

Research Article

Intuitionistic Fuzzy Possibilistic C Means Clustering Algorithms

Arindam Chaudhuri

Samsung Research & Development Institute, Noida 201304, India

Correspondence should be addressed to Arindam Chaudhuri; arindamphdthesis@gmail.com

Received 24 August 2014; Accepted 4 October 2014

Academic Editor: Ferdinando Di Martino

Copyright © 2015 Arindam Chaudhuri. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Intuitionistic fuzzy sets (IFSs) provide mathematical framework based on fuzzy sets to describe vagueness in data. It finds interesting and promising applications in different domains. Here, we develop an intuitionistic fuzzy possibilistic C means (IFPCM) algorithm to cluster IFSs by hybridizing concepts of FPCM, IFSs, and distance measures. IFPCM resolves inherent problems encountered with information regarding membership values of objects to each cluster by generalizing membership and nonmembership with hesitancy degree. The algorithm is extended for clustering interval valued intuitionistic fuzzy sets (IVIFSs) leading to interval valued intuitionistic fuzzy possibilistic C means (IVIFPCM). The clustering algorithm has membership and nonmembership degrees as intervals. Information regarding membership and typicality degrees of samples to all clusters is given by algorithm. The experiments are performed on both real and simulated datasets. It generates valuable information and produces overlapped clusters with different membership degrees. It takes into account inherent uncertainty in information captured by IFSs. Some advantages of algorithms are simplicity, flexibility, and low computational complexity. The algorithm is evaluated through cluster validity measures. The clustering accuracy of algorithm is investigated by classification datasets with labeled patterns. The algorithm maintains appreciable performance compared to other methods in terms of pureness ratio.

1. Introduction

Clustering algorithms [1, 2] form an integral part of computational intelligence and pattern recognition research. Clustering analysis is commonly used as an important tool to classify collection of objects into homogeneous groups, such that objects within a given group are similar to each other whereas objects within different groups are dissimilar to each other. The concept is based on the notion of similarity, which is a basic component of intelligence and ubiquitous to scientific endeavor. Clustering finds numerous applications [3] across a variety of disciplines such as taxonomy, image processing, information retrieval, data mining, pattern recognition, microbiology, archaeology and geographical analysis, and so forth. It is an exploratory tool for deducing the nature of data by providing labels to individual objects that describe how the data separate into groups. It has improved the performance of other systems by separating the problem domain into manageable subgroups [4]. Often researchers are confronted with the challenging datasets that are large and unlabeled. There are many methods available in exploratory

data analysis [5, 6] by which researchers can elucidate these data.

Clustering an unlabeled dataset $X = \{x_1, \dots, x_n\} \subset \mathcal{R}^p$ is partitioning of X into $1 < c < n$ subgroups such that each subgroup represents natural substructure in X . This is done by assigning labels to vectors in X and hence to objects generating X . A c partition of is a set of cn values that can be conveniently represented as $c \times n$ matrix = $[u_{ik}]$. There are generally three sets of partition matrices [7, 8]:

$$M_{pcn} = \{U \in \mathcal{R}^{cn} : 0 \leq u_{ik} \leq 1 \forall i, k; \forall k \exists i \ni u_{ik} > 0\} \quad (1)$$

$$M_{fcn} = \left\{ U \in M_{pcn} : \sum_{i=1}^c u_{ik} = 1 \forall k; \sum_{i=1}^c u_{ik} > 0 \forall i \right\} \quad (2)$$

$$M_{hcn} = \{U \in M_{fcn} : u_{ik} = 0 \forall i \forall k\}. \quad (3)$$

The matrix M_{pcn} in (1) has the property that for any k there exists at least an index i such that u_{ik} is greater than 0. The matrix M_{fcn} in (2) states that if $\sum_{i=1}^c u_{ik}$ is equal to 1 for any k , it is obvious that $\sum_{i=1}^c u_{ik}$ is greater than 0.

The matrix M_{hcn} in (3) is formed by boolean matrices as a subset of matrix M_{fcn} in (2). The equations (1), (2), and (3) thus define the sets of possibilistic, fuzzy, or probabilistic and crisp c partitions of X , respectively. Hence, there are thus four kinds of label vectors, but fuzzy and probabilistic label vectors are mathematically identical having entries between 0 and 1 that sum to 1 over each column. The reason these matrices are called partitions follows from the interpretation of their entries. If U is crisp or fuzzy, u_{ik} is taken as a membership of \mathbf{x}_k in i th partitioning fuzzy subset or cluster of X . If U in M_{fcn} is probabilistic, u_{ik} is the posterior probability $\text{Prob}(i/\mathbf{x}_k)$. If U in M_{pcn} is possibilistic, it has entries between 0 and 1 that do not necessarily sum to 1 over any column. In this case, u_{ik} is taken as the possibility that \mathbf{x}_k belongs to class i . An alternate interpretation of possibility u_{ik} is that it measures the typicality of \mathbf{x}_k to cluster i . It is observed that $M_{hcn} \subset M_{fcn} \subset M_{pcn}$. A clustering algorithm C finds $U \in M_{hcn}(M_{fcn}, M_{pcn})$ which best explains and represents an unknown structure in X with respect to the model that defines C . For U in M_{fcn} , $c = 1$ is represented uniquely by the hard 1 partition, $\mathbf{I}_n = \underbrace{[1 \cdots 1]}_{n \text{ times}}$ which unequivocally assigns all

n objects to a single cluster, and $c = n$ is represented uniquely by $U = I_n$, the $n \times n$ identity matrix up to a permutation of columns. In this case, each object is in its own singleton cluster. Choosing $c = 1$ or $c = n$ rejects the hypothesis that X contains clusters.

In the last few decades, variety of clustering techniques [3, 5, 6, 9, 10] has been developed to classify data. Clustering techniques are broadly divided into hierarchical and partition methods. Hierarchical clustering [5] generates hierarchical tree of clusters called dendrogram which can be either divisive or agglomerative [3]. The former is a top-down splitting technique which starts with all objects in one cluster and forms hierarchy by dividing objects into smaller clusters in an iterative procedure until the desired number of clusters is achieved or considered objects which is constituted as unique cluster. The latter starts by considering each object as cluster, followed by comparing them amongst themselves using distance measure. The clusters with smaller distance are considered as constituting unique group and then merged. The merging procedure is repeated until the desirable number of clusters is achieved or only one cluster is left with all considered objects. Partition clustering method gives single c partition of objects, with c being the predefined number of clusters [11]. One of the most widely used partition clustering algorithms is fuzzy C means (FCM). FCM is a combination of k means clustering algorithm and fuzzy logic [1, 7]. It works iteratively in which the desired number of clusters c and initial seeds are predefined. FCM algorithm assigns memberships to \mathbf{x}_k which are inversely proportional to relative distance of \mathbf{x}_k to c point prototypes $\{\mathbf{v}_i\}$ that are cluster centers. For $c = 2$, if \mathbf{x}_k is equidistant from two prototypes, the membership of \mathbf{x}_k in each cluster will be the same regardless of absolute value of the distance of \mathbf{x}_k from two centroids as well as from other points. The problem this creates is noise points which are far but equidistant from central structure of two clusters that can never be given equal membership, when it seems far more natural that such points

are given very low or no membership in either cluster. This problem was overcome by Krishnapuram and Keller [8], who proposed possibilistic C means (PCM) which relaxes column sum constraint in (2) so that sum of each column satisfies the constraint $0 < \sum_{i=1}^c u_{ik} \leq c$. In other words, each element of k th column can be between 0 and 1, as long as at least one of them is positive. They suggested that value should be interpreted as typicality of \mathbf{x}_k relative to cluster i . They interpreted each row of U as possibility distribution over X . The objective function of PCM algorithm sometimes helps to identify outliers, that is, noise points. However, Barni et al. [12] pointed that PCM pays price for its freedom to ignore noise points such that it is very sensitive to initializations and sometimes generates coincident clusters. Moreover, typicality can be very sensitive to the choice of additional parameters needed by PCM algorithm. The coincident cluster problem of PCM algorithm was avoided by two possibilistic fuzzy clustering algorithms proposed by Timm et al. [13–15]. They modified PCM objective function by adding an inverse function of distances between the cluster centers. This term acts in repulsive nature and avoids coincident clusters. In [13, 14], Timm et al. used the same concept to modify objective function as used by Gustafson and Kessel [16] clustering algorithm. These algorithms exploit benefits of both fuzzy and possibilistic clustering. Pal et al. [17] justified the need for both possibility, that is, typicality and membership values, and proposed a model and corresponding algorithm called fuzzy possibilistic C means (FPCM). This algorithm normalizes possibility values, so that the sum of possibilities of all data points in a cluster is 1. Although FPCM is much less prone to errors encountered by FCM and PCM, possibility values are very small when size of dataset increases.

The notion of intuitionistic fuzzy set (IFS) coined by Atanassov [22] for fuzzy set generalizations has interesting and useful applications in different domains such as logic programming, decision making problems, and medical diagnostics [23–26]. This generalization presents degrees of membership and nonmembership with a degree of hesitancy. Thus, knowledge and semantic representation become more meaningful and applicable [27, 28]. Sometimes it is not appropriate to assume that membership and nonmembership degree of an object are exactly defined [29], but value ranges or value intervals can be assigned. In such cases, IFS can be generalized and interval valued intuitionistic fuzzy set (IVIFS) [29] can be defined whose components are intervals rather than exact numbers. IFSs and IVIFSs have been found to be very useful to describe and deal with vague and uncertain data [28, 30]. With this motivation, it is desirable to develop some practical approaches to clustering IFSs and IVIFSs. Intuitionistic fuzzy similarity matrix was defined by [31] and thereby intuitionistic fuzzy equivalence matrix was developed. The work in [31] gave an approach to transform intuitionistic fuzzy similarity matrices into intuitionistic fuzzy equivalence matrices, based on which a procedure for clustering intuitionistic fuzzy sets was proposed. Some methods for calculating association coefficients of IFSs or IVIFSs and corresponding clustering algorithm were introduced by [32]. The algorithm used derived association coefficients of IFSs or IVIFSs to construct an association matrix and utilized

the procedure to transform it into an equivalent association matrix. Reference [33] introduced an intuitionistic fuzzy hierarchical algorithm for clustering IFSs which is based on traditional hierarchical clustering procedure and intuitionistic fuzzy aggregation operator. These algorithms cannot provide information about membership degrees of objects to each cluster.

In this work, an intuitionistic fuzzy possibilistic C means (IFPCM) algorithm to cluster IFSs is developed. IFPCM is obtained by applying IFSs to FPCM which is a known clustering method based on basic distance measures between IFSs [34, 35]. At each stage of the algorithm seeds are modified and for each IFS membership and typicality degrees to each of the clusters are estimated. The algorithm ends when all given IFSs are clustered according to estimated membership and typicality degrees. It overcomes the inherent problems encountered with information regarding membership values of objects to each cluster by generalizing membership and nonmembership with hesitancy degree. The algorithm is then extended to interval valued intuitionistic fuzzy possibilistic C means (IVIFPCM) for clustering IVIFSs. The algorithms are illustrated through conducting experiments on different datasets. The evaluation of the algorithm is performed through cluster validity measures. The clustering accuracy of the algorithm is determined by classification datasets with labeled patterns. IFPCM algorithm is simple and flexible in nature and provides information about membership and typicality degrees of samples to all clusters with low computational complexity.

