Linear Modeling of Genetic Networks from Experimental Data

E.P. van Someren[a], L.F.A. Wessels[a,b] and M.J.T. Reinders[a]

[a]Information and Communication Theory Group, [b]Control Laboratory
Faculty of Information Technology and Systems,
Delft University of Technology,
P.O. Box 5031, 2600 GA Delft, The Netherlands
E-mail: E.P.vanSomeren@its.tudelft.nl
Tel: +31-152786424 Fax: +31-152781843
Keywords: Genetic Networks, Quasi-Linear Model, Clustering

Abstract

In this paper, the regulatory interactions between genes are modeled by a linear genetic network that is estimated from gene expression data. The inference of such a genetic network is hampered by the dimensionality problem. This problem is inherent in all gene expression data since the number of genes by far exceeds the number of measured time points. Consequently, there are infinitely many solutions that fit the data set perfectly. In this paper, this problem is tackled by combining genes with similar expression profiles in a single prototypical 'gene'. Instead of modeling the genes individually, the relations between prototypical genes are modeled. In this way, genes that cannot be distinguished based on their expression profiles are grouped together and their common control action is modeled instead. This process reduces the number of signals and imposes a structure on the model that is supported by the fact that biological genetic networks are thought to be redundant and sparsely connected. In essence, the ambiguity in model solutions is represented explicitly by providing a generalized model that expresses the basic regulatory interactions between groups of similarly expressed genes. The modeling approach is illustrated on artificial as well as real data.

Introduction

The introduction of the micro-array technology has made it possible to measure the simultaneous expression of thousands of genes with a single experiment. Since then, much research has been done on how this amount of data can be employed to infer the functionality of genes. Such inference is currently mainly performed by means of clustering (Eisen *et al.* 1998; Wessels *et al.* 1999) and pattern recognition techniques (Brown *et al.* 1999). The type of information that is not considered in these approaches is how genes regulate each other. If such relationships are known, more light can be shed on the functionality of genes. This might indicate which genes are responsible for a certain disease or process and can (ultimately) aid in the design of drugs that can cure a disease with minimal side-effects.

Currently, several different types of models are studied, like Boolean networks (Liang, Fuhrman, & Somogyi 1998), Bayesian networks (Friedman, Goldszmidt, & Wyner 1999; Friedman *et al.* 1999), (Quasi)-Linear networks (D'Haeseleer *et al.* 1999), Neural networks (Weaver, Workman, & Stormo 1999) and Differential Equations (Chen, He, & Church 1999). Boolean and Bayesian networks need some way to discretize the continuous measurement values, a process that is sensitive to the quantization scheme and might introduce artifacts which will make these models less realistic. More biologically inspired models, like the Mjolsness Model (E. Mjolsness & Reinitz 1991), contain so many parameters that there are serious limitations on learning these parameters from current real data-sets. The major problem concerning currently available data-sets is that they generally consist of hundreds to thousands of genes whose activation levels are measured at no more than twenty time points. From an information-theoretic point of view this dimensionality problem will render any network model inferred from this data virtually meaningless. We therefore choose an approach that employs a network model containing as few parameters as necessary and aim additionally to extract as much information as possible from ambiguous data. Useful information can be extracted from the data by incorporating sensible constraints on the modeling process based on available biological information.

In this paper, a linear network is used as a basis to model the regulating interactions between genes and the model is learned from measurements of gene activity over consecutive time points. The basic linear model follows the assumption that the activity level[1] of a gene at a certain point in time can be determined by the weighted sum of the activity levels of all genes at the previous time-point[2].

$$x_j(t) = \sum_{i=1}^{N} r_{i,j} \cdot x_i(t-1) \qquad x_j, r_{i,j} \in I\!R \qquad (1)$$

[1]Throughout the paper we define the activity level of a gene as the logarithm of the ratio between normal and sample mRNA level.

[2]In other words, relationships between genes are assumed to be stationary.

, where $x_j(t)$ represents the activity level of gene j at time t, $r_{i,j}$ represents how strongly gene i controls gene j and N is the total number of genes under consideration. Note that the linear model captures negative and positive regulatory interactions between genes, but processes such as mRNA degradation are not explicitly modeled. A more complete modeling approach is covered in the next section, were the linear model is augmented with appropriate pre-and post-processing steps. Our main contribution lies in the way the dimensionality problem is tackled by employing hierarchical clustering to combine genes with similar expression profiles. Genes with similar profiles introduce ambiguity when the linear model is learned, because they cannot be distinguished in terms of the way they control (are being controlled by) other genes. This approach is also biologically plausible because redundancy in genes implies input and output sharing among genes involved within a gene family or pathway (D'Haeseleer, Liang, & Somogyi 2000). This constraint is implemented by forcing genes in the same cluster to share one set of weights. Furthermore, the fact that genes are estimated on average to interact with four to eight other genes (Arnone & Davidson 1997) warrants the restriction of possible inputs that is caused by the clustering process.

The Modeling Approach

Our modeling approach is illustrated in Fig. 1 and consist of three major processes, i.e. a combination of pre- and post-processing, clustering and the linear model. Pre-processing is used to convert the raw measurements $\mathbf{X}$ into a set of useful signals $\mathbf{X}'$ that reflect the important properties of the original data. Signals that are not significantly up- or down-regulated are removed and the remaining signals are normalized. The normalization step actually reflects which signals are thought to be similar and what control actions should be linearly modeled. To tackle the dimensionality problem, the number of signals must be reduced in a way that does not degrade the validity of the resulting model. The clustering is employed to determine groups of similar signals and to compute a prototype signal for each of the clusters (using $\mathbf{Q}$). These prototypes $\mathbf{Y}'$ form a representation of the basic signal shapes among all gene-profiles. These prototypes are used to learn the reduced linear model $\mathbf{R}$ between the prototypes. Such a constrained model removes any ambiguity introduced by the data and represents the basic interactions of the underlying genetic network. After the linear model is learned it can predict the signals of the prototypes $\hat{\mathbf{Y}}'$ given the value of the prototypes at the first time-point. From the estimated prototype signals an estimation of the normalized signals $\hat{\mathbf{X}}'$ can be determined by means of an 'inverse' clustering step $\mathbf{W}$. Similarly, an estimate of the original signals $\hat{\mathbf{X}}$ can be determined by employing an inverse normalization step. A more detailed description of each part of the approach is given in the subsections below.

