RFCM: A Hybrid Clustering Algorithm Using Rough and Fuzzy Sets

Pradipta Maji* **and Sankar K. Pal**
Center for Soft Computing Research
Indian Statistical Institute
203 B. T. Road, Kolkata, 700 108, India
{*pmaji,sankar*}*@isical.ac.in*

Abstract. A hybrid unsupervised learning algorithm, termed as rough-fuzzy c-means, is proposed in this paper. It comprises a judicious integration of the principles of rough sets and fuzzy sets. While the concept of lower and upper approximations of rough sets deals with uncertainty, vagueness, and incompleteness in class definition, the membership function of fuzzy sets enables efficient handling of overlapping partitions. The concept of crisp lower bound and fuzzy boundary of a class, introduced in rough-fuzzy c-means, enables efficient selection of cluster prototypes. Several quantitative indices are introduced based on rough sets for evaluating the performance of the proposed c-means algorithm. The effectiveness of the algorithm, along with a comparison with other algorithms, has been demonstrated on a set of real life data sets.

Keywords: Pattern recognition, data mining, clustering, fuzzy c-means, rough sets.

1. Introduction

Cluster analysis is a technique for finding natural groups present in the data. It divides a given data set into a set of clusters in such a way that two objects from the same cluster are as similar as possible and the objects from different clusters are as dissimilar as possible. In effect, it tries to mimic the human ability to group similar objects into classes and categories [8, 10].

Clustering techniques have been effectively applied to a wide range of engineering and scientific disciplines such as pattern recognition, machine learning, psychology, biology, medicine, computer vision,

*Address for correspondence: Center for Soft Computing Research, Indian Statistical Institute, 203 B. T. Road, Kolkata, 700 108, India

communications, and remote sensing. A number of clustering algorithms have been proposed to suit different requirements [8, 10, 11].

One of the most widely used prototype based partitional clustering algorithms is hard c-means [17]. In hard c-means, each object must be assigned to exactly one cluster. On the other hand, fuzzy c-means relaxes this requirement by allowing gradual memberships [4, 9]. In effect, it offers the opportunity to deal with the data that belong to more than one cluster at the same time. It assigns memberships to an object which are inversely related to the relative distance of the object to cluster prototypes. Also, it can deal with the uncertainties arising from overlapping cluster boundaries.

Although fuzzy c-means is a very useful clustering method, the resulting membership values do not always correspond well to the degrees of belonging of the data, and it may be inaccurate in a noisy environment [13, 14]. In real data analysis, noise and outliers are unavoidable. Hence, to reduce this weakness of fuzzy c-means, and to produce memberships that have a good explanation of the degrees of belonging for the data, Krishnapuram and Keller [13, 14] proposed a possibilistic approach to clustering which used a possibilistic type of membership function to describe the degree of belonging. However, the possibilistic c-means sometimes generates coincident clusters [3]. Recently, the use of both fuzzy (probabilistic) and possibilistic memberships in a clustering algorithm has been proposed in [19].

Rough set theory [22, 23] is a new paradigm to deal with uncertainty, vagueness, and incompleteness. It has been applied to fuzzy rule extraction, reasoning with uncertainty, fuzzy modeling, etc [12, 24]. It is proposed for indiscernibility in classification according to some similarity [22]. In [15], Lingras and West introduced a new clustering method, called rough c-means, which describes a cluster by a prototype (center) and a pair of lower and upper approximations. The lower and upper approximations are weighted different parameters to compute the new centers. Asharaf et al. [1] extended this algorithm that may not require specification of the number of clusters.

Combining fuzzy set and rough set provides an important direction in reasoning with uncertainty [2, 6, 7, 16, 21]. Both fuzzy sets and rough sets provide a mathematical framework to capture uncertainties associated with the data [6, 7]. They are complementary in some aspects. Recently, combining both rough and fuzzy sets, Mitra et al. [18] proposed rough-fuzzy c-means, where each cluster consists of a fuzzy lower approximation and a fuzzy boundary. Each object in lower approximation takes a distinct weight, which is its fuzzy membership value. However, the objects in lower approximation of a cluster should have similar influence on the corresponding centroid and cluster as well as their weights should be independent of other centroids and clusters. Thus, the concept of fuzzy lower approximation, introduced in rough-fuzzy c-means of [18], reduces the weights of objects of lower approximation. In effect, it drifts the cluster prototypes from their desired locations. Moreover, it is sensitive to noise and outliers.

In this paper, we propose a hybrid algorithm, termed as rough-fuzzy c-means, based on rough sets and fuzzy sets. While the membership function of fuzzy sets enables efficient handling of overlapping partitions, the concept of lower and upper approximations of rough sets deals with uncertainty, vagueness, and incompleteness in class definition. Each partition is represented by a set of three parameters, namely, a cluster prototype (centroid), a crisp lower approximation, and a fuzzy boundary. The lower approximation influences the fuzziness of the final partition. The cluster prototype (centroid) depends on the weighting average of the crisp lower approximation and fuzzy boundary. Several quantitative measures are introduced based on rough sets to evaluate the performance of the proposed algorithm. The effectiveness of the proposed algorithm, along with a comparison with crisp, fuzzy, possibilistic, and rough c-means, has been demonstrated on a set of benchmark data sets.

The structure of the rest of this paper is as follows. Section 2 briefly introduces the necessary notions of fuzzy c-means, rough sets, and rough c-means algorithm. In Section 3, we describe rough-fuzzy c-means algorithm based on the theory of rough sets and fuzzy c-means. Several quantitative performance measures are introduced in Section 4 to evaluate the quality of the proposed algorithm. A few case studies and a comparison with other methods are presented in Section 5. Concluding remarks are given in Section 6.

2. Fuzzy C-Means and Rough C-Means

This section presents the basic notions of fuzzy c-means and rough c-means. The proposed rough-fuzzy c-means algorithm is developed based on these algorithms.

2.1. Fuzzy C-Means

Let $X = \{x_1, \cdots, x_j, \cdots, x_n\}$ be the set of n objects and $V = \{v_1, \cdots, v_i, \cdots, v_c\}$ be the set of c centroids, where $x_j \in \Re^m$ and $v_i \in \Re^m$. The fuzzy c-means provides a fuzzification of the hard c-means [4, 9]. It partitions X into c clusters by minimizing the objective function

$$J = \sum_{j=1}^{n}\sum_{i=1}^{c}(\mu_{ij})^{\acute{m}_1}||x_j - v_i||^2 \tag{1}$$

where $1 \leq \acute{m}_1 < \infty$ is the fuzzifier, v_i is the ith centroid corresponding to cluster β_i, $\mu_{ij} \in [0,1]$ is the probabilistic membership of the pattern x_j to cluster β_i, and $||.||$ is the distance norm, such that

$$v_i = \frac{1}{n_i}\sum_{j=1}^{n}(\mu_{ij})^{\acute{m}_1}x_j; \text{ where } n_i = \sum_{j=1}^{n}(\mu_{ij})^{\acute{m}_1} \tag{2}$$

$$\mu_{ij} = (\sum_{k=1}^{c}(\frac{d_{ij}}{d_{kj}})^{\frac{2}{\acute{m}_1 - 1}})^{-1}; \; d_{ij}^2 = ||x_j - v_i||^2; \text{ subject to } \sum_{i=1}^{c}\mu_{ij} = 1, \forall j, 0 < \sum_{j=1}^{n}\mu_{ij} < n, \forall i. \tag{3}$$

The process begins by randomly choosing c objects as the centroids (means) of the c clusters. The memberships are calculated based on the relative distance of the object x_j to the centroids $\{v_i\}$ by Equation 3. After computing memberships of all the objects, the new centroids of the clusters are calculated as per Equation 2. The process stops when the centroids stabilize. That is, the centroids from the previous iteration are identical to those generated in the current iteration. The basic steps are outlined as follows:

1. Assign initial means v_i, $i = 1, 2, \cdots, c$. Choose values for $\acute{m}_1$ and threshold ϵ. Set iteration counter $t = 1$.

2. Compute memberships μ_{ij} by Equation 3 for c clusters and n objects.

3. Update mean (centroid) v_i by Equation 2.

4. Repeat steps 2 to 4, by incrementing t, until $|\mu_{ij}(t) - \mu_{ij}(t-1)| > \epsilon$.

In fuzzy c-means, the memberships of an object are inversely related to the relative distance of the object to the cluster centroids. In effect, it is very sensitive to noise and outliers. Also, from the standpoint of "compatibility with the centroid", the memberships of an object x_j in a cluster β_i should be determined solely by how close it is to the mean (centroid) v_i of the class, and should not be coupled with its similarity with respect to other classes.

