Journal of Bioinformatics and Computational Biology
Vol. 3, No. 4 (2005) 965–988
© Imperial College Press

Imperial College Press
www.icpress.co.uk

NOISE-ROBUST SOFT CLUSTERING OF GENE EXPRESSION TIME-COURSE DATA

MATTHIAS E. FUTSCHIK[*,†]

[*]*Institute of Theoretical Biology, Humboldt-University, Invalidenstr. 43*
10115 Berlin, Germany
[†]*Department of Information Science, University of Otago, PO Box 56*
Dunedin, New Zealand
m.futschik@biologie.hu-berlin.de

BRONWYN CARLISLE

Department of Biochemistry, University of Otago, PO Box 56
Dunedin, New Zealand

Received 14 October 2004
Revised 24 January 2005
Accepted 30 January 2005

Clustering is an important tool in microarray data analysis. This unsupervised learning technique is commonly used to reveal structures hidden in large gene expression data sets. The vast majority of clustering algorithms applied so far produce hard partitions of the data, i.e. each gene is assigned exactly to one cluster. Hard clustering is favourable if clusters are well separated. However, this is generally not the case for microarray time-course data, where gene clusters frequently overlap. Additionally, hard clustering algorithms are often highly sensitive to noise.

To overcome the limitations of hard clustering, we applied soft clustering which offers several advantages for researchers. First, it generates accessible internal cluster structures, i.e. it indicates how well corresponding clusters represent genes. This can be used for the more targeted search for regulatory elements. Second, the overall relation between clusters, and thus a global clustering structure, can be defined. Additionally, soft clustering is more noise robust and *a priori* pre-filtering of genes can be avoided. This prevents the exclusion of biologically relevant genes from the data analysis. Soft clustering was implemented here using the fuzzy c-means algorithm. Procedures to find optimal clustering parameters were developed. A software package for soft clustering has been developed based on the open-source statistical language R. The package called *Mfuzz* is freely available.

Keywords: Microarray data analysis; clustering; noise-robustness.

1. Introduction

Microarrays have made it possible to monitor simultaneously the expression of thousands of genes. They have rapidly become indispensable experimental techniques in biomedical research and have offered new insights into the biology of cells.

The analysis of the large data sets generated by microarrays, however, remains challenging. A major task is the detection of patterns in gene expression data despite a large background noise.

A standard approach for pattern detection is cluster analysis. It has been widely used in numerous fields of scientific research.[1] Clustering can be especially useful if prior knowledge is little or non-existent, since it requires minimal prior assumptions. This feature has made clustering a favourable tool in the analysis of microarray data, where knowledge of the underlying regulatory networks has been limited.

Different cluster algorithms have been applied to the analysis of expression data: k-means, SOM, graph-based and hierarchical clustering to name just a few.[2–5] These methods assign genes to clusters based on the similarity of their expression patterns. Genes with similar patterns should be grouped together, while genes with different patterns should be placed in distinct clusters. The vast majority of these cluster methods used so far have been restricted to a one-to-one mapping: one gene belongs to exactly one cluster (or in case of hierarchical clustering, to exactly one sequence of nested clusters). The borders between clusters are hard, i.e. genes are assigned to exactly one cluster even if their expression profile is similar to several cluster patterns. The underlying assumption for this so-called *hard clustering* is that clusters are well separated. While this principle seems reasonable in many fields of cluster analysis, it might be too limited for the study of microarray data. For several time-course experiments, it has been pointed out that there are no well-defined boundaries between classes of temporal patterns. For example, Spellman and collaborators noted that no clear boundary existed between categories defined by the yeast cell cycle phases.[6] Cho and co-workers observed genes that were induced in two different phases.[7] For sporulation of yeast cells, Chu *et al.* remarked that genes were often highly correlated with the patterns of more than one cluster.[8] These observations might be expected, since genes products frequently participate in more than one regulatory mechanism to different degrees. The regulation of a gene is generally not in an "on-off", but gradual manner which allows a finer control of the gene's functions. A cluster algorithm should reflect this finding by differentiating how closely a gene follows the dominant cluster patterns. *Soft clustering* appears as a good candidate for this task since it can assign genes gradual degrees of membership to a cluster. The membership values can vary continuously between zero and one. This feature enables soft clustering to provide more information about the structure of gene expression data. Soft clustering can be implemented using algorithms (such as fuzzy c-means) based on minimization of objective functions.[9,10] Alternatively, probabilistic approaches such Gaussian mixture models combined with expectation-maximization schemes can be applied.[11]

A second reason for applying soft clustering is the high level of noise in microarray data due to numerous biological and experimental factors. A common procedure to reduce noise in microarray data is the setting of a minimum threshold for change in expression. Genes below this threshold are excluded from further analysis. However, the exact threshold value remains arbitrary due to the lack of an established error model. Additionally, filtering may exclude interesting genes from further analysis. Soft clustering is a valuable approach here since it is highly robust to noise and pre-filtering can be avoided.

In this context, an additional problematic feature of conventional clustering methods such as k-means is that they always detect clusters, even in random data. Especially for gene expression analysis, this may lead to false results, as often little prior knowledge exists to readily identify cluster artefacts. Therefore, a favorable clustering method should avoid detecting clusters in random data. We will show that soft clustering can also fulfil this criterion.

This article is structured as follows: First we outline the main differences between hard and soft clustering. We then describe our implementation of soft clustering using fuzzy c-means. To illustrate this approach, we re-analyze the yeast cell-cycle experiment by Cho *et al.*[7] After a description of the data pre-processing, the selection of clustering parameters is addressed. This is followed by a presentation of important features of soft clustering and a comparison of the noise robustness of hard and soft clustering. A discussion of the main results and related studies closes this presentation.

2. Clustering Methods

Clustering methods can be divided into hierarchical and partitional clustering. Hierarchical clustering creates a set of nested clusters, so that clusters on a higher hierarchical level comprise clusters on lower levels. The sequential partitioning is conventionally presented in a dendrogram. This makes hierarchical clustering a popular tool in microarray data analysis, since the internal structure is easily accessible. However, the order of branches in a dendrogram remains undefined making its interpretation difficult. Additionally, hierarchical clustering is sensitive to noise, since the clustering process is based on local data features. To gain noise robustness, partitional clustering can be used. It splits the data in clusters without the definition of a cluster hierarchy. The generated partition divides the data into several clusters and can be represented by a partition matrix U that contains the membership values μ_{ij} of each object i for each cluster j.

2.1. *Hard partitional clustering*

For clustering methods, which is based on classical set theory, clusters are mutually exclusive. This leads to the so-called hard partitioning of the data. Relationships between objects in one cluster remain generally unspecified. Hard partitions are defined as

$$M_{hc} = \left\{ U_{ij} \in R^{N \times k} \;\middle|\; \begin{array}{l} \mu_{ij} \in \{0,1\} \;\; \forall\, i,j \\ \sum_{j=1}^{k} \mu_{ij} = 1 \; \forall\, i \\ 0 < \sum_{i=1}^{N} \mu_{ij} < N \; \forall\, j \end{array} \right\}, \tag{1}$$

where k is the number of clusters, N is the number of data objects and $\mu_{ij} \in \{0, 1\}$ denotes that μ_{ij} is either 0 or 1. Partitional clustering is frequently based on the optimization of a given objective function. If the data is given as a set of n-dimensional vectors, a common objective function is the square error function

$$E = \sum_i \sum_j d(\mathbf{x}_i, \mathbf{c}_j)^2, \tag{2}$$

where d is the distance metric and $\mathbf{c}_j$ is the centre of cluster j. The square error function E describes the within-cluster variation. Minimization of E results in clusters with a minimal sum of the distances between the data vectors $\mathbf{x}$ and the cluster centres $\mathbf{c}$. Since the total variation of the data is fixed, minimizing the within-cluster variation leads to maximizing the between-cluster variation. A popular approach for minimizing E is k-means clustering which iteratively assigns objects to clusters until no objects change in consecutive iterations, i.e. self-consistency of the clustering process is reached. For the distance metric D, the Euclidean distance is generally chosen.

