A NOVEL EVOLUTIONARY ALGORITHM FOR DATA CLUSTERING IN N DIMENSIONAL SPACE

Roohollah Etemadi
Department of Electrical and computer engineering, Islamic Azad University Bonab Branch,
Bonab, Azarbayjan-e shargi,Iran
r.etemadi@bonabiau.ac.ir

Alireza Hajieskandar
Department of Electrical and computer engineering, Islamic Azad University Bonab Branch,
Bonab, Azarbayjan-e shargi,Iran
Hajieskandar@Gmail.com

Abstract

K-means clustering algorithm is one of the main algorithms applying in machine learning and pattern recognition. However, as the center of clusters are selected randomly and also due to the dependence of clustering result on the initial centers of clusters we may trap into local optima centers. In this paper a new genetic algorithm approach based on k-means algorithm is suggested in which the centers of clusters are selected better and in an appropriate manner. In order to increase the efficiency of this algorithm, in each stage, the layout of cluster centers which are in the form of chromosomes are changed with respect to the best chromosome. By estimation of results of the proposed approach on a standard data set and also comparison of this algorithm with other related algorithms we can show that our approach is more efficient than k-means algorithm and other algorithms which have been selected in this paper for comparison purposes.

***Keywords*:** Data mining; K-means Clustering Algorithm; Genetic Algorithm;

1. Introduction

The process of automatic finding of information among the enormous volume of saved data in data bases, data warehouses and also other data saving places is called data mining. Data mining emerged in late 1980s and in 1990s it developed significantly and we expect its progress and development in current century. One of the most important data mining techniques is clustering. It refers to the process of sample classifications in which similar samples are located in the same groups. These groups are called clusters [Han et al. 2001]. Generally, clustering algorithms and techniques have been presented based on various methods. We can classify them into 5 categories: a. partitioning methods, b. hierarchical methods, c. density based methods, d. grid based methods and e. model based methods.

In the partitioning model at first k numbers of partitions is generated from data. Every partition contains at least 1 data. (k<=n; n is the number of data). If the partitioning be hard type, then a given data could lay just in one cluster and if the partitioning be fuzzy type then a given data could lay in various clusters (with different membership grade). Two known hard type heuristic algorithms for partitioning purposes are k-means and k-methods. The equivalents of them in fuzzy type algorithms are Fuzzy k-means and Fuzzy k-methods [Han et al. 2001] & [Keller F.]. K-means algorithm is one of the most famous clustering methods. Despite of its simplicity it is considered a base for more other clustering approaches. In a very simple type of this method, at first a number of data equal to the number of necessary clusters are selected randomly. Then, we assign them into one of the available clusters based on their similarity to each other. Thus, new clusters are generated. By repetition of the work, we can compute new centers for data in each repetition through data averaging. This procedure will be continued until we see no change in data. Despite of this fact that we can ensure this algorithm will be ended but the final solution is not a unique value and depends on the initial centers of clusters and is not necessarily always an optimum solution. [Sanghamitra et al. 2002]. Various methods have been suggested in order to solve the problems of k-means algorithm including: Genetic Algorithm[Sanghamitra et al. 2002],[Dong-Xia Chang et al. 2009], particle Swarm optimization, Ant Colony optimization, simulated annealing algorithm, Tabu research and combined approaches[Yi-Tung et al. 2008].

The rest of this paper is structured as follow. In section 2 and 3, we present the concepts of clustering and k-means algorithm respectively. In section 4, we propose our approach. The results of applying this proposed approach is reported in section 5. Section 6 covers conclusion and decision issues.

2. Clustering Concept

The process of grouping a collection of physical or abstract objects into similar categories groups is called clustering. The objects in a cluster are more different and are different from the objects of other group [Han et al. 2001], [Keller F.].

