Comparing different cities using Foursquare API

Héctor Aristizábal

April 29th, 2020

Introduction

The problem to be addressed in this project has been thought as a hypothetical case of someone that has a job offer in 2 different cities. Let's, call him John. John lives in the city of Toronto and works as a data scientist and got a job offer from the same company in two cities, Bogota Colombia and Berlin, Germany. John has live his entire life in Toronto and wonders how similar these cities are. Also, he would like to know which neighborhoods are similar to the one he is living in, and maybe other Neighborhoods in his hometown.

Data acquisition and cleaning

Data sources

A Neighborhood list of the cities of Toronto, Bogota and Berlin is available online in Wikipedia and can be found in the following links:

Toronto

https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M

Bogota

https://en.wikipedia.org/wiki/Boroughs_and_neighborhoods_of_Berlin#Localities

Berlin

https://en.wikipedia.org/wiki/Boroughs_and_neighborhoods_of_Berlin#Localities

The places by Foursquare API is a database of more than 105 million places worldwide, and is going to be consulted for this project. To explore the cities, we will use the *Venue Recommendation* which returns a list of recommended venues near a certain location. In order to make the query in the Foursquare API we need the coordinates given in

Latitude and Longitude for a given Neighborhood. So the first thing me we need to do is to get coordinates for each Neighborhood in each of the cities.

To get the coordinates given a city name and a Neighborhood we use ArcGIS geocoding services. Geocoder is a geocoding library very easy to use written in python that find locations of addresses, coordinates business names, and so on. The Arcgis geocoder documentation can be found in the following link:

https://geocoder.readthedocs.io/providers/ArcGIS.html

Cleansing

The data scraped from Wikipedia with the list of Neighborhoods for the different cities comes with different information for each city. So first, we get rid of all the information that is not of our interest. In order to consult nearby venues in each Neighborhood we only need the coordinates of every Neighborhood. So at the end we end up with a data frame like this:

	City	Borough	Neighborhood	Latitude	Longitude
0	Toronto	North York	Parkwoods	43.758872	-79.320292
1	Toronto	North York	Victoria Village	43.731540	-79.314280
2	Toronto	Downtown Toronto	Regent Park / Harbourfront	43.660690	-79.360310
3	Toronto	North York	Lawrence Manor / Lawrence Heights	43.723570	-79.437110
4	Toronto	Downtown Toronto	Queen's Park / Ontario Provincial Government	43.666630	-79.393268
...	...	...	...	...	...
306	Berlin	Reinickendorf	Waidmannslust	52.575450	13.349700
307	Berlin	Reinickendorf	Lübars	52.575450	13.349700
308	Berlin	Reinickendorf	Wittenau	52.575450	13.349700
309	Berlin	Reinickendorf	Märkisches Viertel	52.596800	13.358310
310	Berlin	Reinickendorf	Borsigwalde	52.575450	13.349700

311 rows × 5 columns

Figure 1. Dataframe used with the relevant information for this project

A total of 311 Neighborhoods for the 3 cities and the coordinates in latitude and longitude for each Neighborhood. In order to keep track of the city every Neighborhood belongs to we added an extra column in the data frame with the city.

This data frame is the input for the Foursquare API in order to start exploring the different Neighborhoods of these 3 cities. The results for each Neighborhood were limited to 100 Venues and to a radius of 700m from the coordinates given.

Methodology

The category of the places and businesses in each area will be used to characterize each Neighborhood and compare them. To achieve this we will use K-means clustering in order to group and categorize the Neighborhoods in the different cities.

With the results from the Foursquare API the first thing we do before starting the clustering of the Neighborhoods is to determine which categories of venues are the most common or popular, in order to take only the most important categories into account. After trying different numbers we decide to work with the 6 most popular categories for each Neighborhood. Changing the number after 6 the most common doesn't affect very much the results.

K-means clustering

In order to find the best number of clusters for our K means algorithm we run the algorithm with K=3, 4, 6 and 8 clusters and see the results to find the one that makes more sense to our categorization problem.