2.2. *Soft partitional clustering*

Algorithms for hard clustering perform favorably, if clusters are well separated. In many situations, however, this might not be the case. Clusters may be overlapping with data objects between clusters sharing attributes of several clusters. To accommodate this situation, soft clustering generalizes the partitioning of hard clustering. Soft clustering is based on the concept of soft partitioning of the data. In contrast to hard clustering, a data object can be member of several clusters. Membership values μ_{ij} define a "grade" of membership and can take any value between zero and one. This results in soft partitions that take the form of

$$M_{sc} = \left\{ U_{ij} \in R^{N \times c} \;\middle|\; \begin{array}{l} \mu_{ij} \in [0, 1] \;\; \forall\, i, j \\ \sum_{j=1}^{c} \mu_{ij} = 1 \;\forall\, i \\ 0 < \sum_{i=1}^{N} \mu_{ij} < N \;\forall\, j \end{array} \right\}, \tag{3}$$

where c is the number of soft clusters and $\mu_{ij} \in [0, 1]$ denotes that μ_{ij} is a real value between 0 or 1 (including 0 and 1). Note that space of soft partitions M_{sc} fully contains the set of hard partitions M_{hc} . Hard k-means clustering can be seen as a special case of soft clustering where the membership values are either zero or one.

3. Implementation of Soft Clustering

3.1. *Soft clustering by fuzzy c-means algorithm*

An important objective function for soft clustering is the c-means function J_m which is similar to the square error function E for k-means clustering. It weights, however, the distances of the data vector $\mathbf{x}_i$ to the cluster centre $\mathbf{c}_j$ according to the membership values of $\mathbf{x}_i$:

$$J_m = \sum_i \sum_j (\mu_{ij})^m \|\mathbf{x}_i - \mathbf{c}_j\|^2_{\mathbf{A}}, \tag{4}$$

where m is a parameter with $m > 1$ and $\|\cdot\|_{\mathbf{A}}$ is a distance norm of the quadratic form $\|\mathbf{x}\|_{\mathbf{A}} = \sqrt{\mathbf{x}^T\mathbf{A}\mathbf{x}}$ with $\mathbf{A}$ being a positive definite matrix. If $\mathbf{A}$ is the identity matrix, $\|\cdot\|_{\mathbf{A}}$ is the Euclidean distance. In contrast to E, the objective function J_m contains, besides the number of clusters, a further parameter m. Choosing m determines the influence of $\mathbf{x}_i$ on the clustering process depending on their membership values μ_{ij}. If parameter m is increased, poorly classified objects which have small membership values μ_{ij} contribute less to the calculation of the cluster centres $\mathbf{c}_j$. Data objects with a large noise content thus have a reduced influence on the outcome of the clustering process. This makes soft clustering especially suitable for noisy data such as microarray measurements.

Several methods for minimizing the objective function J_m have been proposed.[9,10] Fuzzy c-means (FCM) clustering is the most common algorithm for solving this problem. It is based on the first order conditions for a minimum of J_m for c cluster centres and N data vectors:

$$\mathbf{c_j} = \frac{\sum_{i=1}^N (\mu_{ij})^m \mathbf{x}_i}{\sum_{i=1}^N (\mu_{ij})^m} \quad \forall j \in [1, c] \tag{5}$$

$$u_{ij} = \frac{1}{\left(\frac{\|\mathbf{x}_i - \mathbf{c}_j\|_{\mathbf{A}}}{\sum_{k=1}^c \|\mathbf{x}_i - \mathbf{c}_k\|_{\mathbf{A}}}\right)^{\frac{2}{m-1}}} \quad \forall i \in [1, N], \quad j \in [1, c]. \tag{6}$$

A Picard iteration alternating between the evaluation of Eq. 5 and 6 adjusts μ_{ij} and $\mathbf{c}_j$ until the change in J_m falls below a threshold e or a maximal number of iterations $T_{\max}$ is reached. Note that $\mathbf{c}_j$ is the weighted mean of all $\mathbf{x}_i$ in cluster j.

3.2. *Microarray data set*

To illustrate our approach, we applied soft clustering to expression data of the yeast cell cycle by Cho *et al.*[7] This enables us to compare directly the results obtained by soft clustering to those obtained by other methods, since several clustering schemes have been applied to this data set: clustering by visual inspection,[7] by SOMs,[12] by stimulated annealing[13] and k-means.[2]

3.3. *Data pre-processing*

In the yeast cell cycle experiment by Cho and co-workers, 6178 genes were monitored at 17 time points over a span of 160 minutes using Affymetrix chips. The expression values for the time point $t = 90$ minutes were excluded in our analysis as these data were considered erroneous.[2] Further, genes with less than 75% of the measurements present were excluded. This reduced the number of genes for the cluster analysis to 6101. The arrays were globally normalized, i.e. the total intensity of each array was linearly scaled to have the same value. To convert the Affymetrix data into ratios, the measured intensities of each gene were divided by their average values. To treat positive and negative fold changes equally, the data were $\log_2$-transformed. Missing values were replaced by estimates derived by the *knn* method.[14]

3.4. *Filtering*

Most cluster analyses previously performed have included *a prior* filtering step to remove genes that are expressed at low levels or that show only small changes in expression. Different filtering procedures have been proposed for the analysis of the expression data analysed here. Heyer and co-workers excluded all genes with a mean or variance in the lower 25% of the data.[15] Tavazoie and collaborators included only 3000 genes, which showed the largest variation.[2] Tamayo *et al.* reduced the number of genes for analysis to as few as 823 by setting thresholds for the relative and the absolute change.[12] Inspection of the different measures proposed, however, revealed that no obvious threshold for filtering existed. For example, Fig. 1 shows the standard deviation of gene expression in the experiment. The transition between low and high values for variation in gene expression is smooth and no particular cut-off point is indicated. Thus, the value of a filtering threshold remains arbitrary.

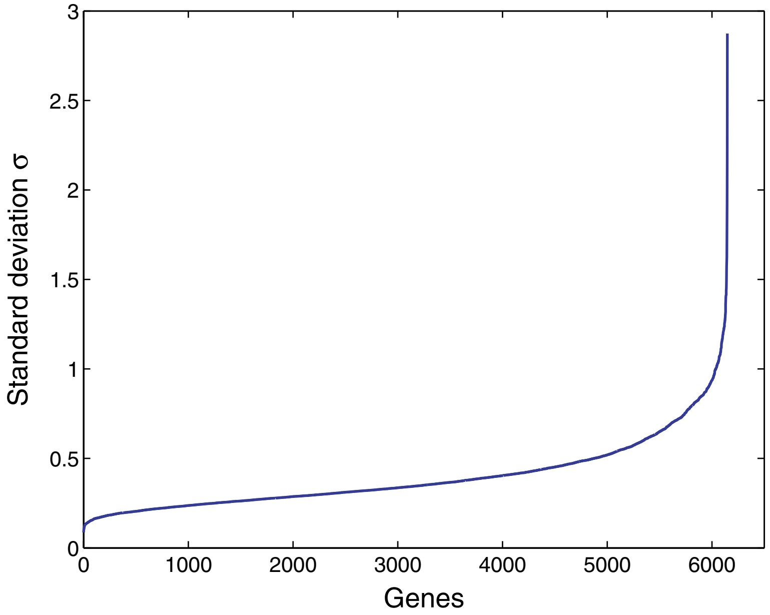


Fig. 1. Standard deviation of gene expression vectors before standardization. The genes were ordered by the standard deviation of the corresponding expression vector. A unique cut-off value for filtering is not prominent.

As no stringent filtering procedure currently exists, we avoided any prior filtering of data. This prevents the loss of biologically important information, as many genes show only small changes in transcription.[16] The inclusion of all genes in the analysis demands, however, a clustering method which is robust against noise. We latter demonstrate that this is the case for soft clustering by FCM.

3.5. *Standardisation*

Since the clustering is performed in Euclidian space, the expression values of genes were standardized to have a mean value of zero and a standard deviation of one. This ensures that vectors of genes with similar changes in expression are close in Euclidean space.

