

Capstone Project - The Battle of Neighborhoods (Weeks 1-2)

Parisian restaurants

A description of the problem and a discussion of the background.

The last 2 lessons of the IBM Applied Data Science Capstone course allow us to use machine learning techniques on real-world projects. During this course we've learned segmentation and clustering methods. For this we've used Toronto, Ontario, Canada and New York city, the USA datasets. This data included 2 data sources: neighborhoods and Foursquare API data with its users reviews. So now, I'll try to implement received knowledge using Paris, France data about its Arrondissements (districts) and corresponding Quarters (neighborhoods).

The main goal of this project is to make a recommendation where in a city of my choice I would suggest to open a restaurant. I was interested in Paris, France as a city of my choice. The task for this project says about Paris - "very diverse and is the financial capital of country". In France there's a lot of tourist places, but with Paris it is very difficult to compete with any other city in France, and probably there will be enough data on restaurants and cafes.

A description of the data and how it will be used to solve the problem.

The following data (20 boroughs and 80 neighborhoods) was used to determine the best place to open a restaurant in Paris:

1. Arrondissements data (This dataset comes from a certified public service - data.gouv.fr). This data include information about districts names, postal_codes, their coordinates, and perimeter as well as density information. File name - arrondissements.csv

<https://www.data.gouv.fr/en/datasets/arrondissements-1/#>
(<https://www.data.gouv.fr/en/datasets/arrondissements-1/#>)

1. Quarters data (Wikipedia). This data includes quarter names and their areas, as well as corresponding arrondissements. To get this data BeautifulSoup module was used.

https://fr.wikipedia.org/wiki/Liste_des_quartiers_administratifs_de_Paris
(https://fr.wikipedia.org/wiki/Liste_des_quartiers_administratifs_de_Paris)

https://en.wikipedia.org/wiki/Quarters_of_Paris (https://en.wikipedia.org/wiki/Quarters_of_Paris) (English version)

1. Latitude and longitude of the given address using OpenStreetMap's Nominatim service. Neighborhood names, arrondissements' postal codes and "Paris, FR" info were used to get neighborhood coordinates through geolocator.geocode

<https://geopy.readthedocs.io/en/stable/> (<https://geopy.readthedocs.io/en/stable/>)

1. Foursquare API. It was used to get information about neighborhood venues.

<https://foursquare.com/developers/apps> (<https://foursquare.com/developers/apps>)

Methodology.

There were made 3 steps to achieve a goal of this project:

Part 1: Load and prepare data (data collection, cleaning, joining all datasets together).

Part 2: Get the latitude and the longitude coordinates of each neighborhood.

Part 3: Exploring and clustering the neighborhoods in Paris (visualization the resulting clusters). Clustering (K-Means) was used to analyze neighborhood data.

In []:

In [1]: !jupyter trust assignment2.ipynb

Signing notebook: assignment2.ipynb

Part 1: *Load and prepare data*

Import libraries.

```
In [2]: import numpy as np # Library to handle data in a vectorized manner

import pandas as pd # Library for data analysis
import copy
import os
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

import json # Library to handle JSON files
from pandas.io.json import json_normalize # tranform JSON file into a pandas d
ataframe

import requests # Library to handle requests
from bs4 import BeautifulSoup

import geocoder # convert an address into latitude and longitude values
from geopy.geocoders import Nominatim # convert an address into latitude and l
ongitude values

# 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

import folium # map rendering library

print('Libraries imported.')
```

Libraries imported.

In []:

Arrondissements of Paris - municipal boroughs.

```
In [3]: # Construct file name

file_dir = 'C:\\Sasha\\udacity\\coursera\\Applied Data Science Specialization
\\4 Applied Data Science Capstone\\4 The Battle of Neighborhoods'
file_name = 'arrondissements.csv'
result_path = os.path.join(file_dir, file_name)
```

In [4]: *# Load data*

```
boroughs = pd.read_csv(result_path, sep=';')
boroughs.head(2)
```

Out[4]:

	Identifiant séquentiel de l'arrondissement	Numéro d'arrondissement	Numéro d'arrondissement INSEE	Nom de l'arrondissement	Nom officiel de l'arrondissement	N_S
0	750000003	3	75103	3ème Ardt	Temple	7500
1	750000007	7	75107	7ème Ardt	Palais-Bourbon	7500



In [5]: *# Sort data, delete usefull columns, set index, and rename index and columns*

```
boroughs.sort_values('Numéro d'arrondissement INSEE', inplace=True)
boroughs.reset_index(drop=True, inplace=True)

list_to_drop = ['Identifiant séquentiel de l'arrondissement', 'N_SQ_CO', 'Geom
etry X Y', 'Geometry']
boroughs.drop(list_to_drop, axis=1, inplace=True)

boroughs.set_index('Numéro d'arrondissement', inplace=True)

boroughs.index.name = 'Identifiant'
boroughs.columns = ['PostalCode', 'Arrondissement', 'Borough', 'Arrondissement
\'s Area', 'Arrondissement\'s Perimeter']
```

In [6]: *# Edit PostalCode column*

```
boroughs.PostalCode = boroughs.PostalCode - 100
```

In [7]: *# Look at preliminary result*

boroughs

Out[7]:

	PostalCode	Arrondissement	Borough	Arrondissement's Area	Arrondissement's Perimeter
Identifier					
1	75001	1er Ardt	Louvre	1.824613e+06	6054.936862
2	75002	2ème Ardt	Bourse	9.911537e+05	4554.104360
3	75003	3ème Ardt	Temple	1.170883e+06	4519.263648
4	75004	4ème Ardt	Hôtel-de-Ville	1.600586e+06	5420.908434
5	75005	5ème Ardt	Panthéon	2.539375e+06	6239.195396
6	75006	6ème Ardt	Luxembourg	2.153096e+06	6483.686786
7	75007	7ème Ardt	Palais-Bourbon	4.090057e+06	8099.424883
8	75008	8ème Ardt	Élysée	3.880036e+06	7880.533268
9	75009	9ème Ardt	Opéra	2.178303e+06	6471.588290
10	75010	10ème Ardt	Entrepôt	2.891739e+06	6739.375055
11	75011	11ème Ardt	Popincourt	3.665442e+06	8282.011886
12	75012	12ème Ardt	Reuilly	1.631478e+07	24089.666298
13	75013	13ème Ardt	Gobelins	7.149311e+06	11546.546526
14	75014	14ème Ardt	Observatoire	5.614877e+06	10317.483310
15	75015	15ème Ardt	Vaugirard	8.494994e+06	13678.798315
16	75016	16ème Ardt	Passy	1.637254e+07	17416.109657
17	75017	17ème Ardt	Batignolles-Monceau	5.668835e+06	10775.579516
18	75018	18ème Ardt	Buttes-Montmartre	5.996051e+06	9916.464176
19	75019	19ème Ardt	Buttes-Chaumont	6.792651e+06	11253.182479
20	75020	20ème Ardt	Ménilmontant	5.983446e+06	10704.940486

In []:

Quartiers (Quarters) of Paris - municipal neighborhoods.

In [8]: neighborhood_num, neighborhood, area = [], [], []

```
In [9]: adress = 'https://fr.wikipedia.org/wiki/Liste_des_quartiers_administratifs_de_Paris'
page = requests.get(adress)
soup = BeautifulSoup(page.text, 'html.parser')

tags_table = soup.find('table', {'class': 'wikitable sortable'}) # find table with postal codes
table = tags_table.find_all('tr') # Links to cities
```

```
In [10]: for line in table:
    rows = line.find_all('td', style=None)
    if rows:
        if len(rows) == 7:
            cnt = 0
            for word in rows:
                if 'arrondissement' in word.text:
                    cnt = 1
                elif cnt == 1:
                    neighborhood_num.append(word.text)
                    cnt += 1
                elif cnt == 2:
                    neighborhood.append(word.text)
                    cnt += 1
                elif ',' in word.text:
                    area.append(word.text.replace(',', '.'))
                    cnt += 1
            elif len(rows) == 5:
                cnt = 0
                for word in rows:
                    if cnt == 0:
                        neighborhood_num.append(word.text)
                        cnt += 1
                    elif cnt == 1:
                        neighborhood.append(word.text)
                        cnt += 1
                    elif ',' in word.text:
                        area.append(word.text.replace(',', '.'))
                        cnt += 1
```

```
In [11]: # Create List of boroughs

borough = []
for i in range(1,21):
    borough += [i]*4
print('Len borough is', len(borough))
borough[:20]
```

Len borough is 80

```
Out[11]: [1, 1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5]
```

```
In [12]: # Check length of 4 lists (borough, neighborhood, neighborhood_num, and area)
len(borough), len(neighborhood), len(neighborhood_num), len(area)
```

Out[12]: (80, 80, 80, 80)

```
In [13]: # Instantiate the dataframe

neighborhoods = pd.DataFrame(zip(*(borough, neighborhood, neighborhood_num, ar
ea)),
                             columns=['Borough', 'Neighborhood', 'Neighborhood
\'s number', 'Neighborhood\'s Area'])

# Set index
neighborhoods.set_index('Borough', inplace=True)
neighborhoods.index.name = 'Identifier'

neighborhoods.head()
```

Out[13]:

	Neighborhood	Neighborhood's number	Neighborhood's Area
Identifier			
1	Saint-Germain-l'Auxerrois	1er	86.9
1	Halles	2e	41.2
1	Palais-Royal	3e	27.4
1	Place-Vendôme	4e	26.9
2	Gaillon	5e	18.8

In []:

Join boroughs and neighborhoods DataFrames.

