Trending Business Ventures in Zurich, Switzerland

June 7, 2021

0.1 Introduction

Switzerland is vibrant and international. Even though migrating to Switzerland is not an easy task, it's still every ones dream to live and explore different parts of Switzerland. The aim of this capstone project is to demonstrate how Foursquare API can be used to explore and compare cities in Canton Zurich, Switzerland. This project will help people in exploring better facilities around their neighborhood in Zurich. It will help people making smart and efficient decision on selecting great neighborhood out of numbers of other neighborhoods in Zurich, Switzerland.

0.2 Table of Contents

Before we get the data and start exploring it, let's download all the dependencies that we will need.

[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␣
↪completed 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 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.');
```

```
Collecting package metadata (current_repodata.json): done
Solving environment: done

# All requested packages already installed.

Libraries imported.
```

0.3 1. Download and Explore Dataset

[2]:
```
# Source: https://raw.githubusercontent.com/zauberware/
 ↪postal-codes-json-xml-csv/master/data/CH/zipcodes.ch.csv
swiss_data = pd.read_csv('zipcodes.ch.csv')
#swiss_data.head()
print(swiss_data.shape)
swiss_data.sample(n=10)
```

```
(4356, 11)
```

[2]:

	country_code	zipcode	place	state	state_code	province	province_code	community
3833	CH	1912	Dugny (Leytron)	Canton du Valais	VS	Martigny District	2307	Leytron
3925	CH	3965	Chippis	Canton du Valais	VS	Sierre District	2311	Chippis
3776	CH	1961	Vernamiège	Canton du Valais	VS	Hérens District	2305	Mont-Noble
2131	CH	6372	Ennetmoos	Kanton Nidwalden	NW			
3624	CH	1278	La Rippe	Canton de Vaud	VD			
2756	CH	8376	Au TG	Kanton Thurgau	TG			
2355	CH	9657	Unterwasser	Kanton St. Gallen	SG			
598	CH	3000	Bern 60 UPD	Canton de Berne	BE			
2947	CH	6614	Brissago	Ticino	TI			
3788	CH	1969	St-Martin VS	Canton du Valais	VS			

2131	Nidwalden	700	Ennetmoos
3624	Nyon District	2228	La Rippe
2756	Münchwilen District	2014	Fischingen
2355	Wahlkreis Toggenburg	1727	Wildhaus-Alt St. Johann
598	Bern-Mittelland District	246	Bern
2947	Locarno District	2104	Brissago
3788	Hérens District	2305	Saint-Martin (VS)

	community_code	latitude	longitude
3833	6135	46.1875	7.1982
3925	6235	46.2802	7.5396
3776	6090	46.2117	7.4313
2131	1506	46.9559	8.3388
3624	5726	46.3810	6.1505
2756	4726	47.3984	8.9542
2355	3359	47.1970	9.3086
598	351	46.9476	7.4065
2947	5097	46.1201	8.7118
3788	6087	46.1615	7.4510

Load and explore the data

[3]:
```
swiss_data["state_code"].value_counts()
```

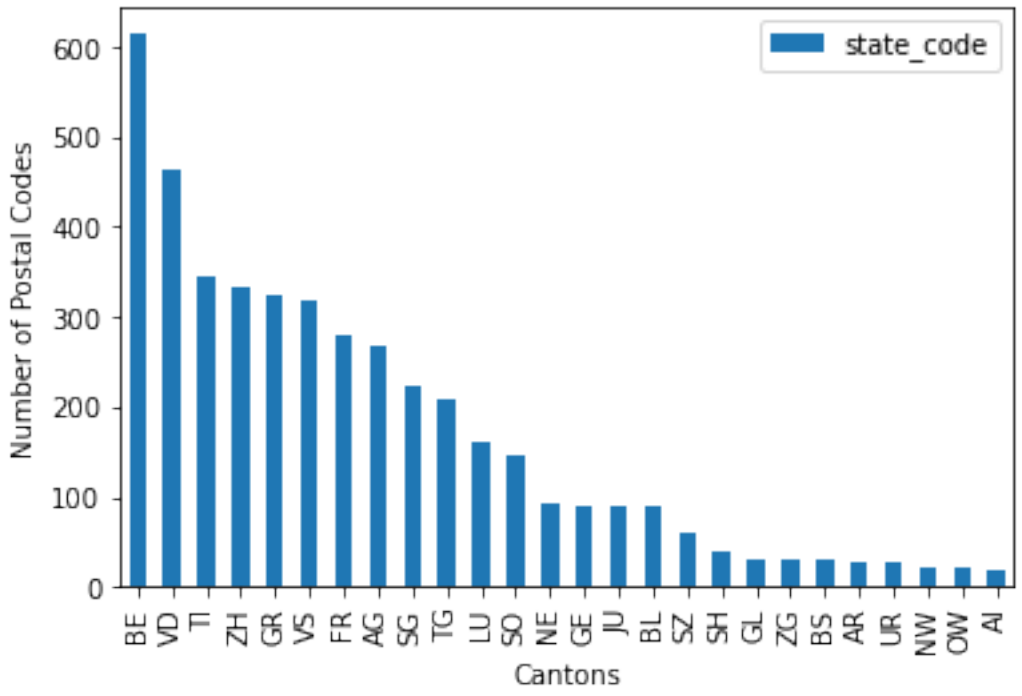
[3]:
```
BE    615
VD    464
TI    345
ZH    332
GR    325
VS    318
FR    281
AG    269
SG    223
TG    208
LU    162
SO    146
NE     93
GE     91
JU     91
BL     90
SZ     60
SH     38
GL     31
ZG     31
BS     30
AR     27
UR     27
```

```
NW     21
OW     21
AI     17
Name: state_code, dtype: int64
```

However, for illustration purposes, let's simplify the above map and segment and cluster only the Towns in Zurich. So let's slice the original dataframe and create a new dataframe of the Zurich data.

[4]:
```
df = pd.DataFrame(swiss_data["state_code"].value_counts())
df.plot(kind="bar", xlabel= "Cantons", ylabel= "Number of Postal Codes")
```

[4]:
```
<AxesSubplot:xlabel='Cantons', ylabel='Number of Postal Codes'>
```



[5]:
```
swiss_data_ZH = swiss_data[swiss_data['state_code'].str.
 ↪contains('ZH',regex=False)]
print(swiss_data_ZH.shape)
swiss_data_ZH.sample(n=10)
```

```
(332, 11)
```

[5]:

	country_code	zipcode	place	state	state_code	\
4166	CH	8815	Horgenberg	Kanton Zürich	ZH	
4306	CH	8002	Zürich	Kanton Zürich	ZH	
4270	CH	8474	Dinhard	Kanton Zürich	ZH	

```
4197           CH     8317         Tagelswangen  Kanton Zürich         ZH
4137           CH     8496      Steg im Tösstal  Kanton Zürich         ZH
4324           CH     8042               Zürich  Kanton Zürich         ZH
4322           CH     8040               Zürich  Kanton Zürich         ZH
4118           CH     8156            Oberhasli  Kanton Zürich         ZH
4280           CH     8542         Wiesendangen  Kanton Zürich         ZH
4147           CH     8626  Ottikon (Gossau ZH)  Kanton Zürich         ZH

               province  province_code     community  community_code  \
4166      Bezirk Horgen            106        Horgen             133
4306      Bezirk Zürich            112        Zürich             261
4270  Bezirk Winterthur            110       Dinhard             216
4197   Bezirk Pfäffikon            108        Lindau             176
4137      Bezirk Hinwil            105   Fischenthal             114
4324      Bezirk Zürich            112        Zürich             261
4322      Bezirk Zürich            112        Zürich             261
4118   Bezirk Dielsdorf            104   Niederhasli              90
4280  Bezirk Winterthur            110  Wiesendangen             298
4147      Bezirk Hinwil            105   Gossau (ZH)             115

      latitude  longitude
4166   47.2485     8.5877
4306   47.3667     8.5500
4270   47.5500     8.7667
4197   47.4307     8.6728
4137   47.3506     8.9342
4324   47.3828     8.5307
4322   47.3828     8.5307
4118   47.4657     8.4988
4280   47.5217     8.7897
4147   47.2946     8.7819
```

[6]:
```
swiss_data_ZH = swiss_data_ZH.reset_index()
swiss_data_ZH.head()
del swiss_data_ZH['index']
```

Let's get the geographical coordinates of Manhattan.

[7]:
```
swiss_data_ZH.head()
```

[7]:
```
  country_code  zipcode       place          state state_code  \
0           CH     8143   Uetliberg  Kanton Zürich         ZH
1           CH     8143   Stallikon  Kanton Zürich         ZH
2           CH     8906  Bonstetten  Kanton Zürich         ZH
3           CH     8907    Wettswil  Kanton Zürich         ZH
4           CH     8908    Hedingen  Kanton Zürich         ZH
```

```
           province  province_code          community  community_code  \
0  Bezirk Affoltern            101          Stallikon              13
1  Bezirk Affoltern            101          Stallikon              13
2  Bezirk Affoltern            101         Bonstetten               3
3  Bezirk Affoltern            101  Wettswil am Albis              14
4  Bezirk Affoltern            101           Hedingen               5

   latitude  longitude
0   47.3521     8.4875
1   47.3258     8.4897
2   47.3150     8.4684
3   47.2695     8.4744
4   47.2979     8.4483
```

0.4 Selecting the Bezirks (Province) for Analysis

[64]:
```
CLIENT_ID = 'xxxxxxxxxxxx' # your Foursquare ID
CLIENT_SECRET = 'xxxxxxxxxxxx' # your Foursquare Secret
ACCESS_TOKEN = 'xxxxxxxxxxxx' # your FourSquare Access Token
VERSION = 'xxxxxxxxxxxx'
LIMIT = 50
print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
```

