# TO LOCATE THE BEST LOCATION FOR A BUSINESS IN MADRID

TABLE OF CONTENTS:

# PROBLEM DESCRIPTION

In this assignment, I attempted to answer a question that has arrived in my personal life. I am Spaniard who lived in China and is married to a Colombian. My wife and I want to open a bar/restaurant in Madrid, however finding a location is hard. There are too many factors to consider and the foreign diversity does not make it easier. Thankfully, for this assignment I came across Madrid's foreign population and their numbers, which drove me to believe that I could use this assignment to help my wife.

The Coursera Data Science Professional Certificate has taught me to use various tools, such as data visualization techniques, data analysis and preprocessing, the use of Jupyter Notebooks and various advanced Machine Learning techniques. For this project, I have compiled everything I have learned in order to help us during the decision-making process.

# DATA PRESENTATION

I used three data sources to complete this project, namely:

a) **The official foreign population statistics offered by Madrid's City Hall website**: You will find the excel document in my GitHub 'CAPSTONE' folder. This information is critical, not only because we got the names of all the 'neighborhood' we are investigating, but because before we open our restaurant, we must familiarize ourselves with the foreign population in that region. Even though its obvious, we cannot invest in a project destined for Latin people, including typical foods, music and culture-based inside jokes, if they are found halfway across the Autonomous Region.

b) **Google Maps**: Considering I was unable to find a document containing all the coordinated I needed, I had to pull them from Google maps. Every coordinate is an exact replica of the information in their database.

c) **Foursquare**: This tool has been made available for all students (not exclusively). From this database, I pulled all the pertinent metadata of each venue in a 500m radius and filtered it to fit my needs. For example, venue location (Lat, Lng), type of venue, name, etc…

# METHODOLOGY

The first section of the code is divided into two main data frames, the foreign population excel file offered my Madrid's City Hall (1) and finding the coordinates of each 'Neighborhood' (2). The latter was easy enough, it consists of making a simple data frame and plotting every value manually. The former on the other hand, needed a little polishing. I mainly deleted the first few rows as they were irrelevant, changes the first three column names to suit the data, and dropped any NA values. Both data frames ended up like this:

| | Nationality | Madrid City | Centro | Arganzuela | Retiro | Salamanca | Chamartin | Tetuan | Chamberi | Fuencarral-El Pardo | Moncloa-Aravaca | Latina | Carabanchel | Usera |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Rumanía | 45036 | 815 | 754 | 480 | 753 | 680 | 1466 | 597 | 1830 | 991 | 4904 | 5873 | 2241 |
| 1 | China | 37276 | 1508 | 1356 | 564 | 755 | 652 | 1988 | 816 | 1733 | 960 | 2554 | 4396 | 9207 |
| 2 | Ecuador | 23953 | 647 | 741 | 265 | 619 | 380 | 1395 | 453 | 632 | 387 | 2194 | 3674 | 1806 |
| 3 | Venezuela | 23359 | 1563 | 913 | 638 | 1564 | 933 | 1310 | 794 | 1428 | 630 | 1448 | 1870 | 875 |
| 4 | Colombia | 22618 | 996 | 717 | 483 | 803 | 551 | 822 | 659 | 999 | 454 | 1786 | 3395 | 1752 |
| 5 | Marruecos | 21909 | 1101 | 390 | 184 | 322 | 280 | 1393 | 320 | 930 | 342 | 1539 | 2223 | 942 |
| 6 | Italia | 20308 | 3030 | 1219 | 840 | 1817 | 1060 | 1194 | 1640 | 1195 | 710 | 826 | 915 | 412 |
| 7 | Perú | 18829 | 563 | 521 | 253 | 612 | 419 | 965 | 567 | 805 | 368 | 2026 | 2425 | 1131 |
| 8 | Paraguay | 18682 | 364 | 474 | 237 | 521 | 657 | 3311 | 584 | 1024 | 636 | 2061 | 2152 | 727 |
| 9 | República Dominicana | 17511 | 365 | 654 | 204 | 344 | 322 | 2272 | 443 | 589 | 536 | 1501 | 1607 | 1202 |
| 10 | Honduras | 15981 | 149 | 228 | 232 | 332 | 337 | 755 | 317 | 863 | 335 | 2021 | 2870 | 1115 |
| 11 | Bolivia | 14930 | 284 | 407 | 182 | 342 | 315 | 576 | 280 | 401 | 225 | 1458 | 2625 | 2827 |
| 12 | Filipinas | 12628 | 1344 | 640 | 142 | 578 | 661 | 4473 | 771 | 442 | 568 | 629 | 400 | 225 |
| 13 | Portugal | 9860 | 769 | 372 | 262 | 695 | 534 | 590 | 509 | 693 | 365 | 533 | 658 | 283 |
| 14 | Francia | 9561 | 1608 | 455 | 370 | 968 | 554 | 387 | 699 | 366 | 347 | 196 | 188 | 66 |
| 15 | Ucrania | 9453 | 152 | 214 | 133 | 220 | 176 | 221 | 149 | 312 | 168 | 1745 | 1251 | 428 |
| 16 | Brasil | 9324 | 677 | 309 | 244 | 431 | 280 | 567 | 322 | 361 | 234 | 1159 | 1596 | 410 |
| 17 | Bulgaria | 7842 | 262 | 137 | 115 | 113 | 123 | 245 | 74 | 316 | 170 | 996 | 988 | 395 |
| 18 | Estados Unidos de América | 6791 | 1637 | 406 | 385 | 749 | 389 | 300 | 657 | 297 | 426 | 207 | 125 | 86 |
| 19 | Reino Unido | 5915 | 1274 | 324 | 256 | 550 | 466 | 329 | 501 | 313 | 304 | 211 | 170 | 81 |
| 20 | Bangladesh | 5886 | 2742 | 381 | 33 | 32 | 21 | 210 | 48 | 27 | 14 | 257 | 410 | 175 |
| 21 | Cuba | 5725 | 435 | 243 | 111 | 169 | 161 | 257 | 175 | 305 | 133 | 395 | 650 | 301 |
| 22 | Polonia | 5487 | 246 | 173 | 117 | 138 | 130 | 220 | 151 | 184 | 133 | 801 | 706 | 296 |
| 23 | Argentina | 5061 | 588 | 262 | 186 | 323 | 206 | 271 | 291 | 261 | 188 | 284 | 426 | 230 |
| 24 | Alemania | 4707 | 592 | 199 | 175 | 494 | 442 | 245 | 367 | 607 | 247 | 140 | 111 | 60 |
| 25 | Nicaragua | 4697 | 82 | 107 | 101 | 174 | 217 | 248 | 111 | 215 | 137 | 430 | 556 | 291 |
| 26 | México | 4503 | 644 | 213 | 173 | 656 | 263 | 266 | 359 | 263 | 184 | 205 | 199 | 89 |
| 27 | El Salvador | 3228 | 71 | 58 | 82 | 82 | 91 | 135 | 81 | 115 | 72 | 226 | 413 | 205 |
| 28 | Rusia | 2522 | 178 | 117 | 89 | 215 | 138 | 132 | 113 | 127 | 102 | 186 | 131 | 92 |
| 29 | Chile | 2520 | 285 | 124 | 84 | 212 | 76 | 122 | 161 | 161 | 85 | 146 | 164 | 69 |
| 30 | Senegal | 2106 | 507 | 235 | 24 | 31 | 18 | 63 | 18 | 37 | 14 | 40 | 239 | 62 |
| 31 | India | 1704 | 138 | 67 | 40 | 126 | 142 | 136 | 72 | 71 | 31 | 102 | 74 | 74 |
| 32 | Países Bajos | 1615 | 218 | 77 | 90 | 134 | 103 | 78 | 127 | 68 | 61 | 47 | 89 | 48 |

