BIOINFORMATICS

Vol. 17 no. 2 2001
Pages 126–136

A hierarchical unsupervised growing neural network for clustering gene expression patterns

Javier Herrero[1], Alfonso Valencia[2] and Joaquín Dopazo[1,*]

[1]Bioinformatics, CNIO, Ctra. Majadahonda-Pozuelo, Km 2, Majadahonda,
28220 Madrid and [2]Protein Design Group CNB-CSIC, 28049 Madrid, Spain

Received on August 6, 2000; revised on September 29, 2000; accepted on October 6, 2000

ABSTRACT

Motivation: We describe a new approach to the analysis of gene expression data coming from DNA array experiments, using an unsupervised neural network. DNA array technologies allow monitoring thousands of genes rapidly and efficiently. One of the interests of these studies is the search for correlated gene expression patterns, and this is usually achieved by clustering them. The Self-Organising Tree Algorithm, (SOTA) (Dopazo,J. and Carazo,J.M. (1997) *J. Mol. Evol.*, **44**, 226–233), is a neural network that grows adopting the topology of a binary tree. The result of the algorithm is a hierarchical cluster obtained with the accuracy and robustness of a neural network.

Results: SOTA clustering confers several advantages over classical hierarchical clustering methods. SOTA is a divisive method: the clustering process is performed from top to bottom, i.e. the highest hierarchical levels are resolved before going to the details of the lowest levels. The growing can be stopped at the desired hierarchical level. Moreover, a criterion to stop the growing of the tree, based on the approximate distribution of probability obtained by randomisation of the original data set, is provided. By means of this criterion, a statistical support for the definition of clusters is proposed. In addition, obtaining average gene expression patterns is a built-in feature of the algorithm. Different neurons defining the different hierarchical levels represent the averages of the gene expression patterns contained in the clusters.

Since SOTA runtimes are approximately linear with the number of items to be classified, it is especially suitable for dealing with huge amounts of data. The method proposed is very general and applies to any data providing that they can be coded as a series of numbers and that a computable measure of similarity between data items can be used.

Availability: A server running the program can be found at: http://bioinfo.cnio.es/sotarray

Contact: jdopazo@cnio.es

INTRODUCTION

DNA array technologies (Schena *et al.*, 1995; Shalon *et al.*, 1996; Lockhart *et al.*, 1996) have opened new ways of looking at organisms in a wide-genomic manner. The study of the expression of the genes of a complete genome, in the case of yeast (DeRisi *et al.*, 1997; Eisen *et al.*, 1998; Wodicka *et al.*, 1997; Cho *et al.*, 1998), is now possible using such techniques. Studies involving human genes (Alon *et al.*, 1999; Iyer *et al.*, 1999; Perou *et al.*, 1999) or other eukaryotic organisms (Lockhart *et al.*, 1996) have been carried out using DNA arrays too. And most probably, in only a few years, DNA arrays of the complete human genome will be available. Drug discovery will be a field to especially benefit by the use of DNA array technologies (Debouck and Goodfellow, 1999). For example, these technologies have been successfully applied to drug target identification (Kozian and Kirschbaum, 1999), development (Gray *et al.*, 1998) and validation (Marton *et al.*, 1998).

A problem inherent to the use of DNA array technologies is the huge amount of data produced, whose analysis in itself constitutes a challenge. Several approaches, including hierarchical clustering, multivariate analysis and neural networks have been applied to the analysis of gene expression data. Despite the arsenal of methods used, the optimal method for analysing such data is still open to discussion.

Hierarchical clustering (Sneath and Sokal, 1973) is the most widely used method for the analysis of patterns of gene expression. It produces a representation of the data with the shape of a binary tree, in which the most similar patterns are clustered in a hierarchy of nested subsets. These techniques have already been applied to the study of gene expression patterns (Eisen *et al.*, 1998; Iyer *et al.*, 1999; Wen *et al.*, 1998). Nevertheless, classical hierarchical clustering presents drawbacks when dealing with data containing a non-negligible amount of noise, as is the case. Several authors (Tamayo *et al.*, 1999) have noted that hierarchical clustering suffers from a lack of robustness and solutions may not be unique, and

[*]To whom correspondence should be addressed.

dependent on the data order. Also, the deterministic nature of hierarchical clustering and the impossibility of re-evaluating the results in light of the complete clustering of the data, can cause some clusters of patterns to be based on local decisions rather than on the global picture (Tamayo *et al.*, 1999). Other different clustering methods have recently been proposed (Heyer *et al.*, 1999; Ben-Dor *et al.*, 1999), but their performance remains to be evaluated by the user community.

These arguments lead to the use of neural networks as an alternative to hierarchical cluster methods (Tamayo *et al.*, 1999; Törönen *et al.*, 1999). Unsupervised neural networks, and in particular self-Organising Maps (SOM) (Kohonen, 1990, 1997), provide a more robust and accurate approach to the clustering of large amounts of noisy data. Neural networks have a series of properties that make them suitable for the analysis of gene expression patterns. They can deal with real-world data sets containing noisy, ill-defined items with irrelevant variables and outliers, and whose statistical distributions do not need be parametric. SOM are reasonably fast and can be easily scaled to large data sets. They can also provide a partial structure of clusters that facilitate the interpretation of the results. SOM structure, unlike in the case of hierarchical cluster, is a two-dimensional grid usually of hexagonal or rectangular geometry, having a number of nodes fixed from the beginning. The nodes of the network are initially random patterns. During the training process, that implies slight changes in the nodes after repeated comparison with the data set, the nodes change in a way that captures the distribution of variability of the data set. In this way, similar gene expression patterns map close together in the network and, as far as possible from the different patterns. At the end of the training process, the nodes of the SOM grid have clusters of patterns assigned, and the trained nodes represent an average pattern of the cluster of data that map into it. This reduction of the data space is a very interesting property when dealing with big data sets, which is often the case in DNA array data (Herwig *et al.*, 1999).

Nevertheless, this approach presents several problems (Fritzke, 1994). Firstly, the SOM is a topology-preserving neural network. In other words: the number of clusters is arbitrarily fixed from the beginning. This makes the recovering of the natural cluster structure of the data set a very difficult and subjective task. The training of the network (and, consequently, the clusters) depends on the number of items. Thus the clustering obtained is not proportional. If irrelevant data (e.g. invariant, ‘flat’ profiles) or some particular type of profile is abundant, SOM will produce an output in which this type of data will populate the vast majority of clusters. Because of this, the most interesting profiles will map in few clusters and their resolution might be low. Finally, the lack of a tree structure makes it impossible to detect higher order relationships between clusters of profiles.