The dataframe will consist of six columns: PostalCode, Borough, Arrondissement's Area, Arrondissement's Perimeter, Neighborhood, and Neighborhood's Area.

```
In [14]: # Join the dataframe

df = boroughs.join(neighborhoods)
df.reset_index(drop=True, inplace=True)
df.head()
```

Out[14]:

	PostalCode	Arrondissement	Borough	Arrondissement's Area	Arrondissement's Perimeter	Neighborhood	N
0	75001	1er Ardt	Louvre	1.824613e+06	6054.936862	Saint-Germain-l'Auxerrois	
1	75001	1er Ardt	Louvre	1.824613e+06	6054.936862	Halles	
2	75001	1er Ardt	Louvre	1.824613e+06	6054.936862	Palais-Royal	
3	75001	1er Ardt	Louvre	1.824613e+06	6054.936862	Place-Vendôme	
4	75002	2ème Ardt	Bourse	9.911537e+05	4554.104360	Gaillon	

```
In [15]: # Delete usefull columns

df.drop(['Arrondissement', 'Neighborhood\'s number'], axis=1, inplace=True)
```

Print the number of rows & columns in a dataframe and look at 1st 5 rows.

```
In [16]: print('Shape:', df.shape)
df.head()
```

Shape: (80, 6)

Out[16]:

	PostalCode	Borough	Arrondissement's Area	Arrondissement's Perimeter	Neighborhood	Neighborhood's Area
0	75001	Louvre	1.824613e+06	6054.936862	Saint-Germain-l'Auxerrois	86.9
1	75001	Louvre	1.824613e+06	6054.936862	Halles	41.2
2	75001	Louvre	1.824613e+06	6054.936862	Palais-Royal	27.4
3	75001	Louvre	1.824613e+06	6054.936862	Place-Vendôme	26.9
4	75002	Bourse	9.911537e+05	4554.104360	Gaillon	18.8

In []:

In []:

Part 2: Get the latitude and the longitude coordinates of each neighborhood


```
In [17]: def get_latlon(neighborhood, postal_code):  
  
    # initialize your variable to None  
    lat_lng_coords = None  
  
    # loop until you get the coordinates  
    while(lat_lng_coords is None):  
        g = geocoder.arcgis('{} {}, Paris, FR'.format(neighborhood, postal_code))  
        lat_lng_coords = g.latlng  
  
    latitude = lat_lng_coords[0]  
    longitude = lat_lng_coords[1]  
  
    return latitude, longitude
```

```
In [18]: # Apply function  
  
coordinates = []  
latitude, longitude = [], []  
  
for idx,row in df[['Neighborhood', 'PostalCode']].iterrows():  
    neighbor, code = row['Neighborhood'], row['PostalCode']  
    coordinates = get_latlon(neighbor, code)  
    latitude.append(coordinates[0])  
    longitude.append(coordinates[1])
```

```
In [19]: # Check the results  
  
latitude[:5], longitude[:5]
```

```
Out[19]: ([48.86140804265944,  
          48.86218743953672,  
          48.864708042972204,  
          48.867780000000004,  
          48.869020000000035],  
         [2.3331228017315646,  
          2.345242624547943,  
          2.336495988403402,  
          2.3301100000000474,  
          2.3344500111085558])
```

```
In [20]: df['Latitude'] = latitude  
         df['Longitude'] = longitude
```

In [21]: `df.head()`

Out[21]:

	PostalCode	Borough	Arrondissement's Area	Arrondissement's Perimeter	Neighborhood	Neighborhood's Area	I
0	75001	Louvre	1.824613e+06	6054.936862	Saint-Germain- l'Auxerrois	86.9	48
1	75001	Louvre	1.824613e+06	6054.936862	Halles	41.2	48
2	75001	Louvre	1.824613e+06	6054.936862	Palais-Royal	27.4	48
3	75001	Louvre	1.824613e+06	6054.936862	Place- Vendôme	26.9	48
4	75002	Bourse	9.911537e+05	4554.104360	Gaillon	18.8	48

In []:

In []:

Part 3: *Exploring and clustering the neighborhoods in Paris*

Examine the resulting dataframe.

```
In [22]: neighborhoods = df.copy()

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

neighborhoods.head()
```

The dataframe has 20 boroughs and 80 neighborhoods.

Out[22]:

	PostalCode	Borough	Arrondissement's Area	Arrondissement's Perimeter	Neighborhood	Neighborhood's Area	
0	75001	Louvre	1.824613e+06	6054.936862	Saint-Germain-l'Auxerrois	86.9	48
1	75001	Louvre	1.824613e+06	6054.936862	Halles	41.2	48
2	75001	Louvre	1.824613e+06	6054.936862	Palais-Royal	27.4	48
3	75001	Louvre	1.824613e+06	6054.936862	Place-Vendôme	26.9	48
4	75002	Bourse	9.911537e+05	4554.104360	Gaillon	18.8	48

Use geopy library to get the latitude and longitude values of Palais-Royal, Paris, FR. In order to define an instance of the geocoder, a user_agent is defined. The name of an agent is ny_explorer, as shown below.

```
In [23]: address = 'Palais-Royal, Paris, FR'

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

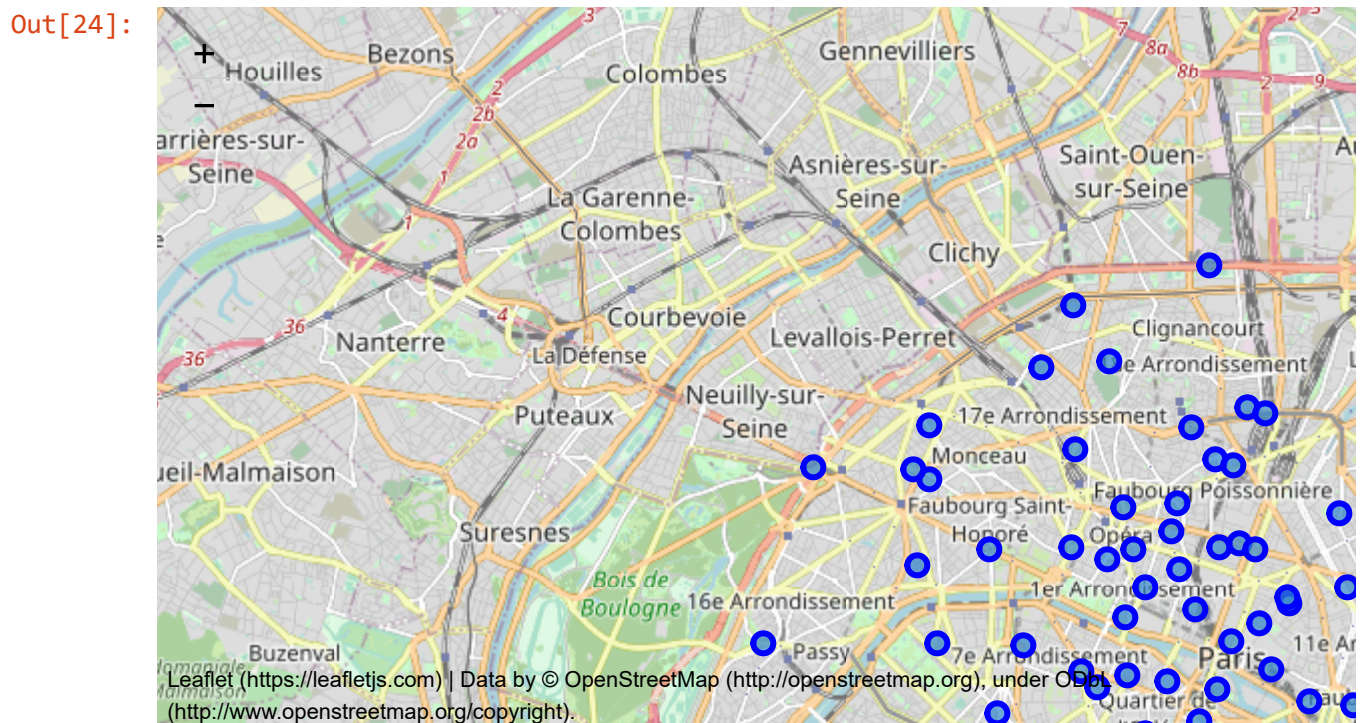
The geograpical coordinate of Palais-Royal, Paris are 48.863584700000004, 2.3362042200938715.

Create a map of Paris with neighborhoods superimposed on top.