This paper is organized as follows. In the next section, the concepts of IFSs and IVIFSs are defined. FPCM clustering algorithm is given in Section 3. The next section presents IFPCM clustering algorithms for IFSs and IVIFSs, respectively. The experimental results on both real world and simulated datasets are illustrated in Section 5. Finally, in Section 6 conclusions are given.

2. Intuitionistic Fuzzy Sets and Interval Valued Intuitionistic Fuzzy Sets

In this section, we present some basic definitions associated with IFSs and IVIFSs.

Definition 1. Considering X as universe of discourse [22], then IFS is defined as

$$V = \{ \langle x, \mu_V(x), \nu_V(x) \rangle \mid x \in X \}. \quad (4)$$

In (4) $\mu_V(x)$ and $\nu_V(x)$ are the membership and nonmembership degrees, respectively, satisfying the following constraints:

$$\begin{aligned} \mu_V : X &\longrightarrow [0, 1], & x \in X &\longrightarrow \mu_V(x) \in [0, 1], \\ \nu_V : X &\longrightarrow [0, 1], & x \in X &\longrightarrow \nu_V(x) \in [0, 1]. \end{aligned} \quad (5)$$

Equation (5) is subject to the condition that $\mu_V(x) + \nu_V(x) \leq 1$, $\forall x \in X$.

Definition 2. For each IFS V in X , if $\pi_V(x) = 1 - \mu_V(x) - \nu_V(x)$, then $\pi_V(x)$ is called hesitation degree (or intuitionistic index) [36] of x to V . Obviously $\pi_V(x)$ is specified in the range $0 \leq \pi_V(x) \leq 1$; especially if $\pi_V(x) = 0 \forall x \in X$, then IFS V is reduced to fuzzy set. If $\pi_V(x)$ and $\nu_V(x)$ have 0 values such that $\mu_V(x) = \nu_V(x) = 0 \forall x \in X$, then IFS V is completely intuitionistic.

Considering the fact that the elements x_i ; $i = 1, \dots, n$ in universe X have different importance, let us assume $w = (w_1, \dots, w_n)$ should be the weight vector of x_i ; $i = 1, \dots, n$ with

$$w_i \geq 0, \quad \sum_{i=1}^n w_i = 1. \quad (6)$$

Xu [37] defined the following weighted Euclidean distance between IFSs V and W :

$$\begin{aligned} D_\alpha(V, W) = & \left(\frac{1}{2} \sum_{i=1}^n w_i \left((\mu_V(x_i) - \mu_W(x_i))^2 \right. \right. \\ & + (\nu_V(x_i) - \nu_W(x_i))^2 \\ & \left. \left. + (\pi_V(x_i) - \pi_W(x_i))^2 \right) \right)^{1/2}. \end{aligned} \quad (7)$$

In particular, if $w = (1/n, \dots, 1/n)$, then (7) is reduced to normalized Euclidean distance [34] which is defined as follows:

$$\begin{aligned} D_\alpha(V, W) &= \left(\frac{1}{2n} \sum_{i=1}^n \left((\mu_V(x_i) - \mu_W(x_i))^2 \right. \right. \\ & \left. \left. + (\nu_V(x_i) - \nu_W(x_i))^2 \right. \right. \\ & \left. \left. + (\pi_V(x_i) - \pi_W(x_i))^2 \right) \right)^{1/2}. \end{aligned} \quad (8)$$

Atanassov and Gargov [29] pointed out that sometimes it is not appropriate to assume that membership and nonmembership degrees of the element are exactly defined but value ranges or value intervals can be given. In this context, Atanassov and Gargov [29] extended IFS and introduced the concept of IVIFS, which is characterized by a membership degree and a nonmembership degree, whose values are intervals rather than exact numbers.

Definition 3. An IVIFS \tilde{V} over X is an object having the following form [29]:

$$\tilde{V} = \{ \langle x, \tilde{\mu}_{\tilde{V}}(x), \tilde{\nu}_{\tilde{V}}(x) \rangle \mid x \in X \}. \quad (9)$$

Here, $\tilde{\mu}_{\tilde{V}}(x) = [\tilde{\mu}_{\tilde{V}}^L(x), \tilde{\mu}_{\tilde{V}}^U(x)] \subset [0, 1]$ and $\tilde{\nu}_{\tilde{V}}(x) = [\tilde{\nu}_{\tilde{V}}^L(x), \tilde{\nu}_{\tilde{V}}^U(x)] \subset [0, 1]$ are intervals $\tilde{\mu}_{\tilde{V}}^L(x) = \inf \tilde{\mu}_{\tilde{V}}(x)$, $\tilde{\mu}_{\tilde{V}}^U(x) = \sup \tilde{\mu}_{\tilde{V}}(x)$, $\tilde{\nu}_{\tilde{V}}^L(x) = \inf \tilde{\nu}_{\tilde{V}}(x)$, $\tilde{\nu}_{\tilde{V}}^U(x) = \sup \tilde{\nu}_{\tilde{V}}(x)$, and $\tilde{\mu}_{\tilde{V}}^U(x) + \tilde{\nu}_{\tilde{V}}^U(x) \leq 1, \forall x \in X$ and $\tilde{\pi}_{\tilde{V}}(x) = [\tilde{\pi}_{\tilde{V}}^L(x), \tilde{\pi}_{\tilde{V}}^U(x)]$, where

$\pi_{\tilde{V}}^L(x) = 1 - \mu_{\tilde{V}}^U(x) - \nu_{\tilde{V}}^U(x)$, $\pi_{\tilde{V}}^U(x) = 1 - \mu_{\tilde{V}}^L(x) - \nu_{\tilde{V}}^L(x)$, $\forall x \in X$. In particular, if $\mu_{\tilde{V}}(x) = \tilde{\mu}_{\tilde{V}}^L(x) = \tilde{\mu}_{\tilde{V}}^U(x)$ and $\nu_{\tilde{V}}(x) = \tilde{\nu}_{\tilde{V}}^L(x) = \tilde{\nu}_{\tilde{V}}^U(x)$, then \tilde{V} is reduced to an IFS.

Now we extend the weighted Euclidean distance measure given in (7) to IVIFS theory:

$$\begin{aligned}
D_\gamma(\tilde{V}, \tilde{W}) &= \left(\frac{1}{4} \sum_{l=1}^n w_l \left((\tilde{\mu}_{\tilde{V}}^L(x_l) - \tilde{\mu}_{\tilde{W}}^L(x_l))^2 \right. \right. \\
&\quad + (\tilde{\mu}_{\tilde{V}}^U(x_l) - \tilde{\mu}_{\tilde{W}}^U(x_l))^2 + (\tilde{\nu}_{\tilde{V}}^L(x_l) - \tilde{\nu}_{\tilde{W}}^L(x_l))^2 \\
&\quad + (\tilde{\nu}_{\tilde{V}}^U(x_l) - \tilde{\nu}_{\tilde{W}}^U(x_l))^2 \\
&\quad + (\tilde{\pi}_{\tilde{V}}^L(x_l) - \tilde{\pi}_{\tilde{W}}^L(x_l))^2 \\
&\quad \left. \left. + (\tilde{\pi}_{\tilde{V}}^U(x_l) - \tilde{\pi}_{\tilde{W}}^U(x_l))^2 \right) \right)^{1/2}. \tag{10}
\end{aligned}$$

Particularly, if $w = (1/n, \dots, 1/n)$, then (10) is reduced to normalized Euclidean distance which is given as follows:

$$\begin{aligned}
D_\gamma(\tilde{V}, \tilde{W}) &= \left(\frac{1}{4n} \sum_{l=1}^n w_l \left((\tilde{\mu}_{\tilde{V}}^L(x_l) - \tilde{\mu}_{\tilde{W}}^L(x_l))^2 \right. \right. \\
&\quad + (\tilde{\mu}_{\tilde{V}}^U(x_l) - \tilde{\mu}_{\tilde{W}}^U(x_l))^2 + (\tilde{\nu}_{\tilde{V}}^L(x_l) - \tilde{\nu}_{\tilde{W}}^L(x_l))^2 \\
&\quad + (\tilde{\nu}_{\tilde{V}}^U(x_l) - \tilde{\nu}_{\tilde{W}}^U(x_l))^2 + (\tilde{\pi}_{\tilde{V}}^L(x_l) - \tilde{\pi}_{\tilde{W}}^L(x_l))^2 \\
&\quad \left. \left. + (\tilde{\pi}_{\tilde{V}}^U(x_l) - \tilde{\pi}_{\tilde{W}}^U(x_l))^2 \right) \right)^{1/2}. \tag{11}
\end{aligned}$$

3. Fuzzy Possibilistic C Means Clustering Algorithm

This section illustrates FPCM clustering algorithm proposed by Pal et al. [17] in 1997 to exploit the benefits of fuzzy and possibilistic modeling while circumventing their weaknesses. To correctly interpret the data substructure, FPCM clustering uses both memberships (relative typicality) and possibilities (absolute typicality). When we want to crisply label a data point, membership is a plausible choice as it is natural to assign a point to cluster whose prototype is closest to the point. On the other hand, while estimating the centroids, typicality is an important means for alleviating the undesirable effects of outliers. Here, the number of clusters is fixed a priori to a default value considering the dataset used in the application such that it is completely data driven. Generally it is advisable to avoid trivial clusters which may be either too large or small.

FPCM extends FCM clustering algorithm [17] by normalizing possibility values so that sum of possibilities of all data points in a cluster is 1. Although FPCM is much less prone to the problems of both FCM and PCM, the possibility values are very small when size of the dataset increases. Analogous to FCM clustering algorithm, the membership term in FPCM is a function of data point and all centroids. The typicality term in FPCM is a function of data point and cluster prototype alone. That is, the membership term is influenced by the positions of all cluster centers whereas typicality term is affected by only one. Incorporating the abovementioned facets the FPCM model is defined by the following optimization problem [17]:

$$\min_{(U,T,V)} \left\{ J_{m,\eta}(U, T, V; X) = \sum_{k=1}^p \sum_{i=1}^c (u_{ik}^m + t_{ik}^\eta) D_{ik}^2 \right\} \tag{12}$$

$$\text{subject to } m > 1, \quad \eta > 1, \quad 0 < u_{ik}, \quad t_{ik} < 1$$

$$\sum_{i=1}^c u_{ik} = 1 \quad \forall k \text{ i.e., } U \in M_{f_{cn}} \tag{13}$$

$$\sum_{k=1}^p t_{ik} = 1 \quad \forall i \text{ i.e., } T \in M_{f_{nc}}. \tag{14}$$

The transpose of admissible T 's is member of set $M_{f_{nc}}$. T is viewed as a typicality assignment of n objects to c clusters. The possibilistic term $\sum_{k=1}^p \sum_{i=1}^c t_{ik}^\eta D_{ik}^2$ distributes $\{t_{ik}\}$ with respect to all n data points, but not with respect to all c clusters. Under the usual conditions placed on c -means optimization problems, the first order necessary conditions for extrema of $J_{m,\eta}$ are stated in terms of the following theorem.

