International Journal of Computational Intelligence Systems
Vol. **13(1)**, 2020, *pp.* 757–770
DOI: https://doi.org/10.2991/ijcis.d.200601.001; ISSN: 1875-6891; eISSN: 1875-6883
https://www.atlantis-press.com/journals/ijcis/

Research Article

An Efficient Clustering Algorithm for Mixed Dataset of Postoperative Surgical Records

Hemant Petwal*, Rinkle Rani

Department of Computer Science and Engineering, Thapar Institute of Engineering and Technology, Patiala, Punjab, India

ARTICLE INFO

Article History

Received 21 Oct 2019
Accepted 19 May 2020

Keywords

Data clustering
Meta-heuristic
Artificial electric field algorithm
Distance measure
Mixed dataset

ABSTRACT

In data mining, data clustering is a prevalent data analysis methodology that organizes unlabeled data points into distinct clusters based on a similarity measure. In recent years, several clustering algorithms found, dependent on a predefined number of clusters and centered around the dataset with either numeric or categorical attributes only. However, many real-world engineering, scientific, and industrial applications involve datasets with mixed numeric as well as categorical attributes but lack domain knowledge (target labels). Clustering unlabeled-mixed datasets is a challenging task as (1) it is difficult to estimate the number of clusters in the absence of domain knowledge and (2) mathematical operations cannot be applied directly to the mixed dataset. In this paper, an efficient searching and fast convergent automatic data clustering algorithm based on population-based meta-heuristic optimization is proposed to deal with the mixed dataset. The proposed clustering algorithm aims to find the optimal number of cluster partitions automatically. It utilizes a real-coded variable-length candidate solution to detect the optimal number of clusters automatically. The concepts of threshold setting and cut-off ratio are used in the optimization process to refine the clusters. The similarity between data points and different cluster centers is measured using Euclidean distance (for numeric attributes) and the probability of co-occurrence of values (for categorical attributes). The proposed algorithm is compared with existing mixed data clustering techniques based on a statistical significance test and two robustness measures: Average accuracy and Standard deviation. Finally, the proposed algorithm is validated by applying to a real historical postoperative surgical mixed data set obtained from a surgical department of a multispecialty hospital in India. Results show the effectiveness, robustness, and usefulness of the proposed clustering algorithm.

1. INTRODUCTION

Data clustering is a data analysis approach that arranges unlabeled data into different groups based on a similarity measure. Each group is called a "cluster," which shows similarity among the data in it and differs from the set of data in other clusters. Clustering is most widely used in those disciplines where multivariate data analysis is required. In recent years, cluster analysis has played a significant role in the distinct domains of various fields such as engineering, life, and medical sciences, earth sciences, and economics [1–3]. The elementary problem of the clustering analysis is to accurately determine the approximate number of clusters, as this number influences the clustering outcomes to a large extent [4]. Clustering algorithms are classified into two main categories: partitional clustering [5] and hierarchical clustering [6]. The hierarchical clustering arranges data points in a hierarchical tree structure based on the homogeneity among the data points. It overlooks the shape and size of the formed clusters. Further, this clustering allocates a single cluster to a data point at a time that renders the cluster structure static. Contrary, partitioning clustering analyzes the dataset and organizes data points into clusters based on the similarity among data points. The partitioning clustering aims to optimize a global criterion involving minimizing the similarity among the elements within a cluster and maximizing the disparity between different clusters. Although both of these algorithms prove their usefulness and performance in various domains, both still have some critical limitations. The efficacy of these algorithms depends on the foreknowledge of the number of clusters present in the datasets. Since different datasets, especially in real-world applications, have diverse patterns, cluster analysts lack information on how many appropriate clusters exist in the dataset [7]. Hence, these algorithms that require the cluster number as an onset parameter cannot be used effectively. Since most real-world datasets do not have class labels, there are no specific criteria for directing clustering analysis. It is considered a major limitation [4] of the dataset, which makes it a challenging task to find a suitable number of clusters. Therefore, determining the optimum number of clusters in a data set has become an essential research issue to address such limitations. In recent years, the concept of the automatic clustering method has been used to overcome this limitation in clustering. Automatic clustering is defined as an analytic process of determining a suitable number of clusters in the dataset, irrespective of any prior knowledge related to the dataset [8]. Several automatic clustering algorithms, mostly inspired by natural phenomena, involving genetic algorithm (GA) [9,10], particle swarm optimization (PSO) [11,12],

* *Corresponding author. Email: hemant.petwal@thapar.edu*

gravitational search algorithm (GSA) [13,14], differential evolution (DE) [15], bacterial evolutionary algorithm (BEA) [16], and bee colony optimization algorithm (BCA) [17,18], etc., are introduced in recent years. In these algorithms, clustering is considered as an optimization activity, aiming to maximize the similarity among the data of a cluster and maximize the disparity among disjoint clusters [19] . These algorithms have proven their higher convergence speed and efficacy in producing quality outcomes not only in optimization but also in clustering analysis. In this paper, clustering analysis is framed as an optimization problem, and an efficient clustering algorithm based on a meta-heuristic optimization algorithm, called Artificial Electric Field Algorithm (AEFA) [20], is proposed to address the automatic clustering problem. AFEA is a recent inclusion in the global pool of population-based meta-heuristic algorithms that contributes to global optimization. To the extent of our knowledge, there is no such contribution where AEFA has been employed for automatic clustering analysis. The proposed clustering algorithm attempts to address two issues. The first is to select the optimal clusters automatically, and the second is to focus on mixed datasets with numerical as well as categorical attributes contrary to recent studies that focus on data set with either numerical or categorical datasets only. AEFA simulates the Coulomb's law of attraction electrostatic force and law of motion. Traditional AEFA begins with an initial population of charged particles (candidate solutions). During the optimization process of AEFA, the fitness of each charged particle is computed that helps in determining the charges and forces on each charged particle, which further enables the global movement of all charged particles toward a heavier charged particle. Finally, a particle with heavier charge is selected as a globally optimum solution. Similarly, in the proposed clustering algorithm, the initial population of charged particles is formed by selecting a set of random cluster centers from the dataset where each charged particle is encoded using a real-coded variable-length encoding approach. The fitness of each charged particle is computed using a fitness function based on the SD index [21]. The probability of co-occurrence of value for categorical attributes along with Euclidean distance for numeric attributes is used as a distance measure for assigning the data points to a particular cluster. The optimization process is carried out until the optimum clusters are obtained.

1.1. Our Contribution

1. An efficient data clustering algorithm for the mixed dataset based on population-based meta-heuristic is proposed.
2. The concepts of threshold setting and cut-off ratio are used in the optimization process to refine the clusters.
3. The proposed algorithm is evaluated on real-life datasets and compared with existing mixed dataset clustering techniques. Further, it is also validated using the real postoperative surgical dataset of a multispecialty hospital.

This paper is organized in the following way: Section 2 covers an overview of the existing literature on clustering analysis; Section 3 describes the preliminaries and background algorithms; Section 4 describes the proposed clustering algorithm in detail; Section 5 presents results and performance of the proposed algorithm in comparison to existing clustering techniques; Section 6 sums up the findings and discuss the future work of this research in concluding remarks.

2. RELATED WORK

K-means is a center-based clustering approach widely used in the partitioning of the data set for its simplicity and efficacy. The major drawback of the k-means algorithm involves its dependency on an initial number of the cluster centers and its faster convergence to the local optima [22,23]. Although several clustering algorithms [24–26] have been proposed to prevent solution from being stuck in the local optima, still the reliance of the algorithm on prior cluster number information and its adverse effect on clustering performance has emerged as an issue [27–30]. For addressing such an issue, several optimization algorithms have been adapted to implement automatic clustering analysis. The first-ever contribution toward automatic clustering was based on evolutionary computing, called EP-clustering [31]. EP-clustering was aimed to minimize the DB index and WGS indices for improving the efficiency of exploration and exploitation. This algorithm produced better results as compared K-means algorithm when implemented on real-life datasets. In recent years, Chen *et al.* [32] introduced a GEP-cluster algorithm based on gene expression [33] programming for the automatic clustering of data. GEP-cluster comprised of clustering algebra as a new concept for identifying the best cluster with no prior knowledge, and an automatic merging clustering algorithm for merging clusters automatically. The results indicated that the GEP-cluster algorithm is found noise sensitive as well as incompetent to high-dimensional datasets. Lee and Antonsson [34] introduced evolutionary strategy-based clustering, also called, ES-clustering for automatic clustering of data. In ES-clustering, the initial population is encoded using genomes of variable-length and (10 + 60) ES selection strategy, and an improved mean square error is selected as a fitness function. Tseng and Yang [35] introduced an automatic clustering, called CLUSTERING, based on GA. In CLUSTERING, the data points are grouped into the small clusters using the nearest neighbor clustering method, then by GA, and using a difference between the WGS and BGS indices as the fitness function, all small clusters are combined to a larger cluster. Finally, a heuristic strategy is implemented to determine the cluster partition. The results revealed that the CLUSTERING algorithm outperformed when compared with K-means, complete-link, and single-link. Bandyopadhyay and Maulik [36] proposed a nonparametric VGA-clustering algorithm. In the VGA-clustering algorithm, a variable-length encoding is utilized for encoding chromosomes, composed of cluster coordinates, in the population, and the I-index is used to compute the fitness of each solution. The algorithm is evaluated on real-life datasets and produced an appropriate number of clusters. Bandyopadhyay and Saha [37] introduced a variable length and point-symmetry-based genetic clustering algorithm (VGAPS) for the automatic partitions of the dataset. The algorithm utilized a point symmetry (PS)-based distance as similarity measure, and the Sym-index as for fitness computation. The experimental analysis revealed that the VGAPSA produced better results. Liu *et al.* [7] presented an automatic clustering algorithm based on the GA, also called AGCUK. This algorithm aimed to determine cluster partition in the absence of the prior number of clusters. The algorithm adopted a real-code variable-length representation for encoding cluster centers in chromosomes

