Rough Fuzzy C-means and Particle Swarm Optimization Hybridized Method for Information Clustering Problem

F. Cai and F. J. Verbeek
Section Imaging and Bioinformatics, LIACS, Leiden University, Leiden, The Netherlands
Email: {f.cai, f.j.verbeek}@liacs.leidenuniv.nl

Abstract—This paper presents a hybrid unsupervised clustering algorithm, referred to as the Rough Fuzzy C-Means (RFCM) algorithm and Particle Swarm Optimization (PSO). The PSO algorithm features high quality of searching in the near-optimum. At the same time, in RFCM, the concept of lower and upper approximation can deal with uncertainty, vagueness and indiscernibility in cluster relations while the membership function in a fuzzy set can handle overlapping partitions. To illustrate the competence of this method, a number of state-of-the-art hybrid methods (FPSO, Fuzzy-FPSO, RCM-PSO, K-means PSO) are compared through application on datasets obtained from the UC Irvine Machine Learning Repository. The reported results and extensive numerical analysis indicate an excellent performance on the proposed method.

Index Terms—Rough fuzzy c-means, hybrid approach, particle swarm optimization, clustering problems

I. Introduction

Among pattern finding methods, i.e., summarization, association and prediction etc. [1], information clustering is of the great importance and popularity both in research and implementation. Clustering analysis is a technique aiming at grouping a set of objects, based on the similarities and dissimilarities between the data objects. Clustering can be processed in a supervised, semi-supervised and unsupervised manner and consequently it has received considerable amount of attentions from researchers.

However, the exact number of natural groups in the data is sensitivity to outliers and local maxima or minima, algorithmic complexity, and degeneracy [2], etc., are the sorts of issues that cause bottlenecks in the performance of a particular clustering technique [3]. To tackle these problems, nowadays, an amount of approaches and diverse cross-discipline theories are being proposed. Specifically, optimization algorithms are increasingly hybridized with information clustering algorithms.

Particle Swarm Optimization (PSO) was first introduced [4]. Particle optimization evolved from Swarm Intelligence (SI). PSO is one the optimization techniques which has been successfully applied as an approach to a range of clustering quests. It is a population-based metaheuristic algorithm that is inspired by the movement of individuals in a bird flock. PSO consists of a collection of particles, as well as rules to update the status of those particles. The process of updating is based on the history information of the individual and the behavior of its neighbor. Based on these intrinsic properties of PSO, recently hybridized clustering using PSO approaches have been widely and successfully applied in a range of different disciplines [5], i.e., clustering analysis [6], image clustering [7], network clustering [8], and bioinformatics clustering [9].

Research shows that natural behavior of group animals can be successfully used as an inspiration to solve clustering problems in natural systems [10]. Due to its robust ability to perform a global search, approaches such as K-means, K-Harmonic mean, Fuzzy c-means, etc., can be significantly improved with the help of PSO. Ahmadyfar proposed [11] a new method combining PSO with the K-means clustering algorithm, i.e. PSO-KM. An initial process is set up by randomly choosing *k* centroids, and PSO operates by searching all dimensions for a global optimization. In [12], a hybrid PSO and K-means algorithm method on document clustering is presented. The initial centroids are constructed via PSO and subsequently, the K-means algorithm continues until the termination conditions are no longer satisfied. Alternatively, a faster convergent result can be produced [13] with a low computational cost, which is based on a K-Harmonic means with a PSO-based data clustering algorithm (KHM-PSO). The hybridization approaches in fuzzy clustering problems also produce acceptable results. It is stated that [14] the clustering quality is highly correlated with the initialization of centroids in a typical Fuzzy C-Means (FCM) approach. Such approach is referred to as FPSO and it results in a better performance if centroids are initialized by PSO; traditional FCM can deal well with the fuzzy clustering problem. Additionally, a more fuzzyness, hybridized FPSO method named FCM-FPSO [15], is proposed to further reduce the minima in the objective function. Based on FPSO, this algorithm initiates an extra FCM approach to re-search the centroid space in order to reduce the possibilities of being trapped into local minima. In this manner it provides a better convergence. In addition to such an approach, fuzzy c-means algorithm based on Picard iteration hybridized with PSO (PPSO-FCM) is proposed [16] in order to overcome the drawbacks of the typical FCM.

Manuscript received October 20, 2016; revised December 22, 2016.
This work was partially supported by the Chinese Scholar Council (CSC).
Corresponding author email: f.j.verbeek@liacs.leidenuniv.nl.
doi:10.12720/jcm.11.12.1106-1113

The rough c-means approach has shown successful utilization in feature selection, in addition, in clustering analysis it can also provide good results. Rough set theory is pioneered and introduced by Pawlak [17]. Moreover, a method is proposed to combine Rough c-means with PSO [6], i.e. Rough-PSO. In this method, each cluster is modelled as rough set and PSO is employed to tune the threshold and the relative importance of upper and lower approximation of the rough set.