```
In [24]: # create map of Paris using Latitude and Longitude values
map_paris = folium.Map(location=[latitude, longitude], zoom_start=12)

# add markers to map
for lat, lng, borough, neighborhood in zip(neighborhoods['Latitude'], neighbor
hoods['Longitude'], neighborhoods['Borough'], neighborhoods['Neighborhood']):
    label = '{} , {}'.format(neighborhood, borough)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(map_paris)

map_paris
```



In []:

Foursquare API.

Define Foursquare Credentials and Version

```
In [25]: CLIENT_ID = 'D5HCS5AUYRJUG2YL10JFLX05V3VJASRMXAXGSF1Y111GXKSP' # Foursquare ID
CLIENT_SECRET = 'CBS4MJPQFHVM41IL0E5LEJ4QRARFWFPPYBOVSYZ4F5GKQCMY' # Foursquare Secret
VERSION = '20200620' # Foursquare API version

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

Your credentails:

CLIENT_ID: D5HCS5AUYRJUG2YL10JFLX05V3VJASRMXAXGSF1Y111GXKSP

CLIENT_SECRET: CBS4MJPQFHVM41IL0E5LEJ4QRARFWFPPYBOVSYZ4F5GKQCMY

Let's explore the first neighborhood in our dataframe. Get the neighborhood's name.

```
In [26]: neighborhoods.loc[0, 'Neighborhood']
```

```
Out[26]: "Saint-Germain-l'Auxerrois"
```

Get the neighborhood's latitude and longitude values.

```
In [27]: neighborhood_latitude = neighborhoods.loc[0, 'Latitude'] # neighborhood latitude value
neighborhood_longitude = neighborhoods.loc[0, 'Longitude'] # neighborhood longitude value

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

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

Latitude and longitude values of Saint-Germain-l'Auxerrois are 48.86140804265944, 2.3331228017315646.

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

```
In [28]: LIMIT = 100 # limit of number of venues returned by Foursquare API

radius = 500 # define radius

url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
    CLIENT_ID,
    CLIENT_SECRET,
    VERSION,
    neighborhood_latitude,
    neighborhood_longitude,
    radius,
    LIMIT)
url # display URL
```

```
Out[28]: 'https://api.foursquare.com/v2/venues/explore?&client_id=D5HCS5AUJRJUG2YL10JFLX05V3VJASRMXAXGSF1Y111GXKSP&client_secret=CBS4MJPQFHVM41IL0E5LEJ4QRARFWFPPYB0VSYZ4F5GKQCMY&v=20200620&ll=48.86140804265944,2.3331228017315646&radius=500&limit=100'
```

Send the GET request and examine the results

```
In [29]: results = requests.get(url).json()
# results
```

Clean the json and structure it into a pandas dataframe.

```
In [30]: # function that extracts the category of the venue
def get_category_type(row):
    try:
        categories_list = row['categories']
    except:
        categories_list = row['venue.categories']

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

```
In [31]: venues = results['response']['groups'][0]['items']

nearby_venues = json_normalize(venues) # flatten JSON

# filter columns
filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat', 'venue.location.lng']
nearby_venues = nearby_venues.loc[:, filtered_columns]

# filter the category for each row
nearby_venues['venue.categories'] = nearby_venues.apply(get_category_type, axis=1)

# clean columns
nearby_venues.columns = [col.split(".")[1] for col in nearby_venues.columns]

nearby_venues.head()
```

Out[31]:

	name	categories	lat	lng
0	Vestige de la Forteresse du Louvre	Historic Site	48.861577	2.333508
1	Cour Napoléon	Plaza	48.861172	2.335088
2	Musée des Arts Décoratifs	Art Museum	48.863077	2.333393
3	Pavillon des Sessions – Arts d'Afrique, d'Asie...	Art Museum	48.860724	2.332121
4	LouLou	Italian Restaurant	48.862804	2.333500

```
In [32]: print('{} venues were returned by Foursquare.'.format(nearby_venues.shape[0]))

47 venues were returned by Foursquare.
```

In []:

Explore Neighborhoods in Paris.

```

In [33]: 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)

```

Run the above function on each neighborhood and create a new dataframe called paris_venues.


```
In [34]: paris_venues = getNearbyVenues(names=neighborhoods['Neighborhood'],  
                                         latitudes=neighborhoods['Latitude'],  
                                         longitudes=neighborhoods['Longitude']  
                                         )
```

Saint-Germain-l'Auxerrois
Halles
Palais-Royal
Place-Vendôme
Gaillon
Vivienne
Mail
Bonne-Nouvelle
Arts-et-Métiers
Enfants-Rouges
Archives
Sainte-Avoye
Saint-Merri
Saint-Gervais
Arsenal
Notre-Dame
Saint-Victor
Jardin-des-Plantes
Val-de-Grâce
Sorbonne
Monnaie
Odéon
Notre-Dame-des-Champs
Saint-Germain-des-Prés
Saint-Thomas-d'Aquin
Invalides
École-Militaire
Gros-Caillou
Champs-Élysées
Faubourg-du-Roule
Madeleine
Europe
Saint-Georges
Chaussée-d'Antin
Faubourg-Montmartre
Rochechouart
Saint-Vincent-de-Paul
Porte-Saint-Denis
Porte-Saint-Martin
Hôpital-Saint-Louis
Folie-Méricourt
Saint-Ambroise
Roquette
Sainte-Marguerite
Bel-Air
Picpus
Bercy
Quinze-Vingts
Salpêtrière
Gare
Maison-Blanche
Croulebarbe
Montparnasse
Parc-de-Montsouris
Petit-Montrouge
Plaisance
Saint-Lambert

Necker
 Grenelle
 Javel
 Auteuil
 Muette
 Porte-Dauphine
 Chaillot
 Ternes
 Plaine-de-Monceaux
 Batignolles
 Épinettes
 Grandes-Carrières
 Clignancourt
 Goutte-d'Or
 Chapelle
 Villette
 Pont-de-Flandre
 Amérique
 Combat
 Belleville
 Saint-Fargeau
 Père-Lachaise
 Charonne

Check the size of the resulting dataframe.

```
In [35]: print(paris_venues.shape)
paris_venues.head()
```

(5309, 7)

Out[35]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Saint-Germain-l'Auxerrois	48.861408	2.333123	Vestige de la Forteresse du Louvre	48.861577	2.333508	Historic Site
1	Saint-Germain-l'Auxerrois	48.861408	2.333123	Cour Napoléon	48.861172	2.335088	Plaza
2	Saint-Germain-l'Auxerrois	48.861408	2.333123	Musée des Arts Décoratifs	48.863077	2.333393	Art Museum
3	Saint-Germain-l'Auxerrois	48.861408	2.333123	Pavillon des Sessions – Arts d'Afrique, d'Asie...	48.860724	2.332121	Art Museum
4	Saint-Germain-l'Auxerrois	48.861408	2.333123	LouLou	48.862804	2.333500	Italian Restaurant

Check how many venues were returned for each neighborhood.

```
In [36]: paris_venues.groupby('Neighborhood').count()
```

Out[36]:

	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Neighborhood						
Amérique	37	37	37	37	37	37
Archives	94	94	94	94	94	94
Arsenal	71	71	71	71	71	71
Arts-et-Métiers	13	13	13	13	13	13
Auteuil	41	41	41	41	41	41
Batignolles	58	58	58	58	58	58
Bel-Air	30	30	30	30	30	30
Belleville	45	45	45	45	45	45
Bercy	47	47	47	47	47	47
Bonne-Nouvelle	83	83	83	83	83	83
Chaillot	100	100	100	100	100	100
Champs-Élysées	90	90	90	90	90	90
Chapelle	49	49	49	49	49	49
Charonne	56	56	56	56	56	56
Caussée-d'Antin	83	83	83	83	83	83
Clignancourt	18	18	18	18	18	18
Combat	31	31	31	31	31	31
Croulebarbe	52	52	52	52	52	52
Enfants-Rouges	78	78	78	78	78	78
Europe	35	35	35	35	35	35
Faubourg-Montmartre	87	87	87	87	87	87
Faubourg-du-Roule	100	100	100	100	100	100
Folie-Méricourt	100	100	100	100	100	100
Gaillon	100	100	100	100	100	100
Gare	50	50	50	50	50	50
Goutte-d'Or	28	28	28	28	28	28
Grandes-Carières	56	56	56	56	56	56
Grenelle	74	74	74	74	74	74
Gros-Caillou	100	100	100	100	100	100
Halles	61	61	61	61	61	61
Hôpital-Saint-Louis	76	76	76	76	76	76
Invalides	54	54	54	54	54	54

	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Neighborhood						
Jardin-des-Plantes	60	60	60	60	60	60
Javel	76	76	76	76	76	76
Madeleine	100	100	100	100	100	100
Mail	100	100	100	100	100	100
Maison-Blanche	72	72	72	72	72	72
Monnaie	91	91	91	91	91	91
Montparnasse	82	82	82	82	82	82
Muette	37	37	37	37	37	37
Necker	45	45	45	45	45	45
Notre-Dame	65	65	65	65	65	65
Notre-Dame-des-Champs	47	47	47	47	47	47
Odéon	63	63	63	63	63	63
Palais-Royal	100	100	100	100	100	100
Parc-de-Montsouris	18	18	18	18	18	18
Petit-Montrouge	63	63	63	63	63	63
Picpus	35	35	35	35	35	35
Place-Vendôme	100	100	100	100	100	100
Plaine-de-Monceaux	51	51	51	51	51	51
Plaisance	59	59	59	59	59	59
Pont-de-Flandre	44	44	44	44	44	44
Porte-Dauphine	70	70	70	70	70	70
Porte-Saint-Denis	69	69	69	69	69	69
Porte-Saint-Martin	100	100	100	100	100	100
Père-Lachaise	18	18	18	18	18	18
Quinze-Vingts	71	71	71	71	71	71
Rochechouart	77	77	77	77	77	77
Roquette	100	100	100	100	100	100
Saint-Ambroise	66	66	66	66	66	66
Saint-Fargeau	18	18	18	18	18	18
Saint-Georges	100	100	100	100	100	100
Saint-Germain-des-Prés	100	100	100	100	100	100