Definition: Consider the set x={x1,x2,…,xn} including n object, the aim of clustering is grouping the objects in k cluster as c={c1,c2,… ck} such that every cluster is as following:

$$1) C_1 \cup C_2 \cup \ldots \cup C_k = X$$
$$2) C_i \neq \varphi \quad i = 1, 2, \ldots, k$$
$$3) C_i \cap C_j = \phi$$

Regarding the above definition, the various states for clustering n object to k cluster equals:

$$NW(n,k) = \frac{1}{k!} \sum_{i=0}^{k} (-1)^i \binom{k}{i} (k-i)^n \qquad (1)$$

In most of methods, the rate of clusters that is K is determined by the user. From equation (1) it is understood that even if K is obvious, finding the best state of clustering is not easy. In addition , the methods of clustering n object to k cluster increases as $K^n / k!$,so finding the best state for clustering n object to k cluster is considered a NP-Complete and complicate problem and should be solved optimally by some techniques [Liu G.L. 1968],[Hruschka E.R. et al. 2003].

3. K-means Clustering Algorithm

Many algorithms have been suggested for solving clustering problems. K-means is one of the most famous of them. The main stages of k-means algorithm for data clustering purposes are as follow [Sanghamitra et al. 2002]:

Step 1: Choose K cluster centers $M = (m_1, m_2, \ldots, m_k)$ randomly from n points $X = \{x_1, x_2, \ldots, x_n\}$.

Step 2: Assign point $x_i \in X$ to cluster $c_j \in C = \{c_1, c_2, \ldots, c_k\}$ if

$$\|x_i - m_j\| < \|x_i - m_p\| \quad 1 \le p \le k,\ j \ne p \qquad (2)$$

Step 3: Compute new cluster centers $M^* = (m_1^*, m_2^*, \ldots, m_k^*)$ as follows:

$$m_i^* = \frac{1}{n_i} \sum_{x_j \in C_i} x_j \qquad 1 \le i \le k \qquad (3)$$

Where n_i is the number of points belonging to cluster c_i.

Step 4: If termination criteria satisfied, stop otherwise continues from step 2. The K-means clustering process terminates when any one of the following criteria is satisfied: when the maximum number of iterations has been exceeded, when there is little change in the centroid vectors over a number of iterations, or when there are no cluster membership changes.

As we can see in the proposed algorithm, the initial centers for clusters are selected randomly. It causes this algorithm to deliver different solutions for given data objects in various runs of it which is considered as one of the disadvantages of this algorithm. [Sanghamitra et al. 2002].

4. Proposed Algorithm

In most genetic algorithms which are employed for clustering purposes, it is the duty of chromosomes to preserve cluster centers. [Sanghamitra et al. 2002] At first, cluster centers within chromosomes are selected among data objects randomly. It is our idea to cluster data objects at this first stage through k-means algorithm in terms of their individual attributes. The numbers of clusters which are generated in this stage are more or equal to the numbers of the main clusters of input data. Next, the cluster centers within each chromosome are selected among these clusters no repeatedly. Then, for every run of the proposed algorithm the best chromosome is selected and considered as the basic chromosome and the structure of other chromosomes are changed with respect to this base chromosome. Algorithm 1 shows the basic genetic algorithm for clustering.

Algorithm1. pseudo-code of the Proposed Genetic K-Means Clustering Algorithm

1. **Input:** Data SET ($X = \{x_1, x_2, ..., x_n\}$) ,Attribute Number,Cluster Number(K),
2. **Output:** Clusters Set($C = \{c_1, c_2, ..., c_k\}$)
3. **Begin**
4. Find_Seed_Cluster_Center( );/* see section 4-1 */
5. Iitialize_Population( );
6. While(r < Reapet)
   - 6.1. Fitness( ); /*see section 4 -3 */
   - 6.2. Rearrangment_choromosome();/*see section 4-4*/
   - 6.3. Crossover( ); /*see section 4 -5 */
   - 6.4. Mutation( ); /*see section 4 -6 */
7. Rturn Clusters
8. **End**

4.1. *Algorithm for Finding Seed Centers for Clusters*

Our proposed method for finding the initial cluster centers in the initial stages of assigning value to chromosomes are as follow. At first we cluster all data objects through k-means algorithm in terms of their individual attributes. Then regarding to the generated clusters, a pattern is generated for each data object based on each attribute in each stage. Objects with common attributes are located within the same cluster. By this approach all objects will be clustered. The number of generated clusters in this stage will be more than that of main clusters. This approach is very similar to the proposed approach in [Shehroz et al. 2004] where the object clustering is done in two stages. First stage is similar to the above mentioned method. In second stage, similar clusters are merged together until we obtain specific numbers of clusters. Algorithm 2 shows the proposed approach for pre clustering of data objects. The obtained centers for these clusters are called seed centers.

