Journal of
Computational and Theoretical Nanoscience
Vol. 2, 1–10, 2005

Evolutionary Fuzzy Clustering Algorithm with Knowledge-Based Evaluation and Applications for Gene Expression Profiling

Han-Saem Park,* Si-Ho Yoo, and Sung-Bae Cho

Department of Computer Science, Yonsei University, 134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea

In microarray data analysis, clustering is a method that groups thousands of genes by their similarities of expression levels, helping to analyze gene expression profiles. This method has been used for identifying unknown functions of genes. The fuzzy clustering method assigns one sample to multiple groups according to their degrees of membership. This method is more appropriate for analyzing gene expression profiles, because a single gene might be involved in multiple functions. General clustering methods, however, have problems in that they are sensitive to initialization and can be trapped into local optima. To overcome these problems, we propose an evolutionary fuzzy clustering method with knowledge-based evaluation. The proposed method uses a genetic algorithm for clustering and prior knowledge of experimental data for evaluation. We have performed experiments to show the usefulness of the proposed method with yeast cell-cycle and SRBCT datasets.

Keywords: Evolutionary Fuzzy Clustering, Knowledge-Based Evaluation, Gene Expression Profiles, Microarray Data Analysis.

RESEARCH ARTICLE

1. INTRODUCTION

Nanotechnology is the creation of functional materials, devices and systems through control of matter on the nanometer length scale, and DNA microarray technology is one of the related technologies.[1] A great amount of gene-level information can be obtained by a single experiment with this technology.

Clustering is a method that groups thousands of genes by their similarities of expression levels, helping to analyze gene expression profiles. This method has been used for identifying functionally related families of genes.[2] It is often difficult to group the data in the real world clearly, since often there are no clear boundaries of clusters.[3] Clustering genes which contain multiple functions and belong to multiple clusters is a representative example. Since fuzzy clustering method assigns one sample to multiple clusters according to their degrees of membership, it is more appropriate for analyzing gene expression profiles.[4]

General clustering algorithms have common problems in that they are very sensitive to initial values and they can be trapped by local optima because their processes are supposed to minimize an objective function.[5,6] Besides, it is difficult to analyze the data correctly without previous knowledge since the number of clusters often needs to be fixed before analyses are performed. It takes much time and cost to cluster the data, if there is no prior information of the number of clusters. There is also a problem of validating cluster results. Because gene expression profiles are variable depending on experiments and environments from which they were collected, it is not proper to validate them by a single criterion.

In this paper, we propose an evolutionary fuzzy clustering and knowledge-based evaluation method. The genetic algorithm is used for the evolutionary fuzzy clustering method, because it is an efficient method to solve optimization problems.[7] Using this method, clustering gets to be less subject to initial values and closer to the optimal solution.[8] There are many publications that are related to evolutionary computation for clustering. Maulik tried to minimize the distances between the data in the same cluster and between cluster centers,[6] and there was a study of a genetic algorithm to minimize objective function value of hard and fuzzy c-means algorithms.[5] However, the number of clusters was fixed and genetic algorithm was used only for the minimization of the objective function, and the authors did not compare several cluster partitions at the same time. The proposed method in this paper encodes

*Author to whom correspondence should be addressed.

one cluster partition as one chromosome and forms various cluster partitions, and finds out the optimal cluster partition with a genetic algorithm. The fuzzy c-means algorithm with GA is used for clustering, and the Bayesian validation method, which is a fuzzy cluster validity measure, is used to evaluate the fitness for every individual. Bayesian validation method requires α-cut value, and it is obtained using prior knowledge of the datasets. We have performed experiments using SRBCT (small round blue cell tumours) and yeast cell-cycle datasets from microarray experiments and compared the results with conventional methods. Finally, we have analyzed the optimal cluster partition to investigate the biological significance of our findings.

2. BACKGROUND

2.1. DNA Microarrays

We can measure expression levels of thousands of genes by a single experiment using the microarray technology. It consists of spatially ordered probes of cDNA or oligonucleotides on a chip. In this paper, two cDNA microarrays, yeast cell-cycle and SRBCT datasets, are used for experiments.

The first step for cDNA microarray experiment is RNA extraction from a tissue sample and RNA amplification. The RNA is reversely transcribed to cDNA labels using different fluorescent dyes mixed (red-fluorescent dye Cy5 and green-fluorescent dye Cy3). Due to the complementarity of base-pairing, the cDNA binds to specific oligonucleotides on the array, and the dye is excited by a laser so that the amount of cDNA can be quantified by measuring the fluorescence intensities.[9, 10] The log ratio of two intensities of each dye is used for the gene expression profiles.

$$\text{gene\_expression} = \log_2 \frac{\text{Int(Cy5)}}{\text{Int(Cy3)}} \tag{1}$$

Here, Int(Cy5) and Int(Cy3) are the intensities of red and green colors.

These measurements are repeated for every sample. After all experiments are finished, the data are incorporated into one table of the gene expression matrix.

2.2. Fuzzy c-Means Algorithm

Fuzzy c-means algorithm proposed by Bezdek is the most widely used fuzzy clustering method. Given dataset, $X = \{x_1, x_2, \ldots, x_n\}$, and the central vector of fuzzy clustering, $V = \{v_1, v_2, \ldots, v_c\}$, an objective function is defined with the membership degree between each data x_j and cluster center v_i.

$$J_m(X, U, V) = \sum_{j=1}^{n} \sum_{i=1}^{c} (\mu_{ij})^m d^2(x_j, v_i) \tag{2}$$

Here, μ_{ij} is the membership degree of x_j and the ith cluster, an element of the membership matrix $U = [\mu_{ij}]$. $d^2(,)$ is the square of the Euclidean distance, and m is the fuzziness parameter, which indicates the degree of fuzziness of each datum's membership degree; it should be bigger than 1.0[4]. In case it is 1.0, the algorithm becomes the same as the hard c-means algorithm.

The process below is one implementation of the fuzzy c-means algorithm.

