Dynamic Bayesian networks for interventional data

Simon E F Spencer [1], Steven M Hill [2] and Sach Mukherjee [2]

[1]Department of Statistics, University of Warwick, Coventry CV4 7AL.
[2]Netherlands Cancer Institute, Amsterdam, The Netherlands.

Abstract

Graphical models are widely used to study biological networks. Interventions on network nodes often form part of experimental designs for the study of biological networks. In this paper, we discuss the modelling of interventional time-course data using graphical models known as dynamic Bayesian networks. Our work is motivated by, and illustrated in the context of, protein signalling networks. We show empirical results that demonstrate the need to carefully model interventions when interventions form part of the design.
Contact: s.e.f.spencer@warwick.ac.uk

1 Introduction

Network inference approaches are widely used to shed light on biological networks, including gene regulatory and signalling networks. Since processes underlying such networks are dynamical in nature, time-course data can help to elucidate regulatory interplay. Network inference methods for time-course data have been investigated in the literature, with contributions including (among many others) Husmeier (2003); Bansal *et al.* (2006); Hill *et al.* (2012).

Interventions, for example gene knockouts, knockdowns by RNA-interference (RNAi) or drug inhibition of kinases, play an important role in experimental designs aimed at elucidating network structure. This is due to the fact that association does not imply causation: interventions are needed to understand whether a given node has a causal influence on another as opposed to merely being co-expressed. As data acquisition costs fall, interventional time-course designs are becoming more common.

Network inference often has the goal of generating testable hypotheses regarding biological interplay. For example, prediction of an edge from node A to node B might be validated by intervening on node A (e.g. by inhibition, knockout or knockdown, possibly in combination with intervention on other nodes) and observation of the effect on node B. Such hypotheses are causal hypotheses, since they concern effects under intervention, and not simply correlation between nodes. In this sense, biological network inference can have a causal aspect.

From the perspective of causal inference (Pearl, 2000, 2009), interventional data require special treatment, because the intervention modifies the causal graph and thereby the likelihood. However, existing methods for network inference from time-course data, including conventional dynamic Bayesian network (DBNs) and their variants, do not account for interventional designs.

In this paper, we put forward an approach to network inference from time-course data that takes account of interventional designs. We proceed within a graphical models framework, using in particular DBNs. This is done by modifying the statistical formulation for those experimental samples in which interventions were carried out, following related literature in causal inference, including Pearl (2000) and Eaton and Murphy (2007). Specifically, for experiments in which interventions were carried out, we modify the structure of the underlying graphical model and explore various parameterizations of the effect of the intervention.

A DBN is a Bayesian network (BN) with an explicit time index (for a detailed treatment see Hill *et al.* (2012) or Murphy (2002)). For multivariate time-course data $\mathbf{X}$, a DBN gives a likelihood $p(\mathbf{X}|G, \boldsymbol{\theta})$ that depends on a graph G and parameters $\boldsymbol{\theta}$ (details of the DBN likelihood appear below). The graph G describes the network edge structure (also referred to as the "topology" in the bioinformatics literature, we use both terms interchangeably) and is the focus of inference in this paper. Let $\mathbf{X}_c$ denote the full set of time course data obtained under experimental condition c. If the conditions are simply replicates, the indices c run over instances of the same experimental design; then the overall likelihood of the complete dataset $\{\mathbf{X}_c\}$ is the familiar product $\prod_c p(\mathbf{X}_c|G, \boldsymbol{\theta})$. However, if the design for experiment c includes interventions, the likelihood itself should be modified in accordance with the nature of the intervention (Pearl, 2000; Eaton and Murphy, 2007). Such modification may include changes to the underlying graphical model structure and/or changes to model parameters; this yields a modified likelihood $p_c(\mathbf{X}_c|G, \boldsymbol{\theta}_c)$. Network inference then proceeds using the overall likelihood $\prod_c p_c(\mathbf{X}_c|G, \boldsymbol{\theta}_c)$. Precisely how the model and/or parameters should be modified for interventional conditions is the main focus of this paper.

We illustrate our approach in the context of protein signalling networks. In this setting, we consider in detail the case of intervention by kinase inhibitors. We illustrate the methods on both simulated and real protein signalling data, exploring the behaviour of a number of approaches by which to model interven-

tions, and comparing their performance with respect to network reconstruction. We find that in the presence of interventional data, analyses that do not account for interventions do not perform well.

Related work on incorporating interventions into network inference include Bansal *et al.* (2006) and, in a more general setting, Eaton and Murphy (2007) for discrete data. Frölich *et al.* (2008) extend the nested effects models of Markowetz *et al.* (2005) to accommodate continuous observations. Here we focus on continuous protein concentration measurements over multiple timecourse experiments. Recently, Bender *et al.* (2010) presented an approach based on hidden Markov models that applies in the setting of interest here. Finally Terfve *et al.* (2012) described CellNetOptimizer – a collection of tools that utilize Boolean logic networks to refine an existing network model. Two tools in particular, CNOdt and CNOode, are appropriate for timecourse data.

The remainder of the paper is structured as follows: first a Bayesian framework for network inference using DBNs is outlined in Sections 2.1.1 to 2.1.4. Next, the interventional models that constitute our main focus, are described in Section 2.2. We illustrate some of key points of the approach using examples in Sections 2.4. Empirical results appear in Section 3. We close with discussion of open questions and future prospects.

2 Methods

We first fix ideas and notation by first reviewing the "classical" DBN formulation (without interventions). We then go on to discuss in detail how the likelihood can be modified to account for interventions. Taken together, this gives an overall approach by which to infer networks from data that includes interventions.