As we can see in algorithm 2, for each attribute of a given data object, an appropriate ticket is generated for that cluster. This ticket is added to the pattern of that data object. Objects with similar patterns are located in the same cluster. In order to generate tickets for each data in terms of all attributes, at first we should compute the mean and variance of a given attribute for all data. Then, the values of that given attribute are divided into k equal reaches based on the obtained mean and variance. The end of each reach will be considered as the initial centers of clusters. Now based on these initial centers all data are clustered through k-means algorithm.

Algorithm2. pseudo-code of the Proposed Find_Seed_ Cluster_ Center Algorithm

1. Input: Data SET ($X = \{x_1, x_2, ..., x_n\}$) ,Attribute Set($A = \{A_1, A_2, ..., A_N\}$),Cluster Number(K),
2. Output: Clusters Seed Set ($SC = \{sc_1, sc_2, ..., sc_H\}, H \geq K$)
3. Begin
4. while ($\forall A_j \in A$)

   Compute Mean(μ_j) and Standard Deviation(σ_j)

   Compute Cluster Center($e = 1,2,...,k$) $X_e = Z_e * \sigma_j + \mu_j$, $Z_e = \frac{2*e-1}{2*k}$

   Execute K-means on this attribute

   Allocate cluster labels obtained from step 4.3 to every data pattern
5. Find unique patterns ($H \geq k$) and clustering each data whit obtained patterns.
6. Return SC
7. End

4.2. *Chromosome Structure*

We use value encoding method to encoding the chromosomes. If k be the number of clusters and N be the number of data objects attributes the length of chromosome would be N*K and is defined as follow:

$M = [m_{11}, m_{12}, ...m_{1N}, m_{21}, m_{22}, ..., m_{2N}, ..., m_{k1}, m_{k2}, ..., m_{kN}]$

According to above structure, we call $m_i = [m_{i1}, m_{i2}, ..., m_{iN}]$ the center of ith cluster. In the start point of this proposed algorithm a specific number of chromosomes with the above mentioned structure are generated. Cluster centers within chromosomes are selected randomly and no repeatedly among the seed centers obtained from the mentioned algorithm of section 4.1 (Algorithm 2).

4.3. *Fitness Function*

In order to define standards and criterion for assessment of chromosome fitness, at first through equation 2, we assign objects to clusters with respect to cluster centers within chromosome. Then according to this new clustering method, new centers for clusters are generated by equation 3. These new centers are presented in the form of chromosome. Based on these new cluster centers, we can calculate the fitness of chromosome through equation 4.

$$Fitness(C) = \sum_{i=1}^{k} \sum_{x_j \in C_i} \left\| x_j - m_i^* \right\| \qquad (4)$$

4.4. *Rearrangement Chromosome's Structure*

In order to increase the efficiency of this algorithm in each stage the best chromosome is selected and the layout of the cluster centers of other chromosomes are varied with respect to this selected chromosome. This Rearrangement prevents the generation of repeated centers and can enhance the efficiency of the combination operator for obtaining new chromosomes purposes.

Suppose chromosome X and Y with follow structures:

$X = [x_{11}, x_{12}, ..., x_{1N}, x_{21}, x_{22}, ..., x_{2N}, ..., x_{k1}, x_{k2}, ..., x_{kN}]$

$= [x_1, x_2, ..., x_k]$

$Y = [y_{11}, y_{12}, ..., y_{1N}, y_{21}, y_{22}, ..., y_{2N}, ..., y_{k1}, y_{k2}, ..., y_{kN}]$

$= [y_1, y_2, ..., y_k]$

According to above structure we can write the center of ith cluster in those two clusters as follow:

$y_i = [y_{i1}, y_{i2}, ..., y_{iN}]$ $x_i = [x_{i1}, x_{i2}, ..., x_{iN}]$

If we suppose the X chromosome as the reference one, then we can vary the structure of the Y chromosome through algorithm 3 so that the neighborhood centers locate in the same gene within two chromosomes.

