Variational Network Inference: Strong and Stable with Concrete Support

Amir Dezfouli [1 2] **Edwin V. Bonilla** [1] **Richard Nock** [3]

Abstract

Traditional methods for the discovery of latent network structures are limited in two ways: they either assume that all the signal comes from the network (i.e. there is no source of signal outside the network) or they place constraints on the network parameters to ensure model or algorithmic stability. We address these limitations by proposing a model that incorporates a Gaussian process prior on a network-independent component and formally proving that we get algorithmic stability for free, while providing a novel perspective on model stability as well as robustness results and precise intervals for key inference parameters. We show that, on three applications, our approach outperforms previous methods consistently.

1. Introduction

Networks represent the elements of a system and their interconnectedness as a set of *nodes* and *arcs* (connections) between them. Applications of network analysis range from biological systems such as gene regulatory networks and brain connectivity networks, to social networks and interactions between financial indices.

When dealing with continuous observations, a commonly used framework for this purpose is linear causal models (Bollen, 1989; Pearl, 2000; Spirtes et al., 2000), in which the data-generation process is defined such that the observations from each node are a linear combination of the observations from other nodes and additive noise with – typically – a constant mean and variance. Hence, temporal variations in the observations from a node are either associated to the other nodes in the network, or to the changes in latent confounders; i.e., in the absence of any change in these two components, observations from a node are assumed to follow the noise distribution and thus unaffected by the time-varying signals that come from outside the network. This comes as a significant limitation for real-world problems where observations from a node can also follow a network-independent trend. For example, when considering property prices in the suburbs of a city, some of them can follow a decreasing/increasing trend over time, essentially independent of other suburbs.

Our first contribution overcomes this limitation. We propose a network-structure discovery model that generalizes linear causal models by incorporating a network-independent component for each node, which is determined by a Gaussian process (GP) prior capturing the inter-dependencies between observations over time. Consequently, the output of a node is now given by a sum of the network-independent component and a (noisy) linear combination of the observations from the other nodes. Our model considers the parameters of this linear combination (which determine the structure of the network) as random variables. This modeling approach provides a more flexible data-generation process due to the non-parametric nature of the GP prior but, of course, raises the question of what algorithms can be used to learn such more general models.

Our second contribution provides an answer to this question, including an efficient variational inference approach for the posterior over the network-dependent parameters. A key part relies on showing that, by marginalizing the latent functions corresponding to the network-independent components, our approach is closely related to multi-task GP models under a product covariance (Bonilla et al., 2008; Rakitsch et al., 2013). This allows us to exploit properties of Kronecker products in order to compute the marginal likelihood (conditioned on the network parameters) efficiently. We estimate the posterior over the network-dependent parameters building upon recent breakthroughs in variational inference (Rezende et al., 2014; Kingma & Welling, 2014; Maddison et al., 2016).

Computational efficiency is not the only concern of previous popular approaches: all rely on more or less stringent assumptions to make sure that parameters do not deviate from a prescribed, *finite* regime. In short, "*There is a stability caveat to retrieve the network*". What finiteness is related

[1]UNSW, Sydney. [2]Started work at Data61. [3]Data61, the Australian National University and the University of Sydney. Correspondence to: Amir Dezfouli <akdezfuli@gmail.com>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.

to takes various forms: this can be the number of events for Hawkes process modeling (Linderman & Adams, 2014), the variance of the process (Shimizu et al., 2006), the iterated dynamical system parameters (Hyvärinen & Smith, 2013), the non-singularity of the mixing matrix (Shimizu et al., 2011), etc. It is legitimate to ask where our more general model and algorithm position us with respect to this caveat.

Our third contribution is a formal proof that stability is not an issue in our case. Concrete distributions (Maddison et al., 2016) happen to be important in our setting not just for their convenience in the reparameterization trick (Kingma & Welling, 2014): they help to get stability for free. Furthermore, we investigate what we get with the assumptions used in previous work to guarantee stability (Linderman & Adams, 2014). What we get, which to our knowledge has never been documented, is that key parameters are not just stable: with high probability, they are easy to bound and meet some form of statistical robustness. The variance of the network signal, for example, is of the same order as that of the network-independent parameters, and therefore robust to changes in the network-dependent distributions. This result is highly relevant, considering the choices we make to get tractable families of distributions over network parameters for variational inference.

Experiments. We benchmark our approach against the state of the art on three very different and challenging problems: discovering brain functional connectivity, modeling property prices in Sydney, and understanding regulation in the yeast genome. We provide a quantitative evaluation of our approach, showing that it consistently outperforms competitive baselines. Furthermore, when investigating the full yeast genome regulation, our qualitative analyses show that even in a large network (up to 38,000,000+ arcs), our technique is able to recover both high-level and low-level knowledge that is strikingly consistent with the previous literature and hints on original findings.

1.1. Related Work

Our approach is different from standard linear causal models with Gaussian noise (e.g. Bollen, 1989; Pearl, 2000) in three key aspects: (i) we do not assume that the underlying network is a directed acyclic graph (e.g. Spirtes et al., 2000); (ii) we represent the connection strengths using random matrices; and (iii) we incorporate the network-independent Gaussian process component. Concerning aspect (i), cyclic models are particularly important for the analysis of biological data in which the underlying networks typically include reciprocal connections (e.g., connections between different brain regions) or cycles (such as gene regulatory networks). Such models have been explored in the previous works, however, they differ from the current model in aspects (ii) and (iii) mentioned above. Examples of these studies include early work of Richardson (1996) and more recent works such as Hyttinen et al. (2012); Mooij et al. (2011) and Hyvärinen & Smith (2013). On a different vein and regarding aspect (ii) mentioned above, our use of random matrices representing network structure is similar to the model in Linderman & Adams (2014), but that model is focused on point-process data rather than continuous-valued observations.

With regard to aspect (iii), as observations in our model are generated from several latent Gaussian processes, our framework is related to GP latent variable models (Lawrence, 2005). However, our goal is to recover the underlying network structure, instead of carrying out dimensionality reduction or predicting observations for the nodes. Other models in this class can be used for causal inference (Zhang et al., 2010; Huang et al., 2015), which are different from our model in aspects (i) and (ii). With regard to multi-task GP models (Bonilla et al., 2008; Rakitsch et al., 2013) and more general frameworks for modeling vector-valued outputs (Wilson & Ghahramani, 2010), other approaches have considered Bayesian inference in multi-task learning subject to specific constraints, such as rank constrains (Koyejo & Ghosh, 2013). However, their work is mostly focused on dealing with the problem of high-dimensional data instead of network discovery. Finally, unlike our work, other approaches assume a non-Gaussian additive noise (Shimizu et al., 2006) or a nonlinear transformation of the network-dependent component (Hoyer et al., 2009).

