Chapter 6

Learning the structure of discrete-variable graphical models with hidden variables

6.1 Introduction

One of the key problems in machine learning and statistics is how to learn the structure of graphical models from data. This entails determining the dependency relations amongst the model variables that are supported by the data. Models of differing complexities can be rated according to their posterior probabilities, which by Bayes' rule are related to the marginal likelihood under each candidate model.

In the case of fully observed discrete-variable directed acyclic graphs with Dirichlet priors on the parameters it is tractable to compute the marginal likelihood of a candidate structure and therefore obtain its posterior probability (or a quantity proportional to this). Unfortunately, in graphical models containing hidden variables the calculation of the marginal likelihood is generally intractable for even moderately sized data sets, and its estimation presents a difficult challenge for approximate methods such as asymptotic-data criteria and sampling techniques.

In this chapter we investigate a novel application of the VB framework to approximating the marginal likelihood of discrete-variable directed acyclic graph (DAG) structures that contain hidden variables. We call approximations to a model's marginal likelihood *scores*. We first derive the VB score, which is simply the result of a VBEM algorithm applied to DAGs, and then assess its performance on a model selection task: finding the particular structure (out of a small class of structures) that gave rise to the observed data. We also derive and evaluate the BIC and Cheeseman-Stutz (CS) scores and compare these to VB for this problem.

We also compare the BIC, CS, and VB scoring techniques to annealed importance sampling (AIS) estimates of the marginal likelihood. We consider AIS to be a "gold standard", the best method for obtaining reliable estimates of the marginal likelihoods of models explored in this chapter (personal communication with C. Rasmussen, Z. Ghahramani, and R. Neal). We have used AIS in this chapter to perform the first serious case study of the tightness of variational bounds. An analysis of the limitations of AIS is also provided. The aim of the comparison is to convince us of the reliability of VB as an estimate of the marginal likelihood in the general incomplete-data setting, so that it can be used in larger problems, for example embedded in a (greedy) structure search amongst a much larger class of models.

In section 6.2 we begin by examining the model selection question for discrete directed acyclic graphs, and show how exact marginal likelihood calculation rapidly becomes computationally intractable when the graph contains hidden variables. In section 6.3 we briefly cover the EM algorithm for ML and MAP parameter estimation in DAGs with hidden variables, and discuss the BIC, Laplace and Cheeseman-Stutz asymptotic approximations. We then present the VBEM algorithm for variational Bayesian lower bound optimisation, which in the case of discrete DAGs is a straightforward generalisation of the MAP EM algorithm. In section 6.3.5 we describe in detail an annealed importance sampling method for estimating marginal likelihoods of discrete DAGs. In section 6.4 we evaluate the performance of these different scoring methods on the simple (yet non-trivial) model selection task of determining which of all possible structures within a class generated a data set. Section 6.5 discusses some related topics which expand on the methods used in this chapter: first, we give an analysis of the limitations of the AIS implementation and suggest possible extensions for it; second, we more thoroughly consider the parameter-counting arguments used in the BIC and CS scoring methods, and reformulate a more successful score. Finally we conclude in section 6.6 and suggest directions for future research.

6.2 Calculating marginal likelihoods of DAGs

Consider a data set of size n, $\mathbf{y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$, modelled by the discrete directed acyclic graph consisting of hidden and observed variables $\mathbf{z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_n\} = \{\mathbf{s}_1, \mathbf{y}_1, \ldots, \mathbf{s}_n, \mathbf{y}_n\}$. The variables in each plate $i = 1, \ldots, n$ are indexed by $j = 1, \ldots, |\mathbf{z}_i|$, of which some $j \in \mathcal{H}$ are hidden and $j \in \mathcal{V}$ are observed variables, i.e. $\mathbf{s}_i = \{\mathbf{z}_{ij}\}_{j \in \mathcal{H}}$ and $\mathbf{y}_i = \{\mathbf{z}_{ij}\}_{j \in \mathcal{V}}$.

