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

Eduardo Manuel Pérez Rodríguez

August 2019

# Introduction

- **Background**

Our client is a virtual international r**eal 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.

# Introduction

- **Problem**

The company is 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**

Choose **which Madrid city neighborhoods are most suitable for the expansion of our company** to buy land for a new office.

Basic requirements of the new placement:

  - Accesibility to **Metro stations**.
  - **Hotels** nearby.
  - **Restaurants** nearby.
  - **Land value is outside the scope of the study in this phase.**

# Data sources

- **Banco de datos de Madrid.** Callejero Oficial del Ayuntamiento de Madrid. (CSV file) (https://datos.madrid.es/). August 2019.

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

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

# Data preprocessing and wrangling

Difficulties overcame while studying the provided data:

- Standarization of Madrid **neighborhood names**.
- Original **format of coordinates** (WGS84 instead of decimal).
- **Missing or zero values** in the original dataset.

# Methodology

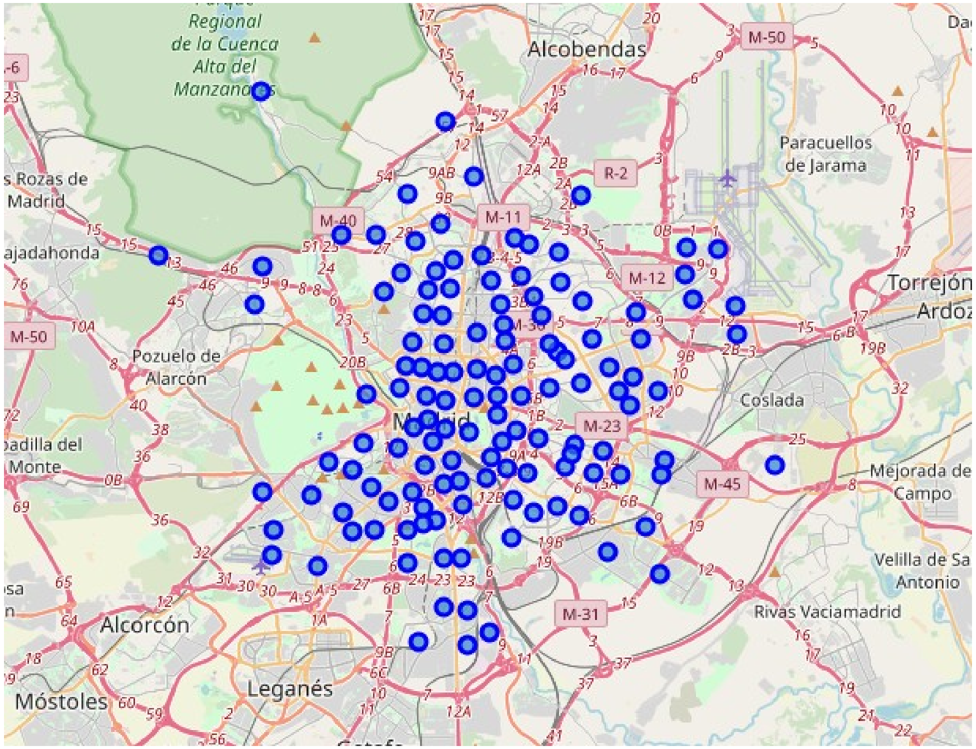
- **Venues obtention**
  - Using Foursquare API.
- **Geocoding**
  - Using geocode library (Python) and OpenStreetMaps API.
- **Plotting**
  - Using folium library (Python).
- **Clustering**
  - Using K-means algorithm (unsupervised).
  - Using sklearn library (Python).
- **Scoring**
  - Quantitative scoring using StandardScaler.

# Methodology

- **Geocoding** a centered spot in Madrid.

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

- **Plotting** with **folium**: Geographical location of each neighborhood.