2. Model Specification

Given a dataset $\mathcal{D}$ of vector-valued observations $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^N$ and their corresponding times[1] $\{\mathbf{t}_i\}_{i=1}^N$ from N nodes in a network, our goal is to infer the existence and strength of the arcs between the nodes. For simplicity in the notation, we assume that each observation $\mathbf{y}_i$ is T-dimensional and denote $n = N \times T$ as the total number of observations. Let $y_i(t)$ be the output of node i at time t,

$$y_i(t) = f_i(t) + \epsilon_{it}, \quad \epsilon_{it} \sim \mathcal{N}(0, \sigma_y^2), \tag{1}$$

where σ_y^2 is the observation-noise variance. To model latent function f_i, we assume that it is generated by two sources: (i) a network-independent component, denoted by $z_i(t)$, and (ii) a network-dependent component, i.e., a weighted sum of the inputs received from the rest of the network:

$$\begin{aligned} f_i(t) &= z_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^{N} A_{ij} W_{ij} \left[f_j(t) + \xi_{jt} \right], \\ z_i(t) &\sim \mathcal{GP}(\mathbf{0}, \kappa(t, t'; \boldsymbol{\theta})), \quad \xi_{jt} \sim \mathcal{N}(0, \sigma_f^2), \end{aligned} \tag{2}$$

[1] Although we refer to time indexes throughout this paper, the applicability of our method is not constrained to time-series data or one-dimensional inputs.

where $A_{ij} \in \{0, 1\}$ represents the existence of an arc from node j to node i and $W_{ij} \in \mathbb{R}$ determines the weight of the connection from node j to node i (assuming $A_{ii} = W_{ii} = 0$). These are elements of the adjacency matrix $\mathbf{A}$ and weight matrix $\mathbf{W}$, respectively, which we will refer to as network parameters. The network-independent component $z_i(t)$ is drawn from a Gaussian process (GP; Rasmussen & Williams, 2006) with covariance function $\kappa(t, t'; \boldsymbol{\theta})$ and hyperparameters $\boldsymbol{\theta}$.

2.1. Prior over Network Parameters

Eq. (1) defines the likelihood of our observations and eq. (2) defines the prior over the latent functions given the network parameters $\mathbf{A}, \mathbf{W}$. We assume these parameters are also random variables and their prior is defined as:

$$p(\mathbf{A}, \mathbf{W}) = p(\mathbf{A})p(\mathbf{W}) = \prod_{ij} p(A_{ij})p(W_{ij}),$$
$$p(A_{ij}) = \text{Bern}(\rho), \quad p(W_{ij}) = \mathcal{N}(0, \sigma_w^2), \tag{3}$$

where $\text{Bern}(\rho)$ denotes a Bernoulli distribution with parameter ρ. Note that defining separate priors over $\mathbf{A}$ and $\mathbf{W}$ – as above – allows us to specify separately our beliefs about the sparsity of the network connections and their strengths. Furthermore, these priors can be more effective than weak-sparsity inducing priors (Mohamed et al., 2012).

3. Inference

Our main inference task is to estimate the posterior over the network parameters $p(\mathbf{A}, \mathbf{W}|\mathcal{D})$. To this end, by exploiting the closeness of GPs under linear operators, we will first show in §3.1 the exact expression for the (conditional) marginal likelihood $p(\mathbf{Y}|\mathbf{A}, \mathbf{W})$ obtained when marginalizing the latent functions. Furthermore, by establishing a relationship of our model to multi-task learning (Rakitsch et al., 2013; Bonilla et al., 2008), we show how to compute this marginal likelihood efficiently. Subsequently, due to the highly nonlinear dependence of $p(\mathbf{Y}|\mathbf{A}, \mathbf{W})$ on $\mathbf{A}, \mathbf{W}$, we will approximate the posterior over these network parameters using variational inference in §3.2.

3.1. Marginal Likelihood given Network Parameters

Let us denote the values of all latent functions $f_i(t)$ at time t with $\mathbf{f}(t) = [f_1(t), \ldots, f_N(t)]$, and similarly $\mathbf{z}(t) = [z_1(t), \ldots, z_N(t)]$. Hence, we can rewrite eq. (2) as:

$$\mathbf{f}(t) = (\mathbf{I} - \mathbf{A} \odot \mathbf{W})^{-1}(\mathbf{z}(t) + \mathbf{A} \odot \mathbf{W}\boldsymbol{\xi}_t), \tag{4}$$

where $\odot$ is the Hadamard product. We refer to the model in eq. (4) as the *inverse model* and its the detailed derivation is in the supplement (§II.4). We can see now that, for fixed $\mathbf{A}, \mathbf{W}$, the resulting distribution over f_i is also a Gaussian process. Hence, we only need to figure out the mean function and the covariance function of the resulting process. Below we present the main results and leave the details to the supplement (§II.4).

Let $\mathbf{B} \stackrel{\text{def}}{=} \mathbf{A} \odot \mathbf{W}$ and define the following intermediate matrices (which are a function of the network parameters):

$$\mathbf{E} = (\mathbf{I} - \mathbf{B})^{-1}\mathbf{B}\mathbf{B}^T(\mathbf{I} - \mathbf{B})^{-T},$$
$$\mathbf{K}_f = (\mathbf{I} - \mathbf{B})^{-1}(\mathbf{I} - \mathbf{B})^{-T}. \tag{5}$$

Then we have that the mean function and covariance function of the latent processes are:

$$\mu_i(t) = \mathbb{E}\left[f_i(t)\right] = 0,$$
$$\mathbb{C}\text{ov}[f_i(t), f_j(t')] = [\mathbf{K}_f]_{i,j}\kappa(t, t'; \boldsymbol{\theta}) + [\mathbf{E}]_{i,j}\sigma_f^2, \tag{6}$$

where $[\mathbf{M}]_{i,j}$ denotes the i, j entry of matrix $\mathbf{M}$. Consequently, the distribution of the noisy process y_i is also a Gaussian process. Let us assume synchronized observations, i.e. that observations for all nodes lie on a grid in time, $t = 1, \ldots, T$. Furthermore, let $\mathbf{Y}$ be the $N \times T$ matrix of observations and define $\mathbf{y} = \text{vec}(\mathbf{Y})$, where $\text{vec}(\cdot)$ takes the columns of the matrix argument and stacks them into a single vector. Therefore, the log-marginal likelihood conditioned on the network parameters is given by ($\otimes$ is Kronecker product):

$$\log p(\mathbf{y}|\mathbf{A}, \mathbf{W}) = -\frac{1}{2}\log|\boldsymbol{\Sigma}_y| - \frac{1}{2}\mathbf{y}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{y} + C,$$
$$\text{with} \quad \boldsymbol{\Sigma}_y = \mathbf{K}_f \otimes \mathbf{K}_t + (\sigma_f^2\mathbf{E} + \sigma_y^2\mathbf{I}) \otimes \mathbf{I}, \tag{7}$$

where $C = -n/2\log(2\pi)$; $\mathbf{K}_t$ is the $T \times T$ covariance matrix induced by evaluating the covariance function $\kappa(t, t'; \boldsymbol{\theta})$ at all observed times; $\mathbf{E}$ and $\mathbf{K}_f$ are defined as in eq. 5; and, as before, $n = N \times T$ is the total number observations.

