GENETIC k-MEANS CLUSTERING ALGORITHM FOR MIXED NUMERIC AND CATEGORICAL DATA SETS

Dharmendra K Roy and Lokesh K Sharma

Department of Information Technology and MCA
Rungta College of Engineering and Technology, Bhilai (CG) - INDIA
roy.dharmendra@gmail.com, lksharmain@gmail.com

ABSTRACT

Clustering is one of the major data mining tasks and aims at grouping the data objects into meaningful classes (clusters) such that the similarity of objects within clusters is maximized, and the similarity of objects from different clusters is minimized. In this paper we present a clustering algorithm based on Genetic k-means paradigm that works well for data with mixed numeric and categorical features. We propose a modified description of cluster center to overcome the numeric data only limitation of Genetic k-mean algorithm and provide a better characterization of clusters. The performance of this algorithm has been studied on benchmark data sets.

KEYWORDS

Data mining, Genetic Algorithm, Clustering Algorithm, Numeric data, Categorical data

1. INTRODUCTION

Partitioning a set of objects in databases into homogeneous groups or clusters is a fundamental operation in data mining. It is useful in a number of tasks, such as classification (unsupervised) aggregation and segmentation or dissection [4]. The problem of clustering in general deals with partitioning a data set consisting of n points embedded in m-dimensional space into k distinct set of clusters, such that the data points within the same cluster are more similar to each other than to data points in other clusters. The three sub-problems[1] addressed by the clustering process are (i) defining a similarity measure to judge the similarity (or distance) between different elements (ii) implementing an efficient algorithm to discover the clusters of most similar elements in an unsupervised way and (iii) derive a description that can characterize the elements of a cluster in a succinct manner. Traditional clustering algorithms used Euclidean distance measure to judge the similarity of two data elements [5] [6]. This works well when the defining attributes of a data set are purely numeric in nature. However, Euclidean distance measure fails to capture the similarity of data elements when attributes are categorical or mixed. Increasingly, the data mining community is inundated with a large collection of categorical data [3] like those collected from banks, or health sector, web-log data and biological sequence data. Banking sector or health sector data are primarily mixed data containing numeric attributes like age, salary, etc. and categorical attributes like sex, smoking or non-smoking, etc. Clustering mixed data sets into meaningful groups is a challenging task in which a good distance measure, which can adequately capture data similarities, has to be used in conjunction with an efficient clustering algorithm [2]. In order to handle mixed numeric and categorical data, some of the strategies that have been employed are as follows:

(1) The numeric distance measures can be applied for computing similarity between object pairs after conversion of categorical and nominal attribute values to numeric integer values. However, it is very difficult to give correct numeric values to categorical values.

(2) Another approach has been to discretize numeric attributes and apply categorical clustering algorithm. But the discretization process leads to loss of information.

10.5121/ijaia.2010.1203

In terms of clustering, we are interested in Genetic Algorithms (GA) which can efficiently cluster large data sets containing mixed numeric and categorical values because such data sets are frequently encountered in data mining applications. This paper is organized as follows. A brief review of the GA based clustering techniques in the next. After Next section proposed genetic k-means clustering algorithm for mixed numeric and categorical data is illustrated. Finally, experiment result is shown and we concluded our work.

2. RELATED WORK ON GENETIC CLUSTERING ALGORITHM

Evolutionary algorithms are stochastic optimization algorithms based on the mechanism of natural selection and natural genetics [5]. They perform parallel search in complex search spaces. Evolutionary algorithms include genetic algorithms, evolution strategies and evolutionary programming. Genetic algorithms (GA's) were originally proposed by Holland [1] [5]. GA's have been applied to many function optimization problems and are shown to be good in finding optimal and near optimal solutions. Their robustness of search in large search spaces and their domain independent nature motivated their applications in various fields like pattern recognition, machine learning, VLSI design, etc. Krishna and Murty proposed a new clustering method called genetic k-means algorithm (GKA) [5], which hybridizes a genetic algorithm with the k-means algorithm. This hybrid approach combines the robust nature of the genetic algorithm with the high performance of the k-means algorithm. As a result, GKA will always converge to the global optimum faster than other genetic algorithms.

