | 63 KB     | ##################################### | 100%
geographiclib-1.50   | 34 KB     | ##################################### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
Collecting package metadata (current_repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible
solve.
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/jupyterlab/conda/envs/python

  added / updated specs:
    - folium=0.5.0
```

```
The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    altair-4.1.0               |             py_1         614 KB  conda-forge
    branca-0.4.1               |             py_0          26 KB  conda-forge
    brotlipy-0.7.0             |py36h8c4c3a4_1000         346 KB  conda-forge
    chardet-3.0.4              |py36h9f0ad1d_1006         188 KB  conda-forge
    cryptography-2.9.2         |   py36h45558ae_0         613 KB  conda-forge
    folium-0.5.0               |             py_0          45 KB  conda-forge
    pandas-1.0.5               |   py36h830a2c2_0        10.1 MB  conda-forge
    pysocks-1.7.1              |   py36h9f0ad1d_1          27 KB  conda-forge
    requests-2.24.0            |     pyh9f0ad1d_0          47 KB  conda-forge
    toolz-0.10.0               |             py_0          46 KB  conda-forge
    vincent-0.4.4              |             py_1          28 KB  conda-forge
    ------------------------------------------------------------
                                           Total:        12.0 MB

The following NEW packages will be INSTALLED:

  altair             conda-forge/noarch::altair-4.1.0-py_1
  attrs              conda-forge/noarch::attrs-19.3.0-py_0
  branca             conda-forge/noarch::branca-0.4.1-py_0
  brotlipy           conda-forge/linux-64::brotlipy-0.7.0-py36h8c4c3a4_1000
  chardet            conda-forge/linux-64::chardet-3.0.4-py36h9f0ad1d_1006
  cryptography       conda-forge/linux-64::cryptography-2.9.2-py36h45558ae_0
  entrypoints        conda-forge/linux-64::entrypoints-0.3-py36h9f0ad1d_1001
  folium             conda-forge/noarch::folium-0.5.0-py_0
  idna               conda-forge/noarch::idna-2.9-py_1
  importlib_metadata conda-forge/noarch::importlib_metadata-1.6.1-0
  jinja2             conda-forge/noarch::jinja2-2.11.2-pyh9f0ad1d_0
  jsonschema         conda-forge/linux-64::jsonschema-3.2.0-py36h9f0ad1d_1
  markupsafe         conda-forge/linux-64::markupsafe-1.1.1-py36h8c4c3a4_1
  pandas             conda-forge/linux-64::pandas-1.0.5-py36h830a2c2_0
  pyopenssl          conda-forge/noarch::pyopenssl-19.1.0-py_1
  pyrsistent         conda-forge/linux-64::pyrsistent-0.16.0-py36h8c4c3a4_0
  pysocks            conda-forge/linux-64::pysocks-1.7.1-py36h9f0ad1d_1
  pytz               conda-forge/noarch::pytz-2020.1-pyh9f0ad1d_0
  requests           conda-forge/noarch::requests-2.24.0-pyh9f0ad1d_0
  toolz              conda-forge/noarch::toolz-0.10.0-py_0
  urllib3            conda-forge/noarch::urllib3-1.25.9-py_0
  vincent            conda-forge/noarch::vincent-0.4.4-py_1


Downloading and Extracting Packages
```

```
pysocks-1.7.1         | 27 KB     | ##################################### | 100%
toolz-0.10.0          | 46 KB     | ##################################### | 100%
chardet-3.0.4         | 188 KB    | ##################################### | 100%
folium-0.5.0          | 45 KB     | ##################################### | 100%
branca-0.4.1          | 26 KB     | ##################################### | 100%
cryptography-2.9.2    | 613 KB    | ##################################### | 100%
brotlipy-0.7.0        | 346 KB    | ##################################### | 100%
altair-4.1.0          | 614 KB    | ##################################### | 100%
requests-2.24.0       | 47 KB     | ##################################### | 100%
pandas-1.0.5          | 10.1 MB   | ##################################### | 100%
vincent-0.4.4         | 28 KB     | ##################################### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
```

[46]:
```
import pandas as pd  # library for data analsysis

import numpy as np # library to handle data in a vectorized manner

from urllib.request import urlopen # module to open URLs

from bs4 import BeautifulSoup # package used to extract data from html file

import re # module provides regular expression matching operations

import geocoder #library to get latitude and longitude

from geopy.geocoders import Nominatim # convert an address into latitude and␣
↪longitude values

print('Libraries imported.')

import requests # library to handle requests

import json # library to handle JSON files

from pandas.io.json import json_normalize # tranform JSON file into a pandas␣
↪dataframe

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
```

```
Libraries imported.
```

1.1.1 Download and Explore Dataset for NEW YORK CITY:

New York City Data with emphasis on Manhattan Neighborhood has a total of 5 boroughs and 306 neighborhoods. In order to segement the neighborhoods and explore them, we will essentially need a dataset that contains the 5 boroughs and the neighborhoods that exist in each borough as well as the the latitude and logitude coordinates of each neighborhood.

Luckily, this dataset exists for free on the web. Here is the link to the dataset: https://geo.nyu.edu/catalog/nyu_2451_34572

We run a wget command and access the data.

[9]:
```
!wget -q -O 'newyork_data.json' https://cocl.us/new_york_dataset
print('Data downloaded!')
```

```
Data downloaded!
```

Load and explore the data

[10]:
```
with open('newyork_data.json') as json_data:
    newyork_data = json.load(json_data)
```

All the relevant data is now in the features key, which is basically a list of the neighborhoods. Let's define a new variable that includes this data, and look at the first few lines in the list

[11]:
```
neighborhoods_data = newyork_data['features']
neighborhoods_data[0]
```

[11]:
```
{'type': 'Feature',
 'id': 'nyu_2451_34572.1',
 'geometry': {'type': 'Point',
  'coordinates': [-73.84720052054902, 40.89470517661]},
 'geometry_name': 'geom',
 'properties': {'name': 'Wakefield',
  'stacked': 1,
  'annoline1': 'Wakefield',
  'annoline2': None,
  'annoline3': None,
  'annoangle': 0.0,
  'borough': 'Bronx',
  'bbox': [-73.84720052054902,
   40.89470517661,
   -73.84720052054902,
   40.89470517661]}}
```

Tranform the data into a *pandas* dataframe

[12]:
```
# define the dataframe columns
column_names = ['Borough', 'Neighborhood', 'Latitude', 'Longitude']

# instantiate the dataframe
neighborhoods = pd.DataFrame(columns=column_names)

#Take a look at the empty dataframe to confirm that the columns are as intended.

neighborhoods
```

[12]:
```
Empty DataFrame
Columns: [Borough, Neighborhood, Latitude, Longitude]
Index: []
```

Then let's loop through the data and fill the dataframe one row at a time.

[13]:
```
for data in neighborhoods_data:
    borough = neighborhood_name = data['properties']['borough']
    neighborhood_name = data['properties']['name']

    neighborhood_latlon = data['geometry']['coordinates']
    neighborhood_lat = neighborhood_latlon[1]
    neighborhood_lon = neighborhood_latlon[0]

    neighborhoods = neighborhoods.append({'Borough': borough,
                                          'Neighborhood': neighborhood_name,
                                          'Latitude': neighborhood_lat,
                                          'Longitude': neighborhood_lon},␣
 ↪ignore_index=True)
```

[14]:
```
#And make sure that the dataset has all 5 boroughs and 306 neighborhoods.

print('The dataframe has {} boroughs and {} neighborhoods.'.format(
        len(neighborhoods['Borough'].unique()),
        neighborhoods.shape[0]
    )
)
```

```
The dataframe has 5 boroughs and 306 neighborhoods.
```

Use geopy library to get the latitude and longitude values of New York City. In order to define an instance of the geocoder, we need to define a user_agent. We will name our agent ny_explorer, as shown below.

1.1.2 Use Folium to create a map for data visualization

let's slice the original dataframe and create a new dataframe of the Manhattan data.

[115]:
```
manhattan_data = neighborhoods[neighborhoods['Borough'] == 'Manhattan'].
↪reset_index(drop=True)
manhattan_data.head(15)
```

[115]:

	Borough	Neighborhood	Latitude	Longitude
0	Manhattan	Marble Hill	40.876551	-73.910660
1	Manhattan	Chinatown	40.715618	-73.994279
2	Manhattan	Washington Heights	40.851903	-73.936900
3	Manhattan	Inwood	40.867684	-73.921210
4	Manhattan	Hamilton Heights	40.823604	-73.949688
5	Manhattan	Manhattanville	40.816934	-73.957385
6	Manhattan	Central Harlem	40.815976	-73.943211
7	Manhattan	East Harlem	40.792249	-73.944182
8	Manhattan	Upper East Side	40.775639	-73.960508
9	Manhattan	Yorkville	40.775930	-73.947118
10	Manhattan	Lenox Hill	40.768113	-73.958860
11	Manhattan	Roosevelt Island	40.762160	-73.949168
12	Manhattan	Upper West Side	40.787658	-73.977059
13	Manhattan	Lincoln Square	40.773529	-73.985338
14	Manhattan	Clinton	40.759101	-73.996119

Let's get the geographical coordinates of Manhattan.

[16]:
```
address = 'Manhattan, NY'

geolocator = Nominatim(user_agent="ny_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geograpical coordinate of Manhattan are {}, {}.'.format(latitude,␣
↪longitude))
```

```
The geograpical coordinate of Manhattan are 40.7896239, -73.9598939.
```

let's visualize Manhattan the neighborhoods in it.

[20]:
```
# create map of Manhattan using latitude and longitude values
map_manhattan = folium.Map(location=[latitude, longitude], zoom_start=11)

# add markers to map
for lat, lng, label in zip(manhattan_data['Latitude'],␣
↪manhattan_data['Longitude'], manhattan_data['Neighborhood']):
    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_manhattan)

map_manhattanCLIENT_ID = 'TSZIAANQKD0CDSMJURVP3YMYF4FAACDSDJ2L3KKOWEQY4ND3' #␣
 ↪your Foursquare ID
CLIENT_SECRET = 'AT34D3MC0GOIAZDMUOQZEJ4FNM00BOBE35NS4X5255TS5TXO' # your␣
 ↪Foursquare Secret
VERSION = '20180605' # Foursquare API version
```

Utilize the Foursquare API to explore the neighborhoods and segment them.

Define Foursquare Credentials and Version

[21]:
```
CLIENT_ID = 'TSZIAANQKD0CDSMJURVP3YMYF4FAACDSDJ2L3KKOWEQY4ND3' # your␣
 ↪Foursquare ID
CLIENT_SECRET = 'AT34D3MC0GOIAZDMUOQZEJ4FNM00BOBE35NS4X5255TS5TXO' # your␣
 ↪Foursquare Secret
VERSION = '20180605' # Foursquare API version
```

[22]:
```
manhattan_data.loc[0, 'Neighborhood']
```

[22]:
```
'Marble Hill'
```

Get the neighborhood's latitude and longitude values.

[23]:
```
neighborhood_latitude = manhattan_data.loc[0, 'Latitude'] # neighborhood␣
 ↪latitude value
neighborhood_longitude = manhattan_data.loc[0, 'Longitude'] # neighborhood␣
 ↪longitude value

neighborhood_name = manhattan_data.loc[0, 'Neighborhood'] # neighborhood name

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

```
Latitude and longitude values of Marble Hill are 40.87655077879964,
-73.91065965862981.
```

Now, let's get the top 100 venues that are in Marble Hill within a radius of 500 meters.
First, let's create the GET request URL. Name your URL **url**.

[24]:
```
# type your answer here
LIMIT = 100 # limit of number of venues returned by Foursquare API
radius = 500 # define radius
# create URL
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)
url # display URL
```

[24]:
```
'https://api.foursquare.com/v2/venues/explore?&client_id=TSZIAANQKDOCDSMJURVP3YM
YF4FAACDSDJ2L3KKOWEQY4ND3&client_secret=AT34D3MCOGOIAZDMUOQZEJ4FNMOOBOBE35NS4X52
55TS5TXO&v=20180605&ll=40.87655077879964,-73.91065965862981&radius=500&limit=100
'
```

Double-click **here** for the solution.

Send the GET request and examine the resutls

[25]:
```
results = requests.get(url).json()
```

From the Foursquare lab in the previous module, we know that all the information is in the *items* key. Before we proceed, let's borrow the **get__category__type** function from the Foursquare lab.

