Soft Computing (2020) 24:15529–15549
https://doi.org/10.1007/s00500-020-04881-0

METHODOLOGIES AND APPLICATION

# A new hybridization of DBSCAN and fuzzy earthworm optimization algorithm for data cube clustering

Mina Hosseini Rad[1] · Majid Abdolrazzagh-Nezhad[2]

Published online: 6 April 2020


**Abstract**
Data aggregation from different databases into a data warehouse creates multidimensional data such as data cubes. With regard to the 3D structure of data, data cube clustering has significant challenges to perform on data cube. In this paper, new preprocessing techniques and a novel hybridization of DBSCAN and fuzzy earthworm optimization algorithm (EWOA) are proposed to solve the challenges. Proposed preprocessing consists of an assigned address to each cube cell and dimension move to create a related 2D data from the data cube and new similarity metric. The DBSCAN algorithm, as a density-based clustering algorithm, is adopted based on both Euclidean and newly proposed similarity metric, which are called DBSCAN1 and DBSCAN2 for the related 2D data. A new hybridization of the EWOA and DBSCAN is proposed to improve the DBSCAN, and it is called EWOA–DBSCAN. Also, to dynamically tune parameters of EWOA, a fuzzy logic controller is designed with two fuzzy group rules of Mamdani (EWOA–DBSCAN-Mamdani) and Sugeno (EWOA–DBSCAN-Sugeno), separately. These ideas are proposed to present efficient and flexible unsupervised analysis for a data cube by utilizing a meta-heuristic algorithm to optimize DBSCAN's parameters and increasing the efficiency of the idea by applying dynamic tuning parameters of the algorithm. To evaluate the efficiency, the proposed algorithms are compared with DBSCAN1 and GA-DBSCAN1, GA-DBSCAN1-Mamdani and GA-DBSCAN1-Sugeno. The experimental results, consisting of 20 runs, indicate that the proposed ideas achieved their targets.

**Keywords** Data cube · Dimension move · DBSCAN clustering · Fuzzy logic controller · Dynamic tuning parameters · Earthworm optimization algorithm

## 1 Introduction

There is a natural requirement for the effective methods for accessing data and extracting useful knowledge, with regard to the increasing expansion of data on different storage media. Data mining consists of the most effective methods in this field. The data mining is an iterative process in order to make the discovery of knowledge which is done manually and automatically. The data mining searches valuable and new information from the huge volume of data (Gnanapriya et al. 2010).

Description and prediction are the main aims of the data mining. In the first category, data attributes are described in a dataset and its focus is about finding patterns from the dataset so that the found patterns can be described by human. The second category is based on data deduction, looking for unknown variables and values of the data (Mining 2006). Each of these categories includes different patterns such as exploring frequent patterns, classification and regression, clustering and exploring outline patterns, each of which has its own application and features. The aim of this study is to investigate the clustering analysis, which is part of the descriptive patterns with regard to the type of data used for data mining (Pujari 2001).

In clustering, we can create a grouping of data and so, its main aim is to maximize similarity between samples of a cluster as well as minimizing similarity between samples of

Communicated by V. Loia.

✉ Majid Abdolrazzagh-Nezhad
abdolrazzagh@buqaen.ac.ir

Mina Hosseini Rad
hosseinirad.edu@gmail.com

[1] Department of Computer Engineering, Birjand Branch, Islamic Azad University, Birjand, Iran

[2] Department of Computer Engineering, Faculty of Engineering, Bozorgmehr University of Qaenat, Qaen, Iran

the various clusters (Berkhin 2006). The clustering widely helps discovery of unknown patterns in data and has a vast application in the various fields including Web searching, security and business intelligence as well (Vercellis 2011; Han et al. 2011). The data mining in business intelligence as a powerful and advanced technology will enable companies to have more focus on important data in data warehouses. It can help corporations to adopt effectively knowledge-based decisions in order to increase business profits using data mining tools (Hema and Malik 2010).

The multidimensional data analysis is one of the most important factors in improving efficiency and increasing the data mining speed in business intelligence. In this study, due to clustering, data cube-contained datasets were used; this type of data provides the possibility of analysis in various aspects. In the following, some of the works done in the data cube are investigated.

The data warehouses that contain collected data from data sources and are around a specific topic provide possible widespread. The data require the complex analysis for managers by using OLAP tools (Hema and Malik 2010). The data warehouse and OLAP tools are based on a multidimensional data model; therefore, the data cube is the best concept for data modeling in several dimensions, in which data are represented by dimensions and facts. In addition, it is possible to use the OLAP operation in order to create views, interactive query and perform data analysis in the data cube (Chaudhuri and Dayal 1997). With regard to Liço (2017), OLAP is introduced as the main component of business intelligence and data cube is considered as an OLAP's main component. Moreover, it considers the data cube as the most powerful tool for using in Big Databases. The study introduces intelligent cube in order to reduce system response time and also addresses using compression techniques to reduce storage memory space.

Introducing clustering algorithm for modeling of the data cube and collecting information from cuboid has been already done by Woo et al. (2015). In this study, the amounts of special attributes contain flow of large data and cuboid is used for saving different parts of flow data and so clustering is carried out on this type of data. A research was done on hierarchical-based clustering algorithm (Ceci et al. 2015) through continuous data, and their aim was using the algorithm in applications including wireless sensor network.

In the current research, data cube clustering is considered to prepare an efficient unsupervised analysis through the data. The challenge of cube data clustering is the existence of irregular and specific data (when data size is high), which makes it difficult to cluster. There are several approaches for clustering (Joshi and Kaur 2013), which include partitioning method, hierarchical-based clustering, density-based clustering and grid-based clustering, and among these four approaches, only a density-based approach has the ability to identify nonconvex clusters.

Therefore, in order to achieve higher efficiency, we use density-based clustering methods. Among the density-based clustering methods, the method of DBSCAN is widely used in comparison with other methods. The popularity reasons of the DBSCAN are its simplicity to performance and its ability to recognize clusters with different sizes and nonconvex shapes (Berkhin 2006; Cheng 2017). Hence, in the current research, the DBSCAN algorithm is selected for density-based clustering. The DBSCAN is a very good candidate to find nonconvex clusters in data space (Kumar and Reddy 2016). The challenge of the DBSCAN clustering is the cluster's dependence on its parameters such as the neighborhood radius and the minimum points. These parameters are empirically chosen according to the type of data. So, the fine-tuning of these parameters has a significant role in identifying proper clusters.

There are several studies which tried to improve DBSCAN. In Smiti and Eloudi (2013), fuzzy set theory was applied to design fuzzy clustering and improve DBSCAN that the authors called Soft DBSCAN. The Soft DBSCAN is a new fuzzy clustering, which offers appropriate primal degrees for data's membership to express proximities of data entities to the cluster centers. A graph-based index structure method group (Kumar and Reddy 2016) was proposed to improve the performance of DBSCAN on high-dimensional dataset that accelerated the neighbor search operations. A new measurement criterion (Cheng 2017) was utilized to obtain a distance which was calculated based on the threshold analysis of the nearest neighbor with the total neighbors. Darong and Peng (2012) combined partition technique with DBSCAN. The goal was to obtain the proper input parameters for DBSCAN. However, the effectiveness of this method was not evaluated for datasets with different densities. A combination of Gaussian-means method with DBSCAN (Smiti and Eloudi 2012) was proposed to improve the determination of DBSCAN parameters. However, Gaussian-means creates circular clusters that are not density-based and do not act very well against dense data as well. The DBSCAN clustering was combined with binary differential evolution (Karami and Johansson 2014) to determine the parameters of the DBSCAN. Recently, many meta-heuristic algorithms have been presented to improve clustering on various algorithms for reducing clustering sensitivity to the important parameters of the algorithms (Chen 2012; Pei et al. 2008; Zhao 2007; Aydilek and Arslan 2013).

Among them, there is a lack of improvement in the DBSCAN as a density-based clustering with a meta-heuristic algorithm such as earthworm optimization algorithm (EWOA) (Wang et al. 2018). Therefore, in this

paper, the EWOA is considered to identify the best parameters for the DBSCAN. The proposed clustering algorithm, which is called the improved DBSCAN (EWOA–DBSCAN), contains a hybridization of the DBSCAN with the EWOA on the obtained 2D data from the data cube based on the new dimension move and similarity metric.

The EWOA's challenge is tuning Reproduction 1 and Reproduction 2 parameters that are also empirically determined. The parameters have a significant impact on the efficiency and convergence of the algorithm. There have been several attempts to dynamically tune the other meta-heuristic parameters such as proposing a diversity measure between GA's chromosomes in the population, designing a variation depending on the fitness function values of GA's chromosomes and planning fuzzy logic controllers (FLCs) (Angelova and Pencheva 2011; Karafotias et al. 2015), but this type of tuning parameters has not been considered for the EWOA. An FLC is constructed with a knowledge base, which consists of the information given by linguistic control rules, a fuzzification interface, an inference system and a defuzzification interface (Liu and Lampinen 2005; Herrera and Lozano 2003). Therefore, in order to modify the EWOA–DBSCAN, another data cube clustering algorithm is proposed, which attempts to improve accurate selection of effective EWOA parameters by using a new simple FLC based on two group linguistic control rules such as Mamdani's rules and Takagi–Sugeno's rules. We name the second algorithm as "soft improved DBSCAN" (EWOA–DBSCAN- Mamdani and EWOA–DBSCAN-Sugeno). The proposed ideas and their achievements will be considered to design 3D clustering in our future work.

The rest of the paper is organized as follows. In Sect. 2, we describe preprocessing of data cube based on data cube structure, assigning an address to each cube cell, designing a dimension move to create a related 2D data from the data cube and proposing a new similarity metric. The proposed clustering algorithms are explained in Sect. 3 and consist of three investigated evaluation indices, DBSCAN algorithm, improved and soft improved DBSCAN algorithms. Then, in the following, the experimental results are illustrated in Sect. 4. Finally, conclusion and future works are explained in Sect. 5.

## 2 Preprocessing of data cube

In order to smoothly explain the proposed clustering algorithms and demonstrate their scalability, it is necessary to first describe in four steps the cube data structure, the normalizing data cube, the dimension move and the new similarity metric in this section. To clarify these

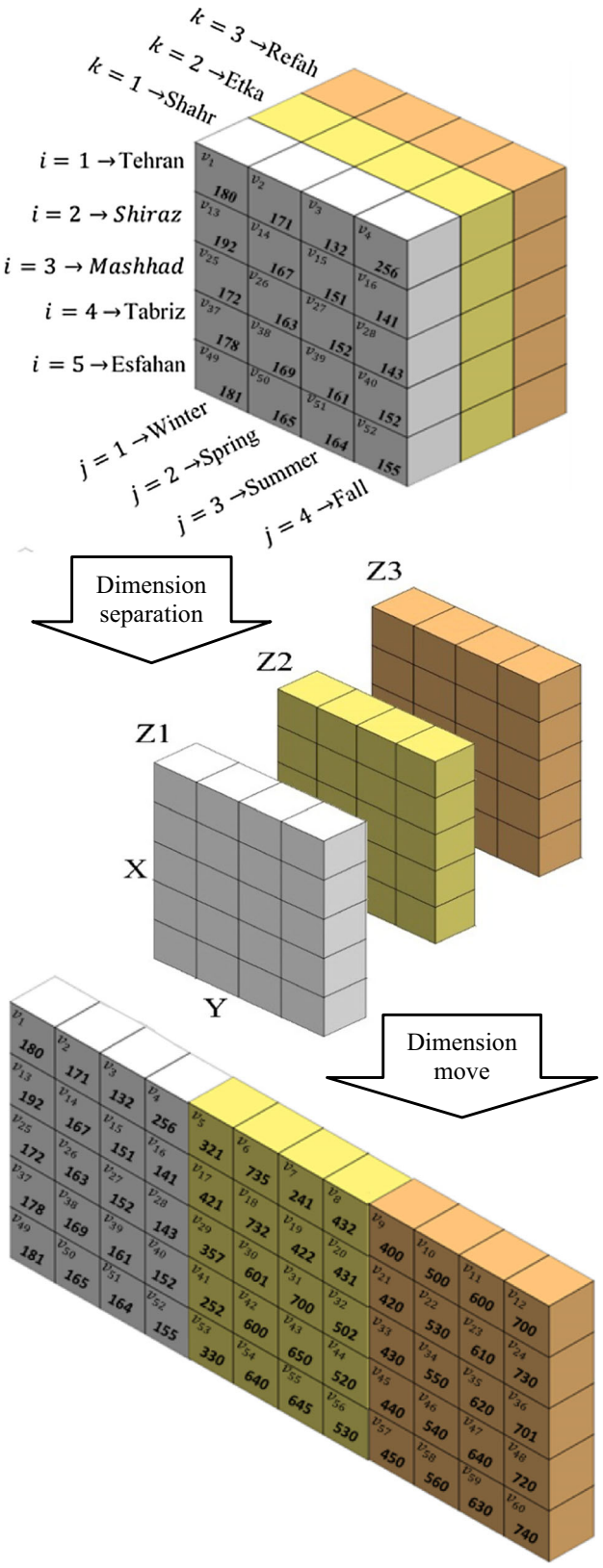



**Fig. 1** A data cube's structure, dimension separation and dimension move

descriptions, a data cube sample is designed with size $5 \times 4 \times 3$, which is shown in Fig. 1. The sample's values belong to the sales of three well-known hypermarkets in Iran, namely Refah (Ref), Etkah (Etk) and Shahrvand (Sha). The sales are obtained in five cities in Iran, namely Tehran (Teh), Shiraz (Shi), Mashhad (Mash), Tabriz (Tab) and Esfahan (Esf). Each city's hypermarket sales have been measured in four seasons, namely winter, spring, summer and fall. It is assumed that the cities are represented by $X$ ($|X| = 5$), the seasons by $Y$ ($|Y| = 4$) and the hypermarkets by $Z$ ($|Z| = 3$). So, the steps of preprocessing are described as follows.

## 2.1 Data cube structure

A normal datum is a two-dimensional array of values in which each row is called an object and each column is called as attribute. Since, data cube is a multidimensional array of values that it has more than 2D. Each dimension of data cube belongs to an attribute. The investigated data cube of the current research consists of 3D array of values, which is called $\text{cube}(x, y, z) = \text{value}$. In Fig. 1, there are three attributes, namely city ($x$), season ($y$) and hypermarket ($z$), and for example, $\text{cube}(3, 2, 1) = 163$ represents 163 sale units of Shahrvand Hypermarket on Mashhad city in spring. In the research, an address is assigned to each cube cell based on the following function:

$$V_r : r = |Z \times Y| \times (i-1) + |Y| \times (k-1) + j \tag{1}$$

where $i$, $j$ and $k$ represent the cell indices for $X$, $Y$ and $Z$, respectively. For example, in Fig. 1, $v_{42}$ is the address of $\text{cube}(4, 2, 2) = 600$. Also, the above function has the following inverse function:

$$i = \begin{cases} \text{div}(r, |Z \times Y|), & \text{if } \text{mod}(r, |Z \times Y|) = 0 \\ \text{div}(r, |Z \times Y|) + 1, & \text{otherwise} \end{cases} \tag{2}$$

$$k = \begin{cases} \text{div}((r - |Z \times Y| \times (i-1)), |Y|), & \text{if } \text{mod}((r - |Z \times Y| \times (i-1)), |Y|) = 0 \\ \text{div}((r - |Z \times Y| \times (i-1)), |Y|) + 1, & \text{otherwise} \end{cases} \tag{3}$$

$$j = r - |Z \times Y| \times (i-1) - |Y| \times (k-1) \tag{4}$$

The address function and its inverse create a two-way relationship between 3D and 2D search spaces. Because the proposed data cube clustering converts the 3D space of data cube into a two-dimensional space by moving one dimension along the other dimensions using the Dimension Move Algorithm.