Lu et al [8] proposed fast genetic k-means cluster technique (FGKA). It is a faster version of GKA and FGKA that features several improvements over GKA including an efficient evaluation of the objective value TWCV (Total Within-Cluster Variation), avoiding illegal string elimination overhead, and a simplification of the mutation operator. These improvements result that FGKA runs 20 times faster than GKA [5]. Although FGKA outperforms GKA significantly, it suffers from a potential disadvantage. If the mutation probability is small, then the number of allele changes will be small, and the cost of calculating centroids and TWCV from score can be much more expensive. Lu et al proposed an incremental genetic k-means algorithm (IGKA) [6] to overcome problem of FGKA. IGKA inherits all the advantages of FGKA including the convergence to the global optimum, and outperforms FGKA when the mutation probability is small. The main idea of IGKA is to calculate the objective value TWCV and to cluster centroids incrementally. IGKA performs well compare as FGA when mutation probability is smaller than some threshold but not when mutation probability is larger than some threshold. Therefore, a hybrid genetic k-means algorithm (HGKA) is proposed. HGKA combines the benefits of FGKA and IGKA and performs well in smaller and larger mutation probability. Basic foundations of these GA based clustering techniques are k-means clustering and it can deal only numeric data sets. Therefore, a genetic clustering algorithm (called GKMODE) is proposed. GKMODE integrates a k-modes algorithm [6] introduced by Chaturvedi et al and the genetic algorithm. GKMODE works only for categorical data but it is not able to handle mixed numeric and categorical data.

3. GENETIC K-MEANS CLUSTERING ALGORITHM FOR MIXED NUMERIC AND CATEGORICAL DATA

In this section we will describe proposed genetic k-means clustering algorithm for mixed numeric and categorical data.

3.1. Objective Function

The data for clustering consists of N genes and their corresponding N patterns. Each pattern is a vector of D dimensions recording the expression levels of the genes under AGKA each of the D monitored conditions or at each of the D time points. The goal of AGKA algorithm is to partition the N patterns into user-defined K groups. The Total Within-Cluster Variation (TWCV) is used to minimize for clustering in GKA, FGKA and IGKA. It can define as Eq. (1). Let *X1, X2,…, XN* be the *N* patterns, and *Xnd* denotes the *dth* feature of pattern *Xn* (*n*=1…*N*). Each partitioning is represented by a string, a sequence of numbers *a1…aN*, where *an* takes a value from {1, 2,…, K} representing the cluster number that pattern *Xn* belongs to. Let *Gk* denote the *kth* cluster and *Zk* denote the number of patterns in *Gk*. The Total Within-Cluster Variation (*TWCV*) is define as [6]

$$TWCV = \sum_{n=1}^{N}\sum_{d=1}^{D} X_{nd}^{2} - \sum_{k=1}^{K}\frac{1}{Z_k}\sum_{d=1}^{D} SF_{kd}^{2} \qquad (1)$$

SFkd is the sum of the *d*th features of all the patterns in *Gk*. Above function can use to handle the numeric attribute. Here we are using modified cost function specified in Eq. (2), which is to be minimized for clustering mixed data sets has two distinct components, one for handling numeric attributes and another for handling categorical attributes. The cost function can define for clustering mixed data sets with n data objects and m attributes (m_r numeric attributes, m_c categorical attributes, $m = m_r + m_c$) as

$$\Psi = \sum_{i=1}^{n} V(d_i, C_j), \qquad (2)$$

Where $V(d_i, C_j)$, is the distance of a data object d_i from the closest cluster center C_j. $V(d_i, C_j)$ is defined as Eq. (3)

$$V(d_i, C_j) = \sum_{t=1}^{m_r-1}(w_t(d_{it}^{r} - C_{jt}^{r}))^2 + \sum_{t=1}^{m_c-1}\Omega\left(d_{it}^{c}, C_{jt}^{c}\right)^2 \qquad (3)$$

Where $\sum_{t=1}^{m_r-1}(w_t(d_{it}^{r} - C_{jt}^{r}))^2$denotes the distance of object di from its closest cluster center Cj, for numeric attributes only, w_t denotes the significance of the *t*th numeric attribute, which is to be computed from the data set $\sum_{t=1}^{m_c-1}\Omega(d_{it}^{c}, C_{jt}^{c})^2$ denotes the distance between data object d_i and its closest cluster center C_j in terms of categorical attributes only.