2.1 Dynamic Bayesian network model

A DBN uses a graph to describe probabilistic relationships between variables through time, with associated parameters specifying the temporal relationships. Following (Friedman *et al.*, 1998; Husmeier, 2003), we consider DBNs with edges forward in time only (i.e. first-order Markov with no within-time-slice-edges) and assume stationarity in the sense that neither network topology nor parameters change through time. Then, each variable at time t depends only on its regulators at the previous time step. Further, since the graph structure does not change with time and edges are always forward in time, the topology can be described by a graph G with exactly one vertex for each protein under study

and edges understood to mean that the child depends on the state of the parent at the previous time-step.

2.1.1 Statistical formulation

Let $x_{j,c,t}$ denote log-expression of variable $j \in \{1 \dots p\}$, at time $t \in \{1 \dots T\}$ in the time-course obtained under experimental conditions $c \in \mathcal{C}$. The edge set of the graph G is $E(G)$. Let $\gamma^{(j)} = \{i : (i,j) \in E(G)\}$ denote the set of parents for node j. Then, for conditions c without intervention, the DBN model for node j is

$$x_{j,c,t} = \begin{cases} \alpha_1^{(j)} + \sum_{i \in \gamma^{(j)}} x_{i,c,t-1} \beta_i^{(j)} + \epsilon_{j,c,t} & t > 0, \\ \alpha_2^{(j)} + \epsilon_{j,c,0} & t = 0. \end{cases} \tag{1}$$

where, $\beta_i^{(j)}$ denotes parameters that govern the dependence on parent nodes in the previous time step, $\alpha_1^{(j)}, \alpha_2^{(j)}$ are coefficients that do not depend on the parent set $\gamma^{(j)}$ and $\epsilon_{j,c,t} \sim N(0, \sigma_j^2)$ is a noise term. The use of two intercept parameters, one for the initial time point, allows the model more flexibility to incorporate the unmeasured effects of the parents acting on the first observations.

2.1.2 Variable selection

Under the stationarity and Markov assumptions above, there is a close relationship between inference concerning the DBN graph G and variable selection for the above regression formulation. As discussed in detail in Hill *et al.* (2012), exploiting this connection allows efficient inference regarding the graph G. Specifically, if $\mathrm{P}(i \in \gamma^{(j)}|\mathbf{X})$ is the posterior probability that variable i appears in the regression model for variable j above (i.e. the posterior inclusion probability in the variable selection sense) and assuming a graph prior $\mathrm{P}(G)$ that is decomposable over nodes, we have

$$\begin{aligned} \mathrm{P}((i,j) \in E(G)|\mathbf{X}) &= \mathrm{P}(i \in \gamma^{(j)}|\mathbf{X}) \\ &= \sum_{\gamma \in \mathcal{M}} I(i \in \gamma) \mathrm{P}(\gamma^{(j)} = \gamma|\mathbf{X}) \end{aligned} \tag{2}$$

where $\mathcal{M}$ denotes the set of all possible variable subsets and I is the indicator function.

Thus, due to the structure of the DBN model, for estimation of posterior probabilities of edges in graph G it suffices to perform variable selection for each node in turn, with variables at the previous time point considered as potential predictors. To ease notational burden we leave implicit dependence on node j

in the following sections. Let $\mathbf{x} = \{x_{j,c,t}\}$ denote all observations for protein j; we assume there are total of n such observations. Then, model (1) can be written as

$$\mathbf{x} \quad = \quad \boldsymbol{X}_0\boldsymbol{\alpha} + \boldsymbol{X}_\gamma\boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{3}$$

where $\boldsymbol{\epsilon} \sim N_n(\mathbf{0}_n, \sigma^2\boldsymbol{I}_n)$, N_n denotes the n-dimensional multivariate Normal density, $\mathbf{0}_n$ is the n dimensional vector of zeros and $\boldsymbol{I}_n$ is the $n \times n$ identity matrix. The design matrix is split into two parts: $\boldsymbol{X}_0 = \left[\mathbf{1}_{\{t>0\}}\mathbf{1}_{\{t=0\}}\right]_{n\times 2}$, which is the same for every model and has parameter vector $\boldsymbol{\alpha}$; and $\boldsymbol{X}_\gamma$ which depends on the choice of parents given by γ and has parameter vector $\boldsymbol{\beta}$. Let a be the length of $\boldsymbol{\alpha}$ and b be the length of $\boldsymbol{\beta}$, then $\boldsymbol{X}_0$ has dimension $n \times a$ and $\boldsymbol{X}_\gamma$ has dimension $n \times b$. In the absence of interventions, observations of the parent proteins from the previous timepoint form the columns of $\boldsymbol{X}_\gamma$ (we discuss interventions below). For the first observation, where there are no previous observations, zeros are inserted into $\boldsymbol{X}_\gamma$ in the place of the parent observations.

We can assume without loss of generality that the two parts of the design matrix ($\boldsymbol{X}_0$ and $\boldsymbol{X}_\gamma$) are orthogonal, ie. $\boldsymbol{X}_0^T\boldsymbol{X}_\gamma = \mathbf{0}_{a\times b}$. This reparameterisation ensures the predictors have mean zero, for details see Appendix A.

2.1.3 Marginal likelihood

The marginal likelihood $p(\mathbf{x}|\gamma)$ for node j is obtained by marginalising over all model parameters, that is $p(\mathbf{x}|\gamma) = \int p(\mathbf{x}|\boldsymbol{\theta},\gamma)p(\boldsymbol{\theta}|\gamma)\,\mathrm{d}\boldsymbol{\theta}$ where $\boldsymbol{\theta} = (\boldsymbol{\alpha},\boldsymbol{\beta},\sigma)$ is the full set of model parameters. We make use of widely-used parameter priors from the Bayesian literature (Denison *et al.*, 2002). Firstly, we assume improper priors for $\boldsymbol{\alpha}$ and σ, namely that $p(\boldsymbol{\alpha},\sigma|\gamma) \propto \frac{1}{\sigma}$ for $\sigma > 0$. Note that as this prior is improper, for meaningful comparisons to be made between models in $\mathcal{M}$ this prior must be the same for all of the models. Secondly, we assume Zellner's g-prior for the regression coefficients, so that $\boldsymbol{\beta}|(\boldsymbol{\alpha},\sigma,\gamma) \sim N_b\left(\mathbf{0}_b, g\sigma^2(\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma)^{-1}\right)$. Following Smith and Kohn (1996); Kohn *et al.* (2001), we set $g = n$. With this prior the covariance matrix for $\boldsymbol{\beta}$ is proportional to $(\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma)^{-1}$ which has some nice properties, for example invariance to rescaling of the columns of $\boldsymbol{X}_\gamma$ (Smith and Kohn, 1996). Using standard results (Denison *et al.*, 2002), the marginal likelihood is then given in closed form as,