(1a)

| Puente de Vallecas | Moratalaz | Ciudad Lineal | Hortaleza | Villaverde | Villa de Vallecas | Vicálvaro | San Blas-Canillejas | Barajas |
|---|---|---|---|---|---|---|---|---|
| 4784 | 1286 | 2888 | 1466 | 3646 | 3384 | 2606 | 2929 | 661 |
| 3802 | 564 | 1960 | 1104 | 1236 | 685 | 472 | 972 | 190 |
| 3290 | 491 | 2471 | 401 | 2017 | 496 | 439 | 1015 | 138 |
| 1829 | 480 | 1858 | 1434 | 909 | 762 | 321 | 1486 | 314 |
| 1733 | 482 | 1792 | 910 | 1618 | 740 | 384 | 1282 | 258 |
| 3437 | 258 | 1011 | 426 | 3372 | 1655 | 802 | 649 | 333 |
| 704 | 310 | 1258 | 1109 | 330 | 427 | 189 | 786 | 337 |
| 2079 | 668 | 1726 | 603 | 1280 | 564 | 338 | 810 | 106 |
| 1354 | 360 | 1619 | 583 | 870 | 217 | 199 | 581 | 151 |
| 1989 | 223 | 1581 | 359 | 1881 | 296 | 151 | 889 | 103 |
| 2483 | 281 | 1062 | 349 | 1040 | 387 | 165 | 596 | 62 |
| 1573 | 227 | 1086 | 479 | 737 | 222 | 159 | 454 | 71 |
| 418 | 76 | 590 | 300 | 140 | 25 | 24 | 152 | 30 |
| 597 | 173 | 612 | 727 | 302 | 262 | 168 | 565 | 191 |
| 138 | 85 | 599 | 1802 | 68 | 59 | 52 | 357 | 197 |
| 912 | 239 | 348 | 282 | 1649 | 256 | 152 | 390 | 56 |
| 604 | 201 | 421 | 388 | 417 | 223 | 81 | 306 | 91 |
| 760 | 285 | 780 | 283 | 516 | 261 | 269 | 662 | 92 |
| 139 | 232 | 252 | 206 | 54 | 51 | 33 | 101 | 57 |
| 142 | 59 | 285 | 319 | 61 | 49 | 27 | 120 | 74 |
| 649 | 23 | 179 | 27 | 489 | 57 | 32 | 78 | 2 |
| 513 | 118 | 483 | 248 | 362 | 175 | 88 | 325 | 78 |
| 424 | 88 | 266 | 194 | 506 | 164 | 273 | 213 | 62 |
| 319 | 74 | 317 | 270 | 162 | 119 | 50 | 157 | 75 |
| 93 | 39 | 240 | 327 | 39 | 46 | 25 | 130 | 89 |
| 601 | 212 | 523 | 186 | 203 | 69 | 58 | 147 | 29 |
| 133 | 38 | 217 | 244 | 60 | 77 | 19 | 158 | 43 |
| 575 | 102 | 264 | 113 | 244 | 39 | 43 | 189 | 28 |
| 166 | 50 | 141 | 195 | 120 | 67 | 33 | 93 | 37 |
| 172 | 80 | 187 | 112 | 90 | 46 | 25 | 105 | 14 |
| 270 | 11 | 52 | 38 | 248 | 86 | 31 | 69 | 13 |
| 93 | 14 | 142 | 97 | 99 | 19 | 6 | 146 | 15 |
| 74 | 21 | 99 | 117 | 41 | 22 | 11 | 55 | 35 |

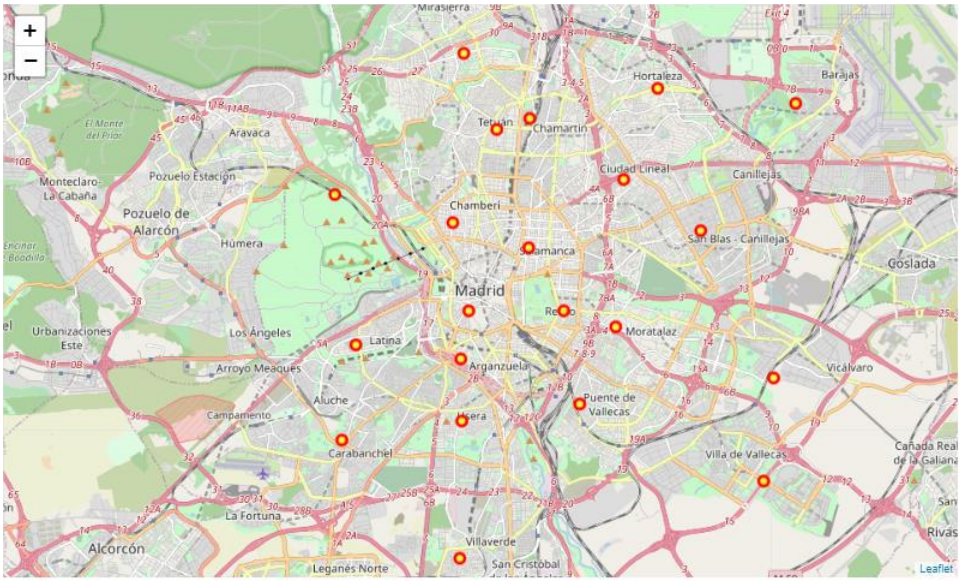
(1b)