and utilized DB index to compute the fitness of an individual solution. This algorithm produced better outcomes in terms of the misclassification rate. Ahmad and Dey [38] proposed K-means for the mixed (numeric and categorical) dataset, also called KMCMD, and outperformed the K-means algorithm. K-harmonic mean with mixed data set was, also called KHMCMD, proposed by Ahmad and Hashmi [39] to cluster the mixed dataset. KHMCMD found competent in producing satisfactory clustering outcomes in comparison to K-means algorithm. Chang *et al.* [40] suggested an algorithm for automatic clustering. The algorithm adopted dynamic niching GA with niche migration for determining the cluster centers and the number of clusters automatically. The algorithm is compared with species-conserving GA (SCGA) [40], and the dynamic fitness sharing (DFS) [40] algorithm using real-life datasets. It outperformed the compared algorithm in terms of determining a suitable number of clusters. A framework called ETSAs, based on tabu search, and genetic operators, is proposed by Pan and Cheng [41] for automatic clustering. The framework determines the optimal number of clusters utilizing an evolutionary approach that assists in producing a competition population by utilizing two parallel reproduction process. An automatic clustering algorithm, based on the DE approach, and called ACDE, is proposed by Das *et al.* [15] In the proposed algorithm, each individual is encoded as a real value vector, containing the activation threshold linked to the cluster centers. ACDE performed better in terms of classification error. Das *et al.* [16] proposed an automatic clustering algorithm based on a bacterial evolutionary algorithm, called (ACBEA). The proposed algorithm consists of two operations: bacterial mutation and the gene transfer operation for population generation. These two operations are modified for managing the variable length in a chromosome that encodes cluster centers. Further, the fitness of each individual is computed using the CS index. Omran *et al.* [42] proposed a PSO-based dynamic clustering approach (DCPSO) for image [43,44] segmentation. In the algorithm, the "best" number of clusters is selected using binary PSO. Then, the K-means is implemented to refine the chosen clusters. The results showed that DCPSO produced suitable cluster numbers. Cura [45] introduced a novel PSO clustering algorithm, called CPSO, that determine the number of clusters in both scenarios, whether the number of clusters is known or unknown, respectively. Further, two fitness functions are used in the algorithm: WGS index when the number of clusters is specified and the difference between the BGS and WGS indices when the number of clusters is unknown. The CPSO produced better results in terms of cluster quality. Chowdhury *et al.* [46] proposed an IWO-clustering for the automatic evolution of clusters. In this algorithm, the weed strings that represent the population and encode cluster centers, are encoded using a variable-length encoding approach. Further, a modified Sym Index is used as the fitness function in this algorithm. Kumar and Kumar [13] introduced an adaptive harmony search-based automatic clustering algorithm. This algorithm adapted a real-coded variable-length approach for encoding the harmony vector to detect the number of clusters automatically. The algorithm produced optimally compacted and well-separated clusters. Kumar and Kumar [14] proposed an automatic clustering and feature selection algorithm based on the GSA. This algorithm aimed to produce an optimal number of clusters and to select relevant cluster features at run time. The algorithm utilized a modified I-index as fitness computation. The algorithm produced better results in terms of classification accuracy. A summary of literature is presented in Table 1.

3. PRELIMINARIES AND BACKGROUND

This section briefly discusses the basic concepts of partitioning clustering, AEFA, and the distance measure for the mixed dataset.

3.1. Partitioning Clustering

Partitioning clustering is a data clustering approach that groups the data points into disjoint clusters. Let us consider a dataset $Z = \{Z_1, Z_2, Z_3, \ldots, Z_n\}$ of n data points. In the dataset, each data point is represented by D attributes. For example, $Z_j = (Z_{j1}, Z_{j2}, \ldots, Z_{jD})$ is a vector representing the j^{th} data point, and Z_{ji} represents the i^{th} attribute of Z_j. The objective of the partitional clustering algorithm is to determine the disjoint cluster (C_i) that satisfies the following condition:

$$C_i \neq \phi,\ i = 1, 2 \ldots k \tag{1}$$

$$\sum_{i=1}^{k} C_i = Z \tag{2}$$

$$C_i \cap C_j \neq \phi\ \forall i, j \tag{3}$$

where k represents the number of clusters.

3.2. Distance Measure for Mixed Datasets

The belonging of a data point to a cluster is measured using the distance computed between the cluster and data point. Distance measure ensures the similarity among the data points belonging to the same cluster and dissimilarity between disjoint clusters. Grouping a mixed dataset into clusters is a significantly challenging task. It requires a suitable distance measure, which can compute the similarity/dissimilarity between data points effectively, to be involved in the clustering process. In this paper, the distance measure proposed by Ahmad and Dey [38] is used. The distance (ϑ) between any data point (d_i) and cluster center (c_j) is computed as follows:

$$\vartheta(d_i, c_j) = \sum_{t=1}^{m_r} \left(w_t \left(d_{it}^r - c_{jt}^r \right) \right)^2 + \sum_{t=1}^{m_c} (d_{it}^c, C_{jt}^c)^2 \tag{4}$$

Here, the first component represents the distance between t^{th} numeric attribute value of data point d_i; (d_{it}^r) and cluster center (c_{jt}^r) and the second component represents the distance between the t^{th} categorical attribute value of data point d_i ; (d_{it}^c) and cluster center (c_{jt}^c). w_t represents the significance of t^{th} numeric attribute. It is calculated by computing the average distance between all pairs of discretized numeric attribute values. The distance between two attribute values of a categorical attribute is measured by computing co-occurrence of these values with attribute values of other categorical attributes.

3.3. Artificial Electric Field Algorithm

AEFA [20] is a population-based meta-heuristic, which mimics the Coulomb's law of attraction electrostatic force and law of motion. In AEFA, the possible candidate solutions of the given problem

Table 1 | Summary of existing clustering methods.

Author(s)	Objective/Work Done	Technique Proposed/Used	Performance Parameters	Research Gap(s)
Liu *et al.* [7]	The author proposed an automatic clustering algorithm based on the genetic algorithm	Automatic genetic clustering for unknown K (AGCUK)	1. Degree of population diversity 2. Degree of selection pressure 3. Average and standard deviation of DB-index 4. Average and standard deviation of the number of clusters	The algorithm is limited to either numeric or categorical datasets only
Kumar *et al.* [13]	The author proposed an automatic data clustering using an adaptive harmony search algorithm (AHSA)	Adaptive harmony search algorithm (AHSA)	1. Inter-cluster distance 2. Intra-cluster distance 3. Trace	The algorithm is limited to numeric attributes only
Kumar and Kumar [14]	The author proposed an automatic data clustering and feature selection using the gravitational search algorithm (GSA)	Automatic clustering and feature selection using gravitational search algorithm (GSA_CFS)	1. Silhouette index 2. Classification error	The algorithm is limited to either numeric or categorical datasets only
Das *et al.* [15]	The author proposed an automatic clustering algorithm using an improved differential evolution algorithm	Automatic clustering DE algorithm (ACDE)	1. Mean classification error 2. Standard deviation	The algorithm is limited to either numeric or categorical datasets only
Das *et al.* [16]	The author proposed an automatic clustering algorithm based on the bacterial evolutionary approach	Automatic clustering using the bacterial evolutionary algorithm (ACBEA)	1. CS-measure 2. Inter-cluster distance 3. Intra-cluster distance	The algorithm is limited to either numeric or categorical datasets only
Sarkar *et al.* [31]	The author proposed data clustering based on evolutionary programming	EP-clustering	DB-Index	This algorithm is sensitive to the initial cluster number. Further, this algorithm is limited to numeric attributes only
Chen *et al.* [32]	The author proposed an automatic clustering algorithm based on gene expression programming	1. GEP-cluster algorithm 2. Automatic merging cluster algorithm	1. Mean square error 2. Success rate	This algorithm is sensitive to noise and found efficient for high dimensional data
Lee and Antonsson [34]	The author proposed a dynamic clustering algorithm based on evolutionary strategy approach	Evolutionary strategy-based clustering (ES-clustering)	Heuristic mean square error	The efficacy of the proposed algorithm is not tested on high-dimensional data
Tseng and Yang [35]	The author proposed an automatic clustering algorithm using genetic algorithm and heuristic strategy	CLUSTERING	The average distance from the cluster center	The algorithm did not focus on mixed datasets
Bandyopadhyay and Saha [37]	The author proposed a point symmetry-based genetic clustering algorithm for automatic portioning of data	Variable-length point symmetry-based genetic clustering algorithm (VGAPS)	1. Minkowski score 2. Sym-index	This algorithm is focused on point-based symmetry only. Further, the algorithm is limited to either numeric or categorical data only
Ahmad and Dey [38]	The author proposed K-mean clustering algorithm for the mixed data type	K-mean clustering for mixed dataset (KMCMD)	1. Micro-precision 2. Micro-recall	The cluster center initialization problem persists
Chang *et al.* [40]	The author proposed an automatic clustering algorithm based on dynamic niching and niche migration	Dynamic niching and niche migration clustering (DNNM-Clustering)	1. The average number of niches 2. Standard error	The algorithm is sensitive to the size of the initial population