$$p(\mathbf{x}|\gamma) \quad = \quad \frac{C}{(n+1)^{b/2}}\left(\mathbf{x}^T(\boldsymbol{I}_n - \boldsymbol{P}_0 - \tfrac{n}{n+1}\boldsymbol{P}_\gamma)\mathbf{x}\right)^{-\frac{n-a}{2}} \tag{4}$$

where $\boldsymbol{P}_0 = \boldsymbol{X}_0(\boldsymbol{X}_0^T\boldsymbol{X}_0)^{-1}\boldsymbol{X}_0^T$, $\boldsymbol{P}_\gamma = \boldsymbol{X}_\gamma(\boldsymbol{X}_\gamma^T\boldsymbol{X}_\gamma)^{-1}\boldsymbol{X}_\gamma^T$ and the normalising constant $C = \frac{1}{2}\Gamma(\frac{n-a}{2})\pi^{-(n-a)/2}|\boldsymbol{X}_0^T\boldsymbol{X}_0|^{-1}$.

We also wish to consider the model in which $b = 0$, which corresponds to $\gamma = \emptyset$. The regression equation is simply $\mathbf{x} = \boldsymbol{X}_0\boldsymbol{\alpha} + \boldsymbol{\epsilon}_p$ and the marginal likelihood is given by $p(\mathbf{x}|\gamma = \emptyset) = C\left(\mathbf{x}^T(\boldsymbol{I}_n - \boldsymbol{P}_0)\mathbf{x}\right)^{-(n-a)/2}$.

2.1.4 Model prior

Following Scott and Berger (2010) we include a multiplicity correction to properly weight models in light of number of possible parent sets. Since there are $\binom{p}{k}$ models for node j with k parents, the prior probability is chosen so that $\mathrm{P}(\gamma^{(j)}) \propto \binom{p}{k}^{-1}$.

2.1.5 Computation

Combining the marginal likelihood (4) and model prior $\mathrm{P}(\gamma^{(j)})$ gives the posterior $\mathrm{P}(\gamma^{(j)}|\mathbf{x}) \propto p(\mathbf{x}|\gamma^{(j)})\mathrm{P}(\gamma^{(j)})$ over parent sets (so far, for the no-intervention case). Then, posterior probabilities $\mathrm{P}((i,j) \in G|\mathbf{x})$ for individual edges in the graph are obtained directly from the posterior over parent sets by Eq. 2. As discussed in detail in Hill *et al.* (2012), placing a bound m on graph in-degree, following common practice in structural inference for graphical models (e.g. Husmeier, 2003), allows exact computation of the posterior scores.

2.2 Modelling interventions

In statistical terms, interventions may alter the edge structure of the graphical model, or model parameters, or both. Here, we discuss the modelling of interventional data in the context of the DBN model outlined above. For experimental conditions c that involve an intervention, the approaches below give different approaches by which to form the likelihood $p_c(\mathbf{x}|G, \theta_c)$ for an interventional experiment c. We first give a general typology of interventions following Eaton and Murphy (2007), with extensions to accommodate the wide range of interventions seen in biological experiments, and then go on to discuss kinase inhibition in detail.

2.2.1 Approaches for modelling interventions

In a *perfect intervention* the target of the intervention is treated as known and certain edges that the target node participates in are removed. We call an intervention that corresponds to removal of edges leading out of the target node

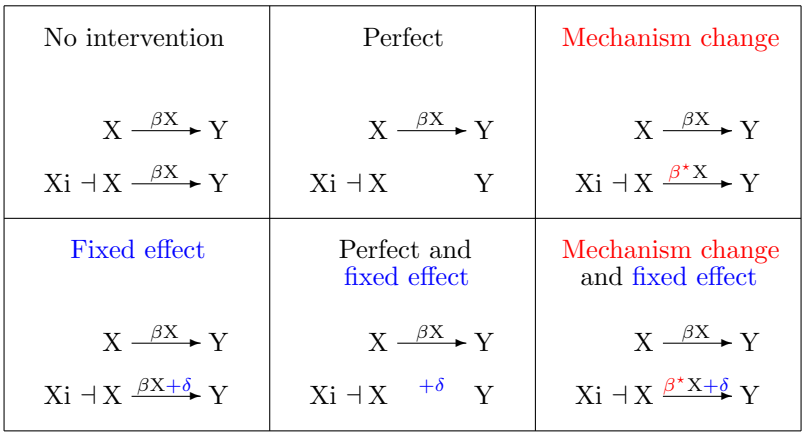

Figure 1: Diagrammatic representation of intervention models in their '-out' forms for an inhibitor Xi that targets X (see text for details).

perfect-out interventions and one that corresponds to removal of edges leading in to the target node a *perfect-in* intervention. For example, a knockout with known target gene j can be thought of as externally setting the transcription level of node j to zero. This removes the causal influence of other nodes on j and therefore constitutes a perfect-in intervention. However, since the change to j may have causal influences on other nodes, outgoing links are allowed to remain.

In a *mechanism change intervention* the structure of the graph remains unchanged, but parameters associated with edges that the target participates in are allowed to change. In a *mechanism-change-out* intervention, parameters are re-estimated for the case where the target is a parent; in a *mechanism-change-in* intervention parameters are re-estimated when the target is the child.