3.1.1. Relationship with multi-task learning

Remarkably, the marginal likelihood of the model described in eq. (7) reveals an interesting relationship with multi-task learning when using Gaussian process priors. Indeed, it boils down to the marginal likelihood of multi-task GP models under a product covariance (Bonilla et al., 2008; Rakitsch et al., 2013). In our case, the nodes in the network can be seen as the tasks in a multi-task GP model and are associated with a task-dependent covariance $\mathbf{K}_f$, which is fully determined by the parameters of the network $\mathbf{A}, \mathbf{W}$. This contrasts with multi-task models where $\mathbf{K}_f$ is, in general, a free parameter (Bonilla et al., 2008). Similarly, the input covariance $\mathbf{K}_t$ is the covariance of the observation times.

Finally, conditioned on $\mathbf{A}, \mathbf{W}$, our model's marginal likelihood exhibits a more complex noise covariance $\sigma_f^2\mathbf{E} + \sigma_y^2\mathbf{I}$, which depends strongly on the network parameters. Such a covariance structured was not studied by Bonilla et al.

(2008), as they considered only diagonal noise-covariances. However, Rakitsch et al. (2013) did consider the more general case of Gaussian systems with a covariance given by the sum of two Kronecker products. In the following section, we exploit their results in order to compute, for fixed $\mathbf{A}, \mathbf{W}$, the marginal likelihood of our model.

3.1.2. COMPUTATIONAL EFFICIENCY

In this section we show an efficient expression for the computation of the log-marginal likelihood in eq. (7). For simplicity, we consider the synchronized case where all the N nodes in the network have T observations at the same times and, as before, we denote the total number of observations with $n = N \times T$. The main difficulties of computing the log-marginal likelihood above are the calculation of the log-determinant of an n dimensional matrix, as well as solving an n-dimensional system of linear equations. Our goal is to show that we never need to solve these operations on an n-dimensional matrix, which are $\mathcal{O}(n^3)$ but instead use $\mathcal{O}(N^3 + T^3)$ operations. The results in this section have been previously shown by Rakitsch et al. (2013) for covariances with a sum of two Kronecker products.

We show our derivations in the supplement (§II.5) and present the results specific to our model here. To give some intuition behind such derivations, the main idea is to "factor-out" the noise matrix $\sigma_f^2\mathbf{E} + \sigma_y^2\mathbf{I}$ from the covariance matrix $\mathbf{\Sigma}_y$ and then apply properties of the Kronecker product. Hence, given the following matrix definitions along with their eigen-decompositions:

$$\begin{aligned} \mathbf{\Omega} &\stackrel{\text{def}}{=} (\sigma_f^2\mathbf{E} + \sigma_y^2\mathbf{I}) = \mathbf{Q}_\Omega \mathbf{\Lambda}_\Omega \mathbf{Q}_\Omega^T, \\ \tilde{\mathbf{K}}_f &\stackrel{\text{def}}{=} \mathbf{\Lambda}_\Omega^{-1/2} \mathbf{Q}_\Omega^T \mathbf{K}_f \mathbf{Q}_\Omega \mathbf{\Lambda}_\Omega^{-1/2} = \tilde{\mathbf{Q}}_f \tilde{\mathbf{\Lambda}}_f \tilde{\mathbf{Q}}_f^T; \end{aligned} \tag{8}$$

the log-determinant and the quadratic term in eq. (7) are

$$\begin{aligned} \log|\mathbf{\Sigma}_y| &= T\sum_{i=1}^{N} \log \lambda_\Omega^{(i)} + \sum_{i=1}^{N}\sum_{j=1}^{T} \log(\tilde{\lambda}_f^{(i)} \tilde{\lambda}_t^{(j)} + 1), \\ \mathbf{y}^T \mathbf{\Sigma}_y^{-1} \mathbf{y} &= \text{tr}(\tilde{\mathbf{Y}}^T \tilde{\mathbf{Q}}_t \tilde{\mathbf{Y}}_{tf} \tilde{\mathbf{Q}}_f^T), \end{aligned} \tag{9}$$

where: $[\tilde{\mathbf{Y}}_{tf}]_{i,j} = [\tilde{\mathbf{Q}}_t^T \tilde{\mathbf{Y}} \tilde{\mathbf{Q}}_f)]_{i,j} / [\tilde{\boldsymbol{\lambda}}_t \tilde{\boldsymbol{\lambda}}_f^T + 1]_{i,j}$, $\tilde{\mathbf{Y}} = \mathbf{Y}\mathbf{Q}_\Omega \mathbf{\Lambda}_\Omega^{-1/2}$, and $\tilde{\mathbf{Q}}_t \tilde{\mathbf{\Lambda}}_t \tilde{\mathbf{Q}}_t^T$ is the eigen-decomposition of $\mathbf{K}_t$. We see that the above computations only require the eigen-decompositions of the $N \times N$ matrix $\tilde{\mathbf{K}}_f$ and the $T \times T$ matrix $\mathbf{K}_t$, while avoiding matrix operations on the whole $n \times n$ matrix of covariances $\mathbf{\Sigma}_y$.

3.2. Variational Inference over Network Parameters

Having marginalized the latent functions $\mathbf{f}$ corresponding to the network-independent component, our next step is to use variational inference to approximate the true posterior $p(\mathbf{A}, \mathbf{W}|\mathcal{D})$ with a tractable family of distributions $q(\mathbf{A}, \mathbf{W})$ that factorizes as

$$\begin{aligned} q(\mathbf{A}, \mathbf{W}) =& q(\mathbf{A})q(\mathbf{W}) = \prod_{i,j} q(A_{ij})q(W_{ij}), \\ & i, j = 1 \dots N, \text{ and } i \neq j. \end{aligned} \tag{10}$$

Following standard variational-inference arguments, we aim to optimize the variational objective, so-called evidence lower-bound ($\mathcal{L}_{\text{elbo}}$), which is given by:

$$\begin{aligned} \mathcal{L}_{\text{elbo}} &\stackrel{\text{def}}{=} \mathcal{L}_{\text{kl}} + \mathcal{L}_{\text{ell}}, \\ \mathcal{L}_{\text{kl}} &= -\text{KL}(q(\mathbf{A}, \mathbf{W}) || p(\mathbf{A}, \mathbf{W})), \\ \mathcal{L}_{\text{ell}} &= \mathbb{E}_{q(\mathbf{A},\mathbf{W})}[\log p(\mathbf{Y}|\mathbf{A}, \mathbf{W})], \end{aligned} \tag{11}$$

where $\text{KL}(q||p)$ denotes the Kullback-Leibler divergence between distributions q and p, and $p(\mathbf{A}, \mathbf{W})$ is the prior over the network-dependent parameters as defined in eq. (3).

