Joint Structure Learning of Multiple Non-Exchangeable Networks

Chris. J. Oates
Department of Statistics, University of Warwick, UK.
Sach Mukherjee
Department of Biochemistry, Netherlands Cancer Institute, NL.

February 25, 2014

Abstract

Several methods have recently been developed for joint structure learning of multiple (related) graphical models or networks. These methods treat individual networks as exchangeable, such that each pair of networks are equally encouraged to have similar structures. However, in many practical applications, exchangeability in this sense may not hold, as some pairs of networks may be more closely related than others, for example due to group and sub-group structure in the data. Here we present a novel Bayesian formulation that generalises joint structure learning beyond the exchangeable case. In addition to a general framework for joint learning, we (i) provide a novel default prior over the joint structure space that requires no user input; (ii) allow for latent networks; (iii) give an efficient, exact algorithm for the case of time series data and dynamic Bayesian networks. We present empirical results on non-exchangeable populations, including a real data example from biology, where cell-line-specific networks are related according to genomic features.

1 Introduction

Structure learning remains an important and challenging problem. Often we seek to learn multiple graphs or *networks* $\{G_i\}_{i\in\mathcal{I}}$ that are expected to be related but that may be non-identical. For example in biomedical applications, multivariate data $\{\boldsymbol{y}_i\}_{i\in\mathcal{I}}$ pertaining to the same biological process (e.g. gene regulation or protein signaling) may be obtained from multiple, related samples $i \in \mathcal{I}$ (e.g. patients or laboratory models) that are expected to be non-identical with respect to conditional independence structure (Penfold *et al.*, 2012; Danaher *et al.*, 2014; Oates *et al.*, 2013). In such situations, it is natural to consider joint learning that allows for information sharing between the inference problems indexed by $i \in \mathcal{I}$. Several techniques have been proposed for such joint structure learning, including Bayesian techniques for graphical models (Werhli

and Husmeier, 2008; Penfold *et al.*, 2012; Oates *et al.*, 2013) and penalised likelihood estimators for Gaussian graphical models (GMMs; Chiquet *et al.*, 2011; Guo *et al.*, 2011; Yang *et al.*, 2012; Danaher *et al.*, 2014; Mohan *et al.*, 2014). These methods have been shown to improve estimation of individual graphs (or networks, we use both terms interchangeably) G_i, especially in the regime where local sample sizes n_i are not large.

Existing joint structure learning methods operate by shrinking estimated networks towards each other under an exchangeability assumption (i.e. the $\{G_i\}_{i\in\mathcal{I}}$ are treated as exchangeable random variables). However, in practice, relationships between datasets $\{\boldsymbol{y}_i\}_{i\in\mathcal{I}}$ (and their underlying networks) may be complex, e.g. hierarchical, with group and sub-group structure. For example, in biology, datasets from multiple species may be related according to a complex evolutionary history (Baumbach *et al.*, 2009), while cells within a tumour are related according to their lineage within the tumour (Gerlinger *et al.*, 2012). Similarly, in a data mining application, networks with nodes corresponding to products in an inventory (Taylor and Fox, 2011) may be arranged into groups and sub-groups based on market structure or region.

This paper introduces a richer class of Bayesian joint estimators known as structure learning trees (SLTs) that subsume previous exchangeable formulations whilst permitting more complex, non-exchangeable relationships between networks. An SLT is a rooted tree T whose vertices are themselves networks and whose edges describe relationships between the networks (e.g. group and sub-group membership). The tree T encodes possibly non-exchangeable relationships between networks that are exploited during joint structure learning. In this paper we restrict attention to the case where the tree T can be reasonably specified *a priori*. For example in biology, depending on the setting, established taxonomies such as phylogeny, tissue type, disease (sub)-type etc. can be used to specify T. Such taxonomies are often supported by a wealth of experimental evidence, and it is therefore natural to leverage them for improved structure learning. In the case where T may be uncertain, we provide empirical results that investigate the extent to which SLT estimators are robust to T misspecification.

This paper is organised as follows: In Section 2.1 we introduce SLTs, generalising existing work on joint structure learning to the non-exchangeable setting. Prior specification for SLTs is achieved by appealing to the intuitive notion that model constraints should be inherited along the edges of the tree. This heuristic allows specification of a default structural prior over all networks jointly that has essentially no user-set hyperparameters. Section 2.2 provides an exact belief propagation algorithm for inference of both data-generating and latent network structures, while Section 2.3 focuses on time series data and dynamic Bayesian networks. Empirical results on simulated data in Section 3.1 assess the performance of SLTs, including cases where the joint structural prior is misspecified. Section 3.2 shows results on proteomic time series data from multiple cancer cell lines that illustrate the use of SLTs in a topical application. Finally we close with a discussion in Section 4.

2 Methods

2.1 A Bayesian hierarchical model

Structure learning trees We consider joint structure learning of multiple *networks* $G_i = (V, E_i)$, $i \in \mathcal{I}$, that share the same *vertex set* $V = \{1, \ldots, P\}$ but may differ with respect to their *edge sets* $E_i \subseteq V \times V$. Let $\mathcal{G}$ denote the space of all networks over vertex set V, up to restrictions associated with any particular model class (e.g. acyclicity, undirected edges etc.). We define a *structure learning tree* $T = (\mathcal{I}, E_T)$ as a rooted tree whose vertices are used[1] to index individual networks G_i, with all edges $e \in E_T$ directed away from the root. Examples of SLTs are displayed in Figs. 1 and 3. The root network is denoted by G_1. Existing methods for joint estimation (see Introduction) can be regarded as a special case of the general SLT where T has a star topology with centre G_1.

Latent networks. In classical structure learning, network structure is latent in the sense that it not directly observed. SLTs allow for further latency; specifically we consider the situation in which data $\{\mathbf{y}_i\}_{i \in \iota}$ are available conditional upon only a subset $\iota \subseteq \mathcal{I}$ of the networks of interest. The remaining nodes $\mathcal{I} \setminus \iota$ are *doubly latent* in the sense that neither they, nor data directly conditional upon them, are observed. Latent nodes may be used to describe hidden (e.g. group level) structure (as in our biological example in Sec. 3.2). Learning in an SLT exploits relationships between networks as encoded in T to allow joint estimation of *all* networks $\{G_i\}_{i \in \mathcal{I}}$, whilst respecting non-exchangeable relationships between these networks.

A default, subset prior. To formulate a joint statistical model we begin by placing a prior $p(G_1|G^0)$ on the root network $\overline{G_1}$. Henceforth $\overline{G_i}$ represents the true (unknown) value of the network corresponding to $i \in \mathcal{I}$ whilst G_i will be used to denote a possible structure for $\overline{G_i}$, and G^0 is a fixed "prior network" (see below). Then we define a joint structural prior over all networks $\{\overline{G_i}\}_{i \in \mathcal{I}}$ that factorises along the edges of T:

$$p(\{G_i\}_{i \in \mathcal{I}}|G^0, E_T) = p(G_1|G^0) \prod_{(i,j) \in E_T} p(G_j|G_i) \tag{1}$$

Previously proposed structural priors (e.g. Mukherjee and Speed, 2008; Werhli and Husmeier, 2008) could in principle be used to specify the conditional density $p(G'|G)$. Recent work on the joint estimation of multiple exchangeable networks has focused on Boltzmann priors $p(G'|G) \propto \exp(-\lambda d(G, G'))$ for some measure of distance $d : \mathcal{G} \times \mathcal{G} \to [0, \infty)$ (Werhli and Husmeier, 2008; Penfold *et al.*, 2012; Oates *et al.*, 2013) and analogous penalised likelihoods (Chiquet *et al.*, 2011; Guo *et al.*, 2011; Yang *et al.*, 2012; Danaher *et al.*, 2014; Mohan *et al.*,

[1] It will be convenient to interchange between an index $i \in \mathcal{I}$ and its corresponding network G_i.

2014). However, such priors can be difficult to specify in the exchangeable case (Werhli and Husmeier, 2008; Penfold *et al.*, 2012; Oates *et al.*, 2013) and generalise poorly to the non-exchangeable case since each edge $e \in E_T$ in principle introduces an associated hyperparameter $\lambda_e \in [0, \infty)$.

To control complexity of prior specification, we make use of the simple heuristic that network structure must be a subset of the structure of all network ancestors according to T:

$$p(G_j|G_i) \propto \mathbb{I}\{E_j \subseteq E_i\}\eta(G_j) \tag{2}$$

Here $\mathbb{I}$ is the indicator function and η provides multiplicity correction for varying $G_j \in \mathcal{G}$ (see below). Under Eqn. 2 the prior $p(G_1|G^0)$ encodes prior certainty that particular edges cannot exist, in *any* network in $\{\overline{G_i}\}_{i\in\mathcal{I}}$. If the networks are interpreted as causal graphical models then G^0 describes a set of conditional independence assumptions. Thus in our formulation, inferred causation is explicitly conditional on prior causal hypotheses G^0 (Pearl, 2009).

Multiplicity correction. Multiplicity correction plays an important role in Bayesian structure learning beyond the penalty on model complexity provided by the marginal likelihood. This is clearly illustrated in the context of variable selection, where a uniform prior over variable subsets has the undesirable property that the prior mass on all models with exactly one predictor goes to zero as the number of predictors grows large; such a prior cannot make sense in settings where one expects that a single predictor should have some reasonable prior mass. Following Scott and Berger (2010) we employ a binomial multiplicity correction

$$\eta(G) = \prod_{p\in V} \binom{P}{d_p(G)}^{-1} \mathbb{I}\{d_p(G) \leq d_{\max}\} \tag{3}$$

where $d_p(G)$ is the in-degree of vertex p in G. Here $d_{\max}$ represents a constraint on in-degree; such constraints are widely used to facilitate inference in graphical models (e.g. Hill *et al.*, 2012).