In a *fixed-effect intervention*, the effect of the inhibitor is modelled by an additional, additive parameter in the regression equation. In a *fixed-effect-in* intervention the effect appears in the equation for the target itself, while in a *fixed-effect-out* intervention the effect appears in the equations for the children of the target. As we discuss further below, these formulations can be useful in certain settings where the intervention results in a change in the average level of the target or its causal descendants.

These intervention models can be used in combination (see Figure 1) to reflect understanding of the biological action of the interventions. The perfect and mechanism change intervention models cannot be used together as this would introduce a column of zeros into the design matrix.

2.2.2 Example: intervention by kinase inhibition

We now discuss in detail a specific type of inhibition, namely drug inhibition of kinases, as used in studies of protein signalling. This example illustrates the need to consider the mechanism of the intervention in selecting amongst the interventional formulations outlined above. Kinase inhibition blocks the kinase domain of the target, removing the ability of the target to enzymatically influence other nodes. However, such inhibitors may not prevent phosphorylation of the target itself. Therefore, we focus on "-out" interventions for modelling kinase inhibitors.

As we argue below, perfect interventions in combination with fixed effect interventions are well suited to modelling kinase inhibition using log-transformed data, since they capture the blocking of enzymatic ability and also allow estimation of the quantitative effect of inhibition on child nodes. Recalling the regression equation for a node j with parents γ (Eq. 3), we see that $\boldsymbol{X}_\gamma$ is the parent-specific design matrix. In combined perfect and fixed effect interventions, a zero is inserted in the design matrix when a parent protein is inhibited and for these observations an additive parameter is also estimated. Inserting a zero into $\boldsymbol{X}_\gamma$ is equivalent to forcing the protein under inhibition to take its mean value (as we have orthogonalised $\boldsymbol{X}_\gamma$ to the intercepts $\boldsymbol{X}_0$, which causes the columns of $\boldsymbol{X}_\gamma$ to have mean zero). The fixed effect term appears as a binary column in the design matrix, (with ones in rows corresponding to conditions where the inhibitor was active) and an additional parameter in the vector $\boldsymbol{\beta}$. This allows data-driven estimation of the effect of inhibition on the children of the inhibitor's targets. This effect must be estimated when working with relative log-concentrations, since the locations of specific values on the scale are lost.

The regression equation for perfect fixed effect interventions is therefore

$$\mathbf{x} = \boldsymbol{X}_0\boldsymbol{\alpha} + (\boldsymbol{I}_\gamma \cdot \boldsymbol{X}_\gamma)\,\boldsymbol{\beta} + \sum_{i \in \mathcal{I}_\gamma} \mathbf{1}_i \delta_i + \boldsymbol{\epsilon},$$

where $\boldsymbol{I}_\gamma$ is an $n \times |\gamma|$ binary matrix with entry (i, j) equal to 0 if the jth parent in γ is inhibited in sample i and 1 otherwise; $\boldsymbol{A} \cdot \boldsymbol{B}$ denotes the element-wise product of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$; $\mathcal{I}_\gamma$ is the set of inhibitors that target nodes in γ; and $\mathbf{1}_i$ is a vector with jth entry 1 if inhibitor i is active in sample j and 0 otherwise. The matrix $\boldsymbol{I}_\gamma$ accounts for the perfect interventons and the δ_i (for $i \in \mathcal{I}_\gamma$) are the interventional fixed effects.

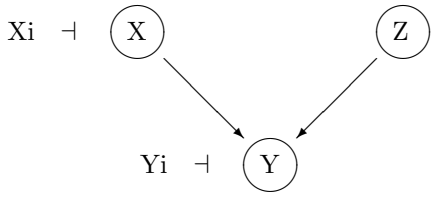

Figure 2: An example protein network showing the action of inhibitors Xi and Yi on proteins X, Y and Z.

2.3 A toy example

We now illustrate the foregoing interventional models and the modifications to the DBN that they induce by considering a toy, three-node protein phosphorylation example. The causal graph G is illustrated in Figure 2: Proteins X and Z act as kinases for protein Y, with kinase inhibitors against X and Y ("Xi" and "Yi" respectively) available. Experimental conditions include no inhibitors, inhibitor Xi only, inhibitor Yi only and both inhibitors together. Below we show how the model for Y, conditional on parent set $\gamma^{(Y)} = \{X, Z\}$, is modified under various intervention schemes. The vector observations $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ with components indexed by $c \in \{0, \mathrm{Xi}, \mathrm{Yi}, \mathrm{Xi}+\mathrm{Yi}\}$ are taken to be phospho-protein levels, with $\boldsymbol{x}, \boldsymbol{z}$ observed at a certain time t and $\boldsymbol{y}$ at the following time $t+1$, such that the former are potential predictors for the latter. Without accounting for interventions we have the familiar model for $\mathbf{y}$ (as in Eq. 3; here rows in the design matrix correspond to experiments c and columns to the parents)

$$\begin{bmatrix} \boldsymbol{y}_0 \\ \boldsymbol{y}_{\mathrm{Xi}} \\ \boldsymbol{y}_{\mathrm{Yi}} \\ \boldsymbol{y}_{\mathrm{Xi+Yi}} \end{bmatrix} = \boldsymbol{X}_0\boldsymbol{\alpha} + \begin{bmatrix} \boldsymbol{x}_0 & \boldsymbol{z}_0 \\ \boldsymbol{x}_{\mathrm{Xi}} & \boldsymbol{z}_{\mathrm{Xi}} \\ \boldsymbol{x}_{\mathrm{Yi}} & \boldsymbol{z}_{\mathrm{Yi}} \\ \boldsymbol{x}_{\mathrm{Xi+Yi}} & \boldsymbol{z}_{\mathrm{Xi+Yi}} \end{bmatrix} \boldsymbol{\beta} + \boldsymbol{\epsilon}.$$

