IBM Data Science Professional Certificate
Applied Data Science Capstone

Week 4 Project

Cluster analysis of commuter belt locations around Brussels, Belgium

Edward Christie, 14 June 2020

1. Introduction

The goal is to produce an application that carries out cluster analyses of locations around cities - with the specific case of Brussels, Belgium as an example - and using Foursquare data on venues. I decided to focus on food- and drink-related venues[1], i.e. all types of restaurants, eateries, bars, cafes, and shops selling food or drink. The geographical scope is commuter belt locations, that is, locations whose distances from the centre of a designated major city are within a specified range (e.g. at least 10 kilometres, not more than 25 kilometres).

Potential applications for this type of analysis may include, among others, providing comparisons between commuter belt locations for purposes of real estate market analyses (for both real estate agents and for their customers) or for market research purposes, e.g. market gap analyses. The application developed here is simplified and uses only K-Means clustering, rather than more elaborate modelling techniques, so the application should be seen as a work in progress, in the context of the IBM Applied Data Science Capstone course on Coursera. The output of this assignment is a map of the region around Brussels, with colour-coded clusters. The final notebook containing my results can be accessed at: https://github.com/EdChristie/myfirstrepository/blob/master/BruBeltFinalV2.ipynb

2. Data sources

The application uses two datasets. The first is a freely available geographical information dataset[2] which covers all of the municipalities in Belgium, and contains their names, postal codes, and geographic coordinates (longitude and latitude). This dataset covers 2757 postal codes.

The second is an extract from Foursquare, covering commercial outlets in selected municipalities in Belgium, obtained through a standard API request, following the approach presented in the course. The extract was limited to a ring of postal codes around the city of Brussels (see below for scope). This yielded a total of 3205 venues. The dataset was then restricted to food- and drink-related venues, yielding a total of 1084 venues.

[1] In my first week submission, I had initially expressed an interest in focusing on shops and stores, but I found in practice that food and drink venues were more numerous in the suburban locations I had.

[2] Source: Jean-Francois Monfort (jief), see: https://github.com/jief/zipcode-belgium

3. Methodology

The methodological approach relies on two successive components: first, the definition and use of a function to compute the Haversine distance between any two given locations defined by their geographical coordinates; second, the application of the standard K-Means clustering method, as included in the scikit-learn library, to carry out the cluster analysis on the data extracted from Foursquare. I experimented with variations in the dataset on venues in order to stress-test the ability of K-Means clustering to yield relevant results.

3.1 Haversine distance

As we are using data on municipalities and on venues that include geolocation information, it would be useful to be able to compute the geographical distances between them. A very good approximation is given by the Haversine formula. The Haversine formula determines the great-circle distance between two points on a sphere, given their longitudes and latitudes[3]. The great-circle distance is the shortest distance between two points on a sphere, while remaining on the surface of that sphere.

Several Python implementations of the Haversine formula are available. I chose to borrow an explicit block of code[4] – rather than use a library solution[5] – for this assignment.

Having an augmented dataset on locations or venues which includes their respective distances to a given centre point may have multiple practical applications. For example, one may focus on locations within a given radius of a given point, e.g. everything within 5 kilometres of the centre of a city. The application I chose was to define a ring or belt of locations around a given centre, but excluding the centre, in particular the commuter belt around the city of Brussels, Belgium. This simply requires selecting locations that are at least x, but not more than y, kilometres away from a designated centre, with x<y. Based on local knowledge, I set the inner distance at 7 kilometres, and the outer distance at 20 kilometres. The outer distance could arguably be higher given actual patterns of commuting – however this would then entail capturing provincial towns and cities that have significant local economies in their own right (such as Leuven or Mechelen). I also wanted to limit the total request volume and resulting application run-time.

3.2 K-Means Clustering and the Elbow Method

As in the introduction to the Capstone course, I use K-Means clustering as the main analytical method, implemented through the standard scikit-learn version of it. However, I experimented with several modifications to the dataset, as well as with some visualisation techniques, in order to make the best use of K-Means clustering. One key question is how to choose the optimal number of clusters. I therefore implemented the 'Elbow Method' to visualise whether there might be a key inflection point in the relationship between the

[3] See e.g. https://en.wikipedia.org/wiki/Haversine_formula

[4] https://towardsdatascience.com/heres-how-to-calculate-distance-between-2-geolocations-in-python-93ecab5bbba4

[5] E.g. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.haversine_distances.html or https://pypi.org/project/haversine/

number of clusters and the remaining sum of squared distances[6]. The key challenge was that the initial data on venues led to no clear pattern, on the basis of Elbow Graphs, regarding an optimal number of clusters.

