Genetic fuzzy clustering

4 authors, including:

James C. Bezdek

University of Missouri

428 PUBLICATIONS   **64,457** CITATIONS

Genetic Fuzzy Clustering

L.O. Hall[1], J.C. Bezdek[2], S. Boggavarpu[1], and A. Bensaid[1]

[1]Department of Computer Science and Engineering
University of South Florida
Tampa, Fl. 33620

[2]Department of Computer Science
University of West Florida
Pensacola, FL. 32514

Abstract

This paper describes a genetic guided fuzzy clustering algorithm. The fuzzy-c-means functional J_m is used as the fitness function. In two domains the approach is shown to avoid some higher values of J_m to which the fuzzy c-means algorithm will converge under some initializations. Hence, the genetic guided approach shows promise as a clustering tool.

1 Introduction

Unsupervised clustering is useful in exploratory data analysis, image segmentation and with some added class knowledge may be used for classification. In this paper we present a genetic algorithm (GA) approach to clustering with fuzzy c-means (FCM). Clustering algorithms that use calculus-based methods can have the problem of being trapped at local extrema in the process of optimizing the clustering criterion. Genetic algorithms can avoid local minima because they are based on (constrained) random search.

The GA guided clustering algorithm presented here attempts to minimize the same function as the FCM clustering algorithm [1]. The method of minimization is genetically guided rather than by an iterative steepest descent type procedure. Avoidance of local minima of the FCM functional is an important goal in the development of our approach.

An evolution-based clustering strategy has been reported in [2] and two artificial data sets from that paper are used here. The approach to crossover and mutation used in [2] is quite different from ours and the results discussed concern only initializing FCM with no coverage of multiple random initial populations. Our approach is distinct from that in [3], where partitions of data are represented as binary strings.

2 Clustering with FCM

Consider a set of n objects $X = \{x_1, x_2, \ldots x_n\}$ to be clustered into groups of like objects. Each $x_i \in R^s$ is a feature vector consisting of s real-valued measurements describing the features of the object represented by x_i. The features could be length, width, color etc. Particular clusterings of the objects can be represented by a fuzzy membership matrix, which is called a fuzzy partition. The set of all $c \times n$ non-degenerate constrained fuzzy partition matrices is denoted by M_{fcn} and is defined as

$$M_{fcn} = \{U \in R^{c \times n} | \sum_{i=1}^{c} U_{ik} = 1, 0 < \sum_{k=1}^{n} U_{ik} < n, \; and$$
$$U_{ik} \in [0,1] \, ; 1 \le i \le c; 1 \le k \le n\}. \quad (1)$$

Given a method for creating fuzzy clusters, the problem is to find the corresponding *best* partition matrix in M_{fcn}.

The clustering criterion considered here is the function:

$$J_m(U, V) = \sum_{i=1}^{c} \sum_{k=1}^{n} (U_{ik})^m D_{ik}^2(v_i, x_k) \quad (2)$$

where $U \in M_{fcn}$ is a fuzzy partition matrix; $m \in [1, \infty)$ is the weighting exponent on each fuzzy membership; $V = [v_1, \ldots, v_c]$ is the matrix of prototype parameters; and $D_{ik}(v_i, x_k)$ is a measure of the distance from x_k to the ith cluster prototype. The *Euclidean distance* metric is used in all work reported here. A search is undertaken for a good representation of the cluster structure in X based upon a (U,V) minimizer of (2). Typically, the optimal (U,V) pair is sought using an alternating optimization scheme of the type generally described in [4].

3 Genetic guided clustering

Element i of the population is V_i, the $c \times s$ matrix of cluster centers in FCM notation. The initial population is constructed by random assignment of real numbers to each feature of the c cluster centers of the initial population of size P. The initial values are chosen to be in the range (determined from the data set) of the feature to which they are assigned.

