Analysing Contagion Rates in Madrid

Luis Deyá Navarro

December 23 2020

1. Introduction

1.1 Background

During the first months of the pandemic caused by the new SARS-CoV-2 coronavirus, many experts pointed out that the high population density of cities such as Madrid was responsible for the large metropolises became sources of expansion for Covid-19. Months later, the expansion of the scientific literature on the virus suggests that population density is not the only factor that explains the speed of its spread. The best example is found in the United States, one of the most affected countries where the coronavirus has managed to spread rapidly beyond the main urban areas. Many expert researchers in public health and urban planning agree that the concentration of people within a given area is not telling us the full story about the spread of the virus. On this basis, various studies carried out throughout the pandemic indicate that, in the spread of the virus, other factors such as existing connections between different communities, access to medical care or the overcrowding of certain areas play a crucial role.

1.2 Problem

Researcher points out that the population density of a city or region is not a faithful reflection of more subtle aspects such as the meetings of its inhabitants in smaller spaces such as the family and social environment. When we talk about overcrowding, we must understand that this concept is applicable to such common events as concerts, entertainment venues or family gatherings. But it also refers to the fact that, due to socioeconomic conditions, many people are forced to live together in small spaces. Even cultural elements influence, for example, families living in multigenerational households. This concept can even be extended to public transport, which can be saturated with large crowds of people.

1.3 Interest

The food and restoration industry has been one of the most heavily affected by the restrictions imposed by governments to reduce the expansion of Covid-19, being forced to restrict opening hours, venue capacity, or close altogether. Bars, restaurants, and cafes are the most affected. It would be interesting to use publicly available contagion rate data and population density data, along with Foursquare venue data (which focuses on food and restoration mostly) to see if the conclusions drawn by the studies mentioned before can be duplicated with Foursquare and publicly available data.

In theory, we should conclude that population density and contagion rates are not fully correlated, and that bars, restaurants, and cafes are more likely to be related with higher contagion rates. Using K-Means clustering (more on that later in the methodology) we could also see how different districts in Madrid do in terms of contagion rates based on the cluster they belong to.

2. Data acquisition and cleaning

2.1 Data sources

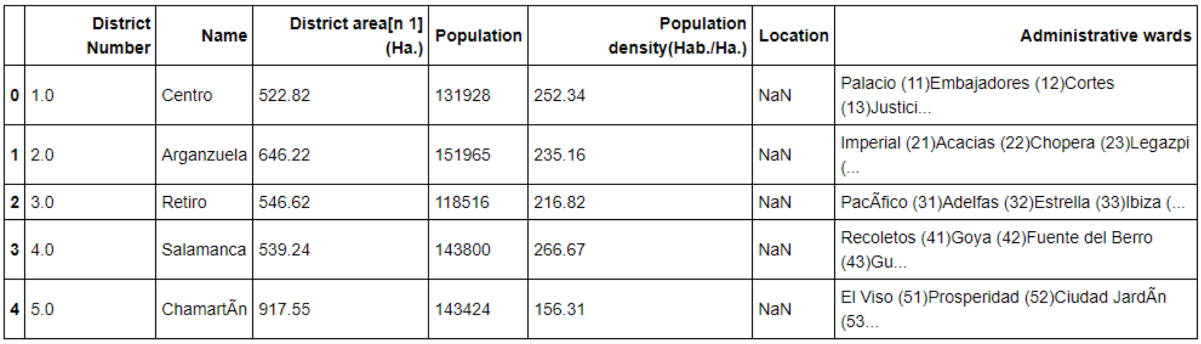
The following data is required for the project:

1) Neighbourhood (District) data of Madrid
2) Geographical coordinates of Madrid and all 21 districts in Madrid
3) Covid-19 contagion data of Madrid
4) Venue data for Madrid’s districts
5) A geojson map of Madrid to create a choropleth map

The data of the Neighbourhood (District) data of Madrid was scraped from (https://es.wikipedia.org/wiki/Anexo:Distritos_de_Madrid). The data is read into a pandas data frame using the read_html() method. The main reason for doing so is that the Wikipedia page provides a comprehensive and detailed table of the data which can easily be scraped using the read_html() method of pandas. In fact, ‘Wikipedia’ is a Python library that makes it easy to access and parse data from Wikipedia.

Search Wikipedia, get article summaries, get data like links and images from a page, and more. Wikipedia wraps the MediaWiki API so you can focus on using Wikipedia data, not getting it. The top 5 rows of the dataframe are shown in Figure 1. Covid-19 contagion rate of the las 14 days for Madrid was taken from this website. (https://madrid.maps.arcgis.com/apps/opsdashboard/index.html#/7965c30d54f94d9cbd804c7b8ab3a40a (As of 05/12/2020).