## 2.2 Normalizing data cube

There are different scaling sizes between the 3D attributes of the data, and application of normalizing the attributes causes removal of the effect of larger scale attributes on smaller one. The min–max normalization (Han et al. 2011) is a linear converter of $v_i \in A$ into $v'_i \in A'$ with new bound $[\text{new\_min}_{A'}, \text{new\_max}_{A'}]$. The min–max normalization was proposed for 2D data, and a 3D draft of the normalization (see Fig. 1), which is linearly converted to $[0, 1]$, is designed in the current research whose procedure is shown for the first layer as follows:

$$\begin{cases} \text{cube}'(i,j,1) = \dfrac{\text{cube}(i,j,1) - \min\limits_{\substack{1 \le i \le |X| \\ 1 \le j \le |Y|}} \text{cube}(i,j,1)}{\max\limits_{\substack{1 \le i \le |X| \\ 1 \le j \le |Y|}} \text{cube}(i,j,1) - \min\limits_{\substack{1 \le i \le |X| \\ 1 \le j \le |Y|}} \text{cube}(i,j,1)} \\ \vdots \\ \text{cube}'(i,j,|Z|) = \dfrac{\text{cube}(i,j,|Z|) - \min\limits_{\substack{1 \le i \le |X| \\ 1 \le j \le |Y|}} \text{cube}(i,j,|Z|)}{\max\limits_{\substack{1 \le i \le |X| \\ 1 \le j \le |Y|}} \text{cube}(i,j,|Z|) - \min\limits_{\substack{1 \le i \le |X| \\ 1 \le j \le |Y|}} \text{cube}(i,j,|Z|)} \end{cases} \tag{5}$$

where $i = 1, \ldots, |X|$ and $j = 1, \ldots, |Y|$.

## 2.3 Dimension move

There are technical reshape data cube and 3D matrix such as Scovanner et al. (2007), Zhao and Yang (2015) and Johnson et al. (2013) which have different complexity levels and are applied in image processing. In the current research, a dimension move is designed to convert a 3D matrix into a 2D matrix (see Fig. 1). To achieve the aim, first, sub-2D matrixes of the 3D matrix are considered by dimension separation along a given dimension. For example, three sub-2D matrixes ($X \times Y$) are separated along $Z$ in Fig. 1. Second, the 2D matrixes are joined together along one of their dimensions. In Fig. 1, the 2D separated matrixes are joined together along $Y$ which create a 2D matrix with $|X|$ rows and $|Y| \times |Z|$ columns. The procedure of dimension move is presented in Algorithm 1.

Algorithm 1 : Dimension Move

**Input:** $Cube()$ as a given 3D matrix
1: for $i = 1 : size(Cube, 1)$
2: Count = 1;
3: for $j = 1 : size(Cube, 2)$.
4: for $k = 1 : size(Cube, 3)$
5: $2D\_Data(i, Count) = Cube(i, j, k)$;
6: Count++;
7: end for 3
8: end for 2
9: end for 1

### 2.4 New similarity metric

As mentioned, each cell of a data cube is indicated with four values such as Cube$(i, j, k) =$ value, where $i, j, k$ are dimension values of $X$ ($X$Cube), $Y$ ($Y$Cube) and $Z$ ($Z$Cube), respectively. Because the obtained 2D data from a given data cube consist of quantitative values for its rows and column labels, a new similarity metric is proposed to calculate distance between rows $A$ and $B$ in the 2D data. The similarity's procedure is presented in Algorithm 2. The similarity metric is utilized to calculate dissimilarity of two rows in the procedure of DBSCAN clustering (Algorithm 3) and the investigation evaluation indices between the samples in each cluster [(7) and (8)].

Algorithm 2 : Similarity Metric

**Input:** $Cube()$ as a given 3D matrix, $r_A$ and $r_B$ are assigned addresses of each cell in rows $A$ and $B$, respectively.
1: for $count = 1 : size(row\ A)$
2: Calculate $(i_A, j_A, k_A)$ by (2), (3) and (4) from $r_A^{count}$;
3: Calculate $(i_B, j_B, k_B)$ by (2), (3) and (4) from $r_B^{count}$;
4: if $XCube(i_A, j_A, k_A) == XCube(i_B, j_B, k_B)$ then
5: $XValue = 0$;
6: else
7: $XValue = 1$;
8: if $YCube(i_A, j_A, k_A) == YCube(i_B, j_B, k_B)$ then
9: $YValue = 0$;
10: else
11: $YValue = 1$;
12: if $ZCube(i_A, j_A, k_A) == ZCube(i_B, j_B, k_B)$ then
13: $ZValue = 0$;
14: else
15: $ZValue = 1$;
16: $dis += |Cube(i_A, j_A, k_A) - Cube(i_B, j_B, k_B)| + \delta(XValue + YValue + ZValue)$;
17: end for

Here, $\delta$ is an effectiveness parameter for comparing quantitative values of $X$Cube, $Y$Cube and $Z$Cube. The parameter is tuned in the range of $[0.2, 0.4]$, since 3D matrix was normalized.

## 3 The proposed clustering algorithms

With regard to the proposed preprocessing, the data cube with its 3D matrix structure was converted to a related 2D data by the dimension move (Algorithm 1) and the distance between rows A and B were calculated by a new similarity metric (Algorithm 2) in the 2D data. Now, two improved drafts of density-based clustering are proposed to solve data cube clustering. In the next subsections, first, three investigation evaluation indices are explained. Second, DBSCAN algorithm is presented and also the challenge of DBSCAN algorithm and novel strategies are proposed to improve it.

### 3.1 The investigation evaluation indices

Three investigation evaluation indices, namely the Davies–Bouldin index (DBI) (Davies and Bouldin 1979), Dunn index (DI) (Bezdek and Pal 1995) and Silhouette index (SI) (Rousseeuw 1987), are considered to evaluate the obtained clusters with DBSCAN. The DBI calculates within-cluster distance and between-cluster distance. The best choice of clusters will be done, since the DBI is minimized and the index is formulated as follows:

$$\mathrm{DBI} = \frac{1}{N}\sum_{i=1}^{N}\left(\max_{\substack{j=1,\ldots,N \\ j\neq i}}\left(\frac{S_i + S_j}{d_{ij}}\right)\right) \tag{6}$$

where $N$ is the number of clusters and $d_{ij}$ is the average linkage as between-cluster distance of clusters $C_i$ and $C_j$. Also, $S_i$ and $S_j$ are the average distance of within-cluster $C_i$ and within-cluster $C_j$, respectively.

$$d_{ij} = \frac{\sum_{p_r\in C_i, p_s\in C_j} p_r - p_s}{C_i C_j} \tag{7}$$

$$S_i = \frac{\sum_{p_r, p_s\in C_i} p_r - p_s}{C_i(C_i - 1)} \tag{8}$$

where $\|\cdot\|$ is Algorithm 2 and $p_r \in C_i$ means point $r$ belongs to the cluster $i$.

The DI focuses on compactness of clusters which are well-separated from others based on inter-cluster distance and cluster's diameter, respectively. So, it calculates the minimum inter-cluster distance divided by the maximum cluster size. The best choice of clusters will be done, since the DI is maximized and the index is formulated as follows:

$$\mathrm{DI} = \frac{\min_{\substack{i,j=1,\ldots,N \\ j\neq i}}(D_{ij})}{\max_{k=1,\ldots,N}(\mathrm{diam}_k)} \tag{9}$$

$$D_{ij} = \min_{p_r\in C_i, p_s\in C_j}\|p_r - p_s\| \tag{10}$$

$$\mathrm{diam}_k = \max_{p,p'\in C_k}\|p - p'\| \tag{11}$$

where $\|\cdot\|$ is Algorithm 2, $D_{ij}$ is the inter-cluster distance between clusters $C_i$ and $C_j$ and $\mathrm{diam}_k$ is the diameter of cluster $C_k$.

The SI focuses on consistency within clusters based on comparison of the similarity of an object to its own cluster to other clusters. The index is calculated for a data point $p_i$ in the cluster $C_i$ and ranges from $-1$ to $+1$, where the values close to 1 show that the object is well matched to its own cluster and poorly matched to neighboring clusters. So, the best choice of clusters will be done, since the DI is maximized by being close to 1 and the SI's average of the all data points is formulated as follows:

$$\mathrm{SI} = \frac{1}{|\mathrm{Data}|}\sum_{i=1}^{|\mathrm{Data}|}\left(\frac{b(p_i)-a(p_i)}{\max\{a(p_i),b(p_i)\}}\right) \tag{12}$$

$$b(p_i) = \min_{\substack{k\neq i \\ k=1,\ldots,N}} \frac{1}{|C_k|}\sum_{p_j\in C_k}\left\|p_i-p_j\right\| \tag{13}$$

$$a(p_i) = \frac{1}{|C_i|-1}\sum_{p_j,p_i\in C_k}\left\|p_i-p_j\right\| \tag{14}$$

where $\|\cdot\|$ is Algorithm 2 and |Data| and $|C_k|$ are the number of all data points and the number of data points in cluster $k$, respectively. $a(p_i)$ is the mean distance between $p_i$ and all other data points in the same cluster, and $b(p_i)$ is the mean dissimilarity of $p_i$ to the other clusters.

### 3.2 DBSCAN algorithm

DBSCAN (Han et al. 2011) is an information clustering method based on the data density whose brief procedure is presented in Algorithm 1. Two parameters, namely the neighborhood radius ($\varepsilon$) and minimum points (MinPts) ($\mu$), needed to form a cluster have been used in order to evaluate the distributed density of points. This algorithm begins from an optional point, and then it accounts for the points which are located in the neighborhood radius of this point at a distance less than $\varepsilon$. If the number of points is more than $\mu$ parameter, they produce a cluster; otherwise, the intended point is known as outliner data. In the next step, this point may be recognized as a part of a cluster. The advantage of this method is the ability to distinguish and separate the outliner data from other data.

Algorithm 3 : DBSCAN Clustering

**Input:** 2*D_Data* obtained by *Algorithm1* as the input data, |Data| objects to be clustered, the neighborhood radius (ε) and minimum points (μ)

1: Randomly select a point $P$
2: Retrieve all points density-reachable from $P$ based on ε and μ and Similarity Metric (*Algorithm2*)
3: If $P$ is a core point, a cluster is formed.
4: If $P$ is a border point, no points are density-reachable from P and DBSCAN selects the next no-visited point randomly.
5: Continue the procedure until all points have been processed.

In this algorithm, the most important role is to find the proper $\varepsilon$ and $\mu$ points. Commonly, by using statistical and classical methods of combining different data mining ways, we can find these points. In many cases, despite consuming too much time, this is not run with high precision. Therefore, in the research, we try to use the earthworm optimization algorithm (EWOA) (Wang et al. 2018), as a metaheuristic algorithm, to estimate the exact values of these parameters and achieve significant improvements.

### 3.3 The improved DBSCAN

To design the improved BDSCAN (EWOA–DBSCAN), the EWOA is adapted to find the optimum values of $P$, $\mu$ and $\varepsilon$ in Algorithm 3. In the adapted EWOA to improve DBCSAN for an obtained 2D dataset (Algorithm 1) with |Data| objects/points and $M$ attributes, each earthworm is an $M+2$-dimensional array such as (15). The first $M$ elements represent an initial point $P$ at which DBSCAN starts. The element of $M+1$ represents the neighboring radius ($\varepsilon$), and the last element represents the value of MinPts ($\mu$).

$$\mathrm{EW}_i = \begin{cases} \min_{ij} \le \mathrm{ew}_{ij} \le \max_{ij}, & \text{if } 1 \le j \le M \\ \mathrm{dis}_{\min} \le \mathrm{ew}_{ij} \le \mathrm{dis}_{\max}, & \text{if } j = M+1 \\ 2 \le \mathrm{ew}_{ij}, & \text{if } j = M+2 \end{cases} \tag{15}$$

$$\min_{ij} = \min_{1\le j\le |\mathrm{Data}|} 2D\_\mathrm{Data}(i,j) \tag{16}$$

$$\max_{ij} = \max_{1\le j\le |\mathrm{Data}|} 2D\_\mathrm{Data}(i,j) \tag{17}$$

$$\mathrm{dis}_{\min} = \min_{\substack{1\le i,r\le |\mathrm{Data}| \\ i\neq r}} \|p_i-p_r\| \tag{18}$$

$$\mathrm{dis}_{\max} = \max_{\substack{1\le i,r\le |\mathrm{Data}| \\ i\neq r}} \|p_i-p_r\| \tag{19}$$

where $i, r = 1,\ldots,\mathrm{pop\_size}$, $\min_{ij}$ and $\max_{ij}$ are the minimum and the maximum values of $j$th columns of $2D$_Data, $\mathrm{dis}_{\min}$ and $\mathrm{dis}_{\max}$ are the minimum and the maximum distances between objects/points. Note that $\min_{ij} = 0$ and $\max_{ij} = 1$ for a normalized 2D data.

The EWOA's inputs are population size (pop_size), similarity factor ($\alpha$), the number of kept earthworms ($n$KEW), maximum generations (MaxGen) and/or other terminate criteria (Carvalho and Freitas 2004; Nagar and Srivastava 2008; Huang 1997; Freitas 2003). In Algorithm 4, a brief procedure of the EWOA is presented based on Initialization, Reproduction 1, Reproduction 2 and Selection.

**Initialization:** With regard to the earthworm's structure (15), pop_size earthworms are generated, as initial population in the zero generation $\mathrm{EW}(0) = \{\mathrm{EW}_1^0,\ldots,\mathrm{EW}_{\mathrm{pop\_size}}^0\}$

randomly that is $\mathrm{EW}_i^0 = \left(\mathrm{ew}_{i1}, \ldots, \mathrm{ew}_{i,M+2}\right)$. To evaluate each earthworm, Algorithm 3 runs and the DBI (6) calculates its fitness function. The worms of EW(0) are sorted in ascending order, and the best one is saved as $\mathrm{EW}_{\mathrm{best}}$.

