제22회 한국정보처리학회 추계학술발표대회 논문집 제11권 제2호 (2004. 11)

유전자 알고리즘 기반 적응 군집화 알고리즘

박남현, 안창욱, R.S. Ramakrishna
광주과학기술원 정보통신공학과
e-mail : nhpark@gist.ac.kr

An Adaptive Clustering Algorithm Based on Genetic Algorithm

Namhyun Park, Chang Wook Ahn, R.S. Ramakrishna
Dept. of Information Communication, Gwangju Institute of Science and Technology

Abstract

This paper proposes a genetically inspired adaptive clustering algorithm. The algorithm automatically discovers the actual number of clusters and efficiently performs clustering without unduly compromising cluster purity. Chromosome encoding that ensures the correct number of clusters and cluster purity is discussed. The required fitness function is designed on the basis of modified similarity criteria and genetic operators. These are incorporated into the proposed adaptive clustering algorithm. Experimental results show the efficiency of the clustering algorithm on synthetic data sets and real world data sets.

1 Introduction

Clustering refers to the operation of grouping objects with similar patterns within given data sets. This is an important task in many applications including bio-informatics, web data analysis, information retrieval, CRM (customer relationship management), text mining, and scientific data analysis [1]. The literature is replete with distance-based, density-based, graph-based, model-based, and evaluation-based clustering methods [2]. An appropriate decision threshold is crucial to the success of these methods. This threshold is a priori unknown in real world applications.

Accurate clustering involves the two features listed below [4]:

- The number of clusters: This is the number of clusters that results after optimal clustering.
- Purity: This is the number of objects primarily belonging to a single class that each cluster contains. The larger the purity value, the better is the result of clustering.

If the number of clusters is the same as the number of clusters that is already known and the purity is as high as possible, then the clustering procedure is effective and acceptable. The number of clusters is intimately related to purity. Purity deteriorates rapidly when the number of predicted clusters is small and the cluster size is large. Cluster validity index helps in this regard [3]. An appropriate decision threshold is very important for acceptable results.

For example, in the clustering algorithm based on partitions, the predicted number of clusters (k) is a crucially important variable that impacts on whether some objects are included in the same cluster or not. The threshold and related variables play an important role in ensuring the discovery of the actual number of accurate clusters with acceptable purity. They are decided by trial and error through complicated procedures [1], [5]. Cluster validity indices are the culmination of efforts in this direction. These approaches do not solve the primary problem of selecting the threshold for real world data sets.

Evolution-based clustering and genetic algorithm-based clustering offer hope in this regard. It must be pointed out, however, that early genetic algorithm-based clustering algorithms such as GKA still select the appropriate threshold from among many suggested thresholds [6], [7]. Fitness functions arising out of cluster validity indices have also been studied extensively. There is also the problem of the decision that transforms the threshold to the specific, related variable. Automatic computation of decision thresholds is under investigation of late.

CHyGA is a genetic algorithm-based clustering algorithm [8]. However, CHyGA adopts an existing cluster validity index, viz., the Calinski and Harabasz criterion (CH). This algorithm uses a fitness function and mutation operators.

This paper proposes an adaptive clustering algorithm based on genetic algorithm that maintains the number of actual clusters and purity. The proposed algorithm

automatically decides the threshold. The paper is divided into two parts. The first part discusses a chromosome encoding method using a new mechanism that involves the number of actual clusters and purity. The second concerns the fitness function designed on the basis of similarity and dissimilarity discussed in section 2. The proposed clustering algorithm is not hampered by the threshold decision problem like many existing clustering algorithms. It returns the patterns with the correct number of accurate clusters and maintains cluster purity.

Section 2 shows the similarity criterion that is needed in the design of the fitness function. Section 3 develops the proposed algorithm. The new chromosome encoding method and the fitness function are described there. Sections 4 and 5 present experimental results and conclusions respectively.

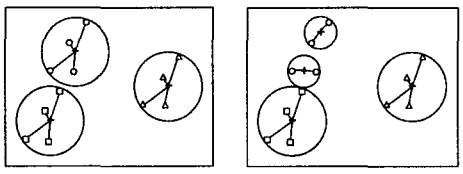
2 Similarity criteria

This section presents a modification to the existing distance-based similarity. This leads to an adaptive clustering algorithm that forms clusters while continuously monitoring cluster accuracy and purity.

