Data Clustering Using Virtual Population based Incremental Learning Algorithm with Similarity Matrix Encoding Strategy

Yi Hong
Department of Comp. Sci.
City University of Hong Kong
yihong@cityu.edu.hk

Sam Kwong
Department of Comp. Sci.
City University of Hong Kong
CSSAMK@cityu.edu.hk

Hui Xiong
MSIS Department
Rutgers University
hxiong@rutgers.edu

Qingsheng Ren
Dep. of Comp. Sci. and Eng.
Shanghai Jiao Tong University
ren-qs@cs.sjtu.edu.cn

ABSTRACT

Data clustering is a good benchmark problem for testing the performance of many combinatory optimization methods. However, very few works have been done on using the estimation of distribution algorithms for solving the problem of data clustering. The purpose of this paper is to demonstrate the effectiveness of the estimation of distribution algorithms for solving the problem of data clustering. In particular, a novel encoding strategy termed as the Similarity Matrix Encoding strategy (SME) and a Virtual Population Based Incremental Learning algorithm using SME encoding strategy (VPBIL-SME) are proposed for clustering a set of unlabeled instances into groups. Effectiveness of VPBIL-SME is confirmed by experimental results on several real data sets.

Categories and Subject Descriptors

I.5 [**Pattern Recognition**]: Clustering; I.2.8 [**Artificial Intelligence**]: Heuristic methods

General Terms

Algorithms

Keywords

Data Clustering, Similarity Matrix Encoding Strategy, Virtual Population based Incremental Learning Algorithm

1. SME

Let $D = \{x_i, x_2, ..., x_L\}$ denote a data set containing L instances, x_{ij} represent the j^{th} feature of the instance x_i and each instance has N features and K be the number of groups that has been known beforehand. A clustering solution of the data set D can be defined as an integer label string $I = \{I_1, I_2, ..., I_L\}$ as follows:

$$I_i = k, \quad \text{if } x_i \text{ is classified into the } k^{th} \text{ group} \tag{1}$$

Copyright is held by the author/owner(s).
GECCO'08, July 12–16, 2008, Atlanta, Georgia, USA.
ACM 978-1-60558-130-9/08/07.

where $i = 1, 2, ..., L$ and $k = 1, 2, ..., K$. Provided that the data set has L instances, SME encodes each chromosome as a $L \times L$ binary matrix S as follows:

$$S_{jk} = \begin{cases} 1 & \text{if } x_j \text{ and } x_k \text{ are classified into the same group;} \\ 0 & \text{otherwise;} \end{cases} \tag{2}$$

where $j = 1, 2, ..., L$ and $k = 1, 2, ..., L$. It is considered that the similarity matrix is a binary matrix, therefore the whole code space of chromosomes is equal to $2^{(L^2)}$ that is significantly larger than $\frac{K^L}{K!}$ of the problem space. The redundancy of the code space of the SME encoding strategy means that one clustering solution can be encoded by around $\frac{2^{(L^2)} \times K!}{K^L}$ different chromosomes. However, unlike other existing encoding strategies these $\frac{2^{(L^2)} \times K!}{K^L}$ chromosomes are very similar. This characteristic guarantees that the SME encoding strategy has the ability of avoiding the problem of context insensitivity of other existing encoding strategies.

Although SME is capable of avoiding the problem of context insensitivity of other encoding strategies, it also leads to the compounded difficulty of evaluation of candidate chromosomes, the increased memory requirements and the expanded search space of the minimization of the within-cluster variation. However, the following paragraphs will illustrate that the above three shortcomings of SME can be solved.

2. FITNESS EVALUATION METHOD

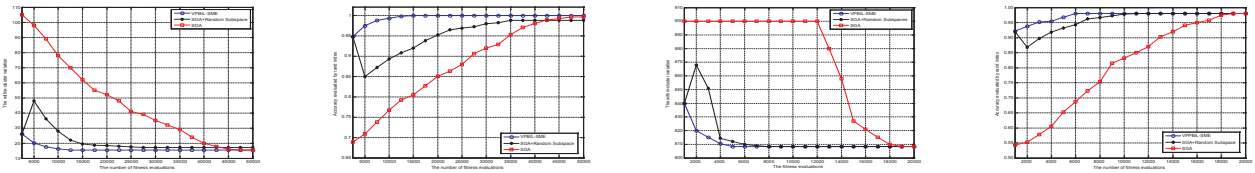
Let S denote a chromosome of a $L \times L$ binary matrix in the population, the similarity matrix should be firstly transformed into a distance matrix as follows:

$$D_{jk} = 1 - S_{jk} \tag{3}$$

where $j = 1, 2, ..., L$ and $k = 1, 2, ..., L$. Then a clustering solution I of the data set can be obtained through executing the average link clustering algorithm on the distance matrix D as follows:

$$I = AverageLink(D) \tag{4}$$

The fitness value of the chromosome S of the similarity matrix is set as the within-cluster variation of the clustering solution I. Based on the definition of the within-cluster