At the same time, based on a combination of local knowledge and human judgment regarding venue categories, one possible outcome one would expect a cluster analysis of venues data to be able to detect – indirectly – would be differences in average incomes of local populations. Even though direct indicators such as average wages or average house prices per municipality are not included in the dataset, one could reason that a location with relatively more high-end restaurants would have relatively higher income residents, as compared to a location with relatively fewer higher-end restaurants, and perhaps relatively more fast food outlets.

So the chosen approach was to develop a 2nd version of the analysis, on a derived and much reduced dataset that imposes (reasonable) interpretations of venue category labels. This was put in place by considering that certain venue categories are upmarket ('Status' in the assignment code), others downmarket ('Budget'), and all others midmarket ('Neutral'). The allocation was as shown in Table 1. This allocation reflects my personal judgment.

Table 1: Mapping of Foursquare venue categories into market segments

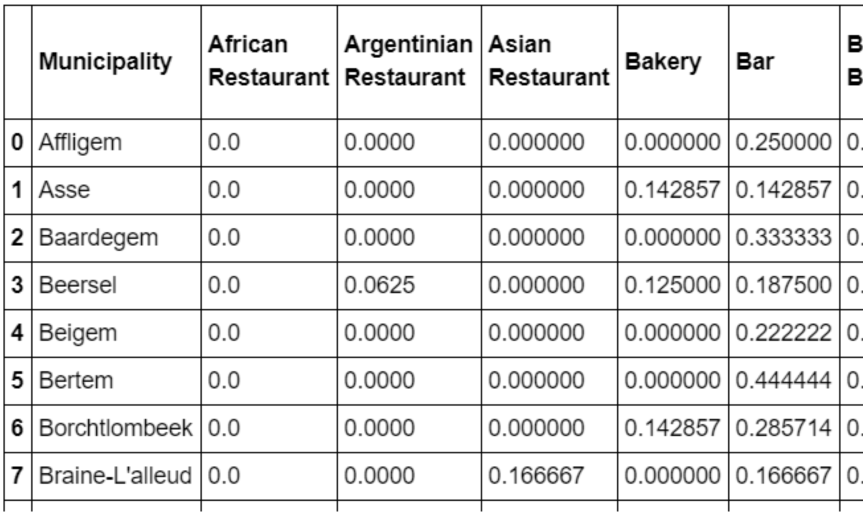
Market Segment	Venue Categories
'Status'	French Restaurant Cheese Shop Gourmet Shop Wine Bar Gastropub Wine Shop Whisky Bar
'Budget'	Snack Place Fast Food Restaurant Pizza Place Sports Bar Cafeteria
'Neutral'	All other venue categories

This allocation was implemented by computing three new columns in the data frame, each equal to the sum of occurrences, within each municipality, of the venue categories belonging to each market segment. For illustration, if a municipality had 2 French Restaurants and 1 Wine Bar, it got a count of 2+1=3 in the 'Status' column. After these new columns were computed, they were divided by the total number of venues in each municipality. The other columns (initial venue categories from Foursquare) were discarded. The resulting new data frame thus had three columns, as named above, with the shares of venues falling under each market segment, such that the sum of shares equals 1.

[6] https://blog.cambridgespark.com/how-to-determine-the-optimal-number-of-clusters-for-k-means-clustering-14f27070048f

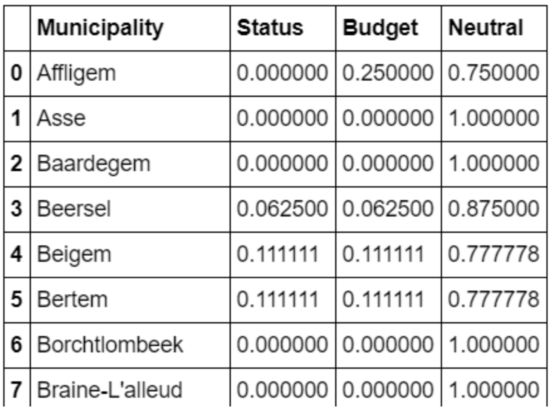
K-Means clustering was applied to both the full data frame (68 venue categories from Foursquare) and the market segments data frame (3 market segments). In both cases, an Elbow Graph was generated by carrying out K-Means clustering for K ranging from 1 to 25, storing the results, and plotting them at the end. For illustration, extracts of the data frames used under each approach are shown below. Data frame heads can be viewed from the respective notebook files in my github repository.