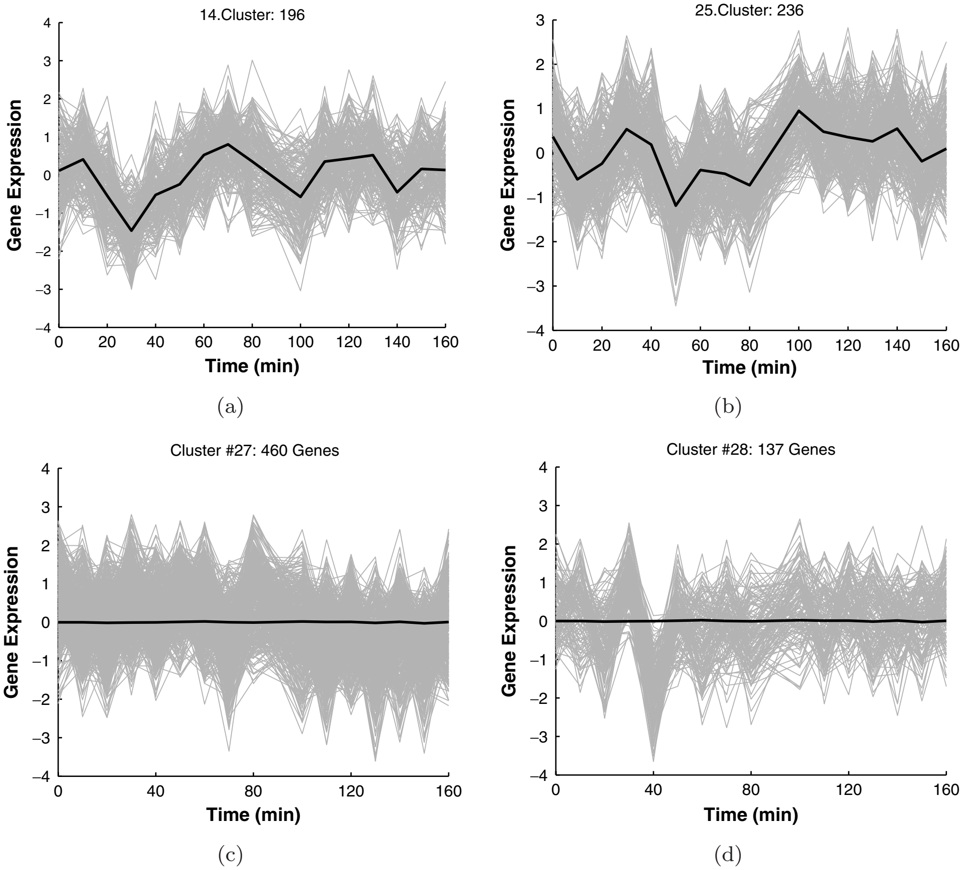
3.6. *Determination of the clustering parameters*

To use FCM for cluster analysis, several parameters have to be specified. Besides the number of clusters c and the FCM parameter m, users must choose values of the minimal change e in the objective function for termination and the maximal number Tmax of iterations. For termination of the clustering process, we specified the minimal change $e = 0.001$ and the maximal number of iterations $T_{\max} = 100$. Setting a larger $T_{\max}$ or a smaller e yielded only minor changes in the clustering results. For the distance metric $\mathbf{A}$, the Euclidean distance was chosen.

The FCM parameter m is a crucial parameter since it determines the influence of noise on the cluster analysis. For $m \to 1$, it can be shown that the clustering becomes hard.[9] The FCM algorithm is then equivalent to the k-means clustering. The membership values are either one or zero. All genes of a cluster are treated equally for the calculation of the cluster centre. Increasing the parameter m reduces the influence of genes with low membership values as can be seen in Eq. (5). Gene expression vectors with large noise component generally have a low membership value, since the corresponding genes are not well represented by a single cluster, but rather are partially assigned to several clusters. Hence, selection of the FCM parameter m determines the influence of noise on the clustering process. For $m \to \infty$, the partition matrix becomes uniform, i.e. genes are assigned to all clusters equally. Monitoring the clustering results for increasing m therefore gives insights into the data structure. We will use this feature to prevent the detection of clusters in random data.

3.7. *Construction of a baseline clustering*

A major problem with hard clustering algorithms such as k-means or SOMs is that they always assign objects clusters. Even if the data are random, distinct clusters are formed. This is illustrated in Fig. 2(a), (b) which shows clusters detected by k-means for randomized yeast cell cycle data. The randomization was achieved by random permutation of the time order of every gene independently. Data structures



(a) (b) (c) (d)

Fig. 2. Clustering of randomized data: (a,b) Example clusters produced by k-means ($k = 30$) clustering of randomised expression data. Cluster centroids are indicated by solid lines. (c,d) Example clusters produced by FCM ($m = 1.25$, $c = 30$) clusterings of randomized expression data. Similar results are produced for cluster parameter c ranging between 2 and 60 with $m = 1.25$. The illustrated cluster structures are somewhat misleading, since genes were assigned to clusters for which they had a maximal membership value. The membership values showed, however, only minimal variation. The mean membership value was $0.033 \approx 1/30$ with a standard deviation $1.3 \cdot 10^{-5}$. Essentially, all genes were equally included for the calculation of each cluster centre, so that $\mathbf{c}_k \approx \vec{0}$. Note that the clusters in (a) and (b) do not correpond to clusters in (c) or (d), but are randomly chosen from the hard and soft clusterings.

occurring by chance were identified as clusters by k-means. This feature of hard clustering is problematic, as it can easily lead to false results, especially if no prior biological knowledge exists which allows for rapid identification of such artefacts.

This pitfall in hard clustering can be overcome by soft clustering. Since the FCM parameter m controls the sensitivity of the clustering process to noise, we can adjust m to prevent the detection of clusters in the randomized data. For the determination of parameter m, randomized data was clustered with parameter values between 1.05 and 3.05. The number of clusters c was varied between 2 and 60. In this range of c, inspection of the FCM clustering results showed that no clusters are detected for $m \geq 1.25$. The partition matrices became uniform, i.e. every gene was approximately equally assigned to all clusters. All genes were included equally for the calculation of each cluster centre. Therefore, the cluster centroids derived were approximately null vectors, i.e. vectors with all coordinates equalling zero. Examples are shown in Fig. 2(c), (d) for the number of clusters set to 30. The FCM parameter was therefore set to $m = 1.25$ for the following analysis.

3.8. *Determination of the number of clusters*

After the selecting the FCM parameter m, the number of clusters has to be determined. For this task, the cluster number c in the FCM algorithm was gradually increased and the results of the clustering were examined. It was observed that the membership values of genes tend to spread more between clusters as the produced clusters become more similar for increasing c. Especially for less isolated clusters, the number of genes with a membership value larger than 0.5 decreased. Finally, clusters are generated for which none of the genes surpasses the membership value of 0.5. We call these clusters *empty* clusters as no gene is primarily assigned to them (see also Fig. 9). Note that the term cluster corresponds here to the list of membership values μ_{ij} stored in a chosen column of partition matrix U_{ij}. An empty cluster thus corresponds to a column of the partition matrix with all values smaller than 0.5. This enlarges the common definition of a cluster, where cluster are sets of genes which either do or do not belong to a chosen cluster. For soft clustering, genes can belong to a cluster to a certain degree between 0 and 1.

The appearance of empty clusters in the clustering process allows the setting of the parameter c. Soft clustering by FCM was repeatedly performed and the number of non-empty clusters was monitored. Figure 3 shows that all of the repeated clusterings produced empty clusters for $c > 20$. Hence, we selected $c = 20$ to prevent the persistent appearance of empty clusters in repeated clusterings for the following analysis. The same number of clusters was found by Luskashin and Fuchs using stimulated annealing.[13] For $c = 20$ and $m = 1.25$, an average of 1560 genes was assigned maximal membership values less than 0.5, i.e. over 25% of the genes were not primarily assigned to a single cluster.

A further increase of c always produced empty clusters, although the number of non-empty clusters also increased. However, since empty clusters can be easily detected, the setting of the cluster number is less problematic for soft clustering compared with hard clustering which does not indicate the quality of clusters. Increasing the cluster number to $c > 20$ may even be favorable for the study of local structures as we discuss further below.

A drawback of our method to selecting the clustering parameters m and c is its high computational cost due to repeated clustering and data randomization.

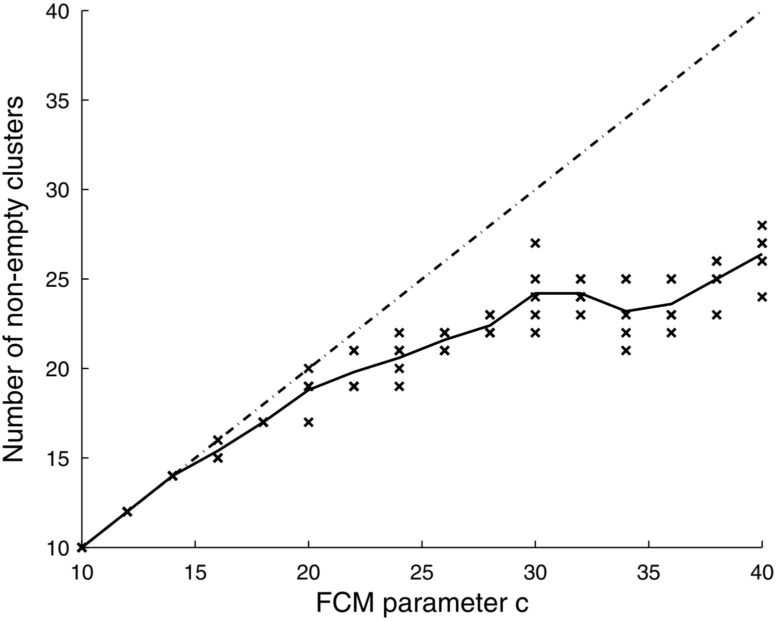


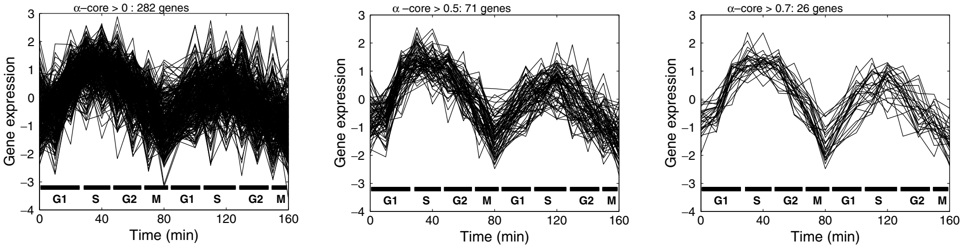
Fig. 3. Determination of cluster number c for FCM clustering. The number of non-empty clusters is shown for increasing c. Five repeated clustering with random initiations were performed. The FCM parameter m was set to 1.25. The dotted line shows the maximum possible number of non-empty clusters.

However, as it can be performed in "batch mode" without requiring human interaction, this drawback might be less severe in practice.

4. Results

4.1. *Differentiation in cluster membership and profiling of cluster cores*

For soft clustering, the cluster centroids $\mathbf{c}_k$ result from the weighted sum of all cluster members and show the overall expression patterns of clusters. The membership values μ_{ij} indicate how well the gene expression vector $\mathbf{g}_i$ is represented by $\mathbf{c}_k$. Low values μ_{ij} point to a poor representation of gene i by $\mathbf{c}_k$. Large values μ_{ij} point to a high correlation of $\mathbf{g}_i$ with $\mathbf{c}_k$. Membership values can also indicate the similarity of vectors to each other. If two gene expression vectors have a high membership value for a specific cluster, they are generally similar to each other. This is the basis for the definition of the core of a cluster. We define that genes with membership values larger than a chosen threshold α belong to the *α-core* of the cluster. This allows us to define relationships between genes within a cluster. Similarly to hierarchical clustering, the internal structures of clusters become accessible.

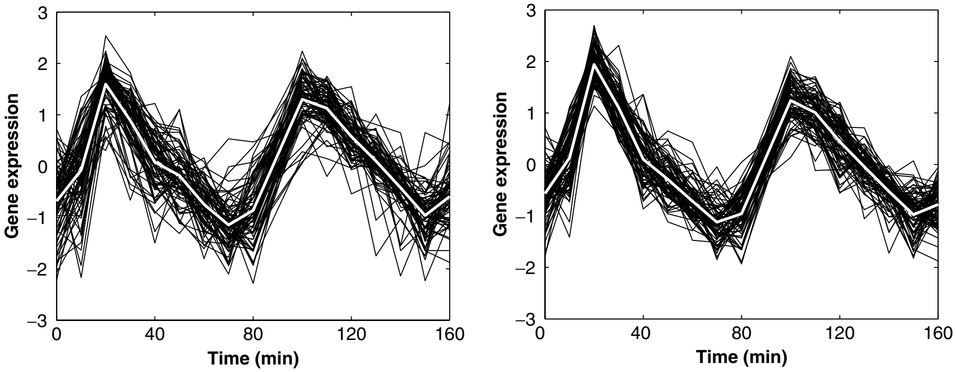


(a) $\alpha = 0$ (b) $\alpha = 0.5$ (c) $\alpha = 0.7$

Fig. 4. Different α-cores for soft cluster ($c = 20$ and $m = 1.25$): Left sub-figure ($\alpha = 0$) displays only those genes which were primarily assigned to the S-phase cluster.

Figure 4 presents different α-cores for an expression cluster (peaking in S-phase). Genes can be differentiated by examining whether they are included in a certain α-core. Figure 4(a) shows all genes which were primarily assigned to the cluster. This cluster structure is equivalent to hard clustering. The within-cluster variation of the gene expression values is considerable indicating a high background noise. Setting the α-threshold to 0.5 decreased within-cluster variation. Genes were excluded if they were poorly correlated with the overall cluster pattern. The periodicity of the remaining genes became more prominent. Increasing the α-threshold to 0.7 led to a decreased number of genes included in the α-core. Only 26 genes of 282 originally assigned to the cluster remained. Simultaneously, the average within-cluster variation was reduced from 0.78 for $\alpha = 0$ to 0.40 for $\alpha = 0.7$. The use of the α-threshold can therefore act as *a posteriori* filtering of genes. This contrasts with previously discussed procedures which demand the problematic setting of a threshold *a priori* to the cluster analysis. Soft clustering thus avoids the exclusion of biologically relevant genes from cluster analysis.

To study the differences of both filtering procedures in more detail, we compared the within-cluster variation resulting from *a priori* filtering with subsequent k-means clustering and from *a posteriori* with preceding FCM clustering. For *a priori* filtering, we followed the approaches by Heyer *et al.* excluding over 1150 genes,[15] Tavazoie *et al.* excluding over 3100 genes[2] and Tamayo *et al.* excluding over 5200 genes.[12] Each of these filtered data sets was clustered by k-means ($k = 20$). The average within-cluster variation of the generated hard clusters was compared to α-cores of corresponding clusters produced by FCM ($c = 20$, $m = 1.25$) applied to the original data set. To enable direct comparison of both filtering approaches, the threshold α was chosen so that the number of remaining genes equals the number of genes in the corresponding hard cluster. This represents the *a posteriori* filtering step. The results of both filtering procedure are illustrated in Fig. 5 for the case of the G_1 cluster. Using the filtering procedure by Tamayo *et al.* and k-means clustering led to a G_1 cluster with 80 genes and an average within-cluster variation of 0.48. A reduced within-cluster variation of 0.38 was achieved based on the 80 genes with the largest membership values for the G_1 cluster determined



(a) *A priori* filtering

(b) *A posteriori* filtering

Fig. 5. Comparison of *a priori* and *a posteriori* filtering: Both G_1 clusters include 80 genes. The solid white lines represent the cluster centers. (a) The cluster was detected by k-means after *a priori* filtering was applied. The filtering procedure followed Tamayo *et al.* excluding over 5200 genes. The average within-cluster variation is 0.48. (b) The cluster was detected by FCM in the original data. Subsequently, *a posteriori* filtering was applied leading to the same number of included genes as in cluster (a). The average within-cluster variation is reduced to 0.38. An increased tightness of the *a posteriori* filtered cluster is apparent. Similar tendencies were found for other clusters.

by FCM. Similarly, the *a priori* filtering used by Tavazoie *et al.* combined with k-means produced a G_1 cluster with 215 genes and an average variation of 0.58. The corresponding core of the soft G_1 cluster had an average variation of 0.53. Finally, the filtering procedure of Heyer *et al.* led to a G_1 cluster with 313 and an average variation of 0.68. *A posteriori* filtering resulted in a reduced variation of 0.61. This indicates that *a posteriori* filtering can outperform conventional *a priori* filtering regarding the tightness of the detected clusters.

The possibility to rank genes based on their membership values also facilitates the identification of transcription mechanisms. To examine this feature of soft clustering further, a search for regulatory elements in the up-stream regions of the open reading frames (ORFs) was performed using the AlignACE software. ALIGNACE (for Align Nucleic Acid Conserved Elements) is a Gibbs sampling algorithm that identifies over-represented motifs in a set of DNA sequences. It has been optimized to find multiple motifs using a masking procedure.[17,18] Here we employed the on-line version available at *http://atlas.med.harvard.edu*, which is dedicated to detect sequence motifs in the up-stream region of ORFs in the yeast genome.

The top ranked 30 genes of clusters generated by soft clustering were used as input for the Gibbs alignments produced by AlignACE. Besides the sequence alignments, AlignACE delivers several statistical measures for the quality assessment of alignments: The maximum *a priori* log likelihood (MAP) score measures the degree to which a motif is over-represented in the sequences considered. The group specificity score measures the degree to which the motif is associated with the set of

Table 1. Comparison of promoter identification by hard and soft clustering for several known motifs. Results of alignments by AlignACE are shown for FCM and corresponding k-means clusters. MAP stands for the maximum *a priori* log likelihood score and is a measure for the degree to which a motif is over-represented. Group specificity (Sp) measures the degree to which the motif is associated with the set of selected genes. The similarity score (Sim) indicates the similarity of the identified motifs with known motifs. Dashes denote that motifs were not found. Besides the statistical measures produced by AlignACE, the average membership value $\bar{\mu}$ of genes used in the alignment are shown. Alignments were generally more group specific if $\bar{\mu}$ was large.

	SOFT (FCM) CLUSTERING				K-MEANS CLUSTERING		
Motif	*MAP*	Log*10(Sp)*	*Sim.*	$\bar{\mu}$	*MAP*	Log*10(Sp)*	*Sim.*
PAC	45.6	−5.2	1	0.79	14.5	−3.8	0.83
STE12	10.9	−6.4	0.72	0.79	—	—	—
RAP1	37.7	−14.1	0.96	0.97	21.3	−10.8	0.75
ECB	29.6	−10.8	0.69	0.88	—	—	—
MCB	67.8	−26.6	0.98	0.99	35.1	16.7	0.79

selected genes. A smaller score denotes a higher specificity. If an alignment produces a motif similar to a known motif, a similarity score ranging from −1 to 1 is additionally produced for this comparison. The list of cis-regulatory motifs known from the literature were identified and assigned in the study by Hughes *et al.*[18] In the context of our study, it allowed us to compare the quality of soft and hard clustering. Thus, AlignACE results for soft clustering were compared with the alignments of the same number of genes from the corresponding cluster produced by hard k-means clustering. Generally, we observed that alignments are of improved quality if they are based on top ranked genes of soft clusters. The results for several known regulatory elements are shown in Table 1. Larger MAP and smaller group specificity scores were achieved by alignments of genes clustered by FCM. Some motifs such as the early cell-cycle box (ECB) were found only for genes highly ranked by soft clustering and were not detected for the corresponding set of genes derived from hard clustering. This demonstrated that using all genes contained in a cluster may introduce noise and hinders proper alignment of motifs. Weakly correlated genes in the cluster may lack the sequence motifs of the genes in the strongly correlated cluster core. This indicates that genes with large membership value for a cluster are more likely to be co-regulated than the remaining genes in the cluster. It is also notable that the similarity scores for motifs were generally larger for alignments based on soft clustering. This feature is especially favorable for the detection of novel regulatory sequences since it enables more accurate reconstruction of motif sequences. It also shows that soft clustering can be used for an improved identification of regulatory elements while avoiding *a priori* filtering of genes.

4.2. *Stability of clusters*

Based on the definition of α-cores, variation of the FCM parameter m can be used to gain insight into the internal structure of single clusters, since the choice of m controls the distributions of the cluster membership values. Small m led to large cluster cores with little variation of the membership functions. Increasing m yielded partitions with more distributed membership values. The α-core of the clusters became more differentiated, e.g. a larger α-threshold resulted in smaller cluster cores. By varying m, the distribution of membership values, and thus the internal cluster structures, can be examined. To support this examination, we color-coded the α-core of clusters (Fig. 6). This facilitates the identification of temporal patterns in gene cluster.

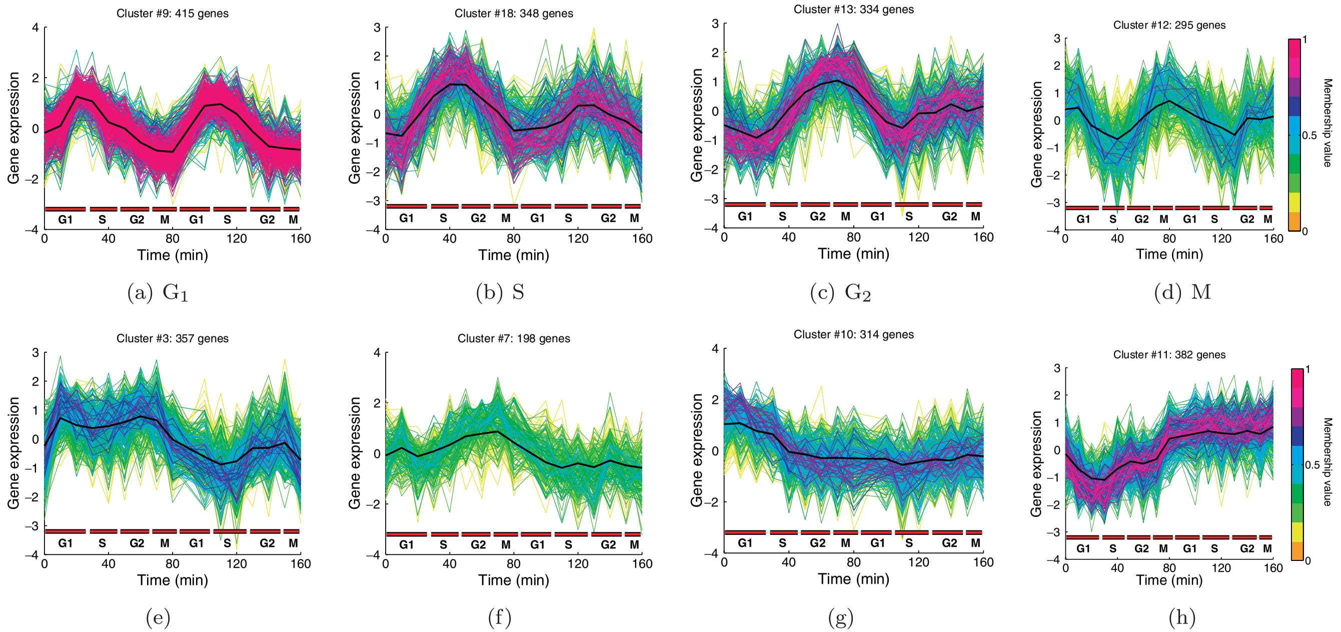
Variation of the FCM parameter m also allows investigation of the stability of clusters. We define stable clusters as clusters that show only minor changes in their structure with variation of the parameter m. Stable clusters are generally isolated and compact. This is contrasted by weak clusters that lose their internal structure or disappear if m was increased. Example clusters in Figs. 7(a)–7(d) illustrate this procedure. Both clusters seemed to have a well-defined α-core for $m = 1.15$. Differences appeared, however, for $m = 1.25$. For the stable cluster in Fig. 7(a), most genes retained their large membership values (Fig. 7(b)), whereas the membership values decreased considerably for the weak cluster in Fig. 7(c). Additionally, the number of genes assigned to the strong cluster stayed approximately the same. This is contrasted by the weak cluster that lost genes for larger m (Fig. 7(d)).

By continually increasing m, it is possible to rank clusters according to their stability. Biologically, this may give indications of how strongly genes are co-regulated in the underlying genetic networks. We found that frequently the motifs discovered in alignments based on genes of weak cluster are less statistically significant than motifs discovered in stable clusters. Especially for weak clusters, however, selection of genes based on their membership values can support the identification of co-regulation. This can be illustrated, for example, for the weak cluster presented in Fig. 7. The 100 top ranked genes of this cluster had an average membership of only 0.50. Using this set of genes for promoter identification by AlignACE did not yield any motif with a MAP score larger than 3 and a group specificity smaller than 10^{-3}. If the set of genes, however, is reduced to the top 30 ranked genes, two novel motifs surpasses the threshold for MAP and group specificity. Thus, soft clustering allows for refinement of gene selection that can support the follow-up analysis.

