Clustering the Cities of Grand Paris

IBM Data Science Professional Certificate

Capstone Project

Frédéric BENOIT – September 2021

Introduction / Business Problem

With the covid-19 crisis, more and more people want to move from Paris, where the housing prices are very high and therefore the house surfaces small, either to a different region or to the area around Paris where the real estate is less expensive.

For the people who want to move from Paris to the area around Paris but who don't know accurately all cities around Paris, it would be convenient to have a tool capable of proposing some areas corresponding to their needs (transports, employment, education, etc) and their centers of interest (entertainment, sport, etc).

This tool could be used by individuals, could be helpful for estate agents as well.

The idea is to cluster the cities around Paris (officially grouped under the name "Grand Paris") by the categories of venues they have the most. The people who want to relocate to this area would simply have to choose the cluster that corresponds the best to their needs and center of interests to have the list of cities best suited for them

Data

For this study, we will need to list all venues available in the neighborhood of each city of the Grand Paris area. This requires the following steps of data acquisition:

- List the cities of Grand Paris.
- Get their geospatial data.
- From these geospatial data, list the venues available in each city.

List of cities of Grand Paris.

A survey in a search engine led to different possibilities, but the most obvious one is the use of open data provided by the French Government:

https://www.data.gouv.fr/fr/datasets/communes-de-la-metropole-du-grand-paris-par-ept/

These data can conveniently be downloaded as an excel file, which contains:

- The city names (column "Libellé géographique").
- Some city codes (column"code géographique"), which are not the postal code therefore not directly usable by interfaces using the postal code.
- The codes "région", "département" and "EPT", which are not relevant for our study.

Geospatial data

We will use the geocoder python package, which takes the city names in string format as inputs and sends out the longitude and latitude of the city center.

Venues in each city

In order to get a list of the venues available in each city, I will use the Foursquare API. Foursquare is a technology company that built a massive dataset of crowd-sourced location data.

By using the Foursquare API, one can get the location, the category, and many information of the venues located in a specified radius around a geospatial position (defined by a longitude and a latitude).

Methodology

Step 1: List the cities of the Grand Paris area

As mentioned in the "Data section", this data is available as open data from the French Governement (https://www.data.gouv.fr/fr/datasets/communes-de-la-metropole-du-grand-paris-par-ept/), and can be downloaded in Excel format. Data scraping from other sites is also possible but would require more effort for the same result.

Let us import this excel file in a dataframe:

	Code géographique	Région	Département	Libellé géographique	EPT
0	75056	11	75	Paris	Ville de Paris - T1
1	94015	11	94	Bry-sur-Marne	Paris-Est-Marne et Bois - T10
2	94017	11	94	Champigny-sur-Marne	Paris-Est-Marne et Bois - T10
3	94018	11	94	Charenton-le-Pont	Paris-Est-Marne et Bois - T10
4	94033	11	94	Fontenay-sous-Bois	Paris-Est-Marne et Bois - T10

The data cleaning is quite straightforward:

- I keep only the column containing the cities' names (column"Libellé géographique").
- I remove the first row (the city of Paris) because I exclude it from the study.

Now I get a dataframe containing the list of cities in the Grand Paris area:

	index	Libellé géographique
0	1	Bry-sur-Marne
1	2	Champigny-sur-Marne
2	3	Charenton-le-Pont
3	4	Fontenay-sous-Bois
4	5	Joinville-le-Pont
5	6	Maisons-Alfort
6	7	Nogent-sur-Marne
7	8	Le Perreux-sur-Marne
8	9	Saint-Mandé
9	10	Saint-Maur-des-Fossés

The dataframe contains 130 cities.

Step 2: Add the geospatial data

As mentioned in the “data” section, I want to identify the venues available in each city. For this, the Foursquare API proposes two possibilities:

- Either use the city name as input, in this case the API will send out all venue in this city.
- Or use latitude and longitude as inputs to the API, in this case the API will list the venues available in a given radius of this geographic point.

I tried both possibilities but in the principle, it makes more sense to list the venue situated in a given radius around the city center, so I go for the second possibility. This solution requires to the latitude and longitude of each city center. I use the geocoder python library to add the longitude and latitude in the previous dataframe:

	index	Libellé géographique	Latitude	Longitude
0	1	Bry-sur-Marne	48.835287	2.519332
1	2	Champigny-sur-Marne	48.813776	2.510738
2	3	Charenton-le-Pont	48.819848	2.415951
3	4	Fontenay-sous-Bois	48.849072	2.474935
4	5	Joinville-le-Pont	48.818372	2.466808

Step 3: List the venues in each city

I use the Foursquare API with an input radius of 2 000 m to list in a dataframe the venues around each city center:

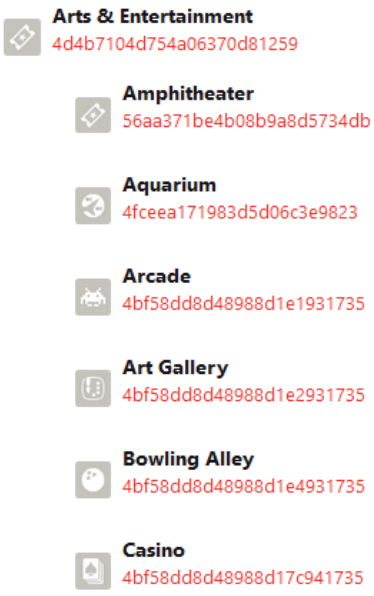
	City	City Latitude	City Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Bry-sur-Marne	48.835287	2.519332	Keftedes & Tzatziki	48.835981	2.513415	Greek Restaurant
1	Bry-sur-Marne	48.835287	2.519332	Studios de Bry	48.835987	2.534052	Film Studio
2	Bry-sur-Marne	48.835287	2.519332	Les Délices de Fred	48.835291	2.498486	Bakery
3	Bry-sur-Marne	48.835287	2.519332	IKEA	48.827993	2.530133	Furniture / Home Store
4	Bry-sur-Marne	48.835287	2.519332	Quai Est	48.834266	2.516968	French Restaurant

Step 4: Group the venues by super-categories

At this point, the Foursquare API provides us with too many venue categories, which in my opinion makes a good clustering difficult for my study. It would for example be much more relevant to group all restaurants, whichever their specialty they have, in a category "Food".

Foursquare gives us this possibility since super-categories are already defined in the developer section of their website: https://developer.foursquare.com/docs/build-with-foursquare/categories/

For example, the venues Amphitheater, Aquarium, Arcade, Art Gallery are grouped into a super-category called “Arts & Entertainment”:



I create a function to make a table listing the venue categories and the super-categories they belong to, and save this table in a dataframe:

	index	venue	venue_category	level
0	0	Amphitheater	Arts & Entertainment	0
1	1	Aquarium	Arts & Entertainment	0
2	2	Arcade	Arts & Entertainment	0
3	3	Art Gallery	Arts & Entertainment	0
4	4	Bowling Alley	Arts & Entertainment	0
5	5	Casino	Arts & Entertainment	0
6	6	Circus	Arts & Entertainment	0
7	7	Comedy Club	Arts & Entertainment	0
8	8	Concert Hall	Arts & Entertainment	0
9	9	Country Dance Club	Arts & Entertainment	0
10	10	Disc Golf	Arts & Entertainment	0

Note: the category also come with a level, which represents their position in a tree. I decide to keep only the level 0 super-categories, because I am only interested in the highest categories.

I then add these super-categories to my dataframe containing the cities' names, their longitudes, their latitudes and their venues:

	City	City Latitude	City Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	Venue Group
0	Bry-sur-Marne	48.835287	2.519332	Keftedes & Tzatziki	48.835981	2.513415	Greek Restaurant	Food
1	Bry-sur-Marne	48.835287	2.519332	Studios de Bry	48.835987	2.534052	Film Studio	Shop & Service
2	Bry-sur-Marne	48.835287	2.519332	Les Délices de Fred	48.835291	2.498486	Bakery	Food
3	Bry-sur-Marne	48.835287	2.519332	IKEA	48.827993	2.530133	Furniture / Home Store	Shop & Service
4	Bry-sur-Marne	48.835287	2.519332	Quai Est	48.834266	2.516968	French Restaurant	Food

I notice that there are 8 super-categories:

- Food
- Shop & Service
- Outdoors & Recreation
- Travel & Transport

- Arts & Entertainment
- Nightlife Spot
- Professional & Other Places
- Residence.

Step 5: Cluster the cities

In order to meet the study target, determine which city fits the best with the inhabitants groups, I want to cluster the cities using these super-categories of venues as criteria.

I run the k-means to cluster the neighborhood into 4 clusters, which seems to be the best suited to define meaningful groups and I add the cluster labels to the dataframe listing the cities characteristics:

	Cluster Labels	City	Arts & Entertainment	Food	Nightlife Spot	Outdoors & Recreation	Professional & Other Places	Residence	Shop & Service	Travel & Transport
0	1	Ablon-sur-Seine	0.000000	0.400000	0.000000	0.000000	0.00	0.0	0.333333	0.266667
1	1	Alfortville	0.047619	0.333333	0.047619	0.214286	0.00	0.0	0.285714	0.071429
2	1	Antony	0.033333	0.400000	0.033333	0.200000	0.00	0.0	0.266667	0.066667
3	0	Arcueil	0.060000	0.490000	0.020000	0.160000	0.01	0.0	0.160000	0.100000
4	0	Argenteuil	0.066667	0.466667	0.000000	0.000000	0.00	0.0	0.200000	0.266667