Out[1014]:

| | Neighborhood | Latitude | Longitude |
|---|---|---|---|
| 0 | Arganzuela | 40.3989 | -3.7102 |
| 1 | Barajas | 40.4654 | -3.5952 |
| 2 | Carabanchel | 40.3778 | -3.7512 |
| 3 | Chamartin | 40.4615 | -3.6866 |
| 4 | Centro | 40.4115 | -3.7076 |
| 5 | Chamberí | 40.4344 | -3.7132 |
| 6 | Ciudad Lineal | 40.4457 | -3.6544 |
| 7 | Fuencarral-El Pardo | 40.4786 | -3.7092 |
| 8 | Latina | 40.4025 | -3.7465 |
| 9 | Hortaleza | 40.4694 | -3.6425 |
| 10 | Moratalaz | 40.4072 | -3.6570 |
| 11 | Moncloa-Aravaca | 40.4418 | -3.7537 |
| 12 | Puente de Vallecas | 40.3870 | -3.6695 |
| 13 | Retiro | 40.4113 | -3.6749 |
| 14 | Salamanca | 40.4279 | -3.6868 |
| 15 | San Blas-Canillejas | 40.4322 | -3.6279 |
| 16 | Tetuán | 40.4588 | -3.6978 |
| 17 | Usera | 40.3826 | -3.7099 |
| 18 | Villaverde | 40.3469 | -3.7108 |
| 19 | Villa de Vallecas | 40.3670 | -3.6061 |
| 20 | Vicálvaro | 40.3940 | -3.6029 |

(2)

I then had to make sure that the coordinate values were properly transformed into floats, so I plotted them in a map:

(3) 


Then came the second part of the data quest, finding the venues. This process could have been shorter, but I wanted to first pull the venue data from one region and then apply that as an iteration through the rest of the neighborhoods. I started with the first value alphabetically, Arganzuela.

```
close_venues.head()
```

Out[973]:

| | Name | Category | Latitude | Longitude |
|---|---|---|---|---|
| 0 | Puente Monumental del Parque de la Arganzuela | Bridge | 40.397671 | -3.711777 |
| 1 | La Gelateria di Angelo | Ice Cream Shop | 40.397951 | -3.707739 |
| 2 | Toboganes Parque de la Arganzuela | Playground | 40.398026 | -3.710241 |
| 3 | Madrid Río (Sector Central) | Park | 40.396744 | -3.712842 |
| 4 | Parque de la Arganzuela | Park | 40.398330 | -3.708686 |

[974]:
```
# How many venues did we find?
print('{} venues were located'.format(close_venues.shape[0]))
```

```
50 venues were located
```

(4)

Considering it worked perfectly, I itereated through all the regions, pulling data from all the venues in a 500m radius with a 90 venue cap per region.

(5)

```
In [977]: # Let's take a quick look
          print(Mad_Ven.shape)
          Mad_Ven
```

```
(563, 7)
```

