Scientia Iranica D (2023) 30(1), 104–115

Sharif University of Technology
Scientia Iranica
Transactions D: Computer Science & Engineering and Electrical Engineering
http://scientiairanica.sharif.edu

A semi-supervised clustering approach using labeled data

A. Taghizabet[a], J. Tanha[b], A. Amini[a,*], and J. Mohammadzadeh[a]

a. *Department of Computer Engineering, Karaj Branch, Islamic Azad University, Karaj, Iran.*
b. *Computer and Electrical Engineering Department, University of Tabriz, Tabriz, Iran.*

Received 15 June 2021; received in revised form 19 June 2022; accepted 19 September 2022

KEYWORDS
Semi-supervised;
Clustering;
Label-based clustering;
Semi-supervised learning;
Convex hull.

Abstract. Over recent decades, there has been a growing interest in the use of semi-supervised clustering. Compared to the supervised or unsupervised clustering methods for solving different real-life problems, the review of relevant articles shows that semi-supervised clustering methods are more powerful, and even a small amount of supervised information can significantly improve the results of unsupervised methods. One popular method for incorporating partial supervised information is the use of labeled data. In this study, a semi-supervised clustering algorithm called ConvexClust is proposed. The proposed method improves data clustering using a geometric view borrowed from the Lune concept in the connectivity index and 10% of labeled data. Use of labeled data and formation of a convex hull are the beginning steps toward clustering. Next, labeling of non-labeled data and updating of the convex hull in an iterative process are the next steps. Evaluations of three UCI datasets and sixteen artificial datasets indicate that the proposed method outperforms other semi-supervised and traditional clustering techniques.

1. Introduction

Clustering is known as one of the most important unsupervised learning methods in the machine learning field, widely used for all kinds of practical events like image segmentation [1,2], identification and analysis of optimal faces in seismic datasets [3], character recognition [4,5], network security issues [6–8], clustering of sensor nodes [9–12], intrusion detection [13], blind channel equalizer design [13], human action classification [14], document clustering [15], tourism market segmentation [13], analysis of gene expression patterns [16,17], etc. as pre-processing techniques. The target of clustering is to categorize unlabeled samples into multiple classes based on the similarity between them, and these classes are often called clusters [18]. Supervised learning methods are more time-consuming and cost-sensitive due to the need for the sufficient extent of prior knowledge and its lack in the real world [19]. In contrast, unsupervised learning methods work without prior information, but they suffer from local trap problems since clustering results are trapped into a local optimum, hence undesirable clustering results. To overcome these problems, semi-supervised methods are employed to improve the quality and performance of clustering methods with a small amount of prior knowledge. These new methods are considered by researchers because they are more practical than the methods mentioned above for resolving real problems.

*. *Corresponding author.*
E-mail addresses: a.taghizabet@gmail.com (A. Taghizabet); tanha@tabrizu.ac.ir (J. Tanha); aamini@kiau.ac.ir (A. Amini); j.mohammadzadeh@kiau.ac.ir (J. Mohammadzadeh)

doi: 10.24200/sci.2022.58519.5772

Generally, semi-supervised clustering algorithms fall into three categories: distance-based, constraint-based, and combined approaches [20]. In the first approach, often, an existing clustering method is applied, but the distance measure of the method is adopted based on prior knowledge. The distance criterion is adjusted such that the distance of data points in the same clusters (Must-Link constraints: ML) is reduced, while the same distance in different clusters (Cannot-Link constraints: CL) increases [21]. In other words, an adjusted distance measure is parameterized, and the parameters are detected based on prior supervision information in the form of constraints such as the above-mentioned constraints (ML and CL) [22]. However, in the distance-based method, the modified distance measure might not work accurately, e.g., two instances associated with a Must-Link are still far away from each other and, thus, separated into different clusters. Some studies that applied this method for solving clustering problems include [23–25] and [26].

Constraint-based approaches modify the existing clustering algorithm so that prior knowledge can be labeled data or constraints to guide the algorithms for better clustering results. This was done by modifying the objective function of the clustering algorithm in different ways. Constrained COBWEB [27] embeds the constraints into the incremental partitioning process by optimizing its clustering objective. Seeded K-means [28] is the result of incorporating prior knowledge of labeled data into only the initialization step of the conventional K-means algorithm. At the same time, constrained K-means [28] is the result of combining prior knowledge in both initialization and assignment steps of the K-means algorithm. Combined methods benefit from both approaches, i.e., distance-based and constraint-based.

In the case of traditional constraint-based semi-supervised clustering methods that use labeled data, such as seeded K-means and constrained K-means, these data are employed to guide the algorithm and this guidance involves the use of labeled data in either the initialization step, during the learning process, or both. In the case of seeded K-means, labeled data is used only for the initialization step. In constrained K-means, which is the basis for the development of the proposed method, labeled data are used in both the initialization step and learning process step. The proposed approach uses the labeled data more effectively to guide the algorithm both in the initialization stage and at the learning phase. In the traditional methods mentioned above, only the distance criterion is used to assign unlabeled data to clusters. In contrast, the proposed method uses a more efficient assignment by presenting a new objective function in which, in addition to the distance factor, data density is also considered implicitly.

In this work, the objective is to develop a novel semi-supervised clustering algorithm that functions based on the constrained K-means algorithm. It not only is easy to implement, but also improves the performance of the clustering process. The contributions of the current paper are as follows:

- More efficient use of labeled data;
- Introducing a new objective function that considers both the distance criterion and data density (implicitly) to assign unlabeled data to clusters;
- Easy to implement;
- A new perspective on the clustering process (geometric viewpoint).

The rest of this paper is organized as follows. Section 2 describes the semi-supervised clustering approaches related to our method. Section 3 presents the proposed method. Section 4 gives experimental results. Finally, Section 5 conclude this paper.

2. Related works

2.1. *K*-means

K-means algorithm, which was proposed by Macqueen in 1967, is one of the simplest unsupervised learning algorithms. The main idea is to find K groups in a given dataset $X = \{x_1, x_2, ..., x_n\}$. The algorithm starts with initial estimates for the K cluster centers (each center represents one cluster), which can be randomly generated or randomly selected from the dataset. The algorithm then iterates between two steps: data assignment and center update.

In the first step, each data point is assigned to its nearest center based on the squared Euclidean distance, and in the second step, data points for the centers are recomputed. This is done by taking the mean value of all data points assigned to the first cluster center. The algorithm iterates between these steps until a stopping criterion is met. The basic steps of the K-means algorithm are given below.