K=3

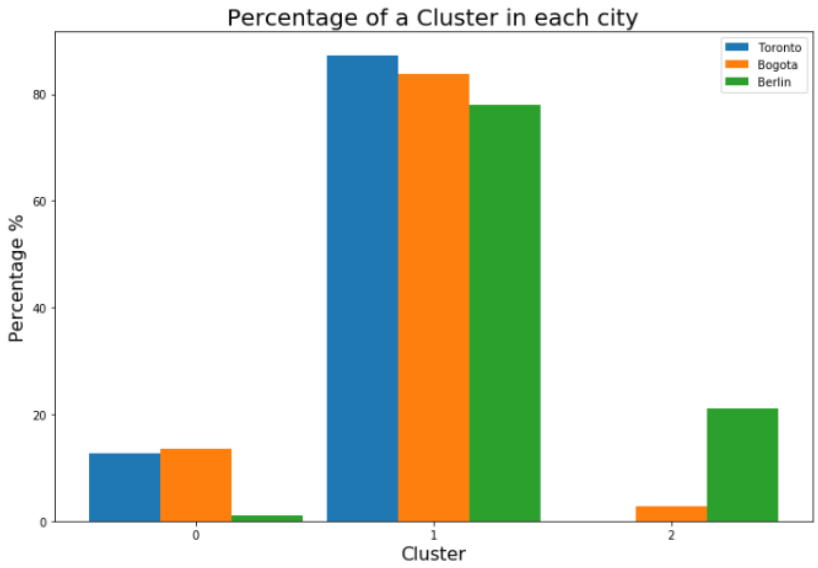

Figure 2. Percentage of a cluster for each city for K=3

In the bar chart above you can see the results of a K-means clustering with a number of clusters of K=3. Each bar represents the percentage of clusters for each city. Toronto been color blue, Bogota orange and Berlin color green. There is no much of a bigger difference between the cities for this case, been slightly more similar Bogota and Toronto. The chart below shows the results for K=4, showing no bigger difference for all cities. The Cluster 2 has almost no neighborhoods in any of the cities.

K=4

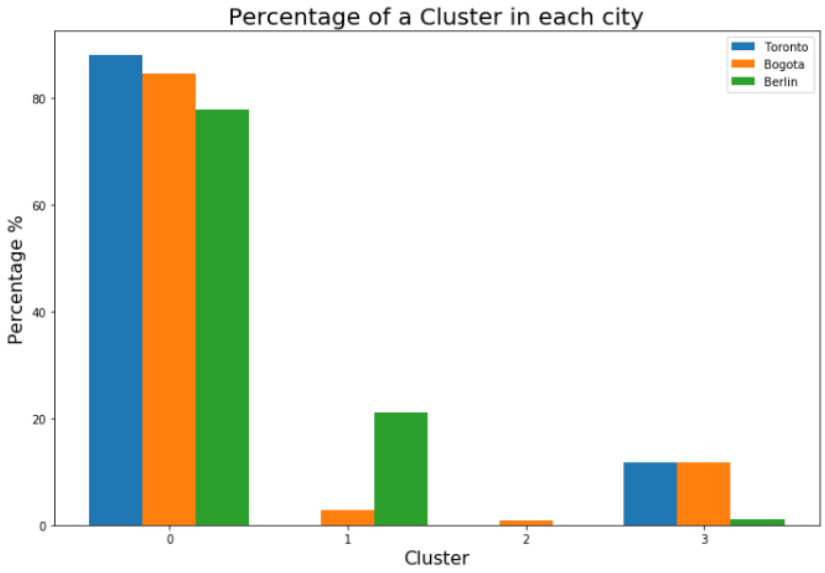

Figure 3. Percentage of a cluster for each city for K=4

For K=6 there is something very remarkable and is that the city of Bogota starts looking very different to the other 2 cities having most of its Neighborhood in labeled in cluster 4. There is also some cluster with almost no neighborhoods in any of the cities.

