Study of venues for office land acquisiton in the city of Madrid.

Eduardo Manuel Pérez Rodríguez

August 2019

Table of Contents

1. Introduction

Background

Our client is a virtual international real estate company focused in renting offices and coworking spaces, based in Central Europe. The company is extending its business to Southern Europe, choosing **Madrid** for the expansion, since the company have some partner companies already there. Madrid, capital of Spain and the 4th most populated city in Europe, also has the advantages of very good communications to the city and regional and international relevance to make business in Southern Europe.

Problem

The company is facing problems since it is not familiar with the city area to acquire land for new offices, so they are requiring an analysis from data science experts in order to process all the relevant data about the city.

Objective

The main objective is to choose the Madrid city's neighbourhood most suitable for the expansion of our client to buy land for a new office, with issues not related with price.

The business managers of our client, our target audience and main stakeholder, transmitted to the data science team some clear ideas about the basic requirements of the new placement:

- Accesibility to **public transport**, specially Metro stations (Subway), to ease the access to the users of the new offices and/or coworking spaces.
- **Hotels** nearby, making easier to receive international clients.
- **Restaurants** of any kind, so the personnel does not need to take long breaks for meals.

We will need to make an exhaustive study regarding this in order to get the preferred neighborhoods for our client purposes.

2. Data

Data sources

Regarding the data to be used, the data science team has found the following sources:

Public data bank of Madrid municipality

- Website: https://datos.madrid.es
- Data provided mainly in CSV or TXT files directly from the website.
- CSV and TXT can be imported to a dataframe using **Pandas** (https://pandas.pydata.org/) Python library.
- Data to be obtained: Districts (Nombre del distrito), neighborhoods(Nombre del barrio), postal codes(Codigo Postal), geographical locations(Longitud, Latitud) and land value.

Example (Python):

data=pd.read_csv('https://datos.madrid.es/egob/catalogo/200075-1-callejero.csv', encoding='iso-8859-1', sep=';')

Foursquare API

- Website: https://es.foursquare.com/developers/
- Data provided using Foursquare API.
- Foursquare API data can be turned to dataframes using **Pandas** (https://pandas.pydata.org/) and **Requests** (https://2.python-requests.org/en/master/) Python libraries.
- Data to be obtained: Services (Public transport, offices/business centers, hotels, restaurants…) and geographical data of the venues.

Example (Python):

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

OpenStreetMap API

- Website: https://www.openstreetmap.org/
- Data provided using OpenStreetMap API.
- Geographical data can be extracted using **geocoder** library (https://geocoder.readthedocs.io/).
- Data to be obtained: Geographical data of Madrid areas, neighborhoods or postal codes.

Example (Python):

```
import geocoder g = geocoder.osm('Nuevos Ministerios, Madrid') Longitude=g.x Latitude=g.y
```

Preprocessing and data wrangling

For this problem, we have decided to select the **neighborhood** as the minimal division of Madrid urban area. There are in total 131 different administrative neighborhoods to be analysed.

We will get the data related to neighborhoods and geographical coordinates from the official **Madrid municipality data bank** previously mentioned in **CSV** format. This data has the columns names in Spanish language and sometimes uses acronyms, so we will preprocess the data to get the columns in a readable English language in this case.

The coordinates in the data of Madrid's data bank are given in **WGS84 coordinates** format. We need to convert them to **decimal coordinates** format using libraries some customized functions in Python, what generates some aditional tasks for the data wrangling.

Once we have the neighborhoods and coordinates are obtained, we can start working in the acquisition of data from **Foursquare API**. We will first get all the main venues for each neighborhood for clustering the neighborhoods later.

3. Methodology

As we mentioned in the data description, we will need to obtain a complete dataframe with the following columns in order to apply clustering:

- **Neighborhood**: Division of Madrid's urban area.
- **Top 10 venues per neighborhood**: One in each column of the data frame.

After that, we will cluster the neighborhoods using the **k-means** algorithm. This way, we will separate the 131 neighborhoods of Madrid in 10 different clusters by similarity. We will obtained a 'Cluster Label' column with the obtained cluster and plot the neighborhoods cordinates in a map, painting the ones of each cluster in a different color to make the information easily understandable.

As an extra, once we get all the clusters, we will do a **weighted scoring** regarding the most important venues mentioned by our client: Hotels, restaurants and metro stations. We will need a column for each of these venues and an extra one with the obtained score.