With perfect(-out) interventions are assumed, this equation becomes

$$\begin{bmatrix} \boldsymbol{y}_0 \\ \boldsymbol{y}_{\mathrm{Xi}} \\ \boldsymbol{y}_{\mathrm{Yi}} \\ \boldsymbol{y}_{\mathrm{Xi+Yi}} \end{bmatrix} = \boldsymbol{X}_0\boldsymbol{\alpha} + \begin{bmatrix} \boldsymbol{x}_0 & \boldsymbol{z}_0 \\ \mathbf{0} & \boldsymbol{z}_{\mathrm{Xi}} \\ \boldsymbol{x}_{\mathrm{Yi}} & \boldsymbol{z}_{\mathrm{Yi}} \\ \mathbf{0} & \boldsymbol{z}_{\mathrm{Xi+Yi}} \end{bmatrix} \boldsymbol{\beta} + \boldsymbol{\epsilon},$$

since inhibitor Xi prevents X from influencing the phosphorylation state of Y. Since the inhibitor Yi targets the kinase domain of Y, but may nonetheless allow phosphorylation of Y, the model does not force Y to be unphosphorylated.

For fixed effect interventions the parameter vector $\boldsymbol{\beta}$ is extended to include an

additional parameter:

$$\begin{bmatrix} \boldsymbol{y}_0 \\ \boldsymbol{y}_{\mathrm{Xi}} \\ \boldsymbol{y}_{\mathrm{Yi}} \\ \boldsymbol{y}_{\mathrm{Xi+Yi}} \end{bmatrix} = \boldsymbol{X}_0\boldsymbol{\alpha} + \begin{bmatrix} \boldsymbol{x}_0 & \mathbf{0} & \boldsymbol{z}_0 \\ \boldsymbol{x}_{\mathrm{Xi}} & \mathbf{1} & \boldsymbol{z}_{\mathrm{Xi}} \\ \boldsymbol{x}_{\mathrm{Yi}} & \mathbf{0} & \boldsymbol{z}_{\mathrm{Yi}} \\ \boldsymbol{x}_{\mathrm{Xi+Yi}} & \mathbf{1} & \boldsymbol{z}_{\mathrm{Xi+Yi}} \end{bmatrix} \boldsymbol{\beta} + \boldsymbol{\epsilon}.$$

For perfect interventions with fixed effect interventions the model becomes:

$$\begin{bmatrix} \boldsymbol{y}_0 \\ \boldsymbol{y}_{\mathrm{Xi}} \\ \boldsymbol{y}_{\mathrm{Yi}} \\ \boldsymbol{y}_{\mathrm{Xi+Yi}} \end{bmatrix} = \boldsymbol{X}_0\boldsymbol{\alpha} + \begin{bmatrix} \boldsymbol{x}_0 & \mathbf{0} & \boldsymbol{z}_0 \\ \mathbf{0} & \mathbf{1} & \boldsymbol{z}_{\mathrm{Xi}} \\ \boldsymbol{x}_{\mathrm{Yi}} & \mathbf{0} & \boldsymbol{z}_{\mathrm{Yi}} \\ \mathbf{0} & \mathbf{1} & \boldsymbol{z}_{\mathrm{Xi+Yi}} \end{bmatrix} \boldsymbol{\beta} + \boldsymbol{\epsilon}.$$

Finally, for mechanism change interventions we have:

$$\begin{bmatrix} \boldsymbol{y}_0 \\ \boldsymbol{y}_{\mathrm{Xi}} \\ \boldsymbol{y}_{\mathrm{Yi}} \\ \boldsymbol{y}_{\mathrm{Xi+Yi}} \end{bmatrix} = \boldsymbol{X}_0\boldsymbol{\alpha} + \begin{bmatrix} \boldsymbol{x}_0 & \mathbf{0} & \boldsymbol{z}_0 \\ \mathbf{0} & \boldsymbol{x}_{\mathrm{Xi}} & \boldsymbol{z}_{\mathrm{Xi}} \\ \boldsymbol{x}_{\mathrm{Yi}} & \mathbf{0} & \boldsymbol{z}_{\mathrm{Yi}} \\ \mathbf{0} & \boldsymbol{x}_{\mathrm{Xi+Yi}} & \boldsymbol{z}_{\mathrm{Xi+Yi}} \end{bmatrix} \boldsymbol{\beta} + \boldsymbol{\epsilon}.$$

Given our focus on kinase inhibitors, we considered only "-out" interventions above. For comparison, perfect-in interventions, which act upon the parents of a protein instead of its children, (see for example Eaton and Murphy, 2007) produce the model:

$$\begin{bmatrix} \boldsymbol{y}_0 \\ \boldsymbol{y}_{\mathrm{Xi}} \end{bmatrix} = \boldsymbol{X}_0\boldsymbol{\alpha} + \begin{bmatrix} \boldsymbol{x}_0 & \boldsymbol{z}_0 \\ \boldsymbol{x}_{\mathrm{Xi}} & \boldsymbol{z}_{\mathrm{Xi}} \end{bmatrix} \boldsymbol{\beta} + \boldsymbol{\epsilon}.$$

Note that for '-in' forms of the intervention it is inhibitor Yi which directly affects Y rather than inhibitor Xi.

2.4 Protein data example

We now illustrate the foregoing approaches using a simple, real data example (Figure 3) in which a known three-node network is interrogated by inhibition (data courtesy Gray Lab, OHSU Knight Cancer Institute, Portland, OR, USA). Three phospho-proteins – the receptor EGFR, phosphorylated on tyrosine residue #1173 ("EGFRpY1173"), and two nodes downstream of EGFR, namely AKTpS473 and MAPKpT202 – were observed through time under several experimental conditions. The conditions included: no inhibitors (green in Figure 3), with an AKT inhibitor (blue), with an EGFR inhibitor (red) and with both EGFR and AKT inhibition (purple). In line with the known network, the data show a clear reduction in the observed level of AKTpS473 and MAPKpT202 under EGFR inhibition.

To investigate the behaviour of the interventional schemes described above, we carried out inference for these data using DBNs coupled to the respective intervention scheme. We show the data itself, fitted output from the various models and the corresponding inferred networks. Strikingly, although several of the methods fit the data reasonably well, only fixed effect and perfect-fixed-effect are able to both fit the data and estimate what we believe to be the correct network.

It is noteworthy that even in this simple example it is possible to fit the data well whilst estimating a plainly incorrect network. For example, the no intervention model fits the data (including the inhibitor time courses) reasonably well, but does not estimate the known edges from EGFR to MAPK and AKT, despite the fact that both MAPK and AKT change dramatically under EGFR inhibition in the very data being analysed. This is an example of statistical confounding that arises due to the fact that the data are analysed "blind": the analysis does not know which time course was obtained under EGFR inhibition, rendering the easily seen causal effect of EGFR on AKT and MAPK invisible to network inference. In contrast, the fixed effect intervention approaches can directly incorporate this information in the overall network inference. Note also that the inhibitors can be seen to affect the concentration of their target proteins, most likely due to feedback mechanisms that are represented by self-edges in the estimated network. For more discussion about the role of the self edge in the network, see Section 4.

3 Results

We performed a simulation study to compare the network inference methods with different intervention models. Data for 15 nodes were simulated from a DBN using a data-generating graph G^* (see Figure S1). Mimicking the design of typical real proteomic experiments, for each protein we simulated a small number of timepoints (8) in several experimental conditions (4, namely: no inhibitor; 'A' inhibition; 'B' inhibition; both 'A' and 'B' inhibition), giving $n = 32$ multivariate datapoints. We sampled coefficients for the node-specific linear models uniformly at random from the set $(-1, -0.5) \cup (0.5, 1)$ in order to create associations of various strengths, whilst avoiding associations close to zero. To simulate data under *in silico* inhibition requires an intervention model: since each interventional scheme also corresponds to such a data-generating model, to avoid bias we simulated data based on all four intervention models that were considered, namely perfect, fixed effects, perfect with fixed effect and mechanism change. Network inference was then carried out as described above using these four intervention models plus the model with no intervention and simple, marginal correlations between nodes (i.e. a co-expression network).

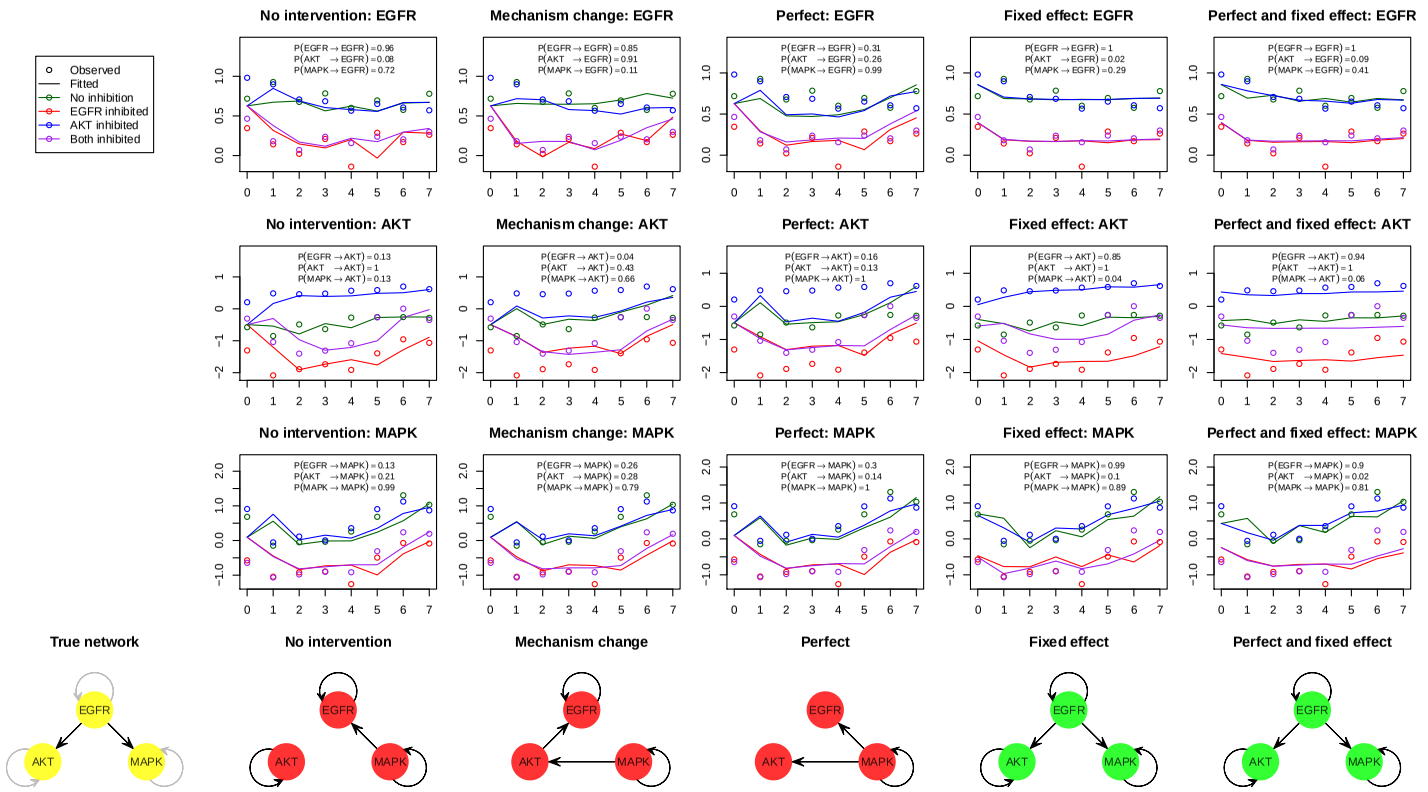

Figure 3: Observations (circles), fitted values (lines), and posterior edge probabilities (insets) based on data from cell line AU565 for three proteins EGFRpY1173, AKTpS473 and MAPKpT202, for five intervention models: no intervention, mechanism change, perfect, fixed effect, and perfect with fixed effect.