[26]:
```
# function that extracts the category of the venue
def get_category_type(row):
    try:
        categories_list = row['categories']
    except:
        categories_list = row['venue.categories']

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

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

[27]:
```
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.head()
```

```
/home/jupyterlab/conda/envs/python/lib/python3.6/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
```

[27]:

	name	categories	lat	lng
0	Arturo's	Pizza Place	40.874412	-73.910271
1	Bikram Yoga	Yoga Studio	40.876844	-73.906204
2	Tibbett Diner	Diner	40.880404	-73.908937
3	Starbucks	Coffee Shop	40.877531	-73.905582
4	Dunkin'	Donut Shop	40.877136	-73.906666

And how many venues were returned by Foursquare?

[28]:
```
print('{} venues were returned by Foursquare.'.format(nearby_venues.shape[0]))
```

```
25 venues were returned by Foursquare.
```

1.2 Explore Neighborhoods in Manhattan, New York City

Create a function to repeat the same process to all the neighborhoods in Manhattan

[29]:
```
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)
```

Code to run the above function on each neighborhood and create a new dataframe called *manhattan_venues*:

[30]:
```
# type your answer here

manhattan_venues = getNearbyVenues(names=manhattan_data['Neighborhood'],
                                   latitudes=manhattan_data['Latitude'],
                                   longitudes=manhattan_data['Longitude']
                                  )
```

```
Marble Hill
Chinatown
Washington Heights
Inwood
Hamilton Heights
Manhattanville
Central Harlem
East Harlem
```

```
Upper East Side
Yorkville
Lenox Hill
Roosevelt Island
Upper West Side
Lincoln Square
Clinton
Midtown
Murray Hill
Chelsea
Greenwich Village
East Village
Lower East Side
Tribeca
Little Italy
Soho
West Village
Manhattan Valley
Morningside Heights
Gramercy
Battery Park City
Financial District
Carnegie Hill
Noho
Civic Center
Midtown South
Sutton Place
Turtle Bay
Tudor City
Stuyvesant Town
Flatiron
Hudson Yards
```

Let's check the size of the resulting dataframe

[31]:
```
print(manhattan_venues.shape)
manhattan_venues.head()
```

```
(3142, 7)
```

[31]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	\
0	Marble Hill	40.876551	-73.91066	Arturo's	
1	Marble Hill	40.876551	-73.91066	Bikram Yoga	
2	Marble Hill	40.876551	-73.91066	Tibbett Diner	
3	Marble Hill	40.876551	-73.91066	Starbucks	
4	Marble Hill	40.876551	-73.91066	Dunkin'	

```
   Venue Latitude  Venue Longitude Venue Category
```

0	40.874412	-73.910271	Pizza Place
1	40.876844	-73.906204	Yoga Studio
2	40.880404	-73.908937	Diner
3	40.877531	-73.905582	Coffee Shop
4	40.877136	-73.906666	Donut Shop

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

[32]:
```
manhattan_venues.groupby('Neighborhood').count()
```

[32]:

Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue
Battery Park City	65	65	65
Carnegie Hill	89	89	89
Central Harlem	43	43	43
Chelsea	100	100	100
Chinatown	100	100	100
Civic Center	100	100	100
Clinton	100	100	100
East Harlem	40	40	40
East Village	100	100	100
Financial District	100	100	100
Flatiron	100	100	100
Gramercy	82	82	82
Greenwich Village	100	100	100
Hamilton Heights	60	60	60
Hudson Yards	57	57	57
Inwood	58	58	58
Lenox Hill	100	100	100
Lincoln Square	96	96	96
Little Italy	100	100	100
Lower East Side	49	49	49
Manhattan Valley	43	43	43
Manhattanville	42	42	42
Marble Hill	25	25	25
Midtown	100	100	100
Midtown South	100	100	100
Morningside Heights	43	43	43
Murray Hill	92	92	92
Noho	100	100	100
Roosevelt Island	29	29	29
Soho	98	98	98
Stuyvesant Town	16	16	16
Sutton Place	100	100	100
Tribeca	75	75	75
Tudor City	74	74	74
Turtle Bay	100	100	100

Upper East Side	90		90	90
Upper West Side	90		90	90
Washington Heights	86		86	86
West Village	100		100	100
Yorkville	100		100	100

Neighborhood	Venue Latitude	Venue Longitude	Venue Category
Battery Park City	65	65	65
Carnegie Hill	89	89	89
Central Harlem	43	43	43
Chelsea	100	100	100
Chinatown	100	100	100
Civic Center	100	100	100
Clinton	100	100	100
East Harlem	40	40	40
East Village	100	100	100
Financial District	100	100	100
Flatiron	100	100	100
Gramercy	82	82	82
Greenwich Village	100	100	100
Hamilton Heights	60	60	60
Hudson Yards	57	57	57
Inwood	58	58	58
Lenox Hill	100	100	100
Lincoln Square	96	96	96
Little Italy	100	100	100
Lower East Side	49	49	49
Manhattan Valley	43	43	43
Manhattanville	42	42	42
Marble Hill	25	25	25
Midtown	100	100	100
Midtown South	100	100	100
Morningside Heights	43	43	43
Murray Hill	92	92	92
Noho	100	100	100
Roosevelt Island	29	29	29
Soho	98	98	98
Stuyvesant Town	16	16	16
Sutton Place	100	100	100
Tribeca	75	75	75
Tudor City	74	74	74
Turtle Bay	100	100	100
Upper East Side	90	90	90
Upper West Side	90	90	90
Washington Heights	86	86	86
West Village	100	100	100

```
Yorkville                      100            100            100
```

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

[33]:
```
print('There are {} uniques categories.'.format(len(manhattan_venues['Venue␣
 ↪Category'].unique())))
```

```
There are 331 uniques categories.
```

1.3 Analyze Each Neighborhood in New York City's Manhattan borough

[34]:
```
# one hot encoding
manhattan_onehot = pd.get_dummies(manhattan_venues[['Venue Category']],␣
 ↪prefix="", prefix_sep="")

# add neighborhood column back to dataframe
manhattan_onehot['Neighborhood'] = manhattan_venues['Neighborhood']

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

manhattan_onehot.head()
```

[34]:

	Neighborhood	Accessories Store	Adult Boutique	Afghan Restaurant	African Restaurant	American Restaurant	Antique Shop	Arcade	Arepa Restaurant	Argentinian Restaurant	…	Video Store	Vietnamese Restaurant	Volleyball Court	Waterfront	Whisky Bar	Wine Bar
0	Marble Hill	0	0	0	0	0	0	0	0	0	…	0					
1	Marble Hill	0	0	0	0	0	0	0	0	0	…	0					
2	Marble Hill	0	0	0	0	0	0	0	0	0	…	0					
3	Marble Hill	0	0	0	0	0	0	0	0	0	…	0					
4	Marble Hill	0	0	0	0	0	0	0	0	0	…	0					

```
   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

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

   [5 rows x 332 columns]
```

And let's examine the new dataframe size.

[35]:
```
manhattan_onehot.shape
```

[35]:
```
(3142, 332)
```

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

[36]:
```
manhattan_grouped = manhattan_onehot.groupby('Neighborhood').mean().
↪reset_index()
manhattan_grouped
```

[36]:

	Neighborhood	Accessories Store	Adult Boutique	Afghan Restaurant	\
0	Battery Park City	0.000000	0.00	0.00	
1	Carnegie Hill	0.000000	0.00	0.00	
2	Central Harlem	0.000000	0.00	0.00	
3	Chelsea	0.000000	0.00	0.00	
4	Chinatown	0.000000	0.00	0.00	
5	Civic Center	0.000000	0.00	0.00	
6	Clinton	0.000000	0.00	0.00	
7	East Harlem	0.000000	0.00	0.00	
8	East Village	0.000000	0.00	0.00	
9	Financial District	0.000000	0.00	0.00	
10	Flatiron	0.000000	0.00	0.00	
11	Gramercy	0.000000	0.00	0.00	
12	Greenwich Village	0.000000	0.00	0.00	
13	Hamilton Heights	0.000000	0.00	0.00	
14	Hudson Yards	0.000000	0.00	0.00	
15	Inwood	0.000000	0.00	0.00	
16	Lenox Hill	0.000000	0.00	0.01	
17	Lincoln Square	0.000000	0.00	0.00	
18	Little Italy	0.000000	0.00	0.00	

19	Lower East Side	0.000000	0.00	0.00
20	Manhattan Valley	0.000000	0.00	0.00
21	Manhattanville	0.000000	0.00	0.00
22	Marble Hill	0.000000	0.00	0.00
23	Midtown	0.000000	0.00	0.00
24	Midtown South	0.000000	0.00	0.00
25	Morningside Heights	0.000000	0.00	0.00
26	Murray Hill	0.000000	0.00	0.00
27	Noho	0.000000	0.00	0.00
28	Roosevelt Island	0.000000	0.00	0.00
29	Soho	0.000000	0.00	0.00
30	Stuyvesant Town	0.000000	0.00	0.00
31	Sutton Place	0.000000	0.01	0.00
32	Tribeca	0.000000	0.00	0.00
33	Tudor City	0.000000	0.00	0.00
34	Turtle Bay	0.000000	0.00	0.00
35	Upper East Side	0.000000	0.00	0.00
36	Upper West Side	0.000000	0.00	0.00
37	Washington Heights	0.011628	0.00	0.00
38	West Village	0.000000	0.00	0.00
39	Yorkville	0.000000	0.00	0.00

	African Restaurant	American Restaurant	Antique Shop	Arcade	\
0	0.000000	0.015385	0.00	0.000000	
1	0.000000	0.011236	0.00	0.000000	
2	0.046512	0.046512	0.00	0.000000	
3	0.000000	0.040000	0.00	0.000000	
4	0.000000	0.030000	0.00	0.000000	
5	0.000000	0.030000	0.01	0.000000	
6	0.000000	0.030000	0.00	0.000000	
7	0.000000	0.000000	0.00	0.000000	
8	0.000000	0.010000	0.00	0.000000	
9	0.000000	0.050000	0.00	0.000000	
10	0.000000	0.010000	0.00	0.000000	
11	0.000000	0.036585	0.00	0.012195	
12	0.000000	0.010000	0.00	0.000000	
13	0.000000	0.000000	0.00	0.000000	
14	0.000000	0.052632	0.00	0.000000	
15	0.000000	0.034483	0.00	0.000000	
16	0.000000	0.000000	0.00	0.000000	
17	0.000000	0.031250	0.00	0.000000	
18	0.000000	0.000000	0.00	0.000000	
19	0.000000	0.000000	0.00	0.000000	
20	0.000000	0.000000	0.00	0.000000	
21	0.000000	0.023810	0.00	0.000000	
22	0.000000	0.040000	0.00	0.000000	
23	0.000000	0.020000	0.00	0.000000	

```
24              0.000000              0.020000          0.00  0.000000
25              0.000000              0.069767          0.00  0.000000
26              0.000000              0.032609          0.00  0.000000
27              0.000000              0.030000          0.00  0.000000
28              0.000000              0.000000          0.00  0.000000
29              0.000000              0.010204          0.00  0.000000
30              0.000000              0.000000          0.00  0.000000
31              0.000000              0.020000          0.00  0.000000
32              0.000000              0.053333          0.00  0.000000
33              0.000000              0.013514          0.00  0.000000
34              0.000000              0.020000          0.00  0.000000
35              0.000000              0.011111          0.00  0.000000
36              0.000000              0.022222          0.00  0.000000
37              0.000000              0.011628          0.00  0.000000
38              0.000000              0.040000          0.00  0.000000
39              0.000000              0.000000          0.00  0.000000