```
Your credentails:
CLIENT_ID: xxxxxxxxxxxx
CLIENT_SECRET:xxxxxxxxxxxx
```

[14]:
```
resList = pd.DataFrame()
```

[15]:
```
swiss_data_ZH.head()
```

[15]:
```
  country_code  zipcode       place          state state_code  \
0           CH     8143   Uetliberg  Kanton Zürich         ZH
1           CH     8143   Stallikon  Kanton Zürich         ZH
2           CH     8906  Bonstetten  Kanton Zürich         ZH
3           CH     8907    Wettswil  Kanton Zürich         ZH
4           CH     8908    Hedingen  Kanton Zürich         ZH

           province  province_code          community  community_code  \
0  Bezirk Affoltern            101          Stallikon              13
1  Bezirk Affoltern            101          Stallikon              13
2  Bezirk Affoltern            101         Bonstetten               3
3  Bezirk Affoltern            101  Wettswil am Albis              14
4  Bezirk Affoltern            101           Hedingen               5
```

```
   latitude  longitude
0   47.3521     8.4875
1   47.3258     8.4897
2   47.3150     8.4684
3   47.2695     8.4744
4   47.2979     8.4483
```

[16]:
```
resList["PLZ"] = swiss_data_ZH.zipcode
resList["Latitude"] = swiss_data_ZH.latitude
resList["Longitude"] = swiss_data_ZH.longitude
resList["Bezirk"] = swiss_data_ZH.province
```

[19]:
```
search_query = "Restaurant"
radius= 500
print(search_query+"....OK")

resCount= []
for i in range(len(resList["Latitude"])):
    latitude_1 = resList["Latitude"].iloc[i]
    longitude_1 = resList["Longitude"].iloc[i]
    url_1 = 'https://api.foursquare.com/v2/venues/search?
 ↪client_id={}&client_secret={}&ll={},{}&oauth_token={}&v={}&query={}&radius={}&limit={}'.
 ↪format(CLIENT_ID, CLIENT_SECRET, latitude_1, longitude_1,ACCESS_TOKEN,␣
 ↪VERSION, search_query, radius, LIMIT)
    results_1 = requests.get(url_1).json()
    # assign relevant part of JSON to venues
    venues_1 = results_1['response']['venues']

    # tranform venues into a dataframe
    dataframe_1 = json_normalize(venues_1)
    resCount.append(len(dataframe_1))
resCount

resList["N_Restaurants"] = resCount
resList.head()
#Now use this list of cities for the analysis
# Remove NaN, and Bar Plot and analysi on
```

```
Restaurant....OK

/home/jupyterlab/conda/envs/python/lib/python3.6/site-
packages/ipykernel_launcher.py:15: FutureWarning: pandas.io.json.json_normalize
is deprecated, use pandas.json_normalize instead
  from ipykernel import kernelapp as app
```

[19]:

	PLZ	Latitude	Longitude	Bezirk	N_Restaurants
0	8143	47.3521	8.4875	Bezirk Affoltern	2
1	8143	47.3258	8.4897	Bezirk Affoltern	2

```
2  8906   47.3150     8.4684  Bezirk Affoltern             1
3  8907   47.2695     8.4744  Bezirk Affoltern             1
4  8908   47.2979     8.4483  Bezirk Affoltern             2
```

[20]:
```
resList.shape
```

[20]:
```
(332, 5)
```

[21]:
```
resList.to_csv(r'wholeresList.csv', index = False, header=True)
#resList_ZH = pd.read_csv('Restaurant_Zurich_dataframe.csv')
#swiss_data.head()
resList_ZH = pd.read_csv('wholeresList.csv')
print(resList_ZH.head())
resList_ZH.sample(n=10)
```

```
    PLZ  Latitude  Longitude            Bezirk  N_Restaurants
0  8143   47.3521     8.4875  Bezirk Affoltern              2
1  8143   47.3258     8.4897  Bezirk Affoltern              2
2  8906   47.3150     8.4684  Bezirk Affoltern              1
3  8907   47.2695     8.4744  Bezirk Affoltern              1
4  8908   47.2979     8.4483  Bezirk Affoltern              2
```

[21]:
```
       PLZ  Latitude  Longitude            Bezirk  N_Restaurants
68    8196   47.6045     8.5081     Bezirk Bülach              0
9     8913   47.2823     8.4043  Bezirk Affoltern              3
129   8636   47.2760     8.9140     Bezirk Hinwil              7
97    8162   47.4971     8.4522  Bezirk Dielsdorf              0
14    8926   47.2304     8.4954  Bezirk Affoltern              0
109   8340   47.2943     8.8439     Bezirk Hinwil              0
100   8165   47.4975     8.3975  Bezirk Dielsdorf              0
184   8484   47.4294     8.7888  Bezirk Pfäffikon              1
0     8143   47.3521     8.4875  Bezirk Affoltern              2
13    8925   47.2264     8.5496  Bezirk Affoltern              2
```

[22]:
```
resList_ZH_sorted = resList_ZH.sort_values(by="N_Restaurants",ascending=False,␣
↪inplace= False, kind="quicksort",na_position="last")
```

[23]:
```
resList_ZH_sorted2 = resList_ZH_sorted.drop_duplicates(
  subset = ['Latitude', 'Longitude'],
  keep = 'last').reset_index(drop = True)
```

[24]:
```
for i in range(len(resList_ZH_sorted2.Bezirk)):
    resList_ZH_sorted2.Bezirk.values[i] = resList_ZH_sorted2.Bezirk.values[i].
↪strip("Bezirk ")
```

0.5 Collecting the available Restaurant data for Analysis

[25]:
```
resList_ZH_sorted2_filtered = resList_ZH_sorted2.drop_duplicates(
  subset = ['Bezirk'],
  keep = 'first').reset_index(drop = True)
```

[30]:
```
selected_bezirk = resList_ZH_sorted2_filtered.head(n=12)
selected_bezirk
```

[30]:

	PLZ	Latitude	Longitude	Bezirk	N_Restaurants
0	8086	47.3828	8.5307	Zürich	38
1	8403	47.4967	8.7342	Winterthu	18
2	8953	47.4017	8.4001	Dietikon	16
3	8302	47.4515	8.5849	ülach	12
4	8610	47.3471	8.7209	Ust	12
5	8800	47.2918	8.5635	Horgen	10
6	8623	47.3264	8.7978	Hinwil	9
7	8105	47.4341	8.4687	Dielsdorf	8
8	8910	47.2774	8.4513	Affoltern	7
9	8712	47.2425	8.7234	Meilen	5
10	8320	47.3878	8.7515	Pfäffikon	5
11	8245	47.6905	8.6436	Andelfingen	5

0.5.1 Now we need to pass the Lat, Long and retrieve the Ratings – Sort them and return only the Top 5 Restaurant names Later, Plot it

Set the Query, URL, GET JSON

[ ]:
```
search_query = "Restaurant"
radius= 500
print(search_query+"....OK")
```

0.5.2 Latitude and Logitude of the Bezirk

[ ]:
```
latitude = selected_bezirk["Latitude"].iloc[0]
longitude = selected_bezirk["Longitude"].iloc[0]
```

0.5.3 Create the URL to get JSON data

[ ]:
```
url = 'https://api.foursquare.com/v2/venues/search?
 ↪client_id={}&client_secret={}&ll={},{}&oauth_token={}&v={}&query={}&radius={}&limit={}'.
 ↪format(CLIENT_ID, CLIENT_SECRET, latitude, longitude,ACCESS_TOKEN, VERSION,␣
 ↪search_query, radius, LIMIT)
url
```

0.5.4 Send the Request

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

0.5.5 JSON to Pandas Dataframe

[ ]:
```
# assign relevant part of JSON to venues
venues = results['response']['venues']

# tranform venues into a dataframe
dataframe = json_normalize(venues)
dataframe.head()
```

What we need from this is only the id of the restaurants -then we do the same above methods and get the rating – sort the rating and return top 5 with info about Restaurant name, Category, Address, Lat, Long, Rating

[ ]:
```
# Req 1: Venue ID
venue_id = dataframe["id"].iloc[0]
```

[ ]:
```
# Req 2: URL
url_venue = 'https://api.foursquare.com/v2/venues/{}?
 ↪client_id={}&client_secret={}&oauth_token={}&v={}'.format(venue_id,␣
 ↪CLIENT_ID, CLIENT_SECRET,ACCESS_TOKEN, VERSION)
url_venue
```

[ ]:
```
# Now send the GET request
result_venue = requests.get(url_venue).json()
```

[ ]:
```
#result_venue
```

[ ]:
```
#Here we take only the ratings of each location
try:
    rate = float(result_venue['response']['venue']['rating'])
    # tranform venues into a dataframe
    #dataframe = json_normalize(venues)
    print(rate)
except:
    rate = float("NaN")
    #print(result['response']['venue'])
    #print('This venue has not been rated yet.')
```

[32]:
```
# Function to Extract Categories
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']
```

[ ]:
```
import copy
dataframe2 = copy.deepcopy(dataframe)
dataframe2['categories'] = dataframe2.apply(get_category_type, axis=1)
dataframe2.drop(['venuePage.id'],axis=1,inplace=True)
dataframe2.columns = [column.split('.')[-1] for column in dataframe2.columns]
dataframe2.
 ↪drop(['referralId','hasPerk','address','postalCode','labeledLatLngs','distance','cc','city'
dataframe2.head()
```

[33]:
```
import copy
#ratings= []
top5 = pd.DataFrame(columns=['id','name','categories','location.
 ↪address','location.lat','location.lng','Rating'])
# 1
for Num in range(len(selected_bezirk)):
    ratings= []
    search_query = "Restaurant"
    radius= 500
    print(search_query+"....OK")

