Multivariate Statistics

Cluster Analysis

of peripheral municipalities in Austria
based on selected variables

Seminar Thesis

856.134
Winter Term 2024/25

Max Schneeberger

Isabella Tkalec

Table of contents

1 Introduction

Clustering methods are used for grouping data points together based on their similarities and therefore identifying homogeneous subgroups. In many cases, the approach of clustering is used for data exploration for a better understanding of complex data (Gao et al., 2023, p. 2; Backhaus et al., 2018, p. 21). In the domain of computer science, cluster analysis can also be understood as an unsupervised learning approach, hence the classification parameters and characteristics of the data, i.e. class labels, are unknown. This implies that there are no predetermined classes yet and instead, classes are generated based on similarities within features. For analyzing data via clustering, a wide variety of algorithms exist from which the chosen method influences the outcome of the analysis. Therefore, choices on the algorithm and input parameters must be made in advance to perform clustering (Mahdi et al., 2021, p. 4).

Different classifications of clustering methods exist, but they can roughly be organized into four groups, namely hierarchical clustering, density-based clustering, center-based partition clustering, and model-based clustering (Gao et al., 2023, p.3). Hierarchical clustering builds typically on a bottom-up approach where a nested hierarchy of clusters is generated by grouping the most similar data points and then progressively merging smaller clusters into larger ones. The result can be visualized in a dendrogram. In contrast, partitional methods such as k-means assume that data can be represented in a set of spherical shaped clusters, based on the data points within a cluster's distance to its center and processes in an iterative manner. While partitional clustering assumes equal-sized clusters, density-based methods are used for irregular, atypically shaped data, due to the underlying mechanism of the density distribution that helps for the identification of outliers and clusters of varying shapes and sizes (Gao et al., 2023, p. 5; Jäckle, 2017, pp. 57–72).

In case of our analysis, we decided to use the center-based partitioning clustering algorithm k-means, since it is described as effective on metric datasets, used for explorative approaches to detect connections between variables and relatively simple to interpret the generated clusters (Backhaus et al.,2018, p. 15; Gao et al., 2023, p. 5). We analyzed a preprocessed dataset, which contains municipalities in Austria based on their municipality type, as well as

different socioeconomic variables. To gain an understanding and insights into the dataset, we inspected the dataset and selected four variables for the analysis.

Our goal was to determine whether distinctions exist between these municipality types across the chosen variables. To achieve this, we applied the k-means clustering algorithm using Python and the according libraries, tested with different values of k (numbers of clusters) and two distance matrices (Euclidean distance and Manhattan distance) and then evaluated the clustering results to explore the differences between municipality types in accordance with our variables. The aim of our study and the chosen variables lead us to formalizing the following hypotheses:

> ***H0:*** *"There are no significant differences (distributed homogeneously) between the municipalities across clusters with respect to the selected variables."*
>
> ***HA:*** *"For at least one variable, there are significant differences between municipality types across clusters."*

2 K-Means clustering method

The k-means clustering algorithm is a frequently used unsupervised learning method, that divides a dataset into a previously defined number of k clusters by using the similarity measurements. With the choice of Euclidean Distance as a similarity measurement, the process of clustering can be described as following:

The algorithm starts by initializing K centroids (Figure 1), which can be selected randomly but also based on other methods like k-means++ which help in better centroid placement. Regarding the number of clusters, the user selects the number k based on a pre-assessment of the dataset and other measurements like elbow plots. The elbow method helps to determine this number by visualizing the relationship between heterogeneity (also called inertia) and the number of clusters in a scree plot (Jäckle 2017, p. 476). Each data point is then assigned to its nearest centroid based on the chosen distance metric. The centroid is calculated as the mean distance of all data points of the according cluster and represented as a point in the center of each cluster. Usually, every data point can only be assigned to one

cluster. Afterwards, the centroids are updated and re-partitioned. The steps of assigning the data points to the nearest centroid and recalculating centroids are repeated in an iterative way, until a certain convergence is reached. This implies that either the assignments to clusters or the centroids will no longer change with ongoing iterations and the process is completed (Jäckle, 2017, p. 64; Gao et al., 2023, p. 6).

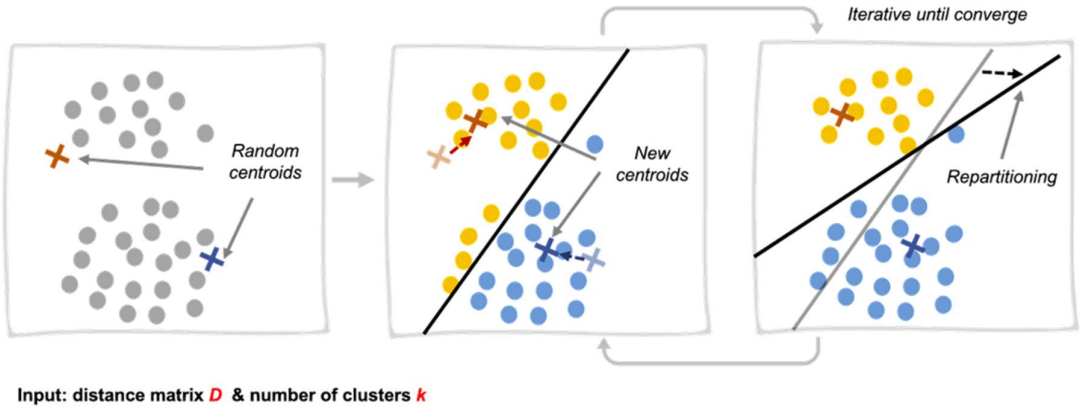

Figure 1: The steps of k-means clustering (Gao et al. 2023, p. 7).