    Arepa Restaurant  Argentinian Restaurant  …  Video Store  \
0           0.000000                0.000000  …         0.00
1           0.000000                0.011236  …         0.00
2           0.000000                0.000000  …         0.00
3           0.000000                0.000000  …         0.00
4           0.000000                0.000000  …         0.00
5           0.000000                0.000000  …         0.00
6           0.000000                0.000000  …         0.00
7           0.000000                0.000000  …         0.00
8           0.010000                0.010000  …         0.00
9           0.000000                0.000000  …         0.00
10          0.000000                0.000000  …         0.00
11          0.000000                0.000000  …         0.00
12          0.000000                0.000000  …         0.00
13          0.000000                0.000000  …         0.00
14          0.000000                0.000000  …         0.00
15          0.000000                0.000000  …         0.00
16          0.000000                0.000000  …         0.00
17          0.000000                0.000000  …         0.00
18          0.000000                0.000000  …         0.00
19          0.000000                0.020408  …         0.00
20          0.000000                0.000000  …         0.00
21          0.000000                0.000000  …         0.00
22          0.000000                0.000000  …         0.00
23          0.000000                0.000000  …         0.00
24          0.000000                0.000000  …         0.00
25          0.000000                0.000000  …         0.00
26          0.000000                0.000000  …         0.00
27          0.000000                0.010000  …         0.00
28          0.000000                0.000000  …         0.00
```

29	0.000000	0.000000	…	0.00
30	0.000000	0.000000	…	0.00
31	0.000000	0.000000	…	0.00
32	0.000000	0.013333	…	0.00
33	0.000000	0.000000	…	0.00
34	0.000000	0.000000	…	0.00
35	0.000000	0.000000	…	0.00
36	0.000000	0.000000	…	0.00
37	0.011628	0.000000	…	0.00
38	0.000000	0.000000	…	0.00
39	0.000000	0.000000	…	0.01

	Vietnamese Restaurant	Volleyball Court	Waterfront	Whisky Bar	Wine Bar	\
0	0.000000	0.000000	0.000000	0.000000	0.000000	
1	0.022472	0.000000	0.000000	0.000000	0.011236	
2	0.000000	0.000000	0.000000	0.000000	0.000000	
3	0.000000	0.000000	0.000000	0.000000	0.010000	
4	0.030000	0.000000	0.000000	0.000000	0.000000	
5	0.000000	0.000000	0.000000	0.000000	0.010000	
6	0.000000	0.000000	0.000000	0.000000	0.010000	
7	0.000000	0.000000	0.000000	0.000000	0.000000	
8	0.020000	0.000000	0.000000	0.000000	0.040000	
9	0.000000	0.000000	0.000000	0.000000	0.000000	
10	0.000000	0.000000	0.000000	0.000000	0.000000	
11	0.000000	0.000000	0.000000	0.000000	0.000000	
12	0.020000	0.000000	0.000000	0.000000	0.010000	
13	0.000000	0.000000	0.000000	0.000000	0.016667	
14	0.000000	0.000000	0.000000	0.000000	0.000000	
15	0.000000	0.000000	0.000000	0.000000	0.034483	
16	0.000000	0.000000	0.000000	0.000000	0.010000	
17	0.000000	0.000000	0.000000	0.000000	0.020833	
18	0.010000	0.000000	0.000000	0.000000	0.010000	
19	0.020408	0.000000	0.000000	0.000000	0.000000	
20	0.023256	0.000000	0.000000	0.000000	0.000000	
21	0.000000	0.000000	0.000000	0.000000	0.000000	
22	0.000000	0.000000	0.000000	0.000000	0.000000	
23	0.000000	0.000000	0.000000	0.000000	0.000000	
24	0.000000	0.000000	0.000000	0.000000	0.010000	
25	0.000000	0.000000	0.000000	0.000000	0.000000	
26	0.010870	0.000000	0.000000	0.000000	0.010870	
27	0.000000	0.000000	0.000000	0.000000	0.030000	
28	0.000000	0.000000	0.034483	0.000000	0.000000	
29	0.000000	0.000000	0.000000	0.000000	0.010204	
30	0.000000	0.000000	0.000000	0.000000	0.000000	
31	0.000000	0.000000	0.000000	0.000000	0.010000	
32	0.000000	0.013333	0.000000	0.013333	0.040000	
33	0.027027	0.000000	0.000000	0.000000	0.000000	

34	0.000000	0.000000	0.000000	0.000000	0.020000
35	0.000000	0.000000	0.000000	0.000000	0.000000
36	0.011111	0.000000	0.000000	0.000000	0.033333
37	0.000000	0.000000	0.000000	0.000000	0.011628
38	0.000000	0.000000	0.000000	0.000000	0.040000
39	0.020000	0.000000	0.000000	0.000000	0.010000

	Wine Shop	Wings Joint	Women's Store	Yoga Studio
0	0.015385	0.000000	0.000000	0.000000
1	0.033708	0.000000	0.000000	0.033708
2	0.000000	0.000000	0.000000	0.000000
3	0.010000	0.000000	0.000000	0.000000
4	0.000000	0.000000	0.000000	0.010000
5	0.020000	0.010000	0.000000	0.030000
6	0.040000	0.000000	0.000000	0.000000
7	0.000000	0.000000	0.000000	0.000000
8	0.010000	0.010000	0.000000	0.000000
9	0.020000	0.000000	0.010000	0.010000
10	0.020000	0.000000	0.000000	0.020000
11	0.012195	0.000000	0.000000	0.012195
12	0.000000	0.000000	0.000000	0.010000
13	0.000000	0.000000	0.000000	0.033333
14	0.017544	0.000000	0.000000	0.000000
15	0.017241	0.000000	0.000000	0.017241
16	0.010000	0.000000	0.010000	0.010000
17	0.031250	0.000000	0.000000	0.010417
18	0.010000	0.000000	0.020000	0.000000
19	0.000000	0.000000	0.020408	0.020408
20	0.023256	0.023256	0.000000	0.046512
21	0.000000	0.000000	0.000000	0.000000
22	0.000000	0.000000	0.000000	0.040000
23	0.000000	0.000000	0.010000	0.010000
24	0.000000	0.000000	0.000000	0.010000
25	0.000000	0.000000	0.000000	0.000000
26	0.010870	0.000000	0.000000	0.010870
27	0.020000	0.000000	0.000000	0.020000
28	0.000000	0.000000	0.000000	0.000000
29	0.000000	0.000000	0.020408	0.010204
30	0.000000	0.000000	0.000000	0.000000
31	0.020000	0.000000	0.000000	0.020000
32	0.013333	0.000000	0.000000	0.000000
33	0.027027	0.000000	0.000000	0.013514
34	0.000000	0.000000	0.000000	0.000000
35	0.022222	0.000000	0.022222	0.033333
36	0.011111	0.000000	0.000000	0.011111
37	0.023256	0.000000	0.011628	0.000000
38	0.000000	0.000000	0.000000	0.000000

```
39   0.030000      0.000000        0.000000      0.000000

[40 rows x 332 columns]
```

Let's confirm the new size

[37]:
```
manhattan_grouped.shape
```

[37]:
```
(40, 332)
```

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

[38]:
```
num_top_venues = 5

for hood in manhattan_grouped['Neighborhood']:
    print("----"+hood+"----")
    temp = manhattan_grouped[manhattan_grouped['Neighborhood'] == hood].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_venues))
    print('\n')
```

```
----Battery Park City----
           venue  freq
0           Park  0.12
1          Hotel  0.06
2    Coffee Shop  0.06
3  Memorial Site  0.05
4            Gym  0.05


----Carnegie Hill----
                venue  freq
0         Coffee Shop  0.09
1         Pizza Place  0.04
2                Café  0.04
3  Italian Restaurant  0.04
4         Yoga Studio  0.03


----Central Harlem----
                venue  freq
0                 Bar  0.05
1   Chinese Restaurant  0.05
```

```
2   African Restaurant  0.05
3  American Restaurant  0.05
4    French Restaurant  0.05


----Chelsea----
                 venue  freq
0          Coffee Shop  0.08
1          Art Gallery  0.07
2       Ice Cream Shop  0.04
3  American Restaurant  0.04
4               Bakery  0.03


----Chinatown----
                 venue  freq
0   Chinese Restaurant  0.08
1               Bakery  0.06
2         Dessert Shop  0.04
3                  Spa  0.03
4  American Restaurant  0.03


----Civic Center----
                  venue  freq
0           Coffee Shop  0.07
1          Cocktail Bar  0.05
2  Gym / Fitness Center  0.05
3                   Spa  0.04
4                 Hotel  0.04


----Clinton----
                  venue  freq
0               Theater  0.08
1    Italian Restaurant  0.05
2  Gym / Fitness Center  0.05
3             Wine Shop  0.04
4           Coffee Shop  0.04


----East Harlem----
                       venue  freq
0         Mexican Restaurant  0.12
1            Thai Restaurant  0.08
2                     Bakery  0.08
3  Latin American Restaurant  0.05
4             Sandwich Place  0.05
```

```
----East Village----
                venue  freq
0                 Bar  0.06
1  Mexican Restaurant  0.05
2        Cocktail Bar  0.05
3   Korean Restaurant  0.04
4            Wine Bar  0.04


----Financial District----
                 venue  freq
0          Coffee Shop  0.07
1  American Restaurant  0.05
2                  Bar  0.05
3         Cocktail Bar  0.04
4          Pizza Place  0.04


----Flatiron----
                      venue  freq
0      Gym / Fitness Center  0.06
1  Mediterranean Restaurant  0.05
2                     Café  0.05
3              Coffee Shop  0.04
4   New American Restaurant  0.04


----Gramercy----
                venue  freq
0                 Bar  0.06
1         Pizza Place  0.05
2          Bagel Shop  0.05
3         Coffee Shop  0.05
4  Italian Restaurant  0.04


----Greenwich Village----
                venue  freq
0  Italian Restaurant  0.09
1               Café  0.05
2    Sushi Restaurant  0.05
3      Clothing Store  0.04
4  Chinese Restaurant  0.03


----Hamilton Heights----
```

```
                venue  freq
0         Pizza Place  0.10
1         Coffee Shop  0.07
2       Deli / Bodega  0.07
3  Mexican Restaurant  0.05
4               Café  0.05


----Hudson Yards----
                  venue  freq
0                 Hotel  0.07
1  Gym / Fitness Center  0.05
2    Italian Restaurant  0.05
3   American Restaurant  0.05
4                   Gym  0.04


----Inwood----
                venue  freq
0  Mexican Restaurant  0.07
1               Café  0.05
2         Restaurant  0.05
3             Lounge  0.05
4       Deli / Bodega  0.03


----Lenox Hill----
                venue  freq
0         Coffee Shop  0.07
1  Italian Restaurant  0.07
2         Pizza Place  0.04
3               Café  0.04
4        Cocktail Bar  0.04


----Lincoln Square----
                  venue  freq
0                 Café  0.05
1                Plaza  0.05
2   Gym / Fitness Center  0.04
3     Italian Restaurant  0.04
4  Performing Arts Venue  0.04


----Little Italy----
                     venue  freq
0                   Bakery  0.06
1        Chinese Restaurant  0.05
```

```
2        Italian Restaurant  0.04
3  Mediterranean Restaurant  0.04
4           Bubble Tea Shop  0.04


----Lower East Side----
                venue  freq
0  Chinese Restaurant  0.06
1               Café  0.04
2        Cocktail Bar  0.04
3         Coffee Shop  0.04
4         Art Gallery  0.04


----Manhattan Valley----
                venue  freq
0         Coffee Shop  0.09
1                 Bar  0.07
2  Mexican Restaurant  0.05
3         Pizza Place  0.05
4         Yoga Studio  0.05


----Manhattanville----
                venue  freq
0         Coffee Shop  0.10
1       Deli / Bodega  0.07
2  Mexican Restaurant  0.05
3  Seafood Restaurant  0.05
4  Italian Restaurant  0.05


----Marble Hill----
            venue  freq
0  Sandwich Place  0.12
1     Coffee Shop  0.08
2             Gym  0.08
3     Yoga Studio  0.04
4            Bank  0.04


----Midtown----
            venue  freq
0     Coffee Shop  0.07
1           Hotel  0.07
2         Theater  0.04
3          Bakery  0.04
4  Clothing Store  0.04
```

```
----Midtown South----
                 venue  freq
0    Korean Restaurant  0.11
1                Hotel  0.08
2                 Café  0.04
3  Japanese Restaurant  0.04
4         Burger Joint  0.03


----Morningside Heights----
                 venue  freq
0                 Park  0.07
1           Bookstore  0.07
2  American Restaurant  0.07
3          Coffee Shop  0.07
4       Sandwich Place  0.05


----Murray Hill----
                 venue  freq
0       Sandwich Place  0.05
1          Coffee Shop  0.05
2  Japanese Restaurant  0.04
3                Hotel  0.04
4          Pizza Place  0.03


----Noho----
                venue  freq
0         Pizza Place  0.06
1  Italian Restaurant  0.06
2       Grocery Store  0.04
3         Coffee Shop  0.04
4   French Restaurant  0.03


----Roosevelt Island----
            venue  freq
0            Park  0.07
1     Coffee Shop  0.03
2        Bus Line  0.03
3  Farmers Market  0.03
4   Metro Station  0.03


----Soho----
```

```
                    venue  freq
0       Italian Restaurant  0.08
1           Clothing Store  0.05
2              Coffee Shop  0.04
3  Mediterranean Restaurant  0.04
4           Sandwich Place  0.04


----Stuyvesant Town----
                  venue  freq
0                  Park  0.12
1        Baseball Field  0.06
2           Pet Service  0.06
3              Fountain  0.06
4  Gym / Fitness Center  0.06


----Sutton Place----
                    venue  freq
0    Gym / Fitness Center  0.06
1      Italian Restaurant  0.05
2                    Park  0.04
3             Coffee Shop  0.04
4  Furniture / Home Store  0.03


----Tribeca----
                 venue  freq
0                 Park  0.07
1   Italian Restaurant  0.07
2  American Restaurant  0.05
3          Coffee Shop  0.04
4                Café  0.04


----Tudor City----
                venue  freq
0               Café  0.07
1               Park  0.07
2  Mexican Restaurant  0.05
3       Deli / Bodega  0.04
4    Greek Restaurant  0.03


----Turtle Bay----
                venue  freq
0  Italian Restaurant  0.06
1         Coffee Shop  0.05
```

```
2                Park  0.04
3    Sushi Restaurant  0.04
4                Café  0.04


----Upper East Side----
                  venue  freq
0    Italian Restaurant  0.09
1           Coffee Shop  0.06
2                Bakery  0.06
3             Juice Bar  0.04
4  Gym / Fitness Center  0.04


----Upper West Side----
                 venue  freq
0  Italian Restaurant  0.06
1                 Bar  0.04
2         Coffee Shop  0.03
3        Dessert Shop  0.03
4              Bakery  0.03


----Washington Heights----
                       venue  freq
0                       Café  0.06
1                     Bakery  0.05
2          Mobile Phone Shop  0.03
3              Grocery Store  0.03
4  Latin American Restaurant  0.02