(*Continued*)

Table 1 Summary of existing clustering methods. *(Continued)*

Author(s)	Objective/Work Done	Technique Proposed/Used	Performance Parameters	Research Gap(s)
Pan and Cheng [41]	The author proposed a framework for automatic clustering algorithm based on tabu search and genetic operators	An evolution-based tabu search approach (ETSA)	Cluster validity index (PBM-index) value	The algorithm is limited to either numeric or categorical datasets only
Cura [45]	The author proposed a data clustering algorithm using enhanced particle swarm optimization (PSO)	Enhanced particle swarm optimization-based clustering (EPSO-clustering)	1. Error rate 2. Intra-cluster distance 3. Inter-cluster distance	Robustness decreases when dealing with problems where the numbers of clusters are unknown
Chowdhury *et al.* [46]	The author proposed an automatic clustering based on invasive weed optimization (IWO) algorithm	IWO-clustering	Minkowski score	The algorithm is limited to either numeric or categorical datasets only

are represented as a collection of the charged particles. The charge associated with each charged particle helps in determining the performance of each candidate solution. Attraction electrostatic force causes each particle to attract toward one another resulting in the global movement toward particle with the heavier charge. A candidate solution to the problem corresponds to the position of charged particles and fitness function, which determines their charge and unit mass. The steps of AEFA are as follows:

Step 1. Initialization of population: A population of P candidate solutions (charged particle) is initialized as follows:

$$CP_i = \left(CP_i^1, CP_i^2, CP_i^k \ldots CP_i^D\right), \forall\, i = 1, 2, 3 \ldots n \tag{5}$$

where, CP_i^k represents the position of i^{th} charged particle in the k^{th} dimension and D is the dimensional space.

Step 2. Fitness evaluation: Performance of each charged particle depends on the fitness values at each iteration. The best and worst fitness is computed as follows:

$$Best\ (t) = \ min_{i=1}^{n}\ Fitness\ (t) \tag{6}$$

$$Worst\ (t) = \ max_{i=1}^{n}\ Fitness\ (t) \tag{7}$$

where $Fitness\ (t)$ and n are the fitness value of i^{th} charged particle and size of the population, respectively at time t. $Best\ (t)$, $Worst\ (t)$ represents the best fitness and the worst fitness of all charged particles at time t, respectively.

Step 3. Computation of Coulomb's constant: At time t, the Coulomb's constant is denoted by $K_c\,(t)$ and computed as follows:

$$K_c\,(t) = \ K_0 * \exp\left(-\alpha \frac{iter}{N}\right) \tag{8}$$

where K_0 represents initial value and α is a random value, respectively. $iter$ and N represents current iteration and maximum number of iterations.

Step 4. Compute the charge of charged particles: At time t, the charge of i^{th} charged particle is represented by $Q_i\,(t)$. It is computed based on the current population's fitness as follows:

$$Q_i\,(t) \ = \ \frac{q_i\,(t)}{\sum_{i=1}^{n} q_i\,(t)} \tag{9}$$

$$q_i\,(t) = \ exp\left(\frac{fitness_{cp_i}\,(t) - Worst\ (t)}{Best\ (t) - Worst\,(t)}\right) \tag{10}$$

Step 5. Compute the electrostatic force and acceleration of the charged particles:

1. The electrostatic force exerted by j^{th} charged particle on the i^{th} charged particle in the D^{th} dimension at time T is computed as

$$F_{ij}^D\,(t) = \ K\,(t)\ \frac{\left(Q_i\,(t) * Q_j\,(t)\right) * \left(P_j^D\,(t) - \ X_j^D\,(t)\right)}{R_{ij}\,(t) + \varepsilon} \tag{11}$$

$$F_i^D\,(t) = \sum_{j=1,\ j\neq i}^{N} rand\,() \ * F_{ij}^D\,(t) \tag{12}$$

where $Q_i\,(t)$ and $Q_j\,(t)$ are the charges of i^{th} and j^{th} charged particle at any time t. ε is a small positive constant and $R_{ij}\,(t)$ is the distance between two charged particles i and j. $P_j^D\,(t)$ and $X_j^D\,(t)$ are the global best and current position of the charged particle at time t. $F_i^D\,(t)$ is the net force exerted on i^{th} charged particle by all other charged particles at time t. $rand\,()$ is a uniform random number in [0, 1] interval.

1. The acceleration $a_i^D(t)$ of i^{th} charged particle at time t in D^{th} dimension is computed using the Newton law of motion as follows:

$$a_i^D\,(t) = \frac{Q_i\,(T) * E_i^D\,(t)}{M_i^D\,(t)}, E_i^D\,(t) = \ \frac{F_i^D\,(t)}{Q_i\,(T)} \tag{13}$$

where $E_i^D\,(t)$ and $M_i^D\,(t)$ represents the electric field and unit mass of i^{th} charged particle at any time and in D^{th} dimension respectively

Step 6. Updation of velocity and position of charged particle: At time t, the position and velocity of i^{th} charged particle in D^{th} dimension is updated as follows:

$$vel_i^D\,(t+1) = \ rand_i() * \ vel_i^D\,(t) + a_i^D\,(t) \tag{14}$$

$$CP_i^D(t+1) = CP_i^D(t) + vel_i^D(t) \quad (15)$$

where *rand*() is a uniform random number in the interval [0, 1].

4. PROPOSED CLUSTERING ALGORITHM

In this section, the proposed clustering algorithm is described in detail. The algorithm starts with accepting pre-processed mixed dataset as input and initializes parameters. Then, a population of candidate solutions is generated, where each solution composed of two segments: the first segment represents threshold values and the second segment represents cluster centers. The threshold values of the first segment determine whether or not the corresponding cluster centers are active in the second segment. Further, by computing fitness value for each candidate solution, the best solution is selected. The population is iteratively updated until the termination conditions are satisfied, and the optimal solution is returned. The symbols used in the proposed algorithm are presented in Table 2. The detailed workflow and pseudocode of the proposed clustering algorithm are presented in Figure 1 and Algorithm 1, respectively.

4.1. Population Generation

In this step, the population of candidate solutions is initialized. For the dataset, which has n data point with dimension D each, the maximum number of cluster center is computed as follows:

$$K_{max} = \sqrt{n} \quad (16)$$

Each candidate solution (X_i) is represented by $D_x = K_{max} + K_{max} * D$ dimension. The first segment of X_i contains K_{max} threshold values (Th), which are generated randomly within the interval of [0, 1]. The second segment comprises the k_{max} cluster centers, where each of which has D dimension. The threshold values representation of each candidate solution (X_i) is as follows:

Table 2 Symbols used in the proposed algorithm.

Symbol	Definition	Symbol	Definition
ϑ	Distance between the data point and cluster center	a_i^D	Acceleration of i^{th} charged particle in D^{th} dimension at time t
d_{it}^r	t^{th} numeric value of i^{th} data point	E_i^D	Electric field of i^{th} charged particle in D^{th} dimension at time t
d_{it}^c	t^{th} categorical value of i^{th} data point	M_i^D	Mass of i^{th} charged particle in D^{th} dimension at time t
CP_i^D	Position of i^{th} charged particle (candidate solution) in D^{th} dimension	vel_i^D	Velocity of i^{th} charged particle in D^{th} dimension at time t
CP_i^j	j^{th} cluster center belonging to i^{th} charged particle	$K_{\max}$	Maximum number of clusters centers
D	Dimension of objective space that helps in determining the overall dimension of a candidate solution	n	Population size (number of data points)
Fitness()	Objective fitness function	D_x	Overall dimension of a charged particle (candidate solution)
Best(t)	Best (lowest value for minimization problem and highest value for maximization problem) fitness value at time t	N	Maximum number of iterations
Worst(t)	Worst (highest value for minimization problem and lowest for maximization problem) fitness value at time t	Th_i^j	Threshold value of j^{th} cluster center belonging to i^{th} candidate solution
K_0	Initial value for coulomb's constant	Th_{cl}	The selection threshold value of a cluster center
K_c	Coulomb's constant at time t	T_{cot}	The cut-off threshold value of a cluster center
$q_i(t)$	Small value of charge on i^{th} charged particle at time t that helps in determining the total charge of (i^{th}) charged particle	$SD(CP_i)$	SD index value of a i^{th} charged particle
$Q_i(t)$	Total charge on a i^{th} charged particle at time t	$Scat(CP_i)$	Intra-cluster distance of a i^{th} charged particle
$F_{ij}^D(t)$	Force exerted by j^{th} charge particle on i^{th} charge particle in D^{th} dimension at time t	$Dist(CP_i)$	Inter-cluster distance a i^{th} charged particle
$F_i^D(t)$	Net force on i^{th} charged particle in D^{th} dimension at time (t)	$\sigma_{CP_i}^D$	Variance of clusters belonging to i^{th} charged particle
$P_j^D(t)$	Global best position of j^{th} charged particle in D^{th} dimension at time t	σ_x^D	Variance of dataset (X)
$X_j^D(t)$	Current position of j^{th} charged particle in D^{th} dimension at time t	D_{max}	Maximum distance between the cluster centers of a charged particle
$R_{ij}(t)$	Distance between i^{th} and j^{th} charged particle at time t	D_{min}	Minimum distance between the cluster centers of a charged particle
k_{active}	Total number of active clusters		

Th_i^1	Th_i^j	...	$Th_i^{K_{max}}$	CP_i^1	CP_i^2	CP_i^j	..	$CP_i^{K_{max}}$

Threshold values Cluster centers

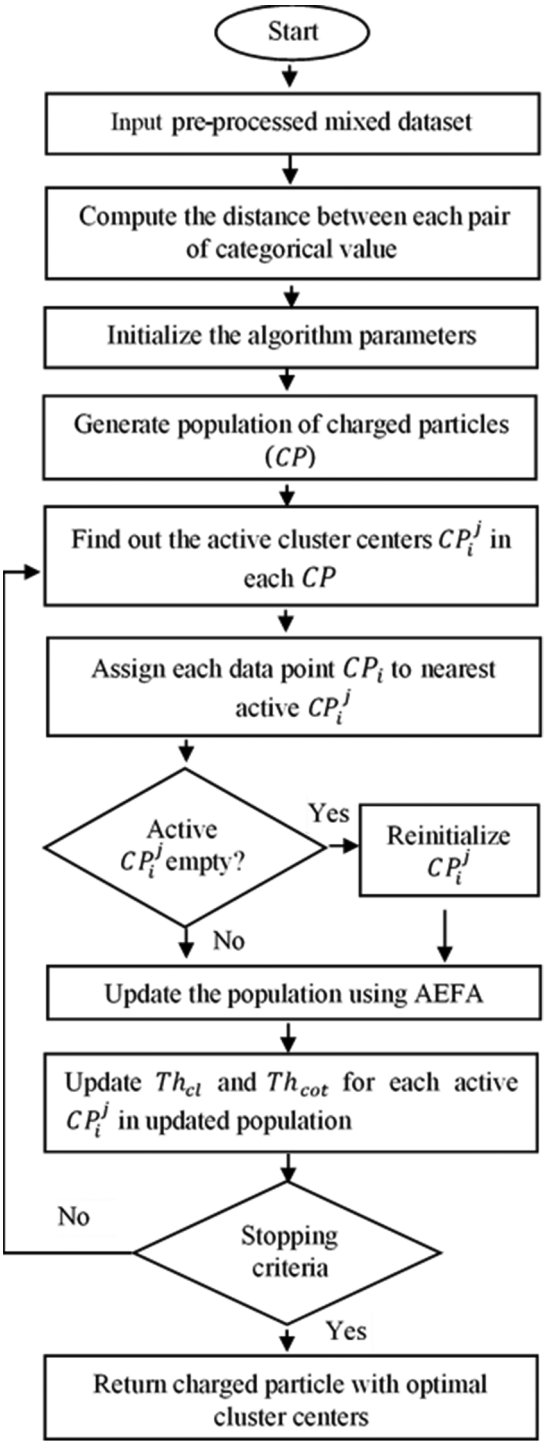


Figure 1 The workflow of the proposed clustering algorithm.

where Th_i^j represent the threshold value of CP_i^j cluster center and CP_i^j represents j^{th} cluster center of i^{th} candidate solution.

4.2. Active Cluster Center Selection

In this step, active cluster centers among k_{max} cluster centers of a candidate solution are selected. Selection of the active cluster centers is based on the following condition:

$$CP_i^j = \begin{cases} 1, & \text{if } Th_i^j > T_{cot} \\ 0, & \text{otherwise} \end{cases} \tag{17}$$

where T_{cot} represents the cut-off threshold for every cluster center. It is initially set to a random value between [0, 1] interval. In succeeding iterations T_{cot} is computed as follows:

$$T_{cot} = \frac{1}{k_{active}} \sum_{l=1}^{k_{active}} Th_{cl} \tag{18}$$

$$Th_{cl} = \sqrt{\frac{1}{n_{cl}} \left(\sum_{i=1}^{m_r} (d_i^r,\ CP_l^r)^2 + \sum_{i=1}^{m_c} (d_i^c,\ CP_l^c)^2 \right)} \tag{19}$$

where Th_{cl} represents the selection threshold of a cluster center. $(d_i^r,\ CP_l^r)$ and $(d_i^c,\ CP_l^c)$ represents distance between i^{th} data points and corresponding cluster center of the numeric and categorical attribute, respectively. k_{active} represents number of active cluster centers. n_{cl} represents the total number of data-points belongs to an l^{th} active cluster center.

4.3. Empty Clusters Validation

A cluster center is said to be empty if no data points or less than 2 data points are assigned to it. Such problems are resolved by reinitializing the cluster center for that candidate solution. The candidate solution is reinitialized by assigning n/k_{active} data points to each nearest active cluster center.

4.4. Fitness Evaluation

In this step, the performance of the candidate solutions (cluster centers) is measured. As the performance critically relies upon a suitable cluster validation criterion. A random selection of criteria for clustering may lead to poor results. Therefore, the SD index [21] is chosen for cluster validation, and the resulting fitness of the candidate solution (charged particle) is computed as follows:

$$Fitness\ (CP_i) = SD\ (CP_i) * \frac{K_{max} - k_{active}}{k_{active} + 1} \tag{20}$$

$$SD\,(CP_i) = a * Scat\,(CP_i) + Dist\ (CP_i) \tag{21}$$

where $Scat\,(CP_i) = \frac{1}{k_{active}} \sum_{i=1}^{k_{active}} \frac{\left\|\sigma_{(CP_i)}\right\|}{\|\sigma_{(x)}\|}$, and

$Dist\ (CP_i) = \frac{D_{max}}{D_{min}} \sum_{k=1}^{k_{active}} \frac{1}{\left(\sum_{z=1}^{k_{active}} \left\| \vartheta \left(CP_i^k, CP_i^z \right) \right\| \right)}$

Here,

1. $\sigma_{(CP_i)}^D = \frac{\sum_{k=1}^{n} \vartheta \left(x_k^D, CP_i^D \right)^2}{n_i}$ and $\sigma_{(x)}^D = \frac{\sum_{k=1}^{n} \vartheta \left(x_k^D, x^D \right)^2}{n}$

Algorithm 1: Proposed clustering algorithm for the mixed dataset of postoperative surgical records

Data: Dataset with both mixed numeric and categorical attributes and n data points
Result: Optimal number of cluster centers
1. Define the maximum number of iterations (N), the maximum number of clusters (K_{max}), population size (n), selection threshold (Th_{cl}) and cut-off threshold (T_{cot})
2. Compute the dimension (D_x) and generate an initial population of charged particles (CP) as cluster centers
3. Initialize $iter = 1$

while $iter < N$ **do**
  for *j=1 to n* **do**
    for *j=1 to* K_{max} **do**
      if $Th_{ij} > T_{cot}$ **then**
        Select and activate the cluster center CP_i^j using Eq. 17
      else
        CP_i^j is set to inactive
      end
    end
    foreach *data point*CP_i *in given mixed dataset* **do**
      Compute the distance between CP_i and active CP_i^j using Eq. 4 and assign the CP_i to the closest active CP_i^j
      Verify and re-initialize the empty CP_i^j as described in section 4.3
    end
  end
  Update the population using the AEFA algorithm (mentioned in section 3.3). The exploration process is guided by the fitness function using Eq. 20 and Eq. 21, and the distance measure for the mixed dataset using Eq. 4.
  Update Th_l and T_{cot} for each CP in the updated population using Eq. 18 and Eq. 19.
  $iter = iter + 1$
end
Return the charged particle (CP) with optimal cluster centers.