2.1 Benefits and constraints of k-means clustering

One of the key benefits of k-means is its simplicity and easy implementation. It is computationally efficient and therefore suitable for medium to large sized datasets. K-means performs especially well with handling large, multidimensional datasets and has a high scalability. It can be applied to a wide range of applications with complex data, including social sciences (King, 2015, pp. 68–76; Mahdi et al., 2021, p. 7). In contrast to hierarchical clustering methods, where formed clusters and assigned data points cannot be dissolved during the analysis process, partitioning methods such as k-means have the advantage that data entities can be exchanged and reassigned between clusters during the process (Backhaus et al. 2018, pp. 457-459).

Major limitations include a high sensitivity to outliers due to the mathematical processing of calculating mean values for centroid construction, that k-means only works for roughly spherical shapes of data distributions, and due to its centroid-method, which might only identify the local optimum of the dataset within a computing process (Jäckle 2017, p. 64; Gao et al., 2023, p.6). The following Table (Table 1) summarizes the key features of partitional-

based clustering techniques such as k-means and provides a small comparison to hierarchical and density-based clustering.

Table 1: Comparison of partitional clustering to hierarchical and density-based techniques. Based on Gao et al. (2023, pp. 5–10), Mahdi et al. (2021, pp. 7-11) and Jäckle (2017, pp. 64-67).

	partitional	hierarchical	density-based
Description	Provide fixed number of clusters; centroid calc. and rearrange.	„bottom up“, starting with 1 datapoint = 1 cluster and morph clusters stepwise (single linkage vs. Ward)	Clustering based on density distribution
Input	Nr. of clusters (a priori), numerical data	Numerical/mixed; similarity m. (Euclid), Linkage criteria	Num. data, similarity m., neighborhood dist., min. density
Advantages	Fast to estimate; easy to interpret; works well with large datasets; diff. dist. metrics can be applied	Combination with other clustering methods possible; quick interpretation	robust to outliers; can cluster atypical shaped clusters; best number of clusters auto-est.
Constraints	Assumes equal-sized clusters, sensitive to outliers	Hard to handle/compute large datasets/diff. density	Parameter sensitivity; diff. for high-dim, data
Examples	K-means	Agglomerat. Hierarchical Clustering, (Dendrogram)	DBSCAN
Application	Large datasets with equally sized/spherical clusters	Small-medium datasets with hierarchical relationships	Medium datasets with atypically shaped clusters/outliers

2.2 Similarity Measurement

The majority of clustering algorithms requisite the calculation of similarity or dissimilarity to assign datapoints to the according clusters. While there exist a broad variety of similarity and dissimilarity measures, a common approach is to use distance. The simplest way to calculate distance is by utilizing Euclidean distance, the length of the path between two data entities in two-dimensional space. Despite its easiness, Euclidean distance may not be applicable for every input data. It is unsuitable for noncontinuous variables or if there are differences in scale of the variables. Furthermore, it is susceptible to outliers. Average Euclidean distance and weighted Euclidean distance have been developed to cater for some of the limitations of the normal Euclidean distance method. Often considered as the counterpart to Euclidean distance is Manhattan distance: it is more reliable with regards to outliers because it calculates the sum of the absolute differences in each dimension and therefore does not include squared terms. One visual difference that can often be noticed when comparing

Euclidean distance with Manhattan distance is the shape of the formed clusters: Euclidean distance tends to form clusters more circular in shape while Manhattan distance often produces clusters more diamond shaped. Other common distance measures for continuous variables include Chebyshev distance, Mahalanobis distance, Cosine distance, Dot product, Chord distance, and Canberra distance (Gao et al. 2023, pp. 3-5). A usual choice of similarity measurements for metric data and calculating centroids in k-means clustering is Euclidean distance (Jäckle 2017, pp.57-65).

3 Method and workflow

In the following section, the initial investigation of the provided dataset is presented and the workflow of our clustering analysis is described with its key steps that have been applied and the decisions that were made during the process. The code that was used for the python execution is presented as well.

3.1 Initial data examination and exploration

The data for our analysis represents a municipality dataset from Austria. Each row in the dataset stands for one of Austrias municipalities that have either the categorization

- 330 = *ländlicher Raum im Umland von Zentren,*" with the subcategory peripheral which translates to rural area near urban centers, or
- 430 = "*ländlicher Raum"* with the subcategory peripheral which translates to rural areas (Statistik Austria 2021, p. 5).

Rural municipalities are typically defined by having a lower population density and less developed infrastructure than urban regions. These are often characterized by a higher prevalence of agriculture, forests, and natural landscapes. Therefore, the economy in rural municipalities might often be centered around farming, forestry, and small local businesses.