----West Village----
                     venue  freq
0       Italian Restaurant  0.07
1  New American Restaurant  0.05
2                     Park  0.04
3             Cocktail Bar  0.04
4      American Restaurant  0.04


----Yorkville----
                 venue  freq
0  Italian Restaurant  0.06
1                 Gym  0.06
2                 Bar  0.05
3         Coffee Shop  0.05
4    Sushi Restaurant  0.04
```

Let's put that into a *pandas* dataframe First, let's write a function to sort the venues in descending order.

[39]:
```
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]
```

Now let's create the new dataframe and display the top 10 venues for each neighborhood.

[40]:
```
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'] = manhattan_grouped['Neighborhood']

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

neighborhoods_venues_sorted.head()
```

[40]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
0	Battery Park City	Park	Coffee Shop	Hotel	Memorial Site	Gym
1	Carnegie Hill	Coffee Shop	Café	Italian Restaurant	Pizza Place	Yoga Studio
2	Central Harlem	Chinese Restaurant	American Restaurant	Bar	French Restaurant	Seafood Restaurant
3	Chelsea	Coffee Shop	Art Gallery	American Restaurant	Ice Cream Shop	French Restaurant
4	Chinatown	Chinese Restaurant	Bakery			

```
4          Dessert Shop  Vietnamese Restaurant          Cocktail Bar

  6th Most Common Venue 7th Most Common Venue 8th Most Common Venue  \
0         Boat or Ferry                Plaza          Gourmet Shop
1                   Gym  Gym / Fitness Center             Bookstore
2  Gym / Fitness Center    African Restaurant          Cycle Studio
3                 Café                Bakery                Market
4                  Bar                   Spa       Bubble Tea Shop

  9th Most Common Venue 10th Most Common Venue
0            Food Court          Shopping Mall
1             Wine Shop  Vietnamese Restaurant
2   Fried Chicken Joint                Library
3    Italian Restaurant                Theater
4        Ice Cream Shop    American Restaurant
```

1.4 Cluster Neighborhoods in New York City

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

[43]:
```
# set number of clusters
kclusters = 5

manhattan_grouped_clustering = manhattan_grouped.drop('Neighborhood', 1)

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

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

[43]:
```
array([1, 0, 0, 1, 0, 1, 1, 2, 0, 0], dtype=int32)
```

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

[44]:
```
# add clustering labels
neighborhoods_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)

manhattan_merged = manhattan_data

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

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

[44]:

	Borough	Neighborhood	Latitude	Longitude	Cluster Labels	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	Manhattan	Marble Hill	40.876551	-73.910660	2	Sandwich Place	Coffee Shop	Gym	Yoga Studio	Tennis Stadium	Supplement Shop	Steakhouse	Seafood Restaurant	Donut Shop	Kids Store
1	Manhattan	Chinatown	40.715618	-73.994279	0	Chinese Restaurant	Bakery	Dessert Shop	Vietnamese Restaurant	Cocktail Bar	Bar	Spa	Bubble Tea Shop	Ice Cream Shop	American Restaurant
2	Manhattan	Washington Heights	40.851903	-73.936900	2	Café	Bakery	Mobile Phone Shop	Grocery Store	Deli / Bodega	Pizza Place	Latin American Restaurant	Supermarket	Tapas Restaurant	Sandwich Place
3	Manhattan	Inwood	40.867684	-73.921210	2	Mexican Restaurant	Lounge	Restaurant	Café	Wine Bar	Pizza Place	American Restaurant	Caribbean Restaurant	Frozen Yogurt Shop	Deli / Bodega
4	Manhattan	Hamilton Heights	40.823604	-73.949688	2	Pizza Place	Deli / Bodega	Coffee Shop	Mexican Restaurant	Café	Yoga Studio	Sandwich Place	Sushi Restaurant	Bakery	Caribbean Restaurant

Finally, let's visualize the resulting clusters

[47]:
```
# 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(manhattan_merged['Latitude'],␣
 ↪manhattan_merged['Longitude'], manhattan_merged['Neighborhood'],␣
 ↪manhattan_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
```

[47]: <folium.folium.Map at 0x7f49b4179160>

1.5 Examine Clusters in New York City's Manhattan borough

Now, you can examine each cluster and determine the discriminating venue categories that distinguish each cluster. Based on the defining categories, you can then assign a name to each cluster. I will leave this exercise to you.

Cluster 1

[48]:
```
manhattan_merged.loc[manhattan_merged['Cluster Labels'] == 0, manhattan_merged.
 ↪columns[[1] + list(range(5, manhattan_merged.shape[1]))]]
```

[48]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
1	Chinatown	Chinese Restaurant	Bakery	Dessert Shop	Vietnamese Restaurant	Cocktail Bar
6	Central Harlem	Chinese Restaurant	American Restaurant	Bar	French Restaurant	Seafood Restaurant
9	Yorkville	Gym	Italian Restaurant	Bar	Coffee Shop	Sushi Restaurant
12	Upper West Side	Italian Restaurant	Bar	Indian Restaurant	Wine Bar	Bakery
19	East Village	Bar	Mexican Restaurant			
20	Lower East Side	Chinese Restaurant	Bakery			
25	Manhattan Valley	Coffee Shop	Bar			
27	Gramercy	Bar	Coffee Shop			
29	Financial District	Coffee Shop	Bar			
30	Carnegie Hill	Coffee Shop	Café			

```
19          Cocktail Bar               Wine Bar           Coffee Shop
20      Ramen Restaurant                   Café          Art Gallery
25           Yoga Studio            Pizza Place   Mexican Restaurant
27           Pizza Place             Bagel Shop   Italian Restaurant
29   American Restaurant           Cocktail Bar          Pizza Place
30    Italian Restaurant            Pizza Place          Yoga Studio

   6th Most Common Venue 7th Most Common Venue 8th Most Common Venue  \
1                    Bar                   Spa       Bubble Tea Shop
6   Gym / Fitness Center    African Restaurant          Cycle Studio
9            Pizza Place         Deli / Bodega    Mexican Restaurant
12          Dessert Shop           Coffee Shop        Ice Cream Shop
19           Pizza Place     Korean Restaurant   Japanese Restaurant
20           Coffee Shop          Cocktail Bar           Yoga Studio
25     Indian Restaurant    Chinese Restaurant  Ethiopian Restaurant
27    Mexican Restaurant   American Restaurant            Playground
29    Italian Restaurant             Juice Bar            Steakhouse
30                   Gym  Gym / Fitness Center             Bookstore

       9th Most Common Venue    10th Most Common Venue
1              Ice Cream Shop       American Restaurant
6        Fried Chicken Joint                   Library
9        Japanese Restaurant                     Diner
12  Mediterranean Restaurant        Mexican Restaurant
19             Ice Cream Shop       Filipino Restaurant
20             Clothing Store  Mediterranean Restaurant
25             Farmers Market            Clothing Store
27              Grocery Store              Cocktail Bar
29                       Park               Event Space
30                 Wine Shop     Vietnamese Restaurant
```

Cluster 2

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

[49]:
```
         Neighborhood 1st Most Common Venue 2nd Most Common Venue  \
13    Lincoln Square                 Plaza                 Café
14           Clinton               Theater   Italian Restaurant
15           Midtown                 Hotel          Coffee Shop
16       Murray Hill        Sandwich Place          Coffee Shop
17           Chelsea           Coffee Shop          Art Gallery
28  Battery Park City                  Park          Coffee Shop
32       Civic Center           Coffee Shop         Cocktail Bar
33     Midtown South     Korean Restaurant                Hotel
34       Sutton Place  Gym / Fitness Center   Italian Restaurant
38           Flatiron  Gym / Fitness Center                Café
```

```
39       Hudson Yards                 Hotel  Gym / Fitness Center

       3rd Most Common Venue    4th Most Common Venue   5th Most Common Venue  \
13     Performing Arts Venue                  Theater      Italian Restaurant
14      Gym / Fitness Center          Sandwich Place              Wine Shop
15                   Bakery                  Theater         Clothing Store
16       Japanese Restaurant                   Hotel            Pizza Place
17       American Restaurant          Ice Cream Shop      French Restaurant
28                     Hotel           Memorial Site                    Gym
32      Gym / Fitness Center                    Spa      French Restaurant
33                     Café      Japanese Restaurant            Burger Joint
34                     Park             Coffee Shop  Furniture / Home Store
38  Mediterranean Restaurant  New American Restaurant             Coffee Shop
39       American Restaurant       Italian Restaurant             Coffee Shop

   6th Most Common Venue 7th Most Common Venue 8th Most Common Venue  \
13  Gym / Fitness Center          Concert Hall   American Restaurant
14                   Gym           Coffee Shop         Cocktail Bar
15             Steakhouse      Cuban Restaurant           Pizza Place
16  Gym / Fitness Center   American Restaurant         Deli / Bodega
17                 Café                Bakery                Market
28         Boat or Ferry                 Plaza          Gourmet Shop
32                 Hotel          Yoga Studio                 Café
33         Indie Theater           Coffee Shop        Scenic Lookout
34            Bagel Shop                   Gym           Yoga Studio
38                  Spa                   Gym    Italian Restaurant
39                 Park                   Gym               Dog Run

   9th Most Common Venue         10th Most Common Venue
13           Coffee Shop            Indie Movie Theater
14    American Restaurant                          Hotel
15     Salon / Barbershop                Cosmetics Shop
16            Bagel Shop              Jewish Restaurant
17     Italian Restaurant                        Theater
28            Food Court                Shopping Mall
32                  Park            American Restaurant
33                Lounge                            Spa
34           Pizza Place                Lingerie Store
38    Japanese Restaurant  Vegetarian / Vegan Restaurant
39            Restaurant                           Bar
```

Cluster 3

[50]:
```
manhattan_merged.loc[manhattan_merged['Cluster Labels'] == 2, manhattan_merged.
↪columns[[1] + list(range(5, manhattan_merged.shape[1]))]]
```

[50]:

	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	Marble Hill	Sandwich Place	Coffee Shop	Gym	Yoga Studio	Tennis Stadium	Supplement Shop	Steakhouse	Seafood Restaurant	Donut Shop	Kids Store
2	Washington Heights	Café	Bakery	Mobile Phone Shop	Grocery Store	Deli / Bodega	Pizza Place	Latin American Restaurant	Supermarket	Tapas Restaurant	Sandwich Place
3	Inwood	Mexican Restaurant	Lounge	Restaurant	Café	Wine Bar	Pizza Place	American Restaurant	Caribbean Restaurant	Frozen Yogurt Shop	Deli / Bodega
4	Hamilton Heights	Pizza Place	Deli / Bodega	Coffee Shop	Mexican Restaurant	Café	Yoga Studio	Sandwich Place	Sushi Restaurant	Bakery	Caribbean Restaurant
5	Manhattanville	Coffee Shop	Deli / Bodega	Italian Restaurant	Mexican Restaurant	Seafood Restaurant	Lounge	Bar	Spanish Restaurant	Food & Drink Shop	Climbing Gym
7	East Harlem	Mexican Restaurant	Thai Restaurant	Bakery	Latin American Restaurant	Deli / Bodega	Sandwich Place	Spa	Liquor Store	Taco Place	Gas Station
11	Roosevelt Island	Park	Soccer Field	School	Scenic Lookout	Sandwich Place	Liquor Store	Coffee Shop	Metro Station	Dry Cleaner	Outdoors & Recreation
26	Morningside Heights	Park	Bookstore	American Restaurant	Coffee Shop	Burger Joint	New American Restaurant	Sandwich Place	Deli / Bodega	Café	Greek Restaurant
36	Tudor City	Park	Café	Mexican Restaurant	Deli / Bodega	Diner	Greek Restaurant	Pizza Place	Dog Run	Coffee Shop	Garden

Cluster 4

[51]:
```
manhattan_merged.loc[manhattan_merged['Cluster Labels'] == 3, manhattan_merged.
↪columns[[1] + list(range(5, manhattan_merged.shape[1]))]]
```

[51]:

	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
8	Upper East Side	Italian Restaurant	Coffee Shop	Bakery	Gym / Fitness Center	Juice Bar	Yoga Studio	French Restaurant	Spa	Women's Store	Sushi Restaurant
10	Lenox Hill	Coffee Shop	Italian Restaurant	Pizza Place	Café	Cocktail Bar	Sushi Restaurant	Gym / Fitness Center	Gym	Burger Joint	Thai Restaurant
18	Greenwich Village	Italian Restaurant	Sushi Restaurant	Café	Clothing Store	Ice Cream Shop	Chinese Restaurant	French Restaurant	Bar	Caribbean Restaurant	Coffee Shop
21	Tribeca	Park	Italian Restaurant	American Restaurant	Greek Restaurant	Wine Bar	Spa	Café	Coffee Shop	Burger Joint	Playground
22	Little Italy	Bakery	Chinese Restaurant	Spa	Bubble Tea Shop	Mediterranean Restaurant	Italian Restaurant	Ice Cream Shop	Café	Cosmetics Shop	Thai Restaurant
23	Soho	Italian Restaurant	Clothing Store	Sandwich Place	Coffee Shop	Mediterranean Restaurant	Bakery	Ice Cream Shop	Hotel	Café	French Restaurant
24	West Village	Italian Restaurant	New American Restaurant	Cocktail Bar	Park	Wine Bar	American Restaurant	Bakery	Jazz Club	Coffee Shop	Pizza Place
31	Noho	Pizza Place	Italian Restaurant	Coffee Shop	Grocery Store	French Restaurant	Sandwich Place	Cocktail Bar	Mexican Restaurant	American Restaurant	Wine Bar
35	Turtle Bay	Italian Restaurant	Coffee Shop	Park	Café	Sushi Restaurant	Deli / Bodega	Seafood Restaurant	French Restaurant	Turkish Restaurant	Garden

Cluster 5

[52]:
```
manhattan_merged.loc[manhattan_merged['Cluster Labels'] == 4, manhattan_merged.
↪columns[[1] + list(range(5, manhattan_merged.shape[1]))]]
```

[52]:
```
        Neighborhood 1st Most Common Venue 2nd Most Common Venue  \
37  Stuyvesant Town                  Park                   Bar