	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Neighborhood						
Saint-Germain-l'Auxerrois	47	47	47	47	47	47
Saint-Gervais	100	100	100	100	100	100
Saint-Lambert	53	53	53	53	53	53
Saint-Merri	98	98	98	98	98	98
Saint-Thomas-d'Aquin	82	82	82	82	82	82
Saint-Victor	94	94	94	94	94	94
Saint-Vincent-de-Paul	83	83	83	83	83	83
Sainte-Avoye	69	69	69	69	69	69
Sainte-Marguerite	91	91	91	91	91	91
Salpêtrière	79	79	79	79	79	79
Sorbonne	100	100	100	100	100	100
Ternes	100	100	100	100	100	100
Val-de-Grâce	48	48	48	48	48	48
Villeite	45	45	45	45	45	45
Vivienne	59	59	59	59	59	59
École-Militaire	31	31	31	31	31	31
Épinettes	36	36	36	36	36	36

Find out how many unique categories can be curated from all the returned venues.

```
In [37]: print('There are {} uniques categories.'.format(len(paris_venues['Venue Category'].unique())))
```

There are 288 uniques categories.

```
In [ ]:
```

Analyze Each Neighborhood.

```
In [55]: # one hot encoding
paris_onehot = pd.get_dummies(paris_venues[['Venue Category']], prefix="", prefix_sep="")

# add neighborhood column back to dataframe
paris_onehot['Neighborhood'] = paris_venues['Neighborhood']

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

# paris_onehot.head()
```

Examine the new dataframe size.

```
In [39]: paris_onehot.shape
```

```
Out[39]: (5309, 289)
```

Group rows by neighborhood and by taking the mean of the frequency of occurrence of each category.


```
In [40]: paris_grouped = paris_onehot.groupby('Neighborhood').mean().reset_index()  
paris_grouped
```

Out[40]:

	Neighborhood	Accessories Store	Afghan Restaurant	African Restaurant	Alsatian Restaurant	American Restaurant	Antique Shop	Arger Restaurant
0	Amérique	0.000000	0.000000	0.027027	0.000000	0.000000	0.000000	0.0
1	Archives	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
2	Arsenal	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
3	Arts-et-Métiers	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
4	Auteuil	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
5	Batignolles	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
6	Bel-Air	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
7	Belleville	0.000000	0.000000	0.044444	0.000000	0.000000	0.000000	0.0
8	Bercy	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
9	Bonne-Nouvelle	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
10	Chaillot	0.020000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
11	Champs-Élysées	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
12	Chapelle	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
13	Charonne	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
14	Chaussée-d'Antin	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
15	Clignancourt	0.000000	0.000000	0.055556	0.000000	0.000000	0.000000	0.0
16	Combat	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
17	Croulebarbe	0.000000	0.000000	0.000000	0.000000	0.019231	0.000000	0.0
18	Enfants-Rouges	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
19	Europe	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
20	Faubourg-Montmartre	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
21	Faubourg-du-Roule	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
22	Folie-Méricourt	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
23	Gaillon	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
24	Gare	0.000000	0.000000	0.020000	0.000000	0.000000	0.000000	0.0
25	Goutte-d'Or	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
26	Grandes-Carrières	0.000000	0.000000	0.000000	0.000000	0.017857	0.017857	0.0
27	Grenelle	0.000000	0.000000	0.000000	0.000000	0.027027	0.000000	0.0
28	Gros-Caillou	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
29	Halles	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0

	Neighborhood	Accessories Store	Afghan Restaurant	African Restaurant	Alsatian Restaurant	American Restaurant	Antique Shop	Arger Rest:
30	Hôpital-Saint-Louis	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
31	Invalides	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
32	Jardin-des-Plantes	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
33	Javel	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
34	Madeleine	0.010000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
35	Mail	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
36	Maison-Blanche	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
37	Monnaie	0.000000	0.000000	0.010989	0.000000	0.000000	0.000000	0.0
38	Montparnasse	0.000000	0.000000	0.000000	0.012195	0.000000	0.000000	0.0
39	Muette	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
40	Necker	0.000000	0.000000	0.000000	0.022222	0.000000	0.000000	0.0
41	Notre-Dame	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
42	Notre-Dame-des-Champs	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
43	Odéon	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
44	Palais-Royal	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
45	Parc-de-Montsouris	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
46	Petit-Montrouge	0.000000	0.000000	0.015873	0.000000	0.000000	0.000000	0.0
47	Picpus	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
48	Place-Vendôme	0.010000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
49	Plaine-de-Monceaux	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
50	Plaisance	0.000000	0.000000	0.000000	0.000000	0.016949	0.000000	0.0
51	Pont-de-Flandre	0.000000	0.000000	0.022727	0.000000	0.022727	0.000000	0.0
52	Porte-Dauphine	0.000000	0.000000	0.000000	0.000000	0.028571	0.000000	0.0
53	Porte-Saint-Denis	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
54	Porte-Saint-Martin	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
55	Père-Lachaise	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
56	Quinze-Vingts	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
57	Rochechouart	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0

	Neighborhood	Accessories Store	Afghan Restaurant	African Restaurant	Alsatian Restaurant	American Restaurant	Antique Shop	Arger Rest:
58	Roquette	0.010000	0.000000	0.010000	0.000000	0.000000	0.000000	0.0
59	Saint-Ambroise	0.000000	0.015152	0.000000	0.000000	0.000000	0.000000	0.0
60	Saint-Fargeau	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
61	Saint-Georges	0.000000	0.000000	0.010000	0.000000	0.000000	0.000000	0.0
62	Saint-Germain-des-Prés	0.000000	0.000000	0.000000	0.000000	0.010000	0.000000	0.0
63	Saint-Germain-l'Auxerrois	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
64	Saint-Gervais	0.000000	0.000000	0.010000	0.000000	0.000000	0.000000	0.0
65	Saint-Lambert	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
66	Saint-Merri	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
67	Saint-Thomas-d'Aquin	0.000000	0.000000	0.000000	0.000000	0.024390	0.000000	0.0
68	Saint-Victor	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
69	Saint-Vincent-de-Paul	0.000000	0.000000	0.012048	0.000000	0.000000	0.000000	0.0
70	Sainte-Avoye	0.000000	0.000000	0.014493	0.000000	0.000000	0.000000	0.0
71	Sainte-Marguerite	0.010989	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
72	Salpêtrière	0.000000	0.000000	0.000000	0.000000	0.012658	0.000000	0.0
73	Sorbonne	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
74	Ternes	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
75	Val-de-Grâce	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
76	Villette	0.000000	0.000000	0.022222	0.000000	0.000000	0.000000	0.0
77	Vivienne	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
78	École-Militaire	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
79	Épinettes	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0

Confirm the new size.

In [41]: paris_grouped.shape

Out[41]: (80, 289)

Print each neighborhood along with the top 5 most common venues.

```
In [42]: num_top_venues = 5

for hood in paris_grouped['Neighborhood']:
    print("-----"+hood+"-----")
    temp = paris_grouped[paris_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')
```

----Amérique----

	venue	freq
0	Bar	0.14
1	Bakery	0.08
2	French Restaurant	0.08
3	Restaurant	0.05
4	Japanese Restaurant	0.05

----Archives----

	venue	freq
0	Japanese Restaurant	0.06
1	French Restaurant	0.06
2	Italian Restaurant	0.04
3	Wine Bar	0.04
4	Cocktail Bar	0.04

----Arsenal----

	venue	freq
0	French Restaurant	0.21
1	Hotel	0.08
2	Plaza	0.04
3	Park	0.04
4	Italian Restaurant	0.03

----Arts-et-Métiers----

	venue	freq
0	Italian Restaurant	0.08
1	Hotel	0.08
2	Park	0.08
3	Brasserie	0.08
4	Middle Eastern Restaurant	0.08

----Auteuil----

	venue	freq
0	Tennis Court	0.24
1	French Restaurant	0.15
2	Sporting Goods Shop	0.05
3	Stadium	0.05
4	Brewery	0.05

----Batignolles----

	venue	freq
0	French Restaurant	0.24
1	Bar	0.07
2	Italian Restaurant	0.05
3	Restaurant	0.05
4	Pizza Place	0.05

----Bel-Air----

	venue	freq
0	Hotel	0.17

1	Japanese Restaurant	0.10
2	Farmers Market	0.07
3	Convenience Store	0.07
4	Supermarket	0.07

----Belleville----

	venue	freq
0	Bar	0.18
1	French Restaurant	0.07
2	Italian Restaurant	0.04
3	African Restaurant	0.04
4	Pizza Place	0.04

----Bercy----

	venue	freq
0	Hotel	0.06
1	Italian Restaurant	0.04
2	Bus Stop	0.04
3	Bakery	0.04
4	French Restaurant	0.04

----Bonne-Nouvelle----

	venue	freq
0	Hotel	0.10
1	Cocktail Bar	0.08
2	Wine Bar	0.08
3	Bar	0.06
4	Bakery	0.06

----Chaillot----

	venue	freq
0	French Restaurant	0.20
1	Hotel	0.16
2	Italian Restaurant	0.05
3	Art Museum	0.05
4	Lebanese Restaurant	0.03

----Champs-Élysées----

	venue	freq
0	French Restaurant	0.16
1	Hotel	0.12
2	Garden	0.06
3	Art Gallery	0.03
4	Plaza	0.03

----Chapelle----

	venue	freq
0	Indian Restaurant	0.16
1	Hotel	0.08
2	French Restaurant	0.08
3	Supermarket	0.06

4 Bakery 0.06

----Charonne----

	venue	freq
0	French Restaurant	0.20
1	Bar	0.11
2	Italian Restaurant	0.05
3	Farmers Market	0.05
4	Supermarket	0.04

----Chaussée-d'Antin----

	venue	freq
0	French Restaurant	0.12
1	Hotel	0.10
2	Salad Place	0.06
3	Bistro	0.05
4	Tea Room	0.04

----Clignancourt----

	venue	freq
0	Flea Market	0.28
1	Fast Food Restaurant	0.11
2	Beer Garden	0.06
3	Hotel	0.06
4	Fried Chicken Joint	0.06

----Combat----

	venue	freq
0	French Restaurant	0.26
1	Bakery	0.06
2	Supermarket	0.06
3	Bar	0.06
4	Burger Joint	0.03

----Croulebarbe----

	venue	freq
0	French Restaurant	0.17
1	Hotel	0.08
2	Italian Restaurant	0.06
3	Japanese Restaurant	0.06
4	Sushi Restaurant	0.06

----Enfants-Rouges----

	venue	freq
0	Italian Restaurant	0.06
1	Japanese Restaurant	0.06
2	Vietnamese Restaurant	0.04
3	Cocktail Bar	0.04
4	Coffee Shop	0.04

----Europe----

	venue	freq
0	Hotel	0.14
1	Wine Bar	0.11
2	French Restaurant	0.09
3	Sushi Restaurant	0.09
4	Italian Restaurant	0.06

----Faubourg-Montmartre----

	venue	freq
0	French Restaurant	0.18
1	Hotel	0.09
2	Italian Restaurant	0.06
3	Japanese Restaurant	0.05
4	Chinese Restaurant	0.03

----Faubourg-du-Roule----

	venue	freq
0	Hotel	0.19
1	French Restaurant	0.19
2	Italian Restaurant	0.09
3	Bakery	0.03
4	Bistro	0.03

----Folie-Méricourt----

	venue	freq
0	French Restaurant	0.11
1	Bar	0.08
2	Bakery	0.05
3	Restaurant	0.05
4	Japanese Restaurant	0.05

----Gaillon----

	venue	freq
0	Hotel	0.10
1	French Restaurant	0.09
2	Japanese Restaurant	0.07
3	Jewelry Store	0.05
4	Chocolate Shop	0.04

----Gare----

	venue	freq
0	Nightclub	0.06
1	French Restaurant	0.06
2	Beer Garden	0.06
3	Bakery	0.04
4	Food Truck	0.04

----Goutte-d'Or----

	venue	freq
0	Indian Restaurant	0.14

1	Bookstore	0.07
2	Supermarket	0.07
3	Bar	0.07
4	Bakery	0.07

----Grandes-Carières----

	venue	freq
0	French Restaurant	0.29
1	Bar	0.09
2	Hotel	0.07
3	Italian Restaurant	0.05
4	Bakery	0.05

----Grenelle----

	venue	freq
0	French Restaurant	0.15
1	Hotel	0.12
2	Café	0.07
3	Plaza	0.04
4	American Restaurant	0.03

----Gros-Caillou----

	venue	freq
0	French Restaurant	0.27
1	Italian Restaurant	0.11
2	Hotel	0.09
3	Café	0.05
4	Ice Cream Shop	0.03

----Halles----

	venue	freq
0	French Restaurant	0.20
1	Ice Cream Shop	0.05
2	Bakery	0.05
3	Sandwich Place	0.03
4	Pedestrian Plaza	0.03

----Hôpital-Saint-Louis----

	venue	freq
0	French Restaurant	0.09
1	Italian Restaurant	0.05
2	Coffee Shop	0.05
3	Bar	0.05
4	Bistro	0.04

----Invalides----

	venue	freq
0	French Restaurant	0.35
1	Hotel	0.11
2	Plaza	0.07
3	History Museum	0.06

4 Café 0.06

----Jardin-des-Plantes----

	venue	freq
0	French Restaurant	0.13
1	Science Museum	0.10
2	Hotel	0.07
3	Garden	0.07
4	Zoo Exhibit	0.05

----Javel----

	venue	freq
0	French Restaurant	0.11
1	Hotel	0.08
2	Bakery	0.05
3	Korean Restaurant	0.05
4	Persian Restaurant	0.05

----Madeleine----

	venue	freq
0	Hotel	0.15
1	French Restaurant	0.11
2	Boutique	0.10
3	Women's Store	0.04
4	Men's Store	0.03

----Mail----

	venue	freq
0	French Restaurant	0.14
1	Italian Restaurant	0.06
2	Wine Bar	0.06
3	Bistro	0.04
4	Bakery	0.03

----Maison-Blanche----

	venue	freq
0	Vietnamese Restaurant	0.22
1	Asian Restaurant	0.17
2	Chinese Restaurant	0.12
3	Thai Restaurant	0.11
4	Cantonese Restaurant	0.04

----Monnaie----

	venue	freq
0	French Restaurant	0.13
1	Hotel	0.03
2	Historic Site	0.03
3	Bookstore	0.03
4	Bistro	0.03

----Montparnasse----

	venue	freq
0	French Restaurant	0.16
1	Hotel	0.09
2	Japanese Restaurant	0.07
3	Creperie	0.07
4	Italian Restaurant	0.04

----Muette----

	venue	freq
0	French Restaurant	0.16
1	Bakery	0.11
2	Italian Restaurant	0.08
3	Garden	0.05
4	Supermarket	0.05

----Necker----

	venue	freq
0	French Restaurant	0.16
1	Hotel	0.13
2	Italian Restaurant	0.07
3	Bar	0.04
4	Café	0.04

----Notre-Dame----

	venue	freq
0	French Restaurant	0.15
1	Plaza	0.08
2	Bookstore	0.06
3	Bar	0.05
4	Bakery	0.05

----Notre-Dame-des-Champs----

	venue	freq
0	French Restaurant	0.19
1	Creperie	0.09
2	Café	0.04
3	Hotel	0.04
4	Italian Restaurant	0.02

----Odéon----

	venue	freq
0	Café	0.06
1	Hotel	0.06
2	Bistro	0.05
3	Plaza	0.03
4	Bakery	0.03

----Palais-Royal----

	venue	freq
0	Japanese Restaurant	0.12

1	French Restaurant	0.10
2	Hotel	0.06
3	Italian Restaurant	0.04
4	Plaza	0.04

----Parc-de-Montsouris----

	venue	freq
0	Bus Stop	0.17
1	Japanese Restaurant	0.11
2	Italian Restaurant	0.11
3	Gym	0.06
4	Bistro	0.06

----Petit-Montrouge----

	venue	freq
0	Hotel	0.17
1	French Restaurant	0.14
2	Supermarket	0.06
3	Italian Restaurant	0.05
4	Café	0.05

----Picpus----

	venue	freq
0	Hotel	0.11
1	Pizza Place	0.09
2	Farmers Market	0.06
3	Supermarket	0.06
4	Japanese Restaurant	0.06

----Place-Vendôme----

	venue	freq
0	Hotel	0.12
1	Japanese Restaurant	0.10
2	French Restaurant	0.09
3	Bookstore	0.05
4	Plaza	0.04

----Plaine-de-Monceaux----

	venue	freq
0	French Restaurant	0.31
1	Hotel	0.10
2	Bakery	0.08
3	Bistro	0.04
4	Bar	0.04

----Plaisance----

	venue	freq
0	French Restaurant	0.10
1	Supermarket	0.07
2	Bistro	0.05
3	Bar	0.05

4 Bakery 0.05

----Pont-de-Flandre----

	venue	freq
0	Café	0.07
1	French Restaurant	0.07
2	Hotel	0.07
3	Asian Restaurant	0.05
4	Tram Station	0.05

----Porte-Dauphine----

	venue	freq
0	French Restaurant	0.20
1	Hotel	0.13
2	Italian Restaurant	0.07
3	Gym / Fitness Center	0.04
4	Chinese Restaurant	0.04

----Porte-Saint-Denis----

	venue	freq
0	French Restaurant	0.09
1	Cocktail Bar	0.09
2	Bar	0.09
3	Hotel	0.06
4	Burger Joint	0.06

----Porte-Saint-Martin----

	venue	freq
0	French Restaurant	0.11
1	Bar	0.07
2	Hotel	0.06
3	Italian Restaurant	0.05
4	Coffee Shop	0.05

----Père-Lachaise----

	venue	freq
0	Bar	0.11
1	French Restaurant	0.11
2	Beer Bar	0.11
3	Italian Restaurant	0.06
4	Cemetery	0.06

----Quinze-Vingts----

	venue	freq
0	French Restaurant	0.10
1	Cocktail Bar	0.08
2	Bar	0.07
3	Italian Restaurant	0.06
4	Bookstore	0.06

----Rochechouart----

	venue	freq
0	French Restaurant	0.18
1	Bar	0.06
2	Hotel	0.06
3	Bakery	0.05
4	Coffee Shop	0.05

----Roquette----

	venue	freq
0	French Restaurant	0.11
1	Bar	0.06
2	Italian Restaurant	0.05
3	Café	0.05
4	Bistro	0.05

----Saint-Ambroise----

	venue	freq
0	French Restaurant	0.11
1	Restaurant	0.09
2	Bar	0.08
3	Bakery	0.05
4	Café	0.05

----Saint-Fargeau----

	venue	freq
0	Bakery	0.11
1	French Restaurant	0.11
2	Supermarket	0.11
3	Plaza	0.06
4	Indian Restaurant	0.06

----Saint-Georges----

	venue	freq
0	French Restaurant	0.21
1	Italian Restaurant	0.06
2	Hotel	0.06
3	Japanese Restaurant	0.05
4	Restaurant	0.05

----Saint-Germain-des-Prés----

	venue	freq
0	French Restaurant	0.13
1	Hotel	0.10
2	Italian Restaurant	0.08
3	Café	0.04
4	Plaza	0.04

----Saint-Germain-l'Auxerrois----

	venue	freq
0	Japanese Restaurant	0.11

1	French Restaurant	0.09
2	Coffee Shop	0.09
3	Art Museum	0.06
4	Historic Site	0.06

----Saint-Gervais----

	venue	freq
0	French Restaurant	0.11
1	Clothing Store	0.06
2	Ice Cream Shop	0.05
3	Hotel	0.04
4	Wine Bar	0.03

----Saint-Lambert----

	venue	freq
0	French Restaurant	0.17
1	Italian Restaurant	0.08
2	Hotel	0.06
3	Bistro	0.06
4	Bakery	0.06

----Saint-Merri----

	venue	freq
0	French Restaurant	0.14
1	Ice Cream Shop	0.06
2	Art Gallery	0.05
3	Bakery	0.04
4	Plaza	0.03

----Saint-Thomas-d'Aquin----

	venue	freq
0	French Restaurant	0.22
1	Hotel	0.12
2	Café	0.05
3	Italian Restaurant	0.04
4	Art Gallery	0.04

----Saint-Victor----

	venue	freq
0	French Restaurant	0.16
1	Bar	0.09
2	Italian Restaurant	0.07
3	Pub	0.04
4	Hotel	0.04

----Saint-Vincent-de-Paul----

	venue	freq
0	French Restaurant	0.19
1	Hotel	0.10
2	Japanese Restaurant	0.05
3	Italian Restaurant	0.04

4 Seafood Restaurant 0.04

----Sainte-Avoye----

	venue	freq
0	Burger Joint	0.06
1	Ice Cream Shop	0.06
2	Deli / Bodega	0.04
3	Bakery	0.04
4	Art Gallery	0.04

----Sainte-Marguerite----

	venue	freq
0	French Restaurant	0.13
1	Bar	0.10
2	Bistro	0.04
3	Wine Bar	0.04
4	Cocktail Bar	0.04

----Salpêtrière----

	venue	freq
0	Hotel	0.15
1	French Restaurant	0.09
2	Italian Restaurant	0.08
3	Bar	0.06
4	Supermarket	0.06

----Sorbonne----

	venue	freq
0	French Restaurant	0.10
1	Hotel	0.07
2	Plaza	0.06
3	Bookstore	0.05
4	Indie Movie Theater	0.05

----Ternes----

	venue	freq
0	Hotel	0.24
1	French Restaurant	0.18
2	Italian Restaurant	0.11
3	Seafood Restaurant	0.03
4	Bar	0.02

----Val-de-Grâce----

	venue	freq
0	French Restaurant	0.12
1	Bar	0.10
2	Hotel	0.08
3	Creperie	0.04
4	Asian Restaurant	0.04

----Villette----

	venue	freq
0	French Restaurant	0.20
1	Bar	0.09
2	Café	0.07
3	Restaurant	0.07
4	Italian Restaurant	0.04

----Vivienne----

	venue	freq
0	Hotel	0.08
1	French Restaurant	0.07
2	Italian Restaurant	0.05
3	Japanese Restaurant	0.05
4	Pizza Place	0.05

----École-Militaire----

	venue	freq
0	Hotel	0.29
1	French Restaurant	0.16
2	Plaza	0.06
3	Café	0.06
4	Italian Restaurant	0.03

----Épinettes----

	venue	freq
0	Supermarket	0.14
1	Japanese Restaurant	0.08
2	French Restaurant	0.08
3	Bus Stop	0.06
4	Bakery	0.06

Put that into a pandas dataframe:

A function to sort the venues in descending order.

```
In [43]: 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]
```

Create the new dataframe and display the top 10 venues for each neighborhood.

```

In [44]: 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'] = paris_grouped['Neighborhood']

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

neighborhoods_venues_sorted.head()

```

Out[44]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue
0	Amérique	Bar	Bakery	French Restaurant	Café	Restaurant	Japanese Restaurant	Gourmet Shop
1	Archives	French Restaurant	Japanese Restaurant	Cocktail Bar	Art Gallery	Italian Restaurant	Wine Bar	Bakery
2	Arsenal	French Restaurant	Hotel	Plaza	Park	Gastropub	Italian Restaurant	Tapas Restaurant
3	Arts-et-Métiers	Diner	Kebab Restaurant	Theater	Middle Eastern Restaurant	Brasserie	Pizza Place	BBQ Joint
4	Auteuil	Tennis Court	French Restaurant	Plaza	Sporting Goods Shop	Brewery	Stadium	Pool

Look at establishments' frequencies in dataset.

```

In [45]: establishment_occurencies = []
for venues in neighborhoods_venues_sorted.iloc[:, 2:].values:
    for establishments in venues:
        establishment_occurencies.append(establishments)

caffe = ['Café', 'Burger Joint', 'Sandwich Place', 'Bistro', 'Gastropub', 'Diner', 'Pizza Place', 'BBQ Joint', 'Salad Place', 'Tea Room', 'Steakhouse', 'Food Truck', 'Buffet', 'Fried Chicken Joint', 'Creperie', 'Poke Place']
bar = ['Bar', 'Pub', 'Brasserie', 'Brewery', 'Beer Garden']
shop = ['Store', 'Beer Store', 'Flea Market', 'Health Food Store', 'Furniture / Home Store', 'Bookstore', 'Accessories Store', 'Department Store', 'Electronics Store', 'Shoe Store', 'Women\'s Store', 'Men\'s Store', 'Clothing Store', 'Grocery Store', 'Jewelry Store', 'Candy Store', 'Convenience Store', 'Boutique', 'Shop', 'Farmers Market', 'Supermarket', 'Organic Grocery', 'Grocery', 'Market', 'Bakery', 'Deli / Bodega']
plaza = ['Plaza', 'Pedestrian Plaza']
culture = ['Theater', 'Indie Movie Theater', 'Art Gallery', 'Street Art', 'Art Museum', 'History Museum', 'Science Museum', 'Concert Hall', 'Multiplex', 'Zoo Exhibit', 'Music Venue', 'Outdoor Sculpture']
rest_and_sport = ['Park', 'Boat or Ferry', 'Historic Site', 'Tennis Court', 'Stadium', 'Pool', 'Botanical Garden', 'Spa', 'Gym', 'Soccer Field', 'Lounge', 'Fountain', 'Forest', 'Lounge', 'Trail', 'Roof Deck', 'Gym Pool', 'Gym / Fitness Center']
transport = ['Tram Station', 'Bus Stop']

establishments_list = []

for i in establishment_occurencies:
    if 'restaurant' in i.lower():
        establishments_list.append('Restaurant')
    elif 'hotel' in i.lower():
        establishments_list.append('Hotel')
    elif str(caffe).lower().find(i.lower()) != -1:
        establishments_list.append('Caffe')
    elif (str(bar).lower().find(i.lower()) != -1) or ('Bar' in i):
        establishments_list.append('Bar')
    elif (str(shop).lower().find(i.lower()) != -1) or ('Shop' in i):
        establishments_list.append('Shop')
    elif str(plaza).lower().find(i.lower()) != -1:
        establishments_list.append('Plaza')
    elif str(culture).lower().find(i.lower()) != -1:
        establishments_list.append('Culture')
    elif str(rest_and_sport).lower().find(i.lower()) != -1:
        establishments_list.append('Rest and Sport')
    elif str(transport).lower().find(i.lower()) != -1:
        establishments_list.append('Transport')
    else:
        establishments_list.append('Other')

df_hist = pd.DataFrame({'establishments': establishments_list})
df_hist.reset_index(inplace=True)
df_hist.columns = ['count', 'establishments']
print('Total number of establishments:', df_hist.establishments.value_counts().sum())