Approach 1: full data frame (extract)



	Municipality	African Restaurant	Argentinian Restaurant	Asian Restaurant	Bakery	Bar	B B
0	Affligem	0.0	0.0000	0.000000	0.000000	0.250000	0.
1	Asse	0.0	0.0000	0.000000	0.142857	0.142857	0.
2	Baardegem	0.0	0.0000	0.000000	0.000000	0.333333	0.
3	Beersel	0.0	0.0625	0.000000	0.125000	0.187500	0.
4	Beigem	0.0	0.0000	0.000000	0.000000	0.222222	0.
5	Bertem	0.0	0.0000	0.000000	0.000000	0.444444	0.
6	Borchtlombeek	0.0	0.0000	0.000000	0.142857	0.285714	0.
7	Braine-L'alleud	0.0	0.0000	0.166667	0.000000	0.166667	0.

Approach 2: reduced data frame (extract)



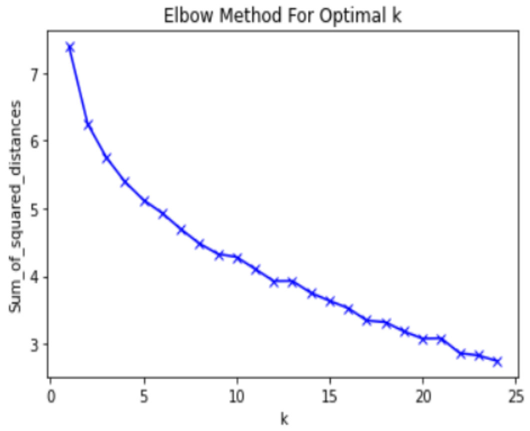
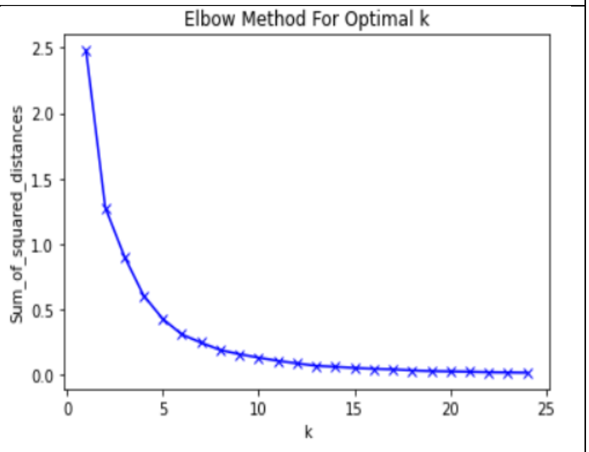
	Municipality	Status	Budget	Neutral
0	Affligem	0.000000	0.250000	0.750000
1	Asse	0.000000	0.000000	1.000000
2	Baardegem	0.000000	0.000000	1.000000
3	Beersel	0.062500	0.062500	0.875000
4	Beigem	0.111111	0.111111	0.777778
5	Bertem	0.111111	0.111111	0.777778
6	Borchtlombeek	0.000000	0.000000	1.000000
7	Braine-L'alleud	0.000000	0.000000	1.000000

4. Results

The first key intermediate result is the comparison of Elbow Graphs between the two approaches described in the methodology section – the first based on the full data frame, the second based on the reduced data frame. The results are shown in Table 2. The codes leading to each of the two Elbow Graphs can be consulted on my github repository[7] – the notebook files are, respectively, BruBeltFinalV1.ipynb and BruBeltFinalV2.ipynb.

[7] https://github.com/EdChristie/myfirstrepository

Table 2: Comparison of K-Means Elbow Graphs

Approach 1: full data frame with 68 venue categories	Approach 2: reduced data frame with 3 market segments
	