Figure 1. Raw dataframe from Wikipedia

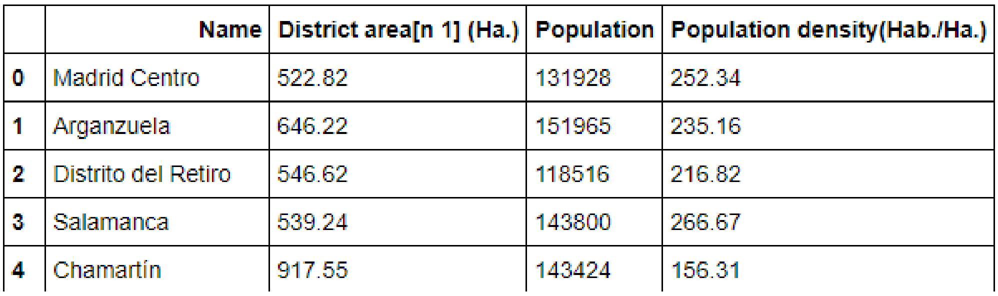


	District Number	Name	District area[n 1] (Ha.)	Population	Population density(Hab./Ha.)	Location	Administrative wards
0	1.0	Centro	522.82	131928	252.34	NaN	Palacio (11)Embajadores (12)Cortes (13)Justici...
1	2.0	Arganzuela	646.22	151965	235.16	NaN	Imperial (21)Acacias (22)Chopera (23)Legazpi (...
2	3.0	Retiro	546.62	118516	216.82	NaN	PacÃfico (31)Adelfas (32)Estrella (33)Ibiza (...
3	4.0	Salamanca	539.24	143800	266.67	NaN	Recoletos (41)Goya (42)Fuente del Berro (43)Gu...
4	5.0	ChamartÃn	917.55	143424	156.31	NaN	El Viso (51)Prosperidad (52)Ciudad JardÃn (53...

2.2 Data cleaning

Drop useless columns, “District Number","Location","Administrative wards". Change the names that are wrong, this will be useful to find them with the geocoder later. See Figure2.

Figure 2. Cleaned table

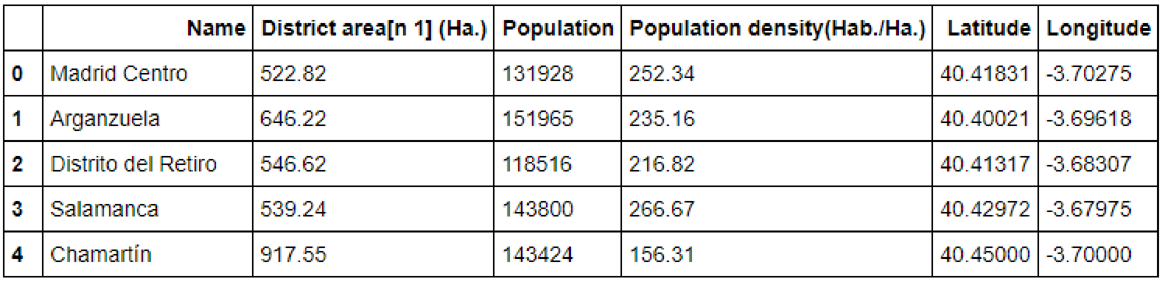


	Name	District area[n 1] (Ha.)	Population	Population density(Hab./Ha.)
0	Madrid Centro	522.82	131928	252.34
1	Arganzuela	646.22	151965	235.16
2	Distrito del Retiro	546.62	118516	216.82
3	Salamanca	539.24	143800	266.67
4	Chamartín	917.55	143424	156.31

The geographical coordinates for Madrid have been obtained from the GeoPy library in python. In order to define an instance of the geocoder, we need to define a user_agent. We will name our agent Madrid_explorer. This data is relevant for plotting the map of Madrid using the Folium library in python.

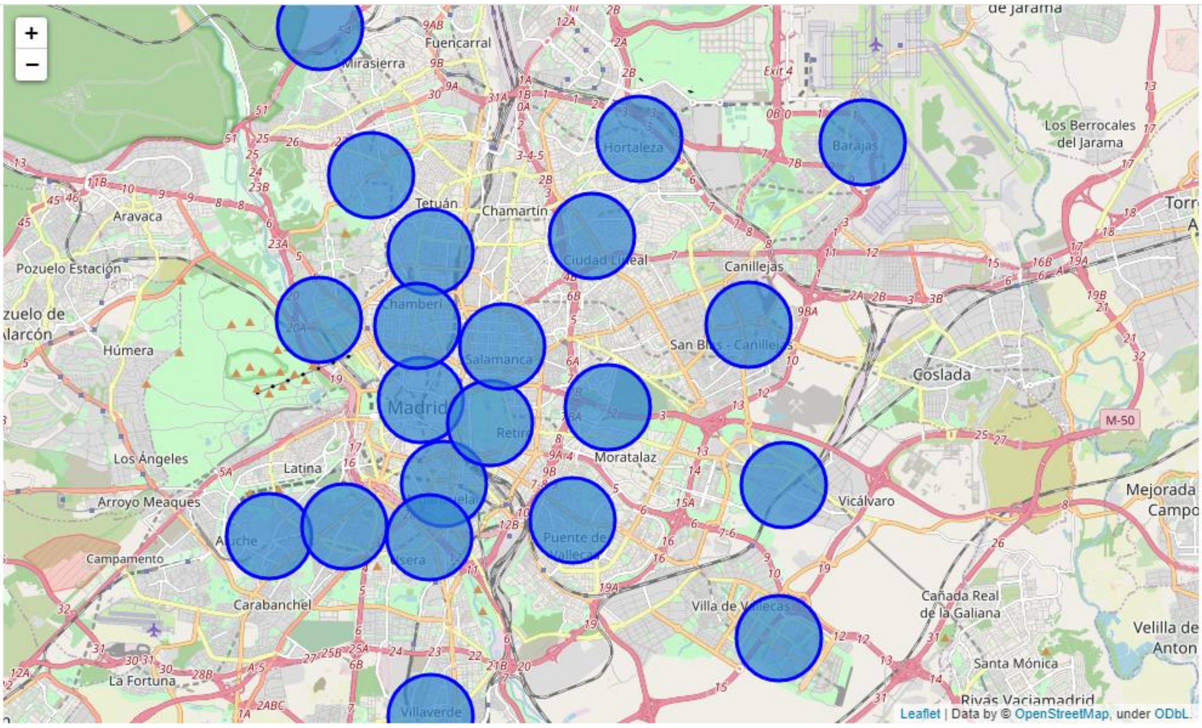
The geocoder library in python has been used to obtain latitude and longitude data for the 21 districts in Madrid. Figure 3 shows the coordinates of the districts obtained from. geocoder as ‘Latitude’ and ‘Longitude’. Then we created a map using the coordinates as shown in Figure 4, to make sure our data is correct.

