Segmenting and Clustering Popular Cities in Poland

# Introduction

## A.1. Description & Discussion of the Background

Currently, we are in a crisis; the COVID-19 epidemic. Every country takes its own countermeasures to avoid spreading and protect its people. Some countries even use a lockdown. In this case, you can only go outside for necessary causes, like groceries. Unfortunately, this means that travelling is not possible anymore. We need to wait until the epidemic is over.

Because this might take a while, we need to use our time to travel as effective as possible. Therefore, we will have to prioritize which city, we want to visit first. That is why, I want to analyze different cities and get to know how they differ. Questions I want to answer are; What are the differences and similarities of each city? What cities are more interesting for social activities? What cities are more interesting for visiting restaurants? And what for cultural activities?

In this particular case, we are going to investigate four cities in Poland; Łódź, Warsaw, Kraków and Poznań. I have chosen those cities because I am myself in Łódź for an internship. And after the epidemic, I want to travel to other cities. Of course, this program could be used for other cities as well. Therefore, it might be helpful for other people interested in travelling.

## A.2. Data Description

To answer the questions, we need the following applications:

* *Google Maps;*
  + To get the center coordinates of each city.
* *Foursquare API*;
  + To get the venues of each city.
* *Python*;
  + To generate the location points in each city, request the venues, get the most common venues, cluster them and show them on a map.

So the data that is involved consists of:

* Coordinates (longitude and latitude) of each city and its location points.
* The venues of each location point.

# Methodology

We want to have the data of multiple venues for each city. Unfortunately, the Foursquare API allows to get a maximum of only 100 venues per location. Therefore, we need to create multiple location points within the city.

Before we can do this, we need to determine at what radius the maximum of 100 venues is returned. Based on this information, we can segment the city and create multiple location points. We will use the center coordinates retrieved from *Google Maps*.

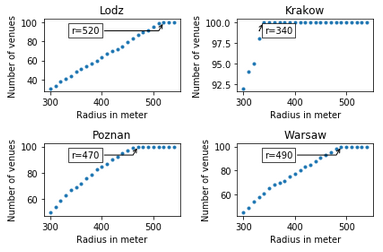
In order to be a bit more certain that we do not exceed the limit of 100 venues per location point, we use the a radius of 20 meters smaller than the maximum value. This results into a maximum radius of 500 meters for Łódź, 320 meters for Krakow, 450 meters for Poznan and 470 meters for Warsaw.

Figure : The radii with the maximum amount of venues for each city

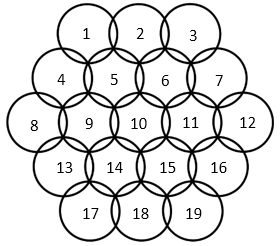
With those radii, we can create 19 location points in a hexagonal lattice. This lattice is the most effective way to cover a bigger circle (Conway & Sloane, 1999), as shown in figure 2.

Figure 2: 19-circle hexagonal lattice

We will use a function to create those location points. Because the earth is a sphere, we need to convert the radii in meters to degrees (for latitude/longitude). Based on the coordinates of the cities, we can estimate how much 1 degree of latitude/longitude is. This gives 111 kilometers for each degree of latitude and 80 kilometers for the longitude (Rosenberg, 2020). So 1 meter of latitude is degrees and 1 meter of latitude is degrees.

The first few location points with the city name and coordinates are shown in figure 3.

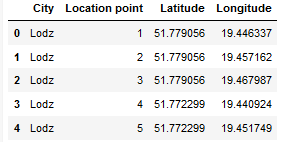


Figure : The head of the location points dataframe

Now we can use the *Python Folium* library to visualize the geographic details of each location point. We superimpose them on top of the map of Poland. The results for Łódź are shown in figure 4.

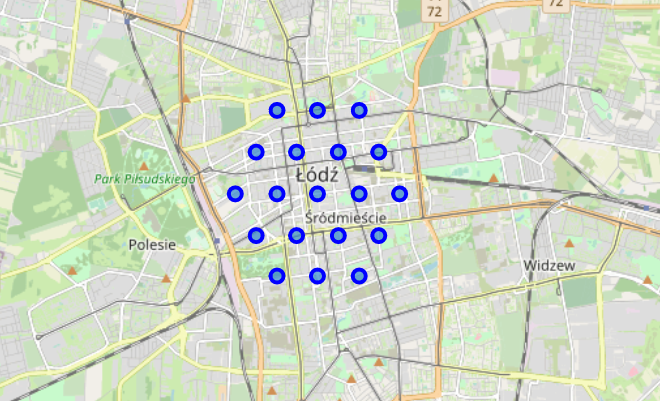


Figure : The location points of Łódź

The hexagonal lattice is clearly visible and the location points are numbered the same way as in figure 2.

Now we can get the data of the nearby venues using the *Foursquare API*. The head of the venues dataframe is shown in figure 5.

