A major challenge in systems biology is to quantitatively understand how cells evolve during development and differentiation. However, one challenge in estimating such evolving networks is that their host cells not only contiguously evolve, but also branch over time. For example, a stem cell evolves into two more specialized daughter cells at each division, forming a tree of networks. Another example is in a laboratory setting: a biologist may apply several different drugs individually to malignant cancer cells to analyze the effects of each drug on the cells; the cells treated by one drug may not be intrinsically similar to those treated by another, but rather to the malignant cancer cells they were derived from.

We propose a novel algorithm, Treegl, an $l_1$ plus total variation penalized linear regression method, to effectively estimate multiple gene networks corresponding to cell types related by a tree-genealogy, based on only a few samples from each cell type. Treegl takes advantage of the similarity between related networks along the biological lineage, while at the same time exposing sharp differences between the networks. We demonstrate that our algorithm performs significantly better than existing methods via simulation. Furthermore we explore an application to a breast cancer dataset, and show that our algorithm is able to produce biologically valid results that provide insight into the progression and reversion of breast cancer cells.

1 INTRODUCTION

A major challenge in systems biology is to quantitatively understand and model the topological and functional properties of cellular networks, such as the transcriptional regulatory circuitry and signal transduction pathways that control cell behavior in complex biological processes. In complex organisms, biological processes such as differentiation and development are often controlled by a large number of molecules that exchange information in a spatial-temporally specific and context-dependent manner. These cellular networks are inevitably changing to take on different functions and reacting to changing environments. This necessitates studying different networks for each condition, such as each different developmental stage, tissue subtype, and cell lineage. Most existing techniques for reconstructing molecular networks based on high-throughput data ignore the intricate dependencies between networks of closely related biological subjects.

For example, when studying cancer development, it is common to infer gene networks based on microarray data from different cancer specimens or cell lines separately and independently, despite that these biomaterials are usually collected over a contiguous disease progression course. As we discuss in detail, such an ‘isolationist’ strategy can compromise both the statistical power and biological insight of the inferred networks. In this article, we present a new methodology called Treegl, which adopts a statistically more powerful and biologically more natural ‘connectionist’ principle. Treegl reconstructs gene networks in related biological subjects via an inter-dependent approach such that the inferred networks directly embody and exploit the relationships of the biological subjects they represent. As a result, this reveals deeper insight on how the structure, function, and behavior of such networks evolve during evolution, differentiation and environmental perturbation.

To better understand our rationale, take the analysis of stem cell differentiation as an example. It is well known that all organ- and tissue-specific cells in a multicellular organism are differentiated from a stem cell, following a well-known genealogy (Figure 1). To date, gene networks from many of these organs and tissues have been derived using a variety of computational or experimental technologies (Basso et al., 2005; Hyatt et al., 2006; Li, 2004). However, knowledge about the cell lineage has rarely been utilized in constructing these networks. For example, according to the genealogy in Figure 1, the platelets are more closely related to the red blood cells than to the lymphoblasts. The gene networks present in the platelets and red blood cells are thus expected to be more similar; microarray data from red blood cells should reflect the topology of a platelet’s network to a greater extent than that of a lymphoblast network.

Is it therefore legitimate and possible to use the red blood cell’s microarray in addition to the platelet’s microarray to infer the platelet’s network? And, if yes, how? Essentially, what one needs to handle is a network of networks. In this article, we focus on the class of tree-shaped biological genealogies. This class of genealogies can be naturally found in crop and animal breeding, species evolution, cell-lineage construction, and carcinogenesis.

© The Author(s) 2011. Published by Oxford University Press.
This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/2.5), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
1.2 Our contribution

In this work, we move beyond the static and time-varying assumptions, and focus on the more general case of tree-evolving genealogies that we believe are more natural for the biological phenomena that we seek to explore. We propose an algorithm called Tree-smoothed graphical lasso (Treegl), that can effectively and jointly recover evolving regulatory networks present in multiple cell-types related by a tree genealogy.

Our approach takes advantages of the similarities of networks nearby in the genealogy, but can also reveal sharp differences. Moreover, by building on the method of neighborhood selection via the lasso (Meinshausen and Bühlmann, 2006), our approach works well even when the number of genes is much larger than the number of samples.

We were motivated by the many applications discussed above in the development of Treegl. However, in this article, we focus on applying Treegl to study the progression and reversion of breast cancer cells in 3D organotypic cultures (Itoh et al., 2007; Liu et al., 2004; Weaver et al., 1997). The cell-line in question begins as non-malignant, organized and non-tumorigenic cells that progress to apolar, disorganized and tumorigenic cancer cells. Several different drugs are applied and the genealogy then branches to different reverted cells with partially polarized structures. Although our dataset is small, we are able to show that we obtain biologically valid and intriguing results through our method.