Figure 3. Table with latitude and longitude



	Name	District area[n 1] (Ha.)	Population	Population density(Hab./Ha.)	Latitude	Longitude
0	Madrid Centro	522.82	131928	252.34	40.41831	-3.70275
1	Arganzuela	646.22	151965	235.16	40.40021	-3.69618
2	Distrito del Retiro	546.62	118516	216.82	40.41317	-3.68307
3	Salamanca	539.24	143800	266.67	40.42972	-3.67975
4	Chamartín	917.55	143424	156.31	40.45000	-3.70000

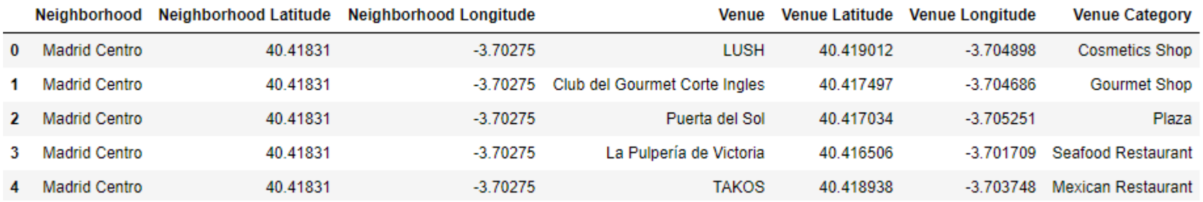
Figure 4. District map



2.3 Acquiring venue data

The venue data has been extracted using the Foursquare API. This data contains venue details for all districts in Madrid and is used to study the popular venues of different neighbourhoods as well as build the unsupervised learning model to cluster neighbourhoods. The venue recommendations of all districts were obtained with a limit of 300, that is, maximum of 300 venue recommendations per neighbourhood and a radius of 2 km around the districts' geographical coordinates. Figure 5 shows the top 5 rows depicting the results obtained after cleaning the data from Foursquare API.

Figure 5. Venues



	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Madrid Centro	40.41831	-3.70275	LUSH	40.419012	-3.704898	Cosmetics Shop
1	Madrid Centro	40.41831	-3.70275	Club del Gourmet Corte Ingles	40.417497	-3.704686	Gourmet Shop
2	Madrid Centro	40.41831	-3.70275	Puerta del Sol	40.417034	-3.705251	Plaza
3	Madrid Centro	40.41831	-3.70275	La Pulpería de Victoria	40.416506	-3.701709	Seafood Restaurant
4	Madrid Centro	40.41831	-3.70275	TAKOS	40.418938	-3.703748	Mexican Restaurant

4 Classification models

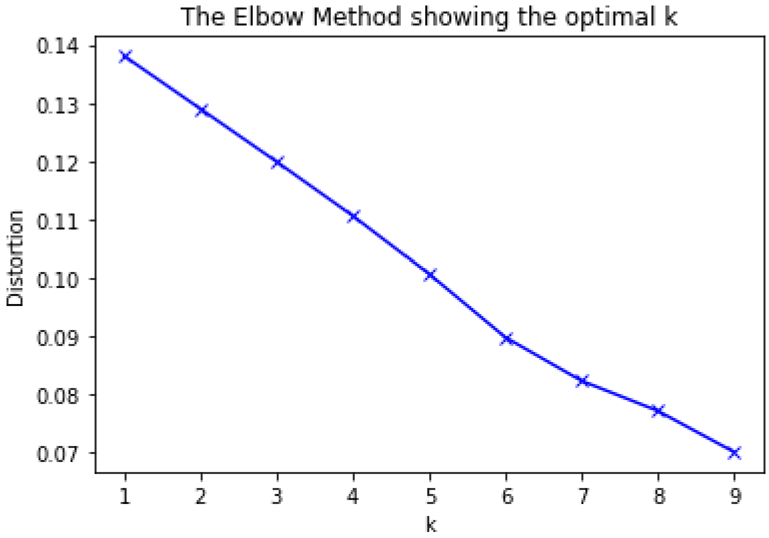
4.1 K-means clustering

To find clusters of venue types in the different city districts, we need to first transform the data frame with the restaurant venues, associated to city districts, by one-hot encoding (0/1). Now group rows by districts and by taking the mean of the frequency of occurrence of each category. Then, define a function that will put the most common venues by district in a dataframe. Create the new dataframe and display the top 10 venues for each neighbourhood as shown in figure

4.2 Unsupervised Learning

K-means unsupervised learning technique was used to cluster the districts based on the most common category in each venue. One important aspect of the k-means model is to determine the number of clusters to use in model development. We can determine the best number of cluster "k" using the "elbow" method. The elbow method runs k-means clustering on the dataset for a range of values for k (say from 1-10) and then for each value of k computes an average score for all clusters. By default, the distortion score is computed, the sum of square distances from each point to its assigned centre. The resulting number of clusters and their respective Silhouette scores are shown in Figure 6.