In this paper, we propose an efficient approach hybridized with evolutional PSO and RFCM clustering method. We intend to contribute to the further development of hybrid methodology, in which a sensible integration of rough and fuzzy c means approach with particle swarm optimization algorithm is realized. In clustering problems, the principle of the membership in a fuzzy set enables efficient handling of overlapping partitions, the lower and upper sets of rough theory deal with uncertainty, vagueness and incompleteness in the class definition. At the same time, PSO has the characteristic to be reasonably accurate and able to avoid being trapped into local optima.

The remaining of this paper is organized in the following manner: Section 2 gives a primary overview of RFCM and PSO, respectively. The proposed rough fuzzy c-means hybridized with PSO method is illustrated in Section 3. Section 4 elaborates experimental results, and consequently we conclude the paper in Section 5.

II. Primary Theory Bound

A. Rough Fuzzy C-Means Algorithm

The idea of dealing with uncertainty information in a dataset has led to a combination of employing both fuzzy set and rough set theory. These hybridized algorithms referred as rough fuzzy c-means (RFCM), [18]-[20] and [21], have been widely and frequently used in real life data clustering problems. In this manner, RFCM algorithm is elaborated as follows.

First, fuzzy c-means is used as an partition-based algorithm that clusters a set of *n* objects $\{x_1, \cdots, x_j, \cdots, x_n\}$ into *c* fuzzy centroids with $\{v_1, \cdots, v_i, \cdots, v_c\}$. The membership index assigned "fuzziness" characteristic of a set depicted as level of belonging, can be expressed as u_{ij}.

$$u_{ij} = \left(\sum_{k=1}^{c} \left(\frac{d_{ij}}{d_{kj}} \right)^{\frac{2}{mf-1}} \right)^{-1} \tag{1}$$

where $mf \in (1, \infty]$ is a scalar referred to as the fuzzifier for FCM algorithm and d_{ij} is the distance from object x_j to the cluster centroid v_i.

Taking the advantage of FCM, the boundary domain of a cluster is roughened through incorporation with the approximation sets. The sets are characterized by the lower and upper approximations $\underline{R}(X)$ and $\overline{R}(X)$, respectively, with the following properties: (i) an object x can be part of at most one lower approximation; (ii) if x is not a part of any lower approximation, then it belongs to two or more upper approximations; and (iii) if $x \subseteq \underline{R}(X)$ of class *X*, then simultaneously $x \subseteq \overline{R}(X)$. Based on the defined approximations, the R-positive and R-boundary are defined:

$$\begin{cases} \textbf{Positive}_R(\textbf{in short } P_R) = \underline{R}(X) \\ \textbf{Boundary}_R(\textbf{in short } B_R) = \overline{R}(X) - \underline{R}(X) \end{cases} \tag{2}$$

Consequently, the objective function of RFCM needs to be minimized and subsequently broken into three conditional equations [21]:

$$J = \begin{cases} \omega \times \left(\sum_{i=1}^{c} \sum_{x_j \in P_\delta(v_i)} d_{ij}^2 \right) + \tilde{\omega} \times \left(\sum_{i=1}^{c} \sum_{x_j \in B_\delta(v_i)} u_{ij}^{mf} d_{ij}^2 \right), & if\ P_\delta(v_i) \neq \emptyset,\ B_\delta(v_i) \neq \emptyset \\ \sum_{i=1}^{c} \sum_{x_j \in P_\delta(v_i)} d_{ij}^2 & ,if\ P_\delta(v_i) \neq \emptyset,\ B_\delta(v_i) = \emptyset \\ \sum_{i=1}^{c} \sum_{x_j \in B_\delta(v_i)} u_{ij}^{mf} d_{ij}^2 & ,if\ P_\delta(v_i) = \emptyset,\ B_\delta(v_i) \neq \emptyset \end{cases} \tag{3}$$

where the parameters ω and $\tilde{\omega} = (1 - \omega)$ are the weighting factors that are tuned to balance the relative importance between the crisp region and fuzzy boundary. Since objects lying in a lower set denote definite belongings, and will be assigned with a higher weight ω compared with $\tilde{\omega}$ of objects lying in a boundary set. In RFCM algorithm, each cluster is characterized by its own boundary set and lower approximation, which influences the fuzziness of the final partition. Therefore, the values of the weighting factors are given by [0, 1]. In one cluster, all data are grouped into either lower approximation set or boundary set via a selected attribute δ, which is practically defined as:

$$\delta = \frac{1}{n} \sum_{j=1}^{n} (u_{vh|j} - u_{sh|j}) \tag{4}$$

here *n* is the total number of objects, $u_{vh|j}$ and $u_{sh|j}$ are the highest and second highest membership indexes of object x_j. The meaning of δ is to determine in a degree if one object is "close" enough to the center it belongs to. Therefore, a good clustering procedure should have a value of δ as high as possible. According to the definitions of lower approximation and boundary set, and based on the predefined attribute δ, one object x_j can be characterized as:

$$x_j \begin{cases} \begin{cases} \in P_\delta(v_{vh}) \\ \notin P_\delta(v_{sh}), \\ \notin B_\delta(v_{vh}) \end{cases} & \delta < u_{vh|j} - u_{sh|j} \\ \begin{cases} \in B_\delta(v_{vh}) \\ \in B_\delta(v_{sh}), \end{cases} & \delta \geq u_{vh|j} - u_{sh|j} \end{cases} \tag{5}$$