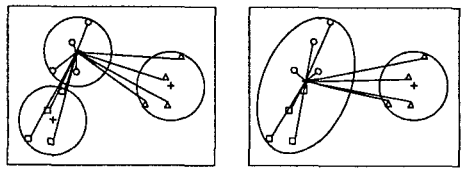
Similarity The proposed algorithm adopts distance-based similarity. Distance measurements generally include Euclidean, Manhattan, and Chebychev distances. The proposed algorithm uses Euclidean distance for similarity measurements. It involves relations between centroids and objects [2]. The proposed intra-similarity and inter-dissimilarity follow this kind of approach.

Intra-similarity The intra-similarity is the mean of the distances among centroids and objects in the same cluster. In the case of three and four clusters, the intra-cluster distance of four clusters is smaller than that of three clusters, and the intra-similarity of four clusters is larger than that of three clusters (See Figure 1a). Moreover, the intra-similarity induces larger clusters during grouping. This is the same as the well known definition of intra-similarity.

Inter-dissimilarity The proposed inter-dissimilarity is different from existing inter-dissimilarity. This involves relations between centroids of clusters and all the objects during clustering. The inter-cluster distance for a single cluster is given by the sum of distances between the cluster and the objects. In the case of three and two clusters for instance, the inter-cluster distance of two clusters is smaller and the inter-dissimilarity is larger than that of three clusters (See Figure 1b). The proposed inter-dissimilarity different from intra-similarity derives smaller clusters during grouping.



(a) Clustering results derived by intra-similarity (from three to four clusters)



(b) Clustering results derived by inter-dissimilarity (from three from two clusters)

Figure 1. The trade-off between intra- and inter-dissimilarity

The proposed algorithm uses the larger of the intra-similarity and inter-dissimilarity values. The correct number of accurate clusters and high purity occur when both have larger values. If the intra-cluster distance and inter-cluster distance are small during grouping, then the objects in the same cluster have similar patterns and they are therefore naturally included in the same cluster. This is similar to the cluster validity index - if the cluster validity index after merging and splitting of objects has a larger value than before, then the two objects are similar and are included in the same cluster. If cluster validity index has to exhibit the smaller value, the optimum value is small after grouping the objects. In contrast, if the cluster validity index has to exhibit a larger value, then the optimum value is larger after grouping the objects [3]. However, this paper is also concerned with the design of a fitness function so as not to lose the information on the number of accurate clusters and cluster purity during evolution.

3 The proposed algorithm

This section describes the genetic algorithmic framework of the proposed adaptive clustering algorithm. Two keys issue in this regard pertain to a new chromosome encoding method and a new fitness function. The genetic operators of the proposed algorithm are also described.

3.1 Chromosome encoding

Existing clustering algorithms based on genetic algorithms involve cluster purity. The number of accurate clusters can not be accommodated as they rely primarily on binary chromosome encoding. Non-binary encoding for purity is controlled by cluster indices for chromosomes as in Figure 2a [6], [7]. Thus, these chromosomes do not have the information about the number of accurate clusters for dominant genes (dominant cluster indices) and recessive genes (recessive cluster indices) in chromosomes. If chromosomes are evolved with the information about dominant and recessive genes, then the chromosomes with the dominant character can be propagated to the next generation (See Figure 2b).

In this study, an approximate range of threshold is suggested. The chromosome encoding is illustrated by a simple example (See Figure 2b).

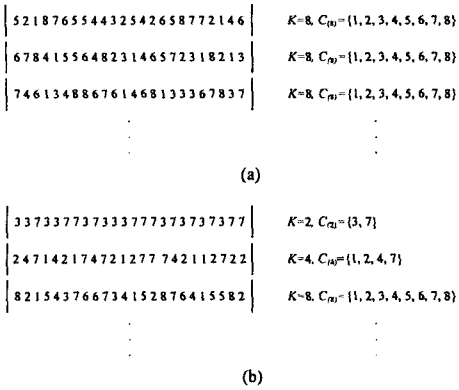


Figure 2. Chromosome encoding

Hence, K is the number of distinct values that genes can take in chromosomes. K is decided from a uniform distribution over the range from one to eight. $C_{(K)}$ is the set of K characters selected from the same distribution. In other words, K is the number of cluster indices that objects can take when clusters are grouped, and $C_{(K)}$ is the set of indices of K clusters. Each chromosome is considered with the number of clusters (threshold) according to K and the purity according to $C_{(K)}$. For example, in Figure 2b, Chromosomes are encoded by a range of eight clusters.