Algorithm 1: K-means

1. **Initialization:** Begin with initial cluster centers $\mu_j, j = 1, 2, ..., k$ in the given dataset $X = \{x_1, x_2, ..., x_n\}, \forall x_i \in R^m$.
2. **Repeat**
3. **Data assignment:** Assign each data point x_i to the closest cluster j^* and let $x_i \in c_{j^*}$, where $j^* = \arg\min_j ||x_i - \mu_j||$.
4. **Center update:** Update the cluster centers by averaging the data points assigned to each of them, i.e., $x_i = \sum_{x_i \in c_j} \frac{x_i}{|c_j|}$.

5. **Until** convergence is achieved.
6. **Return** clustering result $\{c_1, c_2, ..., c_k\}$ of X.

2.2. Seeded K-means (S-Kmeans)

In this algorithm, a subset L of labeled data of the original dataset is used to guide the clustering process through initialization [28]. The L is composed of K groups of labeled data of each cluster. Thus, rather than initializing K-means with K random means, the mean value of the ith cluster is initialized with the mean value of the ith partition of labeled data in L. The values for these groups of labeled data are merely used for the initialization step rather than in other stages of the algorithm. The S-Kmeans phases are given in Algorithm 2.

Algorithm 2: S-Kmeans

1. **Input:** K = number of clusters, dataset $X = \{x_1, x_2, ..., x_n\}, \forall x_i \in R^m, L = \{l_1, l_2, ..., l_k\}$, and l_i set of labeled data in the ith cluster.
2. **Output:** clustering result $\{c_1, c_2, ..., c_k\}$ of X.
3. **Initialization:** $\mu_j = \sum_{x_i \in l_j} \frac{x_i}{|l_j|}, j = 1, 2, ..., k$ in the given dataset $X = \{x_1, x_2, ..., x_n\}$, $\forall x_i \in R^m$.
4. **Repeat**
5. **Data assignment:** Assign each data point x_i to the closest cluster j^* and let $x_i \in c_{i^*}$, where $j^* = \arg\min_j ||x_i - \mu_j||$.
6. **Center update:** Update the cluster centers by averaging the data points assigned to each of them, i.e., $x_i = \sum_{x_i \in c_j} \frac{x_i}{|c_j|}$.
7. **Until** convergence is achieved.
8. **Return** clustering result $\{c_1, c_2, ..., c_k\}$ of X.

2.3. Constrained K-means (C-Kmeans)

In the constrained K-means, the set L is used for initialization as described for the seeded K-means algorithm. However, in the data assignment step of the algorithm, the reassignment of those labeled data is not performed. More precisely, labeled data are kept unchanged in constrained K-means, while the other data are grouped the same as the K-means algorithm. The steps of the C-Kmeans are shown in Algorithm 3.

Algorithm 3: C-Kmeans

1. **Input:** K = number of clusters, dataset $X = \{x_1, x_2, ..., x_n\}, \forall x_i \in R^m, L = \{l_1, l_2, ..., l_k\}$, and l_i set of labeled data in the ith cluster.
2. **Output:** clustering result $\{c_1, c_2, ..., c_k\}$ of X.
3. **Initialization:** $\mu_j = \sum_{x_i \in l_j} \frac{x_i}{|l_j|}, j = 1, 2, ..., k$ in the given dataset $X = \{x_1, x_2, ..., x_n\}, \forall x_i \in R^m$.
4. **Repeat**
5. **Data assignment:** Assign each data point $x_i \notin L$ to the closest cluster j^* and let $x_i \in c_{j^*}$, where $j^* = \arg\min_j ||x_i - \mu_j||$; otherwise, assign x_i to the cluster to which it belongs.
6. **Center update:** Update the cluster centers by averaging the data points assigned to each of them, i.e., $x_i = \sum_{x_i \in c_j} \frac{x_i}{|c_j|}$.
7. **Until** convergence is achieved.
8. **Return** clustering result $\{c_1, c_2, ..., c_k\}$ of X.

2.4. Fuzzy C-Means (FCM)

FCM is a method of clustering that allows a data point to belong to two or more clusters. It is based on the minimization of the objective function as Eq. (1):

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^m ||x_i - c_j||, \quad 1 \leq m \leq \infty. \tag{1}$$

At each iteration of the algorithm, the value of J is minimized. N is the number of data points, C the number of clusters, c_j the cluster center of cluster j, and m (the fuzziness coefficient) is any real number greater than one. This value determines how much the clusters can overlap with one another. The higher the value of m, the larger the overlap between clusters, u_{ij} is the degree of membership of data point x_i to cluster j, and $|| * ||$ is any norm expressing the similarity (or closeness) between any measured data x_i and the center c_j. At each iteration, cluster centers are measured as the weighted average of the data points, where the weights are given by the degrees of membership. Fuzzy clustering is carried out through an iterative optimization of the objective function shown above by updating the degree of membership u_{ij} and the cluster center c_j as in Eqs. (2) and (3):

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{||x_i - c_j||}{||x_i - c_k||}\right)^{\frac{2}{m-1}}}, \tag{2}$$

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^m x_i}{\sum_{i=1}^{N} u_{ij}^m}. \tag{3}$$

The required accuracy of the membership degree determines the number of iterations completed by the FCM algorithm. Iterations will stop when convergence is achieved, which means the membership degrees between two iterations are no more than a predefined threshold. The steps of the algorithm are given below:

Algorithm 4: FCM

1. **Initialize** $U = [u_{ij}]$ randomly.

2. **Repeat**

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^m x_i}{\sum_{i=1}^{N} u_{ij}^m}.$$

3. Update cluster centers $C^{(k)} = [c_j]$ by $U^{(k)}$:

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^m x_i}{\sum_{i=1}^{N} u_{ij}^m}.$$

4. Update $U^{(k)}$:

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{||x_i - c_j||}{||x_i - c_k||}\right)^{\frac{2}{m-1}}}.$$

5. **Until** convergence is achieved.

2.5. Fuzzy G-K and fuzzy G-G

The fuzzy G-K algorithm is a developed version of FCM, aims to solve the FCM problem of identifying clusters of different sizes, and requires the following steps to automatically estimate the number of clusters [29]:

1. Calculation of the initial cluster centers;
2. Covariance matrix estimation for each cluster;
3. Computation of the distance using the covariance matrix found in the previous step to evaluate the resulting clusters according to the XB measure having the initial centers of the cluster and the membership function;
4. Calculation of the new membership matrix; and
5. Repeating of the algorithm from step 1 until convergence be achieved.