To alleviate this problem, Krishnapuram and Keller [13, 14] introduced possibilistic c-means algorithm, where the objective function can be formulated as

$$J = \sum_{i=1}^{c}\sum_{j=1}^{n}(\nu_{ij})^{\acute{m}_2}||x_j - v_i||^2 + \sum_{i=1}^{c}\eta_i\sum_{j=1}^{n}(1-\nu_{ij})^{\acute{m}_2} \tag{4}$$

where $1 \leq \acute{m}_2 \leq \infty$ is the fuzzifier and η_i represents the scale parameter. The membership matrix ν generated by the possibilistic c-means is not a partition matrix in the sense that it does not satisfy the constraint

$$\sum_{i=1}^{c}\nu_{ij} = 1 \tag{5}$$

The update equation of ν_{ij} is given by

$$\nu_{ij} = \frac{1}{1+\mathrm{D}};\ \text{where } \mathrm{D} = \left\{\frac{||x_j - v_i||^2}{\eta_i}\right\}^{1/(\acute{m}_2-1)} \tag{6}$$

$$\text{subject to } \ \nu_{ij} \in [0,1], \forall i,j; 0 < \sum_{j=1}^{n}\nu_{ij} \leq n, \forall i; \text{ and } \max_i \nu_{ij} > 0, \forall j.$$

The scale parameter η_i represents the zone of influence of the cluster β_i. The update equation for η_i is

$$\eta_i = \mathrm{K}\cdot\frac{\mathrm{P}}{\mathrm{Q}};\ \text{where } \mathrm{P} = \sum_{j=1}^{n}(\nu_{ij})^{\acute{m}_2}||x_j - v_i||^2;\ \text{and } \mathrm{Q} = \sum_{j=1}^{n}(\nu_{ij})^{\acute{m}_2} \tag{7}$$

Typically K is chosen to be 1. In each iteration, the updated value of ν_{ij} depends only on the similarity between the object x_j and the centroid v_i. The resulting partition of the data can be interpreted as a possibilistic partition, and the membership values may be interpreted as degrees of possibility of the objects belonging to the classes, i.e., the compatibilities of the objects with the means (centroids). The updating of the means proceeds exactly the same way as in the case of the fuzzy c-means algorithm.

2.2. Rough Sets

The theory of rough sets begins with the notion of an approximation space, which is a pair $< U, R >$, where U be a non-empty set (the universe of discourse) and R an equivalence relation on U, i.e., R is reflexive, symmetric, and transitive. The relation R decomposes the set U into disjoint classes in such a way that two elements x, y are in the same class iff $(x, y) \in R$. Let denote by U/R the quotient set of U by the relation R, and

$$U/R = \{X_1, X_2, \cdots, X_m\}$$

where X_i is an equivalence class of R, $i = 1, 2, \cdots, m$. If two elements x, y in U belong to the same equivalence class $X_i \in U/R$, we say that x and y are indistinguishable. The equivalence classes of R and the empty set $\emptyset$ are the elementary sets in the approximation space $< U, R >$. Given an arbitrary set $X \in 2^U$, in general it may not be possible to describe X precisely in $< U, R >$. One may characterize X by a pair of lower and upper approximations defined as follows [22]:

$$\underline{R}(X) = \bigcup_{X_i \subseteq X} X_i; \quad \overline{R}(X) = \bigcup_{X_i \cap X \neq \emptyset} X_i$$

That is, the lower approximation $\underline{R}(X)$ is the union of all the elementary sets which are subsets of X, and the upper approximation $\overline{R}(X)$ is the union of all the elementary sets which have a non-empty intersection with X. The interval $[\underline{R}(X), \overline{R}(X)]$ is the representation of an ordinary set X in the approximation space $< U, R >$ or simply called the rough set of X. The lower (resp., upper) approximation $\underline{R}(X)$ (resp., $\overline{R}(X)$) is interpreted as the collection of those elements of U that definitely (resp., possibly) belong to X. Further, we can define:

- a set $X \in 2^U$ is said to be definable (or exact) in $< U, R >$ iff $\underline{R}(X) = \overline{R}(X)$.
- for any $X, Y \in 2^U$, X is said to be roughly included in Y, denoted by $X \tilde{\subset} Y$, iff $\underline{R}(X) \subseteq \underline{R}(Y)$ and $\overline{R}(X) \subseteq \overline{R}(Y)$.
- X and Y is said to be roughly equal, denoted by $X \simeq_R Y$, in $< U, R >$ iff $\underline{R}(X) = \underline{R}(Y)$ and $\overline{R}(X) = \overline{R}(Y)$.

In [22], Pawlak discusses two numerical characterizations of imprecision of a subset X in the approximation space $< U, R >$: accuracy and roughness. Accuracy of X, denoted by $\alpha_R(X)$, is simply the ratio of the number of objects in its lower approximation to that in its upper approximation; namely

$$\alpha_R(X) = \frac{|\underline{R}(X)|}{|\overline{R}(X)|}$$

The roughness of X, denoted by $\rho_R(X)$, is defined by subtracting the accuracy from 1:

$$\rho_R(X) = 1 - \alpha_R(X) = 1 - \frac{|\underline{R}(X)|}{|\overline{R}(X)|}$$

Note that the lower the roughness of a subset, the better is its approximation. Further, the following observations are easily obtained:

1. As $\underline{R}(X) \subseteq X \subseteq \overline{R}(X)$, $0 \leq \rho_R(X) \leq 1$.
2. By convention, when $X = \emptyset$, $\underline{R}(X) = \overline{R}(X) = \emptyset$ and $\rho_R(X) = 0$.
3. $\rho_R(X) = 0$ if and only if X is definable in $< U, R >$.

2.3. Rough C-Means

Let $\underline{A}(\beta_i)$ and $\overline{A}(\beta_i)$ be the lower and upper approximations of cluster β_i, and $B(\beta_i) = \{\overline{A}(\beta_i) - \underline{A}(\beta_i)\}$ denote the boundary region of cluster β_i. In rough c-means algorithm, the concept of c-means algorithm is extended by viewing each cluster β_i as an interval or rough set. However, it is possible to define a pair of lower and upper bounds $[\underline{A}(\beta_i), \overline{A}(\beta_i)]$ or a rough set for every set $\beta_i \subseteq U$, U be the set of objects of concern [22]. The family of upper and lower bounds are required to follow some of the basic rough set properties such as:

1. an object x_j can be part of at most one lower bound;
2. $x_j \in \underline{A}(\beta_i) \Rightarrow x_j \in \overline{A}(\beta_i)$; and
3. an object x_j is not part of any lower bound $\Rightarrow x_j$ belongs to two or more upper bounds.

Incorporating rough sets into c-means algorithm, Lingras and West [15] introduced rough c-means algorithm. It adds the concept of lower and upper bounds into c-means algorithm. It classifies the object space into two parts - lower approximation and boundary region. The mean (centroid) is calculated based on the weighting average of the lower bound and boundary region. All the objects in lower approximation take the same weight w while all the objects in boundary take another weighting index $\tilde{w}$ $(= 1 - w)$ uniformly. Calculation of the centroid is modified to include the effects of lower as well as upper bounds. The modified centroid calculation for rough c-means is given by:

$$v_i = \begin{cases} w \times \mathcal{A} + \tilde{w} \times \mathcal{B} & \text{if } \underline{A}(\beta_i) \neq \emptyset, B(\beta_i) \neq \emptyset \\ \mathcal{A} & \text{if } \underline{A}(\beta_i) \neq \emptyset, B(\beta_i) = \emptyset \\ \mathcal{B} & \text{if } \underline{A}(\beta_i) = \emptyset, B(\beta_i) \neq \emptyset \end{cases} \tag{8}$$

$$\mathcal{A} = \frac{1}{|\underline{A}(\beta_i)|} \sum_{x_j \in \underline{A}(\beta_i)} x_j; \text{ and } \mathcal{B} = \frac{1}{|B(\beta_i)|} \sum_{x_j \in B(\beta_i)} x_j$$

β_i represents the ith cluster associated with the centroid v_i. $\underline{A}(\beta_i)$ and $B(\beta_i)$ represent the lower bound and the boundary region of cluster β_i. The parameter w and $\tilde{w}$ correspond to the relative importance of lower bound and boundary region, and $w + \tilde{w} = 1$. The main steps of rough c-means are as follows:

1. Assign initial means v_i, $i = 1, 2, \cdots, c$. Choose value for threshold δ.
2. For each object x_j, calculate distance d_{ij} between itself and the centroid v_i of cluster β_i.
3. If d_{ij} is minimum for $1 \leq i \leq c$ and $(d_{ij} - d_{kj}) \leq \delta$, then $x_j \in \overline{A}(\beta_i)$ and $x_j \in \overline{A}(\beta_k)$. Furthermore, x_j is not part of any lower bound.
4. Otherwise, $x_j \in \underline{A}(\beta_i)$ such that d_{ij} is minimum for $1 \leq i \leq c$. In addition, by properties of rough sets, $x_j \in \overline{A}(\beta_i)$.
5. Compute new centroid as per Equation 8.
6. Repeat steps 2 to 5 until no more new assignments can be made.