One important fact to point out is that out of the 398 municipalities listed in the dataset, only 38 are of type 330. In general, such an imbalance can introduce the following problems: With such a small proportion of municipalities belonging to rural type 330, the algorithm may struggle to form a meaningful cluster for this group, as the majority of the municipalities (rural type 430) dominate the centroid calculations. This imbalance could result in clusters that do

not accurately represent the characteristics of the 330 municipalities, as their small number may not influence the centroids enough, leading to misclassification or the merging of the smaller group with the larger one. Additionally, the few 330 municipalities may be treated as outliers, affecting the overall clustering results, and potentially skewing the cluster definitions towards the majority class. The disproportionate distribution could lead to incorrect classifications and interpretation in of the rural type 330, as K-means tends to form clusters based on minimizing intra-cluster variance and therefore often favoring larger groups.

Despite the imbalance of the categorized rural types, we still decided to center our study objectives around this variable. Specifically, we wanted to find out if municipalities with rural type 330 respectively 430 spread homogeneously among clusters based on a chosen set of attributes. The attributes we chose for our analysis were “percentage of people employed in the third sector” (data from 2019), “percentage of outbound commuters” (data from 2019), and “percentage of foreign population” (data from 2019). We chose these attributes based on the following assumptions:

We anticipated significant differences in the percentage of people employed in the third sector when comparing the two different types of regions, meaning that geographical differences (types of municipalities) correlate with economical differences (employment, commuting). While both municipality types may have a lower percentage in this employment sector, we expected it to be even lower in rural areas (type 430) since they are usually more dominated by agriculture and forestry. Furthermore, we assumed that rural areas near urban centers (type 330) might be categorized by an even higher number of outbound commuters than type 430 municipalities, since they are located around urban centers which offer more employment options within service sector. For the third dependent variable, we estimated that both municipality types are characterized by a lower percentage of foreigners compared to urban regions, but rural municipalities type 430 might even consist of lower numbers due to reduced social and infrastructural networks.

3.2 Workflow of the clustering analysis

The first step to start our project was to select the **clustering method** of our choice. As our goal was to investigate the distinction between municipality types in relation to metric

variables, partitional clustering, specifically K-means, was our choice as a clustering method. Since our dataset consists of a couple of hundred data points, a fast and simple clustering algorithm like k-means can be considered. Due to the dataset consisting of percentage values which are of the same magnitude, possible outliers are less likely to distort the clustering process, which could be an issue with k-means clustering algorithms in general. Another reason was that partitional clustering is due to its unsupervised method a reasonable choice for discovering possible hidden patterns in data.

In the next step, we had to choose a technology stack for implementing our clustering algorithm. Since we wanted to make use of open-source **software** that is freely accessible but still refers to a standardized processing and analysis, we opted for the Python Jupyter Notebooks instead of a statistical, licensed software package. To foster coworking, we went for Google Colab as an environment. To have maximum control over the analysis and visualization process, we decided to implement the k-means clustering algorithm from scratch instead of relying on prebuilt libraries. This was achieved by merging the approaches from McDonald (2022) and Dataquest (2023).

For k-means, the **k number of clusters** has to be defined a priori (Backhaus et al., 2018, p. 475). To define an optimal number of clusters, the elbow method was chosen and applied with the python script accordingly. The results can be found in section 4.

The last step before the cluster analysis could be implemented, was to select the **similarity measurement** for calculating the cluster centroids. K-means usually uses Euclidean distance as a similarity measurement. However, some studies suggest that other distance measures, such as Manhattan distance work better in a high dimensional space (Aggarwal et al. 2001). Therefore, both measurements have been tested, with no significant differences in the outcome.

Regarding the **clustering process** and running the code, our python code can be structured in the following parts (see next page):

```
df = pd.read_csv(file_name_input, index_col='GEM_NAME')
print(df)
print("Missing values " + str(df.isnull().sum().sum()))
df.describe()
```

GEM_NAME	p_employ_sec_III_19	p_comm_out_19	p_foreign_19
Bocksdorf	45.28	85.9	2.37
Burgauberg-Neudauberg	54.55	81.1	4.85
Eberau	66.67	72.7	11.89
Gerersdorf-Sulz	63.93	82.5	5.51
Güssing	77.45	50.7	9.90
...	...	...	...
Mittelberg	82.23	12.4	46.98
Schnepfau	32.59	70.8	13.94
Schoppernau	67.47	57.8	9.63
Schröcken	85.71	56.9	15.64
Warth	90.67	26.1	6.10

[398 rows x 3 columns]
Missing values 0

Figure 2: importing datasets into Google Colab

The first step, which can be seen in Figure 2, was to import our datasets into the Google Colab environment. In addition to that, we performed basic data exploration – for example checking for the validity of Pearson Correlation Coefficient and further assessing possible multicollinearity between our variables.