In order to work only with V's, equation (2) can be rewritten by substitution for U of the necessary conditions:

$$U_{ik} = 1/\sum_{j=1}^{c} \left(\frac{D_{ik}(\mathbf{v}_i, \mathbf{x}_k)}{D_{jk}(\mathbf{v}_j, \mathbf{x}_k)} \right)^{2/(m-1)} \quad for \ 1 \leq i \leq c;$$

$$1 \leq k \leq n. \quad (3)$$

The genetic guided clustering algorithm works in the following way. Given an initial population of V's of size P, a new generation is created by selecting some of the members of the population for reproduction. Reproduction is accomplished by the use of the crossover and mutatiɔn operators. Selection, crossover and mutation are combined to create the next generation of P new V's. This means selection is performed exactly P times with crossover occurring $P/2$ times since each crossover results in 2 children.

3.1 Selection

A roulette wheel selection mechanism is used for selecting population members to reproduce [5]. Each member of the population gets a percentage of the roulette wheel based on its fitness. The technique selects with high probability a member which has a maximum function value. Since J_m is to be minimized, the construction of the roulette wheel is actually done with the reciprocals of J_m values. In order to prevent a V with a very low J_m from having a prohibitively high probability (large percentage of the wheel) of repeat selection, a linear scaling [5] is applied to the raw reciprocal J_m values.

Under this selection paradigm, two population members are chosen for reproduction by "spinning" the roulette wheel twice. Crossover and mutation are then performed using the two chosen members.

3.2 Crossover

Each feature v_{ij} of a cluster center is a real number. In order to do crossover the feature value is converted into an integer which captures three decimal places of precision. The integer is then represented as a binary bit string. For example, if the largest value for a feature is 7.999 then it will be converted to 7999 which may be represented by a 13 bit binary number. The binary representation of a V matrix with c clusters and s features each represented by 13 bits requires $c \times s \times 13$ bits.

Figure 1 shows the crossover operation for 2 chosen parents (P_i and P_j of a population, where c=3, s=2 and each feature is represented by 3 bits). The bits of each feaʰure of a cluster center are separated by a thick line. For each cluster center of a parent, a crossover point and block of bits are randomly selected such that the number of bits to the left and right of the crossover point is the same. This block is swapped with the corresponding block of the other parent. The bit blocks shown in Figure 1 have by dashed lines have widths of 3, 3, and 1, respectively for the three centers. This crossover operation is done on each of the c cluster centers of the mating parents and produces two children (one of which is shown in Figure 1) for the next generation.

The reason that c crossovers are done is to exploit the fact that each cluster center is an individual entity and might need to be adjusted during each crossover. A single crossover during reproduction has been tried with less success and is not reported here due to space limitations.

3.3 Mutation

After every crossover each bit of the children is considered for mutation with a mutation probability pm. Mutation consists of the process of flipping the value of the chosen bit from 1 to 0 or vice versa. The mutation rate is set so that approximately one bit in every child cluster will be transformed.

3.4 Elite algorithm vs. non-elite algorithm

The roulette wheel selection process may result in a population at generation i+1 that does not contain the best member (lowest J_m) from the population at generation i. This, coupled with an ineffective set of crossovers, can cause the reduction in J_m to be non-monotonic. To remedy this problem, we use an "elite" algorithm in which the best 2 members of the population at generation i are passed through to the population at generation i+1. The remaining (p-2) members of the new generation are created through the usual process of selection, crossover and mutation as described above. All members of a population are included on the roulette wheel.

4 Data set descriptions

Four different domains are used to generate the results reported in the next section. Three of the domains are artificial, but useful for this study because they each have multiple local minima for J_m.