For non-trivial models and approximate posteriors, the expectations required in the objective above are analytically intractable. Modern variational inference methods estimate $\mathcal{L}_{\text{elbo}}$ and its gradients using Monte Carlo samples and the re-parameterization trick (see e.g. Kingma & Welling, 2014; Rezende et al., 2014). Thus, we set our approximate posterior over W_{ij} as $q(W_{ij}) = \mathcal{N}(\mu_{ij}, \sigma_{ij}^2)$, which can be re-parameterized easily using $W_{ij} = \mu_{ij} + \sigma_{ij} z_w$, where $z_w \sim \mathcal{N}(0, 1)$. Furthermore, since the re-parameterization trick cannot be applied to discrete distributions, we use a continuous relaxation of discrete random variables known as the Concrete distribution (Maddison et al., 2016; Jang et al., 2016). In particular we set $q(A_{ij}) = \text{Concrete}(\alpha_{ij}, \lambda_c)$, and sample from it using its re-parameterization:

$$\begin{aligned} \mathcal{U} &\sim \text{Uniform}(0, 1), \\ a_{ij} &= (\log \alpha_{ij} + \log \mathcal{U} - \log(1 - \mathcal{U}))/\lambda_c, \\ A_{ij} &= 1/(1 + \exp(-a_{ij})), \end{aligned} \tag{12}$$

where α_{ij} are variational parameters and λ_c is a constant. Analogously to Maddison et al. (2016), we also relax our priors and estimate the log-probabilities in $\mathcal{L}_{\text{kl}}$ using:

$$\begin{aligned} \log q(A_{ij}) =& \log \lambda_c - \lambda_c a_{ij} + \log \alpha_{ij} \\ & - 2\log(1 + \exp(-\lambda_c a_{ij} + \log \alpha_{ij})), \end{aligned} \tag{13}$$

and similarly for $p(A_{ij})$. Having relaxed our discrete variables, we proceed with optimization of the $\mathcal{L}_{\text{elbo}}$ in eq. (11) by using Monte Carlo samples from $q(\mathbf{W}, \mathbf{A})$ to estimate $\mathcal{L}_{\text{ell}}$. For computing $\text{KL}(q(\mathbf{A})||p(\mathbf{A}))$ we use samples from $q(\mathbf{A}), p(\mathbf{A})$ and their log-probabilities as defined in eq. (13). Finally, for $\text{KL}(q(\mathbf{W})||p(\mathbf{W}))$ we use the analytical form for the KL-divergence between two Gaussians.

As a consequence of the relaxation of the prior and posterior over $\mathbf{A}$ via the Concrete distribution, defining a sound joint (dependent) distribution between $\mathbf{A}$ and $\mathbf{W}$ is quite

challenging, which motivates the independence assumptions made in eq. (10). However, $\mathbf{A}$ and $\mathbf{W}$ still interact in the likelihood and these interactions are captured during variational learning.

4. Stability and Robustness

$\mathcal{L}_{\text{kl}}$ is straightforward to compute as explained in section 3.2, so the eventual stability burden relies on calculating $\log p(\mathbf{y}|\mathbf{W}, \mathbf{A})$ in $\mathcal{L}_{\text{ell}}$ using samples of $\mathbf{W}$ and $\mathbf{A}$. At the core of this problem lies the non-singularity of $(\mathbf{I} - \mathbf{B})$. This problem appears in the most popular approaches as well, sometimes as is (Shimizu et al., 2011), sometimes coming with stronger constraints on the boundedness of the coordinates of $\mathbf{B}$ (Hyvärinen & Smith, 2013) or its eigenvalues (Linderman & Adams, 2014). Such constraints can be related to stability issues because, when they fail to hold, parameters diverge. What we now show is that stability is not an issue for our model: we get it for free.

Theorem 1 *For any value of the parameters of the concrete distributions ($\lambda_c \geq 0$ and $\alpha_{ij} \geq 0$ ($\forall i \neq j$)), $\mathbf{I} - \mathbf{A} \odot \mathbf{W}$ is non-singular with probability one.*

Proof sketch: The proof (given in extenso in supplement, §II.1) is non-trivial to handle the limit cases of $\lambda_c = 0$ or $\alpha_{ij} = 0$. To understand the importance of concrete distributions, we sketch here the case $\lambda_c > 0$ or $\alpha_{ij} > 0$ ($\forall i \neq j$). For any[2] $N \geq 2$, denote $\mathbf{g}_1, \mathbf{g}_2, ..., \mathbf{g}_N$ the columns of $\mathbf{I} - \mathbf{A} \odot \mathbf{W}$. Each can be thought of as a random vector where one coordinate takes value 1 with probability 1, and this coordinate is different for each two vectors. $\mathbf{I} - \mathbf{A} \odot \mathbf{W}$ is non invertible iff $\mathbf{g}_1, \mathbf{g}_2, ..., \mathbf{g}_N$ are linearly dependent. *None* of the $\mathbf{g}_j$s can be the null vector, so if $\mathbf{I} - \mathbf{A} \odot \mathbf{W}$ is not invertible, then $\exists j > 1 : \mathbf{g}_j \in \text{span}(\mathbf{g}_1, \mathbf{g}_2, ..., \mathbf{g}_{j-1})$. As a consequence,

$$\begin{aligned} &\Pr(\det(\mathbf{I} - \mathbf{A} \odot \mathbf{W}) = 0) \\ &\quad \leq \sum_j \Pr(\mathbf{g}_j \in \text{span}(\mathbf{g}_1, \mathbf{g}_2, ..., \mathbf{g}_{j-1})) \ , \quad (14) \end{aligned}$$