3.2. The Selection Operator

Proportional selection is used for the selection operator in which, the population of the next generation is determined by *Z* independent random experiments. Each experiment randomly selects a solution from the current population $\{S_1, S_2, \dots, S_z\}$ according to the probability distribution $\{p_1, p_2, \dots, p_z\}$ defined by[6]

$$p_z = \frac{F(Sz)}{\sum_{z=1}^{Z} F(S_z)} \quad (z = 1, \dots\dots, Z), \qquad (4)$$

F(Sz) denotes the fitness value of solution *Sz*. In our context, the objective is to minimize the V which can obtain from equation (3). Therefore, solutions with smaller *V*s should have higher probabilities for survival and should be assigned with greater fitness values. In addition, illegal strings are less desirable and should have lower probabilities for survival, and thus should be assigned with lower fitness values. We define F(Sz) as follows,

$$F(S_z) = \begin{cases} 1.5 * V_{max} - V(S_Z), if\ S_Z\ is\ leagal \\ e(S_Z) * F_{min}, otherwise \end{cases} \qquad (5)$$

Where V_{max} is the maximum *V* that has been encountered till the present generation, F_{min} is the smallest fitness value of the legal strings in the current population if they exist, otherwise F_{min} is defined as 1

3.3. The Mutation Operator

The mutation operator performs the function of shaking the algorithm out of a local optimum, and moving it towards the global optimum. During mutation, we replace a_n by a_n' for $n=1,\ldots,N$ simultaneously. a_n' is a cluster number randomly selected from {1,…,K} with the probability distribution $\{p_1,p_2,\ldots,p_K\}$ defined by

$$p_k = \frac{1.5 * d_{max}(X_n) - d(X_n, c_k) + 0.5}{\sum_{k=1}^{K} (1.5 * d_{max}(X_n) - d(X_n, c_k) + 0.5)} \qquad (6)$$

where $d(Xn,ck)$ is the Euclidean distance between pattern X_n and the centroid c_k of the kth cluster, and $d_{max}(X_n) = \max_k \{ d(X_n, c_k) \}$. If the kth cluster is empty, then $d(X_n,c_k)$ is defined as 0. The bias 0.5 is introduced to avoid divide by- zero error in the case that all patterns are equal and are assigned to the same cluster in the given solution.

3.4. The k-Means Operator

In order to speed up the convergence process, one step of the classical K-means algorithm, which we call *K-means operator (KMO)* is introduced. Given a solution that is encoded by $a1\ldots aN$, we replace an by an' for $n=1,\ldots,N$ simultaneously, where an' is the number of the cluster whose centroid is closest to Xn in Euclidean distance.

To account for illegal strings, we define $d(Xn, ck) = +\infty$ if the kth cluster is empty. This definition is different from section 3.2, in which we defined $d(Xn, ck) = 0$ if the kth cluster is empty. The motivation for this new definition here is that we want to avoid reassigning *all* patterns to empty clusters. Therefore, illegal string will remain illegal after the application of KMO.

4. EXPERIMENT AND RESULT ANALYSIS

We experiment our clustering algorithm on some standard data sets such as Iris, Vote, Heart Diseases etc. These data were taken from the UCI repository and KDD Cup data sets. To judge the quality of clustering, we assume that we are given pre-classified data and measure the ‘‘overlap’’ between an achieved clustering and the ground truth classification. We have found comparative good results. In this paper result of Heart Diseases dataset is reported.

Heart Diseases data generated at the Cleveland Clinic, is a mixed data set with eight categorical and five numeric features. It contains 303 instances belonging to two classes - normal (164) and heart patient (139). Average number of distinct attributes values for categorical attributes is taken as number of intervals (S), which is ≈3 for heart disease data set. Table 1 presents the results of clustering obtained on this data set using our algorithm. This table presents the average performance of our clustering algorithm over 100 runs. The population size is set to 50; the generation size is set to 100. The mutation probability ranges from 0.001 to 0.1. It can be seen from this table that the average number of data elements which are not in desired center, is ≈ 46. Standard deviation for error is ≈3.

