Hindawi Publishing Corporation
EURASIP Journal on Bioinformatics and Systems Biology
Volume 2009, Article ID 878013, 13 pages
doi:10.1155/2009/878013

Research Article

Selection of Statistical Thresholds in Graphical Models

Anthony Almudevar

Department of Biostatistics and Computational Biology, University of Rochester, 601 Elmwood Avenue, Box 630, Rochester, NY 14642, USA

Correspondence should be addressed to Anthony Almudevar, anthony_almudevar@urmc.rochester.edu

Received 11 June 2009; Revised 15 September 2009; Accepted 20 November 2009

Recommended by Erchin Serpedin

Reconstruction of gene regulatory networks based on experimental data usually relies on statistical evidence, necessitating the choice of a statistical threshold which defines a significant biological effect. Approaches to this problem found in the literature range from rigorous multiple testing procedures to ad hoc P-value cut-off points. However, when the data implies graphical structure, it should be possible to exploit this feature in the threshold selection process. In this article we propose a procedure based on this principle. Using coding theory we devise a measure of graphical structure, for example, highly connected nodes or chain structure. The measure for a particular graph can be compared to that of a random graph and structure inferred on that basis. By varying the statistical threshold the maximum deviation from random structure can be estimated, and the threshold is then chosen on that basis. A global test for graph structure follows naturally.

1. Introduction

The reconstruction of gene regulatory networks using gene expression data has become an important computational tool in systems biology. A relationship among a set of genes can be established either by measuring the effect of the experimental perturbation of one or more selected genes on the remaining genes or from the use of measures of coexpression from observational data. The data is then incorporated into a suitable mathematical model of gene regulation. Such models vary in level of detail, but most are based on a *gene graph*, in which nodes represent individual genes, while edges between nodes indicate a regulatory relationship.

One important issue that arises is the variability of the data due to biological and technological sources. This leads to imperfect resolution of gene relationships and the need for principled statistical methodology with which to assign statistical significance to any inferred feature.

In many models, the existence or absence of an edge in the gene graph is resolved by a statistical hypothesis test. A natural first step is the ranking of potential edges based on the strength of the statistical evidence for the existence of the implied regulatory relationship. The intuitive approach is to construct a graph consisting of the highest ranking edges, defined by a P-value threshold. The choice of threshold may be ad hoc, typically a conservative significance level such as 0.01. A more rigorous approach is to select the threshold using principles of multiple hypothesis testing (see, e.g., [1]), which may yield an estimate of the error rates of edge classification.

There is a fundamental drawback to this approach, in that the lack of statistical evidence of a regulatory relationship may be as much a consequence of small sample size as of biological fact. Under this scenario, we note that selection of a P-value threshold $P^{\text{thr}} = p$ generates a graph of, say, N_p edges, with N_p increasing in p. Under a null hypothesis of no regulatory structure, P-values are randomly ranked, hence edges will be distributed uniformly, whereas the edges of a true regulatory network will posses structure unlikely to arise by chance. Formulated in terms of statistical hypothesis tests, it should be possible to exploit this evidence in order to make a more informative choice of P^{thr}. This article proposes a method to accomplish this goal.

2. Problem Formulation

The proposed algorithm is intended to be part of the following type of analysis based on gene expression data for N genes.

(S1) Collect gene perturbation data from N experiments coupled with control data. For simplicity, assume that experiment j is a single perturbation of gene j.

(S2) Construct an $N \times N$ matrix D, in which element D_{ij} is a measure of the statistical evidence that perturbation of gene j changes the expression level of gene i. This defines *data matrix D*.

(S3) Use the methodology proposed here to determine a P-value threshold P^{thr}.

(S4) Conclude that perturbing gene j causes a change in gene i if and only if $D_{ij} \leq P^{\text{thr}}$.

(S5) The perturbation responses implied by step (S4) may now be used to fit, for example, a Boolean network as in [2–4].

We assume in (S1)-(S2) that the matrix D is *balanced* in the sense that row i and column i refer to the same gene. The methods proposed here do not rely on this assumption, although a formal treatment of the general case will be deferred to future work. Typically, D_{ij} will be a P-value from a two-sample hypothesis test comparing the expression levels of genes i obtained from cells subject to an experimental perturbation of gene j to those obtained from control (unperturbed) cells. In this case small values of D_{ij} are interpreted as evidence for the existence of directed edge $j \to i$. We adopt this convention below.

It will be useful to introduce some definitions of directed gene graphs (see [5]). We say gene *a regulates* gene b if the gene expression level of a directly influences that of gene b. This is distinct from *transitive regulation*, in which expression levels of one gene affect another only through intermediary genes. For example, if a regulates b and b regulates c, then a and c are in a transitive regulatory relationship (that would not exist without b). In an *accessibility graph* edge $a \to b$ exists if a regulates or transitively regulates b. In contrast, in an *adjacency graph* an edge from a to b exists only if a regulates b. An adjacency graph can be constructed as a parsimonious representation of an accessibility graph ([5–7]). It should be noted that a regulatory relationship implied by a graphical model is relative only to those genes included and does not rule out the existence of intermediary genes not observed.

Step (S3) will be based on the following idea. Data matrix D can generate an estimated accessibility graph $G^{\text{acc}}(D, t)$ by constructing an edge $j \to i$ if and only if $D_{ij} \leq t$. While this is a crude form of network model, we may still expect $G^{\text{acc}}(D, t)$ to contain interesting and measurable structure, provided that t is efficiently chosen. Our intention is to use this structure to guide the choice of P^{thr}. The set of edges in $G^{\text{acc}}(D, P^{\text{thr}})$ is then used to construct a more detailed model, as in step (S5).

Consider a hierarchical sequence of graphs $G_1, G_2, \ldots$ obtained by successively adding edges in increasing order of their P-values. If the data is dominated by statistical noise, we may expect elements of the sequence to consist of random graphs generated by uniform distributions of a fixed number of edges, known as the Erdös-Renyi random graph model (see, e.g., [8]). Actual cellular networks are believed to conform more closely to the power-law model, where the likelihood that a randomly chosen node has d interactions is proportional to $d^{-\tau}$ where $1.5 < \tau < 2.5$ (see [9]). We may also expect more chain structure (longer paths) than would occur by chance. This would allow statistical identification of cellular network structure, which can provide auxiliary information for the selection of P^{thr} beyond what is normally available using standard multiple hypothesis testing methods.

2.1. Conditional Hypothesis Tests. The required elements of our procedure are (i) a data matrix D (steps (S1)-(S2)), (ii) a graph score λ which is sensitive to general graphical structure, and (iii) a distributional model $\mathscr{P}$ for generating graphs under the null hypothesis of no regulatory relationships. In the following development smaller values of λ imply greater structure.

2.1.1. Notational Conventions. We will adopt the following notation. Assume that N is fixed. First, let $\mathscr{S}_1$ be the set of all increasing sequences of positive integers $\tilde{s} = (s_1, \ldots, s_m)$ for which $s_m = N^2$. Then let $\mathcal{V}$ be the set of all N-dimensional vectors of nonnegative integers (which we refer to as *count vectors*). Let $\overline{v}$ denote the sum of the elements of any $v \in \mathcal{V}$. A sequence of vectors $\tilde{v} = (v_1, \ldots, v_m)$ from $\mathcal{V}$, written $v_i = (v_{i1}, \ldots, v_{iN})$, is *increasing* if $v_{ij} \leq v_{(i+1)j}$ for all $1 \leq i \leq m-1$, $1 \leq j \leq N$, and if $(\overline{v}_1, \ldots, \overline{v}_m) \in \mathscr{S}_1$. Let $\mathscr{S}_2$ be the set of all such increasing count vector sequences.

The set of all order N labelled graphs is denoted by $\mathscr{G}$. Let $\mathscr{G}_k \subset \mathscr{G}$ be the subset of graphs with k edges, and for any $v \in \mathcal{V}$ let $\mathscr{G}_v \subset \mathscr{G}$ be the subset of graphs containing v_j edges with parent j. Let $\mathscr{G}_k^{-E} \subset \mathscr{G}_k$, $\mathscr{G}_v^{-E} \subset \mathscr{G}_v$ be the respective subsets which exclude all edges from edge set E. A sequence of graphs $g_1, \ldots, g_m$ from $\mathscr{G}$ is called *increasing* if g_j is a subgraph of g_{j+1}, $1 \leq j \leq m-1$. We say that an increasing graph sequence $(g_1, \ldots, g_m)$ *conforms* to index sequence $\tilde{s} \in \mathscr{S}_1$ and set E if $g_i \in \mathscr{G}_{s_i}^{-E}$, for all $1 \leq i \leq m$.

2.1.2. Data Matrix. Suppose that we are given an $N \times N$ data matrix D of P-values as described in (S1)–(S5). An edge $j \to i$ may be ruled out by setting $D_{ij} = 1$. We will refer to such an edge as a *void* edge, with corresponding *void* matrix element. For example, this should occur when the data cannot predict self-regulation implied by edges $i \to i$. A missing value in D may also represent a void edge.

Let $t_1, t_2, \ldots, t_m$ be the sequence of all unique values represented as elements of D. The value of m varies according to the number of ties as well as the number of void elements. We need to define a system of counts generated by D.

Set

$$\begin{aligned} v_{ij} &= \sum_{k=1}^{N} I\left\{D_{kj} \le t_i\right\}, \quad i = 1,\ldots,m,\ j = 1,\ldots,N, \\ s_k &= \sum_{i=1}^{N}\sum_{j=1}^{N} I\left\{D_{ij} \le t_k\right\}, \quad k = 1,\ldots,m, \end{aligned} \tag{1}$$

where $I\{E\} = 1$ when event E occurs and is zero otherwise. We then define the sequence $G(D) = (G_1(D),\ldots,G_m(D))$, where $G_k(D) = G^{\text{acc}}(D, t_k) \in \mathcal{G}_{s_k}^{-E}$. This sequence is increasing, and consecutive graphs may increase by more than one edge. We refer to the system of counts $S_1(D) = (s_1,\ldots,s_m)$ and $S_2(D) = (v_1,\ldots,v_m)$, $v_i = (v_{i1},\ldots,v_{iN})$ which can be interpreted as random objects in sample spaces $\mathcal{S}_1$, $\mathcal{S}_2$, respectively. The sequence $S_1(D)$ corresponds to the number of edges of the graphs in $G(D)$; that is, s_k is the number of edges in $G_k(D)$. Similarly, $S_2(D)$ corresponds to the number of edges decomposed by parent node in $G(D)$; that is, v_{ij} is the number of edges in $G_i(D)$ with parent node j.

2.1.3. Conditional Inference. Under the simplest null hypothesis of no regulatory structure perturbation conditions are indistinguishable from the control, in which case the P-values of D are uniformly distributed. A number of considerations then need to be made. The uniform distribution assumption depends on a correct characterization of the sampling distribution, which is often problematic in gene expression assays. In addition, when empirical methods (permutation or bootstrap methods) are used to estimate P-values, ties may result which affect graph ordering. Finally, the definition of a null model relies on the independence structure of the data, which must be carefully characterized. Conditional procedures permit the development of tests which do not depend on problematic model identification, and have been extensively used in other applications in statistical genetics.

We will now develop two null models. A conditional inference procedure is defined by data D, a composite null hypothesis H_0 concerning D, a test statistic $T(D)$, an ancillary statistic $S(D)$, such that the distribution of $T(D)$ conditional on $S(D)$ can be characterized, and is the same for all distributions described by H_0.

2.1.4. Null Model 1 (Elementwise Exchangeability). Recall that a multivariate distribution is *exchangeable* if it is invariant under any permutation of its coordinates. This includes *iid* distributions, but also those with identical marginal distributions and permutation invariant dependence structure.

For any $g \in \mathcal{G}$ let $\mathcal{G}_s^{-E}[g]$ be all graphs in $\mathcal{G}_s^{-E}$ for which g is a sub graph.

Definition 1. For void edges E and sequence $\tilde{s} = (s_1,\ldots,s_m) \in \mathcal{S}_1$, a random sequence of graphs $\tilde{h} = (h_1,\ldots,h_m)$ possesses a null distribution $\mathcal{P}_1(E,\tilde{s})$ if h_1 is uniformly distributed on $\mathcal{G}_{s_1}^{-E}$ and if h_j, conditional on $(h_1,\ldots,h_{j-1})$, is uniformly distributed on $\mathcal{G}_{s_j}^{-E}[h_{j-1}]$, for all $j = 2,\ldots,m$.

The sequence $\tilde{h}$ forms a Markov process in which h_{j+1} is obtained from h_j by adding $s_{j+1} - s_j$ edges to h_j at random, excluding E. We then define the null hypothesis:

(H_0^1) The distribution of the nonvoid elements of D is exchangeable.

This leads to the following lemma.

Lemma 1. *Under hypothesis H_0^1 the distribution of $G(D)$ conditional on $S_1(D)$ is given by $\mathcal{P}_1(E, S_1(D))$.*

Proof. Suppose $P(S_1(D) = \tilde{s}) > 0$ for some $\tilde{s} = (s_1,\ldots,s_m) \in \mathcal{S}_1$. For $k > 1$ suppose $g_0 \in \mathcal{G}_{s_{k-1}}^{-E}$ and $g \in \mathcal{G}_{s_k}^{-E}[g_0]$. We have the set equality

$$\begin{aligned} &\left\{G_k(D) = g\right\} \cap \left\{S_1(D) = \tilde{s}\right\} \\ &\quad = \left\{D_{ij} < D_{i'j'} \ \forall\, j \longrightarrow i \in g,\ j' \longrightarrow i' \notin g\right\} \cap \left\{S_1(D) = \tilde{s}\right\}. \end{aligned} \tag{2}$$

Let $g' \in \mathcal{G}_{s_k}^{-E}[g_0]$, and suppose $g \ne g'$. We may define a permutation operator T on the nonvoid elements of D such that the elements associated with g are mapped onto the elements associated with g', with element associated with g_0 mapped into themselves. This implies that the elements not associated with g are mapped into the elements not associated with g'. The quantity $S_1(D)$ is permutation invariant, and by construction $G_{k-1}(D) = g_0 = G_{k-1}(TD) = g_0$ so from (2) we have

$$\begin{aligned} &\left\{G_k(D) = g\right\} \cap \left\{G_{k-1}(D) = g_0\right\} \cap \left\{S_1(D) = \tilde{s}\right\} \\ &\quad = \left\{G_k(TD) = g'\right\} \cap \left\{G_{k-1}(TD) = g_0\right\} \cap \left\{S_1(TD) = \tilde{s}\right\}. \end{aligned} \tag{3}$$

By the exchangeability assumption D and TD have identical distributions, giving

$$\begin{aligned} &P\left(\left\{G_{k-1}(D) = g_0\right\} \cap S_1(D) = \tilde{s}\right) \\ &\quad = P\left(\left\{G_{k-1}(TD) = g_0\right\} \cap S_1(TD) = \tilde{s}\right), \\ &P\left(\left\{G_k(D) = g'\right\} \cap \left\{G_{k-1}(D) = g_0\right\} \cap \left\{S_1(D) = \tilde{s}\right\}\right) \\ &\quad = P\left(\left\{G_k(TD) = g'\right\} \cap \left\{G_{k-1}(TD) = g_0\right\} \cap \left\{S_1(TD) = \tilde{s}\right\}\right). \end{aligned} \tag{4}$$

The argument can be adapted to verify that $G_1(D)$, conditional on $\{S_1(D) = \tilde{s}\}$ is uniformly disributed on $\mathcal{G}_{s_k}^{-E}$. By combining the above equalities the proof follows. □

In the simplest case, the null hypothesis predicts uniformly distributed and independent P-values among nonvoid elements of D. In this case by Lemma 1 $G(D)$ conditioned on $S_1(D)$ has distribution $\mathcal{P}_1(E, S_1(D))$. If the marginal distributions are continuous, then the probability of ties is zero, and with probability 1 the elements of $S_1(D)$ increment by one until the void elements are reached. When distributions are discrete ties, are possible and $S_1(D)$ can be determined directly from the data. It is important to note that the actual marginal distribution of the elements is not important, which is a considerable advantage when null distributions are difficult to estimate accurately.

The testing procedure proposed here is based on simulated sampling from $\mathcal{P}_1(E, S_1(D))$. There are two straightforward ways to do this. First, let D^* be a random matrix obtained by a random permutation of the nonvoid elements of D. We have already argued that $S_1(D) = S_1(D^*)$. We also note that the distribution of nonvoid elements of D^* is exchangeable, hence by Lemma 1 $G(D^*)$ has distribution $\mathcal{P}_1(E, S_1(D))$. Alternatively, suppose D^* is any random matrix with continuously distributed *iid* nonvoid elements. Given any index sequence $\tilde{s} \in \mathcal{S}_1$, we can define a sequence of graphs $G^* = (G_{s_1}(D^*), \dots, G_{s_m}(D^*))$. It is easily verified that G^* has distribution $\mathcal{P}_1(E, \tilde{s})$.

2.1.5. Null Model 2 (Within Column Exchangeability). The use of $\mathcal{P}_1(E, S_1(D))$ as a null distribution rests on the assumption of elementwise exchangeability. A number of commonly encountered conditions may require alternative assumptions. For example, the columns of D may be derived from data obtained from a single high throughput assay. In this case, the columns may be independent, but not identically distributed. Furthermore, normalization procedures and other slide specific factors may affect any independence assumptions within a column. We therefore develop an alternative null model based on within column exchangeability, which is accomplished by conditioning on $S_2(D)$.

Definition 2. Suppose that we are given void edges E and an increasing count vector sequence $\tilde{v} = (v_1, \dots, v_m) \in \mathcal{S}_2$, $v_i = (v_{i1}, \dots, v_{iN})$. A random sequence of graphs $\tilde{h} = (h_1, \dots, h_m)$ possesses a null distribution $\mathcal{P}_2(E, \tilde{v})$ if h_1 is uniformly distributed on $\mathcal{G}_{v_1}^{-E}$, and if h_j conditional on $(h_1, \dots, h_{j-1})$ is uniformly distributed on $\mathcal{G}_{v_j}^{-E}[h_{j-1}]$ for all $j = 2, \dots, m$.

Then define our second null hypothesis:

(H_0^2) The columns of D have an exchangeable distribution among nonvoid elements, and are mutually independent.

This leads to the following lemma.

Lemma 2. *Under hypothesis H_0^2 the distribution of $G(D)$ conditional on $S_2(D)$ is given by $\mathcal{P}_2(E, S_2(D))$.*

Proof. The argument in Lemma 1 may be directly adapted by using only permutations T which map any element into its original column. □

Following the permutation procedure used to simulate $\mathcal{P}_1(E, S_1(D))$, we can simulate $\mathcal{P}_2(E, S_2(D))$ using independent within column permutations of D, resulting in D^*. By Lemma 2, $G(D^*)$ possesses distribution $\mathcal{P}_2(E, S_2(D))$. We note that by construction a graph sequence sampled from $\mathcal{P}_2(E, S_2(D))$ also conforms to $(E, S_1(D))$.

2.2. Hypothesis Test Algorithm. Suppose that we have a sample of graphs $H = (h_1, \dots, h_R) \in \mathcal{G}^R$ from a distribution $\mathcal{P}$, which in turn defines a random variable $\lambda(h)$ with distribution $\mathcal{P}_\lambda$, where h is distributed as $\mathcal{P}$. If $\mathcal{P}$ is a null distribution representing graphs with no significant structure, then the location of $\lambda(g)$ in the lower tail of $\mathcal{P}_\lambda$ is evidence of significant structure within g.

We will assume that when null hypothesis H_0^1 or H_0^2 does not hold, this violation is due to the existence of a true graph g'. In this case, all elements of D conform to the null hypothesis except for any D_{ij} for which $j \to i \in g'$, which are assumed to have smaller means than would be implied under the null distribution.

We therefore define statistics:

$$q(g \mid H) = (R+1)^{-1}\left(1 + \sum_{j=1}^{R} I\left\{\lambda\left(h_j\right) \le \lambda(g)\right\}\right), \tag{5}$$
$$z(g \mid H) = \sigma(H)^{-1}(\lambda(g) - \mu(H)),$$

where $\mu(H)$ and $\sigma(H)$ are the sample mean and standard deviation of sample $(\lambda(h_1), \dots, \lambda(h_R))$. Then $q(g \mid H)$ is the estimated P-value for a test against a null hypothesis that g is sampled from $\mathcal{P}$, and $z(g \mid H)$ is the associated z-score.

Now suppose that we are given $G(D)$, with $S_1(D) = (s_1, \dots, s_m)$, $S_2(D) = (v_1, \dots, v_m)$. We may generate a random sample from either $\mathcal{P}_1(E, S_1(D))$ or $\mathcal{P}_2(E, S_2(D))$, say $\widetilde{G}^* = (G_1^*, \dots, G_R^*)$. Set $G_i^* = (g_{i1}, \dots, g_{im})$, from which we extract sample $H_j = (g_{1j}, \dots, g_{Rj})$ so that when $\widetilde{G}^*$ is a random sample from $\mathcal{P}_1(E, S_1(D))$ or $\mathcal{P}_2(E, S_2(D))$, H_j is a uniformly distributed random sample from $\mathcal{G}_{s_j}^{-E}$ or $\mathcal{G}_{v_j}^{-E}$, respectively. This leads to the two sequences of statistics:

$$Q\left(G(D) \mid \widetilde{G}^*\right) = (q(G_1(D) \mid H_1), \dots, q(G_m(D) \mid H_m)),$$
$$Z\left(G(D) \mid \widetilde{G}^*\right) = (z(G_1(D) \mid H_1), \dots, z(G_m(D) \mid H_m)). \tag{6}$$

These sequences then form measures of the deviation of $G(D)$ which can be used to accomplish two tasks. First, we conjecture that the minimum point of these sequences will define a useful threshold P^{thr}, that is, a point in the sequence $G(D)$ below which most edges are true positives (a selected edge in true graph g'), and above which additional edges are primarily false positives (a selected edge not in true graph g'). Second, by generating further replications, we can estimate a global significance level for the presence of network structure. As will be discussed below, examining the entire range of the sequence $G(D)$ may be problematic, and so it may be truncated. Let $m_K = \max\{i : s_i \le K\}$, where $S_1(D) = (s_1, \dots, s_m)$. Then consider the truncated sequences:

$$Q^K\left(G(D) \mid \widetilde{G}^*\right) = \left(q(G_1(D) \mid H_1), \dots, q\left(G_{m_K}(D) \mid H_{m_K}\right)\right),$$
$$Z^K\left(G(D) \mid \widetilde{G}^*\right) = \left(z(G_1(D) \mid H_1), \dots, z\left(G_{m_K}(D) \mid H_{m_K}\right)\right). \tag{7}$$

Thus, all graphs of order K or less are considered. We first devise a statistic $W(G(D) \mid \widetilde{G}^*)$ which measures statistical differences of $G(D)$ from the sample $\widetilde{G}^*$. We then generate an additional set of R^* null replications from the null distribution, denoted by $\widetilde{G}^{**} = (G_1^{**}, \dots, G_{R^*}^{**})$. An empirical distribution is formed from the sample $W(G_1^{**} \mid \widetilde{G}^*), \dots, W(G_{R^*}^{**} \mid \widetilde{G}^*)$, from which a significance level for statistic $W(G(D) \mid \widetilde{G}^*)$ is directly obtainable. This represents the desired global significance level. We consider the four choices:

$$\begin{aligned} \widehat{W}_Q^K &= \min\left\{Q^K\left(G(D) \mid \widetilde{G}^*\right)\right\}, \\ \overline{W}_Q^K &= \text{mean}\left\{\log_{10}\left(Q^K\left(G(D) \mid \widetilde{G}^*\right)\right)\right\}, \\ \widehat{W}_Z^K &= \min\left\{Z^K\left(G(D) \mid \widetilde{G}^*\right)\right\}, \\ \overline{W}_Z^K &= \text{mean}\left\{Z^K\left(G(D) \mid \widetilde{G}^*\right)\right\}. \end{aligned} \tag{8}$$

We now summarize the proposed algorithm.

Algorithm A. (1) Construct hierarchical graph sequence $G(D)$ for data matrix D.

(2) Generate reference sample $\widetilde{G}^*$ from R replications of null model $\mathcal{P}_1(E, S_1(D))$ or $\mathcal{P}_2(E, S_1(D))$.

(3) Identify threshold P^{thr} as the minimum point of the sequence $Q^K(G(D) \mid \widetilde{G}^*)$ (or alternatively of $Z^K(G(D) \mid \widetilde{G}^*)$).

(4) Generate a new reference sample $\widetilde{G}^{**}$ from R^* replications of the null model.

(5) Calculate statistic $W(G(D) \mid \widetilde{G}^*)$, and determine its quantile position among replications $W(G_1^{**} \mid \widetilde{G}^*), \dots, W(G_{R^*}^{**} \mid \widetilde{G}^*)$. This gives the global significance level for the presence of graphical structure in the network.

The Algorithm A depends on a score λ which is sensitive to general forms of regularity. This is discussed in the next section.

3. Information-Based Scoring for Directed Graphs

Information theoretic methods are becoming increasingly important in bioinformatics (see, e.g., [10]) and have been recently used in various graphical modelling applications. Recent examples include [2–4, 11, 12]. This is generally done using the *minimum description length* (MDL) principle, [13–15], which is a general method of inductive inference based on the idea that a model's goodness of fit can be objectively measured by estimating the amount of data compression that it permits. The work proposed here is not formally an application of these methods but does share an interest in coding techniques for graphs.

3.1. Coding-Directed Graphs. The present objective is to devise a coding algorithm for a directed graph G using efficient coding principles [16]. The object to be coded is first reduced to a list of elements in a predetermined order (letters of a text or pixels of an image). Each element is coded separately into a codeword of binary digits, which are then concatenated to form one single binary string. It is important to ensure that each distinct object is converted to a unique code, and this may be done by ensuring that the codewords possess the *prefix property*; that is, no codeword is a prefix of another codeword. The simplest such code is the *uniform code*. If an element to be coded is one of N^{type} types, then each type can be uniquely assigned a binary string of $\lceil \log_2(N^{\text{type}}) \rceil$ bits, and any concatenation of uniform codes can be uniquely decoded. In the following development we will forgo the practice of rounding up to the next integer, since in the context of inference it is more intuitive for the code length to be a strictly increasing function of N^{type}.

In order to code a nonnegative integer using a uniform code we would have to specify an upper bound $I^{\max}$, giving $I^{\max} + 1$ types, and so a codeword length of $\lceil \log_2(I^{\max} + 1) \rceil$ for each integer. If we expect most integers to be significantly smaller than $I^{\max}$, this would be inefficient. We will therefore make use of a *universal code* $b(i)$ proposed in [17]. One segment of the code consists of a binary representation of the integer, with no leading 0's. The code is prefixed by a string consisting of 0's equal in length to the binary string followed by a 1. Thus, $b(0) = 1$, $b(1) = 011$, $b(2) = 00110$, and so on. In general, we will have code length $|b(i)| = 1 + 2\lfloor 1 + \log_2(i) \rfloor$ when $i > 0$, and $|b(i)| = 1$ for $i = 0$. This code is a prefix code, with the advantage that no upper bound need be specified, and it will be more efficient when smaller integer values are expected to be most prevalent. In the work which follows, we omit the rounding operation, and so accept the approximate code length of $b(i)$ as

$$\bar{b}(i) = \begin{cases} 1, & i = 0, \\ 3 + 2\log_2(i), & i > 0. \end{cases} \tag{9}$$

Again, it is more natural that $\bar{b}(i)$ be strictly increasing.

A directed order n graph may be represented as an $n \times n$ 0-1 *adjacency graph* (the class of such matrices is denoted by $\mathcal{M}^n$). An edge from node j to i is indicated by a 1 entry for row i and column j. Such a matrix may be completely represented by an ordered list of n subsets of $\{1, \dots, n\}$, in which the ith subset represents the entries of row i equaling 1. The graph itself may therefore be coded as a concatenation of n codewords representing the subsets. We assume that the value of n is available to the decoder.

To code a subset, a uniform code may used, so that any subset from n labels would be coded using n bits. However, in the applications considered here, it is often expected that the size of the subset is considerably smaller than n. An alternative strategy is to first specify the size k of the subset and then apply a uniform code to represent all subsets of that size. This involves concatenating a codeword for k (using the universal integer code) and a codeword for the subset (using a uniform code for $\binom{n}{k}$ possible subsets). A subset of size k from m objects will then be assigned a code length of

$$B_0(k, m) = \bar{b}(k) + \log_2 \binom{m}{k}. \tag{10}$$

We refer to this code as an *size indexed* code, in contrast to a uniform code. A code for matrix $M \in \mathcal{M}^n$ is then easily constructed by concatenating codewords for each row subset, giving code length

$$C_0(M) = \sum_{i=1}^{n} B_0(k_i, n), \tag{11}$$

where k_i is the number of 1 entries in row i. This code is similar to the one proposed in [11] but assumes that $\log_2(n)$ bits are used to code k_i, as required by a uniform code on n integers.

There will be some advantage to considering a modification to $C_0(M)$. If only a relatively small subset of nodes possess edges, then we may instead code a submatrix of M. Let $L(M)$ be the set of nodes which are part of at least one edge. Possibly, $|L(M)| \ll n$, in which case it may be advantageous to code only the $L(M) \times L(M)$ submatrix of M. But we would also need to code $L(M)$ itself. This object may be converted to codewords using the size indexed code and will appear in the code as a header, followed by the submatrix coded as described above. Thus, the code length for the $L(M) \times L(M)$ submatrix is

$$C_M(M) = B_0(|L(M)|, n) + \sum_{i \in L(M)} B_0(k_i, |L(M)|). \tag{12}$$

3.2. Properties of Graph Codes. We now examine the properties of the scores. In Algorithm A comparisons of graphs will be between those with equal numbers of edges. We will consider an asymptotic scenario in which the size of the largest subset is bounded by m, with $m \ll n$. Applying Lemma 5 of [18] we may write

$$C_0(M) = N_E(M)\log_2(n) + \left(\sum_{i=1}^{n} \overline{b}(k_i) - \log_2(k_i!)\right) + O(1), \tag{13}$$

where $N_E(M)$ is the number of 1 entries in M (i.e., the number of edges in the graph). If we now let $n \to \infty$, assume that $N_E(M)$ grows proportionally with n, and that the subset sizes remain bounded by m, then $C_0(M) = N_E(M)\log_2(n) + O(n)$. This means that when comparing graphs M, M' with equal numbers of edges the dominant terms of $C_0(M), C_0(M')$ are equal, since $N_E(M) = N_E(M')$ and the comparison will depend on the remaining dominant term

$$C_0^k(M) = \sum_{i=1}^{n} \overline{b}(k_i) - \log_2(k_i!). \tag{14}$$

We let $\mathcal{Z}^n$ be the set of all n-dimensional vectors of nonnegative integers.

Definition 3. A mapping $f : \mathcal{Z}^n \to \mathcal{R}$, $n \geq 2$ is called stepwise monotone when the following holds. Let $\widetilde{k}$ be any element of $\mathcal{Z}^n$ with at least two nonzero elements. Let k_i, k_j be any two components of $\widetilde{k}$ for which $1 \leq k_i \leq k_j$. Then let $\widetilde{k}' \in \mathcal{Z}^n$ be equal to $\widetilde{k}$, except that $k_i' = k_i - 1$ and $k_j' = k_j + 1$. Then $f(\widetilde{k}') \leq f(\widetilde{k})$, and f is called strictly stepwise monotone when the inequality can be replaced with strict inequality.

Note that $C_0^k(M)$ is a function of the vector of subset sizes $\widetilde{k} = (k_1, \ldots, k_n)$. The stepwise operation described in Definition 3 generates a hierarchy of subset lists based on the tendency to concentrate larger subset sizes in fewer subsets. In terms of graphs, the ranking will be based on the tendency for a fixed number of edges to target a smaller number of nodes. We now show that $C_0^k(M)$ is strictly stepwise monotone.

Lemma 3. *The mapping $C_0^k(M)$, interpreted as a function of the row totals $\widetilde{k} = (k_1, \ldots, k_n)$ of M, is strictly stepwise monotone.*

Proof. Noting the form of $\overline{b}(i)$ in (9) it is convenient to write $3 + 2\log_2(i) = 1 + 2\log_2(2i)$. Then from (14) we have

$$C_0^k(M) = n + \log_2\left(\frac{\Pi_{k_i>0} 4k_i^2}{k_i!}\right). \tag{15}$$

Let $\widetilde{k}$ and $\widetilde{k}'$ be two vectors from $\mathcal{Z}^n$ as described in Definition 3. If $k_i > 1$, the ratio of the product in the second term of (15) may be written as

$$\frac{\Pi_{k_i'>0} 4(k_i')^2/k_i'!}{\Pi_{k_i>0} 4k_i^2/k_i!} = \frac{(k_i-1)^2\left(k_j+1\right)^2/(k_i-1)!\left(k_j+1\right)!}{k_i^2 k_j^2/k_i!k_j!}. \tag{16}$$

Consider the quantities $A_L = (ab)^2/(a!b!)$ and $A_U = ((a-1)(b+1))^2/((a-1)!(b+1)!)$, where a, b are any integers for which $1 \leq a \leq b$. Then $(a-1)(b+1) = ab + a - b - 1 < ab$, and $(a-1)!(b+1)! = a!b!(b+1)/(a) > a!b!$, from which it follows $A_U < A_L$. Using (16) this inequality may be applied directly to the second term of (15) to verify the lemma. If $k_i = 1$, then we have the corresponding ratio:

$$\frac{\Pi_{k_i'>0} 4(k_i')^2/k_i'!}{\Pi_{k_i>0} 4k_i^2/k_i!} = \frac{\left(k_j+1\right)^2/\left(k_j+1\right)!}{4k_j^2/k_j!} = \frac{k_j+1}{4k_j^2}, \tag{17}$$

which can easily be shown to be less than one for all $k_j \geq 1$. The lemma therefore holds for this case as well. □

Consider four graphs of n nodes consisting of $k = 4$ edges contained in the subgraphs in Figure 1. Denote the respective adjacency matrices by M_A, M_B, M_C, M_D. It is easily verified that $C_0(M_A) < C_0(M_B) = C_0(M_C) = C_0(M_D)$. This leads to two problems. First, we would like M_A and M_B to be scored equally. Second, graph (C) clearly has more interesting structure than (D), but it has the same score. To address the first problem, we may score the transpose of the adjacency matrices, which gives $C_0(M_A) = C_0(M_B^T)$.

The second problem can be addressed using the modified score C_M. We have, for fixed $k \ll n$,

$$\begin{aligned} C_M(M_C) &= (k+1)\log_2(n) + o\left(\log_2(n)\right), \\ C_M(M_D) &= 2k\log_2(n) + o\left(\log_2(n)\right), \\ C_0(M_C) &= C_0(M_D) = n + o(n). \end{aligned} \tag{18}$$

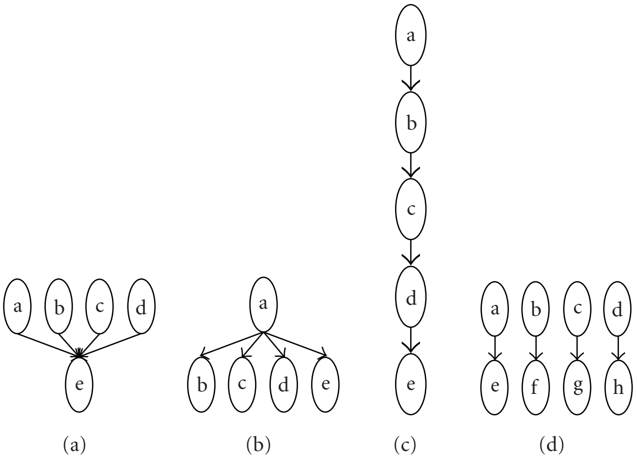


Figure 1: Four sample graphs, each of 4 edges.

Thus, for large enough n, $C_M(M_C) < C_0(M_C)$ and $C_M(M_D) < C_0(M_D)$, so that C_M will be smaller for M_C and M_D. Similarly $C_M(M_C) < C_M(M_D)$, so that C_M can be seen to be sensitive to chain structure, whereas C_0 is not. We then adopt the compound score:

$$\lambda = \min\left\{C_M(M), C_M\left(M^T\right)\right\}. \tag{19}$$

We omit from λ a prefix consisting of a fixed number of bits which will indicate which score was the minimum.

4. Examples

In this section we apply Algorithm A to a set of examples, first a synthetic network based on a typical pathway, then one based on yeast genome perturbation experiments.

4.1. Synthetic Network (MAP Kinase). The pathway illustrated in Figure 2 represents a known MAP kinase signal transduction cascade, used in [6] to illustrate a network model. This pathway possesses 12 genes and 13 edges. We will add $N - 12$ spurious genes to the model, allowing N to vary. The objective is to simulate a data matrix D as defined in steps (S1)-(S2), which might plausibly summarize experiments generated by this network. The strategy will be to first demonstrate the methodology on a statistically favorable case to clarify the objectives. The case will then be modified to present a scenario in which statistical noise plays a more prominent role.

4.1.1. Model Simulation. Let M^* be the adjacency matrix of the graph in Figure 2. Gene j directly regulates i if $M^*_{ij} = 1$. We also expect perturbation of j to affect genes further downstream; so we say that i and j are in an *order k relationship* if there is a path from j to i of k edges, and no shorter path exists. This holds if the ijth element of the product $(M^*)^{k'}$ is nonzero for $k' = k$ and zero for $k' < k$.

If i and j are in an *order k relationship* simulate a normal random variable with mean μ_k and variance 1, then let D_{ij} be the P-value associated with a hypothesis test $H_0 : \mu = 0$ against $H_1 : \mu > 0$. If i and j have no relationship, let D_{ij} be uniformly distributed.

A model is defined by characteristics μ_k and N. To study a given model the data matrix D is replicated 2500 times as described above. For each replication we apply Algorithm A, setting $K = 100$, $R = R^* = 2500$. The compound score λ of (19) is used. We use the elementwise exchangeable null hypothesis H_0^1.

4.1.2. Algorithm Evaluation. A study of the algorithm must take into account its dual purpose. We may accept $G^{\mathrm{acc}}(D, P^{\mathrm{thr}})$ as an estimated accessibility graph which can be compared to the true graph. On the other hand, viewed as a multiple testing procedure, the objective is an efficient choice of P^{thr} along a type of error curve, giving the expected number of true edges as a function of the total number of edges within graphs of the sequence $G(D)$. The properties of the error curve define the accuracy with which a cellular network can be inferred. Ideally, the error curve increases with slope 1 until the graph is constructed and then remains constant. Statistical variation forces deviation from this ideal; so the goal in the selection of P^{thr} is to identify a position along the error curve such that below (or above) this position most new edges are true (or false) positives.

We now discuss the calculation of the error curve. It will be convenient to restrict attention to relationships up to an order k. Suppose that G_0^k is the true order k graph, in the sense that it contains edge $j \rightarrow i$ if and only if i and j have an order $k' \leq k$ relationship. In our example, G_0^1 is equivalent to the graph in Figure 2. Let D' represent a simulated replicate from the given model, from which we construct sequence $G' = G(D')$. Let r_i^k be the number of edges of G_0^k contained in element g_i' of G'. We will estimate two forms of the error curve. For the first, using replicates of G' we calculate the sample mean value $\overline{r}_i^k$ of r_i^k for each $i = 1, \dots, K$. For the second, for each replicate G' we use the edge value i' minimizing $Z(G' \mid \widetilde{G}^*)$, thus identifying P^{thr}, then capturing the pairs $(i', r_{i'}^k)$ to be displayed in the form of a scatter plot.

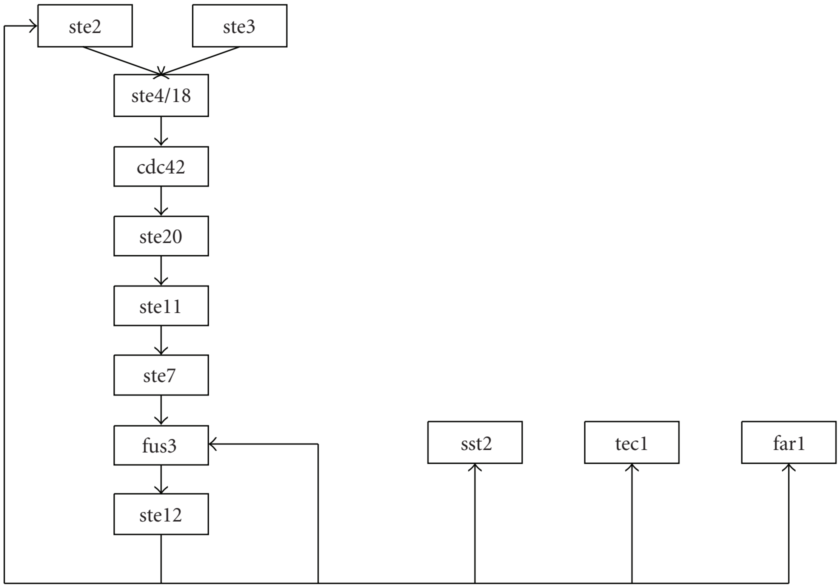


Figure 2: Sample MAP Kinase network.

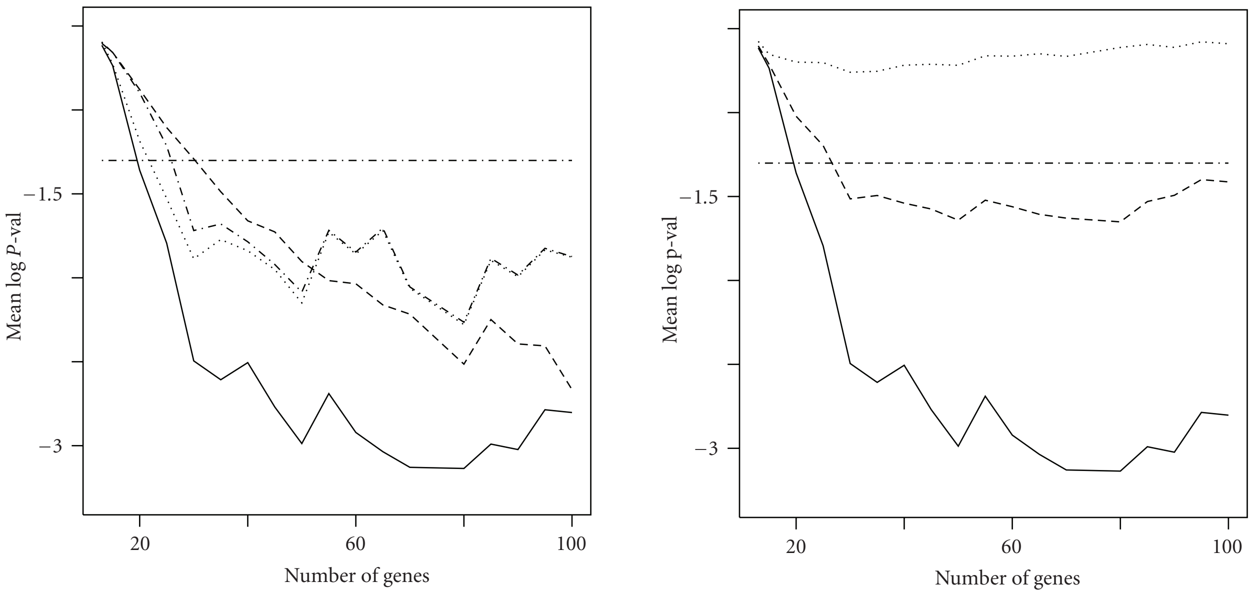
4.1.3. Model 1 (Direct Regulation Only). We will first consider a simplified version of the problem, in which order $k > 1$ regulatory relationships are ignored ($\mu_k = 0$ for $k > 1$). We study models defined by $\mu_1 = 2, 3, 4$ and $N = 13, 15, \ldots, 100$ in increments of 5. Each test network is replicated 2500 times. Figure 3 summarizes the achieved global significance for the statistics proposed in (8). In Plot 1 the four statistics proposed in (8) are applied to the models defined by $\mu_1 = 4$ over the proposed range of N. The plot shows the average attained global significance levels on a log scale (base 10). The horizontal axis represents a significance level of 0.05. Noticeably greater power is demonstrated for statistic $\widehat{W}_Z^K$, and significance levels are well below the 0.05 value over most of the range of N, demonstrating the ability of the procedure to detect overall network structure. Sensitivity to the strength of the statistical evidence is demonstrated in Figure 3, Plot 2, in which average significance levels for models $\mu_1 = 2, 3, 4$ over the proposed range of N are reported. Here we use only statistic $\widehat{W}_Z^K$, which will become our default choice.

An interesting feature of these plots is the increase in power with the increase in the number of spurious genes. This is the opposite of what is usually expected in gene discovery but follows from the use of graphical structure as statistical evidence. The existence of such structure implies higher connectivity of a smaller subset of genes than would occur at random. The existence of a larger pool of unconnected genes should, to some extent, contribute to the significance of graphical discovery, since the existing structure would be less likely to have occurred by chance. Of course, the competing effect of false positives usually associated with multiple hypothesis testing will also exist. The relative importance of these effects remains to be analyzed.

A single simulation will illustrate the implementation of the algorithm. Here we assume $N = 60$ candidate genes, with effect size defined by $\mu_1 = 4$. The Z-score values $Z(G(D) \mid \widetilde{G}^*)$ are shown in Plot 1 of Figure 4. A clear minimum point is evident at 9 edges. Plot 2 shows the cumulative proportion of true positives among the edges defining the graphs in the sequence $G(D)$ (i.e., the number of true edges in $G_i(D)$ by edge). The minimum point at the 9th edge is clearly a point at which the graph is almost completely constructed, and above which most new edges will be false positives.

4.1.4. Model 2 (Including Transitive Regulation). We next include evidence of transitive regulations by simulating perturbation effects of size $(\mu_1, \mu_2, \mu_3, \mu_4) = (3, 2, 1, 1)$, with $\mu_k = 0$ for $k > 4$. We will examine specifically the $N = 60$ gene model and use 5000 replicates. The error curve is shown in Figure 5 (Plot 1) for up to order $k = 4$ relationships. Note that the error curve based on selected minimum points yields slightly higher values. We conjecture that the minimum selection process introduces greater accuracy (discussion in the next subsection pertains to this issue). As expected, both error curves are at first of unit slope, up until a point at which false positives begin to dominate. As emphasized earlier, the error curve represents an inherent limit of the accuracy possible under given experimental conditions. The role of our procedure is therefore to determine a suitable location along that curve. In Figure 5 (Plot 2), a histogram of the captured minimum points i' is shown. Interestingly, the mode of the histogram is located precisely where the error curve is no longer of unit slope, which is the point we wish to identify.

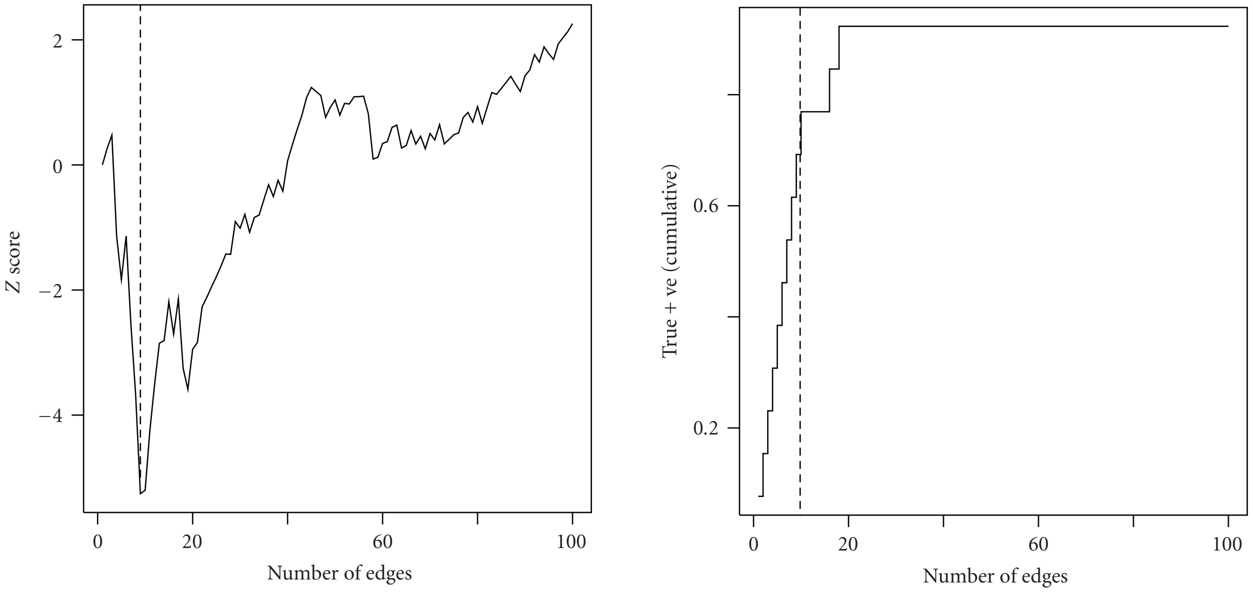
4.2. Yeast Genome Expression Data. In [19] a series of gene deletion and drug experiments are reported, resulting in a compendium of 300 microarray gene expression profiles on the yeast genome. We extracted 266 genes for which



(a)

(b)

Figure 3: Global significance plots for simulation study. Number of genes N is varied within each plot. Probabilities are given on a base 10 logarithmic scale. A $\log_{10}(0.05)$ axis is indicated. (a): Case $\mu_1 = 4$ using statistics $\widehat{W}_Q^K$ [··]; $\overline{W}_Q^K$ [− · −]; $\widehat{W}_Z^K$ [−]; $\overline{W}_Z^K$ [− −]. (b): Cases $\mu_1 = 2$ [··]; $\mu_1 = 3$ [− −]; $\mu_1 = 4$ [−] using statistic $\widehat{W}_Z^K$.

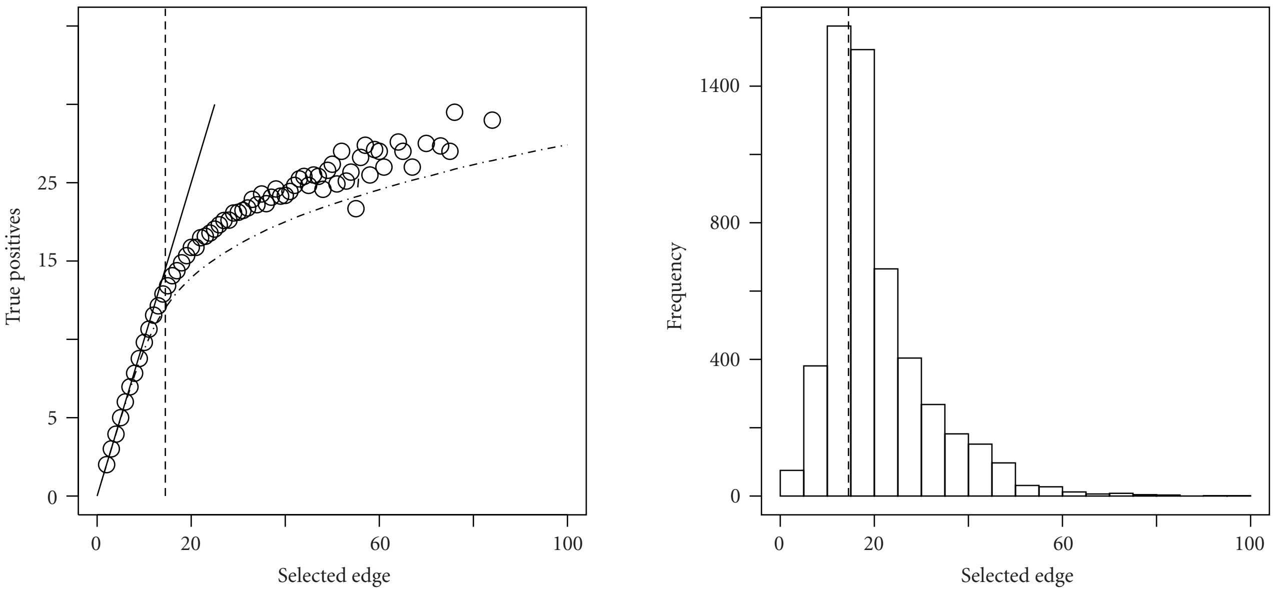


(a)

(b)

Figure 4: Properties of sample data set D, for $N = 60$ genes, $\mu_1 = 4$. (a): Value of Z-score sequence $Z(G(D) \mid \widetilde{G}^*)$. Minimum point is located at edge 9, indicated by dashed line. (b): Cumulative proportion of true positive edges within sequence $G(D)$. There are 13 true edges. Minimum point of $Z(G(D) \mid \widetilde{G}^*)$ (Plot 1) is indicated by dashed line.

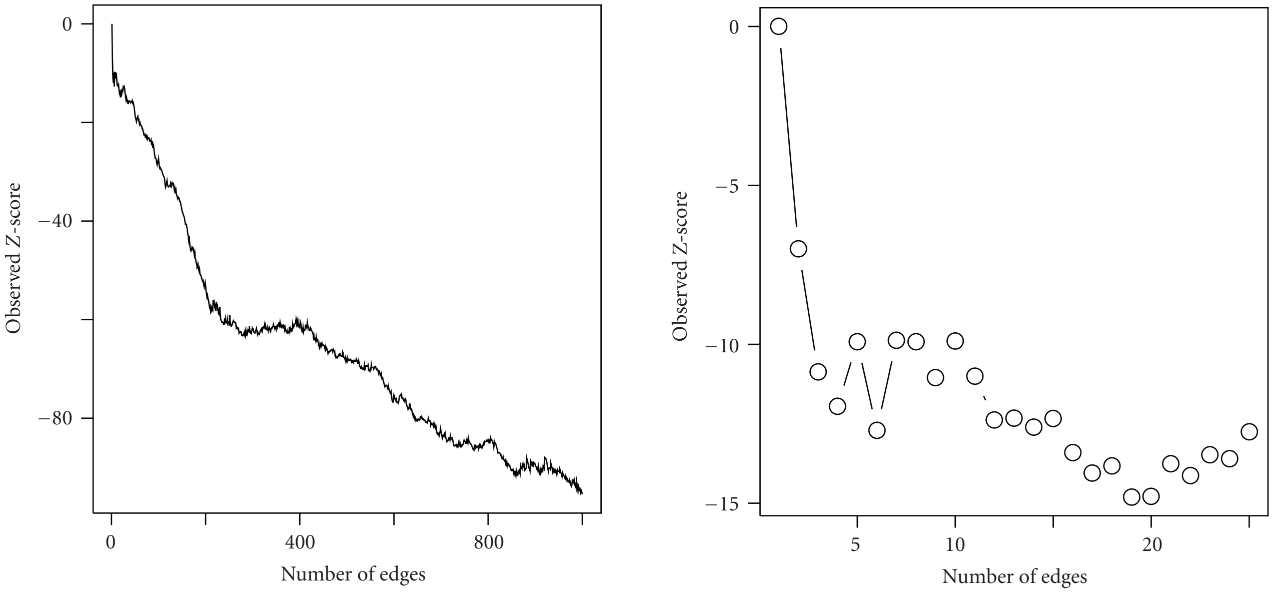
single deletion experiments were performed. By matching the responses for those genes a 266 × 266 data matrix D of perturbation effect P-values was constructed (the P-values used are those reported in [19]). Algorithm A was applied using a maximum of $K = 1000$ edges, using $R = R^* = 500$ replications of a null matrix; then $Z(G(D) \mid \widetilde{G}^*)$ was calculated as above. These replications were supplemented by an application with settings $K = 100$, $R = R^* = 5000$. We use the element wise exchangeable null hypothesis H_0^1.



(a)

(b)

Figure 5: Sample data set D, for $N = 60$ genes, $(\mu_1, \mu_2, \mu_3, \mu_4) = (3, 2, 1, 1)$. (a): error curve calculated by mean true positive $\bar{r}_i^k$ [− · −], and scatter plot of selected minimum points and true positives $(i', r_{i'}^k)$. We use $k = 4$. The dashed line indicates the mode in Plot 2. (b): Histogram of selected minimum points i'. The mode is indicated by the dashed line.



(a)

(b)

Figure 6: Values of $Z(G(D) \mid \widetilde{G}^*)$ for yeast data. Edge ranges 1–1000 and 1–25.

The Z-scores are shown in Figure 6. A significant deviation from zero is shown almost immediately and persists throughout the observed range. The global P-value at 1-2 edges is estimated as 0.0084 and decreases rapidly beyond this point. Table 1 lists the edges associated with the 10 lowest P-values in D. At this point obvious graph structure is apparent, as the first four edges all have a common parent *tup1*. It is interesting to note that the z-score falls as edges 2 to 4 are added, each of which contains a gene found in the previous edges. Edge 5, however, introduces two new genes, at which point the Z-score increases. In fact, this rule persists up to the 12th edge; that is, the z-score decreases if and only if at least one gene of a new edge exists among the previous edges. It also holds among 83%, 74%, and 63% of the first 25, 100, and 1000 edges, respectively.

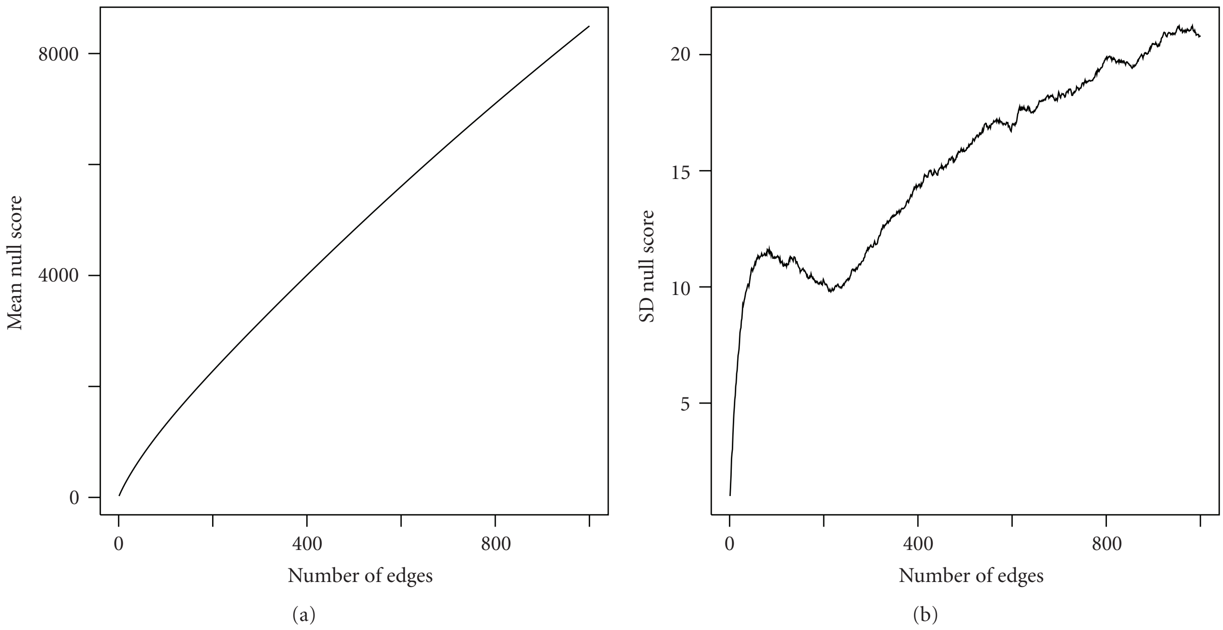


Figure 7: Mean and standard deviation of graph scores for 266 × 266 null perturbation matrix by edge number.

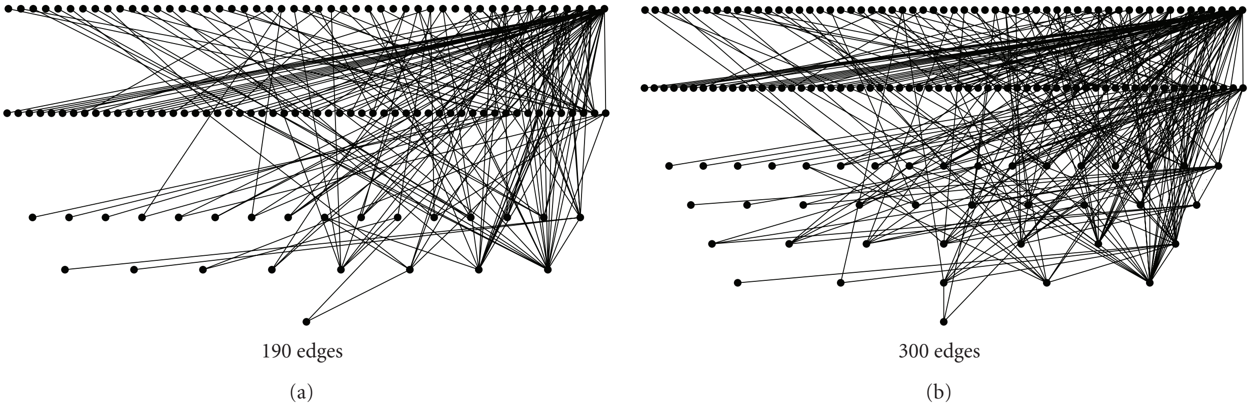


Figure 8: Estimated accessibility graphs for yeast genome data, using 190 and 300 edges. The choice of 190 genes represents the application of the Benjamini-Hochberg FDR control procedure.

In this example, the Z-score is clearly able to distinguish between edges which contribute to graph structure and those that do not up to some number of edges. The application of this principle over a large range of edges is complicated by the increasingly complex statistical properties of the graph score, as suggested in Figure 7. While the mean score increases smoothly, the growth of the standard deviation is more complicated. We would expect this to some degree. The number of offspring for each node would be approximately Poisson distributed when such numbers are small, but eventually this probability law will no longer hold when subsets become large enough. It is therefore problematic to identify interesting features of the Z-score plot over large ranges of edge number.

Finally, we make a note comparing multiple testing procedures (MTPs) and our proposed graph-based procedure. Accepting the P-values reported in [19], two well-known MTPs were applied to the P-values of matrix D (see [1] for details). Using the Bonferroni procedure (FWER = 0.05) 41 P-values are rejected, whereas using the Benjamini-Hochberg procedure (FDR = 0.05) 190 P-values are rejected. Figure 8 displays the connectivity graphs formed from the first 190 and the first 300 edges. All edges point "downward" in the diagram (arrows are omitted for clarity). Three exceptions are indicated by dashed lines, which are bidirectional. The graphs contain no cyclic behavior other than these edges (a simple simulation experiment confirms that this level of cyclicity is compatible with the Erdös-Renyi random graph model).

Table 1: Graph edges associated with 10 lowest ranking P-values for yeast data.

P-val Rank	Response Gene	Perturbed Gene	P-val Rank	Response Gene	Perturbed Gene
1	hpa3	tup1	6	yer066c-a	tup1
2	yor009w	tup1	7	rts1	yor015w
3	pau2	tup1	8	yhr022c	ssn6
4	yhr022c	tup1	9	phd1	tup1
5	ste2	ste12	10	ald5	yer050c

If we accept the 190 edge graph as that resulting from the application of an MTP, we then note that the proposed graph-based method results in significantly more structural discovery. The global significance level for a graph with 1000 edges can be taken as extremely small from an estimated Z-score of -95.4. This significance level applies to subgraphs from the sequence $G(D)$. Similarly, additional structure in the 300 gene graph compared to the 190 gene graph can be clearly seen. In order to define a "highly connected gene," we simulate random graphs to estimate a distribution of a gene's edge order X. For $N = 190$ and $N = 300$ edges among 266 nodes we have $P(X \geq 4) \approx 0.006$ and $P(X \geq 5) \approx 0.006$. Thus, we define any gene with at least 4 and 5 edges as "highly connected" in the respective graphs. Under these criteria, the respective graphs contain 33 and 43 such genes. The most connected gene in the 190 gene graph is $tup1$ with 38 edges. This gene is also the most connected gene in the 300 gene graph (46 edges). In general, more highly connected genes are added between edges 190 and 300, while additional edges are added to already highly connected genes.

5. Conclusion

A common problem in the statistical analysis of high-throughput data is the selection of a threshold for statistical evidence which controls false discovery. Such data is often used to construct graphical models of gene interactions. A threshold selection procedure was proposed which is based on the observed graphical structure implied by a given threshold. This procedure can be used both for threshold selection and to estimate a global significance level for graphical structure. The method was demonstrated on a small simulated network as well as on the "Rosetta Compendium" [19] of yeast genome expression profiles. The methodology proved to be accurate and computationally feasible.

Further investigation is warranted in a number of issues. The graphs investigated here were unconstrained directed graphs. Application to undirected graphs and directed acyclic graphs (DAGs) will require more sophisticated graph simulation algorithms. Additionally, the long range statistical behavior of the proposed graph code is complex. Such issues will need to be carefully examined before a general threshold selection technique can be proposed.

A software implementation of the proposed procedures is available from the author's web site, in the form of an R library at http://www.urmc.rochester.edu/biostat/people/faculty/almudevar.cfm.

Acknowledgment

This work was supported by NIH Grant GM075299.

References

[1] S. Dudoit, J. P. Shaffer, and J. C. Boldrick, "Multiple hypothesis testing in microarray experiments," *Statistical Science*, vol. 18, no. 1, pp. 71–103, 2003.

[2] T. E. Ideker, V. Thorsson, and R. M. Karp, "Discovery of regulatory interactions through perturbation: inference and experimental design," in *Pacific Symposium on Biocomputing*, pp. 305–316, 2000.

[3] W. Zhao, E. Serpedin, and E. R. Dougherty, "Inferring gene regulatory networks from time series data using the minimum description length principle," *Bioinformatics*, vol. 22, no. 17, pp. 2129–2135, 2006.

[4] J. Dougherty, I. Tabus, and J. Astola, "Inference of gene regulatory networks based on a universal minimum description length," *EURASIP Journal on Bioinformatics and Systems Biology*, vol. 2008, Article ID 482090, 11 pages, 2008.

[5] A. Wagner, "Reconstructing pathways in large genetic networks from genetic perturbations," *Journal of Computational Biology*, vol. 11, no. 1, pp. 53–60, 2004.

[6] S. Onami, K. M. Kyoda, M. Morohashi, and H. Kitano, "The DBRF method for inferring a gene network from large-scale steady-state gene expression data," in *Foundations of Systems Biology*, H. Kitano, Ed., pp. 59–75, The MIT Press, Cambridge, Mass, USA, 2001.

[7] A. Wagner, "How to reconstruct a large genetic network from n gene pertubations in fewer than n^2 easy steps," *Bioinformatics*, vol. 17, no. 12, pp. 1183–1197, 2002.

[8] B. Bollobas, *Random Graphs*, Academic Press, London, UK, 1985.

[9] A. Wagner, "Estimating coarse gene network structure from large-scale gene perturbation data," *Genome Research*, vol. 12, no. 2, pp. 309–315, 2002.

[10] J. Rissanen, P. Grünwald, J. Heikkonen, P. Myllymäki, T. Roos, and J. Rousu, "Information theoretic methods for bioinformatics," *EURASIP Journal on Bioinformatics and Systems Biology*, vol. 2007, Article ID 79128, 2 pages, 2007.

[11] N. Friedman and M. Goldszmidt, "Learning Bayesian networks with local structure," in *Learning in Graphical Models*, M. I. Jordan, Ed., pp. 421–459, The MIT Press, Cambridge, Mass, USA, 1998.

[12] A. Almudevar, "A graphical approach to relatedness inference," *Theoretical Population Biology*, vol. 71, no. 2, pp. 213–229, 2007.

[13] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, no. 5, pp. 465–471, 1978.

[14] P. D. Grünwald, *The Minimum Description Length Principle*, The MIT Press, Cambridge, Mass, USA, 2007.

[15] J. Rissanen, *Information and Complexity in Statistical Modeling*, Springer, New York, NY, USA, 2007.

[16] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley, New York, NY, USA, 1991.

[17] J. Rissanen, "A universal prior for integers and estimation by minimum description length," *Annals of Statistics*, vol. 11, pp. 416–431, 1983.

[18] A. Almudevar, "Efficient coding of labelled graphs," in *Proceedings of IEEE Information Theory Workshop (ITW '07)*, pp. 523–528, Lake Tahoe, Calif, USA, September 2007.

[19] T. R. Hughes, M. J. Marton, A. R. Jones, et al., "Functional discovery via a compendium of expression profiles," *Cell*, vol. 102, no. 1, pp. 109–126, 2000.