**Reproduction 1:** The EWOA's procedure includes two strategies to generate new offspring such as Reproduction 1 and Reproduction 2. The behavior of Reproduction 1 is similar mutation in GA, and first the operation is run. The Reproduction 1 has just effects on $\varepsilon$ and $\mu$ values, and one worm ($\mathrm{EW}_i$) gene0 .rates a new offspring ($\mathrm{EW}_i^{\mathrm{new}}$) by the following formula and saves it in $R1(G)$:

Algorithm 4 : EWOA-DBSCAN

**Input:** $pop\_size$, $MaxG$, $\alpha$ and $nKEW$
1: $G = 0$
2: Initialization: Generate $pop\_size$ worms based on (15) randomly as the initial generation EW(G).
3: Evaluate $EW$(G) with run *Algorithm3* and calculate (6).
4: Sort Ascending $EW(G)$ and get its best as $EW_{best}$.
5: while (G≤ $MaxG$) do
6: for $i = 1$: $pop\_size$
7: Generate $EW_i^{new}$ based on (*20*) and save it in $R1(G)$.
8: if $i > nKEW$ then
9: Select a worm from $EW$(G) by Roulette wheel.
10: Run the Scattered Crossover (*Fig.2*) on the selected worms and $EW_i$ and save 2 offspings in $R2(G)$.
11: else
12: Generate $EW_i^{new}$ based on (*21*) and save it in $R2(G)$.
13: end if
14: end for
15: Evaluate $R1(G)$ and $R2(G)$ by *Algorithm3* and (6).
16: Select new generation EW($G + 1$) by Sort Ascending EW($G$)∪ $R1(G) \cup R2(G)$ and save $EW_{best}$.
17: G = G+ 1
18: end while
19: represent the best worm: $EW_{best}$.

$$\mathrm{EW}_i^{\mathrm{new}} = \begin{cases} \mathrm{ew}_{i,j}, & \text{if } 1 \le j \le M \\ \mathrm{dis}_{\max} + \mathrm{dis}_{\min} - \alpha \times \mathrm{ew}_{i,j}, & \text{if } j = M+1 \\ 2 \le \mathrm{ew}_{i,j} + (U(-\alpha, \alpha) \times |\mathrm{Data}|), & \text{if } j = M+2 \end{cases} \tag{20}$$

where $\alpha \in [0, 1]$ is a similarity factor, and it can adjust the distance between the parent earthworm ($\mathrm{EW}_i$) and its offspring ($\mathrm{EW}_i^{\mathrm{new}}$). Decreasing $\alpha$ value causes reduces the distance. $U(-\alpha, \alpha)$ generates random real value in $[-\alpha, \alpha]$.

**Reproduction 2:** There are different strategies to generate offspring in Reproduction 2 since earthworm's index is bigger or smaller than $n$KEW. In the adapted draft of EWOA for the current research, if the index is bigger than $n$KEW ($i > n\mathrm{KEW}$), a earthworm is selected from EW($G$) like $\mathrm{EW}_S$ ($i \neq S$) based on the Roulette wheel. Then, the scattered crossover (Fig. 2) is run on $\mathrm{EW}_S$ and $\mathrm{EW}_i$ to generate two new offspring ($\mathrm{EW}_1^{\mathrm{new}}$ and $\mathrm{EW}_2^{\mathrm{new}}$) and save them in $R2(G)$. If the index is smaller than $n$KEW ($i \le n\mathrm{KEW}$), the earthworm ($\mathrm{EW}_i$) is known as the elite worm and is stored in $R2(G)$ by changing $\varepsilon$ and $\mu$ values as follows:

$$\mathrm{EW}_i^{\mathrm{new}} = \begin{cases} \mathrm{ew}_{i,j}, & \text{if } 1 \le j \le M \\ \mathrm{dis}_{\max} + \mathrm{dis}_{\min} - \dfrac{\alpha}{2} \times \mathrm{ew}_{i,j}, & \text{if } j = M+1 \\ 2 \le \mathrm{ew}_{i,j} + \left(U\left(\dfrac{-\alpha}{2}, \dfrac{\alpha}{2}\right) \times |\mathrm{Data}|\right), & \text{if } j = M+2 \end{cases} \tag{21}$$

**Selection:** There are $\mathrm{EW}(G) \cup R1(G) \cup R2(G)$ earthworms in the last generation $G$, and the evaluation of offspring worms in $R1(G) \cup R2(G)$ is done by running Algorithm 3 and (6). To reduce the number of worms to pop_size one for considering in the next generation ($EW(G+1)$), EW $(G) \cup R1(G) \cup R2(G)$ and the top pop_size worms are sorted in the ascending order. Also, the best one is saved as $\mathrm{EW}_{\mathrm{best}}$.

The challenge for the EWOA is to optimize the determination of the Reproduction 1 ($\alpha$) and Reproduction 2 ($n$KEW) parameters. These parameters are empirically determined and have a significant impact on the efficiency, accuracy and speedup of the algorithm.

### 3.4 The soft improved DBSCAN algorithm

To fill up the above challenge and to design the soft improved DBSCAN (EWOA–DBSCAN–FLC), a new self-adaptive EWOA based on a new fuzzy logic controller (FLC) is designed in this subsection.

Algorithm 5 : EWOA-DBSCAN-FLC

**Input:** $pop\_size$, $MaxG$, $\alpha$ and $nKEW$
1: $G = 0$
2: Initialization: Generate $pop\_size$ worms based on (15) randomly as the initial generation EW(G).
3: Evaluate $EW$(G) with run *Algorithm3* and calculate (6).
4: Sort Ascending EW($G$) and get its best as $EW_{best}$.
5: while (G≤ $MaxG$) do
6: for $i = 1$: $pop\_size$
7: Select a worm from $EW$(G) by Roulette wheel as $EW_S$.
8: Calculate $UG$ and fitBest by (*22*) and (*23*) respectively.
9: Calculate $\alpha$ and $nKEW$ by $FLC(UG, fitBest)$ (*Fig.3*).
10: Generate $EW_i^{new}$ based on (*20*) and save it in $R1(G)$.
11: $nKEW = [nKEW \times pop\_size]$
12: if $i > nKEW$ then
13: Run the Scattered Crossover (*Fig.2*) on the selected worms and $EW_i$ and save 2 offsping in $R2(G)$.
14: else
15: Generate $EW_i^{new}$ based on (*21*) and save it in $R2(G)$.
16: end if
17: end for
18: Evaluate $R1(G)$ and $R2(G)$ by *Algorithm3* and (6).
19: Select new generation EW($G + 1$) by Sort Ascending EW($G$)∪ $R1(G) \cup R2(G)$ and save $EW_{best}$.
20: G = G+ 1
21: end while
22: represent the best worm: $EW_{best}$.

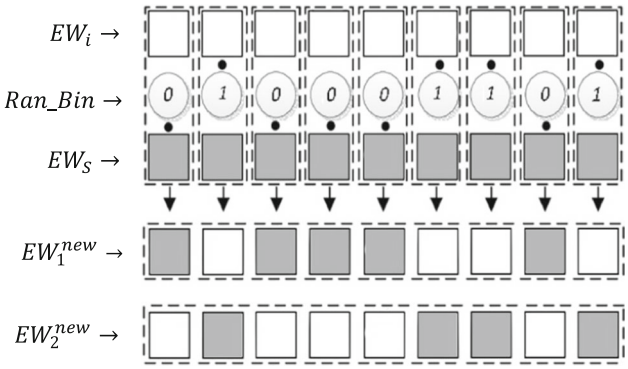



**Fig. 2** Scattered crossover procedure

Algorithm 5 has the procedure like Algorithm 4, except calculating $\alpha$ and $n$KEW by the proposed FLC (see Fig. 3) based on the inputs such as the UG and the fitBest. The FLC's inputs are normalized values related to generation, and the evaluation function (6) that they calculate is as follows:

$$\mathrm{UG} = \frac{G}{\mathrm{Max}G} \tag{22}$$

$$\mathrm{fitBest} = \frac{\mathrm{DBI}(\mathrm{EW}_{\mathrm{best}})}{\min(\mathrm{DBI}(\mathrm{EW}_S), \mathrm{DBI}(\mathrm{EW}_i))} \tag{23}$$

The FLC consists of fuzzifying the inputs, linguistic logic strategy (LLS) and defuzzifying the outputs. The inputs fuzzify based on the presented membership functions in Fig. 4. The LLS includes two main parts: rule based and inference engine. There are two group rules, which are called Mamdani's rules and Takagi–Sugeno's rules, because the FLC is designed to generate dynamic outputs based on Mamdani (Mamdani and Assilian 1975) and Takagi–Sugeno's inferences (Takagi and Sugeno 1993). The LLS creates outputs' surfaces in Figs. 5 and 6 by Mamdani's rules and Takagi–Sugeno's rules, respectively. Maximum and minimum operations are considered for "OR" and "AND" operators in the LLS's inference engine to aggregation functions and reasoning. The center of gravity is used for defuzzification method.

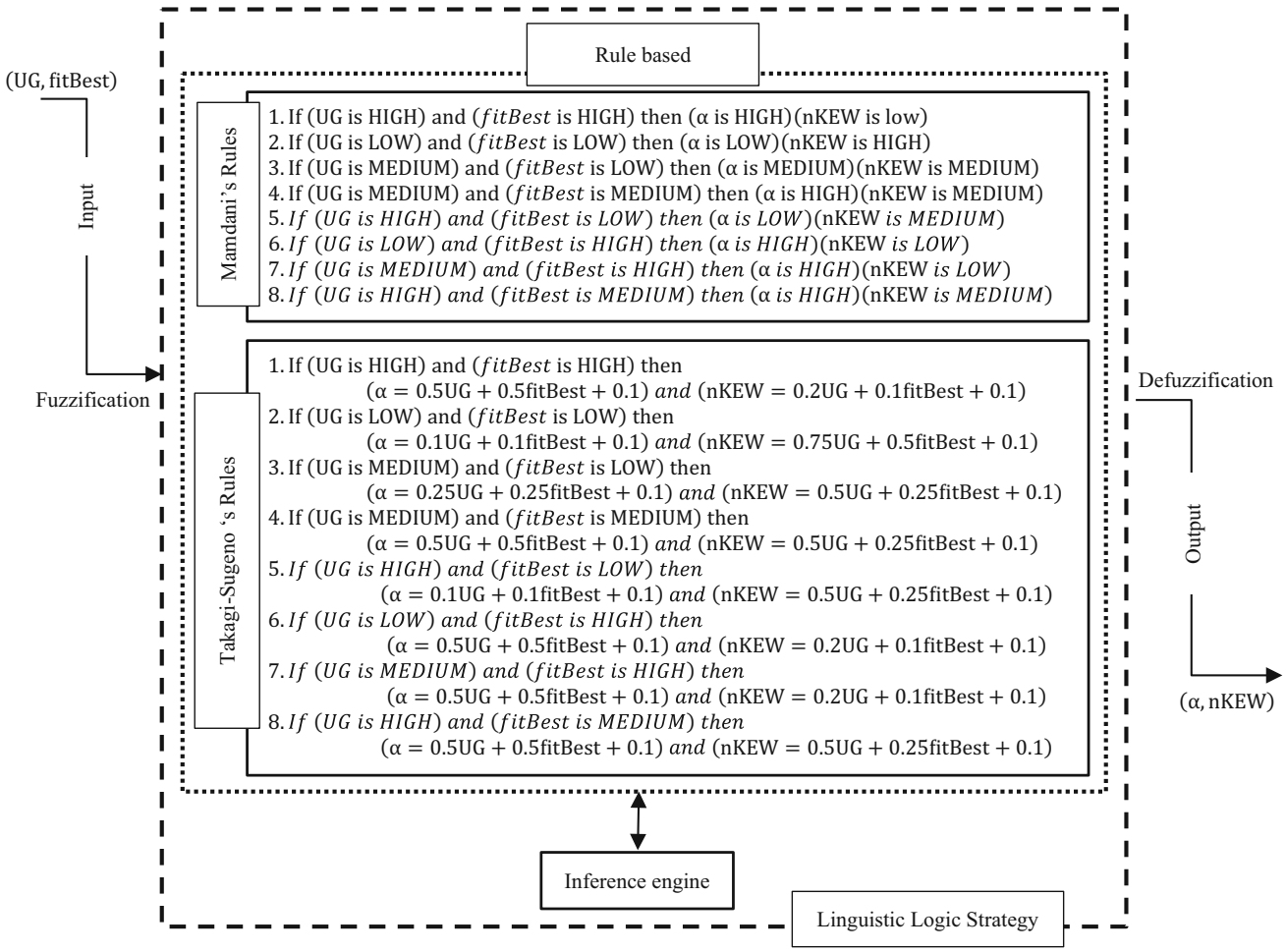



**Fig. 3** Proposed fuzzy logic controller based on two inputs (UG, fitBest) and two outputs ($\alpha$, $n$KEW)

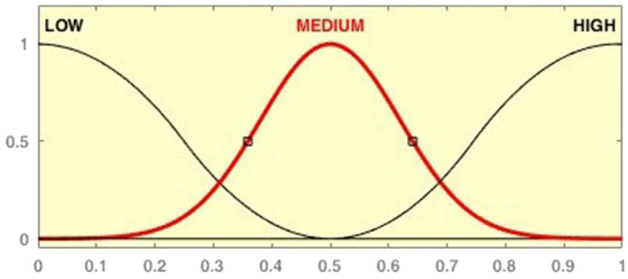



**Fig. 4** Membership functions of two inputs (UG, fitBest) based on linguistic values of low, medium and high

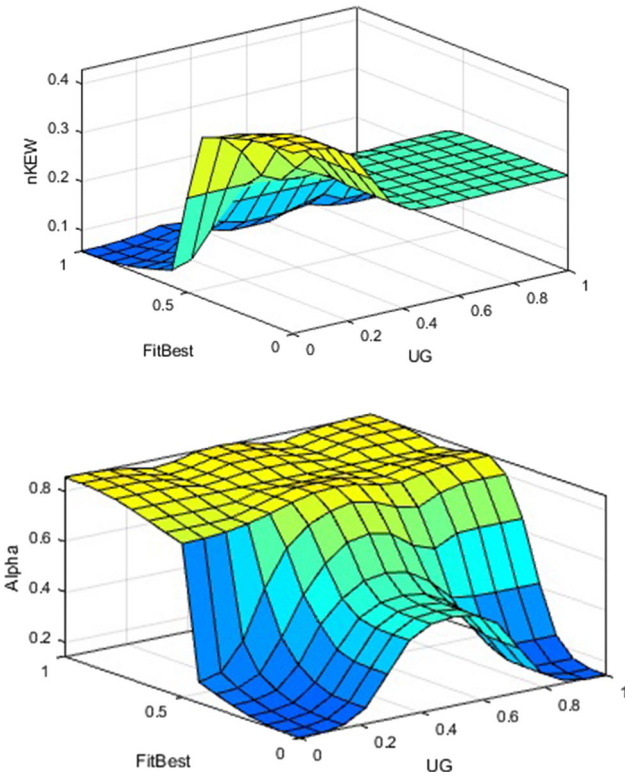



**Fig. 5** Outputs' surfaces of the LLS based on Mamdani's rules

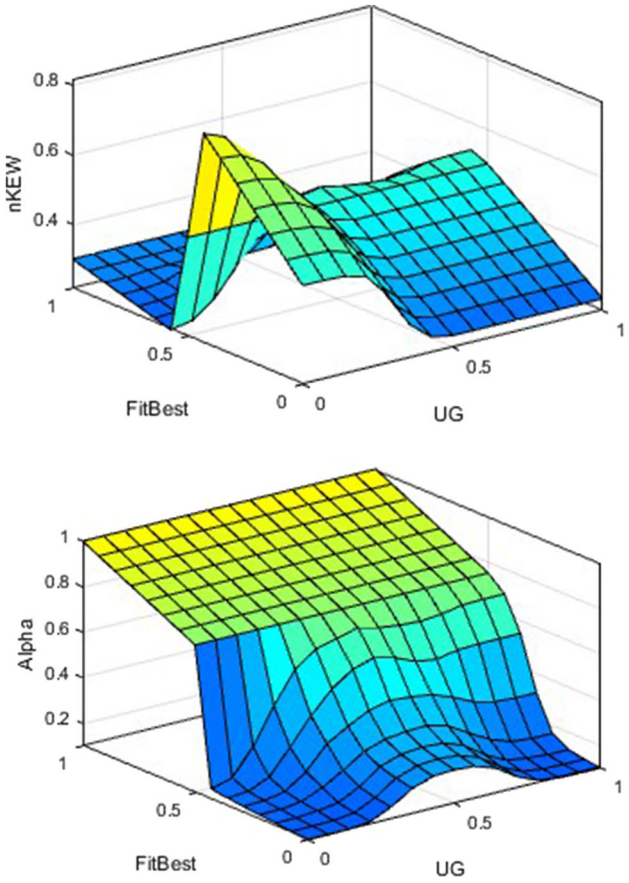



**Fig. 6** Outputs' surfaces of the LLS based on Takagi–Sugeno's rules

## 4 Evaluation of the improved DBSCAN algorithm and the soft improved DBSCAN

To evaluate the effectiveness of the proposed algorithms, experiments were performed on the Intel Core i5 3.2 GHz CPU and 4.00 GB memory. The algorithms were implemented in MATLAB 2017a. Six benchmarks datasets of the data cube, which are available from UCI and considered for experimentation, are shown in Table 1.