Thresholding

Experimental observations (D'Haeseleer, Liang, & Somogyi 2000) have shown that the ratio of gene-expression of the same genes taken from different cultures under similar conditions can vary up to an absolute ratio of two, however when one culture is differentiated by a change in condition the gene-expression ratios of a substantial part of the genes exceed a two-fold and (some) even a five-fold ratio. Such observations indicate that genes with profiles that remain below an absolute value of two are not significantly expressed or repressed and therefore do not participate in the regulation. The removal of such insignificant signals will not only help to reduce the dimensionality problem, but also avoid erroneous relationships when learning the linear model. A signal with a small amplitude contains a large portion of measurement error that introduces local distortions, such as additional peaks. Nevertheless, the erroneous shape of that signal might be such that when multiplied by a large weight, it closely predicts one of the other significant signals. In such cases the linear model will erroneously infer a strong relation between a largely random signal and one of the significant signals. Therefore, in our approach all genes whose activity level over *all* time instances is below a value of two will be removed.

Normalization

In our approach, normalization is an important step as it actually determines which characteristics of the original signals are thought to express dissimilarity as well as the regulatory action between genes. When two signals share a characteristic considered to be important, these two signals should be very similar after normalization. For example, if the shape of the signals are thought to be important this can be emphasized by normalizing the signals with respect to their mean and variance. After such normalization the Euclidean distance measure will express the same similarity as the Pearson Correlation Measure expresses when applied to the original signals. Similarly, a linear model relating normalized signals is equivalent to a quasi-linear model on the original signals. To simplify these model choices we only alter the type of normalization and keep the Euclidean distance measure and the basic linear model fixed. A visualization of the normalized signals indicates what the user exactly determines as being similar c.q. dissimilar, and also depicts the actual control action of each gene that is linearly modeled. The types of normalization can be written as a combination of an additive term c_1 and a multiplicative term c_2. The relation between original signal $\mathbf{x}$ and normalized signal $\mathbf{x}'$ is represented by the following equation:

$$\mathbf{x}' = \frac{\mathbf{x} - c_1}{c_2} \tag{2}$$

The original signals can be reconstructed from the normalized signals by means of a post-processing step that

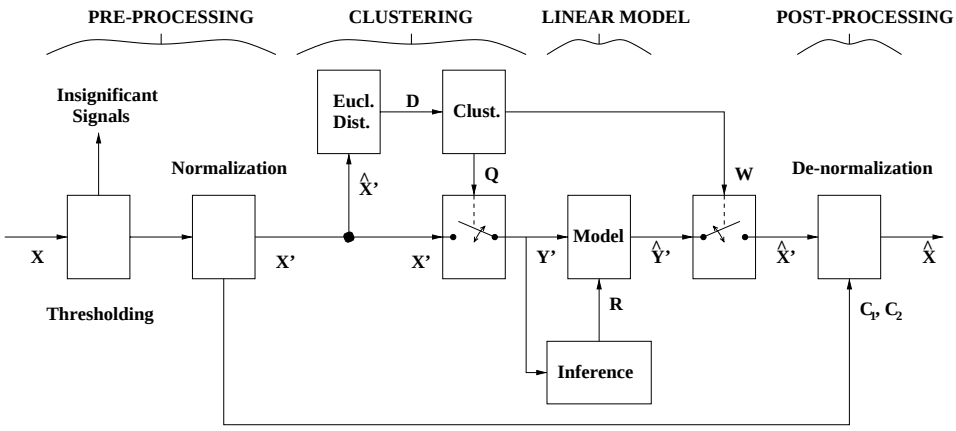

Figure 1: The modeling approach.

Type	c_1	c_2	$\mathbf{x}'$
None	0	1	$\mathbf{x}' = \mathbf{x}$
Mean	$\frac{1}{T}\sum_{t=1}^{T} x(t)$	1	$\mathbf{x}' = \mathbf{x} - c_1$
Variance	0	$\sum_{t=1}^{T} x(t)^2$	$\mathbf{x}' = \frac{\mathbf{x}}{c_2}$
Mean and Variance	$\frac{1}{T}\sum_{t=1}^{T} x(t)$	$\frac{1}{T-1}\sum_{t=1}^{T} (x(t) - c_1)^2$	$\mathbf{x}' = \frac{\mathbf{x}-c_1}{c_2}$

Table 1: Types of normalization

performs the inverse operation of Eq. (2). We distinguish between the types of normalization as depicted in table 1.

The Linear Model

The linear model as defined in Eq. (1) serves as a representation of the regulatory interaction between genes. If the weights, being the parameters of the linear model, and the activity levels of all genes at a certain time-point are known, the activity levels of all genes at later time points can be predicted. However, the weights are not known and must be inferred from a set of measurements of gene-activities at consecutive time points. Such measurements can be represented in a so called gene expression matrix,

$$\mathbf{X} = [\ x_{i,t} \mid i \in 1, \ldots, N \quad t \in 1, \ldots, T\ ] \qquad (3)$$

, where each row, denoted by $\mathbf{x}_i$, represents the gene-profile of gene i taken over T time points. The t-th column of $\mathbf{X}$ is denoted by $\mathbf{x}(t) = [\ x_1(t),\ \ldots,\ x_N(t)\ ]^T$ and determines the state of the system at time t. For ease of notation we represent the linear model using matrix and vector notation rather than by means of Eq. (1).

$$\mathbf{x}(t+1)^T = \mathbf{x}(t)^T \cdot \mathbf{R} \qquad \forall t = 1,\ \ldots,\ T-1 \qquad (4)$$

The first goal is to find all weight-matrices $\mathbf{R}$ that are consistent with our data and thus with Eq. (4), i.e. using $\mathbf{R}$ and a given state will exactly determine the next state. In general, the weight-matrix will be under-constrained which means that there exist multiple solutions which can be written as a combination of a particular solution $\mathbf{P}$, a basis of homogeneous solutions $\mathbf{H}$ and a set of free variables $\mathbf{F}$, i.e.

$$\mathbf{R} = \mathbf{P} + \mathbf{H} \cdot \mathbf{F} \qquad (5)$$

The particular solution $\mathbf{P}$ reflects the information from the data, i.e. it is *one* solution that satisfies Eq. (4) (Note, many other solutions exist).

$$\mathbf{x}(t+1)^T = \mathbf{x}(t)^T \cdot \mathbf{P} \qquad \forall t = 1,\ \ldots,\ T-1 \qquad (6)$$

The homogeneous solution $\mathbf{H} \cdot \mathbf{F}$ reflects the resulting ambiguity in the data, i.e. it is that part of the weight-matrix that reflects the possible changes that do *not influence* the estimation of the signals in the data, i.e.

$$\mathbf{x}(t)^T \cdot \mathbf{H} \cdot \mathbf{F} = \mathbf{0} \qquad \forall t = 1,\ \ldots,\ T-1 \qquad (7)$$

Each column of the basis of homogeneous solutions $\mathbf{H}$ determines how a particular change in one weight must be compensated by the other weights in the same column[3]. The more columns $\mathbf{H}$ contains the more independent directions of change are allowed. The set of free variables $\mathbf{F}$ reflects the degrees of freedom as each element can be substituted with any particular value without changing the estimation of the given data. For a given data-set the particular and homogeneous solution can be found by Gaussian Elimination.

For actual biologically measured gene-expression matrices, the amount of ambiguity, i.e. the number of

[3] If a weight from gene A to gene B is altered, the relations of the other genes to gene B must compensate for the change in order keep the prediction of gene B the same.

columns in $\mathbf{H}$ is large as the number of measurements is significantly less than the number of genes. Although the ambiguity is exactly known it seems almost impossible to represent it in a interpretable way. However, when two genes react similarly one can group these genes together and in this way decrease the ambiguity in the system. When two signals, $\mathbf{x}_i$ and $\mathbf{x}_j$, are very similar and both signals can predict one of the other signals $\mathbf{x}_k$ accurately then removing one of the similar signals still gives a good prediction[4]. In fact, a good prediction is influenced only by the *sum of the weights* that correspond to the input of both signals, i.e. $r_{i,k} + r_{j,k}$. This introduces an ambiguity in the set of possible weight-matrices that can be made explicit by replacing each group of similar signals with a prototype signal and learning the weights between the prototypes instead of between the original signals. A weight that is assigned to a prototype means that one of the signals corresponding to that prototype should be assigned that weight, but given the data no choice can be made. This gives a motivation to perform clustering in order to tackle the dimensionality problem without degrading the interpretation of the resulting simplified rule-base and with minimal loss of the estimation performance.

Clustering

The two main functions of clustering are to find groups (clusters) of signals based on similarity and to conceptualize the data by representing each cluster with a proper prototype. The most important aspect of clustering is the choice of distance measure. It basically expresses the way similarity between signals depends on the values of the signals. Because the similarity of the signals is part of the normalization step it is not longer necessary to consider multiple distance measures and we restrict ourselves to the Euclidean distance measure. An important step which follows clustering is to represent each cluster with a proper prototype. In our view, the choice of distance measure completely determines how a proper prototype must be computed. A prototype is a signal that is the most similar to all signals it represents. In terms of the distance measure, the representative will be that signal $\mathbf{y}$ which has the minimal root mean square (RMS) distance to all signals in the cluster. For the Euclidean distance, this corresponds to taking the mean of all signals in a cluster. The variance of all signals in a cluster represents the error that is made by replacing the signals with their prototype, also denoted by the within scatter of the clustering. Note that the clustering process reduces the noise in the signals due to the averaging of signals within the same cluster.

Complete linkage hierarchical clustering based on the Euclidean distance is performed. The number of clusters is decreased until the system becomes over-constrained (i.e. the homogeneous solution becomes empty). This clustering can be represented by means of matrix multiplication. Transforming signals to prototypes is performed by multiplying the normalized signals with matrix $\mathbf{Q}$.

$$\mathbf{Y}' = \mathbf{X}'^T \cdot \mathbf{Q} \tag{8}$$

, where $\mathbf{Q}$ is a $(N \times P)$ matrix where each row corresponds to each signal and each column corresponds to each prototype. Each element $q_{n,p}$ of the matrix represents the contribution of the n-th signal to the p-th prototype, with a mean prototype this becomes:

$$q_{n,p} = \begin{cases} 1/|C_p| & \hat{\mathbf{x}}_n \in C_p \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

C_p is the set of all signals in cluster p. The inverse operation of transforming prototypes into signals is performed by multiplying the estimated prototype signals with matrix $\mathbf{W}$.

$$\hat{\mathbf{X}}'^T = \hat{\mathbf{Y}}' \cdot \mathbf{W} \tag{10}$$

, where $\mathbf{W}$ is a $(P \times N)$ matrix where the rows correspond to the prototypes and the columns correspond to the signals. Each element $w_{p,n}$ of the matrix represents how the n-th signal can be reconstructed from the p-th prototype. With a mean prototype the prototype itself can directly represent the signal:

$$w_{p,n} = \begin{cases} 1 & \mathbf{x}_n \in C_p \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

To validate the modeling approach two experiments were performed. First, an artificial problem with a known and structured weight-matrix is used to test the capability of our method to uncover the right structure and relations from generated signals. The second experiment was done on a real yeast data-set on which several analysis were performed.

Example: Artificial Problem

In this section, we discuss the experimental results that were obtained on a simple artificial example, which serves to illustrate the principles on which the modeling approach is based. This example mimics the structure present in large scale genetic networks: groups of genes being co-regulated and thus exhibiting similar time responses.

As a first step, a linear system consisting of five genes is constructed by choosing a particular (5×5) matrix $\mathbf{R}^5 = \{ r^5_{i,j} \}$. The superscript indicates the size. In analogy with Eq. (4), the behavior of this system is described by the following equation:

$$\mathbf{x}(t+1)^T = \mathbf{x}(t)^T \cdot \mathbf{R}^5 \tag{12}$$

Starting from any initial state this system will settle into a stable steady state after a finite time interval. The specifically chosen weight-matrix is graphically depicted in Figure 2. From this small network, a network

[4] The remaining signal can compensate for the removed signal, because of its similar shape

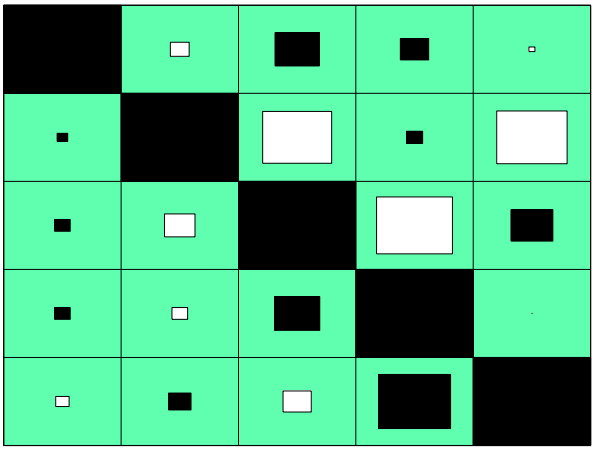

Figure 2: Graphical representation of $\mathbf{R}^5$ used for the artificial example. A black square at position (i, j) represents a positive control action of gene i on gene j, i.e. $r_{i,j} > 0$, while a negative control action is represented by a white square. The size of a square is proportional to the absolute size of the weight value.

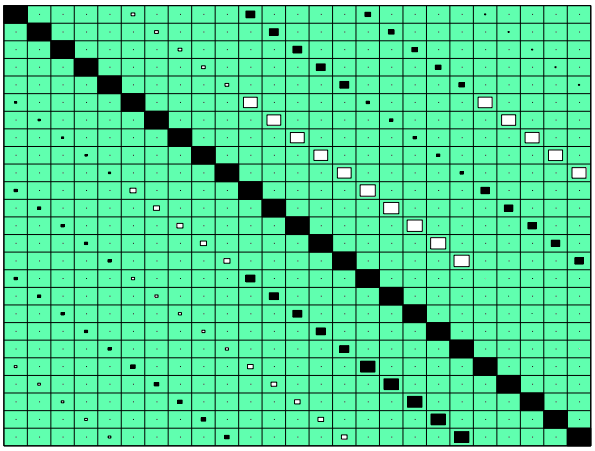

Figure 3: Graphical representation of $\mathbf{R}^{25}$: Expanded from $\mathbf{R}^5$ for the artificial example.

of twenty-five genes was constructed by replicating the simple system five times. This was achieved by constructing $\mathbf{R}^{25}$ in the following fashion: the (i, j)-th 5×5 sub-matrix in $\mathbf{R}^{25}$ was constructed by placing $r^5_{i,j}$ on the diagonal with all other positions in the sub-matrix occupied by zeros. Figure 3 contains a graphical representation of $\mathbf{R}^{25}$. The initial state of the genes is determined such that each signal has an initial state that is more similar to the initial states of the genes in its group than to the initial states of the genes in the other groups. Figure 4 depicts the first 20 time points of the gene activity levels after such an initialization and calculating subsequent time points using weight-matrix $\mathbf{R}^{25}$. We will denote this gene-expression matrix by $\mathbf{X}$. The 'grouping' of the signals within a subsystem is quite apparent, and simulates groups of co-regulated genes in a biological genetic network. If purely judged on the number of signals ($N = 25$) and the number of time points in the gene-expression matrix ($T = 20$) the problem of estimating the values of $\mathbf{R}^{25}$ based on $\mathbf{X}$ is under-constrained, i.e. infinitely many solutions exist. We estimated the model based on the methodology described in the previous sections, but without pre- and post-processing. More specifically, we employed complete linkage hierarchical clustering and the Euclidean distance measure to build a complete dendogram of the signals. For each possible clustering, C^k, ranging from a single cluster ($k = 1$, all signals in one cluster) to 25 clusters ($k = 25$, a single signal per cluster) the following steps were performed:

1. The set of prototypes, $\mathbf{Y}^k$ associated with the clusters in C^k was determined by averaging the signals in each cluster.
2. The weight matrix, $\hat{\mathbf{R}}^k$, corresponding to each clus-

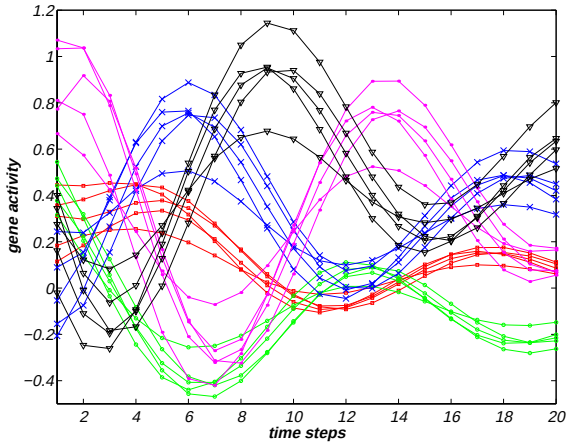

Figure 4: Time response for the 25 x 25 system of the artificial problem.

tering was determined from $\mathbf{Y}^k$. In the under-constrained case, the particular solution was used;

3. Given the complete model $C^k; \mathbf{Y}^k; \hat{\mathbf{R}}^k$, and an initial state, $\mathbf{x}(0)$, approximations to the original signals can be computed as: $\hat{\mathbf{x}}^k(t+1) = f(C^k, \mathbf{Y}^k, \hat{\mathbf{R}}^k, \mathbf{x}(0))$. The following types of approximations were computed:

   - *one-step* approximations, i.e. each prediction is based on the true state of the previous time instance.

$$\hat{\mathbf{x}}^{os,k}(t+1) = f(C^k, \mathbf{Y}^k, \hat{\mathbf{R}}^k, \mathbf{x}(t)), \quad t = 1, 2, \ldots, T-1 \quad (13)$$

   - *free-run* approximations, i.e. predictions are based only on the initial state

$$\hat{\mathbf{x}}^{fr,k}(t+1) = f(C^k, \mathbf{Y}^k, \hat{\mathbf{R}}^k, \hat{\mathbf{x}}^{fr,k}(t)), \quad \hat{\mathbf{x}}^{fr,k}(0) = \mathbf{x}(0), \quad t = 1, 2, \ldots, T-1 \quad (14)$$

4. The mean squared error (MSE) value associated with both the one step and free run approximation of the original signals were computed and are denoted by $E^{os,k}$ and $E^{fr,k}$ respectively.

5. The weighted prototype MSE, $E^{wp,k}$, was also computed; this error represents the free run prototype prediction error, i.e. predicting the values of the prototypes at the next time step in terms of the values estimated on the preceding time step. The error of each prototype was weighted by the number of signals in its cluster.

$$E^{wp,k} = \frac{E^{p,k} \cdot |C_k|}{N} \quad (15)$$

$$E^{p,k} = (\frac{1}{T-1}) \sum_{t=1}^{T} \|\hat{\mathbf{y}}^{fr,k}(t) - \mathbf{x}(t)\|^2 \quad (16)$$

$$\hat{\mathbf{y}}^{fr,k}(t+1) = g(\hat{R}^k, \hat{\mathbf{y}}^{fr,k}(t)), \quad \hat{\mathbf{y}}^{fr,k}(0) = h(C^k, \mathbf{x}(0)), \quad t = 1, 2, \ldots, T-1 \quad (17)$$

Figure 5 depicts $E^{fr,k}$, $E^{os,k}$ and $E^{wp,k}$ as a function of k, the number of clusters. Apart from the expected trend which shows an increase in *signal* approximation error as the number of clusters decreases, the prototype MSE, $E^{wp,k}$, remains zero as long as $k \geq 5$ clusters. For fewer clusters the homogeneous solution is empty as the system can no longer exactly predict the prototype signals. Consequently, only a single solution exists that minimizes the MSE prediction error, i.e. the system is over-constrained. Note that when judging the system based purely on the number of time steps and number of signals in the data set, we expected a non-zero $E^{wp,k}$ for $k \geq 19$ clusters, i.e. equal to one less than the number of time points. However, due to the repetitive structure in the system (five copies of the same system), and the fact that the clustering enables the solution process to exploit this structure, only five

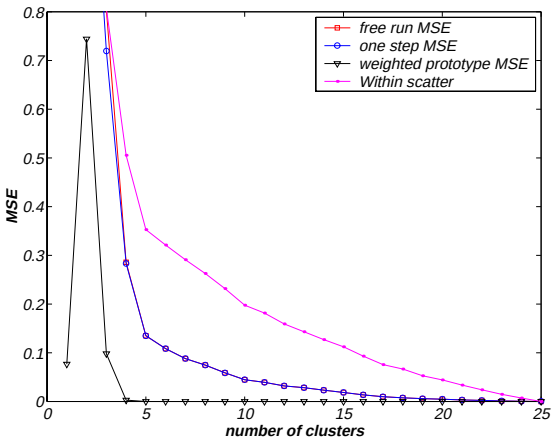

Figure 5: Error curves as a function of the number of clusters employed in the model for the signals generated from $\mathbf{R}^{25}$ in the artificial example.

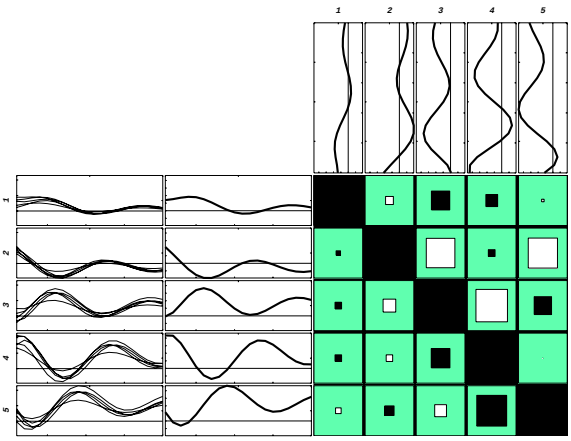

Figure 6: Graphical representation of the model for 5 clusters. The column of plots on the far left hand side depicts the original time responses of the 25 x 25 system, grouped in five clusters. The next column of plots depicts the associated prototypes. The 5 x 5 square represents the estimated weight matrix $\hat{\mathbf{R}}^5$, while the prototypes are repeated at the top. This representation enables us to read off the relation between a given prototype, say above column j of $\hat{\mathbf{R}}^5$ and all the other prototypes to the left of $\hat{\mathbf{R}}^5$, by examining the values in column j of $\hat{\mathbf{R}}^5$.

prototypes are required to exactly predict the signals. The resulting model is depicted in Figure 6. It is quite clear from this figure that the resulting estimate $\hat{\mathbf{R}}^5$ is a good approximation of $\mathbf{R}^5$. However, at the same time we should point out that interpretation of the $\mathbf{R}$ matrix can, in some cases, be fairly difficult. For example, consider prototype 5 directly above column 5 of the $\hat{\mathbf{R}}^5$ matrix in Figure 6. The large value of $r_{5,5}$ implies that it has a large positive relation with itself, which is understandable, However, large values for $r_{2,5}$ and $r_{3,5}$ imply that these prototypes are also involved in producing the time response of prototype 5. Such subtle interactions are fairly difficult to comprehend by looking at the signals. In any event, this artificial example illustrates that the approach holds promise, since it enabled us to extract the essential (original) structure of a system consisting of co-regulated genes.

Experiment: Yeast data-set

The analyzes described in this section were carried out on the gene expression profiles extracted from the 2467 genes in the budding yeast *S. cerevisiae* (Eisen *et al.* 1998). The dataset consists of several sub-sets collected under different conditions: mitotic cell division cycle, sporulation and temperature and reducing shocks. Since the approach described in this paper models the *time behavior* of the genes, appropriate external inputs should be included to model changing external conditions, if the concatenation of the sub-sets were to be employed as input data. Since such inputs are not included in our model yet, we can only apply the approach on a single sub-set at a time.

Thresholding:

In a previous section it was motivated why genes with expression profiles which never exceed ± 2 should be considered as 'insignificant' and why such genes should be removed from the data set. Removal of all such expression profiles reduces the sizes of all the data sets dramatically and consequently alleviates the dimensionality problem. For the ALPHA subset which consists of 18 time points, 45 genes remain as significant signals, for the CDC15 subset consisting of 14 time points 113 genes remain as significant signals.

Normalization:

In order to investigate the effects of the different kinds of normalization, the same experimental procedure as in the artificial experiment was employed for each of the Eisen sub-sets. For each subset the prototype free run error remained zero as long as the number of clusters were equal to or higher than one less than the number of time points ($k = T - 1$). That particular number of clusters $k = T - 1$, corresponds to the solution involving the largest number of clusters for which the problem is over-constrained, i.e. the homogeneous matrix is empty. Figure 7 depicts the one-step MSE ($E^{os,T-1}$) at this point as a function of all the different data sets and

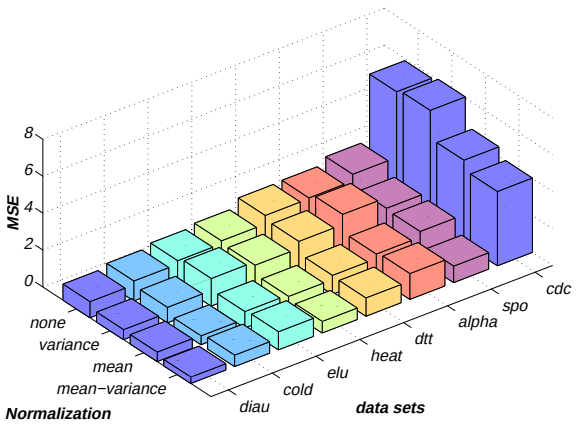

Figure 7: MSE for all data sets and the four kinds of normalization. The MSE is computed for the over-constrained model with the smallest MSE and always corresponded to that model where the number of clusters equalled the number of state-transitions in the data set.

kinds of normalization[5]. From Figure 7 it is quite apparent that 'mean-variance' normalization results in the smallest MSE for all data sets except for ALPHA, ELU and COLD, where the mean normalization is slightly better. The CDC15 and ALPHA subsets were chosen for further experimentation since they contain a relatively large number of time points (15 and 18 resp.).

Fitting the linear model on the CDC15 subset:

The same experimental procedure outlined in the previous section was employed for this data set, resulting in the error curves depicted in Figure 8. From this figure it is clear that E^{wp} becomes zero at 14 clusters, indicating that in order to capture the intrinsic dimensionality a model with at least 14 and probably more prototypes is required. Note that given only 15 time points it is the best we can do. In order to determine the exact intrinsic dimensionality more time points are required. The resulting model is depicted in Figure 9. The $\mathbf{R}$ matrix is, as stated earlier, not always easily interpretable due to the many signals that contribute to the prediction of a single signal. This phenomenon corresponds to a column with multiple large values, such as columns 2, 9 and 14. For example, the ninth prototype is constructed by employing significant contributions from at least six other prototypes. However, columns with few large values correspond to genes that are influenced by only a small number of prototypes. For example, prototype 3 is primarily influenced by prototype 5. This makes sense as prototype 3 matches a one-step delayed version of prototype 5. Similarly, a row with many (few) large values indicates a prototype that regulates many

[5]The free-run error is exactly the same as the one-step error as each intermediate state is exactly estimated.

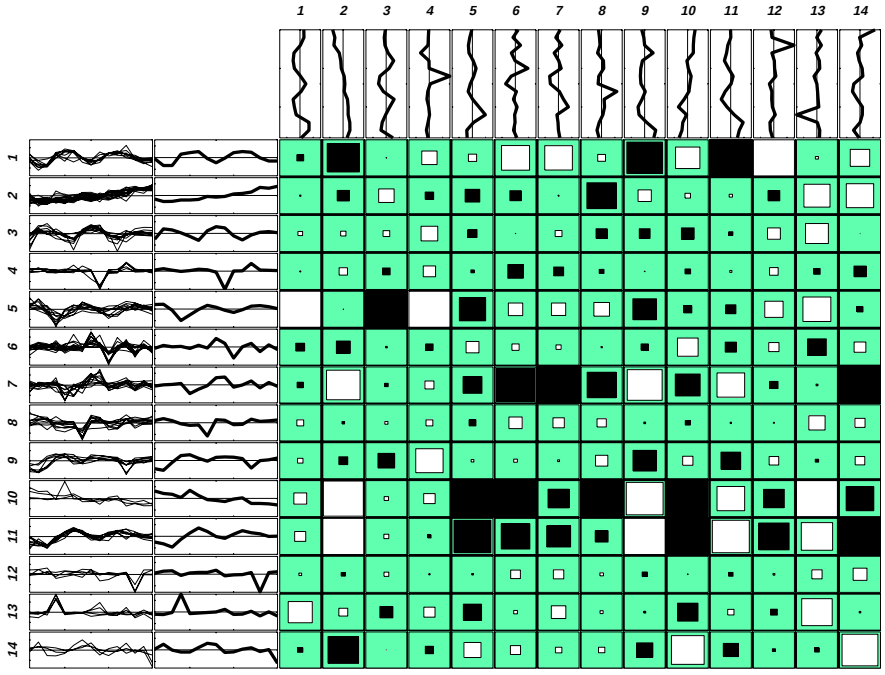

Figure 9: Graphical representation of the model for 14 clusters on data set CDC15 using mean variance normalization. The column of plots on the far left hand side depicts the normalized time of the 113 genes that remain in the data set after thresholding, grouped in 14 clusters. The next column of plots depicts the associated prototypes. The 14 x 14 square represents the associated R matrix. For visualization purposes each column is normalized with respect to the maximum value of that column.

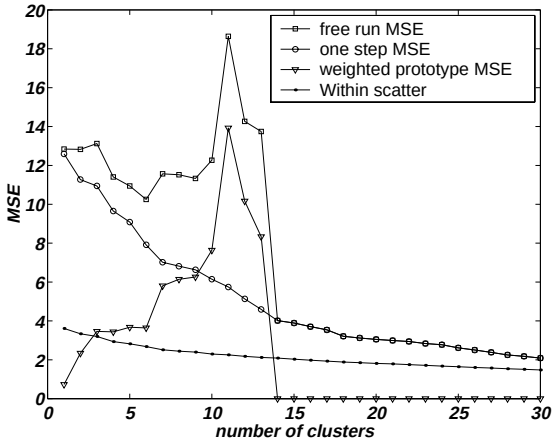

Figure 8: Error curves as a function of the number of clusters employed in the model for the CDC15 dataset.

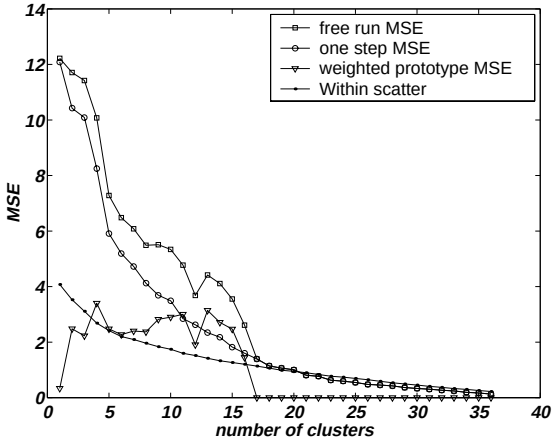

Figure 10: Error curves as a function of the number of clusters employed in the model for the ALPHA data-set.

(few) other genes. Examples of such prototypes are prototypes 7, 10 and 11 which regulate many genes, while prototype 12 has hardly any influence on other genes.

Fitting the linear model on the Alpha sub-set:

The same procedure was also applied to the Alpha-subset, which is the subset containing the largest number of time points. The resulting error curves are shown in Figure 10. Once again the prototype error remains zero when the number of clusters exceeds the number of time points. The resulting model for 17 clusters is depicted in Figure 11 when using mean variance normalization. A striking characteristic of the **R** matrix is the fact that prototypes 1, 7 and 17 strongly influence other genes. Moreover, prototypes 7 and 17 have almost exactly the same influence on all other prototypes. This is probably due to the fact that a summation of these two prototypes results in a more or less constant signal which plays a role similar to a bias term. Another characteristic of this model is the 'peakyness' of the prototypes. For example, prototypes 5, 7, 15, 16 and 17. This stems from the fact that mean variance normalization was employed, which emphasizes small peaks in the original signals. To illustrate the effect of normalization, we also obtained a model without normalization, which is depicted in Fig. 12. From this figure, we observe that there are fewer prototypes that contain a single peak. In addition, the similarity between rows, which was observed in Fig. 11, has disappeared and is more sparse than the matrix obtained with mean variance normalization. Table 2 lists the function names associated with the genes in the clusters. There are two dominant clusters, namely, Cluster 2 which consists mainly of genes involved in MATING and Cluster 7 which consists entirely of CHROMATIN STRUCTURE related genes. There is one prototype, namely Prototype 2 (MATING), which has a strong influence on many other prototypes. There are also several columns which contain a single large value indicating that those genes are influenced primarily by a single prototype. For example, Prototype 14 (MATING) is strongly influenced by Prototype 2 (MATING), which can be easily understood (purely based on signal shapes) since the down regulation of Prototype 14 is preceded by down regulation of Prototype 2. Another interesting example involves Prototypes 7 (CHROMATIN STRUCTURE) and 13, where a single large value in column 7 indicates that Prototype 7 can be well predicted by the behaviour of Prototype 13 (either PROTEIN GLYCOSYLATION or VANADATE RESISTANCE). This follows from the fact that Prototype 7 is a time delayed version of Prototype 13.

When interpreting such results, human beings typically focus on pair-wise comparisons of in- and outputs. When more than two control actions are responsible for a signal's response, the results are often hard to validate by comparing pairs of signals involved. One should bear in mind that a linear model is not primarily intended to be interpreted in such a way. It should also be borne in mind that starting from 2467 genes now only 14 or 17 basic regulatory actions remain! Unfortunately, in none of the sub-sets the number of measured time points is enough to discover the exact intrinsic dimensionality. This is indicated by the fact that the prototype error was only zero when the number of clusters exceeded one less than the number of time points. If the intrinsic dimensionality were lower, the prototype error would have been zero for a smaller number of clusters, as illustrated in the artificial problem.

Summary

In this paper we presented a new methodology for modeling genetic networks that employs clustering to tackle the dimensionality problem and a linear model to represent the relationships between the resulting prototypes.

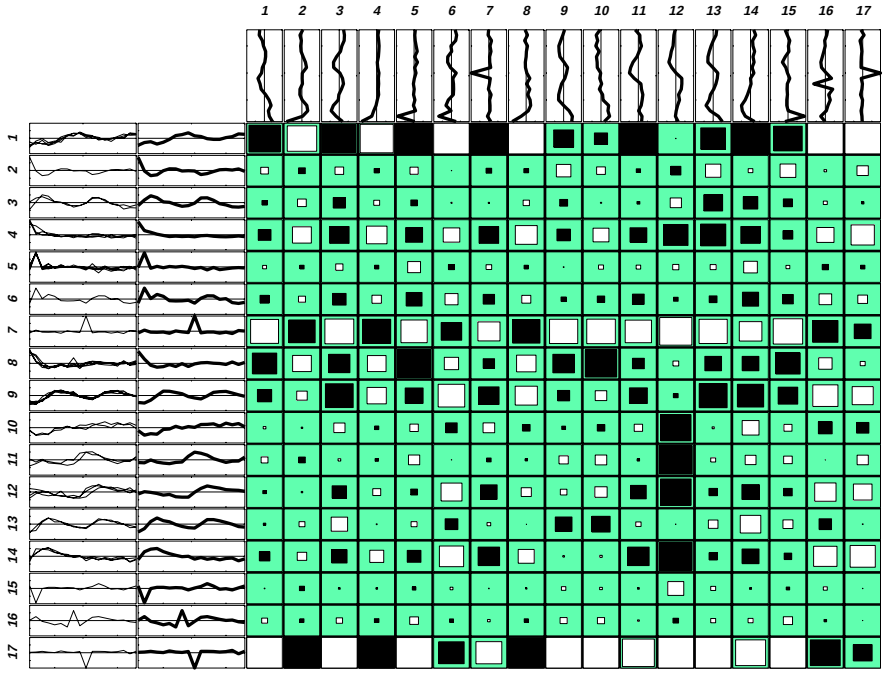

Figure 11: Graphical representation of the model for 17 clusters on data set ALPHA using mean variance normalization. The column of plots on the far left hand side depicts the normalized time of the 45 genes that remain in the data set after thresholding, grouped in 17 clusters. The next column of plots depicts the associated prototypes. The 17 x 17 square represents the associated R matrix. For visualization purposes each column is normalized with respect to the maximum value of that column.

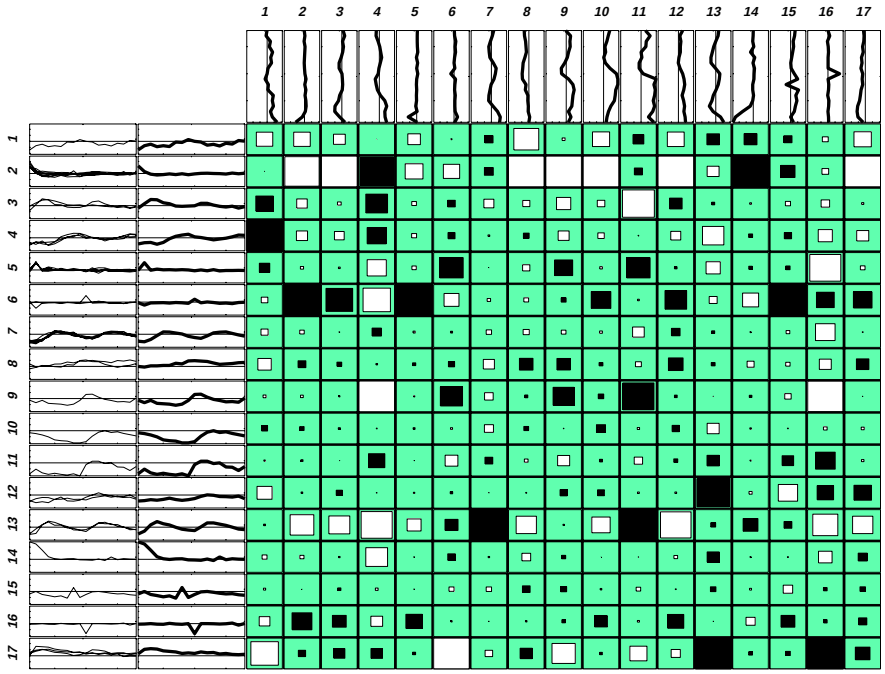

Figure 12: Graphical representation of the model for 17 clusters on data set ALPHA using no normalization. The column of plots on the far left hand side depicts the normalized time of the 45 genes that remain in the data set after thresholding, grouped in 17 clusters. The next column of plots depicts the associated prototypes. The 17 x 17 square represents the associated R matrix. For visualization purposes each column is normalized with respect to the maximum value of that column.

In order to alleviate the dimensionality problem the number of signals must be dramatically reduced, such that the resulting generalized model is a valid representation of the basic regulatory interactions. This can not be done in an ad hoc fashion. However, clustering can be employed to perform such a reduction in a biologically sound fashion. This stems from the fact that clustering combines similar signals in a way that models redundancy and imposes limited connectivity; characteristics which are believed to be present in biological genetic networks. Moreover, similar signals will introduce ambiguity in the set of possible solutions because their regulating behaviour cannot be distinguished based on the data. The validity of our approach is illustrated by applying it on an artificial problem were it is shown that the method recovers the structured underlying network from an under-constrained set of signals. Consequently, some preliminary experiments were performed on the Eisen data-set. The results on two sub-sets, the ALPHA and CDC15 subsets, were presented in this paper. The validity of any interpretation based on the results obtained with the linear model is obviously conditioned on the assumption that the linear model faithfully captures the behaviour of the underlying network. While more complex models, such as differential equations may be more biologically plausible, the estimates of the parameters are highly unreliable. The approach outlined in the paper, which combines clustering with linear modeling, addresses the dimensionality problem directly, thus rendering the estimated parameters more reliable. In our opinion, this approach strikes a good balance between model complexity and parameter validity.

Acknowledgements

This work was funded by the Intelligent Molecular Diagnostic System program of the Delft Inter-Facultary Research Center at the Technical University of Delft.

References

Arnone, A., and Davidson, B. 1997. The hardwiring of development: Organization and function of genomic regulatory systems. *Development*.

Brown, M.; Grundy, W.; Lin, D.; Cristianini, N.; Sugnet, C.; Ares, M.; and Haussler, D. 1999. Support vector machine classification of microarray gene expression data. *Technical report*.

Chen, T.; He, H.; and Church, G. 1999. Modeling gene expression with differential equations. *Pacific Symposium on Biocomputing '99* 4:29–40.

D'Haeseleer, P.; Wen, X.; Fuhrman, S.; and Somogyi, R. 1999. Linear modeling of mrna expression levels during cns development and injury. *Pacific Symposium on Biocomputing '99* 4:41–52.

Cluster	Function Name
1	SECRETION, NON-CLASSICAL
2	CYTOSKELETON
	MATING; CELL FUSION
	MATING
	KARYOGAMY
	MATING
	MATING
	MATING
	MATING; CELL FUSION
3	CELL CYCLE
	TCA CYCLE
4	CELL CYCLE
	CELL WALL BIOGENESIS
	CELL CYCLE
5	CU2+ ION HOMEOSTASIS
	DNA REPLICATION
	CELL WALL BIOGENESIS
	RRNA PROCESSING
	CELL CYCLE
	MRNA SPLICING
6	CELL CYCLE
	SECRETION
7	CHROMATIN STRUCTURE
	CHROMATIN STRUCTURE
	CHROMATIN STRUCTURE
	CHROMATIN STRUCTURE
	CHROMATIN STRUCTURE
	CHROMATIN STRUCTURE
	CHROMATIN STRUCTURE
	CHROMATIN STRUCTURE
8	TRANSPORT
	PHOSPHATE METABOLISM
9	MATING TYPE SWITCHING
10	CELL WALL BIOGENESIS
11	CELL CYCLE
12	AGING
	BUD SITE SELECTION, BIPO
13	PROTEIN GLYCOSYLATION
	VANADATE RESISTANCE
14	MATING
15	MEIOSIS
16	PROTEIN SYNTHESIS
17	TRANSPORT
	SECRETION, NON-CLASSICAL
	CELL CYCLE

Table 2: Function names of signals in clusters

D'Haeseleer, P.; Liang, S.; and Somogyi, R. 2000. Genetic network inference: From co-expression clustering to reverse engineering. *Submitted to Bioinformatics.*

E. Mjolsness, D. S., and Reinitz, J. 1991. A connectionist model of development. *Journal of Theoretical Biology* 152.

Eisen, M.; Spellman, P.; Brown, P.; and Bottstein, D. 1998. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Acadamy of Sciences of the USA* 95(25):14863–14868.

Friedman, N.; Linial, M.; Nachman, I.; and Pe'er, D. 1999. Using bayesian networks to analyze expression data. *Submitted.*

Friedman, N.; Goldszmidt, M.; and Wyner, A. 1999. Data analysis with baysian networks: A bootstrap approach. *Proc. Fifteenth Conf. on Uncertainty in Artificial Intelligence (UAI).*

Liang, S.; Fuhrman, S.; and Somogyi, R. 1998. Reveal, a general reverse engineering algorithm for inference of genetic network architectures. *Pacific Symposium on Biocomputing '98* 3:18–29.

Weaver, D.; Workman, C.; and Stormo, G. 1999. Modeling regulatory networks with weight matrices. *Pacific Symposium on Biocomputing '99* 4:112–123.

Wessels, L.; Reinders, M.; Baldochi, R.; and Gray, J. 1999. Statistical analysis of gene expression data. *Proceedings of the Ninth Belgium-Dutch Conference on Machine Learning (BENELEARN'99).*