2.2 Exact inference

Marginal belief propagation. In this Section we describe how marginalisation and belief propagation combine to facilitate efficient, exact inference in SLT models. Taken together with a "local" likelihood $p(\boldsymbol{y}_i|\boldsymbol{\theta}_i, G_i)$, Eqn. 1 defines a Bayesian network on both discrete ($\overline{G_i}$) and possibly continuous ($\boldsymbol{\theta}_i$) variables (SFig. 4a). Efficient inference will require marginalisation of continuous variables; for data $\boldsymbol{y}_i$ we require that the "marginal likelihoods" $p(\boldsymbol{y}_i|G_i) = \int p(\boldsymbol{y}_i|\boldsymbol{\theta}_i, G_i)p(\boldsymbol{\theta}_i|G_i)d\boldsymbol{\theta}_i$ are pre-computed and cached for all $i \in \iota$. Here $p(\boldsymbol{y}_i|G_i)$ is a convenient shorthand for $p(\boldsymbol{y}_i|E_i)$, the evidence for a particular topology $\overline{E_i} = E_i$, and $\boldsymbol{\theta}_i$ are parameters required to specify the local data-generating model. For many models of interest, including dynamic Bayesian

networks (see Sec. 2.3), marginal likelihood may be computed in closed form by exploiting conjugate prior specifications. Otherwise, Monte Carlo and related numerical techniques may be used to approximate marginal likelihood in more complex models (e.g. Calderhead and Girolami (2009)).

The marginalised SLT (SFig. 4b) is then a discrete Bayesian network with respect to T. A factor graph representation of the marginal SLT model is shown in SFig. 4c. Exact inference over factor graphs can be achieved efficiently using belief propagation (Pearl, 1982), provided the factor graph is acyclic. By restricting attention to tree structures T in Sec. 2.1 we have guaranteed that the factor graph is acyclic. Belief propagation therefore yields posterior distributions $p_i(G_i|\boldsymbol{y})$ over structure for each $i \in \mathcal{I}$. Pseudocode for our approach is provided in Supp. Sec. 5.1.

Model averaging. Evidence in favour of an edge (k, l) in a network $\overline{G_i}$ is summarised by the posterior marginal inclusion probability obtained by averaging over all possible structures G_i for $\overline{G_i}$:

$$p((k,l) \in \overline{E_i}|\boldsymbol{y}) = \sum_{G_i \in \mathcal{G}} \mathbb{I}\{(k,l) \in \overline{E_i}\} p_i(G_i|\boldsymbol{y}). \tag{4}$$

Here $\boldsymbol{y} = \{\boldsymbol{y}_i\}_{i \in \iota}$ contains all data. The subset constraints of Eqn. 2 manifest in the posterior as $p((k,l) \in \overline{E_i}|\boldsymbol{y}) \geq p((k,l) \in \overline{E_j}|\boldsymbol{y})$ whenever j is a descendant of i in T.

2.3 Explicit formulae for time series

FFDBN models. For graphical models and time series data we provide explicit formulae: We follow previous work by Murphy (2002); Hill *et al.* (2012), adopting a "feed-forward" dynamic Bayesian network (FFDBN) model for time series data. For clarity of notation we consider a specific fixed network G, suppressing dependence upon $i \in \mathcal{I}$. FFDBNs prohibit contemporaneous edges; this confers computational advantages (see Hill *et al.* (2012) for full details). Key features of FFDBNs include; (i) feedback can be explicitly modelled through time, (ii) the likelihood factorises over variables $p \in V$, reducing computational complexity (see below), (iii) conjugate priors and closed form expressions for marginal likelihood are available, and (iv) experimental designs involving interventions may be integrated in line with a causal calculus (Spencer and Mukherjee, 2012).

In a FFDBN the value $Y_p(t)$ of variable p at (discrete) time t is dependent upon covariates $\boldsymbol{Y}(t-1) = [Y_1(t-1), \ldots, Y_P(t-1)]$. When multiple time series are available, the vector $\boldsymbol{Y}_p = \left[Y_p^1(1), \ldots, Y_p^1(n)\ ,\ Y_p^2(1), \ldots, Y_p^2(n) \ldots\right]$ denotes the concatenated time series, with the subscript indexing a specific variable $p \in V$. We write $\mathrm{pa}_G(p)$ for the parents of vertex p in the network G. In this paper we restrict attention to linear models that, for variable p, may be expressed as $\boldsymbol{Y}_p = \boldsymbol{X}_0\boldsymbol{\alpha} + \boldsymbol{X}_{\mathrm{pa}_G(p)}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon} \sim N(\boldsymbol{0}_{n\times 1}, \sigma^2\boldsymbol{I}_{n\times n})$. The matrix $\boldsymbol{X}_0 = [\boldsymbol{1}_{\{t=1\}}\ \boldsymbol{1}_{\{t>1\}}]_{n\times 2}$ contains a term for the initial time point in each

series. The elements of $\boldsymbol{X}_{\mathrm{pa}_G(p)}$ corresponding to initial observations $(\boldsymbol{Y}_p)_{\{t=1\}}$ are simply set to zero. Parameters $\boldsymbol{\theta} = \{\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma\}$ are specific to variable p and network G. In the linear case the model-specific component $\boldsymbol{X}_{\mathrm{pa}_G(p)}$ of the design matrix consists of the predictors $\boldsymbol{Y}_{\mathrm{pa}_G(p)}(t-1)$, where $\boldsymbol{Y}_A$ denotes the elements of the vector $\boldsymbol{Y}$ belonging to the set A.

Intervention. In Sec. 3 we consider experimental designs that involve targeted intervention on vertices in the data-generating networks. We followed the approach described in Spencer and Mukherjee (2012) to integrate interventional data in line with a causal calculus. Specifically, for the type of intervention in the experimental data (drug inhibition of kinases), using a "perfect out fixed effects" (POFE) approach (we direct the interested reader to the reference for full details). This changes the network structure to model the intervention in line with the *do*-calculus (Pearl, 2009) and also includes a fixed effect $\boldsymbol{X}_1\boldsymbol{\gamma}$ in the regression model for $\boldsymbol{Y}_p$, such that $\boldsymbol{X}_1$ indicates whether or not intervention(s) were used for each data-point.

Prior specification. We used a standard conjugate formulation for the linear model. Specifically, we employed a Jeffreys prior $p(\boldsymbol{\alpha}, \sigma|\mathrm{pa}_G(p)) \propto 1/\sigma$ for $\sigma > 0$ over the common parameters. Prior to inference, the non-interventional components of the design matrix were orthogonalised (following Deltell *et al.*, 2012) using the transformation

$$(\boldsymbol{X}_{\mathrm{pa}_G(p)})_{ak} \mapsto \sum_{l=1}^{n} (\boldsymbol{I}_{n\times n} - \boldsymbol{P}_0)_{al} (\boldsymbol{X}_{\mathrm{pa}_G(p)})_{lk}, \tag{5}$$

where $\boldsymbol{P}_0 = \boldsymbol{X}_0(\boldsymbol{X}_0^T\boldsymbol{X}_0)^{-1}\boldsymbol{X}_0^T$. We then assumed a unit-information g-prior for regression coefficients (Zellner, 1986), given by

$$\boldsymbol{\beta}|\boldsymbol{\alpha}, \sigma, \mathrm{pa}_G(p) \sim N(\boldsymbol{0}_{b\times 1}, n\sigma^2(\boldsymbol{X}^T_{\mathrm{pa}_G(p)}\boldsymbol{X}_{\mathrm{pa}_G(p)})^{-1}) \tag{6}$$

where $b = \dim(\boldsymbol{\beta})$. (When interventional designs are used, the pair $(\boldsymbol{\beta}, \boldsymbol{\gamma})|\boldsymbol{\alpha}, \sigma, \mathrm{pa}_G(p)$ are jointly assigned a g-prior.)

Marginal likelihood. With the above specification, the evidence in favour of $\mathrm{pa}_G(p)$ can be obtained in closed-form:

$$p(\boldsymbol{y}_p|\mathrm{pa}_G(p)) \propto \frac{1}{(n+1)^{b/2}} \times \left(\boldsymbol{y}_p^T\left(\boldsymbol{I}_{n\times n} - \boldsymbol{P}_0 - \frac{n}{n+1}\boldsymbol{P}_{\mathrm{pa}_G(p)}\right)\boldsymbol{y}_p\right)^{-\frac{n-a}{2}} \tag{7}$$

where $\boldsymbol{P}_{\mathrm{pa}_G(p)} = \boldsymbol{X}_{\mathrm{pa}_G(p)}(\boldsymbol{X}^T_{\mathrm{pa}_G(p)}\boldsymbol{X}_{\mathrm{pa}_G(p)})^{-1}\boldsymbol{X}^T_{\mathrm{pa}_G(p)}$, $a = \dim(\boldsymbol{\alpha})$ and $b = \dim(\boldsymbol{\beta})$. Note that the left hand side of Eqn. 7 is an abuse of notation since dependence on covariates $\boldsymbol{Y}_{\mathrm{pa}_G(p)}(t-1)$ is suppressed (a formal treatment is presented in Oates *et al.* (2013)).

Computation. From the factorisation property of FFDBNs, the total marginal likelihood is simply given by the product

$$p(\boldsymbol{y}|G) = \prod_{p \in V} p(\boldsymbol{y}_p|\mathrm{pa}_G(p)). \tag{8}$$

For FFDBNs the parent sets $\mathrm{pa}_G(p)$ $(1 \leq p \leq P)$ are Fisher-orthogonal; computational complexity may therefore be significantly reduced by decomposing the SLT into P independent SLTs, each targeting one parent set $\mathrm{pa}_G(p)$. MATLAB R2013b code implementing our procedure is provided in the Supplement.

Although we have focussed on FFDBNs, our procedure applies to other classes of network models, such as Bayesian networks and Gaussian graphical models. The availability of explicit formulae for FFDBNs motivates their use for the computational study presented below.

3 Results

3.1 Simulated data

To probe empirical performance of SLTs, we simulated data from a known tree T and assessed ability to infer the true data-generating networks $\{\overline{G_i}\}_{i \in \iota}$. In all experiments we placed $2P$ edges uniformly at random to generate a root network $\overline{G_1}$ subject to the in-degree constraint $d_p(G_1) = 2$ for all $p \in V$. Two child networks $\overline{G_{11}}, \overline{G_{12}}$ were then generated, each containing P edges drawn as described below. Finally 10 networks $\overline{G_{1ij}}$ were generated by sampling ρP edges as described below. We use concatenated subscripts to uniquely identify nodes in T; for example G_{12} corresponds to child 2 of network G_1.

Existing joint structure learning methodologies require exchangeability of networks, while SLT instead imposes a tree structure capturing non-exchangeable relationships. We considered 5 data-generating regimes designed to mimic various applied settings, including those in which the SLT assumptions are violated:

(1) **Disjoint sub-groups.** Edges in each (non-root) network are drawn at random from the parent in T, conditional upon the networks $\overline{G_{11}}$, $\overline{G_{12}}$ having disjoint edge sets. This regime strongly violates the exchangeability assumption implicit in existing joint structure learning methodologies.

(2) **Weakly exchangable.** Here networks $\overline{G_{11}}$, $\overline{G_{12}}$ are generated independently, conditional upon $\overline{G_1}$, such that they are likely to share common edges. As above, all edges in each (non-root) network are drawn at random from the parent in T. Whilst exchangeability is violated, this regime ought to be more favourable to existing exchangeable estimators than regime (1) above.

(3) **Fully exchangeable.** The networks $\overline{G_{11}} = \overline{G_{12}}$ are taken equal, rendering the networks $\overline{G_{1ij}}$ fully exchangeable. In this regime SLT should lack efficiency relative to exchangeable estimators.

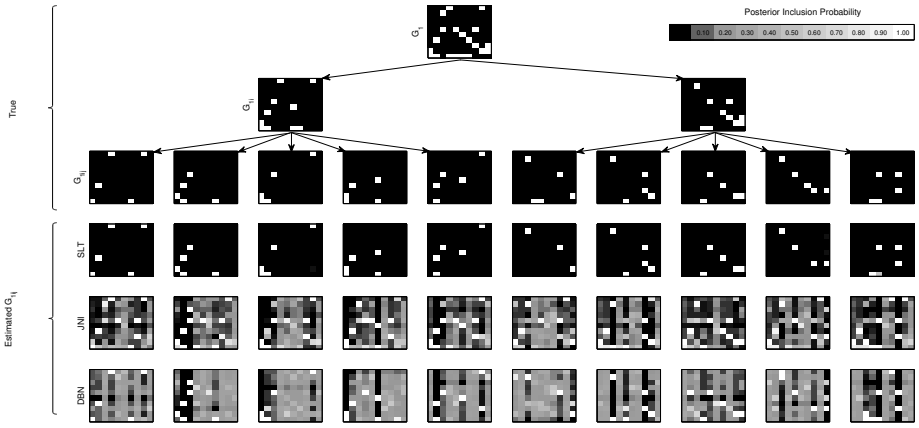


Figure 1: Results on data generated from disjoint sub-groups (regime (1), see text), where "doubly latent" networks $\overline{G_{11}}$, $\overline{G_{12}}$ share no common edges. Top: Data-generating networks and associated tree structure. Bottom: Estimates for individual network structure. [Inference methods: "SLT" = structure learning trees, "JNI" = joint (exchangeable) network inference (Oates *et al.*, 2013), "DBN" = classical network inference applied to each network separately (see text for details). Data consisted of $n = 60$ time points; see Supplement for full details of the data-generating set-up.].

(4) **Misspecified tree.** This data-generating regime is equivalent to the disjoint sub-groups regime (1), however the SLT estimator is based not on the true data-generating tree, but rather on a tree T' uniformly sampled from the space of all trees. Thus while the networks are non-exchangeable, the SLT is misspecified with high probability. This mimics the scenario in which an *a priori* assumed tree is used that is in fact largely incorrect.

(5) **Subset violation.** All edges in each (non-root) network are drawn such that 20% of edges in each child network are not edges in its parent network in T. In this regime, sub-groups exist among the networks, but the key assumption (Eqn. 2) of the parameter-free structural prior is violated with high probability.

Time series data $\boldsymbol{y}_{1ij}$ of length n were generated from each of the 10 networks $\overline{G_{1ij}}$ according to a linear autoregressive process with interventions described in Supp. Sec. 5.2.1. No data were made available on the networks $\overline{G_1}, \overline{G_{11}}, \overline{G_{12}}$, which are doubly latent. For all simulation experiments we fixed $P = 10$. The entire process was repeated 10 times. We compared SLT to:

(A) Non-joint network inference ("DBN"), the default approach of carrying out structural inference using a FFDBN for each dataset $\boldsymbol{y}_{1ij}$ independently.

(B) Joint network inference ("JNI"; Oates *et al.*, 2013). This Bayesian method

is a special (exchangeable) case of our proposed SLT methodology. Hyperparameters were chosen according to the heuristics of Oates *et al.* (2013).

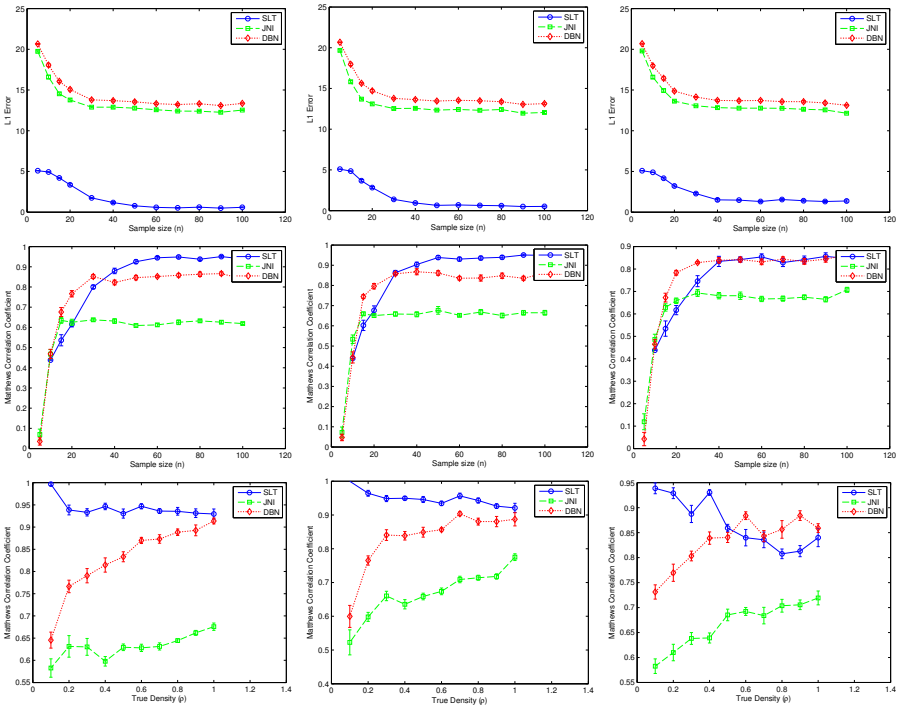
We note that alternative exchangeable estimators to (B) include Danaher *et al.* (2014) and Penfold *et al.* (2012), but the former has not been adapted for time series data and heavy computational demands of the latter preclude systematic empirical comparison. To ensure fair comparison, the same in-degree restriction $d_{\max} = 2$ (which includes the data-generating networks) was used for all methods. Moreover, to prevent confounding by differing formulations of likelihood, we based each method on the same FFDBN likelihood (as described in Sec. 2.3). Thus, all methods share the same basic time series formulation and differ only with respect to whether and how they share information between networks. No specific prior information was given regarding network topology, except for the tree structure T (in regimes 1-3,5) which was exploited by SLT.

We considered the thresholded network estimator, which consists of edges with marginal posterior inclusion probability (Eqn. 4) > 0.5. Performance at sample size n and density ρ was quantified using metrics from classifier analysis (see Supp. Sec. 5.2.2), averaged over all data-generating networks and all datasets. Here we focus on the Matthews correlation coefficient (MCC), which is regarded as a balanced measure, suitable for use when the underlying class distribution is skewed. To quantify performance of the posterior inclusion probabilities themselves, we also considered the ℓ_1 distance to the true data-generating networks. Further details regarding performance measures (including additionally AUPR and AUROC) appear in Supp. Sec. 5.2.2.

Intuitively, SLT should provide an advantage over JNI when the data structure contains distinct sub-groups with respect to network topology. Fig. 1 displays typical inferences in the "disjoint sub-group" regime (1) when $n = 60$, $\rho = 1/2$; SLT is noticeably sparser than JNI and DBN whilst achieving high MCC (Fig. 2a) and essentially perfect precision (SFig. 7). As a consequence, over all sample sizes n which we considered, SLT is considerably closer than JNI and DBN to the true network structures in the ℓ_1 norm (Fig. 2a). This ability to generate a clear decision boundary in the posterior is not demonstrated by JNI and DBN, which produce less sparse matrices of posterior inclusion probabilities (Fig. 2). This is expected, since JNI erroneously shares information *equally* among all networks, whilst DBN is statistically inefficient and therefore subject to higher variance.