In order to achieve scalability of the proposed data cube clustering algorithms, each cell of the investigated 3D benchmarks such as $(i,j,k)$ is assigned by a given address like $r$ based on (1) and the addresses have inverse functions (2), (3) and (4) to calculate their $(i,j,k)$. The data are normalized based on (5), and their related 2D data are obtained by dimension move (Algorithm 1). To analyze DBSCAN2, which utilizes the proposed similarity metric (Algorithm 2) in the procedure of DBSCAN clustering (Algorithm 3) and the investigation evaluation indices, DBSCAN1 is considered based on the Euclidean metric in Algorithm 3 and the indices for its evaluation.

**Table 1** Investigated data cube

| ID | Dataset cube | Dimensions |
|---|---|---|
| 1 | Daily Demand Forecasting Orders | $8 \times 12 \times 5$ |
| 2 | Istanbul Stock Exchange | $20 \times 9 \times 26$ |
| 3 | Dow Jones Index | $330 \times 14 \times 2$ |
| 4 | ADL Recognition | $844 \times 3 \times 10$ |
| 5 | Software Engineering Teamwork | $63 \times 84 \times 5$ |
| 6 | User Identification From Walking Activity | $1144 \times 4 \times 6$ |

Eight drafts of the proposed clustering algorithms are preformed, namely DBSCAN1, DBSCAN2, EWOA–DBSCAN1, EWOA–DBSCAN2, EWOA–DBSCAN1-Mamdani, EWOA–DBSCAN2-Memdani, EWOA–DBSCAN1-Sugeno and EWOA–DBSCAN2-Sugeno.

There are eight parameters in the experimentation, namely $\mu$ and $\varepsilon$ in Algorithm 3, pop_size, $\alpha$, $n$KEW and Max$G$ in Algorithm 4 and pop_size and Max$G$ in Algorithm 5. Tuning parameters of Algorithm 3 are based on $\varepsilon = 0.5$ and $\mu$ is 10% of the investigated data, because other values increased the DBI and number of clusters simultaneously. Also, Algorithm 4 was tested on the data cube of "Daily Demand Forecasting Orders" by different values for $\alpha$ and $n$KEW; then, $\alpha = 0.8$ and $n\text{KEW} = [0.2 \times \text{pop\_size}]$, which had the best results, were considered for experimentations of all data cubes. Finally, pop_size and Max$G$ were tuned with 100 earthworms and 100 generations.

In order to compare the behavior of the EWOA in improving the designed DBSCAN with other metaheuristics, a genetic algorithm (GA) with the scattered crossover (Fig. 2), single point mutation, $P_c = 0.8$ (crossover rate), $P_m = 0.02$ (mutation rate), $\text{pop}_{\text{size}} = 100$ and MaxItr = 100 is performed and called GA-DBSCAN1. Also, $P_c$ and $P_m$ are dynamically tuned by the FLC (Fig. 3) in GA-DBSCAN1-Mamdani and GA-DBSCAN1-Sugeno.

The algorithms are run 20 times on each data cube, and then the best obtained DBI (6), DI (9) and SI (12) are reported as the best quality clustering of the data. The details of the obtained results from implementations are shown in "Appendix" and Tables 6, 7, 8, 9, 10, 11, 12 and 13. The best obtained DBI (6), DI (9) and SI (12) are summarized in Tables 2, 3 and 4, respectively. With regard to the tables, the performance of the EWOA–DBSCAN2-Sugeno (Algorithm 3 based on Takagi–Sugeno's rules) is significantly superior to that of the other performed algorithms.

Also, a comparison of the obtained results of DBSCAN1 and DBSCAN2 can show that except Dow Jones Index dataset for the DBI and except ADL Recognition dataset for Dunn and Silhouette indices, DBSCAN2 on the other datasets obtains between 3.6 and 25.5% better results in Table 2, between 1.6 and 33.6% better results in Table 3 and between 1.5 and 188.3% better results in Table 4. So, the newly proposed similarity metric is more efficient than Euclidean metric.

To compare the functionality of the proposed data cube clustering algorithms, the curve convergences of the best DBI, DI and SI are shown on the datasets in Figs. 7, 8, 9, 10, 11 and 12. The horizontal axis of the figures is measured based on the number of generations from 1 to 100, and the vertical axis is denoted by the best found DBI, DI and SI through improvement. Based on the figures, the EWOA has more control over the search space than the GA. Although the EWOA and the GA have almost the same evolutionary structure, the EWOA generates more offspring than the GA in a generation that the fact enhances the search process and increases the diversity of search points.

**Table 2** Best results (DBI) after 20 runs

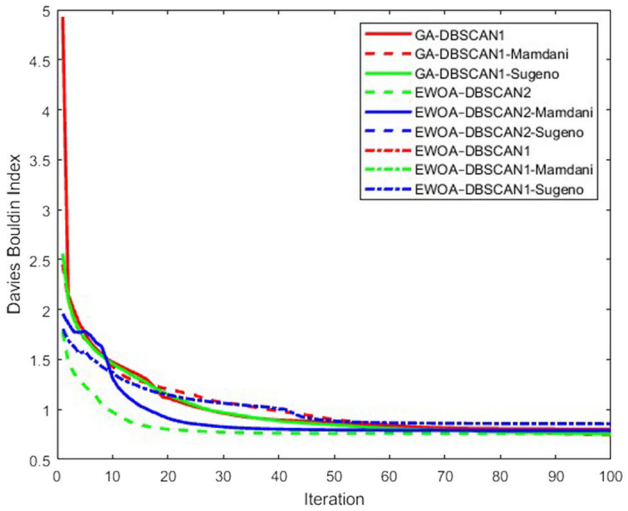
| Datasets | DBSCAN1 | GA-DBSCAN1 | GA-DBSCAN1-Mamdani | GA-DBSCAN1-Sugeno | EWOA – DBSCAN1 | EWOA–DBSCAN1-Mamdani | EWOA–DBSCAN1-Sugeno | DBSCAN2 | EWOA–DBSCAN2 | EWOA–DBSCAN2-Mamdani | EWOA–DBSCAN2-Sugeno |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1.0262 | 0.7777 | 0.7279 | 0.7160 | 0.8391 | 0.7567 | 0.74037 | 0.7644 | 0.7532 | 0.716 | **0.7157** |
| 2 | 0.9610 | 0.7970 | 0.7732 | **0.7676** | 0.8418 | 0.8093 | 0.80452 | 0.8578 | 0.8195 | 0.7719 | 0.7683 |
| 3 | 0.7155 | 0.5120 | 0.4890 | **0.4604** | 0.5626 | 0.6007 | 0.55985 | 0.8011 | 0.4889 | 0.4984 | **0.4604** |
| 4 | 0.8466 | 0.8010 | 0.7977 | 0.7654 | 0.8043 | 0.7994 | 0.7915 | 0.8011 | 0.8018 | 0.7677 | **0.7541** |
| 5 | 0.6418 | 0.5723 | 0.5549 | 0.5520 | 0.5887 | 0.5809 | 0.56895 | 0.6156 | 0.5533 | 0.552 | **0.5471** |
| 6 | 0.6117 | 0.5862 | 0.5245 | **0.5201** | 0.5726 | 0.5667 | 0.5539 | 0.5895 | 0.5815 | 0.5351 | 0.5211 |

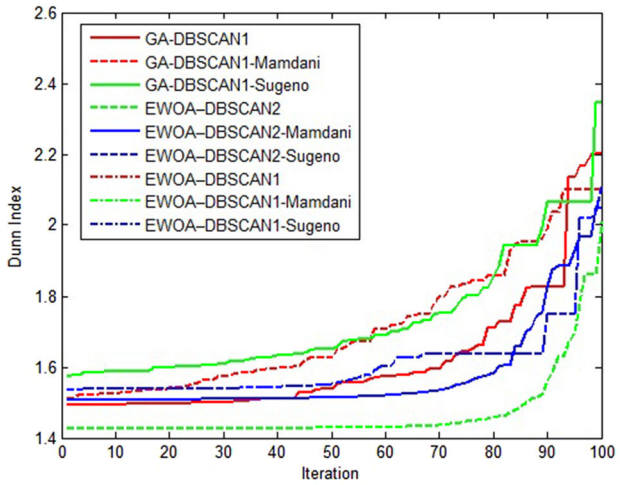
**Table 3** Best results (Dunn index) after 20 runs

| Datasets | DBSCAN1 | GA-DBSCAN1 | GA-DBSCAN1-Mamdani | GA-DBSCAN1-Sugeno | EWOA–DBSCAN1 | EWOA–DBSCAN1-Mamdani | EWOA–DBSCAN1-Sugeno | DBSCAN2 | EWOA–DBSCAN2 | EWOA–DBSCAN2-Mamdani | EWOA–DBSCAN2-Sugeno |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1.215 | 1.4921 | 1.5119 | 1.5749 | 1.4273 | 1.5080 | 1.5367 | 1.5202 | 1.54 | 1.6516 | **1.6855** |
| 2 | 1.1981 | 1.4444 | 1.6691 | 1.709 | 1.4505 | 1.5714 | 1.6576 | 1.6009 | 1.6515 | 1.691 | **1.7168** |
| 3 | 1.4677 | 1.6795 | 1.8908 | 1.9097 | 1.6856 | 1.7431 | 1.8647 | 1.659 | 1.9994 | 1.979 | **2.0339** |
| 4 | 1.6226 | 1.6955 | 1.7021 | 1.699 | 1.6723 | 1.6592 | 1.6424 | 1.5802 | 1.5883 | 1.7095 | **1.7318** |
| 5 | 1.7105 | 1.7829 | 1.8123 | 1.873 | 1.7888 | 1.8571 | 1.8467 | **1.9763** | 1.7259 | 1.9002 | 1.907 |
| 6 | 1.7781 | 1.8148 | 1.852 | 1.88 | 1.8243 | 1.8243 | 1.8404 | 1.8061 | 1.8236 | 1.915 | **1.9752** |

**Table 4** Best results (Silhouette index) after 20 runs

| Datasets | DBSCAN1 | GA-DBSCAN1 | GA-DBSCAN1-Mamdani | GA-DBSCAN1-Sugeno | EWOA–DBSCAN1 | EWOA–DBSCAN1-Mamdani | EWOA–DBSCAN1-Sugeno | DBSCAN2 | EWOA–DBSCAN2 | EWOA–DBSCAN2-Mamdani | EWOA–DBSCAN2-Sugeno |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.2412 | 0.2698 | 0.2398 | 0.2909 | 0.2673 | 0.2647 | 0.2771 | 0.2846 | 0.2932 | 0.3676 | **0.4012** |
| 2 | 0.1591 | 0.2414 | 0.4423 | 0.4766 | 0.2923 | 0.3808 | 0.4621 | 0.4587 | 0.471 | 0.4629 | **0.4791** |
| 3 | 0.1832 | 0.1915 | 0.3798 | 0.3701 | 0.2482 | 0.3438 | 0.4245 | 0.4601 | 0.4883 | 0.4774 | **0.4943** |
| 4 | 0.4692 | 0.4965 | 0.4998 | 0.4644 | 0.4767 | 0.4592 | 0.4339 | 0.3813 | 0.3901 | 0.4772 | **0.4859** |
| 5 | 0.3523 | 0.3552 | 0.3672 | 0.5520 | 0.4198 | 0.446 | 0.5220 | **0.6156** | 0.5533 | 0.4522 | 0.4541 |
| 6 | 0.3898 | 0.401 | 0.3765 | 0.4001 | 0.3969 | 0.3910 | 0.3943 | 0.3956 | 0.4051 | 0.4501 | **0.4963** |



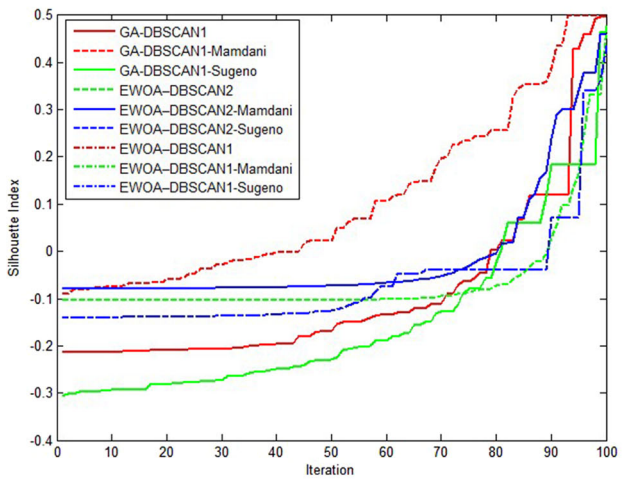
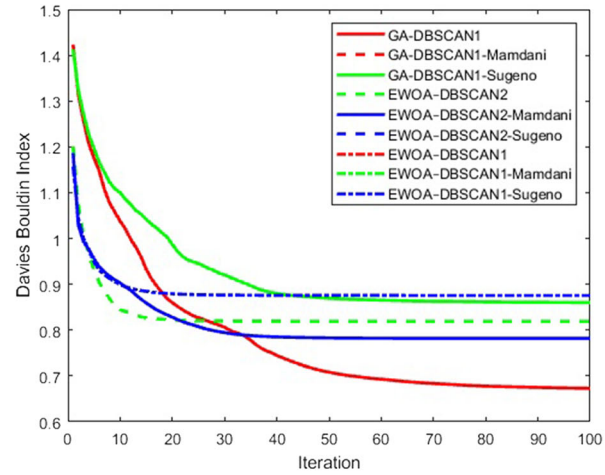
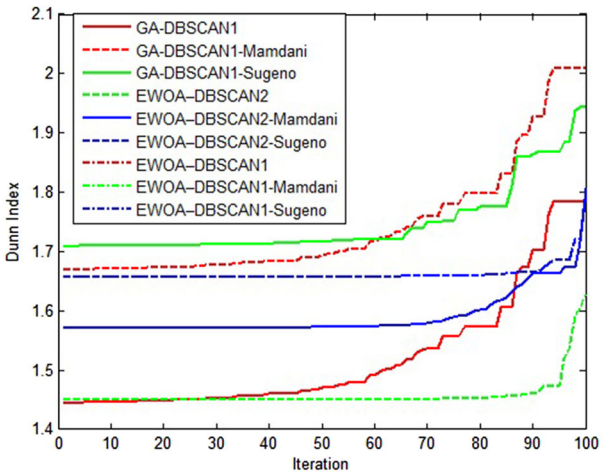