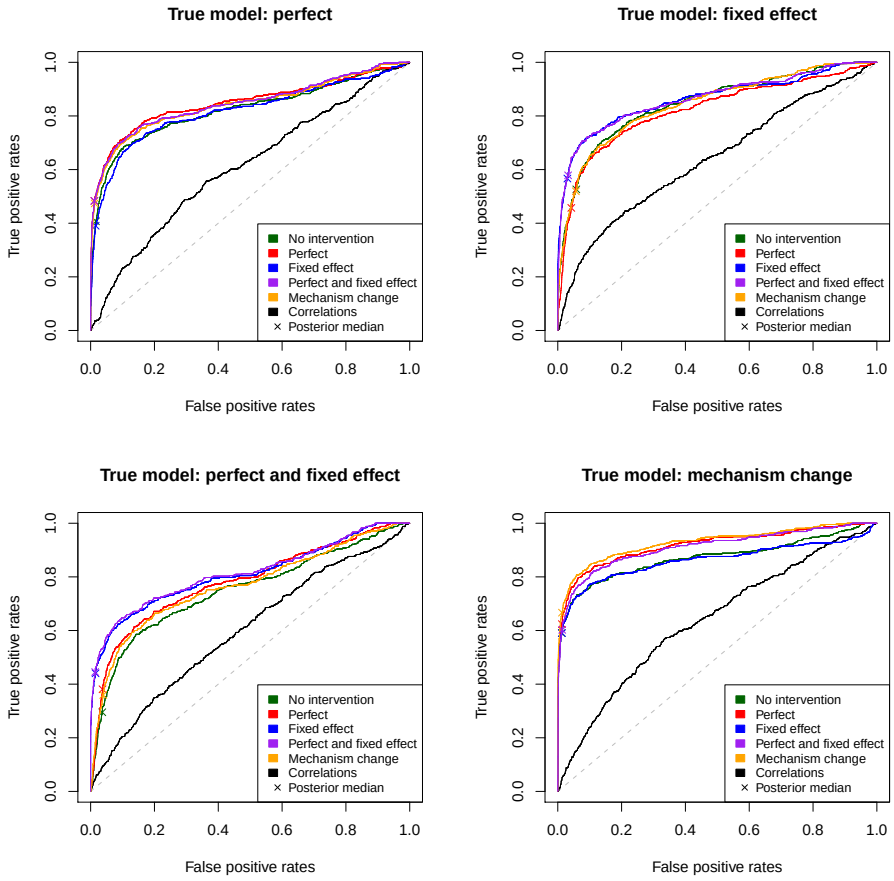

Figure 4: ROC curves for the different intervention models for simulated data based on a) perfect b) fixed effect, c) perfect and fixed effect and d) mechanism change intervention models. The crosses indicate the performance of the posterior median network (50% posterior edge probability threshold).

Figure 4 shows ROC curves (produced from 20 datasets for each regime) for each combination of intervention method and the underlying model. In each case and as expected, analysis under the data-generating model gives the best results. However, the perfect and fixed effect model does consistently well and is generally close to the data-generating model. The mechanism change model generally appears to perform similarly to the perfect intervention model. Co-expression analysis does much less well than all of the DBN models. Note that even under the correct data-generating model, in this noisy, small sample example, the area under the ROC curve can be much lower than unity, highlighting the inherent difficulty of the network inference problem and the challenging nature of the simulation.

4 Discussion

Network inference is widely used to generate hypotheses about regulatory links in biological systems. Hypotheses that are intended to be validated by interventional experiments are by definition causal hypotheses. Interventional data are important for elucidation of causal links. We discussed and illustrated the issues of confounding that can arise in network inference. As we showed in simple toy and real data examples, nodes not linked in terms of regulation can nonetheless exhibit statistical association and thereby easily lead network inference astray. We showed how such confounding can present a concern even with only three nodes but the issue becomes rapidly more severe in higher dimensions. Results on simulated data suggested that the "perfect-fixed-effect" intervention scheme we proposed represents a good default choice for kinase inhibition experiments.

A number of limitations of network inference in real applications are discussed in Oates and Mukherjee (2012), who point to several reasons why network inference may not be possible in practice. A point of particular relevance to this paper is the fact that usually only a subset of the nodes can be intervened upon. Further experimental and computational work is needed to assess whether for a given class of network, intervention on only a subset of nodes is sufficient for recovery of the causal graph.

The self edge (the edge that connects a protein to itself) has two roles in the model. First, it can represent statistical autocorrelation in the protein time-course. Second, it can represent a feedback loop, possibly via some additional unmeasured proteins. Negative feedback would give rise to a negative regression coefficient and positive feedback, a positive coefficient. Since we integrate out the regression coefficient to obtain the marginal likelihood, the posterior signalling network does not give any indication of the direction of this effect, or what role the self edge is performing. In future work we hope to differentiate between inhibition and activation effects in the protein signalling network using

ideas from causal inference; this may help to identify the role of the self-edges.

Appendix A – Orthogonalisation

We can assume without loss of generality that the two parts of the design matrix $\boldsymbol{X}_0$ and $\boldsymbol{X}_\gamma$ are orthogonal, i.e. $\boldsymbol{X}_0^T\boldsymbol{X}_\gamma = \boldsymbol{0}_{a\times b}$. If this is not the case then we can reparameterise from $\mathbf{x} = \boldsymbol{X}_0\boldsymbol{\alpha} + \boldsymbol{X}_\gamma\boldsymbol{\beta} + \boldsymbol{\epsilon}_p$ to $\mathbf{x} = \boldsymbol{X}_0\boldsymbol{\alpha}' + \boldsymbol{X}'_\gamma\boldsymbol{\beta} + \boldsymbol{\epsilon}_p$, where $\boldsymbol{\alpha}' = \boldsymbol{\alpha} - (\boldsymbol{X}_0^T\boldsymbol{X}_0)^{-1}\boldsymbol{X}_0^T\boldsymbol{X}_\gamma\boldsymbol{\beta}$, $\boldsymbol{X}'_\gamma = (\boldsymbol{I}_n - \boldsymbol{P}_0)\boldsymbol{X}_\gamma$, and $\boldsymbol{X}_0^T\boldsymbol{X}'_\gamma = \boldsymbol{0}_{a\times b}$

The orthogonal reparameterisation is useful because the hat matrix of the linear model ($\boldsymbol{H}$) is simpler when written in terms of $\boldsymbol{X}'_\gamma$ rather than $\boldsymbol{X}_\gamma$, i.e. $\boldsymbol{H} = \boldsymbol{P}_0 + \boldsymbol{X}'_\gamma(\boldsymbol{X}'^T_\gamma\boldsymbol{X}'_\gamma)^{-1}\boldsymbol{X}'^T_\gamma$. For more details see Forte Deltell (2011).

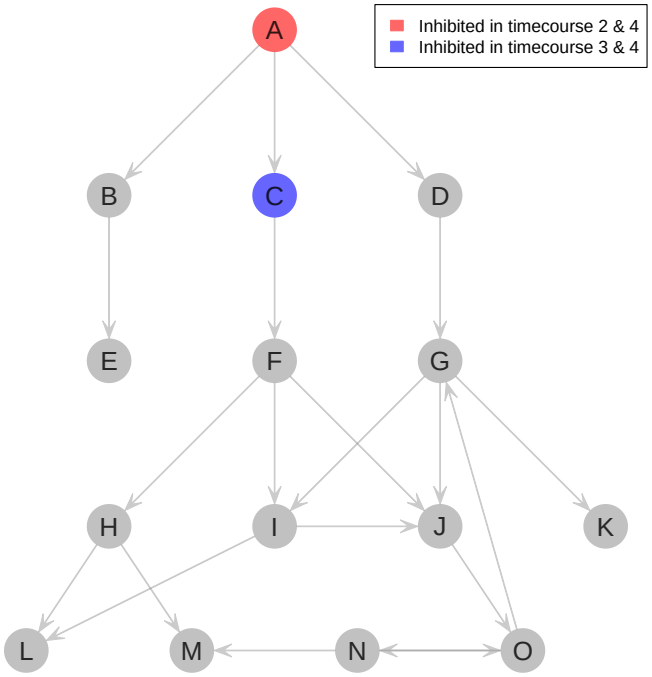

Figure S1: Network used to generate simulated data.

References

Äijö, T. and Lähdesmäki, H. (2009) Learning gene regulatory networks from gene expression measurements using non-parametric molecular kinetics. *Bioinformatics*, **25**, 2937–2944.

Bansal, M., Della Gatta, G., di Bernardo, D. (2006) Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. *Bioinformatics*, **22**, 815–822.

Bender, C., *et al.* (2010) Dynamic deterministic effects propagation networks:learning signalling pathways from longitudinal protein array data. *Bioinformatics*, **26**, i596–i602.

Denison, D.G.T., *et al.* (2002) *Bayesian Methods for non-linear calssification and regression.* John Wiley and sons, Chichester, UK.

Eaton D. and Murphy, K. (2007) Exact Bayesian structure learning from uncertain interventions. In *Proceedings of 11th Conference on Artificial Intelligence and Statistics (AISTATS-07).*

Forte Deltell, A. (2011) Objective Bayes criteria for variable selection. PhD thesis, University of Valencia, Spain.

Friedman, N., Murphy, K., and Russell, S. (1998). Learning the structure of dynamic probabilistic networks. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, 139–147.

Frölich, H. *et al.* (2008) Estimating large-scale signaling networks through nested effects models with intervention effects from microarray data. *Bioinformatics*, **24**, 2650–2656.

Hill, S.M. *et al.* (2012) Bayesian inference of signaling network topology in a cancer cell line. *Bioinformatics*, **28**, 2804–2810.

Husmeier, D. (2003) Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks. *Bioinformatics*, **19**, 2271–2282.

Kohn, R., Smith, M., Chan D. (2001) Nonparametric regression using linear combinations of basis functions. *Statistics and Computing*, **11**, 313–322.

Markowetz, F., Bloch, J. and Spang, R. (2005) Non-transcriptional pathway features reconstructed from secondary effects of RNA interference. *Bioinformatics*, **21**, 4026–4032.

Murphy, K.P. (2002) Dynamic Bayesian networks: representation, inference and learning. PhD thesis, Computer Science, University of California, Berkeley, CA.

Oates, C.J. and Mukherjee, S. (2012) Network inference and biological dynamics. *The Annals of Applied Statistics*, **6**(3), 1209-1235.

Pearl, J. (2000) *Causality: Models, Reasoning and Inference.* Cambridge University Press, Cambridge, UK.

Pearl, J. (2009) Causal inference in statistics: an overview. *Statistics Surveys*, **3**, 96–146.

Scott, J.G. and Berger, J.O. (2010) Bayes and empirical Bayes multiplicity adjustment in the variable selection problem. *Annals of Statistics*, **38**, 2587–2619.

Smith, M. and Kohn, R. (1996) Nonparametric regression using Bayesian variable selection. *Journal of Econometrics*, **75**, 317–344.

Terfve, C. *et al.* (2012) CellNOptR: a flexible toolkit to train protein signalling networks to data using multiple logic formalisms. *BMC Systems Biology*, **6**, 133.