# Detecting urban centers through machine learning

Maja Dall'Acqua (maja.dallacqua@unitn.it)

GitHub content (https://github.com/majada14/Urban_centers_detection)

***Abstract*— The current research aims to provide a model to study the urban development of a city through the analysis of geospatial data. Different unsupervised machine learning algorithms were used to identify new urban centers inside the city of Trento. Preliminary results show a partial correspondence between the urban centers detected and the current administrative districts of Trento.**

***Keywords*— Unsupervised machine learning, DBSCAN algorithm, OPTICS algorithm, Trento, Urban studies.**

## I. Introduction

Urbanization is a spatial process by which urban areas tend to expand and the urban population to grow. Such changes have been studied mainly by demographers, sociologists, and economists.

In the last decades the rise of megacities with multiple urban centers has drawn attention to the changing urban structure and its repercussion on the population.

To observe such changes USS (Urban Spatial Structure) studies have developed in the past few years new unsupervised machine learning models that rely on the use of so-called urban data in order to study the social composition of cities, which can provide useful knowledge for the administration and government of the same cities. The merging of machine algorithms and urban data has led to the rise of urbanization and regional studies which have provided some promising results in the past years [1].

Important topics and concepts belonging to urban studies and relevant for the current project are:

- *Districts*: are sections inside a city with administrative boundaries that serves many administrative and jurisdictional functions for the population within its borders [2].
- *Urban centers*: are areas within the city's borders where human activities tend to concentrate. Urban centers serve as critical nodes of the urban system supporting the operations of urban functions [3].
- *Urban functions*: correspond a set of services that the urban center should provide to the urban residents considering the diverse demands.

Considering this definitions, new urban centers can therefore be defined as so when a series of structures an, providing all the urban services necessary, are presents inside the urban center itself.

Through collecting geospatial data and adopting the prospective of urban function distribution [4], the current project aims therefore to answer the following questions:

1. Is it possible of detect urban centers inside a city using machine learning?
2. If so, do the urban centers detected by the algorithms implemented correspond to the administrative districts of a city or not?
3. Which are the implications of the possible correspondence/divergence?

The city of Trento was selected as case study for the research. It is composed of 12 administrative districts, represented in *Figure 1*, which have different boundaries and socio-demographic compositions.

Consistent with recent studies and the composition of the city of Trento, the urban functions selected to be analyzed concern the fields of economy, education, health, tourism, shopping and catering.

*Figure 1: Visualization of Trento's administrative districts.*


## II. Methodology

### A. DATA COLLECTION

Data is represented as geospatial points serving a urban function within the city's administrative borders.

Various queries were run through *Overpass*' API [4] to collect such data from *OpenStreetMap* application [5]. Starting from the city's center, all points satisfying two requirements were retrieved:

1. being inside a certain radius, calculated through the formula:

$$r = \sqrt{\frac{A}{\pi}}$$

2. containing specific tags, which are key-value pairs that allows to extract nodes belonging to the specified features selected.

Each node is therefore represented through the following attributes:

- name: the Italian name of the structure,
- id: the OpenStreetMap unique code,
- coordinates: latitude and longitude,
- class: the urban function.

In the current project data were collected considering 7 km as radius from Trento's center. *Table 1* shows the number of nodes retrieved for each urban function, with a total of 659 elements around the city of Trento.

*Table 1. Distribution of the data collected for each urban function.*

| Urban function | POI description | POI retrieved |
|---|---|---|
| Economic | Banks | 63 |
| Catering | Bar, Cafè, Restaurants, Fast Foods | 400 |
| Education | Primary school, High-school, Universities, Libraries | 58 |
| Shopping | Mall, Supermarkets | 46 |
| Tourism | Museums, Parks, Nightclub | 57 |
| Health | Hospitals, Pharmacies | 35 |

## B. MODEL IMPLEMENTATION

To study the distribution of urban centers around the city two different algorithms were implemented, the DBSCAN (Density Based Spatial Clustering with Noise Application) and the OPTICS (Ordering Points To Identify the Clustering Structure). Both belong to the clustering technique, which is the most established subcategory of unsupervised machine learning [6]. Among others, the DBSCAN and the OPTICS were selected considering the peculiarity of the dataset, since they both can measure the distance between data using non-flat geometry computations.

The DBSCAN model allows to identify an unspecified number of clusters from a pool of data calculating the distance between data points [7]. One peculiarity of the DBSCAN is the so-called noise application, for which data points can also be considered noises and therefore are not used in the computation steps. The DBSCAN is implemented through the selection of the epsilon parameter, which corresponds to the radius inside which searching for close data points, and the selection of the minimum number of members the cluster must be composed of.

The OPTICS model allows to identify core samples of high density in the distribution and to expand clusters from them, constructing a so-called reachability graph [8]. The OPTICS is implemented through the selection of a minimum number of samples, corresponding to the number of elements in a neighborhood for a data point to be considered as a core point, and the minimum cluster size, which is the minimum number of samples required in an OPTICS cluster.

Both algorithms were implemented using the Haversine distance, which is the most known and used function to compute distance between geospatial points. The Haversine distance determines the great-circle distance between point A and point B on a sphere given their coordinates through the formula:

$$haversine(\theta) = haversine(lat_B - lat_A) + \cos(lat_A)\cos(lat_B)\, haversine(lon_B - lon_A)$$

Such distance was computed with different methods by the two algorithms. Indeed, while the DBSCAN algorithm automatically takes as input latitude and longitude for each data point and can compute the Haversine distance by itself, in the OPTICS implementation the same computation is not available. Thus, it was required to construct a distance matrix containing the Haversine distances between all data points as input for the OPTICS algorithm.

Considering the multiple datasets, fine tuning the parameters of the DBSCAN and the OPTICS algorithms required different implementations based on the domain. Consistent with previous studies and the composition of Trento, a small pool of values for each parameter was picked as starting point for training the two models. Small values (0.2 – 0.9) for the epsilon parameter were chosen, since the area analyzed is quite delimited. At the same time, small values (2 – 10) were selected as minimum number of components for the clusters.

For each dataset the two algorithms were tested using all the possible combination of parameters. Every model's result was then evaluated through selected evaluation measures.

### C. MODEL EVALUATION

To evaluate both models' performances two measures were implemented:

- the Silhouette Coefficient: it is obtained computing the mean of the Silhouette coefficient of all samples [9], each defined as:

$$\frac{(b-a)}{max(a,b)},$$

where $a$ corresponds to the intra-cluster distance and $b$ corresponds to the nearest-cluster distance.
The score ranges between -1 and 1, where 0 indicates overlapping clusters, negative values indicate that a sample has been assigned to the wrong cluster, while positive values indicate the opposite.

- the Davies Bouldin score: it is the average similarity measure of each cluster with its most similar cluster, where the similarity is computed as:

$$\frac{within-cluster\ distances}{between-cluster\ distances}$$

For low values (as close as possible to zero) the clustering is producing well-separated clusters [10].

For each model trained, both these evaluation measures were calculated and then stored in a temporary variable. Among this pool, the results were first filtered considering the range of clusters identified by the algorithm. Indeed, to compare the administrative districts with the clusters' distribution provided by the models, only the combinations identifying a number of clusters in the range of 7 (12 administrative districts -5) and 17 (12 administrative districts + 5) were considered. From this skimming process, the best results stored were then picked considering the best combination of Silhouette Coefficient and Davies Bouldin Score.

## III. Results

Overall, both algorithms provide quite interesting results, although with some differences.

As one can see in *Table 3* the optimal parameters for both algorithms vary among the different datasets.

In both models the minimum number of data points necessary to construct a cluster is two, except for the catering dataset, where such parameter is a little bit higher in the DBSCAN model and a lot higher in the OPTICS model. This could be explained considering that the size of that specific dataset is indeed bigger than the others.

The epsilon parameter in the DBSCAN implementation quite vary across all datasets in a range between 0.2 and 0.7, and the best combination in each dataset usually required the 0.5 value for the epsilon.

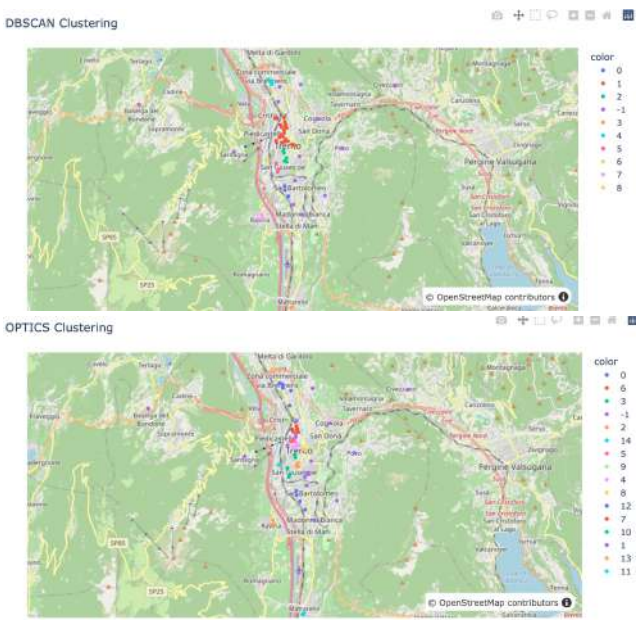
Moreover, *Table 3* shows the best results (with a green background) for each dataset, selected considering both the evaluation metrics and the visualization of the models. In the following sections for each dataset the best combination of the DBSCAN and the OPTICS will be analyzed and compared to the administrative division of Trento.

### A. ECONOMIC DATASET

As one can see in *Figure 3*, the number of clusters identified by the OPTICS models is bigger than one in the DBSCAN model. Such difference emerges specifically in the city center's division, where the OPTICS can identify different clusters inside the Centro storico area. Nonetheless, both algorithm success into identifying the districts of Mattarello, Ravina, Gardolo, Centro – Storico, Santa Chiara and Oltrefersina, while they fail to identify the farthest districts, such as Meano, Povo and Bondone.

Both algorithms seem to identify a new cluster (cluster number 8 in the DBSCAN, cluster number 11 in the OPTICS represented in *Figure 3*) corresponding to Le Albere neighborhoods in the city of Trento.

*Figure 2: visualization of the clusters identified in the economic dataset by the DBSCAN and the OPTICS algorithms respectively.*

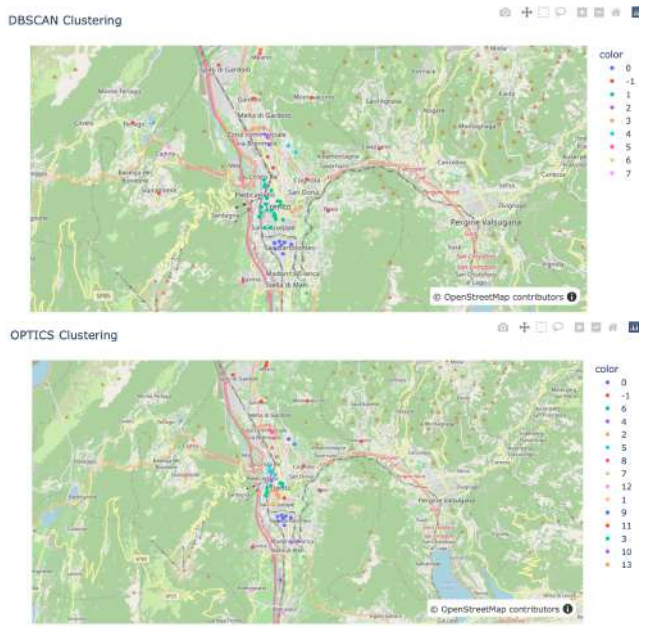



### B. EDUCATION DATASET

In the education dataset the two algorithms differ quite evidently. Indeed, the OPTICS identifies a cluster both in Mattarello and Oltrefersina's districts, while the DBSCAN does not. Both algorithms identify the districts

of Argentario, Meano, Bondone, Centro storico and Santa Chiara. Though in both cases the data near Gardolo are considered as noises (cluster -1), the area corresponding to Via Brennero is identified as a cluster (cluster number 2 in the DBSCAN, cluster number 8 in the OPTICS represented in *Figure 4*). As for the previous dataset, the city's center division appears to be more detailed in the OPTICS algorithm, where more clusters are indeed identified.

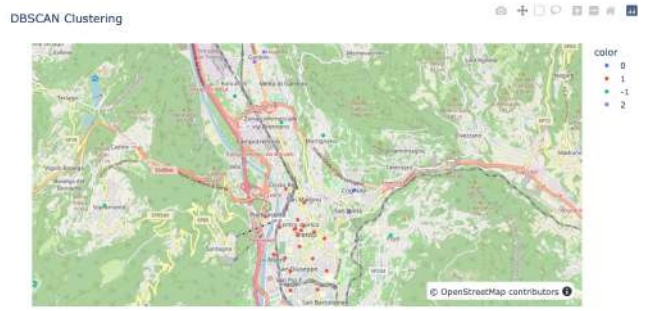
*Figure 3: visualization of the clusters identified in the education dataset by the DBSCAN and the OPTICS algorithms respectively.*

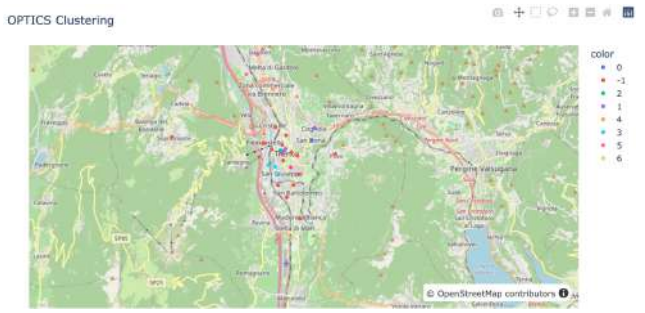



## C. HEALTH DATASET

In the current dataset the OPTICS model provides better results than the DBSCAN. Indeed, while both success into identifying three administrative clusters, corresponding to Centro storico, Gardolo and Argentario, only the former can identify some additional clusters near the area of Mattarello, Le Albere and Via Brennero (respectively clusters number 6, 3 and 4 represented in the OPTICS clustering of *Figure 5*).

*Figure 4: visualization of the clusters identified in the health dataset by the DBSCAN and the OPTICS algorithms respectively.*



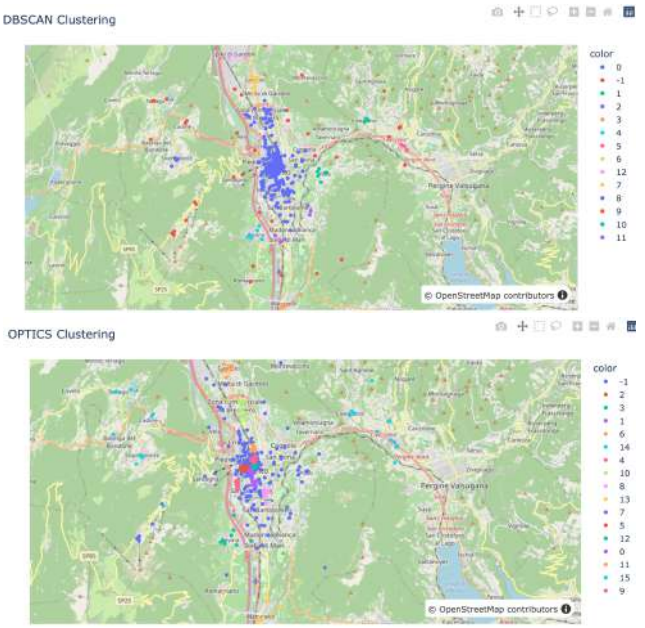




## D. CATERING DATASET

For the catering dataset both algorithms provide interesting results, with similarities and differences.

For starting, the two models identify a similar number of clusters. Secondly, both algorithms success into identifying most of Trento's administrative districts, as Bondone, Gardolo, Mattarello, Villazzano, Ravina and Povo. Similarly, both models identify some clusters in area of Civezzano (clusters number 7 and 12 in the DBSCAN, cluster number 14 in the OPTICS, represented in *Figure 6*), while only the OPTICS implementation identifies a cluster near Via Brennero (cluster number 10 in the OPTICS clustering represented in *Figure 6*).

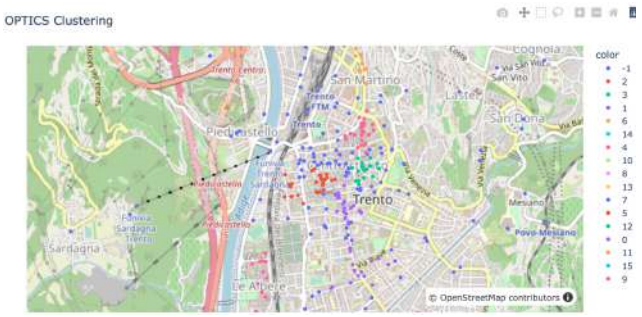
*Figure 5: visualization of the clusters identified in the catering dataset by the DBSCAN and the OPTICS algorithms respectively.*




Considering the high number of data points in this dataset, an insight on the cluster distribution inside the city's center was considered necessary for the OPTICS implementation. Indeed, *Figure 7* shows a clear division of the data points, tracing the neighborhoods division of the city center. From this insight it is possible to see how the OPTICS identifies the area of San Martino (cluster number 6), Piazza Duomo (cluster number 2), Piazza

Santa Maria Maddalena (cluster number 3), Piazza Fiera and Corso 3 Novembre (cluster number 1), and Le Albere (cluster number 9).

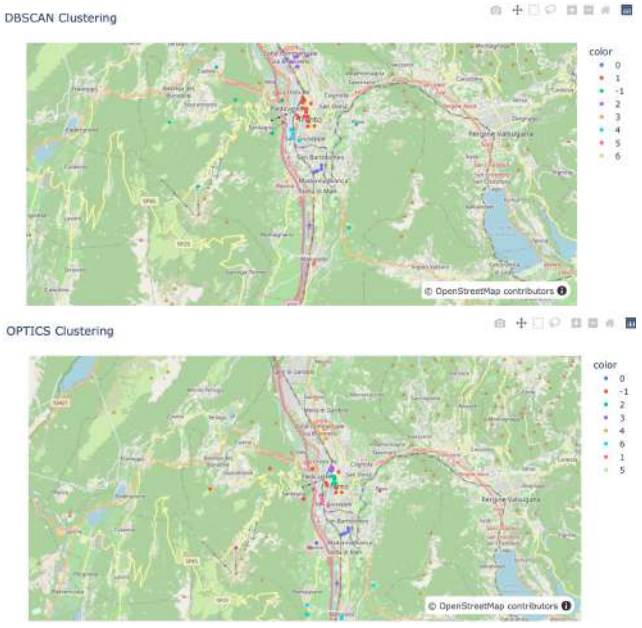
*Figure 6: visualization of the clusters inside the city center identified by the OPTICS algorithm.*




## E. TOURISM DATASET

In the tourism dataset the algorithms provided the same number of clusters identified, though with some differences. Both the DBSCAN and the OPTICS identify the district of Gardolo, although the latter consider Gardolo and Meano as a single cluster, Centro storico, Mattarello and Oltefersina. Moreover, in the DBSCAN it is also possible to see a cluster corresponding to Ravina's district (cluster number 3 in the DBSCAN model represented in *Figure 8*). Both algorithms identify a cluster corresponding to the area of Le Albere (cluster number 4 in the DBSCAN, cluster number 1 in the OPTICS represented in *Figure 8*).

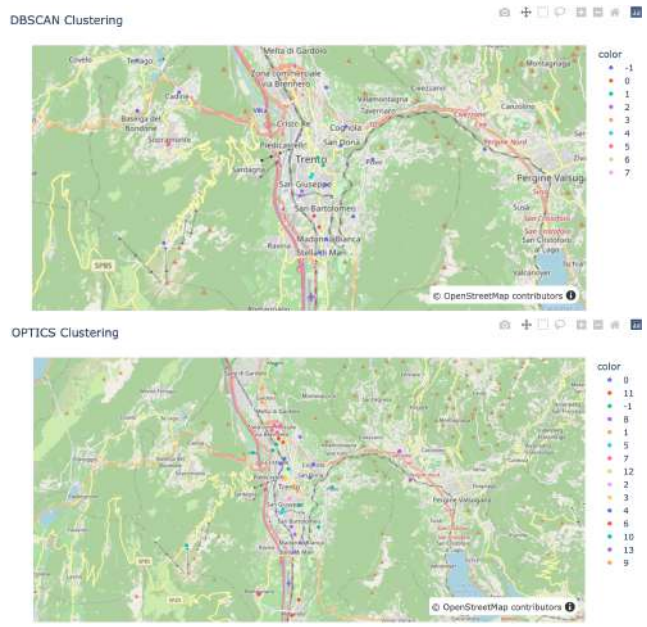
*Figure 7: visualization of the clusters identified in the tourism dataset by the DBSCAN and the OPTICS algorithms, respectively.*




## F. SHOPPING DATASET

In the shopping dataset the results provided by the two algorithms are consistent with the results achieved in the previous datasets. Both algorithms successfully identify clusters corresponding to the districts of Gardolo, Bondone, Centro storico, Oltrefersina and Santa Chiara. At the same time, the two models provide a clear division of the city's center into many clusters corresponding to the city's neighborhoods and identify a cluster in Civezzano (cluster number 7 in the DBSCAN, cluster number 13 in the OPTICS represented in *Figure 9*).

*Figure 8: visualization of the clusters identified in the shopping dataset by the DBSCAN and the OPTICS algorithms respectively.*




# IV. Conclusions

The results provided from the different datasets can serve as useful inputs to draw a complete analysis of the city of Trento with respect to the research questions of the current project.

In all the dataset both the algorithms implemented were able to identify many of the administrative districts of Trento, although with some exceptions. Indeed, the districts of Villazzano and Sardagna never appeared as clusters in any of the datasets. Such discrepancy could be related to some of the problems encountered in the implementation of the project that will be discussed further on.

Consistently with previous studies, the results for the current project seem to confirm that unsupervised machine learning can serve as a useful instrument to study the segmentation of urban areas [11], though it requires new studies and implementations to be as accurate as possible.

Regarding the city of Trento, it is quite interesting how both the algorithms implemented were able to detect in most of the datasets some clusters non corresponding to the administrative districts.

Indeed, Via Brennero and Le Albere have emerged as clusters in most of the datasets, which could mean that

both can satisfy all the urban function required. Although it is not possible to state whether these areas are more similar (in dimension and composition) to a neighborhood or to a district, it is evident that urban processes are shaping the city of Trento in new ways. Further analyses are necessary to explore the consequential implications of the birth of new districts, considering additional factors and variables.

The main challenges of this research appertained to the quality and size of the data and the interpretation of the results [12], which are both common problems in urban studies since unsupervised methods highly rely on the intrinsic structure and quality of the input.

The quality and quantity of data points used in the current project was not optimal due to two main elements. The former regards the fact that only few specific queries were used to collect the data, while the latter regards the adoption of a cleaning process not limiting.

The combination of the two has led to the construction of small datasets with a non-optimal quality level.

A better approach would have been collecting all nodes available from *OpenStreetMap* and then apply an intense and multiple-steps cleaning process, that at the same time would have been time consuming and would have required a high level of human control to select all the possible tags among the available ones from the *OpenStreetMap* dataset.

A second common problem concerns the interpretation of the results. Indeed, to provide a correct interpretation of the data it is necessary for the researchers to have a good knowledge of the city studied, its composition and socio-demographics. Such information are useful to select the hyper-parameters of the machine learning algorithms and in the interpretation of the results.

In conclusion, the current project provides some interesting results to understand the current composition of Trento's urban centers and to capture hidden urban patterns. Further analyses could involve the use of longitudinal data to observe the composition of urban centers in the same city across time.

## APPENDIX

## REFERENCES AMA STYLE


[1] J. Wang, F. Biljecki, *Unsupervised machine learning in urban studies: A systematic review of applications*, *Cities,* Volume 129, 2022.

[2] Enciclopedia Treccani official website (https://www.treccani.it/enciclopedia/circoscrizione_(Enciclopedia-Italiana)/, last visited 31/08/23)

[3] Yu L, Yu T, Wu Y, Wu G. Rethinking the Identification of Urban Centers from the Perspective of Function Distribution: A Framework Based on Point-of-Interest Data. *Sustainability*. 2020; 12(4):1543. https://doi.org/10.3390/su12041543

[4] Overpass-Turbo official website (https://overpass-turbo.eu, last visited 31/08/23)

[5] OpenStreetMap official website (https://www.openstreetmap.org/#map=11/46.0759/11.2081, last visited 31/08/23)

[6] J. Wang, F. Biljecki, *Unsupervised machine learning in urban studies: A systematic review of applications*, *Cities,* Volume 129, 2022.

[7] Scikit-learn official website (https://scikit-learn.org/stable/modules/clustering.html#dbscan, last visited 31/08/23)

[8] Scikit-learn official website (https://scikit-learn.org/stable/modules/clustering.html#optics, last visited 31/08/23)

[9] Scikit-learn official website (https://scikit-learn.org/stable/modules/generated/sklearn.metrics.silhouette_score.html#sklearn.metrics.silhouette_score, last visited 31/08/23)

[10] Scikit-learn official website (https://scikit-learn.org/stable/modules/generated/sklearn.metrics.davies_bouldin_score.html, last visited 31/08/23)

[11] Municipality of Trento official website (https://www.comune.trento.it/Comune/Organi-politici/Circoscrizioni, last visited 31/08/23)

[12] J. Wang, F. Biljecki, *Unsupervised machine learning in urban studies: A systematic review of applications*, *Cities,* Volume 129, 2022.


*Table 3: comparison between the results for the implementation of the DSCAN and the OPTICS algorithms on the different datasets.*

| | DBSCAN implementation | | | | OPTICS implementation | | | |
|---|---|---|---|---|---|---|---|---|
| | Parameters | | Evaluation metrics | | Parameters | | Evaluation metrics | |
| | Eps | Min Parts | Silhouette | DB score | Min sample | Min cluster size | Silhouette | DB score |
| **Economic dataset** | | | | | | | | |
| | 0.2 | 2 | 0.25 | 3.62 | 2 | 2 | 0.34 | 3.85 |
| | 0.3 | 2 | 0.32 | 2.99 | 2 | 3 | 0.12 | 3.63 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0.3 | 3 | 0.22 | 6.05 | 3 | 2 | 0.17 | 4.30 |
| **Education dataset** | | | | | | | |
| 0.3 | 2 | 0.08 | 4.76 | 2 | 2 | 0.36 | 1.22 |
| 0.5 | 2 | 0.28 | 2.47 | 2 | 3 | 0.22 | 4.31 |
| 0.6 | 2 | 0.27 | 2.83 | 3 | 3 | 0.19 | 2.95 |
| **Health dataset** | | | | | | | |
| 0.4 | 2 | -0.06 | 5.60 | 2 | 2 | -0.04 | 6.20 |
| 0.6 | 2 | 0.23 | 4.75 | - | - | - | - |
| 0.7 | 2 | 0.24 | 5.29 | - | - | - | - |
| **Shopping dataset** | | | | | | | |
| 0.4 | 2 | -0.02 | 3.72 | 2 | 2 | 0.36 | 2.0 |
| 0.5 | 2 | 0.04 | 3.12 | 2 | 3 | 0.12 | 2.09 |
| - | - | - | - | 3 | 2 | 0.17 | 5.79 |
| **Tourism dataset** | | | | | | | |
| 0.5 | 2 | 0.46 | 1.10 | 2 | 2 | 0.27 | 1.57 |
| 0.6 | 2 | 0.46 | 1.24 | 3 | 2 | 0.18 | 1.57 |
| 0.7 | 2 | 0.47 | 1.20 | 4 | 2 | 0.29 | 1.65 |
| **Catering dataset** | | | | | | | |
| 0.5 | 3 | 0.38 | 1.93 | 8 | 7 | 0.04 | 3.14 |
| 0.6 | 3 | 0.39 | 2.69 | 8 | 8 | 0.04 | 3.14 |
| 0.7 | 3 | 0.37 | 2.26 | 8 | 9 | 0.0 | 2.83 |