**Fig. 7** Convergence curves for the Daily Demand Forecasting Orders based on three investigation evaluation indices



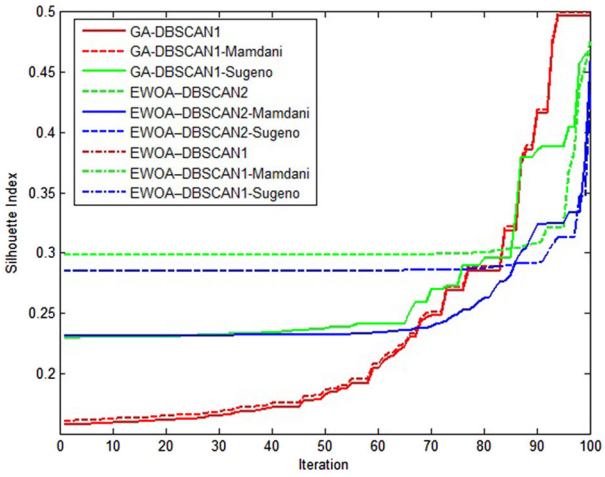
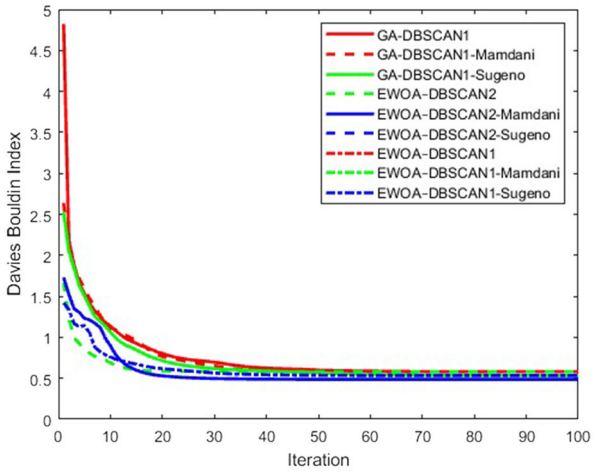
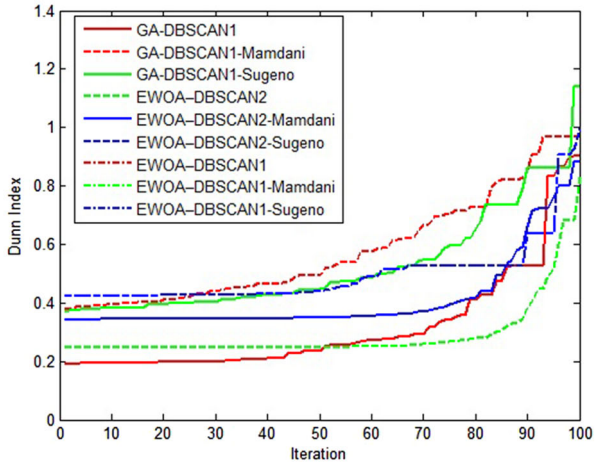




**Fig. 8** Convergence curves for the Istanbul Stock Exchange based on three investigation evaluation indices

To evaluate the significance level of the comparisons for the proposed data cube clustering algorithms, a hypothesis test is done to test the difference in the resulting quality between the algorithms. Because the obtained results of each algorithm are not normally distributed, a nonnormally distributed hypothesis test, such as the Wilcoxon signed-rank test, is utilized in SPSS between two samples at a significant



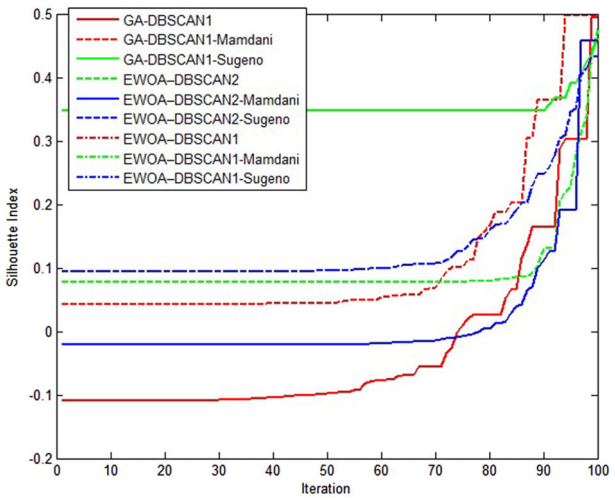
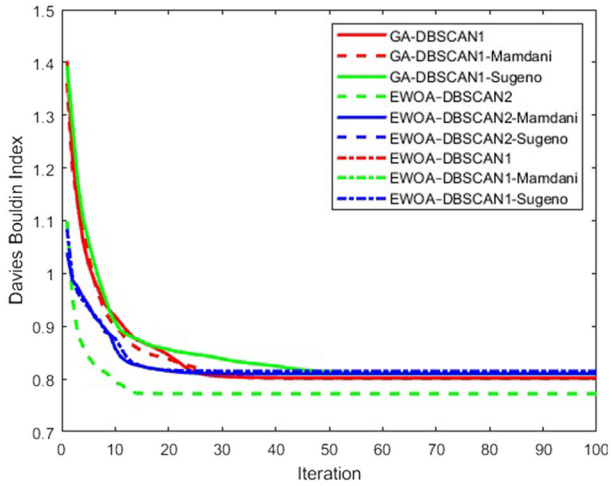
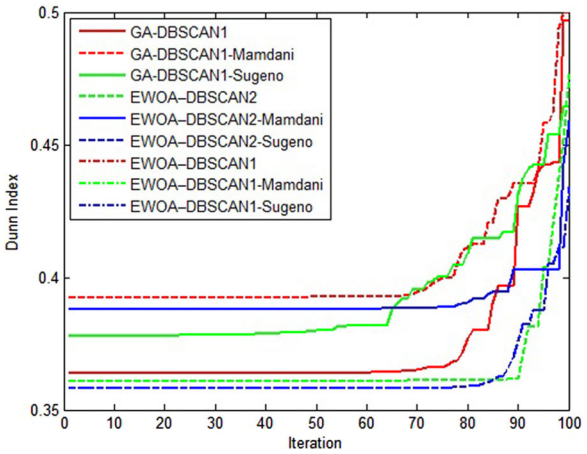




**Fig. 9** Convergence curves for the Dow Jones Index based on three investigation evaluation indices



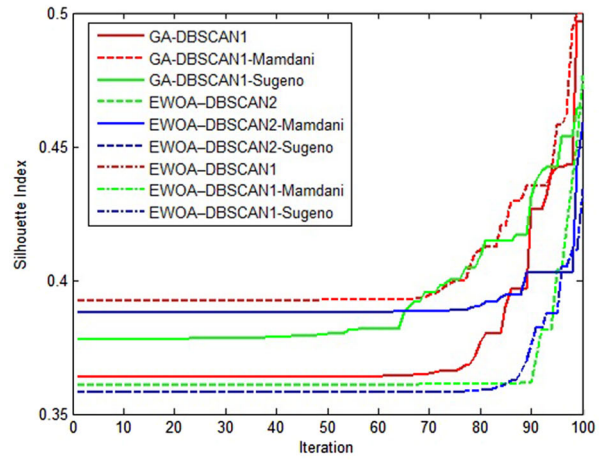




**Fig. 10** Convergence curves for the ADL Recognition based on three investigation evaluation indices

level of $\alpha = 0.05$. The results are presented in the form of [$Z$, $P$] in Table 5. If $P$ value $< 0.05$, then the null hypothesis (the two samples are dependent samples) can be rejected at the 95% level, but if $P$ value $> 0.05$, then the null hypothesis cannot be rejected. Therefore, the bold $P$ values present that the comparison of two mentioned clustering algorithms on the related datasets is significant at the 95% level.

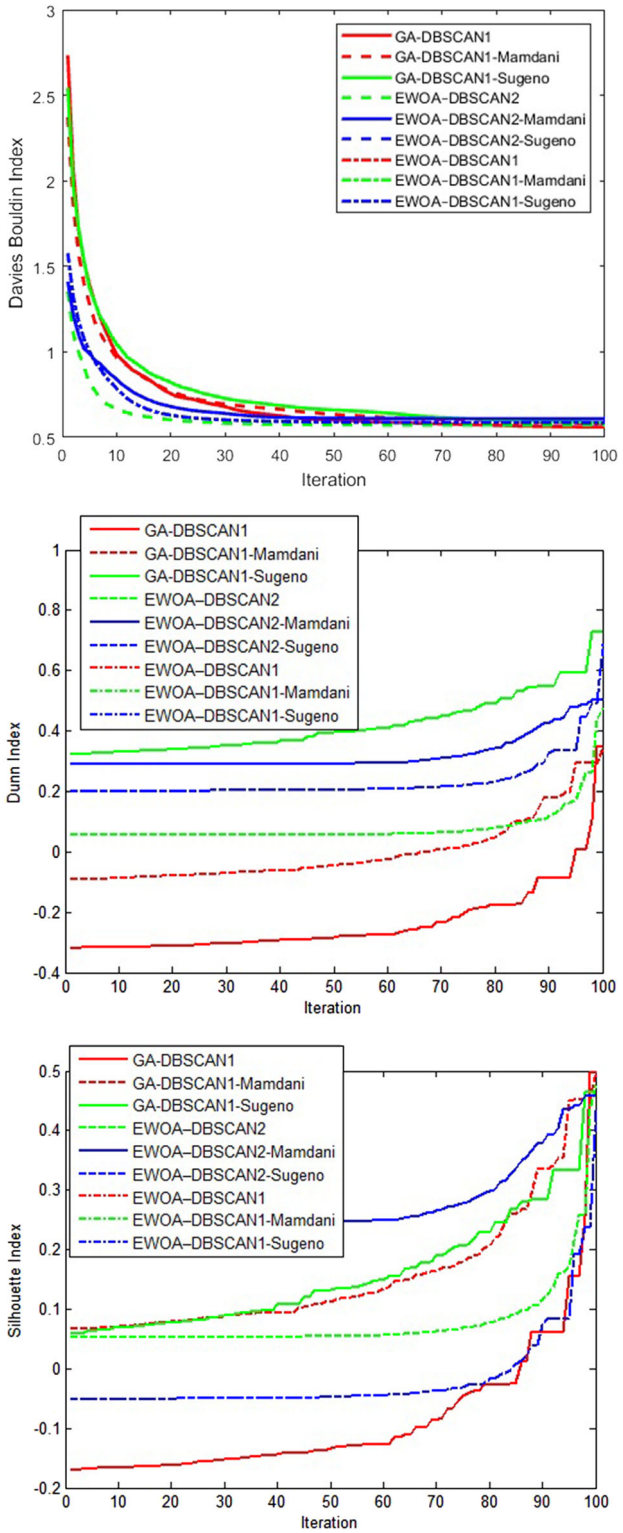



**Fig. 11** Convergence curves for the Software Engineering Teamwork based on three investigation evaluation indices

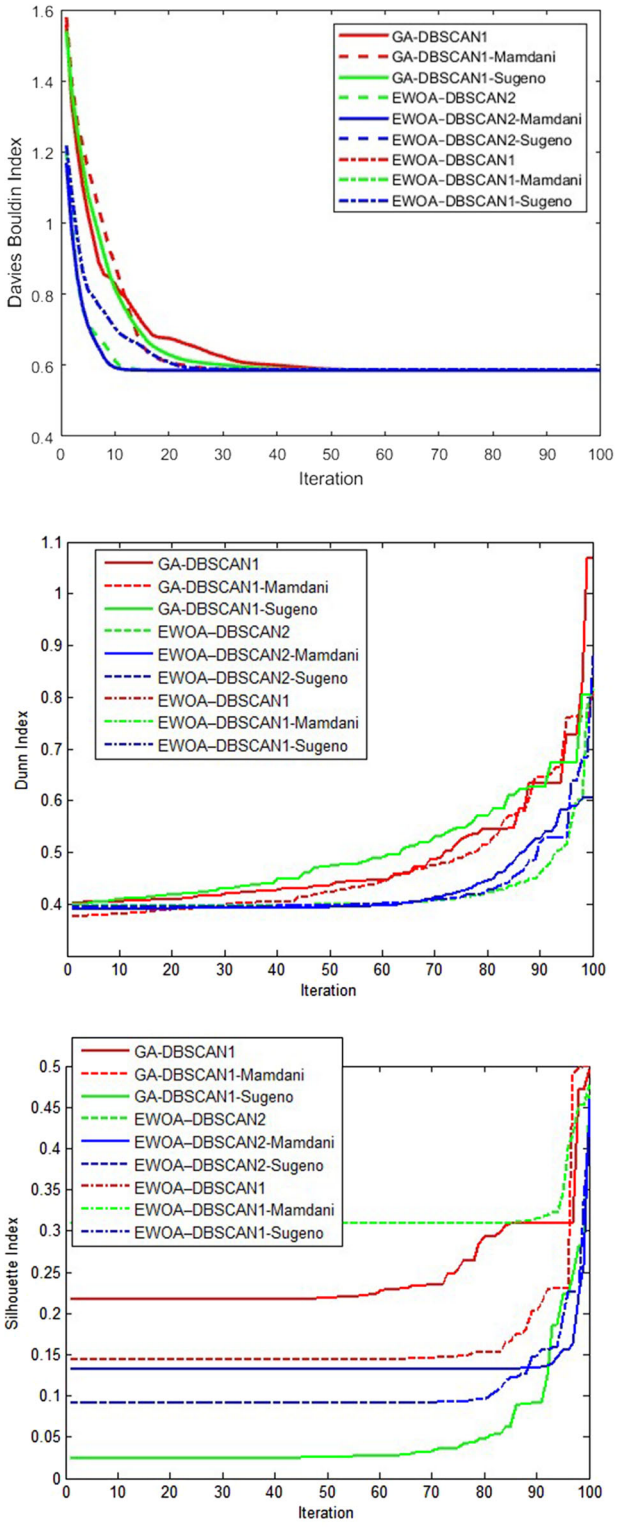



**Fig. 12** Convergence curves for the User Identification from Walking Activity based on three investigation evaluation indices

