

Tailoring Fuzzy C-Means Clustering Algorithm for Big Data Using Random Sampling and Particle Swarm Optimization

Yang Xianfeng¹ and Liu Pengfei²

¹ Yang Xianfeng, School of Information Engineering, Henan Institute of Science and Technology, Henan Xinxiang, China

² HEBI Colleges of Vocation and Technology, Henan Hebi, China
¹ 49377535@qq.com, ² 274772453@qq.com,

Abstract

As one of the most common data mining techniques, clustering has been widely applied in many fields, among which fuzzy clustering can reflect the real world in a more objective perspective. As one of the most popular fuzzy clustering algorithms, Fuzzy C-Means (FCM) clustering combines the fuzzy theory and K-Means clustering algorithm. However, there are some issues with FCM clustering. For example, FCM is very sensitive to the initialization condition, such as the determination of initial clusters; the speed of convergence is limited, and the global optimal solution is hard to be guaranteed. Especially for big data scenario, the overall speed is slow, and it is hard to perform the clustering algorithm on all the original dataset. To solve above challenges, in this paper, we propose a modified FCM based on Particle Swarm Optimization (PSO). Besides, we also present a multi-round random sampling method to deal with the big data problem, by simulating the clustering on the original big dataset with the objective to approximate the clustering results on sample datasets. Our experiments show that both the modified FCM using PSO and the multi-round sampling strategy are efficient and effective.

Keywords: Clustering, Fuzzy C-Means (FCM), Particle Swarm Optimization (PSO), Big data

1. Introduction

Clustering [1] is one of the most common data mining techniques, which refers to group a set of objects in such a way that objects within the same group are similar and objects across different groups are different. Clustering is an unsupervised learning process, and has been widely applied in many fields, such as pattern recognition [2], data analysis [3], image processing [4] and marketing research [5], etc.

Fuzzy clustering (also referred to as soft clustering) can allocate objects to clusters in a fuzzy way, where data objects can belong to more than one clusters and associated with different membership levels [6]. Therefore, fuzzy clustering can reflect the real world in a more objective perspective. As one of the most popular fuzzy clustering algorithms, Fuzzy C-Means (FCM) clustering [7] combines the fuzzy theory and K-Means clustering algorithm.

However, there are some problems with FCM clustering. First, FCM is very sensitive to the initialization condition, such as the determination of initial clusters. Second, the speed of convergence is limited, and the global optimal solution is hard to be guaranteed, especially for big data scenario. Third, for big data, the overall speed is slow, and it is hard to perform the clustering algorithm on all the original dataset.

To solve above challenges, in this paper we propose a modified FCM algorithm especially for big data solution. First, to deal with the initialization sensitivity, we propose to improve FCM using Particle Swarm Optimization (PSO) [8] by optimizing the

initialization procedure of FCM. Second, to solve the challenges of big data, we propose a multi-round random sampling method to simulate the clustering on the original big dataset with the objective to approximate the clustering results on sample datasets.

The remain of this paper is organized as follows. Section 2 provides some related work. In Section 3, we present some preliminaries of FCM and PSO. The proposed modified FCM based on PSO and the procedures of FCM clustering on big data are discussed in section 4. Then, empirical experiments are conducted in Section 5. Finally, the paper is concluded in Section 6.

2. Related Work

Existing clustering methods can be grouped as partitioning methods, hierarchical methods, density-based methods, grid-based methods, and model-based methods. For example, typical partitioning methods include K-Means [9], CLARANS [10], and FLEM [11]. Common hierarchical clustering algorithms are BIRCH [12] and CURE [13]. Density-based methods include DBSCAN [14], OPTICS [15], ST-DBSCAN [16], *etc.* STING [17], CLIQUE [18], WAVE-CLUSTER [19] are the examples for grid-based methods.

Many efforts have done in fuzzy clustering as well. For example, Ruspini [20] first proposed the concept of fuzzy partitioning by introducing fuzzy theory into clustering analysis. Then, many fuzzy clustering methods were proposed, such as the transitive closure based on fuzzy equivalence relations [21-22], method based on similarity relations and fuzzy relations [23], the maximum tree based on fuzzy graph theory [24] and methods based on the dataset convex decomposition [25] and dynamic programming [26], *etc.*

Now one of the most common fuzzy clustering algorithms is Fuzzy C-Means (FCM) clustering [27]. However, FCM has some issues such as sensitivity to the initialization status and premature convergence due to its gradient descent based search strategy. Some researchers introduce Genetic Algorithm (GA) to solve the local optimal problem [28-29]. However, when the size of datasets is very large, premature convergence cannot be avoided. Compared to GA, PSO can not only provide global search, but also present strong local search capability by adjusting parameters, and is easy to implement [30-31]. Therefore, in this paper, we propose to use PSO for improving FCM.

3. Preliminary

3.1. FCM Basics

Let $X = \{x_1, x_2, \dots, x_n\}$ be the set of data points, where n the number of data is objects. The clustering results can be represented by fuzzy matrix $\mathbf{U} = [u_{ij}]_{n \times C}$, where C is the number of clusters, and each element u_{ij} denotes the degree of membership of i -th data point to j -th cluster, $i = 1, 2, \dots, n; j = 1, 2, \dots, C$. And,

$$u_{ij} \text{ s.t. } \begin{cases} \forall i, j, & u_{ij} \in [0,1]; \\ \forall i, & \sum_{j=1}^c u_{ij} = 1; \\ \forall j, & \sum_{i=1}^n u_{ij} > 0. \end{cases} \quad (1)$$

FCM is formulated as the optimization problem with above constraints using the following objective function:

$$\min J_m(\mathbf{U}, \mathbf{C}) = \sum_{i=1}^n \sum_{j=1}^c u_{ij}^m d^2(x_i, c_j) \quad (2)$$

where $\mathbf{C} = (c_1, c_2, \dots, c_C)$, c_j is the centroid of j -th cluster; $d(x_i, c_j) = \|x_i - c_j\|$ is the Euclidean distance between x_i and c_j , d_{ij} for short; m is the fuzzy weighting exponent, used to control the influence of matrix \mathbf{U} , and bigger m means that the clustering results are more fuzzy (it is established that $m \in [1.5, 2.5]$ [32]); and $J_m(\mathbf{U}, \mathbf{C})$ is the weighted sum of squares of the distances between data points and the cluster centroids. In this paper, we set $m = 2$.

Apply Lagrange multipliers to solve Equation (2), we have

$$u_{ij} = 1 / \sum_{k=1}^c \left[\frac{d_{ij}}{d_{ik}} \right]^{\frac{2}{m-1}}, \quad (3)$$

$$c_j = \sum_{i=1}^n u_{ij}^m x_i / \sum_{i=1}^n u_{ij}^m \quad (4)$$

The procedure of FCM can be described as follows:

Step 1: given the number of clusters C , the size of data samples n , the termination threshold ε , fuzzy exponent m , maximum iteration times Num , the randomly generated initial centroid matrix \mathbf{C} , and $t = 0$. Initialize \mathbf{U}^0 using Equation (3);

Step 2: $t = t + 1$. Update centroid matrix \mathbf{C}^t using Equation (4);

Step 3: Recalculate membership matrix \mathbf{U}^t using Equation (3);

Step 4: if $\|U^t - U^{t-1}\| < \varepsilon$, algorithm stops; otherwise, go to Step 2.

3.2. PSO Basics

PSO algorithm simulates the pattern and behavior of bird flock, and each bird is represented as a particle. For each particle, there are two important attributes, i.e., location and speed. Suppose there are N_p particles in a D -dimensional space. Given particle i , its location is $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$, and speed is $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})$. The location and speed of particles are updated during the flying process. Suppose the best location of particle i is $p_i = (p_{i1}, p_{i2}, \dots, p_{iD})$, and the overall best location of all particles is $p_g = (p_{g1}, p_{g2}, \dots, p_{gD})$. The speed and location are updated by:

$$v_{id}(t+1) = wv_{id}(t) + c_1\gamma_1(p_{id}(t) - x_{id}(t)) + c_2\gamma_2(p_{gd}(t) - x_{gd}(t)), \quad (5)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1), \quad (6)$$

where $i = 1, 2, \dots, N_p, d = 1, 2, \dots, D$, c_1, c_2 are learning factors, $\gamma_1, \gamma_2 \in [0, 1]$ are random numbers, and w is an inertia weight (typically $w \in [0.1, 0.9]$). $v_{id}(t)$ is the speed of particle i at iteration t , and $x_{id}(t)$ is the location of particle i at iteration t .

Equation (5) is the update rule of particle's speed, where the first component is the current speed of particle; the second component is the influence of self-recognition, i.e., the distance between the current location and the best location of each particle; and the third component is the influence of the whole population, i.e., the distance between the current location of the particle and the overall best location of all particles. Equation (6) is the update rule of particle's location, which indicates that the location at iteration $t+1$ is calculated as the location at iteration t plus the flying distance during one iteration. Figure 1 gives the pseudo-code of PSO.

4. Proposed Modified FCM

4.1. Improving FCM with PSO

The gradient descent based FCM algorithm is essentially a local search algorithm, and therefore is easily to arrive a local minimum, especially then the size of data samples is large. Besides, FCM is sensitive to the initialization status, and most FCM algorithms simply random choose initial centroids, which in turn affects the convergence speed.

Fortunately, swarm based PSO algorithm has a strong global search capability, and the convergence speed is fast. Therefore, we propose to combine FCM and PSO to optimize the objective function in Equation (2). Basically, the idea is to represent centroids as particles, and utilize PSO for global search and FCM for local search.

Algorithm 1 PSO algorithm

```

1: repeat
2:   Initialize particles, position, and speed;
3:   for each particle  $i$  do
4:     Calculate the fitness  $f(i)$ ;
5:     if  $f(i) > f^*(i)$  then
6:       Update individual best location  $p_i$ ;
7:     end if
8:   end for
9:   if the overall best location is better than  $p_g$  then
10:    Update overall best location  $p_g$ ;
11:   end if
12:   for each particle  $i$  do
13:     for each dimension  $d$  do
14:        $v_{id}(t+1) = W \times v_{id}(t) + c_1 \times rand() \times (p_{id}(t) - x_{id}(t)) + c_2 \times rand() \times$ 
          $(p_{gd}(t) - x_{gd}(t));$ 
15:        $x_{id}(t+1) = x_{id}(t) + v_{id}(t+1);$ 
16:     end for
17:   end for
18:    $t = t + 1;$ 
19: until  $t > Num$ 

```

Figure 1. Algorithm Description of PSO

4.1.1. Particles Encoding: we represent each particle as a feasible solution. That is, the location of each particle consists of C centroids $\mathbf{C} = (c_1, c_2, \dots, c_C)$, where c_j is the centroid of j -th cluster.

4.1.2. Fitness Calculation: recall that our goal is to minimize $J_m(\mathbf{U}, \mathbf{C})$. Therefore, the fitness function can be defined as:

$$f_i = \frac{1}{J_m(\mathbf{U}, \mathbf{C}) + 1}, \quad (7)$$

The larger f_i is, the smaller $J_m(\mathbf{U}, \mathbf{C})$ is, and the better the solution is.

4.1.3. Timing of Combination: Generally, we utilize PSO for global search and FCM for local search. The key is to determine when to apply FCM during the PSO process. In

this paper, we propose to perform FCM when the particle swarm converges. Define the degree of convergence by the variance of fitness:

$$\sigma^2 = -\sum_{i=1}^{N_p} \left(\frac{f_i - f_{avg}}{f} \right)^2, \quad (8)$$

Where f_i is the fitness of i -th particle, f_{avg} is the average fitness of all particles, and N_p is the number of particles. The smaller σ^2 is, the more converge the particles are.

We use σ^2 to determine when to apply FCM during PSO global search. If σ^2 is close to 0, the particles are almost the same, which means PSO arrives premature convergence or global convergence. Otherwise, particles with various fitness are doing random search. If σ^2 is smaller than a threshold, it means that PSO is about to converge, and applying FCM at that time for local search can improve the performance of global search to avoid premature convergence.

4.1.4 Algorithm Procedure: Figure 2 illustrates the flow chart of modified FCM with PSO algorithm. Specifically, the procedure can be described as follows:

Step 1: initialize the particle swarm and parameters. Given the number of clusters C , the size of data samples n , fuzzy exponent m , maximum iteration times Num , and $t = 0$. Randomly generate initial centroid matrix C , and initialize U^0 using Equation (3). Generate a particle from the centroid vector, and initialize its speed using Equation (5). Run the random centroids generation N_p times, and get N_p particles.

Step 2: for each particle i , calculate its fitness using Equation (7), and get its best location P_i .

Step 3: for each particle i , if its best location P_i is better than any other particles, update the global location P_g as P_i .

Step 4: update speed and location of all particles using Equations (5) and (6).

Step 5: if the variance of fitness $\sigma^2 < \mathcal{G}$, where \mathcal{G} is the threshold of the variance of fitness values of all particles, go to Step 6; otherwise, return to Step 2.

Step 6: perform FCM clustering. Calculate the centroids of clusters using Equation (4).

Calculate membership matrix U using Equation (3). Update corresponding particles based on centroids.

Step 7: if termination condition is satisfied, algorithm stops. Otherwise, return to Step 2.

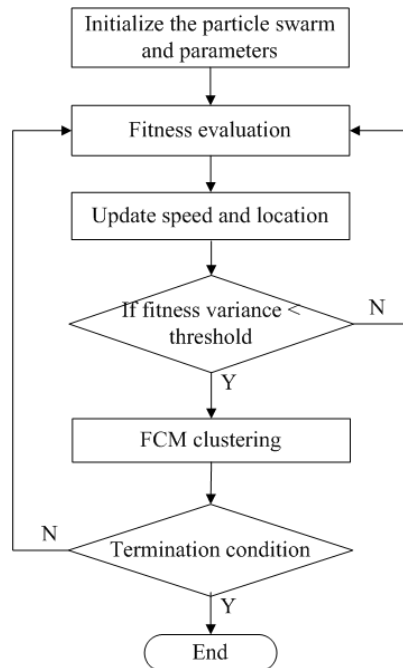


Figure 2. Flow Chart of Modified FCM Clustering based on PSO

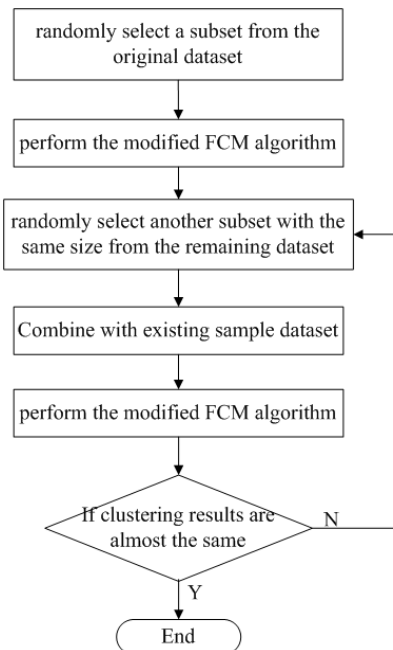


Figure 3. Flow Chart of Multi-round Random Sampling Method

4.2. Procedure for Big Data Clustering

Even though our modified FCM clustering algorithm can solve the issues, such as initialization sensitivity and low convergence speed, there are still problems with big data scenario.

For example, in Step 1 in Section 4.1, where N_p particles are generated by randomly generating centroids N_p times on the original dataset. If the size of dataset n is not very large, the performance is good. However, if n is very big, this step could be extremely time-consuming, and even not feasible to implement.

Therefore, in this section, we propose a multi-round random sampling method to simulate the clustering on the original big dataset with the objective to approximating the clustering results on sample datasets. The basic assumption here is that [33] by performing clustering on a smaller subset, we can approximate the results on the original large dataset.

The multi-round random sampling can be described as follows. First, randomly select a subset from the original dataset D , notated as D_i , and then perform the modified FCM algorithm on D_i as presented in Section 3, and get the clustering results $\mathbf{C}_i = (c_1^i, c_2^i, \dots, c_C^i)$. After that, randomly select another subset with the same size from the remaining dataset, notated as D_j , and combine D_i and D_j , notated as D_{ij} . Using $\mathbf{C}_i = (c_1^i, c_2^i, \dots, c_C^i)$ as the initial centroids, and then perform modified FCM on D_{ij} , and get clustering results $\mathbf{C}_{ij} = (c_1^{ij}, c_2^{ij}, \dots, c_C^{ij})$. Repeatedly randomly select a subset from the remaining dataset and combine to the existing samples, and then perform modified FCM on the new dataset. Until the difference of clustering results between two rounds is small enough. That is, $\|\mathbf{C}_{12\dots t} - \mathbf{C}_{12\dots(t-1)}\| < \varepsilon$, where $\mathbf{C}_{12\dots t}$ is the clustering results of the t -th round, and ε is the threshold. The flow chart is shown in Figure 3, where the modified FCM algorithm is discussed in Section 4.1.