Incorporating both fuzzy and rough sets, recently Mitra et al. [18] have proposed rough-fuzzy c-means, where each cluster consists of a fuzzy lower approximation and a fuzzy boundary. If an object $x_j \in \underline{A}(\beta_i)$, then $\mu_{kj} = \mu_{ij}$ if $k = i$ and $\mu_{kj} = 0$ otherwise. That is, each object $x_j \in \underline{A}(\beta_i)$ takes a distinct weight, which is its fuzzy membership value. Thus, the weight of the object in lower approximation is inversely related to the relative distance of the object to all cluster prototypes.

In fact, the objects in lower approximation of a cluster should have similar influence on the corresponding centroid and cluster. Also, their weights should be independent of other centroids and clusters and should not be coupled with their similarity with respect to other clusters. Thus, the concept of fuzzy lower approximation, introduced in [18], reduces the weights of objects of lower approximation and effectively drifts the cluster centroids from their desired locations.

3. Rough-Fuzzy C-Means Algorithm

Incorporating both fuzzy and rough sets, next we describe a new c-means algorithm, termed as rough-fuzzy c-means (RFCM). The proposed c-means adds the concept of fuzzy membership of fuzzy sets, and lower and upper approximations of rough sets into c-means algorithm. While the membership of fuzzy sets enables efficient handling of overlapping partitions, the rough sets deal with uncertainty, vagueness, and incompleteness in class definition.

3.1. Objective Function

The proposed c-means partitions a set of n objects into c clusters by minimizing the objective function

$$J_{\text{RF}} = \begin{cases} w \times \mathcal{A}_1 + \tilde{w} \times \mathcal{B}_1 & \text{if } \underline{A}(\beta_i) \neq \emptyset,\, B(\beta_i) \neq \emptyset \\ \mathcal{A}_1 & \text{if } \underline{A}(\beta_i) \neq \emptyset,\, B(\beta_i) = \emptyset \\ \mathcal{B}_1 & \text{if } \underline{A}(\beta_i) = \emptyset,\, B(\beta_i) \neq \emptyset \end{cases} \tag{9}$$

$$\mathcal{A}_1 = \sum_{i=1}^{c} \sum_{x_j \in \underline{A}(\beta_i)} (\mu_{ij})^{\acute{m}_1} ||x_j - v_i||^2; \text{ and } \mathcal{B}_1 = \sum_{i=1}^{c} \sum_{x_j \in B(\beta_i)} (\mu_{ij})^{\acute{m}_1} ||x_j - v_i||^2$$

where the parameters w and $\tilde{w}$ $(= 1 - w)$ correspond to the relative importance of lower and boundary region. Note that, μ_{ij} has the same meaning of membership as that in fuzzy c-means.

In proposed RFCM, each cluster is represented by a centroid, a crisp lower approximation, and a fuzzy boundary (Fig. 1). The lower approximation influences the fuzziness of final partition. According to the definitions of lower approximations and boundary of rough sets, if an object $x_j \in \underline{A}(\beta_i)$, then $x_j \notin \underline{A}(\beta_k), \forall k \neq i$, and $x_j \notin B(\beta_i), \forall i$. That is, the object x_j is contained in β_i definitely. Thus, the weights of the objects in lower approximation of a cluster should be independent of other centroids and clusters, and should not be coupled with their similarity with respect to other centroids. Also, the objects in lower approximation of a cluster should have similar influence on the corresponding centroid and cluster. Whereas, if $x_j \in B(\beta_i)$, then the object x_j possibly belongs to β_i and potentially belongs to another cluster. Hence, the objects in boundary regions should have different influence on the centroids and clusters. So, in RFCM, the membership values of objects in lower approximation are $\mu_{ij} = 1$, while

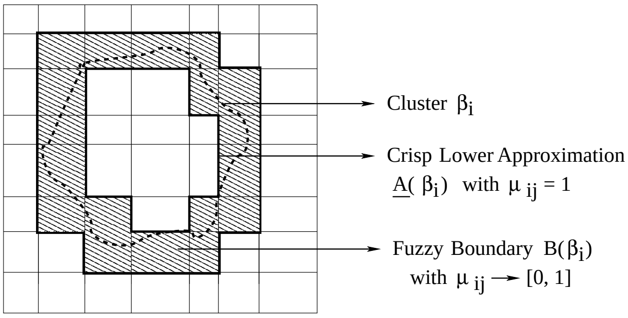


Figure 1. Rough-fuzzy c-means: cluster β_i is represented by crisp lower bound and fuzzy boundary

those in boundary region are the same as fuzzy c-means (Equation 3). In other word, the proposed c-means first partitions the data into two classes - lower approximation and boundary. Only the objects in boundary are fuzzified. Thus, $\mathcal{A}_1$ reduces to

$$\mathcal{A}_1 = \sum_{i=1}^{c} \sum_{x_j \in \underline{A}(\beta_i)} ||x_j - v_i||^2$$

and $\mathcal{B}_1$ has the same expression as that in Equation 9.

3.2. Cluster Prototypes

The new centroid is calculated based on the weighting average of the crisp lower approximation and fuzzy boundary. Computation of the centroid is modified to include the effects of both fuzzy memberships and lower and upper bounds. The modified centroid calculation for RFCM is obtained by solving Equation 9 with respect to v_i:

$$v_i^{\mathrm{RF}} = \begin{cases} w \times \mathcal{C}_1 + \tilde{w} \times \mathcal{D}_1 & \text{if } \underline{A}(\beta_i) \neq \emptyset, B(\beta_i) \neq \emptyset \\ \mathcal{C}_1 & \text{if } \underline{A}(\beta_i) \neq \emptyset, B(\beta_i) = \emptyset \\ \mathcal{D}_1 & \text{if } \underline{A}(\beta_i) = \emptyset, B(\beta_i) \neq \emptyset \end{cases} \quad (10)$$

$$\mathcal{C}_1 = \frac{1}{|\underline{A}(\beta_i)|} \sum_{x_j \in \underline{A}(\beta_i)} x_j; \text{ and } \mathcal{D}_1 = \frac{1}{n_i} \sum_{x_j \in B(\beta_i)} (\mu_{ij})^{\acute{m}_1} x_j; \text{ where } n_i = \sum_{x_j \in B(\beta_i)} (\mu_{ij})^{\acute{m}_1}$$

$|\underline{A}(\beta_i)|$ represents the cardinality of $\underline{A}(\beta_i)$.

Thus, the cluster prototypes (centroids) depend on the parameters w and $\tilde{w}$, and fuzzifier $\acute{m}_1$ rule their relative influence. The correlated influence of these parameters and fuzzifier, makes it somewhat difficult to determine their optimal values. Since the objects lying in lower approximation definitely belong to a cluster, they are assigned a higher weight w compared to $\tilde{w}$ of the objects lying in boundary region. Hence, for RFCM, the values are given by $0 < \tilde{w} < w < 1$.

3.3. Fundamental Properties

From the above discussions, we can get the following properties of RFCM algorithm.

1. $\bigcup \overline{A}(\beta_i) = U$, U be the set of objects of concern.
2. $\underline{A}(\beta_i) \cap \underline{A}(\beta_k) = \emptyset, \forall i \neq k$.
3. $\underline{A}(\beta_i) \cap B(\beta_i) = \emptyset, \forall i$.
4. $\exists i, k, B(\beta_i) \cap B(\beta_k) \neq \emptyset$.
5. $\mu_{ij} = 1, \forall x_j \in \underline{A}(\beta_i)$.
6. $\mu_{ij} \in [0, 1], \forall x_j \in B(\beta_i)$.

Let us briefly comment on some properties of RFCM. The property 2 says that if an object $x_j \in \underline{A}(\beta_i) \Rightarrow x_j \notin \underline{A}(\beta_k), \forall k \neq i$. That is, the object x_j is contained in β_i definitely. The property 3 establishes the fact that if $x_j \in \underline{A}(\beta_i) \Rightarrow x_j \notin B(\beta_i)$, - that is, an object may not be in both lower and boundary region of a cluster β_i. The property 4 says that if $x_j \in B(\beta_i) \Rightarrow \exists k, x_j \in B(\beta_k)$. It means an object $x_j \in B(\beta_i)$ possibly belongs to β_i and potentially belongs to other cluster. The properties 5 and 6 are of great importance in computing the objective function J_{RF} and the cluster prototype v^{RF}. They say that the membership values of the objects in lower approximation are $\mu_{ij} = 1$, while those in boundary region are the same as fuzzy c-means. That is, each cluster β_i consists of a crisp lower approximation $\underline{A}(\beta_i)$ and a fuzzy boundary $B(\beta_i)$.