    # 2 For Each Bezirk new Lat und Long
    latitude = selected_bezirk["Latitude"].iloc[Num]
    longitude = selected_bezirk["Longitude"].iloc[Num]
    # 3
    url = 'https://api.foursquare.com/v2/venues/search?
 ↪client_id={}&client_secret={}&ll={},{}&oauth_token={}&v={}&query={}&radius={}&limit={}'.
 ↪format(CLIENT_ID, CLIENT_SECRET, latitude, longitude,ACCESS_TOKEN, VERSION,␣
 ↪search_query, radius, LIMIT)
    # 4
    results = requests.get(url).json()
    # 5
    # assign relevant part of JSON to venues
    venues = results['response']['venues']

    # tranform venues into a dataframe
    dataframe = json_normalize(venues)
    dataframe.head()

    # 6
    dataframe['categories'] = dataframe.apply(get_category_type, axis=1)
```

```
    dataframe = dataframe[['id','name','categories','location.
↪address','location.lat','location.lng']]
    #dataframe.drop(['venuePage.id'],axis=1,inplace=True)
    #dataframe.columns = [column.split('.')[-1] for column in dataframe.columns]
    #dataframe.
↪drop(['referralId','hasPerk','address','postalCode','labeledLatLngs','distance','cc','city'
    #dataframe.head()

    for i in range(len(dataframe)):
        venue_id = dataframe["id"].iloc[i]
        url_venue = 'https://api.foursquare.com/v2/venues/{}?
↪client_id={}&client_secret={}&oauth_token={}&v={}'.format(venue_id,␣
↪CLIENT_ID, CLIENT_SECRET,ACCESS_TOKEN, VERSION)
        result_venue = requests.get(url_venue).json()
        try:
            rate = result_venue['response']['venue']['rating']
            #print(result_venue['response']['venue']['rating'])
        except:
            rate = float("NaN")
            #print('This venue has not been rated yet.')
        ratings.append(rate)
    ratings

    ##
    dataframe["Rating"] = ratings

    dataframe_ratings_sorted = dataframe.
↪sort_values(by="Rating",ascending=False, inplace= False,␣
↪kind="quicksort",na_position="last")

    ##
    print("Finished Bezirk: "+str(Num)+"------")
    #print("Top 5 Restaurants in Bezirk: "+str(selected_bezirk["Bezirk"].
↪iloc[0])+" are:")
    #print(dataframe_ratings_sorted.head())
    top= pd.DataFrame(dataframe_ratings_sorted.head())
    #print(top)
    top5 = top5.append(top)

print(top5)
```

```
Restaurant…OK

/home/jupyterlab/conda/envs/python/lib/python3.6/site-
packages/ipykernel_launcher.py:23: FutureWarning: pandas.io.json.json_normalize
is deprecated, use pandas.json_normalize instead

Finished Bezirk: 0------
```

```
Restaurant…OK
Finished Bezirk: 1------
Restaurant…OK
Finished Bezirk: 2------
Restaurant…OK
Finished Bezirk: 3------
Restaurant…OK
Finished Bezirk: 4------
Restaurant…OK
Finished Bezirk: 5------
Restaurant…OK
Finished Bezirk: 6------
Restaurant…OK
Finished Bezirk: 7------
Restaurant…OK
Finished Bezirk: 8------
Restaurant…OK
Finished Bezirk: 9------
Restaurant…OK
Finished Bezirk: 10------
Restaurant…OK
Finished Bezirk: 11------
                          id                                    name  \
5   4b058888f964a520f8cb22e3                      Restaurant JOSEF
2   4b0f01b0f964a520095e23e3                 Panama Bar & Restaurant
11  52dd552411d2bf369d4686a0  Restaurant & Bar  Valentin's  Zürich
36  4e41164662e17b948c233c11                      Phuket Asia Center
0   4eb2f06130f85a3c6d89de74                  Restaurant Valentin's
12  4b556dbaf964a5200fe427e3                      Restaurant Widder
5   4cadc4bf2303a14348e1e8f0                 Restaurant Al Giardino
17  4b5f2bcaf964a520c3aa29e3                                  bloom
0   4bf269ba55c7c9b6620e6204                    Restaurant Schäfli
7   4e7f69b777c8c6603259bab5                  Restaurant zur Sonne
12  4c28d2183492a59369f0b728                                 Tomate
13  4fe4eb22e4b0d55efea120fa                                 Heimat
0   4fefc013e4b0c8e6f59365f3              King Fast-Food Restaurant
1   4e9f31297beb6623c124f603                      Migros Restaurant
2   53bd3020498ebfa47ad3f558                         Restaurant Luo
11  4cadaf85d1f8b60cd48c70c6                                 Migros
0   5bbde1cdc0af57002cae3806         Eisdiele im Restaurant Ripasso
1   5b9793281ffe97002c910575           Restaurant Zur Alten Taverne
2   4e53faf6d4c05e0079f96e7e                    Restaurant Sonne
3   4caca1bff47ea14330039321             China Restaurant (Red Lion)
0   4b9b85b0f964a520c30c36e3                 Restaurant zur Kastanie
1   4d984b10647d8cfa1572eb3d              Boulevard Restaurant Uster
2   4caf182cc5e6a1cd69c7c9f6                      Migros Restaurant
3   532ae5ee498e47607f1364c6                  Restaurant Sonnenhof
4   51b47c40498e15fc00d6d2ea                  Restaurant Burg Uster
```

```
7   4d20c97cd7b0b1f7c2b2179f                  Restaurant Etzliberg
1   4b6f1cf0f964a520d5dd2ce3                Restaurant Grundstein
0   504f47def1ba6afe50c964d8              Coop Restaurant Thalwil
2   5022e1d2e4b07c69f5164a2d      Restaurant Trattoria Gallo Nero
3   582f4be0d0bb3e21438ca72a                  Restaurant ComViet
7   521252cb11d24231aa65972d                       Villa Barone
8   4c88d339105237044056c3f1                             Migros
0   5037f9a7e4b027db4f613b52              Restaurant Badi Egelsee
1   4e1f184262e19e27e250298e                      Restaurant West
2   4fdbb0c6e4b0619721a75f6c                    Restaurant Konter
2   4df3499dc65bf55ee52b28a5                  Restaurant Hot Wok
0   501d9ba8e4b0f873f83600c9                     Restaurant Ciao
1   4f99c59ee4b00694c47597a7        Restaurant zum ewigä Liechtli
3   50df30abe4b0a78bf519babb                  Restaurant Hardegg
4   4f8be8afe4b0bae899423ed8                   Migros Restaurant
0   4dd7f5dee4cd37c893b8cb5a               Restaurant Rosengarten
1   4b6c78e9f964a520ea3c2ce3                   Central Restaurant
2   53c6577d498e4e6bafd1ac85                    Restaurant Löwen
3   58b7fb9803cf257b09d5c3cf        Stella Restaurant & Take Away
4   58075f13d67c942a1bdb4394              China Restaurant Papaya
0   501855f6e4b06ec2737da3f0                Restaurant Alte Krone
1   4d4062e246775481fa6957f4              Restaurant Brückenwaage
2   4d767ccb18cc594110a1f13f                 Restaurant Hang Zhou
3   5050ae39e4b00991db9a2ef8                    Restaurant Rössli
4   4cd5142867c7236a9cea1077              Restaurant Schützenhaus
0   4fb38001e4b0271a4c344a04                      Restaurant Oase
1   4f6b7a80e4b00b8301be1964            Restaurant Schützengasse
2   5005e0b3e4b0229e8a19b0c0        17 - Restaurant zum Scharfeck
3   50abc453e4b0ddb08e6bd3a1            Restaurant&Lounge Pumpkin
4   50243330e4b0783685bdbe4e                  Stiftung zur Palme
0   5183997b498e5e7bc8cdc543           Restaurant Schwarzbrünneli
1   4f35172de4b01f8cbc84a5bd                  Restaurant Mühlental
2   5c94b65281635b002ced3fce  Coop Restaurant Feuerthalen Rhymarkt
3   5005b7c7e4b004dc66d91687  SOI 1 Bar . Restaurant . Thai Take Out
4   4d94c7cc078ef04dcade3278                                   Coop