Table 1: Cluster recovery for Heart disease data set with our proposed algorithm

Cluster No.	Normal	Heart Patient
1	130	29
2	34	110

The average convergence of the clustering accuracy and the objective function value over generations for two different mutation probabilities is studied. In both cases, proposed algorithm converges very fast to the extent that it will reach the global optimal clustering in five generations. The convergence of clustering accuracy and the convergence of objective function. The convergence of clustering accuracy and the convergence of objective function value are shown in Figure 1.

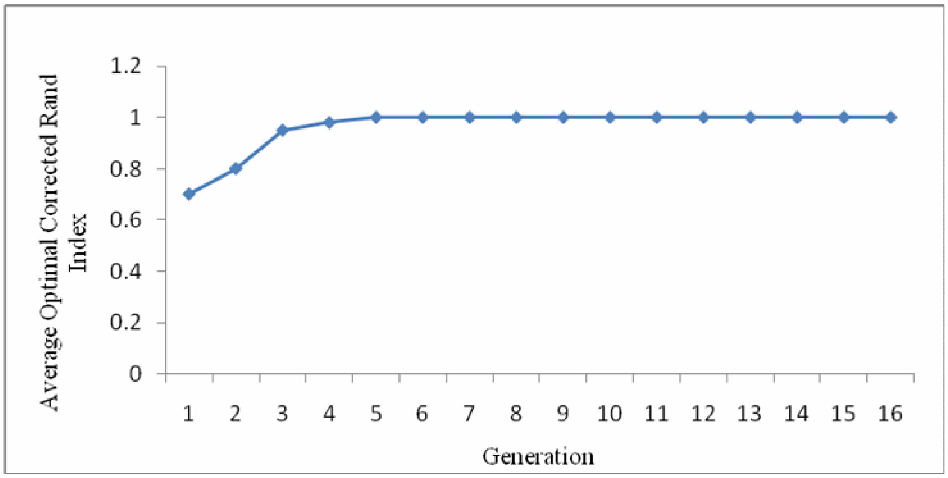

Figure 1. Average optimal corrected rand index changes

We also compare our algorithm with other algorithms such GKMODE, IGKA, and FGKA. Figure 2 shows performance comparison of above algorithms. The comparison is based on the Heart Diseases data set, the population number is set to 50 and the mutation probability is set to 0.0001.

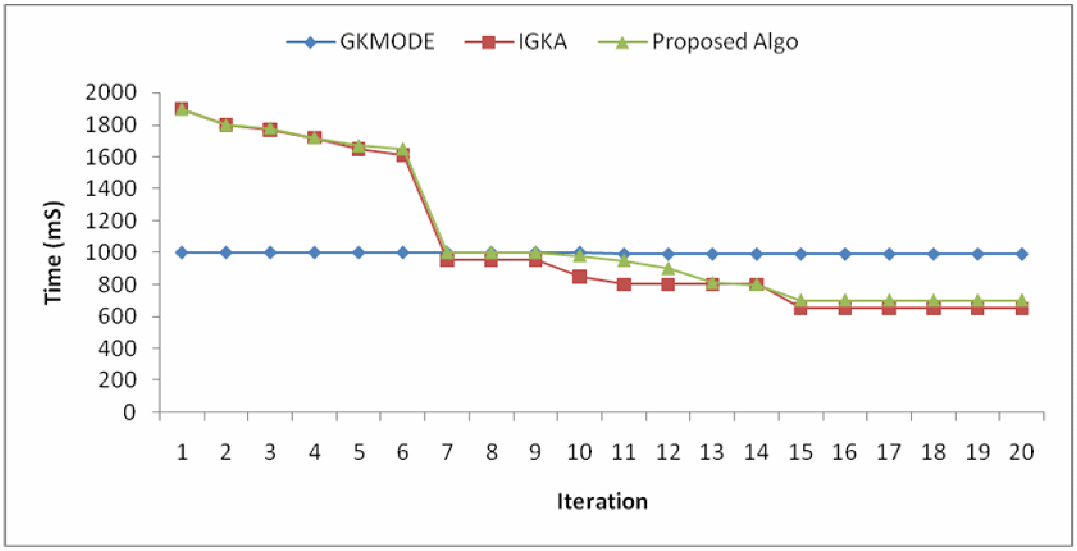

Figure 2. The performance comparison of GKMODE, IGKA and proposed based on iterations.