**Table 5** Results of the Wilcoxon signed-rank test in the form of [*Z*, *P*]

| | DBSCAN1 versus GA-DBSCAN1 | DBSCAN1 versus EWOA–DBSCAN1 | GA-DBSCAN1 versus EWOA–DBSCAN1 | GA-DBSCAN1 versus GA-DBSCAN1-Mamdani | EWOA–DBSCAN1 versus EWOA–DBSCAN1-Mamdani | GA-DBSCAN1-Mamdani versus EWOA–DBSCAN1-Mamdani |
|---|---|---|---|---|---|---|
| 1 | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 2.427, **0.015**] | [− 3.323, **0.001**] | [− 3.211, **0.001**] |
| 2 | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 0.672, 0.502] | [− 1.568, 0.117] | [− 0.411, 0.681] | [− 2.352, **0.019**] |
| 3 | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 3.621, **0.000**] | [− 1.456, 0.145] | [− 2.053, **0.040**] | [− 2.091, **0.037**] |
| 4 | [− 0.821, 0.411] | [− 2.427, **0.015**] | [− 2.501, **0.012**] | [− 1.232, 0.218] | [− 0.672, 0.502] | [− 2.203, **0.028**] |
| 5 | [− 3.584, **0.000**] | [− 3.920, **0.000**] | [− 3.584, **0.000**] | [− 1.904, 0.057] | [− 3.621, **0.000**] | [− 0.672, **0.020**] |
| 6 | [− 0.149, 0.881] | [− 0.821, 0.411] | [− 3.211, **0.001**] | [− 2.613, **0.009**] | [− 3.621, **0.000**] | [− 1.941, 0.052] |

| | GA-DBSCAN1 versus GA-DBSCAN1-Sugeno | EWOA–DBSCAN1 versus EWOA–DBSCAN1-Sugeno | GA-DBSCAN1-Sugeno versus EWOA–DBSCAN1-Sugeno | GA-DBSCAN1-Sugeno versus GA-DBSCAN1-Mamdani | EWOA–DBSCAN1-Sugeno versus EWOA–DBSCAN1-Mamdani | GA-DBSCAN1-Sugeno versus EWOA–DBSCAN1-Sugeno |
|---|---|---|---|---|---|---|
| 1 | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 3.211, **0.001**] | [− 2.763, **0.006**] |
| 2 | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 3.883, **0.000**] | [− 2.352, **0.019**] | [− 2.501, **0.012**] |
| 3 | [− 3.920, **0.000**] | [− 3.883, **0.000**] | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 2.091, **0.037**] | [− 1.680, 0.093] |
| 4 | [− 0.821, 0.411] | [− 2.949, **0.003**] | [− 3.173, **0.002**] | [− 2.501, **0.012**] | [− 2.203, **0.028**] | [− 2.165, **0.030**] |
| 5 | [− 3.584, **0.000**] | [− 3.285, **0.001**] | [− 3.696, **0.000**] | [− 3.584, **0.000**] | [− 0.672, **0.020**] | [− 0.933, 0.351] |
| 6 | [− 0.149, 0.881] | [− 3.323, **0.001**] | [− 3.696, **0.000**] | [− 3.211, **0.001**] | [− 1.941, 0.052] | [− 3.136, **0.002**] |

| | DBSCAN1 versus DBSCAN2 | EWOA–DBSCAN2 versus EWOA–DBSCAN2-Mamdani | DBSCAN2 versus EWOA–DBSCAN2 | DBSCAN2 versus EWOA–DBSCAN2-Mamdani | DBSCAN2 versus EWOA–DBSCAN2-Sugeno | EWOA–DBSCAN2-Mamdani versus EWOA–DBSCAN2-Sugeno |
|---|---|---|---|---|---|---|
| 1 | [− 3.323, **0.001**] | [− 2.613, **0.009**] | [− 1.829, 0.067] | [− 3.472, **0.001**] | [− 3.061, **0.002**] | [− 2.501, **0.012**] |
| 2 | [− 0.411, 0.681] | [− 2.315, **0.021**] | [− 2.912, **0.004**] | [− 3.509, **0.000**] | [− 3.733, **0.000**] | [− 2.576, **0.010**] |
| 3 | [− 2.053, **0.040**] | [− 0.597, **0.050**] | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 0.933, **0.351**] |
| 4 | [− 0.672, 0.502] | [− 2.016, **0.044**] | [− 0.131, 0.896] | [− 3.006, **0.003**] | [− 3.061, **0.002**] | [− 2.109, **0.035**] |
| 5 | [− 3.621, **0.000**] | [− 1.157, 0.247] | [− 2.053, **0.040**] | [− 3.024, **0.002**] | [− 3.323, **0.001**] | [− 1.200, 0.263] |
| 6 | [− 3.621, **0.000**] | [− 1.419, 0.156] | [− 1.195, 0.232] | [− 2.240, **0.025**] | [− 3.547, **0.000**] | [− 3.024, **0.002**] |

| | GA-DBSCAN1 versus EWOA–DBSCAN2 | EWOA–DBSCAN1 versus EWOA–DBSCAN2 | EWOA–DBSCAN1-Mamdani versus EWOA–DBSCAN2-Mamdani | EWOA–DBSCAN1-Sugeno versus EWOA–DBSCAN2-Sugeno | GA-DBSCAN1-Mamdani versus EWOA–DBSCAN2-Mamdani | GA-DBSCAN1-Sugeno versus EWOA–DBSCAN2-Sugeno |
|---|---|---|---|---|---|---|
| 1 | [− 0.885, 0.576] | [− 2.817, **0.005**] | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 3.547, **0.000**] | [− 1.064, 0.287] |
| 2 | [− 3.179, **0.001**] | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 3.883, **0.000**] | [− 3.883, **0.000**] | [− 1.381, 0.167] |
| 3 | [− 3.823, **0.000**] | [− 3.883, **0.000**] | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 3.920, **0.000**] | [− 3.920, **0.000**] |
| 4 | [− 0.483, 0.629] | [− 0.261, 0.794] | [− 3.969, **0.000**] | [− 2.501, **0.012**] | [− 3.435, **0.001**] | [− 1.531, 0.126] |
| 5 | [− 2.817, **0.005**] | [− 2.725, **0.006**] | [− 3.920, **0.000**] | [− 3.584, **0.000**] | [− 3.733, **0.000**] | [− 1.885, 0.059] |
| 6 | [− 0.302, 0.763] | [− 1.755, 0.079] | [− 3.435, **0.001**] | [− 3.211, **0.001**] | [− 2.800, **0.005**] | [− 0.784, 0.433] |

## 5 Conclusion and discussion

This paper focuses on the data cube clustering, and the DBSCAN algorithm is considered as the basic clustering technique, because it successfully recognizes nonconvex clusters and does not depend on a given number of clusters as an input. Since the efficiency of the DBSCAN Algorithm is highly dependent on the parameters of the neighboring radius and the number of neighbors, the study tries to improve this algorithm.

For this purpose, first new techniques were proposed in the preprocessing section. Assigning a unique address (1) to each cube cell for clarifying and interpreting the scalability of the proposed data cube clustering algorithms, obtaining the related 2D data from the 3D data by moving dimension (Algorithm 1) and designing similarity metric (Algorithm 2) are those techniques. The DBSCAN1 and the DBSCAN2 were performed based on embedding Euclidean and new similarity metrics in Algorithm 3, respectively.

The EWOA was adapted to improve DBSCAN (Algorithm 4) and find the optimum values for $P$, $\mu$ and $\varepsilon$ in Algorithm 3 (EWOA–DBSCAN1 and EWOA–DBSCAN2). To fill up the clustering algorithms' challenges and enhance the exploration and exploration capabilities of the algorithms, a new FLC (Fig. 3) was designed to dynamically tune the EWOA's parameters (Algorithm 5) and EWOA–DBSCAN1-Mamdani, EWOA–DBSCAN2-Mamdani, EWOA–DBSCAN1-Sugeno and EWOA–DBSCAN2-Sugeno were performed for analyzing the efficiency of the proposed ideas. The designed FLC has been carried out by Mamdani's rules and Takagi–Sugeno's rules.

For comparison of the obtained results, the GA was considered to embed in Algorithms 4 and 5 and GA-DBSCAN1, GA-DBSCAN1-Mamdani and GA-DBSCAN1–Sugeno were performed on the experimental results. Three investigation evaluation indices were considered to find the most similarities between members of the cluster, and on the other hand, they can provide separation issue which is related to less similarity with other clusters.

To evaluate and compare the proposed clustering algorithms, six datasets of data cube were considered and the details of the obtained results are reported in "Appendix". All experiments indicated the efficiency and improvement of the DBSCAN2 compared to the DBSCAN1 and the EWOA–DBSCAN2–Sugeno compared to the others. In addition, the ρ values of the Wilcoxon signed-rank test were calculated to recognize the significant level of the comparisons for the proposed algorithms. With regard to Table 5, comparisons of the most of the algorithms were significant at the 95% level for all datasets.

**Funding** The study is not funded by any agency.

## Compliance with ethical standards

**Conflict of interest** The authors do hereby declare that there is no conflict of interests of other works regarding the publication of this paper.

**Ethical approval** The manuscript does not contain any studies with human participants or animals performed by any of the authors.

## Appendix

See Tables 6, 7, 8, 9, 10, 11, 12 and 13.

**Table 6** Details of the experimental results for 20 runs of the DBSCAN1

| | Daily Demand Forecasting Orders | | Istanbul Stock Exchange | | Dow Jones Index | | ADL Recognition | | Software Engineering Teamwork | | User Identification From Walking Activity | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** |
| 1 | 1.0776 | 3 | 1.0528 | 3 | 0.7730 | 3 | 0.8672 | 3 | 0.7592 | 4 | 0.6292 | 4 |
| 2 | **1.0262** | **3** | 1.1977 | 4 | 0.8301 | 3 | **0.8466** | **3** | 0.6852 | 3 | 0.6853 | 4 |
| 3 | 1.0239 | 3 | 1.1589 | 4 | 0.8918 | 4 | 0.9471 | 4 | 0.6964 | 3 | **0.6117** | **3** |
| 4 | 1.2061 | 4 | 1.0006 | 3 | **0.7155** | **2** | 0.8639 | 3 | 0.7537 | 4 | 0.6703 | 4 |
| 5 | 1.1385 | 3 | **0.9610** | **3** | 0.8017 | 3 | 0.9257 | 3 | 0.7299 | 4 | 0.6326 | 3 |
| 6 | 1.1123 | 3 | 1.1252 | 3 | 0.6985 | 3 | 0.8863 | 3 | 0.7327 | 4 | 0.7943 | 4 |
| 7 | 1.1833 | 3 | 1.0124 | 3 | 0.7280 | 2 | 0.9833 | 4 | 0.7650 | 4 | 0.6788 | 3 |
| 8 | 1.2829 | 3 | 1.0595 | 3 | 0.8011 | 3 | 0.8868 | 4 | 0.7383 | 4 | 0.6874 | 3 |
| 9 | 1.2433 | 3 | 1.0501 | 3 | 0.8571 | 3 | 0.9303 | 4 | 0.6823 | 3 | 0.6575 | 3 |
| 10 | 1.2922 | 4 | 0.9971 | 2 | 0.8751 | 3 | 0.9115 | 4 | 0.6654 | 2 | 0.6269 | 3 |
| 11 | 1.0393 | 3 | 1.1844 | 4 | 0.7895 | 3 | 0.8720 | 3 | 0.7289 | 4 | 0.6975 | 3 |
| 12 | 1.0551 | 3 | 1.1529 | 4 | 0.8025 | 3 | 0.9771 | 4 | 0.7188 | 3 | 0.6596 | 3 |
| 13 | 1.1059 | 3 | 1.0482 | 3 | 0.8560 | 4 | 0.8625 | 4 | 0.7877 | 4 | 0.7537 | 4 |
| 14 | 1.2973 | 4 | 1.1419 | 4 | 0.8272 | 3 | 0.8771 | 3 | 0.8123 | 4 | 0.6260 | 3 |
| 15 | 1.2534 | 4 | 1.1095 | 3 | 0.8021 | 3 | 0.9849 | 4 | 0.8853 | 4 | 0.6971 | 3 |
| 16 | 1.1245 | 3 | 1.1616 | 4 | 0.8143 | 3 | 0.9947 | 4 | 0.7322 | 4 | 0.7277 | 4 |
| 17 | 1.2078 | 3 | 0.9770 | 3 | 0.7337 | 3 | 0.9437 | 4 | 0.7881 | 4 | 0.6536 | 3 |
| 18 | 1.0307 | 4 | 1.0180 | 2 | 0.8166 | 3 | 0.8598 | 3 | 0.8615 | 4 | 0.6998 | 3 |
| 19 | 1.2267 | 3 | 1.0835 | 2 | 07956 | 3 | 0.9644 | 4 | 0.8718 | 4 | 0.6473 | 3 |
| 20 | 1.2367 | 4 | 1.1481 | 4 | 0.8185 | 3 | 0.9735 | 4 | **0.6418** | **2** | 0.7238 | 4 |

The best obtained clustering is bolded for each dataset

*Denotes Davies Bouldin index (DBI)

**Denotes the number of clusters

**Table 7** Details of the experimental results for 20 runs of the GA-DBSCAN1

| | Daily Demand Forecasting Orders | | Istanbul Stock Exchange | | Dow Jones Index | | ADL Recognition | | Software Engineering Teamwork | | User Identification From Walking Activity | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** |
| 1 | 0.9499 | 3 | 0.8919 | 2 | 0.7152 | 3 | 0.9596 | 3 | 0.6076 | 3 | 0.7239 | 4 |
| 2 | 0.7897 | 3 | **0.7970** | **2** | 0.7341 | 3 | 0.9136 | 3 | 0.6253 | 3 | 0.7159 | 4 |
| 3 | 0.9975 | 3 | 0.8528 | 3 | 0.6781 | 2 | 0.9219 | 3 | 0.7481 | 4 | 0.6311 | 3 |
| 4 | 0.882 | 2 | 0.9419 | 3 | **0.5120** | **2** | **0.8010** | **2** | 0.7453 | 4 | 0.7250 | 4 |
| 5 | 0.939 | 3 | 0.8240 | 3 | 0.5208 | 2 | 0.9352 | 3 | 0.6612 | 3 | 0.6356 | 3 |
| 6 | 0.8475 | 2 | 0.8667 | 2 | 0.6642 | 3 | 0.9466 | 3 | 0.6771 | 3 | 0.6095 | 2 |
| 7 | 0.7816 | 2 | 0.9029 | 4 | 0.5891 | 2 | 0.8730 | 2 | 0.7024 | 3 | 0.7004 | 3 |
| 8 | 0.9584 | 3 | 0.8641 | 3 | 0.5676 | 2 | 0.9792 | 4 | **0.5723** | **2** | 0.6202 | 3 |
| 9 | 0.8214 | 3 | 0.9380 | 4 | 0.5621 | 2 | 0.9944 | 4 | 0.7415 | 4 | **0.5862** | **2** |
| 10 | **0.7777** | **2** | 0.9714 | 4 | 0.5650 | 2 | 0.8032 | 2 | 0.5731 | 2 | 0.6473 | 3 |
| 11 | 0.7782 | 2 | 0.9824 | 4 | 0.5972 | 2 | 0.9095 | 3 | 0.7305 | 4 | 0.7033 | 3 |
| 12 | 0.7644 | 2 | 0.9737 | 4 | 0.6207 | 3 | 0.9421 | 3 | 0.6156 | 3 | 0.6557 | 2 |
| 13 | 0.8651 | 3 | 0.9534 | 3 | 0.6947 | 3 | 0.9859 | 4 | 0.5877 | 2 | 0.6351 | 2 |
| 14 | 0.9289 | 4 | 0.9487 | 3 | 0.6214 | 3 | 0.8078 | 2 | 0.7318 | 4 | 0.6972 | 3 |
| 15 | 0.9624 | 4 | 0.9400 | 3 | 0.6456 | 3 | 0.9831 | 4 | 0.7321 | 4 | 0.6411 | 3 |
| 16 | 0.8532 | 3 | 0.9185 | 3 | 0.7296 | 3 | 0.9160 | 3 | 0.6185 | 3 | 0.7415 | 4 |
| 17 | 0.8899 | 3 | 0.9349 | 3 | 0.5856 | 2 | 0.8289 | 2 | 0.7048 | 3 | 0.6050 | 2 |
| 18 | 0.9404 | 4 | 0.8578 | 2 | 0.7083 | 3 | 0.8625 | 2 | 0.6431 | 3 | 0.7358 | 3 |
| 19 | 0.8586 | 3 | 0.9067 | 3 | 0.5237 | 2 | 0.8503 | 2 | 0.6946 | 3 | 0.7372 | 3 |
| 20 | 0.9448 | 3 | 0.9851 | 3 | 0.6428 | 3 | 0.8104 | 2 | 0.5727 | 2 | 0.5885 | 2 |