                    categories               location.address  \
5             Swiss Restaurant               Gasometerstr. 24
2                          Bar                  Lettensteg 10
11                     Meyhane            Gasometerstrasse 36
36            Asian Restaurant                Schöneggstr. 21
0           Turkish Restaurant            Gasometerstrasse  36
12                   Gastropub                    Metzggasse 9
5           Italian Restaurant              Tösstalstrasse 70
17                   Hotel Bar             Stadthausstrasse 4
0                   Restaurant                            NaN
7             Swiss Restaurant               Marktgasse 13-15
12          Italian Restaurant                   Kirchstr. 16
```

13	Swiss Restaurant	Steinmürlistr. 66
0	Turkish Restaurant	Bremgartnerstr.10
1	Restaurant	Zentralstr. 17
2	Chinese Restaurant	Oberdorfstrasse 36
11	Supermarket	Schaffhauserstr.122
0	Ice Cream Shop	Square/Stadtplatz
1	Restaurant	ItSchaffhausersrrasse 161
2	Swiss Restaurant	NaN
3	Chinese Restaurant	Schaffhauserstrasse 138
0	Italian Restaurant	NaN
1	Fast Food Restaurant	Zürichstrasse 7
2	Restaurant	Zürichstr. 18
3	Gastropub	NaN
4	Restaurant	NaN
7	Restaurant	Etzlibergstrasse 14
1	Restaurant	Dorfstrasse 38
0	None	Florastrasse 1
2	None	NaN
3	Vietnamese Restaurant	Gotthardstrasse 11
7	Italian Restaurant	Bahnhofstrasse 131
8	Grocery Store	Bahnhofstr. 151
0	None	NaN
1	Food	NaN
2	None	NaN
2	Asian Restaurant	Roosstrasse 40
0	None	NaN
1	None	NaN
3	Restaurant	Watterstr. 185
4	Restaurant	NaN
0	Swiss Restaurant	Untere Bahnhofstrassse 33
1	Eastern European Restaurant	Zürichstrasse 100
2	Swiss Restaurant	NaN
3	Kebab Restaurant	Zürichstrasse 81
4	Cantonese Restaurant	Zürichstrasse 90
0	None	NaN
1	Swiss Restaurant	Seestrasse 71
2	Chinese Restaurant	Seestr. 43
3	Restaurant	NaN
4	Restaurant	Seestrasse 48
0	Falafel Restaurant	NaN
1	Swiss Restaurant	Hintergasse 15
2	None	NaN
3	Restaurant	NaN
4	Laundry Service	Hochstrasse 31-33
0	None	Stadtweg 2
1	Indian Restaurant	Durachweg 24, 8200 Schaffhausen
2	None	Schützenstrasse 30
3	Asian Restaurant	Bahnhofstrasse 50

```
4                          None                Schützenstrasse 30

    location.lat  location.lng  Rating
5      47.384006      8.529020     9.0
2      47.385167      8.535383     8.4
11     47.384770      8.529928     8.4
36     47.380210      8.526542     8.0
0      47.384740      8.529904     7.8
12     47.498649      8.727839     8.1
5      47.494340      8.739810     7.8
17     47.500976      8.732610     7.3
0      47.498909      8.731258     7.1
7      47.499683      8.729814     6.9
12     47.405364      8.403364     7.3
13     47.402251      8.395778     6.5
0      47.404198      8.402654     NaN
1      47.405088      8.401644     NaN
2      47.401511      8.400935     NaN
11     47.450963      8.581542     7.9
0      47.450436      8.582993     NaN
1      47.454240      8.579857     NaN
2      47.449273      8.583862     NaN
3      47.452983      8.580237     NaN
0      47.346644      8.715674     NaN
1      47.349439      8.715889     NaN
2      47.349597      8.715463     NaN
3      47.348999      8.720544     NaN
4      47.344826      8.718146     NaN
7      47.293709      8.555386     7.8
1      47.288694      8.565065     7.7
0      47.296668      8.562119     NaN
2      47.293038      8.565930     NaN
3      47.294240      8.564770     NaN
7      47.325855      8.798235     7.4
8      47.327407      8.799791     6.9
0      47.327219      8.797374     NaN
1      47.326458      8.793658     NaN
2      47.327477      8.802418     NaN
2      47.431187      8.471325     5.9
0      47.436742      8.468315     NaN
1      47.436742      8.468315     NaN
3      47.435860      8.472890     NaN
4      47.431388      8.468548     NaN
0      47.279425      8.449746     NaN
1      47.278810      8.454538     NaN
2      47.276272      8.447192     NaN
3      47.277365      8.454025     NaN
4      47.277657      8.454373     NaN
```

```
0      47.241601      8.723448     NaN
1      47.239377      8.720340     NaN
2      47.239319      8.718449     NaN
3      47.239492      8.718948     NaN
4      47.238991      8.718728     NaN
0      47.387905      8.752344     NaN
1      47.386516      8.751646     NaN
2      47.388168      8.751996     NaN
3      47.388284      8.745215     NaN
4      47.386720      8.752900     NaN
0      47.689888      8.643314     NaN
1      47.691850      8.643057     NaN
2      47.690583      8.646911     NaN
3      47.689461      8.638319     NaN
4      47.690721      8.646776     NaN
```

[35]:
```
top5.to_csv(r'top5Restaurants_whole.csv', index = False, header=True)
top5 = pd.read_csv('top5Restaurants_whole.csv')
top5.head()
#top5.drop(['Cluster Labels'],axis=1,inplace=True)
#top5.head()
```

[35]:
```
                         id                                  name  \
0  4b058888f964a520f8cb22e3                       Restaurant JOSEF
1  4b0f01b0f964a520095e23e3                 Panama Bar & Restaurant
2  52dd552411d2bf369d4686a0  Restaurant & Bar  Valentin's  Zürich
3  4e41164662e17b948c233c11                      Phuket Asia Center
4  4eb2f06130f85a3c6d89de74                  Restaurant Valentin's

            categories      location.address  location.lat  location.lng  \
0    Swiss Restaurant      Gasometerstr. 24     47.384006      8.529020
1                 Bar         Lettensteg 10     47.385167      8.535383
2             Meyhane   Gasometerstrasse 36     47.384770      8.529928
3    Asian Restaurant       Schöneggstr. 21     47.380210      8.526542
4  Turkish Restaurant  Gasometerstrasse  36     47.384740      8.529904

   Rating
0     9.0
1     8.4
2     8.4
3     8.0
4     7.8
```

[36]:
```
import folium
from IPython.display import HTML, display
# create map of New York using latitude and longitude values
address = 'Zurich, Switzerland'
```

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

top5_ZH = folium.Map(location=[latitude, longitude], zoom_start=10)

# add markers to map
for name, category, address, lat, lng, ratings in␣
 ↪zip(top5['name'],top5['categories'],top5['location.address'],top5['location.
 ↪lat'],top5['location.lng'],top5['Rating']):
    label = '{}, {},{},{}'.format(name,category,address,ratings)
    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(top5_ZH)

top5_ZH
```

```
The geograpical coordinate of Zurich City are 47.3744489, 8.5410422.
```

[36]:
```
<folium.folium.Map at 0x7f2fbcc99780>
```

0.6 Now try to cluster these restaurants

[38]:
```
# set number of clusters
kclusters = 3

restaurants_ZH_cluster = top5[["location.lat","location.lng"]]

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

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

[38]:
```
array([0, 0, 0, 0, 0, 2, 2, 2, 2, 2], dtype=int32)
```

[39]:
```
# add clustering labels
top5.insert(0, 'Cluster Labels', kmeans.labels_)

top5_merged = top5

# merge manhattan_grouped with manhattan_data to add latitude/longitude for␣
 ↪each neighborhood
#top5_merged = top5_merged.join(neighborhoods_venues_sorted.
 ↪set_index('Neighborhood'), on='Neighborhood')

top5_merged.head() # check the last columns!
```

[39]:

	Cluster Labels	id	name	categories	location.address	location.lat	location.lng	Rating
0	0	4b058888f964a520f8cb22e3	Restaurant JOSEF	Swiss Restaurant	Gasometerstr. 24	47.384006	8.529020	9.0
1	0	4b0f01b0f964a520095e23e3	Panama Bar & Restaurant	Bar	Lettensteg 10	47.385167	8.535383	8.4
2	0	52dd552411d2bf369d4686a0	Restaurant & Bar Valentin's Zürich	Meyhane	Gasometerstrasse 36	47.384770	8.529928	8.4
3	0	4e41164662e17b948c233c11	Phuket Asia Center	Asian Restaurant	Schöneggstr. 21	47.380210	8.526542	8.0
4	0	4eb2f06130f85a3c6d89de74	Restaurant Valentin's	Turkish Restaurant	Gasometerstrasse 36	47.384740	8.529904	7.8

[40]:
```
# 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, cluster in zip(top5_merged['location.lat'], top5_merged['location.
 ↪lng'], top5_merged['Cluster Labels']):
    label = folium.Popup(' 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
```

[40]:
```
<folium.folium.Map at 0x7f2fbcda2e48>
```

0.7 TASK 2: Analyze different venues from each Bezirk

[41]:
```
LIMIT = 50 # limit of number of venues returned by Foursquare API
radius = 500 # define radius
```

[42]:
```
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)
```

[62]:
```
selected_bezirk
```

[62]:

	PLZ	Latitude	Longitude	Bezirk	N_Restaurants
0	8086	47.3828	8.5307	Zürich	38
1	8403	47.4967	8.7342	Winterthu	18
2	8953	47.4017	8.4001	Dietikon	16
3	8302	47.4515	8.5849	ülach	12
4	8610	47.3471	8.7209	Ust	12
5	8800	47.2918	8.5635	Horgen	10
6	8623	47.3264	8.7978	Hinwil	9
7	8105	47.4341	8.4687	Dielsdorf	8
8	8910	47.2774	8.4513	Affoltern	7
9	8712	47.2425	8.7234	Meilen	5
10	8320	47.3878	8.7515	Pfäffikon	5
11	8245	47.6905	8.6436	Andelfingen	5

[44]:
```
bezirk_Venues =␣
↪getNearbyVenues(names=selected_bezirk['Bezirk'],latitudes=selected_bezirk['Latitude'],longi
```