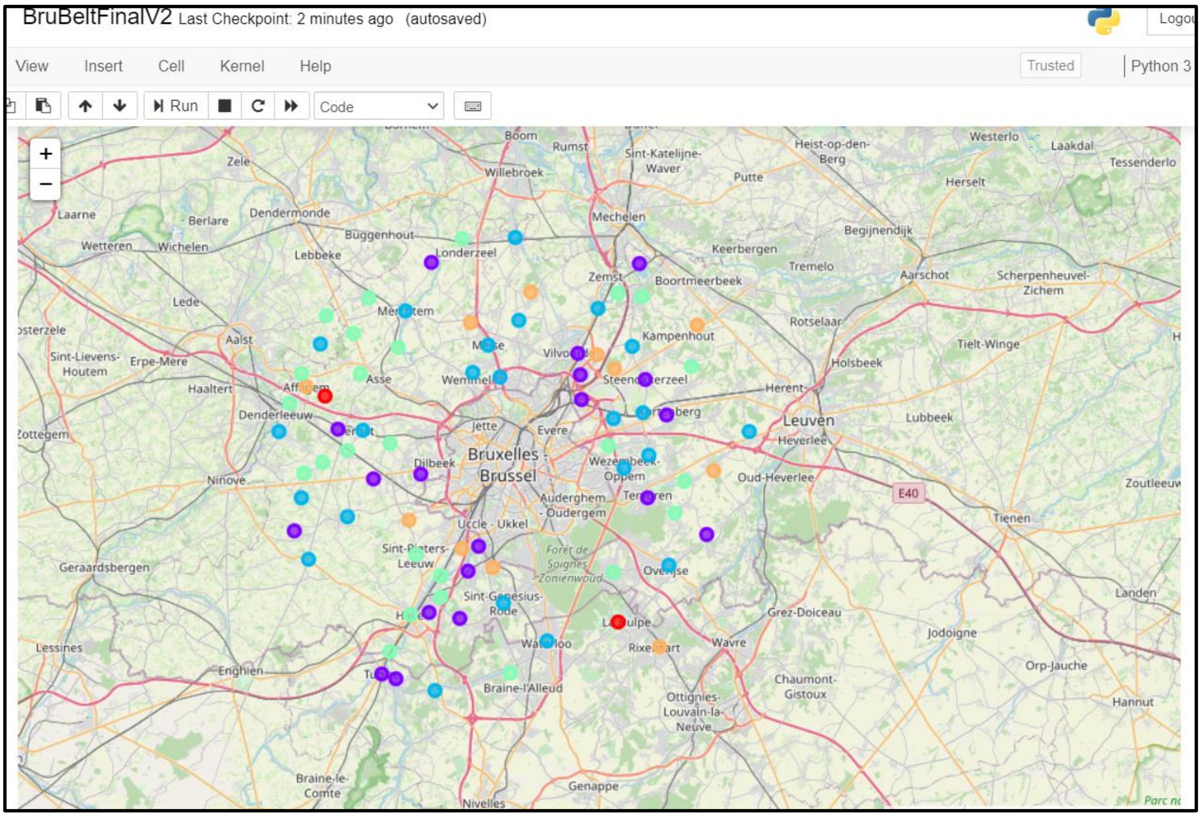
As Table 2 shows, there is considerable difficulty in determining an optimal number of clusters under Approach 1, whereas the results are far clearer under Approach 2, where a range of 4 to 7 clusters would seem efficient. My final choice was therefore to prefer Approach 2 to Approach 1, and to proceed with 5 clusters under Approach 2. Based on that choice, clusters are found that – on visual inspection and with the help of local knowledge – seem to suggest the tendencies described in Table 3 below.

Table 3: Description of clusters

Cluster No	**Tendency**	**Examples**
0	Exceptionally high share of 'Status' venues	La Hulpe
1	High share of 'Budget' venues; no 'Status' venues	Tubize; Buizingen
2	High share of 'Status' venues, lower share of 'Budget' venues	Waterloo; Braine-le-Chateau
3	Low shares of both 'Status' and 'Budget' venues, including many with 0% under both	Braine-L'alleud; Halle
4	High share of 'Budget' venues, lower share of 'Status' venues	Rixensart

