Expert Systems with Applications 36 (2009) 11792–11800

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

A fuzzy clustering algorithm based on evolutionary programming

Hongbin Dong [a,b,*], Yuxin Dong [a], Cheng Zhou [a], Guisheng Yin [a], Wei Hou [a]

[a] National Science Park, Harbin Engineering University, Harbin 150001, China
[b] Department of Computer Science, Harbin Normal University, Harbin 150080, China

ARTICLE INFO

Keywords:
Fuzzy *c*-means algorithm
Evolutionary programming
Cluster validity
EPFCM

ABSTRACT

In this paper, a fuzzy clustering method based on evolutionary programming (EPFCM) is proposed. The algorithm benefits from the global search strategy of evolutionary programming, to improve fuzzy *c*-means algorithm (FCM). The cluster validity can be measured by some cluster validity indices. To increase the convergence speed of the algorithm, we exploit the modified algorithm to change the number of cluster centers dynamically. Experiments demonstrate EPFCM can find the proper number of clusters, and the result of clustering does not depend critically on the choice of the initial cluster centers. The probability of trapping into the local optima will be very lower than FCM.

1. Introduction

Cluster analysis is a major analytic tool in image processing, spatial remote sensing, data mining, gene data processing and signal compression, and so on. Partitional clustering is the most commonly used general algorithm in pattern recognition. Classic partition-based clustering methods includes conventional *c*-means algorithm, fuzzy *c*-means (FCM) algorithm and maximum entropy method. All of these methods cannot optimize the feature the given data set, while to optimize cluster directly. At present, the most well-known fuzzy clustering of which is FCM, the algorithm is available for small and low-dimensional data sets it suffers from several inherent drawbacks: (1) to apply the method, the user has to know a prior knowledge; (2) the random initial choice could generate different clustering solutions, even no solution; (3) this objective function-based algorithm searches the optimum by the gradiential method so that it is easy to get trapped at a local minimum. To address these drawbacks, Gas were utilized to optimize the objective functions of *k*-means algorithm (Murthy & Chowdhury, 1996), it avoids the drawback (2) and (3), since the algorithm encode the cluster and is not a iterative process, the efficiency is lower; a modified algorithm (Bandyopadhyay & Maulik, 2002) is more efficient by encoding the cluster center and adapting iterative *k*-means algorithm in the search of GAs, an evolutionary programming algorithm (Sarkar, Yegnanarayana, & Khemani, 1997) has solved these above problems efficiently by optimizing DB index of *k*-means algorithm; the flexibility of a variable string length genetic algorithm (FVGA) (Pakhira & Bandyopadhyay, 2005), which the cluster validity indices are used as the searched objective functions, can find the optimal number of clusters in search process.

This paper proposes an evolutionary programming-based fuzzy *c*-means clustering algorithm (EPFCM) along with a single-point mutation in evolutionary programming (SPMEP) (Ji, Tang, & Guo, 2004), which utilizes the cluster validity indices to evaluate the result of clustering. To increase the convergence speed of the algorithm, the Modify algorithm is applied for varying the number of cluster centers dynamically. Experiments demonstrate EPFCM can find the proper number of clusters, and the clustering result does not depend critically on the choice of the initial cluster centers, the probability of trapping into the local optima will be very lower than FCM.

2. Basic idea

For most of clustering algorithms, the objective function is not convex, and hence, it may contain local minima. Therefore, while minimizing the objective function, there is a possibility of getting stuck in local minima (also in local maxima and saddle points). To solve this problem, evolutionary algorithms for fuzzy clustering have been proposed and achieved more efficient results; since the objective function decreases monotonically on the number of cluster centers, there are two cluster structures with different number of cluster center, how to determine the best one out of several different cluster partitions? We can use some cluster validity index to find the optimum cluster. The validity indices determine the number of clusters and the corresponding best cluster structure, as well as all of cluster structures, whether the numbers of their cluster centers are same or not, we execute evolutionary algorithm to search the optimum of the validity indices as well as the best result of clustering.

* Corresponding author.
E-mail address: donghongbinbjtu@gmail.com (H. Dong).

0957-4174/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2009.04.031

Evolutionary programming (EP) (Dong, He, & Huang, 2007; Lee & Yao, 2004; Yao, Liu, & Lin, 1999) is advantageous over GA for functions optimization on real numbers space. We utilize the EP as optimization algorithm, encoding the cluster centers as a sequence of real numbers and defining the cluster validity indices as a function of the number of cluster centers. This paper proposes an evolutionary programming along with the iterative process of the FCM clustering method, which fastens the convergent speed of the algorithm. Each cluster center is perturbed in the iterative process of the FCM. we exploit the modified algorithm,which is quite similar to the splitting and merging operations found in ISODATA, to update the number of the cluster centers by splitting the highest scattered cluster or merging two lowest scattered clusters. The algorithm creates new parents from all parents and offspring by clustering the validity indices.

3. EPFCM algorithm

We select single-point mutation evolutionary programming (SPMEP) (Ji et al., 2004) as optimal algorithm, FCM as clustering algorithm, and one of the four indices (Dong, Huang, Zhou, & He, 2007), PC (Bezdek, 1975), PE (Bezdek, 1974), PBMF (Maulik & Bandyopadhyay, 2000; Pakhira & Bandyopadhyay, 2005) or XB (Xie & Beni, 1991) as validity indices.

3.1. EPFCM algorithm as following

(1) Population initialization
Each individual is a vector $(k \times m)$ consisted of some cluster centers, representing a clustering structure, the number of cluster centers, k is a random integer between 2 and $k_{\max}$ (in general, $k_{\max} = \sqrt{n}$). There are $k \times m$ random values in the string.
(2) Mutation
Mutation is performed for parent population $\{P\}$ to achieve offspring population $\{P'\}$ by using SPMEP, in which the vector of cluster centers is regarded as individual vector $\vec{x}$.
(3) Modify the number of cluster centers as individual in population
By exploiting the Modify algorithm, we update the number of cluster centers of each individual with probability p (0.3) to generate cluster structure of cluster center with different number. The Modify algorithm is described in the next section.
(4) Using FCM algorithm to iterate
We execute iterately the step (1) and step (2) of FCM for each individual in population $\{P \cup P'\}$ to generate new population $\{Q \cup Q'\}$, each individual in $\{P \cup P'\}$ is considered as cluster center V of FCM.
(5) Evaluating the fitness of all cluster center vectors
Since a cluster center vector corresponds to a clustering structure, we calculate the fitness value of each individual in $\{Q \cup Q'\}$ by using a cluster validity function.
(6) Selecting the best individuals as new parent generation
We execute the step (4) and step (5) of single-point mutation evolutionary programming (SPMEP) to generate new parent generation.
(7) Termination
Stop algorithm and save the last clustering structure if the number iteration exceeds a given limit, otherwise, go back to step (2).

3.2. Basic idea of the modify algorithm

To change the number of cluster centers of a clustering structure, sometimes one cluster is add to or deleted from clusters (Sarkar et al., 1997). The addition of one cluster is done by splitting an existing cluster of that set. To identify a cluster for deleting, it is required to find the cluster with maximum hypervolume, where hypervolume *volume* (C_k) of cluster C_k is defined as

$$Volume(C_k) = \sqrt{\frac{1}{n}\sum_{x \in C_k} ||x - m_k||^2},$$

C_k is the kth cluster, n is the number of C_k, m_k is the center of C_k.

The cluster m_k with maximum hypervolume is split into two new cluster centers m_k^+ and m_k^-, and then, m_k is deleted. As a result, the number of clusters of this set is incremented by one. The cluster center m_k^+ and m_k^- is formed by adding/subtracting a certain quantity to/from the component of m_k that is

$$m_k^+ = m_k^+ \delta/3, \quad m_k^- = m_k - \delta/3$$

where δ is the maximum component of $\{\delta_1, \delta_2, \ldots, \delta_m\}$, δ_i is the maximum distance value of the ith dimensional data set.

Deletion of one cluster from an offspring set is executed by merging two existing cluster of that set. In order to accomplish it, the two closest clusters with centers are identified randomly for merging. The center of the new cluster is the average of all data of the two merged clusters. Next, the two closet clusters are deleted, and the number of clusters is reduced by one.

EPFCM optimization approach is advantageous over EP, the iterative steps of FCM algorithm is add to the EP-based iterative process in EPFCM, which is the key to this algorithm, otherwise, the algorithm may take a large amount of time to converge. We exploit the modified algorithm to vary the number of some cluster centers and maintain the diversity of the number of the cluster centers, consequently to achieve the optimal number of cluster centers. The experiment results will be provided to show the performance of EPFCM and Modify in next section.

4. Results and discussion

In this section, first, we report the data sets profile of experiments of EPFCM and the results of numerical experiments. This includes a description of the data sets and comparison of the optimal clustering number and the value of validity indices with different indices. The results also show the variation of the value of validity indices with the different number of cluster. Secondly, we give the results of the automatic clustering algorithm, which includes the performance of clustering, and the variation of the number of cluster and the value of validity indices with the number of generations. Finally, We further discuss and analyze the performance of the algorithm.

4.1. Experimental datas

We select five data sets (Figs. 1–5), which include one real data set, two artificial data sets and two literature data sets (Pakhira & Bandyopadhyay, 2005) for testing and comparison. To reach the real data, two artificial data sets follow the normal distribution, there are a lot of data points and actual number of clusters. As Table 1 depicts, Optimal number of clusters of some data sets are not unique, that is reasonable.

We have studied the performance of the algorithm for different parameter values. For illustration, Table 2 shows the variation of the number of cluster centers with the optimal cluster and the value of validity indices with the different validity indices. From this table, it is seen that the number optimal cluster is 8 and the validity index value is 0.9286112 as determined by the Pc index for mydata_8_3, the optimal number of cluster determined by the PBMF index for Iris and mydata_8_3 are different with other data sets, the number of cluster centers determined by the XB index for circular_5_2 and circular_6_2 are different with other data sets.