Next, we relaxed the distinct sub-group architecture that likely favours SLT by allowing $\overline{G_{11}}$, $\overline{G_{12}}$ to share edges ("weakly exchangeable" regime (2); SFig. 5a). Results (Fig. 2b and SFig. 8) in this regime closely mirrored those of the disjoint sub-group regime, suggesting that SLTs offer improved estimation in more realistic weakly exchangeable settings. However in the fully exchangeable regime (3) (SFig. 5b) there was a decrease in performance of SLT with respect to JNI as quantified by AUPR, AUROC and the misclassification rate among top ranked edges (SFig. 9).

In order to probe robustness of SLT to prior mis-specification we considered two scenarios in which assumptions encoded in the joint structural prior are



(a) Disjoint sub-groups (b) Weakly exchangeable (c) Subset violation

Figure 2: Results on simulated data generated from SLTs in different regimes, as described in the Main Text. [Inference methods: "SLT" = structure learning trees, "JNI" = joint network inference (Oates *et al.*, 2013), "DBN" = classical network inference applied to each network separately. Performance metrics: "L1 Error" = average ℓ_1 distance between true and inferred (weighted) adjacency matrices, "Matthews Correlation Coefficient" = average MCC for thresholded network estimators. Error bars display standard error computed over 10 data-generating networks and for each network 10 sampled datasets. We considered both varying n for fixed $\rho = 0.5$ and varying ρ for fixed $n = 60$.]

violated. Firstly, we investigated whether the performance of SLT deteriorates when the tree T itself is misspecified (in fact chosen randomly; regime (4), SFig. 6a). These results showed that SLT remains superior to JNI and DBN terms of MCC, and remains competitive in terms of AUPR and AUROC (SFig. 10). Secondly, we considered strongly violating the subset inclusions (regime (5); SFig. 6b). The MCC performance of SLT in this regime was competitive with JNI and DBN (Fig. 2c). However SLT performed worse than JNI and DBN in terms of AUPR, AUROC and misclassification rate (SFig. 11). Robustness of SLTs is therefore dependent upon which aspects of performance are being considered.

The above experiments were performed at constant edge density $\rho = 1/2$, however SLT tends to produce sparser networks *a priori*. We therefore repeated the above experiments whilst varying the true density ρ and holding the number of samples constant at $n = 60$. Results (Fig. 2, SFigs. 7-11) showed that, in all regimes, performance of SLT improves in sparse settings whilst the performance of both JNI and DBN deteriorate. Examining the density of estimated networks relative to the data-generating networks (SFigs. 7-11) we found that JNI and DBN dramatically over-estimate density in sparse regimes; in contrast SLT automatically adjusts to the density of the data-generating networks. This appealing property results from our novel subset prior of Eqn. 2.

3.2 Biological data

This work was motivated by the problem of inference for protein signalling networks (PSNs) over a diverse panel of breast cancer cell lines. The cell lines under study are expected to differ with respect to PSN structure but can be grouped into sub-types based on underlying biology, as described below. Here independent estimation is likely to be inefficient, since the cell lines have a common lineage and share much of their biology. On there other hand, since sub-types may be quite different from one another, exchangeability within sub-type is arguably a more reasonable assumption than exchangeability between sub-type.

Amplification of the HER2 gene (denoted as "HER2+") is a key biomarker used to stratify breast cancer samples and cell lines. HER2 codes for a receptor that is a member of the EGFR family of receptors and it is believed that signalling related to these receptors may differ between these two sub-types. However, it is challenging to study signalling at the group level *per se*, since within each sub-type there remains considerable genetic diversity. We therefore applied SLT to learn both cell-line-specific and group-level PSNs, whilst controlling for confounding due to both HER2 status and line-specific genomic characteristics. Specifically, we constructed a tree T such that the doubly latent networks $\overline{G_{1i}}$ define HER2+/- sub-types respectively and the data-generating networks $\overline{G_{1ij}}$ correspond to PSNs in cell lines j of sub-type i. We used an informative prior network G^0 derived from the signalling literature (Fig. 3, top left).

Reverse phase protein array data (Hennessy *et al.*, 2010) were obtained over a panel of 10 breast cancer cell lines (Neve *et al.*, 2006) of which half were HER2+ and half HER2-. Data consisted of $P = 17$ protein expression levels, observed at 0.5,1,2,4,8,24,48,72 hours following ligand stimulation. A total of 4 time series were obtained, under treatment with DMSO, a EGFR/HER2 inhibitor (Lapatinib), an AKT inhibitor (AKTi) and Lapatinib + AKTi in combination, giving a total sample size of $n = 4 \times 8 = 32$. From a modelling perspective, the drugs Lapatinib and Akti are perturbations in the causal sense of intervening upon a node in the network. We assumed perfect interventions, corresponding to 100% removal of the target's activity with 100% specificity. Full experimental protocol is provided in the Supp. Sec. 5.3. Fig. 3 displays the inferred root network $\overline{G_1}$,

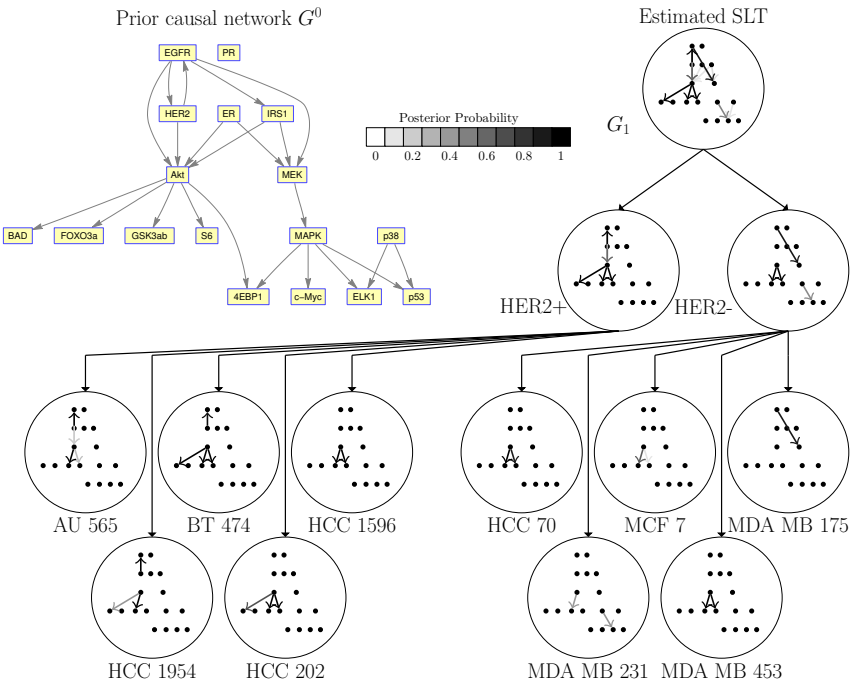


Figure 3: Results, experimental data. Sub-type and cell-line-specific protein signalling networks were inferred from proteomic data obtained from a panel of breast cancer cell lines. [The prior network G^0 (top left) may be used as a key for the vertex labels on smaller networks. Edge shading indicates posterior marginal inclusion probabilities as shown in the legend.]

the sub-type and cell line networks. It is noticeable that HER2 signalling plays a more prominent role in the HER2+ sub-type in line with biological intuition. Interestingly we infer regulation of BAD by HER2 (via AKT); dephosphorylation of BAD initiates apoptosis and this may help to explain a differential efficacy of HER2 inhibitors observed between HER2+/- sub-types. These results illustrate application of SLTs in a topical applied problem; however, inference of network structure from biological data remains extremely challenging (Oates and Mukherjee, 2012) and experimental validation of inferred topology is necessary.

4 Discussion

In this paper we introduced a novel methodology, SLT, which generalises joint estimation of multiple networks to the non-exchangeable setting. Our empirical results support the notion that SLTs can offer improved estimation relative to existing estimators based on exchangeability. We illustrated the use of the

SLT framework using FFDBNs for which joint estimation could be carried out in a computationally efficient manner. However the general SLT approach is applicable in principle to any probabilistic network model for which marginal likelihoods are available. Thus, in principle SLT formulations could be developed for Bayesian networks, GGMs, or more sophisticated local likelihoods, for example based on differential equations (Nelander *et al.*, 2008).

In empirical studies we considered FFDBNs of dimension $P = 10$ and 17; in this setting, exact inference using SLTs was massively faster than (exchangeable) alternatives based on MCMC (Penfold *et al.*, 2012; Werhli and Husmeier, 2008). The (serial) computational complexity of our approach applied to a tree T is at worst $\mathcal{O}(h_1 h_2 \dots h_t c(P))$, where h_i is the number of networks that are tree distance i from the root network G_1 and t is the number of tiers in T. Thus in our cancer example, inclusion of more cancer sub-types or cell lines is computationally cheap (linear in both h_1 and h_2). For FFDBNs, $c(P) = P^{1+2d_{\max}}$ so that SLT has the same computational complexity as a fully exchangeable formulation (JNI), but requires $\mathcal{O}(P^{d_{\max}})$ more computation than the classical non-joint approach.

Extensions of theoretical interest include: (i) The case where T itself is unknown; here the challenge is to jointly learn both individual-specific networks and tree structure. In principle this could be accomplished using the SLT model described here, but further work would be needed to render this tractable for non-trivial applications. (ii) The case of arbitrarily-structured populations, where T need not be a tree, or where data may be associated with multiple networks; here MCMC methods similar to Dondelinger *et al.* (2012) or approximate inference algorithms such as loopy belief propagation may prove effective.

Acknowledgments

The authors wish to thank Frank Dondelinger, Steven Hill, Dan Woodcock and Chris Penfold. Data courtesy James Korkola and Joe W. Gray, Oregon Health and Science University. CJO supported by UK EPSRC EP/E501311/1. SM supported by NCI U54 CA112970 and the Cancer Systems Center grant from the Netherlands Organisation for Scientific Research.

References

Baumbach *et al.* (2009) Reliable transfer of gene regulatory networks between taxonomically related organisms. *BMC Bioinformatics* **3**:8.

Calderhead, B., Girolami, M. (2009) Estimating Bayes factors via thermodynamic integration and population MCMC. *Comput. Stat. Data An.* **53**:4028-4045.

Chiquet, J., Grandvalet, Y., Ambroise, C. (2011) Inferring multiple graphical structures. *Stat. Comput.* **21**(4):537-553.

Danaher, P., Wang, P., Witten, D.M. (2014) The joint graphical lasso for inverse covariance estimation across multiple classes. *J. R. Statist. Soc. B*, in press.

Deltell *et al.* (2012) Criteria for Bayesian Model Choice with Application to Variable Selection. *Ann. Stat.* **40**(3):1550-1577.

Dondelinger, F., Lebre, S., Husmeier, D. (2012) Non-homogeneous dynamic Bayesian networks with Bayesian regularization for inferring gene regulatory networks with gradually time-varying structure. *Mach. Learn.* **90**(2):191-230.

Gerlinger *et al.* (2012) Intratumor heterogeneity and branched evolution revealed by multiregion sequencing. *New Engl. J. Med.* **366**(10):883-892.

Guo *et al.* (2011) Joint estimation of multiple graphical models. *Biometrika* **98**(1):1-15.

Hennessy *et al.* (2010) A Technical Assessment of the Utility of Reverse Phase Protein Arrays for the Study of the Functional Proteome in Nonmicrodissected Human Breast Cancer. *Clin. Proteom.* **6**:129-151.

Hill *et al.* (2012) Bayesian Inference of Signaling Network Topology in a Cancer Cell Line. *Bioinformatics* **28**(21):2804-2810.

Mohan *et al.* (2014) Node-based learning of multiple Gaussian graphical models. *J. Mach. Learn. Res.*, in press.

Mukherjee, S, Speed, T.P. (2008) Network inference using informative priors. *Proc. Nat. Acad. Sci. USA* **105**(38):14313-14318.

Murphy, K. (2002) Dynamic Bayesian Networks: Representation, Inference and Learning. PhD Thesis, University of California, Berkeley.

Nelander *et al.* (2008) Models from experiments: combinatorial drug perturbations of cancer cells. *Mol. Syst. Biol.* **4**(1):216.

Neve *et al.* (2006) A collection of breast cancer cell lines for the study of functionally distinct cancer subtypes. *Cancer Cell* **10**(6):515-527.

Oates, C.J., Mukherjee, S. (2012) Network Inference and Biological Dynamics. *Ann. Appl. Stat.* **6**(3):1209-1235.

Oates *et al.* (2013) Joint Estimation of Multiple Networks from Time Course Data. *CRiSM Working Paper Series, University of Warwick* **13**:03.

Pearl, J. (1982) Reverend Bayes on inference engines: A distributed tree approach. *Proceedings of the Second National Conference on Artificial Intelligence. AAAI-82: Pittsburgh, PA. Menlo Park, California: AAAI Press.*, 133-136.

Pearl, J. (2009) Causal inference in statistics: An overview. *Stat. Surveys* **3**:96-146.

Penfold *et al.* (2012) Nonparametric Bayesian inference for perturbed and orthologous gene regulatory networks. *Bioinformatics* **28**(12):i233-i241.

Scott, J.G., Berger, J.O. (2010) Bayes and Empirical-Bayes Multiplicity Adjustment in the Variable-Selection Problem. *Ann. Stat.* **38**(5):2587-2619.

Spencer, S., Hill, S.M., Mukherjee, S. (2012) Dynamic Bayesian networks for interventional data. *CRiSM Working Paper Series, University of Warwick* **12**:24.

Taylor, M., Fox, C. (2011) Inventory Management with Dynamic Bayesian Network Software Systems. *Business Information Systems: Springer Lecture Notes in Business Information Processing* **87**:290-300.

Werhli, A.V., Husmeier, D. (2008) Gene regulatory network reconstruction by Bayesian integration of prior knowledge and/or different experimental conditions. *J. Bioinf. Comp. Biol.* **6**(3):543-572.

Yang *et al.* (2012) Fused Multiple Graphical Lasso. arXiv:1209.2139.

Zellner, A. (1986) On Assessing Prior Distributions and Bayesian Regression Analysis With g-Prior Distributions, *Bayesian Inference and Decision Techniques - Essays in Honor of Bruno de Finetti, eds. P. K. Goel and A. Zellner*, 233-243.

5 Supplement for "Joint Structure Learning of Multiple Non-Exchangeable Networks", Chris. J. Oates and Sach Mukherjee, AISTATS 2014.

5.1 Belief propagation for SLTs

Following marginalisation of continuous parameters $\boldsymbol{\theta}$, inference for SLTs reduces to inference for a discrete Bayesian network whose nodes are themselves graphical models (SFig. 4). In this Section we describe the use of belief propagation (BP; Pearl (1982)) for inference in this setting and provide pseudocode for the 2-tier SLT model.

Denote by $\boldsymbol{X}$ a vector of random variables whose density factorizes according to

$$p_{\boldsymbol{X}}(\boldsymbol{x}) = \prod_{f \in \mathcal{F}} f(\boldsymbol{x}_f) \tag{9}$$

where $\boldsymbol{x}_f$ denotes the components of vector $\boldsymbol{x}$ upon which the factor f depends. The factor graph corresponding to the 2-tier SLT model is shown in SFig. 4c. We use $\mu_{v \to f}$ to denote a message passed from a variable v to a factor f, whereas $\overline{\mu}_{f \to v}$ will be used to denote a message passed from a factor f to a variable v. The message from a variable v to a factor f takes the following form:

$$\mu_{v \to f}(x_v) = \prod_{f^* \in N(v) \setminus \{f\}} \overline{\mu}_{f^* \to v}(x_v) \tag{10}$$

where $N(v)$ denotes the neighbours of variable v according to the factor graph. Similarly the message from a factor f to a variable v takes the form

$$\overline{\mu}_{f \to v}(x_v) = \sum_{\boldsymbol{x}' : x'_v = x_v} f(\boldsymbol{x}'_f) \prod_{v^* \in N(f) \setminus \{v\}} \mu_{v^* \to f}(x'_{v^*}) \tag{11}$$

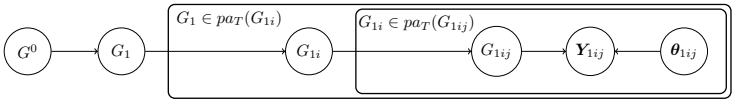
where $N(f)$ denotes the neighbours of factor f according to the factor graph.

To simplify notation, we describe our algorithm using subscript notation as in the Main Text; e.g. $\overline{G_{1ij}}$ denotes the network that is the jth child of the ith child of the root network $\overline{G_1}$ in T. BP nominates one node in the factor graph as a "root"; of the remaining nodes, those with degree one are known as "leaves". For BP applied to SLT we nominate the network $\overline{G_1}$ as the root node. Messages are initiated at the leaves of the factor graph; specifically, in our 2-tier example, each variable node $\boldsymbol{Y}_{1ij}$ is initialised with an atomic distribution $\delta\{\boldsymbol{Y}_{1ij} = \boldsymbol{y}_{1ij}\}$ centered on the observed data $\boldsymbol{y}_{1ij}$. Messages are passed through to the root node before being returned to the leaves.

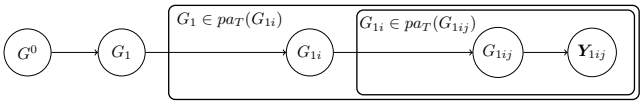
Once the message passing has been completed, it is possible to extract marginals of interest by taking products of messages from factors neighboring the random variable of interest:

$$p_{X_v}(x_v) \propto \prod_{f \in N(v)} \overline{\mu}_{f \to v}(x_v) \tag{12}$$

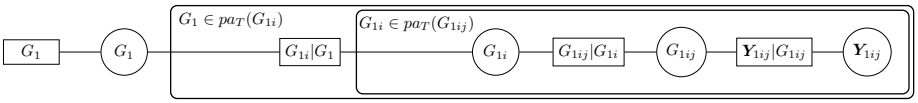
Alg. 1 contains pseudocode for the BP algorithm in the context of 2-tier SLTs.



(a) Graphical model representation.



(b) The marginal SLT.



(c) Factor graph representation.

Figure 4: Structure Learning Trees (SLT); 2-tier example. (a) Graphical model representation. [G^0 = prior network, G_1 = root network, G_{1i} = tier-1 networks, G_{1ij} = tier-2 networks, $\boldsymbol{Y}_{1ij}$ = data available on network G_{1ij}, $\boldsymbol{\theta}_{1ij}$ = parameters describing the distribution of the data $\boldsymbol{Y}_{1ij}$. Bounding boxes are used to denote multiplicity of variables.] (b) The marginal SLT is obtained from (a) by integrating out continuous parameters $\boldsymbol{\theta}_{1ij}$. (c) Factor graph representation of the marginal SLT (b). [Circled nodes are random variables, rectangular nodes are factors. Dependence on the prior network is suppressed.]