This way, in the end, we will two dataframes for our client:

- Most suitable **cluster** of neighborhoods.
- Most suitable **neighborhoods**.

We will provide both in order of preference, so our client can make a decision easier.

Regarding the plotting, we will count on the **geocoder** and **folium** libraries in Python, since they allow us to locate an address in a map and plot the neighborhoods as well. For the geocoding we will use the **OpenStreetMaps** service.

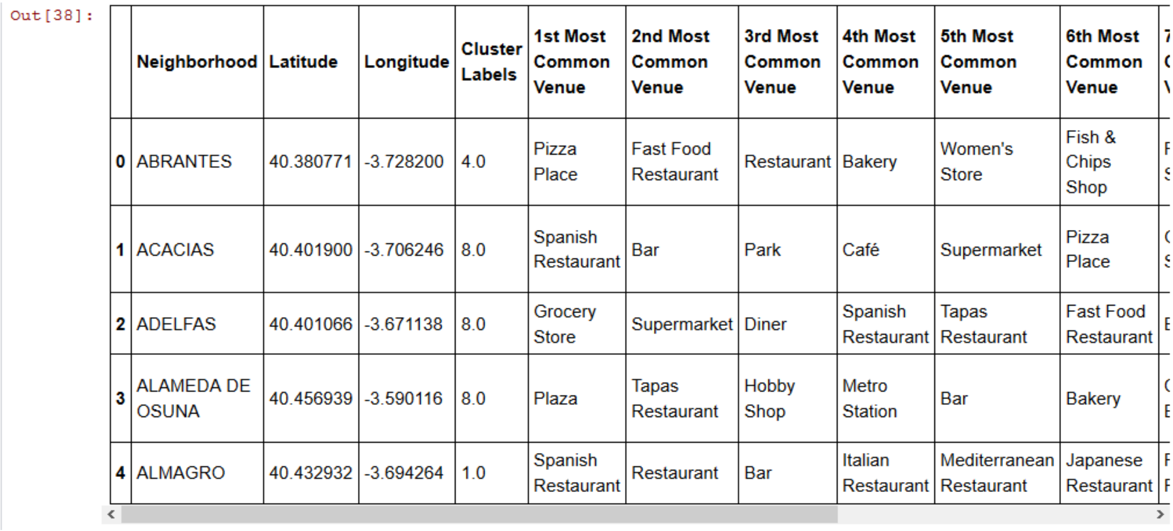
Clustering

The reason to choose clustering is the ability of this method to classify different negihborhoods without any supervision.

The analysis has been done separating ten different clusters of Madrid’s neighborhoods, evaluating the ten most common venues for each neighborhood, obtained from the **Foursquare API** using the coordinates given in the Madrid’s municipality dataset.

It must be said that three neighborhoods have been discarded for the analysis, due to insufficient data from the Foursquare API (Aeropuerto, El Cañaveral and El Goloso). After checking them in a map, we find that there are non-urban constructions and military or reserved areas, so we will skip these outside our study.

After doing the analysis we have obtained the ten clusters without being ranked or scored yet (detailed in the code attached) and a map plotting each neighborhood with a different colour per cluster.



Out[38]:

	Neighborhood	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7 (\
0	ABRANTES	40.380771	-3.728200	4.0	Pizza Place	Fast Food Restaurant	Restaurant	Bakery	Women's Store	Fish & Chips Shop	F S
1	ACACIAS	40.401900	-3.706246	8.0	Spanish Restaurant	Bar	Park	Café	Supermarket	Pizza Place	(S
2	ADELFAS	40.401066	-3.671138	8.0	Grocery Store	Supermarket	Diner	Spanish Restaurant	Tapas Restaurant	Fast Food Restaurant	E
3	ALAMEDA DE OSUNA	40.456939	-3.590116	8.0	Plaza	Tapas Restaurant	Hobby Shop	Metro Station	Bar	Bakery	(E
4	ALMAGRO	40.432932	-3.694264	1.0	Spanish Restaurant	Restaurant	Bar	Italian Restaurant	Mediterranean Restaurant	Japanese Restaurant	F F

Figure 1: Dataframe of clustered neighborhoods, including coordinates and most common venues.