```
Zürich
Winterthu
Dietikon
ülach
Ust
Horgen
Hinwil
Dielsdorf
Affoltern
Meilen
Pfäffikon
Andelfingen
```

[63]:
```
bezirk_Venues.to_csv(r'bezirk_Venues_whole.csv', index = False, header=True)
bezirk_Venues = pd.read_csv('bezirk_Venues_whole.csv')
bezirk_Venues
```

[63]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	\
0	Zürich	47.3828	8.5307	
1	Zürich	47.3828	8.5307	
2	Zürich	47.3828	8.5307	
3	Zürich	47.3828	8.5307	
4	Zürich	47.3828	8.5307	
5	Zürich	47.3828	8.5307	
6	Zürich	47.3828	8.5307	
7	Zürich	47.3828	8.5307	
8	Zürich	47.3828	8.5307	
9	Zürich	47.3828	8.5307	
10	Zürich	47.3828	8.5307	
11	Zürich	47.3828	8.5307	
12	Zürich	47.3828	8.5307	
13	Zürich	47.3828	8.5307	
14	Zürich	47.3828	8.5307	
15	Zürich	47.3828	8.5307	
16	Zürich	47.3828	8.5307	
17	Zürich	47.3828	8.5307	
18	Zürich	47.3828	8.5307	
19	Zürich	47.3828	8.5307	
20	Zürich	47.3828	8.5307	
21	Zürich	47.3828	8.5307	
22	Zürich	47.3828	8.5307	
23	Zürich	47.3828	8.5307	
24	Zürich	47.3828	8.5307	
25	Zürich	47.3828	8.5307	
26	Zürich	47.3828	8.5307	
27	Zürich	47.3828	8.5307	
28	Zürich	47.3828	8.5307	
29	Zürich	47.3828	8.5307	
30	Zürich	47.3828	8.5307	
31	Zürich	47.3828	8.5307	
32	Zürich	47.3828	8.5307	
33	Zürich	47.3828	8.5307	
34	Zürich	47.3828	8.5307	
35	Zürich	47.3828	8.5307	
36	Zürich	47.3828	8.5307	
37	Zürich	47.3828	8.5307	
38	Zürich	47.3828	8.5307	
39	Zürich	47.3828	8.5307	
40	Zürich	47.3828	8.5307	
41	Zürich	47.3828	8.5307	

42	Zürich	47.3828	8.5307
43	Zürich	47.3828	8.5307
44	Zürich	47.3828	8.5307
45	Zürich	47.3828	8.5307
46	Zürich	47.3828	8.5307
47	Zürich	47.3828	8.5307
48	Zürich	47.3828	8.5307
49	Zürich	47.3828	8.5307
50	Winterthu	47.4967	8.7342
51	Winterthu	47.4967	8.7342
52	Winterthu	47.4967	8.7342
53	Winterthu	47.4967	8.7342
54	Winterthu	47.4967	8.7342
55	Winterthu	47.4967	8.7342
56	Winterthu	47.4967	8.7342
57	Winterthu	47.4967	8.7342
58	Winterthu	47.4967	8.7342
59	Winterthu	47.4967	8.7342
60	Winterthu	47.4967	8.7342
61	Winterthu	47.4967	8.7342
62	Winterthu	47.4967	8.7342
63	Winterthu	47.4967	8.7342
64	Winterthu	47.4967	8.7342
65	Winterthu	47.4967	8.7342
66	Winterthu	47.4967	8.7342
67	Dietikon	47.4017	8.4001
68	Dietikon	47.4017	8.4001
69	Dietikon	47.4017	8.4001
70	Dietikon	47.4017	8.4001
71	Dietikon	47.4017	8.4001
72	Dietikon	47.4017	8.4001
73	Dietikon	47.4017	8.4001
74	Dietikon	47.4017	8.4001
75	Dietikon	47.4017	8.4001
76	ülach	47.4515	8.5849
77	ülach	47.4515	8.5849
78	ülach	47.4515	8.5849
79	ülach	47.4515	8.5849
80	ülach	47.4515	8.5849
81	ülach	47.4515	8.5849
82	ülach	47.4515	8.5849
83	ülach	47.4515	8.5849
84	ülach	47.4515	8.5849
85	ülach	47.4515	8.5849
86	ülach	47.4515	8.5849
87	ülach	47.4515	8.5849
88	ülach	47.4515	8.5849

89	ülach	47.4515	8.5849
90	ülach	47.4515	8.5849
91	ülach	47.4515	8.5849
92	ülach	47.4515	8.5849
93	ülach	47.4515	8.5849
94	Ust	47.3471	8.7209
95	Ust	47.3471	8.7209
96	Ust	47.3471	8.7209
97	Ust	47.3471	8.7209
98	Ust	47.3471	8.7209
99	Ust	47.3471	8.7209
100	Ust	47.3471	8.7209
101	Ust	47.3471	8.7209
102	Ust	47.3471	8.7209
103	Ust	47.3471	8.7209
104	Ust	47.3471	8.7209
105	Horgen	47.2918	8.5635
106	Horgen	47.2918	8.5635
107	Horgen	47.2918	8.5635
108	Horgen	47.2918	8.5635
109	Horgen	47.2918	8.5635
110	Horgen	47.2918	8.5635
111	Horgen	47.2918	8.5635
112	Horgen	47.2918	8.5635
113	Horgen	47.2918	8.5635
114	Hinwil	47.3264	8.7978
115	Hinwil	47.3264	8.7978
116	Hinwil	47.3264	8.7978
117	Hinwil	47.3264	8.7978
118	Hinwil	47.3264	8.7978
119	Dielsdorf	47.4341	8.4687
120	Dielsdorf	47.4341	8.4687
121	Dielsdorf	47.4341	8.4687
122	Dielsdorf	47.4341	8.4687
123	Dielsdorf	47.4341	8.4687
124	Dielsdorf	47.4341	8.4687
125	Dielsdorf	47.4341	8.4687
126	Dielsdorf	47.4341	8.4687
127	Dielsdorf	47.4341	8.4687
128	Dielsdorf	47.4341	8.4687
129	Dielsdorf	47.4341	8.4687
130	Affoltern	47.2774	8.4513
131	Affoltern	47.2774	8.4513
132	Affoltern	47.2774	8.4513
133	Affoltern	47.2774	8.4513
134	Affoltern	47.2774	8.4513
135	Affoltern	47.2774	8.4513

136	Affoltern	47.2774	8.4513
137	Meilen	47.2425	8.7234
138	Meilen	47.2425	8.7234
139	Meilen	47.2425	8.7234
140	Meilen	47.2425	8.7234
141	Meilen	47.2425	8.7234
142	Meilen	47.2425	8.7234
143	Meilen	47.2425	8.7234
144	Pfäffikon	47.3878	8.7515
145	Pfäffikon	47.3878	8.7515
146	Pfäffikon	47.3878	8.7515
147	Pfäffikon	47.3878	8.7515
148	Pfäffikon	47.3878	8.7515
149	Pfäffikon	47.3878	8.7515
150	Pfäffikon	47.3878	8.7515
151	Andelfingen	47.6905	8.6436
152	Andelfingen	47.6905	8.6436
153	Andelfingen	47.6905	8.6436
154	Andelfingen	47.6905	8.6436

	Venue	Venue Latitude	Venue Longitude	\
0	Riffraff	47.382831	8.529254	
1	Café Noir	47.382054	8.530643	
2	Restaurant JOSEF	47.384006	8.529020	
3	Co Chin Chin	47.383809	8.528480	
4	SASU Juicebar	47.382506	8.529888	
5	Wesley's Kitchen	47.381832	8.532534	
6	vineria centrale	47.382470	8.529931	
7	Maison Blunt	47.383721	8.528347	
8	The International Beer Bar	47.382793	8.528632	
9	Lily's Stomach Supply	47.382535	8.529540	
10	Paninoteca Il Pentagramma	47.381123	8.533554	
11	25hours Hotel Langstrasse	47.380379	8.528638	
12	Eldorado	47.383783	8.532737	
13	Kosmos	47.380094	8.528910	
14	MAME	47.384863	8.527179	
15	Casa Aurelio	47.383161	8.529686	
16	Tibetasia	47.384629	8.527870	
17	Auer & Co	47.384943	8.533642	
18	Restaurant Valentin's	47.384740	8.529904	
19	Damas	47.384584	8.527610	
20	Kokoro	47.380680	8.526546	
21	Maneez	47.384291	8.528576	
22	Kosmos Bistro	47.380172	8.528535	
23	Chiang Mai Thai Shop	47.380739	8.533633	
24	Flussbad Oberer Letten	47.385223	8.535269	
25	Museum für Gestaltung	47.382870	8.535706	

26	Restaurant Shaba	47.384143	8.531652
27	Der Grüne Libanon	47.380738	8.534835
28	Bäckerei Jung	47.384179	8.532339
29	El Maiz	47.381061	8.533766
30	Cinchona	47.380368	8.528337
31	Panama Bar & Restaurant	47.385167	8.535383
32	Gelati Tellhof	47.378696	8.528853
33	Tessin Grotto	47.386278	8.528736
34	Restaurant Vulkan	47.381918	8.533652
35	Kauz	47.381611	8.536998
36	Phuket Asia Center	47.380210	8.526542
37	New Point	47.381906	8.535398
38	Yoshino	47.380757	8.534579
39	Ynos	47.379202	8.531694
40	4leoni	47.384330	8.530277
41	Total Bar	47.378663	8.528781
42	India Street-Food	47.383168	8.530197
43	El Luchador	47.381852	8.533621
44	Photobastei	47.384764	8.534153
45	Pho Na	47.381502	8.533184
46	Kinski	47.379007	8.527326
47	Primitivo	47.385072	8.536177
48	Il Ristorantino	47.382133	8.535712
49	Sala of Tokyo	47.380990	8.536480
50	La Cyma	47.498514	8.732239
51	Fotomuseum Winterthur	47.495739	8.738024
52	Riva	47.498355	8.730183
53	Tibits	47.499980	8.731118
54	Holy Cow Winterthur	47.499008	8.730335
55	Albani Music Club	47.498395	8.729155
56	Das schmale Handtuch	47.497469	8.727686
57	Restaurant Al Giardino	47.494340	8.739810
58	Cappuccino	47.498760	8.730323
59	Restaurant Schäfli	47.498909	8.731258
60	Park Hotel	47.500937	8.732887
61	bloom	47.500976	8.732610
62	Alltag	47.499256	8.731113
63	Restaurant zur Sonne	47.499683	8.729813
64	Migros	47.499868	8.730988
65	Loge	47.498595	8.731209
66	Unterer Graben	47.499124	8.731070
67	Migros	47.405140	8.402469
68	Tomate	47.405364	8.403364
69	Coop	47.403808	8.402275
70	Hotel Sommerau Ticino Dietikon	47.402416	8.405423
71	Pizza Blitz	47.404090	8.403527
72	Sushi Ko&Ja	47.404287	8.403746

73	ZVV Sommerau	47.402790	8.405746
74	Müller Drogerie	47.403118	8.406105
75	ALDI Suisse	47.403000	8.406100
76	Zermatt Baracca	47.451248	8.583692
77	Rolli's Steakhouse	47.452682	8.580784
78	Migros	47.450963	8.581542
79	Bäckerei Fleischli	47.449901	8.582243
80	Hotel Allegra	47.448372	8.584502
81	Welcome Inn	47.449323	8.581844
82	Coop	47.453253	8.580623
83	Bahnhof Kloten	47.448544	8.583487
84	Suanlong	47.452582	8.582817
85	Zum Goldenen Tor	47.451142	8.583096
86	Asia Lotus Thai Restaurant	47.452347	8.583036
87	Denner	47.453243	8.583020
88	Bar Nelson	47.451778	8.580726
89	Otto's	47.453462	8.580246
90	Hardy's	47.447649	8.583987
91	Zum Wilden Mann	47.453465	8.579886
92	Wilder Mann	47.453631	8.579517
93	Restaurant Wilden Mann	47.453893	8.579492
94	Ichiban 178	47.348300	8.715801
95	Migros	47.350225	8.716156
96	McDonald's	47.350190	8.718192
97	Argentina Steakhouse Uster	47.344916	8.718102
98	Thai-Rütli	47.350303	8.720198
99	Restaurant zur Kastanie	47.346644	8.715674
100	avec. Uster	47.350353	8.718330
101	Peking Garden	47.350408	8.717946
102	Brezelkönig	47.350647	8.717953
103	Dieci	47.350285	8.717162
104	Stadthofsaal Uster	47.349044	8.715345
105	Rebstock	47.289426	8.566911
106	Hotel Sedartis	47.295527	8.565793
107	Luna	47.287976	8.563841
108	Migros	47.294407	8.563798
109	Bahnhof Thalwil	47.295857	8.564399
110	Tempo Sport	47.294567	8.564957
111	Restaurant Höfli	47.293855	8.567156
112	Restaurant Bahnhof	47.295351	8.564653
113	Gnuss pur	47.296213	8.563473
114	Molly Malone Irish Pub	47.327327	8.801669
115	Migros	47.327407	8.799791
116	Coop	47.326200	8.798080
117	Oberland Märt	47.326728	8.800091
118	Kino Palace	47.322019	8.797327
119	Mövenpick Hotel Zürich-Regensdorf	47.431334	8.467797

120	Gasthof Hirschen	47.430130	8.465891
121	Fitnesspark Regensdorf	47.430683	8.469147
122	Coop	47.430948	8.467511
123	Bahnhof Regensdorf-Watt	47.436767	8.472145
124	Restaurant Hot Wok	47.431187	8.471325
125	Zentrum Regensdorf	47.431084	8.469701
126	SportXX	47.431164	8.469065
127	Francos Pizza	47.431110	8.467618
128	Trattoria Ciao	47.431017	8.468219
129	Ex Libris	47.430602	8.469794
130	Migros	47.278246	8.452972
131	Bahnhof Affoltern am Albis	47.275652	8.446448
132	Albis Lodge	47.276993	8.450906
133	Lidl Schweiz	47.278016	8.452600
134	Central Restaurant	47.278810	8.454538
135	Kronenplatz	47.281053	8.453485
136	Coop Restaurant	47.274393	8.446450
137	Migros	47.240507	8.723402
138	Coop	47.241591	8.724710
139	Bahnhof Stäfa	47.240542	8.722031
140	ALDI Suisse	47.241478	8.718436
141	Obstgarten	47.241193	8.728681
142	Restaurant Rössli	47.239492	8.718948
143	Zum Rössli	47.239415	8.718948
144	Steiner	47.387917	8.752306
145	Gasthof Zum Hecht, Fehraltorf	47.387221	8.753142
146	Migros	47.386470	8.751677
147	Kaminfegermeister R. E. Wehrli	47.389040	8.749430
148	Migros Fehraltorf	47.386724	8.754323
149	GameTurnier.ch	47.388464	8.755192
150	Restaurant&Lounge Pumpkin	47.388284	8.745215
151	Coop Rhy Markt	47.691044	8.646599
152	Bhf Feuerthalen	47.691940	8.646147
153	meinekosmetikerin.ch Barbara Pochon	47.693820	8.642540
154	lindli	47.694465	8.646090

	Venue Category
0	Movie Theater
1	Café
2	Swiss Restaurant
3	Vietnamese Restaurant
4	Juice Bar
5	Asian Restaurant
6	Bar
7	Moroccan Restaurant
8	Bar
9	Asian Restaurant