(a) X8D5K (b) X8D5K (c) Wisconsion (d) Wisconsion

Figure 1: Experimental results obtained by VPBIL-SME, SGA+Random Spaces and SGA.

variation, the fitness value $f(S)$ of the chromosome S of the similarity matrix can be calculated as follows:

$$f(S) = v(I) = \sum_{k=1}^{K}\sum_{i=1}^{L}\left[\delta(x_i, c_k)\sum_{j=1}^{N}(x_{ij} - c_{kj})^2\right] \quad (5)$$

3. DENSITY INITIALIZATION

As illustrated in the aforementioned paragraphs, SME increases the size of the search space. Therefore, more time consumption is required to converge. In this paper, VPBIL-SME adopts the random subspaces method to generate its initial density for speeding up its convergence with the following steps employed: part of all features are randomly selected from the full feature set; then a clustering solution is obtained by executing K-means clustering algorithm on the selected features; the above two steps iterate until a population of clustering solutions are achieved; the initial density is obtained through combining these clustering solutions together. Provided that $\{I^{(1)}, I^{(2)}, ..., I^{(T)}\}$ are T clustering solutions obtained by the random subspace method, then the initial density P^1 of VPBIL-SME is set as follows:

$$P^1_{jk} = \frac{\sum_{i=1}^{T}\delta(I^{(i)}_j, I^{(i)}_k)}{T} \quad (6)$$

where $j = 1, 2, ..., L$, $k = 1, 2, ..., L$ and the function $\delta(I^{(i)}_j, I^{(i)}_k)$ can be calculated as follows:

$$\delta(I^{(i)}_j, I^{(i)}_k) = \begin{cases} 1 & \text{if } I^{(i)}_j = I^{(i)}_k; \\ 0 & \text{otherwise;} \end{cases} \quad (7)$$

The inspiration of using random subspaces method for density initialization is that the random subspaces method was known as an effective method for providing us a population of high-quality and diverse clustering solutions.

4. FRAMEWORK OF VPBIL-SME

In order to cut down the large memory requirement of PBIL, this paper adopts the virtual population based incremental learning algorithm (VPBIL) to perform the search task. The concept of virtual population based estimation of distribution algorithms was proposed in [1]. The authors claimed that virtual population based estimation of distribution algorithms were able to obtain a comparative solution under a much less memory requirement when compared with real population based estimation of distribution algorithms. The whole framework of VPBIL-SME is given in Algorithm 1. It can be observed from Algorithm 1 that VPBIL-SME only needs to hold two chromosomes S^{best} and S in the memory during its execution. The finish condition of VPBIL-SME is set as the maximal number of fitness evaluations. Experimental results are given in Figure 1.

Algorithm 1 Framework of VPBIL-SME.

(1) $t = 1$;
(2) $P^t \leftarrow$ set the initial density of promising solutions by using the random subspace method;
(3) $S^{best} \leftarrow$ generate one chromosome by sampling from the density P^t;
(4) $D = 1 - S^{best}$
(4) $I^{best} = AverageLink(D)$;
(5) $f(S^{best}) = v(I^{best})$;
(6) for $i = 1 : M - 1$, do:
  (a) $S \leftarrow$ generate one chromosome by sampling from the density P^t;
  (b) $D = 1 - S$
  (c) $I = AverageLink(D)$;
  (d) $f(S) = v(I)$;
  (e) if $f(S) < f(S^{best})$, then do:
    $I^{best} = I$, $f(S^{best}) = f(S)$;
  (f) remove the chromosome S from the memory;
(7) estimate the density P^{t+1}:
  $P^{t+1}_{jk} = P^t_{jk} + \lambda \cdot (\delta(I^{best}_j, I^{best}_k) - P^t_{jk})$
  for $j = 1, 2, ..., L$ and $k = 1, 2, ..., L$;
(8) if the finish condition is not met, $t = t + 1$ then go to (3).

M : the virtual population size.

5. ACKNOWLEDGMENTS

This work was supported by grant 7002073, CITYU-HK.

6. REFERENCES

[1] Y. Hong, S. Kwong, Q. Ren, and X. Wang. A comprehensive comparison between real population based tournament selection and virtual population based tournament selection. In *IEEE Congress on Evolutionary Computation (CEC2007)*, pages 445–452, 2007.