Algorithm3. pseudo-code of the Proposed Rearrangement chromosome Algorithm

1. **Input**: Reference chromosome X, chromosome Y
2. **Output**: Y' rearrangement chromosome of Y
3. **Begin**
4. F $\leftarrow$ {1,2,…,k}
5. R $\leftarrow \phi$
6. For i=1 to k do
   6.1. h= arg $\min\limits_{j \in F, j \notin R} \| y_i - x_j \|$
   6.2. F $\leftarrow$ F-{h}
   6.3. R(i) $\leftarrow$ h
7. For i=1 to k do
   7.1. $y_i' \leftarrow y_{R(i)}$
8. Return Y'
9. **End**

4.5. *Combination /Crossover Operator*

In order to combine chromosomes with the constant probability of p_c we choose single point crossover method. At first a random point such as z is generated in the range of [1,k]. Then, the two selected chromosomes are cut competitively from the point z*N (N=the number of futures) and the right hand side of chromosomes are replaced with each other.

4.6. *Mutation operator*

Mutation operator with constant probability of P_M on chromosomes is employed for searching intact spaces of solutions. This operator at first selects a random gene from a random chromosome as m_{ir} ,$1 \le i \le k, 1 \le r \le N$. In order to compute the new value of the gene m_{ir} at first we generate a random value of β from the range of $[-1,1]$. The value of the gene m_{ir} is calculated by equation 5.

$$m_{ir} = \begin{cases} m_{ir} + \beta \times (m_{\max}^{ir} - m_{ir}) & \beta > 0 \\ m_{ir} + \beta \times (m_{ir} - m_{\min}^{ir}) & \beta < 0 \end{cases} \qquad (5)$$

In the equation 5, m_{ir} is the value of the ith gene in the selected chromosome and $m_{\max}^{ir}, m_{\min}^{ir}$ are the minimum and maximum values of ith genes of all chromosomes.

5. Experimental results

The proposed algorithm was encoded by C#.net programming language and was run on a Pentium 4 PC with a 3.08 GHz processor and memory of 512 MB. In order to estimate the efficiency of the proposed algorithm we used the standard values indicated in table 1. The results of applying this algorithm on the selected test data set is reported in table 2 of reference [Yi-Tung et al. 2008]. Also this table compares the results of this algorithm with K-means, PSO and K-NM-PSO algorithms.

Table 1. Characteristics of the data sets considered

Name of data set	No. Of attribute	No. of Cluster	**Size of data set (size of clusters in parentheses)**
Iris	4	3	**150(50,50,50)**
Wine	13	3	**178(59,71,48)**
CMC	9	3	**1473 (629, 334, 510)**
Glass	9	6	**214 (70, 17, 76, 13, 9, 29)**

As we can see in the table 2, our proposed algorithm presents better results than k-means and PSO algorithms. In order to assess the efficiency of this algorithm in detail we show in tables 3 to 6 the results of the proposed algorithm and also the results of following algorithms which have been presented in reference [Taher Niknam et al. 2010]; PSO-ACO-K, PSO-ACO, PSO, SA, TS, ACO, GA, HBMO, PSO-SA, ACO-SA, K-means.

Table 2. Comparison of intra-cluster distances for the three clustering algorithms whit proposed algorithm.

Data Set	PSO[Yi-Tung et al. 2008]	K-NM-PSO[Yi-Tung et al. 2008]	**Proposed Alg.**	
			Result	**CPU time(S)**
Iris	96.66	96.66	97.222	**0.018**
Wine	16294.00	16292.00	16530.53	**0.016**
CMC	5538.50	5532.40	5541.64	**0.212**
Glass	271.29	199.68	226.032	**0.095**

It should be noted that the algorithms in reference [Taher Niknam et al. 2010] have been encoded by Matlab 7.1 programming and run on a Pentium IV PC with a 2.8 GHz processor and 512MB memory. The obtained results have been reported in the reference [Taher Niknam et al. 2010]. Tables 3 to 6 present the worse, best and mean solutions in 100 repetitions of the algorithm. Indeed the obtained values indicate the total distances of each data object from the center of that cluster into which the data object has been located. These values are calculated through equation 4. As we can see in tables, in all examples the proposed algorithm delivers acceptable solutions in more appropriate times.