```
10                 Sandwich Place
11                         Hotel
12                           Bar
13               Cultural Center
14                   Coffee Shop
15             Spanish Restaurant
16                Thai Restaurant
17                          Café
18             Turkish Restaurant
19       Middle Eastern Restaurant
20             Japanese Restaurant
21              Indian Restaurant
22                        Bistro
23                Thai Restaurant
24                   Bathing Area
25                         Museum
26                          Café
27             Falafel Restaurant
28                         Bakery
29                   Gourmet Shop
30                      Hotel Bar
31                           Bar
32                Ice Cream Shop
33             Italian Restaurant
34              Indian Restaurant
35                       Speakeasy
36              Asian Restaurant
37                       BBQ Joint
38              Sushi Restaurant
39     Modern European Restaurant
40             Italian Restaurant
41                           Bar
42              Indian Restaurant
43             Mexican Restaurant
44                    Art Gallery
45          Vietnamese Restaurant
46                     Rock Club
47                   Beer Garden
48             Italian Restaurant
49              Asian Restaurant
50                          Café
51                    Art Museum
52                          Café
53  Vegetarian / Vegan Restaurant
54                   Burger Joint
55                      Nightclub
56                           Bar
```

57	Italian Restaurant
58	Bar
59	Restaurant
60	Hotel
61	Hotel Bar
62	Café
63	Swiss Restaurant
64	Grocery Store
65	Hotel Bar
66	Plaza
67	Supermarket
68	Italian Restaurant
69	Supermarket
70	Hotel
71	Italian Restaurant
72	Sushi Restaurant
73	Bus Stop
74	Drugstore
75	Supermarket
76	Swiss Restaurant
77	Steakhouse
78	Supermarket
79	Bakery
80	Hotel
81	Hotel
82	Supermarket
83	Train Station
84	Asian Restaurant
85	Restaurant
86	Thai Restaurant
87	Discount Store
88	Pub
89	Department Store
90	Bar
91	Bus Station
92	Pizza Place
93	Restaurant
94	Japanese Restaurant
95	Grocery Store
96	Fast Food Restaurant
97	Argentinian Restaurant
98	Asian Restaurant
99	Italian Restaurant
100	Snack Place
101	Chinese Restaurant
102	Snack Place
103	Ice Cream Shop

104	Theater
105	Swiss Restaurant
106	Hotel
107	Mexican Restaurant
108	Grocery Store
109	Train Station
110	Sporting Goods Shop
111	Asian Restaurant
112	Falafel Restaurant
113	Bakery
114	Irish Pub
115	Grocery Store
116	Supermarket
117	Shopping Mall
118	Movie Theater
119	Hotel
120	Restaurant
121	Gym / Fitness Center
122	Supermarket
123	Train Station
124	Asian Restaurant
125	Shopping Mall
126	Sporting Goods Shop
127	Pizza Place
128	Restaurant
129	Bookstore
130	Grocery Store
131	Train Station
132	Tapas Restaurant
133	Supermarket
134	Eastern European Restaurant
135	Plaza
136	Restaurant
137	Supermarket
138	Supermarket
139	Train Station
140	Supermarket
141	Bus Station
142	Restaurant
143	Restaurant
144	Bakery
145	Restaurant
146	Food & Drink Shop
147	Construction & Landscaping
148	Food & Drink Shop
149	Soccer Field
150	Restaurant