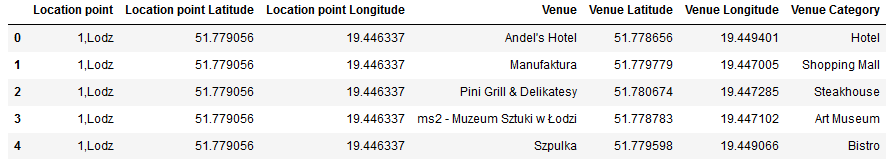


Figure : The venues dataframe

For all 76 location points, this gives a total amount of venues of 3217. Therefore, a lot of the location points did not get to the maximum of 100 venues. So let’s examine the amount of venues for each location point. The head of this dataframe is shown in figure 6.

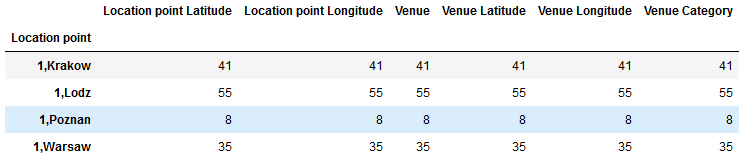


Figure : The amount of venues for each location point

Figure 6 clearly shows that not each location point reaches the maximum amount of 100 venues. The first location point in Poznan only got eight venues. This will not result into useful clusters, we need to clean some of the location points out of it.

When we set a threshold or 20 venues or more, there are 52 location points left. In total, those location points contain 2969 venues.

Now we have cleaned the location points. We can take a closer look at the 10 most common venue for each location point as shown in figure 7.



Figure : The 10 most common venues for each location point

There are some common venues at the location points. But there are also a lot of different venues. Therefore, we will use the unsupervised *k*-means clustering algorithm to segment the location points into clusters. Before we can use this we need to find the optimum *k* (amount of clusters). We can do this by using the elbow method as shown in figure 8.

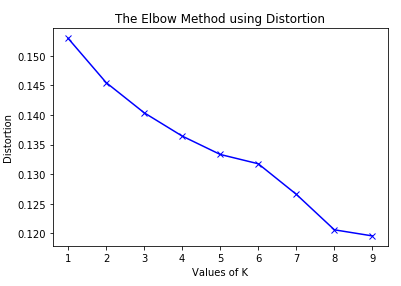


Figure : The elbow method showing the optimum k for clustering

The *k* at which the elbow is located would be the optimum *k*. Unfortunately, in this case there is no really clear elbow. This could be explained by the amount of different venues shown in figure 7. After some trail-and-error, I decided to use a *k* of 4. This resulted into the most diverse clusters.

After we used the *k*-means clustering algorithm to segment the location points into clusters, we can add the cluster labels to the dataframe of figure 7. We will add the coordinates of each location point as well. We need this in order to visualize the clusters later on. This results in the dataframe shown in figure 9.



Figure : The location point with its cluster and most common venues

Now we have clustered each location point, we need to know what each cluster represents and label them. In order to get a clear overview of the most common venues in each cluster, we will use bar charts. They will show the 3 most common venues for the location points of each cluster. Those bar charts are shown in figures 10 to 13.