3.4. Details of the Algorithm

Approximate optimization of J_{RF} (Equation 9) by the RFCM is based on Picard iteration through Equations 3 and 10. This type of iteration is called alternating optimization. The process starts by randomly choosing c objects as the centroids of the c clusters. The fuzzy memberships of all objects are calculated using Equation 3.

Let $\mu_i = (\mu_{i1}, \cdots, \mu_{ij}, \cdots, \mu_{in})$ represent the fuzzy cluster β_i associated with the centroid v_i. After computing μ_{ij} for c clusters and n objects, the values of μ_{ij} for each object x_j are sorted and the difference of two highest memberships of x_j is compared with a threshold value δ. Let μ_{ij} and μ_{kj} be the highest and second highest memberships of x_j. If $(\mu_{ij} - \mu_{kj}) > \delta$, then $x_j \in \underline{A}(\beta_i)$ as well as $x_j \in \overline{A}(\beta_i)$, otherwise $x_j \in \overline{A}(\beta_i)$ and $x_j \in \overline{A}(\beta_k)$. After assigning each object in lower approximations or boundary regions of different clusters based on δ, memberships μ_{ij} of the objects are modified. The values of μ_{ij} are set to 1 for the objects in lower approximations, while those in boundary regions are remain unchanged. The new centroids of the clusters are calculated as per Equation 10. The main steps of RFCM algorithm proceed as follows:

1. Assign initial centroids v_i, $i = 1, 2, \cdots, c$. Choose values for fuzzifier $\acute{m}_1$, and thresholds ϵ and δ. Set iteration counter $t = 1$.
2. Compute μ_{ij} by Equation 3 for c clusters and n objects.
3. If μ_{ij} and μ_{kj} be the two highest memberships of x_j and $(\mu_{ij} - \mu_{kj}) \leq \delta$, then $x_j \in \overline{A}(\beta_i)$ and $x_j \in \overline{A}(\beta_k)$. Furthermore, x_j is not part of any lower bound.

4. Otherwise, $x_j \in \underline{A}(\beta_i)$. In addition, by properties of rough sets, $x_j \in \overline{A}(\beta_i)$.

5. Modify μ_{ij} considering lower and boundary regions for c clusters and n objects.

6. Compute new centroid as per Equation 10.

7. Repeat steps 2 to 7, by incrementing t, until $|\mu_{ij}(t) - \mu_{ij}(t-1)| > \epsilon$.

The performance of RFCM depends on the value of δ, which determines the class labels of all the objects. In other word, the RFCM partitions the data set into two classes - lower approximation and boundary, based on the value of δ. In practice we find that the following definition works well:

$$\delta = \frac{1}{n}\sum_{j=1}^{n}(\mu_{ij} - \mu_{kj}) \quad (11)$$

where n is the total number of objects, μ_{ij} and μ_{kj} are the highest and second highest memberships of x_j. That is, the value of δ represents the average difference of two highest memberships of all the objects in the data set. A good clustering procedure should make the value of δ as high as possible. The value of δ is, therefore, data dependent.

4. Quantitative Measures

In this section we propose some quantitative indices to evaluate the performance of rough-fuzzy clustering algorithm incorporating the concepts of rough sets [22].

α **Index:** It is given by

$$\alpha = \frac{1}{c}\sum_{i=1}^{c}\frac{w\mathrm{A}_i}{w\mathrm{A}_i + \tilde{w}\mathrm{B}_i} \quad (12)$$

$$\text{where } \mathrm{A}_i = \sum_{x_j \in \underline{A}(\beta_i)} (\mu_{ij})^{\acute{m}_1} = |\underline{A}(\beta_i)|; \text{ and } \mathrm{B}_i = \sum_{x_j \in B(\beta_i)} (\mu_{ij})^{\acute{m}_1} \quad (13)$$

μ_{ij} represents the probabilistic memberships of object x_j in cluster β_i. The parameters w and $\tilde{w}$ correspond to the relative importance of lower and boundary region.

The α index represents the average accuracy of c clusters. It is the average of the ratio of the number of objects in lower approximation to that in upper approximation of each cluster. In effect, it captures the average degree of completeness of knowledge about all clusters. A good clustering procedure should make all objects as similar to their centroids as possible. The α index increases with increase in similarity within a cluster. Therefore, for a given data set and c value, the higher the similarity values within the clusters, the higher would be the α value. The value of α also increases with c. In an extreme case when the number of clusters is maximum, i.e., $c = n$, the total number of objects in the data set, the value of $\alpha = 1$. When $\overline{A}(\beta_i) = \underline{A}(\beta_i)$, $\forall i$, that is, all the clusters $\{\beta_i\}$ are exact or definable, then we have $\alpha = 1$. Whereas if $\overline{A}(\beta_i) = B(\beta_i)$, $\forall i$, the value of $\alpha = 0$. Thus, $0 \leq \alpha \leq 1$.

ϱ **Index:** The ϱ index represents the average roughness of c clusters and is defined by subtracting the average accuracy α from 1:

$$\varrho = 1 - \alpha = 1 - \frac{1}{c}\sum_{i=1}^{c}\frac{w\mathrm{A}_i}{w\mathrm{A}_i + \tilde{w}\mathrm{B}_i} \quad (14)$$

where A_i and B_i are given by Equation 13. Note that the lower the value of ϱ, the better is the overall clusters approximations. Also, $0 \leq \varrho \leq 1$. Basically, ϱ index represents the average degree of incompleteness of knowledge about all clusters.

$\alpha^\star$ **Index:** It can be defined as

$$\alpha^\star = \frac{\mathrm{C}}{\mathrm{D}}; \text{ where } \mathrm{C} = \sum_{i=1}^{c} w\mathrm{A}_i; \text{ and } \mathrm{D} = \sum_{i=1}^{c} \{w\mathrm{A}_i + \tilde{w}\mathrm{B}_i\} \tag{15}$$

where A_i and B_i are given by Equation 13. The $\alpha^\star$ index represents the accuracy of approximation of all clusters. It captures the exactness of approximate clustering. A good clustering procedure should make the value of $\alpha^\star$ as high as possible. The $\alpha^\star$ index maximizes the exactness of approximate clustering.

γ **Index:** It is the ratio of the total number of objects in lower approximations of all clusters to the cardinality of the universe of discourse U and is given by

$$\gamma = \frac{\mathrm{R}}{\mathrm{S}}; \text{ where } \mathrm{R} = \sum_{i=1}^{c} |\underline{A}(\beta_i)|; \text{ and } \mathrm{S} = |U| = n. \tag{16}$$

The γ index basically represents the quality of approximation of a clustering algorithm.

5. Performance Analysis

The performance of proposed RFCM algorithm is compared extensively with that of different c-means algorithms. These involve different combinations of the individual components of the hybrid scheme. The algorithms compared are hard c-means (HCM), fuzzy c-means (FCM) [4, 9], possibilistic c-means (PCM) [13, 14], fuzzy-possibilistic c-means (FPCM) [19], rough c-means (RCM) [15], and rough-fuzzy c-means of Mitra et al. (RFCM$^{\mathbf{MBP}}$) [18]. All the methods are implemented in C language and run in LINUX environment having machine configuration Pentium IV, 3.2 GHz, 1 MB cache, and 1 GB RAM. The input parameters used, which are held constant across all runs, are as follows:

Values of fuzzifiers $\acute{m}_1 = 2.0$; and $\acute{m}_2 = 2.0$
Value of threshold $\epsilon = 0.00001$; and Value of $w = 0.95$

To analyze the performance of the proposed method, the experimentation has been done in two parts. In the first part, we have used synthetic as well as real data sets. In the second part, we present the results on segmentation of brain MR images. The major metrics for evaluating the performance of different algorithms are the indices proposed in Section 4 such as α, ϱ, $\alpha^\star$, and γ, as well as some existing measures like Davies-Bouldin (DB) and Dunn index [5], which are described next.