Out[977]:

| | Neighborhood | Neighborhood Lat | Neighborhood Lon | Venue | Venue Lat | Venue Lon | Venue Category |
|---|---|---|---|---|---|---|---|
| 0 | Arganzuela | 40.3989 | -3.7102 | Puente Monumental del Parque de la Arganzuela | 40.397671 | -3.711777 | Bridge |
| 1 | Arganzuela | 40.3989 | -3.7102 | La Gelateria di Angelo | 40.397951 | -3.707739 | Ice Cream Shop |
| 2 | Arganzuela | 40.3989 | -3.7102 | Toboganes Parque de la Arganzuela | 40.398026 | -3.710241 | Playground |
| 3 | Arganzuela | 40.3989 | -3.7102 | Madrid Río (Sector Central) | 40.396744 | -3.712842 | Park |
| 4 | Arganzuela | 40.3989 | -3.7102 | Parque de la Arganzuela | 40.398330 | -3.708686 | Park |
| 5 | Arganzuela | 40.3989 | -3.7102 | Restaurante Peruano Mis Tradiciones | 40.399816 | -3.711022 | Peruvian Restaurant |
| 6 | Arganzuela | 40.3989 | -3.7102 | Le Crust Pizza Bar | 40.400922 | -3.709890 | Pizza Place |
| 7 | Arganzuela | 40.3989 | -3.7102 | Playa de Madrid | 40.396768 | -3.712833 | Beach |
| 8 | Arganzuela | 40.3989 | -3.7102 | Paseo de la Ribera del Manzanares | 40.396753 | -3.712838 | Park |

A top 10 list always made visualization easier, so I applied the same concept here.

(6)

Out[981]:

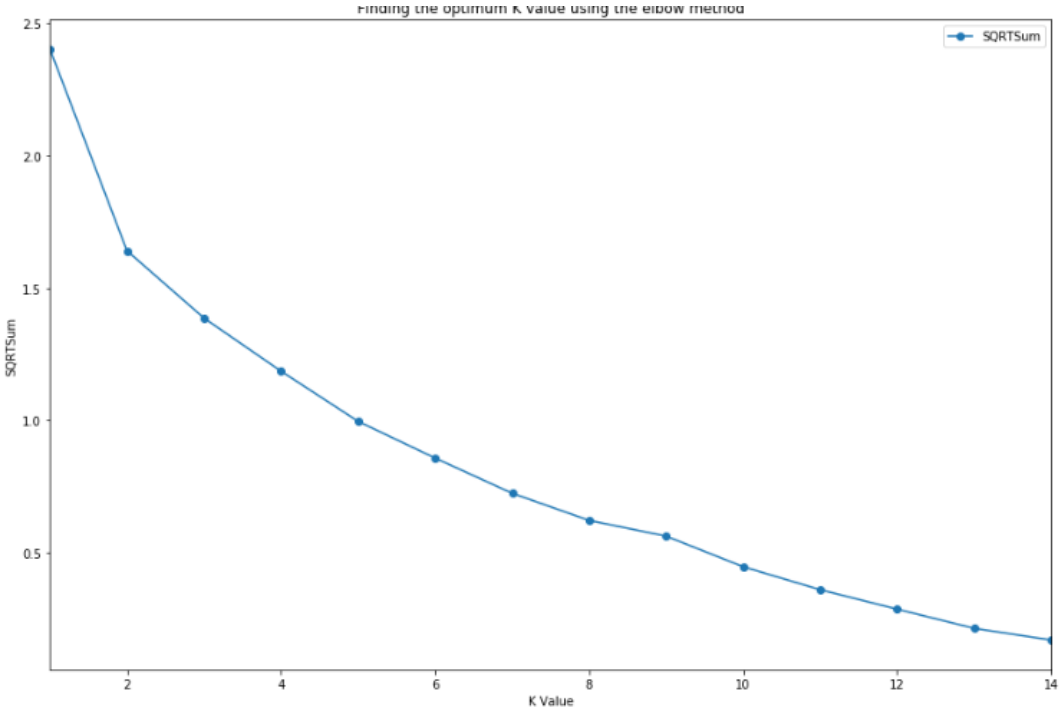
| | Neighborhood | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Arganzuela | Spanish Restaurant | Park | Bar | Grocery Store | Plaza | Playground | Beer Garden | Tapas Restaurant | Hotel | Restaurant |
| 1 | Barajas | Hostel | Coffee Shop | Food & Drink Shop | Café | Tapas Restaurant | Mexican Restaurant | Supermarket | Restaurant | Wine Shop | Dessert Shop |
| 2 | Carabanchel | Bar | Gym / Fitness Center | Gastropub | Mobile Phone Shop | Convenience Store | Cosmetics Shop | Creperie | Deli / Bodega | Department Store | Farmers Market |
| 3 | Centro | Tapas Restaurant | Spanish Restaurant | Bar | Hotel | Plaza | Vegetarian / Vegan Restaurant | Mediterranean Restaurant | Café | Hostel | Dessert Shop |
| 4 | Chamartin | Spanish Restaurant | Restaurant | Mediterranean Restaurant | Tapas Restaurant | Hotel | American Restaurant | Nightclub | Paella Restaurant | Salad Place | Pizza Place |

```
[982]: sorted_venues.shape
```

```
Out[982]: (21, 11)
```