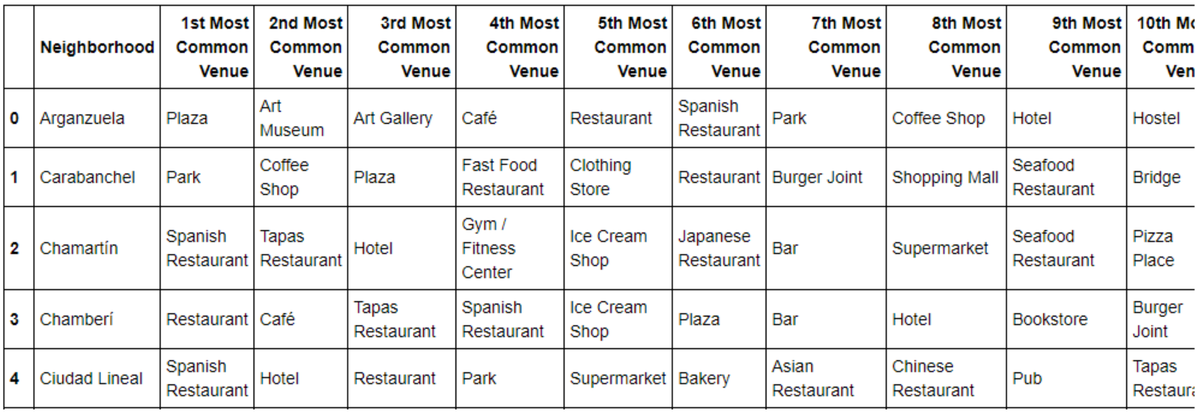
Figure 6. Elbow method for optimal k



4.3 Defining the clusters

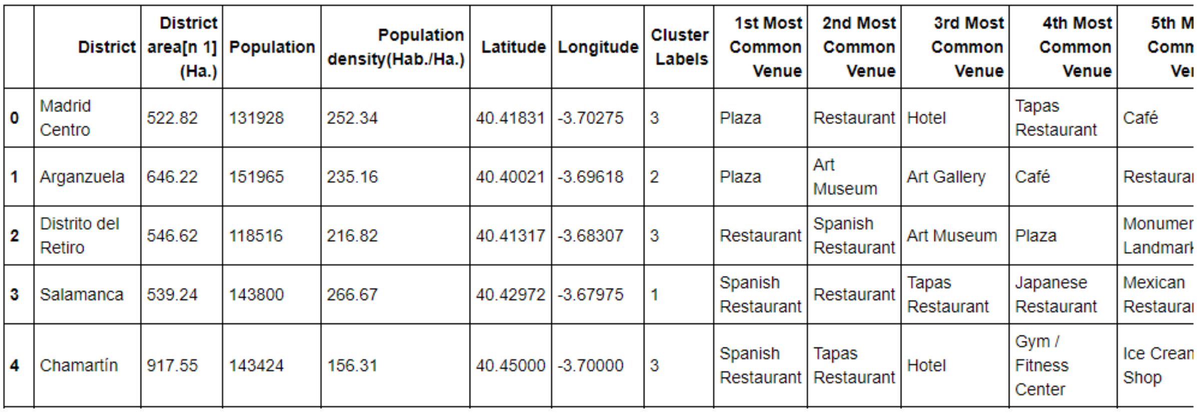
The clustering model then clusters the districts and provides a label for each cluster which is representative of the cluster it belongs to. The cluster labels were then added to the dataframe in Figure 7 along with the Location, Latitude, and Longitude columns to provide a complete summary of the clustering. The top 10 rows are shown in Figure 6. After that we can merge our new venues dataframe with our original dataframe, as shown in Figure 8. With this information, we are also ready to plot our clusters in a map, as shown in Figure 9.