The fuzzy G-G algorithm was introduced after the fuzzy G-K algorithm and created to solve the FCM problem in identifying clusters of different sizes and shapes. The algorithm automatically estimates the number of clusters in a two-layer structure. In the first layer, a modified fuzzy K-means algorithm is implemented by clustering the data without an initial guess of the cluster centers, while in the second layer, clustering is done again with the centers obtained from the previous step to provide an optimal fuzzy result; then, the clustering performance is calculated and finally, the number of clusters increases. This process is repeated until an optimal number of clusters be achieved according to the clustering performance value [30].

3. Proposed method

In traditional semi-supervised clustering methods, such as seeded K-means, data are used in the initialization stage of the algorithm (determination of primary cluster centers). It does not use labeled data in the algorithm learning process, but the constrained K-means algorithm, which is the basis of the proposed method, operates in such a way that it uses labeled data in the initialization step, and with the evolution of the algorithm, the labeled data remain unchanged. For the unlabeled data, the constrained K-means algorithm functions similarly to the traditional K-means algorithm. Therefore, according to Figure 1, the constrained K-means algorithm was decided to be developed such that labeled data in the initialization and learning steps be used more effectively while assigning unlabeled data to clusters more carefully.

This section describes the proposed semi-supervised clustering algorithm, ConvexClust, in detail. The objective is to present a semi-supervised clustering algorithm that divides the dataset into K clusters.

3.1. Objective function

The objective function of the proposed algorithm has been inspired by the concept of Lune used in connectivity-based cluster validity indices in [31]. This objective function implicitly considers two main clustering objectives: cluster compactness and separation.

Before elaborating on the application of the proposed method, it is necessary to explain the convex concept. In a two-dimensional space, convex is a polygon similar to Figure 2a. For any two vertices or any two points inside of it, the connecting line between two vertices or two points is also inside the polygon region and Figure 2b is not a convex polygon.

The convex hull of a set of points S, denoted by $CH(S)$, is the smallest convex polygon P for which each point in S is either on the boundary or interior of P [32].

3.2. Detail of the ConvexClust clustering algorithm

The algorithm works based on creating a convex hull (the smallest convex set) of labeled data in each cluster. First, K groups are formed using the labeled data of each part. These groups are K convex hulls built by labeled data. Then, unlabeled data enclosed in each convex hull are marked with the label of those given data in the construction of their convex hulls. Next, to determine the labels of other unlabeled data, the following iterative process is implemented: First, the mean of the data within each convex hull is obtained. Then, the average distance of the unlabeled data to the mean center of each convex hull is obtained (r). Then the number of labeled data of each cluster, placed within a circle C (an unlabeled data, r) is counted separately. Next, the unlabeled data is assigned to

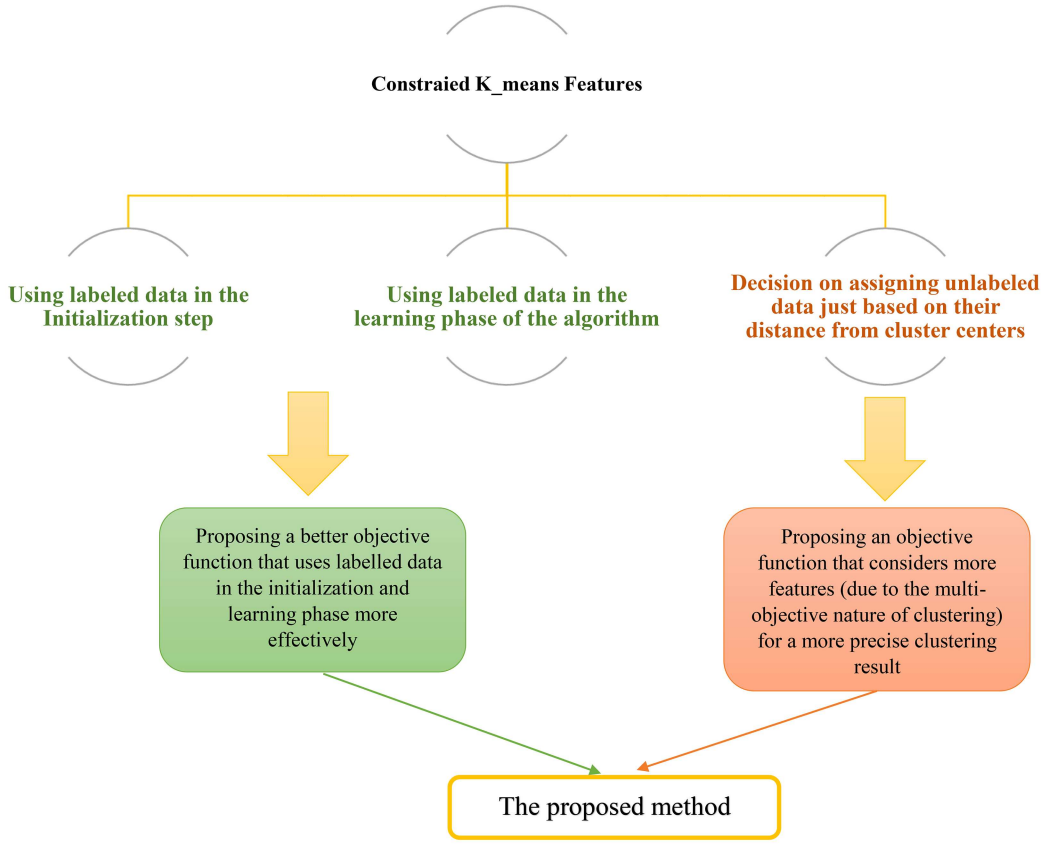

Figure 1. A general overview of the process of proposing a new method.

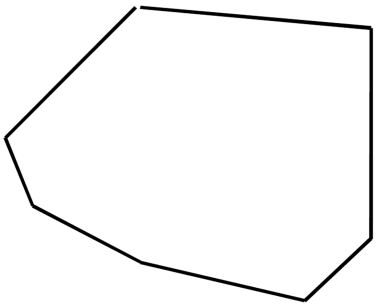

Figure 2a. A convex polygon.

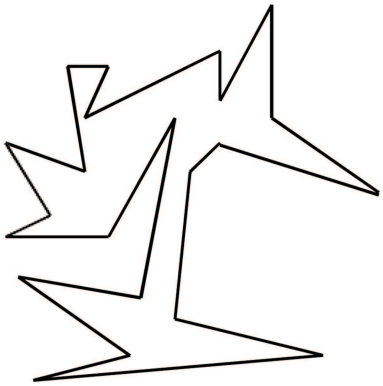

Figure 2b. A non-convex polygon.

the convex, which has the largest amount of labeled data among the aforementioned convex hulls. Finally, the convex hull is renewed with new labeled data and its center is updated. These steps are repeated until no unlabeled data is left. The pseudo code of the ConvexClust algorithm is shown in Algorithm 5.