Davies-Bouldin Index: The Davies-Bouldin (DB) index [5] is a function of the ratio of sum of within-cluster distance to between-cluster separation and is given by

$$\mathrm{DB} = \frac{1}{c} \sum_{i=1}^{c} \max_{k \neq i} \{ \frac{S(v_i) + S(v_k)}{d(v_i, v_k)} \} \text{ for } 1 \leq i, k \leq c. \tag{17}$$

The DB index minimizes the within-cluster distance $S(v_i)$ and maximizes the between-cluster separation $d(v_i, v_k)$. Therefore, for a given data set and c value, the higher the similarity values within the clusters and the between-cluster separation, the lower would be the DB index value. A good clustering procedure should make the value of DB index as low as possible.

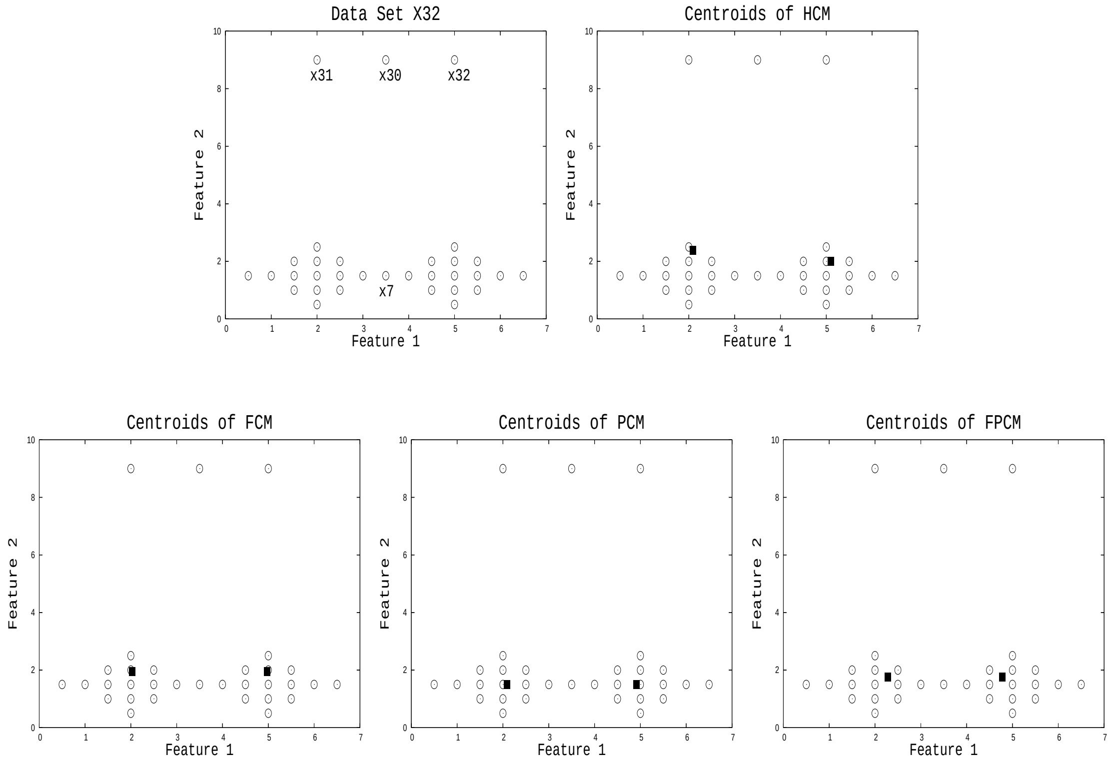
Dunn Index: Dunn's index [5] is also designed to identify sets of clusters that are compact and well separated. Dunn's index maximizes

$$\text{Dunn} = \min_i\{\min_{k \neq i}\{\frac{d(v_i, v_k)}{\max_l S(v_l)}\}\} \quad \text{for} \;\; 1 \leq i, k, l \leq c. \tag{18}$$

5.1. Synthetic Data Set: $X32$

The synthetic data set $X32$ consists of $n = 32$ objects in $\Re^2$ with two clusters. Fig. 2 depicts the scatter plots of the data set $X32$. The objects x_{30}, x_{31}, and x_{32} are outliers (noise), and the object x_7 is the so called inlier or bridge. Two randomly generated initial centroids, along with two scale parameters and the final prototypes of different c-means, are reported in Table 1. Fig. 2 represents the scatter plots of the data set $X32$ along with the clusters prototypes obtained using different c-means algorithms. The objects of $X32$ are represented by $\odot$, while 'box' depict the positions of cluster prototypes.

Table 2 reports the values of α, ϱ, $\alpha^\star$, γ, J_{RF}, Dunn, and DB index of RFCM algorithm over different iterations. All the results reported in Table 2 show that as the number of iteration increases, the values of ϱ, J_{RF}, and DB index decrease, while the values of α, $\alpha^\star$, γ, and Dunn index increase. Finally, all the indices are saturated when RFCM terminates after 4th iterations. Thus, the proposed indices (e.g., α, ϱ, $\alpha^\star$, and γ) can be used to act as the objective function of rough-fuzzy clustering as they reflect good quantitative measures like existing DB and Dunn index.



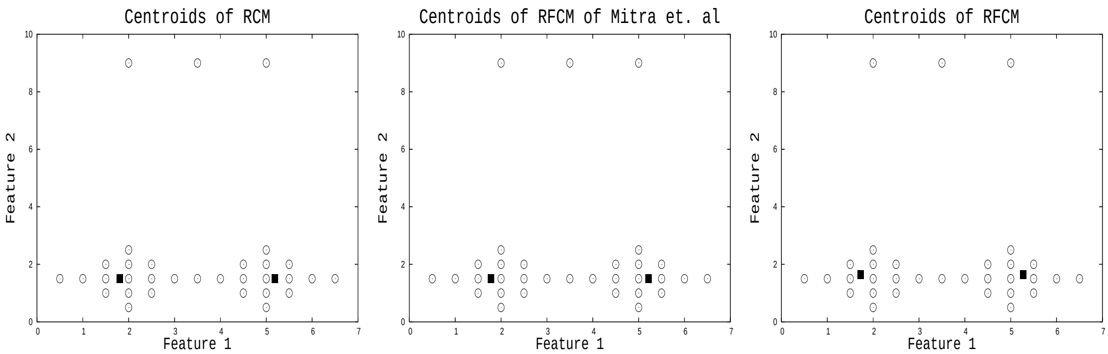


Figure 2. Example data set $X32$ and clusters prototypes of different c-means algorithms

Table 1. Cluster Prototypes of Different C-Means for $X32$

Algorithms	Centroid 1	Centroid 2
Initial	2.088235; 2.382353	5.100000; 2.000000
Scale	η_1 =4.553732	η_2 =3.697741
HCM	2.088235; 2.382353	5.100000; 2.000000
FCM	2.025431; 1.943642	4.974481; 1.943406
PCM	2.087332; 1.500811	4.912668; 1.500811
FPCM	2.281087; 1.749121	4.782719; 1.765219
RCM	1.807862; 1.500375	5.192139; 1.500375
RFCM$^{\mathbf{MBP}}$	1.783023; 1.500408	5.216976; 1.500408
RFCM	1.727380; 1.636481	5.272620; 1.636481

Table 2. Performance of RFCM over Different Iterations

Itr.	ϱ Index	DB Index	J_{RF}	α Index	$\alpha^{\star}$ Index	γ Index	Dunn Index
1	0.000081	0.159813	17.002853	0.999919	0.999905	0.562500	10.022817
2	0.000053	0.140268	16.035562	0.999947	0.999963	0.625000	11.543916
3	0.000037	0.137502	12.191273	0.999963	0.999974	0.812500	13.798013
4	0.000009	0.123709	9.073391	0.999991	0.999991	0.812500	14.621336
5	0.000009	0.123709	9.073391	0.999991	0.999991	0.812500	14.621336

Table 3 provides comparative results of different c-means algorithms. The rough set based clustering algorithms (RCM, RFCM$^{\mathbf{MBP}}$, and RFCM) are found to improve the performance in terms of DB and Dunn index over other algorithms. It is also observed that the proposed RFCM algorithm perform better than FCM, PCM, FPCM, RCM, and RFCM$^{\mathbf{MBP}}$, although it is expected that both PCM and FPCM perform well in noisy environment. Finally, Table 4 shows the comparative results of different rough-fuzzy clustering algorithms in terms of α, ϱ, $\alpha^{\star}$, and γ. The performance of RFCM is better than that of RFCM$^{\mathbf{MBP}}$.