Figure 7. Top 10 venues



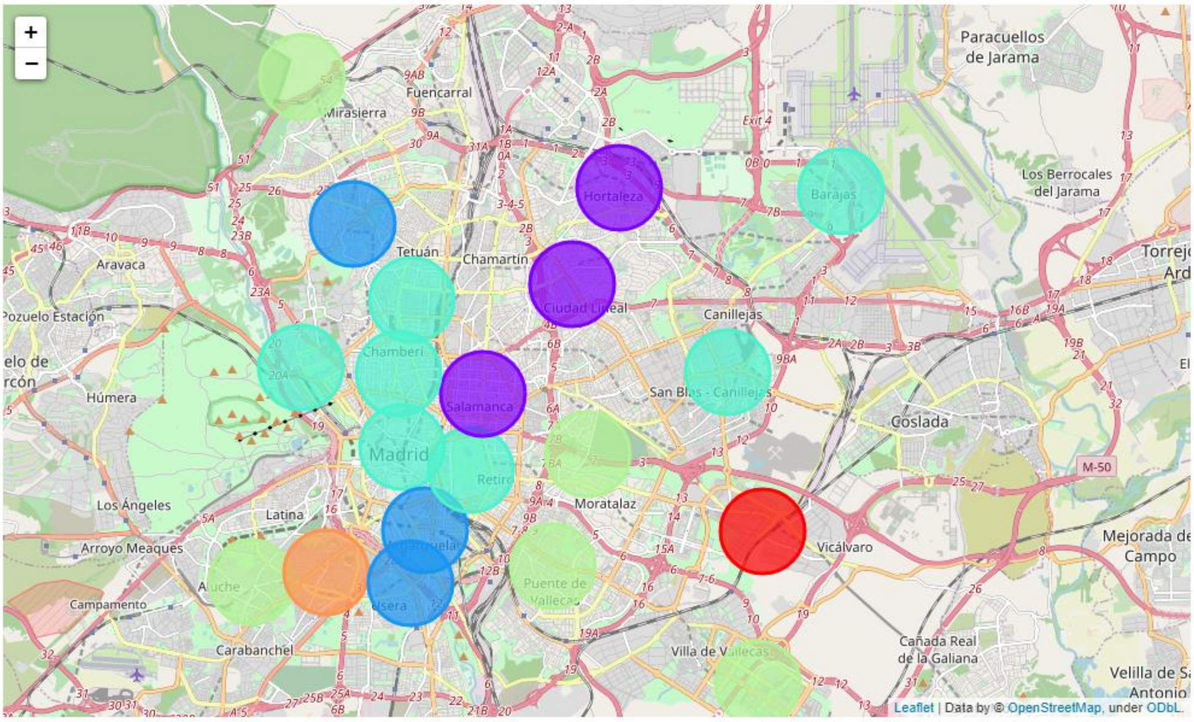
	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Mc Comm Ven
0	Arganzuela	Plaza	Art Museum	Art Gallery	Café	Restaurant	Spanish Restaurant	Park	Coffee Shop	Hotel	Hostel
1	Carabanchel	Park	Coffee Shop	Plaza	Fast Food Restaurant	Clothing Store	Restaurant	Burger Joint	Shopping Mall	Seafood Restaurant	Bridge
2	Chamartín	Spanish Restaurant	Tapas Restaurant	Hotel	Gym / Fitness Center	Ice Cream Shop	Japanese Restaurant	Bar	Supermarket	Seafood Restaurant	Pizza Place
3	Chamberí	Restaurant	Café	Tapas Restaurant	Spanish Restaurant	Ice Cream Shop	Plaza	Bar	Hotel	Bookstore	Burger Joint
4	Ciudad Lineal	Spanish Restaurant	Hotel	Restaurant	Park	Supermarket	Bakery	Asian Restaurant	Chinese Restaurant	Pub	Tapas Restaur

Figure 8. Merged Datafame with Cluster Labels and venues



	District	District area[n 1] (Ha.)	Population	Population density(Hab./Ha.)	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th M Comn Vei
0	Madrid Centro	522.82	131928	252.34	40.41831	-3.70275	3	Plaza	Restaurant	Hotel	Tapas Restaurant	Café
1	Arganzuela	646.22	151965	235.16	40.40021	-3.69618	2	Plaza	Art Museum	Art Gallery	Café	Restaurai
2	Distrito del Retiro	546.62	118516	216.82	40.41317	-3.68307	3	Restaurant	Spanish Restaurant	Art Museum	Plaza	Monumer Landmark
3	Salamanca	539.24	143800	266.67	40.42972	-3.67975	1	Spanish Restaurant	Restaurant	Tapas Restaurant	Japanese Restaurant	Mexican Restaurai
4	Chamartín	917.55	143424	156.31	40.45000	-3.70000	3	Spanish Restaurant	Tapas Restaurant	Hotel	Gym / Fitness Center	Ice Crean Shop

Figure 9. Clusters by colour



4.4 Binning variables

We can bin a few variables to make our interpretation easier. Starting with density we binned the values into 5 distinct categories: Very Sparse, Sparse, Normal, Dense, Very Dense. Then we did the same with the clusters, dividing them into: Restaurants and traditional cuisine, Plazas and historic places, Parks and bars, Supermarkets and train stations, see Figure 10. Once we have created these, we can add them to our dataframe, as in figure 11.

Figure 10. Binning variables

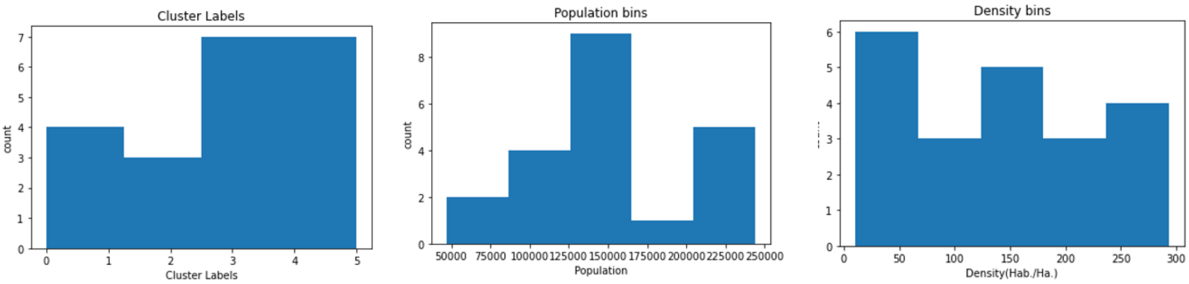
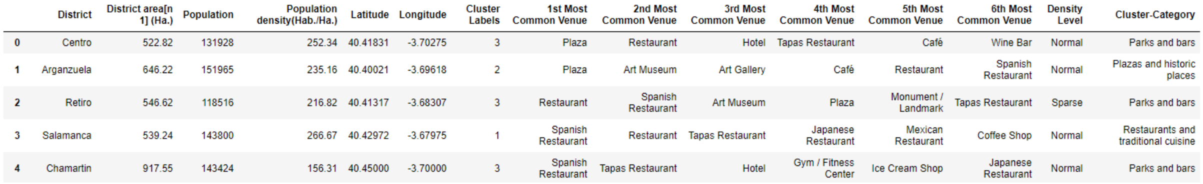


