

# Hierarchical Intuitionistic Fuzzy Possibilistic C Means Kernel Clustering Algorithm for Distributed Networks

Arindam Chaudhuri<sup>1</sup> and Soumya K Ghosh<sup>2</sup>

<sup>1</sup>Samsung R &D Institute Delhi Noida – 201301 India; Email: arindam\_chau@yahoo.co.in

<sup>2</sup>Indian Institute of Technology Kharagpur– 721302 India; Email: skg@iitkgp.ac.in

## Abstract

Advances in distributed networking have resulted in an explosion in size of modern datasets while storage and processing power continue to lag behind. This requires the need for algorithms that are efficient in terms of number of measurements and running time. To combat challenges associated with large datasets in distributed networks we propose hierarchical intuitionistic fuzzy possibilistic *c*-means kernel clustering algorithm. The algorithm executes hierarchically by performing clustering at each peer. The intuitionistic fuzzy degree and typicality membership functions and weight-attribute-entropy factor improves clustering performance. The experiments on artificial and real datasets establish the efficiency and effectiveness of the algorithm.

**Keywords:** Hierarchical clustering, intuitionistic fuzzy set, peer-to-peer network, large datasets, kernel clustering, fuzzy possibilistic *c*-means

## 1. Introduction

In exploratory data analysis clustering [1] is a ubiquitous task. In clustering each object is assigned to one or more groups so that objects in the same group are very similar. The mainstream clustering techniques are based on centralized operation i.e. datasets are of small manageable sizes and usually reside on one central site and then a single process performs clustering the data. The *k*-means algorithm and fuzzy *c*-means (FCM) algorithm [2] are two well-known centralized clustering algorithms. However as the modern datasets gets larger computational burden increases as well as infeasibility of collecting data to central site due to privacy and security requirements. These factors limit the practicality of these algorithms. So it is obvious to develop clustering algorithms that are effective on large scale problems from both measurement and computational perspective. Again for complex distributed network systems the traditional algorithm fails to give desirable clustering results. This entails development of robust clustering algorithms for managing the continuously growing data.

The peer to peer (P2P) [3] is a major architecture for distributed clustering. In P2P network each peer (data site) has equal functionality. A peer is a facilitator as well as worker. A large number of peers are connected in an ad hoc way where each peer can join and leave the network dynamically. Each peer can communicate with others according to the network structure. The P2P distributed clustering algorithms achieves the locally optimized clusters at each peer taking into consideration

the local data in this peer and necessary information exchanged from others. The prototype based partition clustering has given satisfactory results for P2P networks [4]. It is a commonly used machine learning technique for data mining, pattern recognition and statistical analysis [5].

To achieve the stated clustering objectives in this paper we propose a novel hierarchical intuitionistic fuzzy possibilistic *c*-means kernel clustering (HIFPCMCK) algorithm over a distributed P2P network. HIFPCMCK is the hierarchical extension of intuitionistic fuzzy possibilistic *c*-means kernel clustering (IFPCMCK). IFPCMCK is the kernel version of intuitionistic fuzzy possibilistic *c*-means (IFPCM) clustering algorithm proposed by Chaudhuri [6]. The algorithm searches optimized clusters at each peer by collaborating with neighboring peers hierarchically till a global consensus of all peers is reached. It thus reduces the communication overhead among peers. The algorithm easily performs high dimensional sparse data clustering using the intuitionistic fuzzy degree and typicality membership functions. For high dimensional sparse data the cluster structure in the dataset is attributed to a subset of features. A weight-attribute-entropy factor is also inducted into the clustering process according to the significance of different dimensions for cluster identification. This helps to achieve the ideal distribution of attribute weights consistent with available data. This results in optimal clustering so that important features are extracted for cluster identification. The kernelization of the algorithms allows easy handling of non-spherical clustering data. The experimental results on both synthetic and real datasets reveal the superiority of the proposed technique. This paper is structured as follows. In section 2 some work related to hierarchical clustering are given. The mathematical framework of IFPCMCK is highlighted in section 3. This is followed by HIFPCMCK in the next section. The section 5 discusses experiments and results. Finally conclusions are given in section 6.

## 2. Related Work

In recent years a considerable amount of work has been done towards clustering algorithms for distributed networks. The P2P *k*-means algorithm proposed by Datta et al [7] is the first algorithm developed for P2P systems. Kashef et al [8] presented a distributed cooperative clustering method in a two tier hierarchical P2P network. Forero et al [9] proposed a good solution of distributed clustering in WSNs by capitalizing on the

consensus based formulation and parallel optimization tools. Pedrycz et al [10] introduced the concept of collaborative fuzzy clustering where the summarized knowledge structures in different peers are shared by communicating information granules. Coletta et al [11] extended Pedrycz's method to optimize parameters including interaction level for all peer pairs and number of clusters at each peer. These collaborative approaches considered fully connected network structures and with-in a large dynamic network.

### 3. Intuitionistic Fuzzy Possibilistic C-Means Kernel Clustering Analysis

In this section we present the mathematical framework of IFPCM algorithm. The problem description is given followed by the description of the method.

#### 3.1 Problem Description

We consider a distributed network as graph  $G$  with  $Pr$  peers where each peer  $j \in \{1, 2, \dots, Pr\}$  is a node. The edge between two nodes is the communication link. Each peer  $j$  communicates with its intermediate neighbors set  $NH_j$ . The distributed network collects objects and performs clustering task. Each peer  $j$  consists of a set of  $Obj$  objects i.e.  $Y_j = \{y_{jn} | n \in \{1, 2, \dots, Obj\}\}$ . Each object  $y_{jn} = [y_{jn1}, y_{jn2}, \dots, y_{jnD}]$  has  $D$  dimensions. The number of clusters  $Ct$  is same for all peers. The clustering operation of the distributed network assigns each object to cluster  $t$  ( $t \in \{1, 2, \dots, Ct\}$ ) based on suitable criterion to quantify similarity among objects.