```

```
print('\nPercent of establishments by each group:\n')
print(df_hist.establishments.value_counts() / df_hist.establishments.value_counts().sum() * 100)

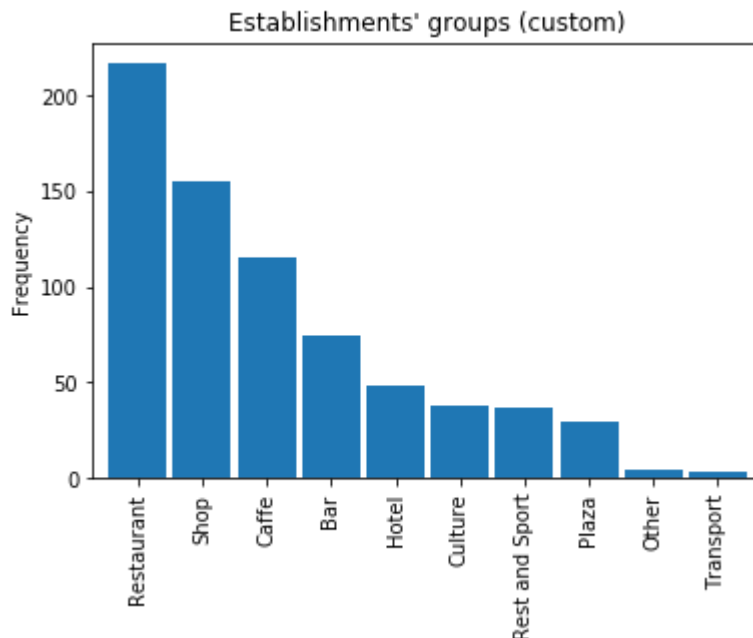
ax = df_hist['establishments'].value_counts().plot(kind='bar', width=.9)
plt.title('Establishments\' groups (custom)')
plt.ylabel('Frequency')
plt.show()
```

Total number of establishments: 720

Percent of establishments by each group:

Restaurant	30.138889
Shop	21.527778
Caffe	15.972222
Bar	10.277778
Hotel	6.666667
Culture	5.277778
Rest and Sport	5.138889
Plaza	4.027778
Other	0.555556
Transport	0.416667

Name: establishments, dtype: float64



Cluster Neighborhoods

Run k-means to cluster the neighborhood into **4 clusters**.

```
In [46]: # set number of clusters
kclusters = 4

paris_grouped_clustering = paris_grouped.drop('Neighborhood', 1)

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

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

Out[46]: array([1, 1, 2, 1, 1, 2, 0, 1, 1, 1])

Create a new dataframe that includes the cluster as well as the top 10 venues for each neighborhood.

```
In [47]: # add clustering labels
neighborhoods_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)

paris_merged = neighborhoods

# merge paris_grouped with paris_data to add Latitude/Longitude for each neighborhood
paris_merged = paris_merged.join(neighborhoods_venues_sorted.set_index('Neighborhood'), on='Neighborhood')
paris_merged.dropna(inplace=True)

paris_merged.head()
```

Out[47]:

	PostalCode	Borough	Arrondissement's Area	Arrondissement's Perimeter	Neighborhood	Neighborhood's Area	
0	75001	Louvre	1.824613e+06	6054.936862	Saint-Germain-l'Auxerrois	86.9	48
1	75001	Louvre	1.824613e+06	6054.936862	Halles	41.2	48
2	75001	Louvre	1.824613e+06	6054.936862	Palais-Royal	27.4	48
3	75001	Louvre	1.824613e+06	6054.936862	Place-Vendôme	26.9	48
4	75002	Bourse	9.911537e+05	4554.104360	Gaillon	18.8	48

```
In [48]: paris_merged.tail() # check the last columns
```

Out[48]:

	PostalCode	Borough	Arrondissement's Area	Arrondissement's Perimeter	Neighborhood	Neighborhood's Area
75	75019	Buttes-Chaumont	6.792651e+06	11253.182479	Combat	129.1
76	75020	Ménilmontant	5.983446e+06	10704.940486	Belleville	80.1
77	75020	Ménilmontant	5.983446e+06	10704.940486	Saint-Fargeau	148.1
78	75020	Ménilmontant	5.983446e+06	10704.940486	Père-Lachaise	159.1
79	75020	Ménilmontant	5.983446e+06	10704.940486	Charonne	209.1

Visualize the resulting clusters.

```

In [49]: # create map
map_clusters = folium.Map(location=[latitude, longitude], zoom_start=12)

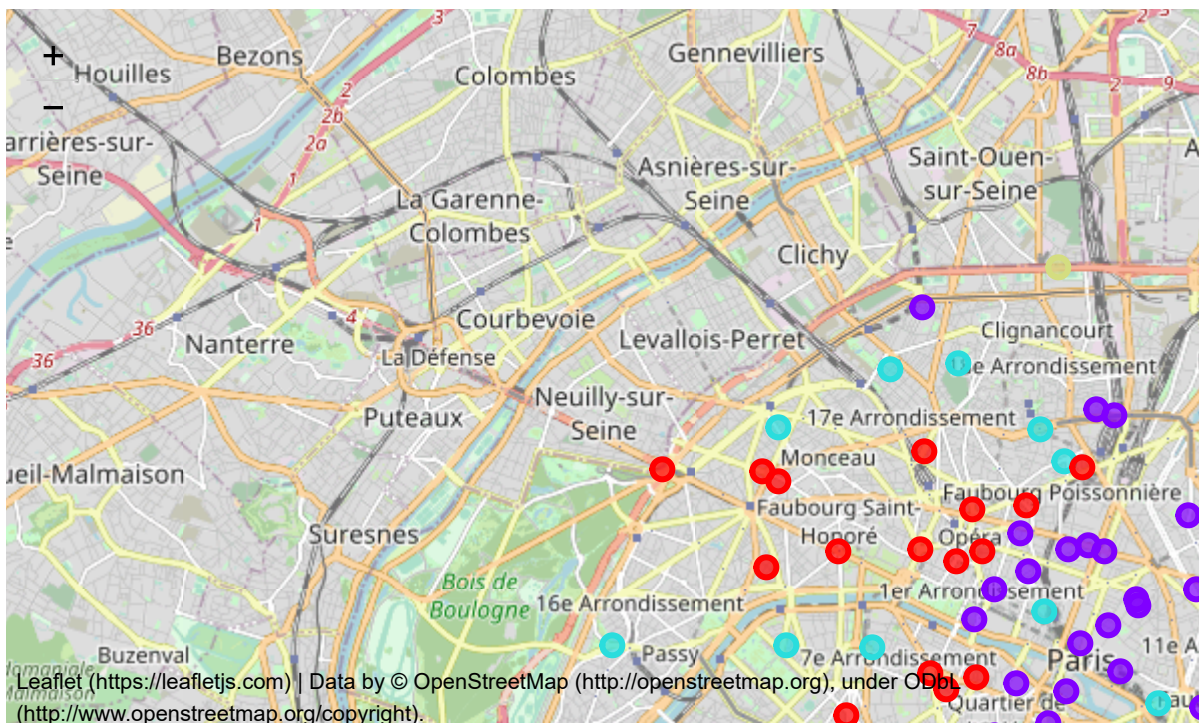
# set color scheme for the clusters
x = np.arange(kclusters)
ys = [i + x + (i*x)**2 for i in range(kclusters)]
colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
rainbow = [colors.rgb2hex(i) for i in colors_array]

# add markers to the map
markers_colors = []
for lat, lon, poi, cluster in zip(paris_merged['Latitude'], paris_merged['Longitude'], paris_merged['Neighborhood'], paris_merged['Cluster Labels']):
    cluster = int(cluster)
    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[49]:



In []:

Examine Clusters.

Two clusters visible more clearly on the map than other clusters (#2, and #13)

Cluster 1

```
In [50]: paris_merged.loc[paris_merged['Cluster Labels'] == 0, paris_merged.columns[[1]  
+ list(range(5, paris_merged.shape[1]))]]
```

Out[50]:

	Borough	Neighborhood's Area	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue
3	Louvre	26.9	48.867780	2.330110	0	Hotel	Japanese Restaurant	French Restaurant
4	Bourse	18.8	48.869020	2.334450	0	Hotel	French Restaurant	Japanese Restaurant
17	Panthéon	79.8	48.842448	2.357202	0	French Restaurant	Science Museum	Hotel
23	Luxembourg	28.2	48.854829	2.333614	0	French Restaurant	Hotel	Italian Restaurant
24	Palais-Bourbon	82.7	48.855315	2.325613	0	French Restaurant	Hotel	Cafe
26	Palais-Bourbon	80.8	48.850436	2.311343	0	Hotel	French Restaurant	Place
28	Élysée	114.1	48.869060	2.309930	0	French Restaurant	Hotel	Garcia
29	Élysée	79.6	48.876818	2.299611	0	French Restaurant	Hotel	Italian Restaurant
30	Élysée	76.1	48.869300	2.323800	0	Hotel	French Restaurant	Boutique
31	Élysée	118.3	48.880341	2.324489	0	Hotel	Wine Bar	Sushi Restaurant
33	Opéra	54.3	48.873750	2.332980	0	French Restaurant	Hotel	Salad Place
34	Opéra	41.7	48.874261	2.342063	0	French Restaurant	Hotel	Italian Restaurant
36	Entrepôt	92.7	48.878490	2.351760	0	French Restaurant	Hotel	Japanese Restaurant
44	Reuilly	138.6	48.845090	2.406380	0	Hotel	Japanese Restaurant	French Restaurant
48	Gobelins	118.2	48.835300	2.358300	0	Hotel	French Restaurant	Italian Restaurant
51	Gobelins	69.2	48.833939	2.347926	0	French Restaurant	Hotel	Italian Restaurant
52	Observatoire	112.6	48.842800	2.323620	0	French Restaurant	Hotel	Crepe
54	Observatoire	134.6	48.826420	2.325200	0	Hotel	French Restaurant	Supermarket
57	Vaugirard	157.8	48.843290	2.313700	0	French Restaurant	Hotel	Italian Restaurant
58	Vaugirard	147.8	48.853310	2.328420	0	French Restaurant	Hotel	Cafe
62	Passy	141.4	48.878293	2.279673	0	French Restaurant	Hotel	Italian Restaurant