Figure 11. Dataframe with added labels “Density Level” and “Cluster Category”



	District	District area[n 1] (Ha.)	Population	Population density(Hab./Ha.)	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	Density Level	Cluster-Category
0	Centro	522.82	131928	252.34	40.41831	-3.70275	3	Plaza	Restaurant	Hotel	Tapas Restaurant	Café	Wine Bar	Normal	Parks and bars
1	Arganzuela	646.22	151965	235.16	40.40021	-3.69618	2	Plaza	Art Museum	Art Gallery	Café	Restaurant	Spanish Restaurant	Normal	Plazas and historic places
2	Retiro	546.62	118516	216.82	40.41317	-3.68307	3	Restaurant	Spanish Restaurant	Art Museum	Plaza	Monument / Landmark	Tapas Restaurant	Sparse	Parks and bars
3	Salamanca	539.24	143800	266.67	40.42972	-3.67975	1	Spanish Restaurant	Restaurant	Tapas Restaurant	Japanese Restaurant	Mexican Restaurant	Coffee Shop	Normal	Restaurants and traditional cuisine
4	Chamartin	917.55	143424	156.31	40.45000	-3.70000	3	Spanish Restaurant	Tapas Restaurant	Hotel	Gym / Fitness Center	Ice Cream Shop	Japanese Restaurant	Normal	Parks and bars

4.5 Examining the Clusters

With all this information, we can now start evaluating the clusters, and also create a choropleth map, see Figure 12, where we display the population density by district and the respective cluster, so we can see how they relate.

Cluster 0 – Is the largest cluster, districts in this cluster present large amounts of traditional food like Spanish restaurants and tapas restaurants and have also a good number of plazas and parks. In addition, the districts present normal to low levels of population density.

Cluster 1 – Districts in cluster 1 are defined by a high number of Spanish restaurants and moderate to dense levels of population density.

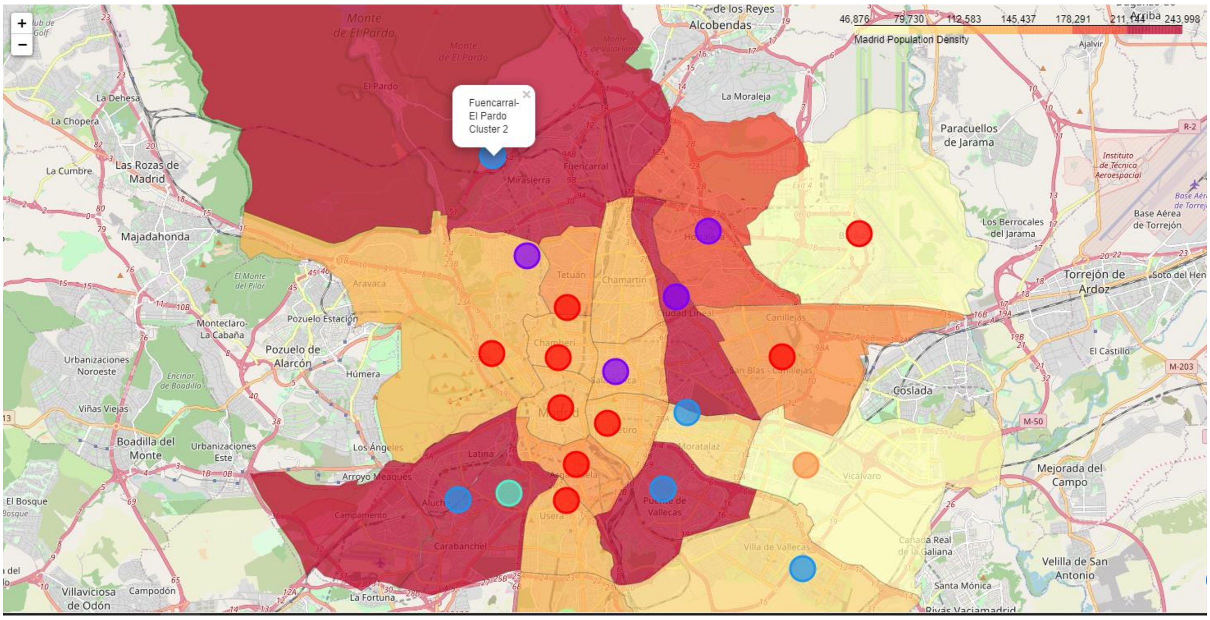
Cluster 2 – Districts in cluster 2 have a more mixed assortment of venues, ranging from cafes, bars, grocery stores and supermarkets, and are generally gave very dense levels of population.

Cluster 3 – Is a single district cluster, Carabanchel. It is a very dense residential area with parks and coffee shops as its main venues.

Cluster 4 – Is a single district cluster, Villaverde. It is a district with the most train stations and normal levels of population density.