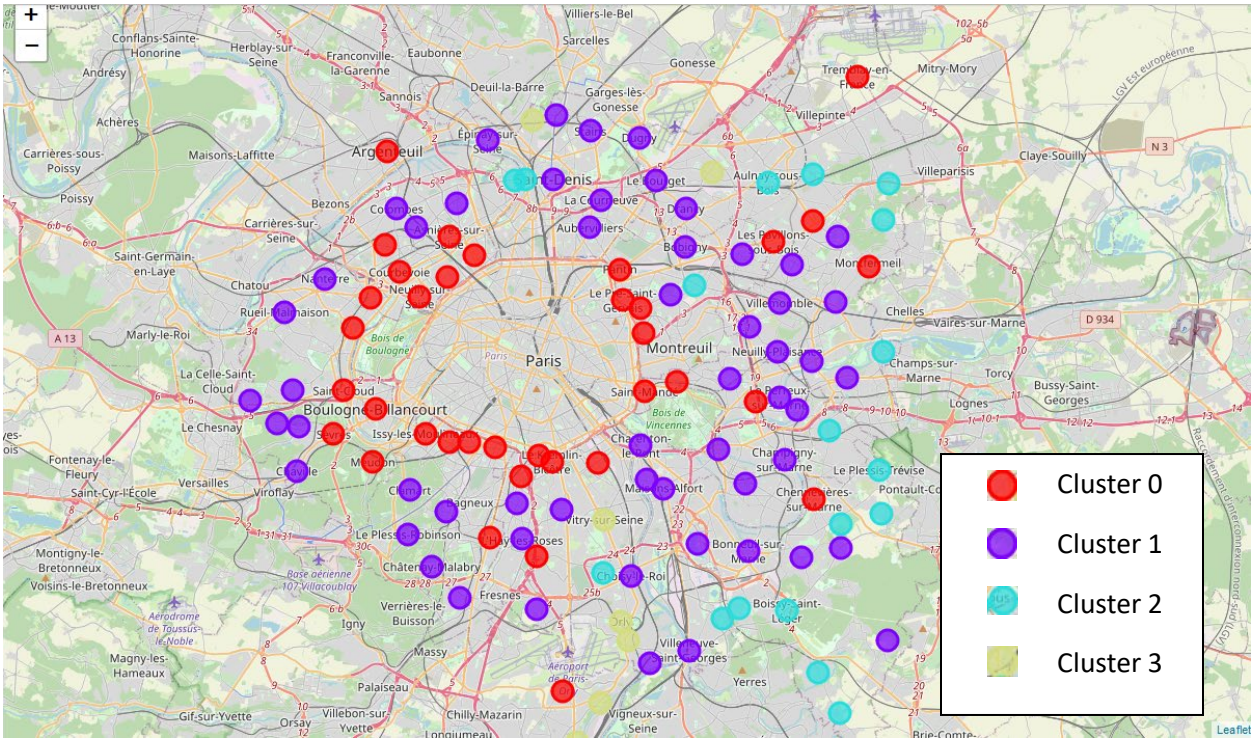
If I analyze the clusters, I notice that:

Cluster Labels	Arts & Entertainment	Food	Nightlife Spot	Outdoors & Recreation	Professional & Other Places	Residence	Shop & Service	Travel & Transport
0	0.055275	0.538116	0.032077	0.112140	0.005029	0.000393	0.156135	0.100835
1	0.048000	0.336544	0.014140	0.174600	0.002478	0.000565	0.272524	0.151149
2	0.026816	0.220651	0.002156	0.101520	0.016951	0.000000	0.537240	0.094666
3	0.021296	0.186731	0.000000	0.027437	0.008547	0.000000	0.274882	0.481107

- The cluster 0 has the higher density for food, nightlife, Arts & Entertainment and is 2nd for the professional category, so I would recommend it to the students, young workers.
- The cluster 1 has more residence and outdoor and recreational areas, so I would recommend it for practitioners of outdoor activities and possibly retired people.
- In the cluster 2, there is a higher density of professional places, shops and services; I would recommend it to install companies, businesses.
- The cluster 3 is the best for Travel and Transport, so I would recommend it for the frequent travellers.

Results

If I represent the clusters on a map:



Discussion

I find some logic by comparing the map and the clusters:

- The first circle around (red dots) Paris in mainly made of cluster 0, it also corresponds to the most urbanized area.
- The outer area around Paris is the cluster 1 (purple dots) (recommended for practitioners of outdoors activities and possibly retired people), not a surprise since it is the least urbanized area.
- The cluster 3 (green dots) (recommended for frequent travelers) is composed of cities located close to the airports.

The results of the study are consistent with the basic knowledge that I have of Paris suburbs.

Conclusion

It is possible to determine by clustering the best areas to live depending on the individuals needs and centers of interests.

However, in the course of the study some limitations of this study appeared:

- The clustering is based on the relative distribution of each venues category in each city, but not based on their density. This means that a city with one restaurant as only venue for example will go into the cluster of cities where foods venues are majority, though its density of food venues is low. In order to solve this issue, a different encoding must be made.

- The Foursquare datasets are sometimes not comprehensive enough:

    - In the Paris area, there is no venue of categories "College and Universities" listed, which makes it impossible to define a cluster recommended for families for example.

    - The category "Residence" includes very few types of venues.

Future directions

This study could be the starting point for an application in which a customer enters its needs and centers of interest and obtains the best groups of cities to live in.

For this, it would be necessary to:

- Make a convenient user interface.

- Enlarge the number of input criteria:
    - increase the venues dataset (ex: with schools)
    - add external data (ex: housing prices).

- Cluster based venues density and not venues distribution.