An example of the proposed method in a two-dimensional space is as follows: Assuming that the given dataset for clustering is given in Figure 3a and the labeled data of each cluster are marked in red points. Then, through steps 3 to 6 of the algorithm, the convex hulls (red lines) are developed and the data inside each convex hull take the label of that convex hull (Figure 3b). After that, though step 7 of the algorithm, the centers of each convex hull are calculated (by calculating the mean of data points in each convex hull). Then, in step 8, for each unlabeled data point (e.g., A), the following actions are taken: First, the distance between the unlabeled data A and each convex hull center is calculated and their average is considered as $(r = (\frac{r_1+r_2+r_3}{3}))$. Then, in this space, the circle intersects with A center and r radius and convex hulls are developed in each cluster (Figure 3c). The number of the labeled data is counted and A is assigned to the convex hull, which has the largest amount of labeled data, and to cluster 2 in this example.

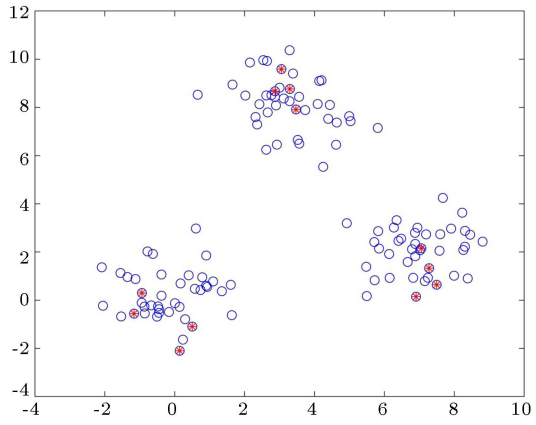

Figure 3a. Given dataset for clustering.

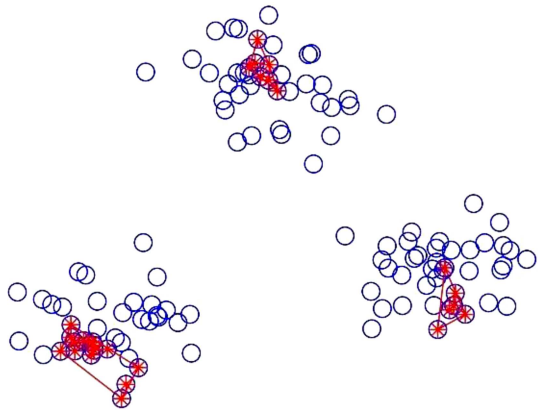

Figure 3b. Dataset corresponding to steps 3 to 6 of the ConvexClust algorithm.

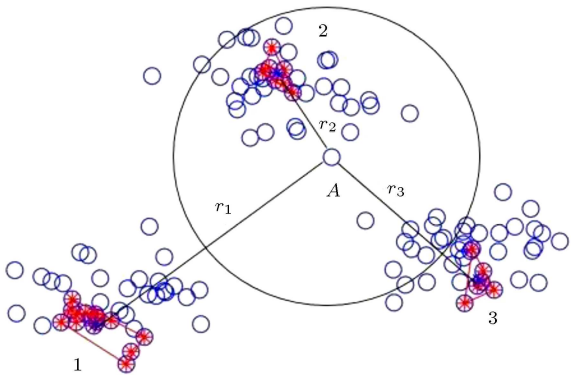

Figure 3c. The space of the intersection of the circle with A center and r radius and the developed convex hulls in each cluster.

Radius r gives rise to a larger shared space between the circle and the cluster to which unlabeled data probably belongs. The radius of the circle to the unlabeled data center is equal to the average distance of that point to the center of the primary clusters. With this method, the distance of each unlabeled data becomes effective in assigning it to its corresponding cluster. On the other hand, according to the proposed method, there may be a situation where some of the labeled data of all the clusters are placed in the circle around that unlabeled data but that unlabeled data does not belong to all those clusters. It belongs to a cluster that has a higher data density. In this way, in the crossing area of the circle with that a growing convex hull, we probably have more labeled data. Therefore, cluster compression and cluster dispersion concepts are realized. Considering these explanations, the objective function is defined as Eq. (4):

$$U_i \in CH_j(L_j) | \arg\max_j(NumberOfDataPoints (circle(U_i, R) \cap CH_j(L_j))), \quad (4)$$

where U_i is any unlabeled data in the dataset, *convex hull*$_j$ (L_j) is any developing convex hull (cluster) with labeled data L_j, and *circle* (U_i, R) is a circle with U_i center and R radius in the way that R equals the distance mean value of U_i to all convex hull centers.

Algorithm 5: Our method

1. Initialize *nCluster*: number of clusters, L: labeled data (ten percent of each cluster), and U: unlabeled data.
2. Perform feature selection if needed (in our experiment, when data dimensions are equal to or more than 4).
3. **For** $i = 1$: *nClusrer*
4. Calculate the convex hull of each group of labeled data.
5. Mark unlabeled data enclosed in each convex hull with the label of those given data in the construction of their convex hull.
6. **End.**
7. Calculate cluster centers by the mean value of the data in each convex hull ($M(j)$, $j = 1$: *nCluster*).
8. **For Each U**
9. Calculate the distance mean value of U from $M(j)$s, called r.
10. Count labeled data enclosed between each convex hull and a circle with U center and r radius.
11. Assign U to the cluster having the largest amount of labeled data.
12. If there is more than one cluster assigning U to them, assign it to the cluster, which has a shorter distance from its center to U.
13. Add U to the intended cluster, develop the related convex hull with existing labeled data, and mark the enclosed unlabeled data in that convex hull.

Table 1. Dataset characteristics.

Dataset	Number of instances (n)	Number of dimensions (d)	Number of clusters (k)
Banknote authentication	1372	4	2
Heart1302	270	13	2
Liver0602	345	6	2
2d-10c-no0	2972	2	10
2d-10c-no1	2525	2	10
2d-10c-no2	3073	2	10
2d-10c-no3	3359	2	10
2d-10c-no4	3291	2	10
2d-10c-no5	3630	2	10
2d-10c-no6	3408	2	10
2d-10c-no8	2830	2	10
2d-4c-no1	1623	2	4
2d-4c-no2	1064	2	4
2d-4c-no3	1123	2	4
2d-4c-no4	863	2	4
2d-4c-no5	1638	2	4
2d-4c-no6	1670	2	4
2d-4c-no7	1028	2	4
2d-4c-no9	876	2	4

14. Update the cluster center of the developed convex hull.
15. **End.**

From the assignment point of view, the minimum Euclidean distance has been considered. A particular point P (Eqs. (5) and (6)) is allocated to the cluster where it has a minimum Euclidean distance from it.

$$P \in i | \arg\min_i \{d_e(C_i, P)\}, \tag{5}$$

$$i = 1...k. \tag{6}$$

C_i is the center of the ith cluster and d_e denotes the Euclidean distance.

4. Experimental results

This section describes the used datasets, including both real and artificial sets, and the measures taken to test the validity of the proposed algorithm. We have compared the proposed algorithm (ConvexClust) with six other algorithms including conventional K-means, CK-means [28], seeded K-means [28], FCM [33], fuzzy G-G [30], and fuzzy Gk [29].

4.1. Dataset and experimental setting

The performance of ConvexClust is evaluated using sixteen artificial and three UCI datasets. Details of these datasets are shown in Table 1. The synthetic datasets were generated using the software provided by Julia Handl [34], and the configurations of datasets are presented as follows:

- Xd-Xc-noX: 'd' indicates attributes, 'c' clusters, and 'no' the dataset number. For example, 2d-10c-no0 is a dataset with two attributes, ten clusters, and zero (the dataset number zero).

In our experiment settings, to make a fair comparison between the proposed algorithm and other algorithms, all the semi-supervised clustering algorithms use the same labeled dataset. 10% of the samples of each cluster are employed as labeled samples. Here, the number of clusters k is set equal to the number of ground-truth clusters, and the values of ARI and NMI are reported in Table 3.

4.2. Evaluation measures

The Normalized Mutual Information (NMI [35]) and the Adjusted Rand Index (ARI [36]) are used to evaluate the performance of ConvexClust with the other clustering approaches after running each algorithm over the same dataset. NMI is an external measure for defining the quality of clustering. Let $X = \{c_1, c_2, ..., c_k\}$ and $Y = \{c'_1, c'_2, ..., c'_{k'}\}$ be the random variables described by the cluster assignments and class labels, respectively. $I(X, Y)$ denotes the mutual information between X and Y; $H(X)$ and

$H(Y)$ are the entropy of X and Y and then, the NMI is defined in Eq. (7):

$$NMI(X,Y) = \frac{2I(X;Y)}{H(X)+H(Y)}, \tag{7}$$

where $I(X,Y) = H(X) - H(X|Y)$ is the mutual information between the random variables X and Y, $H(X)$ is the Shannon entropy of X, and $H(X|Y)$ is the conditional entropy of X given Y. The NMI is normalized; therefore, its value changes between 0 and 1.

The Adjusted Rand Index $ARI(X,Y)$ is also an external measure with a value between –1 and 1. The closer the ARI value to one, the better the clustering quality. The ARI is defined in Eqs. (8) and (9):

$$ARI(X,Y) = \frac{\sum_{h=1}^{k}\sum_{l=1}^{k'}\binom{|C_h - C'_l|}{2} - EIdx}{\frac{1}{2}\left(\sum_{h=1}^{k}\binom{|C_h|}{2} + \sum_{l=1}^{k'}\binom{|C'_l|}{2}\right)}, \tag{8}$$

$$EIdx = \frac{2\sum_{h=1}^{k}\binom{|C_h|}{2}\sum_{l=1}^{k'}\binom{|C'_l|}{2}}{n(n-1)}, \tag{9}$$

where $h \in \{1,...,k\}, l \in \{1,...,k'\}$, n is the total number of data samples, and $|\bullet|$ denotes the number of samples in the cluster. In our experiment, $k = k'$ and is defined initially.

4.3. Experimental results and analysis

For experimental results, the number of labeled data equals ten percent of the total data and is considered a fixed set in all semi-supervised algorithms for a fair comparison. Other settings in the compared algorithms are as follows.

K is equal to the number of clusters in all algorithms and $m = 2$ in the FCM algorithm. The parameters of fuzzy G-K and fuzzy G-G are also set according to the settings mentioned in [30]. It is noteworthy that in order to assign data to clusters, Euclidean distance has been used in all algorithms.

In traditional semi-supervised clustering methods, such as seeded K-means, data are used in the initialization stage of the algorithm (determination of primary cluster centers). It does not use labeled data in the algorithm learning process. Still, the constrained K-means uses labeled data in the learning process so that the algorithm does not decide on the labeled data and the data are labeled according to their cluster. The proposed method uses labeled data more efficiently. In this way, at the beginning of the algorithm, the primary clusters are formed with initial convex hulls and these clusters are the basis of the development and learning process of the algorithm. At the same time, the labeled data are the starting point and remain intact in the learning process. They also play a role in assigning unlabeled data, and for this assignment, not only is the distance criterion measured, but also the data density is essential. However, each method has its advantages and disadvantages. In Table 2, these features are summarized.

Table 3 shows the ARI and NMI values of the proposed semi-supervised clustering algorithm and the other compared algorithms. According to the ARI and NMI values, ConvexClust works better than the other algorithms in most cases. Among 19 datasets based on both criteria in 14 cases, the proposed method and fuzzy G-G algorithm exhibit better performance in 3 and 2 cases, respectively. The proposed method exhibits better outcome than the rest in terms of **one** criterion.

The costliest module in the proposed method is the formation of convex hulls; the corresponding cost increases with increasing data dimensions and this, in turn, reduces the speed and accuracy of the algorithm. Since dimensional data reduction is used as a pre-processing step of the algorithm for high dimensional datasets, the quality of the method is influenced by the feature selection method. As long as the feature selection techniques do not remove much information from the data, the expected results can be produced.

To illustrate the superiority of the proposed method over other comparable algorithms, the error rate associated with all these algorithms on the 2d-4c-no3 dataset has been calculated. Figure 4a shows the dataset with labeled data; Figure 4b presents the clustering result using the proposed method on that dataset; Figure 4c represents the desirable clustering result; and Figure 4d compares the error rates of the proposed method and other methods. As seen in Figure 4d, ConvexClust effectively used the labeled data to

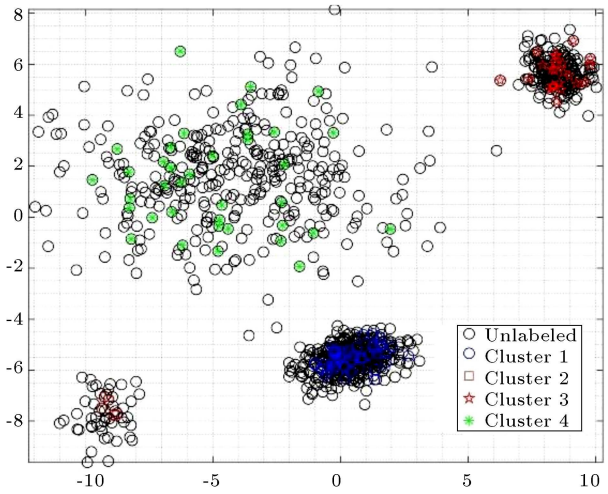

Figure 4a. Labeled and unlabeled examples.