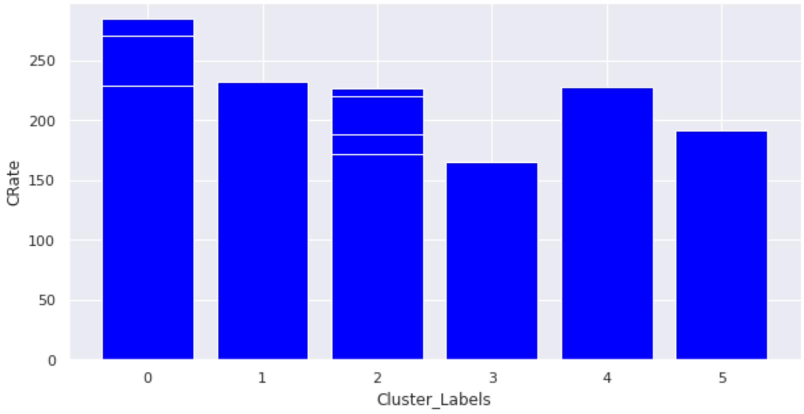
Cluster 5 – Is a single district cluster, Vicalvaro. It is a very sparse residential area defined by the number of exhibits it has.

Figure 12. Choropleth map with population density and venue clusters



Now that we have seen the characteristics of each cluster, we can see how they relate with the contagion rate, for this we can simply visualize it on a bar chart, see Figure 13. We can see that districts in cluster 0 and 1, which present a large number of restaurants, have the highest contagion rates. In comparison, cluster 3 and 5, with large numbers of open spaces such as parks and exhibits, have a comparatively lower contagion rate.

Figure 13. Bar chart CRate vs Cluster_Labels

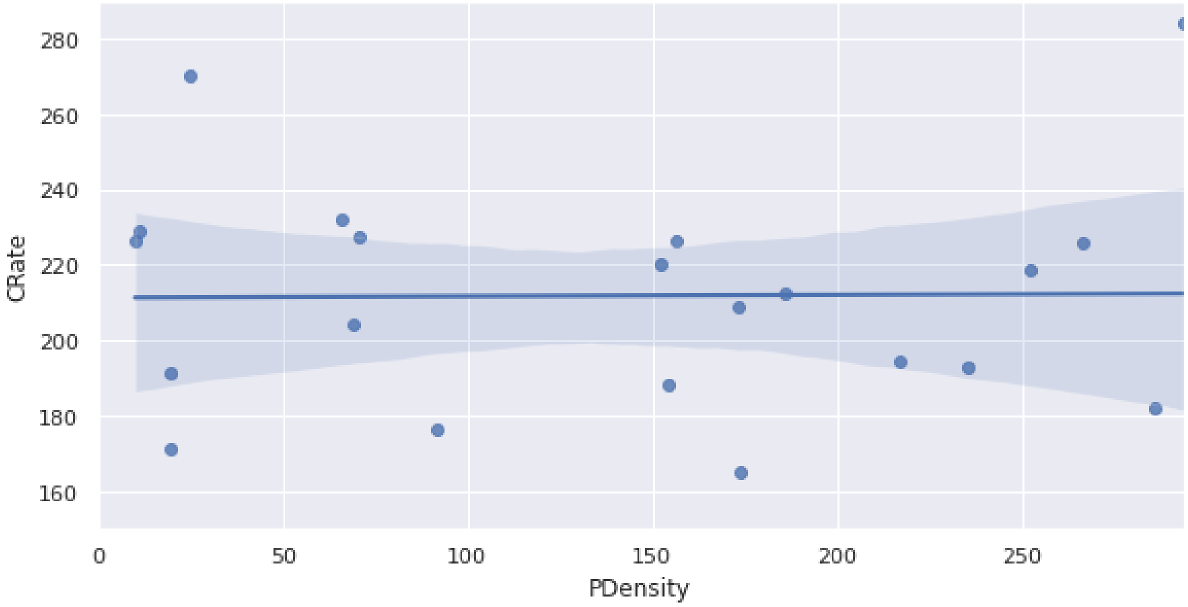


5. Predictive Modelling

5.1 Regression models

Finally, let us see how population density and contagion rate correlate in our dataset. We have the "Population density(Hab./Ha.)", so we will use it to do our analysis. Let's plot Population density(Hab./Ha.) vs Contagion rate last 14 days, to see how linear is their relation, see figure 14. The population density does not seem like a good predictor of the contagion rate at all since the regression line is close to horizontal. Also, the data points are very scattered and far from the fitted line, showing lots of variability. Therefore, it is it is not a reliable variable.