```
151                  Shopping Mall
152                  Train Station
153                Cosmetics Shop
154                          River
```

[46]:
```
print(bezirk_Venues.shape)
bezirk_Venues.head()
```

```
(155, 7)
```

[46]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Zürich	47.3828	8.5307	Riffraff	47.382831	8.529254	Movie Theater
1	Zürich	47.3828	8.5307	Café Noir	47.382054	8.530643	Café
2	Zürich	47.3828	8.5307	Restaurant JOSEF	47.384006	8.529020	Swiss Restaurant
3	Zürich	47.3828	8.5307	Co Chin Chin	47.383809	8.528480	Vietnamese Restaurant
4	Zürich	47.3828	8.5307	SASU Juicebar	47.382506	8.529888	Juice Bar

[47]:
```
bezirk_Venues.groupby('Neighborhood').count()
print('There are {} uniques categories.'.format(len(bezirk_Venues['Venue␣
 ↪Category'].unique())))
```

```
There are 69 uniques categories.
```

0.8 Analyse Each Bezirk

[48]:
```
# one hot encoding
bezirk_onehot = pd.get_dummies(bezirk_Venues[['Venue Category']], prefix="",␣
 ↪prefix_sep="")
# add neighborhood column back to dataframe
bezirk_onehot['Neighborhood'] = bezirk_Venues['Neighborhood']
# move neighborhood column to the first column
fixed_columns = [bezirk_onehot.columns[-1]] + list(bezirk_onehot.columns[:-1])
bezirk_onehot = bezirk_onehot[fixed_columns]
bezirk_onehot.head()
```

[48]:

	Neighborhood	Argentinian Restaurant	Art Gallery	Art Museum
0	Zürich	0	0	0
1	Zürich	0	0	0
2	Zürich	0	0	0
3	Zürich	0	0	0

```
4       Zürich                       0            0          0

   Asian Restaurant  BBQ Joint  Bakery  Bar  Bathing Area  Beer Garden  \
0                 0          0       0    0             0            0
1                 0          0       0    0             0            0
2                 0          0       0    0             0            0
3                 0          0       0    0             0            0
4                 0          0       0    0             0            0

   Bistro  Bookstore  Burger Joint  Bus Station  Bus Stop  Café  \
0       0          0             0            0         0     0
1       0          0             0            0         0     1
2       0          0             0            0         0     0
3       0          0             0            0         0     0
4       0          0             0            0         0     0

   Chinese Restaurant  Coffee Shop  Construction & Landscaping  \
0                   0            0                           0
1                   0            0                           0
2                   0            0                           0
3                   0            0                           0
4                   0            0                           0

   Cosmetics Shop  Cultural Center  Department Store  Discount Store  \
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

   Drugstore  Eastern European Restaurant  Falafel Restaurant  \
0          0                            0                   0
1          0                            0                   0
2          0                            0                   0
3          0                            0                   0
4          0                            0                   0

   Fast Food Restaurant  Food & Drink Shop  Gourmet Shop  Grocery Store  \
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

   Gym / Fitness Center  Hotel  Hotel Bar  Ice Cream Shop  Indian Restaurant  \
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

   Irish Pub  Italian Restaurant  Japanese Restaurant  Juice Bar  \
0          0                   0                    0          0
1          0                   0                    0          0
2          0                   0                    0          0
3          0                   0                    0          0
4          0                   0                    0          1

   Mexican Restaurant  Middle Eastern Restaurant  Modern European Restaurant  \
0                   0                          0                           0
1                   0                          0                           0
2                   0                          0                           0
3                   0                          0                           0
4                   0                          0                           0

   Moroccan Restaurant  Movie Theater  Museum  Nightclub  Pizza Place  Plaza  \
0                    0              1       0          0            0      0
1                    0              0       0          0            0      0
2                    0              0       0          0            0      0
3                    0              0       0          0            0      0
4                    0              0       0          0            0      0

   Pub  Restaurant  River  Rock Club  Sandwich Place  Shopping Mall  \
0    0           0      0          0               0              0
1    0           0      0          0               0              0
2    0           0      0          0               0              0
3    0           0      0          0               0              0
4    0           0      0          0               0              0

   Snack Place  Soccer Field  Spanish Restaurant  Speakeasy  \
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

   Sporting Goods Shop  Steakhouse  Supermarket  Sushi Restaurant  \
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

   Swiss Restaurant  Tapas Restaurant  Thai Restaurant  Theater  \
```

```
0                  0                  0                  0        0
1                  0                  0                  0        0
2                  1                  0                  0        0
3                  0                  0                  0        0
4                  0                  0                  0        0

   Train Station  Turkish Restaurant  Vegetarian / Vegan Restaurant  \
0              0                   0                              0
1              0                   0                              0
2              0                   0                              0
3              0                   0                              0
4              0                   0                              0

   Vietnamese Restaurant
0                      0
1                      0
2                      0
3                      1
4                      0
```

[49]:
```
bezirk_onehot.shape
```

[49]:
```
(155, 70)
```

[50]:
```
bezirk_grouped = bezirk_onehot.groupby('Neighborhood').mean().reset_index()
bezirk_grouped
bezirk_grouped.shape
```

[50]:
```
(12, 70)
```

0.9 Display top 10 Venues from each Bezirk

[51]:
```
num_top_venues = 10

for hood in bezirk_grouped['Neighborhood']:
    #print("----"+hood+"----")
    temp = bezirk_grouped[bezirk_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')
```

```
                      venue  freq
```

```
0               Grocery Store  0.14
1                  Restaurant  0.14
2               Train Station  0.14
3  Eastern European Restaurant  0.14
4                      Plaza  0.14
5           Tapas Restaurant  0.14
6                Supermarket  0.14
7                    Museum  0.00
8                       Pub  0.00
9                Pizza Place  0.00
```

```
          venue  freq
0   Shopping Mall  0.25
1           River  0.25
2   Train Station  0.25
3  Cosmetics Shop  0.25
4       Nightclub  0.00
5       Rock Club  0.00
6      Restaurant  0.00
7             Pub  0.00
8           Plaza  0.00
9     Pizza Place  0.00
```

```
                venue  freq
0          Restaurant  0.18
1           Bookstore  0.09
2       Shopping Mall  0.09
3    Asian Restaurant  0.09
4       Train Station  0.09
5               Hotel  0.09
6  Gym / Fitness Center  0.09
7         Pizza Place  0.09
8   Sporting Goods Shop  0.09
9         Supermarket  0.09
```

```
                venue  freq
0         Supermarket  0.33
1   Italian Restaurant  0.22
2               Hotel  0.11
3           Drugstore  0.11
4            Bus Stop  0.11
5     Sushi Restaurant  0.11
6  Moroccan Restaurant  0.00
7        Movie Theater  0.00
8              Museum  0.00
```

```
9            Rock Club  0.00
```

```
                    venue  freq
0                Irish Pub   0.2
1              Supermarket   0.2
2            Grocery Store   0.2
3            Movie Theater   0.2
4            Shopping Mall   0.2
5                  Theater   0.0
6           Thai Restaurant   0.0
7                Juice Bar   0.0
8       Mexican Restaurant   0.0
9  Middle Eastern Restaurant   0.0
```

```
                venue  freq
0   Mexican Restaurant  0.11
1        Train Station  0.11
2   Falafel Restaurant  0.11
3  Sporting Goods Shop  0.11
4        Grocery Store  0.11
5                Hotel  0.11
6     Swiss Restaurant  0.11
7              Bakery  0.11
8     Asian Restaurant  0.11
9  Moroccan Restaurant  0.00
```

```
                  venue  freq
0             Supermarket  0.43
1              Restaurant  0.29
2           Train Station  0.14
3             Bus Station  0.14
4  Argentinian Restaurant  0.00
5                   River  0.00
6                     Pub  0.00
7                  Plaza  0.00
8             Pizza Place  0.00
9               Nightclub  0.00
```

```
                    venue  freq
0                Restaurant  0.29
1          Food & Drink Shop  0.29
2  Construction & Landscaping  0.14
3               Soccer Field  0.14
4                     Bakery  0.14
```

```
5      Argentinian Restaurant  0.00
6                  Pizza Place  0.00
7                        River  0.00
8                          Pub  0.00
9                        Plaza  0.00
```

```
                   venue  freq
0             Snack Place  0.18
1  Argentinian Restaurant  0.09
2     Japanese Restaurant  0.09
3        Asian Restaurant  0.09
4      Italian Restaurant  0.09
5                 Theater  0.09
6          Ice Cream Shop  0.09
7    Fast Food Restaurant  0.09
8      Chinese Restaurant  0.09
9           Grocery Store  0.09
```

```
                           venue  freq
0                           Café  0.18
1                            Bar  0.12
2                      Hotel Bar  0.12
3                   Burger Joint  0.06
4                  Grocery Store  0.06
5                     Art Museum  0.06
6  Vegetarian / Vegan Restaurant  0.06
7              Italian Restaurant  0.06
8                      Restaurant  0.06
9                       Nightclub  0.06
```

```
                  venue  freq
0                   Bar  0.10
1       Asian Restaurant  0.08
2      Indian Restaurant  0.06
3     Italian Restaurant  0.06
4                   Café  0.06
5  Vietnamese Restaurant  0.04
6        Thai Restaurant  0.04
7           Gourmet Shop  0.02
8                 Hotel  0.02
9             Hotel Bar  0.02
```