   3rd Most Common Venue 4th Most Common Venue 5th Most Common Venue  \
37           Gas Station          Cocktail Bar        Baseball Field

   6th Most Common Venue 7th Most Common Venue 8th Most Common Venue  \
37  Gym / Fitness Center           Coffee Shop       Harbor / Marina

   9th Most Common Venue 10th Most Common Venue
37          Skating Rink                 Bistro
```

1.6 Results

It can be seen from the clustering of the New York city data, the clusters 1 through 4 shows interesting venues indicating a good placement for a new hotel.

1.7 Download and Explore Dataset for Toronto, CANADA

[53]:
```
# provide web address where data exists
url = "https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M"
html = urlopen(url)
```

[54]:
```
# Create BeautifulSoup object
soup = BeautifulSoup(html, 'lxml')
type(soup)
```

[54]:
```
bs4.BeautifulSoup
```

[55]:
```
# Get the title
title = soup.title
print(title)
```

```
<title>List of postal codes of Canada: M - Wikipedia</title>
```

[56]:
```
# Print out the text
text = soup.get_text()
#print(soup.text)
```

[57]:
```
#assign data to variable
Toronto_data = soup.find_all('table')[0]
```

[58]:
```
# check data type
type(Toronto_data)
```

[58]:
```
bs4.element.Tag
```

[59]:
```
# Use for loop to extract relavant data from the table
list_rows = []
for row in Toronto_data.find_all('tr'):
    temp_list = []
    for cell in row.find_all('td'):
        clean = re.compile('<.*?>|\\n')
        clean2 = (re.sub(clean, '',str(cell)))
        temp_list.append(clean2)
    list_rows.append(temp_list)
list_rows[:5]
```

[59]:
```
[[],
 ['M1A', 'Not assigned', 'Not assigned'],
 ['M2A', 'Not assigned', 'Not assigned'],
 ['M3A', 'North York', 'Parkwoods'],
 ['M4A', 'North York', 'Victoria Village']]
```

[60]:
```
# Create DataFrame to store relavant data
Toronto_data1 = pd.DataFrame(list_rows[1:])
Toronto_data1.head()
```

[60]:

	0	1	2
0	M1A	Not assigned	Not assigned
1	M2A	Not assigned	Not assigned
2	M3A	North York	Parkwoods
3	M4A	North York	Victoria Village
4	M5A	Downtown Toronto	Regent Park, Harbourfront

[61]:
```
# Create variable to store row data
Toronto_data.find_all('tr')[0]
row1 = Toronto_data.find_all('tr')[0]
```

[62]:
```
# get column names
col_name=[]
for cell in row1.find_all('th'):
    clean = re.compile('<.*?>|\\n')
    clean2 = (re.sub(clean, '',str(cell)))
    col_name.append(clean2)
print(col_name)
```

```
['Postal Code', 'Borough', 'Neighborhood']
```

[63]:
```
#assign column names to dataframe
Toronto_data1.columns=col_name
Toronto_data1.head()
```

[63]:

	Postal Code	Borough	Neighborhood
0	M1A	Not assigned	Not assigned
1	M2A	Not assigned	Not assigned
2	M3A	North York	Parkwoods
3	M4A	North York	Victoria Village
4	M5A	Downtown Toronto	Regent Park, Harbourfront

[64]:
```
# Select only Boroughs that have assigned names
Toronto_data2=Toronto_data1[Toronto_data1['Borough']!='Not assigned'] # replace␣
↪with filtered data
```

[65]:
```
Toronto_data2.head()
```

[65]:

	Postal Code	Borough	Neighborhood
2	M3A	North York	Parkwoods
3	M4A	North York	Victoria Village
4	M5A	Downtown Toronto	Regent Park, Harbourfront
5	M6A	North York	Lawrence Manor, Lawrence Heights
6	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government

[66]:
```
Toronto_data2[Toronto_data2['Neighborhood']=='Not assigned'] #check if any␣
↪Neighborhood name is "Not Assigned"
```

[66]:
```
Empty DataFrame
Columns: [Postal Code, Borough, Neighborhood]
Index: []
```

[67]:
```
Toronto_data2.shape
```

[67]:
```
(103, 3)
```

[68]:
```
# initialize your variable to None
#lat_lng_coords = None

# loop until you get the coordinates
#while(lat_lng_coords is None):
#  g = geocoder.google('{}, Toronto, Ontario'.format('M5G'))
#  lat_lng_coords = g.latlng

#latitude = lat_lng_coords[0]
#longitude = lat_lng_coords[1]
```

[69]:
```
path = 'https://cocl.us/Geospatial_data'
latlon = pd.read_csv(path)
latlon.head()
```

[69]:

	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

[70]:
```
Toronto_data3 = Toronto_data2.merge(latlon, on='Postal Code') #merge dataframe␣
↪to include latitude and longitude columns
Toronto_data3.shape
```

[70]: (103, 5)

[116]:
```
Toronto_data3.head(15)  #show DataFrame with the 15 columns
```

[116]:

	Postal Code	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	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.662301	-79.389494
5	M9A	Etobicoke	Islington Avenue, Humber Valley Village	43.667856	-79.532242
6	M1B	Scarborough	Malvern, Rouge	43.806686	-79.194353
7	M3B	North York	Don Mills	43.745906	-79.352188
8	M4B	East York	Parkview Hill, Woodbine Gardens	43.706397	-79.309937
9	M5B	Downtown Toronto	Garden District, Ryerson	43.657162	-79.378937
10	M6B	North York			
11	M9B	Etobicoke			
12	M1C	Scarborough			
13	M3C	North York			
14	M4C	East York			

```
10                                        Glencairn  43.709577 -79.445073
11  West Deane Park, Princess Gardens, Martin Grov…  43.650943 -79.554724
12            Rouge Hill, Port Union, Highland Creek  43.784535 -79.160497
13                                        Don Mills  43.725900 -79.340923
14                                  Woodbine Heights  43.695344 -79.318389
```

Use geopy library to get the latitude and longitude values of Toronto

[72]:
```
address = 'Toronto, ON'

geolocator = Nominatim(user_agent="CA_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geograpical coordinate of Toronto are {}, {}.'.format(latitude,␣
 ↪longitude))
```

```
The geograpical coordinate of Toronto are 43.6534817, -79.3839347.
```

Create a map of Toronto with neighborhoods superimposed on top.

[73]:
```
import folium # map rendering library

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

# add markers to map
for lat, lng, borough, neighborhood in zip(Toronto_data3['Latitude'],␣
 ↪Toronto_data3['Longitude'], Toronto_data3['Borough'],␣
 ↪Toronto_data3['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
```

[73]:
```
<folium.folium.Map at 0x7f49ac6d5940>
```

Simplify the above map and segment and cluster only the neighborhoods in Downtown Toronto. So we slice the original dataframe and create a new dataframe of the

Downtown Toronto data

[74]:
```
dttor_data = Toronto_data3[Toronto_data3['Borough'] == 'Downtown Toronto'].
 ↪reset_index(drop=True)
dttor_data.head()
```

[74]:
```
  Postal Code           Borough                                  Neighborhood  \
0         M5A  Downtown Toronto                    Regent Park, Harbourfront
1         M7A  Downtown Toronto  Queen's Park, Ontario Provincial Government
2         M5B  Downtown Toronto                     Garden District, Ryerson
3         M5C  Downtown Toronto                                St. James Town
4         M5E  Downtown Toronto                                   Berczy Park

    Latitude  Longitude
0  43.654260 -79.360636
1  43.662301 -79.389494
2  43.657162 -79.378937
3  43.651494 -79.375418
4  43.644771 -79.373306
```

Let's get the geographical coordinates of Downtown Toronto:

[75]:
```
address = 'Downtown Toronto, ON'

geolocator = Nominatim(user_agent="dttor_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geograpical coordinate of Downtown Toronto are {}, {}.'.
 ↪format(latitude, longitude))
```

```
The geograpical coordinate of Downtown Toronto are 43.6563221, -79.3809161.
```

As we did with all of Toronto, let's visualize Downtown Toronto the neighborhoods in it:

[76]:
```
# create map of Downtown Toronto using latitude and longitude values
map_dttor = folium.Map(location=[latitude, longitude], zoom_start=12)

# add markers to map
for lat, lng, label in zip(dttor_data['Latitude'], dttor_data['Longitude'],␣
 ↪dttor_data['Neighborhood']):
    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_dttor)

map_dttor
```

[76]: <folium.folium.Map at 0x7f49ac531f98>

1.8 Explore Neighborhoods in Toronto

[77]:
```
ATor_data = Toronto_data3[Toronto_data3['Borough'].str.contains('Toronto')].
 ↪reset_index(drop=True)
ATor_data.head()
```

[77]:

	Postal Code	Borough	Neighborhood	\
0	M5A	Downtown Toronto	Regent Park, Harbourfront	
1	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government	
2	M5B	Downtown Toronto	Garden District, Ryerson	
3	M5C	Downtown Toronto	St. James Town	
4	M4E	East Toronto	The Beaches	

	Latitude	Longitude
0	43.654260	-79.360636
1	43.662301	-79.389494
2	43.657162	-79.378937
3	43.651494	-79.375418
4	43.676357	-79.293031

Create a function to explore all neighborhoods in Toronto

[94]:
```
import requests # library to handle requests
# type your answer here
LIMIT = 100
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)
```

Code to run the above function on each neighborhood and create a new dataframe called *toronto_venues*

[95]:
```
toronto_venues = getNearbyVenues(names=ATor_data['Neighborhood'],
                                 latitudes=ATor_data['Latitude'],
                                 longitudes=ATor_data['Longitude']
                                )
```

```
Regent Park, Harbourfront
Queen's Park, Ontario Provincial Government
Garden District, Ryerson
St. James Town
The Beaches
Berczy Park
Central Bay Street
Christie
Richmond, Adelaide, King
Dufferin, Dovercourt Village
Harbourfront East, Union Station, Toronto Islands
Little Portugal, Trinity
The Danforth West, Riverdale
```

```
Toronto Dominion Centre, Design Exchange
Brockton, Parkdale Village, Exhibition Place
India Bazaar, The Beaches West
Commerce Court, Victoria Hotel
Studio District
Lawrence Park
Roselawn
Davisville North
Forest Hill North &amp;amp; West, Forest Hill Road Park
High Park, The Junction South
North Toronto West,  Lawrence Park
The Annex, North Midtown, Yorkville
Parkdale, Roncesvalles
Davisville
University of Toronto, Harbord
Runnymede, Swansea
Moore Park, Summerhill East
Kensington Market, Chinatown, Grange Park
Summerhill West, Rathnelly, South Hill, Forest Hill SE, Deer Park
CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay,
South Niagara, Island airport
Rosedale
Stn A PO Boxes
St. James Town, Cabbagetown
First Canadian Place, Underground city
Church and Wellesley
Business reply mail Processing Centre, South Central Letter Processing Plant
Toronto
```

[96]:
```
#check size of resulting dataframe:

print(toronto_venues.shape)
toronto_venues.head()
```

```
(1627, 7)
```

[96]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude
0	Regent Park, Harbourfront	43.65426	-79.360636	Roselle Desserts	43.653447	-79.362017
1	Regent Park, Harbourfront	43.65426	-79.360636	Tandem Coffee	43.653559	-79.361809
2	Regent Park, Harbourfront	43.65426	-79.360636	Cooper Koo Family YMCA	43.653249	-79.358008
3	Regent Park, Harbourfront	43.65426	-79.360636			
4	Regent Park, Harbourfront	43.65426	-79.360636			

```
3       Body Blitz Spa East       43.654735       -79.359874
4  Dominion Pub and Kitchen       43.656919       -79.358967