Table 2. Merits and demerits of the proposed method.

ConvexClust merits	ConvexClust demerits
Easy implementation	The weak performance of the algorithm in identifying clusters of overlapping datasets and low-density data
The only parameter that must be set in this algorithm is the number of clusters (K)	Requiring at least three labeled data sets per cluster to form primary convex hulls
Good performance of the algorithm in identifying separate hyper-spherical clusters with high density	Requiring data dimension reduction techniques due to the high cost of convex hull formation in high dimensional data

Table 3. Comparison of ARI and NMI values measured by different clustering algorithms.

	Algorithm													
	Proposed method (ConvexClust)		***K*-means**		**S-Kmeans**		**C-Kmeans**		**FCM**		**Fuzzy G-G**		**Fuzzy G-K**	
Dataset	**ARI**	**NMI**	**ARI**	**NMI**	**ARI**	**NMI**	**ARI**	**NMI**	**ARI**	**NMI**	**ARI**	**NMI**	**ARI**	**NMI**
2d-10c-no0	**0.91**	**0.94**	0.86	0.92	0.88	0.93	0.89	0.93	0.79	0.89	0.44	0.66	0.41	0.63
2d-10c-no1	**0.89**	**0.91**	0.72	0.83	0.81	0.86	0.83	0.87	0.61	0.75	0.59	0.71	0.40	0.65
2d-10c-no2	**0.96**	**0.96**	0.92	0.94	0.93	0.95	0.93	0.95	0.84	0.88	0.47	0.67	0.38	0.60
2d-10c-no3	**0.96**	**0.95**	0.94	0.94	0.95	0.95	0.95	0.95	0.87	0.89	0.57	0.76	0.37	0.63
2d-10c-no4	**0.94**	**0.93**	0.82	0.87	0.87	0.89	0.88	0.90	0.83	0.86	0.42	0.60	0.26	0.55
2d-10c-no5	**0.85**	**0.90**	0.76	0.85	0.82	0.87	0.85	0.88	0.73	0.83	0.41	0.62	0.37	0.62
2d-10c-no6	**0.88**	**0.91**	0.70	0.80	0.84	0.87	0.85	0.88	0.70	0.80	0.50	0.70	0.33	0.56
2d-10c-no8	**0.92**	**0.93**	0.67	0.80	0.86	0.88	0.88	0.90	0.62	0.78	0.47	0.69	0.38	0.62
2d-4c-no1	**0.89**	**0.86**	0.84	0.84	0.88	0.83	0.89	0.84	0.87	0.82	0.72	0.80	0.87	0.90
2d-4c-no2	**0.95**	**0.94**	0.56	0.74	0.91	0.90	0.93	0.92	0.91	0.90	0.87	0.86	0.77	0.83
2d-4c-no3	**0.95**	**0.93**	0.92	0.91	0.92	0.91	0.93	0.91	0.82	0.82	0.77	0.82	0.93	0.94
2d-4c-no4	**0.97**	**0.95**	0.97	0.91	0.90	0.95	0.97	0.95	0.97	0.94	0.91	0.91	0.82	0.84
2d-4c-no5	**0.92**	**0.90**	0.79	0.81	0.91	0.89	0.92	0.90	0.77	0.79	0.91	0.88	0.91	0.91
2d-4c-no6	0.97	0.97	0.95	0.95	0.95	0.95	0.96	0.95	0.95	0.95	**0.99**	**0.99**	0.68	0.82
2d-4c-no7	0.91	0.90	0.85	0.86	0.85	0.86	0.87	0.88	0.57	0.75	**0.99**	**0.98**	0.75	0.84
2d-4c-no9	**0.94**	**0.93**	0.90	0.90	0.90	0.90	0.91	0.91	0.89	0.89	0.89	0.89	0.79	0.83
Heart1302	0.03	**0.06**	0.03	0.02	0.03	0.02	0.06	0.04	0.03	0.02	0.01	0.03	0.00	0.01
Liver0602	**0.01**	0.01	-0.01	0.00	-0.01	0.00	0.00	0.00	-0.01	0.01	0.00	0.01	0.00	0.01
BankAuthentication0402	0.39	0.36	0.05	0.03	0.05	0.03	0.09	0.06	0.05	0.03	**0.46**	**0.58**	0.24	0.35

cluster the 2d-4c-no3 dataset and its error rate was lower than the others.

According to the results, the overlapping datasets and the datasets with non-spherical clusters are not suitable for clustering with the proposed algorithm. Moreover, clustering is also affected by the location of labeled data, and the use of different labeled datasets produces different results. The percentage of labeled data is another factor influencing clustering.

5. Conclusion

This paper presented a semi-supervised clustering approach that divided datasets into K groups, creating incremental convex hulls iteratively by using labeled data. This algorithm is an extended version of constrained K-means by which this study used labeled data to create convex hulls instead of applying them for cluster center initialization. Cluster compactness

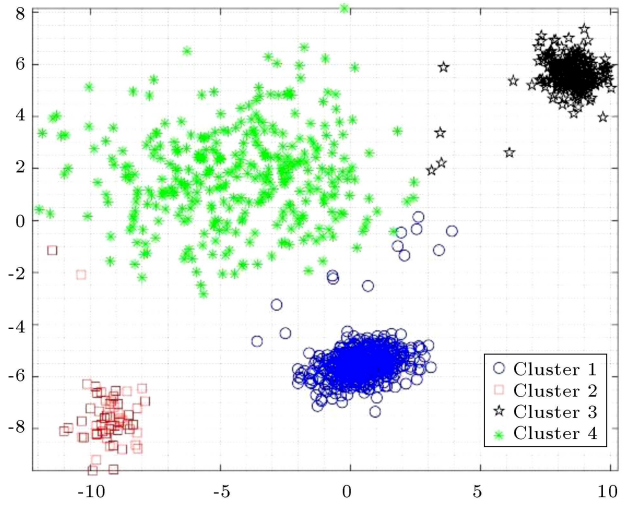

Figure 4b. Clustered examples using CovexClust.