As can be seen in Figure 3, the highest correlation values are around 0,4 which is low enough to continue with our analysis, although some slight correlation is expected. We also started with four variables but disposed of the fourth due to multicollinearity (see section 4 and 5).

calculating Pearson Correlation Coefficient

```
corr = df.corr()
corr
```

	p_employ_sec_III_19	p_comm_out_19	p_foreign_19
p_employ_sec_III_19	1.000000	-0.316096	0.425375
p_comm_out_19	-0.316096	1.000000	-0.358130
p_foreign_19	0.425375	-0.358130	1.000000

Figure 3: Pearson Correlation Coefficient of our variables

The next big section of our code (snippet can be seen in Figure 4) defines numerous functions which when combined can perform comprehensive k-means clustering. Specifically, the defined functions deal with choosing the appropriate number of clusters (elbow method), initializing random centroids and labelling data points in each iteration to a cluster based on Euclidian or Manhattan Distance.

function definitions

```
# function for calculating the optimum number of clusters
def optimise_kmeans(data, max_k):
    means = []
    intertias = []

    for k in range(1, max_k + 1):
        kmeans = KMeans(n_clusters=k)
        kmeans.fit(data)

        means.append(k)
        intertias.append(kmeans.inertia_)

    fig = plt.subplots(figsize=(10, 5))
    plt.plot(means, intertias, 'o-')
    plt.xlabel('Number of Clusters')
    plt.ylabel('Inertia')
    plt.grid(True)
    plt.show
```

Figure 4: Function definitions for k-means clustering

The code section in Figure 5 puts all of the previously defined functions together and visualizes the clustering process in an iterative way. The maximum number of possible iterations was set to 100.

performing analysis with euclidian distance OR manhattan distance

```
t_max_iterations = 100
t_k = 3

t_centroids = random_centroids(cluster_data, t_k)
t_old_centroids = pd.DataFrame()
t_iteration = 1
t_labels = pd.Series(dtype='int64')

while t_iteration < t_max_iterations and not t_centroids.equals(t_old_centroids):
    t_old_centroids = t_centroids
    t_labels = get_labels_eucl(cluster_data, t_centroids) # switch distance methods here
    t_centroids = new_centroids(cluster_data, t_labels, t_k)
    plot_clusters(cluster_data, t_labels, t_centroids, t_iteration)
    t_iteration += 1
```

Figure 5: Combining previously defined functions

4 Results

Our results of the k-means cluster analysis were derived from running the set up python script and deriving output regarding descriptive statistics of the selected variables, elbow plot generation for k numbers of clusters, Pearson correlation for determining multicollinearity issues, as well as cluster analysis results including centroids and cluster composition.

Our input data (Figure 6) has 398 instances (the municipalities) of 38 are of type 330 ("Ländlicher Raum, peripher") and 360 of type 430 ("Ländlicher Raum im Umland von Zentren, peripher"). The variables p_employ_sec_III_19 (percentage of employees in sector III in 2019) and p_comm_out_19 (percentage of outbound commuters in 2019) show a higher standard deviation compared to the variable p_foreign_19 (percentage of foreign population in 2019). The percentage of data instances that lie within the interquartile range for the respective variables are as follows: p_employ_sec_III_19 (47-70%), p_comm_out_19 (60-76%) and p_foreign_19 (3-9%). Especially the outbound commuters have a relatively low deviation in comparison to the overall data range of the respective variable.

	p_employ_sec_III_19	p_comm_out_19	p_foreign_19
count	398.000000	398.000000	398.000000
mean	58.099849	67.047487	6.867990
std	17.104942	14.048994	5.914621
min	0.000000	12.400000	0.120000
25%	46.675000	60.400000	3.042500
50%	57.720000	70.000000	5.470000
75%	70.170000	76.400000	8.977500
max	96.470000	96.200000	64.630000

Figure 6: Descriptive Statistics of input variables

Number of clusters: For determining the number of clusters to be created, we performed an elbow plot analysis (see Figure 7). The elbow plot helps choose the optimal number of clusters k by analyzing how inertia (sum of the squared distances between each data point and its assigned cluster centroid) decreases as k increases. Multiple runs of k-means clustering are executed while increasing the number of clusters in every iteration by one. Then, the inertia is plotted against the respective k for each iteration. The goal is to find the point (k) where adding more clusters doesn't significantly improve inertia anymore. At this point, the selected number of clusters reaches a significant reduction in inertia, compared to lower values of k, while further increasing k would lead to only marginal improvements. In our case, this was achieved with three clusters.

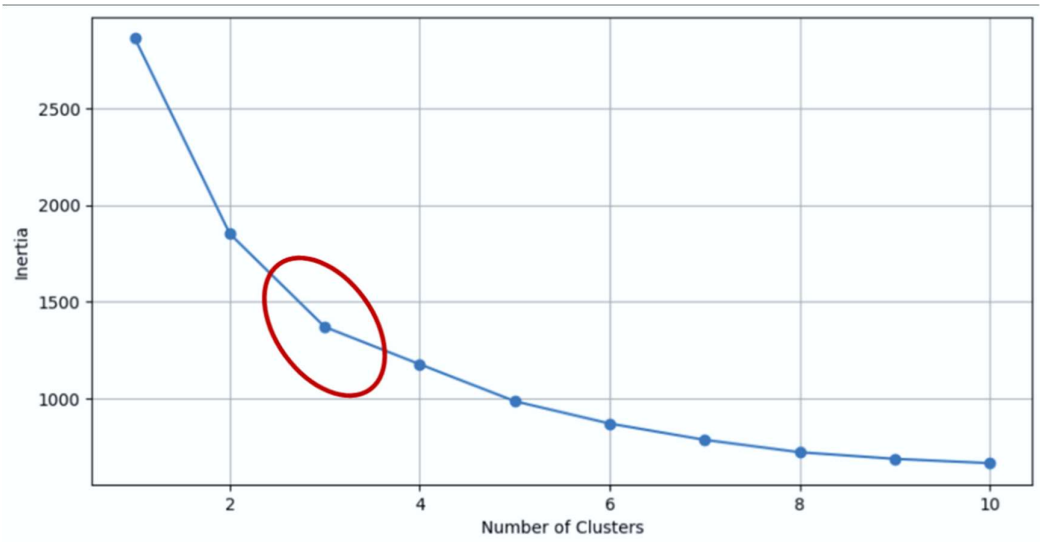

Figure 7: Elbow Plot with reduced inertia at k=3 clusters.

