Journal of Intelligent & Fuzzy Systems 18 (2007) 543–559
IOS Press

Evolutionary fuzzy cluster analysis with Bayesian validation of gene expression profiles

Han-Saem Park and Sung-Bae Cho*
Department of Computer Science, Yonsei University, 134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea

Abstract. Clustering analysis of the gene expression profiles has been used for identifying the functions of unknown genes. Fuzzy clustering method, which is one category of clustering, assigns one sample to multiple clusters as their degrees of membership. It is more appropriate for analyzing gene expression profiles because genes usually belong to multiple functional families. However, general clustering methods have problems that they are sensitive to initialization and can be trapped into local optima. In this paper, we propose an evolutionary fuzzy clustering method with Bayesian validation which uses a genetic algorithm for fuzzy clustering process of gene expression profiles and Bayesian validation method for the fitness evaluation process. We have conducted in-depth experiments to verify the usefulness of the proposed method with well-known gene expression profiles of SRBCT and Saccharomyces.

Keywords: Evolutionary clustering, fuzzy clustering, Bayesian validation, gene expression profiles

1. Introduction

In general, analysis of gene expression profiles can be divided into two groups. One is the classification method based on supervised learning which employs data with known classes. The other is the clustering method based on unsupervised learning which deals data with unknown or partially known classes [1]. Clustering method groups thousands of genes by their similarities of expression levels so that it helps to analyze gene expression profiles [2]. It is also divided into two groups: the hard clustering and fuzzy clustering methods. The fuzzy clustering method, which assigns one sample to multiple clusters at the same time, is appropriate for analyzing genes since a single gene in gene expression profiles may have multiple functions in many cases [3,4].

Normally, clustering algorithms have common problems that they are very sensitive to initialization and they can be trapped into local optima because the processes are supposed to minimize objective function values [5–7]. Another important matter for cluster analyses is how many clusters are actually in the dataset and how good they are [8]. Therefore, it is necessary to evaluate each cluster and this evaluation is called cluster validity. Conventional cluster validity measures focused on the compactness and the variation within cluster. Those measures, however, cannot provide correct representation of fuzzy partition in the data because they are only based on the distance between the clusters [8,9].

In order to solve the problems mentioned above, we propose an evolutionary fuzzy clustering method with Bayesian validation. Evolutionary fuzzy clustering is to find the optimal cluster partition and Bayesian validation method is to evaluate the fitness of evolution process. The proposed method searches the optimal number of clusters and the centers at the same time, and it is applied to cluster DNA microarrays where cluster analyses have been usefully applied to. Experiments with the well-known DNA microarray datasets (SRBCT and Saccharomyces

*Corresponding author. Tel.: +82 2 2123 2720; Fax: +82 2 365 2579; E-mail: sbcho@cs.yonsei.ac.kr.

datasets) are demonstrated, and analysis of Saccharomyces cell cycle expression data follows to show the usefulness of the proposed method.

The rest of this paper is organized as follows. In Section 2, the backgrounds of this paper are introduced. Section 3 describes the proposed method, and experimental results and their analyses are presented in Section 4. Section 5 concludes the paper.

2. Related works

2.1. Fuzzy clustering in DNA microarray analysis

The development of microarray technology has created a great amount of information, and now it is possible to study tens of thousands of genes at once [10]. Since clustering is a very useful method to analyze these microarray data, many research groups have been utilized the cluster analyses. In particular, fuzzy clustering, which assigns one datum to multiple clusters at the same time, has been used to extract useful information from microarrays because a single gene may have multiple functions [11,12]. Gasch and Eisen explored the conditional coregulation of Saccharomyces cell cycle data using fuzzy k-means algorithm [13]. Dembele and Kastner tuned the fuzziness parameter of fuzzy c-means (FCM) algorithm and analyzed Serum, Saccharomyces and human cancer datasets [14]. Belacel et al. exploited fuzzy j-means algorithm, a local heuristic depended only on cetroids information, and variable neighborhood search of optimal searching method in order to analyze human breast cancer and blood datasets [15].

Another important issue in cluster analysis is a validation of the results [16]. Conventional fuzzy cluster validity measures such as Partition Coefficient (PC), Classification Entropy (CE), Fukuyama-Sugeno (FS) and Xie and Beni index (XB) validated the cluster results based only on the compactness and separateness of data within clusters [17]. Therefore, they may not perform the correct validatation of the cluster results.

2.2. Evolutionary clustering

General clustering methods have weaknesses that their results depend on initialization and can be trapped into local optima [18]. In order to solve this problem, evolutionary approaches have been applied the clustering [19]. Genetic algorithm (GA) has been applied to many optimization problems successfully [18]. Maulik tried to minimize the distances between the data in the same clusters and their cluster centers [6,20], and there were studies of GA to minimize objective function value of hard and fuzzy c-means algorithms [4,21]. They fixed the number of clusters and used GA only for the minimization of objective function.

Also, there have been studies that searched the number of clusters and their centers at the same time [22,23]. Nasraoui et al. proposed clustering algorithm based on genetic nitching and applied it to synthetic datasets and image segmentation [23]. Lee and Antonsson used evolutionary strategy to find the optimal number of clusters and clustered synthetic datasets [24]. We have used the genetic algorithm with variable length chromosome to search optimal number of clusters and their centers, and applied the method to clustering DNA microarrays of SRBCT and Saccharomyces datasets.

3. Evolutionary fuzzy clustering with Bayesian validation

We propose an evolutionary fuzzy clustering method with Bayesian validation, which searches optimal cluster partition using the fuzzy c-means algorithm with GA and evaluates the fitness with Bayesian validation method. Figure 1 provides the overall algorithm of the proposed method.

Evolutionary Fuzzy Clustering Algorithm with Bayesian Validation

Input:
Gene expression data information, *DataInfo*.
Output:
Optimal fuzzy partition information including cluster number and centers, *OptimalClusterPartition*.
Function description:
Initialize (*DataInfo*):
A function that initializes the cluster centers and membership matrix using data information.
FuzzyClustering ():
A function that performs fuzzy clustering using fuzzy c-means algorithm and returns its result.
BayesianValidation (*FuzzyClusterResult*, *DataInfo*):
A function that evaluates fuzzy cluster result with Bayesian validation and returns its result.
It requires data information and the results play a role of the fitness for an evolution process.
EvolutionProcess (*ValidationResult*):
A function that performs genetic operations such as selection, crossover and mutation.

```
{
    Initialize (DataInfo);
    for (counter=0; counter<maxIteration; counter++) {
        FuzzyClusterResult = FuzzyClustering ();
        ValidationResult = BayesianValidation (FuzzyClusterResult, DataInfo);
        EvolutionProcess (ValidationResult);
    }
    return OptimalClusterPartition;
}
```

Fig. 1. Algorithm of the proposed method.

3.1. Representation and initialization

Generally, binary representation is used for 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. 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 Fig. 2 illustrates, several chromosomes are in one cluster partition, and each chromosome has different number of clusters and different values of cluster centers.

Population is initialized at random. For a chromosome that contains K clusters, K random samples are extracted from data, and they are used as cluster centers. This is repeated as the number of chromosomes. When clustering a specific dataset, we have set the maximum number of clusters less than the square root of the number of samples [25]. The number of initial cluster K is also decided at random between 2 and the maximum number of clusters.

3.2. Fuzzy clustering

For clustering, the fuzzy c-means algorithm proposed by Ruspini has been used [26]. It 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.

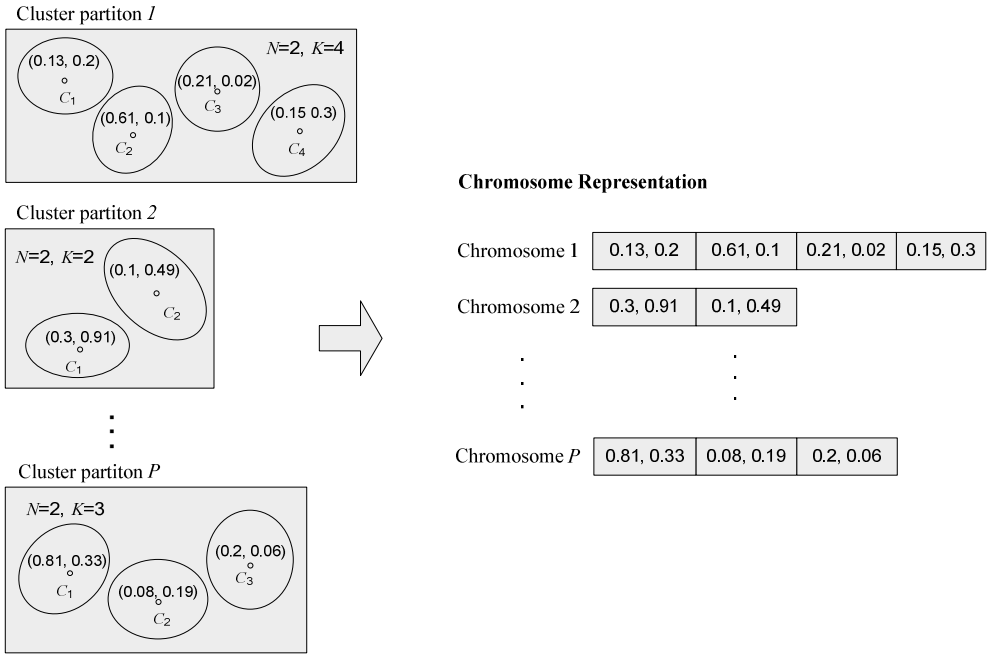


Fig. 2. Representation of the variable length chromosome.

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

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 means the degree of fuzziness of each datum's membership degree that should be bigger than 1.0 [3,12]. When it is set as 1.0, the algorithm comes to be the same as hard c-means algorithm [27].

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

– Step 1: Set c, the number of clusters, and m, the fuzziness parameter.
– Step 2: Initialize$\mu_{ij}$to satisfy Eq. (2).

$$\sum_{i=1}^{c}\mu_{ij} = 1,\ 1 \leqslant j \leqslant n \tag{2}$$

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

$$v_i = \frac{\sum_{j=1}^{n}\mu_{ij}^m x_j}{\sum_{j=1}^{n}\mu_{ij}^m} \tag{3}$$

– Step 4: Compute the membership matrix U.

$$\mu_{ij} = \frac{\left(\frac{1}{d^2(x_j, v_i)}\right)^{\frac{1}{m-1}}}{\sum_{k=1}^{c}\left(\frac{1}{d^2(x_j, v_k)}\right)^{\frac{1}{m-1}}} \tag{4}$$

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

$$|\{J_m^{(l)} - J_m^{(l-1)}\}| \leqslant \varepsilon \tag{5}$$

3.3. Fitness evaluation with Bayesian validation

After conducting clustering process, the results have been evaluated with Bayesian validation. Previous validation measures such as PC, CE, FS and XB focused only on the compactness and the variation within cluster. Those measures, however, are limited in their ability to provide a correct representation on fuzzy partition in the data because separation is computed by considering only the distance between cluster centers. Therefore, these measures lose their ability to validate fuzzy partition when the number of clusters becomes large [8,9].

Bayesian validation method is inspired by the classic Bayeian concept of probability theory, selecting a fuzzy partition with the largest membership degree given the dataset. It chooses the partition that has maximum membership degree, given the dataset as an optimal cluster partition [28].

$$\max P\ (\textit{Cluster}|\ \textit{Dataset}) \tag{6}$$

Transferring the principles of the classic Bayes' theorem to membership, we obtain

$$P(\textit{Cluster}|\ \textit{Dataset}) = \frac{P(\textit{Cluster})P(\textit{Dataset}|\ \textit{Cluster})}{P(\textit{Dataset})} \tag{7}$$

When data set X satisfies the condition $X = \{x1, x2, \ldots, xN\}$, Eq. (7) is approximated as Eq. (8) by multiplication rule and independence rule if each xi is independent one another.

$$\begin{aligned} P(\textit{Cluster}|\ \textit{Dataset}) &= P(\textit{Cluster}|x_1, x_2, \ldots, x_N) \\ &\cong P(\textit{Cluster}|x_1) \times P(\textit{Cluster}|x_2) \times \ldots \times P(\textit{Cluster}|x_N) \end{aligned} \tag{8}$$

Bayesian score (BS) is defined as the sum of all $P(\textit{Cluster}|\ \textit{Dataset})$ such as Eq. (9). We propose that this score indicates how well the fuzzy partition represents the dataset. The higher the BS is, the better the cluster partition is.

$$\begin{aligned} BS &= \frac{\sum_{i=1}^{c} P(C_i|X_i)}{C} = \frac{\sum_{i=1}^{c} P(C_i|x_{i1}, x_{i2}, \ldots, x_{iN})}{C} = \frac{\sum_{i=1}^{c} P(C_i|x_{i1})P(C_i|x_{i2}) \ldots P(C_i|x_{iN})}{C} \\ &= \frac{\sum_{i=1}^{c} \prod_{j=1}^{N_i} P(C_i)P(x_{ij}|C_i)/P(x_{ij})}{C}, X_i = \{x_{ij}|\mu_{ij} > \alpha,\ 1 \leqslant j \leqslant n\}\ ,\ N_i = n(X_i) \end{aligned} \tag{9}$$

In Eq. (9), n(Xi) is the number of Xi, and we select the samples that have larger degree of membership values than a certain probability value because BS computation includes multiplication and it produces wrong value if one of those membership degrees is zero. Besides, it is more approprate to evalute the cluster result with higher membership than a certain threshold. This threshold is defined as α-cut. Each probability is obtained as follows.

$$P(C_i) = \frac{\sum_{j=1,\mu_{ij}>\alpha}^{n} \mu_{ij}}{\sum_{i=1}^{c}\sum_{j=1}^{n} \mu_{ij}} \tag{10}$$

$$P(x_{ij}) = \sum_{i=1}^{c} P(C_i)P(x_{ij}) = \sum_{i=1}^{c} P(C_i)\mu_{ij} \tag{11}$$

When the membership matrix is produced as the 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 $P(x_{ij}|C_i)$. Figure 3 illustrates the overall process of Bayesian validation method where X_1 is the set of samples that belong to c_1 and satisfy the condition of $\mu_{ij} > \alpha$. Cluster results are evaluated using the final BS value.

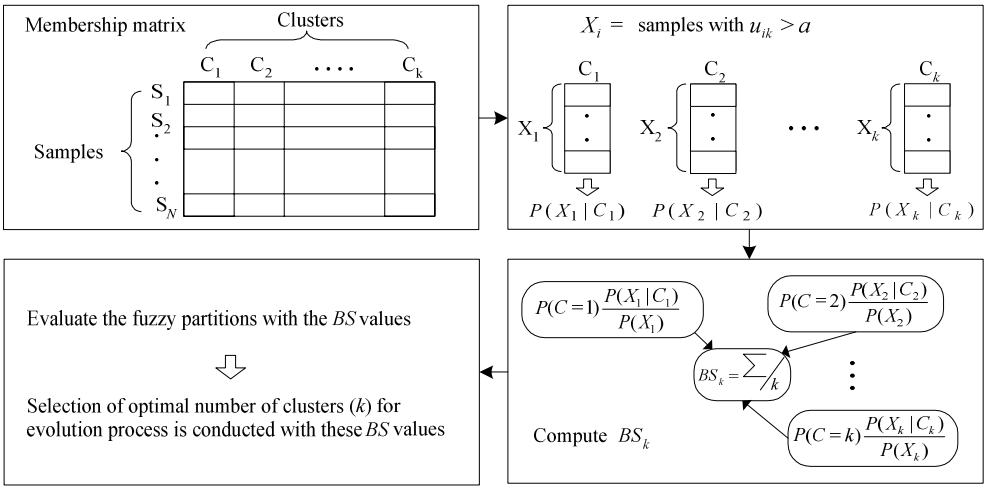


Fig. 3. The process of Bayesian validation method.

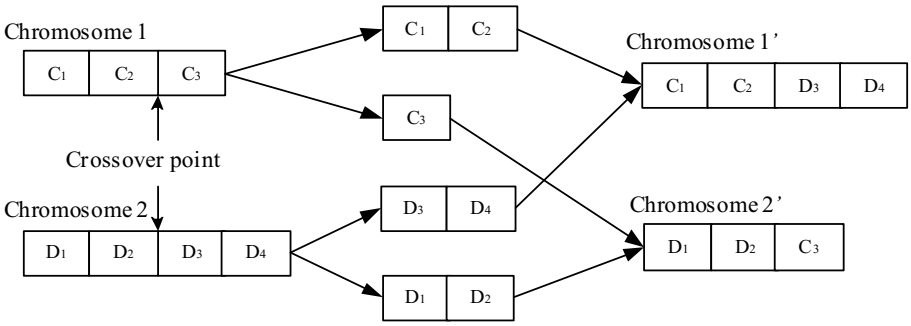


Fig. 4. An example of crossover.

The algorithm of Bayesian validation method is summarized as follow:

- Step 1: Compute the membership matrix U_{ij}.
- Step 2: Construct X_i by selecting samples ($\mu_{ij} > \alpha$) in each cluster.
- Step 3: Compute $P(X_j|C_j)$, $P(X_j)$, and $P(C_j)$ of X_i.
- Step 4: Compute BS using the calculated values in step 2.
- Step 5: Evaluate the fuzzy partition with the maximum value of BS as an optimal partition.

Selection is conducted with these evaluation results. For selection, we have used a roulette wheel strategy that tries to select many copies of individuals corresponding to its fitness [29].

3.4. Crossover and mutation

The proposed method cannot use general crossover operation because the length of chromosome is variable, so crossover operation is performed as shown in Fig. 4. 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 occurred by the fixed mutation rate. Since this paper adopts the floating point representation, mutation is occurred by Eqs (12) and (13). 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 determined as Eqs (12) and (13) [6].

$$\nu \pm 2 \times \delta \times \nu, \ \nu \neq 0 \tag{12}$$

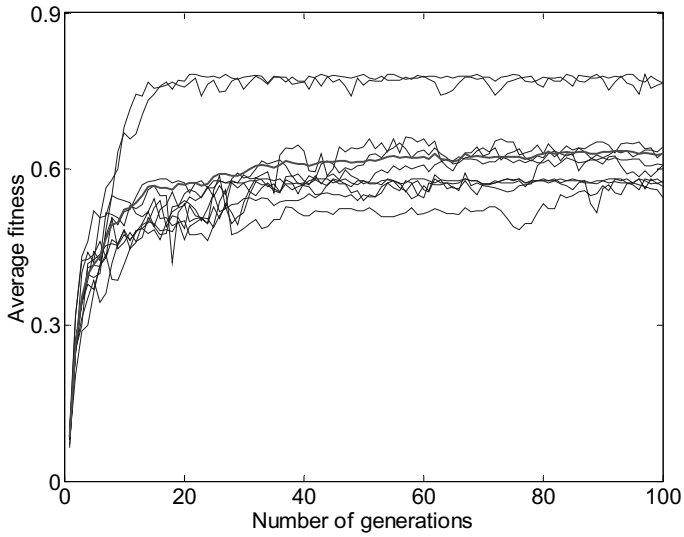


Fig. 5. Average fitness transition (SRBCT dataset, $P = 100$).

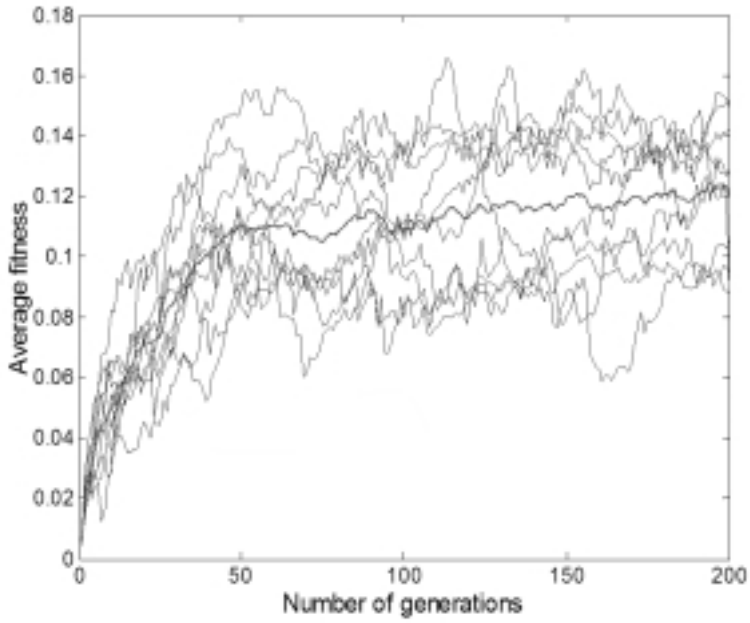


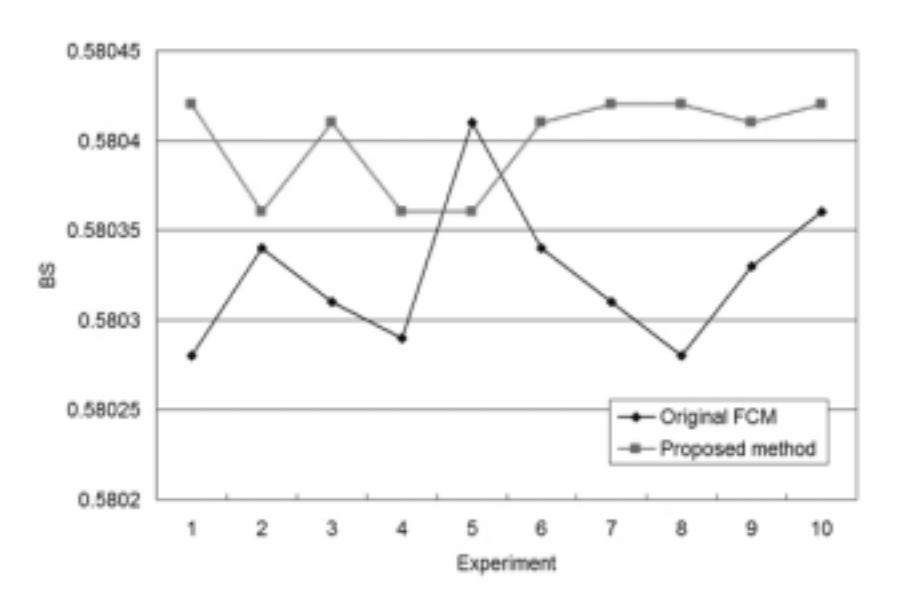
Fig. 6. Average fitness transition (Saccharomyces cell-cycle dataset, $P = 200$).

$$\nu \pm 2 \times \delta,\ \nu = 0 \tag{13}$$

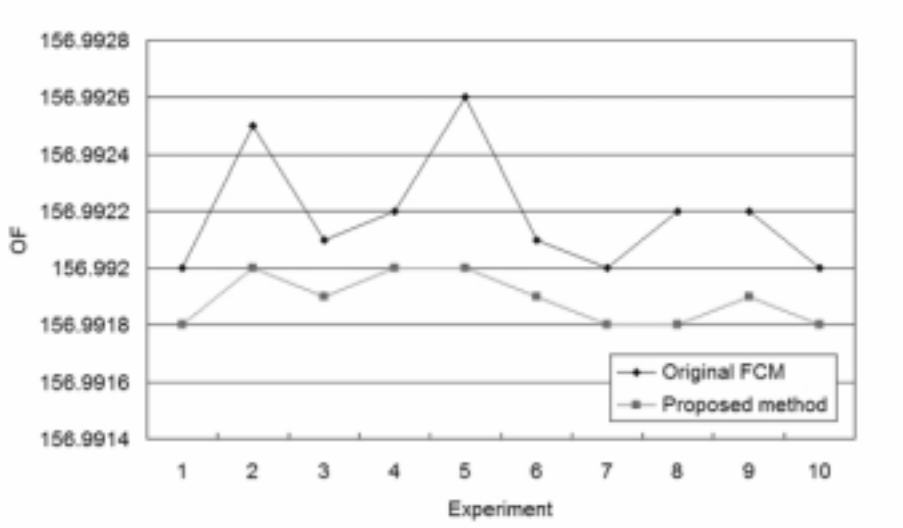
Equation (13) is used if ν is zero, otherwise Eq. (12) is used. The probabilities of sign '$+$' and '$-$' are the same.

4. Experiments

In this paper, thorough experiments are conducted. First, the experiments for an optimal cluster partition search are conducted using evolutionary clustering method. After that, the performance comparisons of the original fuzzy



(a) Comparison using BS



(b) Comparison using OF value

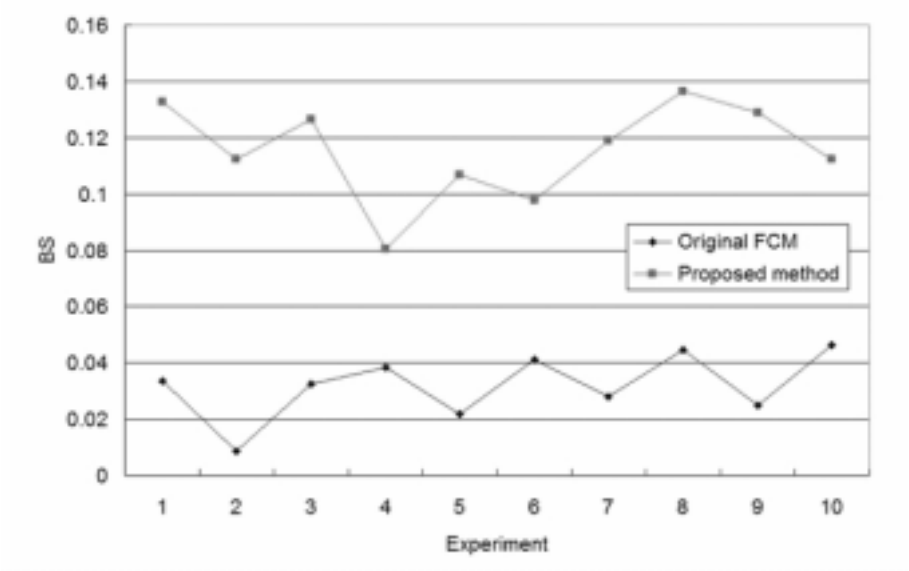
Fig. 7. Comparison between original FCM and the proposed method (SRBCT).

c-means and evolutionary clustering algorithms are performed. Comparisons of BS and PC, one of the most widely used fuzzy cluster validity measures [8], as a fitness evaluation measure followed, and then comparison of the proposed method and hard c-means algorithm was conducted. Finally, the analyses of Saccharomyces cell cycle gene expression data are provided.

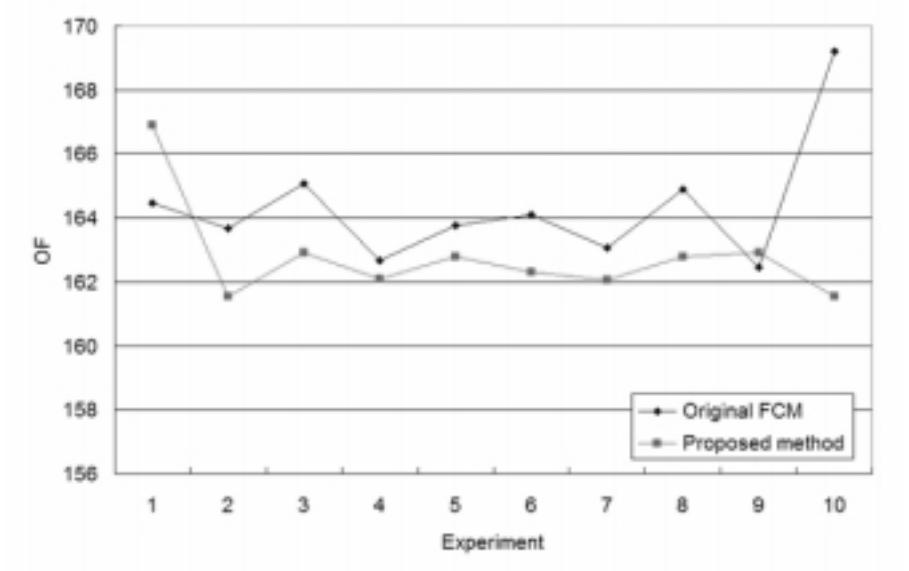
4.1. Experimental environment

4.1.1. Experimental data

The description of SRBCT and Saccharomyces cell cycle expression datasets are as follows.



(a) Comparison using BS



(b) Comparison using OF value

Fig. 8. Comparison between original FCM and the proposed method (Saccharomyces cell-cycle).

– SRBCT dataset: This has 63 samples with 6567 genes and consists of 4 classes, NB (neuroblastoma), RMS (rhabdomhosarcoma), NHL (non-Hodgkin lymphoma) and EWS (Ewing family of tumors). They are kinds of cancer, and each of them has different characteristics. This paper has clustered 63 samples (with 96 attributes) that are known as informative ones [30].
– Saccharomyces cell-cycle dataset: This is a dataset that has expression levels of 6000 genes expressed during 2 cell cycles. Expression levels are measured on 17 different time points every 10 minutes. This dataset is frequently used for genetic analysis since the genes classified by their biological function have different

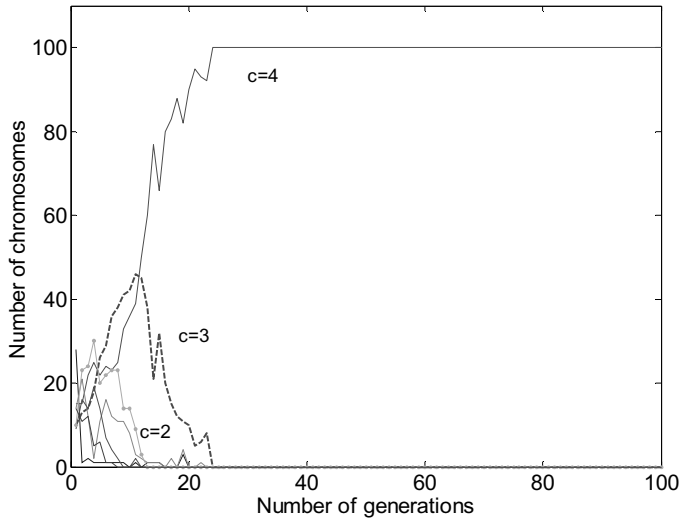


Fig. 9. Change of the number of chromosomes with the specific number of clusters when BS is used for fitness evaluation (SRBCT dataset, $P = 100$).

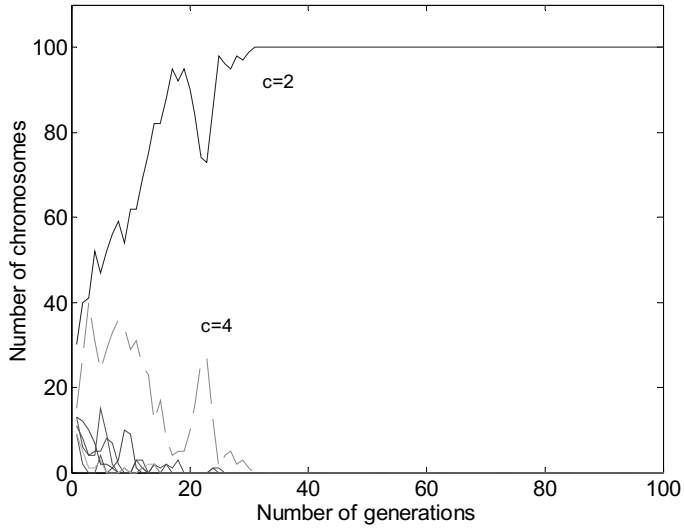


Fig. 10. Change of the number of chromosomes with the specific number of clusters when PC is used for fitness evaluation (SRBCT dataset, $P = 100$).

expression levels according to cycle. 421 genes (with 17 attributes) that show significant change of expression levels are clustered in this paper [31].

4.1.2. Parameters and settings

For Bayesian validation method, the α-cut value of 0.2 and 0.4 are used for SRBCT and Saccharomyces cell cycle datasets. These values have been decided considering the membership degrees of clustered results. Since the samples in SRBCT dataset have small values, 0.2 is selected for SRBCT, and 0.4 is selected for Saccharomyces dataset because its samples have relatively higher membership degrees.

Equation (14) provides the equation of PC as a comparable measure with BS. Here, μ_{ij} is a degree of membership value, and n and c means the number of samples and clusters, respectively. Here, the closer the value of PC is to one, the better clusters are formed.

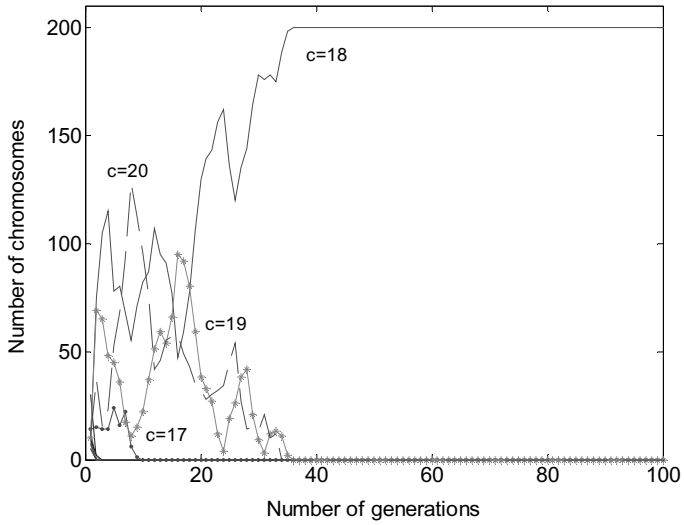


Fig. 11. Change of the number of chromosomes with the specific number of clusters when BS is used for fitness evaluation (Saccharomyces cell-cycle dataset, $P = 200$).

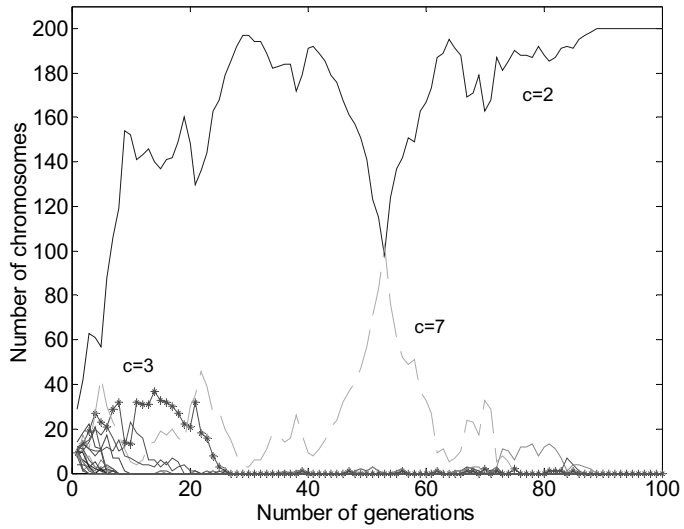


Fig. 12. Change of the number of chromosomes with the specific number of clusters when PC is used for fitness evaluation (Saccharomyces cell-cycle dataset, $P = 200$).

$$PC(U;c) = \frac{\sum_{j=1}^{n}\sum_{i=1}^{c}(\mu_{ij})^2}{n} \tag{14}$$

For evolutionary clustering, maximum generation number is set as 1000, and population sizes of 100 and 200 are used for SRBCT and Saccharomyces cell-cycle datasets, respectively. The size of SRBCT dataset is smaller than Saccharomyces cell-cycle dataset. Maximum numbers of clusters are 8 and 20 for SRBCT and Saccharomyces 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 is set as 1.2 referring Dembele's work [14].

Table 1
fuzzy gene and the membership degrees

Fuzzy gene	The membership degrees to the first & second clusters
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	0.60490 (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	0.46230 (0), 0.36860 (1)
YLL040c	0.44400 (0), 0.34000 (1)
YML110C	0.59168 (1), 0.32380 (0)
YDL119c	0.48350 (0), 0.38849 (2)
YBR158w	0.58690 (1), 0.40988 (2)
YDL179w	0.55259 (1), 0.43413 (2)
YIL009W	0.60611 (1), 0.35258 (2)
YNL046W	0.55810 (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	0.58270 (12), 0.41445 (10)
YMR179W	0.56833 (10), 0.40397 (12)

4.2. Results and analyses

4.2.1. Optimal cluster partition search

Figures 5 and 6 provide the average fitness transition graphs. Figure 5 illustrates one of SRBCT dataset according as generation grows. Experiments have been repeated 10 times, and bold line is the average of them. In SRBCT dataset, it evolves rapidly until the generation number is close to 20, and then converges into about 0.6. Figure 6 illustrates the average fitness transition of Saccharomyces cell-cycle dataset. It converges more slowly than SRBCT dataset and average fitness changes slowly until the 80th generation. It converges into 0.12 with the large oscillation.

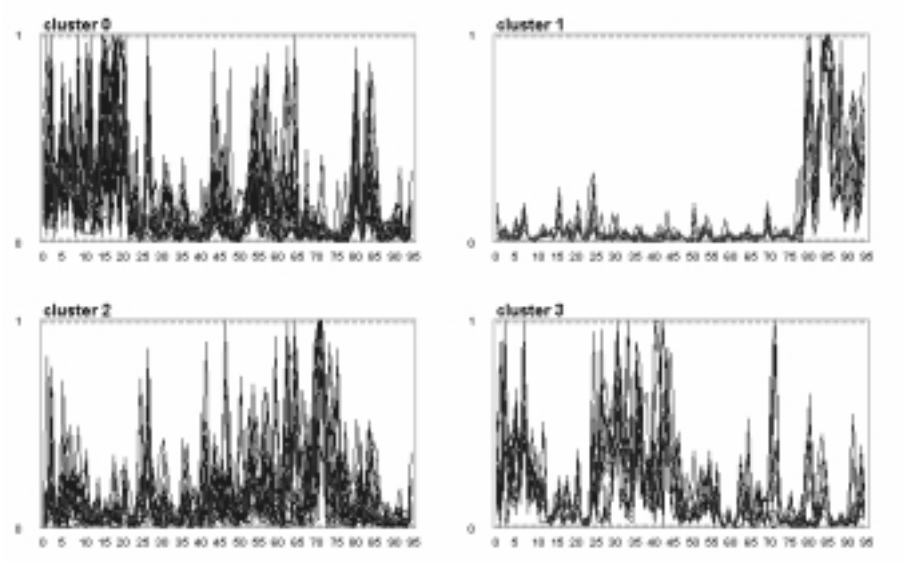
Figures 5 and 6 show different transition patterns, and this can be thought that different characteristics of the datasets had influence on the evolution process. In the fuzzy cluster results, it can be known that most genes of SRBCT dataset have the membership degrees that are larger than 0.9 or smaller than 0.1. Those of Saccharomyces cell-cycle dataset, on the other hand, have various ranges.

4.2.2. Comparison with the original fuzzy c-means algorithm

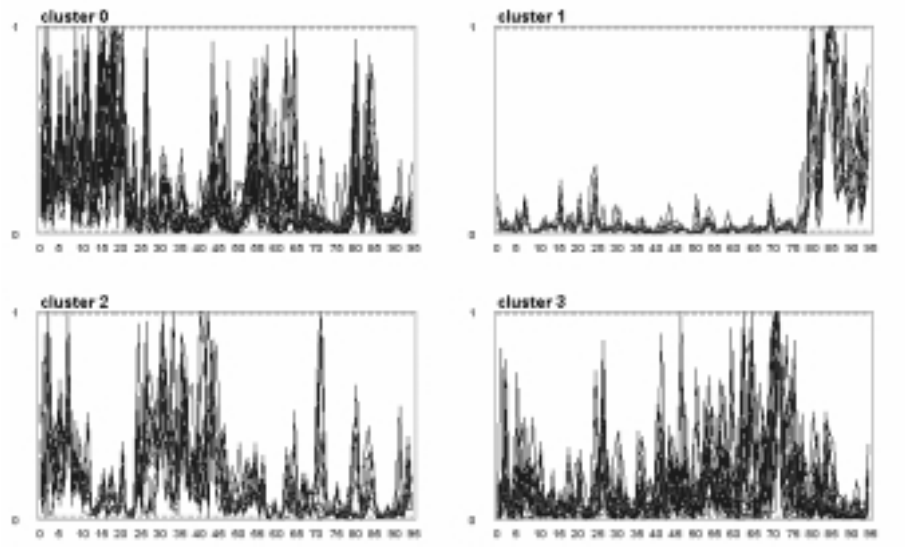
In this section, we have compared the result of the proposed method with the original FCM by means of Bayesian score (BS) and the objective function value (OF) of the FCM. Figure 7 summarizes 10 experimental results of SRBCT dataset. If BS is high and the objective function value is low, it means that the cluster results are good because the objective function value is based on the distances between cluster centers and samples as explained in Eq. (1). The proposed method provides better results than original FCM in both BS and OF value.

Figure 8 summarizes 10 experimental results of Saccharomyces cell-cycle dataset. In both datasets, the result of the proposed method was better even though the difference was not significant in SRBCT dataset. The result of Saccharomyces cell-cycle dataset, however, shows relatively significant difference.

We have confirmed that the result of the proposed method is closer to the optimal solution than one of the original FCM comparing the proposed method with the original FCM.



(a) Clustered result of proposed method



(b) Clustered result of HCM algorithm

Fig. 13. Comparison with HCM algorithm (Number of clusters: 4).

4.2.3. Comparison of BS and PC as a fitness evaluation measure

Figures 9 through 12 demonstrate changes of the number of chromosomes with the specific numbers of cluster during the evolution process. They show the optimal number of clusters when BS and PC are used for fitness evaluation.

Figures 9 and 10 are the results of SRBCT dataset. Figure 9 is the result when BS is used for fitness evaluation. There are various cluster partitions when the number of generations is small, but only chromosomes of $c = 4$ survive after the 25th generation. Here, we can confirm that the proposed method searches the optimal cluster partition because SRBCT dataset has 4 classes. Figure 10 is the result of SRBCT dataset when PC is used for fitness

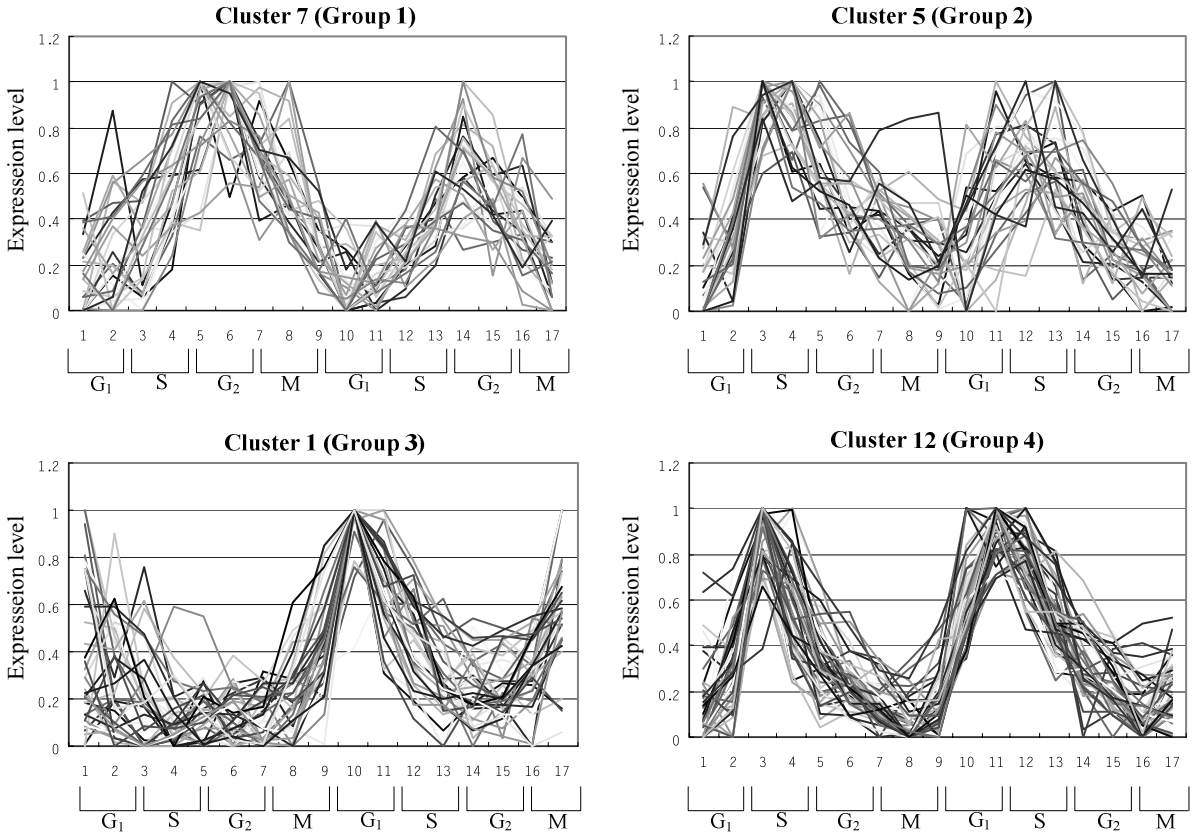


Fig. 14. Change of expression level of representative clusters in four groups over cell-cycle.

evaluation. In this case, chromosomes of $c = 2$, which is wrong cluster partition, have finally survived because fitness evaluation was not correct.

Figures 11 and 12 are the results where Saccharomyces cell-cycle dataset is applied to the same experiments as the previous one. When BS is used for fitness evaluation, the chromosomes of $c = 17$, $c = 18$, $c = 19$, and $c = 20$ compete, and the case of $c = 18$ survives finally. When PC is used, the case of $c = 2$ survives. The number of clusters of Saccharomyces cell-cycle dataset in previous research is various. It has been analyzed from 4 or 5, which is the number of phase in one cell-cycle, through $17 \sim 20$ to more than 30. As it is considered that Saccharomyces cell-cycle dataset consists of expression levels measured from 17 different time points, 18 is significant number comparing with 2 particularly.

We have used the known number of classes (in SRBCT dataset) and the number of phases in one cell-cycle (in Saccharomyces dataset) as a correct number of clusters for evaluation. Even though they may not always provide the correct number of clusters, the results using BS are better than one using PC because 2 is definitely not an optimal number of clusters.

4.2.4. Comparison with the hard c-means Algorithm

With four clusters of SRBCT dataset selected in last section, we have compared the result of proposed method with the one of the hard c-means (HCM) algorithm. If the number of cluster is fixed, proposed method is the same as FCM. As illustrated in Fig. 13, the results are almost the same each other except cluster 2 and cluster 3 are changed each other in Fig. 13(b). Here, y axis represents the expression level, and x axis represents the number of attributes, and each cluster has its own pattern. Even though these results are very similar, the result of fuzzy clustering have more information than one of hard clustering because data can belong to several clusters with the own membership

Table 2
Fuzzy genes and their gene descriptions and cluster numbers

Fuzzy gene	Gene description	Clusters
YBL032w	weak similarity to hnRNP complex protein homology YBR233w	4, 7
YHR031C	similarity to Pif1p	4, 7
YCL063w	weak similarity to yeast translation regulator Gcd6p	7, 11
YBR007c	hypothetical protein	5, 15
YER019w	hypothetical protein	5, 15
YDR297w	suppressor of rvs161 and rvs167 mutations	5, 13
YER118c	involved in the HOG1 high-osmolarity signal transduction	5, 13
YHR173C	hypothetical protein	5, 13
YLL021w	forms schmoos poorly	5, 13
YBR275c	Rap 1p interacting factor. telomere silencing	5, 12
YJL173C	DNA replication factor A, 13 kDa subunit	5, 12
YBR053c	similarity to rat regucalcin	0, 1
YKL163W	protein with internal repeats	0, 1
YLL040c	involved in regulating membrane traffic	0, 1
YML110C	ubiquinone biosynthesis, methyltransferase	0, 1
YDL119c	similarity to bovine Graves disease carrier protein	0, 2
YBR158w	hypothetical protein	1, 2
YDL179w	cyclin like protein interacting with Pho85p	1, 2
YIL009W	acyl CoA synthase	1, 2
YNL046W	hypothetical protein	1, 2
YOR 264W	hypothetical protein	1, 2
YDL127w	cyclin, G1/S-specific	10, 12
YJL187C	phosporylates Cdc28, SPB separation, nuclear division	10, 12
YMR078C	for chromosome transmission in mitosis and normal telomere	10, 12
YMR179W	required for normal transcription at a number of loci	10, 12

degrees in fuzzy clustering while they belong to only one cluster in hard clustering. This characteristic of fuzzy cluster results will be analyzed in the next section with Saccharomyces dataset.

4.2.5. Analyses of saccharomyces cell cycle dataset

We have analyzed the result of Saccharomyces cell-cycle dataset as comparing it with the known genes at Cho's work [31]. We have focused on fuzzy genes, which have membership degrees higher than 0.3 and belong to several clusters simultaneously. Table 1 summarizes the membership degrees and the cluster numbers of fuzzy genes. The number in the right side represents the cluster number.

We have categorized fuzzy genes into 4 groups. First three genes, YBL032w, YHR031C and YCL063w, are members of cluster 4, cluster 7 and cluster 11, respectively, and these are one group. Another group including cluster 5, cluster 12, cluster 13 and cluster 15 is grouped on cluster 5. A group of cluster 0, cluster 1 and cluster 2 and a group of cluster 10 and cluster 12 are the remaings.

Compared these fuzzy genes with known functions of Saccharomyces cell-cycle dataset, discovered information is summarized in Table 2. 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 also presented in Table 2. In this table, 'blue' and 'red' colors mean the expression level is close to 0 and 1 when it is normalized, respectively.

Figure 14 shows the transitions of expression levels of 4 clusters, which are selected by model of each group, according to cluster numbers. Cluster 7 has 26 genes, and they express the most at G2 phase of cell cycle, so cluster 7 is thought to be related to G2 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 highly at G1 phase, and cluster 12 of the last group expresses the most between G1 and S phases. Considering that genes of cluster 12 were grouped with cluster 5 in Table 2, it is also related to the second group.

5. Conclusions

This paper has proposed an evolutionary fuzzy clustering method with Bayesian validation. Evolutionary fuzzy clustering algorithm is to search optimal cluster partition, and Bayesian validation method is to evaluate the fitness. Applying the proposed method to DNA microarrays of SRBCT and Saccharomyces cell cycle datasets, the results have shown the better performance than the results of conventional methods. Finally, we have analyzed the optimal cluster partition of Saccharomyces cell cycle expression data searched by the proposed method.

Future research will include more comparison experiments with other cluster validity measures except the one with the distance measure. Also, the theoretical proof of Bayesian validation will be needed.

Acknowledgements

This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the Biometric Engineering Research Center (BERC) at Yonsei University.

References

[1] N. Belacel et al., Fuzzy j-means: a new heuristic for fuzzy clustering, *Pattern Recognition* **35** (2002), 2193–2200.
[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** (1999), 6745–6750.
[3] N. Bolshakova and F. Azuaje, Cluster validation techniques for genome expression data, *SIGPRO* **21**(82) (2002), 1–9.
[4] D. Fogel and P. Simpson, *Evolving Fuzzy Clusters*, Proc. Int. Conf. on Neural Networks, 1993.
[5] L.O. Hall et al., Clustering with a genetically optimized approach, *IEEE Trans on Evolutionary Computation* **3**(2) (1999), 103–112.
[6] U. Maulik and S. Bandyopadhyay, Genetic algorithm-based clustering technique, *Pattern Recognition* **33** (2000), 1455–1465.
[7] J.N. Bhuyan, V.V. Raghavan and V.K. Elayavalli, *Genetic Algorithm for Clustering with an Ordered Representation*, Proc of the 4th Int. Conf. Genetic Algorithms, 1991.
[8] N.R. Pal and J.C. Bezdek, On cluster validity for the fuzzy c-means model, *IEEE Trans. on Fuzzy Systems* **3**(3) (1995), 370–379.
[9] M.R. Rezaee et al., A new cluster validity index for the fuzzy c-means, *Pattern Recognition Letters* **19** (1998), 237–246.
[10] M.E. Futschik, A. Reeve and N. Kasabov, Evolving connectionist systems for knowledge discovery from gene expression data of cancer tissue, *Artificial Intelligence in Medicine* **28** (2003), 165–189.
[11] 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* **95** (1998), 14863–14868.
[12] F. Hoppner et al., *Fuzzy Cluster Analysis*, Wiley, 1999, 43–39.
[13] A.P. Gasch and M.B. Eisen, Exploring the conditional coregulation of yeast gene expression through fuzzy k-means clustering, *Genome Biology* **3**(11) (2002), 1–22.
[14] D. Dembele and P. Kastner, Fuzzy c-means method for clustering microarray data, *Bioinformatics* **19**(8) (2003), 973–980.
[15] N. Belacel et al., Fuzzy j-means and VNS methods for clustering genes from microarray data, *Bioinformatics* **20**(11) (2004), 1690–1701.
[16] D.W. Kim et al., Fuzzy cluster validation index based on inter-cluster proximity, *Pattern Recognition Letters* **24** (2004), 2561–2574.
[17] J.C. Bezdeck, Cluster validity with fuzzy sets, *Journal of Cybernetics* **3**(3) (1974), 58–72.
[18] L. Chamber, *Practical Handbook of Genetic Algorithm*, CRC Press, 1995.
[19] J.N. Bhuyan et al., *Genetic Algorithm for Clustering with an Ordered Representation*, Proc. 4th Int. Conf. Genetic Algorithms, 1991, 408–415.
[20] S. Bandyopadhyay and U. Maulik, An evolutionary technique based on k-means algorithm for optimal clustering in R^n, *Information Sciences* **146** (2002), 221–237.
[21] A.D. Nola et al., An evolutionary approach to spatial fuzzy c-means clustering, *Fuzzy Optimization and Decision Making* **1** (2002), 195–219.
[22] O. Nasraoui, E. Leon and R. Krishnapuram, *A Novel Approach to Unsupervised Robust Clustering Using Genetic Niching*, Proc. of the 9th IEEE Int. Conf. on Fuzzy Systems, 2000, 170–175.
[23] O. Nasraoui, E. Leon and R. Krishnapuram, Unsupervised Niche Clustering: Discovering an unkown number of clusters in noisy data sets, in: *Invited chapter in "Evolutionary Computing in Data Mining"*, A. Ghosh and L.C. Jain, eds, Springer Verlag, 2005.
[24] C.-Y. Lee and E.K. Antonsson, *Dynamic Partitional Clustering Using Evolution Strategies*, In 3rd Asia Pacific Conf. on Simulated Evolution and Learning, 2000, 2716–2721.
[25] D.W. Kim et al., Fuzzy cluster validation index based on inter-cluster proximity, *Pattern Recognition Letters* **24** (2003), 2561–2574.
[26] E.H. Ruspini, A new approach to clustering, *Information and Control* **15** (1969), 22–32.
[27] J. McQueen, *Some Methods for Classification and Analysis of Multivariate Observations*, (Vol. 1), Proc. 5th Berkeley Symposium on Mathematical Statistics and Probability, 1967, 281–297.
[28] Y. Barash and N. Friedman, Context-specific Bayesian clustering for gene expression data, *Journal of Computational Molecular Cell Biology* **9**(2) (2001), 12–21.

[29] K. Krishna and M.N. Murty, Genetic k-means algorithm, *IEEE Trans. on Systems, Man and Cybernetics* **20**(3) (June 1999).
[30] J. Khan et al., Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks, *Nature* **7**(6) (June 2001), 673–679.
[31] R.J. Cho et al., A genome-wide transcriptional analysis of the mitotic cell cycle, *Molecular Cell* **2** (1998), 65–73.