#### 3.2 Intuitionistic Fuzzy Possibilistic C-Means Kernel Clustering Algorithm

The IFPCM clustering method for IFSs towards the distributed network consists of the following objective function where distributed dissimilarity measure is combined with weight entropy regularization.

$$\begin{aligned} \min_{(U, T, C, W)} \left\{ J_{m, \eta}(U, T, C, W) = \sum_{i=1}^P \sum_{j=1}^{Pr} \sum_{n=1}^{Obj} \sum_{k=1}^{Ct} (u_{jnk}^m + t_{jnk}^\eta) \sum_{d=1}^D w_{jkd} Dt_\alpha^2(Z_i, C_k) \right. \\ \left. + \beta \sum_{j=1}^{Pr} \sum_{k=1}^{Ct} \sum_{d=1}^D (w_{jkd})^{m\eta} \log(w_{jkd})^{m\eta} \right\} \\ \text{subject to } m > 1, \eta > 1; c_{jkd} = c_{ikd}, i \in NH_j \\ \sum_{k=1}^{Ct} u_{jnk} = 1, 0 \leq u_{jnk} \leq 1, \sum_{k=1}^{Ct} t_{jnk} = 1 \wedge 0 \leq t_{jnk} \leq 1 \forall k \\ w_{jkd} = w_{ikd}, i \in NH_j \\ \sum_{d=1}^D w_{jkd} = 1, 0 \leq w_{jkd} \leq 1 \end{aligned} \quad (1)$$

Here  $Z = \{Z_1, \dots, Z_P\}$  are  $P$  IFSs each associated with  $y_{jn}$  objects;  $Ct$  is number of clusters ( $1 \leq Ct \leq Pr$ ) and  $V = \{V_1, \dots, V_{Ct}\}$  are prototypical IFSs i.e. centroid of clusters. In equation (1)  $m$  is fuzzy factor,  $U = [u_{jnk}]$  is membership degree matrix and  $u_{jnk}$  is

membership of  $n^{th}$  object belongs to  $k^{th}$  cluster in  $j^{th}$  peer;  $\eta$  is typicality factor,  $T = [t_{jnk}]$  is typicality matrix and  $t_{jnk}$  is typicality of  $n^{th}$  object to  $k^{th}$  cluster in  $j^{th}$  peer;  $C = [c_{jkd}]$  is cluster prototype matrix and  $c_{jkd}$  is  $d^{th}$  dimension of  $k^{th}$  cluster in  $j^{th}$  peer;  $W = [w_{jkd}]$  is weight matrix and  $w_{jkd}$  is  $d^{th}$  dimension of  $k^{th}$  cluster in  $j^{th}$  peer;  $\beta$  is positive scalar and  $Dt_\alpha^2(\circ)$  is basic distance measure expressed as normalized Euclidean distance [1], [2] which is proximity function of IFPCM and defined as:

$$\begin{aligned} Dt_\alpha^2(Z_i, C_k) = \frac{1}{2} \sum_{j=1}^{Pr} \sum_{n=1}^{Obj} \sum_{k=1}^{Ct} \sum_{d=1}^D \left( \|\mu_{Z_i}(y_{jnd}) - \mu_{C_k}(c_{jkd})\|^2 \right. \\ \left. + \|\nu_{Z_i}(y_{jnd}) - \nu_{C_k}(c_{jkd})\|^2 \right. \\ \left. + \|\pi_{Z_i}(y_{jnd}) - \pi_{C_k}(c_{jkd})\|^2 \right) \end{aligned} \quad (2)$$

The first term of objective function in equation (2) controls shape and size of clusters and encourages their agglomeration and the second term is negative entropy of weights that regularize optimal distribution of weights. Both these quantities can be optimally adjusted through positive regularizing and adjustable parameter  $\beta$ . The constraints  $c_{jkd} = c_{ikd}$  and  $w_{jkd} = w_{ikd}$ ,  $i \in NH_j$  ensures that local cluster prototypes and weights yield at each peer coincide with global ones of all objects. Minimizing  $J_{m, \eta}(U, T, C, W)$  with respect to constraints leads to constrained optimization problem where Picard iteration is applied to solve the problem through Lagrange multipliers. The matrices  $U, T, C$  and  $W$  are updated through  $u_{jnk}$ ,  $t_{jnk}$ ,  $c_{jkd}$  and  $w_{jkd}$ .

IFPCM uses weighted squared distance to evaluate similarity between objects and prototypes. It is only helpful in clustering data with spherical clusters. For data with non-spherical clusters, clustering in high dimensional feature space with Mercer kernel based mapping is considered [2], [6]. The essence of kernel method is to perform a nonlinear mapping  $\Phi$  from the original  $d$ -dimensional space  $\mathbf{R}^d$  to high dimensional kernel space  $HD$  [6]. The linear classifier in kernel space is used to solve the clustering problem which could be highly nonlinear in original feature space. The kernel method has been widely applied to fuzzy clustering [12]. Considering the need of exchanging the cluster prototype and attribute weight messages in original feature space, the objective function is defined as:

$$\begin{aligned} \min_{(U, T, C, W)} \left\{ J_{m, \eta}(U, T, C, W) = \sum_{i=1}^P \sum_{j=1}^{Pr} \sum_{n=1}^{Obj} \sum_{k=1}^{Ct} (u_{jnk}^m + t_{jnk}^\eta) \sum_{d=1}^D w_{jkd} \Phi(Dt_\alpha^2(Z_i, C_k)) \right. \\ \left. + \beta \sum_{j=1}^{Pr} \sum_{k=1}^{Ct} \sum_{d=1}^D (w_{jkd})^{m\eta} \log(w_{jkd})^{m\eta} \right\} \\ \text{subject to } m > 1, \eta > 1; c_{jkd} = c_{ikd}, i \in NH_j \\ \sum_{k=1}^{Ct} u_{jnk} = 1, 0 \leq u_{jnk} \leq 1, \sum_{k=1}^{Ct} t_{jnk} = 1 \wedge 0 \leq t_{jnk} \leq 1 \forall k \\ w_{jkd} = w_{ikd}, i \in NH_j \\ \sum_{d=1}^D w_{jkd} = 1, 0 \leq w_{jkd} \leq 1 \end{aligned} \quad (3)$$

In equation (3)  $U = [u_{jnk}]$ ,  $T = [t_{jnk}]$ ,  $C = [c_{jkd}]$  and  $W = [w_{jkd}]$  are membership degree matrix, typicality degree matrix, cluster prototype matrix and attribute weight matrix respectively. The parameter  $\Phi$  is nonlinear mapping from original feature space to kernel space,  $m$  is fuzzy coefficient,  $\eta$  is typicality coefficient and  $\beta$  is positive scalar. In this method the dot products in kernel space is expressed as Mercer kernel  $K(a, b) = \Phi(a) \Phi(b)$ ,  $a, b \in \mathbf{R}^d$ . Here Gaussian kernel is considered as the representative Mercer kernel so that:

$$K(a, b) = \exp\left(\frac{-\|a - b\|^2}{\sigma^2}\right), \sigma^2 > 0 \quad (4)$$

$$\begin{aligned} \Phi(Dt_{\alpha}^2(Z_i, C_k)) = 3 - \exp\left[\left(\frac{-\|\mu_{Z_i}(y_{jnd}) - \mu_{C_k}(c_{jkd})\|^2}{\sigma^2}\right) \right. \\ \left. + \left(\frac{-\|v_{Z_i}(y_{jnd}) - v_{C_k}(c_{jkd})\|^2}{\sigma^2}\right) \right. \\ \left. + \left(\frac{-\|\pi_{Z_i}(y_{jnd}) - \pi_{C_k}(c_{jkd})\|^2}{\sigma^2}\right)\right] \quad (5) \end{aligned}$$

The Picard iteration is applied to solve the optimization problem in equation (3). The matrices  $U, T, C$  and  $W$  are updated corresponding to the equations (6) – (11) respectively. The equations (6), (7) and (8) are given in Appendix. The equations (6) and (7) are valid for  $1 \leq j \leq Pr, 1 \leq n \leq Ob_j, 1 \leq k \leq Ct$ . The equation (8) is valid for  $1 \leq j \leq Pr, 1 \leq k \leq Ct, 1 \leq d \leq D$ .

$$p_{jikd} = p_{jikd} + \vartheta_1(c_{jkd} - c_{ikd}) \quad (9)$$

The equation (9) is valid for  $1 \leq j \leq Pr, 1 \leq k \leq Ct, 1 \leq d \leq D, i \in NH_j$ .

$$w_{jkd} = \frac{\exp\left(-\beta^{-1} \sum_{n=1}^{Ob_j} (u_{jnk}^m + t_{jnk}^n) \Phi(Dt_{\alpha}^2(Z_i, C_k)) - 2\beta^{-1} \sum_{i \in NH_j} q_{jikd}\right)}{\sum_{p=1}^D \exp\left(-\beta^{-1} \sum_{n=1}^{Ob_j} (u_{jnk}^m + t_{jnk}^n) \Phi(Dt_{\alpha}^2(Z_i, C_k)) - 2\beta^{-1} \sum_{i \in NH_j} q_{jikv}\right)} \quad (10)$$

The equation (10) is valid for  $1 \leq j \leq Pr, 1 \leq k \leq Ct, 1 \leq d \leq D$ .

$$q_{jikd} = q_{jikd} + \vartheta_2(w_{jkd} - w_{ikd}) \quad (11)$$

The equation (11) is valid for  $1 \leq j \leq Pr, 1 \leq k \leq Ct, 1 \leq d \leq D, i \in NH_j$ . Here,  $P = [p_{jikd}]$  and  $Q = [q_{jikd}]$  are two Lagrange multiplier matrices and  $\vartheta_1, \vartheta_2$  are positive scalars.

#### 4. Hierarchical Intuitionistic Fuzzy Possibilistic C-Means Kernel Clustering Analysis

Based on IFPCMCK, HIFPCMCK is formulated in this section. The hierarchical version of IFPCMCK is efficient both in terms of the number of similarities used as well as the running time [13]. To recover any single hierarchy split, IFPCMCK is run on a small subset of the data to compute a seed clustering of the dataset. Using the initial clustering, each remaining object is placed into the seed cluster for which it is most similar on av-

erage. This results in IFPCMCK of the entire dataset, using only similarities to the objects in the small subset. By recursively applying this procedure to each cluster, HIFPCMCK is obtained using a small fraction of the similarities. In this recursive phase, we do not observe any measurements between clusters at the previous split i.e. to partition cluster  $Ct_j$  we only observe similarities between objects in  $Ct_j$ . This results in a robust version of HIFPCMCK that aligns its measurements  $mt$  to resolve the higher resolution cluster structure. The pseudo code for HIFPCMCK is shown in Algorithm 1.