where the distribution is the product distribution over the columns of $\mathbf{I} - \mathbf{A} \odot \mathbf{W}$. Fix *any* $\mathbf{g}_1, \mathbf{g}_2, ..., \mathbf{g}_{j-1}$ belonging to the respective supports of the columns, and let $q_j \stackrel{\text{def}}{=} \Pr(\mathbf{g}_j \in \text{span}(\mathbf{g}_1, \mathbf{g}_2, ..., \mathbf{g}_{j-1}) | \mathbf{g}_1, \mathbf{g}_2, ..., \mathbf{g}_{j-1})$. It is not hard to check that the densities of $\mathbf{g}_j$ for $j \geq 1$ are all absolutely continuous with respect to Lebesgue measure — a key fact, developed in supplement, authorized by the fact that the concrete distribution in eq. (13) "passes through" the absolute continuity of the input distribution (uniform). Along with the fact that $\text{span}(\mathbf{g}_1, \mathbf{g}_2, ..., \mathbf{g}_{j-1})$ has strictly positive codimension for any $j \leq N$, it comes $q_j = 0, \forall j \geq 2, \forall \mathbf{g}_1, \mathbf{g}_2, ..., \mathbf{g}_{j-1}$ fixed. Integrating over the choices of $\mathbf{g}_1, \mathbf{g}_2, ..., \mathbf{g}_{j-1}$, we get $\Pr(\mathbf{g}_j \in \text{span}(\mathbf{g}_1, \mathbf{g}_2, ..., \mathbf{g}_{j-1})) = 0, \forall j \leq N$ and so $\Pr(\det(\mathbf{I} - \mathbf{A} \odot \mathbf{W}) = 0) = 0$ from eq. (14). As a consequence, $\mathbf{I} - \mathbf{A} \odot \mathbf{W}$ is non-singular with probability one, as claimed.

Now, if say $\lambda_c = 0$ or some $\alpha_{ij} = 0$, *some* atom events for column sampling appear with non-zero probability, but the associated determinant can be reduced to that of a squared submatrix for which the previous analysis holds, leading to the same result. □

Given Theorem 1, the following result is not surprising.

Theorem 2 *For any value of the parameters of the concrete distributions ($\lambda_c \geq 0, \alpha_{ij} \geq 0$ ($i \neq j$)), any $\sigma_y^2 > 0$, $|\mathcal{L}_{ell}| \ll \infty$.*

The complete proofs of these theorems are given in the supplement, §II.1, §II.2.

Since we get stability for free, where other popular approaches need to make assumptions to get it, one might ask what more we can get under assumptions that would look alike. Such assumptions constrain the network parameters, typically using the moments or values, eventually including the network size (Hyvärinen & Smith, 2013; Linderman & Adams, 2014). What we now show is that under similar assumptions, we do not just get stability for inference, we make it numerically easy with high probability, and this holds *for a sampling model (M) more general than ours*, meaning that one could make alternative choices to the concrete distributions we use and yet keep the same property:

(M) $(\forall i, j)$ (i) weight W_{ij} is picked as $\mathcal{N}(\mu_{ij}, \sigma_{ij}^2)$ ($\mu_{ij} \in \mathbb{R}, \sigma_{ij} > 0$), and (ii) adjacency A_{ij} is picked as $\text{Bern}(\rho_{ij})$ with $\rho_{ij} \sim \mathcal{V}$, where $\mathcal{V}$ is any random variable with support in $[0, 1]$ (letting $p_{ij} \stackrel{\text{def}}{=} \mathbb{E}[\rho_{ij}]$).

To get our result, we need two functions that aggregate the complete *network signal* coming from (or to) each node, $U, S : \{1, 2, ..., 2N\} \to \mathbb{R}_+$, defined as:

$$U(i) \stackrel{\text{def}}{=} \frac{2}{N} \cdot \left\{ \begin{array}{ll} \sum_j p_{ij}(\mu_{ij}^2 + \sigma_{ij}^2) & \text{if} \quad i \leq N \\ \sum_j p_{ji^*}(\mu_{ji^*}^2 + \sigma_{ji^*}^2) & \text{otherwise} \end{array} \right. ,$$

$$S(i) \stackrel{\text{def}}{=} \frac{1}{N} \cdot \left\{ \begin{array}{ll} \sum_j \mu_{ij}^2 + \sigma_{ij}^2 & \text{if} \quad i \leq N \\ \sum_j \mu_{ji^*}^2 + \sigma_{ji^*}^2 & \text{otherwise} \end{array} \right. ,$$

with $i^* \stackrel{\text{def}}{=} i - N$. For any diagonalizable $\mathbf{U}$, $\lambda(\mathbf{U})$ denotes its eigenspectrum, and $\lambda^{\uparrow}(\mathbf{U}) \stackrel{\text{def}}{=} \max |\lambda(\mathbf{U})|$, $\lambda^{\downarrow}(\mathbf{U}) \stackrel{\text{def}}{=} \min |\lambda(\mathbf{U})|$. Let us now state our main result. We shall comment afterwards the assumptions it makes.

Theorem 3 *Fix any constants $c > 0$ and $0 < \gamma < 1$ and let $\lambda_\circ \stackrel{def}{=} (\lambda^{\downarrow}(\mathbf{K}_t)/2) + \sigma_y^2$, $\lambda_\bullet \stackrel{def}{=} 2\lambda^{\uparrow}(\mathbf{K}_t) + \sigma_f^2 + \sigma_y^2$ and $g(z, \mathbf{y}) \stackrel{def}{=} (N/2) \log z + z\|\mathbf{y}\|_2^2 - C$, where C is defined as in (7). Under sampling model M, suppose that*

$$\max_i U(i) \quad \in \quad \left[\frac{\max_i S(i)}{N^\gamma}, \frac{1}{100N^2} \right] \ . \quad (15)$$

[2] Whenever $N = 1$, $\mathbf{I} - \mathbf{A} \odot \mathbf{W} \stackrel{\text{def}}{=} [1]$ is always invertible.

If N is larger than some constant depending on c and γ, then with probability $\geq 1 - 1/N^c$, we have:

$$-\log p(\mathbf{y}|\mathbf{W}, \mathbf{A}) \quad \in \quad [g(\lambda_\circ, \mathbf{y}), g(\lambda_\bullet, \mathbf{y})] \ , \forall \mathbf{y} \quad (16)$$

Hence, if the non-network dependent "signal" is not flat (say, $\lambda^{\downarrow}(\mathbf{K}_t)$, σ_y^2 are above machine zero), then it is in fact pretty easy to sample $\mathcal{L}_{\text{ell}}$ over most of its support. As discussed in the supplement, the constraint of eq. (15) can be weakened for specific $\mathcal{V}$s (*e.g.* for more "informative" distributions). We also remark that we do not face the sparsity constraints of the model of Linderman & Adams (2014), such as a mandatory sparsity increase with N.