5.2 Simulation study

5.2.1 Data generation

From each network $\overline{G_{1ij}}$ we generated time series data $\boldsymbol{y}_{1ij}$, each containing n time points, according to a linear VAR(1) process. For each time series one variable was selected uniformly at random to be the target of a perfect intervention (Spencer and Mukherjee, 2012). Dynamical parameters were assigned such that for each edge (i, j) we select a data-generating coefficient $\beta \in \{-1, +1\}$ uniformly at random. For all experiments we used a noise magnitude $\sigma = 1$. In each regime we generated data of varying sample size n and edge density ρ. Specifically, we considered both varying n for fixed $\rho = 0.5$ and varying ρ for fixed $n = 60$.

5.2.2 Performance measures

Denote the true data-generating (binary) adjacency matrix by $\boldsymbol{A}^0$. In this work we considered the performance of two kinds of estimator; (i) the weighted adjacency matrices $\boldsymbol{A}$ produced by collecting together posterior marginal inclusion probabilities, and (ii) the binary adjacency matrices $\boldsymbol{A}(\tau)$ with (i, j)th entry $\mathbb{I}(A_{ij} > \tau)$, i.e. including edges if and only if the corresponding posterior marginal inclusion probabilities exceed a threshold τ. Write $\mathrm{TP}(\tau)$, $\mathrm{FP}(\tau)$, $\mathrm{TN}(\tau)$, $\mathrm{FN}(\tau)$ for, respectively, the true positive, false positive, true negative

Algorithm 1 Belief propagation (BP) for the 2-tier SLT model. Here we list all steps of the BP algorithm in order; at each stage messages are passed for all relevant networks indexed by i and j, but we leave this implicit for clarity.

1: $\mu_{\boldsymbol{Y}_{1ij}\to\boldsymbol{Y}_{1ij}|G_{1ij}}(\boldsymbol{Y}_{1ij}) = \delta\{\boldsymbol{Y}_{1ij} = \boldsymbol{y}_{1ij}\}$

2: $\overline{\mu}_{\boldsymbol{Y}_{1ij}|G_{1ij}\to G_{1ij}}(G_{1ij}) = \int_{\boldsymbol{Y}_{1ij}} p(\boldsymbol{Y}_{1ij}|G_{1ij})\mu_{\boldsymbol{Y}_{1ij}\to\boldsymbol{Y}_{1ij}|G_{1ij}}(\boldsymbol{Y}_{1ij})d\boldsymbol{Y}_{1ij}$

3: $\mu_{G_{1ij}\to G_{1ij}|G_{1i}}(G_{1ij}) = \overline{\mu}_{\boldsymbol{Y}_{1ij}|G_{1ij}\to G_{1ij}}(G_{1ij})$

4: $\overline{\mu}_{G_{1ij}|G_{1i}\to G_{1i}}(G_{1i}) = \sum_{G_{1ij}} p(G_{1ij}|G_{1i})\mu_{G_{1ij}\to G_{1ij}|G_{1i}}(G_{1ij})$

5: $\mu_{G_{1i}\to G_{1i}|G_1}(G_{1i}) = \prod_j \overline{\mu}_{G_{1ij}|G_{1i}\to G_{1i}}(G_{1i})$

6: $\overline{\mu}_{G_{1i}|G_1\to G_1}(G_1) = \sum_{G_{1i}} p(G_{1i}|G_1)\mu_{G_{1i}\to G_{1i}|G_1}(G_{1i})$

7: $\overline{\mu}_{G_1\to G_1}(G_1) = p(G_1)$

8: $\mu_{G_1\to G_{1i}|G_1}(G_1) = \overline{\mu}_{G_1\to G_1}(G_1)\prod_{i'\neq i}\overline{\mu}_{G_{1i'}|G_1\to G_1}(G_1)$

9: $\overline{\mu}_{G_{1i}|G_1\to G_{1i}}(G_{1i}) = \sum_{G_1} p(G_{1i}|G_1)\mu_{G_1\to G_{1i}|G_1}(G_1)$

10: $\mu_{G_{1i}\to G_{1ij}|G_{1i}}(G_{1i}) = \overline{\mu}_{G_{1i}|G_1\to G_{1i}}(G_{1i})$

11: $\overline{\mu}_{G_{1ij}|G_{1i}\to G_{1ij}}(G_{1ij}) = \sum_{G_{1i}} p(G_{1ij}|G_{1i})\mu_{G_{1i}\to G_{1ij}|G_{1i}}(G_{1i})$

12: $p(G_1|\boldsymbol{y}) = \overline{\mu}_{G_1\to G_1}(G_1)\prod_i \overline{\mu}_{G_{1i}|G_1\to G_1}(G_1)$

13: $p(G_{1i}|\boldsymbol{y}) = \overline{\mu}_{G_{1i}|G_1\to G_{1i}}(G_{1i})\prod_j \overline{\mu}_{G_{1ij}|G_{1i}\to G_{1i}}(G_{1i})$

14: $p(G_{1ij}|\boldsymbol{y}) = \overline{\mu}_{G_{1ij}|G_{1i}\to G_{1ij}}(G_{1ij})\overline{\mu}_{\boldsymbol{Y}_{1ij}|G_{1ij}\to G_{1ij}}(G_{1ij})$

and false negative counts obtained by comparing $\boldsymbol{A}(\tau)$ to $\boldsymbol{A}^0$. Further write $\mathrm{TPR}(\tau) = \mathrm{TP}(\tau)\ /\ (\mathrm{TP}(\tau) + \mathrm{FN}(\tau))$, $\mathrm{FPR}(\tau) = \mathrm{FP}(\tau)\ /\ (\mathrm{TN}(\tau) + \mathrm{FP}(\tau))$, $\mathrm{PPV}(\tau) = \mathrm{TP}(\tau)\ /\ (\mathrm{TP}(\tau) + \mathrm{FP}(\tau))$.

For (i) we considered the following performance measures:

(1) L1 Error $= \sum_{ij} |A_{ij} - A^0_{ij}|$

(2) Relative Density $= \sum_{ij} |A_{ij}| / \sum_{ij} |A^0_{ij}|$

(3) AUROC $= \int \mathrm{TPR}(\tau) d\mathrm{FPR}(\tau)$

(4) AUPR = $\int \text{PPV}(\tau) d\text{TPR}(\tau)$

For (ii) special attention is afforded to the "median" estimator with $\tau = 0.5$. Specifically we considered the performance measures

(1) Matthews Correlation Coefficient = $(\text{TP}\times\text{TN}-\text{FP}\times\text{FN})/\sqrt{((\text{TP}+\text{FP})(\text{TP}+\text{FN})(\text{TN}+\text{FP})(\text{TN}+\text{FN}))}$

(2) Misclassification Rate = (FP + FN) / P^2

(3) Misclassification Rate (top k edges) = As for the misclassification rate, but with τ chosen such that $\boldsymbol{A}(\tau)$ contains exactly k non-zero entries, where k is the number of edges in the true data-generating network.

(4) Precision = TP / (TP + FP).

5.2.3 Additional results

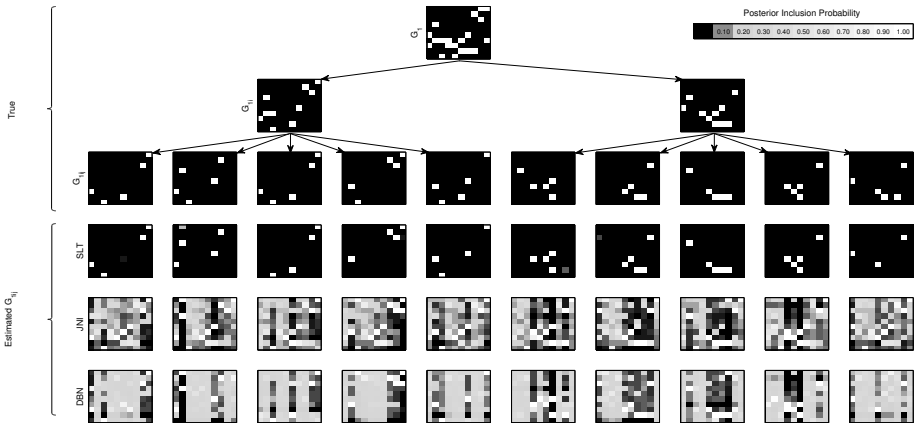
SFigs. 5a, 5b, 6a, 6b display typical simulation examples for regimes 2-5 respectively, and SFigs. 7-11 display full simulation results for each of the the 5 regimes described in the Main Text.

5.3 Experimental protocol

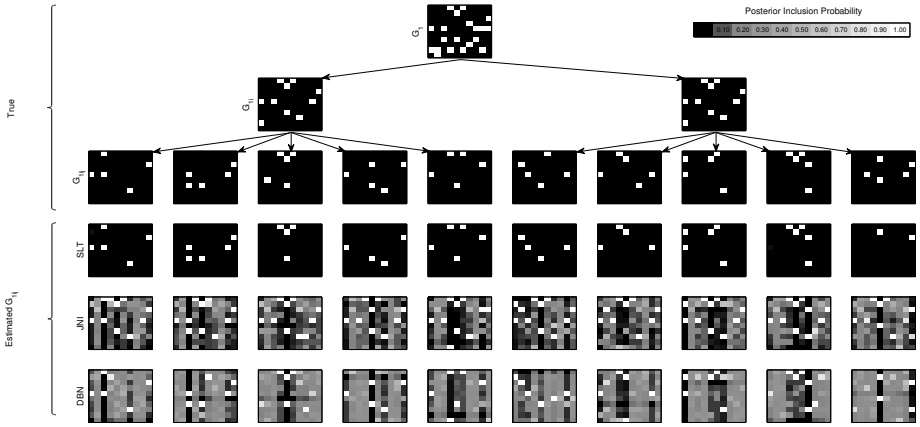
Cells were plated into 10 cm^2 dishes at a density of $1 - 2 \times 10^6$ cells. After 24 hours, cells were treated with 250 nM lapatinib or 250 nM AKTi (GSK690693). DMSO served as a control. Cells were grown in 10% FBS and harvested in RPPA lysis buffer at 30 min, 1h, 2h, 4h, 8h, 24h, 48h, and 72h post-treatment. Cell lysates were quantitated, diluted, arrayed, and probed following Tibes *et al.* (2006). Imaging and quantitation of signal intensity was done following Tibes *et al.* (2006). The particular protein species analysed were 4EBP1(pT37), AKT(pS473), BAD(pS112), c-Myc(pT58), EGFR(pY1173), ELK1(pS383), ER, FOXO3a(pS318), GSK3ab(pS21), HER2, IRS1(pS307), MAPK(pT202), MEK1/2(pS217), p38(pT180), p53, PR and S6(pS240).