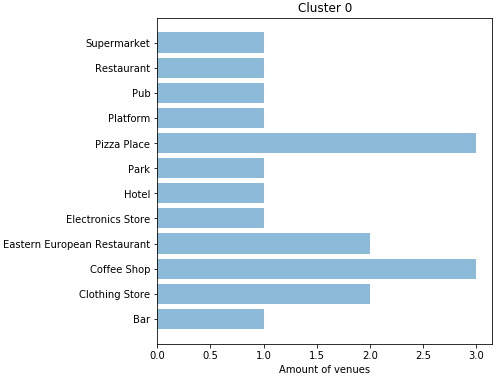


Figure : The 3 most common venues for the location points of cluster 0

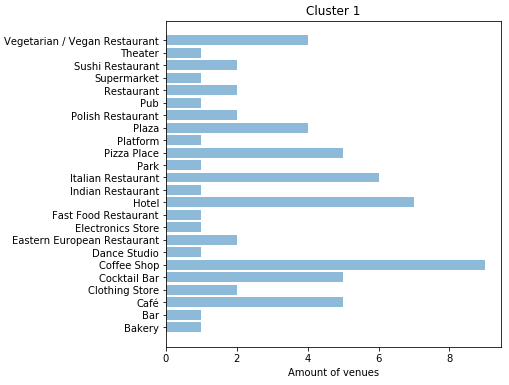


Figure : The 3 most common venues for the location points of cluster 1

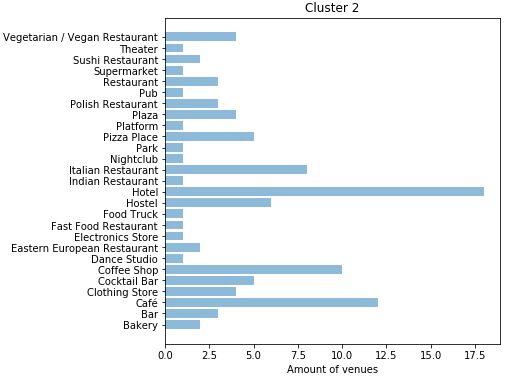


Figure : The 3 most common venues for the location points of cluster 2

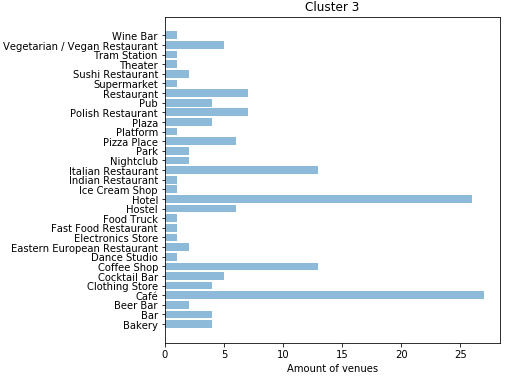


Figure :The 3 most common venues for the location points of cluster 3

From those figures we can see that all of the clusters have a lot of hotels and cafes. So hotels and cafes are ubiquitous. We need to take this into account while deciding how to label the clusters.

Cluster 0 is limited in the amount of venues and has not such one significant feature. Therefore we label cluster 0 as ‘Everyday life’. Cluster 1 contains a lot of social venues, therefore we label cluster 1 as ‘Multiple Social Venues’. Cluster 2 contains a lot hotel and hostels, therefore we label cluster 2 as ‘Accommodation’. Cluster 3 contains multiple different restaurant, therefore we label cluster 3 as ‘Restaurants’.

# Results

Now we can use the Folium library to visualize the location point of each city and its emerging clusters. This is shown in figure 14.

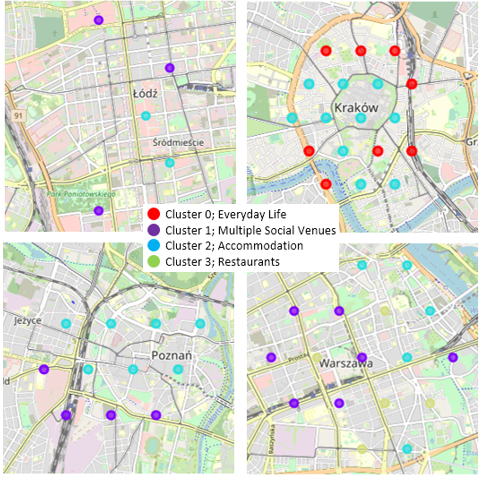


Figure : The location points of each city with its emerging cluster

One of the first things to notice, is the small amount of location points in Łódź. So there were a lot of location points with less than 20 venues in its radius. Second; all cities show cluster for ‘Accommodation’. Third, Kraków is the only city with clusters for ‘Everyday Life’. And finally, Warsaw is the only city with clusters for ‘Restaurants’.

# Discussion

So in Łódź, there are only 5 location points left. Therefore, it might be useful to increase the radius for Łódź and investigate again.

Also, the elbow method did not show a clear optimum *k*. Therefore, a suggestion could be to separate the venue categories in larger overlapping categories. For instance, all the different restaurants could be merged as one general venue category called restaurants. This will result in less diversity and therefore a more optimum *k*.

The final remark are the ubiquitous venues. Some venues just are more general than other venues. Each clusters shows a lot of hotels and cafes. Therefore, it could be useful to use relative frequency compared to other cities. When an overall less frequent venue as a ‘Vegetarian/Vegan Restaurant’ will be more common in one city compared to the other cities, this will be shown. The relative frequency of the venue will be higher.

Also it could be useful to excluded the ubiquitous venues. In this way, the machine learning algorithm won’t be effected by those. This might result in more distinctive clusters. Another way might be to only get the venues of the category of interest. So if we want to know the city that is the most interesting for cultural activities, only get venues which are related to this.

So for further studies, it would be useful to invest more in data cleaning and focusing only less venue categories.

# Conclusion

One similarity for the cities is that they all show clusters for ‘Accommodation’. This is useful to travel and suggests that those are popular cities. The difference of Łódź compared to the other cities is that it has fewer venues. The difference of Warsaw is that it is the only city with clusters for ‘Restaurants’. Kraków is the only city with clusters for ‘Everyday Life’ and also the only one without clusters for ‘Multiple Social Venues’.

Based on this information, Warsaw seems the place to be for visiting restaurants. Because Krakow is one of the biggest cities, but has no social venues, it could imply that it is more common for cultural activities. But this cannot be clearly stated from this data. Besides, Łódź, Poznan and Warsaw seem to be the place for social activities.

# Bibliography

Conway, J., & Sloane, N. J. (1999). Sphere Packings, Lattices and Groups. New York, NY: Springer.

Rosenberg, M. (2020, January 25). *The Distance Between Degrees of Latitude and Longitude*. Retrieved from ThoughtCo.: https://www.thoughtco.com/degree-of-latitude-and-longitude-distance-4070616