When $\delta < u_{vh|j} - u_{sh|j}$, and $x_j \in P_\delta(v_{vh})$, meaning that the impacts of the objects in lower approximation of one cluster should be independent of in-between clusters and centroids, and should have similar influence on within cluster and centroid. Otherwise $x_j \in B_\delta(v_{vh})$, the objects belonging to the boundary set in one cluster can also have a different influence on the other clusters and centroids. Therefore, in the RFCM algorithm, the membership index of an object belonging to the lower approximation has to be reset as $u_{ij} = 1$; while the object belonging to its corresponding boundary set will remain u_{ij} (according to Eq. (1)) as in FCM. The new i^{th} centroid is modified using Eq. (6), which also considers the effect of the lower and upper bounds, as well as the fuzzy membership index. In this manner, the extended RFCM algorithm is obtained via:

$$v_i = \begin{cases} \omega \times \left(\frac{1}{|P_\delta(v_i)|}\sum_{x_j \in P_\delta(v_i)} x_j\right) + \tilde{\omega} \times \left(\frac{\sum_{x_j \in B_\delta(v_i)} u_{ij}^{mf} x_j}{\sum_{x_j \in B_\delta(v_i)} u_{ij}^{mf}}\right), & if\ P_\delta(v_i) \neq \emptyset,\ B_\delta(v_i) \neq \emptyset \\ \frac{1}{|P_\delta(v_i)|}\sum_{x_j \in P_\delta(v_i)} x_j & ,if\ P_\delta(v_i) \neq \emptyset,\ B_\delta(v_i) = \emptyset \\ \frac{\sum_{x_j \in B_\delta(v_i)} u_{ij}^{mf} x_j}{\sum_{x_j \in B_\delta(v_i)} u_{ij}^{mf}} & ,if\ P_\delta(v_i) = \emptyset,\ B_\delta(v_i) \neq \emptyset \end{cases} \quad (6)$$

where $|\cdot|$ represents cardinality operator, and the cluster centroid v_i is calculated by the RFCM procedure.

B. Particle Swarm Optimization Prototype

Particle swarm optimization is a population and generation based algorithm modelled after the movements in a "bird flock" and/or a school of fish. Sharing of experience and information of each individual that takes place during stochastic optimization in PSO procedure. Every individual (particle) in the population (swarm) of one generation is assumed to "fly", in order to gain its own best fitness according to its neighboring individuals and prior knowledge of its former history. In this manner, the PSO algorithm maintains a swarm of candidate solutions of the optimization problem, while, each candidate solution is regarded as a particle.

When particles are flying through search space, their positions adjusted that governed by the distance from their own personal best position, as well as the global best position of the swarm. For a swarm of *n* particles with D-dimension vectors, i^{th} particle ($part_i$) contains the following information (notations):

- $pos_i = (pos_{i1}, pos_{i2}, \ldots, pos_{iD})$, the current position of the i^{th} particle;
- $vel_i = (vel_{i1}, vel_{i2}, \ldots, vel_{iD})$, the current velocity (change of position) of the i^{th} particle;
- $p_i = (p_{i1}, p_{i2}, \ldots, p_{iD})$, the best previous position of the i^{th} particle;
- p_g, the best position of a swarm, and $t = (1, 2, \ldots, G)$, the current generation.

Every particle in a swarm is manipulated via the following updating equation:

$$vel_{id}(t+1) = \alpha \cdot vel_{id}(t) + \beta_1 r_1[p_{id}(t) - pos_{id}(t)] + \beta_2 r_2[p_{gd}(t) - pos_{id}(t)] \quad (7)$$

$$x_{id}(t+1) = x_{id}(t) + vel_{id}(t+1) \quad (8)$$

where $i = 1, 2, \ldots, n$ and $d = 1, 2, \ldots, D$. In Eq. (7), α is the positive inertia weight, β_1 and β_2 are the acceleration constants, meaning the correlation between social and individual behavior, and r_1, r_2 are the displacement deviators in the range [0, 1]. $p_{id}(t) - pos_{id}(t)$ is the personal influence, and $p_{gd}(t) - pos_{id}(t)$ is the social influence on the global experience. At present, research on this simple PSO concept is still being performed. Its success is given by the few parameters that are required for the specification of the problem, i.e. dimensionality of the data space and few weighted factors for control of the convergence.