Theorem FPCM (see [17]). *If $D_{ik} = \|x_k - v_i\| > 0 \forall i$ and $k, m, \eta > 1$ and X contains at least c distinct data points, then $(U, T^t, V) \in M_{f_{cn}} \times M_{f_{nc}} \times \mathfrak{R}^p$ may minimize $J_{m,\eta}$ only if*

$$u_{ik} = \left(\sum_{j=1}^c \left(\frac{D_{ik}}{D_{jk}} \right)^{2/(m-1)} \right)^{-1}; \quad 1 \leq i \leq c, \quad 1 \leq k \leq n, \tag{15}$$

$$t_{ik} = \left(\sum_{j=1}^n \left(\frac{D_{ik}}{D_{ij}} \right)^{2/(\eta-1)} \right)^{-1}; \quad 1 \leq i \leq c, \quad 1 \leq k \leq n, \tag{16}$$

$$v_i = \frac{\sum_{k=1}^n (u_{ik}^m + t_{ik}^\eta) \mathbf{x}_k}{\sum_{k=1}^n (u_{ik}^m + t_{ik}^\eta)}; \quad 1 \leq i \leq c. \tag{17}$$

The proof of the above theorem follows from [38]. FPCM has the same type of singularity as FCM. FPCM does not suffer from the sensitivity problem that PCM seems to exhibit. Unfortunately, when the number of data points is large, the typicality values will be very small. Thus, after FPCM-AO algorithm [38] for approximating solutions to (12) based on iteration through (17) terminates, the typicality values may need to be scaled up. Conceptually, this is not different than scaling typicality as is done in PCM. While scaling seems to solve the small value problem which is caused by row

sum constraint on T , the scaled values do not possess any additional information about points in the data. Thus scaling $\{t_{ik}\}$ is an artificial fix for a mathematical drawback of FPCM.

4. Intuitionistic Fuzzy Possibilistic C Means Clustering Algorithms

In this section, we discuss intuitionistic fuzzy possibilistic C means clustering algorithms for IFSSs and IVIFSSs, respectively.

4.1. *Intuitionistic Fuzzy Possibilistic C Means Algorithm for IFSSs.* We develop the intuitionistic fuzzy possibilistic C means (IFPCM) model and corresponding algorithm for IFSSs. We take the basic distance measure D_j in (7) as proximity function of IFPCM; the objective function of IFPCM model can then be defined as follows:

$$\begin{aligned} \min_{(U,T,V)} \left\{ J_{m,\eta}(U, T, V; X) = \sum_{k=1}^p \sum_{i=1}^c (u_{ik}^m + t_{ik}^\eta) D_\alpha^2(Z_k, V_i) \right\} \\ \text{subject to } m > 1, \quad \eta > 1, \quad 0 < u_{ik}, \quad t_{ik} < 1 \\ \sum_{i=1}^c u_{ik} = 1 \quad \forall k \text{ i.e., } U \in M_{fcn} \\ \sum_{k=1}^p t_{ik} = 1 \quad \forall i \text{ i.e., } T^t \in M_{fnc}. \end{aligned} \quad (18)$$

Here $Z = \{Z_1, \dots, Z_p\}$ are p IFSSs each with n elements, c is the number of clusters ($1 \leq c \leq p$), and $V = \{V_1, \dots, V_c\}$ are the prototypical IFSSs, that is, centroid of the clusters. The parameter m is the fuzzy factor, u_{ik} is the membership degree of j th sample Z_j to the i th cluster, $U = (u_{ik})_{c \times p}$ is matrix of order $c \times p$, parameter η is the typicality factor, t_{ik} is the typicality of j th sample Z_j to the i th cluster, and $T = (t_{ik})_{c \times p}$ is typicality matrix.

To solve the optimization problem stated in (18), we make use of Lagrange multiplier method [39], which is discussed below. Considering

$$\begin{aligned} L = \sum_{k=1}^p \sum_{i=1}^c (u_{ik}^m + t_{ik}^\eta) D_\alpha^2(Z_k, V_i) - \sum_{k=1}^p \lambda_k \left(\sum_{i=1}^c u_{ik} - 1 \right) \\ - \sum_{k=1}^p \xi_k \left(\sum_{i=1}^c t_{ik} - 1 \right), \end{aligned} \quad (19)$$

where,

$$\begin{aligned} D_\alpha^2(Z_k, V_i) = \frac{1}{2} \sum_{l=1}^n \omega_l \left((\mu_{Z_k}(x_l) - \mu_{V_i}(x_l))^2 \right) \\ + (v_{Z_k}(x_l) - v_{V_i}(x_l))^2 \\ + (\pi_{Z_k}(x_l) - \pi_{V_i}(x_l))^2. \end{aligned} \quad (20)$$

Furthermore, $\forall 1 \leq i \leq c, 1 \leq k \leq p$; let

$$\begin{aligned} \frac{\partial L}{\partial u_{ik}} &= 0, \\ \frac{\partial L}{\partial t_{ik}} &= 0, \\ \frac{\partial L}{\partial \lambda_k} &= 0, \\ \frac{\partial L}{\partial \xi_k} &= 0. \end{aligned} \quad (21)$$

From the above system of equations, we have the following expressions:

$$u_{ik} = \frac{1}{\sum_{r=1}^c (D_\alpha(Z_k, V_i) / D_\alpha(Z_k, V_r))^{2/(m-1)}}; \quad 1 \leq i \leq c, \quad 1 \leq k \leq p \quad (22)$$

$$t_{ik} = \frac{1}{\sum_{r=1}^c (D_\alpha(Z_k, V_i) / D_\alpha(Z_k, V_r))^{2/(\eta-1)}}; \quad 1 \leq i \leq c, \quad 1 \leq k \leq p. \quad (23)$$

Now we proceed to compute $V_i; i = 1, \dots, c$, the prototypical IFSSs. Let us assume that

$$\frac{\partial L}{\partial \mu_{V_i}(x_l)} = \frac{\partial L}{\partial v_{V_i}(x_l)} = \frac{\partial L}{\partial \pi_{V_i}(x_l)} = 0. \quad (24)$$

From the above expression we have

$$\mu_{V_i}(x_l) = \frac{\sum_{k=1}^p (u_{ik}^m + t_{ik}^\eta) \mu_{Z_k}(x_l)}{\sum_{k=1}^p (u_{ik}^m + t_{ik}^\eta)}; \quad 1 \leq i \leq c, \quad 1 \leq l \leq n, \quad (25)$$

$$v_{V_i}(x_l) = \frac{\sum_{k=1}^p (u_{ik}^m + t_{ik}^\eta) v_{Z_k}(x_l)}{\sum_{k=1}^p (u_{ik}^m + t_{ik}^\eta)}; \quad 1 \leq i \leq c, \quad 1 \leq l \leq n, \quad (26)$$

$$\pi_{V_i}(x_l) = \frac{\sum_{k=1}^p (u_{ik}^m + t_{ik}^\eta) \pi_{Z_k}(x_l)}{\sum_{k=1}^p (u_{ik}^m + t_{ik}^\eta)}; \quad 1 \leq i \leq c, \quad 1 \leq l \leq n. \quad (27)$$

For simplicity, we define weighted average operator for IFSSs as follows.

Let $A = \{A_1, \dots, A_p\}$ be a set of IFSSs each with n elements; let $\omega = \{\omega_1, \dots, \omega_p\}$ be a set of weights for IFSSs, respectively, with $\sum_{j=1}^p \omega_j = 1$; and then the weighted operator f is defined as

$$\begin{aligned} f(A, \omega) \\ = \left\{ \left\langle x_l, \sum_{j=1}^p \omega_j \mu_{A_j}(x_l), \sum_{j=1}^p \omega_j v_{A_j}(x_l) \right\rangle \mid 1 \leq l \leq n \right\}. \end{aligned} \quad (28)$$

According to (25) to (28), if we assume

$$\omega^{(i)} = \left\{ \frac{(u_{i1} + t_{i1})}{\sum_{k=1}^p (u_{ik} + t_{ik})}, \dots, \frac{(u_{ip} + t_{ip})}{\sum_{k=1}^p (u_{ik} + t_{ik})} \right\}; \quad (29)$$

$$1 \leq i \leq c,$$

the prototypical IFSs $V = \{V_1, \dots, V_c\}$ of the IFPCM model can be computed as follows:

$$V_i = f(Z, \omega^{(i)})$$

$$= \left\{ \left\langle x_l, \sum_{j=1}^p \omega_j^{(i)} \mu_{Z_j}(x_l), \sum_{j=1}^p \omega_j^{(i)} \nu_{Z_j}(x_l) \right\rangle \mid 1 \leq l \leq n \right\},$$

$$1 \leq i \leq c. \quad (30)$$

Since the above equations (22), (23), and (30) are computationally interdependent, we exploit an iterative procedure similar to the FPCM algorithm to solve these equations. The steps of algorithm are as follows.

IFPCM Algorithm

Step 1. Initialize the seed values $V(0)$; let $x = 0$ and set $\varepsilon > 0$.

Step 2(i). Calculate $U(x) = (u_{ik}(x))_{c \times p}$, where

- (a) if $\forall k, r, D_\alpha(Z_k, V_r(x)) > 0$, then $u_{ik}(x) = 1 / (\sum_{r=1}^c (D_\alpha(Z_k, V_r(x)) / D_\alpha(Z_k, V_r(x)))^{2/(m-1)})$; $1 \leq i \leq c, 1 \leq k \leq p$,
- (b) if $\exists k, r$ such that $D_\alpha(Z_k, V_r(x)) = 0$, then let $u_{rk}(x) = 1$ and $u_{ik}(x) = 0 \forall i \neq r$.

Step 2(ii). Calculate $T(x) = (t_{ik}(x))_{c \times p}$, where

- (a) if $\forall k, r, D_\alpha(Z_k, V_r(x)) > 0$, then $t_{ik}(x) = 1 / (\sum_{r=1}^c (D_\alpha(Z_k, V_r(x)) / D_\alpha(Z_k, V_r(x)))^{2/(\eta-1)})$; $1 \leq i \leq c, 1 \leq k \leq p$,
- (b) if $\exists k, r$ such that $D_\alpha(Z_k, V_r(x)) = 0$, then let $t_{rk}(x) = 1$ and $t_{ik}(x) = 0 \forall i \neq r$.

Step 3. Calculate $V(x+1) = \{V_1(x+1), \dots, V_c(x+1)\}$, where

$$V_i(x+1) = f(Z, \omega^{(i)}(x+1)), \quad 1 \leq i \leq c \quad (31)$$

$$\omega^{(i)}(x+1)$$

$$= \left\{ \frac{(u_{i1}(x) + t_{i1}(x))}{\sum_{k=1}^p (u_{ik}(x) + t_{ik}(x))}, \dots, \frac{(u_{ip}(x) + t_{ip}(x))}{\sum_{k=1}^p (u_{ik}(x) + t_{ik}(x))} \right\};$$

$$1 \leq i \leq c. \quad (32)$$

Step 4. If $\sum_{i=1}^c (D_\alpha(V_i(x), V_i(x+1))/c) < \varepsilon$, then go to Step 5; otherwise, let $x = x + 1$, and return to Step 2.

Step 5. End

The pseudocode of the IFPCM algorithm is given in Algorithm 1.