- Step 1: Set c, the number of clusters, and m, the fuzziness parameter.
- Step 2: Initialize μ_{ij} as satisfying Eq. (3).

$$\sum_{i=1}^{c} \mu_{ij} = 1, 1 \le j \le n \tag{3}$$

- Step 3: Compute v_i, each center of all clusters. ($i = 1, 2, \ldots, c$)

$$v_i = \frac{\sum_{j=1}^{n} \mu_{ij}^m x_j}{\sum_{j=1}^{n} \mu_{ij}^m} \tag{4}$$

- Step 4: Compute the membership matrix U.

$$\mu_{ij} = \frac{(1/d^2(x_j, v_i))^{1/m-1}}{\sum_{k=1}^{c} (1/d^2(x_j, v_k))^{1/m-1}} \tag{5}$$

- Step 5: Repeat step 3 and 4 until Eq. (5) is satisfied. l is the iteration step.

$$|\{J_m^{(l)} - J_m^{(l-1)}\}| \le \varepsilon \tag{6}$$

3. PROPOSED METHODS

Here, we propose an evolutionary clustering method to search the optimal cluster partition and knowledge-based evaluation with Bayesian validation and prior knowledge of data. Figure 1 shows the flow chart of the proposed method.

The proposed methods are divided in two parts: evolutionary clustering part, which searches optimal cluster partition using fuzzy clustering and a genetic algorithm, and knowledge-based evaluation part, which obtains the

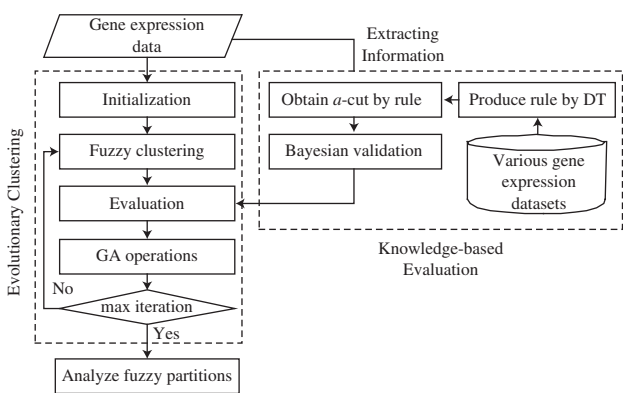


Fig. 1. The flow chart of the proposed method.

optimal α-cuts from data, required by Bayesian validation method and evaluates the clustering result. Evaluating datasets with the same criteria may lead to a wrong result, since each dataset has its own characteristics. Therefore, information for evaluation is extracted from datasets, and it is used for fitness evaluation of chromosomes.

3.1. Knowledge-Based Evaluation of Fuzzy Clustering Result

This section describes a Bayesian validation method which is used for fitness evaluation and for finding optimal α-cut values with the rules obtained by decision tree.

3.1.1. Bayesian Validation Method

This is a method based on probability. Given a dataset, it evaluates clustering results using posterior probability of cluster partition for that dataset. As mentioned before, this method decides an optimal cluster partition when the posterior probability of each cluster for the given data is maximized.[13]

$$\max P(Cluster|Dataset) \tag{7}$$

Applying Bayes' theorem, we can calculate the posterior probability as follows.

$$P(Cluster|Dataset) = \frac{P(Cluster)P(Dataset|Cluster)}{P(Dataset)} \tag{8}$$

When dataset D satisfies the condition $D = \{d_1, d_2, \ldots, d_N\}$, Eq. (8) is represented as Eq. (9) by multiplication rule and independence rule if each d_i is independent on one another.[14]

$$\begin{aligned} P(Cluster|Dataset) &= P(Cluster|d_1, d_2, \ldots, d_N) \\ &= P(Cluster|d_1) \times P(Cluster|d_2) \\ &\quad \times \cdots \times P(Cluster|d_N) \end{aligned} \tag{9}$$

Bayesian score (BS) is defined as the sum of all P (*Cluster|Dataset*) like Eq. (10) using the previous processes. The higher the Bayesian score is, the better the cluster partition is, because it means higher posterior probability.

$$\begin{aligned} BS &= \frac{\sum_{i=1}^{c} P(C_i|D_i)}{c} = \frac{\sum_{i=1}^{c} P(C_i|d_{i1}, d_{i2}, \ldots, d_{iN})}{c} \\ &= \frac{\sum_{i=1}^{c} P(C_i|d_{i1})P(C_i|d_{i2}) \cdots P(C_i|d_{iN})}{c} \\ &= \frac{\sum_{i=1}^{c} \prod_{j=1}^{N_i} P(C_i)P(d_{ij}|C_i)/P(d_{ij})}{c}, \\ &\quad D_i = \{d_{ij}|\mu_{ij} > \alpha, 1 \le j \le n\}, N_i = n(D_i) \end{aligned} \tag{10}$$

In Eq. (10), n(Di) is the number of Di, and we choose the samples that have higher degree of membership values than a certain probability, because the computation process of Bayesian score includes multiplication and it produces a wrong value if one of those degrees of membership is zero. Besides, data of higher membership degree are more correct and informative. α-cut plays a role of this threshold. Eq. (11) shows the computation of each probability.

$$\begin{aligned} P(C_i) &= \frac{\sum_{j=1,\, u_{ij}>\alpha}^{n} u_{ij}}{\sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}} \\ P(d_{ij}) &= \sum_{i=1}^{c} P(C_i)P(d_{ij}) = \sum_{i=1}^{c} P(C_i)u_{ij} \end{aligned} \tag{11}$$

When the membership matrix is produced as a fuzzy cluster result, each degree of membership means the probability that each sample belongs to each cluster. Therefore, the membership degree of each sample, U_{ij}, can be represented as $P(d_{ij}|C_i)$. Figure 2 illustrates the overall process of the Bayesian validation method.

First, specific samples that have higher membership degrees than threshold ($U_{ik} > \alpha$) are selected based on membership degrees, which become the clustering result. U_{ik} means the belongingness of the ith sample to the kth