Figures 2a and 2b show the two-dimensional touching clusters (25 points) and symmetrical data (20 points) [2]. The one-dimensional data set consists of one feature, the output y of the non-linear equation $y = (1 + x_1^{-2} + x_2^{-1.5})^2$, where $1 \leq x_1, x_2 \leq 5$. The equation is taken from [6], where 6 classes were found to provide a useful grouping in the development of fuzzy control rules to model the output of the equation over a set of 50 distinct values. The same 50 y values were used in this study.
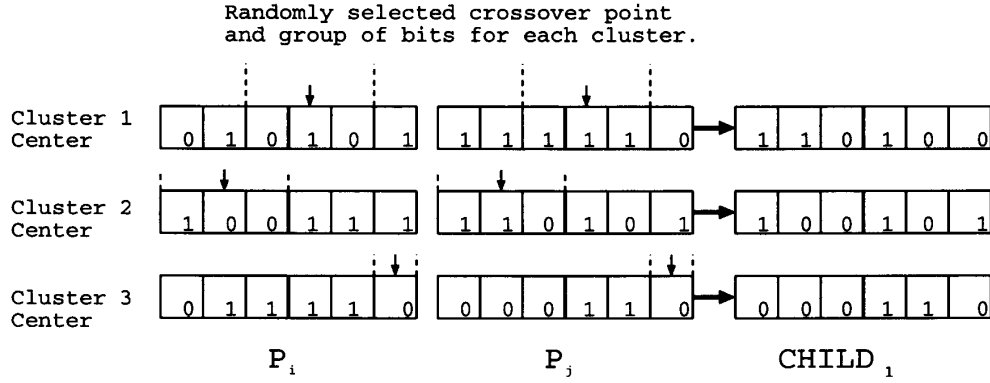
Randomly selected crossover point
and group of bits for each cluster.

Figure 1: The crossover operation for parents with 3 cluster centers each of 2 features. The child created by crossing P_i's bits into P_j is shown.
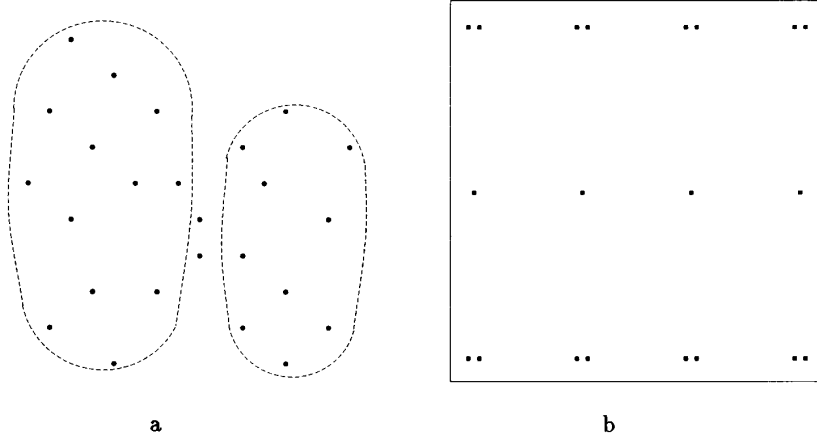

Figure 2: a) Touching Clusters data with artificial boundaries and b) Symmetric data

The multiple sclerosis (MS) data set [7] consists of 2 classes each described by 5 measurements (features). The classes are MS (29 examples) and non-MS (69 examples). The first feature is age, and measurements connected with two different visual stimuli (S1 and S2) provide the other 4 features. The features are the sum and differences of the responses of the stimuli as observed in the left and right eye, respectively. Letting L stand for left eye response and R stand for right eye response, these features are (S1L + S2L), ($|S1L - S2L|$), (S2L + S2R), and ($|S2L - S2R|$).

5 Results

In this section we report results using the Elite and non-Elite GA algorithms to optimize J_m. For each of the three artificial domains FCM might converge to different local minima under different random initializations, as shown in Table 1. The same values for m and c, shown in Table 1, are also used in the GA algorithm for the respective data sets. The one "real world" data set on Multiple Sclerosis (MS) diagnosis exhibited just 1 local minimum for the 1800 different initializations of FCM that were used. The elite method provides a monotonic decrease in J_m as shown in Figure 3 a for the MS data. Figure 3b shows that the non-elite method does not, in general, provide a monotonic decrease in J_m.

All GA guided results are averaged over 30 runs with the standard deviation reported. The GA is run for 500 generations in every case. Each of the 30 runs used a different random seed to initialize the population. The mutation rate, population size (P), m value for J_m and average J_m over the 30 different seeds and standard de-

413

DATA SET	m	c	Minimum A $J_m/\#$ FCM runs	Minimum B $J_m/\#$ FCM runs	Minimum C $J_m/\#$ FCM runs
Single Feat.	2	6	.731/93	.893/181	1.05/146
Touching Clus	2	2	215.441/593	332.320/7	
Symmetrical	3	3	3.167/385	5.432/35	
Multiple Sclerosis	2	2	65766.883/1800		

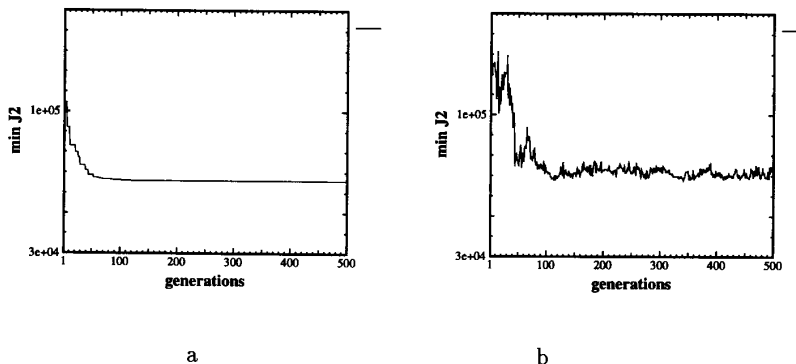Table 1: Different minima for 4 data sets under FCM clustering.

Figure 3: J_m for the a) elite and b) non-elite methods using the MS data.

viation in J_m values are reported in Table 2 for the elite GA guided clustering method. In Table 3, results for the non-elite GA guided clustering method are reported.

One major observation can be made by examining Tables 2 and 3. The elite method outperforms the non-elite method on every data set. J_m and the standard deviation are both lower. Part of the disparity probably is the result of the mutation rate being set to cause, on average, one mutation per crossover. We believe this rate is high enough to cause the non-elite method to mutate away from desirable local minima in many cases.

It is important to compare the best GA guided results to the results that would be obtained using fuzzy c-means. Table 4 shows the results of running FCM using each initial V that was used in a GA population for the respective data sets. This means, for example, that for a population of size 20 with 30 different random seeds, such as the touching clusters data set, $20 \times 30 = 600$ runs of FCM were performed each with a different initialization. In comparing Table 2 to Table 4, it is clear that the GA guided clustering outperforms FCM on average for the 3 artificial data sets. The average performance is not as good for the MS data set. This is probably due to the inability of the GA guided clustering algorithm to make small focussed refinements to cluster centers. Another possible contributing factor is that s=1 or 2 for the three artificial data sets and s=5 for the MS data. Thus the error surface for the MS data is much more complex.

DATA SET	Mut. Rate	P	J_m	std_dev
Touching Clus	0.033	20	215.46	0.03
Symmetrical	0.038	14	3.172	0.01
Multiple Sch.	0.011	60	66126.59	253.54
Single Feat.	0.077	14	0.777	0.04

Table 2: GA guided clustering: Elite method of passing the best 2 population members on to the next generation.

An experiment was run to determine if the best final V from the GA guided elite clustering method when used as an initialization to FCM would enable FCM to converge to the lowest known minimum (where lowest known minimum means the lowest of which we are aware). Initializations were only used from cases in which the GA had not ended up approximately at the best known minimum. FCM *did* converge to the lowest known minimum for all data sets when provided with such an initialization with one exception. The one exception was with the symmetrical data set where the GA initialization enabled FCM to converge to the second best minima shown in Table 1.

6 Summary

For good performance it is necessary to pass some subset of the best members of a population through to the next

414

DATA SET	Mut. Rate	P	J_m	std_dev
Touching Clus	0.033	20	234.22	13.55
Symmetrical	0.038	14	6.027	1.11
Multiple Sch.	0.011	60	74536.03	3798.65
Single Feat.	0.077	14	2.692	0.77

Table 3: GA guided clustering: Non-Elite method.

DATA SET	J_m	std_dev
Touching Clus	216.805	12.55
Symmetrical	3.355	0.63
Multiple Sch.	65766.883	0.76
Single Feat.	0.919	0.13

Table 4: FCM results averaged over all initial V's provided to the GA guided method for each data set.

generation in our GA guided clustering method. Passing the two best members of the population, as done here in the elite method, provides a monotonic decrease in J_m and enables a significantly lower final J_m to be obtained in the general case. The best V's get to participate in reproduction in subsequent generations; this is also likely to contribute to better performance. The results show that the GA guided clustering algorithm described here is effective in bypassing local minima. The GA guided clustering elite method is able to reach the minimum known value for J_m in many cases as shown in Table 5, where the percentage of times the J_m for the best V is within 1-10% of the lowest J_m found with FCM is reported. FCM is generally better able to find the minimal J_m once a specific local minimum has been "settled into". Our GA approach does not appear to have the ability to make focussed small refinements to the cluster centers to provide a final minimization of J_m equivalent to that obtained by FCM.

The GA approach appears to be an effective method to initialize FCM by using the best V from the final population as the initial V of FCM. Such an initialization will generally enable FCM to avoid local minima into which it might otherwise become trapped. Improvements and refinements (discussed below) in the GA guided clustering

DATA SET	x	Percentage of times within x% of FCM minimum over 30 trials.
Touching Clus	1	100
Symmetrical	1	100
Multiple Sch.	1	43
Single Feat.	10	70

Table 5: Comparison of GA final J_m with FCM final J_m.

algorithm may make it even more effective for the initialization task.

Our work leaves a number of questions unanswered. Little experimentation has been done to find optimal mutation rates or to start with a low mutation rate and increase it as the generations increase, to help prevent premature convergence. Ideally, the smallest possible population for the GA which performs well on the FCM initialization task would be used. Experiments with smaller populations have not been done. Since passing the top 2 members of a population on to the next generation is a profitable strategy, it is possible that passing more members onward will be even more effective. The GA guided clustering method is very sensitive to the initial random seed that is used to generate an initial population. A possible remedy under consideration is to generate a large initial population and have most of the population starve before the second generation. The strong members (low J_m's) will be most likely to survive. It is expected that answers to the questions raised here and further improvements to the GA guided clustering approach will better enable excellent initialization of FCM and possibly a clustering technique independent of FCM.

References

[1] Bezdek, J.C. (1981), Pattern Recognition with Fuzzy Objective Functions, Plenum, N.Y.

[2] Babu, G.P. and Murty, M.N. (1994), Clustering with Evolution Strategies, *Pattern Recognition*, V. 27, No. 2, pp. 321-329.

[3] Bhuyan, J.N., Raghavan, V.V. and Elayavalli, V.K. (1991), Genetic Algorithm for Clustering with an Ordered Representation, International Conference on Genetic Algorithms'91, pp. 408-415.

[4] Bezdek, J.C. (1974), Numerical Taxonomy with Fuzzy Sets, Journal of Mathematical Biology, 1, 57-71.

[5] Goldberg, D. (1989), Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley.

[6] Sugeno, M. and Yasukawa, T. (1993), A Fuzzy Logic Based Approach to Qualitative Modeling, *IEEE Transactions on Fuzzy Systems*, V. 1, No. 1, pp. 7-31.

[7] Johnson, R.A. and Wichern, D.W. (1992), Applied Multivariate Statistical Analysis 3rd edition, Prentice Hall, Englewood Cliffs, N.J.