Data distribution and Pearson Correlation: Figure 8 and table 2 show the distribution and correlation values of the variables. For the majority of the variables, it can be assumed that they are normally distributed and have a linear correlation. This qualifies them for further analysis using the Pearson Correlation Coefficient in order to rule out multicollinearity between the variables. As can be seen, we also included a fourth variable, namely p_employ_sec_I_19 (percentage of people employed in the first sector in 2019). However, this variable had a high negative correlation with p_employ_sec_III_2019 (-0.58) when calculating the Pearson Correlation Coefficient (see Figure 8). Therefore, we excluded p_employ_sec_I_19 from our analysis.

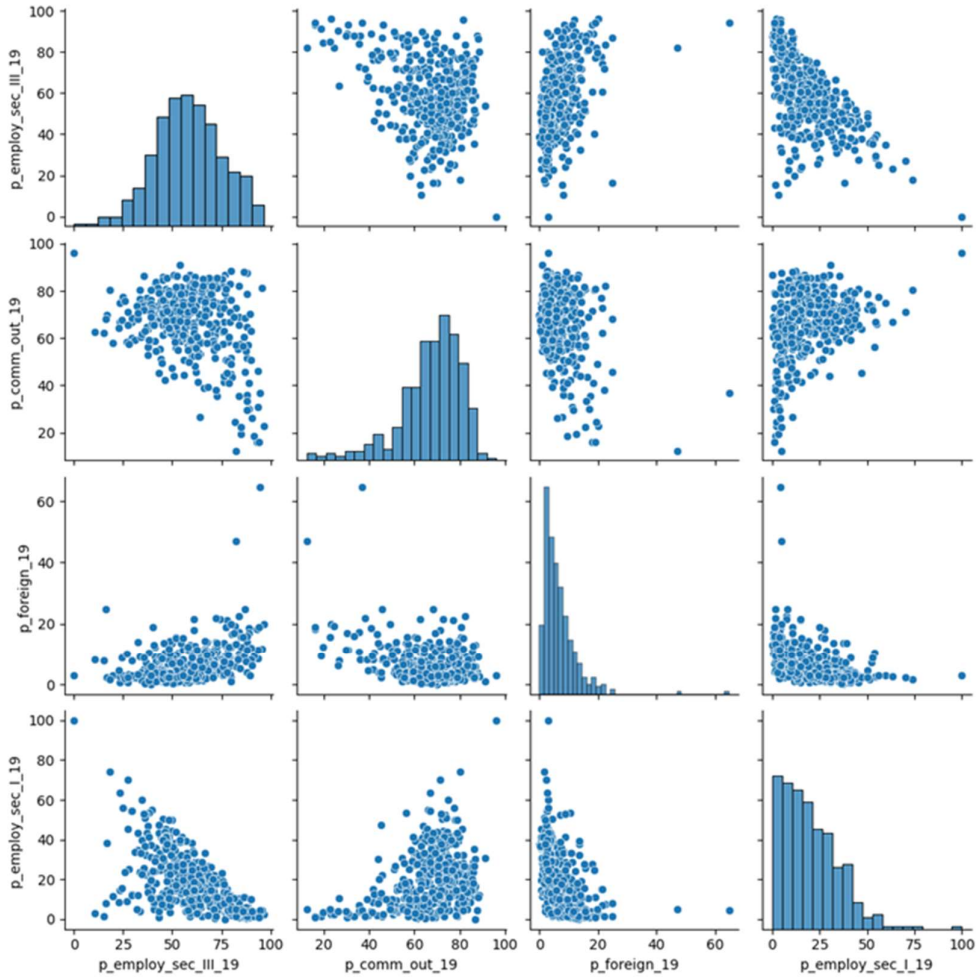

Figure 8: Distribution and correlation of the variables

Table 2: Pearson Correlation Coefficients

	p_employ_sec_III	**p_comm_out**	**p_foreign**	**p_employ_sec_I**
p_employ_sec_III_19	1	-0.316096	0.425375	-0.583236
p_comm_out_19	-0.316096	1	-0.35813	0.308202
p_foreign_19	0.425375	-0.35813	1	-0.40378
p_employ_sec_I_19	-0.583236	0.308202	-0.40378	1

Cluster visualization: The clustering results are represented visually in figure 9. The respective clusters are displayed in different colours (yellow, purple and turquoise) and the associated cluster centroids are depicted as blue coloured points. In order to visualize the results in two-dimensional space, principal component analysis has been conducted to decrease our dataset to two dimensions. While the yellow and the purple cluster form very homogenously with

only a few outliers, the turquoise cluster is more spread. It can also be seen, that the clusters take a slightly spherical shape due to the k-means algorithm and the centroid method with Euclidean distance measurements.

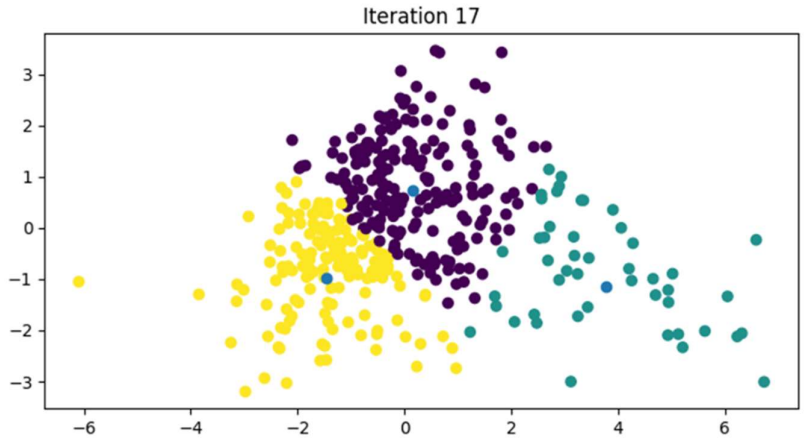

Figure 9: Clustered data points

Cluster centroids: Additionally, the calculation of the cluster centroids was done and an output in form of a table was generated. Table 3 shows the variable values for the cluster centroids of each cluster. To ensure that all features contribute equally to the clustering process, the variable range was scaled to 0-10. Cluster 0 and cluster 2 are similar with high to moderate values for p_employ_sec_III_19 and p_comm_out_19 whereas cluster 1 only shows high values for p_employ_sec_III_19. This also shows that cluster 0 has a high number of people that are employed in the service sector, as well as a high outbound commuting rate. Cluster 1 shows a significantly high number of people within the service sector, but very low values of out-commuters and foreigners, whereas cluster 2 is dominated by outbound commuters but also a significant amount of service sector employees. The variable p_foreign_19 does not play a major role for any of the clusters.

Table 3: Cluster centroids

	Cluster 0	**Cluster 1**	**Cluster 2**
p_employ_sec_III_19	7,04	8,54	4,71
p_comm_out_19	7,32	3,63	7,02
p_foreign_19	1,86	2,73	1,56

Figure 10 shows the rural type percentages among the clusters. The vast majority of municipalities with rural type 330 fall in cluster 0 and a small proportion in cluster 2. Cluster 1 is made up entirely of municipalities with rural type 430. It must be pointed out that the clustered dataset is primarily made up of municipalities with rural type 430 (90,5% overall) and has a small portion of rural type 330 (9,5% overall). Therefore, in relation to the rural type distribution within the dataset, cluster 0 has a significantly higher number of type 330 (14,9%) and cluster 1 a significantly higher number of type 430 municipalities. Cluster 2 has mixed results regarding the municipality type.

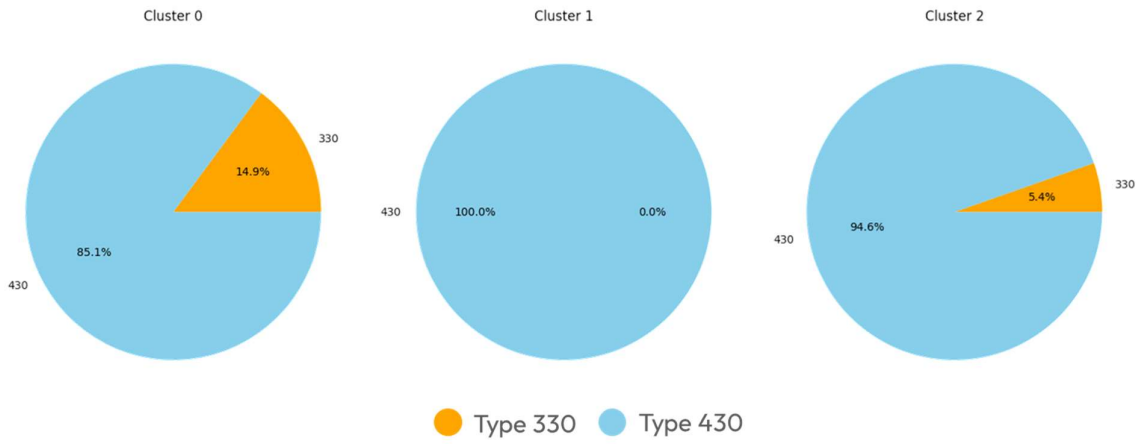

Figure 10: Municipality type percentages among the clusters

5 Discussion

In this section, results of the cluster analysis are interpreted and examined critically. After a thorough interpretation of the clustering outcome and connecting the findings to existing literature, limitations are pointed out shortly. Additionally, the silhouette method was applied for assessing the quality of the clustering quantitively and final conclusions are summarized.

5.1 Interpretation

Results show, that the first cluster is characterized by a high percentage of outbound commuting as well as a high percentage of the service sector (see table 3). The results

indicate, that this cluster contains a significantly larger amount of municipality type 330 (14,9%) compared to the distribution of the overall dataset (9,5%) (see figure 10). This could be interpreted as municipalities of the type 330 “rural areas near urban centers” possibly having a higher number of service sector employees, while commuting to other municipalities or urban centers which are close to these municipality types, resulting in high service sector employment and commuting.