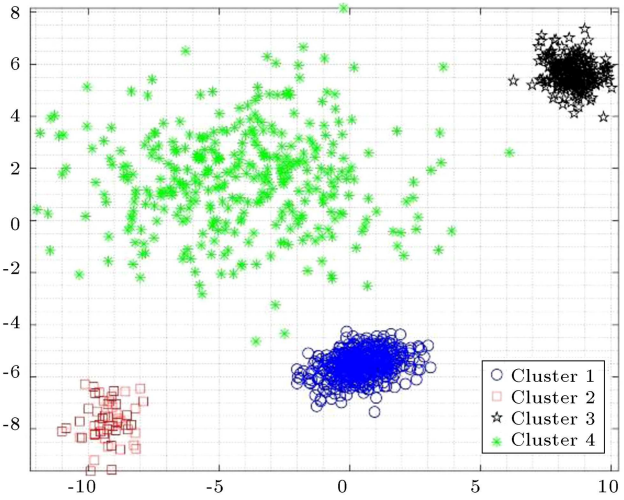

Figure 4c. Original fully labeled data.

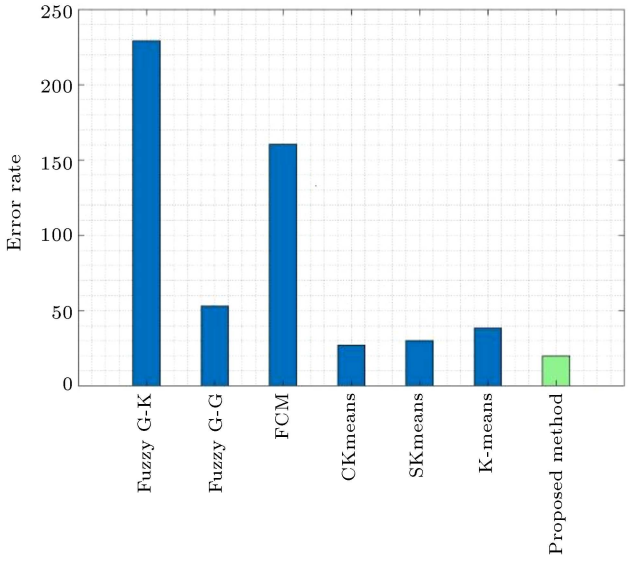

Figure 4d. Error rate comparison for the different algorithms.

and cluster separation concepts were applied by the objective function, which (a) counted the number of labeled data sets enclosed between each convex hull and the circle around the unlabeled data at r radius (the average distance of the unlabeled data to the mean center of each convex hull) and (b) assigned the unlabeled data to the convex hull having the maximum amount of labeled data in the shared spaces. According to the obtained results, the proposed approach outperformed other algorithms.

It is recommended that the following points be considered in future studies:

- Fixing the difficulty of the proposed method in identifying overlapped, non-spherical, and low-density clusters.
- Eliminating the dependence of the algorithm on the existence of at least three labeled data sets in each cluster to form a convex hull for applications in which the datasets have small clusters and ten percent of each cluster does not include the three labeled data.
- Applying the proposed method to each dataset for different sets of labeled data and presenting the average of those results as a clustering result.
- Investigating the proposed method in exchange for incremental percentages of labeled data.
- Applying the proposed method in practical applications such as segmentation and investigating the results.

References

1. Zhang, H., Li, H., Chen, N., et al. "Novel fuzzy clustering algorithm with variable multi-pixel fitting spatial information for image segmentation", *Pattern Recognit.*, **121**, pp. 108–201 (2022).
2. Zhao, F., Cao, L., Liu, H., et al. "Particle competitive mechanism-based multi-objective rough clustering algorithm for image segmentation", *IEEE Trans. Fuzzy Syst.*, **30**(10), pp. 4127–4141 (2022).
3. Hashemi, H., De Beukelaar, P., Beiranvand, B., et al. "Clustering seismic datasets for optimized facies analysis using a SSCSOM technique", *79th EAGE Conf. and Exhibition*, **2017**(1), pp. 1–5 (2017).
4. Gaur, A. and Yadav, S. "Handwritten Hindi character recognition using k-means clustering and SVM", *4th Int. Symp. on Emerg. Trends and Tech. in Libr. and Info. Serv.*, Noida, India, pp. 65–70 (2015).
5. Singh, R., Shukla, A.K., Mishra, R.K., et al., *An Improved Approach for Devanagari Handwritten Characters Recognition System*, pp. 217–226 (2022).
6. Ma, X., Keung, J., Yang, Z., et al. "Combining clustering with attention semantic model for identifying security bug reports", *Inf. Softw. Technol.*, **147**, pp. 106–906 (2022).

7. Ye, W., Wang, H., and Zhong, Y. "Optimization of network security protection situation based on data clustering", *Int. J. Syst. Assur. Eng. Manag.*, pp. 1–8 (2022).

8. Kanthimathi, N., Roshini Roy, J., Saranya, N., et al. "Trust-based security scheme using fuzzy clustering for vehicular Ad Hoc networks", *Soft Comp. for Secu. Appli.,* Singapore, pp. 425–436 (2022).

9. Sathyamoorthy, M., Kuppusamy, S., Dhanaraj, R.K., et al. "Improved K-means based Q learning algorithm for optimal clustering and node balancing in WSN", *Wirel. Pers. Commun.*, **122**(3), pp. 2745–2766 (2021).

10. Sharma, R., Vashisht, V., and Singh, U. "A fuzzy-based clustering algorithm using hybrid technique for wireless sensor networks", *Int. J. Adv. Intell. Paradig.*, **21**(1–2), pp. 129–157 (2022).

11. Jayaraman, G. and Dhulipala, V.R.S. "Fuzzy-based energy-efficient cluster head selection algorithm for lifetime enhancement of wireless sensor networks", *Arab. J. Sci. Eng.*, **47**(2), pp. 1631–1641 (2021).

12. Srinivas, M. and Amgoth, T. "Data acquisition in large-scale wireless sensor networks using multiple mobile sinks: a hierarchical clustering approach", *Wirel. Networks*, **28**(2), pp. 603–619 (2022).

13. Ezugwu, A.E., Ikotun, A.M., Oyelade, O.O., et al. "A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects", *Eng. Appl. Artif. Intell.*, **110**, pp. 104–743 (2022).