4.2. Interval Valued Intuitionistic Fuzzy Possibilistic C Means Algorithm for IVIFSs. If the collected data are expressed as IVIFSs, then we extend IFPCM to interval valued intuitionistic fuzzy possibilistic C means (IVIFPCM) model. We take the basic distance measure D_γ in (10) as the proximity function of the IVIFPCM. The objective function of IVIFPCM model can be defined as follows:

$$\min_{(U, T, \tilde{V})} \left\{ J_{m, \eta}(U, T, \tilde{V}; X) = \sum_{k=1}^p \sum_{i=1}^c (u_{ik}^m + t_{ik}^\eta) D_\gamma^2(\tilde{Z}_k, \tilde{V}_i) \right\}$$

subject to $m > 1, \quad \eta > 1, \quad 0 < u_{ik}, \quad t_{ik} < 1$ (33)

$$\sum_{i=1}^c u_{ik} = 1 \quad \forall k \text{ i.e., } U \in M_{fcn} \quad (34)$$

$$\sum_{k=1}^p t_{ik} = 1 \quad \forall i \text{ i.e., } T^t \in M_{fnc}. \quad (35)$$

Here $\tilde{Z} = \{\tilde{Z}_1, \dots, \tilde{Z}_p\}$ are p IVIFSs each with n elements, c is the number of clusters ($1 < c < p$), and $\tilde{V} = \{\tilde{V}_1, \dots, \tilde{V}_c\}$ are the prototypical IVIFSs, that is, centroids of the clusters. The parameter m is the fuzzy factor, u_{ik} is the membership degree of j th sample \tilde{Z}_j to the i th cluster, $U = (u_{ik})_{c \times p}$ is matrix of order $c \times p$, parameter η is the typicality factor, t_{ik} is the typicality of j th sample \tilde{Z}_j to the i th cluster, and $T = (t_{ik})_{c \times p}$ is typicality matrix.

To solve the optimization problem stated in (30) to (35), we make use of Lagrange multiplier method [39], which is discussed below. Considering

$$L = \sum_{k=1}^p \sum_{i=1}^c (u_{ik}^m + t_{ik}^\eta) D_\gamma^2(\tilde{Z}_k, \tilde{V}_i) - \sum_{k=1}^p \lambda_k \left(\sum_{i=1}^c u_{ik} - 1 \right) - \sum_{k=1}^p \xi_k \left(\sum_{i=1}^c t_{ik} - 1 \right), \quad (36)$$

where,

$$D_\gamma^2(\tilde{Z}_k, \tilde{V}_i) = \frac{1}{4} \sum_{l=1}^n w_l \left(\left(\tilde{\mu}_{\tilde{Z}_k}^L(x_l) - \tilde{\mu}_{\tilde{V}_i}^L(x_l) \right)^2 + \left(\tilde{\mu}_{\tilde{Z}_k}^U(x_l) - \tilde{\mu}_{\tilde{V}_i}^U(x_l) \right)^2 + \left(\tilde{\nu}_{\tilde{Z}_k}^L(x_l) - \tilde{\nu}_{\tilde{V}_i}^L(x_l) \right)^2 + \left(\tilde{\nu}_{\tilde{Z}_k}^U(x_l) - \tilde{\nu}_{\tilde{V}_i}^U(x_l) \right)^2 + \left(\tilde{\pi}_{\tilde{Z}_k}^L(x_l) - \tilde{\pi}_{\tilde{V}_i}^L(x_l) \right)^2 + \left(\tilde{\pi}_{\tilde{Z}_k}^U(x_l) - \tilde{\pi}_{\tilde{V}_i}^U(x_l) \right)^2 \right).$$
(37)

Given an unlabeled dataset $X = \{x_1, \dots, x_n\}$, partition X into $1 < c < n$ clusters such that objective function $J_{m,\eta}(U, T, V; X)$ is minimized

(1) **Input:** Consider the seed values $V(0)$ and assume $x = 0$ and set $\varepsilon > 0$

(2) **Output:** Generate clusters using the IFPCM clustering algorithm for IFSs

(3) **begin procedure**

(4) **repeat**

(5) calculate $U(x) = (u_{ik}(x))_{c \times p}$

(6) **begin**

(7) **if** $(\forall k, r, D_\alpha(Z_k, V_r(x)) > 0)$ **then**

(8)
$$u_{ik}(x) = \frac{1}{\sum_{r=1}^c (D_\alpha(Z_k, V_i(x)) / D_\alpha(Z_k, V_r(x)))^{2/(m-1)}}; \quad 1 \leq i \leq c, 1 \leq k \leq p$$

(9) **end if**

(10) **if** $(\exists k, r, D_\alpha(Z_k, V_r(x)) = 0)$ **then**

(11) $u_{rk}(x) = 1$ and $u_{ik}(x) = 0 \forall i \neq r$

(12) **end if**

(13) **end**

(14) calculate $T(x) = (t_{ik}(x))_{c \times p}$

(15) **begin**

(16) **if** $(\forall k, r, D_\alpha(Z_k, V_r(x)) > 0)$ **then**

(17)
$$t_{ik}(x) = \frac{1}{\sum_{r=1}^c (D_\alpha(Z_k, V_i(x)) / D_\alpha(Z_k, V_r(x)))^{2/(\eta-1)}}; \quad 1 \leq i \leq c, 1 \leq k \leq p$$

(18) **end if**

(19) **if** $(\exists k, r, D_\alpha(Z_k, V_r(x)) = 0)$ **then**

(20) $t_{rk}(x) = 1$ and $t_{ik}(x) = 0 \forall i \neq r$

(21) **end if**

(22) **end**

(23) calculate $V(x+1) = \{V_1(x+1), \dots, V_c(x+1)\}; V_i(x+1) = f(Z, \omega^{(i)}(x+1)), 1 \leq i \leq c; \omega^{(i)}(x+1) = \{(u_{i1}(x) + t_{i1}(x)) / (\sum_{k=1}^p (u_{ik}(x) + t_{ik}(x))), \dots, (u_{ip}(x) + t_{ip}(x)) / (\sum_{k=1}^p (u_{ik}(x) + t_{ik}(x)))\}, 1 \leq i \leq c$

(24) $x = x + 1$

(25) **until** $(\sum_{i=1}^c (D_\alpha(V_i(x), V_i(x+1))) / c) < \varepsilon)$

(26) **end procedure**

ALGORITHM 1

Similar to IFPCM model, we establish the system of partial differential functions of L as follows:

$$\begin{aligned} \frac{\partial L}{\partial u_{ik}} &= 0; \quad 1 \leq i \leq c, \quad 1 \leq k \leq p, \\ \frac{\partial L}{\partial t_{ik}} &= 0; \quad 1 \leq i \leq c, \quad 1 \leq k \leq p, \\ \frac{\partial L}{\partial \lambda_k} &= 0, \quad \frac{\partial L}{\partial \xi_k} = 0; \quad 1 \leq k \leq p, \\ \frac{\partial L}{\partial \mu_{\tilde{V}_i}^L(x_l)} &= \frac{\partial L}{\partial v_{\tilde{V}_i}^L(x_l)} = \frac{\partial L}{\partial \pi_{\tilde{V}_i}^L(x_l)} = 0, \\ \frac{\partial L}{\partial \mu_{\tilde{V}_i}^U(x_l)} &= \frac{\partial L}{\partial v_{\tilde{V}_i}^U(x_l)} = \frac{\partial L}{\partial \pi_{\tilde{V}_i}^U(x_l)} = 0 \\ & \quad 1 \leq i \leq c, \quad 1 \leq l \leq p. \end{aligned}$$

The solution for the above system of equations is

$$u_{ik} = \frac{1}{\sum_{r=1}^c (D_\gamma(\tilde{Z}_k, \tilde{V}_i) / D_\gamma(\tilde{Z}_k, \tilde{V}_r))^{2/(m-1)}}; \quad 1 \leq i \leq c, \quad 1 \leq k \leq p \quad (39)$$

$$t_{ik} = \frac{1}{\sum_{r=1}^c (D_\gamma(\tilde{Z}_k, \tilde{V}_i) / D_\gamma(\tilde{Z}_k, \tilde{V}_r))^{2/(\eta-1)}}; \quad 1 \leq i \leq c, \quad 1 \leq k \leq p, \quad (40)$$

$$\begin{aligned} \tilde{V}_i &= \tilde{f}(\tilde{Z}, \omega^{(i)}) \\ &= \left\{ \left\langle x_l, \left[\sum_{j=1}^p \omega_j^{(i)} \tilde{\mu}_{\tilde{Z}_j}^L(x_l), \sum_{j=1}^p \omega_j^{(i)} \tilde{\mu}_{\tilde{Z}_j}^U(x_l) \right], \right. \right. \\ & \quad \left. \left. \left[\sum_{j=1}^p \omega_j^{(i)} \tilde{v}_{\tilde{Z}_j}^L(x_l), \sum_{j=1}^p \omega_j^{(i)} \tilde{v}_{\tilde{Z}_j}^U(x_l) \right] \right\rangle \mid 1 \leq l \leq n \right\}, \\ & \quad 1 \leq i \leq c, \quad (41) \end{aligned}$$

where

$$\omega^{(i)} = \left\{ \frac{(u_{i1} + t_{i1})}{\sum_{k=1}^p (u_{ik} + t_{ik})}, \dots, \frac{(u_{ip} + t_{ip})}{\sum_{k=1}^p (u_{ik} + t_{ik})} \right\}; \quad 1 \leq i \leq c. \quad (42)$$

Because (41) and (42) are computationally interdependent, we exploit a similar iteration procedure as follows.

IVIFPCM Algorithm

Step 1. Initialize the seed values $\tilde{V}(0)$; let $x = 0$ and set $\varepsilon > 0$.

Step 2(i). Calculate $U(x) = (u_{ik}(x))_{c \times p}$, where

$$(c) \text{ if } \forall k, r, D_\gamma(\tilde{Z}_k, \tilde{V}_r(x)) > 0, \text{ then } u_{ik}(x) = \frac{1}{(\sum_{r=1}^c (D_\gamma(\tilde{Z}_k, \tilde{V}_i(x))/D_\gamma(\tilde{Z}_k, \tilde{V}_r(x)))^{2/(m-1)})};$$

$$1 \leq i \leq c, 1 \leq k \leq p.$$

$$(d) \text{ if } \exists k, r \text{ such that } D_\gamma(\tilde{Z}_k, \tilde{V}_r(x)) = 0, \text{ then let } u_{rk}(x) = 1 \text{ and } u_{ik}(x) = 0 \forall i \neq r.$$

Step 2(ii). Calculate $T(x) = (t_{ik}(x))_{c \times p}$, where

$$(c) \text{ if } \forall k, r, D_\gamma(\tilde{Z}_k, \tilde{V}_r(x)) > 0, \text{ then } t_{ik}(x) = \frac{1}{(\sum_{r=1}^c (D_\gamma(\tilde{Z}_k, \tilde{V}_i(x))/D_\gamma(\tilde{Z}_k, \tilde{V}_r(x)))^{2/(\eta-1)})};$$

$$1 \leq i \leq c, 1 \leq k \leq p,$$

$$(d) \text{ if } \exists k, r \text{ such that } D_\gamma(\tilde{Z}_k, \tilde{V}_r(x)) = 0, \text{ then let } t_{rk}(x) = 1 \text{ and } t_{ik}(x) = 0 \forall i \neq r.$$

Step 3. Calculate $\tilde{V}(x+1) = \{\tilde{V}_1(x+1), \dots, \tilde{V}_c(x+1)\}$, where

$$\tilde{V}_i(x+1) = \tilde{f}(\tilde{Z}, \omega^{(i)}(x+1)), \quad 1 \leq i \leq c$$

$$\omega^{(i)}(x+1)$$

$$= \left\{ \frac{(u_{i1}(x) + t_{i1}(x))}{\sum_{k=1}^p (u_{ik}(x) + t_{ik}(x))}, \dots, \frac{(u_{ip}(x) + t_{ip}(x))}{\sum_{k=1}^p (u_{ik}(x) + t_{ik}(x))} \right\};$$

$$1 \leq i \leq c. \quad (43)$$

Step 4. If $\sum_{i=1}^c (D_\gamma(\tilde{V}_i(x), \tilde{V}_i(x+1))/c) < \varepsilon$, then go to Step 5; otherwise let $x = x + 1$, and return to Step 2.