K=6

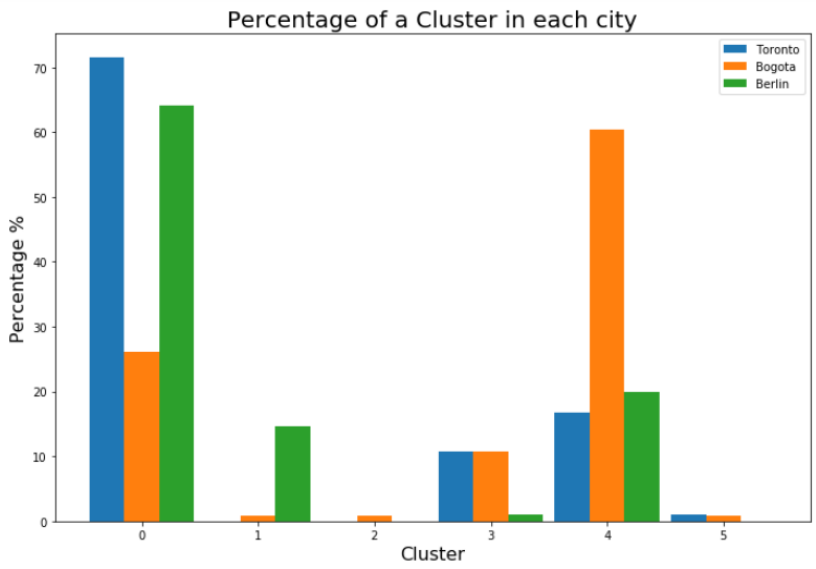

Figure 4. Percentage of a cluster for each city for K=6

In the case of K=8, the 3 cities look again quite similar, this time been berlin slightly different having 20% of its neighborhoods categorized in cluster 2.

K=8

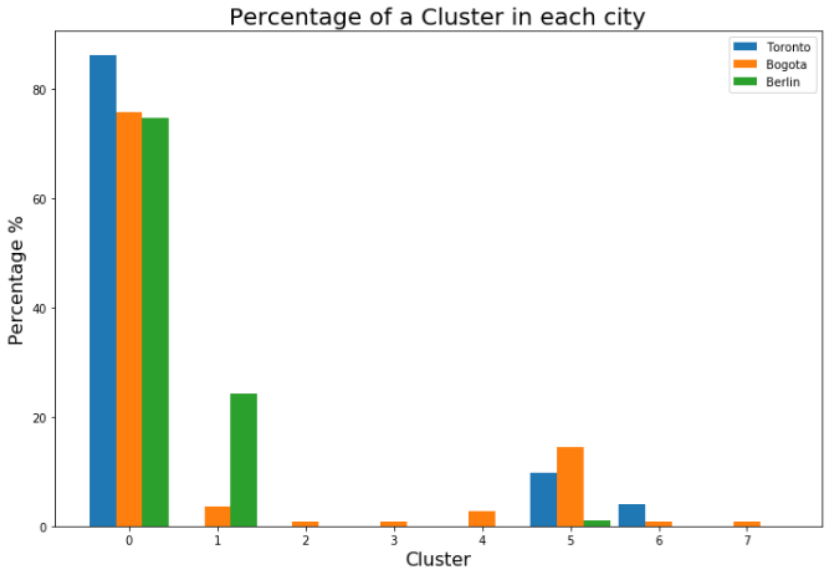

Figure 5. Percentage of a cluster for each city for K=8

In order to help John to visualize the results of the clustering we show the different categorization of the neighborhoods in a Folium map for K=6.

Toronto

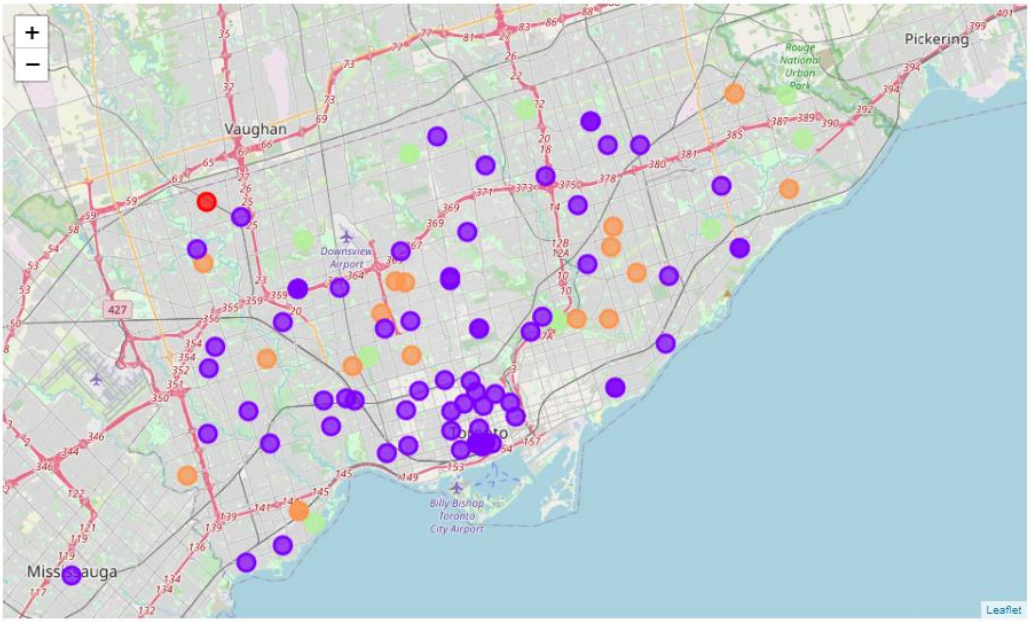

Figure 6. Map of the city of Toronto with clusters for K=6

Bogota

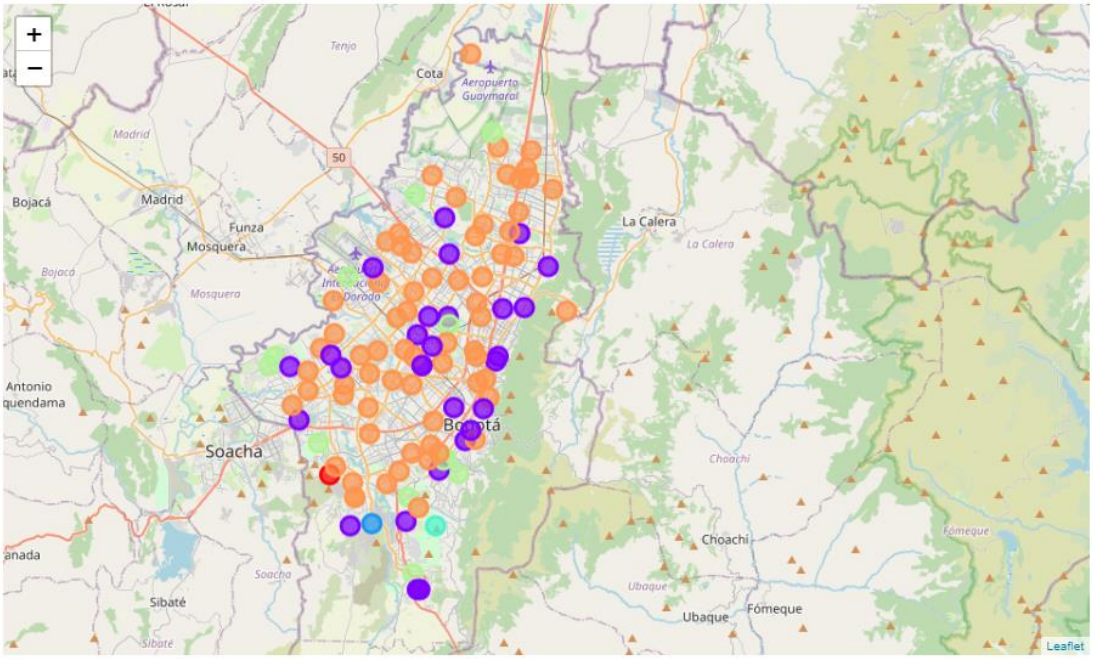

Figure 7. Map of the city of Bogota with clusters for K=6

Berlin

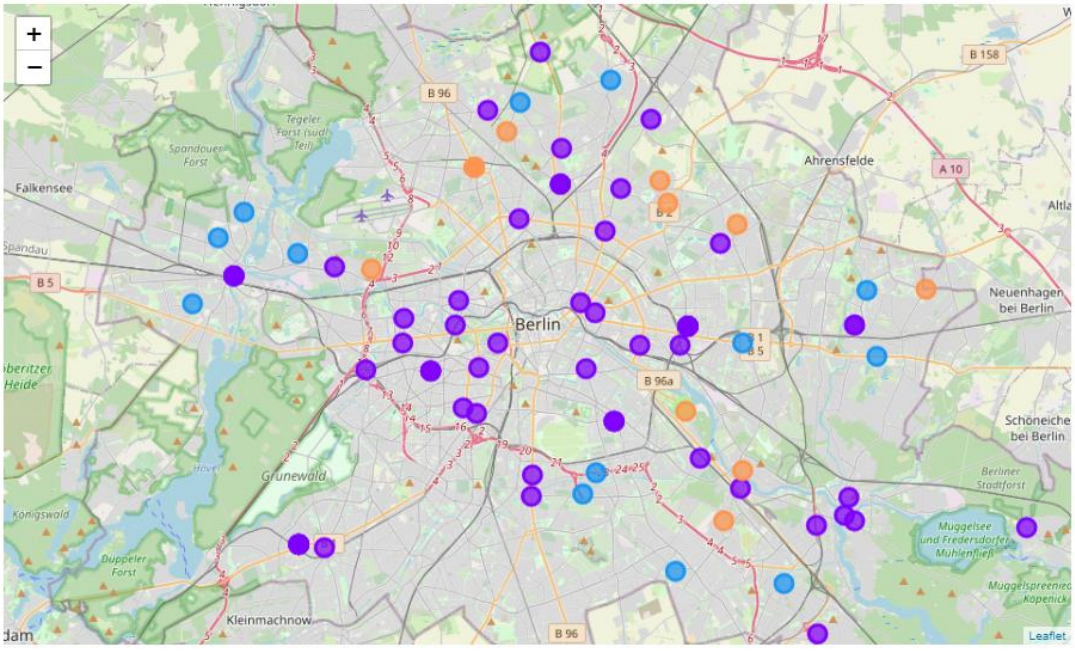

Figure 8. Map of the city of Berlin with clusters for K=6

From the maps you can tell that the city of Bogota has more orange clusters compared to Berlin and Toronto. In the city of Toronto and Berlin the orange label belongs to neighborhoods that are more in the suburbs of the city.

Conclusions

In this project 3 cities and its neighborhoods where studied based on the type of venues in the different neighborhoods. Using K-means clustering the neighborhoods where categorized according to the 7 most common category of venue.

Knowing the socioeconomical context of the different countries maybe it is worthy to take the results of the analysis using k=6 more into detail. It is probably more likely that this clustering tell us more about the neighborhoods of the 3 cities, taking into account that the Foursquare API delivers a lot less results in the case of the city of Bogota. It would be interesting to add some socioeconomic data to the analysis like average income or access to healthcare and other variables in order to make the clustering much more reliable.

After analyzing the clustering results obtained there is no much left to say that this is not an easy decision for John to make, giving that the results observed showed a lot of similarities between the cities. Based on what we have seen in the K-means clustering results John will probably find a good neighborhood to live in any of the cities analyzed.

References

- [1] Toronto neighborhoods – Wikipedia
- [2] Bogota neighborhoods – Wikipedia
- [3] Berlin neighborhoods – Wikipedia
- [4] Foursquare API