2. $D_{max} = \max_{\substack{j\in 1,2,\ldots C \\ j\neq C}} \left(\left\| \vartheta \left(CP_i^j, CP_i^C \right) \right\| \right)$

3. $\max_{\substack{j\in 1,2,\ldots C \\ j\neq C}} \left(\left\| \vartheta \left(CP_i^j, CP_i^C \right) \right\| \right)$

5. EXPERIMENTAL RESULTS AND DISCUSSION

This section is further divided into three subsections. Subsection 5.1 gives a performance comparison of the proposed algorithm with the existing mixed data clustering algorithms. Subsection 5.2 discusses the application of the proposed algorithm to the clustering of postoperative surgical records and Subsection 5.3 presents the statistical significance test of the proposed clustering algorithm.

5.1. Performance Comparison of the Proposed Clustering Algorithm with Existing Mixed Data Clustering Algorithms

At first, 5 real-life datasets are used to evaluate the performance of the proposed clustering algorithm. These datasets are obtained from the UCI machine learning repository https://archive.ics.uci.*edu*/ml/datasets.*php)*. The description of the datasets is shown in Table 3. Then, the performance of the proposed clustering algorithm is compared with existing mixed dataset clustering algorithms. For performance comparison, two cluster quality measures average accuracy [38] and standard deviation [38] are used in this paper. Average accuracy represents the quality and the standard deviation represents the reliability of the clustering algorithm. During each iteration of the proposed clustering algorithm, both average accuracy and standard deviation contribute to the robustness of the clustering algorithm, where a high value of average accuracy and lower standard deviation makes a clustering algorithm more robust.

Table 3 Characteristics of the real-life datasets.

Dataset	Data Points	Attributes		Classes
		Numeric	Categorical	
Heart Disease 1	303	5	8	2
Heart Disease 2	270	6	8	5
Credit approval	690	6	8	2
Iris	150	4	–	3
Soybean	47	–	35	4

5.1.1. Parameter setting

For experiments, the parameters used in the proposed clustering algorithm, i.e., population size (P), the maximum number of clusters $K_{\max}$), Coulomb's constant (K_0), and the maximum number of iterations (N) are initialized as follows: P is set to 20, K_{max} is set to $\sqrt{n}$, where n is the size of the dataset, K_0 is set to 500, and N is set to 20.

5.1.2. Performance comparison

The performance of the proposed clustering algorithm is compared with existing mixed data clustering algorithm: K-means clustering algorithm for mixed dataset [38], and K-harmonic means clustering algorithm for the mixed dataset (KHMCMD) [39], K-prototype (KP) clustering for mixed data1 [47], improved K-prototype (IKP) clustering algorithm for mixed data [48], SBAC [49], and Ji *et al.* [50] The experiments are performed in 50 iterations. In every 10 iterations, the number of correct predictions of data points and corresponding class labels obtained by the proposed algorithm are computed in terms of average accuracy (*AC*). Finally, among the obtained 5 results, the results with the maximum *AC* and minimum standard deviation (*Sd*) of most frequently selected active clusters is chosen as optimum cluster centers. The results are presented in Table 4. Table 4 demonstrates the comparative results of the *AC* and *Sd* among the proposed clustering algorithm and compared algorithms for the used real-life datasets. According to Table 4, for Heart Disease (1) dataset with 303 data points and having 5 numeric and 8 categorical attributes, the proposed clustering algorithm have achieved best *AC* followed by KHMCMD, KMCMD, IKP, Ji *et al.*, SBAC, and KP. The proposed clustering algorithm shows 6%, 0.7%, 2%, 3.8%, 9.4%, and 26.9 % more accurate results than KHMCMD, KMCMD, IKP, Ji *et al.*, SBAC, and KP, respectively. For Heart Disease (2) dataset with 270 datapoints and having 6 numeric and 8 categorical attributes, the proposed clustering algorithm have produced 1.2%, 2.06%, 17.5%, 28.2%, and 28.3% more accurate results than KHMCMD, KMCMD, IKP, KP, and SBAC, respectively. For the credit approval dataset with 690 datapoints and having 6 numeric and 8 categorical attributes, the proposed clustering have achieved 1%, 3.9%, 8.3%, 30%, and 30.7% more accurate results than KHMCMD, KMCMD, IKP, KP, and SBAC, respectively. For the Iris dataset with 150 data points and having 4 numeric attributes, the proposed clustering algorithm have achieved 9.8%, 10.1%, and 54.7% more accurate results than IKP, KP, and SBAC, respectively. For the soybean dataset with 47 data points and having 35 numeric attributes, the proposed clustering achieved 1%, 5.4 %, and 29.3% more accurate results than IKP, KP, and SBAC, respectively. The results in Table 4 demonstrate that for all 5 datasets, the *AC* of the proposed clustering algorithm is higher. In contrast, the *Sd* for the proposed clustering algorithm achieves a lower value as compared to the existing mixed data clustering algorithm. The comparative results of Table 4 are presented as a graph in Figure 2. The graph in Figure 2 shows the *AC* of the proposed clustering algorithm and compared algorithms for the used real-life datasets. The height of the bar represents the measured *AC* in such a manner that the longer the bar, the higher the *AC*. The *Sd* obtained for the corresponding dataset is mentioned at the top of the bar. Figure 2 represents that the proposed clustering algorithm achieved the highest *AC* as well as the lowest *Sd* for all the datasets. Thus, Figure 2 reveals that the proposed clustering algorithm is more robust to optimal cluster center selection.

Table 4 The performance comparison between the proposed clustering algorithm and the existing mixed data clustering algorithm in terms of average accuracy (*AC*) and standard deviation (*Sd*) (in brackets). ~ shows results not available. Bold value shows the best results.

	Proposed Algorithm		Improved K-prototype		K-Prototype		SBAC		KHMCMD		KMCMD		Ji *et al.* [50]	
Dataset	***AC***	***Sd***	***AC***	***Sd***	***AC***	***Sd***	***AC***	***Sd***	***AC***	***Sd***	***AC***	***Sd***	***AC***	***Sd***
Heart Disease (1)	0.846	0.14	0.826	~	0.577	~	0.752	~	0.840	0.15	0.8389	0.15	0.808	~
Heart Disease (2)	0.828	0.18	0.653	~	0.546	~	0.545	~	0.816	0.33	0.8074	1.20	~	~
Credit approval	0.862	0.11	0.779	~	0.562	~	0.555	~	0.852	0.38	0.8223	12.77	0.794	~
Iris	0.92	0.18	0.822	~	0.819	~	0.373	~	~	~	~	~	~	~
Soybean	0.91	0.16	0.90	~	0.856	~	0.617	~	~	~	~	~	~	~

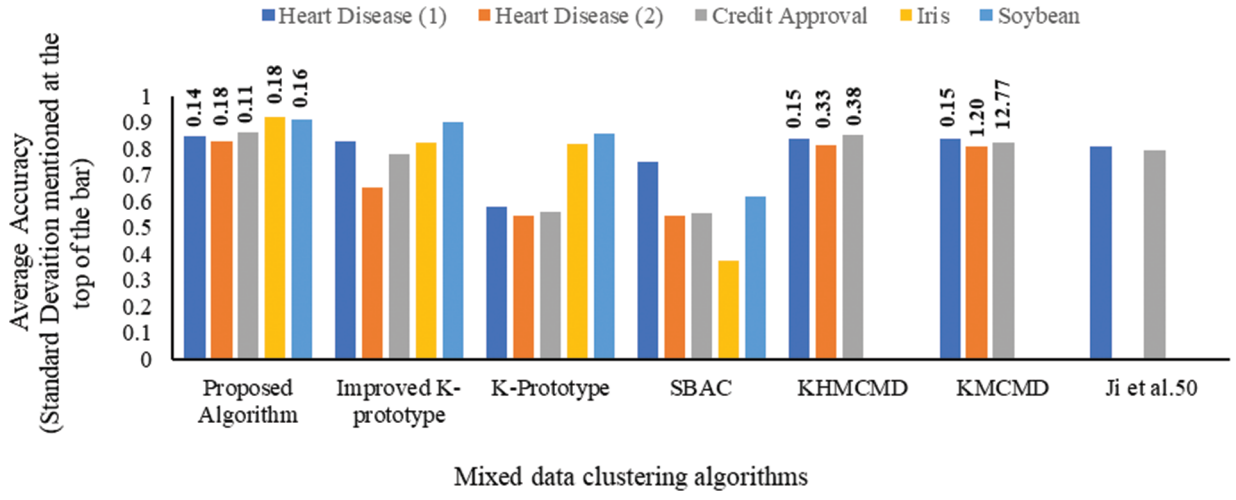


Figure 2 Comparison of clustering accuracy of proposed and existing clustering algorithms.

5.2. Application of the Proposed Clustering Algorithm to the Clustering of Postoperative Surgical Patients

Surgical patient clustering is defined as a process of arranging the patients in distinct groups based on their similarity of characteristics. These characteristics include age, gender, body mass index (BMI), American Society of Anesthesiologists (ASA) fitness grade, etc. For multi-specialty hospitals, where an enormous number of patients receive their surgical care, it is quite a challenging task to manage the surgical records efficiently. An efficiently managed surgical records help hospitals to improve patients care and monitoring to enhance the efficiency of resources within the hospital. In this paper, the surgical record management procedure (SRMP) of a multi-specialty hospital, India is examined, and the proposed algorithm is implemented on the hospital's existing SRMP to cluster surgical records and enhance existing SRMP.

5.2.1. Dataset description

The historical postoperative surgical mixed dataset is obtained from Shri Mahant Indiresh Hospital, Dehradun, India. The description of the datasets is given in Table 5.

5.2.2. Significance of active clusters selected by the proposed clustering algorithm

During each iteration of the algorithm, the significance of the obtained clusters are computed using two parameters: (1) frequency of the number of active clusters selected and (2) average accuracy. [2]. The frequency of clusters (n_c) is computed as

$$F_Q\left(n_{c_i}\right) = \frac{S_{N_{c_i}}}{R_t} \tag{22}$$

where $S_{N_{c_i}}$ is the number of times a particular cluster is selected and R_t represents the total number of iterations. Since the surgical dataset has no class label, the average accuracy of the proposed clustering algorithm is measured based on the SD index. The average accuracy is computed as the inverse of the SD index, and the result with minimum SD index is considered as a highly accurate clustering outcome.

Table 5 Postoperative surgical dataset characteristics.

Attribute	Type	Description
Age	Numeric	Patient's age at the time of surgery
Gender	Categorical	Gender of patient
BMI	Numeric	Body mass index of patient
ASA fitness grade	Numeric	Patients physical status required by the anesthesiologist before surgery
Marital status	Categorical	Marital status of patients
Ethnicity	Categorical	Ethnicity of patient
Comorbidity	Numeric	Charlson comorbidity index
Type of surgery	Categorical	Classifies attempt of surgical procedure
Surgery duration	Numeric	Length of surgical procedure
Procedural code	Categorical	Primary procedure code
Diagnose code	Categorical	Primary diagnosis code
Surgery domain	Categorical	Classifies surgical procedure
Grade of surgery	Categorical	Classifies risk of surgical procedure to the life of the patient
Urgency of surgery	Categorical	Classify the schedule of a surgical patient
LOS	Numeric	Length of stay in hospital after surgery (in days)

5.2.3. Discussion

The experiments are carried out in 50 iterations, and the results obtained in the pair of 10-10 iterations are shown in Table 6. Table 6 demonstrates the frequency of selecting the active clusters and the average accuracy (*AC*) of the clustering outcome, determined by the proposed clustering algorithm. In each pair of iterations, the selection frequency of the active clusters and their corresponding SD index value is computed. The average accuracy (*AC*) and standard deviation (*Sd*) of the obtained clustering outcomes are computed using the obtained SD index value.

Finally, the active cluster with the maximum *AC* and the minimum *Sd* is selected as optimum cluster centers. According to Table 6, the proposed algorithm selects 6 optimal active clusters most frequently with *AC* and *Sd* of 0.77, 0.18, respectively, in the first 10 iterations. In the next 11-20 iterations, 4 optimal active clusters are frequently selected with *AC* and *Sd* of 0.81 and 0.23, respectively. During iterations 21-30, 6 optimal active clusters are selected frequently with *AC* and *Sd* of 0.86 and 0.14, respectively. In 31-40 iterations, the proposed clustering algorithm selects 6 optimal active clusters frequently with *AC* and *Sd* of 0.87, 0.12, respectively. In final 41-50 iterations, 5 optimal active clusters are selected frequently with *AC* and *Sd* of 0.82 and 0.21. The clustered visualization of results, demonstrated in Table 6, is presented in Figure 3. Figure 3 shows the scatter plot of most frequently selected active clusters during each pair of 10 iterations. Since the dataset has both mixed numeric and categorical attributes, the scatter plots are generated based on the distance between the data points and active cluster centers. Each cluster in the figure is represented by a cluster name, and a circle is used to distinguish between the obtained active clusters partitions. It is too noteworthy to mention here that the circle does not represent the shape of the any cluster. Figure 3 clearly shows that the proposed clustering algorithm has produced nonoverlapping and well-separated cluster partitions in every pair of iterations. Further, due to higher average accuracy and lower standard deviation, the clusters shown in Figure 3(d) are considered as optimal cluster partitions. The overall performance is summarized in Table 7 and Figure 4, which shows that the proposed clustering algorithm obtains 6 optimal active clusters with a frequency of 1.2 and an *AC* of 0.87 with *Sd* of 0.12.

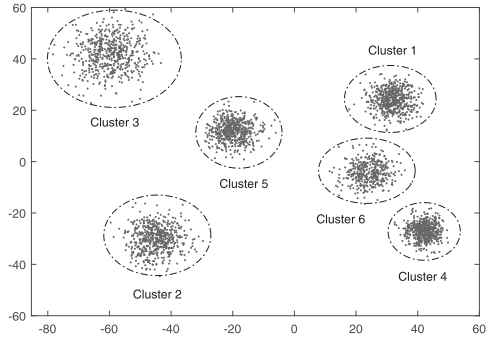
5.3. Statistical Analysis of the Proposed Clustering Algorithm

Although results based on average accuracy (*AC*) and standard deviation (*Sd*) from the Table 4 indicate that the proposed clustering algorithm outperformed compared algorithms, the statistical analysis is performed to validate the performance of the proposed

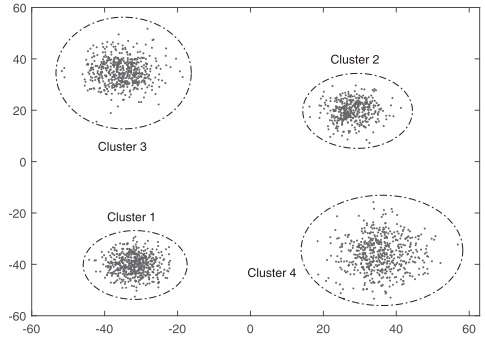
Table 6 Frequency (F_Q) and average accuracy (AC) (standard deviation (Sd) in brackets) of the clusters selected by the proposed clustering algorithm for the postoperative surgical mixed dataset.

Iteration		No of Active Cluster Extraction						
		2	**3**	**4**	**5**	**6**	**7**	**8**
1-10	F_Q	0.0	0.5	0.8	0.4	**1.4**	0.6	0.3
	AC	0.0	0.51 ($\pm$ 0.63)	0.62 ($\pm$ 0.42)	0.47 ($\pm$ 0.36)	**0.77 ($\pm$ 0.18)**	0.59 ($\pm$ 0.48)	0.44 ($\pm$ 0.44)
11-20	F_Q	0.4	0.5	**0.9**	0.6	0.6	0.8	0.2
	AC	0.46 ($\pm$ 0..39)	0.54 ($\pm$ 0.34)	**0.81 ($\pm$0.23)**	0.58 ($\pm$ 0.42)	0.69 ($\pm$ 0.24)	0.75 ($\pm$ 0.21)	0.33 ($\pm$ 0.56)
21-30	F_Q	0.0	0.0	0.8	0.9	**1.5**	0.2	0.6
	AC	0.0	0.0	0.68 ($\pm$ 0.35)	0.78 ($\pm$ 0.21)	**0.86 ($\pm$ 0.14)**	0.37($\pm$ 0.64)	0.51($\pm$ 0.60)
31-40	F_Q	0.3	0.0	0.9	1.0	**1.2**	0.0	0.6
	AC	0.32 ($\pm$ 0.69)	0.0	0.54 ($\pm$ 0.43)	0.62 ($\pm$ 0.31)	**0.87 ($\pm$ 0.12)**	0.0	0.41 ($\pm$ 0.48)
41-50	F_Q	0.0	0.8	0.8	**1.1**	0.6	0.7	–
	AC	0.0	0.46 ($\pm$ 0.36)	0.53 ($\pm$ 0.62)	**0.82 ($\pm$ 0.21)**	0.39 ($\pm$ 0.46)	0.48 ($\pm$ 0.54)	–

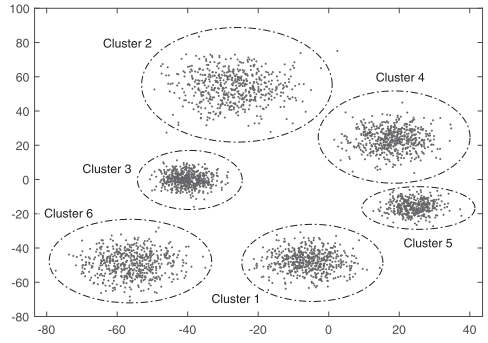
The Numerical bold values represent the best results of Frequency (FQ), Average accuracy (AC), and Standard deviation (Sd) of the active clusters selected by the proposed clustering algorithm in each pair of 10-10 iterations.
Symbolic bold values FQ and AC represent Frequency and Average accuracy, respectively.



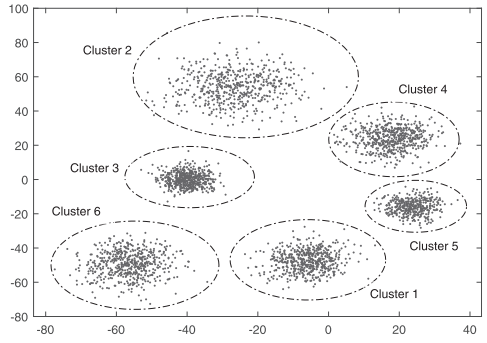
(a) Iteration (1-10): Cluster = 6, F_Q = 1.4, AC = 0.77, Sd = 0.18



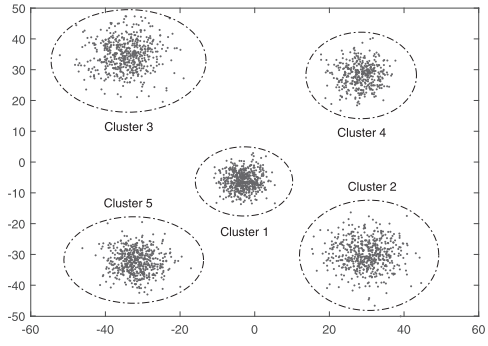
(b) Iteration (11-20): Cluster = 4, F_Q = 0.9, AC = 0.81, Sd = 0.23



(c) Iteration (21-30): Cluster = 6, F_Q = 1.5, AC = 0.86, Sd = 0.14



(d) Iteration (31-40): Cluster = 6, F_Q = 1.2, AC = 0.87, Sd = 0.12



(e) Iteration (41-50): Cluster = 5, F_Q = 1.1, AC = 0.82, Sd = 0.21

Figure 3 Cluster center selection frequency (F_Q), average accuracy (A_C) and standard deviation (S_d) for the postoperative surgical mixed dataset.

clustering algorithm. The statistical analysis helps in determining whether the results obtained by the proposed clustering algorithm are significant (P-value < 0.05) or not. In recent years, several statistical analysis methods [51] are used to compare and validate the performance of clustering algorithms. Since this paper aims to find optimal cluster centroids (means) that are used to determine the accuracy and standard deviation of the proposed clustering algorithm, a statistical analysis method based on "mean"' knows as unpaired t-test [13], is used to validate the proposed clustering algorithm. The unpaired t-test compares the means of results produced by the proposed clustering algorithm and the second-best performing clustering algorithm (KHMCMD) of Table 4. The results obtained in Table 8 demonstrate that for all datasets, P-value satisfies the significance level (< 0.05), which shows the clustering results produced by the proposed clustering algorithm are statistically significant than the other existing clustering algorithm.

6. CONCLUSION AND FUTURE WORK

In this paper, AEFA is used to identify the number of clusters and cluster centers automatically. The proposed approach utilizes threshold setting and cut-off value to select and refine cluster centers. The assignment of data points to different clusters is made based on a distance measure, i.e., Euclidean distance for numeric attributes and the probability of co-occurrence of value for categorical attributes. The proposed approach requires no prior specification of the number of clusters. The proposed approach is compared with existing mixed data clustering algorithms based on robustness measures and statistical testing. The efficacy of the proposed approach is also validated by clustering real historical post-operative surgical patients dataset obtained from a multispecialty hospital. Experimental results evident that the proposed approach is not only able to automatically find the number of clusters, but also the clustering results are more robust and better than the existing clustering techniques. The clusters produced by the proposed approach are also compact and well separated. In the future, the proposed work can be enhanced in various directions: one direction can be solving the complex high dimensional clustering problems by hybridization of the proposed algorithm with other meta-heuristic approaches. Moreover, since the proposed algorithm automatically detects the optimum number of clusters, one can explore other heuristics to improve the optimal selection of clusters. The proposed algorithm can also be used in ensemble clustering.

Table 7 | Performance of the proposed clustering algorithm on the postoperative surgical dataset.

Parameters	Value
No of active cluster obtained	6.0
Frequency of active cluster selection (F_Q)	1.2
Average accuracy (AC)	0.87
Standard deviation (Sd)	$\pm$0.12

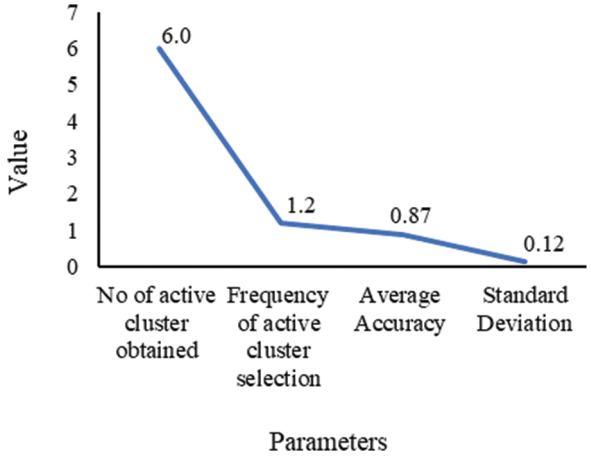


Figure 4 | Cluster quality measures for the proposed algorithm for the surgical dataset.

Table 8 | Unpaired t-test between the proposed and the second-best performing clustering algorithm for real-life datasets.

Dataset	Standard Error	t	95% Confidence Interval	Two-tailed P-value	Significance
Heart Disease (1)	1.000	10.0000	−12.306 to −7.693	0.0001	Highly significant
Heart Disease (2)	1.811	13.0309	−27.78 to −19.42	0.0001	Highly significant
Credit approval	1.00	5.0	−7.306 to −2.693	0.0011	Significant
Iris	1.87	4.06	−11.914 to −3.285	0.0036	Significant
Soybean	1.522	3.8335	−9.28 to −2.39	0.0040	Significant
Postoperative surgical dataset	1.709	12.9450	−9.0725 to −0.9916	0.000032	Highly significant

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHORS' CONTRIBUTIONS

Conceptualization, Methodology, Formal analysis, Validation, Writing-Original Draft Preparation by Hemant Petwal and Supervision by Dr. Rinkle Rani. All authors read and approved the manuscript.

Funding Statement

This research received no external funding.

REFERENCES

[1] I.E. Evangelou, D.G. Hadjimitsis, A.A. Lazakidou, C. Clayton, Data mining and knowledge discovery in complex image data using artificial neural networks, in Proceeding ofWorkshop Complex Reason Geogr Data, Everitt, Paphos, Cyprus, 2001. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.18.6791&rep=rep1&type=pdf

[2] B.S. Everitt, Cluster Analysis, third ed., Edward Arnold / Halsted Press, London, England, 1993.

[3] M.R. Rao, Cluster analysis and mathematical programming, J. Am. Stat. Assoc. 66 (1971), 622–626.

[4] A.J. Graaff, A.P. Engelbrecht, Using sequential deviation to dynamically determine the number of clusters found by a local network neighbourhood artificial immune system, Appl. Soft Comput. 11 (2011), 2698–2713.

[5] H. Frigui, R. Krishnapuram, A robust competitive clustering algorithm with applications in computer vision, IEEE Trans. Pattern Anal. Mach. Intell. 21 (1999), 450–465.

[6] Y. Leung, J.S. Zhang, Z.B. Xu, Clustering by scale-space filtering, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000), 1396–1410.

[7] Y. Liu, X. Wu, Y. Shen, Automatic clustering using genetic algorithms, Appl. Math. Comput. 218 (2011), 1267–1279.

[8] Z. Aliniya, S.A. Mirroshandel, A novel combinatorial merge-split approach for automatic clustering using imperialist competitive algorithm, Expert Syst. Appl. 117 (2019), 243–266.

[9] H. He, Y. Tan, A two-stage genetic algorithm for automatic clustering, Neurocomputing. 81 (2012), 49–59.

[10] D. Doval, S. Mancoridis, B.S. Mitchell, Automatic clustering of software systems using a genetic algorithm, in Proceedings Ninth International Workshop Software Technology and Engineering Practice (STEP'99), IEEE, Pittsburgh, PA, USA, 1999, pp. 73–81.

[11] Z. Izakian, M.S. Mesgari, A. Abraham, Automated clustering of trajectory data using a particle swarm optimization, Comput. Environ. Urban Syst. 55 (2016), 55–65.

