Expert Systems with Applications 39 (2012) 2194–2202

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

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

A genetic clustering algorithm using a message-based similarity measure

Dongxia Chang [a,c,*], Yao Zhao [a], Changwen Zheng [b], Xianda Zhang [c]

[a] Institute of Information Science, Beijing Jiaotong University, Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing 100044, China
[b] National Key Lab of Integrated Information System Technology, Institute of Software, Chinese Academy of Sciences, Beijing 100080, China
[c] Tsinghua Department of Automation, Tsinghua University, Beijing 100084, China

ARTICLE INFO

Keywords:
Clustering
Evolutionary computation
Genetic algorithms
Message passing
K-means algorithm

ABSTRACT

In this paper, a genetic clustering algorithm is described that uses a new similarity measure based message passing between data points and the candidate centers described by the chromosome. In the new algorithm, a variable-length real-value chromosome representation and a set of problem-specific evolutionary operators are used. Therefore, the proposed GA with message-based similarity (GAMS) clustering algorithm is able to automatically evolve and find the optimal number of clusters as well as proper clusters of the data set. Effectiveness of GAMS clustering algorithm is demonstrated for both artificial and real-life data set. Experiment results demonstrated that the GAMS clustering algorithm has high performance, effectiveness and flexibility.

1. Introduction

Clustering analysis is a widely used unsupervised learning technique for data analysis and can be applied in a variety of engineering and scientific disciplines such as biology analysis, psychology, computer vision, communications, and remote sensing. The primary objective of clustering analysis is to partition a given data set of multidimensional vectors (patterns) into several homogeneous clusters such that patterns in the same cluster are similar to each other in some sense and differentiate from those of other clusters in the same sense. Extensive overviews of clustering algorithms can be found in the literature (Everitt, Landau, & Leese, 2001; Jain & Dubes, 1988; Jain, Murty, & Flynn, 1999; Tou & Gonzalez, 1974; Xu & Wunsch, 2005).

As an important tool for data exploration, clustering analysis examines unlabeled data, by either constructing a hierarchical structure, or forming a set of groups according to a prespecified number. Clustering algorithms may be broadly divided into two classes (Everitt et al., 2001; Xu & Wunsch, 2005): hierarchical and partitional. Both hierarchical clustering and partitional clustering have the drawback that the number of clusters need be specified a prior. For hierarchical clustering, the problem of cluster number selection is equivalent to decide in which level to cut the tree. Partitional clustering algorithms typically require the number of clusters as user input. However, the number of clusters in a data set is always not known beforehand in most situations. A variety of methods have been suggested try to estimate the number of clusters. The classical approach of determining the number of clusters is the use of some validity measures (Milligan & Cooper, 1985; Pal & Bezdek, 1995; Xie & Beni, 1991). For a given range of cluster number, the validity measure is evaluated for each given cluster number and then the value that optimizes the validity measure is chosen. The number of clusters searched by this method depends on the selected clustering algorithm, whose performance may rely on the initialization of the algorithm. Another method is progressive clustering (Krishnapuram & Freg, 1992; Krishnapuram, Frigui, & Nasraoui, 1995), the number of clusters is overspecified. After convergence, spurious clusters are eliminated and compatible clusters are merged. The main problem of this method is the measurement of spurious and compatible clusters. Moreover, they cannot guarantee that all clusters in the data set will be found. An alternative version of the progressive clustering is to seek one cluster at a time until no more "good" clusters can be found (Jolion, Meer, & Bataouche, 1991; Zhuang, Huang, Palaniappan, & Zhao, 1996). The performances of these techniques are also dependent on the validity functions, which are used to evaluate the individual clusters. In order to reduce the effect of the validity functions, a Weighted Sum Validity Function (WSVF), which is a weighted sum of several normalized validity functions, is proposed by Sheng, Swift, and Zhang (2005). Using more than one validity function via a weighted sum approach tends to increase the confidence of the clustering solutions. In this paper, we attempt to use a variable-length genetic algorithm to automatically evolve and find the optimal number of clusters as well as proper clusters of the data set.

Genetic algorithms (GAs) (Goldberg, 1989; Holland, 1975; Jong, 1975; Michalewicz, 1994), an imitation of natural selection and

* Corresponding author at: Institute of Information Science, Beijing Jiaotong University, Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing 100044, China.
E-mail address: chang_dongxia@hotmail.com (D. Chang).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.07.009

survival of the fittest, have been proved to be an efficient way in dealing with the optimization problem. In the past years, several clustering algorithm based on GA have been developed. These algorithms fall into two broad categories based on the representations for the clustering solutions. The first category uses a fixed-length string that the user should specify the desired number of clusters in advance to describe the clustering results (Bandyopadhyay & Maulik, 2002; Hall, Bözyurt, & Bezdek, 1999; Laszlo & Mukherjee, 2007; Maulik & Bandyopadhyay, 2000; Murthy & Chowdhury, 1996; Tucker, Crampton, & Swift, 2005). As the a priori knowledge on the number of clusters is often unavailable in most practical applications, it is important to design an algorithm which can automatically evolve a proper value of the center number as well as provide the appropriate clustering. A large variety of the second-category algorithms are adopting variable-length string, in which the number of cluster centers encoded into an individual is variable. Srikanth, George, and Warsi (1995) proposed a Pittsbugh-style GA for clustering where each individual contains a set of ellipsoid-shaped cluster descriptions. In this method, each cluster description consists of a set of parameters specifying the size and shape of an ellipsoid and all the parameters are encoded using binary digits. Ghozeil and Fogel (1996) proposed an evolutionary programming algorithm for clustering where each individual contains a set of hyperbox-shaped cluster descriptions. Both these two algorithms were making an assumption on the shape of the data set, when the data set violates the assumption the clustering results will be unsatisfactory. In order to overcome this drawback, Bandyopadhyay and Maulik (2002) proposed an automatic clustering algorithm which does not assume any particular underlying shape of the data set. But when the clusters are overlapping, this method prefers to class these clusters into one cluster. Saha and Bandyopadhyay (2009) proposed a fuzzy, point symmetry based genetic clustering technique (fuzzy-VGAPS), which can automatically determine the number of clusters present in a data set as well as a good fuzzy partitioning of the data.