Table 3. Performance of Different C-Means Algorithms

Algorithms	DB Index	Dunn Index
HCM	0.266063	5.934918
FCM	0.210139	9.022365
PCM	0.183069	10.309421
FPCM	0.276837	6.894526
RCM	0.127478	14.252816
RFCM$^{\mathbf{MBP}}$	0.125088	14.491059
RFCM	0.123709	14.621336

Table 4. Quantitative Evaluation of Rough-Fuzzy Clustering

Methods	α Index	ϱ Index	$\alpha^{\star}$ Index	γ Index
RFCM$^{\mathbf{MBP}}$	0.999953	0.000047	0.999942	0.625000
RFCM	0.999991	0.000009	0.999991	0.812500

5.2. Real Data Set: Iris

This subsection demonstrates the performance of different c-means algorithms on Iris data set, with $c = 2$ and 3. The Iris data set is a four-dimensional data set containing 50 samples each of three types of Iris flowers. One of the three clusters (class 1) is well separated from the other two, while classes 2 and 3 have some overlap. The data set can be downloaded from http://www.ics.uci.edu/∼mlearn.

Several runs have been made for each of the c-means algorithms on Iris data set with different choices of parameters. The final prototypes of different c-means algorithms, along with initial centroids and scale parameters, are provided in Table 5 for $c = 2$.

Table 5. Cluster Prototypes of Different C-Means for Iris Data ($c = 2$)

Algorithms	Centroid 1	Centroid 2
Initial	4.5 2.3 1.3 0.3	6.3 2.7 4.9 1.8
Scale	$\eta_1 = 0.651171$	$\eta_2 = 1.154369$
FCM	5.02 3.37 1.57 0.29	6.34 2.91 5.01 1.73
PCM	5.04 3.41 1.47 0.24	6.17 2.88 4.76 1.60
FPCM	5.02 3.38 1.56 0.28	6.31 2.90 4.98 1.72
RCM	5.01 3.42 1.46 0.24	6.40 2.94 5.11 1.77
RFCM$^{\mathbf{MBP}}$	5.00 3.42 1.46 0.24	6.38 2.92 5.08 1.76
RFCM	5.01 3.42 1.46 0.24	6.40 2.94 5.11 1.77

Tables 6-7 depict the best results obtained using different c-means algorithms for $c = 2$. In Table 6, the performance of different algorithms is reported with respect to DB and Dunn index. The results reported in Table 6 establish the fact that although each c-means algorithm generates good prototypes with lower values of DB index and higher values of Dunn index for $c = 2$, RFCM provides best result having lowest DB index and highest Dunn index. The results of other versions of rough clustering are quit similar to that of RFCM. Finally, Table 7 compares the performance of different rough-fuzzy clustering with respect to α, ϱ, $\alpha^\star$, and γ. The proposed RFCM performs better than RFCM$^{\mathbf{MBP}}$.

Table 6. Results on Iris Data Set ($c = 2$)

Algorithms	DB Index	Dunn Index
HCM	0.117736	12.081416
FCM	0.114973	12.359162
PCM	0.121456	9.993713
FPCM	0.113514	11.875308
RCM	0.105311	13.495355
RFCM$^{\mathbf{MBP}}$	0.106170	13.425817
RFCM	0.105310	13.495560

Table 7. Quantitative Evaluation of Rough-Fuzzy Clustering

Methods	α Index	ϱ Index	$\alpha^\star$ Index	γ Index
RFCM$^{\mathbf{MBP}}$	0.999991	0.000009	0.999989	0.812500
RFCM	0.999994	0.000006	0.999994	0.906667

Next, the performance of different c-means algorithms on Iris data set is reported for $c = 3$. Several runs have been made with different initializations and different choices of parameters. The final prototypes of different c-means algorithms, along with three random initial centroids and scale parameters, are reported in Table 8.

Table 8. Cluster Prototypes of Different C-Means for Iris Data ($c = 3$)

Methods	Centroid 1	Centroid 2	Centroid 3
Initial	5.8 2.7 5.1 1.9	5.0 2.0 3.5 1.0	5.1 3.5 1.4 0.3
Scale	$\eta_1 = 0.689405$	$\eta_2 = 0.582364$	$\eta_3 = 0.344567$
FCM	6.78 3.05 5.65 2.05	5.89 2.76 4.36 1.40	5.00 3.40 1.49 0.25
PCM	6.17 2.88 4.76 1.61	6.17 2.88 4.76 1.61	5.04 3.41 1.47 0.24
FPCM	6.62 3.01 5.46 1.99	5.92 2.79 4.40 1.41	5.00 3.40 1.49 0.25
RCM	6.87 3.09 5.79 2.12	5.69 2.66 4.11 1.26	5.01 3.42 1.46 0.24
RFCM$^{\mathbf{MBP}}$	6.88 3.09 5.79 2.12	5.70 2.68 4.11 1.26	5.00 3.41 1.47 0.24
RFCM	6.87 3.09 5.79 2.12	5.69 2.66 4.11 1.26	5.01 3.42 1.46 0.24

In each case, except PCM, all the c-means algorithms generate good prototypes. The final prototypes of FCM are used to initialize PCM and FPCM. Even three initial centroids belong to three different classes, PCM generates coincident clusters. That is, two of three final prototypes are identical in case of PCM.

Table 9. Results on Iris Data Set ($c = 3$)

Algorithms	DB Index	Dunn Index
FCM	0.321984	4.316334
FPCM	0.480586	2.627405
RCM	0.225069	6.755984
RFCM$^{\mathbf{MBP}}$	0.224806	6.907512
RFCM	0.224164	6.936064

In Iris data set, since class 2 and class 3 overlap, it may be thought of having two clusters. But, to design a classifier, at least three clusters have to be identified. Thus for such applications, RFCM will be more useful than FCM, PCM, and FPCM, because it is not sensitive to noise, can avoid coincident clusters, and their DB and Dunn index values are far better than that of FCM, PCM, and FPCM, as reported in Table 9.

Table 10. Quantitative Evaluation of Rough-Fuzzy Clustering

Methods	α Index	ϱ Index	$\alpha^\star$ Index	γ Index
RFCM$^{\mathbf{MBP}}$	0.999971	0.000029	0.999963	0.625000
RFCM	0.999986	0.000014	0.999988	0.800000

Finally, Table 10 provides comparative results of different rough-fuzzy clustering with respect to α, ϱ, $\alpha^\star$, and γ. All the results reported here confirm that the performance of RFCM is better than that of RFCM$^{\mathbf{MBP}}$ for Iris data set having $c = 3$.

5.3. Segmentation of Brain MR Images

In this subsection, we present the results of different c-means algorithms on segmentation of brain MR images. Above 100 MR images with different sizes and 16 bit gray levels are tested with different c-means algorithms. All the brain MR images are collected from Advanced Medicare and Research Institute, Salt Lake, Kolkata, India. The comparative performance of different c-means is reported with respect to α, ϱ, $\alpha^\star$, γ, DB, and Dunn index, as well as the β index [20], which is defined next.

β **Index:** The β-index of Pal et al. [20] is defined as the ratio of the total variation and within-cluster variation, and is given by

$$\beta = \frac{\mathrm{N}}{\mathrm{M}}; \text{ where } \mathrm{N} = \sum_{i=1}^{c}\sum_{j=1}^{n_i} ||x_{ij} - \overline{v}||^2; \;\; \mathrm{M} = \sum_{i=1}^{c}\sum_{j=1}^{n_i} ||x_{ij} - v_i||^2; \text{ and } \sum_{i=1}^{c} n_i = n; \tag{19}$$

n_i is the number of objects in the ith cluster ($i = 1, 2, \cdots, c$), n is the total number of objects, x_{ij} is the jth object in cluster i, v_i is the mean or centroid of ith cluster, and $\overline{v}$ is the mean of n objects. For a given image and c value, the higher the homogeneity within the segmented regions, the higher would be the β value. The value of β also increases with c.

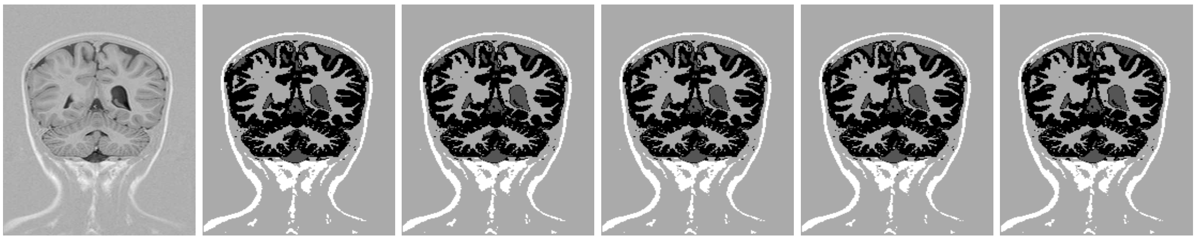


Figure 3. IMAGE-20497774: original and segmented versions of HCM, FCM, RCM, RFCM$^{\text{MBP}}$, and RFCM

Consider Fig. 3 as an example, which represents an MR image (IMAGE-20497774) along with the segmented images obtained using different c-means algorithms. Each image is of size 256×180 with 16 bit gray levels. So, the number of objects in the data set of IMAGE-20497774 is 46080. Table 11 depicts the values of DB index, Dunn index, and β index of FCM and RFCM for different values of c on the data set of IMAGE-20497774. The results reported here with respect to DB and Dunn index confirm that both FCM and RFCM achieve their best results for $c = 4$ (background, gray matter, white matter, and cerebro-spinal fluid). Also, the value of β index, as expected, increases with increase in the value of c. For a particular value of c, the performance of RFCM is better than that of FCM.