2 METHODS

2.1 Probabilistic representation of gene networks

Consider the problem of modeling N different, but independent (we will consider dependency in the next subsection), gene regulatory networks, each corresponding to a unique cell type (say, type n) from a cell bank B where \( B = \{B_1, B_2, \ldots, B_N\} \) with \( S \), i.i.d microarray measurements of all genes in cell type n, and consisting of the same set of p genes across all cell types. Without loss of generality, a gene network can be represented by a probabilistic graphical model, such as a Markov random field (MRF) if the gene states are taken as discrete (Segal et al., 2003), or a Gaussian graphical model (GGM) if the gene states are set to the continuous measurements of the microarray signal (Dobra et al., 2004), or a Bayesian network (Friedman et al., 2000).

In this article, we use cell-type specific undirected Gaussian graphical models to model the gene networks, but the general principle of our method can be extended to discrete Markov random fields as well.

Let \( \mathbf{X}^{(n)} = (X_1^{(n)}, X_2^{(n)}, \ldots, X_p^{(n)}) \) represent a network in cell type n, of which \( X_{s(n)} \) denotes the set of genes, and \( E^{(n)} \) denotes the set of edges over vertices. An edge \( (u,v) \in E^{(n)} \) can represent a relationship (e.g. influence or interaction) between genes u and v. Let \( \mathbf{X}^{(n)} = (X_1^{(n)}, \ldots, X_p^{(n)}) \), where \( n \in N \), \( s \in \{1, \ldots, n_s\} \), and \( p = |V| \), be a random vector of nodal states that are real valued and standardized, such that each dimension has mean 0 and variance 1. We assume that \( \mathbf{X}^{(n)} \) follows a multivariate Gaussian distribution with mean 0 and covariance matrix \( \Sigma^{(n)} \), so that the conditional independence relationships among the genes can be encoded as a Gaussian graphical model. It is a well known fact that for GGMs, edges in the graph correspond to non-zero elements in the inverse covariance matrix (known as the precision matrix), which we denote by \( \mathbf{\Omega}^{(n)} = (\omega_{uv}^{(n)})_{u,v=1}^p \). Thus, estimating the graph structure is equivalent to selecting the non-zero elements of the precision matrix.

As commonly done, instead of directly estimating the precision matrix elements \( \omega_{uv}^{(n)} \), we estimate the partial correlation coefficients \( \rho_{uv}^{(n)} \), where \( \rho_{uv}^{(n)} \) is the correlation between gene u and gene v conditioned on the values of all the other genes. Partial correlation coefficients are related to the precision
matrix elements by Equation (1).

\[
\rho^{uv}_{n} = \frac{\omega^{uv}_{n}}{\sqrt{\omega^{uu}_{n} \omega^{vv}_{n}}}
\]  

As shown in Equation (1), \( \rho^{uv}_{n} \) is zero if and only if \( \omega^{uv}_{n} \) is zero. Therefore, in terms of network structure estimation, the network resultant from the non-zero \( \omega^{uv}_{n} \) is equivalent to that from the non-zero \( \omega^{uu}_{n} \). Furthermore, the partial correlation is quite intuitive in the sense that a high positive value of \( \rho^{uv}_{n} \) indicates that the genes \( u \) and \( v \) are strongly positively correlated (conditioned on the other genes), a low negative value indicates the genes are strongly negatively correlated (conditioned on the other genes), and \( \rho^{uv}_{n} = 0 \) for all \( u, v \in \mathcal{N}(\pi) \). As a result, we simply consider estimating the partial correlation coefficients and designate these as the edge values in \( \mathcal{G}^{(\pi)} \).

\[
\mathcal{L}^{(\pi)} = [\rho^{uv}_{n} |_{u \neq v} > 0].
\]  

2.2 Neighborhood selection

A number of recent papers have studied how to estimate this model from data that are assumed to be i.i.d. samples from the model, and the asymptotic guarantee of the estimator (Bresler et al., 2008; Wainwright et al., 2007). In particular, an efficient neighborhood selection algorithm (Meinshausen and Bühlmann, 2006) based on \( \ell_1 \)-norm regularized regression has been proven effective (often called neighborhood selection). In this approach, the neighborhood of each gene \( u \) is estimated independently using a penalized linear regression with a lasso-style (i.e. \( \ell_1 \)-norm) regularization over edge weights. The regression goes around every gene in the network, leading to completion of a network. In every neighborhood estimation step, gene \( u \) is treated as a response variable, all the other genes are the covariates, and the weights are the correlations between the other genes and \( u \). More formally, let \( X_u \) indicate the \( p \times 1 \) vector of the values of all genes except \( u \). Similarly, \( \Theta_u = [\theta_{u,v} : v \in \mathcal{V} \setminus u] \). Using a well known result (Lauritzen, 1996) that the partial correlation coefficients can be related to the following regression model:

\[
X_u^{(n)} = \sum_{v \neq u} \omega^{uv}_{n} X_v^{(n)} + \epsilon_u^{(n)}, \quad u \in [p],
\]  

where \( \epsilon_u^{(n)} \) is uncorrelated with \( X_u^{(n)} \) if and only if

\[
\rho^{uv}_{n} = \frac{\omega^{uv}_{n}}{\sqrt{\omega^{uu}_{n} \omega^{vv}_{n}}}
\]  

Some algebra gives that

\[
\rho^{uv}_{n} = \text{sign}(\theta^{uv}_{n}) \sqrt{|\theta^{uv}_{n}|} /
\]  

The above equations basically indicate that we can solve for the regression coefficients \( \theta^{uv}_{n} \) using a linear regression, where the response variable corresponds to \( X_u \) and the covariates correspond to \( X_v \). The corresponding partial correlation coefficients can be recovered using Equation (5). As \( \ell_1 \) penalty is applied to encourage a sparse solution, as in the lasso (Tibshirani, 1996).

This surprisingly simple method, when applied over i.i.d. nodal samples (e.g. i.i.d. microarray measurements), has very strong theoretical guarantees about recovering the correct network structure. It has been shown that under certain variable conditions it is possible to obtain an estimator of the edge set \( \mathcal{E} \) that achieves a property known as sparsistency (Meinshausen and Bühlmann, 2006; Wainwright et al., 2007), which refers to the case where a consistent estimator of \( \mathcal{E} \), i.e. the network structure, can be attained when the true degree (i.e. number of neighbors) of each node is much smaller than the size of the graph (even when the sample size is significantly smaller than the number of genes).

Unfortunately, in the case of the tree-evolving network concerned in this article, we have to deal with a much harder problem since our samples are no longer i.i.d., and our networks are no longer independent of each other. For this purpose, we need to extend the basic neighborhood selection lasso algorithm as shown in the following subsections.

2.3 Tree-evolving gene networks over biological lineages

We are interested in reconstructing a set of networks \( \mathcal{G}^{(1)}, \ldots, \mathcal{G}^{(\pi)} \) that are not independent of each other, but are related by a genealogy over their respective host cell-types, thereby constituting a tree evolving network. Formally, given a genealogy over members of a cell bank \( B \), we introduce an ordering over networks \( \mathcal{G}^{(1)}, \ldots, \mathcal{G}^{(\pi)} \) encoded by the following inheritance relationship: for each cell type \( n \in B \), let \( \pi(n) \) be the parent of type \( n \) in the tree, thus \( \mathcal{G}^{(n)} \) is a descendant of \( \mathcal{G}^{(\pi(n))} \). For a pair of networks identified by the genealogy, we assume that their topology should be similar while allowing for differences. For example, consider again Figure 1. In this case, \( \mathcal{G}^{(\text{blood stem cell})} = \mathcal{G}^{(\text{lymphoid stem cell})} = \mathcal{G}^{(\text{blood stem cell})} \), \( \mathcal{G}^{(\text{lymphoid stem cell})} = \mathcal{G}^{(\text{blood stem cell})} \), etc. Note that this framework is flexible and allows for various types of trees since each parent can have a different number of children. We assume without loss of generality that \( \mathcal{G}^{(1)} \) is the root of the tree.

Based on the GGM representation of gene networks described in the previous subsection, we have a set of GGMs whose edges (partial correlation coefficients) \( \rho^{uv} \). Ye are evolving across the genealogy. Since the partial correlations are functions of the conditions rather than constants such a model is an instance of a varying-coefficient model (Fan and Yao, 2005). Varying-coefficient models were popularized in the work of (Cleveland and Grosse, 1991) and ( Hastie and Tibshirani, 1993), and have been applied to a variety of domains to model and predict time- or space- varying response to multidimensional inputs. In our case, we are particularly interested in a certain type of parameter change: the change between zero and non-zero values between \( \rho^{uv} \) and \( \rho^{(\pi(n))v} \), also known as the structural change of the model.

The tree evolving networks described above are effective for modeling a plethora of biological processes such as the growth and reversion of cancer. A biologist may apply several treatments to a malignant cancer cell and would like to analyze the effects of the treatments on the regulatory network. The tree structure naturally expresses the dependence of the treated cells on the malignant cell without forcing the two treated cells to be identical. We explore this application in more detail later in the paper.

2.4 Estimating tree-evolving networks

When the network is tree-evolving, our goal is to learn the structure of a tree-varying GGM, which is a special case of the general varying-coefficient varying-structure (VCVS) model studied in Kolar et al. (2009). This formulation allows us to formally encode the topology of the network into the parameters \( \theta^{uv} \) of the model; for example, the absence of an edge between nodes \( u \) and \( v \) in cell type \( n \), corresponds to the partial correlation coefficient \( \rho^{uv}_{n} = 0 \).

Thus, in our formulation, recovering the structure of the \( N \) gene regulatory networks in the cell genealogy can be done by estimating \( \rho^{uv} \) for each \( 1 \leq n \leq N \). Our goal is to capture the sharp differences (i.e. edge re-wirings), rather than small correlation changes, in the tree evolving network. As a result, we concentrate on recovering the correct edge set \( \mathcal{L}^{(\pi)} \) rather than on the exact values of \( \rho^{uv} \), although these are attainable as a side product of our algorithm.

In line with this goal, we make three assumptions:

- Sparsity: most of the \( \rho^{uv} \) are zero, leading to graphs with few edges.

Note that this is technically not the pairwise potential function in a GGM.
Reconstructing tree-evolving gene networks

- Sparsity of change: the edge set \( E^{(n)} \) is similar to that of its parent \( \hat{E}^{(n+1)} \)
- Sharpness of change: there do exist a few key differences between \( E^{(n)} \) and \( \hat{E}^{(n+1)} \) that must be captured.

These assumptions hold in a wide variety of biological applications. Sparsity is usually well justified. For example, a transcription factor controls (and is controlled by) only a few genes under specific conditions (Davidson, 2001). A sparsity bias can effectively prevent estimating all elements in \( \rho^{(n)} \) to be non-zero, which leads to a meaningless complete graph. Similarly, in many biological processes the gene regulatory network in the parent cell type and the one in the child often contain only a few, but sharp differences. For example, if the parent network is a malignant cancer cell and the child networks are treated cancer cells with various drugs, we expect that the treated and cancer cells should have largely similar networks due to close developmental relationship. However, the genes that are affected by the drug should behave dramatically differently, causing a few large changes in the regulatory networks.

It is important to reiterate here that estimating networks for each cell type separately and independently is either invalid or extremely error-prone, because in common laboratory conditions only a few measurements of the gene expression are obtained, leading to either degeneracy of the likelihood function or high variance in the estimator. We overcome this problem by enabling information sharing across different cell types through a joint estimation of all networks under a loss function defined on each individual network.

To estimate \( \rho^{(1)},...\rho^{(n)} \) jointly, we adopt the neighborhood selection idea described previously, and additionally penalize the difference between the neighborhoods of adjacent cell types in the genealogy. More specifically, to recover the neighborhood of gene \( u \) for all cell types jointly, we propose the following convex optimization problem for estimating tree evolving networks.

\[
\theta^{(1)}_1, ..., \theta^{(n)}_N = \arg \min_{\theta^{(1)}_1, ..., \theta^{(n)}_N} \sum_{i=1}^{N} \sum_{t=1}^{N} (\theta^{(n)}_{uv} - \theta^{(i)}_{uv})^2 + \lambda \sum_{i=1}^{N} \sum_{t=1}^{N} \left( \| \theta^{(n)}_{uv} - \theta^{(i)}_{uv} \|_1 \right)
\]

In Equation (6), \( \hat{X}^{(n)}_{uv} \) refers to the realization of variable \( X^{(n)}_{uv} \). The \( \lambda_1 \) penalty associated with \( \lambda_1 \) enforces sparsity by setting most of the edge weights to 0 as shown in Tibsirian (1996). The total variation (TV) penalty associated with \( \lambda_2 \) enforces sparsity of difference and encourages most of the elements of \( \hat{E}^{(n)} \) to be identical to those of \( \theta^{(i)}_{uv} \) along the genealogy. However, since the \( \lambda_1 \) instead of the \( \lambda_2 \) penalty is used, outliers are not strongly penalized, allowing for large differences for a small set of edges. This allows us to have a large amount of information sharing among samples from related regulatory networks, while still allowing sharp differences to capture key changes as the network evolves.