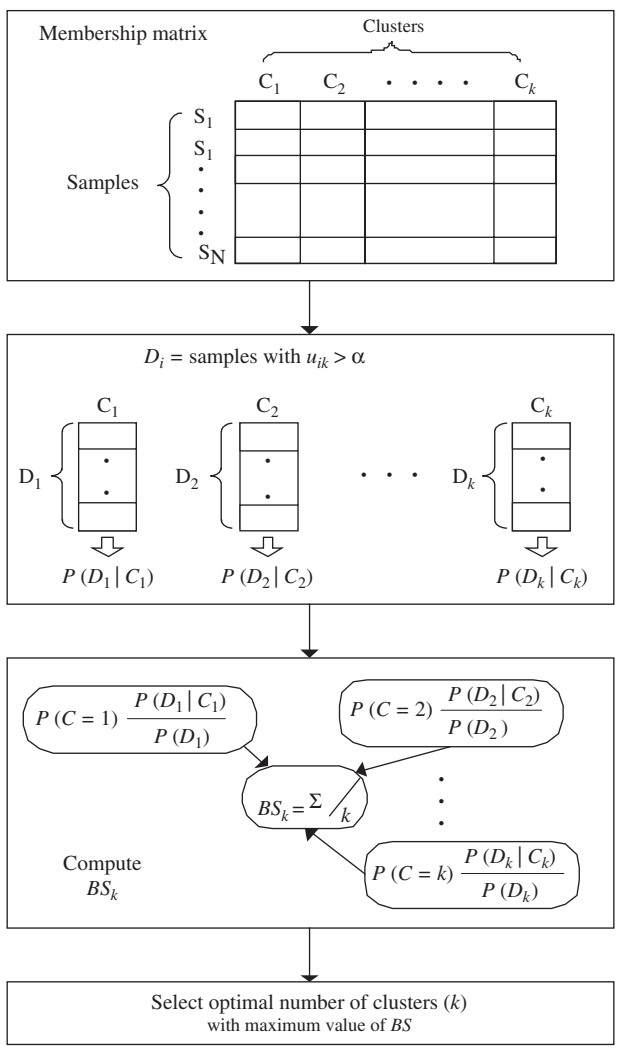


Fig. 2. The overall process of the Bayesian validation method.

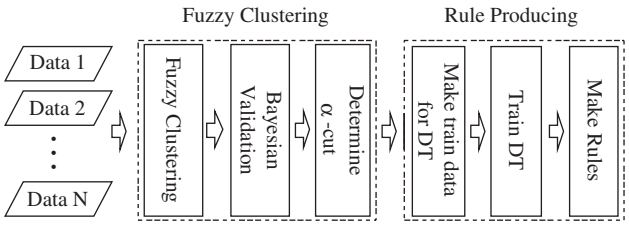


Fig. 3. Decision of α-cut by decision tree rule.

cluster, and it should be between 0 and 1. If it is close to 1, it means high belongingness to its cluster. After samples are selected, $P(D_k|C_k)$ of each cluster is calculated, leading to a Bayesian score.

3.1.2. Finding Optimal α-Cut by a Decision Tree

Generally a decision tree (DT) is used to solve classification and prediction problems. In this paper, C4.5 algorithm that is a representative one for decision trees is used to find an optimal α-cut value for each dataset.[15] Setting α-cut value affects Bayesian score that is the fitness evaluation result. Equation (12) is a definition of α-cut.[16]

$$A_\alpha = \{x \in X | u(x) \geq \alpha\}, \quad 0 < \alpha < 1 \tag{12}$$

A_α is a set of elements that have higher belongingness than α, and x means each element. According to the value of α, A_α can be a variety of sets. If the membership function is linear, the setting of α-cut is simple such as $\alpha = 0.5$ or $\alpha = 1.0$, but if it is not linear, various α-cut values are needed. D_k in Figure 3 can change by α, the proper setting of α-cut for specific datasets is an important problem.

As the original Bayesian validation method used an uniform α-cut value for all datasets,[13] it cannot evaluate correctly, because each dataset has a different sample distribution. This paper obtains an α-cut value for each dataset using the rule of DT. Figure 3 shows the α-cut setting process from fuzzy clustering result.

First, N gene expression profiles are clustered using the fuzzy c-means algorithm, and these clustering results are evaluated by the Bayesian validation method. Subsequently, the optimal α-cut for each dataset is decided, and they are used for the labels of DT training data. Rule production process trains DT, and produces rules.

As Figure 4 indicates, the attributes of DT training data are produced using membership matrices that are the fuzzy cluster results of each dataset. Incrementing the membership degree-2.4 value from 0.0 to 1.0 with the difference of 0.1, attributes are divided into 10 sections. Each section counts the frequency of samples and calculates the attribute by dividing the frequency by the total number of samples. These attributes are $A_1 \sim A_{10}$.

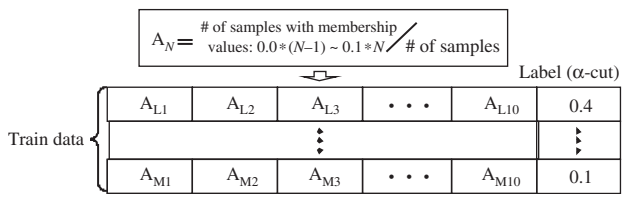


Fig. 4. The training data production process of a decision tree.

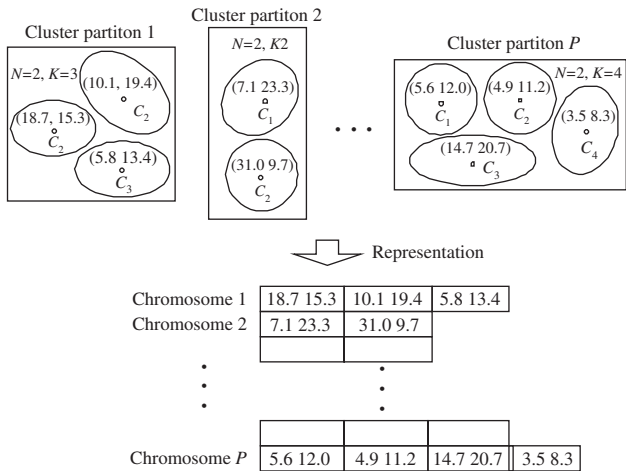


Fig. 5. Representation of variable length chromosome.

The training made by the process in Figure 5 sets α-cut value when experimental dataset is inputted. Using these α-cut values, Bayesian validation method calculates the final Bayesian score of fitness evaluation result.

3.2. Evolutionary Fuzzy Clustering

This section describes the evolutionary fuzzy clustering method, which is the fuzzy clustering method using a genetic algorithm, to search the optimal cluster partition.

3.2.1. Representation

Generally, binary representation is used as a chromosome representation since it is easy to implement and apply. This paper, however, has used floating point representation to represent a set of cluster centers of cluster partition information. One cluster partition consists of K clusters, and a chromosome is represented in a space of $N \times K$ in case that the dimension of each center is N.

This paper evaluates cluster partition with various numbers of clusters, so variable length chromosome has been used. As Figure 5 illustrates, several chromosomes are in one cluster partition, and each chromosome has different number of clusters and different value of cluster centers.

3.2.2. Population Initialization and Fitness Evaluation

Population is initialized at random. For a chromosome that contains K clusters, K random samples are extracted from data, and they are used for cluster centers. This is repeated as the number of chromosomes. When clustering a specific dataset, numbers less than the square value of the number of samples are used.[17] The minimum number of clusters is set as 2.