        Venue Category
0               Bakery
1          Coffee Shop
2  Distribution Center
3                  Spa
4                  Pub
```

[97]:
```
#check how many venues were returned for each neighborhood

toronto_venues.groupby('Neighborhood').count()
```

[97]:

Neighborhood	Neighborhood Latitude	\
Berczy Park	58	
Brockton, Parkdale Village, Exhibition Place	24	
Business reply mail Processing Centre, South Ce…	17	
CN Tower, King and Spadina, Railway Lands, Harb…	16	
Central Bay Street	66	
Christie	17	
Church and Wellesley	83	
Commerce Court, Victoria Hotel	100	
Davisville	33	
Davisville North	9	
Dufferin, Dovercourt Village	15	
First Canadian Place, Underground city	100	
Forest Hill North & West, Forest Hill Road …	4	
Garden District, Ryerson	100	
Harbourfront East, Union Station, Toronto Islands	100	
High Park, The Junction South	24	
India Bazaar, The Beaches West	20	
Kensington Market, Chinatown, Grange Park	60	
Lawrence Park	3	
Little Portugal, Trinity	44	
Moore Park, Summerhill East	2	
North Toronto West, Lawrence Park	20	
Parkdale, Roncesvalles	15	
Queen's Park, Ontario Provincial Government	32	
Regent Park, Harbourfront	47	
Richmond, Adelaide, King	93	
Rosedale	4	
Roselawn	1	
Runnymede, Swansea	34	
St. James Town	80	
St. James Town, Cabbagetown	47	

Stn A PO Boxes	97
Studio District	41
Summerhill West, Rathnelly, South Hill, Forest …	16
The Annex, North Midtown, Yorkville	20
The Beaches	6
The Danforth West, Riverdale	43
Toronto Dominion Centre, Design Exchange	100
University of Toronto, Harbord	36

Neighborhood	Neighborhood Longitude	\
Berczy Park	58	
Brockton, Parkdale Village, Exhibition Place	24	
Business reply mail Processing Centre, South Ce…	17	
CN Tower, King and Spadina, Railway Lands, Harb…	16	
Central Bay Street	66	
Christie	17	
Church and Wellesley	83	
Commerce Court, Victoria Hotel	100	
Davisville	33	
Davisville North	9	
Dufferin, Dovercourt Village	15	
First Canadian Place, Underground city	100	
Forest Hill North & West, Forest Hill Road …	4	
Garden District, Ryerson	100	
Harbourfront East, Union Station, Toronto Islands	100	
High Park, The Junction South	24	
India Bazaar, The Beaches West	20	
Kensington Market, Chinatown, Grange Park	60	
Lawrence Park	3	
Little Portugal, Trinity	44	
Moore Park, Summerhill East	2	
North Toronto West, Lawrence Park	20	
Parkdale, Roncesvalles	15	
Queen's Park, Ontario Provincial Government	32	
Regent Park, Harbourfront	47	
Richmond, Adelaide, King	93	
Rosedale	4	
Roselawn	1	
Runnymede, Swansea	34	
St. James Town	80	
St. James Town, Cabbagetown	47	
Stn A PO Boxes	97	
Studio District	41	
Summerhill West, Rathnelly, South Hill, Forest …	16	
The Annex, North Midtown, Yorkville	20	
The Beaches	6	

The Danforth West, Riverdale	43
Toronto Dominion Centre, Design Exchange	100
University of Toronto, Harbord	36

Neighborhood	Venue	Venue Latitude	\
Berczy Park	58	58	
Brockton, Parkdale Village, Exhibition Place	24	24	
Business reply mail Processing Centre, South Ce…	17	17	
CN Tower, King and Spadina, Railway Lands, Harb…	16	16	
Central Bay Street	66	66	
Christie	17	17	
Church and Wellesley	83	83	
Commerce Court, Victoria Hotel	100	100	
Davisville	33	33	
Davisville North	9	9	
Dufferin, Dovercourt Village	15	15	
First Canadian Place, Underground city	100	100	
Forest Hill North & West, Forest Hill Road …	4	4	
Garden District, Ryerson	100	100	
Harbourfront East, Union Station, Toronto Islands	100	100	
High Park, The Junction South	24	24	
India Bazaar, The Beaches West	20	20	
Kensington Market, Chinatown, Grange Park	60	60	
Lawrence Park	3	3	
Little Portugal, Trinity	44	44	
Moore Park, Summerhill East	2	2	
North Toronto West, Lawrence Park	20	20	
Parkdale, Roncesvalles	15	15	
Queen's Park, Ontario Provincial Government	32	32	
Regent Park, Harbourfront	47	47	
Richmond, Adelaide, King	93	93	
Rosedale	4	4	
Roselawn	1	1	
Runnymede, Swansea	34	34	
St. James Town	80	80	
St. James Town, Cabbagetown	47	47	
Stn A PO Boxes	97	97	
Studio District	41	41	
Summerhill West, Rathnelly, South Hill, Forest …	16	16	
The Annex, North Midtown, Yorkville	20	20	
The Beaches	6	6	
The Danforth West, Riverdale	43	43	
Toronto Dominion Centre, Design Exchange	100	100	
University of Toronto, Harbord	36	36	

	Venue Longitude	\

Neighborhood	
Berczy Park	58
Brockton, Parkdale Village, Exhibition Place	24
Business reply mail Processing Centre, South Ce…	17
CN Tower, King and Spadina, Railway Lands, Harb…	16
Central Bay Street	66
Christie	17
Church and Wellesley	83
Commerce Court, Victoria Hotel	100
Davisville	33
Davisville North	9
Dufferin, Dovercourt Village	15
First Canadian Place, Underground city	100
Forest Hill North & West, Forest Hill Road …	4
Garden District, Ryerson	100
Harbourfront East, Union Station, Toronto Islands	100
High Park, The Junction South	24
India Bazaar, The Beaches West	20
Kensington Market, Chinatown, Grange Park	60
Lawrence Park	3
Little Portugal, Trinity	44
Moore Park, Summerhill East	2
North Toronto West, Lawrence Park	20
Parkdale, Roncesvalles	15
Queen's Park, Ontario Provincial Government	32
Regent Park, Harbourfront	47
Richmond, Adelaide, King	93
Rosedale	4
Roselawn	1
Runnymede, Swansea	34
St. James Town	80
St. James Town, Cabbagetown	47
Stn A PO Boxes	97
Studio District	41
Summerhill West, Rathnelly, South Hill, Forest …	16
The Annex, North Midtown, Yorkville	20
The Beaches	6
The Danforth West, Riverdale	43
Toronto Dominion Centre, Design Exchange	100
University of Toronto, Harbord	36

Neighborhood	Venue Category
Berczy Park	58
Brockton, Parkdale Village, Exhibition Place	24
Business reply mail Processing Centre, South Ce…	17
CN Tower, King and Spadina, Railway Lands, Harb…	16

```
Central Bay Street                                          66
Christie                                                    17
Church and Wellesley                                        83
Commerce Court, Victoria Hotel                             100
Davisville                                                  33
Davisville North                                             9
Dufferin, Dovercourt Village                                15
First Canadian Place, Underground city                     100
Forest Hill North &amp;amp; West, Forest Hill Road …         4
Garden District, Ryerson                                   100
Harbourfront East, Union Station, Toronto Islands          100
High Park, The Junction South                               24
India Bazaar, The Beaches West                              20
Kensington Market, Chinatown, Grange Park                   60
Lawrence Park                                                3
Little Portugal, Trinity                                    44
Moore Park, Summerhill East                                  2
North Toronto West,  Lawrence Park                          20
Parkdale, Roncesvalles                                      15
Queen's Park, Ontario Provincial Government                 32
Regent Park, Harbourfront                                   47
Richmond, Adelaide, King                                    93
Rosedale                                                     4
Roselawn                                                     1
Runnymede, Swansea                                          34
St. James Town                                              80
St. James Town, Cabbagetown                                 47
Stn A PO Boxes                                              97
Studio District                                             41
Summerhill West, Rathnelly, South Hill, Forest …            16
The Annex, North Midtown, Yorkville                         20
The Beaches                                                  6
The Danforth West, Riverdale                                43
Toronto Dominion Centre, Design Exchange                   100
University of Toronto, Harbord                              36
```

[98]:
```
#unique categogries that can be curated from all returned values:

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

```
There are 233 uniques categories.
```

1.8.1 Analyze Each Neighborhood in Toronto

[99]:
```
toronto_venues['Venue Category']= np.where(toronto_venues['Venue Category'] ==␣
↪'Neighborhood','Neighborhood1',toronto_venues['Venue Category'])
toronto_venues[toronto_venues['Venue Category'] == 'Neighborhood1']
```

[99]:
```
                                        Neighborhood  Neighborhood Latitude  \
263                                      The Beaches              43.676357
413                          Richmond, Adelaide, King              43.650571
514  Harbourfront East, Union Station, Toronto Islands              43.640816
954                                  Studio District              43.659526

     Neighborhood Longitude             Venue  Venue Latitude  \
263              -79.293031     Upper Beaches       43.680563
413              -79.384568  Downtown Toronto       43.653232
514              -79.381752      Harbourfront       43.639526
954              -79.340923       Leslieville       43.662070

     Venue Longitude Venue Category
263       -79.292869  Neighborhood1
413       -79.385296  Neighborhood1
514       -79.380688  Neighborhood1
954       -79.337856  Neighborhood1
```

[100]:
```
# one hot encoding
toronto_onehot = pd.get_dummies(toronto_venues[['Venue Category']], prefix="",␣
↪prefix_sep="")
#toronto_onehot.info(verbose=True)
# add neighborhood column back to dataframe
toronto_onehot['Neighborhood'] = toronto_venues['Neighborhood']

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

toronto_onehot.head()

#toronto_onehot.info(verbose=True)
```

[100]:
```
                Neighborhood  Afghan Restaurant  Airport  Airport Food Court  \
0  Regent Park, Harbourfront                  0        0                   0
1  Regent Park, Harbourfront                  0        0                   0
2  Regent Park, Harbourfront                  0        0                   0
3  Regent Park, Harbourfront                  0        0                   0
4  Regent Park, Harbourfront                  0        0                   0

   Airport Gate  Airport Lounge  Airport Service  Airport Terminal  \
```

```
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

   American Restaurant  Antique Shop  …  Toy / Game Store  Trail  \
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

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

   Vietnamese Restaurant  Wine Bar  Wine Shop  Women's Store  Yoga Studio
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

[5 rows x 234 columns]
```

[101]:
```
toronto_onehot.shape
```

[101]:
```
(1627, 234)
```

[102]:
```
# group rows by neighborhood and by taking the mean of the frequency of␣
 ↪occurrence of each category

toronto_grouped = toronto_onehot.groupby('Neighborhood').mean().reset_index()
toronto_grouped
```

[102]:
```
                                      Neighborhood  Afghan Restaurant  \
0                                      Berczy Park           0.000000
1          Brockton, Parkdale Village, Exhibition Place           0.000000
2   Business reply mail Processing Centre, South C…           0.000000
3   CN Tower, King and Spadina, Railway Lands, Har…           0.000000
4                                Central Bay Street           0.000000
5                                          Christie           0.000000
6                           Church and Wellesley           0.012048
```

```
7                      Commerce Court, Victoria Hotel           0.000000
8                                           Davisville          0.000000
9                                    Davisville North           0.000000
10                       Dufferin, Dovercourt Village           0.000000
11              First Canadian Place, Underground city          0.000000
12  Forest Hill North &amp;amp; West, Forest Hill Road…         0.000000
13                             Garden District, Ryerson         0.000000
14  Harbourfront East, Union Station, Toronto Islands           0.000000
15                       High Park, The Junction South          0.000000
16                      India Bazaar, The Beaches West          0.000000
17           Kensington Market, Chinatown, Grange Park          0.000000
18                                       Lawrence Park          0.000000
19                             Little Portugal, Trinity         0.000000
20                          Moore Park, Summerhill East         0.000000
21                  North Toronto West,  Lawrence Park          0.000000
22                               Parkdale, Roncesvalles         0.000000
23         Queen's Park, Ontario Provincial Government          0.000000
24                             Regent Park, Harbourfront        0.000000
25                              Richmond, Adelaide, King        0.000000
26                                            Rosedale          0.000000
27                                            Roselawn          0.000000
28                                 Runnymede, Swansea           0.000000
29                                     St. James Town           0.000000
30                          St. James Town, Cabbagetown         0.000000
31                                       Stn A PO Boxes         0.000000
32                                     Studio District          0.000000
33  Summerhill West, Rathnelly, South Hill, Forest…             0.000000
34                  The Annex, North Midtown, Yorkville         0.000000
35                                        The Beaches           0.000000
36                          The Danforth West, Riverdale        0.000000
37            Toronto Dominion Centre, Design Exchange          0.000000
38                      University of Toronto, Harbord          0.000000

    Airport  Airport Food Court  Airport Gate  Airport Lounge  \
0    0.0000              0.0000        0.0000          0.0000
1    0.0000              0.0000        0.0000          0.0000
2    0.0000              0.0000        0.0000          0.0000
3    0.0625              0.0625        0.0625          0.0625
4    0.0000              0.0000        0.0000          0.0000
5    0.0000              0.0000        0.0000          0.0000
6    0.0000              0.0000        0.0000          0.0000
7    0.0000              0.0000        0.0000          0.0000
8    0.0000              0.0000        0.0000          0.0000
9    0.0000              0.0000        0.0000          0.0000
10   0.0000              0.0000        0.0000          0.0000
11   0.0000              0.0000        0.0000          0.0000
12   0.0000              0.0000        0.0000          0.0000
```

13	0.0000	0.0000	0.0000	0.0000
14	0.0000	0.0000	0.0000	0.0000
15	0.0000	0.0000	0.0000	0.0000
16	0.0000	0.0000	0.0000	0.0000
17	0.0000	0.0000	0.0000	0.0000
18	0.0000	0.0000	0.0000	0.0000
19	0.0000	0.0000	0.0000	0.0000
20	0.0000	0.0000	0.0000	0.0000
21	0.0000	0.0000	0.0000	0.0000
22	0.0000	0.0000	0.0000	0.0000
23	0.0000	0.0000	0.0000	0.0000
24	0.0000	0.0000	0.0000	0.0000
25	0.0000	0.0000	0.0000	0.0000
26	0.0000	0.0000	0.0000	0.0000
27	0.0000	0.0000	0.0000	0.0000
28	0.0000	0.0000	0.0000	0.0000
29	0.0000	0.0000	0.0000	0.0000
30	0.0000	0.0000	0.0000	0.0000
31	0.0000	0.0000	0.0000	0.0000
32	0.0000	0.0000	0.0000	0.0000
33	0.0000	0.0000	0.0000	0.0000
34	0.0000	0.0000	0.0000	0.0000
35	0.0000	0.0000	0.0000	0.0000
36	0.0000	0.0000	0.0000	0.0000
37	0.0000	0.0000	0.0000	0.0000
38	0.0000	0.0000	0.0000	0.0000

	Airport Service	Airport Terminal	American Restaurant	Antique Shop	…	\
0	0.0000	0.000	0.000000	0.000000	…	
1	0.0000	0.000	0.000000	0.000000	…	
2	0.0000	0.000	0.000000	0.000000	…	
3	0.1875	0.125	0.000000	0.000000	…	
4	0.0000	0.000	0.000000	0.000000	…	
5	0.0000	0.000	0.000000	0.000000	…	
6	0.0000	0.000	0.012048	0.000000	…	
7	0.0000	0.000	0.040000	0.000000	…	
8	0.0000	0.000	0.000000	0.000000	…	
9	0.0000	0.000	0.000000	0.000000	…	
10	0.0000	0.000	0.000000	0.000000	…	
11	0.0000	0.000	0.030000	0.000000	…	
12	0.0000	0.000	0.000000	0.000000	…	
13	0.0000	0.000	0.000000	0.000000	…	
14	0.0000	0.000	0.000000	0.000000	…	
15	0.0000	0.000	0.000000	0.041667	…	
16	0.0000	0.000	0.000000	0.000000	…	
17	0.0000	0.000	0.000000	0.000000	…	
18	0.0000	0.000	0.000000	0.000000	…	

19	0.0000	0.000	0.000000	0.000000	…
20	0.0000	0.000	0.000000	0.000000	…
21	0.0000	0.000	0.000000	0.000000	…
22	0.0000	0.000	0.000000	0.000000	…
23	0.0000	0.000	0.000000	0.000000	…
24	0.0000	0.000	0.000000	0.021277	…
25	0.0000	0.000	0.021505	0.000000	…
26	0.0000	0.000	0.000000	0.000000	…
27	0.0000	0.000	0.000000	0.000000	…
28	0.0000	0.000	0.000000	0.000000	…
29	0.0000	0.000	0.037500	0.000000	…
30	0.0000	0.000	0.021277	0.000000	…
31	0.0000	0.000	0.010309	0.010309	…
32	0.0000	0.000	0.048780	0.000000	…
33	0.0000	0.000	0.062500	0.000000	…
34	0.0000	0.000	0.000000	0.000000	…
35	0.0000	0.000	0.000000	0.000000	…
36	0.0000	0.000	0.023256	0.000000	…
37	0.0000	0.000	0.030000	0.000000	…
38	0.0000	0.000	0.000000	0.000000	…

	Toy / Game Store	Trail	Train Station	Vegetarian / Vegan Restaurant	\
0	0.000000	0.000000	0.00	0.017241	
1	0.000000	0.000000	0.00	0.000000	
2	0.000000	0.000000	0.00	0.000000	
3	0.000000	0.000000	0.00	0.000000	
4	0.000000	0.000000	0.00	0.015152	
5	0.000000	0.000000	0.00	0.000000	
6	0.000000	0.000000	0.00	0.000000	
7	0.000000	0.000000	0.00	0.020000	
8	0.060606	0.000000	0.00	0.000000	
9	0.000000	0.000000	0.00	0.000000	
10	0.000000	0.000000	0.00	0.000000	
11	0.000000	0.000000	0.01	0.010000	
12	0.000000	0.250000	0.00	0.000000	
13	0.000000	0.000000	0.00	0.000000	
14	0.000000	0.000000	0.01	0.010000	
15	0.000000	0.000000	0.00	0.000000	
16	0.000000	0.000000	0.00	0.000000	
17	0.000000	0.000000	0.00	0.050000	
18	0.000000	0.000000	0.00	0.000000	
19	0.000000	0.000000	0.00	0.045455	
20	0.000000	0.500000	0.00	0.000000	
21	0.000000	0.000000	0.00	0.000000	
22	0.000000	0.000000	0.00	0.000000	
23	0.000000	0.000000	0.00	0.000000	
24	0.000000	0.000000	0.00	0.000000	

25	0.000000	0.000000	0.00	0.010753
26	0.000000	0.250000	0.00	0.000000
27	0.000000	0.000000	0.00	0.000000
28	0.000000	0.000000	0.00	0.029412
29	0.000000	0.000000	0.00	0.012500
30	0.000000	0.000000	0.00	0.000000
31	0.000000	0.000000	0.00	0.010309
32	0.000000	0.000000	0.00	0.000000
33	0.000000	0.000000	0.00	0.000000
34	0.000000	0.000000	0.00	0.050000
35	0.000000	0.166667	0.00	0.000000
36	0.000000	0.023256	0.00	0.000000
37	0.000000	0.000000	0.01	0.010000
38	0.000000	0.000000	0.00	0.000000

	Video Game Store	Vietnamese Restaurant	Wine Bar	Wine Shop	\
0	0.000000	0.000000	0.000000	0.000000	
1	0.000000	0.000000	0.000000	0.000000	
2	0.000000	0.000000	0.000000	0.000000	
3	0.000000	0.000000	0.000000	0.000000	
4	0.000000	0.000000	0.015152	0.000000	
5	0.000000	0.000000	0.000000	0.000000	
6	0.000000	0.000000	0.000000	0.012048	
7	0.000000	0.000000	0.010000	0.000000	
8	0.000000	0.000000	0.000000	0.000000	
9	0.000000	0.000000	0.000000	0.000000	
10	0.000000	0.000000	0.000000	0.000000	
11	0.000000	0.000000	0.010000	0.000000	
12	0.000000	0.000000	0.000000	0.000000	
13	0.010000	0.010000	0.010000	0.000000	
14	0.000000	0.000000	0.010000	0.000000	
15	0.000000	0.000000	0.000000	0.000000	
16	0.000000	0.000000	0.000000	0.000000	
17	0.000000	0.050000	0.016667	0.000000	
18	0.000000	0.000000	0.000000	0.000000	
19	0.000000	0.022727	0.022727	0.000000	
20	0.000000	0.000000	0.000000	0.000000	
21	0.000000	0.000000	0.000000	0.000000	
22	0.000000	0.000000	0.000000	0.000000	
23	0.000000	0.000000	0.000000	0.000000	
24	0.000000	0.000000	0.000000	0.021277	
25	0.000000	0.000000	0.000000	0.000000	
26	0.000000	0.000000	0.000000	0.000000	
27	0.000000	0.000000	0.000000	0.000000	
28	0.000000	0.000000	0.000000	0.000000	
29	0.000000	0.000000	0.012500	0.000000	
30	0.000000	0.000000	0.000000	0.000000	

31	0.000000	0.000000	0.000000	0.000000
32	0.000000	0.000000	0.024390	0.000000
33	0.000000	0.062500	0.000000	0.000000
34	0.000000	0.000000	0.000000	0.000000
35	0.000000	0.000000	0.000000	0.000000
36	0.000000	0.000000	0.000000	0.000000
37	0.000000	0.000000	0.010000	0.000000
38	0.027778	0.000000	0.000000	0.000000

	Women's Store	Yoga Studio
0	0.000000	0.000000
1	0.000000	0.041667
2	0.000000	0.058824
3	0.000000	0.000000
4	0.000000	0.015152
5	0.000000	0.000000
6	0.000000	0.024096
7	0.000000	0.000000
8	0.000000	0.000000
9	0.000000	0.000000
10	0.000000	0.000000
11	0.000000	0.000000
12	0.000000	0.000000
13	0.000000	0.000000
14	0.000000	0.000000
15	0.000000	0.000000
16	0.000000	0.000000
17	0.000000	0.000000
18	0.000000	0.000000
19	0.000000	0.022727
20	0.000000	0.000000
21	0.000000	0.050000
22	0.000000	0.000000
23	0.000000	0.031250
24	0.000000	0.021277
25	0.010753	0.000000
26	0.000000	0.000000
27	0.000000	0.000000
28	0.000000	0.029412
29	0.000000	0.000000
30	0.000000	0.000000
31	0.000000	0.010309
32	0.000000	0.024390
33	0.000000	0.000000
34	0.000000	0.000000
35	0.000000	0.000000
36	0.000000	0.023256

```
37       0.000000     0.000000
38       0.000000     0.027778

[39 rows x 234 columns]
```

[103]:
```
toronto_grouped.shape
```

[103]:
```
(39, 234)
```

Let's put that into a *pandas* dataframe First, let's write a function to sort the venues in descending order.

[104]:
```
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]
```

Now let's create the new dataframe and display the top 10 venues for each neighborhood.

[105]:
```
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
toronto_venues_sorted = pd.DataFrame(columns=columns)
toronto_venues_sorted['Neighborhood'] = toronto_grouped['Neighborhood']

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

toronto_venues_sorted.head()
```

[105]:
```
                                  Neighborhood 1st Most Common Venue  \
0                                  Berczy Park           Coffee Shop
1      Brockton, Parkdale Village, Exhibition Place                  Café
2  Business reply mail Processing Centre, South C…          Yoga Studio
3  CN Tower, King and Spadina, Railway Lands, Har…      Airport Service
```

```
4                                 Central Bay Street           Coffee Shop

   2nd Most Common Venue 3rd Most Common Venue 4th Most Common Venue  \
0           Cocktail Bar            Restaurant              Beer Bar
1  Performing Arts Venue       Breakfast Spot           Coffee Shop
2          Auto Workshop        Garden Center  Gym / Fitness Center
3       Airport Terminal     Sculpture Garden   Rental Car Location
4         Sandwich Place   Italian Restaurant   Japanese Restaurant

  5th Most Common Venue 6th Most Common Venue 7th Most Common Venue  \
0    Seafood Restaurant           Cheese Shop                Bakery
1          Yoga Studio                   Gym             Pet Store
2  Fast Food Restaurant       Farmers Market    Light Rail Station
3                 Plane          Coffee Shop         Boat or Ferry
4                 Café         Burger Joint      Department Store

  8th Most Common Venue 9th Most Common Venue 10th Most Common Venue
0                 Café              Pharmacy          Shopping Mall
1            Nightclub    Italian Restaurant           Intersection
2           Comic Shop           Pizza Place       Recording Studio
3       Harbor / Marina        Airport Lounge           Airport Gate
4           Salad Place       Thai Restaurant        Bubble Tea Shop
```

1.9 Cluster Neighborhoods in Toronto

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

[106]:
```
# set number of clusters
kclusters = 5

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

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

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

[106]:
```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0], dtype=int32)
```

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

[107]:
```
# add clustering labels
toronto_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)

toronto_merged = ATor_data
```

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

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

[107]:

	Postal Code	Borough	Neighborhood	Latitude	Longitude	Cluster Labels	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	M5A	Downtown Toronto	Regent Park, Harbourfront	43.654260	-79.360636	0	Coffee Shop	Park	Pub	Bakery	Theater	Café	Restaurant	Breakfast Spot	Event Space	Hotel
1	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.662301	-79.389494	0	Coffee Shop	Sushi Restaurant	Bank	Beer Bar	Smoothie Shop	Sandwich Place	Burrito Place	Restaurant	Café	Park
2	M5B	Downtown Toronto	Garden District, Ryerson	43.657162	-79.378937	0	Clothing Store	Coffee Shop	Bubble Tea Shop	Cosmetics Shop	Japanese Restaurant	Italian Restaurant	Middle Eastern Restaurant	Café	Bookstore	Bakery
3	M5C	Downtown Toronto	St. James Town	43.651494	-79.375418	0	Coffee Shop	Café	Cocktail Bar	Gastropub	American Restaurant	Moroccan Restaurant	Cosmetics Shop	Creperie	Department Store	Clothing Store
4	M4E	East Toronto	The Beaches	43.676357	-79.293031	3	Health Food Store	Asian Restaurant	Pizza Place	Pub	Trail	Neighborhood1	Distribution Center	Dessert Shop	Dim Sum Restaurant	Diner

Finally, let's visualize the resulting clusters

[108]:
```
# Matplotlib and associated plotting modules
import matplotlib.cm as cm
```

```
import matplotlib.colors as colors

# 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_merged['Latitude'],␣
 ↪toronto_merged['Longitude'], toronto_merged['Neighborhood'],␣
 ↪toronto_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
```

[108]: <folium.folium.Map at 0x7f49ac1889b0>

1.10 Examine Clusters in Toronto

Cluster 1

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

[109]:

	Borough	Cluster Labels	1st Most Common Venue	\
0	Downtown Toronto	0	Coffee Shop	
1	Downtown Toronto	0	Coffee Shop	
2	Downtown Toronto	0	Clothing Store	
3	Downtown Toronto	0	Coffee Shop	
5	Downtown Toronto	0	Coffee Shop	
6	Downtown Toronto	0	Coffee Shop	
7	Downtown Toronto	0	Grocery Store	
8	Downtown Toronto	0	Coffee Shop	
9	West Toronto	0	Pharmacy	
10	Downtown Toronto	0	Coffee Shop	

```
11      West Toronto               0                   Bar
12      East Toronto               0      Greek Restaurant
13  Downtown Toronto               0           Coffee Shop
14      West Toronto               0                 Café
15      East Toronto               0           Pizza Place
16  Downtown Toronto               0           Coffee Shop
17      East Toronto               0                 Café
20   Central Toronto               0                  Park
22      West Toronto               0    Mexican Restaurant
23   Central Toronto               0        Clothing Store
24   Central Toronto               0        Sandwich Place
25      West Toronto               0             Gift Shop
26   Central Toronto               0        Sandwich Place
27  Downtown Toronto               0                 Café
28      West Toronto               0           Coffee Shop
30  Downtown Toronto               0                 Café
31   Central Toronto               0                  Pub
32  Downtown Toronto               0       Airport Service
34  Downtown Toronto               0           Coffee Shop
35  Downtown Toronto               0                 Café
36  Downtown Toronto               0           Coffee Shop
37  Downtown Toronto               0           Coffee Shop
38      East Toronto               0           Yoga Studio

    2nd Most Common Venue 3rd Most Common Venue  \
0                    Park                   Pub
1       Sushi Restaurant                  Bank
2            Coffee Shop       Bubble Tea Shop
3                   Café          Cocktail Bar
5             Cocktail Bar            Restaurant
6           Sandwich Place    Italian Restaurant
7                   Café                  Park
8                   Café            Restaurant
9                 Bakery         Grocery Store
10              Aquarium                 Café
11           Restaurant                  Café
12           Coffee Shop    Italian Restaurant
13                 Hotel                 Café
14  Performing Arts Venue        Breakfast Spot
15   Fast Food Restaurant        Ice Cream Shop
16                  Café            Restaurant
17                Bakery           Coffee Shop
20           Pizza Place        Sandwich Place
22                  Café       Thai Restaurant
23           Coffee Shop           Yoga Studio
24                  Café           Coffee Shop
25         Breakfast Spot          Dessert Shop
```

```
26           Dessert Shop                   Gym
27                Bakery                   Bar
28                  Café      Sushi Restaurant
30           Coffee Shop    Mexican Restaurant
31           Coffee Shop            Restaurant
32      Airport Terminal      Sculpture Garden
34                  Café    Seafood Restaurant
35           Coffee Shop           Pizza Place
36                  Café                 Hotel
37   Japanese Restaurant      Sushi Restaurant
38         Auto Workshop         Garden Center

          4th Most Common Venue        5th Most Common Venue  \
0                        Bakery                      Theater
1                      Beer Bar               Smoothie Shop
2                Cosmetics Shop          Japanese Restaurant
3                     Gastropub          American Restaurant
5                      Beer Bar           Seafood Restaurant
6            Japanese Restaurant                        Café
7             Athletics & Sports           Italian Restaurant
8                  Deli / Bodega                       Hotel
9                    Supermarket    Middle Eastern Restaurant
10                         Hotel               Scenic Lookout
11  Vegetarian / Vegan Restaurant                  Coffee Shop
12                     Bookstore           Frozen Yogurt Shop
13                    Restaurant           Seafood Restaurant
14                   Coffee Shop                  Yoga Studio
15             Fish & Chips Shop             Sushi Restaurant
16                         Hotel          American Restaurant
17                     Gastropub                      Brewery
20             Food & Drink Shop             Department Store
22                         Bakery                    Speakeasy
23           Sporting Goods Shop       Furniture / Home Store
24                          Park               History Museum
25                 Movie Theater  Eastern European Restaurant
26            Italian Restaurant                        Café
27            Italian Restaurant          Japanese Restaurant
28            Italian Restaurant                 Pizza Place
30         Vietnamese Restaurant                      Bakery
31                   Pizza Place                 Supermarket
32           Rental Car Location                       Plane
34                         Hotel           Italian Restaurant
35            Italian Restaurant           Chinese Restaurant
36                    Restaurant                         Gym
37                    Restaurant                     Gay Bar
38          Gym / Fitness Center         Fast Food Restaurant
```

	6th Most Common Venue	7th Most Common Venue
0	Café	Restaurant
1	Sandwich Place	Burrito Place
2	Italian Restaurant	Middle Eastern Restaurant
3	Moroccan Restaurant	Cosmetics Shop
5	Cheese Shop	Bakery
6	Burger Joint	Department Store
7	Diner	Restaurant
8	Gym	Thai Restaurant
9	Music Venue	Pizza Place
10	Fried Chicken Joint	Sporting Goods Shop
11	Asian Restaurant	Men's Store
12	Ice Cream Shop	Furniture / Home Store
13	Salad Place	Japanese Restaurant
14	Gym	Pet Store
15	Brewery	Food & Drink Shop
16	Gym	Seafood Restaurant
17	American Restaurant	Yoga Studio
20	Hotel	Breakfast Spot
22	Bookstore	Restaurant
23	Fast Food Restaurant	Diner
24	Liquor Store	Burger Joint
25	Dog Run	Italian Restaurant
26	Pizza Place	Toy / Game Store
27	Theater	Restaurant
28	Pub	Yoga Studio
30	Vegetarian / Vegan Restaurant	Grocery Store
31	Sushi Restaurant	Bank
32	Coffee Shop	Boat or Ferry
34	Cocktail Bar	Beer Bar
35	Bakery	Pet Store
36	Salad Place	Japanese Restaurant
37	Café	Pub
38	Farmers Market	Light Rail Station

	8th Most Common Venue	9th Most Common Venue
0	Breakfast Spot	Event Space
1	Restaurant	Café
2	Café	Bookstore
3	Creperie	Department Store
5	Café	Pharmacy
6	Salad Place	Thai Restaurant
7	Baby Store	Candy Store
8	Bookstore	Sushi Restaurant
9	Recording Studio	Café
10	Brewery	Restaurant
11	Cuban Restaurant	Brewery

12	Yoga Studio	Pub	
13	American Restaurant	Italian Restaurant	
14	Nightclub	Italian Restaurant	
15	Restaurant	Italian Restaurant	
16	Japanese Restaurant	Italian Restaurant	
17	Comfort Food Restaurant	Sandwich Place	
20	Gym	Gym / Fitness Center	
22	Cajun / Creole Restaurant	Music Venue	
23	Mexican Restaurant	Cosmetics Shop	
24	Indian Restaurant	Middle Eastern Restaurant	
25	Bar	Bank	
26	Sushi Restaurant	Coffee Shop	
27	Bookstore	Pub	
28	Bar	Fish & Chips Shop	
30	Bar	Pizza Place	
31	Sports Bar	Fried Chicken Joint	
32	Harbor / Marina	Airport Lounge	
34	Restaurant	Japanese Restaurant	
35	Restaurant	Pub	
36	Seafood Restaurant	Steakhouse	
37	Men's Store	Mediterranean Restaurant	
38	Comic Shop	Pizza Place	

	10th Most Common Venue
0	Hotel
1	Park
2	Bakery
3	Clothing Store
5	Shopping Mall
6	Bubble Tea Shop
7	Nightclub
8	Cosmetics Shop
9	Brewery
10	Italian Restaurant
11	Record Shop
12	Pizza Place
13	Concert Hall
14	Intersection
15	Pub
16	Cocktail Bar
17	Cheese Shop
20	Dessert Shop
22	Grocery Store
23	Chinese Restaurant
24	BBQ Joint
25	Restaurant
26	Dance Studio

```
27           Dessert Shop
28    Indie Movie Theater
30                   Park
31             Bagel Shop
32           Airport Gate
34                   Park
35          Grocery Store
36    American Restaurant
37                  Hotel
38       Recording Studio
```

Cluster 2

[110]:
```
toronto_merged.loc[toronto_merged['Cluster Labels'] == 1, toronto_merged.
↪columns[[1] + list(range(5, toronto_merged.shape[1]))]]
```

[110]:
```
            Borough  Cluster Labels 1st Most Common Venue  \
19  Central Toronto               1                Garden

   2nd Most Common Venue 3rd Most Common Venue 4th Most Common Venue  \
19           Yoga Studio         Deli / Bodega     Electronics Store

          5th Most Common Venue 6th Most Common Venue 7th Most Common Venue  \
19  Eastern European Restaurant   Dumpling Restaurant            Donut Shop

   8th Most Common Venue 9th Most Common Venue 10th Most Common Venue
19      Doner Restaurant               Dog Run    Distribution Center
```

Cluster 3

[111]:
```
toronto_merged.loc[toronto_merged['Cluster Labels'] == 2, toronto_merged.
↪columns[[1] + list(range(5, toronto_merged.shape[1]))]]
```

[111]:
```
            Borough  Cluster Labels 1st Most Common Venue  \
29  Central Toronto               2                  Gym

   2nd Most Common Venue 3rd Most Common Venue 4th Most Common Venue  \
29                 Trail           Yoga Studio         Deli / Bodega

   5th Most Common Venue        6th Most Common Venue 7th Most Common Venue  \
29     Electronics Store  Eastern European Restaurant   Dumpling Restaurant

   8th Most Common Venue 9th Most Common Venue 10th Most Common Venue
29            Donut Shop      Doner Restaurant                Dog Run
```

Cluster 4

[112]:
```
toronto_merged.loc[toronto_merged['Cluster Labels'] == 3, toronto_merged.
↪columns[[1] + list(range(5, toronto_merged.shape[1]))]]
```

[112]:
```
            Borough  Cluster Labels 1st Most Common Venue  \
4      East Toronto               3     Health Food Store
18  Central Toronto               3                  Park
21  Central Toronto               3                  Park

   2nd Most Common Venue 3rd Most Common Venue 4th Most Common Venue  \
4       Asian Restaurant           Pizza Place                   Pub
18              Bus Line           Swim School  Ethiopian Restaurant
21         Jewelry Store                 Trail      Sushi Restaurant

   5th Most Common Venue        6th Most Common Venue  \
4                  Trail                Neighborhood1
18     Electronics Store  Eastern European Restaurant
21           Yoga Studio             Department Store

          7th Most Common Venue 8th Most Common Venue 9th Most Common Venue  \
4           Distribution Center          Dessert Shop    Dim Sum Restaurant
18          Dumpling Restaurant            Donut Shop      Doner Restaurant
21  Eastern European Restaurant   Dumpling Restaurant             Donut Shop

   10th Most Common Venue
4                   Diner
18                Dog Run
21       Doner Restaurant
```

Cluster 5

[113]:
```
toronto_merged.loc[toronto_merged['Cluster Labels'] == 4, toronto_merged.
↪columns[[1] + list(range(5, toronto_merged.shape[1]))]]
```

[113]:
```
            Borough  Cluster Labels 1st Most Common Venue  \
33  Downtown Toronto               4                  Park

   2nd Most Common Venue 3rd Most Common Venue 4th Most Common Venue  \
33            Playground                 Trail         Deli / Bodega

   5th Most Common Venue        6th Most Common Venue 7th Most Common Venue  \
33     Electronics Store  Eastern European Restaurant   Dumpling Restaurant

   8th Most Common Venue 9th Most Common Venue 10th Most Common Venue
33            Donut Shop      Doner Restaurant                Dog Run
```

1.11 Results

It can be seen from the clustering of the Toronto data, only the Clusters 1 shows interesting venues indicating a good placement for a new hotel.

1.12 Discussion

From the above methodologies of analyzing the neighborhoods using exploratory data analysis and the processing of data using k-means clustering, it can be seen that the Toronto city has more interesting venues in a closed cluster, whereas New York city has interesting venues in varied clusters located through many neighborhoods of the borough of Manhattan

1.13 Conclusion:

It appears from the process of clustering on the two neighborhoods of New York City and Toronto, the number of locations of interest or interesting venues are found to be very high for Cluster 1 of Toronto compared to Clusters 1 through 4 for New York city. Hence, if a new hotel is to be built, it is preferable to house its location in the Cluster 1 of Toronto. However, since several clusters have good amount of locations of interest for New York City, a new hotel may be housed in either of these clusters.

[ ]: