VARIATIONAL MCMC

Nando de Freitas [†] Pedro Højen-Sørensen [‡] Michael Jordan [†] Stuart Russell [†]

March 7, 2001

[†] UC Berkeley, Computer Science Division
387 Soda Hall, Berkeley
CA 94720-1776 USA
{jfgf,jordan,russell}@cs.berkeley.edu

[‡] Department of Mathematical Modelling
Technical University of Denmark
DK-2800 Kongens Lyngby, Denmark
phs@imm.dtu.dk

Abstract

We propose a new class of learning algorithms that combines variational approximation and Markov chain Monte Carlo (MCMC) simulation. One of these algorithms is a mixture of two MCMC kernels: a random walk Metropolis kernel and a block Metropolis-Hastings (MH) kernel with a variational approximation as proposal distribution. The MH kernel allows one to locate regions of high probability efficiently. The Metropolis kernel allows us to explore the vicinity of these regions. This algorithm outperforms variational approximations because it yields slightly better estimates of the mean and considerably better estimates of higher moments, such as covariances. It also outperforms standard MCMC algorithms because it locates the regions of high probability quickly, thus speeding up convergence. We demonstrate this algorithm on the problem of Bayesian parameter estimation for logistic (sigmoid) belief networks.

1 Introduction

MCMC simulation is a powerful and accurate strategy for inference and learning (Gilks, Richardson and Spiegelhalter 1996, Robert and Casella 1999). However, it often requires the design of complex proposal distributions when applied to new tasks. Otherwise, the algorithms can take very long to converge (*i.e.*, mix poorly). On the other hand, variational methods have been shown to provide fast approximate estimates in many scenarios (Jaakkola and Jordan 1999, Jordan, Ghahramani, Jaakkola and Saul 1999). Yet, they rely on simplifications of the original problem in order to ensure mathematical tractability. This often results in algorithms that yield poor estimates of high order moments, such a covariances and kurtosis.

In this paper, we introduce a class of Markov chain Monte Carlo (MCMC) algorithms that exploits the fact that variational approximations can be used as proposal distributions. We show that naive algorithms exploiting this property can mix poorly, but solve this problem by introducing more sophisticated MCMC kernels based on block sampling and mixtures of MCMC kernels. In particular, we use mixtures with variational kernels that allow the algorithm to detect the regions of high probability quickly and metropolis kernels that enable it to explore the neighbourhood of these regions. The resulting algorithm converges quickly to the regions of high probability and also yields reasonable approximations to the entire distribution of interest. Our approach makes it possible to combine variational and MCMC algorithms within a rigorous probabilistic setting so as to exploit the benefits of both approaches simultaneously.

There have been other attempts at combining specific approximation techniques and simulation methods; indeed, researchers in the statistics community often combine the Laplace approximation with simulation methods (Gilks et al. 1996). However, the Laplace method is based on truncated Taylor expansions of derivative terms that can often lead to poor approximations. Recently, Ghahramani and Beal (2000) showed that using a variational approximation for mixtures of factor analyzers as the proposal for an importance sampler could lead to an improvement in the accuracy of the results. The approach we take here is far more general and surmounts many of the problems encountered in the importance sampling approach.

We demonstrate the approach on the task of Bayesian parameter estimation of logistic (sigmoidal) belief networks with latent variables. This problem is of interest for several reasons. First, it exhibits nonlinearity an non-Gaussianity. Second, it includes the problems of logistic regression and classification with missing observations as a sub-case. That is, our approach can handle situations in which we have many partially observed input signals.

Third, the noise is very uninformative and consequently one has to be very careful when applying model testing techniques such as cross-validation. This motivates the Bayesian paradigm and, in particular, the introduction of a Gaussian prior as a regularisation mechanism. Lastly, this type of network has important connections with research carried out in the area of neural computation.

The remainder of this paper is organised a follows. The probabilistic models and estimation goals are outlined in Section 2. In Section 3, we present the variational approximations to the original models and the expectation maximisation (EM) algorithm to perform the necessary computations. The presentation of variational techniques for parameter estimation begins at a very general level. Subsequently, it focuses on the cases of fully observed Bayesian networks (BNs) and BNs with hidden nodes. A novel strategy that combines MCMC and variational methods is proposed in Section 4. The experimental results obtained with this method for logistic BNs are presented in Section 5. Conclusions and recommendations for future work are drawn in Section 6. Finally, the notation appears in the appendix.

2 Model Specification

In this section, we present our probabilistic model for parameter estimation in belief networks (BNs). These networks provide a convenient pictorial representation of probability distributions that can be factorised as follows[1]

$$p(\mathbf{x}_{1:n_x}|\boldsymbol{\theta}) = \prod_{i=1}^{n_x} p(\mathbf{x}_i|\mathbf{x}_{\pi(i)}, \boldsymbol{\theta}_i)$$

where $\mathbf{x}_{1:n_x} \triangleq \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{n_x}\}$ represents a stacked set of nodes, $\mathbf{x}_i$ denotes the variable associated with node i, $\mathbf{x}_{\pi(i)}$ denotes the parent nodes of node i and $\boldsymbol{\theta}_i$ are some unknown parameters associated with node i. Figure 1 shows a simple BN where all the nodes are observed (A) and a BN where the value of one of the nodes is unknown (B). In both cases, we will show that it is possible to design algorithms to estimate the parameters.

More formally, we consider a countable set of random variables $\mathbf{x}_i \in \mathcal{X}$, and partition the set into a visible part, $\mathbf{x}_i^v \in \mathcal{X}^v$, and a hidden part, $\mathbf{x}_i^h \in \mathcal{X}^h$, such that $\mathcal{X} = \{\mathcal{X}^v \cup \mathcal{X}^h\}$.

[1] For simplicity, we use $\mathbf{x}_t$ to denote both the random variable and its realisation. Consequently, we express continuous probability distributions using $p(d\mathbf{x}_t)$ instead of $\Pr(\mathbf{X}_t \in d\mathbf{x}_t)$ and discrete distributions using $p(\mathbf{x}_t)$ instead of $\Pr(\mathbf{X}_t = \mathbf{x}_t)$. If these distributions admit densities with respect to an underlying measure μ (usually counting or Lebesgue), we denote these densities by $p(\mathbf{x}_t)$. For example, when considering the space $\mathbb{R}^n$, we will use the Lebesgue measure, $\mu = d\mathbf{x}_t$, so that $p(d\mathbf{x}_t) = p(\mathbf{x}_t)\, d\mathbf{x}_t$. To make the material accesible to a wider audience, we shall allow for a slight abuse of terminology by, sometimes, referring to $p(\mathbf{x}_t)$ as a distribution.

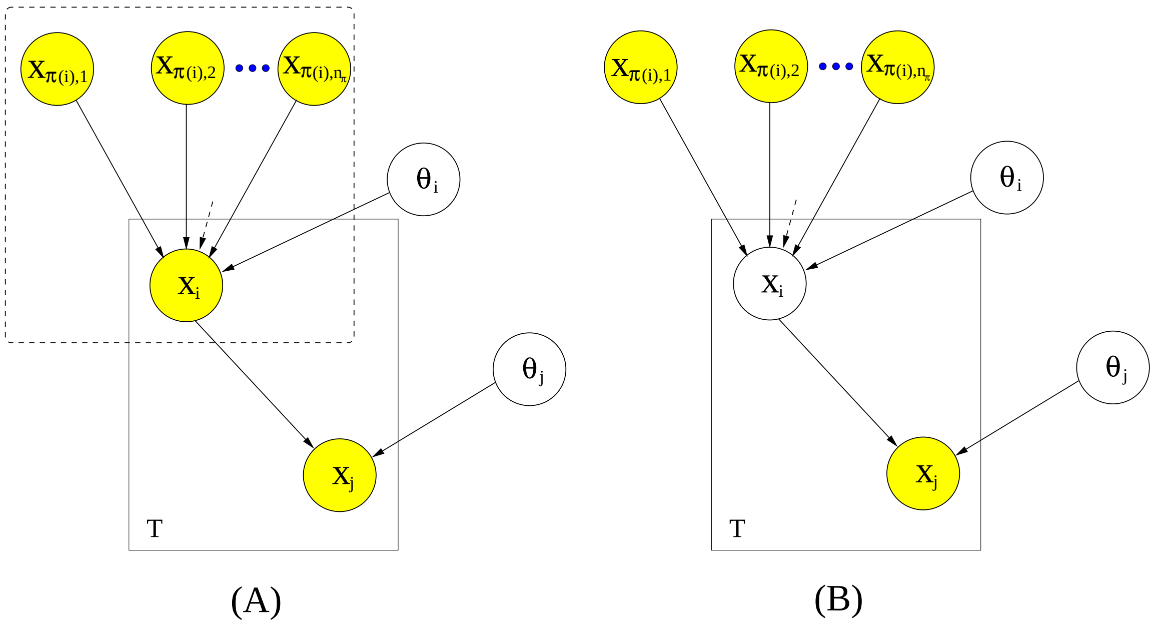

Figure 1: (A) Fully observed belief network. (B) Belief network with one hidden node (right). The parameters $\boldsymbol{\theta}$ are treated as hidden units in the Bayesian framework. The dashed box represents the Markov blanket for node $\boldsymbol{\theta}_i$, while the continuous box is a template indicating that there are T copies of $\mathbf{x}$.

We shall assume that we have T sets of measurements for the observed variables; that is $\mathbf{x}^v \triangleq \mathbf{x}^v_{1:n_{xv},1:T} \in (\mathcal{X}^v)^{n_{xv}\times T}$. The distribution of the random variable $\mathbf{x}_i$ is parameterised by $\boldsymbol{\theta}_i \in \mathbb{R}^{n_{\pi(i)}}$, where $n_{\pi(i)}$ is the number of variables on which $\mathbf{x}_i$ depends; that is the number of parent nodes in the case of a belief network. In general, the cardinality of $\boldsymbol{\theta}$ is $n_\theta \triangleq \sum_i n_{\pi(i)}$. Even though parameters in the Bayesian setting are to be regarded as hidden variables, we will here make a notational distinction between the hidden states and the distributional parameters of the hidden states.

We shall restrict the parameterisation of the conditional probability distributions to the following Bernoulli family with a logistic mapping

$$p(\mathbf{x}_i|\mathbf{x}_{\pi(i)},\boldsymbol{\theta}_i) = \prod_{t=1}^{T} g\left(\varphi_{i,t}\right) = \prod_{t=1}^{T} \frac{1}{1+\exp\left(-\varphi_{i,t}\right)}$$

$$= \prod_{t=1}^{T} \left[\frac{1}{1+\exp\left(-\alpha-\boldsymbol{\theta}_i'\mathbf{x}_{\pi(i),t}\right)}\right]^{\frac{\mathbf{x}_{i,t}+1}{2}} \left[\frac{1}{1+\exp\left(\alpha+\boldsymbol{\theta}_i'\mathbf{x}_{\pi(i),t}\right)}\right]^{-\frac{\mathbf{x}_{i,t}-1}{2}}$$

where $\varphi_{i,t} \triangleq \mathbf{x}_{i,t}(\alpha+\boldsymbol{\theta}_i'\mathbf{x}_{\pi(i),t})$, $\mathbf{x}_i \in \{-1,1\}^T$ and α is assumed to be fixed. (Note that we only make the latter assumption for presentation purposes. One could always introduce an extra node fixed to 1 and treat α as an extra parameter.) To complete the specification of the Bayesian model, we assume a Gaussian prior $\mathcal{N}(\boldsymbol{\mu}_0,\Sigma_0)$ on the parameters $\boldsymbol{\theta}_i$ and prior independence, that is $p(d\boldsymbol{\theta}) = \prod_i p(d\boldsymbol{\theta}_i)$.

The goal of the analysis will be to compute the posterior distribution $p(d\boldsymbol{\theta}|\mathbf{x}^v)$. From this distribution, one can easily derive other quantities of interest, such as predictive distributions and marginal distributions. As illustrated in Figure 1, we need to distinguish between two scenarios: fully observed networks and networks with hidden nodes.

(i) Fully observed BNs: As shown in the left plot of Figure 1, the Markov blanket of $\boldsymbol{\theta}_i$ (the nodes inside the dashed box) does not include any other parameters. As a result of this, the problem of parameter estimation for BNs simplifies to several logistic regression sub-problems; one for each node with parents. The posterior for each of these nodes can be computed using Bayes rule

$$p(d\boldsymbol{\theta}|\mathbf{x}) = \frac{\prod_{i=1}^{n_{xc}} \prod_{t=1}^{T} p(\mathbf{x}_{i,t}|\mathbf{x}_{\pi(i),t}, \boldsymbol{\theta}_i) p(d\boldsymbol{\theta}_i)}{\int_{\boldsymbol{\Theta}} \prod_{i=1}^{n_{xc}} \prod_{t=1}^{T} p(\mathbf{x}_{i,t}|\mathbf{x}_{\pi(i),t}, \boldsymbol{\theta}_i) p(d\boldsymbol{\theta}_i)}$$

where n_{xc} denotes the number of nodes that have at least one parent.

(ii) BNs with hidden nodes: Hidden nodes introduce dependencies between the parameters of the model. For example, in the right plot of Figure 1, the parameters $\boldsymbol{\theta}_j$ depend on the parameters $\boldsymbol{\theta}_i$ because $\mathbf{x}_i$ is unknown. To compute the posterior, we need to marginalise over the hidden variables

$$p(d\boldsymbol{\theta}|\mathbf{x}^v) = \frac{\sum_{\mathbf{x}^h} \prod_{i=1}^{n_{xc}} \prod_{t=1}^{T} p(\mathbf{x}_{i,t}|\mathbf{x}_{\pi(i),t}, \boldsymbol{\theta}_i) p(d\boldsymbol{\theta}_i)}{\int_{\boldsymbol{\Theta}} \sum_{\mathbf{x}^h} \prod_{i=1}^{n_{xc}} \prod_{t=1}^{T} p(\mathbf{x}_{i,t}|\mathbf{x}_{\pi(i),t}, \boldsymbol{\theta}_i) p(d\boldsymbol{\theta}_i)}$$

The posterior distributions, in both cases, cannot be calculated analytically because of the large integrals and sums appearing in the denominators. To circumvent this problem, in the next section we introduce variational methods to obtain approximate solutions. These methods will require that we map the original model to a simplified model that is more amenable to analytical and computational treatment. We shall correct for this change of model using Markov chain Monte Carlo simulation in Section 4.

3 Variational Approximation

We begin this section by presenting a general variational framework for parameter estimation. We then enforce the belief network topological constraints and, finally, derive approximations for parameter estimation in logistic belief networks. The resulting approximations are similar to the ones of (Jaakkola and Jordan 2000), with the exception that we introduce an extra parameter, α, to treat multimodality.

3.1 Variational methods for parameter estimation

The aim of variational methods is to convert a complex problem into a simpler, more tractable problem: see for example (Jordan et al. 1999). The simpler problem is generally characterised by a decoupling of the degrees of freedom in the original problem. This decoupling is achieved by introducing an extra set of parameters, the so-called variational parameters. The variational parameters are then optimised so that the solution to the simpler problem resembles the solution to the complex problem.

Bounds and convexity play an important role in the variational paradigm. In many situations, including our BNs, the likelihood of the data $p(\mathbf{x}^v|\boldsymbol{\theta})$ cannot be easily evaluated. However, if we know a lower bound on the likelihood, we can maximise this bound to obtain an approximate solution. Lower bounds on the likelihood can be easily obtained using Jensen's inequality

$$\log p(\mathbf{x}^v|\boldsymbol{\theta}) = \log \mathbb{E}_{q(\mathbf{x}^h)}\left[\frac{p(\mathbf{x}|\boldsymbol{\theta})}{q(\mathbf{x}^h)}\right] \geq \mathbb{E}_{q(\mathbf{x}^h)}\left[\log p(\mathbf{x}|\boldsymbol{\theta})\right] - \mathbb{E}_{q(\mathbf{x}^h)}\left[\log q(\mathbf{x}^h)\right] \tag{1}$$

where $q(\mathbf{x}^h)$ is an arbitrary density over the hidden states with respect to the Lebesgue or counting measure. The right hand side is the negative Kullback Leibler "distance" between q and p (that is, $-KL(q\|p)$) while the the last term is known as the entropy, $\mathcal{H}(q(\mathbf{x}^h)) \triangleq -\mathbb{E}_{q(\mathbf{x}^h)}\left[\log q(\mathbf{x}^h)\right]$, of the distribution q. It is clear, therefore, that maximising the lower bound is equivalent to minimising the Kullback Leibler "distance".

The distribution q that yields the tightest bound can be found by free-form maximisation, but this typically leads to bounds that cannot be evaluated (Chandler 1987). An alternative approach is to choose a parametric form, $\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})$, of $q(\mathbf{x}^h)$ that makes the right hand side of equation (1) easy to evaluate. The variational parameters $\boldsymbol{\lambda}$ can then be optimised to get a bound that is as tight as possible. This approach is similar to what is done in statistical mechanics where one uses a tractable energy function and the Gibbs-Bogoliubov-Feynman inequality to calculate the partition function (the normalising density in Bayes' rule) of a system with an intractable energy function (Zhang 1993).

It may be impossible, in general, to choose a specific functional form of $\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})$ that makes the evaluation of $\mathbb{E}_{\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})}\left[\log p(\mathbf{x}|\boldsymbol{\theta})\right]$ tractable. However, additional flexibility can be introduced by lower bounding $p(\mathbf{x}|\boldsymbol{\theta})$ with a well-chosen function $\widehat{p}(\mathbf{x}|\boldsymbol{\theta},\boldsymbol{\xi})$, where $\boldsymbol{\xi}$ denotes an additional set of variational parameters. To summarise, the variational approach involves the following two steps

1. Introduce the variational parameters $\boldsymbol{\xi}$ to make the conditional joint distribution of the hidden and visible variables, $p(\mathbf{x}|\boldsymbol{\theta})$, tractable.

2. Introduce the variational distribution q with parameters $\boldsymbol{\lambda}$ to make the conditional marginal distribution of the visible variables, $p(\mathbf{x}^v|\boldsymbol{\theta})$, tractable.

Following these steps, we can obtain an unnormalised lower bound on the likelihood

$$\widehat{p}(\mathbf{x}^v|\boldsymbol{\theta},\boldsymbol{\lambda},\boldsymbol{\xi}) \propto \exp\left\{\mathbb{E}_{\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})}\left[\log\widehat{p}(\mathbf{x}|\boldsymbol{\theta},\boldsymbol{\xi})\right] - \mathbb{E}_{\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})}\left[\log\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})\right]\right\} \tag{2}$$

and, using Bayes' rule, we can easily obtain a lower bound on the posterior distribution

$$\widehat{p}(d\boldsymbol{\theta}|\mathbf{x}^v,\boldsymbol{\lambda},\boldsymbol{\xi}) \propto p(d\boldsymbol{\theta})\exp\left\{\mathbb{E}_{\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})}\left[\log\widehat{p}(\mathbf{x}|\boldsymbol{\theta},\boldsymbol{\xi})\right]\right\} \tag{3}$$

Finally, we can obtain a lower bound, $\widehat{p}(\mathbf{x}^v|\boldsymbol{\lambda},\boldsymbol{\xi})$, on the evidence, $p(\mathbf{x}^v)$, by standard marginalisation

$$p(\mathbf{x}^v) = \mathbb{E}_{p(d\boldsymbol{\theta})}\left[p(\mathbf{x}^v|\boldsymbol{\theta})\right] \geq \mathbb{E}_{p(d\boldsymbol{\theta})}\left[\widehat{p}(\mathbf{x}^v|\boldsymbol{\theta},\boldsymbol{\lambda},\boldsymbol{\xi})\right] = \widehat{p}(\mathbf{x}^v|\boldsymbol{\lambda},\boldsymbol{\xi}) \tag{4}$$

Implicitly, we are replacing the integrand in the normalising expression of the posterior distribution with a tractable lower bound (that is, one that can be integrated easily). We, then, maximise the resulting lower bound on the integral to approximate the true integral. In other words, we have replaced the integration problem by an easier optimisation problem.

An alternative approach to obtain a lower bound on the likelihood was proposed in (Jaakkola and Jordan 2000). The method is also based on convexity and Jensen's inequality. In particular, it is based on the fact that the geometric average, $\prod_i p_i^{q_i}$, where q_i is a probability distribution, is less than or equal to the arithmetic average, $\sum_i q_i p_i$. Following this result, the likelihood can be lower bounded as follows

$$\begin{aligned} p(\mathbf{x}^v|\boldsymbol{\theta}) &= \mathbb{E}_{\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})}\left[\frac{p(\mathbf{x}|\boldsymbol{\theta})}{\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})}\right] \geq \mathbb{E}_{\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})}\left[\frac{\widehat{p}(\mathbf{x}|\boldsymbol{\theta},\boldsymbol{\xi})}{\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})}\right] \\ &\geq \prod_{\mathcal{X}^h}\left(\frac{\widehat{p}(\mathbf{x}|\boldsymbol{\theta},\boldsymbol{\xi})}{\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})}\right)^{\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})} = C(q)\prod_{\mathcal{X}^h}\left(\widehat{p}(\mathbf{x}|\boldsymbol{\theta},\boldsymbol{\xi})\right)^{\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})} \end{aligned}$$

where $\log C(q)$ is the entropy of the random variable $\mathbf{x}^h$ under the distribution $\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})$. The lower bound on the likelihood can be written as follows

$$\widehat{p}(\mathbf{x}^v|\boldsymbol{\theta},\boldsymbol{\lambda},\boldsymbol{\xi}) \propto \prod_{\mathcal{X}^h}\left(\widehat{p}(\mathbf{x}|\boldsymbol{\theta},\boldsymbol{\xi})\right)^{\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda})}$$

That is, the dependencies between the variables $\mathbf{x}$ that would have resulted from performing exact marginalisation have been replaced with dependencies through a shared variational distribution. We shall however use the bound given by equation (2) as it is more general and tractable.

To compute the parameters $\boldsymbol{\theta}$, $\boldsymbol{\lambda}$ and $\boldsymbol{\xi}$, we maximise the lower bound on the evidence, $\widehat{p}(\mathbf{x}^v|\boldsymbol{\lambda},\boldsymbol{\xi})$. This step can be carried out using the coordinate ascent maximum likelihood

1. *Expectation step:* Compute the expectation of the complete log-likelihood using the old values of the variational parameters

$$\mathcal{Q} \triangleq \mathbb{E}_{\widehat{p}(d\boldsymbol{\theta}|\mathbf{x}^v,\boldsymbol{\lambda}^{\text{old}},\boldsymbol{\xi}^{\text{old}})}\left[\log \widehat{p}(\mathbf{x}^v,\boldsymbol{\theta}|\boldsymbol{\lambda},\boldsymbol{\xi})\right]$$

2. *Maximisation step:* Maximise with respect to the variational parameters

$$(\boldsymbol{\lambda}^{\text{new}},\boldsymbol{\xi}^{\text{new}}) = \arg\max_{\boldsymbol{\lambda},\boldsymbol{\xi}} \mathcal{Q}$$

3. Go to 1 until a maximum number of iterations or required error tolerance are reached.

Figure 2: EM algorithm for variational approximation.

algorithm shown in Figure 2 (Dempster, Laird and Rubin 1977). This algorithm is guaranteed to maximise the lower bound on the evidence $\widehat{p}(\mathbf{x}^v|\boldsymbol{\lambda},\boldsymbol{\xi})$, but it is not guaranteed to maximise the actual evidence $p(\mathbf{x}^v)$. That is, monitoring convergence on $\widehat{p}(\mathbf{x}^v|\boldsymbol{\lambda},\boldsymbol{\xi})$ can be misleading. However, if the bounds on the likelihood of the observed and complete data are chosen carefully, some existing empirical results suggest that this framework can perform very well in complex scenarios (de Freitas, Niranjan and Gee 2000, Jaakkola and Jordan 1999, Jaakkola and Jordan 2000). For the BNs introduced in the previous section, the expectation of the complete log-likelihood is defined as

$$\begin{aligned}\mathcal{Q} &\triangleq \mathbb{E}_{\widehat{p}(d\boldsymbol{\theta}|\mathbf{x}^v,\mathbf{x}^v_\pi,\boldsymbol{\lambda}^{\text{old}},\boldsymbol{\xi}^{\text{old}})}\left[\log\left(\prod_{i=1}^{n_{xc}} \widehat{p}(\mathbf{x}^v_i|\mathbf{x}^v_{\pi(i)},\boldsymbol{\theta}_i,\boldsymbol{\lambda}_i,\boldsymbol{\xi}_i)p(d\boldsymbol{\theta}_i)\right)\right]\\ &\propto \mathbb{E}_{\widehat{p}(d\boldsymbol{\theta}|\mathbf{x}^v,\mathbf{x}^v_\pi,\boldsymbol{\lambda}^{\text{old}},\boldsymbol{\xi}^{\text{old}})}\left[\mathcal{H}(\widehat{q}) + \log\left(\prod_{i=1}^{n_{xc}} \exp\left\{\mathbb{E}_{\widehat{q}(\mathbf{x}^h_i|\boldsymbol{\lambda}_i)}\left[\log \widehat{p}(\mathbf{x}_i|\mathbf{x}_{\pi(i)},\boldsymbol{\theta}_i,\boldsymbol{\xi}_i)\right]\right\} p(d\boldsymbol{\theta}_i)\right)\right]\end{aligned} \tag{5}$$

where $\mathbf{x}^h_i$ denotes the hidden nodes in $\{\mathbf{x}_i,\mathbf{x}_{\pi(i)}\}$ and $\mathcal{H}(\widehat{q}) \triangleq \sum_{i=1}^{n_{xc}} \mathcal{H}(\widehat{q}(\mathbf{x}^h_i|\boldsymbol{\lambda}_i))$. In the following two sections, we show how to compute this quantity in the case of logistic belief networks.

3.2 Variational approximation for fully observed logistic BNs

When analysing logistic BNs, we can can lower bound the likelihood of the data using a Gaussian approximation (Jaakkola and Jordan 2000), as follows

$$p(\mathbf{x}_i|\mathbf{x}_{\pi(i)},\boldsymbol{\theta}_i) = g\left(\varphi_i\right) \geq g(\boldsymbol{\xi}_i)\exp\left(\frac{\varphi_i - \boldsymbol{\xi}_i}{2} - \phi(\boldsymbol{\xi}_i)\left(\varphi_i^2 - \boldsymbol{\xi}_i^2\right)\right) \tag{6}$$

where $\varphi_i = \mathbf{x}_i(\alpha + \boldsymbol{\theta}_i'\mathbf{x}_{\pi(i)})$ and $\phi(\boldsymbol{\xi}_i) \triangleq \frac{\tanh(\boldsymbol{\xi}_i/2)}{4\boldsymbol{\xi}_i}$. It is then trivial to apply Bayes' rule to compute a lower bound on the posterior distribution of the parameters

$$\widehat{p}(d\boldsymbol{\theta}_i|\mathbf{x}_i,\mathbf{x}_{\pi(i)},\boldsymbol{\xi}_i) \propto \widehat{p}(\mathbf{x}_i|\mathbf{x}_{\pi(i)},\boldsymbol{\theta}_i,\boldsymbol{\xi}_i)p(d\boldsymbol{\theta}_i)$$

where $\widehat{p}(\mathbf{x}_i|\mathbf{x}_{\pi(i)},\boldsymbol{\theta}_i,\boldsymbol{\xi}_i)$ corresponds to the right hand side of equation (6). Using conjugate analysis and completing squares, we can obtain the following recursive expressions for the mean, $\boldsymbol{\mu}$, and variance, $\boldsymbol{\Sigma}$, of the Gaussian posterior distribution

$$\begin{aligned}
\boldsymbol{\Sigma}_{i,t}^{-1} &= \boldsymbol{\Sigma}_{i,t-1}^{-1} + 2\phi(\boldsymbol{\xi}_{i,t-1})\mathbf{x}_{\pi(i),t}\mathbf{x}_{\pi(i),t}' \\
\boldsymbol{\mu}_{i,t} &= \boldsymbol{\Sigma}_{i,t}\left[\left(\frac{\mathbf{x}_{i,t}}{2} - 2\phi(\boldsymbol{\xi}_{i,t-1})\alpha\right)\mathbf{x}_{\pi(i),t} + \boldsymbol{\Sigma}_{i,t-1}^{-1}\boldsymbol{\mu}_{i,t-1}\right]
\end{aligned}$$

As an instance of equation (5), we can compute the variational parameters by maximising the lower bound on the evidence

$$\boldsymbol{\xi}_i^{\text{new}} = \arg\max_{\boldsymbol{\xi}_i} \quad \mathbb{E}_{\widehat{p}(d\boldsymbol{\theta}_i|\mathbf{x}_i,\mathbf{x}_{\pi(i)},\boldsymbol{\xi}_i^{\text{old}})}\left[\log\widehat{p}(\mathbf{x}_i|\mathbf{x}_{\pi(i)},\boldsymbol{\xi}_i)p(d\boldsymbol{\theta}_i)\right]$$

Since all the distributions are Gaussian, one can take derivatives and equate to zero to obtain the following recursive formula for the variational parameters

$$\begin{aligned}
\boldsymbol{\xi}_{i,t}^2 &= \mathbb{E}_{\widehat{p}(d\boldsymbol{\theta}_i|\mathbf{x}_i,\mathbf{x}_{\pi(i)},\boldsymbol{\xi}_i^{\text{old}})}\left[(\alpha + \boldsymbol{\theta}_i'\mathbf{x}_{\pi(i),t})^2\right] \\
&= \alpha^2 + 2\alpha\boldsymbol{\mu}_{i,t}'\mathbf{x}_{\pi(i),t} + \mathbf{x}_{\pi(i),t}'\left(\boldsymbol{\Sigma}_{i,t} + \boldsymbol{\mu}_{i,t}\boldsymbol{\mu}_{i,t}'\right)\mathbf{x}_{\pi(i),t} \\
&= \alpha^2 + 2\alpha\boldsymbol{\mu}_{i,t}'\mathbf{x}_{\pi(i),t} + \operatorname{tr}\left(\left(\boldsymbol{\Sigma}_{i,t} + \boldsymbol{\mu}_{i,t}\boldsymbol{\mu}_{i,t}'\right)\mathbf{x}_{\pi(i),t}\mathbf{x}_{\pi(i),t}'\right)
\end{aligned}$$

The EM algorithm used for computing the variational approximation of fully observed logistic BNs is shown in Figure 3.

3.3 Variational approximations for logistic BNs with hidden nodes

To obtain the EM update equations for logistic networks with hidden nodes, we first calculate a lower bound on the posterior distribution

$$\begin{aligned}
\widehat{p}(d\boldsymbol{\theta}_i|\mathbf{x}_i^v,\mathbf{x}_{\pi(i)}^v,\boldsymbol{\lambda}_i,\boldsymbol{\xi}_i) &\propto \widehat{p}(\mathbf{x}_i^v|\mathbf{x}_{\pi(i)}^v,\boldsymbol{\theta}_i,\boldsymbol{\lambda}_i,\boldsymbol{\xi}_i)p(d\boldsymbol{\theta}_i) \\
&\propto \exp\left\{\mathbb{E}_{\widehat{q}(\mathbf{x}_i^h|\boldsymbol{\lambda}_i)}\left[\log\widehat{p}(\mathbf{x}_i|\mathbf{x}_{\pi(i)},\boldsymbol{\theta}_i,\boldsymbol{\xi}_i)\right]\right\}p(d\boldsymbol{\theta}_i) \\
&\propto \exp\left\{\mathbb{E}_{\widehat{q}(\mathbf{x}_i^h|\boldsymbol{\lambda}_i)}\left[\frac{\varphi_i - \boldsymbol{\xi}_i}{2} - \phi(\boldsymbol{\xi}_i)\left(\varphi_i^2 - \boldsymbol{\xi}_i^2\right)\right]\right\}p(d\boldsymbol{\theta}_i)
\end{aligned}$$

```
For each child node $\mathbf{x}_i$
    Initialise $\boldsymbol{\mu}_{i,0}$, $\boldsymbol{\Sigma}_{i,0}$ and $\boldsymbol{\xi}_{i,0}$
    For t=1 to t=T
        Initialise iterations counter: $k = 0$
        While ($k <$ maxIterations and error tolerance $\geq$ Tol)
            $k = k + 1$
            $\boldsymbol{\Sigma}_{i,t}^{-1(k)} = \boldsymbol{\Sigma}_{i,t-1}^{-1} + 2\phi(\boldsymbol{\xi}_{i,t-1})\mathbf{x}_{\pi(i),t}\mathbf{x}'_{\pi(i),t}$
            $\boldsymbol{\mu}_{i,t}^{(k)} = \boldsymbol{\Sigma}_{i,t}^{(k)} \left[\left(\frac{\mathbf{x}_{i,t}}{2} - 2\phi(\boldsymbol{\xi}_{i,t-1})\alpha\right)\mathbf{x}_{\pi(i),t} + \boldsymbol{\Sigma}_{i,t-1}^{-1}\boldsymbol{\mu}_{i,t-1}\right]$
            $\boldsymbol{\xi}_{i,t}^{2(k)} = \alpha^2 + 2\alpha\boldsymbol{\mu}_{i,t}^{\prime(k)}\mathbf{x}_{\pi(i),t} + \mathrm{tr}\left(\left(\boldsymbol{\Sigma}_{i,t}^{(k)} + \boldsymbol{\mu}_{i,t}^{(k)}\boldsymbol{\mu}_{i,t}^{\prime(k)}\right)\mathbf{x}_{\pi(i),t}\mathbf{x}'_{\pi(i),t}\right)$
            Compute tolerance
        End While($k$)
    End For($t$)
End For($i$)
```

Figure 3: EM for fully observed logistic BNs.

Proceeding as in the previous section, one can easily obtain the following recursive formulas for $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$ and $\boldsymbol{\xi}$

$$
\begin{aligned}
\boldsymbol{\Sigma}_{i,t}^{-1} &= \boldsymbol{\Sigma}_{i,t-1}^{-1} + 2\phi(\boldsymbol{\xi}_{i,t-1})\mathbb{E}_{\widehat{q}(\mathbf{x}_i^h|\boldsymbol{\lambda}_i)}\left[\mathbf{x}_{\pi(i),t}\mathbf{x}'_{\pi(i),t}\right] \\
\boldsymbol{\mu}_{i,t} &= \boldsymbol{\Sigma}_{i,t}\left(\mathbb{E}_{\widehat{q}(\mathbf{x}_i^h|\boldsymbol{\lambda}_i)}\left[\left(\frac{\mathbf{x}_{i,t}}{2} - 2\phi(\boldsymbol{\xi}_{i,t-1})\alpha\right)\mathbf{x}_{\pi(i),t}\right] + \boldsymbol{\Sigma}_{i,t-1}^{-1}\boldsymbol{\mu}_{i,t-1}\right) \\
\boldsymbol{\xi}_{i,t}^2 &= \alpha^2 + 2\alpha\boldsymbol{\mu}'_{i,t}\mathbb{E}_{\widehat{q}(\mathbf{x}_i^h|\boldsymbol{\lambda}_i)}\left[\mathbf{x}_{\pi(i),t}\right] + \mathrm{tr}\left(\left(\boldsymbol{\Sigma}_{i,t} + \boldsymbol{\mu}_{i,t}\boldsymbol{\mu}'_{i,t}\right)\mathbb{E}_{\widehat{q}(\mathbf{x}_i^h|\boldsymbol{\lambda}_i)}\left[\mathbf{x}_{\pi(i),t}\mathbf{x}'_{\pi(i),t}\right]\right)
\end{aligned}
$$

To obtain an update equation for the variational distribution, q, we introduce the following parametric mean field approximation

$$
\widehat{q}(\mathbf{x}^h|\boldsymbol{\lambda}) = \prod_{\{j;\mathbf{x}_j\in\mathcal{X}^h\}} \boldsymbol{\lambda}_j^{\frac{\mathbf{x}_j+1}{2}}(1-\boldsymbol{\lambda}_j)^{-\frac{\mathbf{x}_j-1}{2}}
$$

That is, each hidden node is represented by an independent Bernoulli distribution. To find the optimal parameters, we need to compute $\arg\max_{\boldsymbol{\lambda}} \mathcal{Q}$, where

$$\begin{aligned}\mathcal{Q} &\propto \mathbb{E}_{\widehat{p}(d\boldsymbol{\theta}|\mathbf{x}^v,\mathbf{x}^v_\pi,\boldsymbol{\lambda}^{\text{old}},\boldsymbol{\xi}^{\text{old}})} \left[\mathcal{H}(\widehat{q}) + \log\left(\prod_{i=1}^{n_{xc}} \exp\left\{\mathbb{E}_{\widehat{q}(\mathbf{x}^h_i|\boldsymbol{\lambda}_i)} \left[\log \widehat{p}(\mathbf{x}_i|\mathbf{x}_{\pi(i)},\boldsymbol{\theta}_i,\boldsymbol{\xi}_i)\right]\right\} p(d\boldsymbol{\theta}_i)\right)\right] \\ &= \mathcal{H}(\widehat{q}) + \mathbb{E}_{\widehat{p}(d\boldsymbol{\theta}|\mathbf{x}^v,\mathbf{x}^v_\pi,\boldsymbol{\lambda}^{\text{old}},\boldsymbol{\xi}^{\text{old}})} \left[\sum_{i=1}^{n_{xc}} \mathbb{E}_{\widehat{q}(\mathbf{x}^h_i|\boldsymbol{\lambda}_i)} \left[\log \widehat{p}(\mathbf{x}_i|\mathbf{x}_{\pi(i)},\boldsymbol{\theta}_i,\boldsymbol{\xi}_i)\right] + \log\left(p(d\boldsymbol{\theta}_i)\right)\right] \\ &= \mathcal{H}(\widehat{q}) + \mathbb{E}_{\widehat{p}(d\boldsymbol{\theta}|\mathbf{x}^v,\mathbf{x}^v_\pi,\boldsymbol{\lambda}^{\text{old}},\boldsymbol{\xi}^{\text{old}})} \left[\sum_{i=1}^{n_{xc}} \mathbb{E}_{\widehat{q}(\mathbf{x}^h_i|\boldsymbol{\lambda}_i)} \left[\frac{\varphi_i - \boldsymbol{\xi}_i}{2} - \phi(\boldsymbol{\xi}_i)\left(\varphi_i^2 - \boldsymbol{\xi}_i^2\right)\right] + \log\left(p(d\boldsymbol{\theta}_i)\right)\right]\end{aligned}$$

We can accomplish this by computing the derivative $\frac{\partial}{\partial \lambda_j}\mathcal{Q}$ (for all j such that $\mathbf{x}_j \in \mathcal{X}^h$) and equating to zero. In doing so, we first notice that $\frac{\partial}{\partial \lambda_j}\mathcal{H}(\widehat{q}) = \log\frac{1-\lambda_j}{\lambda_j}$. Consequently,

$$\lambda_j = \frac{\exp\left(D_j\right)}{1+\exp\left(D_j\right)}$$

where,

$$D_j = \frac{\partial}{\partial \lambda_j}\mathbb{E}_{\widehat{q}(\mathbf{x}^h_j|\boldsymbol{\lambda}_j)}\left[\frac{\mathbb{E}_{\widehat{p}(d\boldsymbol{\theta}|\mathbf{x}^v,\mathbf{x}^v_\pi,\boldsymbol{\lambda}^{\text{old}},\boldsymbol{\xi}^{\text{old}})}\left[\varphi_j - \boldsymbol{\xi}_j\right]}{2} - \phi(\boldsymbol{\xi}_j)\mathbb{E}_{\widehat{p}(d\boldsymbol{\theta}|\mathbf{x}^v,\mathbf{x}^v_\pi,\boldsymbol{\lambda}^{\text{old}},\boldsymbol{\xi}^{\text{old}})}\left[\varphi_j^2 - \boldsymbol{\xi}_j^2\right]\right]. \tag{7}$$

The EM algorithm for logistic BNs with hidden nodes is analogous to the one for fully observed BNs, with the exception that now one has to compute expectations with respect to $\mathbb{E}_{\widehat{q}(\mathbf{x}^h_j|\boldsymbol{\lambda}_j)}$ and include equation (7). As an example, if an observed node, $\boldsymbol{\theta}_i$ has a hidden parent, $\mathbf{x}_{\pi(i),j}$, the second term on the right hand side of equation (7) is equal to zero, yielding

$$D_i = \frac{\partial}{\partial \lambda_j}\mathbb{E}_{\widehat{q}(\mathbf{x}_{\pi(i),j}|\boldsymbol{\lambda}_j)}\left[\frac{\mathbf{x}_i\boldsymbol{\mu}_i'\mathbf{x}_{\pi(i)} - \boldsymbol{\xi}_i}{2}\right]. \tag{8}$$

4 Variational MCMC

In the previous section, we showed how variational methods can be used to map a complex problem to a simpler problem, to which one can apply methods that exploit some of the analytical properties of the functions under consideration. Such a strategy, of course, can result in biased estimates. To correct for this error, we can resort MCMC simulation. In particular, we shall use the variational posterior distribution, $\widehat{p}(d\boldsymbol{\theta}|\mathbf{x}^v, \mathbf{x}^v_\pi, \boldsymbol{\lambda}, \boldsymbol{\xi})$, as the proposal distribution for various MCMC samplers. Before we can explain how this is done, we need to introduce some basic notions of MCMC simulation.

MCMC techniques are a set of powerful simulation methods that may be applied to solve integration and optimisation problems in large dimensional spaces (Gilks et al. 1996, Robert and Casella 1999, Tierney 1994). These two types of problem play a fundamental role in the fields of machine learning, physics, econometrics, statistics and decision analysis. In the context of maximum likelihood estimation, MCMC techniques can be used for carrying out the necessary maximisations (Geyer and Thompson 1992). Within the Bayesian framework, given some unknown variables $\boldsymbol{\theta} \in \boldsymbol{\Theta}$ and data $\mathbf{y} \in \boldsymbol{\mathcal{Y}}$, MCMC simulation can be adopted to solve the following integration problems (Brooks 1998, Gilks, Thomas and Spiegelhalter 1994)

Normalisation: To obtain the posterior distribution $p(d\boldsymbol{\theta}|\mathbf{y})$ given the prior $p(d\boldsymbol{\theta})$ and likelihood $p(\mathbf{y}|\boldsymbol{\theta})$, the normalising factor in Bayes' theorem needs to be computed

$$p(d\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\theta})p(d\boldsymbol{\theta})}{\int_{\boldsymbol{\Theta}} p(\mathbf{y}|\boldsymbol{\theta})p(d\boldsymbol{\theta})}$$

Marginalisation: Given the joint posterior of $(\boldsymbol{\theta}, \mathbf{z}) \in \boldsymbol{\Theta} \times \boldsymbol{\mathcal{Z}}$, we may often be interested in the marginal posterior

$$p(d\boldsymbol{\theta}|\mathbf{y}) = \int_{\boldsymbol{\mathcal{Z}}} p(d\boldsymbol{\theta}, d\mathbf{z}|\mathbf{y})$$

Expectation: The objective of the analysis is often to obtain summary statistics of the form

$$\mathbb{E}(f(\boldsymbol{\theta})|\mathbf{y}) = \int_{\boldsymbol{\Theta}} f(\boldsymbol{\theta})p(d\boldsymbol{\theta}|\mathbf{y})$$

for some function of interest $f : \boldsymbol{\Theta} \to \mathbb{R}^{n_f}$ integrable with respect to $p\left(d\boldsymbol{\theta}|\,\mathbf{y}\right)$. Examples of appropriate functions include the conditional mean, in which case $f\left(\boldsymbol{\theta}\right) = \boldsymbol{\theta}$, or the conditional covariance of $\boldsymbol{\theta}$ where $f\left(\boldsymbol{\theta}\right) = \boldsymbol{\theta}\boldsymbol{\theta}' - \mathbb{E}_{p(\,d\boldsymbol{\theta}|\mathbf{y})}\left[\boldsymbol{\theta}\right]\mathbb{E}'_{p(\,d\boldsymbol{\theta}|\mathbf{y})}\left[\boldsymbol{\theta}\right]$.

We emphasize again that the difficult problem of computing integrals is not only restricted to Bayesian learning. For example, in statistical mechanics, one needs to compute the partition function, Z, of a system with states, s, and Hamiltonian (potential and kinetic energy), $E(s)$,

$$Z = \sum_s \exp\left[-\frac{E(s)}{kT}\right]$$

where k is the Boltzmann's constant and T denotes the temperature of the system. It turns out that the basic problem of equilibrium statistical mechanics is to compute this sum, which becomes and integral continuum systems and a trace for quantum mechanical systems (Baxter 1982).

The idea of perfect Monte Carlo integration methods is to draw an i.i.d. set of samples $\{\boldsymbol{\theta}^{(i)}; i = 1, 2, \ldots, N\}$ from the target distribution $p(d\boldsymbol{\theta})$ (it could be the posterior, $p(d\boldsymbol{\theta}|\mathbf{y})$, in Bayesian analysis) to obtain the following empirical distribution

$$P_N\left(d\boldsymbol{\theta}\right)=\frac{1}{N}\sum_{i=1}^{N}\delta_{\boldsymbol{\theta}^{(i)}}\left(d\boldsymbol{\theta}\right)$$

where $\delta_{\boldsymbol{\theta}^{(i)}}\left(d\boldsymbol{\theta}\right)$ denotes the delta-Dirac mass located in $\boldsymbol{\theta}^{(i)}$. Consequently, one can approximate the integrals, $I\left(f\right)$, by discrete sums, $I_N\left(f\right)$, as follows

$$I_N\left(f\right)=\frac{1}{N}\sum_{i=1}^{N}f(\boldsymbol{\theta}^{(i)})\ \xrightarrow[N\rightarrow\infty]{a.s.}\ I\left(f\right)=\int_{\boldsymbol{\Theta}}f(\boldsymbol{\theta})p(d\boldsymbol{\theta}) \qquad (9)$$

The estimate $I_N\left(f\right)$ is unbiased and by the strong law of large numbers, it will almost surely converge to $I\left(f\right)$. That is

$$P\left(\lim_{N\rightarrow\infty}I_N\left(f\right)=I\left(f\right)\right)=1$$

If the variance of $f\left(\boldsymbol{\theta}\right)$ satisfies $\sigma_f^2\triangleq\mathbb{E}_{p(d\boldsymbol{\theta})}\left[f^2\left(\boldsymbol{\theta}\right)\right]-I^2\left(f(\boldsymbol{\theta})\right)<+\infty$, then the variance of $I_N\left(f\right)$ is equal to $var\left(I_N\left(f\right)\right)=\frac{\sigma_f^2}{N}$ and a central limit theorem yields convergence in distribution of the error

$$\sqrt{N}\left(I_N\left(f\right)-I\left(f\right)\right)\underset{N\rightarrow+\infty}{\Longrightarrow}\mathcal{N}(0,\sigma_f^2)$$

where $\Longrightarrow$ denotes convergence in distribution (Robert and Casella 1999, Section 3.2). The advantage of Monte Carlo integration over deterministic integration arises from the fact that the former positions the integration grid (samples) in regions of high probability. On the other hand, the main disadvantage of simple Monte Carlo methods is that often it is not possible to draw samples from $p(d\boldsymbol{\theta})$ directly. This problem can, however, be circumvented by the introduction of MCMC algorithms. Assuming that we can draw samples from a proposal distribution $\pi(d\boldsymbol{\theta})$, the key idea of MCMC simulation is to design Markov chain mechanisms that cause the proposed samples to migrate so that their empirical distribution approximates $p(d\boldsymbol{\theta})$.

The most popular example of this class of algorithms is the Metropolis-Hastings (MH) algorithm (Hastings 1970, Metropolis, Rosenbluth, Rosenbluth, Teller and Teller 1953). A Metropolis-Hastings step of invariant distribution, say $p\left(d\boldsymbol{\theta}\right)$, and proposal distribution, say $\pi\left(d\boldsymbol{\theta}^{\star}\middle|\,\boldsymbol{\theta}\right)$, involves sampling a candidate value $\boldsymbol{\theta}^{\star}$ given the current value $\boldsymbol{\theta}$ according to $\pi\left(d\boldsymbol{\theta}^{\star}\middle|\,\boldsymbol{\theta}\right)$. The Markov chain then moves towards $\boldsymbol{\theta}^{\star}$ with acceptance probability $\mathcal{A}(\boldsymbol{\theta},\boldsymbol{\theta}^{\star})=\min\{1,\left[p(d\boldsymbol{\theta})\pi(d\boldsymbol{\theta}^{\star}|\boldsymbol{\theta})\right]^{-1}p(d\boldsymbol{\theta}^{\star})\pi(d\boldsymbol{\theta}|\boldsymbol{\theta}^{\star})\}$, otherwise it remains at $\boldsymbol{\theta}$. The pseudo-code is shown in Figure 4.

In the pseudo-code, we assume that the proposal and target distributions admit densities with respect to the Lebesgue or counting measures. The transition kernel associated with

```
1. Initialise $\boldsymbol{\theta}^{(0)}$ and set $i = 0$.
2. Iteration $i + 1$
    • Sample $u \sim \mathcal{U}_{[0,1]}$.
    • Sample $\boldsymbol{\theta}^{(i+1)\star}$ from $\pi(d\boldsymbol{\theta}^{(i+1)\star}|\boldsymbol{\theta}^{(i)})$.
    • If $u < \mathcal{A}(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i+1)\star}) = \min\left\{1, \frac{p(\boldsymbol{\theta}^{(i+1)\star})\pi(\boldsymbol{\theta}^{(i)}|\boldsymbol{\theta}^{(i+1)\star})}{p(\boldsymbol{\theta}^{(i)})\pi(\boldsymbol{\theta}^{(i+1)\star}|\boldsymbol{\theta}^{(i)})}\right\}$
            $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^{(i+1)\star}$
      else
            $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^{(i)}$
3. $i + 1 \leftarrow i + 2$ and go to 2.
```

Figure 4: Metropolis-Hastings algorithm.

the MH algorithm, assuming Lebesgue measure for more generality, is given by

$$K(\boldsymbol{\theta}^{(i)}, A) = \int_A \mathcal{K}(\boldsymbol{\theta}^{(i)}, d\boldsymbol{\theta}^{(i+1)\star}) + r(\boldsymbol{\theta}^{(i)})\mathbb{I}_A(\boldsymbol{\theta}^{(i)}) \tag{10}$$

where

$$\mathcal{K}(\boldsymbol{\theta}^{(i)}, d\boldsymbol{\theta}^{(i+1)\star}) = \pi(d\boldsymbol{\theta}^{(i+1)\star}|\boldsymbol{\theta}^{(i)})\mathcal{A}(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i+1)\star})$$

and

$$\mathcal{A}(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i+1)\star}) = \min\left\{1, \frac{p(d\boldsymbol{\theta}^{(i+1)\star})\pi(d\boldsymbol{\theta}^{(i)}|\boldsymbol{\theta}^{(i+1)\star})}{p(d\boldsymbol{\theta}^{(i)})\pi(d\boldsymbol{\theta}^{(i+1)\star}|\boldsymbol{\theta}^{(i)})}\right\}$$

is the probability associated with a candidate being accepted, while the probability of staying at the same point is $1 - \mathcal{A}(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i+1)\star})$. The rejection term is, therefore, given by

$$r(\boldsymbol{\theta}^{(i)}) = 1 - \int_{\mathcal{X}} \pi(d\boldsymbol{\theta}^{(i+1)\star}|\boldsymbol{\theta}^{(i)})\mathcal{A}(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(i+1)\star})$$

It is fairly easy to prove that the samples generated by MH algorithm will mimic samples drawn from the target distribution (a property known as ergodicity). By construction, $K(\cdot, d\cdot)$ satisfies the detailed balance condition (reversibility). That is,

$$\begin{aligned} p(d\boldsymbol{\theta}^{(i)})\mathcal{K}(\boldsymbol{\theta}^{(i)}, d\boldsymbol{\theta}^{(i+1)\star}) &= p(d\boldsymbol{\theta}^{(i+1)\star})\mathcal{K}(\boldsymbol{\theta}^{(i+1)\star}, d\boldsymbol{\theta}^{(i)}) \\ p(d\boldsymbol{\theta}^{(i)})r(\boldsymbol{\theta}^{(i+1)\star})\mathbb{I}_A(\boldsymbol{\theta}^{(i+1)\star}) &= p(d\boldsymbol{\theta}^{(i+1)\star})r(\boldsymbol{\theta}^{(i)})\mathbb{I}_A(\boldsymbol{\theta}^{(i)}) \end{aligned}$$

it follows that for any measurable set A

$$\begin{aligned}
\int_{\boldsymbol{\Theta}} K(\boldsymbol{\theta}^{(i)}, A)p(d\boldsymbol{\theta}^{(i)}) &= \int_{\boldsymbol{\Theta}}\int_A K(\boldsymbol{\theta}^{(i)}, d\boldsymbol{\theta}^{(i+1)\star})p(d\boldsymbol{\theta}^{(i)}) \\
&= \int_{\boldsymbol{\Theta}}\int_A K(\boldsymbol{\theta}^{(i+1)\star}, d\boldsymbol{\theta}^{(i)})p(d\boldsymbol{\theta}^{(i+1)\star}) \\
&= \int_A p(d\boldsymbol{\theta}^{(i+1)\star}) = p(A) \qquad (11)
\end{aligned}$$

since $\int_{\boldsymbol{\Theta}} \mathcal{K}(\boldsymbol{\theta}^{(i+1)\star}, d\boldsymbol{\theta}^{(i)}) = 1$. Thus, by construction, the MH algorithm admits p as invariant distribution. To show that the MH algorithm converges, we need to ensure that there are no cycles (aperiodicity) and that every state that has positive probability can be reached in a finite number of steps (irreducibility). Since the algorithm always allows for rejection, it follows that it is aperiodic. To ensure irreducibility, we simply need to make sure that $\pi(\cdot) > 0$ over the entire state space. Under these conditions, we obtain the convergence result of equation (9) (Tierney 1994, Theorem 3, page 1717). If the space $\boldsymbol{\Theta}$ is small (for example, bounded in $\mathbb{R}^n$), then it is possible to use minorisation conditions to prove uniform (geometric) ergodicity (Meyn and Tweedie 1993). It is also possible to prove geometric ergodicity using Foster-Lyapunov drift conditions (Meyn and Tweedie 1993, Roberts and Tweedie 1996).

Some properties of the MH algorithm are worth mentioning. Firstly, the normalising constants of the target distribution are not required. We only need to know the target distribution up to a constant of proportionality. Secondly, although the pseudo-code makes use of a single chain, it is easy to simulate several chains in parallel. Finally, *the success or failure of the algorithm often hinges on the choice of proposal distribution.* This is demonstrated in Figure 5. Here the proposal is a simple random walk, $\pi(\boldsymbol{\theta}^{(i+1)\star}|\boldsymbol{\theta}^{(i)}) = \mathcal{N}(0, \sigma^{*2})$. If the proposal is too narrow, only one mode of $p(d\boldsymbol{\theta})$ might be visited. On the other hand, if it is too wide, the rejection rate can be very high. If all the modes are visited while the acceptance probability is high, the chain is said to "mix" well. In the following subsections, we show how one can use the variational approximation as the proposal distribution so as to improve the mixing of the chains in some scenarios.

4.1 Naive variational MCMC approach

The most obvious and immediate way of improving the variational approximation using MCMC is to sample new candidates according to the variational distribution. That is,

$$\pi(d\boldsymbol{\theta}^{(i+1)\star}|\boldsymbol{\theta}^{(i)}) = \widehat{p}(d\boldsymbol{\theta}^{(i+1)\star}|\mathbf{x}^v, \mathbf{x}^v_\pi, \boldsymbol{\lambda}, \boldsymbol{\xi})$$

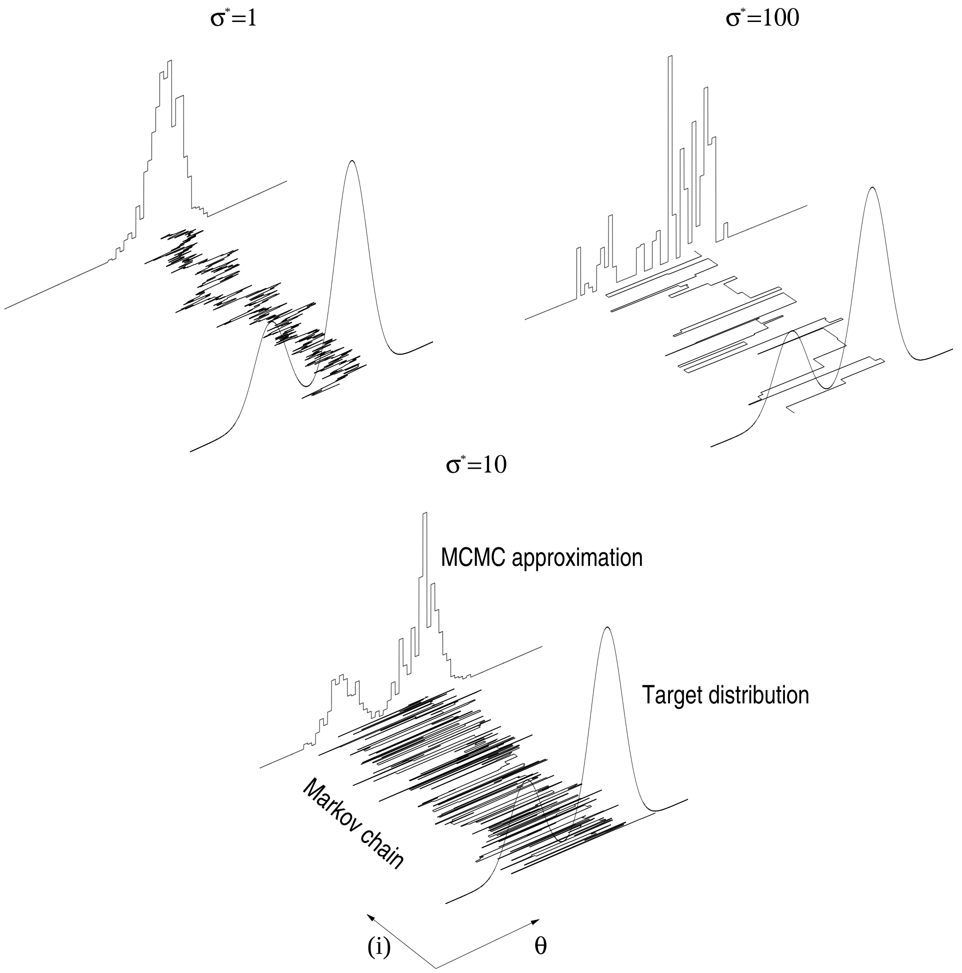

Figure 5: Approximations obtained using the Metropolis algorithm with three Gaussian proposal distributions of different variances.

In this case, the acceptance probability of the MH algorithm simplifies to

$$\begin{aligned}\mathcal{A}(\boldsymbol{\theta}^{(i)},\boldsymbol{\theta}^{(i+1)\star}) &= \min\left\{1,\frac{p(d\boldsymbol{\theta}^{(i+1)\star}|\mathbf{x}^{v},\mathbf{x}_{\pi}^{v})\widehat{p}(d\boldsymbol{\theta}^{(i)}|\mathbf{x}^{v},\mathbf{x}_{\pi}^{v},\boldsymbol{\lambda},\boldsymbol{\xi})}{p(d\boldsymbol{\theta}^{(i)}|\mathbf{x}^{v},\mathbf{x}_{\pi}^{v})\widehat{p}(d\boldsymbol{\theta}^{(i+1)\star}|\mathbf{x}^{v},\mathbf{x}_{\pi}^{v},\boldsymbol{\lambda},\boldsymbol{\xi})}\right\} \\ &= \min\left\{1,\frac{w(\boldsymbol{\theta}^{(i+1)\star})}{w(\boldsymbol{\theta}^{(i)})}\right\}\end{aligned}$$

where $w(\cdot) \triangleq p(\cdot)/\widehat{p}(\cdot)$ denotes the importance weights. This type of algorithm is known as the independent MH algorithm and it is closely related to the standard importance sampler (Geweke 1989). In the previous section, we pointed out that this algorithm will converge to the posterior distribution under mild conditions. Moreover, we can state some encouraging results using "metrics" commonly used in the variational literature; namely, since $p(d\boldsymbol{\theta}|\mathbf{x},\mathbf{x}_{\pi})$ is the unique invariant distribution of the Markov chain, it follows that the relative entropy (Kullback Leibler "distance" between the true posterior and the MCMC approximation) converges to zero as the number of iterations increases (Cover and Thomas

1991). However, both the importance sampler and independent MH algorithm are well known to perform poorly in high dimensions unless the proposal distribution is very close to the target distribution (Geweke 1989, Mengersen and Tweedie 1996). (In practice, the acceptance ratio usually tends to zero after approximately 10 dimensions.) In fact, we have the following result

Proposition 1 (Mengersen and Tweedie 1996, Theorem 2.1) *The independent MH algorithm converges at a uniformly (geometric) rate if there exists a constant $\beta > 0$ such that*

$$\frac{p(\boldsymbol{\theta}|\mathbf{x}^v, \mathbf{x}^v_\pi)}{\widehat{p}(\boldsymbol{\theta}|\mathbf{x}^v, \mathbf{x}^v_\pi)} \leq \beta, \qquad \boldsymbol{\theta} \in supp(p(\boldsymbol{\theta}))$$

in which case,

$$\|K^{(i)}(\boldsymbol{\theta}, .) - p\|_{TV} \leq 2\left(1 - \frac{1}{\beta}\right)^i$$

where $\|\cdot\|_{TV}$ denotes the total variation norm. Conversely, if there exists a set of positive measure where the bound on the importance weights does not hold, then the algorithm is not even geometrically ergodic.

The negative result in this proposition is, perhaps, the most interesting one. Unless we can bound the importance weights in the regions of high probability and in the tails, the approach is bound to fail. One can apply the result of Proposition (1) to obtain the following corollary

Corollary 1 (Uniform Ergodicity of naive variational MCMC) *The independent MH algorithm for logistic BNs, using the variational approximation, $\mathcal{N}(\widehat{\boldsymbol{\mu}}, \widehat{\boldsymbol{\Sigma}})$, as proposal distribution, converges at a uniformly (geometric) rate if*

$$(\boldsymbol{\theta} - \boldsymbol{\mu}_0)'\boldsymbol{\Sigma}_0^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}_0) - (\boldsymbol{\theta} - \widehat{\boldsymbol{\mu}})'\widehat{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{\theta} - \widehat{\boldsymbol{\mu}}) \geq 0 \tag{12}$$

in which case,

$$\|K^{(i)}(\boldsymbol{\theta}, .) - p\|_{TV} \leq 2\left(1 - \frac{1}{\beta}\right)^i$$

The converse result of Proposition (1) also applies.

Proof. Since both the target distribution and the variational approximation to it are proper and since the likelihood is bounded for all possible values of $\boldsymbol{\theta}$, we only require that the ratio of the prior distribution, $\mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$, to the proposal distribution, $\mathcal{N}(\widehat{\boldsymbol{\mu}}, \widehat{\boldsymbol{\Sigma}})$, be bounded. It is then trivial to see that this is the case when condition (12) is satisfied ■

In the one-dimensional case, the bound in the previous corollary is satisfied when the variance of the prior distribution is less than or equal to the variance of the proposal distribution.

4.2 Block MCMC approach

In the previous section, we argued that the acceptance rate of the independent MH sampler can be very low in high dimensions. To surmount this problem to a certain extent, we can exploit the nature of the variational approximation and propose to update the parameters in blocks. The modified algorithm, using b_j to denote the size of the j-th block and n_b to denote the number of blocks, is shown in Figure 6. It uses the notation $\boldsymbol{\theta}^{(i+1)}_{-[b_j+1:b_{j+1}]} \triangleq \{\boldsymbol{\theta}^{(i+1)}_{1:b_1}, \boldsymbol{\theta}^{(i+1)}_{b_1+1:b_2}, \ldots, \boldsymbol{\theta}^{(i+1)}_{b_{j-1}+1:b_j}, \boldsymbol{\theta}^{(i)}_{b_{j+1}+1:b_j}, \ldots, \boldsymbol{\theta}^{(i)}_{b_{n_b}-1+1:b_{n_b}}\}$. (This algorithm includes the Gibbs sampler as a special case; when the proposals correspond to the full conditionals and the acceptance is equal to 1 (Geman and Geman 1984).) Each proposal distribution corresponds to a Gaussian distribution whose mean is a subset of the elements of the mean of the original variational distribution and whose covariance is the corresponding block-diagonal component of the original covariance.

```
1. Initialise $\boldsymbol{\theta}^{(0)}$ and set $i = 0$.
2. Iteration $i + 1$
    • Sample the block $\boldsymbol{\theta}^{(i+1)}_{1:b_1}$ according to an MH step with proposal distribution $\widehat{p}_1(d\boldsymbol{\theta}^{(i+1)}_{1:b_1} | \boldsymbol{\theta}^{(i+1)}_{-[1:b_1]}, \boldsymbol{\theta}^{(i)}_{1:b_1}, \mathbf{x}^v, \mathbf{x}^v_\pi)$ and invariant distribution $p(d\boldsymbol{\theta}^{(i+1)}_{1:b_1} | \boldsymbol{\theta}^{(i+1)}_{-[1:b_1]}, \mathbf{x}^v, \mathbf{x}^v_\pi)$.
    • Sample the block $\boldsymbol{\theta}^{(i+1)}_{b_1+1:b_2}$ according to an MH step with proposal distribution $\widehat{p}_2(d\boldsymbol{\theta}^{(i+1)}_{b_1+1:b_2} | \boldsymbol{\theta}^{(i+1)}_{-[b_1+1:b_2]}, \boldsymbol{\theta}^{(i)}_{b_1+1:b_2}, \mathbf{x}^v, \mathbf{x}^v_\pi)$ and invariant distribution $p(d\boldsymbol{\theta}^{(i+1)}_{b_1+1:b_2} | \boldsymbol{\theta}^{(i+1)}_{-[b_1+1:b_2]}, \mathbf{x}^v, \mathbf{x}^v_\pi)$.
    ⋮
    • Sample the block $\boldsymbol{\theta}^{(i+1)}_{b_{n_{b-1}}+1:b_{n_b}}$ according to an MH step with proposal distribution $\widehat{p}_{n_b}(d\boldsymbol{\theta}^{(i+1)}_{b_{n_b}-1+1:b_{n_b}} | \boldsymbol{\theta}^{(i+1)}_{-[b_{n_b}-1+1:b_{n_b}]}, \boldsymbol{\theta}^{(i)}_{b_{n_b}-1+1:b_{n_b}}, \mathbf{x}^v, \mathbf{x}^v_\pi)$ and invariant distribution $p(d\boldsymbol{\theta}^{(i+1)}_{b_{n_b}-1+1:b_{n_b}} | \boldsymbol{\theta}^{(i+1)}_{-[b_{n_b}-1+1:b_{n_b}]}, \mathbf{x}^v, \mathbf{x}^v_\pi)$.
3. $i + 1 \leftarrow i + 2$ and go to 2.
```

Figure 6: Block variational MH algorithm.

The transition kernel for this algorithm is given by the following expression

$$K(\boldsymbol{\theta}^{(i)}, A) = \prod_{j=1}^{n_b} K_{MH-j}(\boldsymbol{\theta}^{(i)}_{b_{j-1}+1:b_j}, \boldsymbol{\theta}^{(i+1)}_{-[b_{j-1}+1:b_j]}; A_j)$$

where $K_{MH-j}(\cdot; d\cdot)$ denotes the j-th MH algorithm in the cycle. Since this kernel allows one to visit all sets of positive measure, while being aperiodic, the algorithm's simple convergence holds true as the number of samples becomes very large.

Obviously, choosing the size of the blocks poses some trade-offs. If one samples the components of a multi-dimensional vector one-at-a-time, the chain may take a very long time to explore the target distribution. This problem gets worse as the correlation between the components increases. Alternatively, if one samples all the components together, then the probability of accepting this large move tends to be very low.

4.3 Mixtures of MCMC steps

A very powerful property of MCMC is that it is possible to combine several samplers into mixtures and cycles of the individual samplers (Tierney 1994). This way we can have global proposals to explore vast regions of the parameter space and local proposals to discover finer details of the target distribution (Andrieu, de Freitas and Doucet 2000, Andrieu and Doucet 1999). If the transition kernels K_1 and K_2 have invariant distribution $p(\cdot)$ each, then the *cycle hybrid kernel* K_1K_2 and the *mixture hybrid kernel* $\nu K_1 + (1-\nu)K_2$, for $0 \leq \nu \leq 1$, are also transition kernels with invariant distribution $p(\cdot)$.

In this paper, we combine the variational MCMC algorithm discussed in Section 4.2 with a random walk metropolis (also in blocks). This will be useful, for example, when the target distribution has many narrow peaks. Here, the variational proposal locks into a particular peak while the random walk allows one to explore the space around this peak. The pseudo-code for this mixture is shown in Figure 7.

```
1. Initialise $\boldsymbol{\theta}^{(0)}$ and set $i = 0$.
2. Iteration $i+1$
    • Sample $u \sim \mathcal{U}_{[0,1]}$.
    • If $u < \nu$
          Perform the block MH algorithm with the variational proposal.
    • else
          Perform a block Metropolis algorithm with a random walk proposal.
3. $i+1 \leftarrow i+2$ and go to 2.
```

Figure 7: Mixture MCMC algorithm.

5 Simulations

We performed experiments on fully and partially observed logistic BNs. When all the nodes are observed, the posterior is unimodal and symmetric. This allows us to compare the algorithms by evaluating the distance between their estimates of the mean and the optimal mean. The likelihood will be higher for estimates close to the optimal mean. Notice that the optimal mean can be very different from the generating mean. To illustrate this, we used a model with a single parameter set to 1 and generated 1000 observations. We repeated this four times and, each time, we evaluated the likelihood distribution on a discrete grid. As shown in Figure 8, the generating mean is not necessarily equal to the optimal mean. Our non-informative noise model is, therefore, not amenable to model testing techniques such as cross-validation. We also performed experiments on multimodal distributions that show the performance of the algorithm not only in terms of approximating the mean, but in terms of approximating the entire posterior distribution.

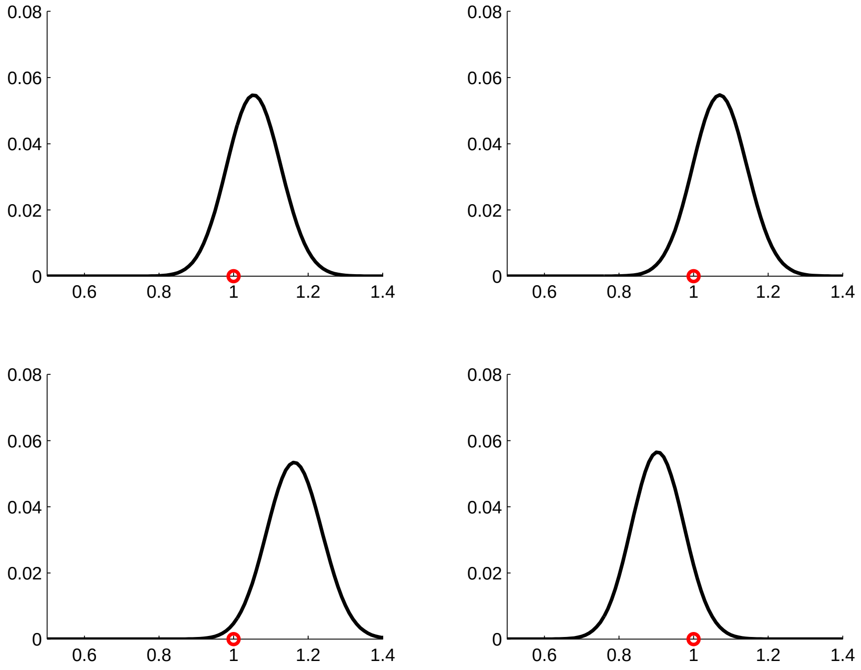

Figure 8: Likelihood of the data (1000 observations) when generated by a Bernoulli logistic node with a single parameter set to 1. Clearly, 1000 observations are not enough to recover the true value of the parameter. We are dealing with a very uninformative noise model and consequently standard cross-validation tests are not expected to perform well.

5.1 Unimodal models

We used a logistic model consisting of one child and a varying number of parents to generate sets of 1000 data samples. We then computed posterior approximations using the variational EM algorithm, the block M-H sampler with the variational proposal distribution (VarMCMC), the random walk Metropolis (RW), and the MCMC mixture with a variational kernel and a Metropolis kernel (VarMixMCMC). We repeated this experiment 10 times to obtain estimates of the performance in terms of means and error bars. We used 5000 MCMC samples, set the random walk variance to 0.01, the bias parameter to 0.5, the Bernoulli mean to 0.5 and the generating parameters to uniformly random values between on $(0, 1]$. We chose a fairly flat prior $\mathcal{N}(\mathbf{0}, 100\mathbf{I})$ The results are shown in Figure 9. It is clear that the VarMixMCMC algorithm outperforms the VarMCMC algorithm, which in turn outperforms the standard variational algorithm. The performance of the RW algorithms varies considerably because it depends on the initialisation and data set realisation. That is, it might or might not perform well depending on whether it is initialised in regions of high probability or not. Of course, as the number of samples goes to infinity, the RW algorithm will approximate the mean according to the central limit theorem. Yet, in

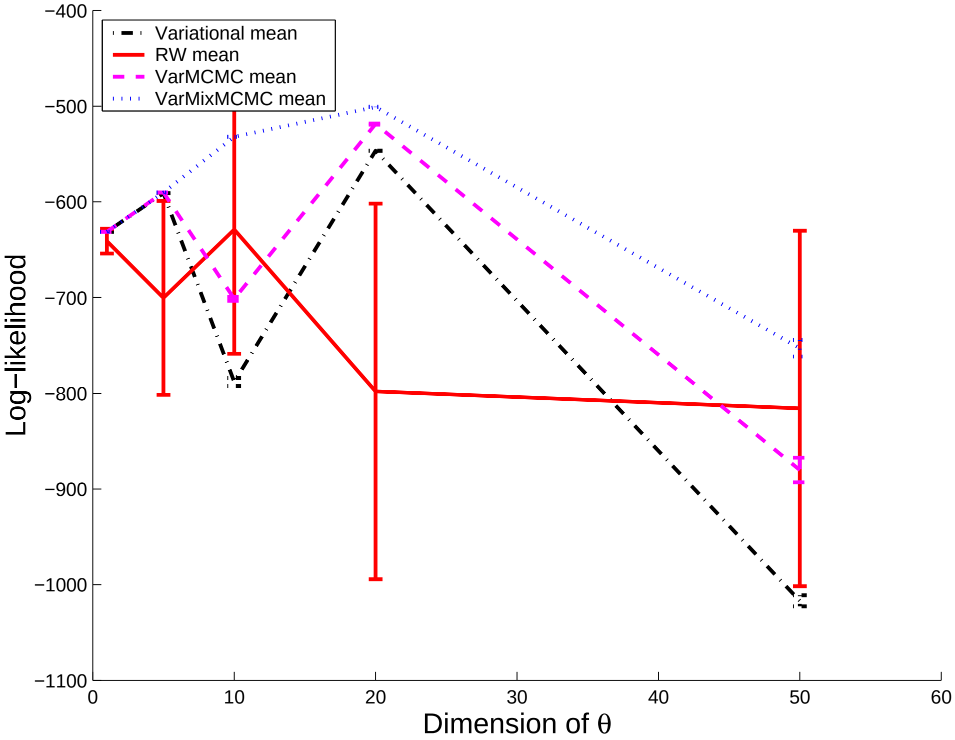

Figure 9: The MCMC mixture with variational and Metropolis kernels provides better estimates of the mean for different numbers of parents.

practical scenarios we often need reliable and faster options. Notice also that this example ...

Computational time ....

other performance measures next section ....

5.2 Multimodal models

In this experiment, we considered a network with two parents (one hidden and one observed). The posterior for $\boldsymbol{\theta}$ is, therefore, bivariate and can have two modes. These modes need not be symmetrical. For demonstration, we set the generating parameters for the hidden and observed nodes to 2 and -1 and the respective Bernoulli means of the hidden variables to 0.6 and 0.5. We set the bias parameter to 2, the number of data 50 and the prior to $\mathcal{N}(3, 10\mathbf{I})$. The posterior in this case can be evaluated numerically on a two-dimensional grid. We show its contour curves in Figure 10.

The posterior is bimodal and asymmetric. The figure also shows the contour plot of the RW MCMC histogram after 1000 iterations and the variational approximation. We notice

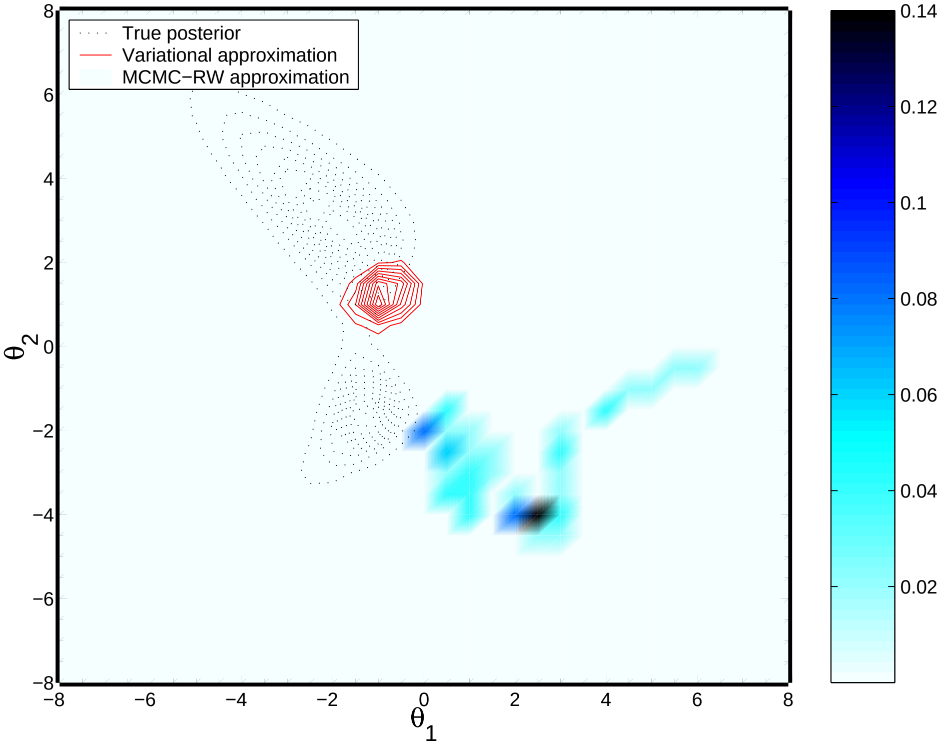

Figure 10: Convergence of the random walk Metropolis algorithm after 1000 iterations for a bivariate model. These contour plots indicate that the random walk can spend a considerable time in regions of low probability.

that the variational approximation fits closely to one of the modes. We also notice that if the random walk starts in a region of low probability, it can take long to locate one of the modes. Its performance will, therefore, be poor when dealing with posteriors with elongated contours.

....

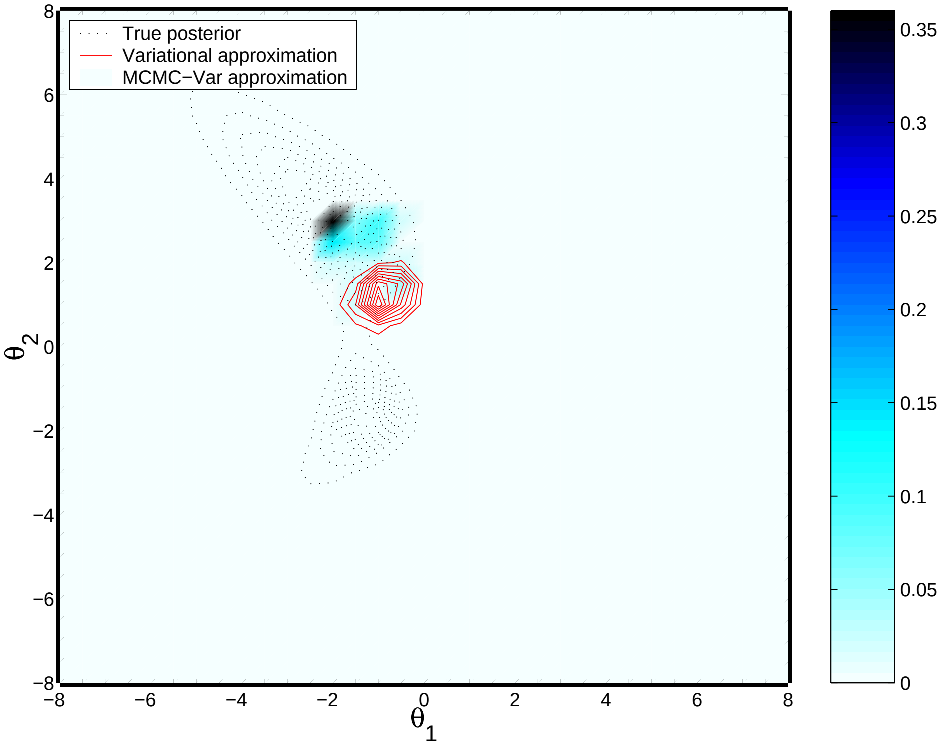

Figure 11: Convergence of the variational MCMC algorithm after 1000 iterations for a bivariate model. The variational approximation allows us to locate a region of high probability.

6 Conclusions

Mention generality of the method

QMR, mixtures.