Theorem 3 is a direct consequence of another Theorem which can be roughly summarized as:

"*modulo an assumption on the network-dependent parameters, the covariance of observations is of the same order as the (co)variance of the network independent component plus that of the noise*."

In short, there is a form of *robustness* (in the statistical sense) achieved on the output with respect to the network-dependent parameters. This can also be viewed as a balance achieved on the second-order moments, between the network-dependent "signal" versus the one which is not network-dependent. We put signal in quotes since the rest includes the noise parameters.

Theorem 4 *Under the conditions of Theorem 3, with probability $\geq 1 - (1/N^c)$ over the sampling of* $\mathbf{W}$ *and* $\mathbf{A}$*, we have that $\lambda(\mathbf{\Sigma}_y) \subset [\lambda_\circ, \lambda_\bullet]$.*

(Proof in supplement, §II.3.) It is not hard to check that eq. (16) is a direct consequence of Theorem 4. Let us finally comment those assumptions on the network-dependent parameters made in eq. (15). To be nonempty, the interval puts the implicit constraint that $\max_i S(i) = O(1/N^\zeta)$ for some constant ζ, *i.e.* roughly, the expected square signal (node-wise) has to be bounded. Such a bound in the signal's values can be found in Hyvärinen & Smith (2013). Consider now the upperbound in eq. (15). It is quantitatively not so different from Linderman & Adams (2014)'s assumptions. They consider two assumptions, the first of which being[3]

$$\sigma^2 \quad \leq \quad \frac{1}{N}, \qquad (17)$$

and also pick network parameters μ, σ in such a way that large deviations for edge weights are controlled with high probability, with a condition that roughly looks like:

$$\mu^2 + \frac{c}{N^2} \cdot \sigma^2 \quad = \quad O\left(\frac{1}{N^2}\right) \ , \qquad (18)$$

for some constant $c > 0$. The constraint put on the maximum node-wise network signal in eq. (15) is in fact quite similar to the constraints imposed by eqs. (17) and (18).

[3]We consider variances for the assumption to rely on the same scales as ours.

Summary of theoretical results and consequences: "Stability", as used in various works, takes on two forms, describing either the stability of the model (Linderman & Adams, 2014; Shimizu et al., 2006) or the numerical stability of the learning algorithm (Shimizu et al., 2011). Our results contribute to *both*: while Theorems 1 and 2 are essentially numerical stability results, the robustness result of Theorem 4 is a model stability result, since it bounds the overall model's signal as a function of the external signal to the network. Theorem 3 lies in between, but its key purpose may be more practical. It says that one can approximate some key variational inference parameters with high probability, which can therefore save time at the expense of an affordable approximation.

5. Experiments

We evaluate our approach on three distinct domains: discovering brain functional connectivity (BRAIN), modeling property prices in Sydney (SYDNEY) and regulation in the yeast genome (YEAST). We considered the methods used recently by Peters et al. (2014) as baselines for comparison. These include: (1) PC algorithm (Spirtes et al., 2000); (2) Conservative PC algorithm (CPC, Ramse et al., 2006); and (3) LiNGAM (Shimizu et al., 2006). In addition to the above, we considered (4) IAMB (Tsamardinos et al., 2003), and (5) Pairwise LiNGAM (PW-LINGAM, Hyvärinen & Smith, 2013), which is a cyclic model and has been developed specifically for discovering connectivity between different brain regions. For the reasons detailed in the supplement (§III), other methods used in Peters et al. (2014) were not applicable to the datasets analyzed here. We used the squared exponential covariance function. For more details of the baseline methods, prior setting and optimization specifics see the supplement, §III. Given the posterior distribution $q(A_{ij})$, the posterior probability of existence of a connection from node j to i was calculated as $\alpha_{ij}/(1 + \alpha_{ij})$.

5.1. BRAIN Domain

Here we want to discover the connectivity between different brain regions, which is a crucial element of neuroscience studies. We analyzed the benchmarks of Smith et al. (2011), in which the activity of different brain regions is simulated for 50 subjects at 200 time points ($T = 200$) for networks with a different number of nodes ($N = 5, 10, 15$). The true underlying network connectivities are reported in Smith et al. (2011), which we used to calculate the area under the ROC curve (AUC) for the links predicted for each subject. Ground-truth data includes whether there is a connection (edge) from node i to j (directional; "positive" is when there is a connection), which were used to calculate true/false positive rates by varying the discrimination threshold (see the supplement, §III.2, for details). As it is not possible to de-

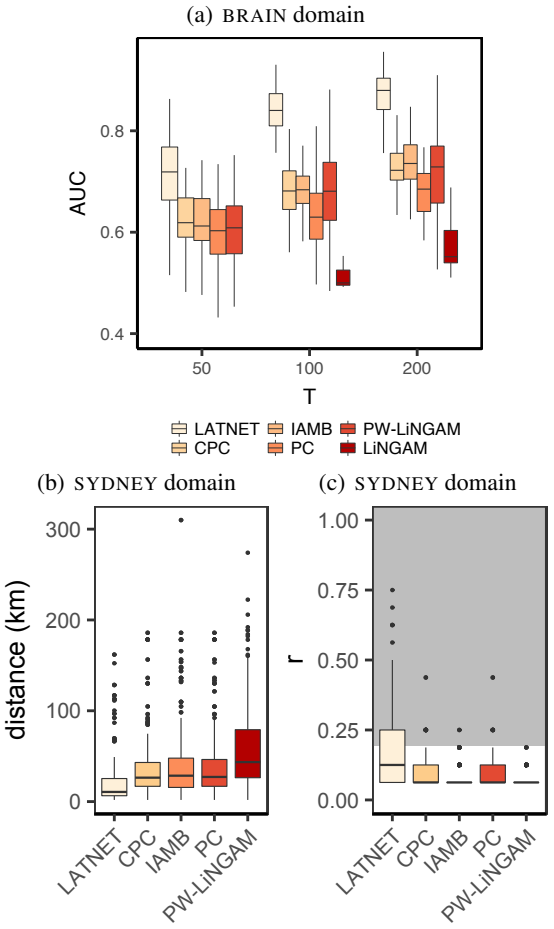

Figure 1. (a) AUC values (the higher the better) on BRAIN, computed using the networks in Smith et al. (2011) as the ground truth. (b) Results on SYDNEY using the air distance as a measure of spatial coherence (the lower the distance the better) between the connections discovered by each method. (c) Results on SYDNEY using the proportion of the networks in which a discovered arc is present (r) as a measure of temporal stability (the higher the better). The shaded area represents r statistically > 0 (risk α = 0.05). LiNGAM was unable to perform inference on BRAIN (a) for $T = 50$ and on SYDNEY (b and c).

fine comparable discriminative thresholds across different methods, using the AUC avoids the selection of a single threshold altogether. We note that the underlying network is a directed acyclic graph (DAG); however networks discovered by LATNET are not restricted to DAGs, and therefore baseline methods assuming the underlying network is a DAG have a favorable bias.

