Genetic Network Inference:
From Co-Expression Clustering to Reverse Engineering

PATRIK D'HAESELEER, SHOUDAN LIANG AND ROLAND SOMOGYI

P. D.
University of New Mexico
Department of Computer Science,
Albuquerque, NM 87131
patrik@cs.unm.edu

S. L.
NASA Ames Research Center
Moffett Field, CA 94035
sliang@mail.arc.nasa.gov
and
Incyte Pharmaceuticals, Inc.
3174 Porter Dr.,
Palo Alto, CA 94304

R. S.
Incyte Pharmaceuticals, Inc.
3174 Porter Dr.,
Palo Alto, CA 94304
rsomogyi@incyte.com

Abstract

Advances in molecular biological, analytical and computational technologies are enabling us to investigate systematically the complex molecular processes underlying biological systems. In particular, using high-throughput gene expression assays, we are able to measure the output of the gene regulatory network. We aim here to review datamining and modeling approaches for conceptualizing and unraveling the functional relationships implicit in these datasets. Clustering of co-expression profiles allows us to infer shared regulatory inputs and functional pathways. We discuss various aspects of clustering, ranging from distance measures to clustering algorithms and multiple cluster memberships. More advanced analysis aims to infer causal connections between genes directly, i.e. who is regulating whom and how. We discuss several approaches to the problem of reverse engineering of genetic networks, from discrete Boolean networks, to continuous linear and non-linear models. We conclude that the combination of predictive modeling with systematic experimental verification will be required to gain deeper insight into living organisms, therapeutic targeting and bioengineering.

1. Introduction

Novel, high-throughput technologies are opening global perspectives of living organisms on the molecular level. Together with a vast experimental literature on biomolecular processes, these data are now providing us with the challenge of *multifunctionality*, implying regulatory *networks* as opposed to isolated, linear pathways of causality (Szallasi, 1999). Questions which have traditionally been posed in the singular are now being addressed in the plural:

- What are the *functions* of this gene?
- Which *genes* regulate this gene?
- Which *genes* are responsible for this disease?
- Which *drugs* will treat this disease?

Beginning with gene sequencing, we are identifying the structure of thousands of genes, and a variety of structural and regulatory features that provide functional clues. However, only the molecular machinery of the cell is able to consistently interpret the sequence information into the functions which determine the complex genetic and biochemical networks that define the behavior of an organism. Since we ultimately seek understanding of the regulatory skeleton of these networks, we are also taking steps to monitor the molecular activities on a global level to reflect the effective functional state of a biological system. Several technologies, ranging from hybridization microarrays, automated RTPCR, to 2-D gel electrophoresis and antibody arrays allow us to assay RNA and protein expression profiles with differing levels of precision and depth.

But how should we organize this process of activity data acquisition? How should we interpret these results to enhance our understanding of living organisms, and identify therapeutic targets and critical processes for bioengineering? Here it is crucial to find the proper abstractions around which to build modeling frameworks and data analysis tools. These abstractions must be centered on two important principles:

1. Genetic information flow: *Defining the mapping from sequence space to functional space*

The genome contains the information for constructing the complex molecular features of an organism, as reflected in the process of development. In information terms, the complexity of the fully developed organism is contained in that of the genome. But which processes link these two types of information, in other words, what are the codes that translate sequence into structure and function? We need to represent these codes in a form understandable to us, so that we may apply them to model building and prediction. We therefore seek methods that allow us to extract these codes from gene sequence and activity data.

2. Complex dynamic systems: *From states to trajectories to attractors*

When addressing biological function, we usually refer to functions in time, i.e. causality and dynamics. On a molecular level, function is manifested in the behavior of complex networks. The dynamics of these networks resemble trajectories of state transitions – this is what we are monitoring when conducting temporal gene expression surveys. The concept of attractors is what really lends meaning to these trajectories, i.e. the attractors are the high dimensional dynamic molecular representations of stable phenotypic structures such as differentiated cells and tissues, either healthy or diseased (Kauffman, 1993, Somogyi & Sniegoski, 1996). Our goal is to understand the dynamics to the point where we can predict the attractor of a molecular network, and know enough about the network architecture to direct these networks to attractors of choice, e.g. from a cancerous cell type to a benign cell type, from degeneration to regeneration etc.

The goal of this review is to discuss principles of genetic network organization, and computational methods for extracting network architectures from experimental data. In Section 2 we will present a brief introduction to Boolean Networks, as a useful conceptual framework to think about the dynamic behavior of large, interacting regulatory networks. Section 3 describes methods and applications for clustering of genes based on their expression patterns. In Section 4, we give an overview of the different modeling methodologies that may be used to model regulatory networks. Finally, Section 5 shows how genetic regulatory networks may be inferred directly from large-scale gene expression data, including estimates of how much data is needed to do so.

2. A Conceptual Approach to Complex Network Dynamics

In higher metazoa, each gene or protein is estimated on average to interact with four to eight other genes (Arnone and Davidson, 1997), and to be involved in ten biological functions (Miklos and Rubin, 1996). The global gene expression pattern is therefore the result of the collective behavior of individual regulatory pathways. In such highly interconnected cellular signaling networks, gene function depends on its cellular context; thus understanding the network as a whole is essential. However, dynamic systems with large numbers of variables present a difficult mathematical problem.

One way to make progress in understanding the principles of network behavior is to radically simplify the individual molecular interactions, and focus on the collective outcome. Boolean networks (Kauffman 1969) represent such a simplification: each gene is considered as a binary variable—either ON or OFF—regulated by other genes through logical or Boolean functions (as can be found in the biochemical literature: see e.g. Yuh *et al*., 1998). Even with this simplification, the network behavior is already extremely rich (Kauffman 1993). Many useful concepts naturally emerge from such a simple mathematical model. For example, cell differentiation corresponds to transitions from one global gene expression pattern to another. Stability of global gene expression patterns can be understood as a

consequence of the dynamic properties of the network, namely that all networks fall into one or more attractors, representing stable states of cell differentiation, adaptation or disease.

For a Boolean network with N genes, the total number of global gene expression patterns can be very large even for moderate N (2^N: each gene can be either on or off independently). We assume that each gene is controlled by up to K other genes in the network. The connectivity K is a very important parameter for network dynamics (with large K, the dynamics tends to be more chaotic). In Random Boolean Network models, these K inputs, and a K-input Boolean function, are chosen at random for each gene (for a justification of this choice, see Kauffman, 1993). The Boolean variables (ON/OFF states of the genes) at time t+1 are determined by the state of the network at time t through the K inputs as well as the logical function assigned to each gene. An excellent tool for calculating and visualizing the dynamics of these networks is the DDLAB software (Wuensche, 1993, 1998).

Because the total number of expression patterns is finite, the system will eventually return to an expression pattern it has visited earlier. Since the system is deterministic, it will from then on keep following the exact same cycle of expression patterns. This periodic state cycle is called the attractor of the network. For example, in Figure 1 we show the repeating 6-state cycle attractor pattern within the trajectory of a 12-element network (right panel). This trajectory is only one of many alternative trajectories also leading to the same attractor, as shown in the "basin of attraction" graph (lower panel). If a state within the basin of attraction is perturbed to any other state within it, the dynamics inexorably converge to the same attractor. This feature helps explain why cell types or cellular adaptations to particular environments are stable with respect to small perturbations from a variety of internal or external noise sources.

The long-term dynamics are determined by the attractors. In the non-chaotic case, which is the only case of practical interest, the number of states in the attractor is typically only a very small fraction of the total number of states, growing as a square root of N (Kauffman, 1993; Bhattacharjya and Liang, 1996), rather than exponentially with N. The notion that gene expression patterns are constrained is in general agreement with the experimental findings of large-scale gene expression data. For example, genes from the same sequence or functional family do not act independently, but tend to fluctuate in parallel, reducing effective N (Wen *et al*., 1998). In terms of attractor cycles, it is rare for a gene to oscillate more than once during the cell cycle in the yeast (Chu *et al*., 1998). Also, the change in gene expression pattern during diauxic shift—when yeast metabolism switches from glucose to ethanol (DeRisi *et al*., 1997)—has been found to be very similar to the change in expression pattern during adaptive evolution to growth in a low glucose medium (Ferea *et al*., 1999).

The number of distinct attractors of a Random Boolean Network also tends to grow as a square root of the number of genes (Kauffman, 1993). If we equate the attractors of the network with individual cell types, as Kauffman suggests, this explains why a large genome of a few billion base pairs is capable of a few hundred stable cell types. This convergent behavior implies immense complexity reduction, convergence and stabilization in networks of constrained architecture.

Boolean networks provide a useful conceptual tool for investigating the principles of network organization and dynamics (Wuensche, 1999). We can study the role of various constraints on global behavior in terms of network complexity, stability and evolvability. From experimental studies we are learning about constraints in the number of inputs and outputs per gene, input and output sharing among genes evolved within a gene family or pathway, and restrictions on rule types (thresholding, no "exclusive or" rules etc.). Investigations into abstract models will help us understand the cybernetic

significance of network features, and provide meaningful questions for targeted experimental exploration.

3. Inference of regulation through clustering of gene expression data

Introduction

Large-scale gene screening technologies such as mRNA hybridization micro-arrays have dramatically increased our ability to explore the living organism at the genomic level (Zweiger, 1999). Large amounts of data can be routinely generated. In order to identify genes of interest, we need software tools capable of selecting and screening candidate genes for further investigation (Somogyi, 1999). At the simplest level, we can determine which genes show significant expression changes compared to a control group in a pair-wise comparison. As data sets become more complex, covering a variety of biological conditions or time series, one may consider several scoring methods for selecting the most interesting genes; e.g. according to a) whether there has been a significant change at any one condition, b) whether there has been a significant aggregate change over all conditions, c) or whether the fluctuation pattern shows high diversity according to Shannon entropy (Fuhrman et al., 2000). A comparison of these criteria in the analysis for toxic response genes has shown that the Shannon entropy allows the clearest distinction of drug-specific expression patterns (Cunningham et al., 2000).

Beyond straightforward scoring methods, we would like to classify gene expression patterns to explore shared functions and regulation. This can be accomplished using clustering methods. The simplest approach to clustering, sometimes referred to as GBA (Guilt By Association), is to select a gene and determine its nearest neighbors in expression space within a certain user defined distance cutoff (Walker et al., 1999). Genes sharing the same expression pattern are likely to be involved in the same regulatory process. Clustering allows us to extract groups of genes that are tightly co-expressed over a range of different experiments. If the promoter regions of the genes are known—as is the case for yeast—it is possible to identify the cis-regulatory control elements shared by the co-expressed genes. Several algorithms to extract common regulatory motifs from gene clusters have been developed (Brazma *et al*., 1998; Roth *et al*., 1998; van Helden *et al*., 1998; Wolfsberg *et al*., 1999). For example, Tavazoie *et al*. (1999) identified eighteen biologically significant DNA-motifs in the promoter region of genes clustered based on cell-cycle expression patterns. Most motifs were highly selective for the cluster in which they were found, and seven were known regulatory motifs for the genes in their respective clusters. In an example from brain development (Figure 2), correlations between cis elements and expression profiles can be established, but are sensitive to the clustering method used.

We will briefly review important issues involved in clustering and some of the main clustering methods used, as well as a few classical clustering methods which have not yet been adopted in gene expression analysis. We should caution the reader that different clustering methods can have very different results (see for example Figure 2), and—at this point—it is not yet clear which clustering methods are most useful for gene expression analysis. Claverie (1999) provides a preliminary review of gene expression analysis techniques with a focus on coexpression clustering. Niehrs and Pollet (1999) provide an overview of very tightly coexpressed groups of genes (which they call "synexpression groups") that have been identified based on large-scale gene expression data. For further reading, some useful textbooks on clustering include Massart and Kaufman (1983), Aldenderfer and Blashfield (1984) and Kaufman and Rousseeuw (1990).

Distance measures and preprocessing

Most clustering algorithms take a matrix of pairwise distances between genes as input. The choice of distance measure—used to quantify the difference in expression profiles between two genes—may be as important as the choice of clustering algorithm. Distance measures can be divided into at least three classes, emphasizing different regularities present within the data: a) similarity according to positive correlations, which may identify similar or identical regulation; b) similarity according to positive and negative correlations, which may also help identify control processes that antagonistically regulate downstream pathways; c) similarity according to mutual information, which may detect even more complex relationships.

So far, most clustering studies in the gene expression literature use either Euclidean distance or Pearson correlation between expression profiles as a distance measure. Other measures used include Euclidean distance between expression profiles and slopes (for time series; Wen *et al.*, 1998), squared Pearson correlation (D'haeseleer *et al.*, 1997), Euclidean distance between pairwise correlations to all other genes (Ewing *et al.*, 1999), Spearman rank correlation (D'haeseleer *et al.*, 1997), and mutual information (D'haeseleer *et al.*, 1997; Michaels *et al.*, 1998; Butte and Kohane, 2000).

Conspicuously absent so far are distance measures that can deal with the large numbers of highly related measurements in the data sets. For example, clustering yeast genes based on all publicly available data will be highly biased towards the large cell cycle data sets: 73 data points in 4 time series, containing almost 8 complete cell cycles (Spellman *et al.*, 1998), whereas only a single data point may be present for various stress conditions, mutations, etc. Correlation between the experiments will also lead to highly elliptical clusters, which form a problem for clustering methods that are biased towards compact, round clusters (such as K-means). A distance measure that can deal with the covariance between experiments in a principled way (e.g. Mahalanobis distance; Mahalanobis, 1936) may be more appropriate here. For even longer time series, distance measures based on Fourier or wavelet transforms may be considered.

A related issue is normalization and other preprocessing of the data. Distance measures that are sensitive to scaling and/or offsets (such as Euclidean distance) may require normalization of the data. Normalization can be done with respect to the maximum expression level for each gene, with respect to both minimum and maximum expression level or with respect to the mean and standard deviation of each expression profile. From a statistical point of view, we recommend using the latter, unless there is a good reason to preserve the mean expression values. See Figure 2 for an example of a single data set clustered using different normalizations and clustering methods. When using relative expression levels (for example, microarray data), the data will tend to be log-normally distributed, so the logarithm of the relative expression values should be used. Califano *et al.* (2000) suggest using a nonlinear transformation into a uniform distribution for each gene instead, which will tend to spread out the clusters more effectively.

Clustering algorithms

All clustering algorithms assume the preexistence of groupings of the objects to be clustered. Random noise and other uncertainties have obscured these groupings. The objectives of the clustering algorithm are to recover the original grouping among the data.

Clustering algorithms can be divided into hierarchical and non-hierarchical methods. Non-hierarchical methods typically cluster N objects into K groups in an iterative process until certain goodness criteria are optimized. Examples of non-hierarchical methods include K-means, EM (expectation-maximization)

and Autoclass. Hierarchical methods return a hierarchy of nested clusters, where each cluster typically consists of the union of two or more smaller clusters. The hierarchical methods can be further distinguished into agglomerative and divisive methods, depending on whether they start with single-object clusters and recursively merge them into larger clusters, or start with the cluster containing all objects and recursively divide it into smaller clusters. In this section, we review several clustering methods for gene expression (see also Figure 2 for comparison of agglomerative and K-means clustering).

The *K-means* algorithm (MacQueen, 1967) can be used to partition N genes into K clusters, where K is pre-determined by the user (see Tavazoie *et al. (*1999) for an application to yeast gene expression). K initial cluster "centroids" are chosen—either by the user, to reflect representative expression patterns, or at random—and each gene is assigned to the cluster with the nearest centroid. Next, the centroid for each cluster is recalculated as the average expression pattern of all genes belonging to the cluster, and genes are reassigned to the closest centroid. Membership in the clusters and cluster centroids are updated iteratively until no more changes occur, or the amount of change falls below a pre-defined threshold. K-means clustering minimizes the sum of the squared distance to the centroids, which tends to result in round clusters. Different random initial seeds can be tried to assess the robustness of the clustering results.

The *Self-Organized Map* (SOM) method is closely related to K-means and has been applied to mRNA expression data of yeast cell cycles as well as hematopoietic differentiation of four well-studied model cell lines (Tamayo 1999). The method is more structured than K-means in that the cluster centers are located on a grid. At each iteration, a randomly selected gene expression pattern attracts the nearest cluster center, plus some of its neighbors in the grid. Over time, fewer cluster centers are updated at each iteration, until finally only the nearest cluster is drawn towards each gene, placing the cluster centers in the center of gravity of the surrounding expression patterns. Drawbacks of this method are that the user has to specify *a priori* the number of clusters (as for K-means), as well as the grid topology, including the dimensions of the grid (typically one, two or three-dimensional) and the number of clusters in each dimension (e.g. 8 clusters could be mapped to a 2x4 2D grid or a 2x2x2 3D cube). The artificial grid structure makes it very easy to visualize the results, but may have residual effects on the final clustering. Optimization techniques for selecting the number of clusters developed for K-means can presumably be used here too.

The *Expectation-Maximization (EM)* algorithm (Dempster *et al.*, 1977) for fitting a mixture of Gaussians (also known as Fuzzy K-Means; Bezdek, 1981) is very similar to K-means, and has been used by Mjolsness *et al.* (1999b) to cluster yeast data. Rather than classifying each gene into one specific cluster, we can assign membership functions (typically Gaussians, or any other parametric probability distribution) to each cluster, allowing each gene to be part of several clusters. As in K-means, we alternately update the membership for each expression pattern, and then the parameters associated with each cluster: centroid, covariance and mixture weight. Cluster boundaries are sharp and linear in K-means, smooth and rounded in EM.

Autoclass is also related to EM, in that it finds a mixture of probability distributions. In addition, it uses Bayesian methods to derive the maximum posterior probability classification, and the optimum number of clusters (Cheeseman and Stutz, 1996).

Wen *et al.* (1998) used the FITCH hierarchical clustering algorithm (Felsenstein, 1993) to group the expression patterns of 112 genes in spinal cord development, producing a graph similar to the

phylogenetic trees familiar to most biologists (Sokal 1958). The expression clusters captured the main waves of gene expression in development. While the algorithm used in this study minimizes the overall distance in the tree, the computational requirement grows with the fourth power of the number of elements, making it impractical for much larger data sets.

Eisen *et al.* (1998) applied a standard agglomerative hierarchical clustering algorithm, average-linkage analysis, to large-scale gene expression data. Starting with N clusters containing a single gene each, at each step in the iteration the two closest clusters are merged into a larger cluster. Distance between clusters is defined as the distance between their average expression pattern. After N-1 steps, all the genes are merged together into a hierarchical tree. Other hierarchical methods may calculate distance between clusters differently. In UPGMA (unweighted pair-group method using arithmetic averages; Sneath and Sokal, 1973) for example, the distance between two clusters is defined as the average distance between genes in the two clusters.

Ben-Dor and Yakhini (1999) have developed a clustering algorithm based on random graph theory. Their method shares features with both agglomerative hierarchical clustering and K-means. Clusters are constructed one at a time. The gene with the largest "affinity" (smallest average distance to all other genes in the cluster) is added to the cluster, if the affinity is larger than a cutoff. A gene can also be removed from the cluster if its affinity drops below the cutoff. A finite number of clusters are constructed depending on the cutoff. The ability to remove ill-fitting genes from the cluster is an attractive feature of this algorithm. Zhu and Zhang (2000) used a similar algorithm to cluster yeast sporulation data.

Alon *et al* (1999) used a divisive hierarchical algorithm to cluster gene expression data of colon cancer. The method relies on the maximum entropy principle and attempts to find the most likely partition of data into clusters at a given "cost" (sum of squared within-cluster distances). Starting from a single cluster with large cost, as the allowed cost is lowered, the cluster breaks up spontaneously into multiple clusters in order to maximize the entropy for the configuration, within the constraint of fixed total cost.

Califano *et al.* (1999) have developed a clustering algorithm to identify groups of genes which can be used for phenotype classification of cell types, by searching for clusters of microarray samples that are highly correlated over a subset of genes. Only the most significant clusters are returned. The same technique could be used to find clusters of genes that are highly coexpressed over a subset of their expression profiles. Han *et al.* (1997) used a similar, partial matching approach to group objects into a *hypergraph* based on correlations over subsets of the data. In a hypergraph, each *hyperedge* (corresponding to a single cluster) connects several nodes (genes), so each node (gene) can be part of several hyperedges (clusters). Mjolsness *et al.* (1999a) developed a hierarchical algorithm that places objects into a directed, acyclic graph, where each cluster can be part of several parent clusters. The algorithm optimizes the number of clusters, cluster positions and partial cluster memberships of objects, such as to provide the most compact graph structure. All three clustering methods allow genes to be part of several clusters, possibly coinciding with multiple regulatory motifs or multiple functional classifications for each gene. This makes them especially appropriate for eukaryotic gene expression where genes are controlled by complex inputs from multiple transcription factors and enhancers.

Other aplications of co-expression clusters

Gene expression clustering is potentially useful in at least three areas: (1) extraction of regulatory motifs (co-regulation from co-expression); (2) inference of functional annotation; (3) as a molecular signature in distinguishing cell or tissue types.

Genes in the same expression cluster will tend to share biological functions. In a system as complex as the developing rat spinal cord, expression clustering clearly led to a segregation according to functional genes families (Wen *et al*., 1998). Moreover, cluster-function relationships exist over several methods of classification; for example, neurotransmitter receptor ligand classes and sequence/pathway groups each selectively map to expression waves (Figure 3). Tavazoie *et al*. (1999), used K-means to partition yeast genes during cell cycle into 30 distinct clusters, and found the members of each cluster to be significantly enriched for genes with similar functions. Functions of unknown genes may be hypothesized from genes with known function within the same cluster. Yeast genes with previously unknown functions have been identified from their temporal pattern of expression during spore morphogenesis and their functional role in sporulation has been confirmed in deletion experiments (Chu *et al*., 1998).

mRNA expression can be regarded as a molecular signature of a cell's phenotype. Clustering of gene expression patterns helps differentiate different cell types, which is useful, for example, in recognizing subclasses of cancers. (Alon *et al*., 1999; Golub *et al*., 1999; Perou *et al*., 1999; Alizadeh *et al*., 2000). Two-way clustering of both the genes and experiments allows for easy visualization (Eisen *et al*., 1998; Alon *et al*., 1999; Alizadeh *et al*., 2000; Weinstein *et al*., 1997). Because activities of genes are often related to each other, gene expression is highly constrained, and gene expression patterns under different conditions can be very similar. Clustering is necessary for identifying the coherent patterns.

Which clustering method to use?

We have discussed several different distance measures and clustering algorithms. Each combination of distance measure and clustering algorithm will tend to emphasize different types of regularities in the data. Some may be useless for what we want to do. Others may give us complementary pieces of information. Jain and Dubes (1988) state:

> There is no single best criterion for obtaining a partition because no precise and workable definition of ``cluster'' exists. Clusters can be of any arbitrary shapes and sizes in a multidimensional pattern space. Each clustering criterion imposes a certain structure on the data, and if the data happens to conform to the requirements of a particular criterion, the true clusters are recovered.

It is impossible to objectively evaluate how "good" a specific clustering is without referring to what the clustering will be used for. However, once an application has been identified, it may be possible to evaluate objectively the quality of the clustering for that particular application. For example, if we want to extract regulatory motifs from clusters, we can compare clustering methods based on the P-values of the resulting motifs. Similarly, for functional classification, we can compare P-values associated with enrichment of clusters in certain functional categories. It is unlikely that there would be a single best clustering method for all applications. Considering the overwhelming number of combinations of distance measures and clustering algorithms—far too many to try them all each time—the field is in dire need of a comparison study of the main combinations for some of the standard applications, such as functional classification or extraction of regulatory motifs. If we want to use gene clusters to infer regulatory interactions, synthetic data generated from small but detailed models of regulatory networks could provide a useful touchstone for comparing clustering methods. Preliminary results comparing SOM, K-means, FITCH and Autoclass—all using Euclidean distance—showed very poor performance of all clustering methods in identifying a metabolic pathway with associated regulation of the enzymes by the metabolites (Mendes, 1999).

The greatest challenge in cluster analysis lies in faithfully capturing complex relationships in biological networks. As stated above, a gene may participate in multiple functions, and is controlled by the activities of many other genes through a variety of cis-regulatory elements. Therefore, for complex datasets spanning a variety of biological responses, a gene should by definition be a member of several clusters, each reflecting a particular aspect of its function and control. As more data becomes available to accurately delineate expression behavior under different conditions, we should consider using some of the clustering methods that partition genes into non-exclusive clusters. Alternatively, several clustering methods could be used simultaneously, allocating each gene to several clusters based on the different regularities emphasized by each method.

4. Modeling Methodologies

Cluster analysis can help elucidate the regulation (or co-regulation) of individual genes, but eventually we will have to consider the integrated behavior of networks of regulatory interactions. Various types of gene regulation network models have been proposed, and the model of choice is often determined by the question one is trying to answer. In this section we will briefly address some of the decisions that need to be made when constructing a network model, and the tradeoffs associated with each.

Level of biochemical detail

Gene regulation models can vary from the very abstract—such as Kauffman's random Boolean networks (Kauffman, 1993)—to the very concrete—like the full biochemical interaction models with stochastic kinetics in Arkin *et al.* (1998). The former approach is the most mathematically tractable, and its simplicity allows examination of very large systems (thousands of genes). The latter fits the biochemical reality better and may carry more weight with the experimental biologists, but its complexity necessarily restricts it to very small systems. For example, the detailed biochemical model of the five-gene lysis-lysogeny switch in Lambda phage (Arkin *et al.*, 1998) included a total of 67 parameters, and required supercomputers for its stochastic simulation.

In-depth biochemical modeling is very important in understanding the precise interactions in common regulatory mechanisms, but clearly we cannot expect to construct such a detailed molecular model of, say, an entire yeast cell with some 6000 genes by analyzing each gene and determining all the binding and reaction constants one-by-one. Likewise, from the perspective of drug target identification for human disease, we cannot realistically hope to characterize all the relevant molecular interactions one-by-one as a requirement for building a predictive disease model. There is a need for methods that can handle large-scale data in a global fashion, and that can analyze these large systems at some intermediate level, without going all the way down to the exact biochemical reactions.

Boolean or continuous

The Boolean approximation assumes highly cooperative binding (very "sharp" activation response curves) and/or positive feedback loops to make the variables saturate in ON or OFF positions. However, examining real gene expression data, it seems clear that genes spend a lot of their time at intermediate values: gene expression levels tend to be continuous rather than binary. Furthermore, important concepts in control theory that seem indispensable for gene regulation systems either cannot be implemented with Boolean variables, or lead to a radically different dynamical behavior: amplification, subtraction and addition of signals; smoothly varying an internal parameter to compensate for a continuously varying environmental parameter; smoothly varying the period of a periodic

phenomenon like the cell cycle, etc. Feedback control (see e.g. Franklin *et al.*, 1994) is one of the most important tools used in control theory to regulate system variables to a desired level, and reduce sensitivity to both external disturbances and variation of system parameters. Negative feedback with a moderate feedback gain has a stabilizing effect on the output of the system. However, negative feedback in Boolean circuits will always cause oscillations, rather than increased stability, because the Boolean transfer function effectively has an infinite slope (saturating at 0 and 1). Moreover, Savageau (1998) identified several rules for gene circuitry (bacterial operons) that can only be captured by continuous analysis methods. In this study, positive and negative modes of regulation were respectively linked to high and low demand for expression, and a relationship was established between the coupling of regulator and effector genes and circuit capacity and demand.

Some of these problems can be alleviated by hybrid Boolean systems. In particular, Glass (1975,1978) has proposed sets of piecewise linear differential equations, where each gene has a continuous-valued internal state, and a Boolean external state. Researchers at the Free University of Brussels (Thomas, 1991; Thiefry and Thomas, 1999) have proposed an asynchronously updated logic with intermediate threshold values. These systems allow for easy analysis of certain properties of networks, and have been used for qualitative models of small gene networks, but still do not seem appropriate for quantitative modeling of real, large-scale gene expression data.

Deterministic or stochastic

One implicit assumption in continuous-valued models is that fluctuations in the range of single molecules can be ignored. Differential equations are already widely used to model biochemical systems, and a continuous approach may be sufficient for a large variety of interesting mechanisms. However, molecules present at only a few copies per cell do play an important role in some biological phenomena, such as the lysis-lysogeny switch in Lambda phage (Ptashne, 1992). In that case, it may be impossible to model the behavior of the system exactly with a purely deterministic model.

These stochastic effects—which have mainly been observed in prokaryotes—may not play as much of a role in the larger eukaryotic cells. In yeast, most mRNA species seem to occur at close to one mRNA copy per cell (Velculescu *et al.*, 1997; Holstege *et al.*, 1998a), down to 0.1 mRNA/cell or less (i.e. the mRNA is only present 10% of the time or less in any one cell). Low copy numbers like these could be due to leaky transcription and not have any regulatory role. Also, if the half-life of the corresponding protein (typically measured in hours or days) is much larger than the half-life of the mRNA (averaging around 20 min in yeast (Holstege *et al.*, 1998b)), the protein level may not be affected by stochastic fluctuations in mRNA. Analysis of mRNA and protein decay rates and abundances may allow us to identify those few genes for which stochastic modeling may prove necessary.

Particle-based models can keep track of individual molecule counts, and often include much biochemical detail and/or spatial structure. Of course, keeping track of all this detail is computationally expensive, so they are typically only used for small systems. A related modeling technique is Stochastic Petri Nets (SPN's), which can be considered a subset of Markov processes, and can be used to model molecular interactions (Goss and Peccoud, 1998). Whereas fitting the parameters of a general particle model to real data can be quite difficult, optimization algorithms exist for SPN's. Hybrid Petri Nets (Mounts and Liebman, 1997; Matsuno *et al.*, 2000) include both discrete and continuous variables, allowing them to model both small-copy number and mass action interactions.

Additional sources of unpredictability can include external noise, or errors on measured data. The Bayesian approach to unpredictability is to construct models that can manipulate entire probability

distributions rather than just single values. Stochastic differential equations could be used for example. Of course, this does add a whole new level of complexity to the models. Alternatively, a deterministic model can sometimes be extended by a simplified analysis of the variance on the expected behavior.

Spatial or non-spatial

Spatiality can play an important role, both at the level of intercellular interactions, and at the level of cell compartments (e.g. nucleus vs. cytoplasm vs. membrane). Most processes in multicellular organisms, especially during development, involve interactions between different cells types, or even between cells of the same type. Some useful information may be extracted using a nonspatial model (see for example D'haeseleer *et al.* (1999, 2000) for a non-spatial model of CNS development and injury), but eventually a spatial model will be needed.

Spatiality adds a whole extra dimension of complexity to the models: spatial development, cell type interactions, reservoirs, diffusion constants, etc. In some cases, the abundance of data— spatial patterns—can more than make up for the extra complexity of the model. For example, Mjolsness *et al.* (1991) used a time series of one-dimensional spatial patterns to fit a simple model of *eve* stripe formation in Drosophila. Models like the ones proposed by Marnellos and Mjolsness (1998) for the role of lateral interactions in early Drosophila neurogenesis provide experimentally testable predictions about potentially important interactions.

Data availability

In general, we must also realize that molecular activity measurements are limited and are carried out over population averages of cells, not on individual cell circuits. Modeling methodologies must therefore be designed around the level of organization for which data is actually accessible. An exhaustive model must take into account RNA and protein concentrations, phosphorylation states, molecular localization and so forth, since each molecular variable carries unique information. However, due to present limitations in measuring technology, these data are not routinely accessible. Modeling is then challenged with providing as much predictive power as possible given limited data on molecular states. The constraints and redundancies in biological networks suggest that much may still be gained even though not all parameters involved in the process may be modeled.

Forward and inverse modeling

Some of the more detailed modeling methodologies listed above have been used to construct computer models of small, well-described regulatory networks. Of course, this requires an extensive knowledge of the system in question, often resulting from decades of research. In this review, we will not focus on this *forward modeling* approach, but rather on the *inverse modeling*, or *reverse engineering* problem: given an amount of data, what can we deduce about the unknown underlying regulatory network? Reverse engineering typically requires the use of a parametric model, the parameters of which are then fit to the real-world data. If the connection structure of the regulatory network (i.e. which genes have a regulatory effect on each other) is unknown, the parametric model will necessarily have to be very general and simplistic. The results of this sort of model only relate to the overall network structure. While this will imply little about the actual molecular mechanisms involved, much helpful information will be gained on genes critical for a biological process, sufficient for e.g. the identification of drug targets. Once the network structure is well known, a more detailed model may be used to estimate individual mechanism-related parameters, such as binding and decay constants.

5. Gene network inference: Reverse engineering

Clustering is a relatively easy way to extract useful information out of large-scale gene expression data sets, however, it typically only tells us which genes are co-regulated, not what is regulating what. In network inference, the goal is to construct a coarse-scale model of the network of regulatory interactions between the genes. This requires inference of the causal relationships among genes, i.e. reverse engineering the network architecture from its activity profiles. As the molecular dynamics data we acquire becomes more copious and complex, we may need to routinely consult reverse engineering methods to provide the guiding hypotheses for experimental design.

One may wonder whether it is at all possible to reverse engineer a network from its activity profiles. A straightforward answer to this question should be obtainable from model networks, e.g. Boolean networks, for which we understand the network architectures and can easily generate activity profiles. In a first attempt, a simple method was introduced that showed that reverse engineering is possible in principle (Somogyi *et al.*, 1997). A more systematic and general algorithm was developed by Liang *et al.* (1998), using Mutual Information to identify a minimal set of inputs that uniquely define the output for each gene at the next time step. Akutsu *et al.* (1999) proposed a simplified reverse engineering algorithm and rigorously examined the input data requirements. For more realistic applications, a further modification was introduced by Akutsu *et al.* (2000) that allows the inference of Boolean networks given noisy input data. Ideker *et al.* (2000) developed an alternative algorithm in which the minimal set covering problem is solved using the branch and bound technique. They devise a perturbation strategy that may be used by laboratory scientists for systematic experimental design. It should be pointed out that the problem of designing a Boolean circuit that corresponds to certain input-output mappings is well studied in electrical engineering, and several efficient algorithms exist (e.g. Espresso; Brayton *et al.*, 1984) that could provide inspiration for reverse engineering of Boolean regulatory networks.

Data requirements

To correctly infer the regulation of a single gene, we need to observe the expression of that gene under many different combinations of expression levels of its regulatory inputs. This implies sampling a wide variety of different environmental conditions and perturbations. Gene expression time series yield a lot of data, but all the data points tend to be about a single dynamical process in the cell, and will be related to the surrounding time points. A 10-point time series generally contains less information than a data set of ten expression measurements under dissimilar environmental conditions, or with mutations in different pathways. The advantage of the time series is that it can provide insight in the dynamics of the process. On the other hand, data sets consisting of individual measurements provide an efficient way to map the attractors of the network. Both types of data, and multiple data sets of each, will be needed to unravel the regulatory interactions of the genes.

Successful modeling efforts will probably have to use data from different sources, and will have to be able to deal with different data types such as time series and steady-state data, different error levels, incomplete data, etc. Whereas clustering methods can use data from different strains, in different growth media etc., combining data sets for reverse engineering of regulatory networks requires that differences between the experimental conditions be quantified much more precisely. Likewise, data will have to be calibrated properly to allow comparison with other data sets. In this respect, there is a growing need for a reliable reference in relative expression measurements. An obvious approach could be to agree on a

standard strain and carefully controlled growth conditions to use in all data collection efforts on the same organism. Alternatively, a reference mRNA population could be derived directly from the DNA itself.

Estimates for various network models

The ambitious goal of network reverse engineering comes at the price of requiring more data points. The space of models to be searched increases exponentially with the number of parameters of the model, and therefore with the number of variables. Narrowing the range of models by adding extra constraints can simplify the search for the best model considerably. Including such information into the inference process is the true art of modeling.

How many data points are needed to infer a gene network of N genes depends on the complexity of the model used to do the inference. Constraining the connectivity of the network (number of regulatory inputs per gene) and the nature of the regulatory interactions can dramatically reduce the amount of data needed. Table 1 provides an overview of some of the models considered, and estimates of the amount of data needed for each. These estimates hold for independently chosen data points, and only indicate asymptotic growth rates, ignoring any constant factors.