3.2 Fitness function

General cluster validity index and fitness function for distance-based clustering algorithms are designed on the basis of similarity [8]. In the design of the fitness function for genetic algorithm-based clustering algorithms, the Euclidean distance is quite appropriate. In this study, we use the Euclidean distance to design the fitness function. The fitness function is based on the intra-similarity and inter-dissimilarity between centroids and objects. This fitness function uses the selection and evaluation of genetic algorithms (as in section 3.3).

Recently, genetic algorithm-based clustering algorithms have introduced fitness functions that involve the intra-similarity and several additional mutation operators. We consider intra-similarity and inter-dissimilarity (Section 2) and design a new fitness function.

In the Equation 1 below, N is the total number of objects in specific data sets, K is the maximum value in an approximate range of the number of predicted clusters, c_k is the centroid (of a cluster), x_i are the objects, and d is the dimension of the data set.

$$\begin{aligned} Fitness &= Fit_{intra} * Fit_{inter} \\ Fit_{intra} &= \sum_{k=0}^{K-1} \sum_{i=0}^{N_k-1} \|c_k - x_i\|^2 \\ Fit_{inter} &= \left[\sum_{k=0}^{K-1} \sum_{i=0}^{N-1} \|c_k - x_i\|^2 \right]^{\frac{1}{d}} \end{aligned} \quad (1)$$

Equation 1 is the fitness function of the adaptive, genetic algorithm-based clustering algorithm. The fitness is accurate when the value is small. Also, this is divided into two parts, Fit_{intra} and Fit_{inter}, which have specific characteristics. If Fit_{intra} alone is used in the fitness function, it forms the largest number of clusters. Conversely, if the Fit_{inter} alone is used, it forms the fewest number of clusters. The larger the number of clusters, the lower is the fitness value. Equation 1 is similar to cluster validity index. However, Fit_{inter} includes a power to $1/d$ because the increase (decrease) rate of Fit_{inter} and Fit_{intra} are different. So, the power of $1/d$ reflects the familiar increasing (decreasing) rate of Fit_{intra} and Fit_{inter}.

3.3 Genetic operators

Genetic algorithms are stochastic search mechanisms. They are efficient and effective search algorithms. They have been successfully used in a variety of applications in business, engineering, and science. Genetic operators needed in the clustering algorithm based on genetic algorithms are discussed below [6], [7], [8].

Initialization The earliest generation uses the new chromosome encoding method suggested in section 3.1. The size of the generation is defined by $n=N*K*d$.

Selection and evaluation The selection operator selects the chromosome for evaluation, and the evaluation operator calculates the fitness value of the chromosome for the next generation. The tournament method is used in the proposed algorithm, and this method extracts the chromosome (that includes dominant genes) from out of the two chromosomes which are randomly selected from among n chromosomes and passes on the dominant chromosome to the next generation. The tournament probability is 0.8, and the fitness function is the one presented in section 3.2.

Crossover The crossover operator forms the new chromosome from two chromosomes that carry dominant genes that are selected as described above. It works with one-point crossover and the crossover probability is 1.0.

Mutation If there is a single cluster index in the entire chromosomes, then there is just one object. Mutation operator is useful in this case.

Termination The algorithm terminates if the chromosomes have the same shape.

4 Experiments

In this section, we describe the performance measures and experimental results.

4.1 Performance measure

The result of clustering is evaluated by the number of accurate clusters and cluster purity. The cluster accuracy is measured by how well the clusters have been formed and by the number of clusters as compared with the number of

suggested clusters. Equation 2 represents cluster accuracy. Here, $|\tilde{c}|$ is the number of computed clusters, and $|c|$ is the number of suggested clusters.

$$Cluster\ Accuracy = \delta_{|\tilde{c}|=|c|} \quad (2)$$

where δ is the Kronecker delta function.