Step 5. End

The pseudocode of the IVIFPCM algorithm is given in Algorithm 2.

5. Experimental Results

In this section, we enumerate the results of experiments performed on both real world and simulated datasets [32] in order to demonstrate the effectiveness of IFPCM clustering algorithm. IFPCM algorithm is implemented through MATLAB. We first explain the steps of the algorithm by the use of some experimental data which is evaluated through cluster validity measures. Next, the algorithm is applied to some classification datasets, that is, data with labeled patterns in order to examine its clustering accuracy.

5.1. Application of IFPCM Algorithm on Experimental Data. The parameters set in IFPCM algorithm are shown in Table 1. It is to be noted that if $\pi(x) = 0 \forall x \in X$, then IFPCM is reduced to FPCM algorithm. Hence, we present a comparative performance of both the algorithms.

The experimental data used here is investment portfolio dataset which contains information regarding ten investments at the disposal of the investor to invest some money to be classified in ICICI prudential financial services, India. Let

TABLE 1: IFPCM parameters.

Parameter	Description
f	Name of input file
c	Number of clusters (default value = 3)
m	Fuzzy factor (default value = 2)
η	Typicality factor (default value = 2)
w	Type of sample weights (default value = 0 (equal); user specified value = 1)
s	Type of initial centroids (default value = 0 (random); user specified value = 1)
i	Maximum number of iterations till convergence (default value = 100)
t	Threshold for iterations stoppage (default value = 0.001)

$I_i; i = 1, \dots, 10$ be the investments described by six attributes, namely a_1 : investment price; a_2 : advance mobilization; a_3 : time period; a_4 : return on investment; a_5 : risk factor; a_6 : security factor. The weight vector of these attributes is $w = (0.20, 0.10, 0.30, 0.15, 0.10, 0.15)$. The characteristics of ten investments under six attributes are represented by IFSs in Table 2. Simulated datasets are used for comparing with the experimental data. We assume that there are three classes in the simulated dataset, $C_i; i = 1, 2, 3$. The number of IFSs in each class is considered as 300. The different classes have different IFSs which are characterized as follows:

- (a) IFSs in C_1 have relatively high and positive scores,
- (b) IFSs in C_2 have relatively high and uncertain scores,
- (c) IFSs in C_3 have relatively high and negative scores.

Considering this, we generate simulated dataset as follows:

- (a) $\mu(x) \sim U(0.7, 1) \wedge v(x) + \pi(x) \sim U(0, 1 - \mu(x)) \forall x \in C_1$,
- (b) $v(x) \sim U(0.7, 1) \wedge \mu(x) + \pi(x) \sim U(0, 1 - v(x)) \forall x \in C_2$,
- (c) $\pi(x) \sim U(0.7, 1) \wedge \mu(x) + v(x) \sim U(0, 1 - \pi(x)) \forall x \in C_3$.

Here, $U(a, b)$ is the uniform distribution on the interval $[a, b]$. We thus generate a simulated dataset which consists of 3 classes comprising 900 IFSs.

5.1.1. Cluster Validity Measures. In IFPCM algorithm big challenge lies in setting the parameter c , that is, the number of clusters. To resolve this, we use two relative measures for fuzzy cluster validity mentioned in [40], namely, partition coefficient (PC) and classification entropy (CE). The descriptions of these two measures are given in Table 3. In PC and CE p is number of samples in the dataset.

5.1.2. IFPCM Algorithm on Investment Portfolio Dataset. IFPCM algorithm is used to cluster ten investments $I_i; i = 1, \dots, 10$, involving the following steps.

Step 1. Let $c = 3$ and $\varepsilon = 0.005$. Now randomly select initial centroids $V(0)$ from the dataset:

$$V(0) = \begin{bmatrix} I_9 \\ I_{10} \\ I_7 \end{bmatrix}. \quad (44)$$

Given an unlabeled dataset $X = \{x_1, \dots, x_n\}$, partition X into $1 < c < n$ clusters such that objective function $J_{m,\eta}(U, T, V; X)$ is minimized

- (1) **Input:** Consider the seed values $\tilde{V}(0)$ and assume $x = 0$ and set $\varepsilon > 0$
- (2) **Output:** Generate clusters using the IFPCM clustering algorithm for IVIFSs
- (3) **begin procedure**
- (4) **repeat**
- (5) calculate $U(x) = (u_{ik}(x))_{c \times p}$
- (6) **begin**
- (7) **if** $(\forall k, r, D_\gamma(\tilde{Z}_k, \tilde{V}_r(x)) > 0)$ **then**
- (8)
$$u_{ik}(x) = \frac{1}{\sum_{r=1}^c (D_\gamma(\tilde{Z}_k, \tilde{V}_i(x)) / D_\gamma(\tilde{Z}_k, \tilde{V}_r(x)))^{2/(m-1)}}; \quad 1 \leq i \leq c, 1 \leq k \leq p$$
- (9) **end if**
- (10) **if** $(\exists k, r, D_\gamma(\tilde{Z}_k, \tilde{V}_r(x)) = 0)$ **then**
- (11) $u_{rk}(x) = 1$ and $u_{ik}(x) = 0 \forall i \neq r$
- (12) **end if**
- (13) **end**
- (14) calculate $T(x) = (t_{ik}(x))_{c \times p}$
- (15) **begin**
- (16) **if** $(\forall k, r, D_\gamma(\tilde{Z}_k, \tilde{V}_r(x)) > 0)$ **then**
- (17)
$$t_{ik}(x) = \frac{1}{\sum_{r=1}^c (D_\gamma(\tilde{Z}_k, \tilde{V}_i(x)) / D_\gamma(\tilde{Z}_k, \tilde{V}_r(x)))^{2/(\eta-1)}}; \quad 1 \leq i \leq c, 1 \leq k \leq p$$
- (18) **end if**
- (19) **if** $(D_\gamma(\tilde{Z}_k, \tilde{V}_r(x)) = 0)$ **then**
- (20) $t_{rk}(x) = 1$ and $t_{ik}(x) = 0 \forall i \neq r$
- (21) **end if**
- (22) **end**
- (23) calculate $\tilde{V}(x+1) = \{\tilde{V}_1(x+1), \dots, \tilde{V}_c(x+1)\}; \tilde{V}_i(x+1) = \tilde{f}(\tilde{Z}, \omega^{(i)}(x+1)), 1 \leq i \leq c; \omega^{(i)}(x+1) = \{(u_{i1}(x) + t_{i1}(x)) / (\sum_{k=1}^p (u_{ik}(x) + t_{ik}(x))), \dots, (u_{ip}(x) + t_{ip}(x)) / (\sum_{k=1}^p (u_{ik}(x) + t_{ik}(x)))\}, 1 \leq i \leq c$
- (24) $x = x + 1$
- (25) **until** $(\sum_{i=1}^c (D_\gamma(\tilde{V}_i(x), \tilde{V}_i(x+1))) / c) < \varepsilon$
- (26) **end procedure**

ALGORITHM 2

Step 2(i). Calculate the membership degrees and centroids iteratively. According to (15), we have

$$U(0) = \begin{bmatrix} 0.4012 & 0.3179 & 0.2812 \\ 0.2155 & 0.2527 & 0.5336 \\ 0.2896 & 0.2312 & 0.4806 \\ 0.8969 & 0.0546 & 0.0512 \\ 0.1666 & 0.6313 & 0.2033 \\ 0.3196 & 0.3906 & 0.2914 \\ 0.1796 & 0.2139 & 0.6079 \\ 0.0000 & 0.0000 & 1.0000 \\ 1.0000 & 0.0000 & 0.0000 \\ 0.0000 & 1.0000 & 0.0000 \end{bmatrix}. \quad (45)$$

Step 2(ii). Calculate the typicality degrees and centroids iteratively. According to (16), we have

$$T(0) = \begin{bmatrix} 0.4215 & 0.3286 & 0.2816 \\ 0.2269 & 0.2626 & 0.5439 \\ 0.2899 & 0.2415 & 0.4809 \\ 0.8996 & 0.0545 & 0.0412 \\ 0.1569 & 0.6416 & 0.2133 \\ 0.3369 & 0.3909 & 0.2813 \\ 0.1696 & 0.2239 & 0.6286 \\ 0.0000 & 1.0000 & 1.0000 \\ 1.0000 & 0.0000 & 1.0000 \\ 1.0000 & 1.0000 & 0.0000 \end{bmatrix}. \quad (46)$$

Step 3. According to (17), we update the centroids as follows:

$$V(1) = \begin{bmatrix} \langle 0.366, 0.386 \rangle & \langle 0.839, 0.086 \rangle & \langle 0.786, 0.156 \rangle & \langle 0.626, 0.186 \rangle & \langle 0.209, 0.707 \rangle & \langle 0.196, 0.739 \rangle \\ \langle 0.760, 0.152 \rangle & \langle 0.679, 0.139 \rangle & \langle 0.586, 0.269 \rangle & \langle 0.496, 0.206 \rangle & \langle 0.709, 0.226 \rangle & \langle 0.509, 0.466 \rangle \\ \langle 0.679, 0.212 \rangle & \langle 0.575, 0.209 \rangle & \langle 0.669, 0.169 \rangle & \langle 0.366, 0.520 \rangle & \langle 0.386, 0.569 \rangle & \langle 0.669, 0.139 \rangle \end{bmatrix}. \quad (47)$$

Step 4. We made a check whether to stop the iterations:

$$\sum_{i=1}^3 \frac{D_{\alpha}(V_i(0), V_i(1))}{3} = 0.086 > 0.005. \quad (48)$$

Since this value exceeds the chosen threshold value, we continue with the next iteration.