Table 3. Results obtained by the algorithms for 100 different runs on Iris

Result	Method											
	PSO–ACO–K	PSO–ACO	PSO	SA	TS	GA	ACO	HBMO	PSO–SA	ACO–SA	k-Means	Proposed ALG.
Best	96.650	96.654	96.8942	97.45	97.365	113.98	97.100	96.752	96.66	96.660	97.333	97.222

Table 4. Results obtained by the algorithms for 100 different runs on Wine

Result	Method											
	PSO–ACO–K	PSO–ACO	PSO	SA	TS	GA	ACO	HBMO	PSO–SA	ACO–SA	k-Means	Proposed ALG.
Best	16,295.31	16,295.34	16,345.96	16,473.48	16,666.22	16,530.53	16,530.53	16,357.28	16,295.86	16,298.62	16,555.68	16,530.537

Table 5. Results obtained by the algorithms for 100 different runs on CMC

Result	Method											
	PSO–ACO–K	PSO–ACO	PSO	SA	TS	GA	ACO	HBMO	PSO–SA	ACO–SA	k-Means	Proposed ALG.
Best	5,694.28	5,694.51	5,700.98	5,849.03	5,885.06	5,705.63	5,701.92	5,699.26	5,696.05	5,696.60	5,842.20	5,541.64

Table 6. Results obtained by the algorithms for 100 different runs on Glass

Result	Method											
	PSO–ACO–K	PSO–ACO	PSO	SA	TS	GA	ACO	HBMO	PSO–SA	ACO–SA	k-Means	Proposed ALG.
Best	199.53	199.57	270.57	275.16	279.87	278.37	269.72	245.73	200.14	200.71	215.74	226.032
Average	199.53	199.6	275.71	282.1	283.79	282.32	273.46	247.71	201.45	201.89	235.5	227.514

According to the results of tables 2 to 6, we can find that the proposed algorithm presents the best solution on CMC data sample, with 1473 data, comparing with other algorithm. May be the reason is this fact that by increasing the number of data within sample data set, other algorithms lose their efficiency and in contrast the proposed algorithm shows its efficiency.

If we look at the results of tables 2 to 6, we can see that the proposed algorithm presents an acceptable solution in a very shorter time than other algorithms, less than 1 second. This again proves the efficiency of this algorithm comparing with others.

6. Conclusion

In this paper we presented a new genetic algorithm for clustering of data objects based on k-means algorithm. We can solve the main weaknesses of k-means algorithm to some extent by generating chromosomes through seed centers. By changing the structure of chromosomes with respect to the best obtained chromosome in each stage, we can improve the performance of the combination/crossover operator. The obtained results from applying the proposed algorithm on a set of test data show higher efficiency of this algorithm comparing k-means. According to the obtained solutions we can find that the proposed algorithm show higher efficiency in high volume of data comparing with other algorithms presented in this paper. Although this algorithm selects cluster centers in chromosome form randomly, this selection however, is done among seed centers and the number of seed centers is very less than the number of data. As a future study we can define a specific standard and criteria by which among the obtained seed centers we try to choose those centers which can improve the proposed approach.

References

[1] Han J.; Kamber M. (2001), *Data Mining: Concepts and Techniques*, San Francisco: Morgan Kaufmann.
[2] Keller F.; *Clustering*, Computer University Saarlandes, Tutorial Slides.
[3] Liu G.L. (1968), *Introduction to Combinatorial Mathematics*, McGraw-Hill.
[4] Hruschka E.R.; N.F.F. Ebecken(2003), *A genetic algorithm for cluster analysis*, Intelligent Data Analysis 7(1) 15–25.
[5] Sanghamitra Bandyopadhyay, Ujjwal Maulik(2002), *An evolutionary technique based on K-Means algorithm for optimal clustering in* R^N, Information Sciences 146, 221–237.
[6] Yi-Tung Kao, Erwie Zahara, I-Wei Kao(2008), *A hybridized approach to data clustering*, Expert Systems with Applications 34 , 1754–1762.
[7] Taher Niknam, Babak Amiri(2010), *An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis*, Applied Soft Computing 10, 183–197.
[8] Shehroz S. Khan, Amir Ahmad(2004), *Cluster center initialization algorithm for K-means clustering*, Pattern Recognition Letters 25, 1293–1302.
[9] Dong-Xia Chang, Xian-Da Zhang, Chang-Wen Zheng(2009), *A genetic algorithm with gene rearrangement for K-means clustering*, Pattern Recognition 42, 1210 – 1222.