Figure 1(a) shows the AUC of each method (for $N = 15$) using box-plots (top and bottom edges of the box correspond to the first and third quartiles respectively). Results for $N = 5, 10$ are given in the supplement (§III.2). In the case of LATNET, we used the *posterior uncertainties* around the predicted connections and their strengths to determine the discriminative thresholds for the AUC calculations. We see that, although other methods are favorably biased about the underlying structure, LATNET consistently outperforms all the baseline methods.

5.2. SYDNEY Domain

Here we aim to discover the relationship between property prices in different suburbs of Sydney. The data include quarterly median sale prices for 51 suburbs in Sydney and surrounding area from 1995 to 2014. Since the underlying network is unknown, we cannot compute the AUC and instead use two other performance measures concerned with *spatial coherence* and *temporal stability*. For this purpose, we compute the air distance between the suburbs that are discovered to be connected (the shorter the better) and the proportion (r) of networks in which a connection was present (for each discovered connection) when our method is applied to different time windows (the higher the better). We set the discrimination threshold for each method so that on average each method finds 17-19 edges in the network.

Figure 1(b) shows the air distance between the connected nodes discovered by each method. The average distance by LATNET is 21km, which is almost half of the average distance by the other methods (CPC:38km, PC:40km, IAMB:43km, PW-LINGAM:60km; p-values < 0.001 for all t-tests between arc distances in LATNET and each of the competitors). Therefore, the networks discovered by LATNET are more spatially coherent than the baselines'. Similarly, Figure 1(c) shows the r values for the different methods. Less than 8%, 5%, 1% of PC, CPC and IAMB arcs were significant (risk $\alpha = 0.05$), respectively, while more than 29% of LATNET arcs are significant. Interestingly, PW-LINGAM did not find any significant arcs. We then conclude that temporal stability of the connections discovered by LATNET is significantly better than all the baseline methods. Additional details and results can be found in the supplement (§III.5).

5.3. YEAST Domain

We use LATNET to infer local and global genome regulation patterns for one extensively studied species, *Saccharomyces cerevisiae* (Spellman et al., 1998). This represents 100,000+ data points and a network with up to 38,000,000+ arcs. The true underlying network is unknown but there is extensive literature about its major features. Here we take as references the cell cycle transcriptionally regulated genes (Rowicka et al., 2007) and *http://www.yeastgenome.org* as a more general resource. For space reasons, we summarize experiments here, and leave to the supplement their exhaustive treatment.

Analysis of the sentinels of the yeast cell cycle (YCC), which represents $\approx$ tenth of the network (Spellman et al., 1998). We extracted key genes and connections discovered by LATNET based on a subset of *strong* arcs. Figure 2 (left)

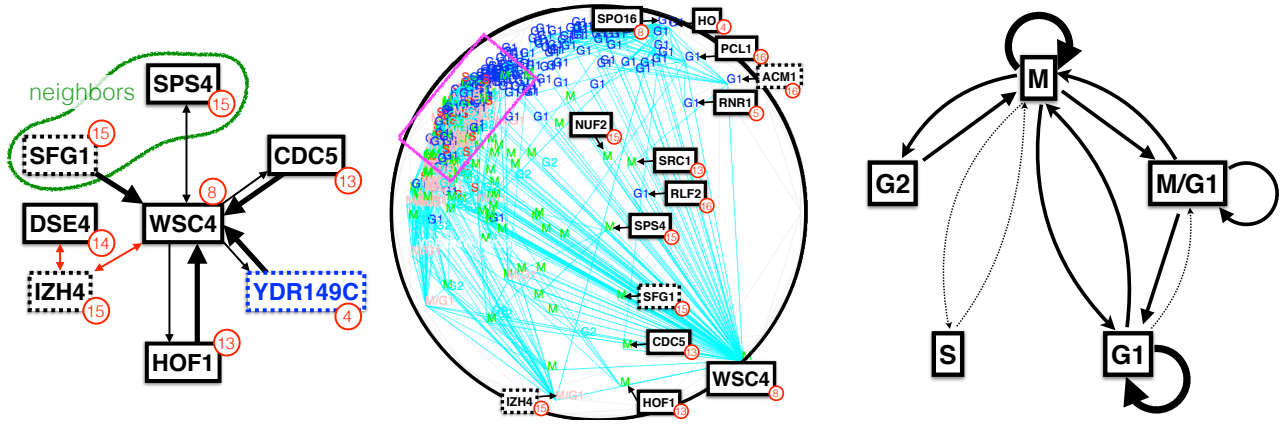

Figure 2. Results on YEAST using LATNET (best viewed in color). *Left*: subgraph G_w containing *all* strong arcs; plain rectangles (vs dashed): reported (vs unreported) cell cycle transcriptionally regulated genes (Rowicka et al., 2007); thick arcs = topmost strong arcs; black arcs (resp red arcs): $\mu > 0$ (resp. $\mu < 0$); red disk: chromosome number; in blue: gene with no known biological process/function/compartment; *Center*: manifold learned from strong arcs, displayed in Klein disk (conformal). Strong arcs in blue segments; only most important gene names shown; pink rectangle: area with comparatively few strong arcs; *Right*: network aggregating strong arcs discovered for the YCC, between cell cycle phases.

summarizes the results. We observe that (i) in terms of nodes, the key discovered genes are known to be involved in the cell structure dynamics. Strikingly, although SFG1 is not among the reported key genes, it happened to be *neighbors* of SPS4 (a key gene) on the same chromosome, and therefore it is likely to be part of the network and can be targeted for future investigations. (ii) In terms of the connections, the topmost strong arcs belong to a small connected component (G_w) that are asymmetrically organized around gene WSC4, which is consistent with the fact that the underlying network should be directed.

We then aimed to obtain a broader network map without filtering arcs. Figure 2 (center) shows the manifold coordinates induced by the network's graph, built upon Meila & Shi (2001). With such a technique, clusters of genes that are "significantly far" from each other should represent different key network structure components. In our case, it is evident that there is a "crowd versus the rest of the crowd" structure, and this rest of the crowd gathers almost only heavily regulated genes, that is, genes that are known to be heavily connected in the true network. It is therefore apparent that LATNET has succeeded in recovering a prominent network structure around such genes. Consistent with the literature, a small number of key genes drive the coordinates. Last, Figure 2 (right) summarizes the broad picture of strong arcs between YCC phases: it should come at no surprise that cell splitting, (M)itosis, has the largest number of these arcs.