**Algorithm 1:** HIFPCMCK (IFPCMCK,  $mt, \{y_i\}_{i=1}^{Ob_j}, Ct_j$ )

**if**  $Ob_j < mt$  **then return**  $\{y_i\}_{i=1}^{Ob_j}$   
 Select  $W \subseteq \{y_i\}_{i=1}^{Ob_j}$  of size  $w$  uniformly at random  
 $C'_1, \dots, C'_{Ct_j} \leftarrow IFPCMCK(W, Ct_j)$   
 Set  $C_1 \leftarrow C'_1, \dots, C_{Ob_j} \leftarrow C'_{Ob_j}$   
**for**  $y_i \in \{y_i\}_{i=1}^{Ob_j} \setminus W$  **do**  
 $\forall k \in [Ct_j], \alpha_k \leftarrow \frac{1}{|C'_j|} \sum_{y_s \in C'_j} S(y_i, y_s)$   
 $C_{\arg\max_{k \in [Ct_j]} \alpha_k} \leftarrow C_{\arg\max_{k \in [Ct_j]} \alpha_k} \cup \{y_i\}$   
**end for**  
**output**  $\{C_k, HIFPCMCK(IFPCMCK, mt, C_k, Ct_j)\}_{k=1}^{Ct_j}$

HIFPCMCK is characterized in terms of probability of success in recovering the true hierarchy  $Ct^*$ , measurement and runtime complexity. Some mild restrictions are placed on the similarity function  $S$  such that the aforementioned specifications work. This ensures that similarities agree with the hierarchy up to some random noise:

**S1** For each  $y_i \in Ct_j \in Ct^*$  and  $j' \neq j$ :  
 $\min_{y_p \in Ct_j} \mathbb{E} \exp[S(y_i, y_p)] - \max_{y_p \in Ct_{j'}} \mathbb{E} \exp[S(y_i, y_p)] \geq \gamma > 0$

Here expectations are taken with respect to the possible noise on  $S$ .

**S2** For each object  $y_i \in Ct_j$ , a set of  $V_j$  objects of size  $v_j$  drawn uniformly from cluster  $Ct_j$  satisfies:

$$\mathbb{P} \left( \min_{y_p \in Ct_j} \mathbb{E} \exp[S(y_i, y_p)] - \sum_{y_p \in V_j} \frac{S(y_i, y_p)}{v_j} > \epsilon \right) \leq 2e^{\left\{ \frac{-2v_j \epsilon^2}{\sigma^2} \right\}}$$

Here  $\sigma^2 \geq 0$  parameterizes the noise on the similarity function  $S$ . Similarly a set  $V_{j'}$  of size  $v_{j'}$  drawn uniformly from cluster  $Ct_{j'}$  with  $j \neq j'$  satisfies:

$$\mathbb{P} \left( \sum_{y_p \in V_{j'}} \frac{S(y_i, y_p)}{v_{j'}} - \max_{y_p \in C_{j'}} \mathbb{E} \exp[S(y_i, y_p)] > \epsilon \right) \leq 2e^{\left\{ \frac{-2v_{j'} \epsilon^2}{\sigma^2} \right\}}$$

The condition **S1** states that the similarity from an object  $y_i$  to its cluster should in expectation are larger than the similarity from that object to any other cluster. This is related to tight clustering condition [13] and less stringent than earlier results which assume that within-

and-between-cluster similarities are constant and bounded in expectation [14]. The condition **S2** enforces that within-and-between-cluster similarities concentrate away from each other. This condition is satisfied if similarities are constant in expectation, perturbed with any subgaussian noise [14], [15].

The algorithm has certain shortcomings that are undesirable for practical applications. Specifically if  $Ct$  is known and constant across splits in the hierarchy and the balance condition are both assumptions that are likely to be violated in any practical situation. This can be resolved by fine tuning the algorithm with several heuristics. The eigengap heuristic is employed in which  $Ct$  is chosen so that the gap in eigenvalues of the Laplacian is large. All the subsampled objects are discarded with low degree when restricted to the sample with the hope of removing underrepresented clusters from the sample. In averaging phase, if an object is not highly similar to any cluster represented in the sample a new cluster for this object is created. It is expected these two heuristics will help in recovering small clusters.

## 5. Experiments and Results

In this section we evaluate the performance of IFPCM KC and HIFPCM KC algorithms with respect to several FCM type clustering algorithms: (a) centralized clustering methods viz. FCM [2] and weighted entropy fuzzy  $c$ -means (WEFCM)[16] (b) kernel based clustering method viz. kernel based fuzzy  $c$ -means and fuzzy clustering (KFCM-F)[17] (c) parallel and distributed clustering methods viz. parallel fuzzy  $c$ -means(PFCM)[18], soft distributed  $k$ -means (SoftDKM)[19] and kernel based collaborative distributed fuzzy  $c$ -means (KCDFCM)[20]. The algorithms are implemented in MATLAB with 3.6 GHz CPU and 6 GB RAM. The Table 1 (given in Appendix) shows the parameter of the clustering algorithms used for experiments on the synthetic and real datasets.

Each algorithm is executed on each dataset for 200 times. The cluster prototypes are randomly initialized at each time. As the range of values of raw data varies widely in clustering algorithms, the objective functions work properly with normalization only. In this work simple normalization method is adopted which rescales the features in the range  $[0, 1]$  to make them independent of each other. For comparative analysis three kinds of performance metrics are applied in the experiments. They are:

- (a) Classification Rate (CR): The classification rate is a measure that used to determine how well the clustering algorithm performs on the given dataset with a known cluster structure [2]. It is expressed in terms of percentage. Better results are obtained from larger classification rate values. For 200 executions the average classification rate is used in experiments.
- (b) Normalized Mutual Information (NMI): The normalized mutual information provides a symmetric

measure to quantify the statistical information shared between two cluster distributions [2]. Better results are obtained from larger normalized mutual information values. For 200 executions the average normalized mutual information is used in experiments.