References

Tibes *et al.* (2006) Reverse phase protein array: validation of a novel proteomic technology and utility for analysis of primary leukemia specimens and hematopoietic stem cells. *Mol. Cancer Ther.* **5**(10):2512-2521.

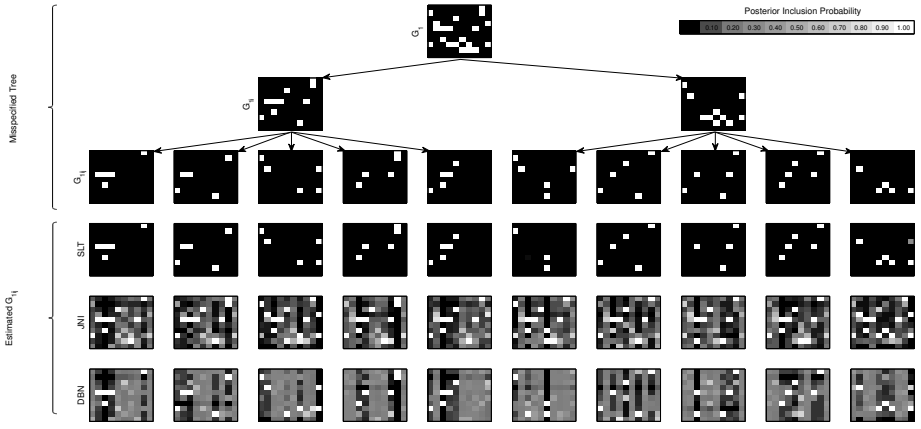


(a) Weakly exchangeable

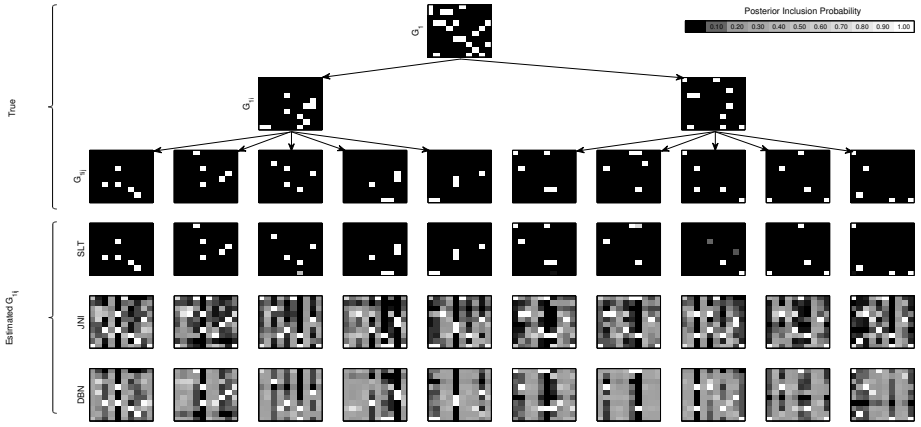


(b) Fully exchangeable

Figure 5: Results on simulated data generated from 2-tier SLTs; (a) a weakly exchangeable population, where $\overline{G_{11}}$, $\overline{G_{12}}$ are likely to share edges, and (b) a fully exchangeable population. [Inference methods: "SLT" = structure learning trees, "JNI" = joint network inference (Oates *et al.*, 2013), "DBN" = independent network inference (this corresponds to structure learning under the same local likelihood as SLT and JNI but applied separately to the data-generating networks located at the leaves of the tree).]



(a) Misspecified Tree



(b) Subset violation

Figure 6: Results on simulated data generated from 2-tier SLTs; (a) a misspecified tree structure T, and (b) a weakly exchangeable population which violates the subset assumptions encoded in the joint structural prior used by SLT. [Inference methods: "SLT" = structure learning trees, "JNI" = joint network inference (Oates *et al.*, 2013), "DBN" = independent network inference (this corresponds to structure learning under the same local likelihood as SLT and JNI but applied separately to the data-generating networks located at the leaves of the tree).]

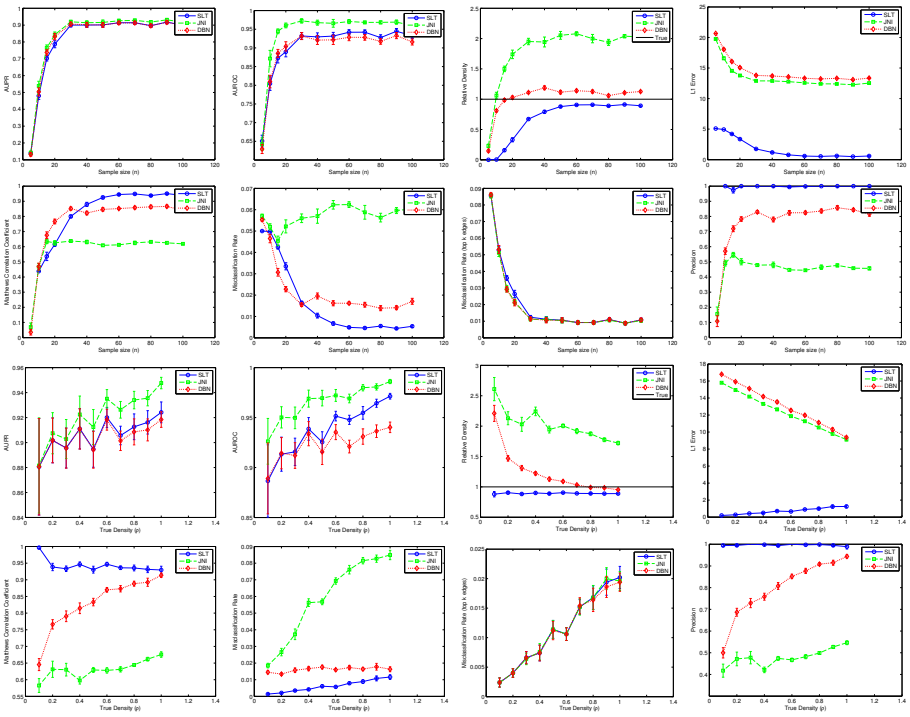


Figure 7: Results on simulated data generated from a 2-tier SLT with disjoint sub-group structure. [Network estimators: "SLT" = structure learning trees; "JNI" = joint network inference; "DBN" = inference for each network independently. For each estimator we considered both thresholded and unthresholded adjacency matrices. Performance scores: "AUROC" = area under the receiver operating characteristic curve; "AUPR" = area under the precision-recall curve; "L1 Error" = ℓ_1 distance from the true adjacency matrices to the inferred weighted adjacency matrices; "top k edges" = the ρP most probable edges. Performance scores were averaged over all 10 data-generating networks and all 10 datasets; error bars denote standard errors of mean performance over datasets. We considered both varying n for fixed $\rho = 0.5$ and varying ρ for fixed $n = 60$.]

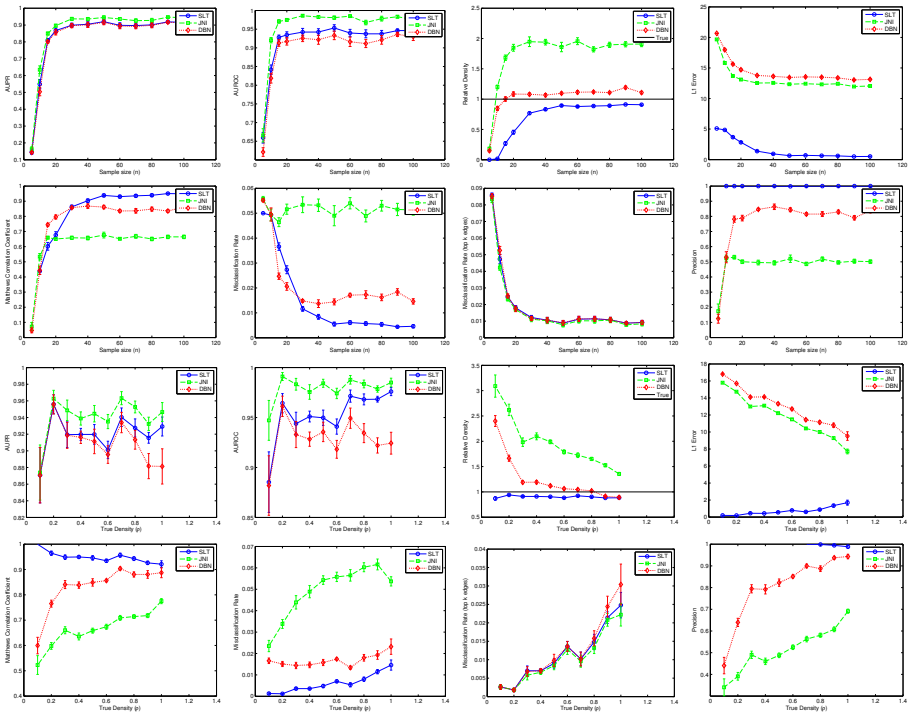


Figure 8: Results on simulated data generated from a 2-tier SLT with weakly exchangeable structure [Network estimators: "SLT" = structure learning trees; "JNI" = joint network inference; "DBN" = inference for each network independently. For each estimator we considered both thresholded and un-thresholded adjacency matrices. Performance scores: "AUROC" = area under the receiver operating characteristic curve; "AUPR" = area under the precision-recall curve; "L1 Error" = ℓ_1 distance from the true adjacency matrices to the inferred weighted adjacency matrices; "top k edges" = the ρP most probable edges. Performance scores were averaged over all 10 data-generating networks and all 10 datasets; error bars denote standard errors of mean performance over datasets. We considered both varying n for fixed $\rho = 0.5$ and varying ρ for fixed $n = 60$.]

