Pattern Recognition Letters 32 (2011) 1613–1621

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

A clustering method combining differential evolution with the *K*-means algorithm

Wojciech Kwedlo *

Faculty of Computer Science, Bialystok University of Technology, Wiejska 45a, 15-351 Białystok, Poland

ARTICLE INFO

Article history:
Received 21 April 2010
Available online 19 May 2011
Communicated by L. Heutte

Keywords:
Cluster analysis
Differential evolution
K-means algorithm

ABSTRACT

The present paper considers the problem of partitioning a dataset into a known number of clusters using the sum of squared errors criterion (SSE). A new clustering method, called DE-KM, which combines differential evolution algorithm (DE) with the well known *K*-means procedure is described. In the method, the *K*-means algorithm is used to fine-tune each candidate solution obtained by mutation and crossover operators of DE. Additionally, a reordering procedure which allows the evolutionary algorithm to tackle the redundant representation problem is proposed. The performance of the DE-KM clustering method is compared to the performance of differential evolution, global *K*-means method, genetic *K*-means algorithm and two variants of the *K*-means algorithm. The experimental results show that if the number of clusters *K* is sufficiently large, DE-KM obtains solutions with lower SSE values than the other five algorithms.

1. Introduction

Clustering (Jain et al., 1999; Kaufman and Rousseeuw, 1990) is an unsupervised classification technique which has applications in many areas, such as social sciences, biology, medicine and signal processing. Clustering can be described as dividing a set of objects into *K* disjoint groups, called clusters, in such a way that objects within one cluster are very similar, whereas objects in the different clusters are very distinct. In some applications, e.g. vector quantization (Gersho and Gray, 1992), the number *K* of clusters is known a priori. Alternatively, this number can be determined during the clustering process. In this paper, it is assumed that the number of clusters is known.

Given the dataset consisting of N feature vectors $X = \{x_1, \ldots, x_i, \ldots, x_N\}$, where $x_i \in \Re^M$, its partition Π can be defined as $\Pi = \{C_1, C_2, \ldots, C_K\}$, where $\forall_{i \neq j} C_i \cap C_j = \emptyset$, $\bigcup_{i=1}^{K} C_i = X$, $\forall_i C_i \neq \emptyset$. Thus, the clustering problem can be formulated as the problem of searching for a partition which minimizes a certain criterion function. One of the most popular criterion functions is the sum of squared errors (SSE) which can be defined as:

$$\mathrm{SSE}(X, \Pi) = \sum_{i=1}^{K} \sum_{x_j \in C_i} \|x_j - m_i\|^2, \tag{1}$$

where $\|\cdot\|$ is the Euclidean distance and m_i is the *centroid* of the cluster C_i, which can be computed as the sample mean:

$$m_i = \frac{1}{|C_i|} \sum_{x_j \in C_i} x_j. \tag{2}$$

The clustering problem with SSE as the clustering criterion is NP-hard for $K \geqslant 2$ (Aloise et al., 2009). Even for moderately sized datasets, heuristics, such as *K*-means algorithm (MacQueen, 1967), must be used to solve it in reasonable time. Unfortunately these heuristics are prone to being trapped in a local minimum of the SSE criterion.

There are several methods of dealing with the problem of local minima. The most straightforward approach involves running the *K*-means algorithm many times, each time starting with random initial conditions. Another approach uses a global optimization method, such as simulated annealing (Selim and Alsultan, 1991) or an evolutionary algorithm (EA). EAs (Michalewicz, 1996) are stochastic search techniques inspired by the concept of Darwinian evolution. Unlike local optimization methods, e.g. *K*-means, they simultaneously process a population of problem solutions, which gives them the ability to escape from local optima. Differential evolution (DE) (Storn and Price, 1997) is a relatively new EA, which has been successfully applied to many optimization problems.

In the paper we describe a DE-KM (differential evolution with *K*-means) algorithm which is able to recover high quality clustering solutions in terms of the SSE criterion. The main contribution of DE-KM is an incorporation of the *K*-means algorithm into the process of DE. The DE-KM uses the *K*-means algorithm in two ways. Firstly, the algorithm is used to obtain the centroids for each initial solution in the DE population. Secondly, the *K*-means algorithm is employed to fine-tune each new solution obtained by the mutation and crossover operators of the DE. In both cases, the *K*-means algorithm is run until convergence. In the

* Tel.: +48 85 7469743; fax: +48 85 7469057.
E-mail address: w.kwedlo@pb.edu.pl

0167-8655/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2011.05.010

experiments the DE-KM algorithm was applied to real-life datasets and its performance was compared with the performance of its two key components, namely DE and the K-means method.

The rest of this paper is organized as follows. The next section discusses research related to our work. Section 3 contains a description of the K-means algorithm. In Section 4 the details of the proposed DE-KM method are described. Section 5 presents the results of computational experiments. The last section of this paper contains conclusions.