On a point of nomenclature, note that $\mathbf{z}_i = \{\mathbf{s}_i, \mathbf{y}_i\}$ contains both hidden and observed variables, and we interchange freely between these two forms where convenient. Moreover, the numbers of hidden and observed variables, $|\mathbf{s}_i|$ and $|\mathbf{y}_i|$, are allowed to vary with the data point index i. An example of such a case could be a data set of sequences of varying length, to be modelled by an HMM. Note also that the meaning of $|\cdot|$ varies depending on the type of its argument, for

example: $|\mathbf{z}|$ is the number of data points, n; $|\mathbf{s}_i|$ is the number of hidden variables (for the ith data point); $|\mathbf{s}_{ij}|$ is the cardinality (number of settings) of the jth hidden variable (for the ith data point).

In a DAG the complete-data likelihood factorises into a product of local probabilities on each variable

$$p(\mathbf{z} \,|\, \boldsymbol{\theta}) = \prod_{i=1}^{n} \prod_{j=1}^{|\mathbf{z}_i|} p(\mathbf{z}_{ij} \,|\, \mathbf{z}_{i\mathbf{pa}(j)}, \boldsymbol{\theta}) \;, \tag{6.1}$$

where $\mathbf{pa}(j)$ denotes the vector of indices of the parents of the jth variable. Each variable in the model is multinomial, and the parameters of the model are different vectors of probabilities on each variable for each configuration of its parents. For example, the parameter for a binary variable which has two ternary parents is a $3^2 \times 2$ matrix with each row summing to one. Should there be a variable j without any parents ($\mathbf{pa}(j) = \emptyset$), then the parameter associated with variable j is simply a vector of its prior probabilities. If we use θ_{jlk} to denote the probability that variable j takes on value k when its parents are in configuration l, then the complete likelihood can be written out as a product of terms of the form

$$p(\mathbf{z}_{ij} \,|\, \mathbf{z}_{i\mathbf{pa}(j)}, \boldsymbol{\theta}) = \prod_{l=1}^{|\mathbf{z}_{i\mathbf{pa}(j)}|} \prod_{k=1}^{|\mathbf{z}_{ij}|} \theta_{jlk}^{\delta(\mathbf{z}_{ij},k)\delta(\mathbf{z}_{i\mathbf{pa}(j)},l)} \tag{6.2}$$

$$\text{with} \quad \sum_k \theta_{jlk} = 1 \quad \forall \, \{j, l\} \;. \tag{6.3}$$

Here we use $\left|\mathbf{z}_{i\mathbf{pa}(j)}\right|$ to denote the number of joint settings of the parents of variable j. That is to say the probability is a product over both all the $\left|\mathbf{z}_{i\mathbf{pa}(j)}\right|$ possible settings of the parents and the $|\mathbf{z}_{ij}|$ settings of the variable itself. Here we use Kronecker-δ notation which is 1 if its arguments are identical and zero otherwise. The parameters of the model are given independent Dirichlet priors, which are conjugate to the complete-data likelihood above (see equation (2.80), which is Condition 1 for conjugate-exponential models). By independent we mean factorised over variables and parent configurations; these choices then satisfy the *global* and *local* independence assumptions of Heckerman et al. (1995). For each parameter $\boldsymbol{\theta}_{jl} = \{\theta_{jl1}, \ldots, \theta_{jl|\mathbf{z}_{ij}|}\}$, the Dirichlet prior is

$$p(\boldsymbol{\theta}_{jl} \,|\, \boldsymbol{\lambda}_{jl}, m) = \frac{\Gamma(\lambda_{jl}^0)}{\prod_k \Gamma(\lambda_{jlk})} \prod_k \theta_{jlk}^{\lambda_{jlk}-1} \;, \tag{6.4}$$

where $\boldsymbol{\lambda}$ are hyperparameters:

$$\boldsymbol{\lambda}_{jl} = \{\lambda_{jl1}, \ldots, \lambda_{jl|\mathbf{z}_{ij}|}\} \tag{6.5}$$

and

$$\lambda_{jlk} > 0 \quad \forall \, k \;, \qquad \lambda_{jl}^0 = \sum_k \lambda_{jlk} \;. \tag{6.6}$$

This form of prior is assumed throughout the chapter. Since the focus of this chapter is not on optimising these hyperparameters, we use the shorthand $p(\boldsymbol{\theta} \,|\, m)$ to denote the prior from here on. In the discrete-variable case we are considering, the complete-data marginal likelihood is tractable to compute:

$$p(\mathbf{z} \,|\, m) = \int d\boldsymbol{\theta}\; p(\boldsymbol{\theta} \,|\, m) p(\mathbf{z} \,|\, \boldsymbol{\theta}) \tag{6.7}$$

$$= \int d\boldsymbol{\theta}\; p(\boldsymbol{\theta} \,|\, m) \prod_{i=1}^{n} \prod_{j=1}^{|\mathbf{z}_i|} p(\mathbf{z}_{ij} \,|\, \mathbf{z}_{i\mathbf{pa}(j)}, \boldsymbol{\theta}) \tag{6.8}$$

$$= \prod_{j=1}^{|\mathbf{z}_i|} \prod_{l=1}^{|\mathbf{z}_{i\mathbf{pa}(j)}|} \frac{\Gamma(\lambda^0_{jl})}{\Gamma(\lambda^0_{jl} + N_{jl})} \prod_{k=1}^{|\mathbf{z}_{ij}|} \frac{\Gamma(\lambda_{jlk} + N_{jlk})}{\Gamma(\lambda_{jlk})} \tag{6.9}$$

where N_{jlk} is defined as the count in the data for the number of instances of variable j being in configuration k with parental configuration l:

$$N_{jlk} = \sum_{i=1}^{n} \delta(\mathbf{z}_{ij}, k) \delta(\mathbf{z}_{i\mathbf{pa}(j)}, l), \qquad \text{and} \quad N_{jl} = \sum_{k=1}^{|\mathbf{z}_{ij}|} N_{jlk} \,. \tag{6.10}$$

The incomplete-data likelihood, however, is not as tractable. It results from summing over all settings of the hidden variables and taking the product over i.i.d. presentations of the data:

$$p(\mathbf{y} \,|\, \boldsymbol{\theta}) = \prod_{i=1}^{n} p(\mathbf{y}_i \,|\, \boldsymbol{\theta}) = \prod_{i=1}^{n} \sum_{\{\mathbf{z}_{ij}\}_{j \in \mathcal{H}}} \prod_{j=1}^{|\mathbf{z}_i|} p(\mathbf{z}_{ij} \,|\, \mathbf{z}_{i\mathbf{pa}(j)}, \boldsymbol{\theta}) \,. \tag{6.11}$$

This quantity can be evaluated as the product of n quantities, each of which is a summation over all possible joint configurations of the hidden variables; in the worst case this computation requires $\mathcal{O}(n \prod_{j \in \mathcal{H}} |\mathbf{z}_{ij}|)$ operations (although this can usually be made more efficient with the use of propagation algorithms that exploit the topology of the model). The incomplete-data marginal likelihood for n cases follows from marginalising out the parameters of the model:

$$p(\mathbf{y} \,|\, m) = \int d\boldsymbol{\theta}\; p(\boldsymbol{\theta} \,|\, m) \prod_{i=1}^{n} \sum_{\{\mathbf{z}_{ij}\}_{j \in \mathcal{H}}} \prod_{j=1}^{|\mathbf{z}_i|} p(\mathbf{z}_{ij} \,|\, \mathbf{z}_{i\mathbf{pa}(j)}, \boldsymbol{\theta}) \,. \tag{6.12}$$

This expression is computationally intractable due to the expectation over the real-valued conditional probabilities $\boldsymbol{\theta}$, which couples the hidden variables across i.i.d. data. In the worst case it can be evaluated as the sum of $\left(\prod_{j \in \mathcal{H}} |\mathbf{z}_{ij}|\right)^n$ Dirichlet integrals. For example, a model with just $|\mathbf{s}_i| = 2$ hidden variables and 100 data points requires the evaluation of 2^{100} Dirichlet integrals. This means that a linear increase in the amount of observed data results in an exponential increase in the cost of inference.

We focus on the task of learning the conditional independence structure of the model, that is, which variables are parents of each variable. We compare structures based on their posterior probabilities. In this chapter we assume that the prior, $p(m)$, is uninformative, and so all our information comes from the intractable marginal likelihood, $p(\mathbf{y} \mid m)$.

In the rest of this chapter we examine several methods to approximate this Bayesian integration (6.12), in order to make learning and inference tractable. For the moment we assume that the cardinalities of the variables, in particular the hidden variables, are fixed beforehand. The related problem of determining the cardinality of the variables from data can be addressed in the same framework, as we have already seen for HMMs in chapter 3.

6.3 Estimating the marginal likelihood

In this section we look at some approximations to the marginal likelihood, which we refer to henceforth as *scores*. We first review ML and MAP parameter learning and briefly present the EM algorithm for a general discrete-variable directed graphical model with hidden variables. From the result of the EM optimisation, we can construct various asymptotic approximations to the marginal likelihood, deriving the BIC and Cheeseman-Stutz scores. We then apply the variational Bayesian framework, which in the case of conjugate-exponential discrete directed acyclic graphs produces a very simple VBEM algorithm, which is a direct extension of the EM algorithm for MAP parameter learning. Finally, we derive an *annealed importance sampling* method (AIS) for this class of graphical model, which is considered to be the current state-of-the-art technique for estimating the marginal likelihood of these models using sampling — we then compare the different scoring methods to it. We finish this section with a brief note on some trivial and non-trivial upper bounds to the marginal likelihood.

6.3.1 ML and MAP parameter estimation

The EM algorithm for ML/MAP estimation can be derived using the lower bound interpretation as was described in section 2.2. We begin with the incomplete-data log likelihood, and lower bound it by a functional $\mathcal{F}(q_{\mathbf{s}}(\mathbf{s}), \boldsymbol{\theta})$ as follows

$$\ln p(\mathbf{y} \mid \boldsymbol{\theta}) = \ln \prod_{i=1}^{n} \sum_{\{\mathbf{z}_{ij}\}_{j \in \mathcal{H}}} \prod_{j=1}^{|\mathbf{z}_i|} p(\mathbf{z}_{ij} \mid \mathbf{z}_{i\mathbf{pa}(j)}, \boldsymbol{\theta}) \tag{6.13}$$

$$\geq \sum_{i=1}^{n} \sum_{\mathbf{s}_i} q_{\mathbf{s}_i}(\mathbf{s}_i) \ln \frac{\prod_{j=1}^{|\mathbf{z}_i|} p(\mathbf{z}_{ij} \mid \mathbf{z}_{i\mathbf{pa}(j)}, \boldsymbol{\theta})}{q_{\mathbf{s}_i}(\mathbf{s}_i)} \tag{6.14}$$

$$= \mathcal{F}(\{q_{\mathbf{s}_i}(\mathbf{s}_i)\}_{i=1}^{n}, \boldsymbol{\theta}) \ , \tag{6.15}$$

where we have introduced a distribution $q_{\mathbf{s}_i}(\mathbf{s}_i)$ over the hidden variables $\mathbf{s}_i$ for each data point $\mathbf{y}_i$. We remind the reader that we have used $\mathbf{s}_i = \{\mathbf{z}_{ij}\}_{j\in\mathcal{H}}$ in going from (6.13) to (6.14). On taking derivatives of $\mathcal{F}(\{q_{\mathbf{s}_i}(\mathbf{s}_i)\}_{i=1}^n, \boldsymbol{\theta})$ with respect to $q_{\mathbf{s}_i}(\mathbf{s}_i)$, the optimal setting of the variational posterior is given exactly by the posterior

$$q_{\mathbf{s}_i}(\mathbf{s}_i) = p(\mathbf{s}_i \,|\, \mathbf{y}_i, \boldsymbol{\theta}) \quad \forall\, i\ . \tag{6.16}$$

This is the E step of the EM algorithm; at this setting of the distribution $q_{\mathbf{s}_i}(\mathbf{s}_i)$ it can be easily shown that the bound (6.14) is tight (see section 2.2.2).

The M step of the algorithm is derived by taking derivatives of the bound with respect to the parameters $\boldsymbol{\theta}$. Each $\boldsymbol{\theta}_{jl}$ is constrained to sum to one, and so we enforce this with Lagrange multipliers c_{jl},

$$\frac{\partial}{\partial\theta_{jlk}}\mathcal{F}(q_{\mathbf{s}}(\mathbf{s}), \boldsymbol{\theta}) = \sum_{i=1}^{n}\sum_{\mathbf{s}_i} q_{\mathbf{s}_i}(\mathbf{s}_i)\frac{\partial}{\partial\theta_{jlk}}\ln p(\mathbf{z}_{ij} \,|\, \mathbf{x}_{i\mathbf{pa}(j)}, \boldsymbol{\theta}_j) + c_{jl} \tag{6.17}$$

$$= \sum_{i=1}^{n}\sum_{\mathbf{s}_i} q_{\mathbf{s}_i}(\mathbf{s}_i)\delta(\mathbf{z}_{ij}, k)\delta(\mathbf{z}_{i\mathbf{pa}(j)}, l)\frac{\partial}{\partial\theta_{jlk}}\ln\theta_{jlk} + c_{jl} \tag{6.18}$$

$$= 0\ , \tag{6.19}$$

which upon rearrangement gives

$$\theta_{jlk} \propto \sum_{i=1}^{n}\sum_{\mathbf{s}_i} q_{\mathbf{s}_i}(\mathbf{s}_i)\delta(\mathbf{z}_{ij}, k)\delta(\mathbf{z}_{i\mathbf{pa}(j)}, l)\ . \tag{6.20}$$

Due to the normalisation constraint on $\boldsymbol{\theta}_{jl}$ the M step can be written

$$\textbf{M step (ML):} \qquad \theta_{jlk} = \frac{N_{jlk}}{\sum_{k'=1}^{|\mathbf{z}_{ij}|} N_{jlk'}}\ , \tag{6.21}$$

where the N_{jlk} are defined as

$$N_{jlk} = \sum_{i=1}^{n}\left\langle\delta(\mathbf{z}_{ij}, k)\delta(\mathbf{z}_{i\mathbf{pa}(j)}, l)\right\rangle_{q_{\mathbf{s}_i}(\mathbf{s}_i)} \tag{6.22}$$

where angled-brackets $\langle\cdot\rangle_{q_{\mathbf{s}_i}(\mathbf{s}_i)}$ are used to denote expectation with respect to the hidden variable posterior $q_{\mathbf{s}_i}(\mathbf{s}_i)$. The N_{jlk} are interpreted as the expected number of counts for observing simultaneous settings of children and parent configurations over observed and hidden variables. In the cases where both j and $\mathbf{pa}(j)$ are observed variables, N_{jlk} reduces to the simple empirical count as in (6.10). Otherwise if j or its parents are hidden then expectations need be taken over the posterior $q_{\mathbf{s}_i}(\mathbf{s}_i)$ obtained in the E step.

If we require the MAP EM algorithm, we instead lower bound $\ln p(\boldsymbol{\theta})p(\mathbf{y} \mid \boldsymbol{\theta})$. The E step remains the same, but the M step uses augmented counts from the prior of the form in (6.4) to give the following update:

$$\textbf{M step (MAP):} \qquad \theta_{jlk} = \frac{\lambda_{jlk} - 1 + N_{jlk}}{\sum_{k'=1}^{|\mathbf{z}_{ij}|} \lambda_{jlk'} - 1 + N_{jlk'}} \,. \tag{6.23}$$

Repeated applications of the E step (6.16) and the M step (6.21, 6.23) are guaranteed to increase the log likelihood (with equation (6.21)) or the log posterior (with equation (6.23)) of the parameters at every iteration, and converge to a local maximum.