Table 11. Performance of FCM and RFCM on IMAGE-20497774

Value of c	DB Index		Dunn Index		β Index	
	FCM	RFCM	FCM	RFCM	FCM	RFCM
2	0.51	0.21	2.30	6.17	2.15	2.19
3	0.25	0.17	1.11	1.62	3.55	3.74
4	0.16	0.15	1.50	1.64	9.08	9.68
5	0.39	0.17	0.10	0.64	10.45	10.82
6	0.20	0.19	0.66	1.10	16.93	17.14
7	0.23	0.27	0.98	0.12	21.63	22.73
8	0.34	0.27	0.09	0.31	25.82	26.38
9	0.32	0.28	0.12	0.13	31.75	32.65
10	0.30	0.24	0.08	0.12	38.04	39.31

Fig. 4 shows the scatter plots of the highest and second highest memberships of all the objects in the data set of IMAGE-20497774 at first and final iterations respectively, considering $w = 0.95$, $\acute{m}_1 = 2.0$, and $c = 4$. The diagonal line represents the zone where two highest memberships of objects are equal. From Fig. 4, it is observed that though the average difference between two highest memberships of the objects are very low at first iteration ($\delta = 0.145$), they become ultimately very high at the final iteration ($\delta = 0.652$). In Fig. 5, the variations of different indices like ϱ, $\alpha^{\star}$, and γ over different iterations are reported for the IMAGE-20497774 data set. All the results reported in Fig. 5 clearly establish the fact that as the iteration increases, the value of ϱ decreases and the values of $\alpha^{\star}$ and γ increase. Ultimately, all the values are saturated after 20 iterations. That is, the proposed rough sets based indices provide good quantitative measures for rough-fuzzy clustering.

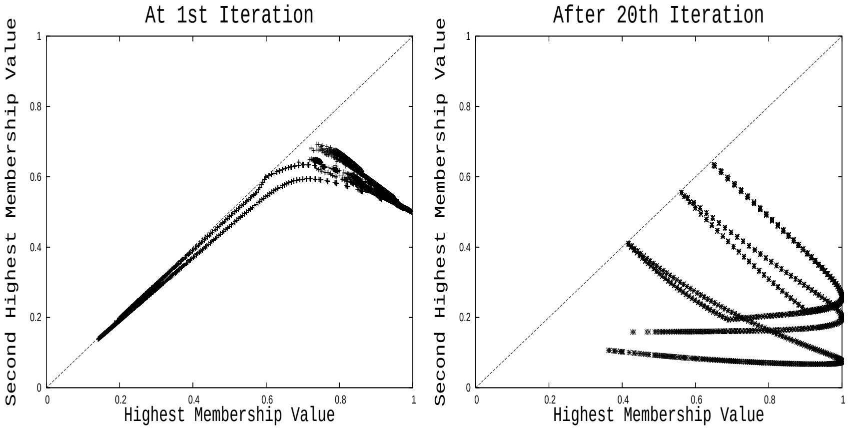


Figure 4. Scatter plots of two highest membership values of all the objects in the data set of IMAGE-20497774

Table 12 compares the performance of different c-means algorithms on some brain MR images with respect to DB, Dunn, and β index considering $c = 4$ (back-ground, gray matter, white matter, and CSF). The original images along with the segmented versions of different c-means are shown in Figs. 6-8. All the results reported in Table 12 and Figs. 6-8 confirm that the proposed algorithm produces segmented images more promising than do the conventional methods. Some of the existing algorithms like PCM and FPCM have failed to produce multiple segments as they generate coincident clusters even when they have been initialized with the final prototypes of FCM. Also, the values of DB, Dunn, and β index of RFCM are better compared to other c-means algorithms.

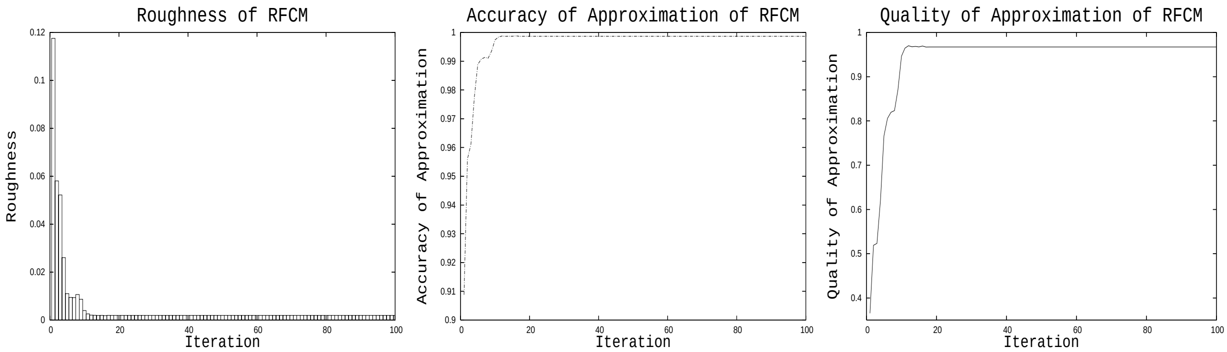


Figure 5. Variation of ϱ, $\alpha^{\star}$, and γ over different iterations for IMAGE-20497774 considering $\acute{m}_1 = 2.0$

Following conclusions can be drawn from the results reported in this paper:

1. It is observed that RFCM is superior to other c-means algorithms. However, RFCM requires higher time compared to FCM/PCM. But, the performance of RFCM is significantly higher than other c-means. Also, RFCM performs better than RFCM$^{\mathbf{MBP}}$.

Table 12. Performance of Different C-Means Algorithms

Data Set	Algorithms	DB Index	Dunn Index	β Index
IMAGE 20497761	HCM	0.16	2.13	12.07
	FCM	0.14	2.26	12.92
	RCM	0.15	2.31	11.68
	RFCM^{MBP}	0.14	2.34	9.99
	RFCM	0.13	2.39	13.06
IMAGE 20497763	HCM	0.18	1.88	12.02
	FCM	0.16	2.02	12.63
	RCM	0.15	2.14	12.59
	RFCM^{MBP}	0.15	2.08	10.59
	RFCM	0.11	2.12	13.30
IMAGE 20497774	HCM	0.18	1.17	8.11
	FCM	0.16	1.50	9.08
	RCM	0.17	1.51	9.10
	RFCM^{MBP}	0.15	1.51	9.02
	RFCM	0.15	1.64	9.68
IMAGE 20497777	HCM	0.17	2.01	8.68
	FCM	0.16	2.16	9.12
	RCM	0.15	2.34	9.28
	RFCM^{MBP}	0.15	2.33	9.69
	RFCM	0.14	2.39	9.81

2. Use of rough sets and fuzzy memberships adds a small computational load to HCM algorithm; however the corresponding integrated method (RFCM) show a definite increase in Dunn index and decrease in DB index.

3. The proposed indices such as α, ϱ, $\alpha^{\star}$, and γ based on the theory of rough sets provide good quantitative measures for rough-fuzzy clustering. The values of these indices reflect the quality of clustering.

The best performance of the proposed RFCM algorithm in terms of α, ϱ, $\alpha^{\star}$, γ, DB, Dunn, and β is achieved due to the following reasons:

1. the concept of crisp lower bound and fuzzy boundary of the proposed algorithm deals with uncertainty, vagueness, and incompleteness in class definition; and

2. membership function of RFCM handles efficiently overlapping partitions.

In effect, good cluster prototypes are obtained using the proposed RFCM algorithm.