When $x = 1$,

$$U(1) = \begin{bmatrix} 0.3866 & 0.3333 & 0.2755 \\ 0.0866 & 0.1022 & 0.8077 \\ 0.1339 & 0.1239 & 0.7333 \\ 0.8386 & 0.0333 & 0.0355 \\ 0.1027 & 0.7525 & 0.2222 \\ 0.2899 & 0.3786 & 0.2779 \\ 0.0655 & 0.8002 & 0.7530 \\ 0.1000 & 0.1955 & 0.6336 \\ 0.7933 & 0.0550 & 0.0500 \\ 0.0720 & 0.7120 & 0.1012 \end{bmatrix},$$

$$T(1) = \begin{bmatrix} 0.3716 & 0.3065 & 0.2515 \\ 0.2024 & 0.2562 & 0.5233 \\ 0.2698 & 0.2210 & 0.4620 \\ 0.8696 & 0.0525 & 0.0312 \\ 0.1455 & 0.6045 & 0.2030 \\ 0.3036 & 0.3799 & 0.2715 \\ 0.1595 & 0.2138 & 0.6185 \\ 0.1000 & 0.8888 & 0.7775 \\ 0.6867 & 0.0444 & 0.0505 \\ 0.0577 & 0.6600 & 0.0005 \end{bmatrix}, \quad (49)$$

$$V(2) = \begin{bmatrix} \langle 0.354, 0.385 \rangle & \langle 0.838, 0.075 \rangle & \langle 0.750, 0.145 \rangle & \langle 0.616, 0.139 \rangle & \langle 0.185, 0.706 \rangle & \langle 0.170, 0.725 \rangle \\ \langle 0.755, 0.142 \rangle & \langle 0.635, 0.174 \rangle & \langle 0.533, 0.222 \rangle & \langle 0.396, 0.160 \rangle & \langle 0.625, 0.222 \rangle & \langle 0.472, 0.399 \rangle \\ \langle 0.536, 0.176 \rangle & \langle 0.512, 0.170 \rangle & \langle 0.650, 0.142 \rangle & \langle 0.279, 0.515 \rangle & \langle 0.327, 0.525 \rangle & \langle 0.596, 0.120 \rangle \end{bmatrix},$$

$$\sum_{i=1}^3 \frac{D_{\alpha}(V_i(1), V_i(2))}{3} = 0.009 > 0.005.$$

Since this value exceeds the chosen threshold value, we continue with the next iteration.

When $x = 2$,

$$U(2) = \begin{bmatrix} 0.3864 & 0.3332 & 0.2752 \\ 0.0865 & 0.1021 & 0.8072 \\ 0.1337 & 0.1236 & 0.7331 \\ 0.8385 & 0.0330 & 0.0354 \\ 0.1026 & 0.7524 & 0.2221 \\ 0.2898 & 0.3784 & 0.2778 \\ 0.0653 & 0.8000 & 0.7529 \\ 0.1000 & 0.1954 & 0.6333 \\ 0.7932 & 0.0550 & 0.0499 \\ 0.0716 & 0.7118 & 0.1010 \end{bmatrix},$$

$$T(2) = \begin{bmatrix} 0.3715 & 0.3064 & 0.2512 \\ 0.2022 & 0.2560 & 0.5232 \\ 0.2697 & 0.2209 & 0.4619 \\ 0.8693 & 0.0524 & 0.0311 \\ 0.1454 & 0.6042 & 0.2028 \\ 0.3035 & 0.3798 & 0.2714 \\ 0.1593 & 0.2137 & 0.6184 \\ 0.9999 & 0.8886 & 0.7772 \\ 0.6862 & 0.0443 & 0.0504 \\ 0.0575 & 0.6599 & 0.0004 \end{bmatrix},$$

$$V(3) = \begin{bmatrix} \langle 0.352, 0.380 \rangle & \langle 0.837, 0.077 \rangle & \langle 0.752, 0.144 \rangle & \langle 0.614, 0.133 \rangle & \langle 0.180, 0.705 \rangle & \langle 0.172, 0.722 \rangle \\ \langle 0.765, 0.140 \rangle & \langle 0.632, 0.172 \rangle & \langle 0.532, 0.220 \rangle & \langle 0.389, 0.156 \rangle & \langle 0.623, 0.222 \rangle & \langle 0.470, 0.396 \rangle \\ \langle 0.528, 0.169 \rangle & \langle 0.510, 0.168 \rangle & \langle 0.647, 0.145 \rangle & \langle 0.276, 0.512 \rangle & \langle 0.326, 0.523 \rangle & \langle 0.586, 0.115 \rangle \end{bmatrix},$$

$$\sum_{i=1}^3 \frac{D_{\alpha}(V_i(2), V_i(3))}{3} = 0.003 < 0.005.$$

(50)

Since this value is less than the threshold value, we stop the iterations and calculate the values of $U(3)$ and $T(3)$ when $x = 3$:

$$U(3) = \begin{bmatrix} 0.3862 & 0.3330 & 0.2750 \\ 0.0665 & 0.1018 & 0.8069 \\ 0.1335 & 0.1236 & 0.7330 \\ 0.8380 & 0.0325 & 0.0352 \\ 0.1023 & 0.7516 & 0.2220 \\ 0.2886 & 0.3765 & 0.2769 \\ 0.0650 & 0.8002 & 0.7527 \\ 0.9999 & 0.1953 & 0.6331 \\ 0.7922 & 0.0545 & 0.0495 \\ 0.0712 & 0.7116 & 0.1009 \end{bmatrix},$$

$$T(3) = \begin{bmatrix} 0.3712 & 0.3050 & 0.2510 \\ 0.2020 & 0.2555 & 0.5226 \\ 0.2695 & 0.2205 & 0.4520 \\ 0.8690 & 0.0520 & 0.0310 \\ 0.1450 & 0.6040 & 0.2025 \\ 0.3033 & 0.3796 & 0.2710 \\ 0.1590 & 0.2135 & 0.6175 \\ 0.9666 & 0.8880 & 0.7769 \\ 0.6860 & 0.0442 & 0.0502 \\ 0.0572 & 0.6596 & 0.0002 \end{bmatrix}. \quad (51)$$

According to $U(4)$ and $T(4)$, cluster validation measures V_{PC} and V_{CE} are calculated as

$$V_{PC} = \frac{1}{3} \sum_{i=1}^3 \sum_{k=1}^{10} (u_{ik}^2 + t_{ik}^2) = 0.633,$$

$$V_{CE} = -\frac{1}{10} \sum_{i=1}^3 \sum_{k=1}^{10} (u_{ik} \log u_{ik} + t_{ik} \log t_{ik}) = 0.865. \quad (52)$$

If we further assume that $u_{ik} \geq 0.75$, $t_{ik} \geq 0.65 \Rightarrow I_j \in C_i$ ($1 \leq j \leq 10$, $1 \leq i \leq 3$), where C_i denotes cluster i , then we have clusters as shown in Table 4.

5.1.3. Convergence of IFPCM Algorithm. Now, we proceed to investigate the convergence of IFPCM algorithm on investment portfolio dataset. The movements of objective function values $J_{m,\eta}(U, T, V; X)$ are shown in Figure 1 along the iterations. As evident from Figure 1, the IFPCM algorithm decreases the objective function value continuously by iterating two phases, namely, updating the membership and typicality degrees in (15) and (16) and updating prototypical IFSSs in (17). The IFPCM algorithm has lower computational

complexity as compared to other clustering algorithms [1–3, 5–7, 9]. The space and time complexities of IFPCM algorithm are $O(p(n+c) + cn)$ and $O(Icpn)$, where p is the number of samples, n is the number of IFSSs in sample, c is the number of clusters, and I is the maximum number of iterations preset for optimal value search process. Some advantages of IFPCM algorithm include simplicity and flexibility, information about the membership, and typicality degrees of samples to all clusters and relatively low computational complexity.

5.1.4. Comparative Performance of IFPCM and FPCM on Investment Portfolio Dataset. In this subsection, we present a comparative performance of IFPCM and FPCM algorithms. We first experiment IFPCM algorithm on the simulated dataset. Here, we set a series of c values in the range of 2 to 10 and compute V_{PC} and V_{CE} measures for each clustering result. The results are given in Table 5, where Obj is the objective function value after convergence of IFPCM algorithm. The optimal values of cluster validity measures are highlighted.

As evident from Table 5, when $c = 4$ V_{PC} reaches its optimal value 0.9664 (maximum) and V_{CE} also reaches its optimal value 0.1866 (minimum), this implies that both V_{PC} and V_{CE} are capable of finding optimal number of clusters, that is, c . However, this is not the case for objective function value. From Figure 2, as the number of clusters increases, Obj decreases continuously and finally reaches 1.1969 when $c = 4$. Hence, the usage of V_{PC} and V_{CE} is justified in the evaluation of clustering results produced by IFPCM algorithm. Next, we experiment FPCM algorithm on the simulated dataset for comparison purpose. The results are given in Table 6. The optimal values of cluster validity measures are highlighted. As indicated by V_{PC} and V_{CE} values in Table 6, FPCM algorithm prefers to cluster the modified simulated datasets into three clusters which are actually away from four true clusters in the data. In other words, FPCM algorithm cannot identify all four classes precisely. This further signifies the importance of uncertainty information in IFSSs.

5.2. Examining Clustering Accuracy of IFPCM Algorithm. To assess the ability of IFPCM algorithm to explore natural clusters in real world data, 11 classification and two clustering datasets with numerical attributes are chosen from the University of California at the Irvine Machine Learning Repository [41] and Knowledge Extraction based on Evolutionary Learning Repository [42]. In Table 7, these datasets are summarized.

The accuracy of IFPCM algorithm is adhered by removing the class labels of data before applying the algorithm.

TABLE 2: Investment portfolio dataset.

	a_1		a_2		a_3		a_4		a_5		a_6	
	$\mu_{I_i}(a_1)$	$\nu_{I_i}(a_1)$	$\mu_{I_i}(a_2)$	$\nu_{I_i}(a_2)$	$\mu_{I_i}(a_3)$	$\nu_{I_i}(a_3)$	$\mu_{I_i}(a_4)$	$\nu_{I_i}(a_4)$	$\mu_{I_i}(a_5)$	$\nu_{I_i}(a_5)$	$\mu_{I_i}(a_6)$	$\nu_{I_i}(a_6)$
I_1	0.30	0.40	0.20	0.70	0.50	0.50	0.80	0.10	0.40	0.50	0.20	0.70
I_2	0.40	0.30	0.50	0.10	0.60	0.20	0.20	0.70	0.30	0.60	0.70	0.20
I_3	0.40	0.20	0.60	0.10	0.80	0.10	0.20	0.60	0.30	0.70	0.50	0.20
I_4	0.30	0.50	0.90	0.00	0.90	0.10	0.70	0.20	0.20	0.90	0.20	0.90
I_5	0.90	0.20	0.70	0.20	0.70	0.10	0.50	0.10	0.90	0.20	0.50	0.70
I_6	0.40	0.30	0.30	0.60	0.20	0.60	0.70	0.10	0.50	0.40	0.30	0.60
I_7	0.50	0.40	0.50	0.20	0.70	0.20	0.30	0.70	0.30	0.70	0.60	0.10
I_8	0.90	0.10	0.70	0.20	0.70	0.10	0.40	0.50	0.40	0.60	0.80	0.00
I_9	0.40	0.40	1.00	0.20	0.90	0.20	0.70	0.20	0.20	0.70	0.20	0.80
I_{10}	0.90	0.10	0.90	0.00	0.60	0.30	0.50	0.20	0.80	0.10	0.60	0.40

TABLE 3: Description of two cluster validity criteria.

Validity criteria	Functional description	Optimal cluster number
PC	$V_{PC} = \frac{1}{P} \sum_{i=1}^c \sum_{k=1}^P (u_{ik}^2 + t_{ik}^2)$	$\underset{c}{\operatorname{argmax}} (V_{PC}, U, T, c)$
CE	$V_{CE} = -\frac{1}{P} \sum_{i=1}^c \sum_{k=1}^P (u_{ik} \log u_{ik} + t_{ik} \log t_{ik})$	$\underset{c}{\operatorname{argmin}} (V_{CE}, U, T, c)$

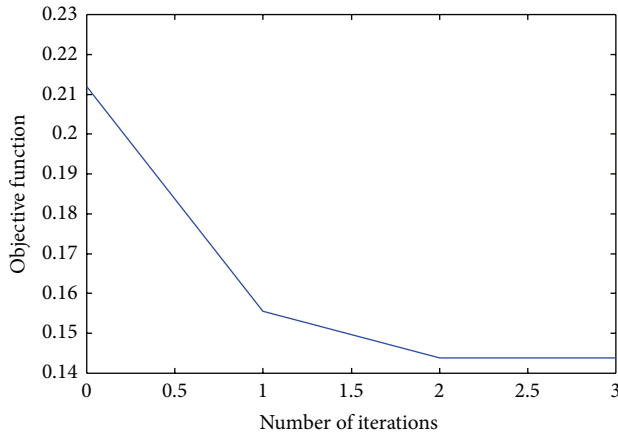


FIGURE 1: The convergence of IFPCM algorithm on investment portfolio dataset.

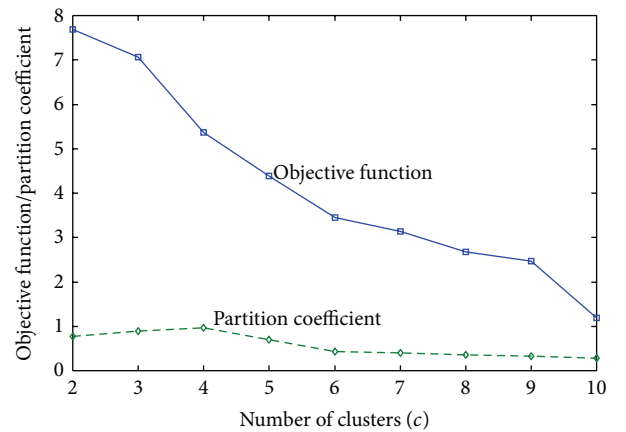
FIGURE 2: Comparative performance of Obj and V_{PC} given different c values.

TABLE 4: Clustering result of the investment portfolio dataset by IFPCM.

Instance	Cluster ID
I_6, I_7	1
I_4, I_8	2
I_2, I_5, I_9	3
I_1, I_3, I_{10}	4

Each attribute value of all datasets is rescaled to a unit interval $[0, 1]$ via linear transformation. The clustering results of the application of IFPCM algorithm on 11 classification datasets are shown in Table 8, where FPCM algorithm results as a benchmark fuzzy clustering method are also provided.

The threshold t for effectiveness measure is set to 0.1 for all the datasets, provided that at least two clusters are explored. For fairness of comparison between IFPCM and FPCM algorithms, the number of clusters that are needed by FPCM as a parameter for each dataset is set to the number of clusters that are explored by IFPCM. In this table, two super cells for each dataset are confusion matrices [43], which represent the clustering accuracy of IFPCM and FPCM algorithms on that data. In confusion matrix, cell c_{ij} contains the number of patterns with class label i which are grouped by cluster j . Accordingly, the cells in each row of actual class label are summed up to the number of patterns in that class. In addition, the summation of each column's cells represents the number of patterns in that cluster. Ideally, optimal clusters are

TABLE 5: IFPCM algorithm with different cluster numbers on investment portfolio dataset.

	c									
	2	3	4	5	6	7	8	9	10	
Obj	7.6896	7.0678	5.3655	4.3867	3.4464	3.1393	2.6695	2.4755	1.1969	
V_{PC}	0.7666	0.8986	0.9664	0.6999	0.4365	0.4050	0.3555	0.3333	0.2757	
V_{CE}	0.5766	0.4672	0.1866	0.7679	1.0009	1.3339	1.4755	1.6462	1.8689	

TABLE 6: FPCM algorithm with different cluster numbers on investment portfolio dataset.

	c									
	2	3	4	5	6	7	8	9	10	
V_{PC}	0.9866	0.6987	0.5366	0.4999	0.3566	0.2999	0.2555	0.2222	0.2009	
V_{CE}	0.0509	0.7000	1.1569	1.2767	1.5509	1.8750	2.0215	2.2554	2.2869	

TABLE 7: Datasets used for evaluating IFPCM algorithm.

Dataset	Number of attributes	Number of classes	Number of samples
Iris	4	3	150
Thyroid	5	3	215
Ecoli	7	8	336
Cancer (Breast)	9	2	684
Glass	9	6	214
Vowel	10	11	990
Wine	13	3	178
Vehicle	18	4	846
WDBC	30	2	569
Ionosphere	33	2	351
Sonar	60	2	208

achieved when patterns of each class are covered by only one cluster and each cluster just contains patterns of one class. Such a case occurred for the first class of the Iris dataset in both clustering methods. As evident from results in Table 8, the performance of IFPCM is better as compared to FPCM algorithm in all datasets.

Although confusion matrices for Vehicle and WDBC datasets show almost identical overall performance, the clustering accuracy of IFPCM algorithm for Iris, Thyroid, Cancer, Glass, and Sonar datasets is comparatively better. On the other hand, FPCM algorithm obtains better performance for Ecoli, Vowel, Wine, and Ionosphere datasets. IFPCM algorithm explores potential clusters that are embedded in datasets and needs only a distinguishing threshold t for effectiveness measure while the number of clusters in FPCM algorithm is provided in advance. The clusters obtained by FPCM algorithm convey no specific cognitive interpretation while those clusters explored by IFPCM algorithm are identified by intuitionistic measure. This intuitionistic interpretability which represents the clusters justifies the claim that IFPCM algorithm is more suitable for knowledge discovery in datasets. The IFPCM algorithm is more robust to outliers and noise in data. Moreover, the computational cost of IFPCM algorithm is higher than that of FPCM algorithm

as given in Table 9. Although some of the datasets which are used in the experiments are high dimensional, they are not too large. The application of IFPCM algorithm on large datasets consumes greater CPU time. Since, threshold t for effectiveness measure is set to 0.1 for all data given in Table 8, the number of clusters that are explored for multiclass datasets is less than their classes. Consequently, IFPCM algorithm needs a lower threshold to explore more clusters. Table 10 illustrates clustering results obtained for these datasets when threshold t is set to 0.01.

To compare the effectiveness of IFPCM algorithm with other fuzzy clustering methods, some recently developed algorithms have been considered and their results on some real world datasets are presented in Table 11. The performance of these methods is expressed in terms of pureness ratio which is the average pureness of clusters after cluster labeling that is based on maximum number of sample classes in each cluster. Along with FPCM algorithms, some other clustering algorithms that run on these datasets are FCM, PCM [8], α -cut FCM (AFCM) [18], entropy based fuzzy clustering (EFC) [19], fuzzy mixture of Student's t factor analyzers (FMSFA) [20], and fuzzy principal component analysis guided robust k -means (FPRk) [21]. The IFPCM algorithm maintains appreciable performance compared to

TABLE 8: Clustering results of IFPCM and FPCM algorithms.

Dataset	Actual class label	IFPCM number of cluster			FPCM number of cluster		
		1	2	3	1	2	3
Iris	1	50			50		
	2		50			47	3
	3		17	33		12	38
Thyroid	1	150			145	5	
	2	12	23			35	
	3	30			30		
Ecoli	1	139	4		139	4	
	2	17	60		15	62	
	3		2			2	
	4		2		1	1	
	5	2	33		3	32	
	6	3	17		5	15	
	7		5			5	
	8	13	39		25	27	
Cancer	1	20	219		22	217	
	2	439	6		438	7	
Glass	1	39	31		70		
	2	39	37		66	10	
	3	9	8		17		
	4	9	4		3	10	
	5	9			5	4	
	6	30			5	25	
Vowel	1	17	22	51		39	51
	2	12	5	73		42	48
	3	2	22	66		51	39
	4	2	55	33	4	65	21
	5	4	44	42	9	51	30
	6	20	42	28		66	24
	7	72	12	6	16	65	9
	8	69	16	5	2	58	30
	9	82	3	5	5	35	50
	10	12	45	33	10	50	30
	11	25	15	50		25	65
Wine	1	59			59		
	2	12	55	4	4	60	7
	3			48			48
Vehicle	1	196	3		196	3	
	2	102	115		98	119	
	3	162	56		162	56	
	4	96	116		95	117	
WDBC	1	46	166		26	186	
	2	350	7		345	12	
Ionosphere	1	116	109		156	69	
	2	30	96		36	90	
Sonar	1	50	47		39	58	
	2	54	57		50	61	

TABLE 9: Computational costs of IFPCM and FPCM algorithms.

Dataset	CPU time (in seconds)	
	IFPCM	FPCM
Iris	0.86	0.15
Thyroid	0.16	0.09
Ecoli	1.00	0.09
Cancer	5.50	0.10
Glass	0.99	0.09
Vowel	21.00	0.12
Wine	0.99	0.09
Vehicle	5.00	0.12
WDBC	2.10	0.13
Ionosphere	2.50	0.09
Sonar	1.60	0.16

other methods in terms of pureness ratio although this is not true for clustering accuracy. The specification of threshold t in IFPCM algorithm for effectiveness measure is more intuitionistic and less data dependent in nature.

6. Conclusion

In this paper, we have proposed IFPCM and IVIFPCM algorithms to cluster IFSs and IVIFSs, respectively. Both the algorithms are developed by integrating concepts of FPCM, IFs, IVIFSs, and basic distance measures. In interval valued intuitionistic fuzzy environments, the clustering algorithm has membership and nonmembership degrees as intervals rather than exact numbers. The algorithms overcome problems involved with membership values of objects to each cluster by generalizing degrees of membership of objects to each cluster. This is achieved by extending membership and nonmembership degrees with hesitancy degree. The algorithms also provide information about membership and typicality degrees of samples to all clusters. Experiments on both real world and simulated datasets show that IFPCM has some notable advantages over FPCM. IFPCM algorithm is simple and flexible. It generates valuable information and produces overlapped clusters where instances have different membership degrees in accordance with different real world applications. The algorithm has relatively lower computational complexity. It also takes into account inherent uncertainty in information captured by IFSs which becomes crucial for success of some clustering tasks. The evaluation of the algorithm is performed through cluster validity measures. The clustering accuracy of the algorithm is determined by classification datasets with labeled patterns. IFPCM maintains appreciable performance compared to other methods in terms of pureness ratio although this is not true for clustering accuracy. For multiclass datasets there is a chance for exploring fewer clusters than classes. This is handled by

TABLE 10: Clustering results of IFPCM algorithm on multiclass datasets.