One complication that results from the above approach is that since each neighborhood is estimated independently and because the regularization encourages some of the coefficients to be zero, the sign of \( \hat{X}^{(n)}_{uv} \) is not guaranteed to equal the sign of \( X^{(n)}_{uv} \) for finite sample sizes. This makes directly using Equation (5) to estimate the partial correlation coefficients difficult. One common way to address ‘max’ symmetrization, which is defined below.

\[
\hat{p}_{vu}^{\text{max}} = \max \{ \hat{p}_{vu}^{(n)} \}, \max \{ \hat{p}_{vu}^{(i)} \} \geq | \hat{p}_{vu}^{(n)} |, | \hat{p}_{vu}^{(i)} | < | \hat{p}_{vu}^{(n)} |
\]

We can now define our estimate of the partial correlation coefficients using Equation (5)^7.

\[
\hat{p}_{vu}^{(n)} = \text{sign}(\hat{p}_{vu}^{\text{max}}) \sqrt{\hat{X}_{vu}^{(n)} \hat{X}_{uv}^{(n)} \hat{X}_{uv}^{(n)} \hat{X}_{vu}^{(n)}}
\]

7 Note that the symmetrization may not make this a good estimate of the magnitude of \( \rho_{vu}^{(n)} \), but it is an accurate estimate of whether or not \( \rho_{vu}^{(n)} \) is positive, negative, or zero, which is all we need to recover the network structure.

The total variation penalty makes this algorithm significantly different from KELLER (Song et al., 2009a). KELLER uses kernel reweighting to recover smoothly evolving networks where the correlations between genes are changing gradually over time. However, in both the stem cell evolution and breast cancer progression-reversion problems that motivate us, the networks are evolving sharply at some points while remaining almost constant in others. For example, different microarray measurements taken from a blood stem cell renewing itself while remaining in the undifferentiated state are expected to exhibit almost the same correlations among the genes. However, once the blood stem cell evolves into a myeloid or lymphoid stem cell as shown in Figure 1, we expect there to be sharp changes in the regulatory network reflecting the new function of the more specialized cell. This sudden change can be effectively captured by the TV penalty in our algorithm but not by the kernel reweighting of KELLER. In this way, our algorithm is similar to that of TESLA (Ahmed and Xing, 2009) which also uses a TV penalty to estimate time evolving networks (a chain of graphs). However, our algorithm generalizes this idea to tree-evolving networks which are more suitable for investigating a wider range of biological processes. Algorithmically, the genealogy-induced TV penalty defines more complex constraints on the model space than that of TESLA, where network structures should be inferred. It also uses a GGM approach, and thus involves a linear regression, instead of the binary MRF approach of TESLA, which involves a logistic regression. We believe that the GGM approach, which allows for continuous measurements, is more suitable for our breast cancer application, because the sample size is small.

2.5 Optimization

We employed the CVX solver (Grant et al., 2008) provided in MATLAB to solve the underlying convex optimization problem for tree-evolving network estimation under our proposed model. At its core, CVX uses the SDPT3 solver (Toh et al., 1999). SDPT3 is an interior point method for solving cone programming problems, where the constraints are convex cones, and the objective function is linear (plus the log-barrier terms for the constraints).

For larger scale problems, one can use the method proposed by Chen et al. (2010) that uses the accelerated gradient method.

3 SIMULATION RESULTS

To assess the performance of Treegl, we evaluated its performance on simulated microarray data with a known topology of the underlying tree-evolving network. Consider the following artificial tree evolving network with \( N = 70 \):

- (1) A graph \( A \) with 30 nodes, average degree 4, and max degree 6 is generated from a Gaussian Graphical Model. For the first 10 generations, i.e., \( n = 1 \) to 10, \( A \) remains unchanged. However, we assume that each of these generations correspond to a different cell type in the genealogy (for reasons that will be made clear later).
- (2) After \( n = 10 \), the graph branches into two child graphs, \( B \) and \( C \). To generate each child graph, 25% of the edges are randomly deleted and the same number are randomly added. This represents a sharp, sparse change in the network. These child graphs stay unchanged for another 10 generations (\( n = 11 \) to 20 for \( B \), \( n = 21 \) to 30 for \( C \)). Again, each generation indicates a different cell type.
- (3) \( B \) and \( C \) then branch further. 25% of the edges are randomly removed/added to generate graphs \( D \) and \( E \) from \( B \), and \( F \) and \( G \) from \( C \). The resulting graphs then stay constant for another

Note that our algorithm does not know at which points the network structure changes. The goal is to examine if it can detect the change points as well as take advantage of the samples that come from cell types with identical structure between the change points.

To evaluate Trueg, we plot a ROC curve showing the recall for different values of precision. Precision = $1/N \sum_{i=1}^{N} \frac{p_i}{\hat{p}_i}$, and Recall = $\frac{\hat{p}_i}{\sum_{i=1}^{N} p_i}$.

To produce the curve, cross validation is used to select $\lambda_1$ and $\lambda_2$. A threshold $\theta$ is then varied from the smallest absolute edge weight to the largest absolute edge weight. An edge is included in the network if and only if its edge weight is greater than $\theta$ (in absolute value). We calculate precision/recall for a large number of values of $\theta$ and produce the curve. To average different trials, we used binning, averaging points using a bin width of 0.05.

The results are shown in Figure 2 for two different sample sizes. Our method (in blue) performs favorably to estimating a single static network (green) or estimating each graph independently (red). It should be noted that our method can produce different graphs compared to the static method which only produces one. The independent method also produces different graphs but it performs very poorly.

4 AN APPLICATION TO BREAST CANCER DATA

We now demonstrate an application of our algorithm to the study of progression and reversion of breast cancer cells. Pioneered by Dr. Mina Bissell’s research team, functional analysis of physiologically more realistic 3D culture models of breast cancer has yielded a wealth of insight into the mechanisms of cancer development (Petersen et al. 2007; Weaver et al. 1992). From tumor cells cultured in 3D matrices, it was found that microenvironmental factors and signaling inhibitors have a dramatic influence on the growth dynamics and malignancy of the cells (Boh et al., 2007; Weaver et al., 1997). Further, tumorigenicity of breast cancer cells is tightly linked to the integrity of their acinar structures (Petersen et al., 1992). However, except for a sketchy outline, little is known about how the cells interpret signaling cues from their surroundings and selectively regulate genes in a temporal-spatially specific manner.

Our goal is to investigate the gene regulatory networks of normal breast cells (S1 cells), malignant breast cancer cells (T4 cells), and non-tumorigenic breast cancer cells reverted by different drugs (T4R cells). The exact treegenalogy underlying these cell-type specific networks is shown in Figure 3: S1 cells with polarized acinar structures evolve into tumorigenic T4 cells which form disorganized apolar colonies, and then three drugs are applied individually to T4 cells and different reverted cells (T4R) with organized structures which resemble S1 cells are produced.

4.1 Experimental setup

We have 15 microarray measurements of 22,000 genes detailed below, that we grouped into five categories of three samples each (based on their similarities): three samples of S1 cells, three samples of T4 cells, three samples of T4R cells reverted by MMP inhibitors (PI3K-MAPKK-T4R), three samples of T4R cells reverted by either PI3K or MAPKK inhibitors (PI3K-MAPKK-T4R) and three samples of T4R cells reverted by either EGFR or integrin p1 inhibitors (EGFR-ITGB1-T4R).

Our experimental procedure started with feature selection to reduce noise. Since some probes on Affymetrix arrays have multiple replicates, we combined measurements from these probes by taking the median, which resulted in 12,977 unique genes. Next, for each gene we calculated its median fold ratios of expression levels among each pair of the five groups of cells. If any of the fold ratios for a gene was greater than 1.3, it was selected for the next step. We picked 5440 genes using this criterion.

Then, we applied Trueg to the 5440 genes. In addition to taking advantage of the similarity between the T4 and T4R networks, we also explicitly penalize the difference between the S1 and T4R networks, since the T4R networks are expected to lie somewhere in between the S1 and T4. As a result, we add extra TV penalty terms between the T4R and S1 to enforce this intuition (dotted lines in Fig. 3). These extra penalty terms are assigned the same parameter $\lambda_2$ as the other total variation penalties. The new optimization problem is given below ($n=1$ corresponds to S1, $n=2$ corresponds to T4, and $n=3,4,5$ correspond to the T4R).

$$\begin{align*}
\hat{\theta}^{(1)}_i, \ldots, \hat{\theta}^{(n)}_i &= \arg \min_{\theta \in \mathbb{R}} \frac{1}{N} \sum_{i=1}^{N} \left( \sum_{x \in \Omega} x \sum_{x \in \Omega} x - \theta_{ij} \right)^2 \\
&+ \lambda_1 \sum_{i=1}^{N} \left( \theta_i^{(S1)} - \theta_i^{(T4)} \right)^2 \\
&+ \lambda_2 \sum_{i=1}^{N} \left( \theta_i^{(T4R)} - \theta_i^{(S1)} \right)^2 \\
&+ \lambda_2 \sum_{i=1}^{N} \left( \theta_i^{(T4R)} - \theta_i^{(T4)} \right)^2 \\
&\text{s.t.} \quad \theta_i^{(S1)} = \theta_i^{(T4)} = \theta_i^{(T4R)} = \theta_i^{(n)} = 0.
\end{align*}$$

All results described here are with the parameter settings of $\lambda_1=4$ and $\lambda_2=2$.
Reconstructing tree-evolving gene networks

5 ANALYSIS OF RESULTS

5.1 Results overview

Figure 4 gives an overview of all the recovered networks using Cytoscape (Shannon et al., 2003). As one can see the networks exhibit many different topologies reflecting their underlying biological differences. To shed more light on these differences, Figure 5 shows the interactions among the second level GO groups in the recovered networks. The thickness of a link between two groups is proportional to the number of edges present between genes that are members of these GO groups. T4 cells display increased activities in cell proliferation and signaling, both indicative of their malignant state, compared to S1 cells. The T4R cells lie somewhere in between: MMP-T4R cells tend to have only a few interactions, since the network is quite sparse. While both the PI3K-MAPKK-T4R and EGFR-ITGB1-T4R networks show reduced activities in growth and locomotion compared to S1 cells, the former network has more activities in cell proliferation and reduced signaling than the latter one. Taken together, these data suggest that although T4 cells can be morphologically reverted back to the normal-looking T4R cells, the underlying molecular mechanisms in the reverted cells are different from those in either S1 or T4 cells.

5.2 GO analysis of networks

Next, we performed GO analysis to discover significantly enriched functional groups specific to each network. Our results are illustrated in Table 1.

Our data shows that highly enriched GO groups in S1 cells correspond to metabolic processes or other housekeeping functions, such as cellular respiration and DNA replication, reflecting the normal nature of these cells. On the other hand, T4 cells are enriched with genes involved in cell proliferation, growth factor activity, intracellular signaling cascade, angiogenesis and actin binding group, all of which are known to play important roles in T4 as well as other cancer cells (Hanahan et al., 2000; Liu et al., 2004; Wang et al., 2002; Weaver et al., 1997). These results show that our algorithm is able to reveal what has already been known about S1 and T4 cells, and thus demonstrate the validity of our method.

Since little is known about T4R cells, we next examined the networks of the different T4R cells to gain more insight into these reverted cells. Our results show that the MMP-T4R network, like S1 cells, contains many enriched GO groups involved in metabolic processes, such as fatty acid and cofactor metabolic processes. On the other hand, however, the PI3K-MAPKK-T4R network contains genes involved mainly in post-translational protein modification, chromatin modification, thiolester hydrolase activity and vacuole, while the EGFR-ITGB1-T4R network is predominantly overrepresented with genes participating in chromatin modification, cytoskeletal protein binding, intracellular junctions, among others. These data therefore suggest that at the molecular level T4R cells are indeed different from S1 and T4 cells, as well as from one another.

5.3 Analysis of Hubs in the T4 network

Finally, to identify potential novel drug targets in T4 cells, we examined several hubs which have high degrees as well as their neighborhood genes in these cells. Figure 6 shows the subnetworks of 5 hubs: ANXA3, CA9, HSF2BP, PTGS2 and SCG5. As expected, many of the functional gene groups enriched in the subnetworks reflect our intuition that these hubs interact closely with genes influential in cancer.

(1) ANXA3 (degree: 61)—encodes a protein belonging to the annexin family, and is known to play a role in the regulation of cell growth and is thought to be a biomarker of cancer (Jung et al., 2010). In the ANXA3-subnetwork, it interacts with a...
number of genes related to cell proliferation, growth factor activity, and the MAP kinase signaling pathway, the latter of which is known to be one of the key signaling pathways in T4 cells (Liu et al., 2004).

(2) CA9 (degree: 37)—encodes carbonic anhydrase IX. It has been implicated in cell proliferation, and has been found to be important in renal cell carcinoma (Jubb et al., 2004). We see that CA9’s neighborhood consists of genes involved in cell proliferation, the MAP kinase signaling pathway, golgi apparatus part, and transcription factor activity.

(3) HSF2BP (degree: 80)—encodes heat shock transcription factor binding protein. Like the previous two hubs, HSF2BP has neighbors related to cell proliferation and the MAP kinase signaling pathway. It also has neighbors related to ‘response to wounding’ which is known to be linked with tumorigenesis and tumor development (Chang et al., 2005; Fukumura et al., 1998).