14. Sun, R. "A recognition method for visual image of sports video based on fuzzy clustering algorithm", *Int. J. Inf. Commun. Technol.*, **20**(1), pp. 1–17 (2022).

15. Sardar, T.H. and Ansari, Z. "MapReduce-based fuzzy C-means algorithm for distributed document clustering", *J. Inst. Eng. Ser. B*, **103**(1), pp. 131–142 (2021).

16. Agapito, G. and Fedele, G. "Clustering methods for microarray data sets", *Methods Mol. Biol.*, **240**(1), pp. 249–261 (2022).

17. Sharma, C.M. and Dinkar, S.K. "A survey on evolutionary clustering algorithms and applications", *Appl. Adv. Optim. Tech. Ind. Eng.,* pp. 23–34 (2022).

18. Wei, S., Li, Z., and Zhang, C. "A semi-supervised clustering ensemble approach integrated constraint-based and metric-based", *7th Int. Conf. on Inte. Multim. Comp. and Serv.*, Hunan, China, pp. 19–21 (2015).

19. Alok, A., Saha, S., and Ekbal, A. "MR brain image segmentation using muti-objective semi-supervised clustering", *Int. Conf. on Signal Process., Inform., Commun. and Energy Sys.*, Kozhikode, India (2015).

20. Qin, Y., Ding, S., Wang, L., et al. "Research progress on semi-supervised clustering", *Cognit. Comput.*, **11**, pp. 599–612 (2019).

21. Dinler, D. "A survey of constrained clustering", In *Unsupervised Learning Algorithms*, Tural, M.K., Springer (2016).

22. Nanda, S.J. and Panda, G. "A survey on nature inspired metaheuristic algorithms for partitional clustering", *Swarm Evol. Comput.,* **16**, pp. 1–18 (2014).

23. Sanodiya, R.K., Saha, S., and Mathew, J. "A kernel semi-supervised distance metric learning with relative distance: Integration with a MOO approach", *Expert Syst. Appl.*, **125**, pp. 233–248 (2019).

24. Zhang, Z., Kwok, J.T., and Yeung, D.Y. "Parametric distance metric learning with label information", Report HKUST-CS 03-02, Depar. of Compu. Scien. The Hong Kong University of Science and Technology Clear Water Bay, Kowloon, Hong Kong (2003).

25. Sanodiya, R., Saha, S., and Mathew, J. "A kernel semi-supervised distance metric learning with relative distance: Integration with a MOO approach", *Expe. Syst. with Applic.*, **125**, pp. 233–248 (2019).

26. Baghshah, M. and Shouraki, S. "Kernel-based metric learning for semi-supervised clustering", *Neurocomput.*, **73**(7–9), pp. 1352–1361 (2010).

27. Wagstaff, K. and Cardie, C. "Clustering with instance-level constraints", *17th Inter. Conf. on Machi. Learn.*, pp. 1103–1110 (2000).

28. Basu, S., Basu, S., Banerjee, A., et al. "Semi-supervised clustering by seeding", *19th Inter. Conf. on Machi. Learn.*, Sydney, Australia, pp. 19–26 (2002).

29. Hashemi, H., Javaherian, A., and Babuska, R. "A semi-supervised method to detect seismic random noise with fuzzy GK clustering", *J. Geophys. Eng.*, **5**(4), pp. 457–468 (2008).

30. Gath, I. and Geva, A.B. "Unsupervised optimal fuzzy clustering", *IEEE Trans. on Patte. Analy. and Machi. Intelli.*, **11**(7), pp. 773–781 (1989).

31. Saha, S. and Bandyopadhyay, S. "Some connectivity based cluster validity indices", *Appl. Soft Comput. J.*, **12**(5), pp. 1555–1565 (2012).

32. Cormen, T.H., *Introduction to Algorithms*, Leiserson, C.E., Rivest, R.L., et al., 3th Edn., pp.1-1313, MIT Press, Massachusetts, UK (2009).

33. Bezdek, J.C., Ehrlich, R., and Full, W. "FCM: The fuzzy C-means clustering algorithm", *Comp. and Geosci.*, **10**(2–3), pp. 191–203 (1984).

34. Handl, J. and Knowles, J. "An evolutionary approach to multiobjective clustering", *IEEE Transa. on Evol. Comput.*, **11**(1), pp. 56–76 (2007).

35. Wei, S., Li, Z., and Zhang, C. "Combined constraint-based with metric-based in semi-supervised clustering ensemble", *Int. J. Mach. Learn. Cybern.*, **9**(7), pp. 1085–1100 (2018).

36. Yu, Z., Kuang, Z., Liu, J., et al. "Adaptive ensembling of semi-supervised clustering solutions", *IEEE Trans. Knowl. Data Eng.*, **29**(8), pp. 1577–1590 (2017).

Biographies

Atiyeh Taghizabet received BSc degree in Computer Engineering from the Islamic Azad University - South Tehran Branch, Iran in 2006. She received her MSc degree in the same field at Payame-Noor University, Rey Branch, Iran in 2011. She is currently a PhD candidate of Software Systems and her research interests include optimization algorithms, metaheuristic algorithms, swarm intelligence, and data mining.

Jafar Tanha received BSc and MSc degrees in Computer Science from the University of Amir Kabir (Polytechnic), Tehran, Iran in 1999 and 2001, respectively, and the PhD degree in Computer Science-Artificial Intelligence from the University of Amsterdam (UvA), Amsterdam, The Netherlands in 2013. He joined INL Institute, Leiden, The Netherland as a researcher from 2013 to 2015. Since 2015, he has been with the Department of Computer Engineering, Payame-Noor University, Tehran, Iran, where he was an Assistance Professor. He has held lecturing positions at the Iran University of Science & Technology, Tehran, Iran in 2016. He is currently an Assistant Professor at the University of Tabriz, Tabriz, Iran. His main areas of research interest are machine learning, pattern recognition, and document analysis.

Amineh Amini received her BSc degree in Software Engineering from Mashhad Azad University in 2001. She obtained her MSc degree in the same field from Najafabad Azad University in 2005. Then, she received her PhD degree and post-doctoral from University of Malaya. She is a faculty member and the Head of the Department of Computer Engineering of Karaj Azad University. Her main research interests include data mining and software remodularization.

Javad Mohammadzadeh received his BSc degree in Computer Science from Shahid Bahonar University of Kerman, Iran in 2004. He received his MSc degree in Computer Science from the University of Tehran in 2007 and PhD degree in Bioinformatics from the University of Tehran in 2014. His research interests include swarm intelligence algorithms, bioinformatics algorithms, complex dynamical networks, parallel computing, and deep learning.