The best obtained clustering is bolded for each dataset

*Denotes Davies Bouldin index (DBI)

**Denotes the number of clusters

**Table 8** Details of the experimental results for 20 runs of the GA-DBSCAN1-Mamdani

| | Daily Demand Forecasting Orders | | Istanbul Stock Exchange | | Dow Jones Index | | ADL Recognition | | Software Engineering Teamwork | | User Identification From Walking Activity | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** |
| 1 | 0.8995 | 3 | 0.8222 | 3 | 0.5066 | 2 | 0.8325 | 4 | 0.5567 | 2 | 0.6325 | 4 |
| 2 | 0.8448 | 3 | 0.8765 | 3 | 0.5427 | 2 | 0.8880 | 4 | 0.6112 | 3 | 0.5418 | 3 |
| 3 | 0.8040 | 3 | 0.7826 | 2 | 0.5624 | 3 | 0.8290 | 4 | 0.6827 | 3 | 0.5394 | 3 |
| 4 | 0.7542 | 2 | 0.9496 | 4 | 0.6782 | 4 | 0.9114 | 5 | 0.7250 | 4 | 0.7082 | 5 |
| 5 | 0.7815 | 2 | 0.8165 | 3 | 0.6407 | 4 | 0.8643 | 4 | 0.6940 | 3 | 0.6238 | 4 |
| 6 | 0.8198 | 3 | 0.8866 | 3 | 0.5441 | 3 | 0.8417 | 4 | 0.6981 | 3 | 0.7167 | 5 |
| 7 | 0.8022 | 3 | **0.7732** | **2** | 0.5161 | 2 | 0.8375 | 5 | 0.7020 | 4 | 0.6651 | 4 |
| 8 | **0.7279** | **2** | 0.7858 | 2 | 0.6492 | 3 | 0.8753 | 5 | 0.5671 | 2 | 0.6639 | 4 |
| 9 | 0.8153 | 2 | 0.8645 | 4 | 0.6588 | 3 | 0.9587 | 5 | 0.7068 | 4 | 0.5820 | 3 |
| 10 | 0.8153 | 2 | 0.7905 | 2 | 0.5447 | 2 | 0.8721 | 5 | 0.5581 | 2 | 0.7093 | 5 |
| 11 | 0.8527 | 3 | 0.8845 | 4 | 0.6578 | 3 | 0.9902 | 5 | 0.6231 | 3 | 0.6038 | 4 |
| 12 | 0.8531 | 3 | 0.8959 | 4 | 0.5796 | 3 | 0.8719 | 4 | 0.6852 | 3 | 0.5439 | 3 |
| 13 | 0.8667 | 3 | 0.8369 | 3 | 0.6764 | 4 | 0.7838 | 3 | 0.6686 | 3 | 0.5324 | 3 |
| 14 | 0.8470 | 3 | 0.9615 | 4 | 0.6533 | 3 | 0.8121 | 3 | 0.7097 | 4 | 0.6017 | 4 |
| 15 | 0.8212 | 3 | 0.9328 | 4 | 0.5449 | 3 | 0.8971 | 3 | 0.5705 | 2 | 0.6414 | 4 |
| 16 | 0.7505 | 2 | 0.9170 | 4 | 0.5331 | 2 | 0.9877 | 4 | 0.5697 | 2 | 0.5267 | 3 |
| 17 | 0.8046 | 2 | 0.9818 | 4 | 0.5683 | 3 | 0.9106 | 5 | **0.5549** | **2** | 0.5832 | 4 |
| 18 | 0.8125 | 3 | 0.9657 | 4 | 0.6576 | 4 | **0.7677** | **3** | 0.5890 | 2 | 0.5924 | 4 |
| 19 | 0.8482 | 3 | 0.9745 | 4 | 0.5402 | 2 | 0.8989 | 4 | 0.5721 | 2 | **0.5245** | **3** |
| 20 | 0.8187 | 3 | 0.9277 | 4 | **0.4890** | **2** | 0.8926 | 4 | 0.5963 | 2 | 0.5903 | 4 |

The best obtained clustering is bolded for each dataset

*Denotes Davies Bouldin index (DBI)

**Denotes the number of clusters

**Table 9** Details of the experimental results for 20 runs of the GA-DBSCAN1-Sugeno

| | Daily Demand Forecasting Orders | | Istanbul Stock Exchange | | Dow Jones Index | | ADL Recognition | | Software Engineering Teamwork | | User Identification From Walking Activity | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** |
| 1 | 0.7393 | 2 | 0.8842 | 4 | 0.6156 | 3 | 0.7867 | 3 | 0.5733 | 2 | **0.5201** | **3** |
| 2 | 0.8141 | 3 | 0.7750 | 2 | 0.6806 | 4 | 0.7685 | 3 | 0.5900 | 3 | 0.6630 | 4 |
| 3 | 0.8824 | 4 | 0.8388 | 3 | 0.6152 | 3 | 0.8461 | 4 | 0.6837 | 4 | 0.5749 | 3 |
| 4 | 0.7680 | 2 | 0.8512 | 4 | 0.5044 | 2 | **0.7654** | **3** | 0.5888 | 2 | 0.6757 | 4 |
| 5 | 0.8131 | 3 | 0.8842 | 4 | 0.6641 | 3 | 0.8312 | 4 | 0.6072 | 3 | 0.6523 | 4 |
| 6 | 0.8622 | 3 | 0.8675 | 3 | 0.6948 | 4 | 0.8991 | 5 | 0.5977 | 2 | 0.5328 | 3 |
| 7 | 0.7720 | 3 | 0.8171 | 3 | 0.5382 | 2 | 0.8345 | 4 | 0.6150 | 3 | 0.6408 | 4 |
| 8 | 0.7187 | 2 | 0.8797 | 4 | 0.5214 | 2 | 0.8169 | 3 | 0.5708 | 2 | 0.5349 | 3 |
| 9 | 0.8531 | 3 | 0.8329 | 3 | 0.5966 | 3 | 0.8389 | 4 | 0.6670 | 4 | 0.5393 | 3 |
| 10 | 0.7684 | 2 | 0.7851 | 2 | 0.6360 | 3 | 0.7754 | 3 | 0.5828 | 2 | 0.6139 | 3 |
| 11 | 0.8867 | 4 | 0.8286 | 3 | 0.6112 | 3 | 0.8775 | 4 | 0.6691 | 3 | 0.6523 | 4 |
| 12 | 0.7171 | 2 | 0.7829 | 2 | 0.4670 | 2 | 0.7996 | 3 | 0.6782 | 3 | 0.5726 | 3 |
| 13 | 0.8028 | 3 | 0.8057 | 2 | 0.5313 | 3 | 0.7883 | 3 | 0.5528 | 2 | 0.6973 | 4 |
| 14 | 0.7413 | 2 | 0.8212 | 2 | 0.4894 | 2 | 0.8289 | 3 | 0.6738 | 4 | 0.6251 | 4 |
| 15 | 0.7686 | 2 | 0.7769 | 2 | 0.4948 | 2 | 0.7966 | 3 | 0.5940 | 2 | 0.6888 | 4 |
| 16 | 0.8150 | 3 | 0.8193 | 2 | 0.6360 | 3 | 0.8064 | 3 | 0.5683 | 2 | 0.5645 | 3 |
| 17 | 0.7781 | 2 | 0.8689 | 3 | 0.5709 | 2 | 0.8324 | 3 | 0.6140 | 3 | 0.6724 | 4 |
| 18 | **0.7160** | **2** | 0.8914 | 4 | 0.5883 | 3 | 0.8789 | 4 | 0.5757 | 2 | 0.6024 | 3 |
| 19 | 0.8123 | 3 | **0.7676** | **2** | 0.5297 | 3 | 0.8763 | 4 | 0.5602 | 2 | 0.5331 | 3 |
| 20 | 0.7502 | 2 | 0.8087 | 2 | **0.4604** | **2** | 0.7731 | 3 | **0.5520** | **2** | 0.5355 | 3 |

The best obtained clustering is bolded for each dataset

*Denotes Davies Bouldin index (DBI)

**Denotes the number of clusters

**Table 10** Details of the experimental results for 20 runs of the DBSCAN2

| | Daily Demand Forecasting Orders | | Istanbul Stock Exchange | | Dow Jones Index | | ADL Recognition | | Software Engineering Teamwork | | User Identification From Walking Activity | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** |
| 1 | 0.8869 | 3 | **0.8578** | **2** | 0.9734 | 4 | 0.9592 | 4 | 0.6753 | 3 | 0.7259 | 4 |
| 2 | 0.8455 | 2 | 0.8644 | 3 | **0.8011** | **3** | 0.9660 | 3 | 0.7248 | 3 | 0.6928 | 3 |
| 3 | 1.2051 | 4 | 0.9826 | 4 | 0.8135 | 3 | 0.8803 | 2 | 0.7988 | 3 | 0.6856 | 3 |
| 4 | 1.0561 | 3 | 0.9814 | 4 | 0.8166 | 3 | 0.9636 | 3 | 0.7605 | 4 | 0.6375 | 3 |
| 5 | 0.7827 | 3 | 1.1416 | 4 | 0.8687 | 2 | 0.8663 | 3 | 0.7405 | 4 | 0.7604 | 3 |
| 6 | 0.8386 | 3 | 0.9629 | 4 | 0.9384 | 4 | 0.8463 | 3 | 0.7422 | 4 | 0.6502 | 3 |
| 7 | 0.8186 | 3 | 1.0524 | 3 | 0.9822 | 4 | 0.9172 | 3 | 0.7481 | 4 | 0.6273 | 3 |
| 8 | 1.1933 | 3 | 0.9280 | 4 | 0.8021 | 3 | 0.8878 | 2 | 0.7337 | 4 | 0.6423 | 3 |
| 9 | 0.8986 | 3 | 1.1377 | 4 | 0.8631 | 3 | 0.8672 | 3 | 0.6764 | 3 | 0.6703 | 4 |
| 10 | 0.8656 | 3 | 1.1729 | 4 | 0.9049 | 4 | 0.9659 | 4 | 0.6753 | 3 | 0.6239 | 3 |
| 11 | 1.2457 | 4 | 1.1577 | 4 | 0.7265 | 2 | 0.9315 | 4 | **0.6156** | **3** | 0.6052 | 2 |
| 12 | 0.8631 | 3 | 1.4506 | 3 | 0.8077 | 3 | 0.8695 | 2 | 0.6436 | 3 | 0.6718 | 3 |
| 13 | 1.1133 | 3 | 0.9634 | 3 | 0.8175 | 3 | 0.8781 | 3 | 0.7329 | 4 | 0.6872 | 3 |
| 14 | 1.2878 | 3 | 0.8541 | 3 | 0.9077 | 3 | 0.8828 | 4 | 0.7484 | 4 | 0.7159 | 4 |
| 15 | 0.8799 | 3 | 1.4182 | 3 | 0.8262 | 3 | 0.8753 | 3 | 0.8716 | 4 | **0.5895** | **2** |
| 16 | **0.7644** | **2** | 0.9167 | 3 | 0.9459 | 3 | 0.9356 | 3 | 0.7532 | 4 | 0.6706 | 4 |
| 17 | 0.8786 | 3 | 1.1877 | 4 | 0.8551 | 3 | 0.9645 | 4 | 0.8423 | 4 | 0.6898 | 3 |
| 18 | 0.7616 | 2 | 1.1180 | 2 | 0.9884 | 4 | 0.9843 | 4 | 0.7665 | 4 | 0.6426 | 3 |
| 19 | 1.0062 | 3 | 1.1716 | 4 | 0.7845 | 3 | 0.8079 | 2 | **0.6156** | **3** | 0.7472 | 3 |
| 20 | 0.9089 | 4 | 0.9124 | 4 | 0.8969 | 2 | **0.801** | **2** | 0.7777 | 4 | 0.6492 | 4 |

The best obtained clustering is bolded for each dataset

*Denotes Davies Bouldin index (DBI)

**Denotes the number of clusters

**Table 11** Details of the experimental results for 20 runs of the EWOA–DBSCAN2

| | Daily Demand Forecasting Orders | | Istanbul Stock Exchange | | Dow Jones Index | | ADL Recognition | | Software Engineering Teamwork | | User Identification From Walking Activity | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** |
| 1 | 0.9484 | 3 | 0.8642 | 3 | 0.5636 | 2 | 0.9492 | 4 | 0.5725 | 2 | 0.6222 | 3 |
| 2 | 0.8446 | 2 | 0.9819 | 4 | 0.5623 | 3 | 0.9196 | 5 | **0.5533** | **2** | 0.5872 | 4 |
| 3 | 0.8387 | 3 | 0.9297 | 4 | **0.4889** | **2** | 0.8976 | 4 | 0.5965 | 2 | 0.5903 | 4 |
| 4 | 0.9804 | 4 | 0.8568 | 2 | 0.7183 | 3 | 0.8515 | 2 | 0.6631 | 3 | 0.7358 | 3 |
| 5 | 0.9039 | 3 | 0.8345 | 3 | 0.5108 | 2 | 0.9556 | 3 | 0.6212 | 3 | 0.6376 | 3 |
| 6 | 0.7082 | 2 | 0.9724 | 4 | 0.5872 | 2 | 0.9015 | 3 | 0.7805 | 4 | 0.7003 | 3 |
| 7 | 0.8199 | 3 | 0.9149 | 3 | 0.5816 | 2 | 0.8239 | 2 | 0.7948 | 3 | 0.6085 | 2 |
| 8 | **0.7532** | **2** | 0.9696 | 4 | 0.6382 | 4 | 0.9214 | 5 | 0.7825 | 4 | 0.7082 | 5 |
| 9 | 0.7882 | 2 | 0.9024 | 4 | 0.5972 | 2 | 0.9091 | 3 | 0.7309 | 4 | 0.7033 | 3 |
| 10 | 0.7915 | 2 | **0.8195** | **3** | 0.6617 | 4 | 0.8603 | 4 | 0.6194 | 3 | 0.6238 | 4 |
| 11 | 0.8653 | 2 | 0.8915 | 4 | 0.6898 | 3 | 0.9589 | 5 | 0.7063 | 4 | **0.5815** | **3** |
| 12 | 0.8714 | 3 | 0.9518 | 4 | 0.5631 | 2 | 0.9942 | 4 | 0.7412 | 4 | 0.5862 | 2 |
| 13 | 0.9939 | 3 | 0.8538 | 3 | 0.5218 | 2 | 0.9353 | 3 | 0.6612 | 3 | 0.6456 | 3 |
| 14 | 0.7944 | 2 | 0.9747 | 4 | 0.6607 | 3 | 0.9426 | 3 | 0.6156 | 3 | 0.6557 | 2 |
| 15 | 0.8691 | 3 | 0.9954 | 3 | 0.6977 | 3 | 0.9819 | 4 | 0.5877 | 2 | 0.6371 | 2 |
| 16 | 0.8819 | 3 | 0.9249 | 3 | 0.5836 | 2 | 0.8281 | 2 | 0.7048 | 3 | 0.6905 | 2 |
| 17 | 0.8945 | 3 | 0.8542 | 3 | 0.5966 | 2 | 0.8327 | 4 | 0.5567 | 2 | 0.6325 | 4 |
| 18 | 0.8922 | 2 | 0.9414 | 3 | 0.5112 | 2 | **0.8018** | **2** | 0.7453 | 4 | 0.7695 | 4 |
| 19 | 0.7702 | 2 | 0.9825 | 4 | 0.5902 | 2 | 0.9995 | 3 | 0.7305 | 4 | 0.7083 | 3 |
| 20 | 0.8607 | 3 | 0.8569 | 3 | 0.6714 | 4 | 0.7438 | 3 | 0.6686 | 3 | 0.5727 | 3 |