2. Related research

Many EA-based approaches to the problem of clustering have been proposed (see Hruschka et al., 2009 for a comprehensive review). The main difference between them is the encoding of the partitions by elements of population (chromosomes). The most natural encoding (e.g., Murthy and Chowdhury, 1996; Krishna and Murty, 1999) uses chromosomes consisting of N elements with integer values from the interval of $[1,K]$, where N is the size of the learning set and K is the number of clusters. The ith element of the chromosome represents the number of a cluster to which the feature vector x_i belongs. For instance, for $N = 5$ and $K = 2$ a chromosome 21121 encodes the following partition of the learning set: $\{\{x_2,x_3,x_5\},\{x_1,x_4\}\}$. This approach is called a label-based representation (Hruschka et al., 2009), since each gene in a chromosome defines a cluster label of an object. A significant shortcoming of this representation is its redundancy. It is easy to notice that each partition can be encoded by $K!$ different chromosomes. For instance, the chromosomes 21121 and 12212 encode the same partition.

In an alternative approach (Fränti et al., 1997; Hall et al., 1999; Maulik and Bandyopadhyay, 2000; Paterlini and Krink, 2006; Laszlo and Mukherjee, 2007), which is employed in this paper, a real-valued chromosome encodes a set of K cluster prototypes (usually centroids). The length of a chromosome is MK, where M is the dimension of the feature space. The first M elements of the chromosome encode the coordinates of the first cluster centroid, the next M elements encode the coordinates of the second cluster centroid, and so forth. To obtain a partition of the dataset, each feature vector is allocated to the cluster represented by the closest centroid. This approach, called a centroid-based representation (Hruschka et al., 2009), is also flawed by the redundant representation problem, since any permutation of centroids in a chromosome will result in the same partition of the dataset.

A disadvantage of EAs, as compared to local search methods, is slower convergence. This shortcoming is particularly severe in the application of EAs to partitional clustering, when computation of an objective function, e.g. (1), requires a pass over the whole learning set X. For the centroid-based representation, the complexity of the computation of (1) is $O(NMK)$. It is a well established fact (Goldberg and Voessner, 1999; Culberson, 1999) that a combination of a local search with an EA can achieve much better efficiency than the EA only. Accordingly, some researchers integrated the K-means algorithm into their EAs. To fine-tune solutions obtained by the mutation, Krishna and Murty (1999) proposed a K-means operator, which performs a single iteration of the K-means algorithm. A similar technique was used by Maulik and Bandyopadhyay (2000). Fränti et al. (1997) employed two iterations of K-means to improve each new solution obtained by recombination operators of a genetic algorithm. In contrast to these approaches, the method of Laszlo and Mukherjee (2007) fine-tunes each solution by running the K-means algorithm until convergence.

Differential evolution (DE) is an evolutionary algorithm proposed by Storn and Price (1997), employing a representation based on real-valued vectors. It has been successfully applied to many optimization problems. DE is based on the usage of vector differences for perturbing the population elements. A version of DE with self-adaptation of control parameters was described by Brest et al. (2006).

The idea of using DE for partitional clustering is not new. Paterlini and Krink (2006) employed a centroid-based representation to investigate the performance difference between DE, genetic algorithm and particle swarm optimizer. The results indicated that the performance of DE is superior to the performance of the other two approaches. Omran et al. (2005) also used the centroid-based representation in the application of DE to clustering of image pixels. A similar experiment was performed by Sudhakar et al. (2010). Das et al. (2007) modified the centroid-based representation by augmenting each centroid with a real number from the interval $[0,1]$, called an activation threshold. If the threshold has a value greater than 0.5, the corresponding centroid is used in the partitioning of the dataset. Otherwise, the centroid remains inactive. In this way, their ACDE (Automatic Clustering with DE) algorithm is able to discover the number of clusters by optimizing a cluster validity index. Tian et al. (2009) combined DE with the K-harmonic means local optimization algorithm (KHM) (Zhang et al., 1999). In their approach one iteration of KHM is applied to all DE population members after the selection. Their results indicate that the hybrid DE-KHM method optimizes a criterion based on the harmonic mean better than DE or KHM. However, as far as we know, no application of a combination of DE with K-means to optimize SSE (1) has been proposed.

3. *K*-means algorithm

The K-means algorithm (MacQueen, 1967) is the most popular clustering algorithm minimizing SSE (1). It is an iterative algorithm which can be described by the following steps.

1. Choose initial centroids $\{m_1,\ldots,m_K\}$ of the clusters $\{C_1,\ldots,C_K\}$.
2. Calculate new cluster membership. A feature vector x_j is assigned to the cluster C_i if and only if

$$i = \operatorname*{arg\,min}_{k=1,\ldots,K} \|x_j - m_k\|^2. \qquad (3)$$

3. Recalculate centroids for the clusters according Eq. (2).
4. If none of the cluster centroids have changed, finish the algorithm. Otherwise go to Step 2.

The K-means algorithm is easy to implement and computationally efficient. However, it has an essential deficiency. Although it converges in a finite number of iterations (Selim and Ismail, 1984), it can be easily trapped in a local minimum of the SSE criterion function (1). Consequently, the quality of the solution obtained by means of the algorithm is heavily dependent on its initial conditions (the centers of initial clusters).

4. DE-KM algorithm

4.1. Differential evolution

Several versions of DE have been proposed. For the purpose of this study the most common variant is used, which, according to the classification proposed by Storn and Price (1997), can be described as DE/rand/1/bin.

Like all EAs, DE maintains a population of S solutions of the optimization problem. At the start of the algorithm, members of the population are initialized randomly with the uniform distribution. Then DE performs multiple iterations in three consecutive steps: reproduction (creation of a temporary population), computing of