# Methodology

- Venues obtention for each neighborhood, using **Foursquare API:**

| | Neighborhood | Yoga Studio | Accessories Store | Adult Boutique | African Restaurant | Airport | American Restaurant | Arcade | Arepa Restaurant | Argentinian Restaurant | Art Gallery | Art Museum | Arts & Crafts Store | Asian Restaurant | Athlet & Spo |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | ABRANTES | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 | 0.0 | 0.0 | 0.0 | 0.00 | 0.00 | 0.0 | 0.000000 | |
| **1** | ACACIAS | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 | 0.0 | 0.0 | 0.0 | 0.00 | 0.00 | 0.0 | 0.015152 | |
| **2** | ADELFAS | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 | 0.0 | 0.0 | 0.0 | 0.00 | 0.00 | 0.0 | 0.027027 | |
| **3** | ALAMEDA DE OSUNA | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.00 | 0.0 | 0.0 | 0.0 | 0.00 | 0.00 | 0.0 | 0.000000 | |
| **4** | ALMAGRO | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.01 | 0.0 | 0.0 | 0.0 | 0.01 | 0.01 | 0.0 | 0.000000 | |

- **Clustering of neighborhoods,** using **kmeans from sklearn library:**

```
[ ] from sklearn.cluster import KMeans
```

```
[ ] # set number of clusters
    kclusters = 10

    Madrid_grouped_clustering = Madrid_grouped.drop('Neighborhood', 1)

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

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

```
array([4, 8, 8, 8, 1, 1, 7, 8, 8, 7], dtype=int32)
```

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

    Madrid_merged = df_geo

    # merge toronto_grouped with toronto_data to add latitude/longitude for each neighborhood
    Madrid_merged = Madrid_merged.join(neighborhoods_venues_sorted.set_index('Neighborhood'), on='Neighborhood')
```