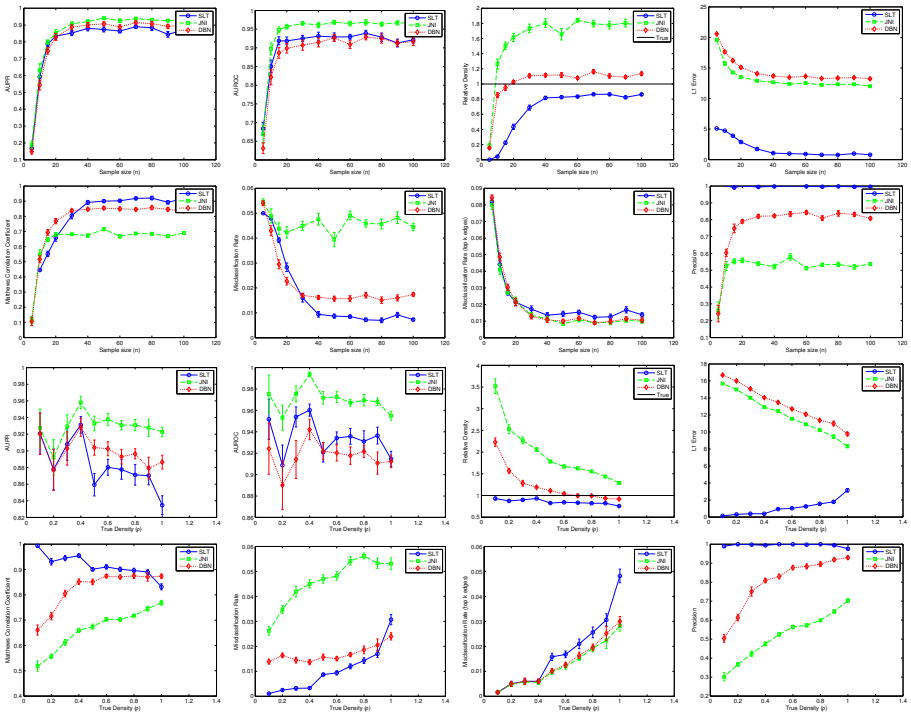


Figure 9: Results on simulated data generated from a fully exchangeable SLT. [Network estimators: "SLT" = structure learning trees (based on 2 tiers); "JNI" = joint network inference; "DBN" = inference for each network independently. For each estimator we considered both thresholded and un-thresholded adjacency matrices. Performance scores: "AUROC" = area under the receiver operating characteristic curve; "AUPR" = area under the precision-recall curve; "L1 Error" = ℓ_1 distance from the true adjacency matrices to the inferred weighted adjacency matrices; "top k edges" = the ρP most probable edges. Performance scores were averaged over all 10 data-generating networks and all 10 datasets; error bars denote standard errors of mean performance over datasets. We considered both varying n for fixed $\rho = 0.5$ and varying ρ for fixed $n = 60$.]

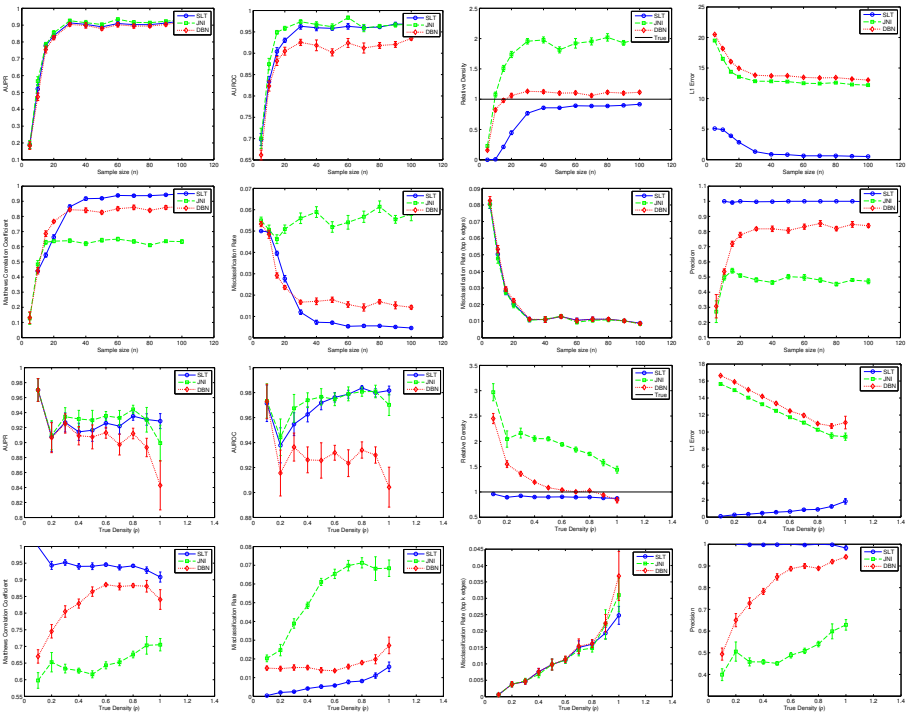


Figure 10: Results on simulated data generated from a 2-tier SLT with misspecified tree structure. [Network estimators: "SLT" = structure learning trees; "JNI" = joint network inference; "DBN" = inference for each network independently. For each estimator we considered both thresholded and unthresholded adjacency matrices. Performance scores: "AUROC" = area under the receiver operating characteristic curve; "AUPR" = area under the precision-recall curve; "L1 Error" = ℓ_1 distance from the true adjacency matrices to the inferred weighted adjacency matrices; "top k edges" = the ρP most probable edges. Performance scores were averaged over all 10 data-generating networks and all 10 datasets; error bars denote standard errors of mean performance over datasets. We considered both varying n for fixed $\rho = 0.5$ and varying ρ for fixed $n = 60$.]

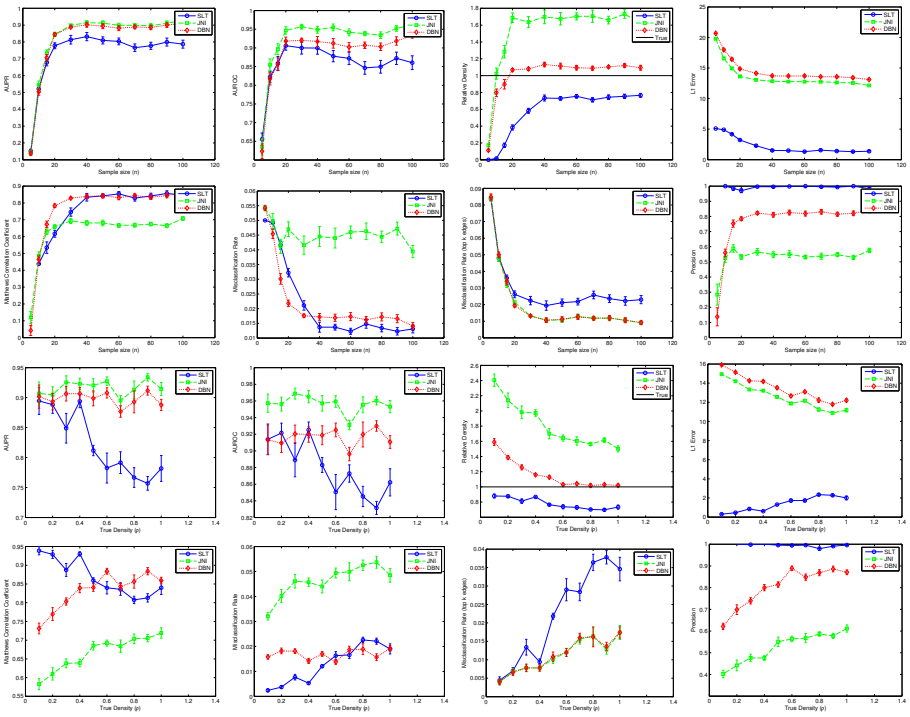


Figure 11: Results on simulated data generated from a 2-tier SLT with structure which violates the subset assumption. [Network estimators: "SLT" = structure learning trees; "JNI" = joint network inference; "DBN" = inference for each tier-3 network independently. For each estimator we considered both thresholded and un-thresholded adjacency matrices. Performance scores: "AUROC" = area under the receiver operating characteristic curve; "AUPR" = area under the precision-recall curve; "L1 Error" = ℓ_1 distance from the true adjacency matrices to the inferred weighted adjacency matrices; "top k edges" = the ρP most probable edges. Performance scores were averaged over all 10 data-generating networks and all 10 datasets; error bars denote standard errors of mean performance over datasets. We considered both varying n for fixed $\rho = 0.5$ and varying ρ for fixed $n = 60$.]