We measure the error rate of clusters after grouping the objects by Equation 3. The error rate counts the number of objects misclassified. Here, e_i is the impurity of a cluster.

$$Error\ Rate = \frac{\sum_{i=0}^{|c|-1} e_i}{N} \quad (3)$$

4.2 Experimental results

We experimented with several data sets - three synthetic data sets and two real world data sets. These synthetic data sets are of 2, 3, and 4 dimensions, and all have four clusters. Each dimension covers a Gaussian distributed interval. Ruspini and Iris are the real world data sets. They have 4 and 3 clusters respectively [9], [10]. Table 1 is the summary of experimental data sets. Table 2 is the error rate for experimental data sets clustered by the *k-means* algorithm. The reported values are the threshold as k and the mean of the error rate of 100 iterations.

Table 1. Summary of experimental data sets

Data	Objects	Dimension	Clusters
2d_data	200	2	4
3d_data	200	3	4
4d_data	200	4	4
Ruspini	75	2	4
Iris	150	4	3

Table 2. The error rate from the *k-means* algorithm

Data	Threshold (k)	Error rate
2d_data	4	0.0171
3d_data	4	0.0156
4d_data	4	0.0174
Ruspini	4	0.0160
Iris	3	0.0913

It is seen that the *k-means* algorithm classify well for given experimental data sets. The error rate is nearly zero. Table 3 is the cluster accuracy and the error rate. Here, we can compare with the error rate of the *k-means* algorithm and the proposed algorithm for experimental data sets.

Table 3. The cluster accuracy and the error rate from the proposed algorithm

Data	Cluster accuracy	Error rate
2d_data	1.0000	0.0001
3d_data	1.0000	0.0005
4d_data	1.0000	0.0000
Ruspini	0.9700	0.0013
Iris	1.0000	0.0283

For Table 3, the reported values are the mean of results of 100 iterations as well. It is seen that cluster accuracy is nearly perfect and error rate is almost zero. This adaptive clustering algorithm based on genetic algorithm is better than the *k-means* algorithm in respect of the error rate. The efficiency of the proposed algorithm and its ability to automatically decide the threshold are noteworthy in terms of cluster accuracy.

5 Conclusion

This paper suggested an adaptive clustering algorithm based on genetic algorithms. It automatically finds the number of accurate clusters and efficiently performs clustering without unduly compromising cluster purity. A modified similarity criterion was suggested. The framework of genetic algorithms was outlined for clarity. A new chromosome encoding method that involves the number of accurate clusters and the fitness function that have an adaptive trait for clustering were also presented. Finally, the efficiency of the proposed algorithm was demonstrated through experiments on several data sets. We are also working on other types of data and other types of fitness function.

References

[1] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2001.

[2] A.K. Jain, M.N. Murty, and P.J. Flynn, "Data Clustering: A Review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264-323, 1999.

[3] D.-J. Kim, Y.-W. Park, and D.-J. Park, "A Novel Validity Index for Determination of Optimal Number of Clusters," *IEICE Trans. Inf. & Syst.*, vol. E84-D, No. 2, Feb. 2001.

[4] New Wave Computing Laboratory, *Clustering Validation Techniques Reflecting Characteristics of Datasets*, Gwangju, Korea, 2003.

[5] Z. Huang, "Extensions to the k-means Algorithm for Clustering Large Data Sets Categorical Values," *Data Mining and Knowledge Discovery*, vol. 2, 1998.

[6] E.Falkenauer, *Genetic Algorithms and Grouping Problems*, John Wiley, 1998.

[7] K. Krishna and M. Narasimha Murty, "Genetic K-Means Algorithm," *IEEE Transaction on Systems, Man and Cybernetics* – Part B: Cybernetics, 1999.

[8] Laetitia Vermeulen-Jourdan, Clarisse Dhaenens, and El-Ghazali Talbi, "Clustering Nominal an Numerical Data: A New Distance Concepts of a Hybrid Genetic Algorithm," in *Proceedings of EvoCOP 2004*, Coimbra, Portugal, 2004.

[9] C.L. Blake and C.J. Merz, *UCI Repository of machine learning databases*, http://www.ics.uci.edu/~mlearn/MLRepository.html, Univ. of California, Irvine, Dept. of Information and Computer Sciences, 1998.

[10] E.H. Ruspini, "Numerical methods for fuzzy clustering," in *Proceedings of Inform. Sci.*, 1970.