```
             venue  freq
0       Restaurant  0.11
```

```
1             Hotel  0.11
2       Supermarket  0.11
3       Bus Station  0.06
4       Pizza Place  0.06
5  Asian Restaurant  0.06
6    Discount Store  0.06
7           Bakery  0.06
8              Bar  0.06
9     Train Station  0.06
```

0.10 to pandas dataframe

[53]:
```
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]
```

0.11 Display 10 Venues for selected Bezirks

[54]:
```
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
bezirk_venues_sorted = pd.DataFrame(columns=columns)
bezirk_venues_sorted['Neighborhood'] = bezirk_grouped['Neighborhood']

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

bezirk_venues_sorted.head()
```

[54]:
```
  Neighborhood 1st Most Common Venue 2nd Most Common Venue  \
0    Affoltern                 Plaza         Train Station
1  Andelfingen         Train Station         Shopping Mall
```

```
2    Dielsdorf            Restaurant  Gym / Fitness Center
3     Dietikon           Supermarket    Italian Restaurant
4       Hinwil             Irish Pub         Grocery Store

  3rd Most Common Venue 4th Most Common Venue  5th Most Common Venue  \
0         Grocery Store      Tapas Restaurant            Supermarket
1        Cosmetics Shop                 River  Vietnamese Restaurant
2           Supermarket                 Hotel          Shopping Mall
3                 Hotel              Bus Stop       Sushi Restaurant
4         Shopping Mall         Movie Theater            Supermarket

  6th Most Common Venue        7th Most Common Venue  8th Most Common Venue  \
0            Restaurant  Eastern European Restaurant  Vietnamese Restaurant
1             Drugstore   Construction & Landscaping        Cultural Center
2             Bookstore                 Pizza Place    Sporting Goods Shop
3             Drugstore           Falafel Restaurant        Cultural Center
4          Gourmet Shop            Food & Drink Shop   Fast Food Restaurant

  9th Most Common Venue      10th Most Common Venue
0        Discount Store  Construction & Landscaping
1      Department Store              Discount Store
2      Asian Restaurant               Train Station
3      Department Store              Discount Store
4    Falafel Restaurant                 Coffee Shop
```

0.12 Cluster the Bezirks

Run k-means to cluster the neighborhood into 3 clusters.

[55]:
```
# set number of clusters
kclusters = 3

bezirk_grouped_clustering = bezirk_grouped.drop('Neighborhood', 1)

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

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

[55]:
```
array([0, 0, 0, 2, 0, 0, 2, 1, 0, 0], dtype=int32)
```

[56]:
```
# add clustering labels
bezirk_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)
```

```
bezirk_merged = selected_bezirk ## Here we need to copy the Bezirk dataframe␣
 ↪with Lat,Long info

# merge manhattan_grouped with manhattan_data to add latitude/longitude for␣
 ↪each neighborhood
bezirk_merged = bezirk_merged.join(bezirk_venues_sorted.
 ↪set_index('Neighborhood'), on='Bezirk')

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

[56]:

	PLZ	Latitude	Longitude	Bezirk	N_Restaurants	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	8086	47.3828	8.5307	Zürich	38	0	Bar	Asian Restaurant	Café	Indian Restaurant	Italian Restaurant	Vietnamese Restaurant	Thai Restaurant	Museum	Moroccan Restaurant	Modern European Restaurant
1	8403	47.4967	8.7342	Winterthu	18	0	Café	Hotel Bar	Bar	Nightclub	Restaurant	Hotel	Burger Joint	Plaza	Swiss Restaurant	Italian Restaurant
2	8953	47.4017	8.4001	Dietikon	16	2	Supermarket	Italian Restaurant	Hotel	Bus Stop	Sushi Restaurant	Drugstore	Falafel Restaurant	Cultural Center	Department Store	Discount Store
3	8302	47.4515	8.5849	ülach	12	0	Hotel	Restaurant	Supermarket	Steakhouse	Pizza Place	Pub	Department Store	Bus Station	Discount Store	Bakery
4	8610	47.3471	8.7209	Ust	12	0	Snack Place	Argentinian Restaurant	Italian Restaurant	Japanese Restaurant	Grocery Store	Fast Food Restaurant	Chinese Restaurant	Ice Cream Shop	Theater	Asian Restaurant

[57]:
```
# 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(bezirk_merged['Latitude'],␣
 ↪bezirk_merged['Longitude'], bezirk_merged['Bezirk'], bezirk_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
```

[57]:
```
<folium.folium.Map at 0x7f2fbde0e9b0>
```

[58]:
```
#Cluster 0
bezirk_merged.loc[bezirk_merged['Cluster Labels'] == 0, bezirk_merged.
 ↪columns[[1] + list(range(2, bezirk_merged.shape[1]))]]
```

[58]:

	Latitude	Longitude	Bezirk	N_Restaurants	Cluster Labels	\
0	47.3828	8.5307	Zürich	38	0	
1	47.4967	8.7342	Winterthu	18	0	
3	47.4515	8.5849	ülach	12	0	
4	47.3471	8.7209	Ust	12	0	
5	47.2918	8.5635	Horgen	10	0	
6	47.3264	8.7978	Hinwil	9	0	
7	47.4341	8.4687	Dielsdorf	8	0	
8	47.2774	8.4513	Affoltern	7	0	
11	47.6905	8.6436	Andelfingen	5	0	

	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	\
0	Bar	Asian Restaurant	Café	
1	Café	Hotel Bar	Bar	
3	Hotel	Restaurant	Supermarket	

```
4            Snack Place  Argentinian Restaurant    Italian Restaurant
5                  Hotel        Swiss Restaurant    Falafel Restaurant
6              Irish Pub           Grocery Store         Shopping Mall
7             Restaurant    Gym / Fitness Center           Supermarket
8                  Plaza           Train Station         Grocery Store
11         Train Station           Shopping Mall        Cosmetics Shop

   4th Most Common Venue  5th Most Common Venue  6th Most Common Venue  \
0      Indian Restaurant     Italian Restaurant  Vietnamese Restaurant
1              Nightclub             Restaurant                 Hotel
3             Steakhouse           Pizza Place                   Pub
4    Japanese Restaurant          Grocery Store   Fast Food Restaurant
5    Sporting Goods Shop                Bakery     Mexican Restaurant
6          Movie Theater           Supermarket           Gourmet Shop
7                  Hotel          Shopping Mall              Bookstore
8       Tapas Restaurant           Supermarket             Restaurant
11                 River  Vietnamese Restaurant              Drugstore

          7th Most Common Venue  8th Most Common Venue 9th Most Common Venue  \
0              Thai Restaurant                Museum   Moroccan Restaurant
1                 Burger Joint                 Plaza      Swiss Restaurant
3             Department Store           Bus Station        Discount Store
4            Chinese Restaurant         Ice Cream Shop               Theater
5             Asian Restaurant          Grocery Store         Train Station
6            Food & Drink Shop   Fast Food Restaurant    Falafel Restaurant
7                  Pizza Place    Sporting Goods Shop      Asian Restaurant
8   Eastern European Restaurant  Vietnamese Restaurant        Discount Store
11   Construction & Landscaping        Cultural Center      Department Store

        10th Most Common Venue
0   Modern European Restaurant
1           Italian Restaurant
3                       Bakery
4              Asian Restaurant
5                  Gourmet Shop
6                   Coffee Shop
7                 Train Station
8   Construction & Landscaping
11              Discount Store
```

[59]:
```
#Cluster 1
bezirk_merged.loc[bezirk_merged['Cluster Labels'] == 1, bezirk_merged.
 ↪columns[[1] + list(range(2, bezirk_merged.shape[1]))]]
```

[59]:
```
    Latitude  Longitude     Bezirk  N_Restaurants  Cluster Labels  \
10   47.3878     8.7515  Pfäffikon              5               1
```

```
   1st Most Common Venue 2nd Most Common Venue 3rd Most Common Venue  \
10            Restaurant     Food & Drink Shop          Soccer Field

   4th Most Common Venue       5th Most Common Venue 6th Most Common Venue  \
10               Bakery  Construction & Landscaping             Drugstore

   7th Most Common Venue 8th Most Common Venue 9th Most Common Venue  \
10        Cosmetics Shop       Cultural Center      Department Store

   10th Most Common Venue
10         Discount Store
```

[60]:
```
#Cluster 2
bezirk_merged.loc[bezirk_merged['Cluster Labels'] == 2, bezirk_merged.
↪columns[[1] + list(range(2, bezirk_merged.shape[1]))]]
```

[60]:
```
   Latitude  Longitude    Bezirk  N_Restaurants  Cluster Labels  \
2   47.4017     8.4001  Dietikon             16               2
9   47.2425     8.7234    Meilen              5               2

  1st Most Common Venue 2nd Most Common Venue 3rd Most Common Venue  \
2           Supermarket    Italian Restaurant                 Hotel
9           Supermarket            Restaurant         Train Station

  4th Most Common Venue        5th Most Common Venue 6th Most Common Venue  \
2             Bus Stop             Sushi Restaurant             Drugstore
9          Bus Station  Eastern European Restaurant        Cosmetics Shop

  7th Most Common Venue 8th Most Common Venue 9th Most Common Venue  \
2    Falafel Restaurant       Cultural Center      Department Store
9       Cultural Center      Department Store        Discount Store

  10th Most Common Venue
2         Discount Store
9              Drugstore
```

[ ]: