Fuzzy Clustering of Macroarray Data

Olga Georgieva[1], Frank Klawonn[2], and Elizabeth Härtig[3]

[1] Institute of Control and System Research
Bulgarian Academy of Sciences
P.O. Box 79, 1113 Sofia, Bulgaria
ogeorgieva@icsr.bas.bg

[2] Department of Computer Science
University of Applied Sciences Braunschweig/Wolfenbuettel
Salzdahlumer Str. 46/48
D-38302 Wolfenbuettel, Germany
f.klawonn@fh-wolfenbuettel.de

[3] Institute for Microbiology
Technical University of Braunschweig
Spielmannstr. 7
D-38106 Braunschweig, Germany
e.haertig@tu-bs.de

1 Introduction

The complete sequence of bacterial genomes provides new perspectives for the study of gene expression and gene function. DNA array experiments allow measuring the expression levels for all genes of an organism in a single hybridization experiment.

Computational analysis of the macroarray data is used extensively to extract groups of similarly expressed genes. The aim is to organize DNA array data so that the underlying structures can be recognized and explored. The gene groups identified as clusters are searched for genes known to be involved in similar biological processes, implying that genes of unknown functions may be involved in the same processes. Commonly used computational techniques include hierarchical clustering, K-means clustering and self-organizing maps. These share many features, particularly the distance metric, which measures the relationship between samples or genes in the data space formed by the expression values. The output from different clustering algorithms usually depends more on the type of distance metric used than on any other factor.

The central limitation of most of the commonly used algorithms is that they are unable to identify genes whose expression is similar to multiple, distinct gene groups, thereby masking the relationships between genes that are coregulated with different groups of genes in response to different conditions. For example, the K-means clustering partitions genes into a defined set of discrete clusters, attempting to maximize the expression similarity of the genes in each cluster assigning each gene to only one cluster, obscuring the relationship between the conditionally coregulated genes. The recently implemented heuristic variant of Fuzzy C-Means (FCM) clustering [4] shows the advantage of the fuzzy clustering technique as a valuable tool for gene expression analysis as it presents overlapping clusters, pointing to distinct features of each gene's function and regulation.

The paper presents a methodology to deal with macroarray data analysis. Each step of the data processing is described in detail. First, the crucial preprocessing step on the raw data to transform the data set into an appropriate and reliable form for clustering is applied. Secondly, subtractive clustering is implemented in order to obtain a good initial data partition. The obtained cluster centres are used to initialize the fuzzy C-means clustering algorithm by which the optimal values of the cluster centres and partition matrix are obtained. The partition matrix is used to determine the gene groups that belong to a given cluster with a prescribed membership degree. The proposed procedure is applied to macroarray data of *B. subtilis*. The obtained results show that informative clusters are obtained. The extracted gene clusters are overlapping, pointing to distinct aspects of the gene function and regulation.

2 Data set

Here we used data obtained from the soil bacterium *B. subtilis* grown under different growth conditions. *B. subtilis* is able to grow in the absence of oxygen using nitrate as alternative electron acceptor or fermentation processes. The switch between aerobic and anaerobic metabolism in *B. subtilis* is regulated mainly at the transcriptional level. To investigate the global changes in gene expression under various anaerobic conditions we used DNA macroarrays containing DNA fragments of all 4107 genes of *B. subtilis*. We analysed mRNA from cells grown aerobic, anaerobic with nitrate, anaerobic with nitrite and under fermentative conditions. When the mRNA levels were compared during exponential growth, several hundred genes were observed to be induced or repressed under the various conditions tested. The data of the macroarrays are obtained by autoradiography using phosphorimaging and the intensities representing the expression levels are transformed into a table. In the obtained numerical table each row corresponds to a gene and each column to one growth condition analyzed.

The considered macroarray data set is a data matrix that consists of the expression levels (ratios) of 4107 genes of *B. subtilis* defining each matrix raw. The cells have been carried out on four different environment condition namely aerobic (A), fermentative (B), nitrite (C) and nitrate (D). The intensities obtained in each experiment are organized in the columns. The ratio in each environment condition has been measured twice. Thus, the original macroarray matrix $Z = (z_{kj})$ $(k = 1, \ldots, N, j = 1, \ldots, n)$ is a large sized data matrix with $N = 4107$ rows and $n = 8$ columns.

Two distinct data sets of the type described above have been obtained in different time instants. The first data set, named identification data set, was used for data partition and determination of the searched gene groups. The second one, named the validation data set, was used only for verification of the obtained clusters.

3 Fuzzy clustering of *B. subtilis* macroarray data

In order to identify the gene clusters and simultaneously to evaluate the level of relationships of genes that are coregulated with different groups, the Fuzzy C-Means clustering technique [2, 5] is applied. It is an objective function-based clustering method. It aims at minimizing an objective function that indicates a kind of fitting error of the clusters to the data. The underlying objective function for most of the clustering algorithms is:

$$J = \sum_{i=1}^{c} \sum_{k=1}^{N} (u_{ik})^m d_{ik}^2 \tag{1}$$

where N is the number of data points; c is the number of clusters; u_{ik} and d_{ik} denote the membership degree and the distance of the point x_k, $k = 1, \ldots, N$, to the i-th cluster prototype (centre), $i = 1, \ldots, c$, respectively, $m \in [1, \infty)$ is the weighted exponent coefficient (fuzzifier) which determines how much clusters may overlap. In order to avoid the trivial solution assigning no data to any cluster, i.e. setting all u_{ik} to zero, and to avoid empty clusters, the constraints

$$u_{ik} \in [0, 1], \qquad 1 \leq i \leq c, \quad 1 \leq k \leq N \tag{2}$$

$$\sum_{i=1}^{c} u_{ik} = 1, \qquad 1 \leq k \leq N \tag{3}$$

$$0 < \sum_{k=1}^{N} u_{ik} < N, \qquad 1 \leq i \leq c \tag{4}$$

are introduced. When the fuzzifier value $m = 1$ is chosen, then $u_{ik} \in \{0, 1\}$ will hold at a minimum of the objective function (1), i.e. the resulting partition will be crisp.

The parameters to be optimized are the membership degrees u_{ik} and the cluster parameters which finally determine the distance values d_{ik}. In the simplest case, a single vector named cluster centre represents each cluster v_i. For FCM clustering the distance of a data point to a cluster is simply the Euclidean distance between the cluster centre v_i and the corresponding data point:

$$d_{ik}^2 = \| x_k - v_i \|^2 = (x_k - v_i)^\top (x_k - v_i), \tag{5}$$

where $x_k = [x_{k1}, \ldots, x_{kn}]$ is the k-th data point defined as a vector in the feature space and $v_i = [x_{i1}, \ldots, x_{in}]$ is the i-th cluster prototype vector.

The minimization of the functional (1) represents a nonlinear optimization problem that is usually solved by means of Lagrange multipliers, applying an alternating optimization scheme [2]. This optimization scheme alternatively considers one of the parameter sets, either the membership degrees or the cluster parameters as fixed, while the other parameter set is optimized, until the algorithm finally converges.

The main problem of the macroarray data clustering is the badly structured data space that lacks well separated (distinguished) data groups. Most of the points are found to be located in one small area close to the zero value. Thus, two important problems arise in applying the FCM clustering algorithm. The first one arises from the specific characteristics of the data set mentioned above. The data set should be presented in an appropriate form in order to guarantee the reliability and authenticity of the extracted information. The second problem concerns the initialization of the FCM algorithm. In the objective function (1), the number of clusters has to be fixed in advance. Since the number of clusters is usually unknown, an additional scheme has to be applied to determine the number of clusters and their prototypes. Both problems are considered in detail and an effective solution for the macroarray data set is described.

3.1 Preprocessing step

The preprocessing step is a preliminary step of the whole data processing (clustering) which aimed to transform the data set into an appropriate and reliable form for clustering. In most situations, the data obtained after scanning must be transformed and normalized before they can be analyzed. By applying several operations on the identification data set the data representation is improved and the quality of the clustering is increased.

One main problem of the macroarray data processing is connected to the data noise. Differences occurred quite frequently in the repeated measurements of the same gene under the same conditions. For this, an average value of the gene expression within every environment condition is determined. In case one of the expressions is zero or invisible (no measured value is provided) the other one is taken into account. Thus, in the considered particular case the data matrix is transformed from an eight to a four column matrix. This in practice means that the clusters will be searched for in a four-dimensional data space.

Since empirical evidence shows that low-intensity spots (low gene expression measurements) are more likely to be noisy, eliminating those spots is a fairly safe option [6]. However, as the most of the data values are close to the zero value it is rather difficult to distinguish the noisy values from the informative ones. That is why only genes that have one or more zero average expression value are taken out from the data set. Eleven genes having a zero average value and by that three of them at two environment conditions were defined and removed from the considered data set.

The second specificity of the illuminated macroarray matrix is that there are so bright spots that some of the surrounding ones could not be distinguished. Their expression levels can not be measured and they are considered as invisible. Genes that have at least one invisible average expression should also be removed from the data set. However, in the particular case under consideration it two invisible expressions for one gene in one condition never occurred.

The upper extreme of the filtered group consists of genes that have large expression that differ drastically from the rest of the data. Normally, they form a small group that is far away from the other data in the data space. In this situation every clustering algorithm will separate both the large and small group of outliers, but this partition will be not informative as the important clusters are within the large amount and narrow spread data points. Genes that have an average expression of over 100 are taken out from the data set. Both cutting values for lower and upper data filtering are subjectively determined. They are subject to an expert choice depending on the concrete data configuration.

The results in many DNA macroarray experiments are ratios. The fact that the output is not symmetric, i.e. twofold change has a ratio of either 2 (up-regulation) or 0.5 (down- regulation), presents a problem for analysis because most distance metrics treat a ratio change differently depending on the direction. Thus, it is essential to convert ratio data into a form that is not sensitive to the direction of the change. The solution of this problem usually applied is a log transform of all ratio values [6]. Here, the natural logarithm was applied but all logarithm bases are equivalent for this task. By this transformation the high ratios are compressed while the small ratios increase their importance by stretching their distances. This effect is very beneficial as it increases the distinguishability among the data due to the fact that the largest amount of expression ratios lies between zero and one.

Generally, the expression ratios are obtained by a comparison to a common reference sample. For each gene the series of ratio values are relative to the expression level of that gene in the reference sample. Thus, in case the uncentred correlation matrix is used, two genes whose expression patterns are identical to each across a range of samples, but different in the reference sample, will not cluster together. Since the reference sample has nothing to do with the experiment, the expression levels should be transformed to be independent from the reference sample. This problem can be solved by subtracting the average (mean) or median log ratio for a gene from the log value for each value for that gene. Median centring the data for columns and rows is applied consequently ten times to the processed identification data set. This transformation removes certain types of biases.

The applied FCM clustering method uses the Euclidean distance metric, which imposes data normalization before clustering in order to ensure reliable clustering results.

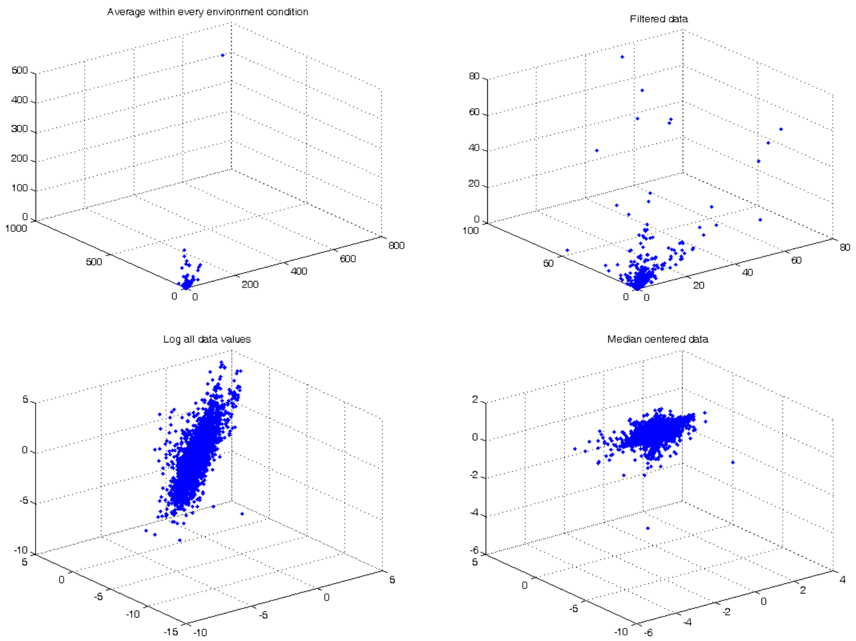

Fig. 1. Data transformations: averaging, filtering, log transform, median centred data

These operations are not associative, so the order in which they are applied is very important. The series of operations for the raw data are:

1. Average of the expression values within every environment condition $Z_{N\times 8} \rightarrow Z_{N\times 4}$;
2. Filtering by removing: $z_{ki} > 100$ (to high expression)
   $z_{ki} = 0$ (no expression)
   $z_{ki} = -1$ (invisible expression)
3. Log transform all values: $\log(Z_{N_1\times 4}) = (\log z_{ki})$, $k = 1,\ldots,N_1$, $i = 1,\ldots,4$, where N_i is the revised number of the genes in the data set;
4. Median centring through data columns and rows consequently ten times;

5. Normalization of the data.

Figure 1 shows the data transformations carried out by the described procedure in the three (A-B-C) dimensional data space.

The data preprocessing does not change the structure of the data set. It only makes it more reliable and informative for the applied clustering algorithm.

3.2 Initialization of the clustering algorithm

The number of clusters and the initial partition matrix should be provided in advance for all objective function-based clustering algorithms. Since, by the clustering optimization scheme usually a local minimum is found, the provided initialization parameters are of a major importance, especially in the case of extreme large data set. An effective solution can be achieved only, if the correct number of clusters and a suitable initial data partition are defined.

The main drawback of the fuzzy clustering algorithm is the lack of an initialization procedure. There is no reliable technique for determining the number of clusters and an initial data partition. In the standard objective function-based clustering additional strategies have to be applied in order to define the number of clusters. There are two commonly applied strategies. Through the first one the clustering is carried out with different numbers of clusters and the partition is evaluated by some global validity measure like average within-cluster distance, fuzzy hypervolume or average partition density [5, 1]:

1. Average within-cluster distance (AWCD)

$$AWCD = \frac{1}{c}\sum_{i=1}^{c}\frac{\sum_{k=1}^{N} u_{ik}^{m} d_{ik}^{2}}{\sum_{k=1}^{N} u_{ik}^{m}}, \tag{6}$$

   This measure monotonically decreases with the number of clusters c. A "knee" in the graph will indicate a suitable partition.

2. Fuzzy hypervolume (Vh)

$$Vh = \sum_{i=1}^{c}[\det(F_i)]^{\frac{1}{2}}, \tag{7}$$

   where F_i, $i = 1, \ldots, c$, are the cluster covariance matrices. Good partitions are indicated by small values of Vh.

3. Average partition density (APD)

$$APD = \frac{1}{c}\sum_{i=1}^{c}\frac{S_i}{[\det(F_i)]^{\frac{1}{2}}}, \tag{8}$$

where S_i is the sum of the membership degrees of the data vectors that lie within a hyperellipsoid whose radius is the standard deviation of the cluster features:

$$S_i = \sum_{k=1}^{N} u_{ik}, \quad \text{for every k, such that} \quad (z_k - v_i)F_i^{-1}(z_k - v_i)^T < 1. \quad (9)$$

Good partitions are indicated by large values of APD.

The clustering procedure is started several times for a given number of clusters and a randomly set partition matrix. The procedure is repeated for different numbers of clusters, varying from a sufficiently small to a sufficiently large amount. Best initial values are those that optimize the chosen cluster validity measures. Normally more then one cluster validity measure should be incorporated. Another strategy is based on compatible cluster merging [1] that starts with a high number of clusters and then deletes bad clusters as well as merges similar clusters together step by step. Both strategies require high computational costs.

In this paper another strategy for a good initialization of the objective function clustering has been applied. A subtractive clustering algorithm [3] is used to partition the identification data set. This algorithm determines the number of clusters and estimates their cluster centres. Then the cluster centres are used as a good initialization for the FCM clustering. As a result by the FCM the optimal cluster centres' coordinates and the partition matrix are obtained. The partition matrix is very useful as its elements are the membership degrees indicating how much each gene belongs to each cluster. So, one obtains not only a partition, but the intensity of the gene's partition. Thus, by defining some cutting level, it is easy to determine groups of those genes that belong to the clusters with desired level of membership.

There are four important parameters of the subtractive clustering algorithm that are responsible for the data partition. The cluster radius r_a is a vector that specifies a cluster centre's range of influence in each data dimension, assuming the data lie within the unit hyperbox. However, after the preprocessing step the transformed data set is fit to assuming spherical clusters, i.e. data points are bounded by a hypercube. Thus, we use the same cluster radius for all data space dimensions. A squash factor s_q is used to multiply the r_a value to determine the neighbourhood of a cluster centre within which the existence of other cluster centres are to be discouraged. The accept ratio ε_a sets the potential, as a fraction of the potential of the first cluster centre, above which another data point will be accepted as a cluster centre. The reject ratio ε_r sets the potential, as a fraction of the potential of the first cluster centre, below which a data point will be rejected as a cluster centre. As the first two parameters are more decisive to the clustering results, they are varied to obtain a good initial partition in terms of the cluster measures AWCD, Vh and APD. The remaining two factors are set to a constant value $\varepsilon_a = 0.5$

and $\varepsilon_r = 0.15$ as it is prescribed in the originally proposed algorithm [3]. The calculated cluster measures by varying first the cluster radii and constantly maintaining the squash factor and after that varying the squash factor while cluster radii have been set to a constant value, are presented in Figure 2. It is difficult to define exactly the best values for r_a and s_q in terms of these cluster measures as there are discrepancies. We can only define an appropriate interval where the best partition is realized. The influential point of both curves in the top of Figure 2 is defined in the interval for $c \in [20, 40]$. In the same interval for the corresponding r_a and s_q values a local minimum can be found in the curves for Vh and APD, respectively. As a reasonable compromise between the optimal values of the cluster measures and a good initial solution for FCM clustering the data partition is chosen that is obtained by the subtractive clustering with parameters $r_a = 0.05$ and $s_q = 1$.

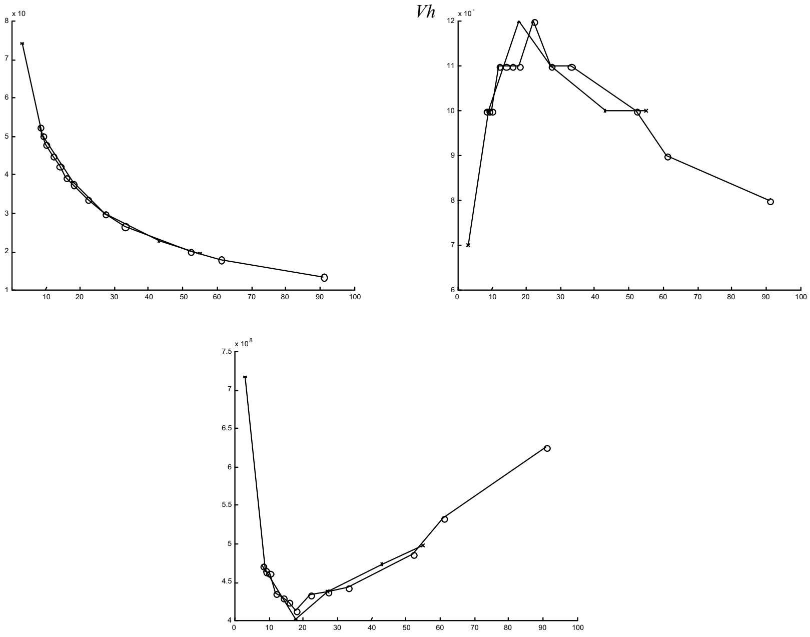

Fig. 2. Cluster measures: -o- r_a varies and $s_q = 1$; -x- $r_a = 0.05$ and s_q varies.

4 Results and discussion

By the procedure described above 27 clusters and their cluster centres are identified. They are used to calculate the initial partition matrix by applying the membership calculation formula of the FCM algorithm. By running the FCM algorithm 27 fuzzy clusters are identified such that genes having similar profiles are assigned to a cluster with highest membership degrees. The degree of membership of each gene to each cluster is given in the partition matrix. It is used to determine the clusters by cutting the membership degrees on a desired level.

Cutting level of the membership degree	0.6	0.5	0.4	0.3	0.2	0.1
Number of all clustered genes	274	447	720	1180	2361	6589

Table 1. Number of clustered genes for different membership level cut

The obtained results show that informative clusters are obtained. The extracted gene clusters are overlapping, which enables us to reveal distinct aspects of the gene function and regulation (Table 1). The number of the clustered genes for the 0.1 cutting level is rather bigger then the number of the clustered genes. This means that a large overlapping of the obtained clusters occurs. By increasing the cutting level the number of the clustered genes decreases.

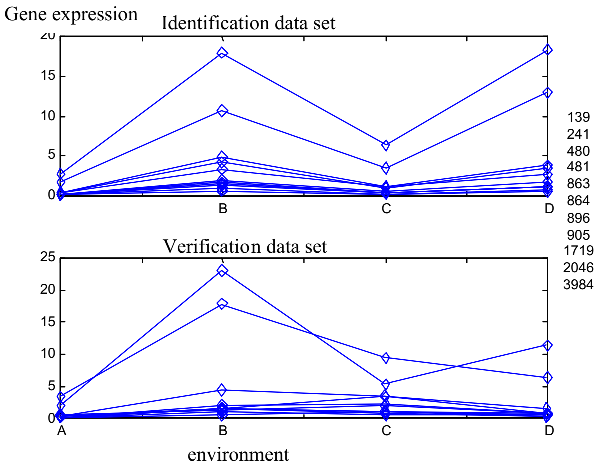

Fig. 3. Profiles of genes (given by numbers) belonging to cluster no. 17

Some typical gene clusters are presented in Figures 3-5. The expression profiles belong to genes that are assigned to the given cluster with a membership degree greater than 0.5 (upper curves in the figures). In order to verify the obtained partition the verification data set is also used. The expression profiles of the genes belonging to this cluster but extracted from the verification data set are shown in the bottom of the figures. The gene expression is changed during the time. This means that it could be expected that the profiles in both data sets could be different. The fact that the extracted genes from the identification data set are still a group with similar profiles in the verification data set proofs the reliability of the obtained partition.

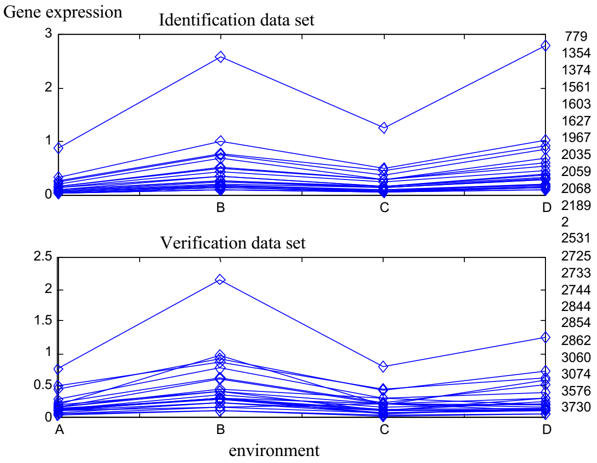

Fig. 4. Profiles of genes (given by numbers) belonging to cluster no. 26

Cluster 17, 26 shows the typical expression pattern: A - low, B - high, C - low and D - high and a lot of genes known to be important for anaerobic life are represented in this cluster. The genes of this cluster are almost not expressed under the A and C conditions (aerobic and anaerobic with nitrate) whereas under the conditions B and D (fermentative and anaerobic with nitrite) the expression of the genes is induced. This expression pattern shows that the condition A and B are more similar to each other than the anaerobic conditions C and D. Under condition A and B the electron transport chain is needed and different electron endacceptors are used: Oxygen in the case of condition A and nitrate at condition B.

Also in cluster 17 we found a lot of genes with unknown functions. Since the expression pattern is similar, we may postulate that they are also needed for anaerobic adaptation. This has to be analyzed in further experiments.

The opposite expression pattern is visible in cluster 19 (A - high, B - low, C - high, D - low), indicating that now genes are expressed that are mainly needed for conditions A and C. Here we found a lot of genes encoding ribosomal proteins.

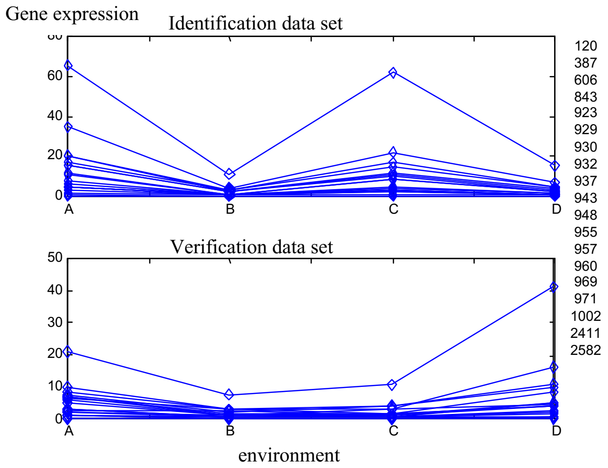

Fig. 5. Profiles of genes (given by numbers) belonging to cluster no. 19

5 Conclusions

In this paper, we have demonstrated how fuzzy clustering can be applied to the analysis of DNA array data using *B. subtilis* as an example. Since the raw data are not suitable for clustering, especially since they contain a high proportion of noise, appropriate transformations have to applied to the data in a preprocessing step. Furthermore, we used a subtractive clustering strategy to determine the number of clusters and to find a good initialization for the fuzzy clustering step. We have used two data sets that were generated independently, but under the same conditions. We have carried out the cluster analysis only in the first data set, whereas the second data set served for verification of the clusters. Assigning the data from the verification set to the clusters derived from the first data set has shown that the clusters group genes together with similar expression characteristics despite the inherent noise.

Taking a closer look at the clusters, we could verify some known correlations between gene expression as well as find some new interesting groups of genes with unknown function which show a similar expression pattern. This might provide some further hints to their function which has to be analysed in further biological experiments.

References

1. R. Babuska. *Fuzzy Modeling for Control.* Kluwer Academic Publishers, Boston, 1998.
2. J.C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Plenum Press, New York, 1981.

3. S.L. Chiu. Fuzzy model identification based on cluster estimation. *Journal of Intelligent and Fuzzy Systems*, 2:267–278, 1994.
4. M.B. Gasch, A.P.and Eisen. Exploring the conditional coregulation of yeast gene expression through fuzzy k-means clustering. *Genome Biology*, 3(11):1–22, 2002.
5. F. Höppner, F. Klawonn, R. Kruse, and T. Runkler. *Fuzzy Cluster Analysis*. Wiley, Chichester, 1999.
6. P.T. Spellman. Cluster analysis and display. In D. Bowtell and J. Sambrook, editors, *DNA Microarrays*, pages 569–581. Cold Spring Harbor Laboratory, Cold Spring Harbor, 2002.