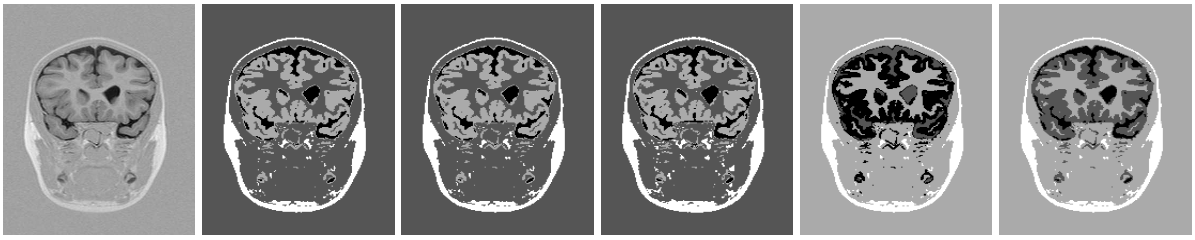


Figure 6. IMAGE-20497761: original and segmented versions of HCM, FCM, RCM, RFCM$^{\mathbf{MBP}}$, and RFCM

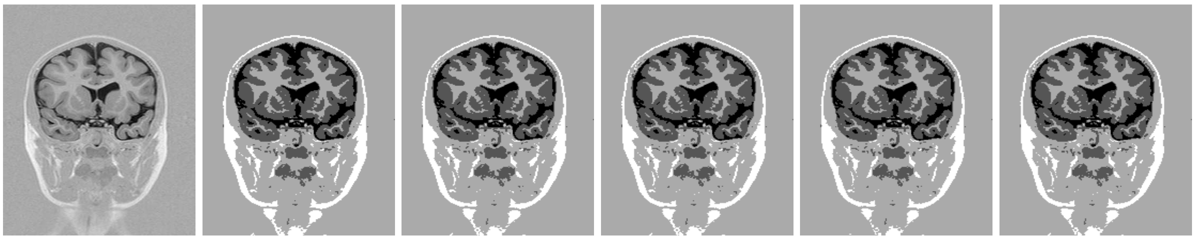


Figure 7. IMAGE-20497763: original and segmented versions of HCM, FCM, RCM, RFCM$^{\mathbf{MBP}}$, and RFCM

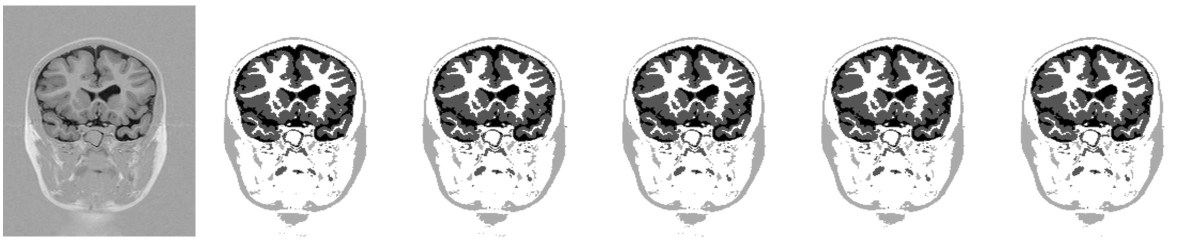


Figure 8. IMAGE-20497777: original and segmented versions of HCM, FCM, RCM, RFCM$^{\mathbf{MBP}}$, and RFCM

6. Conclusion

The contribution of the paper lies in developing a hybrid methodology, which integrates judiciously rough sets and fuzzy c-means algorithm. This formulation is geared towards maximizing the utility of both rough sets and fuzzy sets with respect to knowledge discovery tasks. Several new measures are defined based on rough sets to evaluate the performance of rough-fuzzy clustering algorithms. Finally, the effectiveness of the proposed algorithm is demonstrated, along with a comparison with other related algorithms, on a set of synthetic as well as real life data sets.

Although our methodology of integrating rough sets, fuzzy sets, and c-means algorithm has been efficiently demonstrated for synthetic and real data sets, along with the segmentation of brain MR images, the concept can be applied to other unsupervised classification problems. Some of the indices (e.g., α, $\alpha^{\star}$, ϱ, and γ) used for evaluating the quality of the proposed algorithm may be used in a suitable combination to act as the objective function of an evolutionary algorithm, for rough-fuzzy clustering.

References

[1] Asharaf, S., Murty, M. N.: A Rough Fuzzy Approach to Web Usage Categorization, *Fuzzy Sets and Systems*, **148**, 2004, 119–129.

[2] Banerjee, M., Mitra, S., Pal, S. K.: Rough-Fuzzy MLP: Knowledge Encoding and Classification, *IEEE Transactions on Neural Networks*, **9**(6), 1998, 1203–1216.

[3] Barni, M., Cappellini, V., Mecocci, A.: Comments on A Possibilistic Approach to Clustering, *IEEE Transactions on Fuzzy Systems*, **4**(3), 1996, 393–396.

[4] Bezdek, J. C.: *Pattern Recognition with Fuzzy Objective Function Algorithm*, New York: Plenum, 1981.

[5] Bezdek, J. C., Pal, N. R.: Some New Indexes for Cluster Validity, *IEEE Transactions on System, Man, and Cybernetics, Part B, Cybernetics*, **28**, 1988, 301–315.

[6] Dubois, D., H.Prade: Rough Fuzzy Sets and Fuzzy Rough Sets, *International Journal of General Systems*, **17**, 1990, 191–209.

[7] Dubois, D., Prade, H.: Putting Fuzzy Sets and Rough Sets Together, *in: R. Slowiniski (Ed.), Intelligent Decision Support*, 1992, 203–232.

[8] Duda, R. O., Hart, P. E., Stork, D. G.: *Pattern Classification and Scene Analysis*, John Wiley & Sons, New York, 1999.

[9] Dunn, J. C.: A Fuzzy Relative of the ISODATA Process and its Use in Detecting Compact, Well-Separated Clusters, *J. Cybern.*, **3**, 1974, 32–57.

[10] Jain, A. K., Dubes, R. C.: *Algorithms for Clustering Data*, Englewood Cliffs, N.J.: Prentice Hall, 1988.

[11] Jain, A. K., Murty, M. N., Flynn, P. J.: Data Clustering: A Review, *ACM Computing Surveys*, **31**(3), 1999, 264–323.

[12] Jensen, R., Shen, Q.: Fuzzy-Rough Attribute Reduction with Application to Web Categorization, *Fuzzy Sets and Systems*, **141**, 2004, 469–485.

[13] Krishnapuram, R., Keller, J. M.: A Possibilistic Approach to Clustering, *IEEE Transactions on Fuzzy Systems*, **1**(2), 1993, 98–110.

[14] Krishnapuram, R., Keller, J. M.: The Possibilistic C-Means Algorithm: Insights and Recommendations, *IEEE Transactions on Fuzzy Systems*, **4**(3), 1996, 385–393.

[15] Lingras, P., West, C.: Interval Set Clustering of Web Users with Rough K-Means, *Journal of Intelligent Information Systems*, **23**(1), 2004, 5–16.

[16] Maji, P., Pal, S. K.: Rough-Fuzzy C-Medoids Algorithm and Selection of Bio-Basis for Amino Acid Sequence Analysis, *IEEE Transactions on Knowledge and Data Engineering*, **19**(6), 2007, 859–872.

[17] McQueen, J.: Some Methods for Classification and Analysis of Multivariate Observations, *Proc. Fifth Berkeley Symp. Math. Statistics and Probability*, 1967, 281–297.

[18] Mitra, S., Banka, H., Pedrycz, W.: Rough-Fuzzy Collaborative Clustering, *IEEE Transactions on Systems, Man, and Cybernetics, Part B, Cybernetics*, **36**, 2006, 795–805.

[19] Pal, N. R., Pal, K., Keller, J. M., Bezdek, J. C.: A Possibilistic Fuzzy C-Means Clustering Algorithm, *IEEE Transactions on Fuzzy Systems*, **13**(4), 2005, 517–530.

[20] Pal, S. K., Ghosh, A., Sankar, B. U.: Segmentation of Remotely Sensed Images with Fuzzy Thresholding, and Quantitative Evaluation, *International Journal of Remote Sensing*, **21**(11), 2000, 2269–2300.

[21] Pal, S. K., Mitra, S., Mitra, P.: Rough-Fuzzy MLP: Modular Evolution, Rule Generation, and Evaluation, *IEEE Transactions on Knowledge and Data Engineering*, **15**(1), 2003, 14–25.

[22] Pawlak, Z.: *Rough Sets, Theoretical Aspects of Resoning About Data*, Dordrecht, The Netherlands: Kluwer, 1991.

[23] Skowron, A., Swiniarski, R. W., Synak, P.: Approximation Spaces and Information Granulation, *Transactions on Rough Sets*, **3**, 2005, 175–189.

[24] Wu, W., Zhang, W.: Constructive and Axiomatic Approaches of Fuzzy Approximation Operators, *Information Sciences*, **159**, 2004, 233–254.