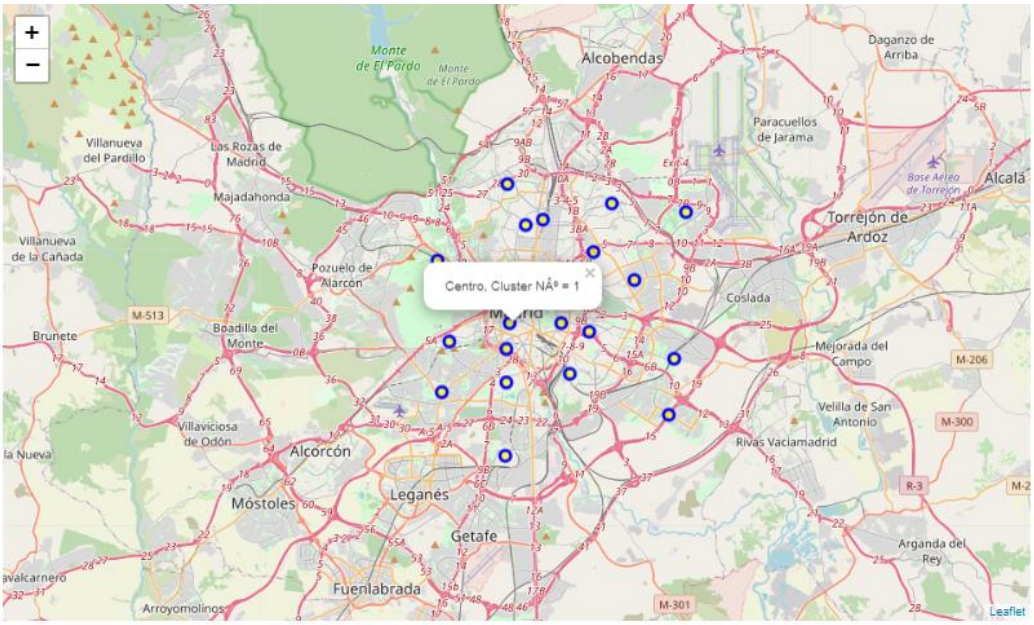
Here is where Jupyter Notebooks had a hiccup. While I was coding, the KMeans graph gave me an elbow value of 6. However, while I was checking the code, I added some notes, and after running the code again, the elbow value in the graph was 8. I decided to stick with 6; the value seemed more appropriate considering the low ratio between neighborhoods:foreign population.

(7)




I merged all my data together and performed a KMeans separation. Each neighborhood was assigned 1 of 6 cluster values according their correlation. In order to make visualiazation easier and more interactive, I added a map showing each neighborhood's name and which cluster it belongs to

(8)



# RESULTS

My main objective is to locate the best location for a Colombian restaurant in Madrid. I already mentioned this in my code, but I would like to reiterate it; I understand that Latin countries are not the same, each culture is different, their Spanish slang varies, and their history ranged from thousands of years back to only a couple hundred. The being said, I am going to make generic assumptions about these countries in order to find the ideal theme for my restaurant. To make it as attractive as possible, we need to DANCE! Because if there is a passion shared among all of these countries, it's the passion for music. This is a must! Also, ceviche… There's only so much fusion that can be done, but a good bachata always does the trick. Taking this into consideration, I searched my data for Latin American countries OR Spanish speaking countries.

CLUSTER 0:

Only North Americans were found in cluster 0, where breakfast spots, bars and Tennis courts were at the lead.