Within this context, the Self-Organising Tree Algorithm (SOTA) (Dopazo and Carazo, 1997), an unsupervised neural network with a binary tree topology provides a good solution. SOTA combines the advantages of both approaches, hierarchical clustering and SOM, and is free of the problems these methods present when applied to gene expression profiles. The result of the algorithm is a hierarchical clustering achieved with the accuracy and robustness of a neural network.

The SOTA was first described by Dopazo and Carazo (1997) as a new type of self-organising neural network based on both the SOM maps of Kohonen (1990) and the growing cell structures (Fritzke, 1994), but implementing a new topology and a different strategy of training. It was applied to cluster sets of aligned sequences. Later it was used for clustering sequences using as data their dipeptide frequencies (Wang *et al.*, 1998b) and to cluster amino acids in classes based on their physico-chemical properties (Wang *et al.*, 1998a). Thus SOTA has demonstrated the ability to successfully cluster data of different nature. We propose here an application of this algorithm to DNA array data, and show how a statistical method for the definition of clusters can be implemented in the network.

ALGORITHM AND IMPLEMENTATION

The Self-Organising Tree Algorithm

SOTA is based both on the SOM (Kohonen, 1990) and the growing cell structures (Fritzke, 1994). The algorithm proposed by Kohonen generates a mapping from a complex input space to a simpler output space. The input space is defined by the experimental input data, whereas the output space consists of a set of nodes arranged according to certain topologies, usually two-dimensional grids. The application of the algorithm maps the input space onto the smaller output space, producing a reduction in the complexity of the analysed data set. In the case of SOTA, the output is a binary tree topology that incorporates the principles of the growing cell structures algorithm of Fritzke (1994). In this algorithm a series of nodes, arranged in a binary tree, are adapted to the intrinsic characteristics of the input data set. As in the growing cell structures, the output space can grow to fit as much as possible to the variability of the input space. The growing of the output nodes can be stopped at the desired taxonomic level or, alternatively, they can grow until a complete classification of every gene in the input data set is reached.

Encoding the data

Each DNA array contains the measures of the level of expression for many genes. These values are usually obtained by measuring the fluorescence intensity and subtracting the background (see, for example, Eisen *et al.*, 1998, for details on the experimental procedure). Each DNA array can be considered as a single measure of the expression of many genes for a given condition (e.g. timepoints, a particular concentration of a product, etc.) Gene expression profiles are obtained from the different DNA arrays of an experiment collecting, for any particular gene, the intensity of its expression in each array. Data are arranged in tables where rows represent all genes for which data has been collected and columns represent the individual expression values obtained in each DNA array. Raw data often display highly asymmetrical distributions that make difficult the use of a distance to assess differences among them. Therefore, it is quite unusual to use the data directly , without a previous transformation. There are several transformations currently used with different purposes, depending on the problem that may affect to the data. *Square* transformation compresses the scale for small values and expands it for large values. The opposite effect is achieved with *square root*, *logarithm* and *inverse* transformations. Since gene expression values are given as ratios of the expression under a given condition to the expression under a reference condition, *logarithmic* transformation can be considered the most suitable option because it provides a symmetrical scale around 0. Each gene profile is a vector identified by the name of the gene, which contains as many values as points have been measured. The values are obtained from the original ones and transformed using logarithm 2. Then, for the sake of the adaptation process of the network, all the vectors were normalised to have a mean of zero and a standard deviation of 1.

The distance function

Depending on the concept by which we want to cluster patterns of expression, different types of distances can be used. Distances are obtained from the pair-wise comparison of patterns of gene expression.

If we have two genes with their corresponding expression patterns: $gene_1$ $(e_{11}, e_{12}, \ldots, e_{1n})$ and $gene_2$ $(e_{21}, e_{22}, \ldots, e_{2n})$, different distances are obtained as follows. *Euclidean* distance is obtained as the square root of the summation of the squares of the differences between all pairs of corresponding values.

$$d_{1,2} = \sqrt{\sum_i (e_{1i} - e_{2i})^2}.$$

An equivalent distance, the *squared Euclidean* distance, is the square of the *Euclidean* distance. Generally speaking, these types of distances are suitable when the aim is to cluster genes displaying similar levels of expression.

Another extensively used distance function is the *Pearson correlation coefficient*, r. It gives values between -1 (negative correlation) and 1 (positive correlation). The more the two profiles have the same trend; the closer to 1 is the r-value, irrespective of their absolute values of expression. The distance is obtained then as follows:

$$d_{12} = (1 - r) = 1 - \frac{\sum_i ((e_{1i} - \hat{e}_1)(e_{2i} - \hat{e}_2))/n}{Se_1 Se_2}.$$

Where: $\hat{e}_n$ and Se_n are the mean and the standard deviation of all the points of the nth profile, respectively. A similar distance for measuring trends is *the correlation coefficient with an offset of 0.* In this case, the distance is obtained from the correlation coefficient in the same way as in the previous case: $d_{12} = (1 - r)$, but considering zero as reference, instead of the mean value of the distribution. This maybe an interesting choice in cases where the data are serial measures with respect to an initial state of reference (time series, dosage series, etc.) This transformation is used by Eisen *et al.* (1998).

SOTA dynamics

The initial system is composed of two external elements, denoted as cells, connected by an internal element (see Figure 1A), that we will call node. Each cell (or node) is a vector with the same size as the gene profiles. Each component in the vector corresponds to a column in the data set, that is, to one of the conditions under which the gene expression has been measured. In the beginning, the entries of the two cells and the node are initialised with the mean values of the corresponding column of the data set. Initialising them to random values produces identical results (data not shown).

In addition to the topology, this type of network has another feature that makes it different from previous growing cell approaches (Fritzke, 1994): only cells, but no nodes, are compared to the input gene profiles. Due to this, the network is trained only through their terminal neurons, or cells. The algorithm proceeds by expanding the output topology starting from the cell having the most heterogeneous population of associated input gene profiles. Two new descendants are generated from this heterogeneous cell that changes its state from cell to node. The series of operations performed until a cell generates two descendants is called a cycle. During a cycle, cells and nodes are repeatedly adapted by the input gene profiles (see below).

This process of successive cycles of generation of descendant cells can last until each cell has one single input gene profile assigned (or several, identical profiles), producing a complete classification of all the gene profiles. Alternatively, the expansion can be stopped at the desired level of heterogeneity in the cells, producing in this way a classification of profiles at a higher hierarchical level.

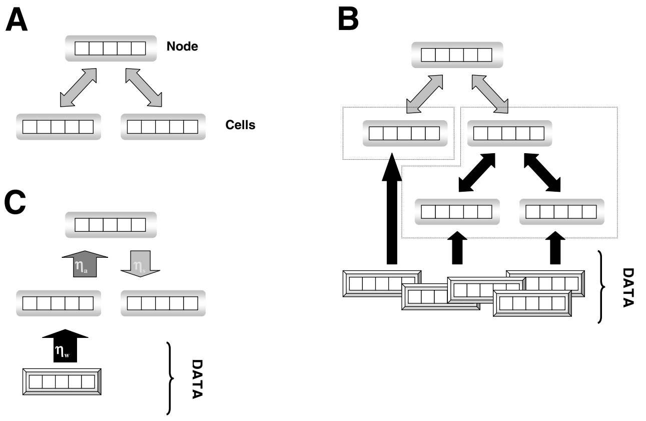

Fig. 1. Schematic representation of the topology of the SOTA network and the growing algorithm. The neurons that compose the network are represented as vectors whose components correspond to the columns of the data matrix, that is, to the conditions at which gene expression values have been measured. (A) Initial state of the network: two terminal neurons, called cells, connected by an internal neuron called node. Arrows account for the possible interactions in the system. (B) The two types of neighbourhood: restrictive and local. As the growth of the network proceeds, internal nodes remain stable (grey arrows means that the corresponding updates are not performed anymore) and updating events only takes place in the external nodes (cells) and their corresponding neighbourhoods (black arrows show the permitted updating events allowed for the topology shown). In order to avoid asymmetrical updating, two different types of neighbourhood are used. The restrictive one can be seen on the left. Since the mother neuron of the cell is not receiving updates from the other side, this cell does not update its mother. The local one can be seen in the right side. Both sister neurons receive update and transmit to each other as well as to their mother node. (C) Pathway of interactions. Once a neuron (bottom left in the example) has been chosen as the winning cell it is adapted (see text) with a factor η_{W}. The strength of the updating decreases as we go further in the neighbourhood. Thus the values of η_{a} and η_{s} for the updating of the mother neuron and the sister neuron, respectively, are consecutively lower. The darker the arrow the stronger is the interaction.

Adaptation process

Adaptation in each cycle is carried out during a series of epochs. Each epoch consists on the presentation of all the expression profiles to the network. A presentation implies two steps: first, finding the best matching cell (winning cell) for each expression profile, that is, the cell with the lowest distance cell-profile ($\boldsymbol{d}_{pc}$) and second, to update this cell and its neighbourhood. Cells are updated by means of the following formula (Kohonen, 1990):

$$\boldsymbol{C}_i(\tau + 1) = \boldsymbol{C}_i(\tau) + \eta \cdot (\boldsymbol{P}_j - \boldsymbol{C}_i(\tau))$$

where η is a factor that accounts for the magnitude of the updating of the ith cell depending on its proximity to the winning cell within the neighbourhood, $\boldsymbol{C}_i(\tau)$ is the ith cell vector at the presentation τ, and $\boldsymbol{P}_j$ is the jth gene expression profile vector. The topological neighbourhood of the winning cell is very restrictive (Dopazo and Carazo, 1997; Fritzke, 1994), unlike in the case of SOM. Two different types of neighbourhood are used. If the sister cell of the winning cell has no descendants (both sister cells are the only descendants of the ancestor node), the neighbourhood includes the winning cell, the ancestor node and the sister cell, otherwise it includes only the winning cell itself (see Figure 1B). We have used the decreasing values η_{w}, η_{a}, and η_{s} for the winning cell, the ancestor node and the sister cell, respectively (see Figure 1C).

There is a particular case for the adaptation process: when both sister cells are equal. This occurs in the initial stage of the network and just after a cell duplicates, giving rise to two new sister cells. In this case the first cell to which the profile is compared is taken as winner by default. Since the adaptation process is asymmetrical, its effect on the winner is stronger than on the sister. Then, the winner is dragged closer to the profile presented by the other cell. This small difference allows that the remainder profiles in the data set, more similar to this one, tend to map in the first cell, and the rest in the other cell. Since the adaptation depends on the expression values of the profiles, the first group to segregate is always the one that is less similar to the average value in the cell, irrespective of the presentation order. The asymmetry is due to the use of different, decreasing values for the η-factors. Typical values are: $\eta_{\mathrm{w}} = 0.01$, $\eta_{\mathrm{a}} = 0.005$ and $\eta_{\mathrm{s}} = 0.001$ (see Dopazo and Carazo, 1997).

The heterogeneity under each cell is computed by its resource, R. This value will be used to direct the growth of the network by replicating, at the end of each cycle, the cell with the largest resource value (Dopazo and Carazo, 1997; Fritzke, 1994; Kohonen, 1990). The resource is defined as the mean value of the distances among a cell and the expression profiles associated to it:

$$R_i = \frac{\sum_{k=1}^{K} d_{P_k C_i}}{K}$$

where the summation is done over the K profiles associated to the cell i.

Growing, convergence and end conditions

The criteria used for monitoring the convergence of the network is the total error, ε, which is a measure of how close the expression profiles are to their corresponding winning cell after an epoch. The error is defined as the summation of the resource values of all the cells that are being presented at the epoch t:

$$\varepsilon_t = \sum_i R_i.$$

Thus, a cycle finishes when the relative increase of the error falls below a given threshold:

$$\left|\frac{\varepsilon_t - \varepsilon_{t-1}}{\varepsilon_{t-1}}\right| < E$$

(see the next section below for a discussion on criteria for choosing thresholds). The network follows its growing process by replicating the cell with the highest resource value. This cell gives rise to two new descendant cells and becomes a node. The values of the two new cells are identical to the node that generated them.

The growing process ends when the heterogeneity of the system falls below a threshold. Two measures of the heterogeneity of the system have been used in this work. One of them is the resource value, R, of the network, that is the maximum resource value among all the cells. And the other one is the variability, V. Lets define D_i as the maximum value among all the possible profile–profile distances between all pairs of genes belonging to cell i. D_i, can be considered an alternative way of measuring the heterogeneity in the cells. Then, the variability V is defined as the maximum value among these D_i-values:

$$V = \max_i\{D_i\};$$
$$D_i = \max_{jk}\{d_{P_j P_k}\}.$$

In this way, the network can be instructed to grow until the desired hierarchical level. If the threshold is chosen to be zero, the network will grow until every cell has associated either one unique profile or several identical profiles. On the other hand, different threshold values will cause the network to stop at higher hierarchical levels, clustering in single cells those groups of profiles whose heterogeneity falls below the threshold.

Confidence intervals and definition of clusters

One of the most complicated problems is the definition of a non-subjective criterion to stop the growing of the tree. Since the aim of clustering is to find groups of genes having a similar expression profile, what we need is to define the upper level of distance at which two genes can be considered to be similar at their profile expression levels. This distance depends on the values contained in the data set. For example, if many genes with profiles of very few points are analysed, it is likely that, randomly, some of them display very similar patterns. On the contrary, if many points compose gene profiles, the possibility for two profiles being similar by chance is very low. Figure 2 shows the distribution of the coefficient of correlation in random, unrelated profiles with different number of points (from 5 to 21). As expected, the mean value is zero, but there exist a significant number of cases of both high positive and negative correlations that arise purely by chance, when the number of points is low.

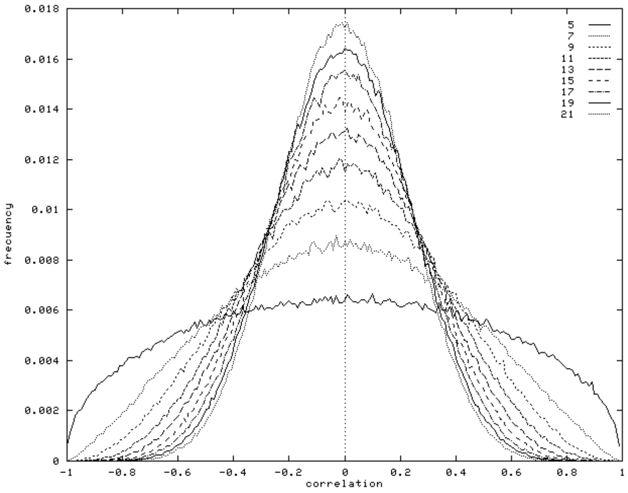

Fig. 2. Distribution of the coefficient of correlation in random, unrelated profiles in the cases of profiles of different numbers of points. Labels account for the distributions obtained for profiles ranging from 5 to 21 points. The mean value is zero in all the cases, because the data are uncorrelated, but the lower the number of points, the higher is the probability of finding positive and negative correlation by chance.

If the random distribution of the values of the measure of distance used for quantifying the degree of similarity between pairs of gene expression profiles were known, a one-tail test could be applied. In this case, a confidence level α could be defined for a given distance value threshold. The confidence level means that distance values like this threshold or lower are found only in the proportion of α when comparing two unrelated genes. Or, in other words, the probability of taking two random profiles as identical is lower than α if the distance between them is smaller than the threshold. In this way, the number of misassignments of items to a cluster is minimised to a threshold fixed by the α-value.

The true random distribution of the distance value is not known, but an approximation can be obtained by resampling the original data set (Efron and Tibsirani, 1991). For each profile, all the points are randomly shuffled. That destroys the actual correlation among the different profiles, whereas the rest of the characteristics of the data set (number of points, ranges of values, frequency of values) are conserved.

Figure 3 shows an example using the data of the yeast cellular cycle (Spellman *et al.*, 1998) (see http://cellcycle-www.stanford.edu). The measure chosen is the correlation coefficient. The internal distribution (continuous line) corresponds to the randomised data, and

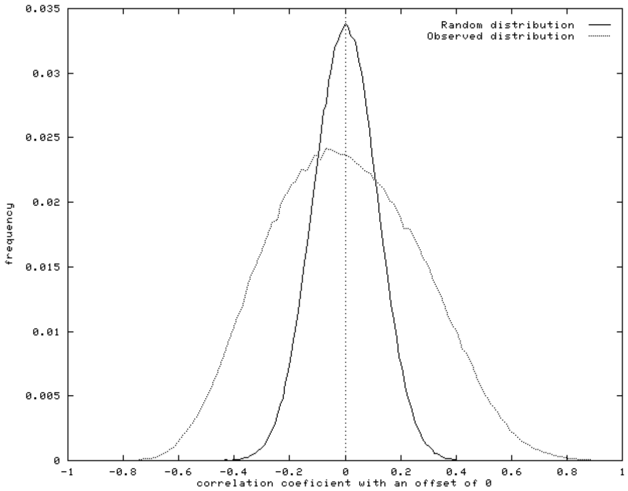

Fig. 3. Distribution of the observed values of coefficient of correlation (dotted line) and the values obtained after shuffling the values of the profiles and calculating again all the coefficients of correlation for the pairs of randomised profiles (continuous line). The data are from the yeast cellular cycle (Spellman *et al.*, 1998) and consist of 800 genes for which 78 data points had been measured. Real data contains much more positive and negative correlation than that expected from a random distribution. It is worth noticing that the actual distribution (dotted line) is biased towards positive correlation values. This points to the fact that in this real data set there are more genes whose expression patterns display a positive correlation than genes with negatively correlated patterns.

the external one (dotted line) is the distribution obtained when comparing all the possible pairs of profiles in the real data. The 95% of the coefficient correlation values are below 0.178, that is, a distance of 0.882 ($= 1 - 0, 178$). So, if we choose this value as threshold, the probability of having two uncorrelated profiles with a correlation coefficient higher than 0.178 is smaller than the 5%. If such a threshold is applied to the data set, the dendogram grows until the variability in any cluster is below this threshold (see Figure 4B). In this way we have a statistical assessment of the content of each cluster.

SOTA allows the dimensionality of the system to be reduced. DNA array data usually consist of a huge amount of genes (several thousands in many cases). What is immediately obvious is that such an amount of data cannot be easily analysed by eye, even in the case of reconstructing a complete clustering of all the items using any hierarchical clustering method. SOTA provides the possibility of reducing the problem to a scale in which differences and similarities among patterns can be analysed in an easier way. Figure 4A shows a low-resolution picture of the clustering obtained for the data set. Clusters contain items whose inter-profile distance (scored as explained above for the coefficient of correlation with an offset of 0) is below 0.75. Figure 4B shows the clustering obtained at higher resolution displaying the 174 patterns that are really different at a confidence level of 5%, out of the 800 gene expression profiles contained in a data set. The number of significantly different profiles is usually not very large compared to the original data size. A similar analysis by hierarchical clustering would have resulted in a densely branched tree of 800 genes, whose interpretation had been very difficult, if not impossible. Moreover, if lower resolution is required two alternatives can be taken. One of them is to reduce the confidence level. If, for example, we choose 10% of reliability, instead of the 5% previously chosen, the corresponding threshold would be a lower value for the coefficient of correlation, giving rise to less clusters, that is, a dendogram reflecting a higher hierarchical level. The other alternative is to choose the number of clusters to which we can reduce the system. In this case the dendogram will grow until the specified number of clusters is reached, irrespective of the reliability of the clusters obtained. Obviously, the higher the hierarchical level the lower the reliability of the clusters. This is due to the fact that as we climb in the hierarchy of clusters, the upper level clusters have less and less defined average profiles, and the probability of having wrongly assigned gene profiles is higher. Or, in other words, it is more difficult to warrant that higher hierarchical relationships are true than affirm the same of lower hierarchical relationships. This fact, which is true in any case independently of the cluster method used, can be monitored by SOTA. The algorithm calculates after each cycle the resource and the variability value. The variability can be related to the random distribution, giving the reliability of the cluster under this node. SOTA has built-in the assessment of the reliability of any cluster in the whole hierarchy. Figure 4A and B show how SOTA allows the management of the resolution with which the system is going to be analysed producing dendograms that describe the system at different hierarchical levels.

Nodes as averages of expression profiles

As previously mentioned the training process causes that the initially random vectors in the cells approach to weighted averages of the profiles associated to them (Kohonen, 1990). Figure 5 shows examples of expression profiles in nodes, cells (terminal nodes) and genes in the cluster corresponding to the cell. The profiles gathered under the cells are highly correlated and the cell vector constitutes an average of them. In fact, the comparison between the cell vector values and the average values obtained from the profiles display a very low discrepancy (less than 0.3%). Due to the way in which the network is trained, this convenient node feature of representing an average of the items (either nodes or genes in the case of terminal nodes) below them can be extended to all the

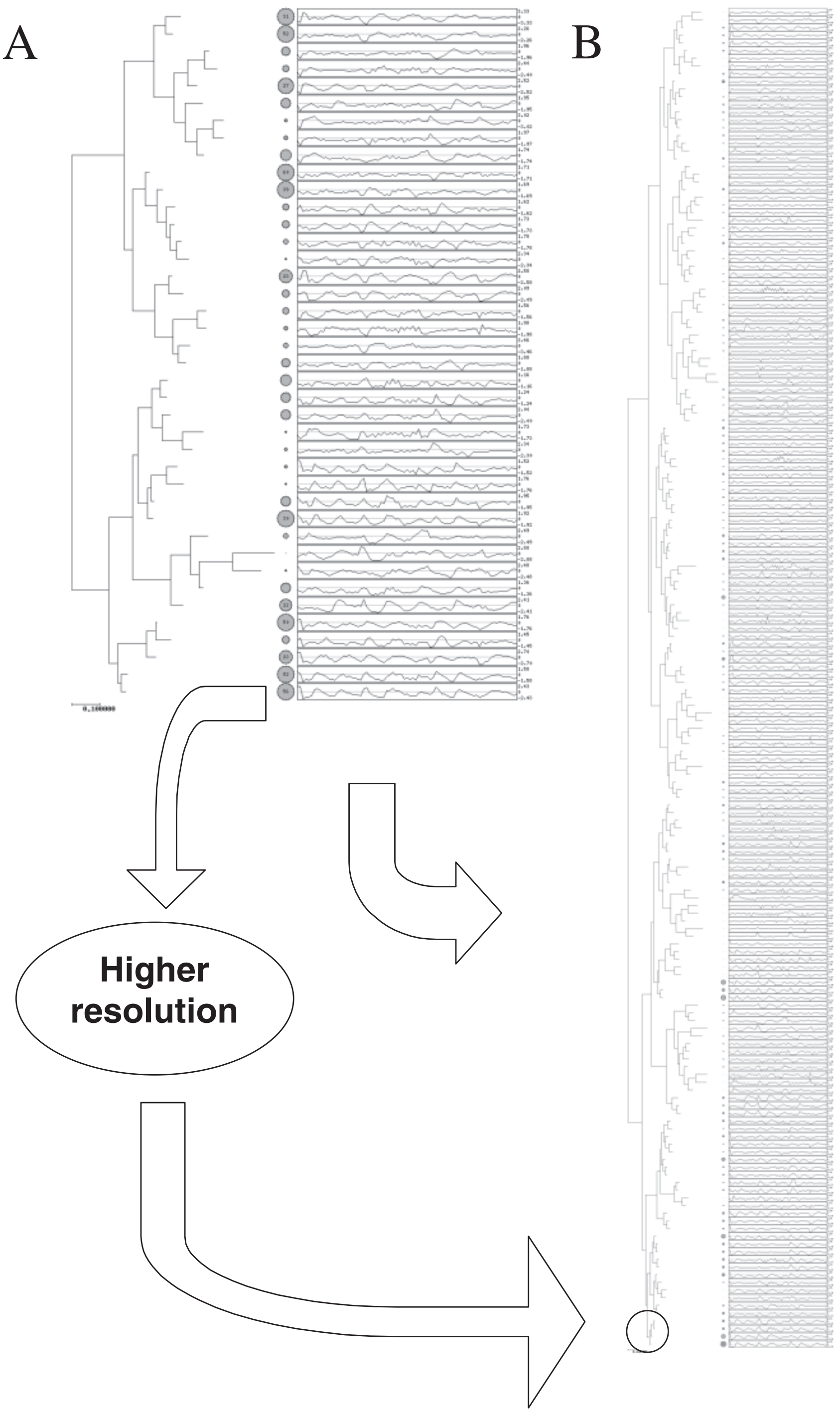

Fig. 4. Dendograms obtained for the data analysed in Figure 2. The data have been normalised (standard deviation of 1 and mean of 0) and the distance used was a coefficient of correlation with an offset of 0. The parameters used for training the network were $\eta_\mathrm{w} = 0.1$, $\eta_\mathrm{a} = 0.05$ and $\eta_\mathrm{s} = 0.01$. (A) The end condition for stopping the growing of the dendogram was to reach a heterogeneity threshold of less than 0.75 for the chosen distance. In this case, the 800 genes involved in the yeast cellular cycle experiment (Spellman *et al.*, 1998) have been grouped into 40 different clusters. (B) The stop condition in this case was to reach a threshold of $\alpha = 5\%$ of probability of including wrong patterns. Now, the 800 have been grouped into 174 different clusters. It can be seen how the cluster in the bottom of the dendogram splits into four new clusters in the higher resolution dendogram. The resolution in the dendogram (B) is approximately four times the resolution of dendogram (A). Circle diameters are proportional to the number of profiles in each cluster.

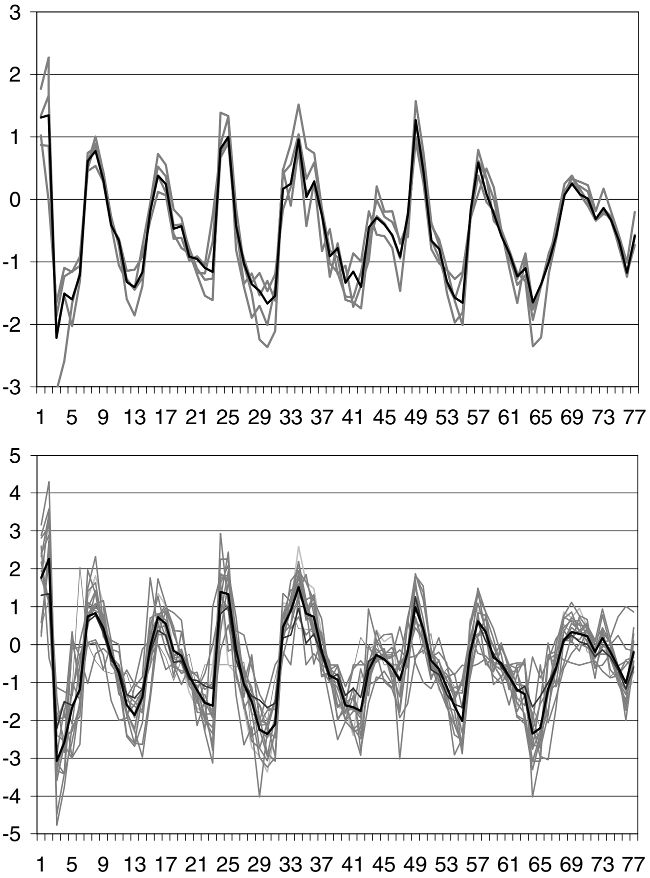

Fig. 5. Values of the node vectors and gene expression profiles obtained after the training process by SOTA. Top Average expression profile of the node in the bottom of the dendogram in (Figure 4A) (black line), together with node vectors corresponding to the four nodes in which this node splits in the higher resolution dendogram shown in (Figure 4B) (grey lines). Bottom Average expression profile in one of the nodes within the circle in (Figure 4B) (thick black line), together with the 18 expression profiles in the cluster (thin grey lines), and the average expression profile in the bottom node of (Figure 4A) (thin black line). The values of the cell vectors (SOTA averages) are very close to the average values obtained from the profiles for each point. Only a 0.3% disagreement among both averages was found for the clusters in the figure.

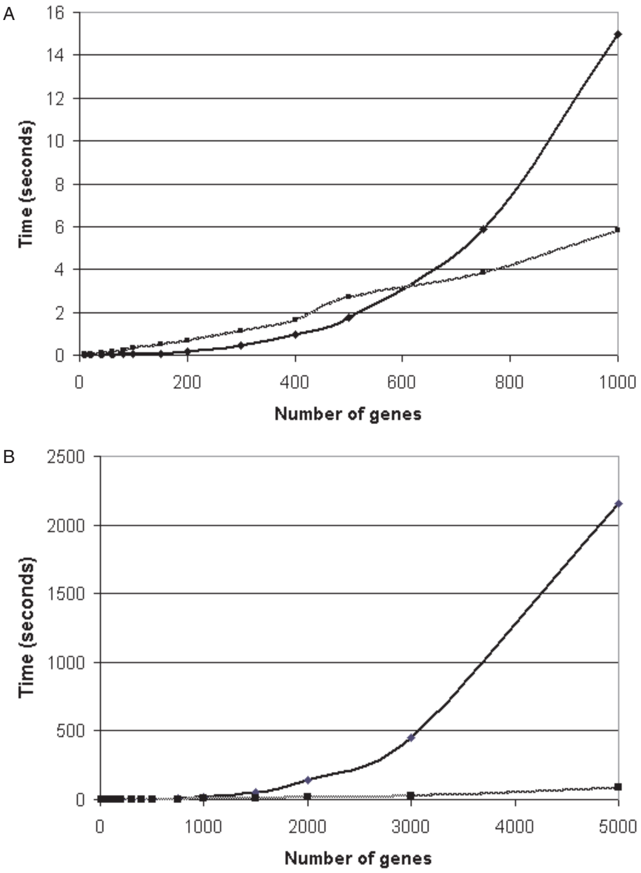

Fig. 6. Comparison between SOTA linear (dotted line) and UPGMA quadratic (continuous line) runtimes. (A) For a moderate number of genes (<1000) SOTA and UPGMA performances are similar. In fact, for less than 600 genes UPGMA is slightly faster. This is a consequence of the time used by SOTA in the initial training steps of the neural network. (B) For a larger number of genes, SOTA is clearly faster than UPGMA. For 5000 genes is around three orders of magnitude faster.

nodes at any level. This allows the study of high-level correlation (either positive or negative) between clusters of genes, instead of among individual genes, and can be very useful for the study of networks of interaction in genomes or in systems for which little information is available.

All against one versus all against all: the way to linear runtimes

One of the interesting properties of this type of neural network is that the most time-consuming comparison operations are performed among the data and one single node (see Figure 1B). The obvious advantage derived from this fact is that the number of comparisons needed for the classification depends, mainly, on the number of items. In the case of classical clustering (UPGMA and related methods, see Hartigan, 1975), the most time-consuming operations are performed on a distance matrix, whose size is proportional to the square of the number of items. In fact, runtime for the clustering procedures used in phylogenetic reconstruction are N^2 to N^4 (Hartigan, 1975).

If runtimes between both approaches are compared (see Figure 6), a similar behaviour can be observed when the number of genes to analyse is small (less than 600). In fact, in this range, UPGMA is faster than SOTA, because the training of the neural network implies a minimum number of presentations. Nevertheless this trend changes drastically for values over 600 genes. For 5000 genes SOTA runtimes are three orders of magnitude faster than UPGMA runtimes (Figure 6B). The runtimes in Figure 6 were obtained in a SGI O200.

DISCUSSION

SOTA is an unsupervised neural network that grows as a binary tree describing, at different levels, the hierarchical relationships between the items analysed, based on an appropriate distance function. Since the growing can be stopped at the desired level of variability, SOTA provides a natural way for defining the actual cluster structure in the set of data. Contrary to the classical hierarchical clustering algorithms (agglomerative), SOTA starts with a high level hierarchy of two neurons (connected by means of a third one that transmit the input signal). After a training cycle, in which the data set is segregated into two groups, the neuron having the most variable population splits in two new sister neurons. The process is repeated until a threshold of heterogeneity is reached for each neuron. The way in which this threshold is defined provides different functionality to SOTA. The heterogeneity threshold can be set to zero for a fully resolved dendogram. In this case the result is similar to that provided by a method of hierarchical clustering. If the heterogeneity threshold is obtained from the randomised distribution of data, SOTA will provide the cluster hierarchy that minimises the probability of having misassigned genes to them. Finally, if the condition to stop the growing of the binary tree is to reach a given number of clusters, SOTA becomes similar to SOM. Nevertheless, SOTA has two crucial advantages: the topology is that of a hierarchical tree, and the clustering obtained is proportional to the heterogeneity of the data, instead of to the number of items. Thus, if a given type of profile is abundant, all the similar items will remain grouped together in a single cluster and they will not directly affect to the rest of the clustering process performed by the network. This last property is due to the fact that SOTA is distribution preserving while SOM is topology preserving (Dopazo and Carazo, 1997; Fritzke, 1994).

In both SOM and SOTA, the training process changes the vectors in the nodes to weighted averages of the gene expression patterns associated to them (Kohonen, 1990). The advantage in the case of SOTA is that the binary topology produces a nested structure in which nodes at each level are averages of the items below them (items that can be nodes or in the case of terminal nodes, genes). This makes it straightforward to compare average patterns of gene expression at different hierarchical levels even for large data sets.

Table 1 lists the differences between classical hierarchical clustering methods, SOM and SOTA. They are related to the type of structure in which the results are arranged, the way in which the different algorithms proceed, and the reliability of the results. SOM and SOTA are, as neural networks, more robust against noise, which is extremely important in the case of data like profiles of gene expression.

Despite the advantages that SOM presents when compared to classical hierarchical cluster methods, it also has some drawbacks. The rectangular, two-dimensional topology is not of much help for the definition of clusters. All the high level hierarchical relationships are lost in this representation. Moreover, the necessity of arbitrarily fixing the number of clusters from the beginning introduces a bias towards this size in the final structure of the results.

On the other hand, classical hierarchical clustering methods, when applied to large amounts of data, produce pictures of difficult interpretation. Obviously, with the appropriate software, the results of a fully developed tree obtained by a classical hierarchical method can be represented at different higher hierarchical levels. But, in any case, the tree must be completely constructed before by a method whose runtimes are quadratic.

An additional advantage of neural networks when compared to classical hierarchical clustering methods is the fact that all the original data are used for defining the clusters during the whole training process. In the case of classical hierarchical clustering the information contained in the data is coded as a distance matrix that is averaged many times. This distance matrix suffers a process of sequential transformations during the definition of the cluster hierarchy that produces a gradual lack of identity of the data that, in addition, can be dependent on the order in which the data are placed in the matrix.

One of the most interesting properties of SOTA is its approximately linear runtimes. This property, together with the possibility of constructing high level trees, makes SOTA a really fast approach to the analysis of large gene expression data sets.

The performance of any method depends critically on the use of an appropriate distance function with the adequate biological meaning. Often it is also necessary to make a transformation of the data before proceeding with the analysis. SOTA includes Euclidean distances (point to point differences between the patterns) and pattern correlation as a distance. Both cases have a clear biological meaning: Euclidean distances are used when the interest is in looking for identical patterns, whereas correlation distances are used in the case of the trends of the patterns. In our experience correlation, that implies looking for clusters of profiles with similar trends, gathers gene expression profiles in biologically meaningful clusters. Euclidean distances are more affected by small variations

Table 1. Comparison of the properties of the different clustering methods for analysing gene expression patterns

	Classical hierarchical clustering	SOM	SOTA
Topology	Hierarchical tree	Hexagonal or rectangular	Hierarchical tree
Growing	Aggregative (from bottom to top)	Size fixed from the beginning	Divisive (from top to bottom)
Number of clusters	As many as items	Fixed from the beginning	Customisable
Statistical definition of cluster	No	No	Yes
Proportional clustering	Yes	No	Yes
Possibility of obtaining clusters at different hierarchical levels	No	No	Yes
Robustness against noise	No	Yes	Yes
Provide average values of the profiles in the cluster	No	Yes	Yes
Runtime	Quadratic	Linear	Linear

in the patterns and produce less interpretable clusters of sequences.

The approach presented here for analysing gene expression profile data combines the advantages of the different clustering methods. DNA array technologies are undergoing a very fast development and in a few years, DNA chips with hundreds of thousands of genes will be available. The management of that huge amount of information will require the application of new approaches like those presented here. We believe that SOTA provides a fast, robust and accurate framework for the study of relationships among large sets of gene expression patterns and can be very useful for analysing gene expression at genomic level in a near future.

ACKNOWLEDGEMENTS

J.H. was supported by contract GW 01/841/98. We are indebted to Dov Stekel for valuable suggestions on the statistic analysis of the data and to Francesco Falciani for helpful comments on the manuscript.

REFERENCES

Alon,U., Barkai,N., Notterman,D.A., Gish,K., Ybarra,S., Mack,D. and Levine,A.J. (1999) Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Natl Acad. Sci. USA*, **96**, 6745–6750.

Ben-Dor,A., Shamir,R. and Yakhini,Z. (1999) Clustering gene expression patterns. *J. Comput. Biol.*, **6**, 281–297.

Cho,R.J., Campbell,M.J., Winzeler,E.A., Steinmetz,E.A., Conway,A., Wodicka,L., Wolfsberg,T.J., Gabrielian,A.E., Landsman,D., Lockhart,D.J. and Davis,R.W. (1998) A genome-wide transcriptional analysis of the mitotic cell cycle. *Mol. Cell*, **2**, 65–73.

Debouck,C. and Goodfellow,P.N. (1999) DNA microarrays in drug discovery and development. *Nature Genet.*, **21**, 48–50.

DeRisi,J.L., Iyer,V.R. and Brown,P.O. (1997) Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, **278**, 680–686.

Dopazo,J. and Carazo,J.M. (1997) Phylogenetic reconstruction using a growing neural network that adopts the topology of a phylogenetic tree. *J. Mol. Evol.*, **44**, 226–233.

Efron,B. and Tibsirani,R. (1991) Statistical data analysis in the computer age. *Science*, **253**, 390–395.

Eisen,M.B., Spellman,P.T., Brown,P.O. and Botstein,D. (1998) Cluster analysis and display of genome-wide expression patterns. *Proc. Natl Acad. Sci. USA*, **95**, 14,863–14,868.

Fritzke,B. (1994) Growing cell structures—a self-organizing network for unsupervised and supervised learning. *Neural Networks*, **7**, 1141–1160.

Gray,N.S., Wodicka,L., Thunnissen,A.M., Norman,T.C., Kwon,S., Espinoza,F.H., Morgan,D.O., Barnes,G., LeClerc,S., Meijer,L., Kim,S.H., Lockhart,D.J. and Schultz,P.G. (1998) Exploiting chemical libraries, structure, and genomics in the search for kinase inhibitors. *Science*, **281**, 533–538.

Hartigan,J.A. (1975) *Clustering Algorithms*. Wiley, New York.

Herwig,R., Poutska,A.J., Müller,C., Bull,C., Lehrach,H. and O'Brien,J. (1999) Large-scale clustering of cDNA-fingerprinting data. *Genome Res.*, **9**, 1093–1105.

Heyer,L.J., Kruglyak,S. and Yooseph,S. (1999) Exploring expression data: identification and analysis of coexpressed genes. *Genome Res.*, **9**, 1106–1115.

Iyer,V.R., Eisen,M.B., Ross,D.T., Schuler,G., Moore,T., Lee,J.C.F., Trent,J.M., Staudt,L.M., Hudson,J., Boguski,M.S., Lashkari,D., Shalon,D., Botstein,D. and Brown,P.O. (1999) The trancriptional program in response of human fibroblasts to serum. *Science*, **283**, 83–87.

Kohonen,T. (1990) The self-organizing map. *Proc. IEEE*, **78**, 1464–1480.

Kohonen,T. (1997) *Self-organizing Maps*. Springer, Berlin.

Kozian,D.H. and Kirschbaum,B.J. (1999) Comparative gene-expression analysis. *TIBTECH*, **17**, 73–78.

Lockhart,D.J., Dong,H., Byrne,M.C., Follettie,M.T., Gallo,M.V., Chee,M.S., Mittmann,M., Wang,C., Kobayashi,M., Horton,H. and Brown,P.O. (1996) Expression monitoring by hybridization to high density oligonucleotide arrays. *Nat. Biotechnol.*, **14**, 1675–1680.

Marton,M.J., DeRisi,J.L., Bennet,H.A., Iyer,V.R., Meyer,M.R., Robert,C.J., Stoughton,R., Burchard,J., Slade,D., Dai,H., Basset,D.E., Hartwell,L.H., Brown,P.O. and Friend,S.H. (1998) Drug target validation and identification of secondary drug target effects using DNA microarrays. *Nat. Med.*, **4**, 1293–1301.

Perou,C.M., Jeffrey,S.S., van de Rijn,M., Rees,C.A., Eisen,M.B., Ross,D.T., Pergamenschikov,A., Williams,C.F., Zhu,S.X., Lee,J.C.F., Lashkari,D., Shalon,D., Brown,P.O. and Botstein,D. (1999) Distinctive gene expression patterns in human mammary epithelian cells and breast cancers. *Proc. Natl Acad. Sci. USA*, **96**, 9112–9217.

Schena,M., Shalon,D., Davis,R.W. and Brown,P.O. (1995) Quantitative monitoring of gene expression patterns with a DNA microarray. *Science*, **210**, 467–470.

Shalon,D., Smith,S.J. and Brown,P.O. (1996) A DNA microarray system for analysing complex DNA samples using two-color fluorescent probe hybridization. *Genome Res.*, **6**, 639–645.

Sneath,P.H.A. and Sokal,R.R. (1973) *Numerical Taxonomy*. Freeman, San Francisco.

Spellman,P.T., Sherlock,G., Zhang,M.Q., Iyer,V.R., Anders,K., Eisen,M.B., Brown,P.O., Botstein,D. and Futcher,B. (1998) Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell*, **9**, 3273–3297.

Tamayo,P., Slonim,D., Mesirov,J., Zhu,Q., Kitareewan,S., Dmitrovsky,E., Lander,E.S. and Golub,T.R. (1999) Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. *Proc. Natl Acad. Sci. USA*, **96**, 2907–2912.

Törönen,P., Kolehmainen,M., Wong,G. and Castrén,E. (1999) Analysis of gene expression data using self-organizing maps. *FEBS Lett.*, **451**, 142–146.

Wang,H.C., Dopazo,J. and Carazo,J.M. (1998a) Self-organizing tree growing network for classifying amino acids. *Bioinformatics*, **14**, 376–377.

Wang,H.C., Dopazo,J., de la Fraga,L.G., Zhu,Y.P. and Carazo,J.M. (1998b) Self-organizing tree-growing network for the classification of protein sequences. *Protein Sci.*, **7**, 2613–2622.

Wen,X., Fuhrman,S., Michaels,G.S., Carr,D.B., Smith,S., Barker,J.L. and Somogyi,R. (1998) Large-scale temporal gene expression mapping of central nervous system development. *Proc. Natl Acad. Sci. USA*, **95**, 334–339.

Wodicka,L., Dong,H., Mittmann,M., Ho,M.H. and Lockhart,D.J. (1997) Genome-wide expression monitoring in *Saccharomyces cerevisiae*. *Nat. Biotechnol.*, **15**, 1359–1366.