Model	Data needed
Boolean, fully connected	2^N
Boolean, connectivity K	$2^K(K+\log(N))$
Boolean, connectivity K, linearly separable	$K \log(N/K)$
Continuous, fully connected, additive	$N+1$
Continuous, connectivity K, additive	$K \log(N/K)$ (*)
Pairwise correlation comparisons (clustering)	$\log(N)$

Table 1: Fully connected: each gene can receive regulatory inputs from all other genes. Connectivity K: at most K regulatory inputs per gene. Additive, linearly separable: regulation can be modeled using a weighted sum. Pairwise correlation: significance level for pairwise comparisons based on correlation must decrease inversely proportional to number of variables. (*): conjecture.

To completely specify a fully connected Boolean network model, where the output of each gene is modeled as a Boolean function of the expression levels of all N genes, we need to measure all possible 2^N input-output pairs. This is clearly inconceivable for realistic numbers of genes. If we reduce the connectivity of the Boolean network to an average of K regulatory inputs per gene, the data requirements decrease significantly. For a slightly simpler model, we can derive a lower bound of $\Omega(2^K(K+\log(N)))$ (see Appendix A), which agrees well with preliminary experimental results by Liang *et al.* (1998) and Akutsu *et al.* (1999) (see Figure 4). Further constraining the Boolean model to use only linearly separable Boolean functions (those that can be implemented using a weighted sum of the inputs, followed by a threshold function) reduces the data requirements to $\Omega(K \log(N/K))$ (Hertz, 1998).

For models with continuous expression levels, the data requirements are less clear. In the case of linear (D'haeseleer *et al.*, 1999) or quasi-linear additive models (Weaver *et al.*, 1999), fitting the model is equivalent to performing a multiple regression, so at least $N+1$ data points are needed for a fully connected model of N genes. Data requirements for sparse additive regulation models are as yet unknown, but based on the similarity with the equivalent Boolean model, we speculate it to be of the

form $\Omega(K \log(N/K))$. A promising avenue of further research in this area may be the results on sample complexity for recurrent neural networks, which have a very similar structure to the models presented here (Koiran and Sontag, 1998; Sontag, 1997).

Finally, to allow for comparison with gene clustering methods, we examined data requirements for clustering based on pairwise correlation comparisons (see Appendix B). As the number of genes being compared increases, the number of data points will have to increase proportional to $\log(N)$, in order to maintain a constant, low level of false positives. Claverie (1999) arrived at a similar logarithmic scaling for binary data (absent/detected).

Note that for reasonably constrained models, the number of data points needed will scale with $\log(N)$ rather than N, and that the data requirements for network inference are at least a factor K larger than for clustering. In practice, the amount of data may need to be orders of magnitude higher because of non-independence and large measurement errors (see also Szallasi, 1999). Higher accuracy methods such as RT-PCR yield more bits of information per data point than cDNA microarrays or oligonucleotide chips, so fewer data points may be required to achieve the same accuracy in the model. Modeling real data with Boolean networks discards a lot of information in the data sets, because the expression levels need to be discretized to one bit per measurement. Continuous models will tend to use the available information in the data set better.

Correlation Metric Construction

Adam Arkin and John Ross have been working on a method called Correlation Metric Construction (Arkin and Ross, 1995; Arkin *et al.*, 1997), to reconstruct reaction networks from measured time series of the component chemical species. This approach is based in part on electronic circuit theory, general systems theory and multivariate statistics. Although aimed more towards cell signaling or metabolic networks, the same methodology could be applied to regulatory networks.

The system (a reactor vessel with chemicals implementing glycolysis) is driven using random (and independent) inputs for some of the chemical species, while the concentration of all the species is monitored over time. A distance matrix is constructed based on the maximum time-lagged correlation between any two chemical species. This distance matrix is then fed into a simple clustering algorithm to generate a tree of connections between the species, and the results are mapped into a 2D graph for visualization. It is also possible to use the information regarding the time lag between species at which the highest correlation was found, which could be useful to infer causal relationships. More sophisticated methods from general systems theory, based on mutual information, could be used to infer dependency.

Systems of differential equations

Simple systems of differential equations have already proven their worth in modeling simple gene regulation systems. For example, the seminal work of Mjolsness *et al.* (1991) used a spatial "gene circuit" approach to model a small number of genes involved in pattern formation during the blastoderm stage of development in Drosophila. The change in expression levels at each point in time depended on a weighted sum of inputs from other genes, and diffusion from neighboring "cells". Synchronized cell divisions along a longitudinal axis (under the control of a maternal clock) were alternated with updating the gene expression levels. The model was able to successfully replicate the pattern of eve stripes in Drosophila, as well as some mutant patterns on which the model was not explicitly trained.

Additive Regulation models

The differential equation systems described above model gene networks using an update rule based on a weighted sum of inputs. Several variants of such models have been proposed, with each group coining

a different name: connectionist model (Mjolsness *et al.*, 1991), linear model (D'haeseleer *et al.*, 1999), linear transcription model (Chen *et al.*, 1999), weight matrix model (Weaver *et al.*, 1999). The core of these seems to be the additive interaction between the regulatory inputs to each gene, so we suggest calling these models collectively *additive regulation models*.

In the simplest case, we can think of these models as being similar to multiple regression:

$$x_i \approx \sum_j w_{ji} x_j + b_i \quad \text{or} \quad x_i(t+\Delta t) = \sum_j w_{ji} x_j(t) + b_i$$

Where x_i is the expression level of gene i at time t, b_i is a bias term indicating whether gene i is expressed or not in the absence of regulatory inputs, and weight w_{ji} indicates the influence of gene j on the regulation of gene i. This can be written equivalently as a difference or differential equation. Given an equidistant time series of expression levels (or an equidistant interpolation of a non-equidistant time series), we can use linear algebra to find the least-squares fit to the data. Weaver *et al.* (1999) showed how a non-linear transfer function can be incorporated into the model as well, and demonstrated that some randomly generated networks can be accurately reconstructed using this modeling technique. D'haeseleer *et al.* (1999, 2000) showed that even a simple linear model can be used to infer biologically relevant regulatory relationships from real data sets (Figure 5).

Chen *et al.* (1999) presented a number of linear differential equation models which included both mRNA and protein levels. They showed how such models can be solved using linear algebra and Fourier transforms. Interestingly, they find that mRNA concentrations alone are not sufficient to solve their model, without at least the initial protein levels. Conversely, the model can be solved given only a time series of protein concentrations.

Models that are more complex may require more general methods for fitting the parameters to the expression data. Mjolsness *et al.* (1991) used Simulated Annealing to fit their hybrid model—incorporating both reaction-diffusion kinetics and cell divisions—to spatial data. Mjolsness *et al.* (1999b) used a similar approach to fit a recurrent neural network with weight decay to clusters of yeast genes. Genetic algorithms (GA's) have been used to model the interaction between four "waves" of coordinately regulated genes (Wahde and Hertz, 1999) previously identified in rat CNS development (Wen *et al.*, 1998). Similarly, Tominaga *et al.* (1999) used a GA to fit a power-law model (Savageau, 1995; Tominaga and Okamoto, 1998) of a small gene network. Networks with larger numbers of genes will likely require stronger optimization algorithms. Akutsu *et al.* (2000) proposes using Linear Programming for both power-law models and qualitative hybrid Boolean-like networks. Efficient gradient descent algorithms developed for continuous-time recurrent neural networks (Pearlmutter, 1995) may be useful for even larger networks (D'haeseleer *et al.*, 1999). Alternatively, the size of the problem can be drastically reduced by combining gene clustering with network inference, deriving a regulation model only for the cluster centers (Wahde and Hertz, 1999; Mjolsness *et al.*, 1999b).

6. Conclusions and Outlook

We are participating in the transition of biology into an information driven science. However, this transition can be meaningful only if we focus on generating models that allow us to systematically derive predictions about important biological processes in disease, development and metabolic control. These will find important applications in pharmaceutical development and bioengineering (Zweiger, 1999). We have reviewed conceptual foundations for understanding complex biological networks, and several

practical methods for data analysis. There are still major challenges ahead, which may be divided into five areas:

1) Measurement quantity, depth and quality. Any attempt at predictive data analysis and model building critically depends on the scope and quality of the input data. Ideally, we would like to gain access to the activities of all important molecular species in a biological process (ranging from mRNA to metabolites and second messengers), with adequate quantitative, anatomical and temporal resolution. However, even though our analytical measurement technologies are undergoing transformations in precision and throughput, there will always be limitations to the amount of data and resolution we can acquire and process. Computational data analysis must therefore identify the most essential molecular parameters to guide experimental measurements, and critically evaluate measurement precision and reproducibility with appropriate statistical measures.
2) Clustering and functional categorization. One priority in this area is to compare the large variety of existing clustering methods (including different normalizations and distance measures), and identify those that give the most biologically relevant results. Just as a gene can play multiple functional roles in various pathways and is subject to different regulatory inputs, co-expression patterns vary according to the cellular and experimental context. Methods for clustering according to co-expression profiles should select the appropriate experimental sets for analysis, and provide flexible solutions with multiple cluster memberships that more accurately reflect the biological reality. Well-designed cluster analysis promises to identify new pathway relationships and gene functions that may be critical to cellular control in health and disease.
3) Reverse Engineering. Since it is the ultimate goal to identify the causative relationships between gene products that determine important phenotypic parameters, top priority should be given to develop reverse engineering methods that provide significant predictions. Alternative computational approaches should be applied to given data sets, and their predictions tested in targeted experiments to identify the most reliable methods.
4) Integrated modeling. While the current focus is on the analysis of large-scale gene expression data, there are other established sources of information on gene function, ranging from sequence homology and cis-regulatory sequences, to disease association and a wide variety of functional knowledge from targeted experiments. Ideally, all of these categories of information should be included in model building. A major challenge here lies in the reliability and compatibility of these data sets.
5) Coupling of modeling with systematic experimental design. Discovery of novel gene function through expression profiling and computational inference depends on the optimal coordination of experimental technology with data analysis methods. While data analysis methods must be centered around data that are realistically accessible, critical predictions from the models must guide experimental design. The hope is that progressive iteration of predictions, experimental measurements and model updates will result in increased fidelity and depth of computational models of biological networks.

Acknowledgements

P.D. gratefully acknowledges the support of the National Science Foundation (grant IRI-9711199), the Office of Naval Research (grant N00014-99-1-0417), and the Intel Corporation. S.L. gratefully

acknowledges support from NASA Collaborative Agreement NCC 2-794. The authors would like to express their appreciation to Xiling Wen, Millicent Dugich and Stefanie Fuhrman for providing data on hippocampal development and injury, and to Stefanie Fuhrman for critically reading and commenting on the manuscript.

Appendix A: Data requirements for sparse Boolean networks

To fully specify a Boolean network with limited connectivity, we need to specify the connection pattern and the rule table for a function of K inputs at each. A lower bound of $\Omega(2^K+K\log(N))$ can be derived using information theory. For a slightly simpler model, where we assume the pattern of connectivity is given, we can calculate how the number of independently chosen data points should scale with K and N. Since this is a simpler model, its data requirements should be a lower bound to the requirements for the model with unknown connections.

Every data point (i.e. every input-output pair, specifying the state of the entire Boolean network at time t and t+1), specifies exactly one of 2^K entries in each rule table: Given this particular combination of the K inputs to each gene at time t, the output of the gene is given by its state at time t+1. We will estimate the probability P that all N rule tables are fully specified by n data points, and calculate how the number of data points n needs to scale with P, the number of genes N, and connectivity K. For $P\approx 1$ (i.e. we have enough data to have a good chance at a fully specified model), the probability for a single rule table to be fully specified by n data points is approximately:

$$1-2^K\left(1-2^{-K}\right)^n$$

The probability that all N rule tables are fully specified by n data points then becomes:

$$P\approx\left(1-2^K\left(1-2^{-K}\right)^n\right)^N$$

Taking base-2 logarithms, and further simplifying using $\log_2(1\text{-}z)\approx -z\log_2(e)$ for $z\ll 1$, we find:

$$\begin{aligned}C_1 &= -\log(P)\\ &\approx -N\log\left(1-2^K\left(1-2^{-K}\right)^n\right)\\ &\approx N2^K\left(1-2^{-K}\right)^n\log(e)\\ C_2 &= -\log\left(C_1/\log(e)\right)\\ &\approx -\log(N)-K-n\log\left(1-2^{-K}\right)\\ &\approx -\log(N)-K+n2^{-K}\log(e)\end{aligned}$$

If $P\approx 1$, C_1 will be a small, and C_2 a large positive constant. We can now express n, the number of data points needed, in terms of N, K and C_2:

$$n\approx 2^K\left(K+\log(N)+C_2\right)/\log(e)$$

which is $O(2^K(K+\log(N)))$.

Appendix B: Data requirements for pairwise correlation comparisons

Let us examine a very simple form of clustering as a representative example of the wide variety of clustering algorithms: we say that two genes cluster together if their correlation is significantly greater (with a significance level α) than a certain cutoff value ρ_c. We test whether we can exclude the null hypothesis $\rho<\rho_c$ based on the measured correlation coefficient r over the available data points. Because of the large number of comparisons being made, we need to reduce the significance level for the correlation test, proportional to the number of tests each gene is involved in: $\alpha=\alpha'/N$ (this will keep the expected number of false positives for each gene constant). In order to be able to use the same cutoff-value for the measured correlation r_α to decide whether two genes cluster together, the number of data points will have to increase as the significance level for each test grows smaller.

If the real correlation coefficient ρ is close to 1.0, the distribution of the measured correlation coefficient r is very asymmetrical. The following z-transformation, developed by Fisher (1969), is approximately normally distributed with mean $z(\rho)$ and variance $1/(n-3)$ (with n the number of data points):

$$z(r) = \frac{1}{2}\ln\left(\frac{1+r}{1-r}\right)$$

We can now devise a single-sided test on $z(r)$ to answer the question: If $z(r)>z(r_\alpha)$, what is the significance level with which we can reject the hypothesis $z(\rho)<z(\rho_c)$ (and thus $\rho<\rho_c$)? At the tail of the normal distribution, the area under the normal curve to the right of $z(r_\alpha)$ can be approximated by:

$$\alpha = \int_{z=z(r_\alpha)}^{\infty} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(z-z(\rho_c))^2}{2\sigma^2}} dz \approx \frac{\sigma}{\sqrt{2\pi}\left(z(r_\alpha)-z(\rho_c)\right)} e^{-\frac{(z(r_\alpha)-z(\rho_c))^2}{2\sigma^2}}$$

with $\sigma = 1/\sqrt{n-3}$, and taking natural logs:

$$\begin{aligned}\ln(\alpha) &\approx \ln(\alpha') - \ln(N) \\ &\approx -\frac{1}{2}\ln(n-3) - \ln\left(\sqrt{2\pi}\left(z(r_\alpha)-z(\rho_c)\right)\right) - (n-3)\left(z(r_\alpha)-z(\rho_c)\right)^2/2 \\ n &\approx 3 + \frac{2}{\left(z(r_\alpha)-z(\rho_c)\right)^2}\left(\ln(N) + \ln(1/\alpha') - \ln(n-3)/2 - \ln\left(\sqrt{2\pi}\left(z(r_\alpha)-z(\rho_c)\right)\right)\right) \\ &= O(\log(N))\end{aligned}$$

In other words, if we want to use the same cutoff value r_α to decide whether $\rho>\rho_c$, we need to scale the number of data points logarithmically with the number of genes. Strictly speaking, this analysis only holds for correlation tests, but we can expect similar effects to play a role in other clustering algorithms.

References

Agnew, B (1998) NIH plans bioengineering initiative. Science, 280:1516-1518

Akutsu, T., Miyano, S. and Kuhara, S. (1999) Identification of Genetic Networks from a Small Number of Gene Expression Patterns Under the Boolean Network Model. Pacific Symposium on Biocomputing 4:17-28. http://www.smi.stanford.edu/projects/helix/psb99/Akutsu.pdf

Akutsu, T., Miyano, S. and Kuhara, S. (2000) Algorithms for inferring qualitative models of biological networks. Pacific Symposium on Biocomputing, in press.

Aldenderfer M.S. and Blashfield R.K. (1984). Cluster analysis. Sage Publications, Newbury Park, CA.

Alizadeh, A.A. et al. (2000) Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature, 403(6769):503-511.

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(12):6745-6750.

Arkin, A. and Ross, J (1995) Statistical construction of chemical reaction mechanism from measured time-series. J. Physical Chemistry, 99, 970-979.

Arkin, A., Ross, J. and McAdams, H.H. (1998) Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected Escherichia coli cells. Genetics, 149, 1633-1648

Arkin, A., Shen, P. and Ross, J (1997) A test case of correlation metric construction of a reaction pathway from measurements. Science, 277, 1275-1279.

Arnone M.I. and Davidson E.H. (1997) The hardwiring of development: organization and function of genomic regulatory systems. Development, 124: 1851-1864.

Ben-Dor A. and Yakhini Z. (1999) Clustering gene expression patterns. International Conference on Computational Molecular Biology.

Bezdek J.C. (1981) Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York.

Bhattacharjya, A and Liang, S. (1996) Power Laws in Some Random Boolean Networks. Phys. Rev. Lett. 77, 1644.

Brayton R.K., Hachtel G.D., McMullen C.T. and Sangiovanni-Vincentelli A.L. (1984) Algorithms for VLSI synthesis. Kluwer Academic Publishers. ftp://ic.eecs.berkeley.edu/pub/Espresso/

Brazma A., Jonassen, I, Vilo J. and Ukkonen E. (1998) Predicting Gene Regulatory Elements in Silico on a Genomic Scale. Genome Research, 8, 1202-1215.

Brown P.O. and Botstein D. (1999) Exploring the new world of genome with DNA microarrays. Nature Genet. 21 33-37.

Butte A.J. and Kohane I.S. (2000) Mutual information relevance networks: Functional genomic clustering using pairwise entropy measurements. Pacific Symposium on Biocomputing, in press.

Califano A., Stolovitzky G. and Tu Y. (2000) Analysis of gene expression microarrays for phenotype classification. Submitted to the International Conference on Computational Molecular Biology 2000. http://www.cs.washington.edu/homes/amirbd/BioClust/Webpn/index.htm

Cheeseman P. and Stutz J. (1996) Bayesian Classification (AutoClass): Theory and Results. in Advances in Knowledge Discovery and Data Mining, Fayyad U.M., Piatetsky-Shapiro G., Smyth P. and Uthurusamy R., Eds. AAAI Press/MIT Press. http://ic-www.arc.nasa.gov/ic/projects/bayes-group/images/kdd-95.ps

Chen, T., He, H. L. and Church, G. M. (1999) Modeling gene expression with differential equations. Pacific Symposium on Biocomputing 4:29-40. http://www.smi.stanford.edu/projects/helix/psb99/Chen.pdf

Chu S., DeRisi J., Eisen M., Mulholland J., Botstein D., Brown P.O.and Herskowitz I. (1998) The transcriptional program of sporulation in budding yeast. Science 282:699-705.

Claverie, J.-M. (1999) Computational methods for the identification of differential and coordinated gene expression. Human Molecular Genetics, 8, 1821-1832.

Cunningham MJ, Liang S, Fuhrman S, Seilhamer JJ, and Somogyi R (2000) Gene Expression Microarray Data Analysis for Toxicology Profiling. Annals of the New York Academy of Sciences, in press

Dempster, A.P., Laird, N.M. and Rubin D.B. (1977) Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society B 39: 1-38.

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(5338):680-686.

D'haeseleer, P. and Fuhrman, S. (2000) Gene network inference using a linear, additive regulation model. Submitted to Bioinformatics.

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 4:41-52. http://www.smi.stanford.edu/projects/helix/psb99/Dhaeseleer.pdf

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(25): 14863-14868.

Erb, R.S. and Michaels, G.S. (1999) Sensitivity of biological models to errors in parameter estimates. Pacific Symposium on Biocomputing 4:53-64. http://www.smi.stanford.edu/projects/helix/psb99/Erb.pdf

Ewing R.M., Kahla A.B., Poirot O., Lopez F., Audic S. and Claverie J.M. (1999) Large-scale statistical analyses of rice ESTs reveal correlated patterns of gene expression. Genome Res 9: 950-9.

Felsenstein, J. (1993) PHYLIP (Phylogeny Inference Package), version 3.5c, distributed by the author, Department of Genetics, University of Washington, Seattle.

Ferea T.L., Botstein D., Brown P.O. and Rosenzweig R.F. (1999) Systematic changes in gene expression patterns following adaptive evolution in yeast. Proc Natl Acad Sci USA, 96:17 9721-6.

Fisher, R.A. In Kendall, M.G. and Stuart, A. (1969), The advanced theory of statistics, Vol. 1, 3rd ed. Hafner Press, New York, page 391.

Franklin, G.F., Powell, J.D. and Emami- Naeini, A. (1994) Feedback control of dynamic systems, 3rd Ed. Addison-Wesley.

Fuhrman S, Cunningham MJ, Wen X, Zweiger G, Seilhamer JJ, and Somogyi R. (2000) The Application of Shannon Entropy in the Identification of Putative Drug Targets. Biosystems, in press

Fuhrman S, D'haeseleer P, and Somogyi R (1999) Tracing Genetic Information Flow from Gene Expression to Pathways and Molecular Networks. 1999 Society for Neuroscience Short Course Syllabus, DNA Microarrays: The New Frontier in Gene Discovery and Gene Expression Analysis

Glass, L. (1975) Combinatorial and topological methods in nonlinear chemical kinetics. J. Chem. Phys., 63, 1325-1335.

Glass, L. (1978) Stable oscillations in mathematical models of biological control systems. J. Math. Biol., 6,207-223.

Golub T.R., Slonim D.K., Tamayo P., Huard C., Gaasenbeek M., Mesirov J.P., Coller H., Loh M.L., Downing J.R., Caligiuri M.A., Bloomfield C.D. and Lander E.S. (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science, 286, 531-537.

Goss, P.J. and Peccoud, J. (1998) Quantitative modeling of stochastic systems in molecular biology by using stochastic Petri nets. Proc Natl Acad Sci USA, 95, 6750-6755.

Han, E.-H., Karypis, G., Kumar, V. and Mobasher, B. (1997) Clustering in a high-dimensional space using hypergraph models. Technical Report # 97-019, Department of Computer Science, University of Minnesota. http://www.cs.umn.edu/tech_reports/1997/TR_97-019__Clustering_Based_on_Association_Rule_Hypergraphs.html

Hertz, J. (1998) Statistical issues in reverse engineering of genetic networks. Poster for Pacific Symposium on Biocomputing. http://www.nordita.dk/~hertz/papers/dgshort.ps.gz.

Heyer, L. J. Semyon Kruglyak and Shibu Yooseph. (1999) Exploring Expression Data: Identification and Analysis of Coexpressed Genes. Genome Research Vol. 9, Issue 11, 1106-1115.

Holstege, F.C.P., Jennings, E.G., Wyrick, J.J., Lee, T.I., Hengartner, C.J., Green, M.R., Golub, T.R., Lander, E.S. and Young, R.A. (1998a) Dissecting the regulatory circuitry of a eukaryotic genome. Cell, 95, 717-728.

Holstege, F.C.P., Jennings, E.G., Wyrick, J.J., Lee, T.I., Hengartner, C.J., Green, M.R., Golub, T.R., Lander, E.S. and Young, R.A. (1998b) Genome-Wide Expression Page. http://web.wi.mit.edu/young/expression/

Huang S. (1999) Gene expression profiling, genetic networks, and cellular states: an integrating concept for tumorigenesis and drug discovery. J Mol Med, 77:6 469-80.

Ideker, T.E., Thorsson, V., and Karp, R.M. (2000) Discovery of Regulatory Interactions Through Perturbation: Inference and Experimental Design. Pacific Symposium on Biocomputing, in press.

Jain A. K. and Dubes R. C. (1988) Algorithms for Clustering Data, Prentice Hall, Englewood Cliffs, NJ.

Kauffman, S.A. (1969) Metabolic stability and epigenesis in randomly connected nets. J Theoret. Biol., 22, 437-467.

Kauffman, S.A. (1993). The Origins of Order, Self-Organization and Selection in Evolution, Oxford University Press

Kaufman L. and Rousseeuw P.J. (1990) Finding groups in data: An introduction to cluster analysis. John Wiley & Sons, New York.

Koiran, P. and Sontag, E.D. (1998) Vapnik-Chervonenkis dimension of recurrent neural networks. Discrete Applied Math, 86, 63-79.

Lance, G. N. and Williams, W. T. (1966) A general theory of classificatory sorting strategies: 1. Hierarchical systems, Computer J. 9, 373-380.

Liang, S., Fuhrman, S. and Somogyi, R. (1998) REVEAL, a general reverse engineering algorithm for inference of genetic network architectures. Pacific Symposium on Biocomputing 3:18-29. http://www.smi.stanford.edu/projects/helix/psb98/liang.pdf

MacQueen J, (1967) Some methods for classification and analysis of multivariate observation. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability.* Vol. I. Ed. by L. M. Le Cam and J. Nyeman. University of California Press.

Mahalanobis, P.C. (1936). On the generalized distance in statistics. Proceedings of the National Institute of Sciences of India 12, 49-55.

Marnellos, G. and Mjolsness, E. (1998) A Gene Network Approach to Modeling Early Neurogenesis in Drosophila. Pacific Symposium on Biocomputing 3:30-41. http://www.smi.stanford.edu/projects/helix/psb98/marnellos.pdf

Massart D.L. and Kaufman L. (1983) The Interpretation of Analytical Chemical Data by the Use of Cluster Analysis. John Wiley & Sons. New York.

Matsuno, H., Doi, A. and Nagasaki, M. (2000) Hybrid Petri Net representation of genetic regulatory network. Pacific Symposium on Biocomputing, in press.

Mendes, P. (1999) Metabolic Simulation as an Aid in Understanding Gene Expression Data. In Workshop on Computation of Biochemical Pathways and Genetic Networks (Bornberg-Bauer, E., De Beuckelaer, A., Kummer, U., Rost, U. eds), Logos Verlag, Berlin, pp. 27-33.

Michaels G, Carr DB, Wen X, Fuhrman S, Askenazi M, Somogyi R (1998) Cluster Analysis and Data Visualization of Large-Scale Gene Expression Data. Pacific Symposium on Biocomputing 3:42-53
http://www.smi.stanford.edu/projects/helix/psb98/michaels.pdf

Miklos G.L. and Rubin G.M. (1996) The role of the genome project in determining gene function: insights from model organisms. Cell, 86:4 521-9.

Mjolsness, E., Castaño, R. and Gray, A. (1999a) Multi-Parent Clustering Algorithms for Large-Scale Gene Expression Analysis. Technical Report JPL-ICTR-99-5, Jet Propulsion Laboratory Section 367.
http://www-aig.jpl.nasa.gov/public/mls/papers/emj/multiparentPreprint.pdf

Mjolsness, E., Mann, T., Castaño, R. and Wold, B. (1999b) From Co-expression to Co-regulation: An Approach to Inferring Transcriptional Regulation among Gene Classes from Large-Scale Expression Data. Technical Report JPL-ICTR-99-4, Jet Propulsion Laboratory Section 365. http://www-aig.jpl.nasa.gov/public/mls/papers/emj/GRN99prprnt.pdf

Mjolsness, E., Sharp, D. H. and Reinitz, J. (1991) A connectionist model of development. J. Theor. Biol., 152, 429--454.

Mounts W.M. and Liebman, M.N. (1997) Application of Petri Nets and Stochastic Activity Nets to Modeling Biological Pathways and Processes. International Journal in Computer Simulation

Niehrs, C. and Pollet, N. (1999) Synexpression groups in eukaryotes. Nature 402: 483–487

Pearlmutter, B.A. (1995) Gradient calculations for dynamic recurrent neural networks: a survey. IEEE Transactions on Neural Networks, 6, 1212-1228.

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., Lashkari D., Shalon D., Brown P.O. and Botstein D. (1999)Distinctive gene expression patterns in human mammary epithelial cells and breast cancers. Proc Natl Acad Sci USA, 96:16 9212-9217.

Ptashne, M. (1992), A genetic Switch. Cell Press & Blackwell scientific publications.

Roth P., Hughes J.D., Estep P.W. and Church G.M. (1998) Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation. Nature Biotechnology, 16, 939-945.

Savageau, M.A. (1995) Power-law formalism: a canonical nonlinear approach to modeling and analysis. In Proceedings of the World Congress of Nonlinear Analysts '92. pp. 3323-3334.

Savageau, M.A. (1998) Rules for the Evolution of Gene Circuitry. Pacific Symposium on Biocomputing 3:54-65.
http://www.smi.stanford.edu/projects/helix/psb98/savageau.pdf

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

Sokal, R.R. and Michener, C. D. (1958) Univ. Kans. Sci. Bull. 38, 1409-1438.

Somogyi R & Sniegoski C (1996). Modeling the Complexity of Genetic Networks. Complexity 1(6):45-63

Somogyi R, Fuhrman S, Askenazi M, Wuensche A (1997) The Gene Expression Matrix: Towards the Extraction of Genetic Network Architectures. Nonlinear Analysis, Proc. of Second World Cong. of Nonlinear Analysts (WCNA96) 30(3):1815-1824
http://rsb.info.nih.gov/mol-physiol/reprints/WCNA.pdf

Somogyi R, Wen X, Ma W, Barker JL (1995) Developmental kinetics of GAD family mRNAs parallel neurogenesis in the rat spinal cord. J Neurosci 15:2575-2591

Somogyi, R. (1999) Making Sense of Gene Expression Data. Pharmainformatics: A Trends Guide (Trends Supplement) pp. 17-24

Sontag, P. (1997) Shattering all sets of k points in general position requires (k-1)/2 parameters. Neural Computation, 9, 337-348.

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: 3273-97.

Szallasi Z (1999) Genetic Network Analysis in Light of Massively Parallel Biological Data Acquisition. Pacific Symposium on Biocomputing 4:5-16. http://www.smi.stanford.edu/projects/helix/psb99/Szallasi.pdf

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.

Tavazoie S., Hughes J.D., Campbell M.J., Cho R.J. and Church G.M. (1999) Systematic determination of genetic network architecture. Nat Genet 22:281-285

Thieffry, D. and Thomas, R. (1998) Qualitative analysis of gene networks. Pacific Symposium on Biocomputing 3:77-88. http://www.smi.stanford.edu/projects/helix/psb98/thieffry.pdf

Thomas, R. (1991) Regulatory networks seen as asynchronous automata: a logical description. J. Theor. Biol., 153, 1-23.

Tominaga, D. and Okamoto, M. (1998) Design of Canonical Model Describing Complex Nonlinear Dynamics. In Yoshida, T. and Shioya, S. (eds.), Computer Applications in Biotechnology 1998, Elsevier Science, pp. 85-90.

Tominaga, D., Okamoto, M., Kami, Y., Watanabe, S. and Eguchi, Y. (1999) Nonlinear Numerical Optimization Technique Based on a Genetic Algorithm, http:/w.bioinfo.de/isb/gcb99/talks/tominaga/

van Helden J., Andre B. and Collado-Vides J. (1998) Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. J Mol Biol, 281, 827-842.

Velculescu V.E., Zhang, L., Zhou, W., Vogelstein, J., Basrai, M.A., Bassett. D.E.Jr., Hieter, P., Vogelstein, B., and Kinzler, K.W. (1997) Characterization of the yeast transcriptome. Cell, 88, 243-251.

Wahde, M. and Hertz, J. (1999) Course-grained reverse engineering of genetic regulatory networks. Proceedings of Information Processing in Cells and Tissues (IPCAT) '99.

Walker M.G., Volkmuth W., Sprinzak E., Hodgson D., Klingler T. (1999) Prediction of gene function by genome-scale expression analysis: prostate-cancer associated genes. Genome Res 9:1198-1203.

Weaver, D. C., Workman, C. T. and Stormo, G. D. (1999) Modeling regulatory networks with weight matrices. Pacific Symposium on Biocomputing 4:112-123. http://www.smi.stanford.edu/projects/helix/psb99/Weaver.pdf

Weinstein J.N., Myers T.G., O'Connor P.M., Friend S.H., Fornace A.J. Jr, Kohn K.W., Fojo T., Bates S.E., Rubinstein L.V., Anderson N.L., Buolamwini J.K., van Osdol W.W., Monks A.P., Scudiero D.A., Sausville E.A., Zaharevitz D.W., Bunow B., Viswanadhan V.N., Johnson G.S.,

Wittes R.E. and Paull K.D. (1997) An information-intensive approach to the molecular pharmacology of cancer. Science 275:343-349.

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, 95:1 334-9.

Wolfsberg T.G., Gabrielian A.E., Campbell M.J., Cho R.J., Spouge J.L., and Landsman D. (1999) Candidate regulatory sequence elements for cell cycle-dependent transcription in saccharomyces cerevisiae. Genome Research, 9, 775-792.

Wuensche, A. (1993) Discrete Dynamics Lab (DDLab) http://www.santafe.edu/~wuensch/ddlab.html

Wuensche, A. (1998) Genomic Regulation Modeled as a Network with Basins of Attraction. Pacific Symposium on Biocomputing 3:89-102. http://www.smi.stanford.edu/projects/helix/psb98/wuensche.pdf

Yuh C.H., Bolouri H. and Davidson E.H. (1998) Genomic cis-regulatory logic: experimental and computational analysis of a sea urchin gene. Science, 279 1896-902.

Zhu J. and Zhang M.Q. (2000) Cluster, function and promoter: analysis of yeast expression array. Pacific Symposium on Biocomputing, in press.

Zweiger, G. (1999) Knowledge discovery in gene-expression-microarray data: mining the information output of the genome. Trends Biotech, 17: 429-436

Figure Legends

Fig 1 Dynamics of Boolean model of a genetic network illustrated using the DDLAB software (Wuensche, 1993, 1999). Wiring, trajectory and basin of attraction. Top panel, wiring diagram of binary gene interactions: The network consists of hypothetical regulatory gene modules (Σ1, Σ3, Σ6) and dependent modules of structural genes (S, U, T - share identical wiring and rules). Right inset, trajectory: As determined by the wiring and rules, the network passes through a series of states from a given starting state, finally arriving at a repeating attractor pattern, a 6-state cycle in this case (dark grey = ON, light gray = OFF; time points numbered at left; modules S, U and T have been collapsed into a single element for simplicity). Bottom panel, basin of attraction: The states of the trajectory shown in the inset are displayed as a series of connected points (labeled by time points) arriving in a cyclic graph resembling the attractor. The additional nodes in the graph resemble other states which also lead to the same attractor, therefore the term "basin of attraction" – all states in this graph merge into a single attractor. (Somogyi and Sniegoski, 1996)

Fig. 2 Comparison of clustering methods on hippocampal development gene expression data. Genes were grouped into 8 clusters with both methods. Note the relative positions of the genes sharing a Krox-24 transcriptional regulatory element among the clusters. A) Gene expression patterns were normalized to their respective minima and maxima, and clustered using an agglomerative algorithm. The cluster bifurcation pattern is shown on the left; cluster boundaries were drawn at the depth shaded in grey (left), based on the 20 level cluster identifier that captures the branching pattern for each gene within the dendrogram (right). B) Gene expression patterns were normalized to their respective maxima, and clustered using a numerical k-means algorithm. (Fuhrman *et al.*, 2000.)

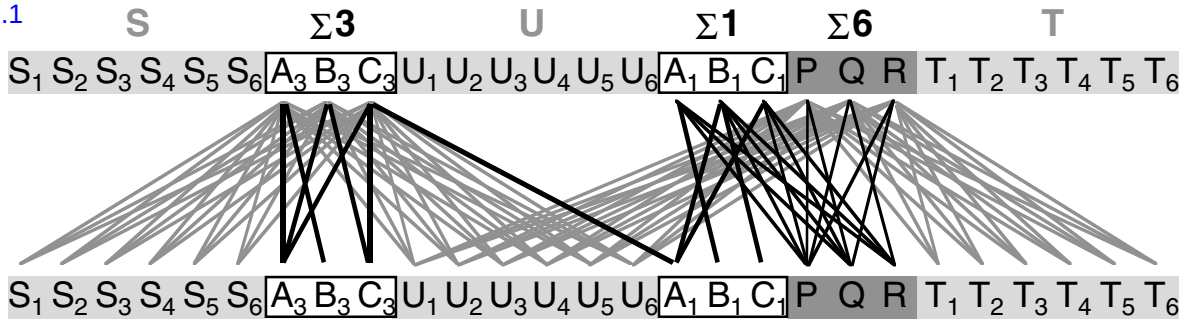
Fig. 3 Gene expression clusters reflect gene families and pathways. Neurotransmitter receptors follow particular expression waveforms according to ligand and functional class. Waves 1 to 4 correspond to major expression clusters found in rat spinal cord development (Wen *et al.*, 1998). Note that the early expression waves 1 and 2 are dominated by ACh and GABA receptors, and by receptor ion-channels in general. Each line represents a gene, and can be traced by following its reflection at each node. (Agnew, 1998.)

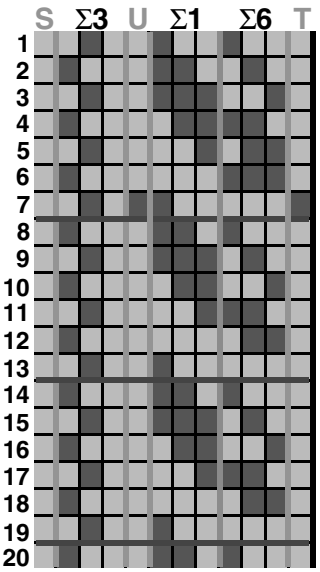
Fig. 4 Dependence of Boolean network reverse engineering algorithm on depth of training data. The probability of finding an incorrect solution is graphed vs. the number of state transition pairs used as input for the algorithm for a N=50 element network. More training data is required for networks of k=3 inputs per gene than for networks of lower connectivity to minimize incorrect solutions. However, only a small fraction e.g. 80 state transition pairs from a total of 2^50=1.13x10^15 is required to obtain reliable results. (Liang *et al.*, 1998.)

Fig. 5 Continuous valued reverse engineering of a CNS genetic network using a linear additive method. A) Experimental gene expression data (circles; development and injury), and simulation using a linear model (lines). The model faithfully reproduces the time series of the training data sets. Dotted line: spinal cord, starting 11 days before birth. Solid line: hippocampus development, starting 7 days before birth. Dashed line: hippocampus kainate injury, starting at postnatal day 25. B) Hypothetical gene interaction

diagram for the GABA signaling family inferred from developmental gene expression data (spinal cord and hippocampus data). While individual proposed interactions have not yet been experimentally verified, the high predicted connectivity within this gene family appears biologically plausible. The positive feedback interaction of the GAD species has been proposed independently in another study (Somogyi *et al.*, 1995). Solid lines correspond to positive interactions, broken lines suggest inhibitory relationships. (D'haeseleer *et al.*, 1999.)

Fig.1


S
Σ3
U
Σ1
Σ6
T
S1 S2 S3 S4 S5 S6 A3 B3 C3 U1 U2 U3 U4 U5 U6 A1 B1 C1 P Q R T1 T2 T3 T4 T5 T6
S1 S2 S3 S4 S5 S6 A3 B3 C3 U1 U2 U3 U4 U5 U6 A1 B1 C1 P Q R T1 T2 T3 T4 T5 T6


S
Σ3
U
Σ1
Σ6
T
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

1
2
3
4
5
6
7
8
9
10
11
12
13

Fig.2A

Hippocampal Development
Min/Max Normalization - Agglomerative Clustering

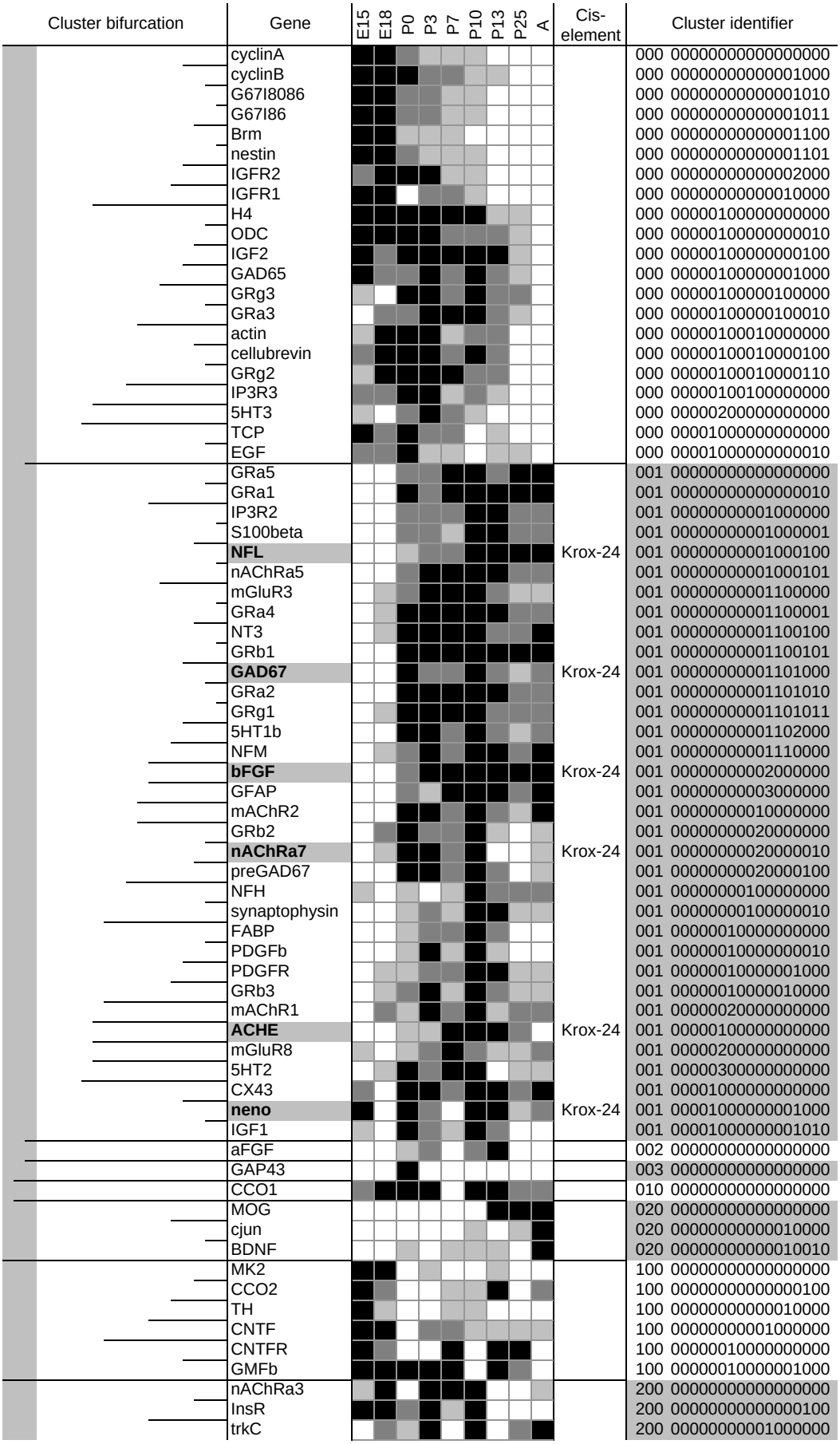


Fig.2B

Hippocampal Development
Max Normalization - K-means Clustering

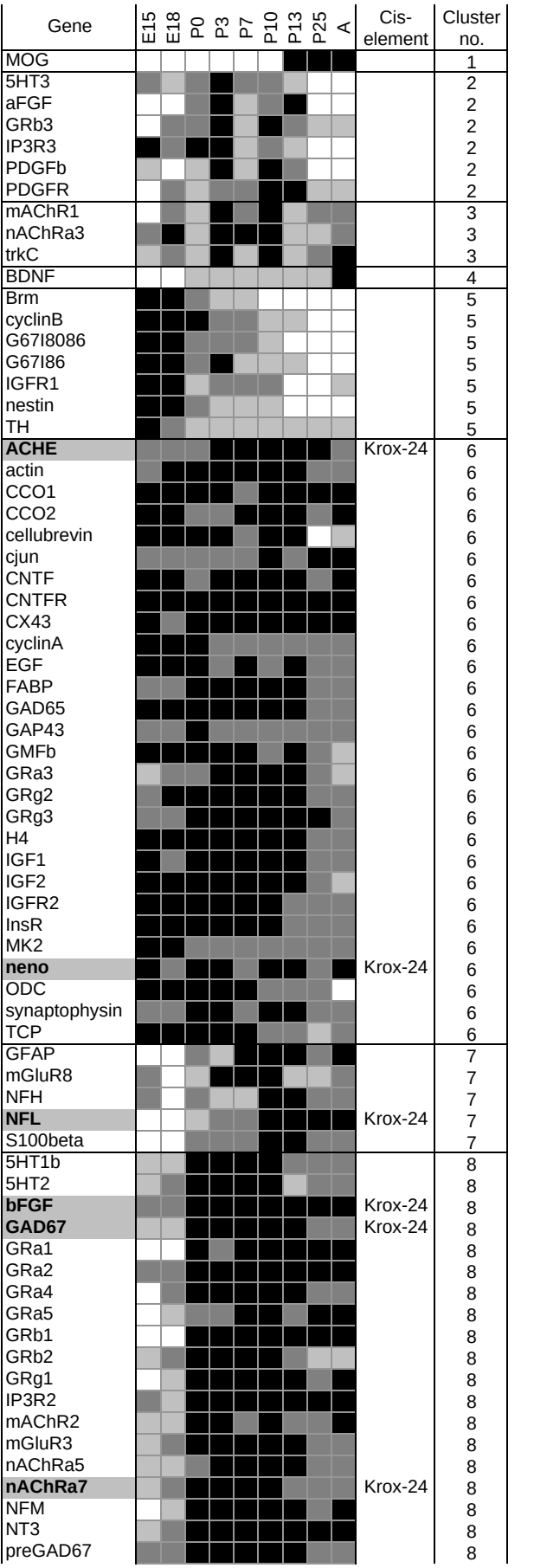


Fig.3

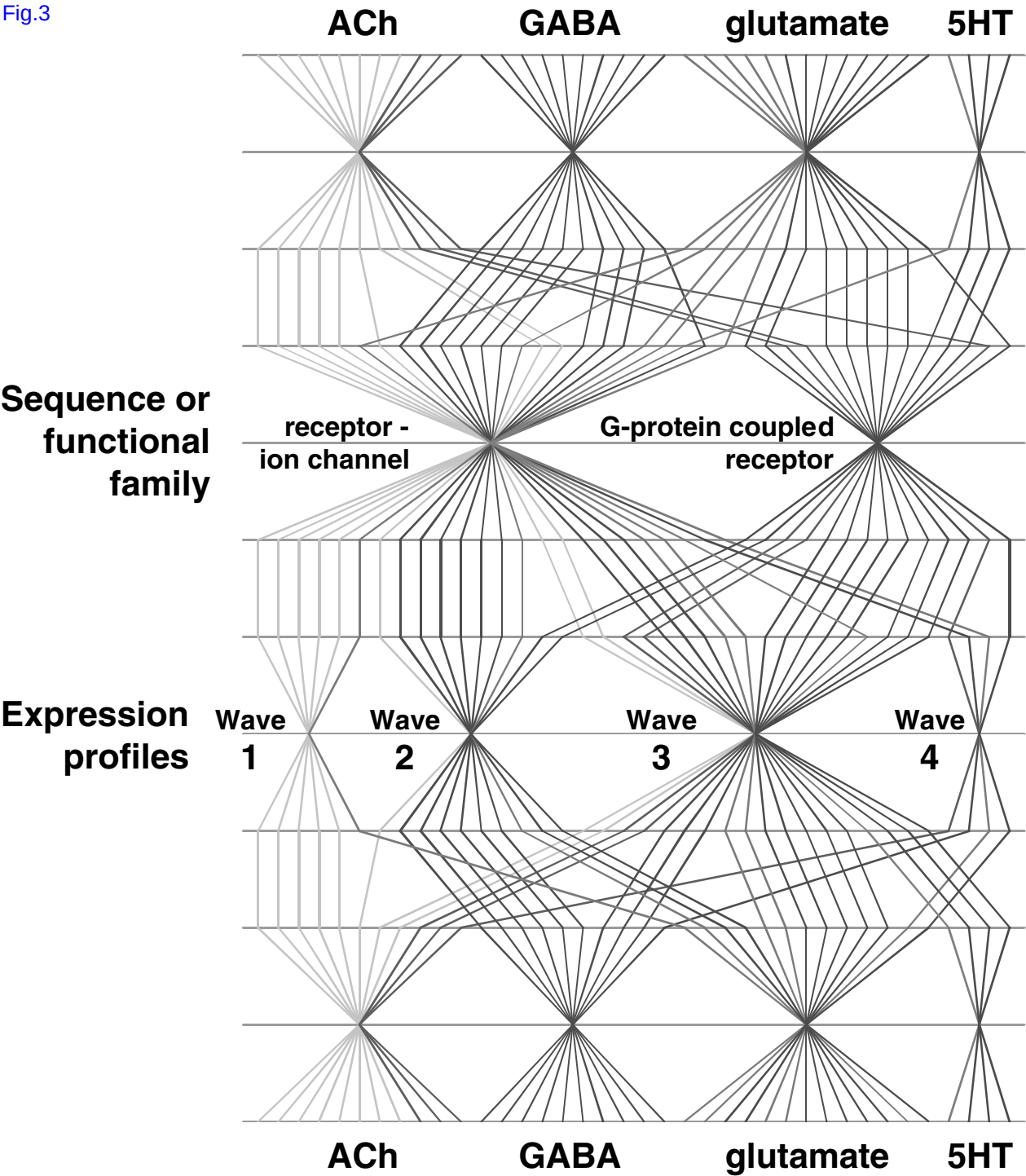

ACh
GABA
glutamate
5HT
Sequence or functional family
receptor - ion channel
G-protein coupled receptor
Expression profiles
Wave 1
Wave 2
Wave 3
Wave 4
ACh
GABA
glutamate
5HT

Fig.4

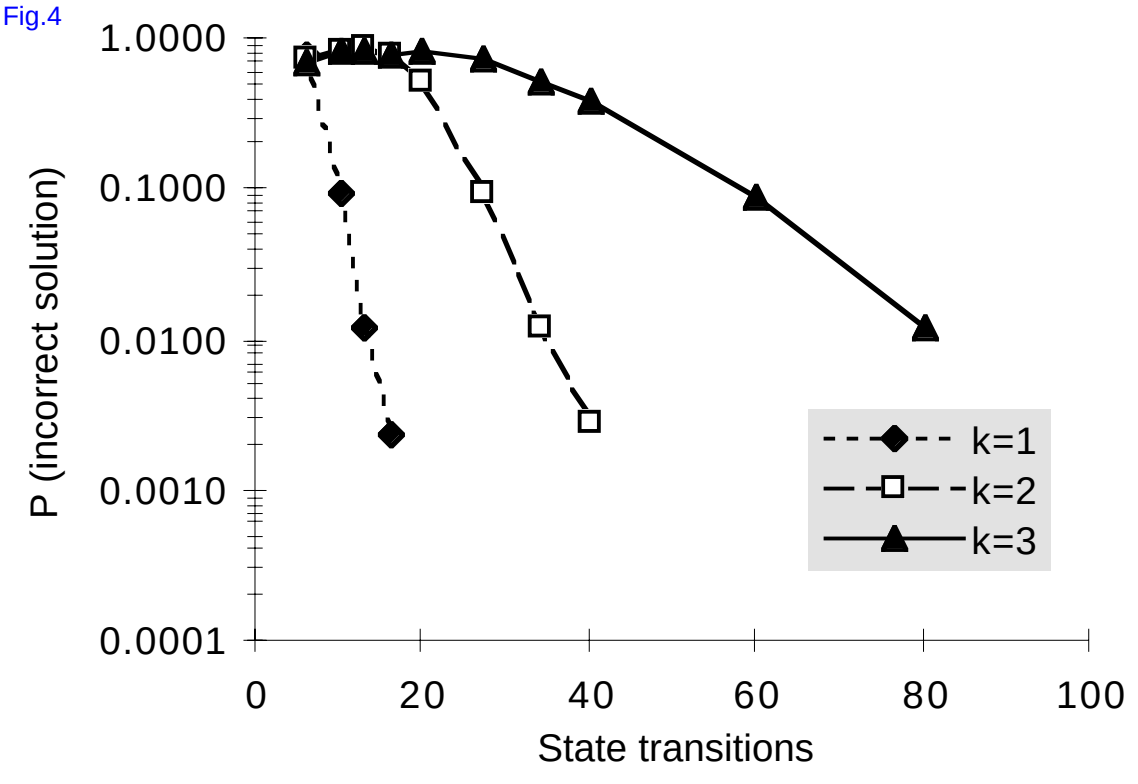

1.0000
0.1000
0.0100
0.0010
0.0001
P (incorrect solution)
0
20
40
60
80
100
State transitions
k=1
k=2
k=3

Fig.5A

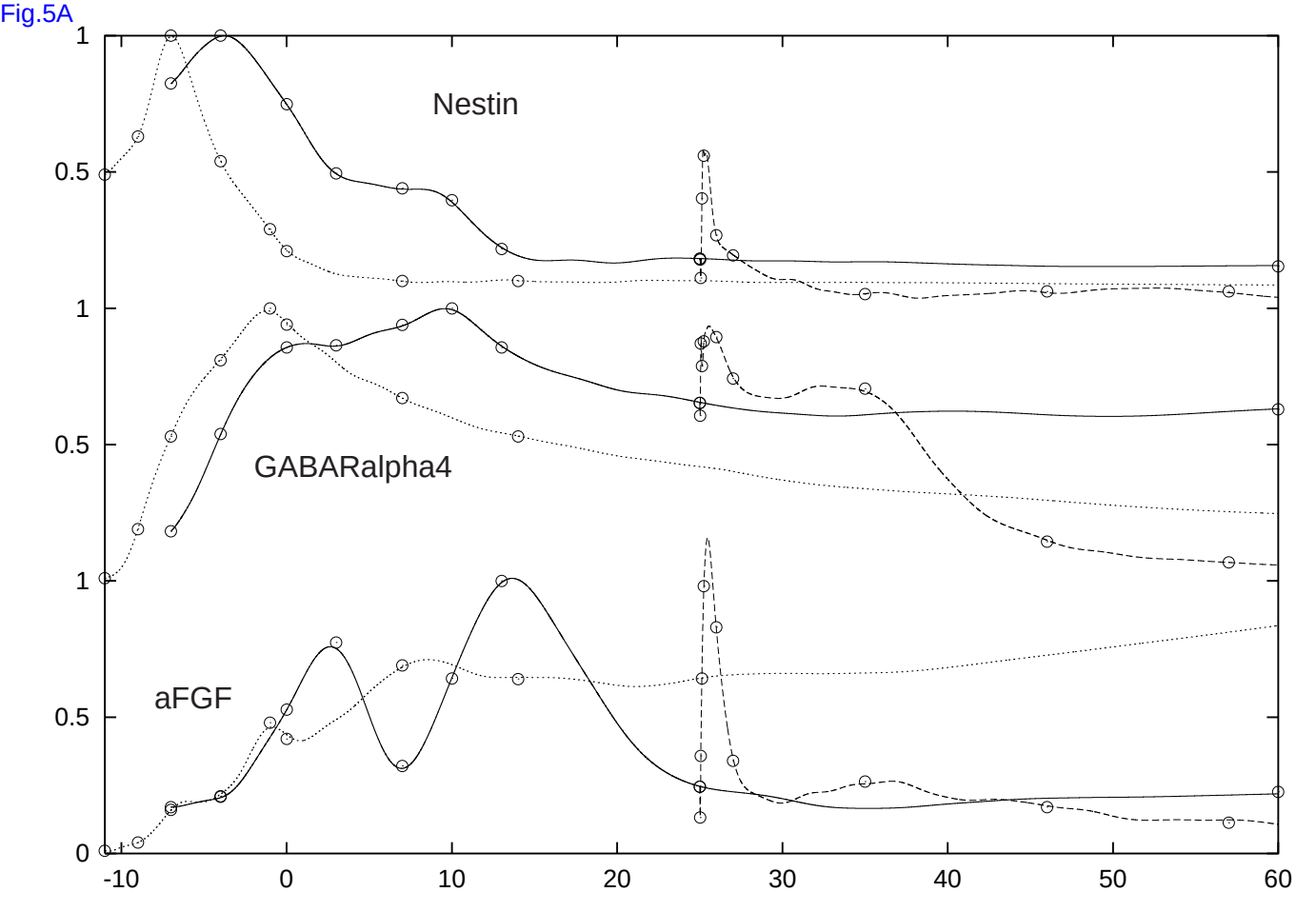
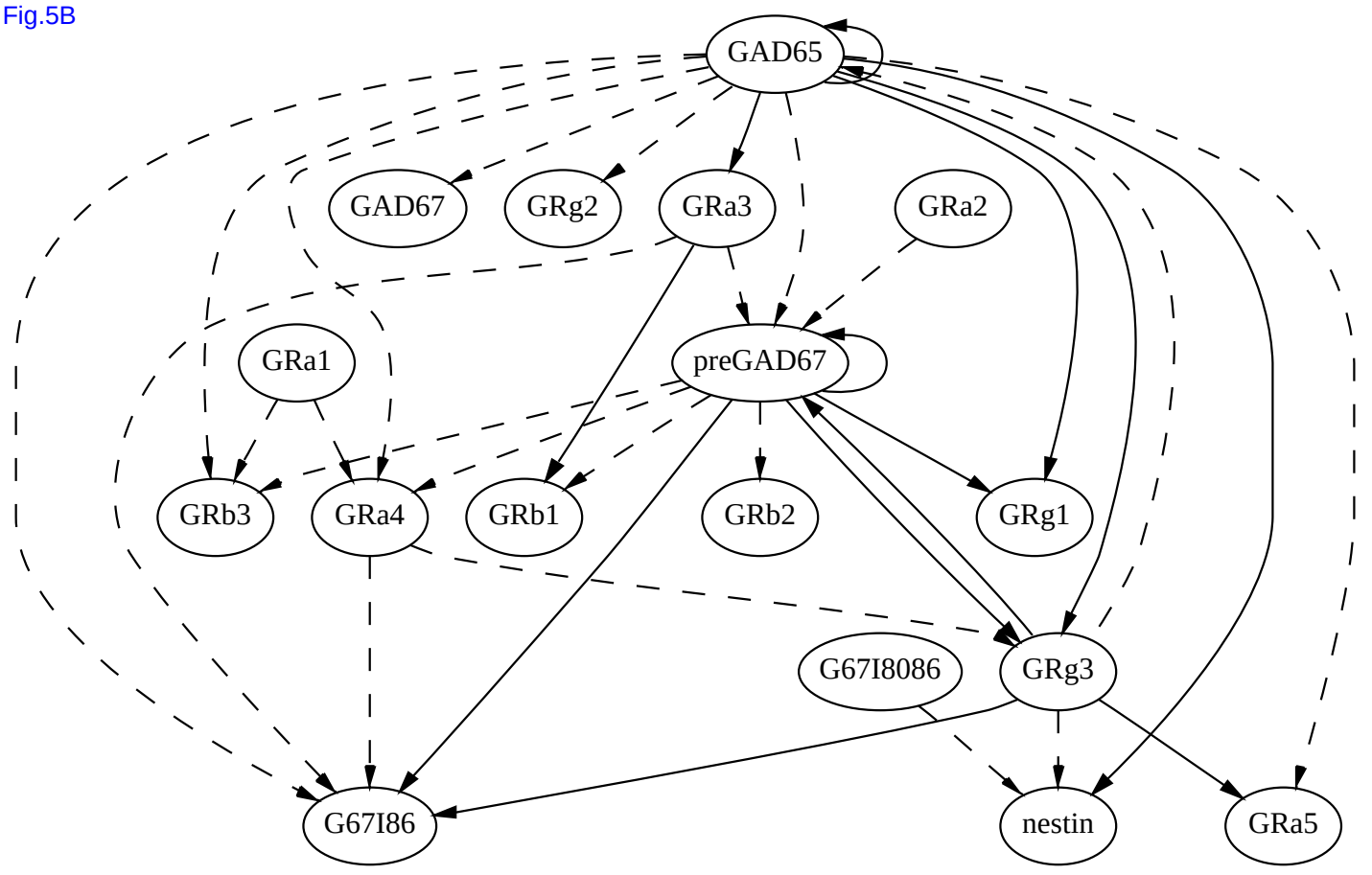


Fig.5B


GAD65
GAD67
GRg2
GRa3
GRa2
GRa1
preGAD67
GRb3
GRa4
GRb1
GRb2
GRg1
G67I8086
GRg3
G67I86
nestin
GRa5