Analysis of the full genome (results in supplement, III.4). A successful technique should recover three essential features of the complete network, from local to global: (i) the fact that it is locally highly asymmetric by nature, with a small number of direct feedbacks relatively to the genome size; (ii) the fact that key sub-networks like YCC should still be in the top rank of the global network and (iii) known connected sub-networks should still appear with as little noise as possible brought by the overall network. LATNET clearly succeeds at (i), with more than twice strong arcs going outside the YCC compared to arcs coming in the YCC from non-YCC genes. LATNET is also good at (ii), and we see that YCC genes tend to be outnumbered by genes that are perhaps more "all-purpose" but still supposed to be involved in heavy regulation mechanisms, which makes sense. The most prominent result is perhaps on (iii): the predominance of gap phase G1 compared to G2 that we still observe with all genes hints on the yeast *species* from which our data comes from: this is a indeed a known feature of *Saccharomyces cerevisiae*, versus other species like *S. pombe* for example. None of these patterns, in particular (ii) and (iii), were discovered using the baseline methods.

6. Conclusion & Discussion

We have proposed a Bayesian framework for network structure discovery for continuous-valued observations; developed an efficient inference algorithm for it; and shown its benefits on real applications. Our theoretical analysis shows that the traditional constraints for stability (Hyvärinen & Smith, 2013; Linderman & Adams, 2014; Shimizu et al., 2006) in networks are alleviated. What we get with such assumptions is a non-negligible uplift in the easiness of the expected log-likelihood part, the bottleneck of the ELBO, and a robustness of the output's variance with respect to the network parameters. These results also hold for a class of posteriors broader than the ones we use, opening interesting avenues of applications for Concrete distributions.

Acknowledgements

AD was supported by a Research Fellowship from UNSW Sydney. We thank Louis Tiao for feedback on the manuscript.

References

Bollen, K.-A. *Structural equations with latent variables*. John Wiley & Sons, 1989.

Bonilla, E.-V., Chai, K.-M. A., and Williams, C.-K.-I. Multi-task Gaussian process prediction. In *NIPS*, 2008.

Hoyer, P.-O., Janzing, D., Mooij, J.-M., Peters, J., and Schölkopf, B. Nonlinear causal discovery with additive noise models. In *NIPS*, 2009.

Huang, B., Zhang, K., and Schölkopf, B. Identification of time-dependent causal model: A Gaussian process treatment. In *IJCAI*, pp. 3561–3568, 2015.

Hyttinen, A., Eberhardt, F., and Hoyer, P. O. Learning linear cyclic causal models with latent variables. *JMLR*, 13 (Nov):3387–3439, 2012.

Hyvärinen, A. and Smith, S.-M. Pairwise likelihood ratios for estimation of non-Gaussian structural equation models. *JMLR*, 14:111–152, 2013.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. *arXiv:1611.01144*, 2016.

Kingma, D. and Welling, M. Auto-Encoding Variational Bayes. In *ICLR*, 2014.

Koyejo, O. and Ghosh, J. Constrained bayesian inference for low rank multitask learning. In *UAI*, 2013.

Lawrence, N. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. *JMLR*, 6:1783–1816, 2005.

Linderman, S.-W. and Adams, R.-P. Discovering Latent Network Structure in Point Process Data. In *ICML*, 2014.

Maddison, C.-J., Mnih, A., and Teh, Y.-W. The Concrete Distribution: A Continuous Relaxation of Discrete Random Variables. *arXiv:1611.00712*, 2016.

Meila, M. and Shi, J. Learning segmentation by random walks. In *NIPS*, volume 14, 2001.

Mohamed, S., Heller, K. A., and Ghahramani, Z. Bayesian and l1 approaches for sparse unsupervised learning. In *ICML*, 2012.

Mooij, J. M., Janzing, D., Heskes, T., and Schölkopf, B. On causal discovery with cyclic additive noise models. In *NIPS*, pp. 639–647, 2011.

Pearl, J. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, New York, NY, USA, 2000. ISBN 0-521-77362-8.

Peters, J., Mooij, J.-M., Janzing, D., and Schölkopf, B. Causal discovery with continuous additive noise models. *JMLR*, 15(1):2009–2053, 2014.

Rakitsch, B., Lippert, C., Borgwardt, K., and Stegle, O. It is all in the noise: Efficient multi-task Gaussian process inference with structured residuals. In *NIPS*, 2013.

Ramse, J., Zhang, J., and Spirtes, P. Adjacency-faithfulness and conservative causal inference. In *UAI*, 2006.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian processes for machine learning*. The MIT Press, 2006.

Rezende, D.-J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, pp. 1278–1286, 2014.

Richardson, T. *Feedback models: Interpretation and discovery*. PhD thesis, Ph. D. thesis, Carnegie Mellon, 1996.

Rowicka, M., Kudlicki, A., Tu, B.-P., and Otwinowski, Z. High-resolution timing of cell cycle-regulated gene expression. *PNAS*, 104(43):16892–16897, 2007.

Shimizu, S., Hoyer, P.-O., Hyvärinen, A., and Kerminen, A. A linear non-Gaussian acyclic model for causal discovery. *JMLR*, 7:2003–2030, 2006.

Shimizu, S., Inazumi, T., Sogawa, Y., Hyvärinen, A., Kawahara, Y., Washio, T., Hoyer, P.-O., and Bollen, K. DirectLiNGAM: A direct method for learning a linear non-gaussian structural equation model. *JMLR*, 12:1225–1248, 2011.

Smith, S.-M., Miller, K.-L., Salimi-Khorshidi, G., Webster, M., Beckmann, C.-F., Nichols, T.-E., Ramsey, J.-D., and Woolrich, M.-W. Network modelling methods for FMRI. *NeuroImage*, 54(2):875–891, 2011.

Spellman, P.-T., Sherlock, G., Zhang, M.-Q., Iyer, V.-R., Anders, K., Eisen, M.-B., Brown, P.-O., Botstein, D., and Futcher, B. Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Molecular Biology of the Cell*, 9:3273–3297, 1998.

Spirtes, P., Glymour, C.-N., and Scheines, R. *Causation, prediction, and search*. MIT press, 2000.

Tsamardinos, I., Aliferis, C.-F., Statnikov, A. R., and Statnikov, E. Algorithms for Large Scale Markov Blanket Discovery. In *FLAIRS*, volume 2, 2003.

Wilson, A.-G. and Ghahramani, Z. Generalised wishart processes. In *UAI*, 2010.

Zhang, K., Schölkopf, B., and Janzing, D. Invariant Gaussian Process Latent Variable Models and Application in Causal Discovery. In *UAI*, 2010.