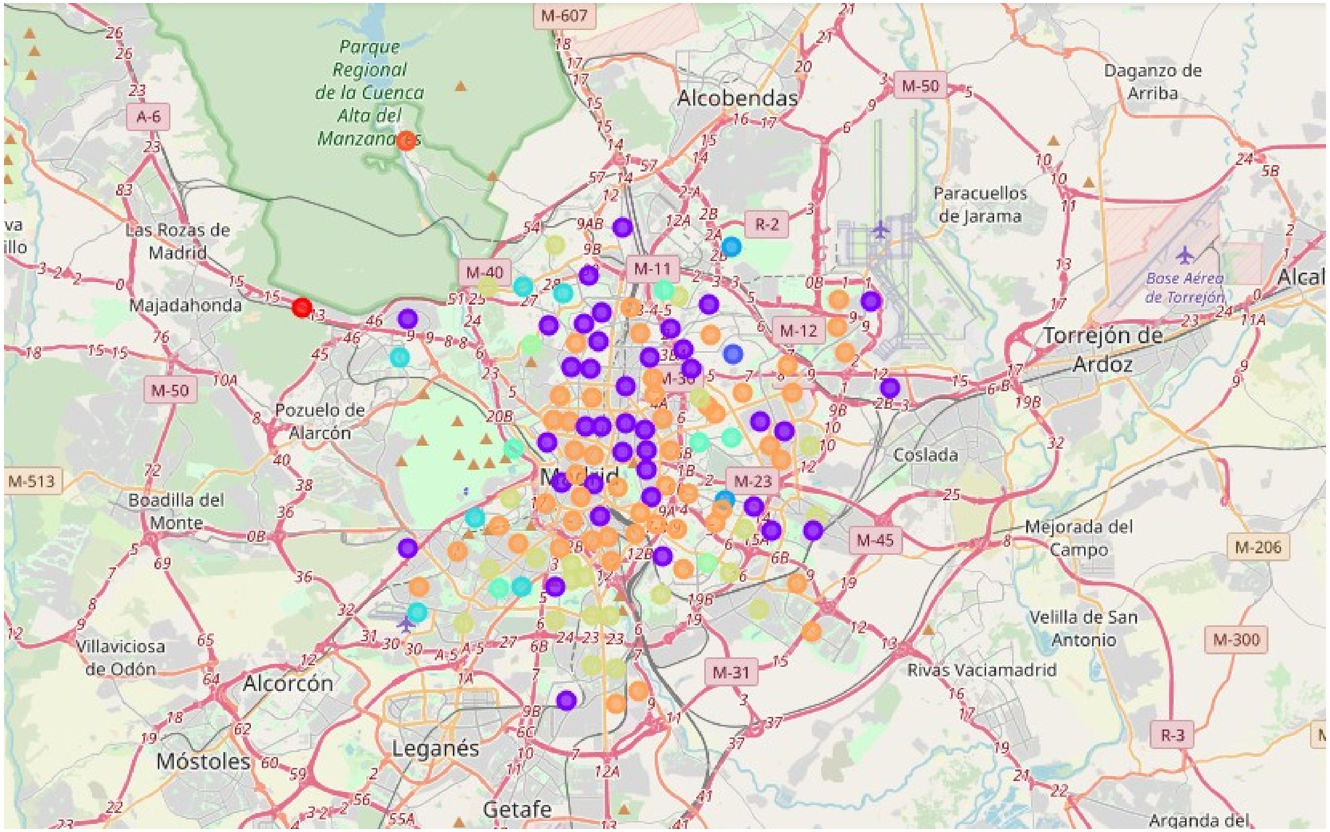
# Methodology

- **Clustering of neighborhoods,** using **kmeans** from **sklearn library (Result):**

| | 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 | 7th Most Common Venue | 8th Most Common Venue | 9th Most Common Venue | 10th Most Common Venue |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | ABRANTES | 40.380771 | -3.728200 | 4.0 | Pizza Place | Fast Food Restaurant | Restaurant | Bakery | Women's Store | Fish & Chips Shop | Fabric Shop | Falafel Restaurant | Farm | Farmers Market |
| **1** | ACACIAS | 40.401900 | -3.706246 | 8.0 | Spanish Restaurant | Bar | Park | Café | Supermarket | Pizza Place | Grocery Store | Restaurant | Food & Drink Shop | Gym |
| **2** | ADELFAS | 40.401066 | -3.671138 | 8.0 | Grocery Store | Supermarket | Diner | Spanish Restaurant | Tapas Restaurant | Fast Food Restaurant | Bar | Breakfast Spot | Football Stadium | Brewery |
| **3** | ALAMEDA DE OSUNA | 40.456939 | -3.590116 | 8.0 | Plaza | Tapas Restaurant | Hobby Shop | Metro Station | Bar | Bakery | Cocktail Bar | Fried Chicken Joint | Chinese Restaurant | Bistro |
| **4** | ALMAGRO | 40.432932 | -3.694264 | 1.0 | Spanish Restaurant | Restaurant | Bar | Italian Restaurant | Mediterranean Restaurant | Japanese Restaurant | French Restaurant | Plaza | Coffee Shop | Café |

# Methodology

- **Plotting** with **folium**: Geographical location of each neighborhood, **clustered**:



# Methodology

- **Scoring** of **neighborhoods**, using **StandardScaler:**

| | 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 |

# Methodology

- **Scoring** of **clusters**, using **StandardScaler:**

| Cluster Labels | SCORE |
|---|---|
| 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 |

# Results

- The most suitable neighborhoods are the ones located in **cluster no. 1** according to our study.

- 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.

# 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.
- Studying **regression and correlation between land value and venues.**
- **Clustering neighborhoods from a new matrix** with all the information **including land value without scoring.**

Last option may be the most cost-effective way for a second phase of this same problem.

# Conclussion

- Clustering represents an easy way of classifying items **without any supervision**, which is the main interest on this kind of analysis.
- The purpose of combining both, clustering and scoring is 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.**

# 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.

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

Eduardo Manuel Pérez Rodríguez

August 2019