5. CONCLUSIONS

In real life database it contains mixed numeric and categorical type data set. In this work, we have proposed a modified genetic k-means algorithm for finding a globally optimal partition of a given mixed numeric and categorical data into a specified number of clusters. This incorporates the genetic algorithm into the k-means algorithm with enhance cost function to handle the categorical data, and our experimental results show that it is effective in recovering the underlying cluster structures from categorical data if such structures exist. Modified representation for the cluster centre is used. This representation can capture cluster characteristics very effectively because it contains the distribution of all categorical values in cluster. Also in this paper, we used additional some features such as efficient calculation of TWCVs, avoiding illegal string elimination overhead, and the simplification of the mutation operator. The initialization phase and the three operators are redefined to achieve these improvements.

ACKNOWLEDGEMENTS

The authors thank to Dr. A. Jagadeesh, Director- Rungta College of Engineering and Technology Bhilai for his encouragement. Also we would like to thank Er. Sourabh Rungta Director (Technical) – GDR Educational Society, Bhilai (CG) for their valuable support.

REFERENCES

[1] A. Ahmad and L. Dey, (2007), A k-mean clustering algorithm for mixed numeric and categorical data', Data and Knowledge Engineering Elsevier Publication, vol. 63, pp 503-527.

[2] Chaturvedi, P. Green and J. Carroll (2001), k-modes clustering. Journal of Classification, vol 18, pp. 35-55.

[3] Jain, M. Murty and P. Flynn (1999), 'Data clustering: A review', ACM Computing Survey., vol. 31, no. 3, pp. 264–323.

[4] G. Gan, Z. Yang, and J. Wu (2005), A Genetic k-Modes Algorithm for Clustering for Categorical Data, ADMA , LNAI 3584, pp. 195–202

[5] J. Z. Haung, M. K. Ng, H. Rong, Z. Li (2005) Automated variable weighting in k-mean type clustering, IEEE Transaction on PAMI 27(5).

[6] K. Krishna and M. Murty (1999), 'Genetic K-Means Algorithm'**,** IEEE Transactions on Systems, Man, and Cybernetics vol. 29, NO. 3, pp. 433-439.

[7] S. Bandyopadhyay and U. Maulik, (2001), Nonparametric genetic clustering: Comparison of validity indices', IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 31, no. 1, pp. 120–125.

[8] S. Guha, R. Rastogi, and K. Shim (2000). ROCK: A robust clustering algorithm for categorical attributes', Information System., vol. 25, no. 5, pp. 345–366.

[9] Y. Lu, S. Lu, F. Fotouhi, Y. Deng, and S. Brown (2004), 'Incremental genetic K-means algorithm and its application in gene expression data analysis', BMC Bioinformatics 5:172.

[10] Y. Lu, S. Lu, F. Fotouhi, Y. Deng, and S. Brown (2004), FGKA: A Fast Genetic K-means Clustering Algorithm', ACM 1-58113-812-1

[11] Z. He, X. Xu, & S. Deng,(2005) Scalable algorithms for clustering categorical data, Journal of Computer Science and Intelligence Systems 20, 1077-1089.

[12] A. Ahmad, L. Dey, A feature selection technique for classificatory analysis, Pattern Recognition Letters 26 (1) (2005) 43–56.

[13] M. Mahdavi and H. Abolhassani, (2009) Harmony K-means algorithm for document clustering, Data Min Knowl Disc (2009) 18:370–391.

[14] H. Yan, K. Chen, L. Liu, and Z. Yi (2010) ' SCALE: a scalable framework for efficiently clustering transactional data', Data Min Knowl Disc (2010) 20:1–27