- (c) Transmission Energy Consumption (TEC): It denotes the energy consumption level of each sensor for data transmission in wireless sensor networks and gives an indication of the network state [20]. The average transmission energy consumption and the maximum transmission energy consumption are calculated 200 times. Better results are obtained from smaller transmission energy consumption indices.

### 5.1 Synthetic Datasets

The wireless sensor network based synthetic datasets are used for experiments. The dataset is generated through sensor nodes which are randomly distributed over  $200\text{ m} \times 200\text{ m}$  region. The communication range of each sensor is set to 60 m approximately. Each sensor exchanges information with the neighbors in communication range. The wireless sensor network is deployed to collect the monitoring data and perform clustering. It is assumed that each sensor node collects 180 objects belonging to 3 clusters. Each object has 6 attributes. Three synthetic datasets are created in the wireless sensor network for the experiments, where S1 collects 4500 data rows using 30 sensor nodes, S2 collects 7500 data rows using 50 sensor nodes and S3 collects 10500 data rows with 70 sensor nodes. The dataset generation algorithm is summarized in [20]. The Table 2 (given in Appendix) shows the clustering results of different algorithms on three wireless sensor network based synthetic datasets S1, S2 and S3. As the results from different datasets are similar, dataset S2 is considered for analysis. We observe that the encapsulation of attribute-weight-entropy regularization in the proposed algorithm improves the clustering performance in terms of ACR and ANMI. In Table 2 WEFCM, KCDFCM and HIFPCM KC show better ACR and ANMI than rest of the algorithms. For the attribute-weight clustering algorithms viz. WEFCM, KCDFCM and HIFPCM KC, the performance in terms of ACR and ANMI parameters are very close. HIFPCM KC need less iterations to reach convergence than KCDFCM. The information exchange between different sensors extends the convergence time of KCDFCM. HIFPCM KC performs clustering hierarchically and shortens the convergence time. The Table 3 (given in Appendix) shows the final assignments of the attribute-weight clustering algorithms viz. WEFCM, KCDFCM and HIFPCM KC. The attribute-1 is more important and contributes much more than other attributes when differing cluster-1 from cluster-2 and cluster-3. So higher weight is assigned to attribute-1 in cluster-1 for each attribute-weight clustering algorithm. A similar situation happens for attribute-2 and attribute-3. These evidences support the efficiency of the attribute-weight-entropy regularization tech-

nique for clustering. The three distributed clustering algorithms viz. Soft-DKM, KCDPCM and HIFPCMKC have less ATEC than other algorithms. This is because distributed clustering algorithms exchange only a small data quantity like cluster prototypes and attribute weights which is not so for other algorithms. Moreover the distributed algorithms yield much more balanced transmission energy consumptions among sensors which leads to longer network lifetime. These aspects prove the superiority of HIFPCMKC for clustering in distributed energy efficient network applications compared with other clustering algorithms. The kernel based algorithms do not provide significant improvements in ACR and ANMI as compared with other approaches. This is explained by the spherical shapes of the generated synthetic data. For these datasets, the capability of kernel based clustering in separating non-spherical data has not shown advantages.

From the above discussion it is obvious that the proposed HIFPCMKC algorithm is best suited in terms of technical constraints such as data size volume as well as privacy and security problems. HIFPCMKC algorithm is also preferred for non-spherical data clustering as illustrated by experiments in the following section.

### 5.2 Synthetic Datasets (Non-spherical)

Three synthetic datasets with non-spherical shaped clusters viz. fuzzy “X”, parabolic and ring are considered [21], [22]. Each sensor node collects objects belonging to 2 clusters and each object has 2 attributes. The fuzzy “X” has 640 data rows using 8 sensor nodes. The parabolic has 960 data rows using 8 sensor nodes. The ring has 750 data rows using 5 sensor nodes. As the number of peers is relatively small, the simple network structure shown in Figure 1 is applied on the datasets. Each peer exchanges messages only with its nearest neighbors.

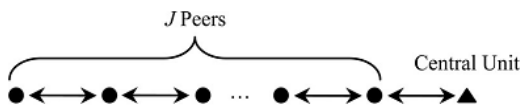


Figure 1: Simple network architecture

The Table 4 (given in Appendix) shows the clustering results of different algorithms on three synthetic datasets. For most of the synthetic datasets the kernel based clustering algorithms viz. KFCM-F, KCDPCM and HIFPCMKC give excellent clustering results in terms of ACR and ANMI. HIFPCMKC algorithm achieves 100 % in ACR on the ring dataset. These results verify the efficiency of HIFPCMKC in comparison to other algorithms for non-spherical shaped data clustering.

### 5.3 UCI Machine Learning Datasets

Three real datasets from UCI repository [23] viz. iris, glass and ionosphere are applied on the clustering algorithms. The iris has 150 data rows using 3 sensor nodes

with each sensor node collecting objects belonging to 3 clusters and each object has 4 attributes. The glass has 214 data rows using 3 sensor nodes with each sensor node collecting objects belonging to 6 clusters and each object has 9 attributes. The ionosphere has 351 data rows using 4 sensor nodes with each sensor node collecting objects belonging to 2 clusters and each object has 33 attributes. The simple network structure shown in Figure 1 is applied. The Table 5 (given in Appendix) shows the clustering results of the algorithms. It is evident from Table 5 that HIFPCMKC shows superior performance in terms of ACR and ANMI. The best case is for the iris dataset. The consideration of attribute weight assignment highlights good clustering results in ACR and ANMI. This performance is as good as the best case which is obtained by the centralized attribute-weight algorithm viz. WEFCM. It again proves the efficiency of HIFPCMKC algorithm.

## 6. Conclusion

In this research work we propose a hierarchical intuitionistic fuzzy possibilistic clustering algorithm viz. HIFPCMKC for distributed P2P networks. The centralized clustering problem is solved in a distributed mode at each peer with neighboring peers. The clustering performance is improved by the intuitionistic fuzzy degree and typicality membership functions and weight-attribute-entropy factor. The intuitionistic membership intervals capture the data semantics more appropriately through its degrees of membership and non-membership. The choice of the membership and non-membership boundary also has interesting implications on modelling relationship between vague data existing in the distributed network. HIFPCMKC has superior clustering performance than the centralized clustering approaches. It also reduces and balances the communication overhead among peers. Based on the attribute-weight-entropy regularization technique important features are extracted through the optimal distribution of attribute weight. The kernelization of the algorithm satisfies the requirements of non-spherical shaped data clusters. The experiments on several synthetic and real datasets shows that HIFPCMKC yields comparative better performance with respect to other clustering methods. The proposed algorithm for distributed P2P network can be generalized for various system control and distributed network applications. The objective function of algorithm can further be optimized through metaheuristics and hybrid optimization approaches which will improve the clustering performance.

## References

- [1] R. Xu and D. C. Wunsch, Survey of clustering algorithms, *IEEE Transactions on Neural Networks*, 16 (3): 645–678, IEEE Computational Intelligence Society, 2005.
- [2] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, 1981.

- [3] K. M. Hammouda, *Distributed Document Clustering and Cluster Summarization in Peer-to-Peer Environments*, PhD Dissertation, Department of Electrical and Computer Engineering, University of Waterloo, Canada, 2007.
- [4] H. Kargupta and P. Chan, *Advances in Distributed and Parallel Knowledge Discovery*, MIT Press, Cambridge, MA, 2000.
- [5] J. W. Han and M. Kamber, *Data Mining: Concept and Techniques*, Morgan Kaufmann, San Mateo, CA, 2001.
- [6] A. Chaudhuri, Intuitionistic fuzzy possibilistic  $c$ -means clustering algorithms, *Advances in Fuzzy Systems*, Volume 2015: 2–18, Hindawi Press, 2015.
- [7] S. Datta, C. Giannella and H. Kargupta, K-means clustering over large, dynamic network, proceedings of the *SIAM international conference on data mining*, pages 153–164, 2006.
- [8] R. Kashef, *Cooperative clustering model and its applications*, PhD Dissertation, Department of Electrical and Computer Engineering, University of Waterloo, Canada, 2008.
- [9] P. A. Forero, A. Cano and G. B. Giannakis, Distributed clustering using wireless sensor networks, *IEEE Journal of Selected Topics in Signal Processing*, 5(4): 707–724, IEEE Computational Intelligence Society, 2011.
- [10] W. Pedrycz and P. Rai, Collaborative clustering with the use of fuzzy  $c$  means and its quantification, *Fuzzy Sets and Systems*, 159(18): 2399–2427, Elsevier Press, 2008.
- [11] L. F. S. Coletta, L. Vendramin, E. R. Hruschka, R. J. G. B. Campello and W. Pedrycz, Collaborative fuzzy clustering algorithms: Some refinements and design guidelines, *IEEE Transactions on Fuzzy Systems*, 20(3): 444–462, IEEE Computational Intelligence Society, 2012.
- [12] K. M. Bataineh, M. Najia and M. Saqera, A comparison study between various fuzzy clustering algorithms, *Jordan Journal of Mechanical and Industrial Engineering*, 5(4): 335–343, The Jordanian Ministry of Higher Education and Scientific Research, 2011.
- [13] B. Eriksson, G. Dasarathy, A. Singh and R. Nowak, *Active Clustering: Robust and efficient Hierarchical Clustering using adaptively selected similarities*, CoRR, 2011.
- [14] K. Rohe, S. Chatterjee and B. Yu, *Spectral clustering and the high dimensional stochastic block model*, Technical Report 791, Department of Statistics, University of California, Berkeley, 2010.
- [15] S. Balakrishnan, M. Xu, A. Krishnamurthy and A. Singh, Noise thresholds for spectral clustering, proceedings of the *Neural Information Processing Systems*, 2011.
- [16] J. Zhou and C. L. P. Chen, Attribute weighted entropy regularization in fuzzy  $c$ -means algorithm for feature selection, proceedings of the *IEEE international conference on system science engineering*, pages 59–64, 2011.
- [17] D. Graves and W. Pedrycz, Kernel based fuzzy clustering and fuzzy clustering: A comparative experimental study, *Fuzzy Sets and Systems*, 161(4): 522–543, Elsevier Press, 2010.
- [18] T. Kwok, K. A. Smith, S. Lozano and D. Taniar, Parallel fuzzy  $c$ -means clustering for large datasets, proceedings of the *8<sup>th</sup> international Euro-Par conference of parallel processing*, pages 365–374, 2002.
- [19] P. A. Forero, A. Cano and G. B. Giannakis, Distributed clustering using wireless sensor networks, *IEEE Journal of Selected Topics on Signal Processing*, 5(4): 707–724, IEEE Computational Intelligence Society, 2011.
- [20] J. Zhou, C. L. P. Chen, L. Chen and Han-Xiong Li, A collaborative fuzzy clustering algorithm in distributed network environments, *IEEE Transactions on Fuzzy Systems*, 22(6): 1443–1456, IEEE Computational Intelligence Society, 2014.
- [21] N. R. Pal and K. Sarkar, What and when can we gain from the kernel versions of  $c$ -means algorithm ?, *IEEE Transactions on Fuzzy Systems*, 22(2): 363–379, IEEE Computational Intelligence Society, 2014.
- [22] J. Huang, X. Chen, P. C. Yuen, J. Zhang, W. S. Chen and J. H. Lai, Kernel parameter optimization of kernel based LDA methods, proceedings of the *IEEE international joint conference on neural networks*, pages 3840–3846, 2008.
- [23] UCI Repository: <http://archive.ics.uci.edu/ml/>

**Appendix**  
**[Equations (6), (7) and (8); Tables 1–5]**

$$u_{jnk} = \frac{1}{\sum_{h=1}^{C_t} \left( \frac{\sum_{d=1}^D w_{jkd} \left( \frac{1}{2} \sum_{j=1}^{P_r} \sum_{n=1}^{O_{b_j}} \sum_{k=1}^{C_t} \sum_{d=1}^D \left( \left\| \Phi(\mu_{Z_i}(y_{jnd}) - \Phi(\mu_{C_k}(c_{jkd}))) \right\|^2 + \left\| \Phi(v_{Z_i}(y_{jnd}) - \Phi(v_{C_k}(c_{jkd}))) \right\|^2 + \left\| \Phi(\pi_{Z_i}(y_{jnd}) - \Phi(\pi_{C_k}(c_{jkd}))) \right\|^2 \right) \right)}{\sum_{d=1}^D w_{jhm} \left( \frac{1}{2} \sum_{j=1}^{P_r} \sum_{n=1}^{O_{b_j}} \sum_{k=1}^{C_t} \sum_{d=1}^D \left( \left\| \Phi(\mu_{Z_i}(y_{jnd}) - \Phi(\mu_{C_k}(c_{jkd}))) \right\|^2 + \left\| \Phi(v_{Z_i}(y_{jnd}) - \Phi(v_{C_k}(c_{jkd}))) \right\|^2 + \left\| \Phi(\pi_{Z_i}(y_{jnd}) - \Phi(\pi_{C_k}(c_{jkd}))) \right\|^2 \right) \right)} \right)^{2/(m-1)}} \quad (6)$$

$$t_{jnk} = \frac{1}{\sum_{h=1}^{C_t} \left( \frac{\sum_{d=1}^D w_{jkd} \left( \frac{1}{2} \sum_{j=1}^{P_r} \sum_{n=1}^{O_{b_j}} \sum_{k=1}^{C_t} \sum_{d=1}^D \left( \left\| \Phi(\mu_{Z_i}(y_{jnd}) - \Phi(\mu_{C_k}(c_{jkd}))) \right\|^2 + \left\| \Phi(v_{Z_i}(y_{jnd}) - \Phi(v_{C_k}(c_{jkd}))) \right\|^2 + \left\| \Phi(\pi_{Z_i}(y_{jnd}) - \Phi(\pi_{C_k}(c_{jkd}))) \right\|^2 \right) \right)}{\sum_{d=1}^D w_{jhm} \left( \frac{1}{2} \sum_{j=1}^{P_r} \sum_{n=1}^{O_{b_j}} \sum_{k=1}^{C_t} \sum_{d=1}^D \left( \left\| \Phi(\mu_{Z_i}(y_{jnd}) - \Phi(\mu_{C_k}(c_{jkd}))) \right\|^2 + \left\| \Phi(v_{Z_i}(y_{jnd}) - \Phi(v_{C_k}(c_{jkd}))) \right\|^2 + \left\| \Phi(\pi_{Z_i}(y_{jnd}) - \Phi(\pi_{C_k}(c_{jkd}))) \right\|^2 \right) \right)} \right)^{2/(\eta-1)}} \quad (7)$$

$$c_{jkd} = \frac{\sum_{n=1}^{O_{b_j}} (u_{jnk}^m + t_{jnk}^\eta) w_{jkd} \exp \left[ \left( \frac{-\|\mu_{Z_i}(y_{jnd}) - \mu_{C_k}(c_{jkd})\|^2}{\sigma^2} \right) + \left( \frac{-\|v_{Z_i}(y_{jnd}) - v_{C_k}(c_{jkd})\|^2}{\sigma^2} \right) + \left( \frac{-\|\pi_{Z_i}(y_{jnd}) - \pi_{C_k}(c_{jkd})\|^2}{\sigma^2} \right) \right] y_{jnd} - \frac{\sigma^2}{2} \sum_{i \in NH_j} p_{jikd}}{\sum_{n=1}^{O_{b_j}} (u_{jnk}^m + t_{jnk}^\eta) w_{jkd} \exp \left[ \left( \frac{-\|\mu_{Z_i}(y_{jnd}) - \mu_{C_k}(c_{jkd})\|^2}{\sigma^2} \right) + \left( \frac{-\|v_{Z_i}(y_{jnd}) - v_{C_k}(c_{jkd})\|^2}{\sigma^2} \right) + \left( \frac{-\|\pi_{Z_i}(y_{jnd}) - \pi_{C_k}(c_{jkd})\|^2}{\sigma^2} \right) \right]} \quad (8)$$

Algorithm	Parameters
FCM	$\alpha = 2$
WEFCM	$\alpha = 2$ ; $\gamma$ varied from 0.01 to 1
KFCM-F	$\alpha = 2$ ; $\sigma^2$ varied from $2^{-10}$ to $2^5$
PFCM	$\alpha = 2$
Soft-DKM	$\alpha = 2$ ; $\eta$ varied from $10^{-8}$ to 1
KCDFCM	$\alpha = 2$ ; $\sigma^2$ varied from $2^{-10}$ to $2^5$ ; $\gamma$ varied from 0.01 to 1; $\eta_1$ and $\eta_2$ varied from $10^{-8}$ to 1
HIFPCM KC	$m = 2$ ; $\eta = 2$ ; $\beta$ varied from 0.01 to 1; $v_1$ and $v_2$ varied from $10^{-8}$ to 1

Table 1: The parameters of FCM-type clustering algorithms

Datasets	FCM	WEFCM	KFCM-F	PFCM	Soft-DKM	KCDFCM	HIFPCM KC
S1	90.31	97.23	91.63	90.33	88.73	<b>97.63</b>	<b>98.63<sup>+</sup></b>
	0.7103	0.8863	0.7193	0.7103	0.7063	<b>0.8912</b>	<b>0.9013<sup>+</sup></b>
	4.3506	4.3506	4.3506	2.7517	0.7093 <sup>+</sup>	<b>2.2773</b>	<b>2.2763</b>
S2	90.62	97.16 <sup>+</sup>	91.72	90.63	89.70	<b>97.00</b>	<b>98.33</b>
	0.7203	0.8883 <sup>+</sup>	0.7269	0.7203	0.7198	<b>0.8842</b>	<b>0.8942</b>
	4.1833	4.1836	4.1836	2.9069	1.0183 <sup>+</sup>	<b>3.0603</b>	<b>3.0303</b>
S3	90.33	97.03 <sup>+</sup>	91.66	90.33	90.33	<b>96.83</b>	<b>98.63</b>
	0.7107	0.8879 <sup>+</sup>	0.7196	0.7107	0.7107	<b>0.8843</b>	<b>0.8943</b>
	4.3750	4.3750	4.3750	2.7393	1.1003 <sup>+</sup>	<b>2.7103</b>	<b>2.7003</b>

<sup>+</sup>The best performance among the group

Table 2: The results of clustering algorithms on wireless sensor network based synthetic datasets in terms of ACR, ANMI and ATEC

WEFCM						
	Attribute1	Attribute2	Attribute3	Attribute4	Attribute5	Attribute6
Cluster1	<b>0.4308</b>	0.1213	0.0655	0.1482	0.1379	0.0963
Cluster2	0.1096	<b>0.3309</b>	0.0953	0.1753	0.1575	0.1316
Cluster3	0.0859	0.1273	<b>0.3723</b>	0.1477	0.1406	0.1263
KCDFCM						
	Attribute1	Attribute2	Attribute3	Attribute4	Attribute5	Attribute6
Cluster1	<b>0.4009</b>	0.1279	0.0788	0.1463	0.1413	0.1049
Cluster2	0.0723	<b>0.3370</b>	0.1073	0.2027	0.1153	0.1656
Cluster3	0.0797	0.1386	<b>0.3503</b>	0.1818	0.1247	0.1249
HIFPCMCK						
	Attribute1	Attribute2	Attribute3	Attribute4	Attribute5	Attribute6
Cluster1	<b>0.3933</b>	0.1283	0.0869	0.1433	0.1513	0.1096
Cluster2	0.0623	<b>0.3393</b>	0.1096	0.2323	0.0933	0.1933
Cluster3	0.0696	0.1396	<b>0.3303</b>	0.2326	0.1033	0.1233

Table 3: Assignments of attribute weight obtained by attribute-weight clustering algorithms on the wireless sensor network based synthetic dataset

Datasets	FCM	WEFCM	KFCM-F	PFCM	Soft-DKM	KCDFCM	HIFPCMCK
Fuzzy “X”	50.16	50.90	65.67	50.16	50.12	<b>67.48</b>	<b>69.43<sup>+</sup></b>
	0.0008	0.0033	0.1487	0.0008	0.0003	<b>0.1923</b>	<b>0.2023<sup>+</sup></b>
Parabolic	88.13	88.43	88.53	88.13	88.10	<b>88.73</b>	<b>89.63<sup>+</sup></b>
	0.4766	0.4843	0.4878	0.4766	0.4743	<b>0.4943</b>	<b>0.4986<sup>+</sup></b>
Ring	50.13	52.09	99.59	50.13	50.13	<b>100.00</b>	<b>100.30<sup>+</sup></b>
	0.0013	0.0023	0.9653	0.0013	0.0013	<b>1.00</b>	<b>1.30<sup>+</sup></b>

<sup>+</sup>The best performance among the group

Table 4: The results of clustering algorithms on three synthetic datasets in terms of ACR and ANMI

Datasets	FCM	WEFCM	KFCM-F	PFCM	Soft-DKM	KCDFCM	HIFPCMCK
Iris	89.33	96.66 <sup>+</sup>	92.03	89.33	87.33	<b>96.23</b>	<b>98.23</b>
	0.7433	0.8801 <sup>+</sup>	0.7773	0.7433	0.7293	<b>0.8793</b>	<b>0.8993</b>
Glass	42.08	53.33	50.83	42.03	40.53	<b>55.73</b>	<b>56.93<sup>+</sup></b>
	0.2973	0.4263	0.3333	0.2973	0.2843	<b>0.4593</b>	<b>0.4633<sup>+</sup></b>
Ionosphere	70.93	76.53	73.36	70.93	67.73	<b>78.63</b>	<b>79.33<sup>+</sup></b>
	0.1299	0.2023	0.1823	0.1299	0.1033	<b>0.2353</b>	<b>0.2633<sup>+</sup></b>

<sup>+</sup>The best performance among the group

Table 5: The results of clustering algorithms on the UCI machine learning datasets in terms of ACR and ANMI