III. Proposed Rough Fuzzy C-Means and Particle Swarm Optimization Hybridized Method (RFCM–PSO)

Taking both RFCM clustering and the intrinsic properties of PSO into account, we propose an efficient model-combined algorithm, namely RFCM-PSO. In RFCM algorithm, each centroid is considered a vector that updates according to an iterative operation. A representation of the centroid vectors therefore, can refer as elements in particles. In other words, the i^{th} particle ($part_i$) can be defined as $part_i = (v_1, v_2, \ldots, v_i \ldots, v_c)$, where v_i, $i = 1,2, \ldots, c$ is the cluster center.

Consequently, a swarm in PSO represents an amount of candidate solutions of centroids in RFCM algorithm. Thus a fuzzy membership function and roughness definitions are assigned on every single object for its clustering decision making. For each iteration in the RFCM-PSO procedure, the centroids in clusters change and their positions are updated based on the particles. Several extra notations (cf. Section 2.B) for RFCM-PSO need to be considered before employing this algorithm:

- *n*, number of objects;
- *c*, number of pre-defined centroids;
- v_i, vectors of centroids containing $pos_i(t)$;
- $pos_i(t)$, the current position of the i^{th} particle at generation *t*; and
- $u_{ij,k}(t)$, the RFCM membership index of the i^{th} object with respect to the j^{th} cluster of the k^{th} particle at generation *t* it belongs to.

Due to the fast convergence and tenable setup of membership index, we suggest an improvement of the performance of PSO searching algorithm, is to initialize the swarm with FCM. The fit, or in other words the objective function, is then measured and minimized by Eq. (3).

The approximation optimization of RFCM is based on Picard iteration through Eq. (1) and (6). The process calls the training of the RFCM parameters which starts by randomly choosing centroids and initiating membership in FCM. Subsequently, it progresses in approximation evaluation for modifying u_{ij} parameter. With a pre-set number of particles, the resulting centroids from RFCM are represented by particles that are given as inputs to optimization procedure of PSO. The best solution, i.e., global optimum, is looked for by a stochastic search from solution space of candidates.

In the proposed method, PSO performs as a standard optimizer in FES/per iteration, where FES represents the maximum amount of function evaluations allowed. Thus, time complexity cost of RFCM-PSO tends to be determined by the cost function in RFCM, which is $O(n^2)$. Furthermore, the implementation of the RFCM – PSO method is described in the pseudo-code as:

Schema 1 *Rough fuzzy c-means and PSO hybrid algorithm*:

Input: fuzzifer *mf*, weighting factor ω, cluster number *c*, α, β_1, β_2
Given: integral generation $t \in (1, \infty]$.
Initializing: stochastic centroid v_i, membership matrix $u_{ij,k}$, *vel* velocity, pos position of particles at generation t=1.
for each *t* generation **do**
  training RFCM parameter:
    Compute the norm distance d_{ij} for each *n* objects and *c* clusters.
    if δ check **then**
      Reset $u_{ij,k(t)}$.
    end if
    Update new centroid as v_i (*t+1*) per Eq. (6).
    Update $u_{ij,k}$ to (*t+1*) via Eq. (1).
  Optimization procedure:
    Training the personal best and global best position, p_i and p_g.
    Update pos_i (*t+1*) and vel_i (*t+1*) for each particle using Eq. (7) and (8).
  Convergence check; break
end for

IV. Experimental Results

TABLE I: RFCM-PSO PARAMETER SETTINGS

Parameter settings				
Clustering	*mf* =2		ω=0.95	
Optimizing	$\alpha \in [0.1,0.9]$	$\beta_1 = \beta_2 = \sqrt{2}$	Population = 10	Generation = 50

The main objective of this section is to assess relative performance of clustering technique hybridized with particle swarm optimization algorithm. The algorithms that are compared with proposed method are: k-means PSO (K-PSO) [11], fuzzy c-means PSO (FPSO) [14], fuzzy c-means and fuzzy PSO (FCM-FPSO) [15] and rough c-means PSO (RPSO) [6]. All the methods are coded and implemented in the *Matlab* 2014a environment running on an Intel (R) Core (TM) i7-3770 (CPU 3.4GHz, 16GB RAM) machine. In practice, our input parameters produce with higher performance compared to other settings. We kept the input parameters constant across all runs (cf. Table I). To analyze the clustering performance of our method, two indices are introduced in the next subsection.

A. Quantitative Measurement

The problem of validation in a clustering algorithm is an important consideration since all of its applications have their own sets of partially successful validation scheme. None of any separate index can comprehensively depict the performance of these clustering algorithms [22] of unlabelled data. After conducting a study in several indexes that are used for performance validation, we propose:

Dunn's Index: Given by [23] is:

$$Dunn = min_j \left\{ min_{k,\ k \neq j} \left\{ \frac{d(v_j,\ v_k)}{max_i\{\Delta(v_i)\}} \right\} \right\} \quad (9)$$

Davies-Bouldin Index: Introduced in [24] is:

$$DB = \frac{1}{c} \sum_{j=1}^{c} max_{j,j \neq k} \{ \frac{S(v_j) + S(v_k)}{d(v_j,\ v_k)} \} \quad (10)$$

Validation standard build: the higher the similarities in within-cluster and dissimilarities in between-cluster, the lower the DB value will have; the well-separated the clusters are, the larger the Dunn index will obtain.

B. Validatioin of Clustering Algorithm

The PSO-combined algorithms have been applied on several bench mark datasets obtained from UCI repository, which cover a range of different type of problems in information science.

Five algorithms are implemented and applied on these datasets (i.e. Table II), and the quality of each algorithm is investigated. The particular test dataset is Iris, with different pre-set cluster numbers, namely cluster = 2 and cluster = 3. The Iris dataset represents a four-dimensional structure that contains 50 samples in each of the three flower categories. One of the three clusters, *Iris setosa*, is well separated with the other two, while there are some overlaps within the *Iris sirginica* cluster and the *Iris sersicolor* cluster. We have setup a separate test of the different partition strategies.

TABLE II: ATTRIBUTE OF SELECTED DATASETS

Dataset	Feature	Instance	category
Iris	4	150	3
Glass	9	214	7
CMC	9	1473	3
Wine	13	178	3
WBCD	30	569	2

Performances of different algorithms are depicted in Fig. 1. This shows that RFCM-PSO has better results by having the lowest DB index and the highest Dunn index

in case of Cluster = 2 and Cluster = 3. An evident difference of the Dunn value occurred in case Cluster = 3. This which is a result of the fact that our method outperforms the others while dealing with overlapped clustering problem. The likely range of variation is coherent and acceptable compared to the four clustering methods in our evaluation. Additionally, the interquartile ranges (IQR) of FPSO and FCM-FPSO are smaller in the relative sense compared with RFCM-PSO. This is because in fuzzy c-means, the membership of an individual is inversely related to the relative distance from every centroid, thus tenable results of FCM-/FPSO can be obtained in a dataset of low dimensions. Nonetheless, it is very sensitive to noise and outliers and it will easily fall into local optima when confronted with dataset of higher dimension.

In the Iris dataset, there are two overlapping clusters of the three clusters in total, this may sometimes result in a clustering of just two clusters. An efficient classifier (the clustering algorithm in unsupervised learning), however, should be able to identify the boundless and vague features classes. As an example, in Fig. 2, it is shown as scatterplots depicting the different views of feature. It is observed that the three different *Iris* species, through inspection of the flowers can be well categorized using sepal width, sepal length, petal width and petal length.

Fig. 3 shows an example of the performance on RFCM – PSO and FPSO in the Iris dataset by minimizing the distance error of all the contained objects, considering cluster = 3. For all algorithms 100 independent runs per generation have been performed. Recorded in every generation steps, the distance error shows the convergence of particles in a single swarm. The Distance Error (DE) is calculated by the mean distance deviation of every single object to the centroid it belongs to after clustering.

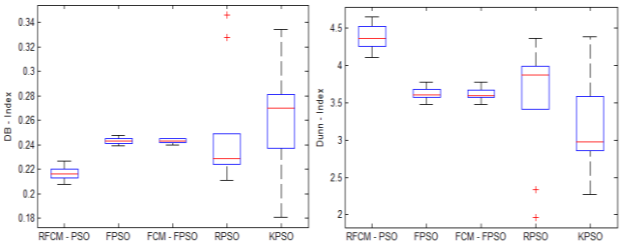


Fig. 1. Box – plot of investigated algorithm on Iris dataset (left: DB Index and, right: Dunn Index, Cluster = 3.

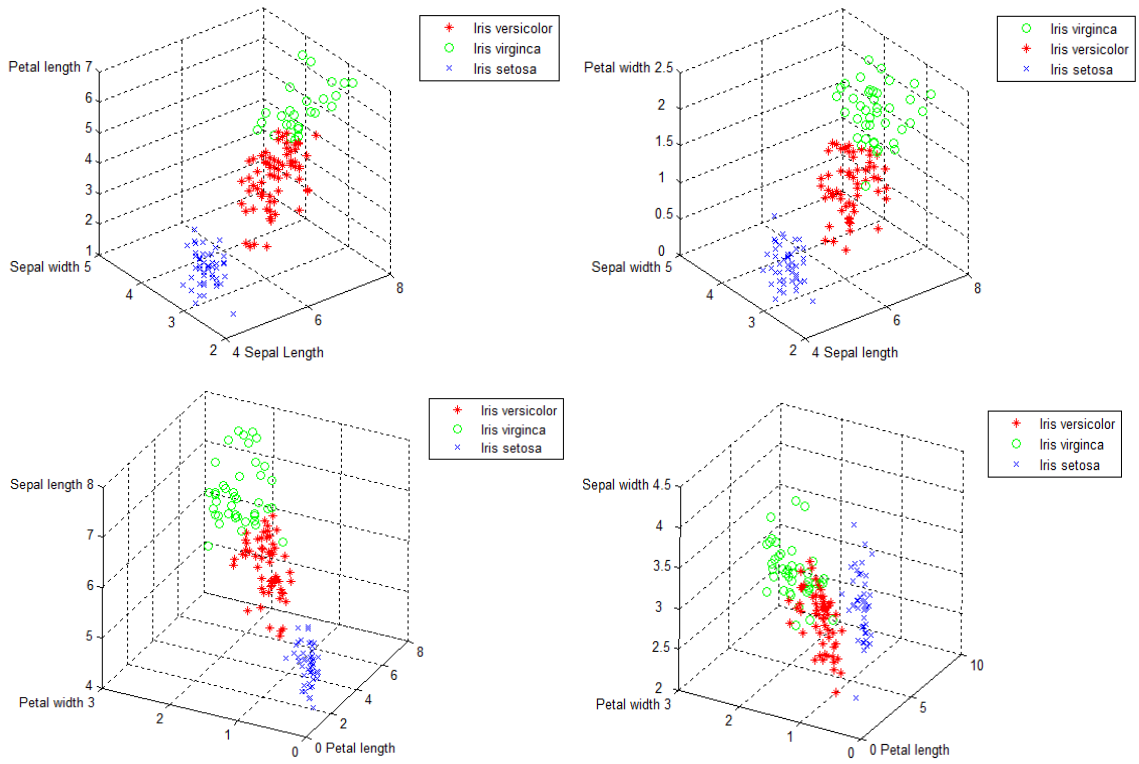


Fig. 2. Scatter plot result of RFCM – PSO on Iris (Cluster = 3, feature = 4).

Depicted in Fig. 3, the proposed RFCM – PSO outperforms the prevalent FPSO in terms of the smaller mean DE error in every generation, lower IQR, faster convergence speed and less outliers.

A well-performed clustering algorithm does not only support on its property of anti-noise or the resultants of less outlier in clusters, but also on its capability of sampling scale-invariance. Applications of most clustering algorithms provide plausible results only on low dimensionality and small population dataset. The handling with sparse and skewed distributions of the samples in a certain clustering space remains a challenge. When sampling scale in a research population, is relatively small then, the higher the dimension of the attributes, the less accuracy and efficiency of the clustering algorithm will perform. Given the definition of

DB Index and Dunn Index, the value of both DB and Dunn should be invariant in spite of a change in the of sampling scale since they have the same overall population. In other words, when different selection of a subset of individuals from within one same research population takes place, as the estimation of the performance of the clustering algorithm, DB and Dunn Index should produce stable results.

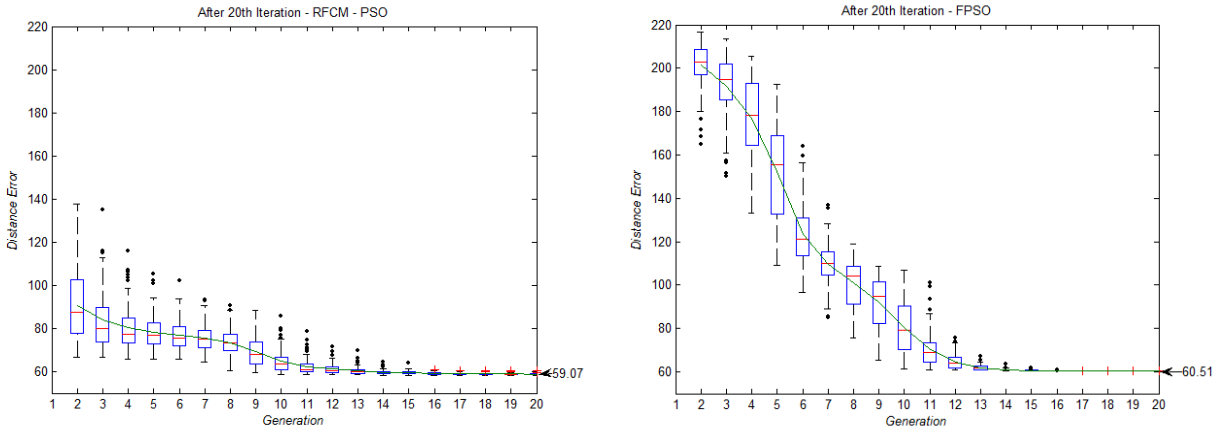


Fig. 3. The clustering performance. When convergent condition met, the DE value for RFCM – PSO (left) and FPSO (right) is 59.07 and 60.51 respectively.

The CMC dataset (cf. Table II) is employed to test the capacity of scale-invariance of each algorithm. We utilize five different scales in the sampling population, i.e. of 300, 600, 900, 1200 and the full population of 1473 instances. To assess a valid estimation of median and to derive acceptable standard errors from a complex and high-dimensional population, the bootstrapped sampling approach is being used. For each different scale, 100 bootstrap runs have been independently applied. The results for each run are summarized and calculated in terms of their max- and minimum, average value and standard deviation. From Table III, one can be seen that the smallest standard deviation of DB and Dunn values are observed on the proposed RFCM – PSO. This result draws a conclusion that the proposed method has acceptable and steady clustering results when sampling scale are differentiated, although skewed and sparse distribution of sample instances are encountered.

In Table IV, the performance of the different PSO hybridized clustering algorithms on the selected benchmark datasets are compared in terms of DB and Dunn index. For all five benchmark sets, every separate algorithm is applied and the value of DB and Dunn are computed respectively. Since the KPSO algorithm produces non-convergent results in the DB and Dunn value of the Glass and Wine datasets, thus these have not been included in Table IV. The results reported here, however convincingly confirm that the proposed method conducts more promising compared to the recognized methods.

TABLE III: SCALE INVARIANT EVALUATION RESULTS ON IRIS DATASET

Algorithm	DB_{max}	DB_{ave}	DB_{std}	$Dunn_{min}$	$Dunn_{ave}$	$Dunn_{std}$
KPSO	0.1457	0.1027	0.2835	13.4826	14.3864	0.9336
FPSO	0.0594	0.0958	0.0820	20.4155	20.6475	0.1453
FCM - FPSO	0.0595	0.0592	0.0817	20.3119	20.5035	0.1409
RPSO	0.0590	0.0586	0.0789	19.4677	19.9972	0.3349
RFCM - PSO	**0.0547**	**0.0542**	**0.0757**	**21.3450**	**21.4935**	**0.0867**

TABLE IV: SCALE INVARIANT EVALUATION RESULTS ON IRIS DATASET

Dataset	KPSO		FPSO		FCM - FPSO		RPSO		RFCM - PSO	
	DB	Dunn	DB	Dunn	DB	Dunn	DB	Dunn	DB	Dunn
Iris	0.241	3.130	0.248	3.567	0.245	3.568	0.249	3.556	**0.216**	**4.382**
Glass	-	-	0.648	0.125	0.644	0.128	0.458	0.197	**0.441**	**0.238**
CMC	0.0732	15.405	0.0594	20.615	0.0592	20.616	0.058	20.007	**0.0544**	**21.494**
Wine	-	-	0.00129	472.4	0.00129	474.4	0.00140	411.8	**0.00124**	**483.9**
WBCD	0.0127	18.8	0.00476	297.6	0.00476	297.9	0.00479	267.3	**0.00475**	**301.2**

V. CONCLUSION

In this paper, we have briefly discussed the evolution of clustering techniques based on Particle Swarm Optimization. A literature survey revealed that there is an enormous increase in the popularity of PSO based clustering techniques. A review of the rough and fuzzy clustering technique is introduced. We use this to present

a novel and efficient hybrid method, namely the Rough Fuzzy C-means and PSO (RFCM - PSO) clustering approach. The performance of our method is compared with the K- means PSO (KPSO), Fuzzy PSO (FPSO), Fuzzy C- means FPSO (FCM - FPSO) and Rough PSO (RPSO) algorithm. The reported results show that our state-of-the-art approach outperforms the rest of the methods in terms of its efficiency, reliability and solution quality which based on geometrical DB and Dunn Index.

The contribution of this paper is in developing a hybridized methodology, which carefully integrates rough and fuzzy c-means approach and particle swarm optimization algorithm. In a clustering problem, the membership of fuzzy set enables efficient handling of overlapping partitions, the lower and upper sets of rough theory deal with uncertainty, vagueness and incompleteness in class definition; while PSO has a tenable quality to be more accurate in searching a best solution from candidate sets as well as avoiding being trapped into local optima.

ACKNOWLEDGMENT

This work is partially supported by the Chinese Scholar Council (CSC).

REFERENCES

[1] S. Mitra, S. K. Pal, and P. Mitra, "Data mining in soft computing framework: A survey," *IEEE Transactions on Neural Networks*, vol. 13, pp. 3–14, 2002.

[2] R. J. Hathaway and J. C. Bezdek, "Optimization of clustering criteria by reformulation," *IEEE Transactions on Fuzzy Systems*, vol. 3, 1995.

[3] S. Alam, G. Dobbie, Y. S. Koh, P. Riddle, and S. U. Rehman, "Research on particle swarm optimization based clustering: A systematic review of literature and techniques," *Swarm and Evolutionary Computation*, 17, 1–13, 2014.

[4] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. Sixth International Symposium on Micro Machine and Human Science*, 1995.

[5] A. Kaur and M. D. Singh, "An overview of pso-based approaches in image segmentation," *International Journal on Engineering Technology*, vol. 2, no. 8, pp. 1349-1357, 2012.

[6] S. Das, A. Abraham, and S. K. Sarkar, "A hybrid rough set--particle swarm algorithm for image pixel classification," in *Proc. Sixth International Conference on Hybrid Intelligent Systems*, 2006.