	Borough	Neighborhood's Area	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue
63	Passy	142.4	48.867196	2.297359	0	French Restaurant	Hotel	Art Museum
64	Batignolles-Monceau	146.6	48.878100	2.296790	0	Hotel	French Restaurant	Italian Restaurant

Cluster 2

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

Out[51]:

	Borough	Neighborhood's Area	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3 C
0	Louvre	86.9	48.861408	2.333123	1	Japanese Restaurant	Coffee Shop	Re:
2	Louvre	27.4	48.864708	2.336496	1	Japanese Restaurant	French Restaurant	
5	Bourse	24.4	48.871100	2.341280	1	Hotel	French Restaurant	Pizz
6	Bourse	27.8	48.866816	2.342632	1	French Restaurant	Italian Restaurant	V
7	Bourse	28.2	48.869320	2.349510	1	Hotel	Cocktail Bar	V
8	Temple	31.8	48.817694	2.334053	1	Diner	Kebab Restaurant	
9	Temple	27.2	48.862940	2.361240	1	Japanese Restaurant	Italian Restaurant	Coffe
10	Temple	36.8	48.863570	2.360890	1	French Restaurant	Japanese Restaurant	Coc
11	Temple	21.3	48.860610	2.356056	1	Ice Cream Shop	Burger Joint	Art
12	Hôtel-de-Ville	31.3	48.858533	2.351467	1	French Restaurant	Ice Cream Shop	Art
13	Hôtel-de-Ville	42.2	48.855403	2.358123	1	French Restaurant	Clothing Store	Ice
15	Hôtel-de-Ville	37.9	48.853130	2.348860	1	French Restaurant	Plaza	Bc
18	Panthéon	70.4	48.841862	2.344037	1	French Restaurant	Bar	
19	Panthéon	43.3	48.849558	2.345876	1	French Restaurant	Hotel	
20	Luxembourg	29.3	48.854185	2.340354	1	French Restaurant	Hotel	
21	Luxembourg	71.6	48.848315	2.336586	1	Hotel	Café	
37	Entrepôt	47.2	48.869690	2.352660	1	Bar	Cocktail Bar	Re:
38	Entrepôt	60.9	48.869010	2.355640	1	French Restaurant	Bar	
39	Entrepôt	88.4	48.873030	2.369970	1	French Restaurant	Italian Restaurant	Coffe
40	Popincourt	72.6	48.864715	2.371208	1	French Restaurant	Bar	Re:
41	Popincourt	83.8	48.861662	2.376290	1	French Restaurant	Restaurant	

	Borough	Neighborhood's Area	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3
42	Popincourt	117.2	48.856947	2.378354	1	French Restaurant	Bar	Re:
43	Popincourt	93.0	48.852670	2.381060	1	French Restaurant	Bar	
45	Reuilly	186.3	48.844420	2.402270	1	Hotel	Pizza Place	Conv
46	Reuilly	190.3	48.833591	2.386031	1	Hotel	Bus Stop	
47	Reuilly	123.6	48.851370	2.372210	1	French Restaurant	Cocktail Bar	
49	Gobelins	304.4	48.835537	2.375975	1	Beer Garden	French Restaurant	N
50	Gobelins	223.2	48.822630	2.358390	1	Vietnamese Restaurant	Asian Restaurant	(Re:
53	Observatoire	135.7	48.824410	2.337780	1	Bus Stop	Italian Restaurant	J& Re:
55	Observatoire	178.5	48.844550	2.389940	1	French Restaurant	Supermarket	
59	Vaugirard	260.9	48.843870	2.286130	1	French Restaurant	Hotel	Re:
60	Passy	303.0	48.845444	2.254189	1	Tennis Court	French Restaurant	
67	Batignolles-Monceau	137.8	48.896496	2.324302	1	Supermarket	French Restaurant	J& Re:
70	Buttes-Montmartre	109.0	48.885010	2.354130	1	Indian Restaurant	Supermarket	Bc
71	Buttes-Montmartre	134.8	48.884362	2.357151	1	Indian Restaurant	Hotel	Re:
73	Buttes-Chaumont	237.7	48.897546	2.381922	1	Hotel	French Restaurant	
74	Buttes-Chaumont	183.6	48.876200	2.393620	1	Bar	Bakery	Re:
76	Ménilmontant	80.7	48.870759	2.385372	1	Bar	French Restaurant	Pizz
77	Ménilmontant	148.7	48.871551	2.403990	1	French Restaurant	Supermarket	
78	Ménilmontant	159.9	48.859890	2.389020	1	Beer Bar	Bar	Re:

Cluster 3

```
In [52]: paris_merged.loc[paris_merged['Cluster Labels'] == 2, paris_merged.columns[[1]
+ list(range(5, paris_merged.shape[1]))]]
```

Out[52]:

	Borough	Neighborhood's Area	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue
1	Louvre	41.2	48.862187	2.345243	2	French Restaurant	Bakery	Ice Cream Shop
14	Hôtel-de-Ville	48.7	48.851864	2.364707	2	French Restaurant	Hotel	Police Station
16	Panthéon	60.4	48.847723	2.353412	2	French Restaurant	Bar	Italian Restaurant
22	Luxembourg	86.1	48.842822	2.332133	2	French Restaurant	Creperie	Hotel
25	Palais-Bourbon	107.4	48.858253	2.315687	2	French Restaurant	Hotel	Police Station
27	Palais-Bourbon	138.2	48.858413	2.300991	2	French Restaurant	Italian Restaurant	Hotel
32	Opéra	71.7	48.879270	2.348840	2	French Restaurant	Hotel	Italian Restaurant
35	Opéra	50.1	48.882824	2.344555	2	French Restaurant	Hotel	
56	Vaugirard	283.1	48.841179	2.297599	2	French Restaurant	Italian Restaurant	Bar
61	Passy	203.7	48.858340	2.270916	2	French Restaurant	Bakery	Italian Restaurant
65	Batignolles-Monceau	138.4	48.883091	2.299712	2	French Restaurant	Hotel	Bar
66	Batignolles-Monceau	144.2	48.889600	2.318920	2	French Restaurant	Bar	Restaurant
68	Buttes-Montmartre	190.6	48.890303	2.330582	2	French Restaurant	Bar	Hotel
72	Buttes-Chaumont	128.6	48.877085	2.387862	2	French Restaurant	Bar	Cafe
75	Buttes-Chaumont	129.5	48.883945	2.385625	2	French Restaurant	Supermarket	
79	Ménilmontant	209.1	48.854188	2.396330	2	French Restaurant	Bar	Farm Market

Cluster 4


```
In [53]: paris_merged.loc[paris_merged['Cluster Labels'] == 3, paris_merged.columns[[1]
+ list(range(5, paris_merged.shape[1]))]]
```

Out[53]:

	Borough	Neighborhood's Area	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue
69	Buttes-Montmartre	165.3	48.900988	2.347675	3	Flea Market	Fast Food Restaurant	Fried Chicken Joint

```
In [54]: len(paris_merged.loc[paris_merged['Cluster Labels'] == 0, paris_merged.columns
[[1] + list(range(5, paris_merged.shape[1]))]]), len(paris_merged.loc[paris_me
rged['Cluster Labels'] == 1, paris_merged.columns[[1] + list(range(5, paris_me
rged.shape[1]))]]), len(paris_merged.loc[paris_merged['Cluster Labels'] == 2,
paris_merged.columns[[1] + list(range(5, paris_merged.shape[1]))]]))
```

Out[54]: (23, 40, 16)

In []:

In []:

Results

After conducting K-Means algorithm, the data was clustered into 4 groups / clusters.

1st cluster consists of **23 neighborhoods** with hotels and french restaurants as their 1st Most Common Venue (marked with red color on the map above). This establishments are located on the western part of Paris.

2nd cluster consists of **40 neighborhoods** with restaurants, bars and hotels as their 1st Most Common Venue (marked with violet / purple color on the map above). This establishments are located on the central, eastern, and southern part of Paris.

3rd cluster consists of **16 neighborhoods** with only french restaurants as their 1st Most Common Venue, bars and hotels as their 2nd Most Common Venue (marked with blue color on the map above). This establishments are evenly distributed throughout the city.

4th cluster include the **only Flea Market** in the data (marked with yellow color on the map above). It is located on the north of Paris.

Discussion and Conclusion

In future research it would be additional help to use population data of each neighborhood in Paris. Since the most preferable place to open a restaurant is that where the number of competitors is the lowest. For our data it's western part of Paris up to public park Bois de Boulogne. However, it's not possible to conclude decisively about the exact place.

The best way to determine the exact place for opening a restaurant in western part of Paris is to use additional data, for example, anonymized data of mobile operators. These data will show a more detailed picture of potential visitors to the new restaurant

In []: