Information Fusion 104 (2024) 102164

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Clustering pipeline for vehicle behavior in smart villages

Daniel Bolaños-Martinez *, Maria Bermudez-Edo, Jose Luis Garrido

University of Granada, C/Pdta. Saucedo Aranda s/n, 18014, Granada, Spain

ARTICLE INFO

MSC:
62H30
68T05

Keywords:
Internet of Things (IoT)
Sensors
Clustering
Smart villages
Explainability

ABSTRACT

Smart cities and villages present a plethora of opportunities for fusing and managing multi-source data. However, in the analysis of mobility patterns, the use of only one data source (i.e., road sensors) without considering other contextual data sources, limits the understanding of the process. To address this gap, we propose a pipeline that integrates multiple data sources, providing valuable information for pattern extraction, mainly based on vehicle mobility behavior and provenance. Our research also highlights the critical role of selecting the appropriate normalization algorithm to scale input features from heterogeneous data sources, which has not received sufficient attention in the literature. We conducted our analysis using data from four License Plate Recognition (LPR) cameras, spanning nine months, and incorporating several databases that include provenance, gross income, and holiday information, resulting in a dataset of over 50,000 vehicles. Using this data and our clustering pipeline, we identified various traffic patterns among residents and visitors in a rural touristic area. Our findings assist data analysts in choosing algorithms for analyzing heterogeneous datasets. Moreover, policymakers could use our results to adjust the resources, such as new parking zones.

1. Introduction

Currently, there are 13.4 billion Internet of Things (IoT) devices. Statista predicted that this figure will increase to 29.4 billion by 2030.[1] These devices form an interconnected network that produces extensive data in numerous social domains. Access to a large volume of data collected by various sensors makes it possible to supervise and manage different aspects of society, including evacuation systems, smart environments, and transportation [1–4]. This trend boosted cities to deploy sensor networks and IoT platforms, for example, to monitor the flow of vehicles on their roads. The data obtained by these sensors have led to numerous studies in several areas, such as traffic behavior [5–8]. Extracting and combining information from multiple sources, not only sensor data, but also information stored on the Internet, can lead to a better understanding of the problem to be solved. For instance, traffic in cities is partially dependent on local holidays. Some approaches have enhanced the analysis of traffic data (from vehicle counter sensors) with context information to understand the traffic conditions on roads using events data, parking information, or weather conditions [9,10].

However, most solutions using License Plate Recognition (LPR) sensors [11,12] did not use additional contextual datasets. Only few works combine LPR with location information [13,14], but none of them include other contextual information. They also did not explore calculated variables that enhance the raw data, such as distance traveled or visit frequency. Furthermore, in traffic analysis works [15], and in ML pipelines, in general, [16,17], the normalization stage was understudied. They usually apply one normalization method without studying the suitability of that method. Moreover, the smart city trend had yet to reach villages, as the solutions found for large cities did not always apply directly to small villages. For example, solutions monitoring traffic behavior in large cities with numerous streets and several traffic lines in some avenues do not extrapolate to villages with mostly pedestrian streets and just one road with a single lane in each direction. Additionally, even if we try to add some explanation to the behavioral cluster in smart villages, the residency of vehicle owners is not straightforward. Recent movements of people relocating from cities to villages or spending extended periods in second residences have made actual residency information unclear in rural areas.

The contribution of this article was twofold. First, we explored the integration of LPR sensor data with contextual information from multiple sources (such as holidays, provenance, or demographic information). One of these sources incorporated data on the origin of each vehicle, which could enhance the results by adding the economic status of the region of origin or the distance traveled to reach the area. Second, we conducted an exploration of different normalization algorithms. To achieve that, we utilized various visualization tools to determine the optimal algorithms based on empirical tests.

In particular, this paper proposes a clustering pipeline based on vehicle behavior in small villages, with information from license plate

* Corresponding author.
E-mail address: danibolanos@ugr.es (D. Bolaños-Martinez).

[1] https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/

https://doi.org/10.1016/j.inffus.2023.102164
Received 28 April 2023; Received in revised form 27 November 2023; Accepted 28 November 2023
Available online 29 November 2023

recognition (LPR) devices and contextual information, such as owners' residence location. We applied the study directly to each individual (vehicle) and defined their spatio-temporal behavior based on their spatial frequencies of visitation. To that end, we fused several datasets and calculated new valuable variables such as the time spent in the area; total distance traveled there, etc. Our pipeline comprised eight steps: data collection, cleaning, fusion, normalization, dimensionality reduction, clustering, evaluation, and visualization. We applied the proposed pipeline to a touristic rural region, with the problems mentioned above of a single small road and the lack of reliable residency information. We paid special attention to optimizing the normalization algorithms to our data. Furthermore, we analyzed the results with residential information and identified the variables that had the most influence on each cluster. With this information, we explained the behavior patterns of each cluster.

Our results are useful for policymakers to improve tourism policy and bring benefits to the area. For example, policymakers could tailor parking fees in the area by identifying visitor clusters and their average stay duration. They could also designate schoolyards or streets for parking during peak tourist periods to reduce road congestion. This work is also useful for developers and data scientists to formalize and choose the clustering and normalization algorithms for their analyses.

The remainder of the paper is organized as follows. In Section 2 related work is summarized. Section 3 presents the theoretical bases discussed throughout the paper, describing the main normalization and clustering algorithms and metrics. Section 4 presents the unsupervised learning pipeline, including a background of the use case, the sensor setup, and the different sources of information used for the construction of the dataset, and Sections 5 and 6 show the analysis and discussion of the results. Finally, Section 7 concludes the paper.

2. Related work

The concept of information fusion has been applied to the specific problem of tourism flows and smart cities. These approaches used data analysis techniques to combine multiple sources of information, providing valuable insights for developing smart tourism applications in cities and designing sustainable environments. Smart city applications were built on top of data, and data fusion provided a wide variety of techniques to improve the input data for an application [18]. Examples of these techniques included data association, state estimation, unsupervised machine learning, or statistical inference. For example, combining different tourist information was used to predict the tourist flow with graph neural networks [19]. The data used in the solution were composed of tourist infrastructure information, such as camping and tourist housing from OpenStreetMap and the National Statistics Institute (Spanish: Instituto Nacional de Estadistica, INE); reports released by the Spanish Ministry of Transportation (SMT); and human mobility data, including the number of movements between administrative areas per hour extracted from geotagged Twitter data. Most of these applications were focused either on user recommendations or tourist flow, but little attention was paid to studying the individual behavior of the tourist inside an area (for a detailed survey, see [18,20]).

The increasing deployment of IoT platforms in smart cities has boosted the proliferation of sensors, including those that monitor traffic. These sensor data allow us to analyze vehicle behavior. The most common works in this area were to analyze mobility patterns in order to improve traffic congestion [5,7], and to aggregate vehicles to obtain useful conclusions for urban management [6,21].

To infer mobility patterns from raw data, unsupervised ML has been widely adopted. Clustering analysis was used to detect behavioral patterns in the field of pedestrian-vehicle mobility, and in the field of indoor-outdoor (IO) positioning systems [22]. Algorithms such as GaussianMixture were used to perform segment analysis, where individuals were defined by their movement routines, and the data was related to the frequency and period of stay in different areas. From the movement information provided by smart cards, several papers applied this algorithm to identify market segments based on temporal travel patterns [23], defined tourist patterns based on frequency and areas where transactions were made [24], or identified changes in functional areas of cities over time [25].

Some studies highlighted the importance of employing normalization techniques, such as in the context of time series analysis [26,27]. In the field of pattern extraction, and specifically in other clustering frameworks, some works use one normalization [15–17]. However, to the best of our knowledge, no work has studied the influence of using different normalization algorithms.

Few works related to clustering analysis in mobility use LPR cameras as the main source of information [28]. For example, [28] analyzed commuting patterns by constructing the spatio-temporal similarity matrix using the Dynamic Time Warping (DTW) algorithm and subsequently analyzed the characteristics of commuting patterns with the density-based spatial clustering of applications with noise (DBSCAN) algorithm. Similarly, [12] analyzed the change in traffic patterns during the pandemic using K-Means. However, none of these works combined LPR data with vehicle provenance nor studied the touristic behavior of the vehicle.

3. Fundamentals

In clustering pipelines, besides choosing the right algorithm and evaluation metrics, sometimes other analyses are needed. For example, to analyze attributes in different scales, such as nights ranging from 0 to 269 and gross income per capita from 12,638 to 79,327, we had to normalize them first. Sometimes, it is worth reducing the dimensionality to simplify the data matrix and facilitate their understanding by the human mind [29]. The most used dimensionality reduction algorithm is Principal Component Analysis (PCA), and it can be used with at least five variables and five samples [30]. Data distributions come in various shapes (scattered, curved, flat), and understanding this geometry can help choose appropriate clustering algorithms.

3.1. Main clustering algorithms

Unsupervised machine learning automates the knowledge discovery process without needing labeled or previously classified data [18]. Most taxonomies group the algorithms into at least five categories [31], although we have identified seven, as some of them did not fit in the 5 elements taxonomy:

Partitional Clustering: decomposes a dataset into distinct clusters through an iterative process of distance calculations between individuals.

Hierarchical Clustering: constructs clusters in either an agglomerative or divisive manner by adding or removing individuals, respectively.

Density-based Clustering: identifies dense regions of objects in the data space separated by low-density regions.

Distribution-based Clustering: creates clusters based on the probability that each individual belongs to the same distribution.

Grid-based Clustering: divides the space into a finite number of cells.

Message-Passing Clustering: creates clusters by exchanging messages between different data points until convergence.

Spectral Clustering: uses the spectral radius of a similarity matrix of the data in a multidimensional problem.

Table 1 shows the main algorithms in each category described in this section, and examples of applications for each algorithm, in the field of mobility pattern analysis in the last three years (2020–2023).

Table 1
Examples of works using clustering to infer mobility pattern in 2020–2023.

Clustering category	Algorithms	Application	Related work
Partitional	K-Means, MiniBatchKMeans, ISODATA	Target classes, analyze patterns	[12,15,32]
Hierarchical	Agglomerative clustering, Divisive clustering, BIRCH	Behavioral patterns, feature extraction	[33–35]
Density-based	DBSCAN, OPTICS, HDBSCAN, MeanShift	Complexity reduction, anomaly detection	[7,28,36,37]
Distribution-based	Gaussian Mixture	Density estimation, outlier detection	[23–25]
Grid-based	STING, WaveCluster, CLIQUE	Spatial-based segmentation	Not found
Message passing-based	Affinity Propagation, IWC-KAP, ScaleAP	Clustering indoor location patterns	[38–40]
Spectral	Spectral Clustering, ASC	Graph partitioning, image segmentation	[41–43]

3.2. Clustering performance

The three most popular internal evaluation metrics in the literature [44] are silhouette coefficient, Calinski–Harabasz score, and Davies–Bouldin index. All of these metrics are based on distances between data points and are commonly used to evaluate the effectiveness of virtually any clustering algorithm, working especially well in algorithms that work with distances, such as those included in the hierarchical, partitional, or spectral categories.

These distance-based metrics may not be suitable for algorithms that use the Expectation Maximization (EM) method, such as the GaussianMixture algorithm. This is because the EM method models the data using probability distributions rather than distances between data points. Therefore, we might get some imprecision when comparing the performance of algorithms of this type if we use these metrics. Instead of using distance-based metrics, distribution-based algorithms typically use statistical criteria to determine the optimal number of clusters or components that best fit the data [45]. One of these metrics is the information criterion (IC), which measures how well a statistical model fits the data distribution while penalizing overfitting [46].

$$IC(k) = -2 \cdot L(\hat{\theta}_k) + c_N \cdot k \tag{1}$$

where $\hat{\theta}_k$ is the estimator of the parameter vector relating to the mixture model with order k, L the log-likelihood function, N the number of observations, and c_N an increasing function of N. The optimal number of clusters is the one that minimizes the IC.

The following are two of the best-known variations of information criteria used in the literature [47]:

- **Akaike information criterion** (AIC): AIC is a particular specification of the general information criterion (IC), in which $c_N = 2$. This criterion is known to overestimate the order of the model.

$$AIC(k) = -2 \cdot L(\hat{\theta}_k) + 2 \cdot k \tag{2}$$

- **Bayesian information criterion** (BIC): Tries to overcome the overestimate of AIC. The penalty term depends on the sample size N, so as $N \to \infty$ the penalty is larger and does not overestimate the order of the mixture as much as AIC does [48].

$$BIC(k) = -2 \cdot L(\hat{\theta}_k) + \log N \cdot k \tag{3}$$

3.3. Principal component analysis

The Principal Component Analysis (PCA) method condenses the information provided by multiple variables $(X_1, \ldots, X_p)$ from a given sample into a smaller number of variables, finding a number s of underlying factors that explain approximately the same variance as the original variables with $s < p$. Each of the new variables $(Z_1, \ldots, Z_p)$ are called principal components, which are linear combinations of the original variables. We define each Z_i as:

$$Z_i = \Phi_{1i} X_1 + \Phi_{2i} X_2 + \cdots + \Phi_{pi} X_p \tag{4}$$

Each Φ represents the weight or importance that each variable X_i has in each Z_i and, explains the information collected by each of the principal components [49]. It is advisable to apply prior normalization to the data, since this method is highly sensitive to variables of different scales. Furthermore, the PCA only works with numerical data, so it is necessary to perform a previous preprocessing on categorical variables that may exist in the input dataset [50].

3.4. Normalization

Normalization compresses or expands the values of each variable to fit them in the same range of values, normally [0,1], or [−1, 1], making them comparable in subsequent processes (PCA or ML algorithms). The choice of the normalization algorithm usually depends on the specific application and the dataset used, as different methods may yield different results and interpretations. For example, in clustering analysis, normalization can be particularly important for comparing similarities between characteristics based on certain distance measures. Among the most commonly used normalization methods are min–max normalization and z-score standardization [51,52]. We have also tested two other methods that are commonly used in the literature [53,54] and occasionally produce better results than min–max or z-score.

1. **Min–max normalization:** Uses the minimum and maximum in the attribute domain to normalize the values to the interval, $[0, 1]$ keeping the distances for each data point X.

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{5}$$

2. **Z-score standardization:** scales the values so that the mean (μ) of the data domain is 0 and the standard deviation (σ) is equal to 1.

$$X' = \frac{X - \mu}{\sigma} \tag{6}$$

3. **Median Absolute Deviation (MAD) normalization:** normalizes the data such that the median of each attribute is 0 and the median absolute deviation is equal to 1.

$$X' = \frac{X - median(X)}{MAD(X)} \tag{7}$$

Where $median(X)$ is the median of the values in attribute X, and $MAD(X)$ is the median absolute deviation of X.

4. ℓ^2 **normalization:** normalizes the data by dividing it by its Euclidean norm. This ensures that all feature vectors have the same length and is commonly used in machine learning and information retrieval. The formula for ℓ^2 normalization is shown below:

$$X' = \frac{X}{\|X\|_2} \tag{8}$$

Where $\|X\|_2$ is the Euclidean norm of, X given by $\sqrt{\sum_{i=1}^{n} X_i^2}$.

3.5. Dataset geometry

In data analysis, we refer to flat and non-flat geometry as the measurement of distances between points by Euclidean or non-Euclidean geometric methods, respectively. In flat geometry, the distance is measured following a straight line between two points, while in non-flat geometry, the distance is measured following a curve. We can detect whether our data follow flat or non-flat geometry by representing the data in a scatter plot, where each point represents an individual in the population. Visually we can only represent 3 dimensions, which normally are the most representative variables of the cluster, or the firsts principal components of a dimensional reduction algorithm. If

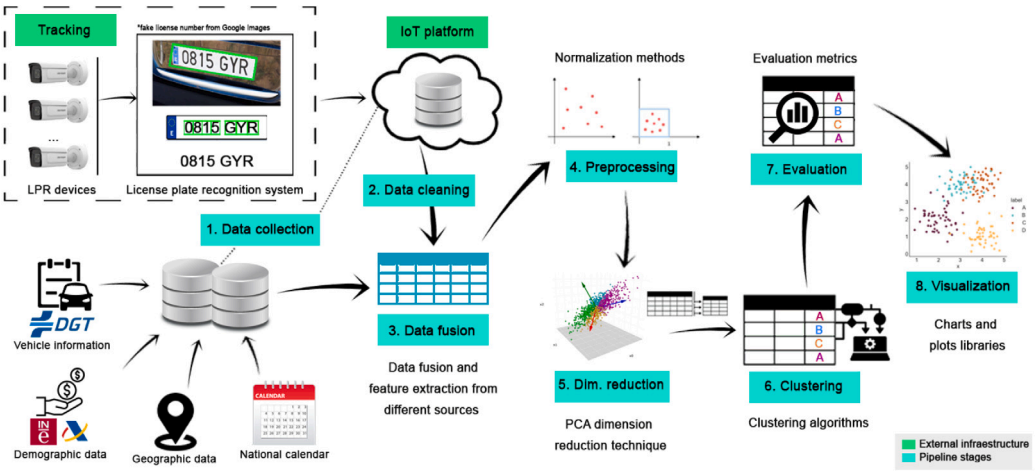


Fig. 1. Overview of the clustering pipeline.

Table 2
Configuration of each stage of the pipeline with the values used in this study.

Stage	Configuration parameters	Experimental values
Data Collection	Data collection from different sources	Storage in own BD and external IoT platform
Data Cleaning	Recovery and treatment of lost data	1. License plate matching 2. Recover movement of vehicles not detected by any camera in their total route
Data Fusion	Fusion of information data and feature extraction	Detailed process in Table 3
Preprocessing	Normalization methods	Min–max normalization, z-score standardization, MAD normalization, ℓ^2 normalization
Dimension reduction	Dimension reduction techniques	Principal Component Analysis (PCA)
Clustering	Clustering algorithms	K-Means, MiniBatchKMeans, Agglomerative clustering, BIRCH, DBSCAN, HDBSCAN, MeanShift, Gaussian Mixture, Spectral Clustering
Evaluation	Evaluation metrics	Silhouette, Davies–Bouldin, Calinski–Harabasz, number of clusters, Bayesian Information Criterion,Akaike Information Criterion
Visualization	Visualization plots	box plot, scatter-plot, elbow method, PCA variance plot

the resulting figure shows a roughly circular, rectangular, or elliptical shape, the data are likely to follow a flat geometry. However, if the figure has an irregular, twisted, or folded shape, the data are likely to follow a non-flat geometry. From different studies [55], it has been found that partitional or distribution category clustering algorithms work best with data cases that follow a flat geometry, while density-based and message-passing algorithms work best with non-flat geometries.[2]

4. Clustering pipeline

We designed an information fusion pipeline to analyze vehicle behavior that divides the analysis into different stages. In general, the pipeline begins with extracting and collecting data from heterogeneous sources and finally produces a grouping result from a clustering model based on the decisions made along the pipeline (see in Fig. 1). Table 2, describes the different stages proposed in the pipeline and the experimental values considered in each stage. The pipeline consists of the following stages: data collection, data cleaning, data fusion, preprocessing, dimension reduction, clustering, evaluation, and visualization.

4.1. Background

Recent years have seen a growing trend of urban exodus, with many people leaving the cities searching for a quieter life. This trend has been boosted by COVID-19 [56]. With the rise of telecommuting, this trend is likely to continue in the future. These migratory flows include both foreign immigrants and the arrival of resident citizens from other parts of the country [57]. In our use case, we take data from 3 small villages in the Alpujarra, an area close to a national park, and attracting tourists from diverse backgrounds [58]. It is especially favored by local and foreign retirees and "neo-rurals", individuals drawn by environmental concerns or a quieter lifestyle, often becoming residents for extended periods [59]. These groups, referred to as "false residents" [60] or non-registered residents, maintain their vehicle registrations from previous residences. Understanding the patterns of the vehicles in the zone is the first step to generating suitable policies to preserve the area's sustainability.

4.2. Data collection

The main source of information for our work was the vehicle tracking system, particularly the license plate recognition (LPR) cameras. The data were collected by four Hikvision LPR IP devices with Automatic number-plate recognition (ANPR) based on Deep Learning. The devices have a 2MP resolution, 2.8–12 mm varifocal optics, and IR LEDs with a range of 50 m.

To cover the entrances and exits of each village in the target area, we strategically positioned the four cameras, as shown in Fig. 2. The locations were (i) entrance to Pampaneira from the western part of the Alpujarra, (ii) entrance to Pampaneira from the eastern part of the Alpujarra, (iii) entrance to Bubión via a single road, and (iv) entrance to Capileira via a single road. By taking advantage of the road structure, we could monitor the mobility of all vehicles in the area using only

[2] https://scikit-learn.org/stable/modules/clustering.html

Fig. 2. Setup of the 4 LPR that obtain the data from the license plates of the vehicles.

four LPRs, minimizing the cost and complexity of the system. The information collected by the cameras was stored on a cloud platform. The rest of the data were collected from different datasets described in Section 4.4.

4.3. Data cleaning

In the field of the IoT, the production of sensor data can often be inaccurate and lead to the loss of some records. In our case, we presented two cleaning steps for the main dataset (LPR cameras). The first step, "license plate matching", aimed to reduce the error rate of incomplete or wrongly detected license plates by the LPRs. About 2% of the stored 1,050,760 records had missing values in the license plate number. For example, if we had a record with a correct license plate 0000AAA, and another record with the value 0#00AAA, missing the second digit, we could, by probability, infer that both records belong to the same plate number and assign the correct value, 0000AAA, to both records. In our case, we assigned the same plate number to all those records whose license plate matches at least four characters out of seven in the same position. The second step, "route recovery", aimed to reduce the percentage of vehicles not detected by any LPR device. These errors occurred when the camera did not detect a vehicle that passes through the road. This error was difficult to detect, but in our setup, if a vehicle moves on the road from camera 1 to 3, and camera 2 (in the middle of the unique road connecting cameras 1 and 3), did not detect the car, we could infer that the car has passed through camera 2. In our process, if the vehicle was detected in less than 30 min in two non-consecutive cameras, our system infers that the vehicle is still in the area and calculates its time of stay based on the new registered values.

4.4. Data fusion

Combining data from provenance, mobility in the area, and the holiday calendar offered the opportunity to gain an understanding of the region, its inhabitants, and visitors. This section explains each source of information and the feature extraction and construction process of each dataset to allow the merging. We will detail the structure and variables obtained for each data source, creating a joint database. Table 3 schematically shows the information fusion process we followed.

License plate recognition data

The LPRs return information on four variables: the vehicle license plate (license_plate), the time stamp (time_stamp), and a variable (direction) indicated as "IN" when a vehicle enters the village or "OUT" when it exits. Each camera is uniquely identified by its (camera_id). The dataset contains information for nine months (February to October 2022). In total, we have 1,050,760 records, of which 25.69% correspond to the camera PAMPANEIRA 1 (i), 29.25% to PAMPANEIRA 2 (ii), 19.16% to BUBION (iii) and 25.9% to CAPILEIRA (iv) (see in Fig. 2). We grouped the records based on the new vehicle identifier (num_plate_ID), taking into account the mobility behavior of each vehicle. For each vehicle, we built a record per each time the vehicle visits the area, containing the date of entry (entry_time_stamp) and exit (exit_time_stamp) to the area and a list of all the cameras (route) by which it has been registered during its stay, this allows us to calculate the total distance traveled in kilometers (total_distance). This calculation is based on the road distance between each installed device, which we recorded in a small dataset. By summing up the distances between the cameras that a vehicle has passed by, we could determine the distance covered within the area. From the above records, we could also calculate the duration of stay (avg_visit) expressed in days and the number of nights spent there. In case of missing data, i.e., we could not calculate the time of entry or exit of a vehicle in the area, we removed the individual from the dataset.

After that, we performed a grouping at the license plate level so that each row corresponded to a different individual. In this way, we fused the information of all the vehicle visits in the area. Finally, we obtained a dataset with the total number of visits (total_entries), the average time (avg_visit) in days, the complete vehicle routing (route), the total accumulated distance traveled (total_distance), the standard deviation of the average time of each visit (std_visit) in days, the total time spent (total_time) in the area, and the total number of nights spent there (nights). From the new record structure, we could calculate the visits of each vehicle in different weeks (visits_dif_weeks) and months (visits_dif_months) to study the fidelity of the individual in the area. Finally, we obtained a dataset with 50,901 vehicle records and ten attributes.

Table 3
Detailed schematic of the data fusion stage in the pipeline.

Phase	Tasks	Values
	Calendar Data	
Importing Data	Read the dataset with information on public holidays at national level in Spain	270 days, 3 attributes (date, day_type, holiday_period)
Set holiday periods	Establish the important holiday periods in Spain: Summer Holiday, Christmas and Holy Week	Summer Holiday (from 1 aug. to 31 aug.) Christmas (from 12 dec. to 6 jan.) Holy Week (from 10 apr. to 17 apr.)
Encode variables	Convert categorical holiday periods into binary variables	270 days, 5 attributes (date, day_type, Summer, Christmas, Holy_Week)
	License Plate Recognition Data	
Importing Data	Read the cleaned dataset produced from the detection of vehicle license plates	1,050,760 rows, 4 attributes (license_plate, time_stamp, direction, camera_id)
Calculate associate variables	Calculate variables combining the 4 cameras + LPR location	(license_plate, entry_time_stamp, exit_time_stamp, route, total_distance)
Group information	Group the information for each record by vehicle	50,901 rows, 10 attributes (license_plate, total_entries, avg_visit, std_visit, total_time, nights, route, total_distance, visits_dif_weeks, visits_dif_months)
	Vehicle information Data	
Importing Data	Reads the dataset with vehicle information and its origin	45,132 license plates, 4 attributes (license_plate, postcode, co2_emissions, num_seats)
	Demographic and Economic data	
Importing Data	Reads demographic information about the region of origin of the vehicle	11,752 regions, 4 attributes (postcode, population, gross_income, disposable_income)
Merging Data	Merge the two sources	INE
Validate Data	Validate information common to the two sources	INE
	Geographic data	
Importing Data	Reads information regarding the region of origin of the vehicle	11,752 regions, 7 attributes (postcode, autonomous_community, province, county, district, town, km_to_dest)
Merging Data	Mix and validate information from the two sources used	geopy and pgeocode
Standardize values	Treatment of equivalences between names of regions in different co-official languages	Elimination of accents, spaces and translation to Spanish of all values related to region names
Validate Data	Validate postcodes and geolocation	geopy, pgeocode and INE
	Fusion Dataset	
Merging Data	Unification of header names and data formats, Mix postcode and license plate fields, Delete rows with some null fields	49,224 vehicles, 22 attributes (license_plate, total_entries, avg_visit, std_visit, total_time, nights, route, total_distance, visits_dif_weeks, visits_dif_months, co2_emissions, num_seats, postcode, autonomous_community, province, county, district, town, km_to_dest, population, gross_income, disposable_income)
Generate new variables	Calculate variables related to the type of dates in the calendar during the period of stay of each vehicle	49,224 vehicles, 27 attributes (license_plate, total_entries, avg_visit, std_visit, total_time, nights, route, total_distance, visits_dif_weeks, visits_dif_months, co2_emissions, num_seats, postcode, autonomous_community, province, county, district, town, km_to_dest, population, gross_income, disposable_income, total_holiday, total_workday, entry_in_holiday, total_high_season, total_low_season)
Exporting Data	Obtaining the resultant dataset	CLUSTERING_VEHICLES BD

Vehicle information data

The Spanish Directorate-General for Traffic (DGT) provided us with data relating to vehicle information[3] including details such as the vehicle's CO_2 emissions (co2_emissions), the number of seats (num_seats), and the postcode of the vehicle's address (postcode). Each vehicle was associated with a fiscal address used to pay road tax. This generally matched the driver's place of origin, although as described in Section 4.1, this was not entirely true. This dataset helped us understand the distribution of vehicle types and ownership in the different regions. We had a dataset with 45,132 vehicles registered in Spain and four attributes. Unfortunately, we did not have this information for vehicles registered outside of Spain. The percentage of foreigners in the data sample was less than 9.5%. Therefore, we determined these individuals exclusively by their mobility behavior in the area. All information related to vehicle information, demographic, economic, and calendar holidays was restricted to Spanish-registered vehicles.

Demographic and economic data

We accessed data regarding population size (population), average gross income (gross_income), and average disposable income (disposable_income) per person for each region linked to a postcode (postcode). This information came from the National Statistics Institute (Spanish: Instituto Nacional de Estadística, INE).[4] The data were available for regions with more than 1000 inhabitants and were updated until 2020. The information collected in this database allowed us to understand each region's economic and demographic characteristics, which was valuable for analyzing patterns in the data related to the

[3] https://sede.dgt.gob.es/es/vehiculos/informe-de-vehiculo/

[4] https://www.ine.es/dynt3/inebase/es/index.htm?padre=7132&capsel=5693

drivers' economic capacity and willingness to travel. We obtained a database with 11,752 postcode records from Spain and four attributes.

National calendar data

We obtained the holiday data using a holiday library, which also allowed the creation of custom calendars for local holidays, long weekends, and bank holidays. The library was designed to quickly and efficiently generate holiday sets specific to each country and subdivision (such as state or province).[5] It aimed to determine whether a particular date was a public holiday and to set national and regional holidays for multiple countries. As we mentioned before, due to the small percentage of foreign individuals in the sample and the complexity of dealing with a different set of holidays for different vehicles, we restricted the analysis of the holidays to Spain. However, we included Saturdays and Sundays in the holidays, so we also considered the idea of a weekly holiday for any origin. For each day, represented by a date (date), we specified with a binary variable whether it is a holiday or working day (day_type). In addition, holiday periods were defined to establish high and low tourist seasons based on the three most important national holidays in Spain: Summer, Christmas, and Holy Week,[6] which represented a binary variable, indicating whether the date belonged to that holiday period (Summer, Christmas, Holy Week). We obtained a database with 270 days and five attributes.

Geographic data

We obtained the geographic origin of the vehicles using the postcode and two libraries: pgeocode and geopy. pgeocode[7] allowed fast and efficient queries of GPS coordinates, region name, and municipality name from postcodes. geopy[8] is a Python client that provided access to several popular geocoding web services. We used data from both sources to validate and complement each other's vehicle location information at different levels, such as municipality, county, or suburb. Furthermore, we also used data from the INE[9] to verify the province and autonomous community code of the vehicle, which was directly related to the postcode. Hence, we created a database that contained, for each postcode, information about (autonomous_community), (province), (county), (district), (town), and the distance in kilometers between the origin of the vehicle and the destination region (km_to_dest). We obtained a database with 11,752 postal code records and nine attributes.

Merge of all the processed datasets

Finally, we fused all constructed databases, crossing the information from the license plate and postcode variables. After merging the tables, we eliminated records with any of the aforementioned attributes null. The information from the national calendar allowed us to add to the vehicle database information related to the stay and its total number of holidays (total_holiday), workdays (total_workday), high season (total_high_season), low season (total_high_season) and a binary variable indicating whether the vehicle enters the area on a holiday or a workday (entry_in_holiday). The resulting dataset contains information on the behavior in the area for 49,224 vehicles and 27 attributes.

4.5. Preprocessing

Our dataset contains 27 attributes with different scales and units. Hence, some variables may be more influential than others in our analysis. To solve this problem, we will apply normalization to the data. Normalization must be applied to numerical data, so we must first convert the categorical variables (in our use case: route, postcode, autonomous_community, province, county, district, town) to numerical values. In particular, the numeric variable, total_distance, kept the information of the kilometers traveled in the variable route. The rest of the categorical variables related to the provenance: town, postal code, etc., and we converted them into the variable km_to_dest. We removed the variables co2_emissions and num_seats, because they had a high percentage of missing values (about 25%), which could introduce noise. During this phase, we also excluded vehicles with a total stay time (total_time) of less than 1 h. This subset comprised 16.98% (8360 vehicles) of the entire dataset. Given their role as transient passers-by in the area and their brief stays, which did not contribute to any discernible benefits for the locality, we omitted them from our analysis. We finally obtained a dataset with 40,864 vehicles and 17 numerical attributes: total_entries, avg_visit, std_visit, total_time, nights, total_distance, visits_dif_weeks, visits_dif_months, km_to_dest, population, gross_income, disposable_income, total_holiday, total_workday, entry_in_holiday, total_high_season, total_low_season.

4.6. Dimensionality reduction

We reduced the dataset's dimensionality to improve efficiency in clustering. This involved simplifying the feature matrix by removing low-variance features that would not contribute much to our goal of clustering different vehicle behaviors. We used PCA to reduce dimensionality. We found that removing variables with very high correlation substantially improved the results and the performance of the clustering models for our data. Furthermore, correlated variables increased the data's variance, making the visual interpretation of the PCA results difficult, as the first principal components might not have accurately reflected the underlying structure of the data.

4.7. Clustering and evaluation

Our study explored all the algorithms mentioned in Section 3.1 to determine the optimal approach for pattern recognition and evaluated whether they could find a realistic solution.

4.8. Visualization

Data visualization was essential in our work, as it helped to determine and make decisions about parameter settings, algorithms, and normalization methods. It also made our machine learning results more understandable. For instance, we used the elbow method to find the best number of clusters for various algorithms. This method plots the number of clusters and a given evaluation metric. The number of clusters at the curve's bend ("elbow") balances the model's complexity and accuracy. We used scatter plots to visualize the first two principal components for each normalization method, helping us grasp the data's structure and cluster distribution. Box plots were another tool we used to show how features were distributed within clusters. This allowed us to spot common patterns in each cluster.

4.9. Data privacy and security

The LPR cameras sent the license plates to a secured server on our provider's premises. We only used the anonymized dataset (see), which we openly published.[10] The other datasets were public, except the DGT dataset. The DGT shared with us sensitive data with license plates and its associate owner's postal code only for research purposes. This information was stored encrypted and was accessible only to

[5] https://python-holidays.readthedocs.io/en/latest/

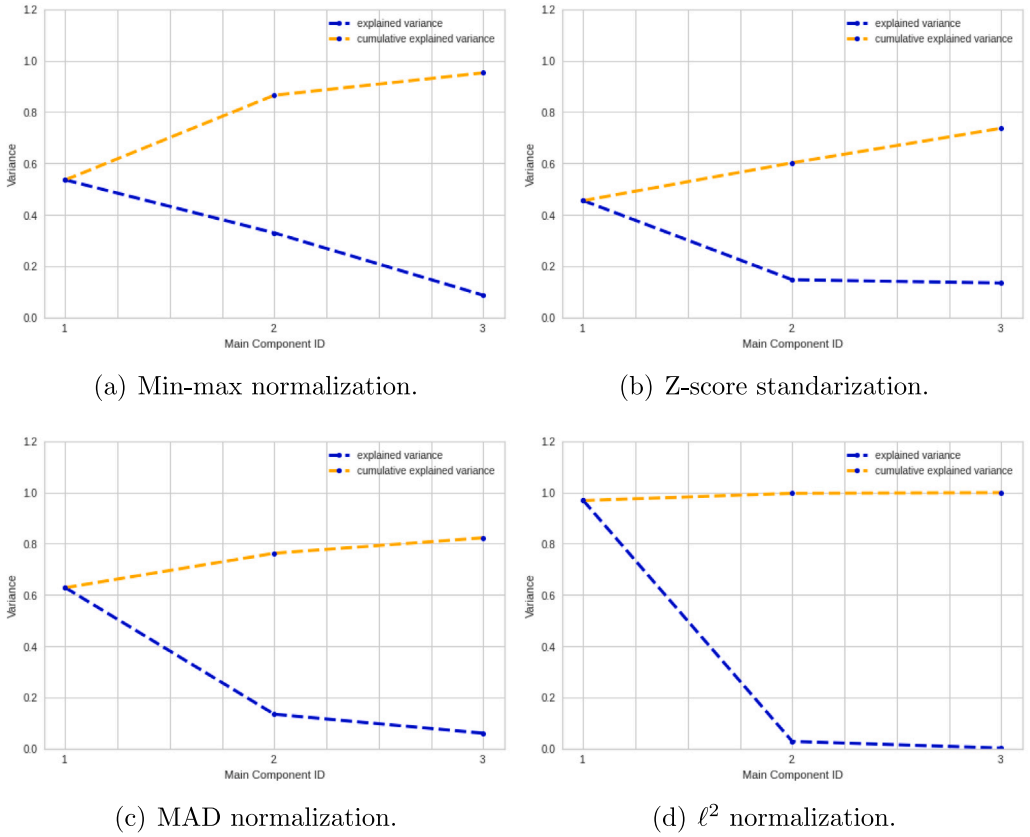
[6] https://es.statista.com/temas/3585/vacaciones-en-espana/#topicOverview

[7] https://pgeocode.readthedocs.io/en/latest/

[8] https://geopy.readthedocs.io/en/latest/

[9] https://www.ine.es/daco/daco42/codmun/cod_ccaa_provincia.htm

[10] https://zenodo.org/record/8356386



(a) Min-max normalization. (b) Z-score standarization.

(c) MAD normalization. (d) ℓ^2 normalization.

Fig. 3. Variance with 3 principal components.

authorized researchers. Furthermore, we used clustering, which means that we did not evaluate the individual behavior of each person but considered them part of a group. Hence, the privacy of the activities of the individuals is not compromised.

5. Results

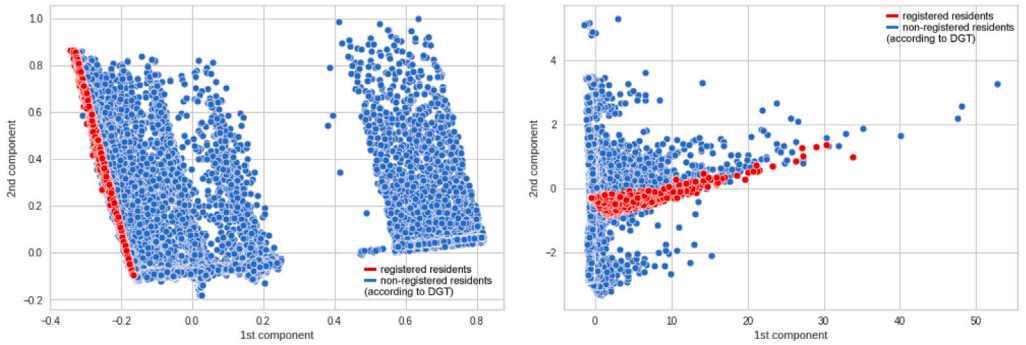
To model traffic behavior and distinguish between residents and visitors. We labeled vehicles as 1 for those registered in the area (resident) and 0 for others. We identified several variables with non-significant correlations (correlation < 0.2): avg_visit, std_visit, and population, and removed them. We showed the results relating to these analyses in Appendix (Fig. A.1 and Table A.1).

5.1. Preprocessing and dimension reduction results: Normalization selection

We performed preprocessing and dimension reduction stages together because they are interdependent. We found that removing highly correlated variables before applying PCA improved the variance explained and the scatter plots of PCA components. Specifically, we removed variables with a correlation coefficient > 0.9: total_entries, nights, visit_dif_weeks, visit_dif_months, km_to_POQ, gross_income, entry_in_holiday, total_distance and total_high_season Appendix (Fig. A.2).

After applying the four most common normalizations to the data (see in Section 3), we applied PCA analysis. Fig. 3 showed the variance carried by each PCA component for each normalization. We could appreciate that two components explained most of the variance in all normalizations. Hence, we performed an exploratory visual analysis by plotting the first two principal components to study their underlying geometry. In Fig. 4, we overlaid on the plots, in red, the points representing the vehicles of the registered residents, in blue, non-registered residents.

The normalization method that obtained the highest cumulative variance was ℓ^2, indicating that it retained the most information in only two components (see in Fig. 3(d)). In addition, the variance of each dimension was high compared to the other techniques analyzed, suggesting that the data were well distributed in both dimensions. The graph in Fig. 4(d) shows a clear separation between the two groups, and the registered residents (in red) were well confined. The min–max normalization method obtained the second-best cumulative variance and the highest variance for each dimension, preserving a reasonable amount of information in only two components (see in Fig. 3(a)). The graph also shows a clear separation between the two groups, and the actual residents were defined along a vertical line on the left cluster in Fig. 4(a). In contrast, the MAD normalization method had a lower cumulative variance and variance for each dimension (see in Fig. 3(c)) than the ℓ^2 and min–max normalization methods. The 2-dimensional scatter plot showed no apparent clusters (see in Fig. 4(c)), and the actual residents were highly dispersed, which made it unusable for our analysis. We had similar results in a scatter plot of three principal components. Finally, the mean normalization, z-score, method presented the lowest cumulative variance, indicating that it lost more information during dimensionality reduction than other techniques (see in Fig. 3(b)). The graph shows that the actual residents were grouped together, but for the 2-components, there were no apparent significant clusters (see in Fig. 4(c)). The trend of the cumulative variance explained was rising, suggesting that the current normalization method could be enhanced by including more components. By adding more dimensions, it may be possible to identify a dimension where the group of registered residents conformed to a clearer distribution. PCA typically worked better with z-score standardization than with min–max normalization. However, normalization techniques that better handled outliers (such as z-score) may not always have been effective for all datasets because they tried to distribute the individuals uniformly, softening the outliers. For example, we observed that the min–max normalization



(a) Min-max normalization. (b) Z-score standarization.

(c) MAD normalization. (d) ℓ^2 normalization.

Fig. 4. Scatter-plot of the first two principal components for the different normalizations.

method performed better than the z-score standardization, possibly due to the presence of small clusters that z-score detected as outliers. In particular, the dataset have a low proportion of registered residents (less than 2% of the total sample), which could be considered outliers (see in Table A.1). In these cases, the min–max normalization method, which was more sensitive to small clusters, may have given better results. With all this information, we decided to apply the two best normalizations for our data (ℓ^2 and min–max) and compare the results obtained in the clustering.

5.2. Exploration of clustering algorithm categories

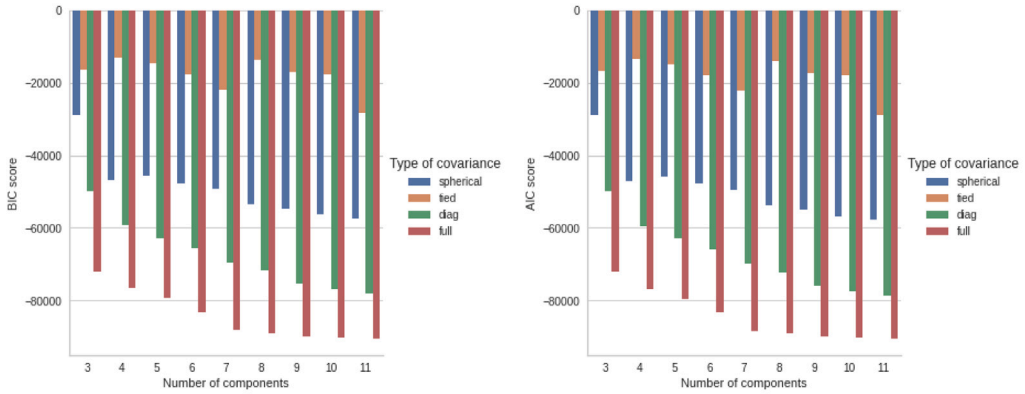
From the scatter plots in Fig. 4, we observed that the data points were spread relatively flat. This suggested that the data points were concentrated in a lower-dimensional space within the original feature space. In other words, the data appeared to exist in a more compressed space rather than being spread out across multiple dimensions. Hence, partition and distribution-based clustering models were the most suitable for this geometry (see Section 3.5). We tested various algorithms from other categories to verify this. However, we did not report the results because none of the tested techniques identified a cluster for the correctly registered residents. For example, density and spectral-based algorithms performed poorly, probably because of the non-flat geometry but also because they worked best for detecting outliers. Hierarchical algorithms performed poorly, probably because of the non-flat geometry, but they also had difficulties with highly concentrated datasets, creating distinct groups only when the separation was obvious. Consequently, we focused on the partitional and distribution-based algorithms, which worked well with flat geometry data. In particular, we tried Gaussian Mixture, K-Means, and MiniBatchKMeans.

Gaussian Mixture models were more flexible and could handle different cluster shapes and sizes, while K-Means assumed a spherical shape of the clusters and a uniform size. In addition, Gaussian Mixture models could estimate the probability that a data point belonged to a cluster, which could be useful in specific applications where we needed to make decisions based on uncertain data or when we wanted to assign a data point to multiple clusters with different probabilities. In the tests carried out, we discovered that K-Means and MiniBatchKMeans were not able to find any cluster that contained the majority of individuals of registered residents (see in Fig. 4(a) and (d)). This was because the distribution of these individuals followed an elliptical geometry, which was not amenable to partition-based algorithms directly. Based on these results, we used the Gaussian Mixture clustering algorithm given the geometry of our data and the distribution followed by registered residents.

5.3. Evaluation results

After choosing the algorithm, we had to configure its settings and hyperparameters. For the GaussianMixture algorithm, a ‘mixture’ meant a blend of multiple Gaussian distributions, with each component representing one of these distributions [61]. We could adjust the number of mixture components, determining how many Gaussian distributions to use for modeling the data. Another configurable aspect was the covariance type, which influenced how variables in the data were correlated, impacting the model’s accuracy and efficiency. The common types of covariance were:

- Full: all components have their own covariance matrix. This means that each component can have a complex correlation structure between the different variables.
- Tied: all components share the same overall covariance matrix. This can be useful if different variables are highly correlated.
- Diagonal: each component has its own diagonal in the covariance matrix. This means that the correlation structure between the different variables is limited to correlations between pairs of variables.



(a) BIC score. (b) AIC score.

Fig. 5. Information criteria for the GaussianMixture on min–max normalization.

- Spherical: each mixture component has its own unique variance. This means that the correlation structure between the different variables is limited to the variance of each variable individually.

To select the best hyperparameters, we calculated the performance of the resulting model with the metrics presented in Section 3.2, which were appropriate for clustering algorithms based on distributions (BIC and AIC). In the next subsections, we performed the evaluation for the different types of covariance of the GaussianMixture algorithm on the two normalizations chosen in the previous subsection: min–max and ℓ^2 normalization.

5.3.1. Evaluation results: Min–max normalization

Fig. 5 represents the values of the BIC and AIC metrics with respect to the number of components and type of covariance used as parameters of the GaussianMixture algorithm. We noted that the 'full' covariance type was the one that minimized both metrics in all cases, so it was the one chosen for the subsequent analysis. This value meant that each component had its own overall covariance matrix, which meant it could capture any correlation between variables. We noted no significant differences between the values obtained for AIC and BIC scores. Therefore, we calculated the elbow method on the BIC score to select the optimal number of mixture components. In Fig. 6, we could detect two "elbow" points. One occurred at seven components (−87,585 BIC), marking a 4591 unit difference from the preceding six components (−82,994 BIC) and a 1632 unit difference from the following eight components (−89,217 BIC). The other point was at four components (−76,798 BIC), with a 4407 unit difference from the preceding three components (−72,391 BIC) and a 3098 unit difference from the subsequent five components (−79,896 BIC). The change from seven components to their previous value was more substantial than the change from three to four, and the difference with the following eight components was less pronounced, indicating a more abrupt change in slope.

5.3.2. Evaluation results: ℓ^2 normalization

Fig. 7 represents the values of the BIC and AIC metrics with respect to the number of components and type of covariance, used as parameters of the GaussianMixture algorithm for ℓ^2 normalization. We observed that the 'tied' covariance type was slightly superior for three components, but the 'full' covariance type was again the best for more than three components. Similarly to the min–max normalization, there was no significant difference between the AIC and BIC score values. Therefore, we calculated the elbow method on the BIC score and the 'full' covariance type. Fig. 8 shows a clear change in four components, showing an increase of 36,977 units in the BIC score (the highest in the graph), going from three components (−232,851 BIC) to four components (−269,828 BIC).

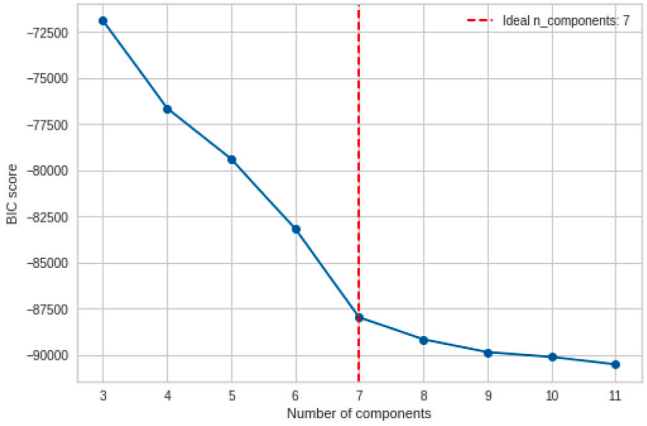


Fig. 6. Elbow method for BIC using min–max normalization.

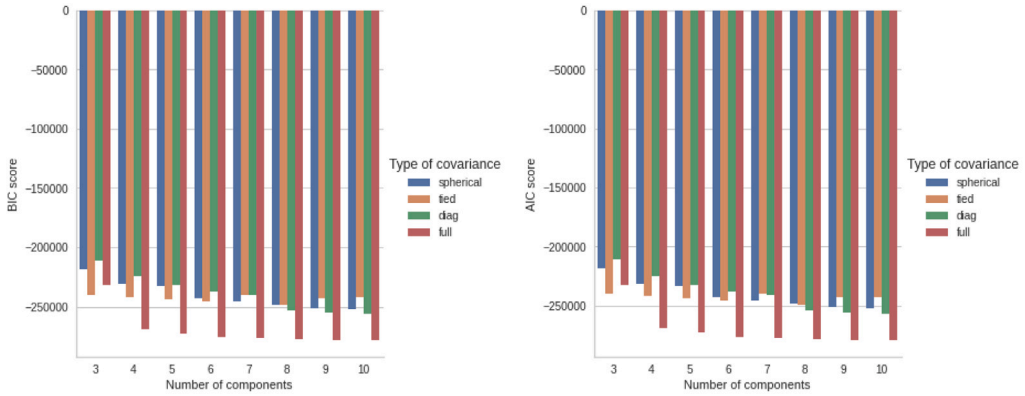
5.4. Visualization results

Once we selected the clustering algorithm and the hyperparameters, we discussed the visualization of the generated clusters over the two chosen normalizations: min–max normalization and ℓ^2.

5.4.1. Visualization: Min–max normalization

Fig. 9(a) shows a 2D scatter plot, where each axis represented 1st and 2nd principal components. Fig. 9(b) highlights registered residents in red. Fig. 9(c) displays a 3D scatter plot with 3 principal components in each axis. Table 4 shows vehicle percentages and registered resident counts in 7 clusters. Cluster 3 correctly grouped over 96% of individuals, and cluster 5 contained nearly 45% of the total sample. Cluster 3, with the most registered residents, represented around 14% of the total population.

Fig. 10 presents the box plots for the 7 clusters for the nights (Fig. 10(a)) and km_to_dest (Fig. 10(b)) variables, which showed significant differences in explaining the groups. Figs. 11 and 12 presents the box plots of the most relevant variables for the 7 clusters obtained. Table 5 complements Figs. 11 and 12, indicating the exact number of the mean of each variable in each cluster. To facilitate visualization, we separated some of the box plots according to the value of the variable nights, which seemed to discriminate well between 2 groups of clusters: (0, 1, 2, 5) with lower values and (3, 4, 6) with higher values (see in Fig. 10(a)). Clusters 3,4,6 had a number of nights close to the behavior of a resident in the area and represented 27.44% of the data (see Table 4). Clusters 0,1,2,5 had visitor behavior because they spent fewer nights in the area and represented 72.56% of the total sample.



(a) BIC score. (b) AIC score.

Fig. 7. Information criteria for the GaussianMixture on ℓ^2 normalization.

Table 4
Clusters based on registered resident labels using min–max normalization.

Data points	Nº cluster						
	0	1	2	3	4	5	6
Percentage of sample	14.13%	5.74%	8.06%	13.47%	10.30%	44.63%	3.67%
Real Residents	8	0	0	641	3	9	0
Rest of individuals	5766	2347	3293	4862	4205	18,230	1500

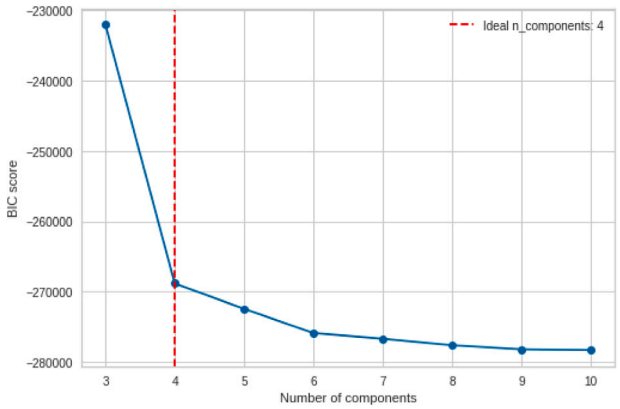


Fig. 8. Elbow method for BIC using ℓ^2 normalization.

For clusters 3, 4, and 6, a key factor was the distance in kilometers from the vehicle's registered address to the area (see Fig. 11(c)). Despite significant differences in origin, these three clusters exhibited similar patterns in terms of nights spent, indicating that they resided or stayed in the area. Cluster 3, with an average distance of 19.39 km (see Table 5), primarily consisted of vehicles registered in the study area (registered residents) and nearby villages. Cluster 6, with an average distance of 1747.30 km for the variable km_to_dest, comprised non-registered residents from abroad, as defined in Section 4.1. Cluster 4, with an average distance of 318.36 km, represented individuals from other regions of Spain who were also non-registered residents, as discussed in the same section. Additionally, the gross income variable was significantly higher in cluster 4 compared to clusters 3 and 6 (almost 34% higher) (see Fig. 12(c)). This suggests that a majority of individuals in cluster 4 (non-registered residents from other Spanish regions) came from regions with above-average incomes. Residents living farther away (clusters 4 and 6) had lower average values for total_distance, total_high_season, and total_entries (see Table 5). This is because they tended to visit less often, cover shorter distances in the area, and have fewer visits during the high season compared to residents in closer proximity (cluster 3) (see Fig. 11(e) and Fig. 12(a, e)).

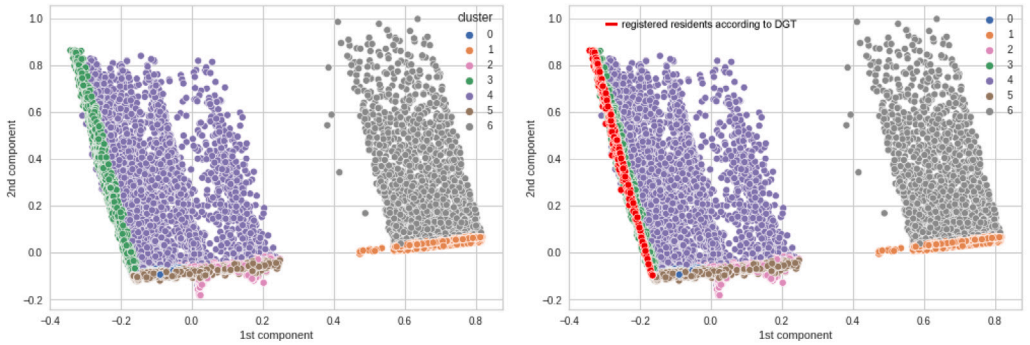
Clusters 0, 1, 2, and 5 represented different visitor behaviors (see in Table 5). Cluster 0, with an average distance of 128.55 km, corresponded to visitors from the province, typically staying 1.57 nights. They made an average of 1.54 visits, mostly during weekends and holidays, and around 65% of these visits occurred in high season (see in Fig. 12(b, f)). Cluster 1, averaging 1742.97 km, consisted of foreign visitors who stayed for only 0.26 nights. They tended to visit during low seasons, primarily using the main road to reach the first village in the area and not visiting the other villages. Cluster 2, with an average distance of 474.21 km, attracted visitors from outside the province, spending around 1.55 nights. This cluster had the highest average gross income (see in Fig. 12(d)) and visits the area during high season, likely by tourists from northern Spain. Cluster 5, averaging 253.70 km, represented visitors from other nearby provinces. They rarely stayed overnight (0 nights on average) and predominantly visited during the day, making up 44.63% of the sample (see Table 4). Only 27% of their visits occurred during high season (see Fig. 12(f)), suggesting day trips from neighboring provinces.

5.4.2. Visualization: ℓ^2 normalization

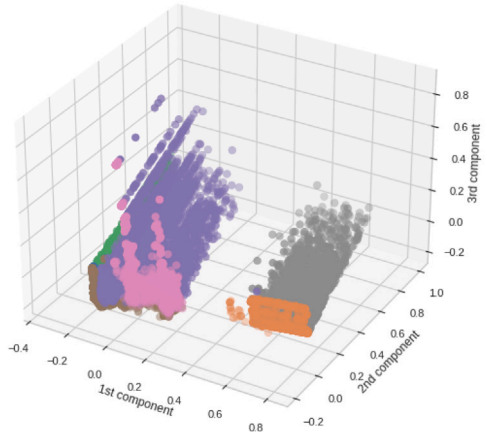
Fig. 13 shows the data distribution using ℓ^2 normalization. Fig. 13(a) depicts a 2D scatter plot of principal components (1st and 2nd axes). In Fig. 13(b), registered residents are marked in red, and Fig. 13(c) presents a 3D scatter plot. Table 6 shows cluster details: Cluster 0 accurately includes over 89% of registered residents, representing 10.30% of the total population. Cluster 3 contains 75.95% of the total sample.

Fig. 14 shows the box plots of the relevant variables for the 4 clusters, and Table 7 displays the mean of each of these variables in each cluster. We distinguished two clusters that contained a high value of the variable "nights" (cluster 0 and 2), while the rest of the clusters (clusters 1 and 3) had a low value. Although there were outliers (see in Fig. 14(a)) that increased the mean number of nights for these clusters (clusters 1 and 3), 50% of the individuals had a number of nights lower than 2 for cluster 3 and lower than 15 nights for cluster 1.

Cluster 0, which included over 89% of area residents, had an average stay of 144.93 nights, covering an average distance of 25.54

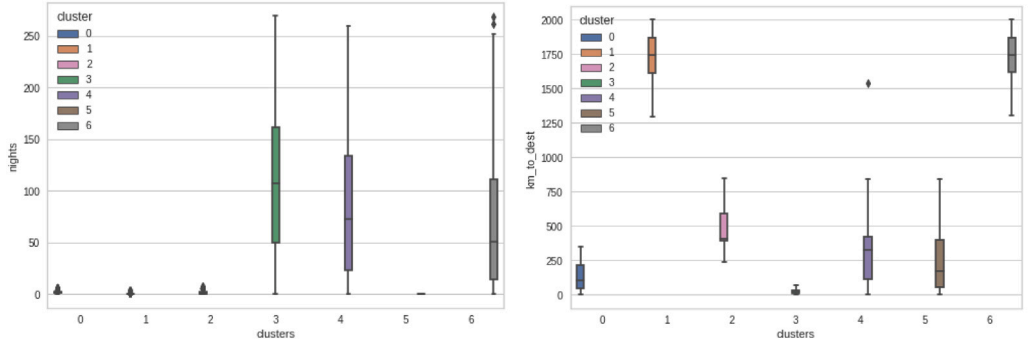


(a) Segmentation for 7 mixture components. (b) Highlighted registered residents.



(c) 3D plot with 3rd component.

Fig. 9. Scatter-plot of the first three components (PCA) using min–max normalization.

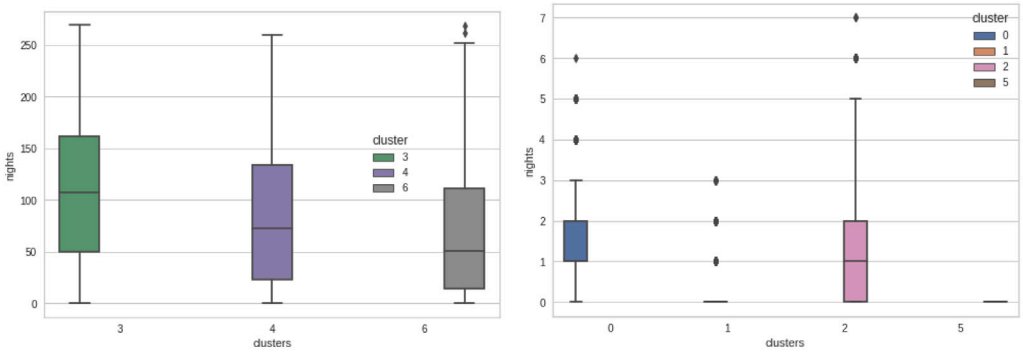


(a) Total nights of stay by cluster. (b) Distance in Km. to the area by cluster.

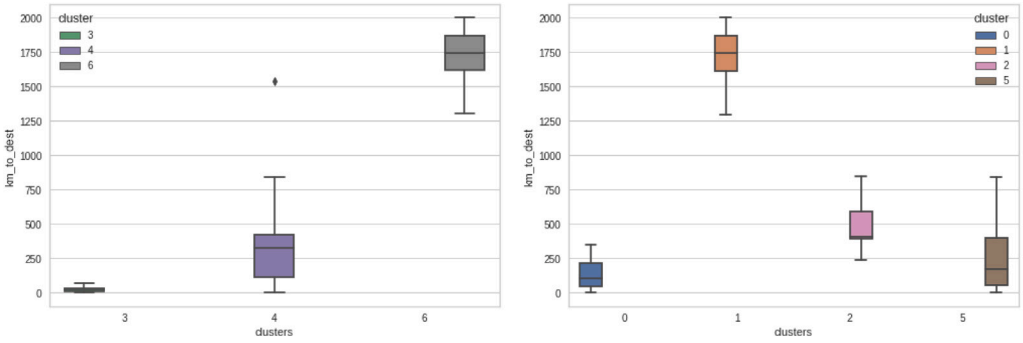
Fig. 10. Box plots for min–max normalization (I).

Table 5
Mean of variables for each cluster performed using min–max normalization.

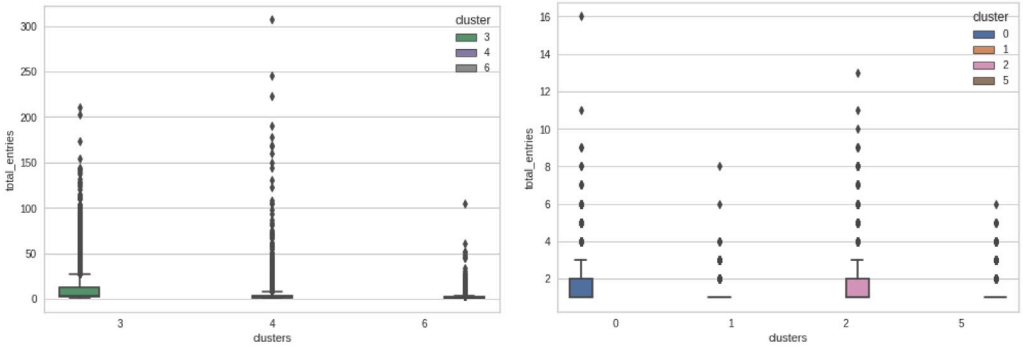
Variables	Nº cluster						
	0	1	2	3	4	5	6
nights	1.57	0.26	1.55	108.62	84.66	0.00	68.73
km_to_dest	128.55	1742.97	474.21	19.39	318.36	253.70	1747.30
total_entries	1.54	1.12	1.58	10.34	4.36	1.12	2.71
total_distance	11.64	4.90	10.67	70.24	30.77	4.86	14.42
gross_income	23,085.36	19,482.10	35,547.66	20,972.17	26,902.26	25,151.75	19,179.54
total_high_season	1.01	0.31	1.14	18.85	15.10	0.31	11.24



(a) Nights of stay for clusters **3,4,6**. (b) Nights of stay for clusters **0,1,2,5**.



(c) Distance to the area for clusters **3,4,6**. (d) Distance to the area for clusters **0,1,2,5**.



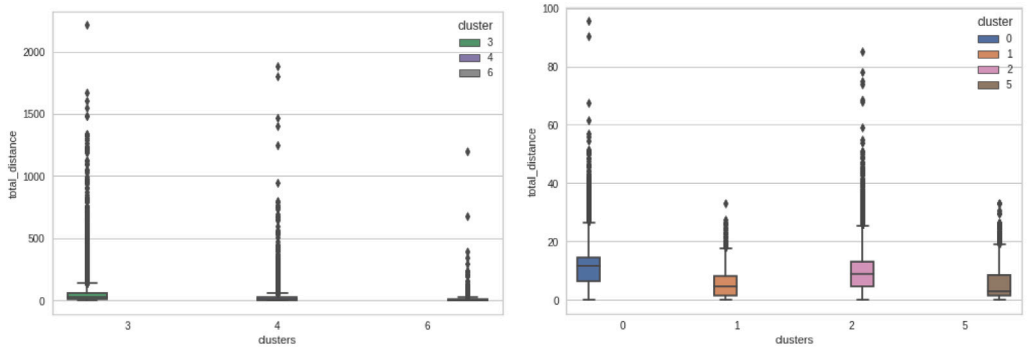
(e) Total entries for clusters **3,4,6**. (f) Total entries for clusters **0,1,2,5**.

Fig. 11. Box plots for min–max normalization (II).

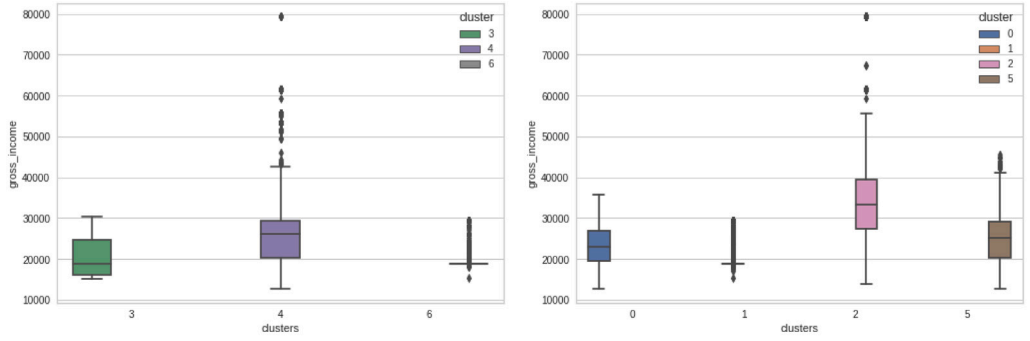
Table 6
Clusters based on actual resident labels using ℓ^2 normalization.

Data points	Nº cluster			
	0	1	2	3
Percentage of sample	10.30%	8.50%	5.25%	75.95%
Real Residents	589	0	0	62
Rest of individuals	3620	3473	2146	30,974

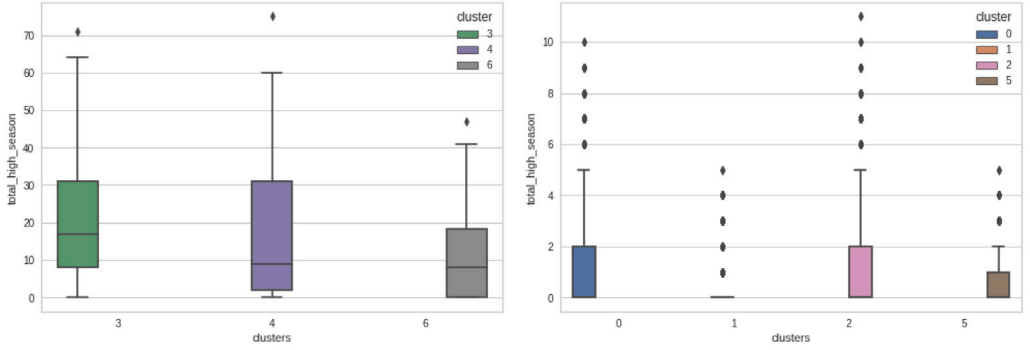
km. Most non-registered residents in this cluster were from the province (see Fig. 14(b)). Cluster 2 represented non-registered residents from outside Granada, making up only 5.25% of the total sample. They stayed an average of 84.62 nights and came from an average distance of 598.01 km. For both groups, total_distance, total_high_season, and total_entries (see Table 7) were inversely proportional to km_to_dest, indicating that visitors from further away tended to visit during the low season, move less within the area, and visit fewer times a year (see Fig. 14(c, d, f)). Cluster 1 comprised foreign visitors and some non-registered foreign residents, covering an average distance of 1750.68 km. They stayed an average of 22.81 nights, with only 17% of stays in the high season. Cluster 3, the largest group (75.95% of the sample), had an average stay of 4.82 nights (although most did not stay overnight). They covered an average distance of 240.01 km and rarely visited in the high season (28% of the total stay) (see Fig. 14(f)). It also had the highest income, with an average of 26,158.32.



(a) Distance run in area for clusters **3,4,6**. (b) Distance run in area for clusters **0,1,2,5**.



(c) Avg. gross income for clusters **3,4,6**. (d) Avg. gross income for clusters **0,1,2,5**.



(e) Total high season for clusters **3,4,6**. (f) Total high season for clusters **0,1,2,5**.

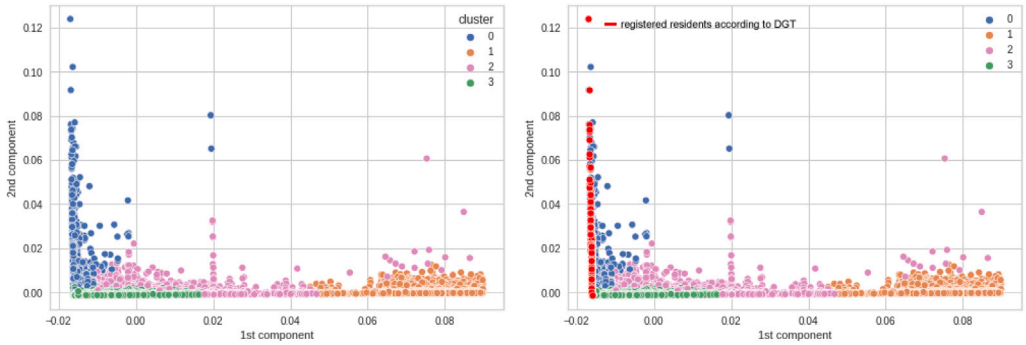
Fig. 12. Box plots for min–max normalization (III).

Table 7
Mean of variables for each cluster performed using ℓ^2 normalization.

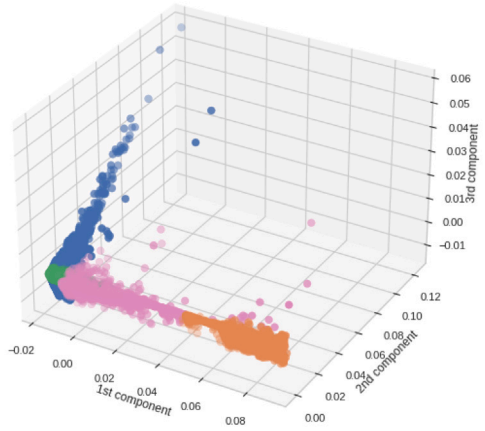
Variables	Nº cluster			
	0	1	2	3
nights	144.93	22.81	84.62	4.82
km_to_dest	25.54	1750.68	598.01	240.01
total_entries	13.18	1.52	3.53	1.49
total_distance	95.43	6.89	26.43	8.01
gross_income	20,268.41	19,018.55	22,886.88	26,158.32
total_high_season	24.83	3.87	14.47	1.36

6. Discussion

Table 8 shows the equivalence by clusters and percentage of the total set for the two normalizations analyzed. Additionally, it briefly describes the general profile of individuals in each cluster. For the group of registered residents, we could see that both normalization methods grouped them into a single cluster (cluster 3 in min–max and 0 in ℓ^2). However, there was a 3.17% difference in the size of these clusters, with the ℓ^2 cluster size being smaller. The min–max normalization distinguished between foreign visitors and foreign non-registered residents (clusters 1 and 6, respectively), while the ℓ^2 normalization grouped all foreign individuals into a single cluster (cluster 1). The clusters of national non-registered residents were also similar in both normalization methods (cluster 4 in min–max and 2 in ℓ^2). Still, there



(a) Segmentation for 4 mixture components. (b) Highlighted registered residents.



(c) 3D plot with 3rd component.

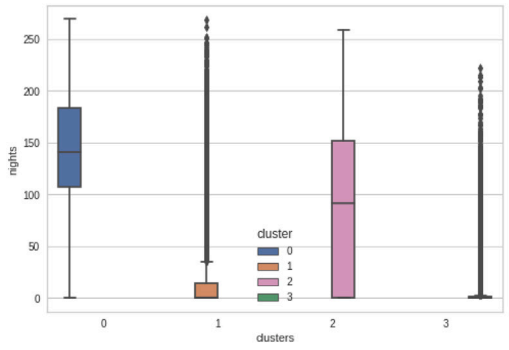
Fig. 13. Scatter-plot of the first three components (PCA) using ℓ^2 normalization.

Table 8
Equivalence of the clusters made for each normalization.

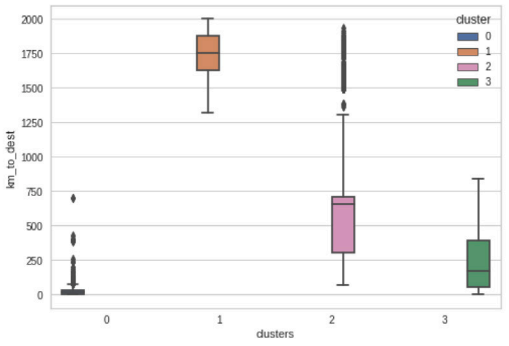
Normalization								
Min–max	Nº cluster	3	1	6	4	0	2	5
	% sample	11.17%	6.11%	3.05%	8.55%	15.04%	8.58%	47.50%
	Description	Registered residents	International visitors	Non-registered international residents	Non-registered national residents	Visitors from Granada who stay for 1–2 nights	National visitors	Visitors from Granada who do not stay night
	Additional characteristics	Long stays and travel frequently in the area	No overnight and visits mostly in low season	Long stays and above-average distance of provenance	Long stays and above-average income and visits	Overnights mostly in high season and weekends	Overnights mostly in high season and above-average income	No overnight and visits mostly in low season
ℓ^2	Nº cluster	0	1		2	3		
	% sample	8.55%	8.76%		4.36%	78.33%		
	Description	Registered residents	International visitors		Non-registered national residents	National visitors		
	Additional characteristics	Long stays and travel frequently in the area	Medium-short stays and visits mostly in low season		Long stays and above-average income and visits	Short stays and visits mostly in low season		

was a 5.05% difference in the size of these clusters, with the size of the ℓ^2 cluster also being smaller. Finally, the ℓ^2 normalization grouped all national visitors into a single cluster (cluster 3), while the min–max normalization divided these into three distinct clusters (clusters 0, 2, and 5). It should be noted that in the ℓ^2 normalization, cluster 3 is larger than the sum of clusters 0, 2, and 5, because it contained individuals with resident behaviors that were not included in the other clusters. This explained the significant differences in the sample sizes of clusters 0 and 2 compared to their equivalents in the min–max normalization.

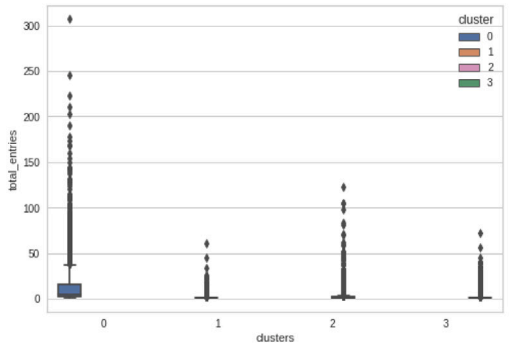
Fig. 15 shows a hierarchical graph comparing the equivalences presented in Table 8 between the two normalizations. We could quickly discern the descriptions that corresponded to each normalization for each cluster type. The min–max normalization seemed more efficient since it allowed a more detailed segmentation of individuals than ℓ^2, and ℓ^2 showed more outliers in the box plots for all the variables. While



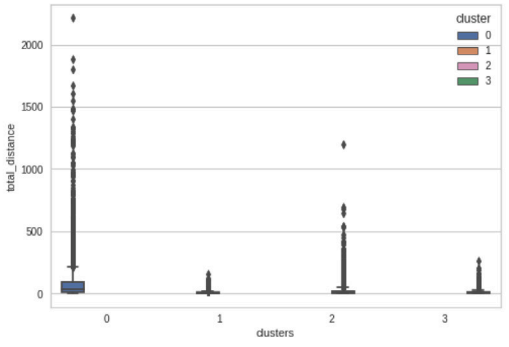
(a) Total nights of stay by cluster.



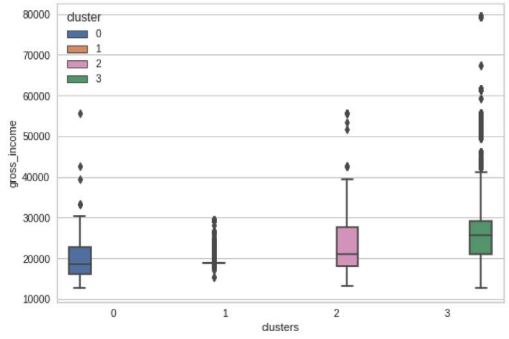
(b) Distance in Km. to the area by cluster.



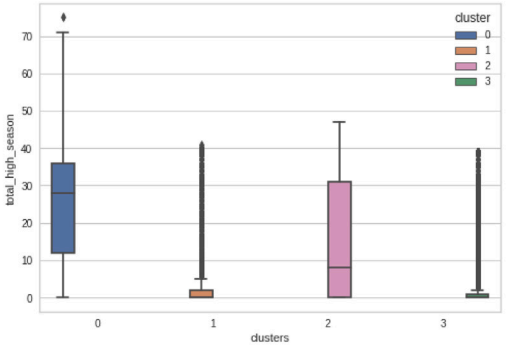
(c) Total entries to the area by cluster.



(d) Distance run in area by cluster.



(e) Average gross income by cluster.



(f) Total high season by cluster.

Fig. 14. Box plots for ℓ^2 normalization.

min–max seemed to distinguish the residents from the visitors, with the variable representing the number of nights spent in the area, ℓ^2 seemed to have a clear segmentation based on the distance to their home. Hence, for our purposes, min–max offered better segmentations. In addition, min–max detected atypical behaviors of individuals not officially registered as residents of the area, but that behaved as residents. In contrast, the ℓ^2 normalization could be useful for excluding foreigners from the analysis and focusing only on comparing registered and non-registered residents at the national level, grouping visitors in a single cluster. Our work, as many in machine learning in real environments, has some limitations related to uncontrolled variables. In particular, we acknowledge that there could be some rented cars with a national plate number that does not match the occupants' provenance; unfortunately, we could not access any rented car database. Likewise, we could not find any good local event calendars, which could affect the traffic.

In summary, our methodology comprises eight steps (see Fig. 1). Initially, we gathered data from various sources, cleaned it, and merged it based on vehicle licenses. In this merge step, we also calculated additional variables from the existing ones (e.g., route and total distance in the area). Next, we followed a systematic sequence involving preprocessing, reducing dimensions, and clustering. Ultimately, we evaluate outcomes through visualization techniques. This approach enriches LPR data with contextual information, uncovering novel patterns within the data. Additionally, it facilitates the comparison of algorithm performance, such as comparing different normalization algorithms in the performance of vehicle-behavior clustering. In smart villages, it is important to select suitable LPR locations to cover the towns entries and exits, and it is also important to consider that the official residence could only be partially reliable.

7. Conclusions

The paper presented an effective pipeline for clustering analysis, using data from different sensors and sources to detect registered and non-registered residents and visitors and their behavior in a given

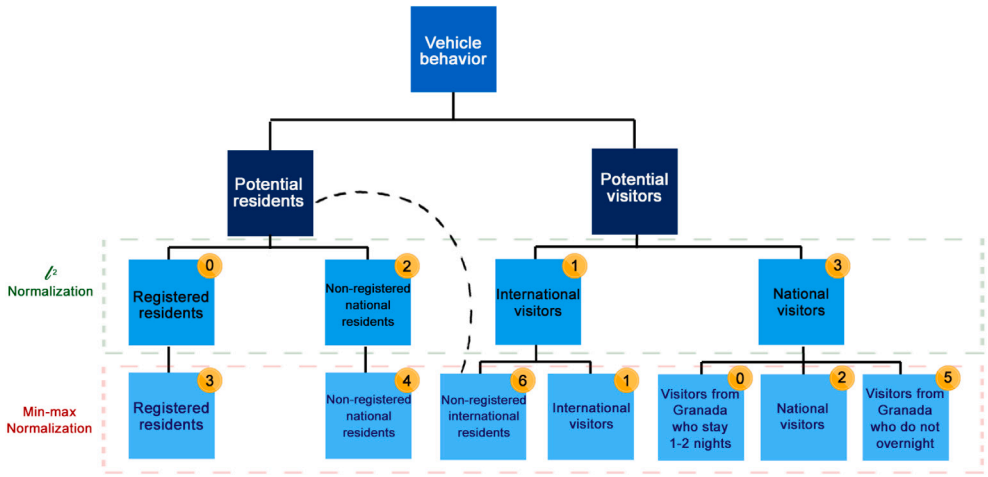


Fig. 15. Hierarchical graph of the two clusters made for each normalization.

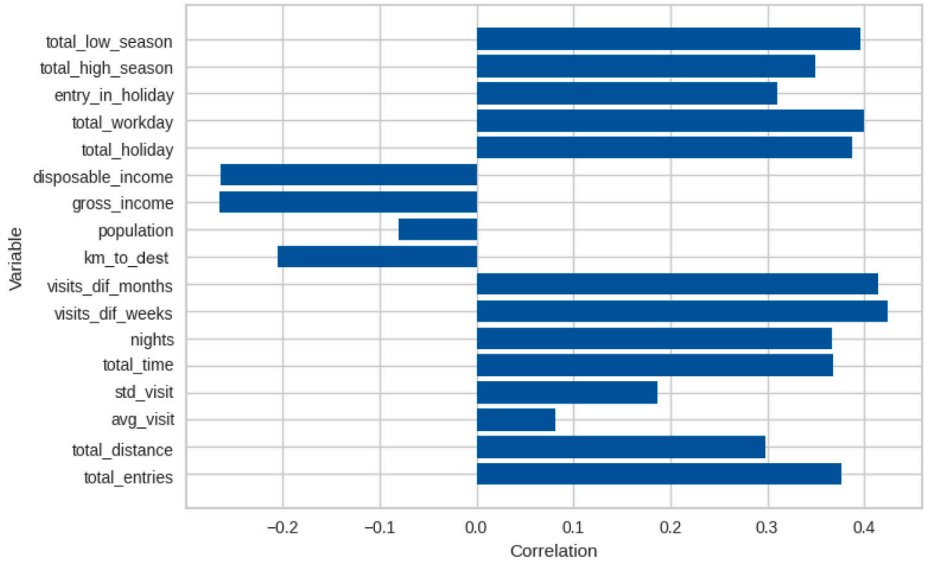


Fig. A.1. Correlation between the registered resident label and the rest of the variables.

area. We selected an optimal clustering algorithm based on the data distribution and two potential normalization algorithms. We found that the min–max normalization was the most effective for detailed segmentation of individuals and their visiting behavior in the area and detection of atypical behavior of individuals not registered as residents of the area but showing resident behavior. The ℓ^2 normalization could be useful in specific situations requiring a distinction from the region of origin. This analysis could assist area managers in crafting tailored strategies to keep certain tourists, considering their income and origin, and promoting overnight stays. This could boost the local economy and reduce traffic. Additionally, these patterns could inform policies to engage non-registered residents in the community, such as tax breaks or social programs. In Spain, this data is crucial for tasks like licensing pharmacies, investing in public health, and scheduling security forces based on seasonal fluctuations. Our pipeline and analysis could also assist data analysts in improving their solutions and making informed decisions. In the future, we aim to conduct an independent clustering analysis on the dataset of passing vehicles in the area. The objective is to identify movement patterns and promote longer stays within the vicinity. Likewise, we will try to find useful datasets that could enhance the results, such as vacation accommodation occupancy or local events, although in small villages, it could be a challenge to find good datasets.

CRediT authorship contribution statement

Daniel Bolaños-Martinez: Conceptualization, Formal analysis, Methodology, Software, Validation, Visualization, Writing – original draft. **Maria Bermudez-Edo:** Conceptualization, Investigation, Methodology, Project administration, Resources, Supervision, Writing – review & editing. **Jose Luis Garrido:** Investigation, Project administration, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

I have share a link to my data in the Manuscript File.

Acknowledgments

This publication is part of the R&D&i Project Ref. PID2019-109644RB-I00 funded by Ministerio de Ciencia e Innovación / Agencia Estatal de

Table A.1
Mean and std. deviation for registered residents and rest of individuals in dataset.

	nights		total_distance		total_entries		entry_in_holiday	
	Residents	Others	Residents	Others	Residents	Others	Residents	Others
mean	158.47	19.99	205.82	13.60	19.46	2.35	4.26	0.72
std	72.37	48.07	238.52	47.78	23.57	6.58	5.49	1.70
	gross_income		km_to_dest		visits_dif_weeks		total_high_season	
	Residents	Others	Residents	Others	Residents	Others	Residents	Others
mean	16,084	25,007.07	1.02	374.73	4.57	1.48	27.53	3.84
std	0.00	7671.19	0.59	486.97	4.03	1.97	14.75	9.00
	total_holiday		avg_visit		std_visit		population	
	Residents	Others	Residents	Others	Residents	Others	Residents	Others
mean	52.54	6.83	23.60	10.54	20.26	4.15	406.66	19,8175.90
std	23.71	15.06	34.85	31.87	23.35	16.05	121.16	56,7183.30

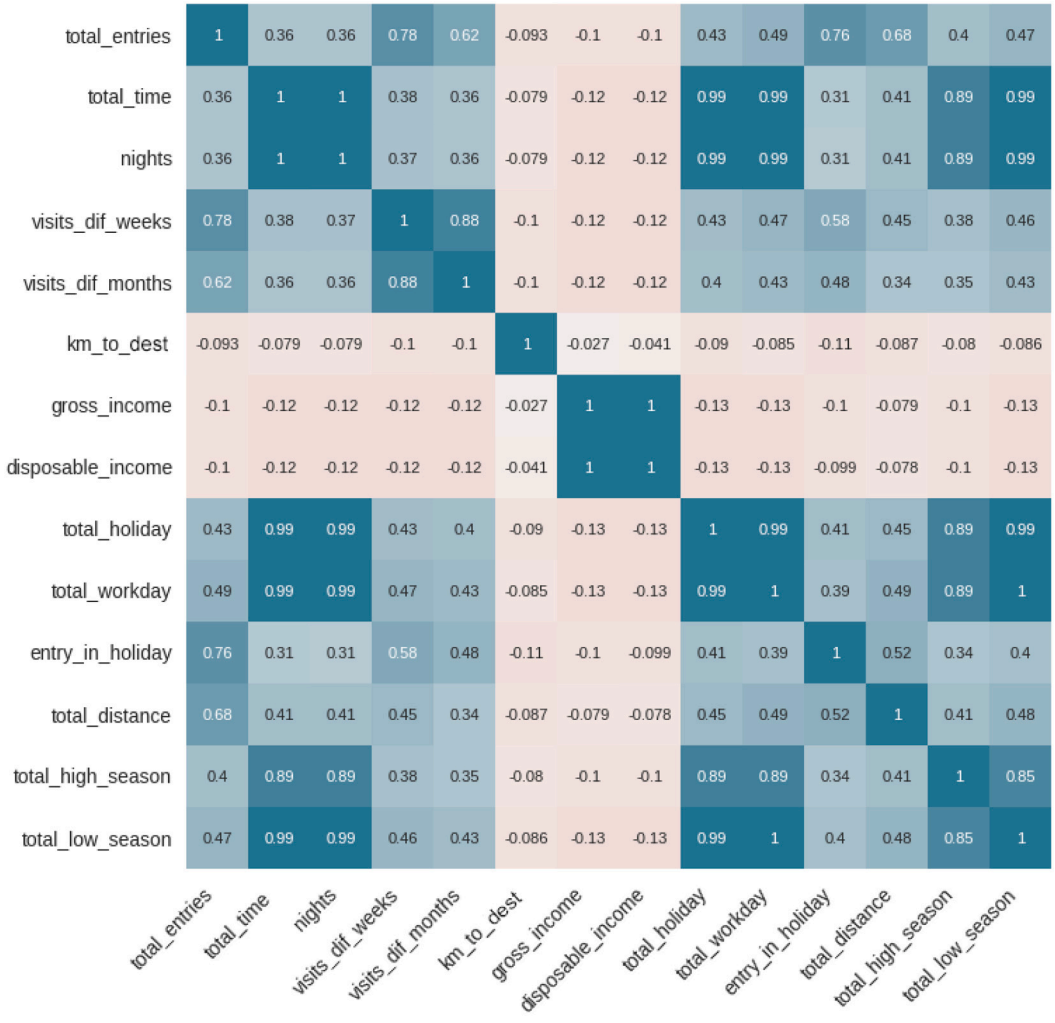


Fig. A.2. Correlation matrix for all variables in the proposed dataset.

Investigación / 10.13039/501100011033, and the R&D&i Project Ref. C-SEJ-128-UGR23 funded by Junta de Andalucía and "ERDF A way of making Europe", and also by the project "Thematic Center on Mountain Ecosystem & Remote sensing, Deep learning-AI e-Services University of Granada-Sierra Nevada" (LifeWatch-2019-10-UGR-01), which has been co-funded by the Ministry of Science and Innovation through the FEDER funds from the Spanish Pluriregional Operational Program 2014–2020 (POPE), LifeWatch-ERIC action line. The project has also been co-financed by the Provincial Council of Granada. Funding for open access charge: Universidad de Granada / CBUA.

Appendix. Supplementary correlation and variable statistics

See Figs. A.1 and A.2 and Table A.1.

References

[1] L. Atzori, A. Iera, G. Morabito, The internet of things: A survey, Comput. Netw. 54 (15) (2010) 2787–2805.

[2] M. Bermudez-Edo, P. Barnaghi, K. Moessner, Analysing real world data streams with spatio-temporal correlations: Entropy vs. Pearson correlation, Autom. Constr. 88 (2018) 87–100.

[3] F.M. Garcia-Moreno, M. Bermudez-Edo, E. Rodríguez-García, J.M. Pérez-Mármol, J.L. Garrido, M.J. Rodríguez-Fórtiz, A machine learning approach for semi-automatic assessment of IADL dependence in older adults with wearable sensors, Int. J. Med. Inf. 157 (2022) 104625.

[4] R.P. Centelles, F. Freitag, R. Meseguer, L. Navarro, S.F. Ochoa, R.M. Santos, A lora-based communication system for coordinated response in an earthquake aftermath, Multidiscip. Digit. Publ. Inst. Proc. 31 (1) (2019) 73.

[5] M.A. Mondal, Z. Rehena, Identifying traffic congestion pattern using k-means clustering technique, in: 2019 4th International Conference on Internet of Things: Smart Innovation and Usages, IoT-SIU, IEEE, 2019, pp. 1–5.

[6] M. Lin, X. Zhao, Application research of neural network in vehicle target recognition and classification, in: 2019 International Conference on Intelligent Transportation, Big Data & Smart City, ICITBS, IEEE, 2019, pp. 5–8.

[7] M.L.M. Peixoto, A.H. Maia, E. Mota, E. Rangel, D.G. Costa, D. Turgut, L.A. Villas, A traffic data clustering framework based on fog computing for VANETs, Veh. Commun. 31 (2021) 100370.

[8] Z. Ning, J. Huang, X. Wang, Vehicular fog computing: Enabling real-time traffic management for smart cities, IEEE Wirel. Commun. 26 (1) (2019) 87–93.

[9] Ş. Kolozali, M. Bermudez-Edo, N. FarajiDavar, P. Barnaghi, F. Gao, M.I. Ali, A. Mileo, M. Fischer, T. Iggena, D. Kuemper, et al., Observing the pulse of a city: A smart city framework for real-time discovery, federation, and aggregation of data streams, IEEE Internet Things J. 6 (2) (2018) 2651–2668.

[10] O. Golovnin, Data-driven profiling of traffic flow with varying road conditions.

[11] G. Yang, D. Coble, C. Vaughan, C. Peele, A. Morsali, G.F. List, D.J. Findley, Waiting time estimation at ferry terminals based on license plate recognition, J. Transp. Eng. A: Syst. 148 (9) (2022) 04022064.

[12] W. Yao, J. Yu, Y. Yang, N. Chen, S. Jin, Y. Hu, C. Bai, Understanding travel behavior adjustment under COVID-19, Commun. Transp. Res. (2022) 100068.

[13] P. Wang, J. Lai, Z. Huang, Q. Tan, T. Lin, Estimating traffic flow in large road networks based on multi-source traffic data, IEEE Trans. Intell. Transp. Syst. 22 (9) (2020) 5672–5683.

[14] Z. Liu, Y. Liu, Q. Meng, Q. Cheng, A tailored machine learning approach for urban transport network flow estimation, Transp. Res. C 108 (2019) 130–150.

[15] H. Sun, Y. Chen, J. Lai, Y. Wang, X. Liu, Identifying tourists and locals by K-means clustering method from mobile phone signaling data, J. Transp. Eng. A: Syst. 147 (10) (2021) 04021070.

[16] C. Morris, J.J. Yang, A machine learning model pipeline for detecting wet pavement condition from live scenes of traffic cameras, Mach. Learn. Appl. 5 (2021) 100070.

[17] J. Enes, R.R. Expósito, J. Fuentes, J.L. Cacheiro, J. Touriño, A pipeline architecture for feature-based unsupervised clustering using multivariate time series from HPC jobs, Inf. Fusion 93 (2023) 1–20.

[18] B.P.L. Lau, S.H. Marakkalage, Y. Zhou, N.U. Hassan, C. Yuen, M. Zhang, U.-X. Tan, A survey of data fusion in smart city applications, Inf. Fusion 52 (2019) 357–374.

[19] F.T. Sáenz, F. Arcas-Tunez, A. Muñoz, Nation-wide touristic flow prediction with Graph Neural Networks and heterogeneous open data, Inf. Fusion 91 (2023) 582–597.

[20] Z. Doborjeh, N. Hemmington, M. Doborjeh, N. Kasabov, Artificial intelligence: a systematic review of methods and applications in hospitality and tourism, Int. J. Contemp. Hosp. Manag. 34 (3) (2022) 1154–1176.

[21] D. Bolaños-Martinez, M. Bermudez-Edo, J.L. Garrido, Clustering study of vehicle behaviors using license plate recognition, in: Proceedings of the International Conference on Ubiquitous Computing & Ambient Intelligence, UCAmI 2022, Springer, 2022, pp. 784–795.

[22] M. Mallik, A.K. Panja, C. Chowdhury, Paving the way with machine learning for seamless indoor-outdoor positioning: A survey, Inf. Fusion (2023).

[23] O. Cats, F. Ferranti, Unravelling individual mobility temporal patterns using longitudinal smart card data, Res. Transp. Bus. Manag. 43 (2022) 100816.

[24] A. Gutiérrez, A. Domènech, B. Zaragozí, D. Miravet, Profiling tourists' use of public transport through smart travel card data, J. Transp. Geogr. 88 (2020) 102820.

[25] Z. Wang, H. Liu, Y. Zhu, Y. Zhang, A. Basiri, B. Büttner, X. Gao, M. Cao, Identifying urban functional areas and their dynamic changes in Beijing: using multiyear transit smart card data, J. Urban Plann. Dev. 147 (2) (2021) 04021002.

[26] F.T. Lima, V.M. Souza, A large comparison of normalization methods on time series, Big Data Res. (2023) 100407.

[27] M. Nicholson, R. Agrahari, C. Conran, H. Assem, J.D. Kelleher, The interaction of normalisation and clustering in sub-domain definition for multi-source transfer learning based time series anomaly detection, Knowl.-Based Syst. 257 (2022) 109894.

[28] W. Yao, C. Chen, H. Su, N. Chen, S. Jin, C. Bai, Analysis of key commuting routes based on spatiotemporal trip chain, J. Adv. Transp. 2022 (2022).

[29] S. Wold, K. Esbensen, P. Geladi, Principal component analysis, Chemometr. Intell. Lab. Syst. 2 (1–3) (1987) 37–52.

[30] C.C.D. Oliveira, V.M.D.A. Calado, G. Ares, D. Granato, Statistical approaches to assess the association between phenolic compounds and the in vitro antioxidant activity of Camellia sinensis and Ilex paraguariensis teas, Crit. Rev. Food Sci. Nutr. 55 (10) (2015) 1456–1473.

[31] M. Halkidi, Y. Batistakis, M. Vazirgiannis, Clustering algorithms and validity measures, in: Proceedings Thirteenth International Conference on Scientific and Statistical Database Management, SSDBM 2001, IEEE, 2001, pp. 3–22.

[32] W. Yao, M. Zhang, S. Jin, D. Ma, Understanding vehicles commuting pattern based on license plate recognition data, Transp. Res. C 128 (2021) 103142.

[33] S. Pasupathi, V. Shanmuganathan, K. Madasamy, H.R. Yesudhas, M. Kim, Trend analysis using agglomerative hierarchical clustering approach for time series big data, J. Supercomput. 77 (2021) 6505–6524.

[34] B. Yu, J. Xiong, A novel WSN traffic anomaly detection scheme based on BIRCH, J. Electron. Inf. Technol. 44 (1) (2022) 305–313.

[35] K. Kim, Spatial contiguity-constrained hierarchical clustering for traffic prediction in bike sharing systems, IEEE Trans. Intell. Transp. Syst. 23 (6) (2021) 5754–5764.

[36] X. Bai, Z. Ma, Y. Hou, D. Yang, A data-driven iterative multi-attribute clustering algorithm and its application in port congestion estimation, 2022, Available at SSRN 4086627.

[37] A. Belhadi, Y. Djenouri, G. Srivastava, D. Djenouri, J.C.-W. Lin, G. Fortino, Deep learning for pedestrian collective behavior analysis in smart cities: A model of group trajectory outlier detection, Inf. Fusion 65 (2021) 13–20.

[38] A.J. Martín, I.M. Gordo, J.J.G. Domínguez, J. Torres-Sospedra, S.L. Plaza, D.G. Gómez, Affinity propagation clustering for older adults daily routine estimation, in: 2021 International Conference on Indoor Positioning and Indoor Navigation, IPIN, IEEE, 2021, pp. 1–7.

[39] S. Zhao, K. Zhao, Y. Xia, W. Jia, Hyper-clustering enhanced spatio-temporal deep learning for traffic and demand prediction in bike-sharing systems, Inform. Sci. 612 (2022) 626–637.

[40] F.S. de Moura, C.T. Nodari, Application of the Affinity Propagation Clustering Technique to obtain traffic accident clusters at macro, meso, and micro levels, 2022, arXiv preprint arXiv:2202.05175.

[41] B. Priambodo, A. Ahmad, R.A. Kadir, Predicting traffic flow propagation based on congestion at neighbouring roads using hidden Markov model, IEEE Access 9 (2021) 85933–85946.

[42] J. Park, J. Jeong, Y. Park, Ship trajectory prediction based on bi-LSTM using spectral-clustered AIS data, J. Mar. Sci. Eng. 9 (9) (2021) 1037.

[43] H. Li, J.S.L. Lam, Z. Yang, J. Liu, R.W. Liu, M. Liang, Y. Li, Unsupervised hierarchical methodology of maritime traffic pattern extraction for knowledge discovery, Transp. Res. C 143 (2022) 103856.

[44] Y. Liu, Z. Li, H. Xiong, X. Gao, J. Wu, Understanding of internal clustering validation measures, in: 2010 IEEE International Conference on Data Mining, IEEE, 2010, pp. 911–916.

[45] A. Oliveira-Brochado, F.V. Martins, et al., Assessing the number of components in mixture models: a review, in: FEP Working Papers (194), Universidade do Porto, Faculdade de Economia do Porto, 2005.

[46] C. Olivier, F. Jouzel, A. Matouat, Choice of the number of component clusters in mixture models by information criteria, in: Proc. Vision Interface, 1999, pp. 74–81.

[47] Z. Hu, Initializing the EM Algorithm for Data Clustering and Sub-Population Detection (Ph.D. thesis), The Ohio State University, 2015.

[48] J.-P. Baudry, CLADAG 2015. Book of abstracts, ISBN: 978888467749-9, 2015, Ch. Estimation and model selection for model-based clustering with the conditional classification likelihood.

[49] G. James, D. Witten, T. Hastie, R. Tibshirani, An Introduction to Statistical Learning, Vol. 112, Springer, 2013.

[50] J.A. Rodrigo, Análisis de Componentes Principales (Principal Component Analysis, PCA) y t-SNE, 2017, Cienciadedatos.Net. available under a Attribution 4.0 International (CC BY 4.0). (Accessed 29 March 2023).

[51] H. Henderi, T. Wahyuningsih, E. Rahwanto, Comparison of min-max normalization and Z-score normalization in the K-nearest neighbor (kNN) algorithm to test the accuracy of types of breast cancer, Int. J. Inf. Inf. Syst. 4 (1) (2021) 13–20.

[52] S. Patro, K.K. Sahu, Normalization: A preprocessing stage, 2015, arXiv preprint arXiv:1503.06462.

[53] K. Polat, U. Sentürk, A Novel ML Approach to Prediction of Breast Cancer: Combining of mad normalization, KMC based feature weighting and AdaBoostM1 classifier, in: 2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies, ISMSIT, IEEE, 2018, pp. 1–4.

[54] M. Ayub, E.-S.M. El-Alfy, Impact of normalization on BiLSTM based models for energy disaggregation, in: 2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy, ICDABI, IEEE, 2020, pp. 1–6.

[55] R. Gallardo García, B. Beltrán, D. Vilariño, C. Zepeda, R. Martínez, Comparison of clustering algorithms in text clustering tasks, Comput. Sist. 24 (2) (2020) 429–437.

[56] S.D. Whitaker, Did the COVID-19 pandemic cause an urban exodus? Clevel. Fed Dist. Data Brief (20210205) (2021).

[57] V. Pinilla, M.-I. Ayuda, L.-A. Sáez, Rural depopulation and the migration turnaround in Mediterranean Western Europe: a case study of Aragon, J. Rural Commun. Dev. 3 (1) (2008).

[58] Á.D.R. Escudero, La Alpujarra granadina: un espacio rural diverso y complejo. De Sierra Nevada al litoral, in: Nuevas realidades rurales en tiempos de crisis: territorios, actores, procesos y políticas: XIX Coloquio de Geografía Rural de la Asociación de Geógrafos Españoles y II Coloquio Internacional de Geografía Rural, Universidad de Granada, 2018, pp. 782–794.

[59] A. Bertuglia, S. Sayadi, A. Guarino, C. López, et al., Reverse migration: from the city to the countryside. The Spanish case of Alpujarra Granadina, Agriregionieuropa 7 (27) (2011) 62–64.

[60] V. Rodriguez, G. Fernandez-Mayoralas, F. Rojo, International retirement migration: Retired Europeans living on the Costa del Sol, Spain, Popul. Rev. 43 (1) (2004) 1–36.

[61] D. Reynolds, Gaussian mixture models, in: S.Z. Li, A. Jain (Eds.), Encyclopedia of Biometrics, Springer US, Boston, MA, 2009, pp. 659–663.