**Trip Advisor: Clustering German Cities by Venues**

by Tobias (<https://github.com/TobiasGuggemos>), May 2020.

1. **Introduction**:

This project is about German cities. The goal is to support decisions about the next spot for a city trip. Potential stakeholders are all people interested in city trips in Germany. We will cluster the cities based on the venues near to the city centre. This will help to choose an appropriate spot for the next city depending on the purpose of the trip and the interest in different venue types. The city clusters can be used to find similar cities to already visited and much appreciated cities and trips.

1. **Data**:

First, we want to identify the largest cities in Germany. Therefore we will use a list on wikipedia of large german cities, large defined as: with a population of min 100k. We also need to have all the city latitude and longitude coordinates which we can explore with Geopy, a Geocoding libraries for Python.

An extract of the dataframe with the cities and their geo coordinates as well as the cities plotted on a map are displayed below:

A screenshot of a cell phone

Description automatically generatedA close up of a map

Description automatically generated

For all of these cities we need to list the venues around the city center. We will use Foursquare API and Get Venue Recommendations for a given place. Furthermore, let's define the number of requested venues per city and test/start with 100. We consider venues within a circle with a radius of 2,000m around the city center. For the city clustering we also need to know the venue category. This data is also collected via Foursquare.

To prepare the data for the cluster analysis, we use One Hot Encoding. In doing so, we create dummy variables for the column 'venue category' and add the column ‘city’ to this dataframe. Next, we group this dataframe by ‘city’ and calculate the mean for all the venue categories. Finally, we display the top 10 venue categories per city:

A screenshot of a cell phone

Description automatically generated

1. **Methodology**:

To support travel decisions we aim at a clustering of the cities. We start with the K-Means algorithm since it is most popular for clustering in general, easy to understand and implement as well as efficient also with large data sets (even though we do not handle large data here). For the decision about the number of clusters we draw an elbow curve. As evaluation metric we use inertia, the sum of intracluster distances (within-cluster sum-of-squares). Inertia can be regarded as a criterion of how internally coherent clusters are. We have also analyzed the mean of the Euclidean distances from all data points to corresponding cluster centroids with very similar results. We should mention that Inertia assumes convex and isotropic clusters and euclidean distances tend to become inflated in high-dimensional spaces (see <https://scikit-learn.org/stable/modules/clustering.html>). For details please go to scikit-learn.org.

Later on, we will also conduct hierarchical clustering for verification of our results.

A clear elbow is visible at k=2. Since we want more and smaller city clusters, we start our analysis with k=4 clusters in the first run.

**A picture containing screen, group

Description automatically generated**

1. **Results**:

The resulting clusters are shown in different colors in following map:

**A close up of a map

Description automatically generated**

Extract Cluster 0:

**A screenshot of a cell phone

Description automatically generated**

Extract Cluster 1:

**A screenshot of a cell phone

Description automatically generated**

Extract Cluster2:

**A screenshot of a cell phone

Description automatically generated**

Extract Cluster 3:

**A screenshot of a cell phone

Description automatically generated**

Cluster 2 seems to be an outlier including only one city. The city of Salzgitter is the smallest of the whole data set. Furthermore, the Foursquare API call delivered only very few venues for this city. Since K-Means is sensitive to outliers

(<https://www.researchgate.net/publication/293061584_Comparative_Study_of_K-Means_and_Hierarchical_Clustering_Techniques>),

we exclude this city for our further analysis. For verification, we also applied hierarchical clustering. Again, Salzgitter represents an own cluster.

We’re quite happy with the results of the other clusters: Scanning the cities of each cluster and their most common venues, we identify that Cluster 0 seems to be dominated by Cafes/Coffee shops (followed by Hotel and Plaza), Cluster 1 includes a lot of supermarkets and drugstores () and Cluster 3 seems to be somewhere in the middle between Cluster 0 and 1 with Supermarkets, Cafes and Hotels.

Running the analysis again and without the outlier city, the results of hierarchical clustering are:

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

We can quickly recognize 2 main clusters. The first cluster is colored red and including many cities with supermarkets as most common venue type, the second one in green with a lot of “café cities”.

We have also applied the K-Means algorithm excluding the smallest city in the dataset. The update elbow curve:

A picture containing white, lot, black, large

Description automatically generated

We see again a strong kink/break at k=2. Running K-Means with k=2 delivers following results:

A close up of a map

Description automatically generated

The first cluster includes the cities with red markers. The top venues are:

A picture containing knife, table

Description automatically generated

The cities of the second cluster are colored in purple on the map above. Their most common venues are:

A picture containing knife, table

Description automatically generated

1. **Discussion**:

The results clearly point out that there are two heterogeneous clusters. Repeated analyses indicate quite robust results. The same applies for different methods: Hierarchical clustering as well as K-Means with k=2 steadily separate the data points into “supermarket” vs. “cafe” city centers. The break in the elbow curve at k=2 refers to this finding. Further clusters seem to be difficult to interpret. For that reason we finish the analysis with a focus on two clusters. Nevertheless, it is very useful to have the city hierarchy and to have a look at the sub clusters. Users can e.g. search for favorite cities and identify similar cities for the next trip.

For our stakeholders we propose to focus on the purple cluster of the city map (clustered by K-Means) and the green colored cities in the city hierarchy (hierarchical clustering). Both clusters represent cities with cafes and restaurants as most common venues in the city center and strongly overlap. Since our stakeholders are coffee lovers and also look for a city with a large selection of good restaurants, this cluster seems to be a good starting point to find great locations for the next trips. Therefore we schedule a meeting for further discussion – especially on the list of cities of the favored cluster.

1. **Conclusion**:

The clustering of Germany city centers discovers two significantly different clusters. As a result we could narrow down the number of potential cities for the next journey. For our stakeholders, we have identified a cluster with cities that are characterized by a lot of cafes and restaurants nearby the city center. For further insights it would be interesting to see how more criteria besides venues affect the results.