5. Experiment

Our datasets are originally obtained from UCI database, i.e., iris, glass and wine. To simulate the big data scenario, we enlarge original datasets by 1000 times in this experiment. The parameter settings are $C = 3, m = 2, \varepsilon = 10^{-5}, N^p = 30, Num = 100, \Delta\% = 3\%$.

We define three baselines for comparison. (1) **FCM**: basic FCM algorithm on

randomly sampled dataset. (2) **FCM-PSO**: modified FCM based on PSO on randomly sampled dataset. (3) **FCM-PSO-MRS**: modified FCM based on PSO using the multi-round random sampling method on the original large dataset.

Table 1 gives the comparison of average precision for three methods on the UCI datasets and large scale datasets, respectively. We have the following observations. First, the performance of FCM-PSO is better than FCM, which means our modified strategy based on PSO can indeed improve the precision of clustering. Second, FCM-PSO-MRS is better than the other two methods, which means that our multi-round random sampling strategy is better than randomly sampling. Third, for big data scenario, FCM-PSO-MRS can preserve good performance, while FCM and FCM-PSO have some degradation of performance, and thus FCM-PSO-MRS is suitable for big data clustering. Specifically, comparing the smaller UCI datasets and the 1000 times big datasets, the precision for FCM decreases most when the size of dataset is large, while the performance of proposed FCM-PSO-MRS is almost the same despite the size of dataset.

Table 1. The Average Precision of Clustering

Dataset	FCM	FCM-PSO	FCM-PSO-MRS
Iris	80.58%	84.37%	89.12%
Glass	71.34%	78.92%	81.05%
Wine	63.41%	69.56%	73.87%
Iris*1000	67.23%	70.84%	88.92%
Glass*1000	60.67%	74.43%	80.85%
Wine*1000	58.50%	67.32%	73.28%

Besides, we evaluate the convergence performance of our modified FCM-PSO algorithm on big dataset in Figures 4, 5 and 6. Note that in these experiments, to be fair, both FCM and FCM-PSO are based on one round random sampling, in order to compare the convergence speed. For all three datasets, we have the same conclusion, that is, our modified FCM-PSO converges faster than FCM with better fitness value.

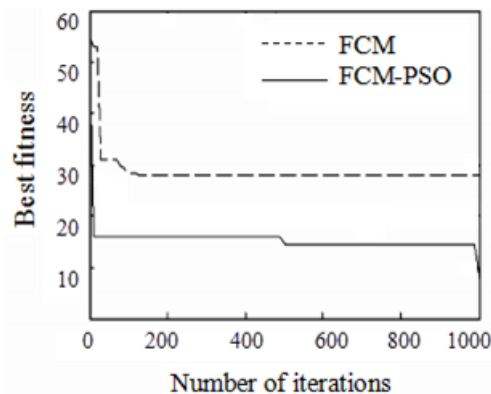


Figure4. Performance of FCM-PSO for iris*1000 Dataset

6. Conclusion

In this paper, we focus on the modification of FCM clustering considering its limitations such as sensitivity to initialization status, low convergence, *etc.* Specifically, we propose to combine PSO into FCM. Moreover, we present a multi-round random sampling method to deal with the big data challenge.

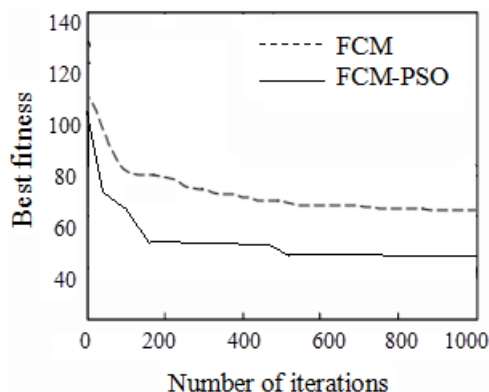


Figure 5. Performance of FCM-PSO for Glass*1000 Dataset

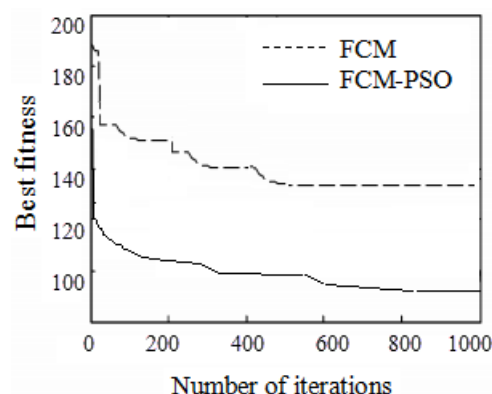


Figure 6. Performance of FCM-PSO for Wine*1000 Dataset

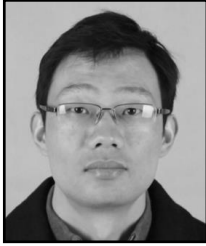
However, the experiments in this study in obtained by expanding UCI datasets, which is more, like simulated datasets. In future works, we would like to embrace the proposed clustering into real world big data scenarios, such as large scale social networks, and try to explore more clustering applications in big data.

References

- [1] A. K. Jain, M. N. Murty and P. J. Flynn. "Data clustering: a review", ACM computing surveys (CSUR), vol. 31, no. 3, (1999), pp. 264-323.
- [2] A. Baraldi and P. Blonda, "A survey of fuzzy clustering algorithms for pattern recognition. I." Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions, vol. 29, no. 6, (1999), pp. 778-785.
- [3] P. Berkhin, "A survey of clustering data mining techniques", Grouping multidimensional data, Springer Berlin Heidelberg, (2006), pp. 25-71.
- [4] B. N. Li, "Integrating spatial fuzzy clustering with level set methods for automated medical image segmentation", Computers in Biology and Medicine, vol. 41, no. 1, (2011), pp. 1-10.

- [5] D. Won, B. M. Song and D. McLeod, "An approach to clustering marketing data", Proceedings of the 2nd International Advanced Database Conference, (2006).
- [6] D. Graves and W. Pedrycz, "Kernel-based fuzzy clustering and fuzzy clustering: A comparative experimental study", Fuzzy sets and systems, vol. 161, no. 4, (2010), pp. 522-543.
- [7] J. C. Bezdek, R. Ehrlich and W. Full, "FCM: The fuzzy c-means clustering algorithm", Computers & Geosciences, vol. 10, no. 2, (1984), pp. 191-203.
- [8] J. Kennedy, "Particle swarm optimization", Encyclopedia of Machine Learning, Springer US, (2010), pp.760-766.
- [9] A. K. Jain, "Data clustering: 50 years beyond K-means", Pattern Recognition Letters, vol. 31, no 8, (2010), pp. 651-666.
- [10] R. T. Ng and J. Han, "CLARANS: A method for clustering objects for spatial data mining", Knowledge and Data Engineering, IEEE Transactions, vol. 14, no. 5, (2002), pp. 1003-1016.
- [11] C. Ordonez and E. Omiecinski, "FREM: fast and robust EM clustering for large data sets", Proceedings of the eleventh international conference on Information and knowledge management, ACM, (2002).
- [12] T. Zhang, R. Ramakrishnan and M. Livny, "BIRCH: an efficient data clustering method for very large databases", ACM SIGMOD Record, vol. 25, no. 2, (1996).
- [13] S. Guha, R. Rastogi and K. Shim, "CURE: an efficient clustering algorithm for large databases", ACM SIGMOD Record, vol. 27, no. 2, ACM, (1998).
- [14] M. Ester, "A density-based algorithm for discovering clusters in large spatial databases with noise", Kdd, vol. 96, (1996).
- [15] M. Ankerst, "Optics: Ordering points to identify the clustering structure", ACM Sigmod Record, vol. 28, no. 2, ACM, (1999).
- [16] D. Birant and A. Kut, "ST-DBSCAN: An algorithm for clustering spatial-temporal data", Data & Knowledge Engineering, vol. 60, no. 1, (2007), pp. 208-221.
- [17] W. Wang, J. Yang and R. Muntz, "STING: A statistical information grid approach to spatial data mining", VLDB, vol. 97, (1997).
- [18] R. Agrawal, "Automatic subspace clustering of high dimensional data for data mining applications", ACM, vol. 27, no. 2, (1998).
- [19] G. Sheikholeslami, S. Chatterjee, A. Zhang, "Wave-Cluster: A Multi-Resolution Clustering Approach for Very Large Spatial Databases", Proc. 24th Int. Conf. on Very Large Data Bases, (1998), New York.
- [20] E. H. Ruspini, "A new approach to clustering", Information and control, vol. 15, no. 1, (1969), pp. 22-32.
- [21] G.-S. Liang, T.-Y. Chou and T.-C. Han, "Cluster analysis based on fuzzy equivalence relation", European Journal of Operational Research, vol. 166, no.1, (2005), pp. 160-171.
- [22] D. Boixader, J. Jacas and J. Recasens, "Fuzzy equivalence relations: advanced material", Fundamentals of Fuzzy Sets. Springer US, (2000), pp. 261-290.
- [23] M.-S. Yang and H.-M. Shih, "Cluster analysis based on fuzzy relations", Fuzzy Sets and Systems, vol. 120, vol. 2, (2001), pp. 197-212.
- [24] Z. Wu and R. Leahy, "An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation", Pattern Analysis and Machine Intelligence, IEEE Transactions, (1993).
- [25] J. C. Bezdek and J. D. Harris, "Fuzzy partitions and relations; an axiomatic basis for clustering", Fuzzy sets and systems, vol. 1, no. 2, (1978), pp. 111-127.
- [26] A. O. Esogbue, "Optimal clustering of fuzzy data via fuzzy dynamic programming", Fuzzy Sets and Systems, vol. 18, no. 3, (1986), pp. 283-298.
- [27] D. Graves and W. Pedrycz, "Kernel-based fuzzy clustering and fuzzy clustering: A comparative experimental study", Fuzzy sets and Systems vol. 161, no. 4, (2010), pp. 522-543.
- [28] K. Krishna and M. N. Murty, "Genetic K-means algorithm" Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions, vol. 29, no. 3, (1999), pp. 433-439.
- [29] U. Maulik and S. Bandyopadhyay, "Genetic algorithm-based clustering technique", Pattern recognition vol. 33, no. 9, (2000), pp. 1455-1465.
- [30] J.-C. Zeng and Z.-H Cui, "A guaranteed global convergence particle swarm optimizer", Journal of Computer Research and Development, vol. 41, no. 8, , (2004), pp. 1333-1338.
- [31] L. Xiao-qing and J. Su-min, "PSO spatical clustering with obstacles constraints", Computer Engineering and Design, vol. 28, no. 24, , b, pp. 5924-5926.
- [32] N. R. Pal and J. C. Bezdek, "On cluster validity for the fuzzy c-means model", Fuzzy Systems, IEEE Transactions, vol. 3, no. 3, (1995), pp. 370-379.
- [33] M.-C. Hung, and D.-L. Yang, "An efficient fuzzy c-means clustering algorithm", Data Mining, (2001), ICDM 2001, Proceedings IEEE International Conference, (2001).

Authors



Yang Xianfeng, he received his bachelor's degree in Computer Application Technology from Henan Normal University, Xinxiang, Henan, China, in 2001, the master's degree in Computer Application Technology from China University of Petroleum, Dongying, China, in 2007. He is now a lecturer at School of Information Engineering, Henan Institute of Science and Technology, Xinxiang, China. His current research interests include pattern recognition, image processing, neural networks, natural language processing.



Liu Pengfei, he received his bachelor's degree in Computer Science and Technology from Anyang Normal University, Anyang, China, in 2004, the master degree in Computer Application from Huazhong University of Science and Technology, Wuhan, China, in 2010. He is now a lecturer at Hebi College of Vocation And Technology. His current research interests include computer network, web application, and network operating system.