Figure 14. Population density (Hab./Ha.) vs Contagion rate last 14 days



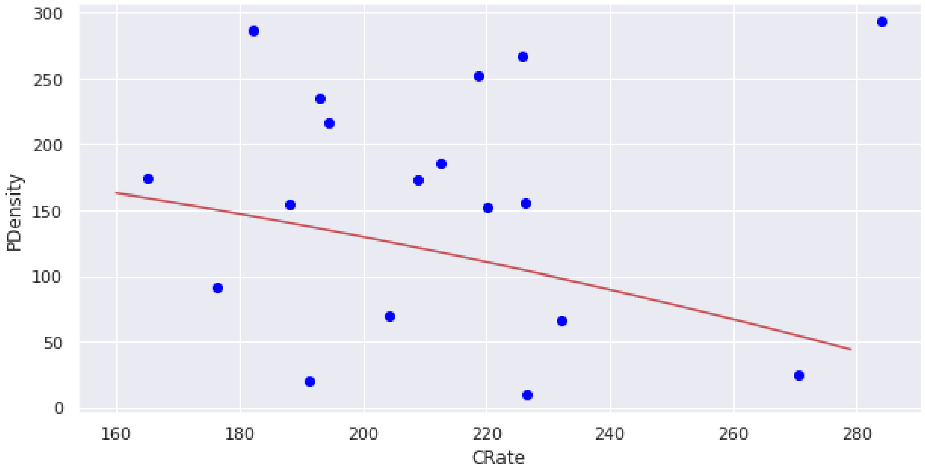
As we can see in Figure 14, and as stated at the beginning on our introduction, the population density and the contagion rate do not seem to be very linearly correlated. In fact, using the simple .corr() function. Pearson Correlation is the default method of the function. The Pearson Correlation Coefficient is 0.012475 with a P-value of P = 0.95719, so we can say that there is no evidence that the correlation is significant.

Sometimes, the trend of data is not really linear, so Pearson may not be the best way to evaluate it. In this case we can use Polynomial regression methods. Many different regressions exist that can be used to fit whatever the dataset looks like, such as quadratic, cubic, and so on, and it can go on and on to infinite degrees. In essence, we can call all of these, polynomial regression, where the relationship between the independent variable x and the dependent variable y is modelled as an nth degree polynomial in x.

We can split our dataset into train and test sets, 80% of the entire data for training, and the 20% for testing. We create a mask to select random rows using np.random.rand() function. To create a train and test dataset we can import from sklearn.preprocessing PolynomialFeatures and also import from sklearn the linear_model. With these we can split our dataset in train and test subsets.

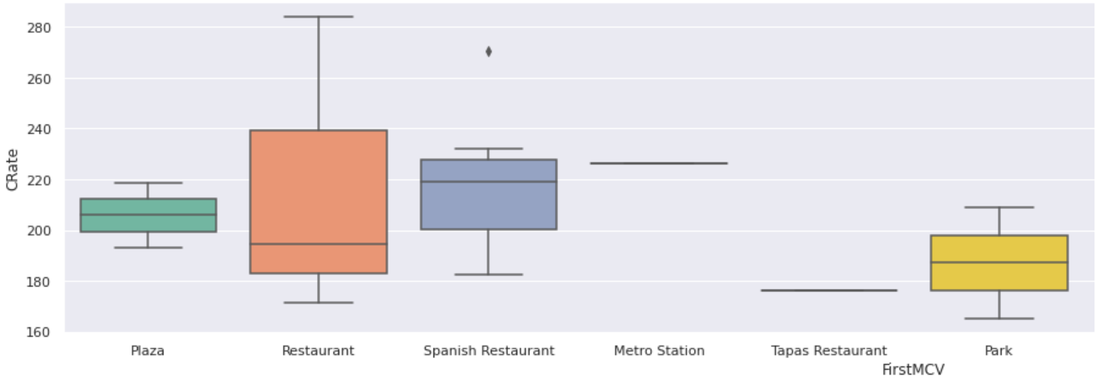
We tried several polynomial degrees to see which one fits the data best, ranging from 2 to 6, and decided that the one which yields the best results is degree 3, see Figure 15. In any case, the mean absolute error is 26.03, the residual sum of squares (MSE) is 954.51, and the R2-score: -1.05. The takeaway for a Negative R2 means we are doing worse than the mean value, so we have thoroughly discarded the correlation between those two variables with this method too.

Figure 15. Polynomial regression grade 3



As expected, we have discarded that contagion rate and population density are correlated, so we plot some BoxPlots to see how different venues relate to contagion rates. As we can see in Figure 16, places that have restaurants as the most common venue present the highest number of contagions rates.

Figure 16. Boxplot Crate vs FirstMCV



6. Conclusions

By using data collected in Foursquare and publicly available sources, we have been able to conduct a small project about analysing Madrid venue information by district and how it may relate to Covid-19 contagion rates. Even though the amount of available data from Foursquare for Madrid area is still quite limited, the results seem in line with the reality faced by the different districts and venues.

Before any conclusion can be drawn, we should acknowledge that Foursquare data is not all-encompassing as the highest number of venues recorded in their app are in the food services and restaurant industry, although that is serviceable or our analysis as these kinds of establishments are shown to be focal points for contagion. A limitation, for example, is that Foursquare data does not take into account a venue's size (and thus how that attracts more people). While the data is limited, it provides a glimpse into a city dynamic and when combined with other sources (e.g., contagion rate in our case) it provides more insightful results.

After using k-means clustering, we can conclude that districts in clusters that have more restaurants present higher contagion rates. A word of caution on k-means clustering is that it is an iterative method and may not reach the optional solution. The initial result of running this algorithm may not be the best possible outcome and rerunning it with different randomized starting centroids might provide a better performance. In our project, we run the k-means clustering several times, obtaining different clusters every time. Also, if you see the shape of the "elbow", that is, the number of ks in relation to the distortion, we can see that there is no apparent optimal k for this model.

We have also seen that the correlation between population density and contagion rates by district is weak or non-existent, and that the spread of Covid-19, as stated by government and scholars, is more related to where people gather. Our data shows that plazas, restaurants, and parks are the most common venues in each district and show more correlation with contagion rates than the population density of each district.