The second cluster has a significantly higher percentage of people working in the service sector, but relatively low values of foreigner percentages and outbound commuting. Interestingly, all of the municipalities in this cluster are of type 430 (rural areas). This indicates, that some of these municipalities might have a high employment rate in the third sector, which could be due to some of the rural areas forming more independent regional centers or local hubs within these micro-regions, as seen in (Vaishar & Zapletalová 2009). Another reason for the high percentage of service sector employment could also be at least partially related to touristic, rural regions, where tourism acts as an economic driver which is dominant in many Austrian areas (Hummelbrunner & Miglbauer 1994). The high demand for services in this industry could be therefore connected to the composition of this cluster.

The third cluster also has a high percentage of outbound commuters, but a more moderate portion of the service sector. Since peripheral areas often have a higher portion of agriculture and farming (Shaikh et al. 2023), this could also be the reason for lower proportions of service sector employees. The percentage of foreigners seems also low, but since this variable has a relatively high variance, this has to be considered carefully and might not lead to conclusive results or explanations. The composition of the two municipality types is mixed, suggesting that economic patterns in these municipalities do not strictly align with their classifications. A possible explanation could be the local diversification of rural areas which might be highly diverse and individual per region.

Other findings of the analysis suggest, that the variable of percentual foreigners (*p_foreign_19*) might not play a major role in defining the cluster composition. This could be due to the broad range (interquartile range of 3-9%) and might be interpreted as diverse distributions of foreigners across municipalities.

The significant overrepresentation of rural municipalities near urban centers in the first cluster suggests, that these areas are more economically integrated to geographically close urban centers, while the second cluster is exclusively consisting of the rural area type. This could support the theory, that some of the rural areas have a more localized centrality with service sector employment within these regions and therefore less commuting to other areas for employment.

Overall, findings show a significant difference between municipality types across clusters, implying that economic and commuting patterns vary significantly, implying the adoption of the alternative hypothesis HA. Nonetheless, the H0 hypothesis cannot be rejected with absolute certainty, since results are mixed and might imply a more complex or locally varying relationship of economy, employment and regional type, meaning that additional factors might influence the patterns, especially for some variables like the percentage of foreigners.

5.2 Limitations

It is also important to mention, that the disproportion of the municipality types (90.5% of type 430) might distort or influence the results, since the outcome might be more likely reflecting economic variations within the type 430 municipalities, while insights to the type 330 municipalities could be limited due to the smaller percentage.

Another factor that limits possible conclusions is, that the third cluster is relatively inconclusive and widely spread, which suggests that the data within does not necessarily fit very well into a (separate) cluster.

Overall, there are also limitations regarding the clustering technique of K-means. Firstly, some assumptions like the spherical shape and rather equal density of data and therefore clusters were made, which might not perfectly reflect the true nature of the underlying dataset. Although this is partly due to the mathematical approach of Euclidean distance measurements and Manhattan distance was used as well to counteract possible errors, there is still no assurance for a perfect method selection for this dataset. Other clustering methods, such as hierarchical clustering, a previously applied PCA for the larger dataset or different multivariate statistical methods could yield different results by capturing different or more complex relationships.

5.3 Quality assessment

After conducting our cluster analysis, we sought am additional quantitative measure to assess the quality of the clustering results. Several methods exist, one of which is the silhouette plot. While used for evaluating the number of clusters for k-means, it can also be applied for interpreting the quality of a k-means clustering result. The silhouette score therefore provides insight into how well each data point is assigned to its respective cluster by quantifying the separation between clusters and the cohesion within them (Gao et al. 2023, p. 20; Backhaus et al. 2018, pp. 476-486). Positive values close to 1 indicate, that the average distance to the next cluster is significantly larger than the average distance within the cluster, which means that clusters are more coherent within and can be seen as well separated from the other clusters. Values close to 0 suggests ambiguous results regarding which cluster is better suited for the according data point(s), while negative values imply misclassification, and therefore better fit to other clusters (Jäckle 2017, p. 61). In case of our analysis, the overall silhouette score for the clustering solution was computed as well as the silhouette plot (figure 11). In case of our analysis, the vast majority of data points achieve a silhouette score above 0 with an overall silhouette score of 0,32, implying that most of the municipalities are well matched to their assigned clusters accordingly. The score of 0,32 stands for an overall moderate fit of the cluster structure, with a certain degree of overlap. This is coherent with the conclusions from the results. A higher score would indicate more distinct clusters, but the result of a more moderate value in our case suggests that some municipalities may share characteristics across cluster boundaries and that the chosen variables might not fully reflect and describe the municipality types. This could be based on complex economic differences and ambiguities in rural areas, with certain municipalities not directly fitting into a single classification or cluster. However, the analysis still provided valuable insights into employment and commuting in relation to demographic patterns.