Dataset	Actual class label	Number of cluster							
		1	2	3	4	5	6	7	8
Thyroid	1	150							
	2	12	23						
	3	9		21					
Ecoli	1	21	20	102					
	2	59	9	9					
	3	1		1					
	4	1		1					
	5	30	2	3					
	6	2	2	16					
	7	2		3					
	8	9	4	39					
Glass	1	46	24						
	2	39	25	12					
	3	12	5						
	4	2	3	8					
	5	5		4					
	6	5		24					
Vowel	1				54	9	27		
	2		24		45	9	12		
	3		16	33		35			6
	4		55	3	20				12
	5		33	14	17	5			21
	6	5	26	2	35		12		10
	7	25	5		55		5		
	8	39	2	3	25	3	3	3	12
	9	46			9	12	5	12	6
	10	15	36	4	12	4		3	16
	11					46	11	33	
Vehicle	1	115	2	82					
	2	64	114	39					
	3	72	60	86					
	4	60	119	33					

decreasing value of threshold for effectiveness measure. The specification of threshold is more intuitionistic and less data dependent in nature. For an unknown dataset, IFPCM must compute cluster accuracy measure for all potential clusters. A sudden drop in values should be considered as stopping criterion whereby the number of clusters is determined which can be explored. The different drawbacks of FPCM are

TABLE II: Comparative analysis of IFPCM algorithm with other Fuzzy clustering methods in terms of Purenness ratio.

Fuzzy clustering method	Parameters	Datasets				
		Iris	Wine	Thyroid	Cancer	Sonar
IFPCM	$t = 0.1$	93.62	93.86	98.55	98.16	55.39
FPCM	$t = 0.1$	91.60	93.09	96.46	96.28	53.36
FCM [7]	$c = 3, m = 2$	89.93	90.46	96.08	95.83	53.61
PCM [8]	$c = 3, m = 2$ $\varepsilon = 0.01$	80.80	91.42	87.78	77.17	52.72
AFCM [18]	$c = 3, m = 2$ $\varepsilon = 0.01$	89.79	94.68	69.46	73.90	56.47
EFC [19]	$\beta = 0.5 - 0.7$ $\gamma = 0 - 0.05m_t$	96.68	87.15	45.15	95.60	—
FMSFA [20]	$c = 3, q = 1$ $\lambda = 0.7 - 0.9$	98.72	96.26	91.44	94.96	—
FPRk [21]	$k = 3, \lambda = 0.1$ $\beta = 1.5$	79.60	96.50	96.18	—	—

effectively handled by possibilistic fuzzy C means (PFCM) model proposed by Pal et al. in 2005. Our future work entails development of IFs framework for PFCM.

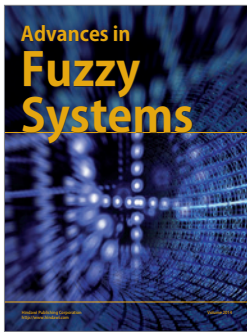
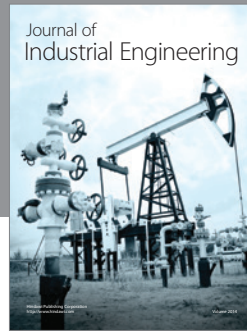
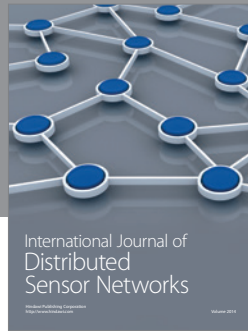
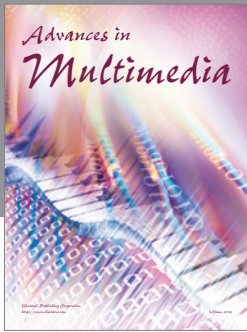
Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

References

- [1] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, John Wiley & Sons, New York, NY, USA, 2nd edition, 2000.
- [2] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1988.
- [3] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, 1999.
- [4] H. Frigui, "Simultaneous clustering and feature discrimination with applications," in *Advances in Fuzzy Clustering and Feature Discrimination with Applications*, pp. 285–312, John Wiley & Sons, New York, NY, USA, 2007.
- [5] B. S. Everitt, S. Landau, and M. Leese, *Cluster Analysis*, Oxford University Press, Oxford, UK, 2001.
- [6] W. Pedrycz, *Knowledge Based Clustering*, John Wiley & Sons, Hoboken, NJ, USA, 2005.
- [7] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, NY, USA, 1981.
- [8] R. Krishnapuram and J. M. Keller, "A possibilistic approach to clustering," *IEEE Transactions on Fuzzy Systems*, vol. 1, no. 2, pp. 98–110, 1993.
- [9] M. R. Anderberg, *Cluster Analysis for Applications*, Academic Press, New York, NY, USA, 1972.
- [10] F. D. A. T. de Carvalho, "Fuzzy c-means clustering methods for symbolic interval data," *Pattern Recognition Letters*, vol. 28, no. 4, pp. 423–437, 2007.
- [11] G. Beliakov and M. King, "Density based fuzzy c-means clustering of non-convex patterns," *European Journal of Operational Research*, vol. 173, no. 3, pp. 717–728, 2006.
- [12] M. Barni, V. Cappellini, and A. Mecocci, "Comments on "a possibilistic approach to clustering"," *IEEE Transactions on Fuzzy Systems*, vol. 4, no. 3, pp. 393–396, 1996.
- [13] H. Timm, C. Borgelt, C. Doring, and R. Kruse, "Fuzzy cluster analysis with cluster repulsion," in *Proceedings of the European Symposium in Intelligent Technologies*, Tenerife, Spain, 2001.
- [14] H. Timm and R. Kruse, "A modification to improve possibilistic fuzzy cluster analysis," in *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE '02)*, pp. 1460–1465, Honolulu, Hawaii, USA, May 2002.
- [15] H. Timm, C. Borgelt, C. Doring, and R. Kruse, "An extension to possibilistic fuzzy cluster analysis," *Fuzzy Sets and Systems*, vol. 147, no. 1, pp. 3–16, 2004.
- [16] D. E. Gustafson and W. C. Kessel, "Fuzzy clustering with a fuzzy covariance matrix," in *Proceedings of the IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes*, pp. 761–766, San Diego, Calif, USA, January 1979.
- [17] N. R. Pal, K. Pal, and J. C. Bezdek, "A mixed c-means clustering model," in *Proceedings of the 6th IEEE International Conference on Fuzzy Systems*, vol. 1, pp. 11–21, Barcelona, Spain, July 1997.
- [18] M.-S. Yang, K.-L. Wu, J.-N. Hsieh, and J. Y. Hsieh, "Alpha-cut implemented fuzzy clustering algorithms and switching regressions," *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, vol. 38, no. 3, pp. 588–603, 2008.
- [19] J. Yao, M. Dash, S. T. Tan, and H. Liu, "Entropy-based fuzzy clustering and fuzzy modeling," *Fuzzy Sets and Systems*, vol. 113, no. 3, pp. 381–388, 2000.
- [20] S. Chatzis and T. Varvarigou, "Factor analysis latent subspace modeling and robust fuzzy clustering using t-distributions," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 3, pp. 505–517, 2009.
- [21] K. Honda, A. Notsu, and H. Ichihashi, "Fuzzy PCA-guided robust k-means clustering," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 1, pp. 67–79, 2010.
- [22] K. T. Atanassov, "Intuitionistic fuzzy sets," *Fuzzy Sets and Systems*, vol. 20, no. 1, pp. 87–96, 1986.
- [23] K. T. Atanassov and G. K. Gargov, "Intuitionistic fuzzy logic," *Computing Research Academy of Bulgarian Sciences*, vol. 43, no. 3, pp. 9–12, 1990.
- [24] K. Atanassov and C. Georgiev, "Intuitionistic fuzzy prolog," *Fuzzy Sets and Systems*, vol. 53, no. 2, pp. 121–128, 1993.

- [25] E. Szmidt and J. Kacprzyk, "Intuitionistic fuzzy sets in group decision making," *Notes on Intuitionistic Fuzzy Sets*, vol. 2, no. 1, pp. 15–32, 1996.
- [26] S. K. De, R. Biswas, and A. R. Roy, "An application of intuitionistic fuzzy sets in medical diagnosis," *Fuzzy Sets and Systems*, vol. 117, no. 2, pp. 209–213, 2001.
- [27] K. T. Atanassov, "New operations defined over the intuitionistic fuzzy sets," *Fuzzy Sets and Systems*, vol. 61, no. 2, pp. 137–142, 1994.
- [28] K. T. Atanassov, *Intuitionistic Fuzzy Sets: Theory and Applications*, vol. 35 of *Studies in Fuzziness and Soft Computing*, Physica, Heidelberg, Germany, 1999.
- [29] K. Atanassov and G. Gargov, "Interval valued intuitionistic fuzzy sets," *Fuzzy Sets and Systems*, vol. 31, no. 3, pp. 343–349, 1989.
- [30] H. Bustince, F. Herrera, and J. Montero, *Fuzzy Sets and Their Extensions: Representation, Aggregation and Models*, Physica, Heidelberg, Germany, 2007.
- [31] H. M. Zhang, Z. S. Xu, and Q. Chen, "Clustering approach to intuitionistic fuzzy sets," *Control and Decision*, vol. 22, no. 8, pp. 882–888, 2007.
- [32] Z. S. Xu, J. Chen, and J. J. Wu, "Clustering algorithm for intuitionistic fuzzy sets," *Information Sciences*, vol. 178, no. 19, pp. 3775–3790, 2008.
- [33] Z. Xu, "Intuitionistic fuzzy hierarchical clustering algorithms," *Journal of Systems Engineering and Electronics*, vol. 20, no. 1, pp. 90–97, 2009.
- [34] E. Szmidt and J. Kacprzyk, "Distances between intuitionistic fuzzy sets," *Fuzzy Sets and Systems*, vol. 114, no. 3, pp. 505–518, 2000.
- [35] Z. Xu, "Some similarity measures of intuitionistic fuzzy sets and their applications to multiple attribute decision making," *Fuzzy Optimization and Decision Making*, vol. 6, no. 2, pp. 109–121, 2007.
- [36] P. Burillo and H. Bustince, "Entropy on intuitionistic fuzzy sets and on interval-valued fuzzy sets," *Fuzzy Sets and Systems*, vol. 78, no. 3, pp. 305–316, 1996.
- [37] Z. S. Xu, "Some similarity measures of intuitionistic fuzzy sets and their applications to multiple attribute decision making," *Fuzzy Optimization and Decision Making*, vol. 6, no. 2, pp. 109–121, 2007.
- [38] N. R. Pal, K. Pal, J. M. Keller, and J. C. Bezdek, "A possibilistic fuzzy c-means clustering algorithm," *IEEE Transactions on Fuzzy Systems*, vol. 13, no. 4, pp. 517–530, 2005.
- [39] K. Ito and K. Kunisch, *Lagrange Multiplier Approach to Variational Problems and Applications*, SIAM Advances in Design and Control, SIAM, Philadelphia, Pa, USA, 2008.
- [40] E. N. Nasibov and G. Ulutagay, "A new unsupervised approach for fuzzy clustering," *Fuzzy Sets and Systems*, vol. 158, no. 19, pp. 2118–2133, 2007.
- [41] A. Asuncion and D. J. Newman, *UCI Machine Learning Repository*, Department of Information and Computer Science, University of California, Irvine, Calif, USA, 2007.
- [42] J. Alcalá-Fdez, A. Fernández, J. Luengo et al., "KEEL data-mining software tool: data set repository, integration of algorithms and experimental analysis framework," *Journal of Multiple-Valued Logic and Soft Computing*, vol. 17, no. 2-3, pp. 255–287, 2011.
- [43] R. Kohavi and F. Provost, "Glossary of terms," *Machine Learning*, vol. 30, pp. 271–274, 1998.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