[12] S. Das, A. Abraham, A. Konar, Automatic kernel clustering with a multi-elitist particle swarm optimization algorithm, Pattern Recognit. Lett. 29 (2008), 688–699.

[13] V. Kumar, J.K. Chhabra, D. Kumar, Automatic cluster evolution using gravitational search algorithm and its application on image segmentation, Eng. Appl. Artif. Intell. 29 (2014), 93–103.

[14] V. Kumar, D. Kumar, Automatic clustering and feature selection using gravitational search algorithm and its application to microarray data analysis, Neural Comput. Appl. 31 (2019), 3647–3663.

[15] S. Das, A. Abraham, A. Konar, Automatic clustering using an improved differential evolution algorithm, IEEE Trans. Syst. Man Cybern. Part A Syst. Hum. 38 (2007), 218–237.

[16] S. Das, A. Chowdhury, A. Abraham, A bacterial evolutionary algorithm for automatic data clustering, in 2009 IEEE Congress on Evolutionary Computation, IEEE, Trondheim, Norway, 2009, pp. 2403–2410.

[17] R.J. Kuo, Y.D. Huang, C.C. Lin, Y.H. Wu, F.E. Zulvia, Automatic kernel clustering with bee colony optimization algorithm, Inf. Sci. 283 (2014), 107–122.

[18] Z.G. Su, P.H. Wang, J. Shen, Y.G. Li, Y.F. Zhang, E.J. Hu, Automatic fuzzy partitioning approach using Variable string length Artificial Bee Colony (VABC) algorithm, Appl. Soft Comput. 12 (2012), 3421–3441.

[19] A. José-García, W. Gómez-Flores, Automatic clustering using nature-inspired metaheuristics: asurvey, Appl. Soft Comput. 41 (2016), 192–213.

[20] A. Yadav, AEFA: artificial electric field algorithm for global optimization, Swarm Evol. Comput. 48 (2019), 93–108.

[21] M. Halkidi, M. Vazirgiannis, Y. Batistakis, Quality scheme assessment in the clustering process, in: D.A. Zighed, J. Komorowski, J. Żytkow (Eds.), Principles of Data Mining and Knowledge Discovery, European Conference on Principles of Data Mining and Knowledge Discovery, Springer, Berlin, Heidelberg, 2000, pp. 265–276.

[22] A.K. Jain, R.C. Dubes, Algorithms for Clustering Data, Prentice-Hall, Inc., Upper Saddle River, New Jersey, 1988. https://www.cse.msu.edu/~jain/Clustering_Jain_Dubes.pdf

[23] J. MacQueen, Some methods for classification and analysis of multivariate observations, in Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, California, USA, 1967, vol. 1, pp. 281–297. https://projecteuclid.org/download/pdf_1/euclid.bsmsp/1200512992

[24] S.U. Mane, P.G. Gaikwad, Hybrid particle swarm optimization (HPSO) for data clustering, Int. J. Comput. Appl. 97 (2014), 1–5.

[25] J. Nasiri, F.M. Khiyabani, A whale optimization algorithm (WOA) approach for clustering, Cogent Math. Stat. 5 (2018), 1483565.

[26] D.W. Van der Merwe, A.P. Engelbrecht, Data clustering using particle swarm optimization, in The 2003 Congress on Evolutionary Computation, IEEE, Canberra, Australia, 2003, pp. 215–220.

[27] P. Berkhin, A survey of clustering data mining techniques, in: J. Kogan, C. Nicholas, M. Teboulle (Eds.), Grouping Multidimensional Data, Springer, Berlin, Heidelberg, 2006, pp. 25–71.

[28] T. Velmurugan, T. Santhanam, A survey of partition based clustering algorithms in data mining: an experimental approach, Inf. Technol. J. 10 (2011), 478–484.

[29] R.J. Kuo, H.S. Wang, T.L. Hu, S.H. Chou, Application of ant K-means on clustering analysis, Comput. Math. Appl. 50 (2005), 1709–1724.

[30] S. Äyrämö, T. Kärkkäinen, Introduction to Partitioning-based Clustering Methods with a Robust Example, Reports of the Department of Mathematical Information Technology, Series C, Software Engineering and Computational Intelligence, University of Jyväskylä, Jyväskylä, Finland, 2006. http://urn.fi/URN:ISBN:951-39-2467-X

[31] M. Sarkar, B. Yegnanarayana, D. Khemani, A clustering algorithm using an evolutionary programming-based approach, Pattern Recognit. Lett. 18 (1997), 975–986.

[32] Y. Chen, C. Tang, J. Zhu, C. Li, S. Qiao, R. Li, J. Wu, Clustering without prior knowledge based on gene expression programming, in Third International Conference on Natural Computation (ICNC 2007), IEEE, Haikou, China, 2007, pp. 451–455.
[33] J. Liu, Y. Chi, Z. Liu, S. He, Ensemble multi-objective evolutionary algorithm for gene regulatory network reconstruction based on fuzzy cognitive maps, CAAI Trans. Intell. Technol. 4 (2019), 24–36.
[34] C.Y. Lee, E.K. Antonsson, Dynamic partitional clustering using evolution strategies, in 2000 26th Annual Conference of the IEEE Industrial Electronics Society (IECON 2000), 2000 IEEE International Conference on Industrial Electronics, Control and Instrumentation, 21st Century Technologies, IEEE, Nagoya, Japan, 2000, vol. 4, pp. 2716–2721.
[35] L.Y. Tseng, S.B. Yang, A genetic approach to the automatic clustering problem, Pattern Recognit. 34 (2001), 415–424.
[36] S. Bandyopadhyay, U. Maulik, Nonparametric genetic clustering: comparison of validity indices, IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. 31 (2001), 120–125.
[37] S. Bandyopadhyay, S. Saha, A point symmetry-based clustering technique for automatic evolution of clusters, IEEE Trans. Knowl. Data Eng. 20 (2008), 1441–1457.
[38] A. Ahmad, L. Dey, A k-mean clustering algorithm for mixed numeric and categorical data, Data Knowl. Eng. 63 (2007), 503–527.
[39] A. Ahmad, S. Hashmi, K-Harmonic means type clustering algorithm for mixed datasets, Appl. Soft Comput. 48 (2016), 39–49.
[40] D.X. Chang, X.D. Zhang, C.W. Zheng, D.M. Zhang, A robust dynamic niching genetic algorithm with niche migration for automatic clustering problem, Pattern Recognit. 43 (2010), 1346–1360.
[41] S.M. Pan, K.S. Cheng, Evolution-based tabu search approach to automatic clustering, IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. 37 (2007), 827–838.
[42] M.G. Omran, A. Salman, A.P. Engelbrecht, Dynamic clustering using particle swarm optimization with application in image segmentation, Pattern Anal. Appl. 8 (2006), 332.
[43] B. Gupta, M. Tiwari, S.S. Lamba, Visibility improvement and mass segmentation of mammogram images using quantile separated histogram equalisation with local contrast enhancement, CAAI Trans. Intell. Technol. 4 (2019), 73–79.
[44] A. Almagbile, Estimation of crowd density from UAVs images based on corner detection procedures and clustering analysis, Geo-spatial Inf. Sci. 22 (2019), 23–34.
[45] T. Cura, A particle swarm optimization approach to clustering, Expert Syst. Appl. 39 (2012), 1582–1588.
[46] A. Chowdhury, S. Bose, S. Das, Automatic clustering based on invasive weed optimization algorithm, in: B.K. Panigrahi, P.N. Suganthan, S. Das, S.C. Satapathy (Eds.), Swarm, Evolutionary, and Memetic Computing, International Conference on Swarm, Evolutionary, and Memetic Computing, Springer, Berlin, Heidelberg, 2011, pp. 105–112.
[47] Z. Huang, Clustering large data sets with mixed numeric and categorical values, in Proceedings Of 1st Pacific-Asia Conference on Knowledge Discouvery And Data Mining, Singapore, 1997. https://pdfs.semanticscholar.org/d42b/b5ad2d03be6d8fefa63d25d02c0711d19728.pdf
[48] J. Ji, T. Bai, C. Zhou, C. Ma, Z. Wang, An improved k-prototypes clustering algorithm for mixed numeric and categorical data, Neurocomputing. 120 (2013), 590–596.
[49] C. Li, G. Biswas, Unsupervised learning with mixed numeric and nominal data, IEEE Trans. Knowl. Data Eng. 14 (2002), 673–690.
[50] J. Ji, W. Pang, Y. Zheng, Z. Wang, Z. Ma, An initialization method for clustering mixed numeric and categorical data based on the density and distance, Int. J. Pattern Recognit. Artif. Intell. 29 (2015), 1550024.
[51] B. Desgraupes, Clustering Indices, University of Paris Ouest-Lab Modal'X, France, 2013, pp. 1–34. https://cran.biodisk.org/web/packages/clusterCrit/vignettes/clusterCrit.pdf