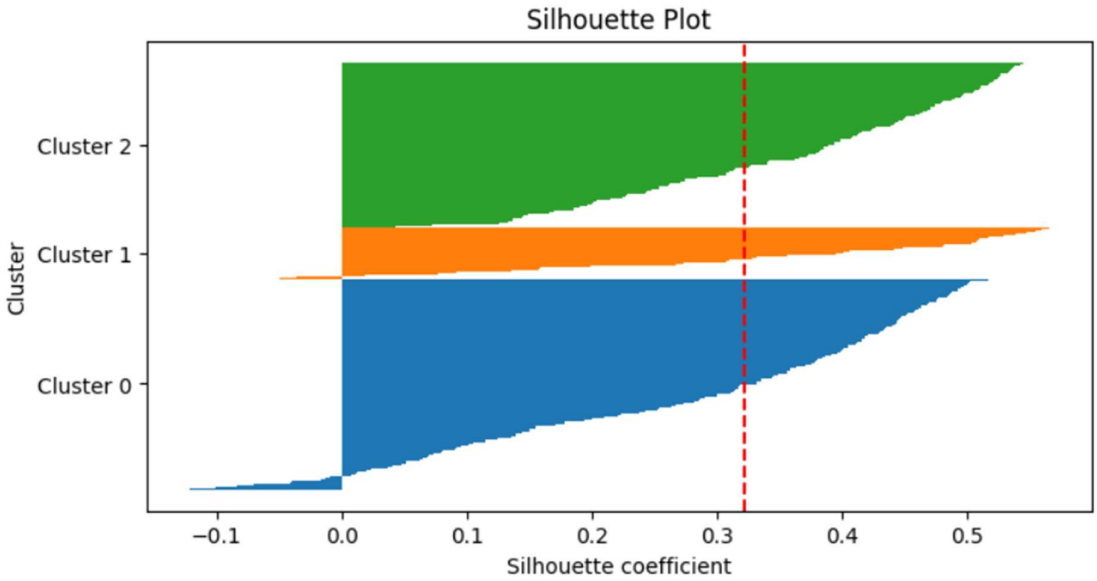

Figure 11: Silhouette plot and average silhouette score

To conclude, the analysis we successfully applied the k-means clustering analysis and identified three distinct clusters, which highlight a certain degree of dependence from economy and commuting differences in relation to their municipality type. Our findings highlight, that the proximity to urban centers can have an important role in economic composition, with rural areas closer to urban centers having a higher commuting percentage, while some more rural areas show more locally centered characteristics due to high service sector employment rates. Lastly, the low influence of the foreign population percentage and the distribution of the third, mixed cluster suggests, that further investigation should be done to detect patterns and explore additional influencing factors.

References

Aggarwal, C. C., Hinneburg, A., & Keim, D. A. (2001). On the surprising behavior of distance metrics in high-dimensional space. In J. Van den Bussche & V. Vianu (Eds.), Database theory — ICDT 2001 (Vol. 1973, pp. 420–434). Springer. https://doi.org/10.1007/3-540-44503-X_27

Backhaus, K., Erichson, B., Gensler, S., Weiber, R., Weiber, T. (2023). Clusteranalyse. In: Multivariate Analysemethoden. Springer Gabler, Wiesbaden. https://doi.org/10.1007/978-3-658-40465-9_8

Dataquest. (2023, January 6). K-means clustering from scratch in Python [Machine learning tutorial] [Video]. YouTube. https://www.youtube.com/watch?v=lX-3nGHDhQg&t=959s

Gao, C. X., Dwyer, D., Zhu, Y., Smith, C. L., Du, L., Filia, K. M., Bayer, J., Menssink, J. M., Wang, T., Bergmeir, C., Wood, S., & Cotton, S. M. (2023). An overview of clustering methods with guidelines for application in mental health research. Psychiatry Research, 327, 115265. https://doi.org/10.1016/j.psychres.2023.115265

Hummelbrunner, R., & Miglbauer, E. (1994). Tourism promotion and potential in peripheral areas: The Austrian case. Journal of Sustainable Tourism, 2(1–2), 41–50. https://doi.org/10.1080/09669589409510682

Jäckle, S. (2017). Neue Trends in den Sozialwissenschaften Innovative Techniken für qualitative und quantitative Forschung. Springer SV. Wiesbaden.

King, R. S. (2015). Cluster analysis and data mining: An introduction. Mercury Learning and Information.

M. A. Mahdi, K. M. Hosny and I. Elhenawy (2021). Scalable Clustering Algorithms for Big Data: A Review. In: IEEE Access, vol. 9, 80015–80027. https://ieeexplore.ieee.org/document/9440980

McDonald, A. (2022, January 27). K-Means clustering algorithm with Python tutorial [Video]. YouTube. https://www.youtube.com/watch?v=iNlZ3IU5Ffw&t=507s

Partridge, M. D., Ali, K. & M. R. Olfert (2010). Rural-to-Urban Commuting: Three Degrees of Integration. In: Growth and Change 41, 2. https://doi.org/10.1111/j.1468-2257.2010.00528.x

Shaikh, P. A., Shaikh, A. A., & Muhammad, F. (2023). Decoding the challenges of promoting decent work in rural and urban labor markets. Pakistan Journal of International Affairs, 6(2). https://doi.org/10.52337/pjia.v6i2.779

Statistik Austria (2021). Urban-Rural-Typologie. Bundesanstalt Statistik Österreich, Wien. https://www.statistik.at/fileadmin/pages/453/urbanRuralTypolgie.pdf

Vaishar, A. & Zapletalová, J. Small towns as centres of rural micro-regions. European Countryside, 2009, Sciendo, vol. 1 no. 2, pp. 70-81. https://doi.org/10.2478/v10091-009-0006-4