Fitness evaluation is divided into two parts. First, all samples are clustered with cluster centers in chromosomes using the fuzzy c-means algorithm. For the construction of a membership matrix using distances among all samples and clusters, we have updated cluster centers as Eq. (13) and Eq. (14). The information of cluster centers in chromosomes is changed through updating these cluster centers

$$u_{ij} = \frac{(1/d^2(x_j, \nu_i))^{1/m-1}}{\sum_{k=1}^{c}(1/d^2(x_j, \nu_i))^{1/m-1}} \tag{13}$$

$$\nu_i = \frac{\sum_{j=1}^{n} u_{ij}^m x_j}{\sum_{j=1}^{n} u_{ij}^m} \tag{14}$$

Second part evaluates the changed chromosomes by Bayesian validation method. Various evaluation values are calculated since each chromosome has different information of cluster partition. Selection is done by these values. For selection, we have used a roulette wheel strategy that tries to select many copies of individuals corresponding to its fitness.[18]

3.2.3. Crossover and Mutation

This paper cannot use general crossover operation, because the length of chromosome is variable, so crossover operation is performed as in Figure 6. After deciding the crossover point, the length of one part is fixed for crossover, and the other part of the chromosomes is crossed over.

In case of the chromosome of length l, crossover point is decided randomly in $[1, l-1]$ with the fixed crossover rate.

Mutation is set to occur by a fixed mutation rate. Since this paper adopts the floating point representation, mutation is set to occur by Eq. (15) and Eq. (16). When δ is a variable of uniform distribution in [0, 1] and ν is a value of mutation point, a value of new ν is decided as in Eq. (15) and Eq. (16).[6]

$$\nu \pm 2 \times \delta \times \nu, \ \nu \neq 0 \tag{15}$$

$$\nu \pm 2 \times \delta, \ \nu = 0 \tag{16}$$

Equation (15) is used when ν is not zero, and Eq. (16) is used when ν is zero. The probabilities of sign '+' and '−' are the same.

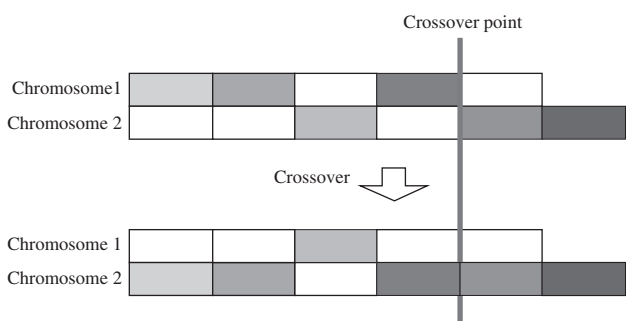


Fig. 6. An example of crossover.

4. EXPERIMENTS

4.1. Experimental Environment

4.1.1. Experimental Data

Lymphoma, leukemia, and serum datasets are used for experiments of optimal α-cut decision by a decision tree, and for most comparisons and analyses, SRBCT and Yeast cell-cycle datasets are used. The information of SRBCT and Yeast cell-cycle datasets are as follows.

- SRBCT dataset: This has 63 samples with 6567 genes and consists of 4 classes, NB (neuroblastoma), RMS (rhabdomyosarcoma), NHL (non-Hodgkin lymphoma) and EWS (Ewing family of tumours). They are all different kinds of cancer, each with different characteristics. This paper used 96 genes that are known as informative ones[20] to apply the proposed method to 63 samples.
- Yeast cell-cycle dataset: This is a dataset that has expression levels of 6000 genes expressed during 2 cell-cycles.[21] Expression levels are measured on 17 different time points every 10 minutes. This dataset is frequently used for genetic analysis since many genes classified by their biological function have different expression levels according to the cell cycle. A total of 421 genes that show significant change of expression levels are used in this paper.[21]

4.1.2. Parameters and Settings

In our experiments, the maximum generation number is 1000, and population sizes of 100 and 200 are used for SRBCT and yeast cell-cycle datasets. The size of SRBCT dataset is smaller than yeast cell-cycle dataset. Maximum numbers of clusters are 8 and 20 for SRBCT and yeast cell-cycle datasets, respectively. Crossover rate of 0.8 and mutation rate of 0.01 are used. The fuzziness parameter of the fuzzy c-means algorithm set as 1.2.

4.2. Experimental Results

4.2.1. Decision of Optimal α-Cut

We have produced training data of decision trees using five gene expression profiles: lymphoma,[22] leukemia,[23] SRBCT, serum[24] and yeast cell-cycle datasets. A rule produced by decision tree using these datasets is shown in Figure 7. The first α-cut, 0.8, is decided by the first attribute (A_1) which is defined in Figure 4, and 0.2 is decided by the third attribute (A_3), and by the last attribute (A_{10}), 0.1, and 0.4 are decided. These are α-cut values for several different training datasets.

By a decision tree, α-cuts of SRBCT and yeast cell-cycle datasets are decided to be 0.2 and 0.4, respectively. Two datasets go to the same direction through A_1, but they are distinguished by A_3. Training data of two datasets are shown in Table I. SRBCT dataset has smaller values of

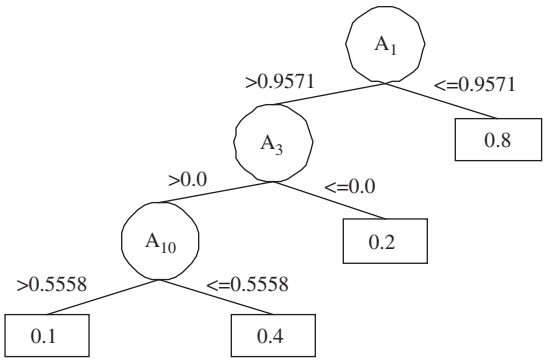


Fig. 7. A rule produced by decision tree.

A_2 and A_9 but has larger value of A_{10}. It means that samples of SRBCT dataset have clearer boundaries and higher membership degrees. This example shows that different gene expression profiles have different distributions, therefore they should be evaluated differently considering their characteristics.

4.2.2. Result of Optimal Cluster Partition Search

Figure 8 illustrates average fitness transition of SRBCT dataset as the generation grows. Experiments have been repeated 10 times, and red line shows the average. It evolves rapidly until the generation number is close to 20 and converges thereafter. The convergence value of SRBCT dataset is about 0.6. Figure 9 illustrates the same graph of yeast cell-cycle dataset. It converges more slowly than SRBCT dataset and average fitness changes slowly until the generation number is around 80. It converges to 0.12 with large oscillations.

Figures 8 and 9 show different transition patterns, and this can be thought to be due to different characteristics of two datasets influencing the evolution process. Observing the fuzzy cluster result, most genes of SRBCT dataset have membership degrees that are larger than 0.9 or smaller than 0.1. On the other hand, yeast cell-cycle dataset has various ranges of membership degrees.

4.2.3. Comparison with the Original Fuzzy c-Means Algorithm

In the previous section, we confirmed that the result of the FCM with GA evolves well. Here, we compare it with the original FCM by means of Bayesian score (BS) and the objective function value (OF value) of the FCM to show the usefulness of the Bayesian validation method. Table II shows 10 experimental results of SRBCT dataset. If BS is high and the objective function value is low, it means that clustering performed well since the objective function value is based on the distances between cluster centers and samples as mentioned in Eq. (2). The proposed method shows better results than original FCM in both BS and OF value.

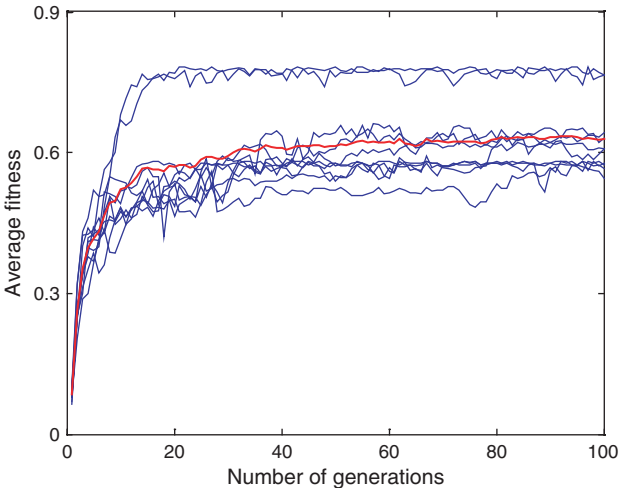


Fig. 8. Average fitness transition of SRBCT dataset ($P = 100$).

Table III represents 10 experimental results of yeast cell-cycle dataset. In case of SRBCT dataset, though the result of the proposed method was slightly better, its difference was not significant. The result of yeast cell-cycle dataset, however, shows a relatively significant difference.

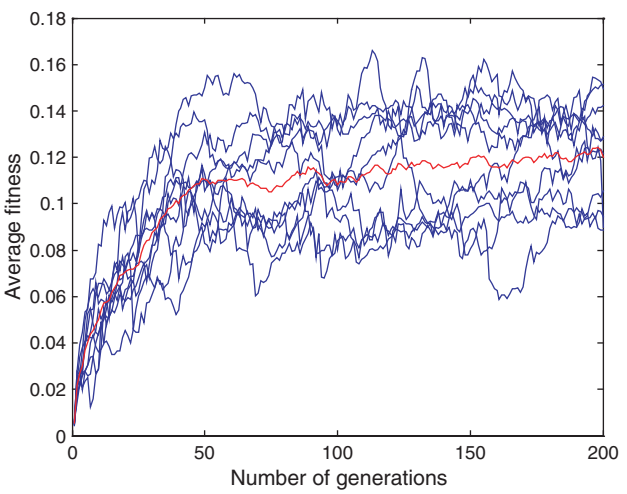


Fig. 9. Average fitness transition of yeast cell-cycle dataset ($P = 200$).

Table I. Training data of decision tree.

	Attributes of training data									
Dataset name	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}
Yeast	1.000	0.231	0.114	0.086	0.071	0.064	0.059	0.069	0.162	0.546
SRBCT	1.000	0.016	0.000	0.016	0.000	0.016	0.000	0.000	0.048	0.937

RESEARCH ARTICLE

Table II. Comparison experiment (SRBCT).

	Original FCM		GA + FCM	
Count	BS	OF value	BS	OF value
1	0.58028	156.9920	0.58042	156.9918
2	0.58034	156.9925	0.58036	156.9920
3	0.58031	156.9921	0.58041	156.9919
4	0.58029	156.9922	0.58036	156.9920
5	0.58041	156.9926	0.58036	156.9920
6	0.58034	156.9921	0.58041	156.9919
7	0.58031	156.9920	0.58042	156.9918
8	0.58028	156.9922	0.58042	156.9918
9	0.58033	156.9922	0.58041	156.9919
10	0.58036	156.9920	0.58042	156.9918
Average	0.58033	156.9922	0.58040	156.9919

Comparing the proposed method with the original FCM, we have confirmed that the result of the proposed method is closer to the optimal solution than the original FCM.

4.3. Analyses of Results

4.3.1. Analysis of SRBCT Dataset

We have compared and analyzed the result of SRBCT dataset with Khan's work.[20] Figure 10 shows the 4 clusters that the proposed method has searched and actual clusters. All 63 samples are clustered to their class correctly. The

Table III. Comparison experiment (Yeast cell-cycle).

	Original FCM		GA + FCM	
Count	BS	OF value	BS	OF value
1	0.03354	164.472	0.13256	166.883
2	0.00875	163.670	0.11246	161.542
3	0.03238	165.057	0.12661	162.911
4	0.03825	162.653	0.08058	162.073
5	0.02165	163.758	0.10667	162.798
6	0.04096	164.086	0.09778	162.312
7	0.02806	163.052	0.11873	162.042
8	0.04473	164.877	0.13659	162.773
9	0.02478	162.452	0.12898	162.905
10	0.04645	169.216	0.11246	161.542
Average	0.03195	164.329	0.11534	162.778

Table IV. List of samples.

Fuzzy sample	First cluster	Second cluster
EWS-T19	0.549942 (3)	0.379079 (4)
NB-C5	0.564467 (6)	0.313732 (7)
RMS-C6	0.349232 (7)	0.333511 (6)
RMS-C8	0.456904 (5)	0.339977 (1)
EWS-T13	0.718476 (3)	0.210988 (4)

sample EWS-T13, which is marked with a line, belongs to EWS class with the membership degree of 0.5043 and also to the RMS class with the membership degree of 0.3860.

We have conducted an additional experiment with the 2308 samples used by Khan[20]—the obtained clusters are shown in Table IX.

Table IV represents fuzzy samples that have higher membership degrees than 0.3 and belong to several clusters simultaneously. Membership degrees and cluster numbers are shown in the table. The last sample EWS-C13 is searched when only 96 meaningful samples are used for experiments. Considering it belongs to cluster 3 and

Table V. Clustering result of SRBCT dataset (2308 samples).

Cluster number	Samples
Cluster 0	EWS-C8 EWS-C6 EWS-C9 EWS-C11 EWS-C10
Cluster 1	EWS-C3 EWS-C2 EWS-C1 BL-C1 BL-C2 BL-C3 BL-C4 NB-C1
Cluster 2	BL-C5 BL-C6 BL-C7 BL-C8
Cluster 3	EWS-T6 EWS-T7 EWS-T9 EWS-T11 EWS-T12 EWS-T14 EWS-T15 **EWS-T19**
Cluster 4	EWS-T13 RMS-C4 RMS-T6 RMS-T7 **RMS-T8** RMS-T5 RMS-T10 RMS-T11
Cluster 5	EWS-T1 EWS-T2 EWS-T3 EWS-T4 EWS-C4 RMS-C8 RMS-C11 RMS-T1 RMS-T4 RMS-T2 RMS-T3
Cluster 6	NB-C2 NB-C3 NB-C6 NB-C12 NB-C7 NB-C4 **NB-C5** NB-C10 NB-C11 NB-C9 NB-C8
Cluster 7	EWS-C7 RMS-C3 RMS-C9 RMS-C2 RMS-C5 **RMS-C6** RMS-C7 RMS-C10

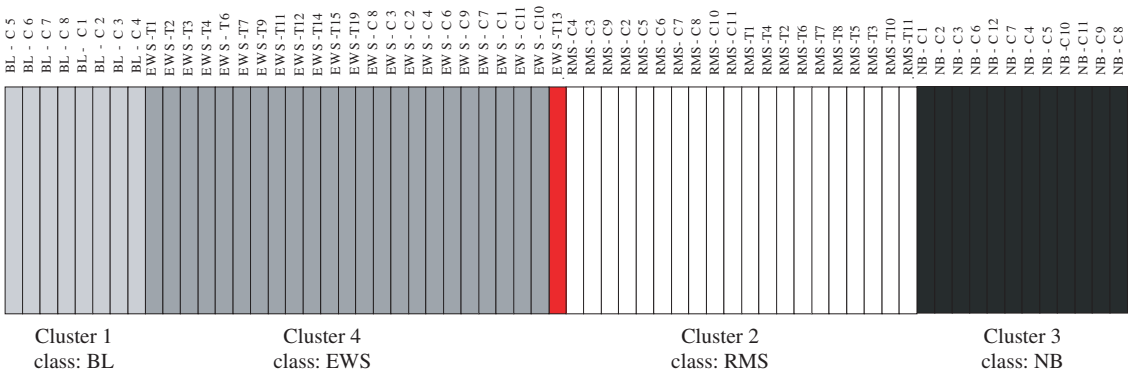


Fig. 10. Clustering result of the SRBCT dataset (96 samples).

RESEARCH ARTICLE

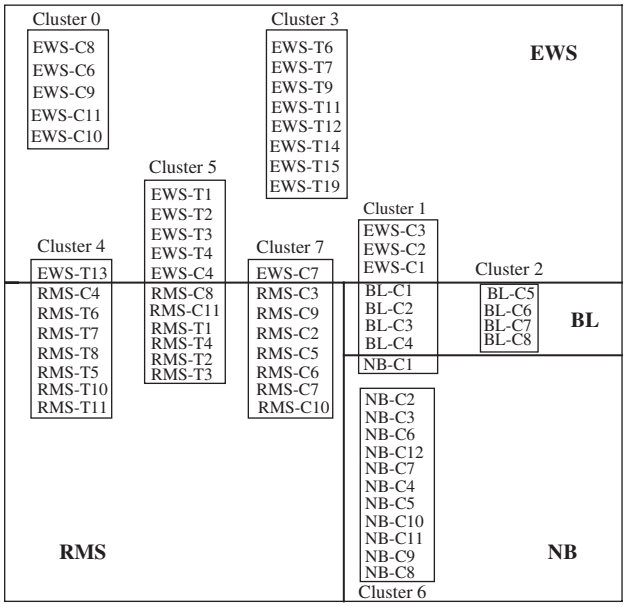


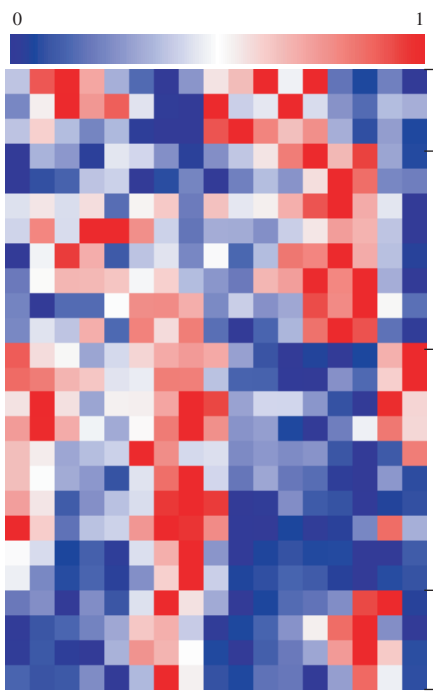
Fig. 11. Classification of clusters of SRBCT dataset (2308 samples).

4, we can conjecture they are related to class EWS or RMS. Table V shows clusters and samples of those clusters. Some clusters have samples of several classes. Fuzzy samples are marked with bold type.

Analyzing the relationship of each cluster based on Table V, the result is presented in Figure 11. Cluster 0 and cluster 3 belong to only class EWS, and cluster 2 and cluster 6 belong to class BL and NB respectively. On the other hand, cluster 4, cluster 5, and cluster 7 belong to both class EWS and class RMS. Cluster 4 and cluster 7 can be thought of as the member of class RMS since all samples except one belong to the class RMS. In case of cluster 5, half of them belong to class EWS, and the other half belong to class RMS, so it can be thought of as a cluster having both characteristics of class EWS and RMS. Using these results of 2308 samples, we can find

Table VI. Fuzzy clustering of genes from the yeast cell-cycle dataset.

Gene	First cluster	Second cluster
YBL032w	0.35035 (7)	0.33226 (4)
YHR031C	0.40455 (4)	0.38120 (7)
YCL063w	0.40413 (7)	0.39001 (11)
YBR007c	0.52122 (5)	0.39115 (15)
YER019w	0.43167 (5)	0.32937 (15)
YDR297w	0.62344 (5)	0.31825 (13)
YER118c	0.6049 (5)	0.33987 (13)
YHR173C	0.39546 (13)	0.38228 (5)
YLL021w	0.66923 (5)	0.31998 (13)
YBR275c	0.59041 (5)	0.37740 (12)
YJL173C	0.43414 (5)	0.41046 (12)
YBR053c	0.45555 (0)	0.44679 (1)
YKL163W	0.4623 (0)	0.36860 (1)
YLL040c	0.44400 (0)	0.34000 (1)
YML110C	0.59168 (1)	0.32380 (0)
YDL119c	0.4835 (0)	0.38849 (2)
YBR158w	0.5869 (1)	0.40988 (2)
YDL179w	0.55259 (1)	0.43413 (2)
YIL009W	0.60611 (1)	0.35258 (2)
YNL046W	0.5581 (2)	0.43928 (1)
YOR264W	0.69030 (1)	0.30635 (2)
YDL127w	0.52180 (12)	0.44541 (10)
YJL187C	0.56953 (12)	0.33059 (10)
YMR078C	0.5827 (12)	0.41445 (10)
YMR179W	0.56833 (10)	0.40397 (12)



Gene description	Clusters
weak similarity to hnRNP complex protein homolog YBR233w	4, 7
similarity to Pif1p	4, 7
weak similarity to yeast translation regulator Gcd6p	7, 11
hypothetical protein	5, 15
hypothetical protein	5, 15
suppressor of rvs161 and rvs167 mutations	5, 13
involved in the HOG1 high-osmolarity signal transduction	5, 13
hypothetical protein	5, 13
forms schmoos poorly	5, 13
Rap1p interacting factor. telomere silencing	5, 12
DNA replication factor A, 13 kDa subunit	5, 12
similarity to rat regucalcin	0, 1
protein with internal repeats	0, 1
involved in regulating membrane traffic	0, 1
ubiquinone biosynthesis, methyltransferase	0, 1
similarity to bovine Graves disease carrier protein	0, 2
hypothetical protein	1, 2
cyclin like protein interacting with Pho85p	1, 2
acyl CoA synthase	1, 2
hypothetical protein	1, 2
hypothetical protein	1, 2
cyclin, G1/S-specific	10, 12
phosporylates Cdc28, SPB separation, nuclear division	10, 12
for chromosome transmission in mitosis and normal telomere	10, 12
required for normal transcription at a number of loci	10, 12

Fig. 12. Expression level, gene description and cluster number of "fuzzy" genes of yeast cell-cycle dataset searched by the proposed method.

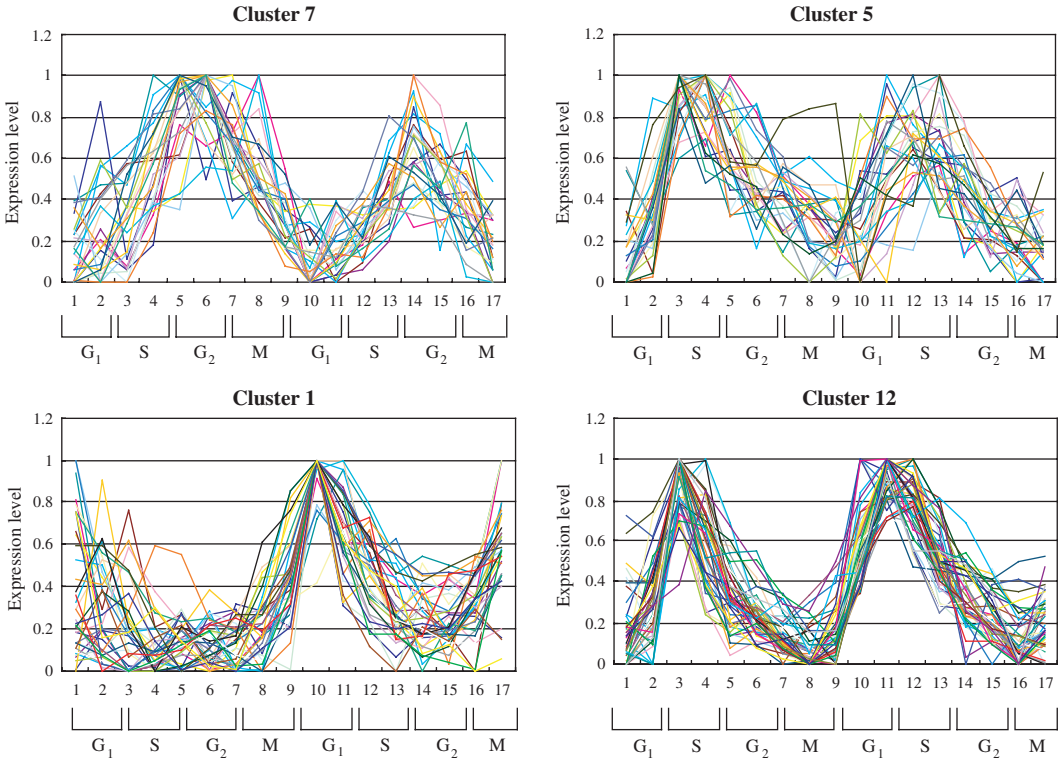


Fig. 13. Change of gene expression level over a time course.

further characteristics of samples and various analyses are possible though the accuracy is lower.

4.3.2. Analysis of Yeast Cell-Cycle Dataset

We have compared and analyzed the result of yeast cell-cycle dataset with the known genes at Cho's work.[21] In particular, we have focused on fuzzy genes, which have membership degrees higher than 0.3 and belong to several clusters at once. Table VI shows the membership degrees and cluster numbers of "fuzzy" genes. The number in parentheses means the cluster number.

We have divided the genes in Table VI into 4 groups. First three genes, YBL032w, YHR031C, and YCL063w, are members of cluster 4, cluster 7, and cluster 11, respectively. All of them belong to cluster 7, and they are one group. Another group of cluster 5, cluster 12, cluster 13, and cluster 15 is grouped on cluster 5. Group of cluster 0, cluster 1, and cluster 2 and group of cluster 10 and cluster 12 are the remaining two groups. Observing expression levels on the left figure, it is easily confirmed that patterns are distinguished by group. Compared these fuzzy genes with known functions of yeast cell-cycle dataset, discovered information is provided in Figure 12. For example, YDR297w of the second group is known as a suppressor of rvs161 and rvs167 mutations, and YJL173C in the same group is known as a DNA replication factor. The known functions of the other genes are described in Figure 12.

Each of four cluster groups is related to specific phase of the cell division. Figure 13 illustrates the transitions of expression levels of four clusters, which is represented in each group, according to cluster numbers. Cluster 7 has 26 genes, and they express the most at G_2 phase of cell-cycle, so cluster 7 is thought to be related to G_2 phase. In case of cluster 5, most genes express with high level in S phase, and that time point is a little earlier than genes of cluster 7. Cluster 1 of the third group expresses the most G_1 phase, and cluster 12 of the last group expresses the most between G_1 and S phase. Considering that genes of cluster 12 were grouped with cluster 5 in Figure 12, it is also related to the second group.

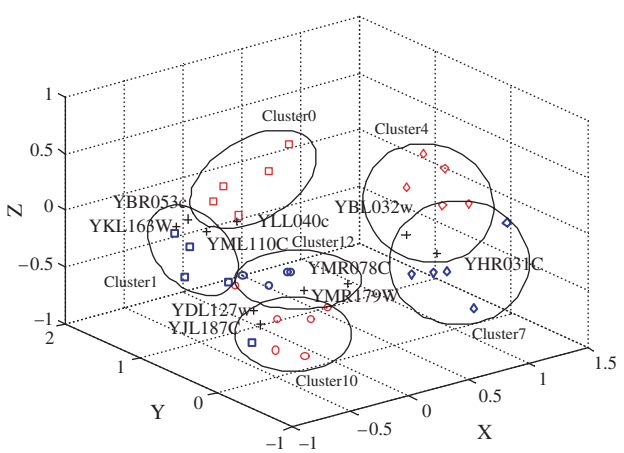


Fig. 14. A distribution of fuzzy clustered genes (yeast cell-cycle dataset).

RESEARCH ARTICLE

Figure 14 is a 3 dimensional distribution of the genes from Table VI. This is obtained using PCA (Principal Component Analysis). Genes of each group are distinguished by their shapes, and "fuzzy" genes are marked with '+'. As shown in this figure, the fuzzy genes are located near the boundaries of two clusters.

5. CONCLUSIONS AND FUTURE WORK

This paper has proposed an evolutionary clustering method to search for optimal cluster partitioning and knowledge-based cluster validation. Applying the proposed method to SRBCT and yeast cell-cycle gene expression datasets and comparing the results with previous methods has shown a better performance than the original fuzzy c-means method. Finally, we have analyzed the optimal cluster partitions searched by the proposed method to find their biological significance. Though the proposed method provides better results than conventional methods, there are some disadvantages. Genetic algorithm evolution process, including variable length chromosomes and their operations, takes longer time than conventional methods, and evaluation also takes longer since it requires the creation of the validation rules in advance.

Future research includes experiments on various datasets since we have performed experiments only on two datasets. More comparisons with other cluster validation methods are needed. The proposed here method shows potentials for a detailed analysis of microarray data, especially for genes involved in more than one biological function. It thus contributes to the better utilization of the massive amount of data coming from DNA microarray chips for the creation of novel nanotechnologies.

Acknowledgments: This work was supported by a grant of Korea Health 21 R & D Project, Ministry of Health & Welfare, the Republic of Korea. The clustering software used in this paper is available from: http://sclab.yonsei.ac.kr/software/SClusteringTool.zip.

References

1. Y. Ando et al., *Materials Today* 7, 22 **(2004)**.
2. U. Alon et al., Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Natl. Acad. Sci. USA* 96, 6745 **(1999)**.
3. A. P. Gasch and M. B. Eisen, *Genome Biology* 3, research 0059.1-0059.22 **(2002)**.
4. N. Bolshakova and F. Azuaje, *SIGPRO* 21, 1 **(2002)**.
5. L. O. Hall et al., Clustering with a genetically optimized approach. *IEEE Trans. On Evolutionary Computation* 3, 103 **(1999)**.
6. U. Maulik and S. Bandyopadhyay, *Pattern Recognition* 33, 455 **(2000)**.
7. L. Chamber, *Practical Handbook of Genetic Algorithm*, CRC Press **(1995)**.
8. J. N. Bhuyan et al., Genetic algorithm for clustering with an ordered representation. *Proc. 4th Int. Conf. Genetic Algorithms* **(1991)**, pp. 408–415.
9. M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein, Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA* **(1998)**, Vol. 95, pp. 14863–14868.
10. M. E. Futschik, A. Reeve, and N. Kasabov, *Artificial Intelligence in Medicine*. 28, 165 **(2003)**.
11. R. E. Hammah and J. H. Curran, Validity measures for the fuzzy cluster analysis of orientations. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **(2000)**, Vol. 22.
12. F. Hoppner et al., *Fuzzy Cluster Analysis.* Wiley, **(1999)**, pp. 43–39.
13. S.-H. Yoo et al., Analyzing fuzzy partitions of Saccharomyces cerevisiae cell-cycle gene expression data by Bayesian validation method. *Proc. of the 2004 IEEE Symposium on CIBCB* **(2004)**, pp. 116–122.
14. T. M. Mitchell, *Machine Learning.* Carnegie Mellon University **(1997)**.
15. S. Ruggieri, Efficient C4.5. *IEEE Transactions on Knowledge and Data Engineering* **(2002)** Vol. 14, pp. 438–444.
16. P. Baranyi et al., A new method for avoiding abnormal conclusion for α-cut based rule interpolation. *8th IEEE Int. Conf. on Fuzzy Systems*, Seoul, Korea **(1999)**, pp. 383–388.
17. D. Dembele and P. Kastner, *Bioinformatics* 19, 973 **(2003)**.
18. K. Krishna and M. N. Murty, Genetic k-means algorithm. *IEEE Trans. On Systems, Man and Cybernetics* **(1999)**, Vol. 20, pp. 433–439.
19. J. C. Bezdeck, *J. Cybernit* 3, 58 **(1974)**.
20. J. Khan et al., *Nature* 7, 673 **(2001)**.
21. R. J. Cho et al., *Molecular Cell* 2, 65 **(1998)**.
22. A. A. Alizadeh et al., *Nature* 403, 503 **(2000)**.
23. T. R. Golub et al., *Science* 286, 531 **(1999)**.
24. V. R. Iyer et al., *Science* 283, 83 **(1999)**.

Received: 16 November 2004. Accepted: 4 May 2005.