[7] M. G. H. Omran, "Particle swarm optimization methods for pattern recognition and image processing by doctor," PhD thesis, 2004.

[8] M. Achankunju, R. Pushpalakshmi, and A. V. A. Kumar, "Particle swarm optimization based secure QoS clustering for mobile ad hoc network," in *Proc. International Conference on Communications and Signal Processing*, 2013 pp. 315–320.

[9] X. Xiao, E. R. Dow, R. Eberhart, Z. B. Miled, and R. J. Oppelt, "Gene clustering using self-organizing maps and particle swarm optimization," in *Proc. International Parallel and Distributed Processing Symposium*, 2003.

[10] S. Rana, S. Jasola, and R. Kumar, "A review on particle swarm optimization algorithms and their applications to data clustering," *Artificial Intelligence Review*, vol. 35, pp. 211–222, 2011.

[11] A. Ahmadyfard and H. Modares, "Combining PSO and k-means to enhance data clustering," in *Proc. International Symposium on Telecommunications*, 2008, pp. 688–691.

[12] X. Cui, T. E. Potok, and P. Palathingal, "Document clustering using particle swarm optimization," in *Proc. Swarm Intelligence Symposium*, 2005, pp. 185–191.

[13] F. Yang, T. Sun, and C. Zhang, "An efficient hybrid data clustering method based on K-harmonic means and Particle Swarm Optimization," *Expert Systems with Applications*, vol. 36, pp. 9847–9852, 2009.

[14] L. Wang, Y. Liu, X. Zhao, and Y. Xu, "Particle swarm optimization for fuzzy c-means clustering," in *Proc. Sixth World Congress on Intelligent Control and Automation*, 2006, pp. 6055–6058.

[15] I. Hesam and A. Abraham, "Fuzzy c-means and fuzzy swarm for fuzzy clustering problem," *Expert Systems with Applications*, vol. 38, no. 3, pp. 1835-1838, 2011.

[16] H. C. Liu, H. J. M. Yi, D. B. Wu, and S. W. Liu, "Fuzzy C-mean clustering algorithms based on picard iteration and particle swarm optimization," in *Proc. International Workshop on Education Technology and Training and 2008 International Workshop on Geoscience and Remote Sensing*, 2008, vol. 2, pp. 838–842.

[17] Z. Pawlak, "Rough sets," *International Journal of Computer and Information Science*, vol. 11, pp. 341–356, 1982.

[18] S. K. Pal, A. Ghosh, and B. U. Shankar, "Segmentation of remotely sensed images with fuzzy thresholding, and quantitative evaluation," *International Journal of Remote Sensing*, 2000.

[19] Z. Ji, Q. Sun, Y. Xia, Q. Chen, D. Xia, and D. Feng, "Generalized rough fuzzy c-means algorithm for brain MR image segmentation," *Computer Methods and Programs in Biomedicine*, vol. 108, no. 2, pp. 644–55, 2012.

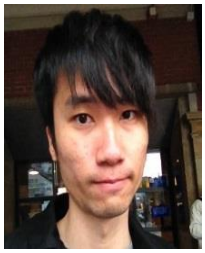
[20] S. Mitra and B. Barman, "Rough-fuzzy clustering: An application to medical imagery," in *Proc. International Conference on Rough Sets and Knowledge Technology*, Springer Berlin Heidelberg, May 17, 2008, pp. 300-307.

[21] P. Maji and S. K. Pal, "RFCM : A hybrid clustering algorithm using rough and fuzzy sets," *Fundamenta Informaticae*, vol. 80, pp. 475–496, 2007.

[22] J. C. Bezdek and N. R. Pal, "Some new indexes of cluster validity," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 28, pp. 301–315, 1998.

[23] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," *Journal of Cybernetics*, pp. 32-57, 1973.

[24] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, pp. 224–227, 1979.



Fuyu Cai was born in Canton, China on November 16th, 1988. He graduated with a master's degree from the Department of Biomedical Engineering at Northeastern University in China. He is a Ph.D candidate currently studying in the Bioinformatics and imaginary group of Institute of Advanced Computer Science, Leiden University, The Netherlands. He focuses on the High Through put or/and High Content Microscopy image processing, including the analysis before, or after Fluorescence and Bright field microscope acquired image, and effect of generic expressing via pattern in image.

Fons J. Verbeek received his Ph.D degree in Applied Physics (pattern recognition group) in 1995 from Delft University of Technology, the Netherlands. Currently, he chairs the Imaging and Bioinformatics group at the Leiden Institute of Advanced Computer Science (LIACS), Leiden University, the Netherlands. In his group the research focusses on development of robust methods for "large-scale" image processing and analysis in the biosciences. The image processing domain is in multi-dimensional microscope images. In addition, methods are being developed to connect image analysis results to biosciences repositories so as to further augment information with available knowledge. Pattern recognition procedures are complemented with Information Visualization to communicate outcomes in the best possible way. In his research the zebrafish model system is used in a range of different application areas.