It is possible to construct more complex and powerful sampling algorithms than the ones describe so far, while still exploiting the variational approximations. e.g. adaptive MCMC, parallel chains, approximating marginals only,

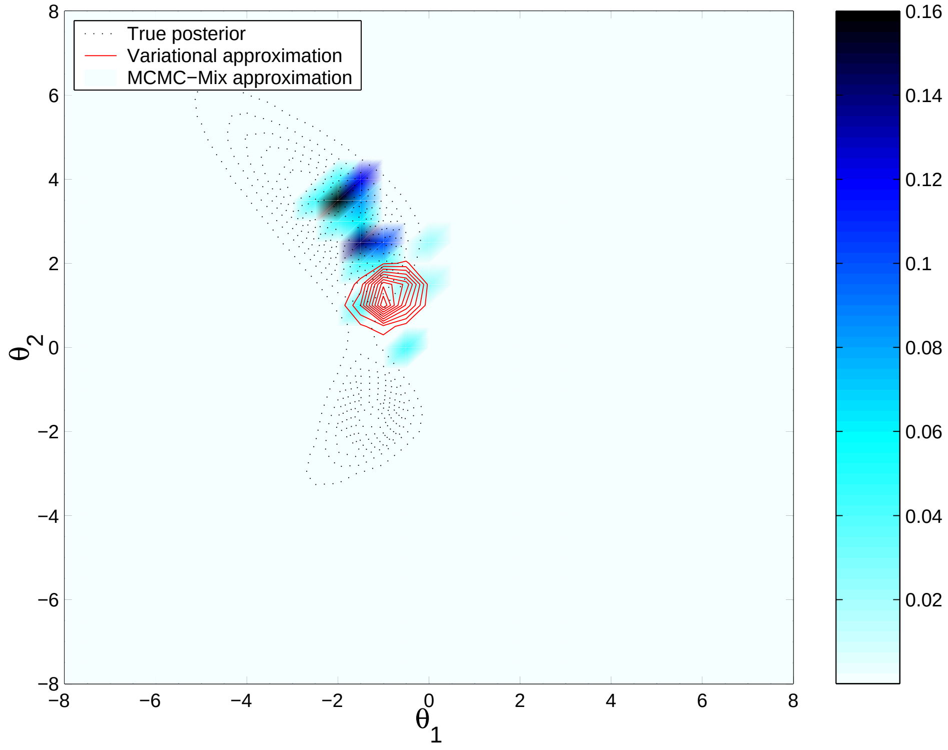

Figure 12: Convergence of the mixture of a random walk and variational MCMC algorithm after 1000 iterations for a bivariate model. The variational component allows us to locate a region of high probability and the random walk allows us to explore the neighbourhood of this region.

Acknowledgements

We would like to thank Christophe Andrieu, Mathew Beal, Arnaud Doucet, Zoubin Ghahramani, Tommi Jaakkola and Kevin Murphy.

Notation

`*** To be finished at the end ***`

Symbols

$\mathbf{z}_{1:t}$	Stacked vector $\mathbf{z}_{1:t} \triangleq (z_1, ..., z_{j-1}, z_j, z_{j+1}, ..., z_t)'$.
$\mathbf{z}_{-j}$	Vector with j-th component missing $\mathbf{z}_{-j} \triangleq (z_1, ..., z_{j-1}, z_{j+1}, ..., z_k)'$.
$\mathbf{A}_{i,j}$	Entry of the matrix $\mathbf{A}$ in the i^{th} row and j^{th} column.
$\mathbf{A}_{1:p,1:q,1:r}$	Three-dimensional matrix of size $p \times q \times r$.
$\mathbf{I}_n$	Identity matrix of dimension $n \times n$.
$\mathbb{R}^n$	Euclidean n-dimensional space.
$\mathbb{N}$	The set of natural numbers (positive integers).
$p(\mathbf{z})$	Distribution of $\mathbf{z}$.
$p(\mathbf{z}\mid\mathbf{y})$	Conditional distribution of $\mathbf{z}$ given $\mathbf{y}$.
$p(\mathbf{z}, \mathbf{y})$	Joint distribution of $\mathbf{z}$ and $\mathbf{y}$.
$\mathbf{z} \sim p(\mathbf{z})$	$\mathbf{z}$ is distributed according to $p\left(\mathbf{z}\right)$.
$\mathbf{z}\mid\mathbf{y} \sim p\left(\mathbf{z}\right)$	The conditional distribution of $\mathbf{z}$ given $\mathbf{y}$ is $p\left(\mathbf{z}\right)$.
$\mathcal{B}(\mathbf{\Theta})$	Sigma field of subsets of the space $\mathbf{\Theta}$.
$\mathcal{O}(N)$	The computation complexity is order N operations.

Operators and functions

$\mathbf{A}'$	Transpose of matrix $\mathbf{A}$.
$\mathbf{A}^{-1}$	Inverse of matrix $\mathbf{A}$.
$\mathrm{tr}(\mathbf{A})$	Trace of matrix $\mathbf{A}$.
$\vert\mathbf{A}\vert$	Determinant of matrix $\mathbf{A}$.
$\mathbb{I}_E(\mathbf{z})$	Indicator function of the set E (1 if $\mathbf{z} \in E$, 0 otherwise).
$\delta_{\mathbf{z}_i}(d\mathbf{z})$	Dirac delta function (impulse function).
$\lfloor z \rfloor$	Highest integer strictly less than z.
$\mathbb{E}(\mathbf{z})$	Expectation of the random variable $\mathbf{z}$.
$var(\mathbf{z})$	Variance of the random variable $\mathbf{z}$.
$\exp(\cdot)$	Exponential function.
$\Gamma(\cdot)$	Gamma function.
$\log(\cdot)$	Logarithmic function of base e (ln).
min, max	Extrema with respect to an integer value.
inf, sup	Extrema with respect to a real value.
$\arg\min\limits_{\mathbf{z}}$	The argument $\mathbf{z}$ that minimises the operand.
$\arg\max\limits_{\mathbf{z}}$	The argument $\mathbf{z}$ that maximises the operand.
$\Vert\mu\Vert_{\mathrm{TV}}$	Total variation norm $\Vert\mu\Vert_{\mathrm{TV}} \triangleq \sup\limits_{A \in \mathcal{B}(\mathbf{\Theta})} \mu(A) - \inf\limits_{A \in \mathcal{B}(\mathbf{\Theta})} \mu(A)$.

Standard probability distributions

Bernoulli	$\mathcal{B}r(\alpha)$	$\alpha^{\mathbf{z}}(1-\alpha)^{(1-\mathbf{z})}$
Gamma	$\mathcal{G}a\left(\alpha,\beta\right)$	$\frac{\beta^{\alpha}}{\Gamma(\alpha)}z^{\alpha-1}\exp\left(-\beta z\right)\mathbb{I}_{[0,+\infty)}\left(z\right)$
Gaussian	$\mathcal{N}\left(\mathbf{m},\Sigma\right)$	$\lvert 2\pi\Sigma\rvert^{-1/2}\exp\left(-\frac{1}{2}\left(\mathbf{z}-\mathbf{m}\right)'\Sigma^{-1}\left(\mathbf{z}-\mathbf{m}\right)\right)$
Inverse Gamma	$\mathcal{IG}\left(\alpha,\beta\right)$	$\frac{\beta^{\alpha}}{\Gamma(\alpha)}z^{-\alpha-1}\exp\left(-\beta/z\right)\mathbb{I}_{[0,+\infty)}\left(z\right)$
Poisson	$\mathcal{P}n\left(\lambda\right)$	$\frac{\lambda^{z}}{z!}\exp(-\lambda)\mathbb{I}_{\mathbb{N}}\left(z\right)$
Student t	$\mathcal{S}t(m,\lambda,\alpha)$	$\frac{\Gamma\left(\frac{1}{2}(\alpha+1)\right)}{\Gamma\left(\frac{1}{2}\alpha\right)}\left(\frac{\lambda}{\alpha\pi}\right)^{1/2}\left[1+\alpha^{-1}\lambda(z-m)^{2}\right]^{-(\alpha+1)/2}$
Uniform	$\mathcal{U}_{A}$	$\left[\int_{A}d\mathbf{z}\right]^{-1}\mathbb{I}_{A}\left(\mathbf{z}\right)$

References

Andrieu, C. and Doucet, A. (1999). Joint Bayesian detection and estimation of noisy sinusoids via reversible jump MCMC, *IEEE Transactions on Signal Processing*. To appear.

Andrieu, C., de Freitas, J. F. G. and Doucet, A. (2000). Robust full Bayesian methods for neural networks, *in* S. Solla, T. Leen and K.-R. Müller (eds), *Advances in Neural Information Processing Systems 12*, MIT Press, pp. 379–385.

Baxter, R. J. (1982). *Exactly Solved Models in Statistical Mechanics*, Academic Press.

Brooks, S. P. (1998). Markov chain Monte Carlo method and its application, *The Statistician* **47**(1): 69–100.

Chandler, D. (1987). *Introduction to Modern Statistical Mechanics*, Oxford University Press, Oxford, UK.

Cover, T. M. and Thomas, J. A. (1991). *Elements of Information Theory*, Wiley Series in Telecommunications, New York.

Cowell, R. G., Dawid, A. P., Lauritzen, S. L. and Spiegelhalter, D. J. (1999). *Probabilistic Networks and Expert Systems*, Springer-Verlag, New York.

de Freitas, J. F. G., Niranjan, M. and Gee, A. H. (2000). Dynamic learning with the EM algorithm for neural networks, *Journal of VLSI Signal Processing Systems* **26**(1/2): 119–131.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society Series B* **39**: 1–38.

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6**(6): 721–741.

Geweke, J. (1989). Bayesian inference in econometric models using Monte Carlo integration, *Econometrica* **24**: 1317–1399.

Geyer, C. J. and Thompson, E. A. (1992). Constrained Monte Carlo maximum likelihood estimation for dependent data (with discussion), *Journal of the Royal Statistical Society* **B 54**: 657–700.

Ghahramani, Z. and Beal, M. J. (2000). Variational inference for Bayesian mixtures of factor analysers, *in* S. Solla, T. Leen and K.-R. Müller (eds), *Advances in Neural Information Processing Systems 12*, MIT Press, pp. 449–455.

Gilks, W. R., Richardson, S. and Spiegelhalter, D. J. (eds) (1996). *Markov Chain Monte Carlo in Practice*, Chapman and Hall, Suffolk.

Gilks, W. R., Thomas, A. and Spiegelhalter, D. J. (1994). A language and program for complex Bayesian modelling, *The Statistician* **43**: 169–178.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their Applications, *Biometrika* **57**: 97–109.

Jaakkola, T. and Jordan, M. I. (1999). Variational methods and the QMR-DT database, *Journal of Artificial Intelligence* **10**: 291–322.

Jaakkola, T. and Jordan, M. I. (2000). Bayesian parameter estimation via variational methods, *Statistics and Computing* **10**: 25–37.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S. and Saul, L. K. (1999). An introduction to variational methods for graphical models, *Machine Learning* **37**: 183–233.

Mengersen, K. L. and Tweedie, R. L. (1996). Rates of convergence of the Hastings and Metropolis algorithms, *The Annals of Statistics* **24**: 101–121.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E. (1953). Equations of state calculations by fast computing machines, *Journal of Chemical Physics* **21**: 1087–1091.

Meyn, S. P. and Tweedie, R. L. (1993). *Markov Chains and Stochastic Stability*, Springer-Verlag, New York.

Robert, C. P. and Casella, G. (1999). *Monte Carlo Statistical Methods*, Springer-Verlag, New York.

Roberts, G. and Tweedie, R. (1996). Geometric convergence and central limit theorems for multidimensional Hastings and Metropolis algorithms, *Biometrika* **83**: 95–110.

Tierney, L. (1994). Markov chains for exploring posterior distributions, *The Annals of Statistics* **22**(4): 1701–1762.

Zhang, J. (1993). The application of the Gibbs-Bogoliubov-Feynman inequality in mean field calculations for Markov random fields, *IEEE Transactions on Image Processing* **5**(7): 1208–1214.