CLUSTER 1:

There's a larger variation of countries and regions in this cluster. We can mainly find Latin American countries, occupying 5/12 rows in this cluster. The top 3 most popular venues in each region are related to the food industry, such as: 'Spanish Restaurants', 'Tapas Restaurants', and 'Grocery Stores'.

CLUSTER 2:

The country density decreases again, however, it's not all bad new, Honduras is the only participant in this cluster, making this a very useful data set. Moreover, several of the top 10 venues in this region are related to food.

CLUSTER 3:

I found cluster 3 to be just as lucky, as Ecuador is the sole participant of this cluster. There's a lower density of food-related venues in this region, however, wcan find a couple of Bars and Pubs in the data.

CLUSTER 4:

The Philippines stood alone in cluster 4, nonetheless, 4/10 venues are directly related to the food industry, namely: 'Spanish Restaurant', 'Restaurant', 'Pizza Place', 'Supermarket', and 'Donut Shop'.

CLUSTER 5:

Cluster 5's data differed slightly from our objective. Nevertheless, we can still find several Restaurants and Grocery Stores leading the top 10 list.

# DISCUSSION

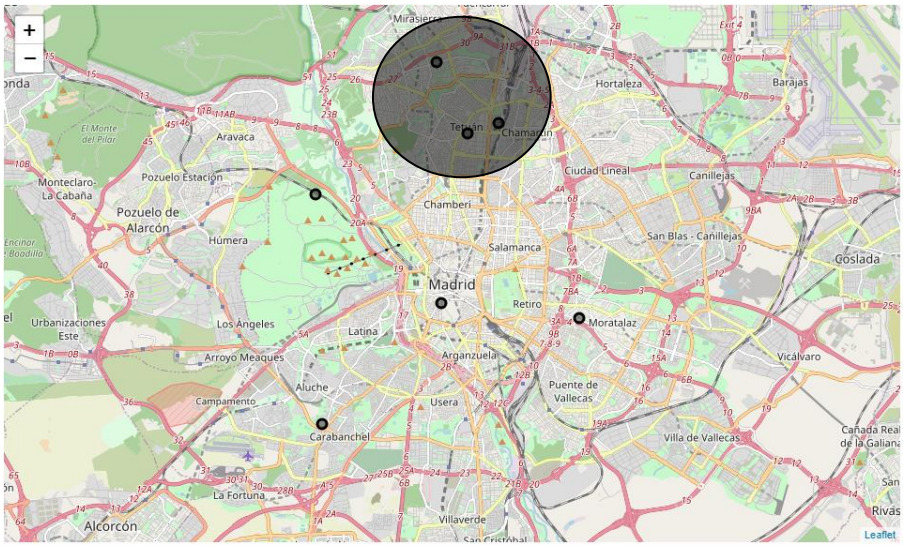
When analyzing they clusters, the density of what we are looking for is almost as important as the quality. For example, say there's a high density of Venezuelan and Colombian foreigners in Madrid Centro, but the most popular venues, given the location, are high end restaurants that are close the main touristic areas of Madrid.

Combining clusters and analyzing their content is a big part of the conclusion. Whereas Madrid is not small per se, their highway system and public transportation make it easy to move around the city. Given this, starting a restaurant next to CEBO Madrid Restaurant, located in Madrid Centro, would be a bad idea, despite the hypothetical high density of Venezuelan and Colombian population.

# CONCLUSION

By clustering clusters 1, 2, and 3, we get the ultimate trio. Most of these clusters have food-related venues as their top 5. In cluster 1, we find a high density of Latin countries, occupying 5/12 rows in the data frame. To make the conclusion easier to visualize, I made a map containing the Latin counties in Clusters 1, 2, and 3:

(9)




In the code, the map is interactive, and we can see how in the northern region we find foreigners from Peru, Colombia, and Brazil. Madrid's subway system offers fast and easy connectivity between Madrid Centro and Chamartín. Considering all of these factors simultaneously, I conclude that the best location of our restaurant is in the region of Chamartín. It's far enough from the Center, so it won't be shadowed by other venues with higher reputation, but it's close enough to the Center, allowing the subway system to do its magic. Moreover, because our main theme is based around Colombia, and Chamartín's main foreign population are Colombians, this creates an added value.