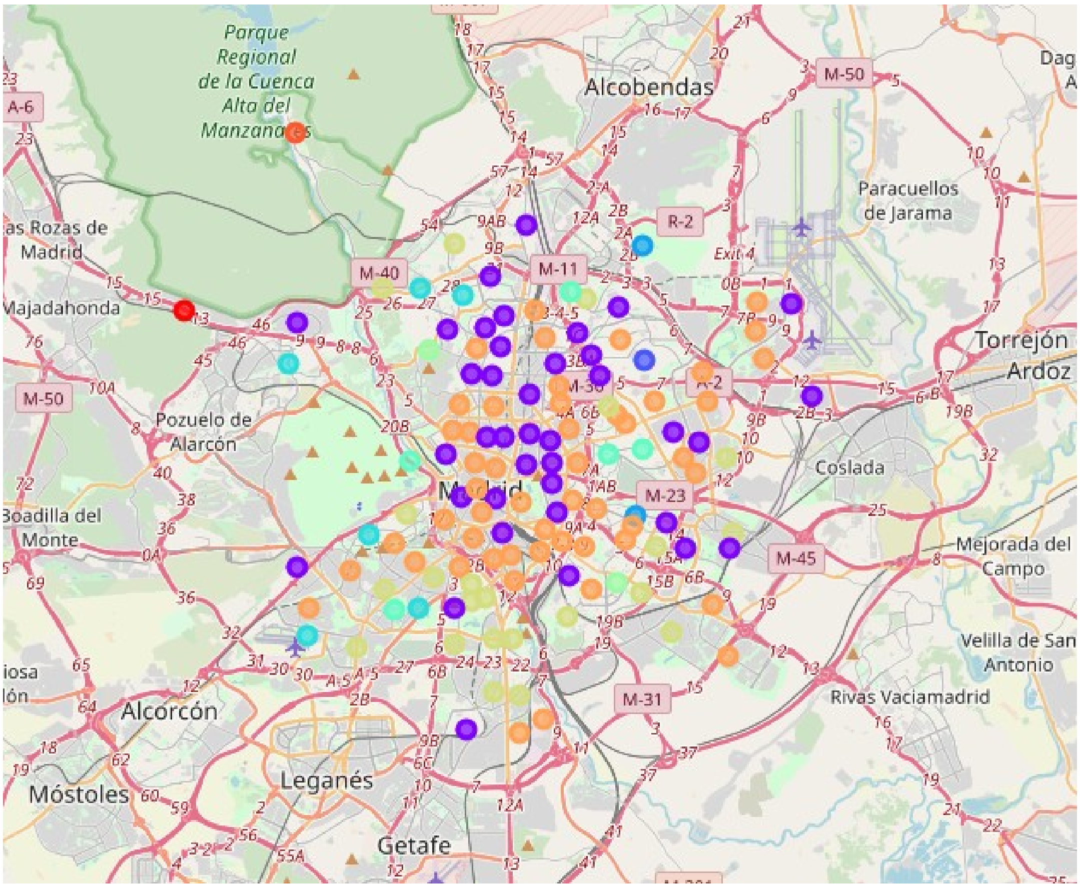


Figure 2: Interactive map with clustered neighborhoods located geographically.

Scoring

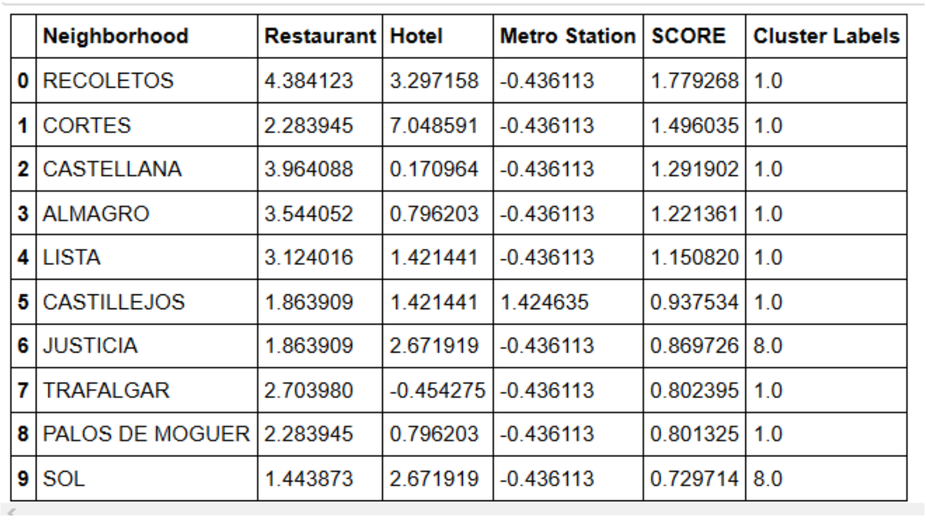
The motivation for quantitative scoring, is getting a rank of best to worst neighborhoods and clusters, making easier to make a decision. Regarding this method, the analysis has been done weighting the three main venues demanded by our client equally as it follows:

- Metro Stations (33%)
- Hotels (33%)
- Restaurants (33%)

Standarization of the values has been needed in order to ensure the correct weighting of these venues for a fair scoring. We will use **StandardScaler** from **sklearn** package in Python, since it allow us to find relevant differences between neighborhoods.