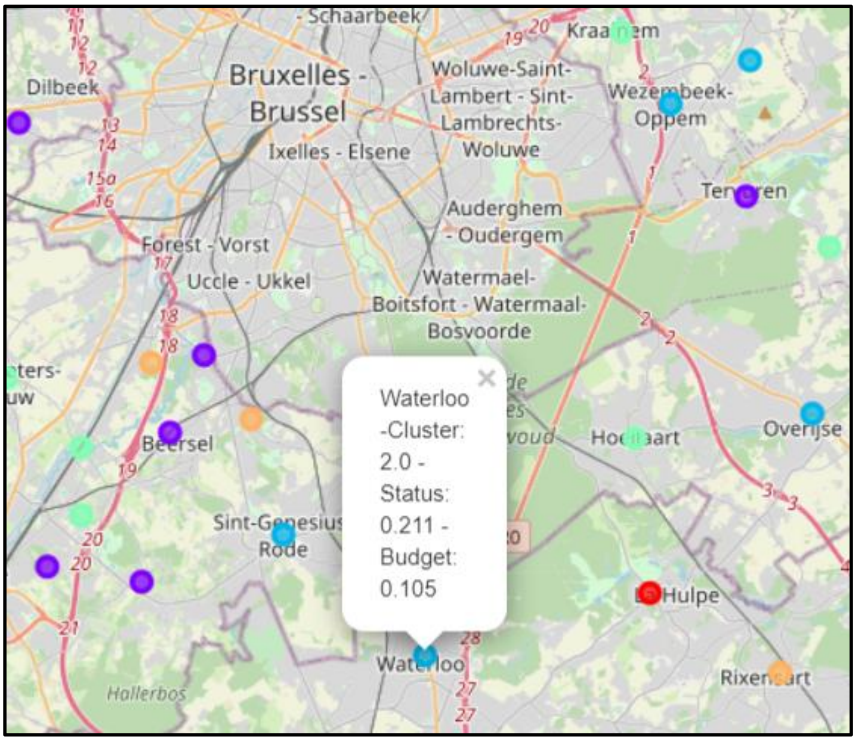
For purposes of visualisation, the clusters were then mapped using the Folium library, as covered in the course. Although the maps display normally in the Jupyter Notebook, they are not necessarily shown when one opens the notebook file I saved on github. I therefore provide screenshots below. Figure 1 shows the overall map. Figure 2 provides a detail on the cluster labels: in addition to what was covered in the course material, I supplement the cluster labels with the shares of 'Status' and 'Budget' venues, rounded to 3 decimals.

Figure 1: Full map with clusters



BruBeltFinalV2 Last Checkpoint: 2 minutes ago (autosaved)

Figure 2: Detail of cluster label for the municipality of Waterloo



Waterloo
-Cluster:
2.0 -
Status:
0.211 -
Budget:
0.105

5. Discussion

The main lesson from this assignment seems to be that clustering using K-Means has certain limitations. It is very easy to implement, but it may fail to generate actionable insights depending on the data set it is applied to. Unless there are particularly clear patterns in the data, adding an additional cluster may yield only a moderate improvement in reducing the sum of squared distances, without any strong inflection point along that relationship which would lead to an especially compelling number of clusters.

This is arguably not surprising – clustering is an unsupervised learning method, intended for unlabelled data – whereas if one has additional insights into the nature of the data, supervised learning methods may be applicable, which have more explanatory power.

In this assignment I experimented with a simple and quite radical simplification of the dataset. Using a combination of human insight and local knowledge, it seemed reasonable to effectively classify venue categories by deeming some of them to be high-market, or low-market, or mid-market. This alternative approach greatly reduced dimensionality while also injecting external information into the dataset (though some potentially interesting information was lost – it's always a trade-off). After this step was carried out, K-Means clustering was still useful as an analytical workhorse, however. One could easily misjudge how similar or dissimilar locations are to each other without applying an objective quantitative method. In the case of specific locations on the outskirts of a city, one might misjudge the numbers or proportions of various kinds of venues, and what that might imply in terms of local consumer preferences and likely average incomes of local populations.

6. Conclusions

With some improvements, I believe that this type of analysis could be further developed and leveraged for a range of practical business applications. The raw data on venues from Foursquare is a powerful resource, and there are many potential use cases.

One possible application could be to further stress-test and validate the market segment cluster analysis I carried out. This could in turn be developed into an attractive visualisation tool, for example for real estate agencies who would like to offer their clients a rapid first impression of the types of venues they would live close to, should they decide to rent or buy accommodation in a particular location. In the same vein, the implementation of the Haversine distance, defining suburban or commuter rings or belts of locations, might also be a simple extension for the websites of real estate agencies: potential tenants or buyers may appreciate a simple kilometre-based filtering tool, alongside classical options such as number of rooms and price range, when searching through a real estate agent's web-site.

More elaborate extensions of what was developed for this assignment could include market research services for potential investors. K-Means clustering could be leveraged by looking at how different a given municipality may be from its cluster centroid. That in turn could be used to compute 'market gaps' – e.g. if there is no Italian Restaurant in a location where, on average, one might expect to find one or more venues of that category.