In order to improve the performance of the GA-based algorithms, a new genetic clustering algorithm using a message-based similarity measure (GAMS) is presented in this paper. By utilizing a problem-specific chromosome structure and a set of genetic operators, the GAMS clustering algorithm can find the optimal number of clusters as well as proper structure of the data set automatically. In the new algorithm, a new similarity measure which we call the message-based similarity is proposed. This measure takes into account the messages exchanged between the data points and the candidate centers described by the chromosome. The usage of this new similarity improves the performance of the clustering greatly.

The rest of this paper is organized as follows. Section 2 provides the message-based similarity measure. Then a description of our GAMS clustering algorithm is presented in Section 3. The details of the new algorithm including the representation, the fitness evaluation function, the genetic operators are given in this section. Experimental results are provided for several artificial and real-life data sets are given in Section 4. The experimental results demonstrate the effectiveness of the GAMS clustering algorithm. Finally, conclusions are drawn in Section 5.

2. Message-based similarity

In this section, we propose a new similarity measure for the clustering criteria, which we call the message-based similarity. The measure is so called because it uses two kinds of message, responsibility and availability, exchanged between data points and the candidate centers, and each takes into account a different kind of competition. Here, the responsibility and availability between the data set $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ and the candidate centers set $\mathbf{C} = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_K\}$ are defined.

For the candidate center set $\mathbf{C}$, an input preference that candidate center $\mathbf{c}_k \in \mathbf{C}$ be chosen as a center is defined firstly. The candidate centers with larger values of input preference are more likely to be chosen as a center. If a priori, this value can be set according to the priori information. Here, we define it as

$$IP(k) = -\frac{1}{n}\sum_{i=1}^{n} d(\mathbf{x}_i, \mathbf{c}_k) = -\frac{1}{n}\sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{c}_k\|^2, \quad k = 1, 2, \ldots, K. \tag{1}$$

And this is the mean distance between a center and all the data points in the data set. Obviously, this value will be optimized when c_k is the center of the data set. Note that the distance measure here is chosen with the Euclidean norm. However, any suitable distance measure can be used to replace the Euclidean norm. Throughout this paper, we use the Euclidean norm. In the following, the responsibility and availability are defined.

The responsibility $r(i,k)$, sent from data point $\mathbf{x}_i$ to the candidate center $\mathbf{c}_k$, reflects the evidence for how well-suited $\mathbf{c}_k$ is to sever as the center for point $\mathbf{x}_i$, taking into account other potential centers for point $\mathbf{x}_i$. The availability $a(i,k)$, sent from candidate center $\mathbf{c}_k$ to point $\mathbf{x}_i$, reflects the evidence for how appropriate it would be for point $\mathbf{x}_i$ to choose $\mathbf{c}_k$ as its center, taking into account the support from other points that $\mathbf{c}_k$ should be an center. The responsibilities are computed using the rule

$$r(i,k) = d(i,k) - \max_{k', s.t. k' \neq k} \{d(i,k')\}, \quad i = 1, 2, \ldots, n, \quad k = 1, 2, \ldots, K, \tag{2}$$

where $d(i,k)$ denotes the distance between data point $\mathbf{x}_i$ and the candidate cluster center $\mathbf{c}_k$. Here the distance measure used is the Euclidean distance, i.e., $d(i,k) = -\|\mathbf{x}_i - \mathbf{c}_k\|^2$. A self-responsibility, $R(k)$, is defined as

$$R(k) = IP(k) - \max_{k', s.t. k' \neq k} \{d(\mathbf{c}_k, \mathbf{c}_{k'})\}, \quad k = 1, 2, \ldots, K, \tag{3}$$

i.e., the self-responsibility of $\mathbf{c}_k$ is defined as its input preference $IP(k)$ minus the largest of the similarities between center $\mathbf{c}_k$ and all other candidate centers. This self-responsibility reflects evidence that center $\mathbf{c}_k$ is a center, based on its input preference tempered by how ill-suited it is to be assigned to another center. A negative self-responsibility $R(k)$ indicates that center $\mathbf{c}_k$ is currently better suited as belonging to another center rather than being a center itself.

Whereas the above responsibility update lets all candidate centers compete for ownership of a data point, the following availability update gathers evidence from data points as to whether each candidate centers would make a good center

$$a(i,k) = \min\left\{0, R(k) + \sum_{i', s.t. i' \neq i} \max\{0, r(i',k)\}\right\},$$
$$i = 1, 2, \ldots, n, \quad k = 1, 2, \ldots, K. \tag{4}$$

This availability $a(i,k)$ reflects evidence that point $\mathbf{c}_k$ is a center, based on the positive responsibilities sent to candidate center from other points. Here, only the positive responsibilities are added, because it is only necessary for a good center to explain some data points well, regardless of how poorly it explains other data points.

After the computation of the responsibility and availability, the similarity between the data point and the candidate center is defined by the sum of the responsibility r and the availability a. That is to say, the similarities between data point $\mathbf{x}_i$ and the candidate centers $\mathbf{C} = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_K\}$ are

$$s(i,j) = r(i,j) + a(i,j), \quad j = 1, 2, \ldots, K, \tag{5}$$

then $\mathbf{x}_i$ will be assign to the cluster with the maximum similarity.