The best obtained clustering is bolded for each dataset

*Denotes Davies Bouldin index (DBI)

**Denotes the number of clusters

**Table 12** Details of the experimental results for 20 runs of the EWOA–DBSCAN2-Mamdani

| | Daily Demand Forecasting Orders | | Istanbul Stock Exchange | | Dow Jones Index | | ADL Recognition | | Software Engineering Teamwork | | User Identification From Walking Activity | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** |
| 1 | 0.8247 | 3 | 0.9185 | 4 | 0.6093 | 3 | 0.8211 | 3 | 0.7197 | 4 | 0.6047 | 4 |
| 2 | 0.884 | 4 | 0.8428 | 3 | 0.6122 | 3 | 0.8641 | 4 | 0.6737 | 4 | 0.5769 | 3 |
| 3 | 0.8415 | 3 | 0.8913 | 2 | 0.636 | 3 | 0.8024 | 3 | 0.6083 | 2 | 0.5745 | 3 |
| 4 | 0.7545 | 2 | 0.9137 | 4 | 0.5331 | 2 | 0.9827 | 4 | 0.6097 | 2 | 0.5867 | 3 |
| 5 | 0.8022 | 3 | 0.7842 | 2 | 0.5611 | 2 | 0.8275 | 5 | 0.7102 | 4 | 0.6641 | 4 |
| 6 | 0.8125 | 3 | 0.9567 | 4 | 0.6567 | 4 | **0.7677** | **3** | 0.8589 | 2 | 0.6924 | 4 |
| 7 | 0.8242 | 3 | 0.9238 | 4 | 0.5494 | 3 | 0.8941 | 3 | 0.7705 | 2 | 0.6814 | 4 |
| 8 | 0.8187 | 3 | 0.9831 | 4 | 0.498 | 2 | 0.8966 | 4 | 0.6963 | 2 | 0.6903 | 4 |
| 9 | 0.7792 | 2 | 0.7588 | 2 | 0.6432 | 3 | 0.8123 | 5 | 0.6671 | 2 | 0.6939 | 4 |
| 10 | 0.755 | 2 | 0.917 | 4 | 0.5471 | 2 | 0.9817 | 4 | 0.6597 | 2 | 0.5967 | 3 |
| 11 | 0.7343 | 2 | 0.848 | 4 | 0.6165 | 3 | 0.7897 | 3 | 0.5973 | 2 | 0.5901 | 3 |
| 12 | 0.8247 | 3 | 0.8545 | 4 | 0.6578 | 3 | 0.9102 | 5 | 0.6321 | 3 | 0.6308 | 4 |
| 13 | 0.7866 | 2 | **0.7719** | **2** | 0.5468 | 2 | 0.7566 | 3 | 0.5954 | 2 | 0.6678 | 4 |
| 14 | 0.7952 | 2 | 0.8087 | 2 | 0.6604 | 2 | 0.7831 | 3 | 0.5992 | 2 | **0.5351** | **3** |
| 15 | 0.8028 | 3 | 0.8057 | 2 | 0.5613 | 3 | 0.7883 | 3 | 0.5528 | 2 | 0.6173 | 4 |
| 16 | 0.7732 | 3 | 0.8171 | 3 | 0.5832 | 2 | 0.8445 | 4 | 0.6125 | 3 | 0.6438 | 4 |
| 17 | 0.7178 | 2 | 0.8194 | 4 | 0.5873 | 3 | 0.8779 | 4 | 0.5957 | 2 | 0.6164 | 3 |
| 18 | **0.716** | **2** | 0.8941 | 4 | 0.5786 | 3 | 0.8219 | 4 | 0.5987 | 2 | 0.6204 | 3 |
| 19 | 0.7520 | 2 | 0.8077 | 2 | **0.4984** | **2** | 0.7931 | 3 | **0.552** | **2** | 0.5515 | 3 |
| 20 | 0.761 | 2 | 0.8947 | 4 | 0.5838 | 3 | 0.8789 | 4 | 0.6537 | 2 | 0.6204 | 3 |

The best obtained clustering is bolded for each dataset

*Denotes Davies Bouldin index (DBI)

**Denotes the number of clusters

**Table 13** Details of the experimental results for 20 runs of the EWOA–NDBSCAN2-Sugeno

| | Daily Demand Forecasting Orders | | Istanbul Stock Exchange | | Dow Jones Index | | ADL Recognition | | Software Engineering Teamwork | | User Identification From Walking Activity | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** | DBI* | NC** |
| 1 | 0.8767 | 4 | 0.8268 | 3 | 0.6132 | 3 | 0.8775 | 4 | 0.6591 | 3 | 0.6253 | 4 |
| 2 | 0.8631 | 3 | 0.8599 | 4 | 0.4796 | 3 | 0.8719 | 4 | 0.6752 | 3 | 0.5429 | 3 |
| 3 | 0.8185 | 3 | 0.9567 | 4 | 0.6576 | 4 | 0.7677 | 3 | 0.5989 | 2 | 0.5824 | 4 |
| 4 | 0.8424 | 4 | 0.8838 | 4 | 0.6152 | 3 | 0.8461 | 4 | 0.6637 | 4 | 0.5479 | 3 |
| 5 | 0.8023 | 3 | 0.7676 | 3 | 0.5297 | 3 | 0.8673 | 4 | 0.5902 | 2 | **0.5211** | **3** |
| 6 | 0.8105 | 3 | 0.9567 | 4 | 0.6576 | 4 | 0.7717 | 3 | 0.5589 | 2 | 0.5944 | 4 |
| 7 | 0.8244 | 3 | 0.9038 | 4 | 0.5621 | 2 | 0.9494 | 4 | 0.7115 | 4 | 0.5862 | 2 |
| 8 | 0.7947 | 3 | 0.9165 | 4 | 0.6533 | 3 | 0.8211 | 3 | 0.7907 | 4 | 0.617 | 4 |
| 9 | 0.7682 | 2 | 0.7581 | 2 | 0.636 | 3 | 0.7574 | 3 | 0.5818 | 2 | 0.6129 | 3 |
| 10 | **0.7157** | **2** | 0.8977 | 4 | **0.4604** | **2** | 0.8259 | 3 | 0.5908 | 2 | 0.5329 | 3 |
| 11 | 0.8634 | 4 | 0.8338 | 3 | 0.6152 | 3 | 0.8416 | 4 | 0.6387 | 4 | 0.5469 | 3 |
| 12 | 0.8351 | 3 | 0.8329 | 3 | 0.4966 | 3 | 0.8398 | 4 | 0.6607 | 4 | 0.5339 | 3 |
| 13 | 0.7831 | 2 | 0.8897 | 4 | 0.5251 | 2 | 0.8166 | 3 | 0.578 | 2 | 0.5329 | 3 |
| 14 | 0.8135 | 2 | 0.8465 | 4 | 0.6588 | 3 | 0.9578 | 5 | 0.7078 | 4 | 0.5812 | 3 |
| 15 | 0.8202 | 3 | **0.7623** | **2** | 0.5161 | 2 | 0.8357 | 5 | 0.7022 | 4 | 0.6615 | 4 |
| 16 | 0.8124 | 3 | 0.9380 | 3 | 0.5621 | 2 | 0.9494 | 4 | 0.7451 | 4 | 0.5826 | 2 |
| 17 | 0.7291 | 2 | 0.7578 | 2 | 0.6492 | 3 | 0.8573 | 5 | **0.5471** | **3** | 0.6629 | 4 |
| 18 | 0.8104 | 3 | 0.7926 | 2 | 0.5624 | 3 | 0.8912 | 4 | 0.6827 | 3 | 0.5349 | 3 |
| 19 | 0.7680 | 2 | 0.8152 | 4 | 0.5864 | 3 | **0.7541** | **3** | 0.5888 | 2 | 0.6777 | 4 |
| 20 | 0.7718 | 2 | 0.8869 | 3 | 0.5709 | 3 | 0.8321 | 3 | 0.6114 | 3 | 0.6714 | 4 |

The best obtained clustering is bolded for each dataset

*Denotes Davies Bouldin index (DBI)

**Denotes the number of clusters

## References


Angelova M, Pencheva T (2011) Tuning genetic algorithm parameters to improve convergence time. Int J Chem Eng 2011:1–7

Aydilek IB, Arslan A (2013) A hybrid method for imputation of missing values using optimized fuzzy *c*-means with support vector regression and a genetic algorithm. Inf Sci 233:25–35

Berkhin P (2006) A survey of clustering data mining techniques. In: Kogan J, Nicholas C, Teboulle M (eds) Grouping multidimensional data. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-28349-8_2

Bezdek JC, Pal NR (1995) Cluster validation with generalized Dunn's indices. In: Proceedings 1995 second New Zealand international two-stream conference on artificial neural networks and expert systems. IEEE

Carvalho DR, Freitas AA (2004) A hybrid decision tree/genetic algorithm method for data mining. Inf Sci 163(1):13–35

Ceci M, Cuzzocrea A, Malerba D (2015) Effectively and efficiently supporting roll-up and drill-down OLAP operations over continuous dimensions via hierarchical clustering. J Intell Inf Syst 44(3):309–333

Chaudhuri S, Dayal U (1997) An overview of data warehousing and OLAP technology. ACM Sigmod Rec 26(1):65–74

Chen J (2012) Hybrid clustering algorithm based on PSO with the multidimensional asynchronism and stochastic disturbance method. J Theor Appl Inf Technol 46(1):434–440

Cheng T (2017) An improved DBSCAN clustering algorithm for multi-density datasets. In: Proceedings of the 2nd international conference on intelligent information processing. ACM

Darong H, Peng W (2012) Grid-based DBSCAN algorithm with referential parameters. Phys Procedia 24:1166–1170

Davies DL, Bouldin DW (1979) A cluster separation measure. IEEE Trans Pattern Anal Mach Intell 2:224–227

Freitas AA (2003) A survey of evolutionary algorithms for data mining and knowledge discovery. In: Ghosh A, Tsutsui S (eds) Advances in evolutionary computing. Natural Computing Series, Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-18965-4_33

Gnanapriya S et al (2010) Data mining concepts and techniques. Data Min Knowl Eng 2(9):256–263

Han J, Pei J, Kamber M (2011) Data mining: concepts and techniques. Elsevier, Amsterdam

Hema R, Malik N (2010) Data mining and business intelligence. In: Proceedings of the 4th national conference

Herrera F, Lozano M (2003) Fuzzy adaptive genetic algorithms: design, taxonomy, and future directions. Soft Comput 7(8):545–562

Huang Z (1997) A fast clustering algorithm to cluster very large categorical data sets in data mining. DMKD 3(8):34–39

Johnson RJ, Williams JP, Bauer KW (2013) AutoGAD: an improved ICA-based hyperspectral anomaly detection algorithm. IEEE Trans Geosci Remote Sens 51(6):3492–3503

Joshi A, Kaur R (2013) A review: comparative study of various clustering techniques in data mining. Int J Adv Res Comput Sci Softw Eng 3(3)

Karafotias G, Hoogendoorn M, Eiben ÁE (2015) Parameter control in evolutionary algorithms: trends and challenges. IEEE Trans Evol Comput 19(2):167–187

Karami A, Johansson R (2014) Choosing DBSCAN parameters automatically using differential evolution. Int J Comput Appl 91(7):1–11

Kumar KM, Reddy ARM (2016) A fast DBSCAN clustering algorithm by accelerating neighbor searching using Groups method. Pattern Recognit 58:39–48

Liço L (2017) Data mining techniques in database systems

Liu J, Lampinen J (2005) A fuzzy adaptive differential evolution algorithm. Soft Comput 9(6):448–462

Mamdani EH, Assilian S (1975) An experiment in linguistic synthesis with a fuzzy logic controller. Int J Man Mach Stud 7(1):1–13

Mining WID (2006) Data mining: concepts and techniques. Morgan Kaufinann, Burlington

Nagar P, Srivastava S (2008) Application of genetic algorithms in data mining. In: 2nd National conference on challenges and opportunities in information technology

Pei Z, Hua X, Han J (2008) The clustering algorithm based on particle swarm optimization algorithm. In: 2008 International conference on intelligent computation technology and automation (ICICTA). IEEE

Pujari AK (2001) Data mining techniques. Universities Press, Cambridge

Rousseeuw PJ (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math 20:53–65

Scovanner P, Ali S, Shah M (2007) A 3-dimensional sift descriptor and its application to action recognition. In: Proceedings of the 15th ACM international conference on multimedia. ACM

Smiti A, Eloudi Z (2012) DBSCAN-GM: An improved clustering method based on Gaussian means and DBSCAN techniques. In: 2012 IEEE 16th International conference on intelligent engineering systems (INES). IEEE

Smiti A, Eloudi Z (2013) Soft DBSCAN: improving DBSCAN clustering method using fuzzy set theory. In: 2013 The 6th international conference on human system interaction (HSI). IEEE

Takagi T, Sugeno M (1993) Fuzzy identification of systems and its applications to modeling and control. In: Kozma R (ed) Readings in fuzzy sets for intelligent systems. IEEE Transactions on Systems, Man, and Cybernetics: Systems, pp 387–403

Vercellis C (2011) Business intelligence: data mining and optimization for decision making. Wiley, New York

Wang G-G, Deb S, dos Santos Coelho L (2018) Earthworm optimisation algorithm: a bio-inspired metaheuristic algorithm for global optimisation problems. IJBIC 12(1):1–22

Woo HJ, Joo KH, Park NH (2015) A clustering OLAP analysis in a big data stream environment

Zhao Y-Q, Yang J (2015) Hyperspectral image denoising via sparse representation and low-rank constraint. IEEE Trans Geosci Remote Sens 53(1):296–308

Zhao B et al (2007) Image segmentation based on ant colony optimization and *K*-means clustering. In: 2007 IEEE International conference on automation and logistics. IEEE