In the end, we have obtained the two expected dataframes:

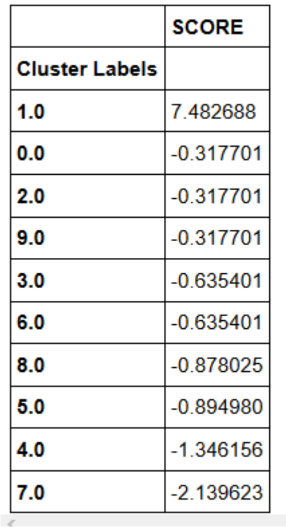
- **Scored neighborhoods:**



	Neighborhood	Restaurant	Hotel	Metro Station	SCORE	Cluster Labels
0	RECOLETOS	4.384123	3.297158	-0.436113	1.779268	1.0
1	CORTES	2.283945	7.048591	-0.436113	1.496035	1.0
2	CASTELLANA	3.964088	0.170964	-0.436113	1.291902	1.0
3	ALMAGRO	3.544052	0.796203	-0.436113	1.221361	1.0
4	LISTA	3.124016	1.421441	-0.436113	1.150820	1.0
5	CASTILLEJOS	1.863909	1.421441	1.424635	0.937534	1.0
6	JUSTICIA	1.863909	2.671919	-0.436113	0.869726	8.0
7	TRAFALGAR	2.703980	-0.454275	-0.436113	0.802395	1.0
8	PALOS DE MOGUER	2.283945	0.796203	-0.436113	0.801325	1.0
9	SOL	1.443873	2.671919	-0.436113	0.729714	8.0

Figure 3: Scored neighborhoods, according to our company requests.

- **Scored clusters**, where the score of each cluster is the sum of all the scores of each neighborhood:



	SCORE
Cluster Labels	
1.0	7.482688
0.0	-0.317701
2.0	-0.317701
9.0	-0.317701
3.0	-0.635401
6.0	-0.635401
8.0	-0.878025
5.0	-0.894980
4.0	-1.346156
7.0	-2.139623

Figure 4: Scored clusters

4. Results

We can finally advise our client that the most suitable neighborhoods are the ones located in **cluster no. 1** according to our study.

Regarding neighborhoods in particular, after using the scoring technique, we find that the following neighborhoods seem the best for our purposes: **Recoletos, Cortes, Castellana, Almagro, Lista, Castillejos, Justicia, Trafalgar, Palos de Moguer and Sol**. Most of them belong to **cluster no. 1** as well.

5. Discussion

In order to have even more information to make the decission of our client, several additional studies can be done:

- Studying **land value** for scoring: It will be convenient to get a more optimal solution to our problem.
- Studying **regression and correlation between land value and venues:** This way we can study if some venues have stronger impact on the land value and find some underrated neighborhoods that may be interesting for our client.
- Clustering neighborhoods including the land value as another parameter (the cheaper the better) and skip scoring. This could be the a better cost-effective solution.

6. Conclussion

This kind of study using clustering represents an easy way of cluster items without any supervision, which is the main interest on this kind of analysis. It is interesting that when using quantitative scoring, the results of both methods agree. The purpose of combining both is still useful since it is possible to rank neighborhoods separately. For our case, our company has now a much clearer view, not only about the clustering of different areas in Madrid, but also after ranking both clusters and neighborhoods in order to make a final decission.

7. References

1. Banco de datos de Madrid. Callejero Oficial del Ayuntamiento de Madrid. (https://datos.madrid.es/). August 2019.

2. Foursquare API (https://developer.foursquare.com/). August 2019.

3. Geocoder library documentation for OpenStreetMaps (https://geocoder.readthedocs.io/providers/OpenStreetMap.html). August 2019.

Eduardo Manuel Pérez Rodríguez

August 2019