Battle of Neighbourhoods

1. Introduction

1.1 Description of the Problem

New York is known as a state of diverse population. That explains that there are a lot of restaurants with food from different parts of the world. The problem starts with finding a suitable neighborhood place to open a Latin food restaurant.

2.2 Discussion of the Background

A customer wants to open a Latin food restaurant in the Bronx district of New York City. Since the district is wide, he wants to know a specific neighborhood which can be conducive to opening his business.

2. Description of data

In this project we will use a public database used in previous examples seen in previous modules and Foursquare data

First, libraries and tools necessary for the project are imported

In [1]:

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

import pandas as pd # library for data analsysis
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

import json # library to handle JSON files

!conda install -c conda-forge geopy --yes # uncomment this line if you haven't complete
d the Foursquare API lab
from geopy.geocoders import Nominatim # convert an address into latitude and longitude
 values

import requests # library to handle requests
from pandas.io.json import json_normalize # tranform JSON file into a pandas dataframe

# Matplotlib and associated plotting modules
import matplotlib.cm as cm
import matplotlib.colors as colors
import matplotlib.pyplot as plt
# import k-means from clustering stage
from sklearn.cluster import KMeans

#!conda install -c conda-forge folium=0.5.0 --yes # uncomment this line if you haven't
 completed the Foursquare API lab
import folium # map rendering library

print('Libraries imported.')
```

```
Solving environment: done


==> WARNING: A newer version of conda exists. <==
  current version: 4.5.11
  latest version: 4.7.12

Please update conda by running

    $ conda update -n base -c defaults conda



## Package Plan ##

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

  added / updated specs:
    - geopy


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    scikit-learn-0.20.1        |   py36h22eb022_0         5.7 MB
    liblapack-3.8.0            |      11_openblas          10 KB  conda-fo
rge
    scipy-1.3.2                |   py36h921218d_0        18.0 MB  conda-fo
rge
    geographiclib-1.50         |             py_0          34 KB  conda-fo
rge
    libopenblas-0.3.6          |       h5a2b251_2         7.7 MB
    liblapacke-3.8.0           |      11_openblas          10 KB  conda-fo
rge
    numpy-1.17.3               |   py36h95a1406_0         5.2 MB  conda-fo
rge
    libcblas-3.8.0             |      11_openblas          10 KB  conda-fo
rge
    libblas-3.8.0              |      11_openblas          10 KB  conda-fo
rge
    geopy-1.20.0               |             py_0          57 KB  conda-fo
rge
    blas-2.11                  |         openblas          10 KB  conda-fo
rge
    ------------------------------------------------------------
                                           Total:        36.8 MB

The following NEW packages will be INSTALLED:

    geographiclib: 1.50-py_0                              conda-forge
    geopy:         1.20.0-py_0                            conda-forge
    libblas:       3.8.0-11_openblas                      conda-forge
    libcblas:      3.8.0-11_openblas                      conda-forge
    liblapack:     3.8.0-11_openblas                      conda-forge
    liblapacke:    3.8.0-11_openblas                      conda-forge
    libopenblas:   0.3.6-h5a2b251_2

The following packages will be UPDATED:

    blas:          1.1-openblas                           conda-forge -->
```

```
2.11-openblas         conda-forge
    numpy:         1.16.2-py36_blas_openblash1522bff_0    conda-forge [bla
s_openblas] --> 1.17.3-py36h95a1406_0 conda-forge
    scipy:         1.2.1-py36_blas_openblash1522bff_0     conda-forge [bla
s_openblas] --> 1.3.2-py36h921218d_0  conda-forge

The following packages will be DOWNGRADED:

    scikit-learn:  0.20.1-py36_blas_openblashebff5e3_1200 conda-forge [bla
s_openblas] --> 0.20.1-py36h22eb022_0


Downloading and Extracting Packages
scikit-learn-0.20.1  | 5.7 MB    | ##################################### |
100%
liblapack-3.8.0      | 10 KB     | ##################################### |
100%
scipy-1.3.2          | 18.0 MB   | ##################################### |
100%
geographiclib-1.50   | 34 KB     | ##################################### |
100%
libopenblas-0.3.6    | 7.7 MB    | ##################################### |
100%
liblapacke-3.8.0     | 10 KB     | ##################################### |
100%
numpy-1.17.3         | 5.2 MB    | ##################################### |
100%
libcblas-3.8.0       | 10 KB     | ##################################### |
100%
libblas-3.8.0        | 10 KB     | ##################################### |
100%
geopy-1.20.0         | 57 KB     | ##################################### |
100%
blas-2.11            | 10 KB     | ##################################### |
100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
Libraries imported.
```

2.1 New York data

The New York neighborhood information database is imported and they are sorted by columns in a data table

In [2]:

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

```
Data downloaded!
```

In [3]:

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

In [4]:

```
neighborhoods_data = newyork_data['features']
```

In [5]:

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

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

In [6]:

```
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)
```

Only Bronx district data is filtered in a different data table.

In [7]:

```
bronx_data=neighborhoods[neighborhoods['Borough'] == 'Bronx'].reset_index(drop=True)
bronx_data.head()
```

Out[7]:

	Borough	Neighborhood	Latitude	Longitude
0	Bronx	Wakefield	40.894705	-73.847201
1	Bronx	Co-op City	40.874294	-73.829939
2	Bronx	Eastchester	40.887556	-73.827806
3	Bronx	Fieldston	40.895437	-73.905643
4	Bronx	Riverdale	40.890834	-73.912585

2.2 Foursquare data

Foursware API will be used to get Bronx neighborhoods venues information to be able to determinate de best neighborhood to open a Latin Food restaurant. This data is determined by neighborhoods's latitud and longitude and it is limited to 100 venues and 500 meters of radio

2.3 How data will be used to solve the problem

Both, NY neighborhoods and Foursquare data are going to determine the most suitable neighborhood to open a latin food restaurant by using k-means metodology.

3. Methodology

Firt we have to obtain Foursquare data for Bronx's neighborhoods.

In [8]:

```
address = 'Bronx, NY'

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

```
The geograpical coordinate of Bronx are 40.85048545, -73.8404035580209.
```

In [9]:

```
CLIENT_ID = 'VQJQ4ZOFBMGM0L3J43GXUM52IXIPNXNLX1KGGNY2KXZHSFT0' # your Foursquare ID
CLIENT_SECRET = 'V3GSJ32HU1CFEXBEL5TSEVN43AESJDNSZUCNKZYEA2PCVAHM' # your Foursquare Se
cret
VERSION = '20180605' # Foursquare API version

print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
```

```
Your credentails:
CLIENT_ID: VQJQ4ZOFBMGM0L3J43GXUM52IXIPNXNLX1KGGNY2KXZHSFT0
CLIENT_SECRET:V3GSJ32HU1CFEXBEL5TSEVN43AESJDNSZUCNKZYEA2PCVAHM
```

In [10]:

```
bronx_data.shape
```

Out[10]:

```
(52, 4)
```

In [11]:

```
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)
```

In [12]:

```
bronx_venues = getNearbyVenues(names=bronx_data['Neighborhood'],
                                   latitudes=bronx_data['Latitude'],
                                   longitudes=bronx_data['Longitude']
                                  )
```

```
Wakefield
Co-op City
Eastchester
Fieldston
Riverdale
Kingsbridge
Woodlawn
Norwood
Williamsbridge
Baychester
Pelham Parkway
City Island
Bedford Park
University Heights
Morris Heights
Fordham
East Tremont
West Farms
High  Bridge
Melrose
Mott Haven
Port Morris
Longwood
Hunts Point
Morrisania
Soundview
Clason Point
Throgs Neck
Country Club
Parkchester
Westchester Square
Van Nest
Morris Park
Belmont
Spuyten Duyvil
North Riverdale
Pelham Bay
Schuylerville
Edgewater Park
Castle Hill
Olinville
Pelham Gardens
Concourse
Unionport
Edenwald
Claremont Village
Concourse Village
Mount Eden
Mount Hope
Bronxdale
Allerton
Kingsbridge Heights
```

Now, lets find out which neighborhood has most latin american restaurants

In [13]:

```
bronx_latin=bronx_venues.loc[bronx_venues['Venue Category']=='Latin American Restauran
t'].reset_index()
```

In [14]:

```
bronx_latin.head(18)
```

Out[14]:

	index	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude
0	65	Kingsbridge	40.881687	-73.902818	Leche y Miel	40.883709	-73.901606
1	102	Kingsbridge	40.881687	-73.902818	Silhouette Restaurant & Lounge	40.880706	-73.902687
2	118	Kingsbridge	40.881687	-73.902818	Hola España	40.879160	-73.904553
3	307	University Heights	40.855727	-73.910416	Liberato	40.853744	-73.907966
4	325	Morris Heights	40.847898	-73.919672	Mamajuana	40.844938	-73.920950
5	328	Fordham	40.860997	-73.896427	188 Bakery Cuchifritos	40.861602	-73.898311
6	379	Fordham	40.860997	-73.896427	Parilla Latina	40.861009	-73.891945
7	456	West Farms	40.839475	-73.877745	El Salvadoreño, bar & restaurante	40.840689	-73.872961
8	469	High Bridge	40.836623	-73.926102	Justine Restaurant	40.835502	-73.921439
9	528	Mott Haven	40.806239	-73.916100	Rincon Ecuatoriano	40.803689	-73.911951
10	537	Port Morris	40.801664	-73.913221	Rincon Ecuatoriano	40.803689	-73.911951
11	546	Longwood	40.815099	-73.895788	El Valle Restaurant	40.816113	-73.896310
12	604	Soundview	40.821012	-73.865746	Don Leo	40.818336	-73.862740
13	670	Westchester Square	40.840619	-73.842194	El Bohio Tropical Restaurant	40.840607	-73.843177
14	1032	Unionport	40.829774	-73.850535	Brisas Del Caribe Restaurant	40.832128	-73.851270
15	1036	Unionport	40.829774	-73.850535	Sabrosura	40.831936	-73.851019
16	1138	Mount Eden	40.843826	-73.916556	D Angies Resturant	40.843632	-73.911529

	index	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude
17	1200	Kingsbridge Heights	40.870392	-73.901523	La Caridad	40.869219	-73.903219

In [15]:

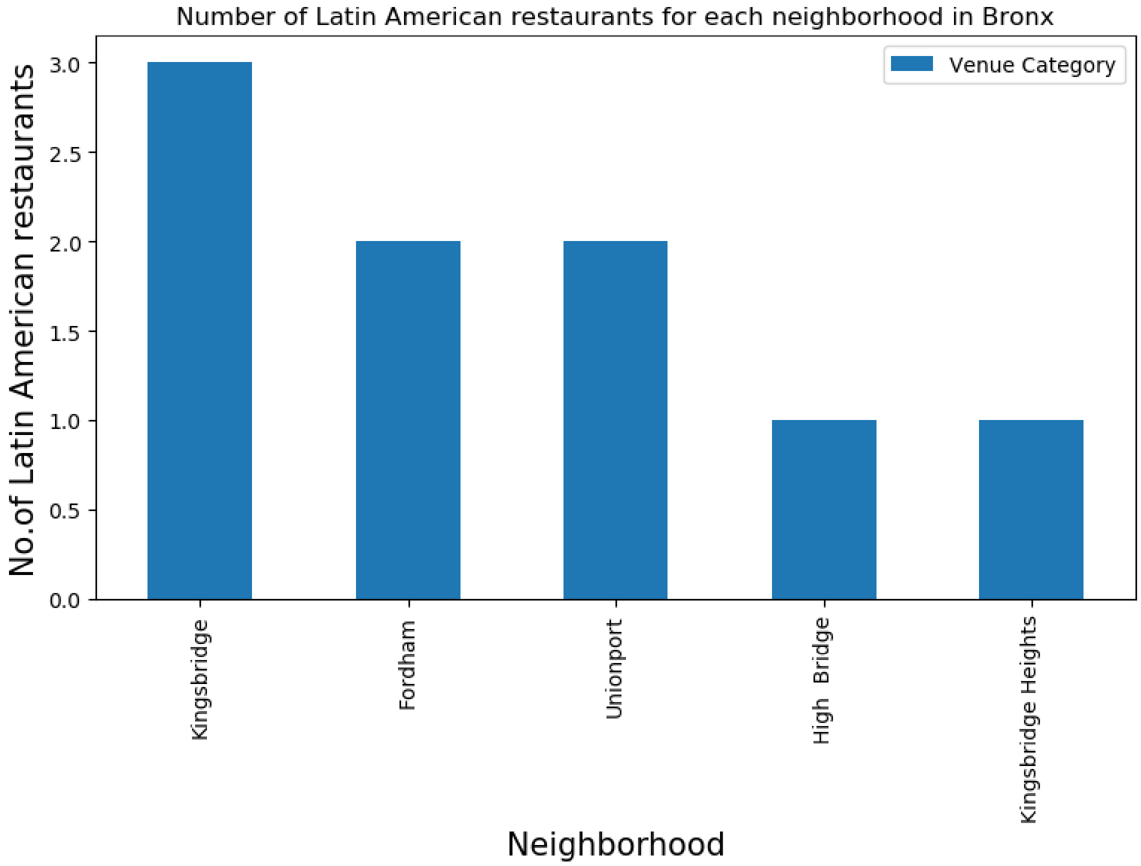
```
bronx_latin.shape
```

Out[15]:

```
(18, 8)
```

In [40]:

```
plt.figure(figsize=(9,5), dpi = 100)
# title
plt.title('Number of Latin American restaurants for each neighborhood in Bronx')
#On x-axis
plt.xlabel('Neighborhood', fontsize = 15)
#On y-axis
plt.ylabel('No.of Latin American restaurants', fontsize=15)
#giving a bar plot
bronx_latin.groupby('Neighborhood')['Venue Category'].count().nlargest(5).plot(kind='ba
r')
#legend
plt.legend()
#displays the plot
plt.show()
```



According to last bar graph the neighborhood with most latin american restaurants is Kingsbridge. Now, lets find out every venue from each neighborhood and it's predominance

In [17]:

```
# one hot encoding
bronx_onehot = pd.get_dummies(bronx_venues[['Venue Category']], prefix="", prefix_sep=
"")

# add neighborhood column back to dataframe
bronx_onehot['Neighborhood'] = bronx_venues['Neighborhood']

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

bronx_onehot.head()
```

Out[17]:

	Neighborhood	Accessories Store	African Restaurant	American Restaurant	Arcade	Arepa Restaurant	Art Gallery	Art Museum
0	Wakefield	0	0	0	0	0	0	0
1	Wakefield	0	0	0	0	0	0	0
2	Wakefield	0	0	0	0	0	0	0
3	Wakefield	0	0	0	0	0	0	0
4	Wakefield	0	0	0	0	0	0	0

In [18]:

```
bronx_grouped = bronx_onehot.groupby('Neighborhood').mean().reset_index()
bronx_grouped
```

Out[18]:

	Neighborhood	Accessories Store	African Restaurant	American Restaurant	Arcade	Arepa Restaurant	Art Gallery	Mu
0	Allerton	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
1	Baychester	0.000000	0.000000	0.045455	0.045455	0.00	0.000000	0.0
2	Bedford Park	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
3	Belmont	0.000000	0.000000	0.010000	0.000000	0.00	0.000000	0.0
4	Bronxdale	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
5	Castle Hill	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
6	City Island	0.000000	0.000000	0.037037	0.000000	0.00	0.037037	0.0
7	Claremont Village	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
8	Clason Point	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
9	Co-op City	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
10	Concourse	0.000000	0.035714	0.000000	0.000000	0.00	0.035714	0.03
11	Concourse Village	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
12	Country Club	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
13	East Tremont	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
14	Eastchester	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
15	Edenwald	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
16	Edgewater Park	0.000000	0.000000	0.041667	0.000000	0.00	0.000000	0.0
17	Fieldston	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
18	Fordham	0.011364	0.011364	0.000000	0.000000	0.00	0.000000	0.0
19	High Bridge	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
20	Hunts Point	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
21	Kingsbridge	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
22	Kingsbridge Heights	0.000000	0.000000	0.029412	0.000000	0.00	0.000000	0.0
23	Longwood	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
24	Melrose	0.000000	0.000000	0.000000	0.000000	0.00	0.041667	0.0
25	Morris Heights	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
26	Morris Park	0.000000	0.000000	0.000000	0.000000	0.04	0.000000	0.0
27	Morrisania	0.000000	0.000000	0.034483	0.000000	0.00	0.000000	0.0
28	Mott Haven	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
29	Mount Eden	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
30	Mount Hope	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
31	North Riverdale	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0
32	Norwood	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0

	Neighborhood	Accessories Store	African Restaurant	American Restaurant	Arcade	Arepa Restaurant	Art Gallery	Mu:
33	Olinville	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0(
34	Parkchester	0.000000	0.000000	0.057143	0.000000	0.00	0.000000	0.0(
35	Pelham Bay	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0(
36	Pelham Gardens	0.000000	0.000000	0.043478	0.000000	0.00	0.000000	0.0(
37	Pelham Parkway	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0(
38	Port Morris	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0(
39	Riverdale	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0(
40	Schuylerville	0.000000	0.000000	0.058824	0.000000	0.00	0.000000	0.0(
41	Soundview	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0(
42	Spuyten Duyvil	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0(
43	Throgs Neck	0.000000	0.000000	0.100000	0.000000	0.00	0.000000	0.0(
44	Unionport	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0(
45	University Heights	0.000000	0.052632	0.000000	0.000000	0.00	0.000000	0.0(
46	Van Nest	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0(
47	Wakefield	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0(
48	West Farms	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0(
49	Westchester Square	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0(
50	Williamsbridge	0.000000	0.000000	0.000000	0.000000	0.00	0.000000	0.0(
51	Woodlawn	0.000000	0.000000	0.041667	0.000000	0.00	0.000000	0.0(

In [19]:

```
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]
```

In [20]:

```
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'] = bronx_grouped['Neighborhood']

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

neighborhoods_venues_sorted.head()
```

Out[20]:

	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 Cor
0	Allerton	Pizza Place	Deli / Bodega	Chinese Restaurant	Supermarket	Fast Food Restaurant	Martial Arts Dojo	Bre
1	Baychester	Electronics Store	Donut Shop	Pizza Place	Bank	Men's Store	Fast Food Restaurant	Ma
2	Bedford Park	Diner	Mexican Restaurant	Pizza Place	Supermarket	Chinese Restaurant	Sandwich Place	S
3	Belmont	Italian Restaurant	Pizza Place	Deli / Bodega	Bakery	Grocery Store	Dessert Shop	
4	Bronxdale	Italian Restaurant	Spanish Restaurant	Bank	Pizza Place	Performing Arts Venue	Paper / Office Supplies Store	Cl Rest

K-MEANS

To apply this method, it is necesaary to find out an appropriate value of K, so we are goinf to aplicate Elbow Method and Silhouette Coefficient.

Elbow Method

The elbow method is used to solve the problem of selecting k. Interestingly, the elbow method is not perfect either but it gives significant insight that is perhaps not top optimal but sub-optimal to choosing the optimal number of clusters by fitting the model with a range of values for k.

The approach for this is to run the k-means clustering for a range of value k and for each value of k, the Sum of the Squared Errors (SSE) is calculated., calculate sum of squared errors (SSE). When this is done, a plot of k and the corresponding SSEs are then made. At the elbow (just like arm), that is where the optimal value of k is. And that will be the number of clusters to be used. The whole idea is to have minimum SSE

In [24]:

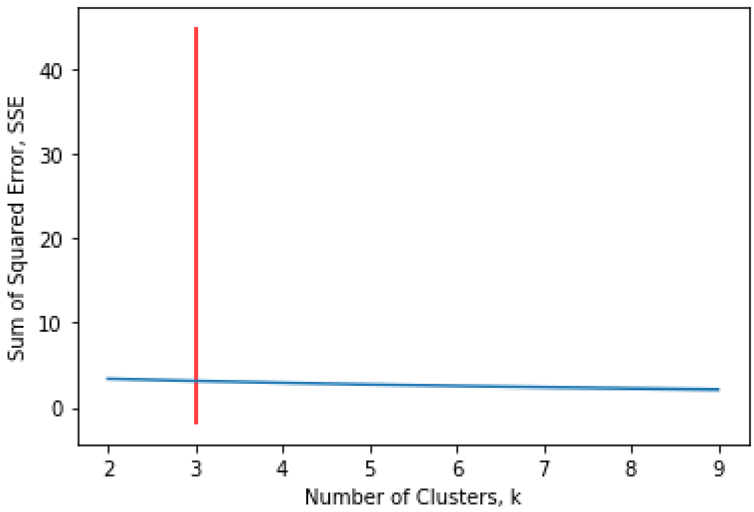
```
bronx_grouped_clustering =bronx_grouped.drop('Neighborhood', 1)
```

In [25]:

```
from sklearn.cluster import KMeans

# SSE is initialize with empty values
# n_clusters is the "k"
sse = {}
for n_cluster1 in range(2, 10):
    kmeans1 = KMeans(n_clusters = n_cluster1, max_iter = 500).fit(bronx_grouped_cluster
ing)
    bronx_grouped_clustering["clusters"] = kmeans1.labels_

    # The inertia is the sum of distances of samples to their closest cluster centre
    sse[n_cluster1] = kmeans1.inertia_
plt.figure()
plt.plot(list(sse.keys()), list(sse.values()))
plt.xlabel("Number of Clusters, k")
plt.ylabel("Sum of Squared Error, SSE")
# vertical line
plt.vlines(3, ymin = -2, ymax = 45, colors = 'red')
plt.show()
```



Depending on the number of iteration (in this case, 500 iterations were used), the number of cluster, k is 3.

Silhouette Coefficient

To find the optimal value of the number of clusters, k, the number of clusters is iterated corresponding Silhouette Coefficientis calculated for each of the k-values used. The highest Silhouette Coefficient gives the best match to its own cluster. Please see below:

In [26]:

```
from sklearn.metrics import silhouette_score
from sklearn.cluster import KMeans

for n_cluster2 in range(2, 10):
    kmeans2 = KMeans(n_clusters = n_cluster2, random_state = 0).fit(bronx_grouped_clust
ering)
    label2 = kmeans2.labels_
    sil_coeff = silhouette_score(bronx_grouped_clustering, label2, metric = 'euclidean'
)
    print("Where n_clusters = {}, the Silhouette Coefficient is {}".format(n_cluster2,
sil_coeff))
```

```
Where n_clusters = 2, the Silhouette Coefficient is 0.6787637908476635
Where n_clusters = 3, the Silhouette Coefficient is 0.6406348689213593
Where n_clusters = 4, the Silhouette Coefficient is 0.6170280718679633
Where n_clusters = 5, the Silhouette Coefficient is 0.6239284792976618
Where n_clusters = 6, the Silhouette Coefficient is 0.650469587277814
Where n_clusters = 7, the Silhouette Coefficient is 0.6537983168861516
Where n_clusters = 8, the Silhouette Coefficient is 0.6647395678798047
Where n_clusters = 9, the Silhouette Coefficient is 0.6550315362017799
```

The value of k in which the coefficient is higher is the chosen one. In this case k = 3

The method is applied with k = 3

In [27]:

```
# set number of clusters
kclusters = 3

bronx_grouped_clustering = bronx_grouped.drop('Neighborhood', 1)

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

# check cluster labels generated for each row in the dataframe
kmeans.labels_[0:50]
```

Out[27]:

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

In [28]:

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

bronx_merged = bronx_data


# merge toronto_grouped with toronto_data to add latitude/longitude for each neighborho
od
bronx_merged = bronx_merged.join(neighborhoods_venues_sorted.set_index('Neighborhood'),
on='Neighborhood')

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

Out[28]:

	Borough	Neighborhood	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue
0	Bronx	Wakefield	40.894705	-73.847201	1	Ice Cream Shop	Sandwich Place	Gas Station
1	Bronx	Co-op City	40.874294	-73.829939	0	Bus Station	Baseball Field	Grocery Store
2	Bronx	Eastchester	40.887556	-73.827806	0	Caribbean Restaurant	Bus Station	Deli / Bodega
3	Bronx	Fieldston	40.895437	-73.905643	0	Plaza	Bus Station	River
4	Bronx	Riverdale	40.890834	-73.912585	0	Bus Station	Park	Food Truck

The clusters formed are shown on the map

In [29]:

```
# 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(bronx_merged['Latitude'], bronx_merged['Longitude'],
bronx_merged['Neighborhood'], bronx_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
```

Out[29]:

Verify the clusters formed

Cluster 1

In [87]:

```
bronx_merged.loc[bronx_merged['Cluster Labels'] == 0, bronx_merged.columns[[1] + list(r
ange(5, bronx_merged.shape[1]))]]
```

Out[87]:

	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	C
1	Co-op City	Bus Station	Baseball Field	Grocery Store	Fast Food Restaurant	Fried Chicken Joint	Mattress Store	Re
2	Eastchester	Caribbean Restaurant	Bus Station	Deli / Bodega	Diner	Metro Station	Platform	
3	Fieldston	Plaza	Bus Station	River	Women's Store	Eastern European Restaurant	Fish Market	Fish
4	Riverdale	Bus Station	Park	Food Truck	Baseball Field	Plaza	Home Service	
17	West Farms	Bus Station	Park	Supermarket	Deli / Bodega	Outdoors & Recreation	Pizza Place	Con
18	High Bridge	Pharmacy	Pizza Place	Bus Station	Chinese Restaurant	Sandwich Place	Sports Club	
24	Morrisania	Discount Store	Bus Station	Pizza Place	Grocery Store	Metro Station	Fast Food Restaurant	Do
25	Soundview	Chinese Restaurant	Pharmacy	Latin American Restaurant	Bus Stop	Breakfast Spot	Fried Chicken Joint	B

Cluster 2

In [88]:

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

Out[88]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Mo Comm Ver
0	Wakefield	Ice Cream Shop	Sandwich Place	Gas Station	Caribbean Restaurant	Laundromat	Donut Sh
5	Kingsbridge	Pizza Place	Bar	Mexican Restaurant	Sandwich Place	Latin American Restaurant	Supermar
6	Woodlawn	Deli / Bodega	Bar	Playground	Pub	Pizza Place	Ind Restaur
7	Norwood	Pizza Place	Park	Bank	Mobile Phone Shop	Pharmacy	Fast Fo Restaur
8	Williamsbridge	Playground	Soup Place	Nightclub	Bar	Caribbean Restaurant	Fo
9	Baychester	Electronics Store	Donut Shop	Pizza Place	Bank	Men's Store	Fast Fo Restaur
10	Pelham Parkway	Pizza Place	Bus Station	Italian Restaurant	Frozen Yogurt Shop	Deli / Bodega	Chine Restaur
11	City Island	Harbor / Marina	Grocery Store	Thrift / Vintage Store	Seafood Restaurant	Pizza Place	Span Restaur
12	Bedford Park	Diner	Mexican Restaurant	Pizza Place	Supermarket	Chinese Restaurant	Sandw Pla
13	University Heights	Pizza Place	Fried Chicken Joint	Fast Food Restaurant	Bakery	Shoe Store	Sandw Pla
14	Morris Heights	Spanish Restaurant	Pizza Place	Pharmacy	Playground	Bus Station	Food Tru
15	Fordham	Donut Shop	Mobile Phone Shop	Pizza Place	Bank	Shoe Store	Gy Fitne Cer
16	East Tremont	Pizza Place	Supermarket	Cosmetics Shop	Paella Restaurant	Discount Store	Donut Sh
19	Melrose	Pizza Place	Pharmacy	Supermarket	Supplement Shop	Department Store	Pape Offi Suppl Sto
20	Mott Haven	Pizza Place	Donut Shop	Gym	Spanish Restaurant	Baseball Field	Disco Sto
21	Port Morris	Spanish Restaurant	Baseball Field	Cupcake Shop	Distillery	Donut Shop	Restaur
22	Longwood	Pizza Place	Diner	Fast Food Restaurant	Chinese Restaurant	Latin American Restaurant	Sandw Pla
23	Hunts Point	Spanish Restaurant	Juice Bar	Shipping Store	Café	Farmers Market	Gourr Sh
27	Throgs Neck	Juice Bar	Sports Bar	Deli / Bodega	American Restaurant	Pizza Place	Ital Restaur

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Mo Comm Ven
28	Country Club	Sandwich Place	Playground	Athletics & Sports	Fried Chicken Joint	Liquor Store	Mob Phone Sh
29	Parkchester	Supermarket	Pizza Place	Women's Store	Chinese Restaurant	Kids Store	Ba
30	Westchester Square	Fast Food Restaurant	Donut Shop	Pharmacy	Pizza Place	Mobile Phone Shop	Me Stat
31	Van Nest	Pizza Place	Bakery	Deli / Bodega	BBQ Joint	Film Studio	Caribbe Restaur
32	Morris Park	Pizza Place	Bakery	Burger Joint	Deli / Bodega	Buffet	Ba
33	Belmont	Italian Restaurant	Pizza Place	Deli / Bodega	Bakery	Grocery Store	Dess Sh
34	Spuyten Duyvil	Tennis Stadium	Bank	Intersection	Park	Pharmacy	Groc St
35	North Riverdale	Pizza Place	Italian Restaurant	Chinese Restaurant	Bank	Deli / Bodega	Social C
36	Pelham Bay	Italian Restaurant	Fast Food Restaurant	Bank	Cosmetics Shop	Donut Shop	Convenier St
37	Schuylerville	Pizza Place	Bank	Pharmacy	Mexican Restaurant	Bar	Donut Sh
38	Edgewater Park	Italian Restaurant	Pizza Place	Deli / Bodega	Coffee Shop	Juice Bar	Donut Sh
39	Castle Hill	Pizza Place	Pharmacy	Deli / Bodega	Cosmetics Shop	Diner	P
40	Olinville	Supermarket	Caribbean Restaurant	Fried Chicken Joint	Chinese Restaurant	Food	Baskett Co
41	Pelham Gardens	Pizza Place	Pharmacy	Donut Shop	Bus Station	Lawyer	Playgrou
42	Concourse	Deli / Bodega	Pizza Place	Clothing Store	Bus Station	Café	Sandw Pla
43	Unionport	Donut Shop	Latin American Restaurant	Pizza Place	Lounge	Dance Studio	Di
44	Edenwald	Grocery Store	Fish Market	Pizza Place	Supermarket	Gas Station	Athletic Spo
45	Claremont Village	Deli / Bodega	Pizza Place	Park	Bakery	Grocery Store	Bus Stat
46	Concourse Village	Fast Food Restaurant	Mexican Restaurant	Sandwich Place	Pharmacy	Sporting Goods Shop	Convenier St
47	Mount Eden	Pharmacy	Fried Chicken Joint	Supermarket	Spanish Restaurant	Fast Food Restaurant	Pizza Pla

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Mo Comm Ven
48	Mount Hope	Deli / Bodega	Grocery Store	Clothing Store	Chinese Restaurant	Donut Shop	Sandw Pla
49	Bronxdale	Italian Restaurant	Spanish Restaurant	Bank	Pizza Place	Performing Arts Venue	Pap Off Suppl St
50	Allerton	Pizza Place	Deli / Bodega	Chinese Restaurant	Supermarket	Fast Food Restaurant	Martial A D
51	Kingsbridge Heights	Pizza Place	Mexican Restaurant	Fried Chicken Joint	Spanish Restaurant	Coffee Shop	Chine Restaur

Cluster 3

In [89]:

```
bronx_merged.loc[bronx_merged['Cluster Labels'] == 2, bronx_merged.columns[[1] + list(r
ange(5, bronx_merged.shape[1]))]]
```

Out[89]:

	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
26	Clason Point	Park	Grocery Store	South American Restaurant	Pool	Boat or Ferry	Bus Stop	Electronics Store

The venues categories of Kingsbridge are verified

In [37]:

```
kingsbridge_data=bronx_venues.loc[bronx_venues['Neighborhood']=='Kingsbridge']
```

In [38]:

```
kingsbridge_data
```

Out[38]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Ca
64	Kingsbridge	40.881687	-73.902818	Garden Gourmet Market	40.881350	-73.903389	G
65	Kingsbridge	40.881687	-73.902818	Leche y Miel	40.883709	-73.901606	An Res
66	Kingsbridge	40.881687	-73.902818	Kingsbridge Social Club	40.884545	-73.901964	Pizz
67	Kingsbridge	40.881687	-73.902818	El Malecon	40.879338	-73.904457	Car Res
68	Kingsbridge	40.881687	-73.902818	Sam's Pizza	40.879435	-73.905859	Pizz
69	Kingsbridge	40.881687	-73.902818	The Bronx Public	40.878377	-73.903481	
70	Kingsbridge	40.881687	-73.902818	Carvel Ice Cream	40.883657	-73.901655	Ice
71	Kingsbridge	40.881687	-73.902818	Bronx Alehouse	40.884749	-73.899699	B
72	Kingsbridge	40.881687	-73.902818	Loeser's Delicatessen	40.879111	-73.905693	Sa
73	Kingsbridge	40.881687	-73.902818	Smashburger	40.884221	-73.900333	Burg
74	Kingsbridge	40.881687	-73.902818	Estrellita Poblana V	40.879687	-73.906257	M Res
75	Kingsbridge	40.881687	-73.902818	BJ's Wholesale Club	40.884104	-73.900267	Ware
76	Kingsbridge	40.881687	-73.902818	Picante Picante Mexican Restaurant	40.878252	-73.902936	M Res
77	Kingsbridge	40.881687	-73.902818	Tilila	40.883872	-73.898209	
78	Kingsbridge	40.881687	-73.902818	Mon Amour Coffee & Wine	40.885009	-73.900332	Coffe
79	Kingsbridge	40.881687	-73.902818	Buffalo Wild Wings	40.884460	-73.900424	Wing
80	Kingsbridge	40.881687	-73.902818	El Economico Restaurant	40.879330	-73.904597	S Res
81	Kingsbridge	40.881687	-73.902818	S & S Cheesecake	40.884793	-73.899861	
82	Kingsbridge	40.881687	-73.902818	The Putnam Trail	40.885189	-73.899450	
83	Kingsbridge	40.881687	-73.902818	Chipotle Mexican Grill	40.884566	-73.900474	M Res
84	Kingsbridge	40.881687	-73.902818	Broadway Pizza & Pasta	40.878822	-73.904494	Pizz

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Ca
85	Kingsbridge	40.881687	-73.902818	MyUnique	40.881966	-73.903584	\
86	Kingsbridge	40.881687	-73.902818	Q'Kachapa	40.880006	-73.904157	S Res
87	Kingsbridge	40.881687	-73.902818	Dunkin'	40.879308	-73.905066	Donu
88	Kingsbridge	40.881687	-73.902818	ALDI	40.877836	-73.904656	Super
89	Kingsbridge	40.881687	-73.902818	Riverdale Diner	40.885183	-73.899484	
90	Kingsbridge	40.881687	-73.902818	Sugarboy Bakery Cafe	40.877832	-73.902669	
91	Kingsbridge	40.881687	-73.902818	Land & Sea Restaurant	40.877885	-73.905873	S Res
92	Kingsbridge	40.881687	-73.902818	Gold Mine Cafe	40.878916	-73.904698	
93	Kingsbridge	40.881687	-73.902818	Leila's Hand Dipped Chocolate	40.879275	-73.905654	Cand
94	Kingsbridge	40.881687	-73.902818	Lot Less Closeouts	40.878270	-73.905265	D
95	Kingsbridge	40.881687	-73.902818	Enterprise Rent-A-Car	40.879866	-73.903847	Ren L
96	Kingsbridge	40.881687	-73.902818	IHOP	40.880422	-73.904019	Br
97	Kingsbridge	40.881687	-73.902818	SUBWAY	40.878493	-73.905385	Sa
98	Kingsbridge	40.881687	-73.902818	Little Caesars Pizza	40.880002	-73.904140	Pizz
99	Kingsbridge	40.881687	-73.902818	The Local	40.878553	-73.903462	
100	Kingsbridge	40.881687	-73.902818	Foodtown of Riverdale	40.878524	-73.905296	Super
101	Kingsbridge	40.881687	-73.902818	Stop & Shop	40.882129	-73.901687	Super
102	Kingsbridge	40.881687	-73.902818	Silhouette Restaurant & Lounge	40.880706	-73.902687	An Res
103	Kingsbridge	40.881687	-73.902818	Dollar Tree	40.881715	-73.903187	D
104	Kingsbridge	40.881687	-73.902818	GNC	40.879760	-73.904484	Supp
105	Kingsbridge	40.881687	-73.902818	T-Mobile	40.884462	-73.899914	Phon
106	Kingsbridge	40.881687	-73.902818	Mattress Firm	40.881580	-73.903277	M
107	Kingsbridge	40.881687	-73.902818	KFC	40.880164	-73.904253	C
108	Kingsbridge	40.881687	-73.902818	Dunkin'	40.884442	-73.900185	Donu
109	Kingsbridge	40.881687	-73.902818	Mr. McGoo's	40.879419	-73.904243	

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Ca
110	Kingsbridge	40.881687	-73.902818	Walgreens	40.878538	-73.904780	Pha
111	Kingsbridge	40.881687	-73.902818	FedEx Office Print & Ship Center	40.882667	-73.901487	Sl
112	Kingsbridge	40.881687	-73.902818	Subway	40.877720	-73.905380	Sa
113	Kingsbridge	40.881687	-73.902818	Petco	40.884123	-73.899339	Pe
114	Kingsbridge	40.881687	-73.902818	Popeyes Louisiana Kitchen	40.879312	-73.904798	C
115	Kingsbridge	40.881687	-73.902818	Chase Bank	40.879229	-73.904968	
116	Kingsbridge	40.881687	-73.902818	The Bridge Tavern	40.883495	-73.901763	Spo
117	Kingsbridge	40.881687	-73.902818	Rite Aid	40.885481	-73.900814	Pha
118	Kingsbridge	40.881687	-73.902818	Hola España	40.879160	-73.904553	An Res
119	Kingsbridge	40.881687	-73.902818	Domino's Pizza	40.884351	-73.902331	Pizza
120	Kingsbridge	40.881687	-73.902818	Bob’s Discount Furniture and Mattress Store	40.877704	-73.904719	Fur Hom
121	Kingsbridge	40.881687	-73.902818	Victoria Nails & Spa	40.883158	-73.901475	Nai
122	Kingsbridge	40.881687	-73.902818	McDonald's	40.884067	-73.901712	Fas Res
123	Kingsbridge	40.881687	-73.902818	Broadway Liquors & Wines	40.884308	-73.900850	Liquo
124	Kingsbridge	40.881687	-73.902818	The Punchbowl	40.885020	-73.900662	
125	Kingsbridge	40.881687	-73.902818	Kennedy Deli	40.880121	-73.907050	E
126	Kingsbridge	40.881687	-73.902818	Kingsbridge-Riverdale Farmers' Market	40.879973	-73.907295	Vege Res
127	Kingsbridge	40.881687	-73.902818	Oni Beauty Salon	40.880132	-73.907452	Cos
128	Kingsbridge	40.881687	-73.902818	24 Hour Fitness	40.880592	-73.908255	I
129	Kingsbridge	40.881687	-73.902818	Bronx express superette	40.882586	-73.897783	Conve
130	Kingsbridge	40.881687	-73.902818	H & A Convenience Store	40.883819	-73.897962	E
131	Kingsbridge	40.881687	-73.902818	Coyote Statue	40.885180	-73.899439	C Sc

Now, let's find out the most common venue at Bronx district

In [33]:

```
bronx_venue_unique_count = bronx_venues['Venue Category'].value_counts().to_frame(name=
'Count')
```

In [36]:

```
bronx_venue_unique_count.head()
```

Out[36]:

	Count
Pizza Place	101
Deli / Bodega	55
Donut Shop	45
Pharmacy	40
Italian Restaurant	39

4. Results

In resume, according to results shown before:

- At Bronx district Pizza Place is the most common venue.
- The neighborhood with the most Latin food restaurants is Kingsbridge. This neighborhood also seems to be a restaurants district and an appropriate place to open a Latin food restaurant.
- Cluster 2 shows that on those neighborhoods there is predominance of restaurants and food places.

5. Discussion and Conclusion

It is very important to note that Clusters 2 is the most viable clusters to create a Latin American Restaurant. Their proximity to other amenities and accessibility to station are paramount.

In conclusion, this project would have had better results if there were more data in terms of traffic access and the quantity of latin people living on those neighborhoods.

In [ ]:

```

```