4.3. *Noise robustness*

The previous sections showed that soft clustering assigns membership values to genes based on the similarity of their expression to the overall pattern of the cluster. Expression vectors with low membership values may thus be considered as noisy. To test this hypothesis, we analyzed the noise robustness of soft clustering against increased noise. For this task, random Gaussian noise was added to the gene expression data. The following formula was used:

$$\tilde{g}_i = g_i + b \cdot N(0,1), \tag{7}$$



(a) G_1

(b) S

(c) G_2

(d) M

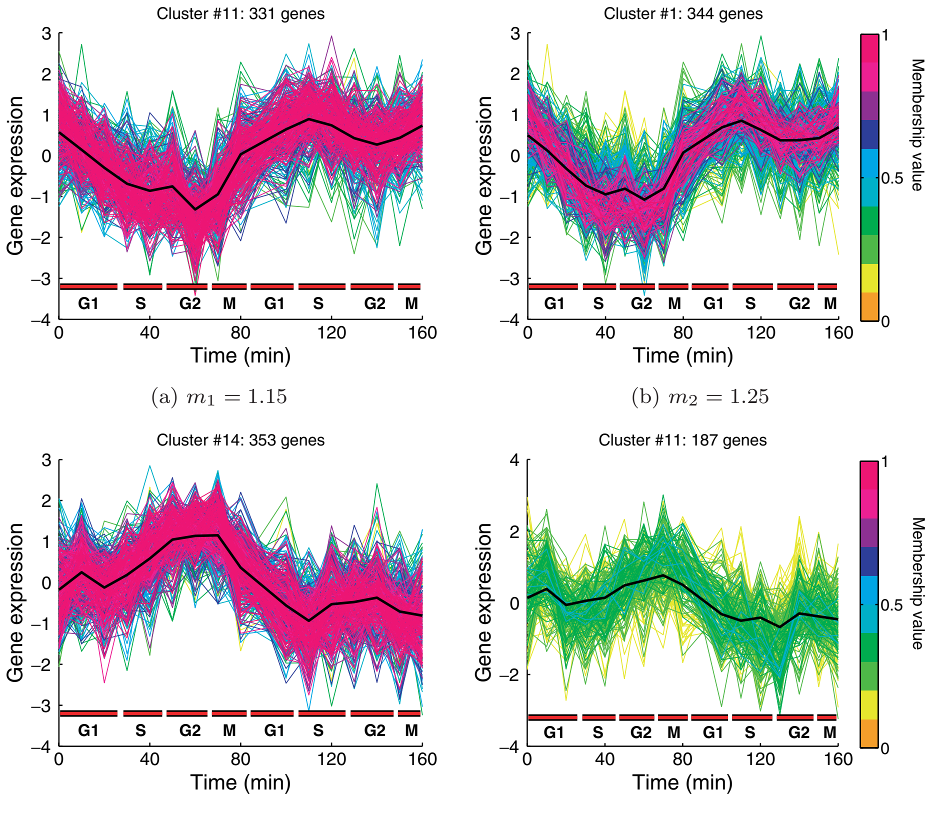
(e)

(f)

(g)

(h)

Fig. 6. Identification of temporal patterns by soft clustering. (a)–(d) Example clusters with periodic gene expression patterns are shown. The labelling of the clusters is based on the peak of gene expression. Periodic clusters had generally large α-cores. (e)–(h) Example clusters with aperiodic expression patterns: These clusters were generally weaker than the periodic clusters detected. FCM parameters were $c = 20$ and $m = 1.25$.



(a) $m_1 = 1.15$

(b) $m_2 = 1.25$

(c) $m_1 = 1.15$

(d) $m_2 = 1.25$

Fig. 7. Cluster stability determined by variation of FCM parameter m. Example clusters are shown for two soft clusterings ($c = 20$, $m_1 = 1.15$, $m_2 = 1.25$). The colour bar on the right indicates the color-encoding of the membership values. (a), (b) Stable clusters maintain their core for increasing m. (c), (d) Weak clusters lose their core for increasing m. The membership values as well as the number of genes are reduced.

where g_i is the original expression vector of gene i, is the expression vector with added noise and $N(0, 1)$ is the standard normal distribution.

The "noisy" gene expression data was clustered and the results compared to the clustering of the original data. To evaluate the noise robustness of soft clustering, we calculated the percentage of pairs of genes assigned in the same clusters for both clusterings. Ideally, gene pairs for the original clustering should also be found for the clustering of the "noisy" data. The results for fraction $b = 0.5$ is shown in Fig. 8 (Choosing other values of b gave concurrent results.) K-means clustering, which was used for comparison, assigns on average only 34% of the gene pairs to the same cluster. This can be improved by soft clustering, since it differentiate gene pairs based on their similarity of expression in the original clusters. Pairs with high membership values for the original cluster are more likely to be clustered together

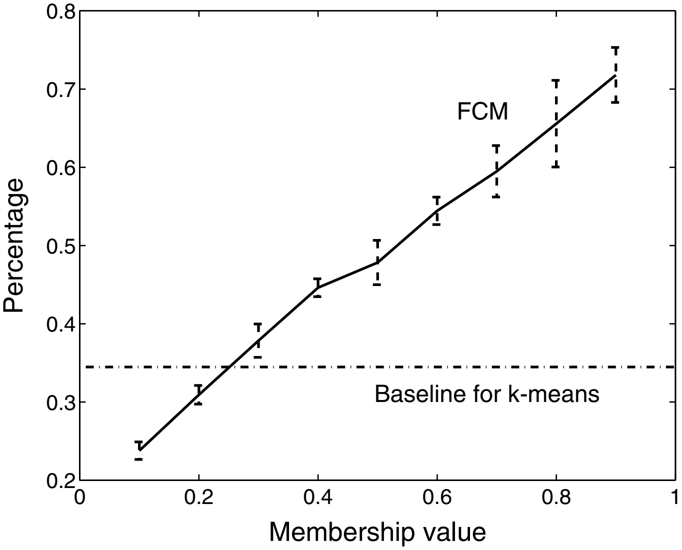


Fig. 8. Noise robustness of k-means and FCM clustering. Percentage of gene pairs which clustered together for original and "noisy" data. Derivation of mean values and error interval were based on five clusterings of independently generated "noisy" data sets. The percentages for membership value x give the fraction of gene pairs clustered together with both genes having a maximal membership value $x \pm 0.05$ for the original clustering.

for noisy data. For example, less than 30% of the pairs were clustered together again if both genes had membership values of less than 0.3 for the original cluster. However, over 70% of pairs, were clustered together if both genes had membership value of 0.85 or higher. Cluster cores obtained by a large threshold are therefore more likely to reflect the "noise-free" cluster structure. This observation justifies the procedure of *a posteriori* filtering introduced previously.

4.4. *Global clustering structures*

An interesting feature of soft clustering is the overlap or coupling between clusters. The coupling coefficient V_{kl} between cluster k and cluster l can be defined by

$$V_{kl} = \frac{1}{N} \sum_{i=1}^{N} \mu_{ik} \mu_{il}, \tag{8}$$

where N is the total number of gene expression vectors. The coupling indicates how many genes are shared by two clusters. Clusters which have a low coupling show distinct overall patterns. If the coupling is large, clusters are more similar. Hence, the coupling defines a similarity measure for pairs of clusters.

This allows the analysis of global clustering structures obtained by soft clustering, since relationships between clusters are defined. Similar to hierarchical clustering, the global clustering structure can be examined at different resolutions determined by the cluster number c. For a small c, only the major clusters present in the data are obtained. If c is increased, sub-clusters with distinct patterns emerge. Sub-clusters derived from a major cluster are generally strongly coupled, since they share the overall expression pattern. Finally, soft clustering produces empty clusters for further increase of c. This approach is illustrated in Fig. 9 showing the overall clustering structure for three different settings of cluster number ($c = 12, 18, 24$). For $c = 12$, the coupling between clusters was generally weak (Fig. 9-I). Several isolated clusters were produced. These clusters are also generally stable cluster for variation of FCM parameter m. An example of such clusters is the G1 cluster that remains isolated with increasing c (Fig. 9(d), (e)). This is contrasted by the G2 cluster which was split into two sub-cluster (Fig. 9(a), (b), (c)). Both clusters are strongly coupled, as the overall pattern is similar. However, the two G2 sub-clusters also show some differences. The first sub-cluster (Fig. 9(b)) has a dominant expression peak during the first cell cycle, whereas peaks in both cell cycle are of similar amplitude for the second sub-cluster (Fig. 9(c)). If the cluster number was increased further ($c = 24$), genes tended to be assigned to several clusters (Fig. 9-III). The coupling between the clusters, thus, became stronger and no isolated cluster remained. Empty clusters were generated and showed strong coupling to many other clusters, as they were poorly isolated. Using color-encoding of α-cores, these cluster artefacts are easily detectable and can readily be excluded for follow-up analysis.

5. Discussion and Conclusions

Hard clustering assumes that clusters are well separated and of homogenous structure. This, however, is seldom the case for gene expression data. The large noise component typical for microarray measurements clouds data structures and leads to diffuse and overlapping clusters. Additionally, genes generally underlie a variety of regulatory mechanisms. The strength of expression has to be fine-tuned to enable the living cell to adapt to a large variety of environmental or developmental conditions. In contrast to hard clustering methods, soft clustering can reflect this complexity in gene expression data by assigning genes to clusters in a gradual manner. This capacity also leads to an increased noise robustness of soft clustering. The soft

⟶

Fig. 9. Global clustering structure generated by soft clustering. Three clusterings (I,II,III) for $c = 12, 18, 24$ (and $m = 1.25$) were analyzed and compared. To visualize the global structure of the clustering, a principle component analysis (PCA) based on the cluster centres of clustering III was performed. Cluster centres (derived in a clustering I,II or III) were projected on the two major principle components and represented by red dots. Thus, the PCA was used to project 16-dimensional vector space to two dimension. The width of connecting blue lines indicates the strength of overlap V between soft clusters as defined by Eq. (8). Sub-figures (a)–(e) shows the core structures for several example clusters. Sub-figure (g) presents the colour-encoding of the membership values.

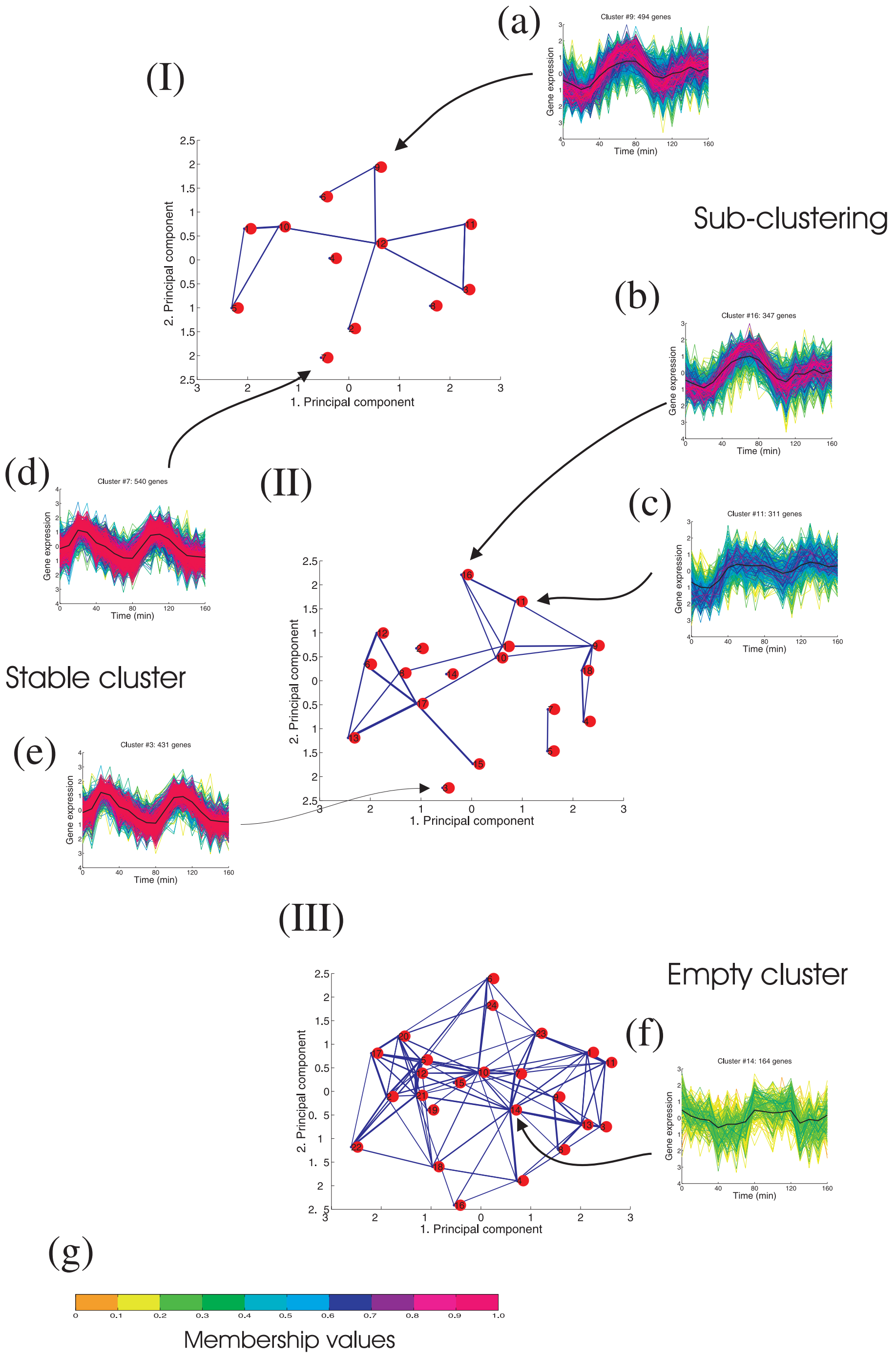


Fig. 9. (*Continued*)

clustering methodology presented in this study can offer the following advantages for microarray data analysis:

(i) *A priori filtering of genes can be replaced by a posteriori filtering*: To reduce the influence of noise, genes are usually filtered before cluster analysis is performed. Setting filtering thresholds, however, remains arbitrary. *A priori* filtering can also prevents the discovery of many important co-regulations where changes in expression are subtle.[16] The use of soft clustering approaches can avoid pre-filtering as it offers increased robustness to noise compared to hard clustering. Expression vectors with a large noise level received a small membership value and contributed less to the determination of the cluster centres. They are distinguished from expression vectors with a large membership value that are generally found close the cluster centroids. Noise can then be reduced *a posteriori* by determining the α-cores of clusters to exclude "noisy" genes from further analysis. The loss of information by pre-filtering of genes can therefore be avoided.

(ii) *Cluster artefacts can be prevented*: Hard clustering such as k-means finds always distinct clusters in a data set, even in purely random data. Generally, cluster algorithms that are based on crisp membership value for cluster members do not indicate how well a data vector is classified, as all members of a cluster have the same membership value. Such cluster artefacts are especially misleading if biological knowledge of the system under study is limited. Soft clustering by soft clustering offers the possibility to avoid the detection of cluster artefacts. This capacity is controlled by tuning the FCM parameter m. Larger values of m lead to a clustering process which is more robust to noise. However, if the parameter m is chosen too large, no cluster at all in the data will be detected. Using the approach presented in this study, an optimal upper threshold for m can be set. Noise robust clustering methods such as FCM are especially desirable, if changes of expression are small or restricted to a subset of genes. Subtle data structures can also be expected for experiments with limited amount of target RNA, which is often the case for studies of human tissue samples. These data frequently contain a high background noise and thus clustering is generally more affected by artefacts.

(iii) *Internal cluster structures become accessible*: Hard partitional clustering methods such as k-means do not generally define any relationships between genes within a cluster and thus lack the ability to indicate sub-structures in clusters. This limitation could be overcome by application of similarity measures to clustering results. Alternatively, soft clustering can be used as it generates naturally internal cluster structures. Relationships between genes within a cluster can be easily defined and visualized. By examining different α-cores for a certain cluster, insight into the underlying cluster structure can be gained. Small α values yield a large variance within the cluster, while larger α values reveal strongly co-expressed genes. This facilitates the discovery of knowledge, as more information about the data structure is captured. The information gain can be used, for example, for better specification of co-expressed genes allowing a more targeted search for regulatory motifs.

Using methods to assess the cluster quality do not address this inherent problem of hard clustering, as they generally only assess the quality of the whole clusters. However, we have observed in the analysis, that the overall correlation within a gene cluster may be weak although a core of strongly correlated genes exists. Applying methods for cluster validity assessment may lead their user to generally discard weak clusters and, thus, to a loss of potentially important information. Using soft clustering with subsequent *a posteriori* filtering can avoid this pitfall and help to reveal the discovery of cis-regulatory motifs even in weak clusters.

(iv) *Global clustering structures can be studied at different resolutions*: A major drawback of most hard partitional clustering method is the lack of relationships generated between single clusters. Especially for genome-wide microarray studies, however, the overall clustering structure can give valuable insights into the underlying biology of the system examined. Notable exceptions are SOMs mapping clusters onto a grid structure.[3,12] However, the initial grid structure remains user-defined and independent of the produced clustering. Hence, it may not reflect the underlying data structure. The use of soft clustering can overcome these limitations, since it allows the definition of coupling between single clusters without inferring with the clustering process. The global clustering structure generated is therefore based solely on the clustering and not on prior assumptions. Ideally, a clustering method should also give insights into the data structures on different scales. The methodology introduced in this study for soft clustering allows the increase in clustering resolution without suffering under the sensitivity of hard clustering towards the detection of clustering artefacts. Local optimal clustering resolution can be reached while cluster artefacts are easily detected as empty clusters. Hence, soft clustering incorporates advantages of current partitional and hierarchical clustering methods. As with partitional clustering, soft clustering as presented here is based on the overall structure allowing for more robust clustering. As with hierarchical clustering methods, it can indicate the different levels of clustering at various resolutions.

Two other clustering studies based on FCM have been recently published. Gasch and Eisen demonstrated for yeast microarray data that FCM identifies clusters previously unrecognised by hierarchical and k-means clustering.[19] They also found that genes were assigned to several clusters if they underlie multiple transcription mechanisms. The work of Dembele and Kastner focused mainly on the selection of parameter m.[20] Our study extends these proposed approaches in several issues: First, the capacity for increased noise robust clustering was examined and demonstrated using the internal cluster structure generated by soft clustering. Second, *a priori* filtering was replaced by *a posteriori* filtering. Thus, pre-filtering was avoided in contrast to both other studies. Third, Dembele and Kastner proposed a heuristically motivated method to select FCM parameter m. However, this method does not guarantee that clustering of random data is prevented. This is also the case for the approach by Gasch and Eisen setting m equals 2 in all their cluster analyses. Fourth, Dembele and Kastner used a graph-theoretical method to select the number of clusters. Since this method is unrelated to FCM, this approach can lead to sub-optimal estimation of the number of clusters. Fifth, both previous studies do not include a construction of global clustering structure that soft clustering can offer by the definition of coupling between clusters.

To our knowledge, this is the first study that focuses the differences between hard clustering and soft clustering. The methodology developed has two main advantages for microarray data analysis: It is robust to noise and produces information-rich cluster structures in contrast to many hard clustering methods. Soft clustering can, therefore, be a desirable complementary approach to current standard clustering methods. The gain in information by applying soft clustering can be used for the discovery of new biological mechanisms. A strong emphasis was put on the visualization of clustering results, since the lack of visualized clustering structures has generally been a drawback of partitional clustering compared to hierarchical clustering. Visualization is, however, an important prerequisite for knowledge discovery in biological and medical research. The development of a software toolbox called *Mfuzz* aims to support the distribution of soft clustering. It is based on the open software R and is freely available from the first author's webpage (http://itb.biologie.hu-berlin.de/∼futschik/software/R/Mfuzz/).

Acknowledgments

We are grateful to Simone Monreal and Anna Ischaut for their on-going assistance in the research project. We also thank the reviewers for their constructive comments on the manuscript. M. F. would like to thank his former PhD supervisors Nik Kasabov, Mike Suillivan, Parry Guilford and Anthony Reeve for past support and Hanspeter Herzel for present support of the research project. M. F. received a PhD scholarship from the University of Otago. The work presented was supported by the *Deutsche Forschungsgemeinschaft (DFG)* by the SFB 618 grant.

References

1. Jain A, Dubes R, *Algorithms for Clustering Data*, Prentice Hall, New Jersey, 1988.
2. Tavazoie S, Hughes JD, Campbell MJ, Cho RJ, Church GM, Systematic determination of genetic network architecture, *Nat Genet*, **22**:281–285, 1999.
3. Törönen P, Kolehmainen M, Wong G, Castren E, Analysis of gene expression data using self-organizing maps, *FEBS Lett*, **451**(2):142–146, 1999.
4. Sharon R, Shamir R, CLICK: a clustering algorithm with applications to gene expression data, *Proceedings of the RECOMB 1999*, pp. 307–316, 1999.

5. Eisen MB, Spellman PT, Brown PO, Botstein D, Cluster analysis and display of genome-wide expression patterns, *Proc Natl Acad Sci USA*, **95**(1):14863–14868, 1998.
6. Spellman PT, Sherlock G, Zhang MQ, Iyer VR, Anders K, Eisen MB, Brown PO, Botstein D, Futcher B, Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridisation, *Mol Biol Cell*, **9**(12):3273–3297, 1998.
7. Cho RJ, Campbell MJ, Winzeler EA, Steinmetz L, Conway A, Wodicka L, Wolfsberg TG, Gabrielian AE, Landsman D, Lockhart DJ, Davis RW, A genome-wide transcriptional analysis of the mitotic cell cycle, *Mol Cell*, **2**:65–73, 1998.
8. Chu S, DeRisi J, Eisen M, Mulholland J, Botstein D, Brown PO, Herskowitz I, The transcriptional program of sporulation in budding yeast, *Science*, **282**(5389):699–705, 1998.
9. Bezdak JC, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, 1981.
10. Geth I, Geva AB, Unsupervised optimal fuzzy clustering, *Trans. Pattern Analysis Machine Intell*, **11**:773–781, 1989.
11. Duda RO, Hart PE, Stork DG, *Pattern Classification*, Wiley, New York, 2001.
12. Tamayo P, Slonim D, Mesirov J, Zhu Q, Kitareewan S, Dmitrovsky E, Lander ES, Golub TR, Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation, *Proc Natl Acad Sci USA*, **96**(6):2907–2912, 1999.
13. Lukashin A, Fuchs R, Analysis of temporal gene expression profiles, *Bioinformatics*, **17**(5):405–414, 2000.
14. Troyanskaya O, Cantor M, Sherlock G, Brown P, Hastie T, Tibshirani R, Botstein D, Altman RB, Missing value estimation methods for DNA microarrays, *Bioinformatics* **17**(6):520–525, 2001.
15. Heyer LJ, Kruglyak S, Yooseph S, Exploring expression data: identification and analysis of coexpressed genes, *Genome Res*, **11**:1106–1115, 1999.
16. Hughes TR, Marton MJ, Jones AR, Roberts CJ, Stoughton R, Armour CD, Bennett HA, Coffey E, Dai H, He YD, Kidd MJ, King AM, Meyer MR, Slade D, Lum PY, Stepaniants SB, Shoemaker DD, Gachotte D, Chakraburtty K, Simon J, Bard M, Friend SH, Functional discovery via a compendium of expression profiles, *Cell*, **102**(1):109–126, 2000.
17. Roth FP, Hughes JD, Estep PW, Church GM, Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation, *Nature Biotech*, **16**:939–945, 1998.
18. Hughes JD, Estep PW, Tavazoie S, Church GM, Computational identification of cis-regulatory elements associated with groups of functionally related genes in Saccharomyces cerevisiae, *J Mol Biol*, **296**:1205–1214, 2000.
19. Gasch AP, Eisen MB, Exploring the conditional coregulation of yeast expression through fuzzy k-means clustering, *Genome Biology*, **3**(11):research0059.1-0059.22, 2002.
20. Dembele D, Kastner P, Fuzzy c-means method for clustering microarray data, *Bioinformatics*, **19**(8):973–980, 2002.

Matthias E. Futschik is currently Research Fellow at the Institute for Theoretical Biology, Berlin, Germany. He received his Ph.D. in Information Science at the University of Otago, Dunedin, New Zealand. His main research interests are integrative bioinformatics and systems biology.

Bronwyn Carlisle holds a Bachelor of Science degree in biochemistry and Graduate Diploma degree in Information Science. She joined the Biochemistry Department of the University of Otago in 1992 as Junior Research Fellow. She is now working as Scientific Officer at the same department.