(4) PTGS2 (degree: 88)—encodes prostaglandin-endoperoxide synthase 2, which is a key enzyme in prostaglandin biosynthesis. Previous evidence suggests that it is associated with risk of breast cancer (Langsenlehner et al., 2006). Again, we see neighbors participating in similar activities to the previous hubs, such as cell proliferation and wound healing. Another interesting group is cell motility which suggests that the subnetwork of PTGS2 potentially plays a role in tumor cell spread (Yamazaki et al., 2005).

(5) SCG5 (degree: 78)—encodes secretogranin V, which has been found to be involved in medullary carcinoma (Marcinkeiwicz et al., 1988) as well as human lung cancer (Roebrock et al., 1989). Again many of its neighbors are involved in cell proliferation, response to wounding, and cell motility. Another interesting group of neighbors is those related to GTPase activity; as ras oncogenes happen to be members of the family of GTPases (Sahai and Marshall, 2002), this group of genes may also have activities implicated in cancer.

In summary, these results suggest that hubs with high degrees in the T4 network contribute to the growth, proliferation, and malignancy of T4 cells, and thus may serve as potential novel targets for breast cancer treatment.

6 DISCUSSION AND CONCLUSION

Statistically and algorithmically, the problem of estimating tree-evolving networks from multiple biological systems in the genealogy simultaneously, as solved by TreeGl, is fundamentally different from estimating multiple networks separately from every cell type, or estimating a single ‘average’ network from samples pooled from all cell-types (or all cell stages) in the genealogy and subsequently ‘trace-out’ active subnetworks corresponding to each cell-type from the average network (Luscombe et al., 2004), which are common practices in current system biology community. The latter two approaches either directly or indirectly assume that

Table 1. Significantly enriched GO groups found in networks for (a) S1, (b) T4, (c) MJD-T4R, (d) PI3K-MAPK-T4R, and (e) EGFR-ITGB1-T4R

<table>
<thead>
<tr>
<th>GO group</th>
<th>score</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Note that the T4 network contains many groups related to cell proliferation, growth, and angiogenesis, while the S1 network does not (see text for details).

Fig. 6. Neighborhoods of a few high-degree hubs in T4 cells. A few enriched GO groups are highlighted in the subnetworks as shown.
the network in question is a static one, and samples of nodal states, such as microarray measurements of gene expressions are i.i.d. within or (when pooled) across cell types. In reality, such an assumption is not only biologically invalid, but is statistically unsubstantiated and hard to leverage. First, such an assumption can lead to severe underuse of the data, and makes an already serious curse-of-dimensionality problem even harder for the following reason. Typically, in many gene expression profiling experiments, especially those from biomedical studies, the size of the sample can be extremely small (e.g. often 2–3 replica per condition or specimen) compared to the number of genes (typically 10³–10⁶ for human) due to the difficulty of procuring many samples in laboratory experiments, which makes the directly estimated network over these genes extremely unreliable. In reality, these different cell types at different positions in the genealogy should not be drastically different, and one should expect that samples from closely related types may offer additional information to the cell type in question. Thus, estimating each point in the genealogy independently using a static reverse engineering algorithm would be largely ineffective, because there is not enough data and there are too many variables. Next, due to the presence of the genealogy that related all cell-type-specific networks, the samples from all types are not identically distributed. Therefore when naively pooling them together to obtain an average network, the result may suffer from high variance, since the regulatory network could change significantly from the beginning to the end of the genealogy. The TreeGP algorithm elegantly couples all the inference problems pertained to each network in the beginning to the end of the genealogy. The TreeGP algorithm cleverly avoids the curse-of-dimensionality problem even harder for the following reasons.

To demonstrate our method, we applied our algorithm to a microarray dataset obtained from a progression and reversion series of breast cancer cells. Our results showed that we not only were able to identify previously known molecular signatures specific to different cell types, but also that we could provide deeper insight into the gene interactions. Such a Bayesian network which is directed and can offer causal insight into the gene interactions.

ACKNOWLEDGEMENTS

We are grateful to Drs Mina Bissell and Ren Xu for their guidance as well as providing us the cancer dataset.

Funding: This research was made possible by (National Science Foundation DB1-0546594, IIS-0713379); National Institutes of Health (1R01GM093156), and an Alfred P Sloan Fellowship (to E.P.X.).

Conflict of Interest: none declared.

REFERENCES


Curtis,R.E. et al. (2011) TVNViewer: an interactive visualization tool for exploring networks that change over time or space. Bioinformatics, in press [Ishp ahead of print; doi: 10.1093/bioinformatics/BTR273].