As mentioned in section 1.3.1, we note that MAP estimation is basis-dependent. For any particular $\boldsymbol{\theta}^*$, which has non-zero prior probability, it is possible to find a (one-to-one) reparameterisation $\boldsymbol{\phi}(\boldsymbol{\theta})$ such that the MAP estimate for $\boldsymbol{\phi}$ is at $\boldsymbol{\phi}(\boldsymbol{\theta}^*)$. This is an obvious drawback of MAP parameter estimation. Moreover, the use of (6.23) can produce erroneous results in the case of $\lambda_{jlk} < 1$, in the form of negative probabilities. Conventionally, researchers have limited themselves to Dirichlet priors in which every $\lambda_{jlk} \geq 1$, although in MacKay (1998) it is shown how a reparameterisation of $\boldsymbol{\theta}$ into the softmax basis results in MAP updates which do not suffer from this problem (which look identical to (6.23), but without the -1 in numerator and denominator).

6.3.2 BIC

The Bayesian Information Criterion approximation, described in section 1.3.4, is the asymptotic limit to large data sets of the Laplace approximation. It is interesting because it does not depend on the prior over parameters, and attractive because it does not involve the burdensome computation of the Hessian of the log likelihood and its determinant. For the size of structures considered in this chapter, the Laplace approximation would be viable to compute, subject perhaps to a transformation of parameters (see for example MacKay, 1995). However in larger models the approximation may become unwieldy and further approximations would be required (see section 1.3.2).

For BIC, we require the number of free parameters in each structure. In the experiments in this chapter we use a simple counting argument; in section 6.5.2 we discuss a more rigorous method for estimating the dimensionality of the parameter space of a model. We apply the following counting scheme. If a variable j has no parents in the DAG, then it contributes $(|\mathbf{z}_{ij}| - 1)$ free parameters, corresponding to the degrees of freedom in its vector of prior probabilities (constrained to lie on the simplex $\sum_k p_k = 1$). Each variable that has parents contributes

$(|\mathbf{z}_{ij}| - 1)$ parameters for each configuration of its parents. Thus in model m the total number of parameters $d(m)$ is given by

$$d(m) = \sum_{j=1}^{|\mathbf{z}_i|} (|\mathbf{z}_{ij}| - 1) \prod_{l=1}^{|\mathbf{z}_{i\mathbf{pa}(j)}|} |\mathbf{z}_{i\mathbf{pa}(j)l}| \ , \tag{6.24}$$

where $|\mathbf{z}_{i\mathbf{pa}(j)l}|$ denotes the cardinality (number of settings) of the lth parent of the jth variable. We have used the convention that the product over zero factors has a value of one to account for the case in which the jth variable has no parents, i.e.:

$$\prod_{l=1}^{|\mathbf{z}_{i\mathbf{pa}(j)}|} |\mathbf{z}_{i\mathbf{pa}(j)l}| = 1 \ , \quad \text{if} \quad |\mathbf{z}_{i\mathbf{pa}(j)l}| = 0 \ . \tag{6.25}$$

The BIC approximation needs to take into account aliasing in the parameter posterior (as described in section 1.3.3). In discrete-variable DAGs, parameter aliasing occurs from two symmetries: first, a priori identical hidden variables can be permuted; and second, the labellings of the states of each hidden variable can be permuted. As an example, let us imagine the parents of a single observed variable are 3 hidden variables having cardinalities $(3, 3, 4)$. In this case the number of aliases is 1728 ($= 2! \times 3! \times 3! \times 4!$). If we assume that the aliases of the posterior distribution are well separated then the score is given by

$$\ln p(\mathbf{y} \,|\, m)_{\text{BIC}} = \ln p(\mathbf{y} \,|\, \hat{\boldsymbol{\theta}}) - \frac{d(m)}{2} \ln n + \ln S \tag{6.26}$$

where S is the number of aliases, and $\hat{\boldsymbol{\theta}}$ is the MAP estimate as described in the previous section. This correction is accurate only if the modes of the posterior distribution are well separated, which should be the case in the large data set size limit for which BIC is useful. However, since BIC is correct only up to an indeterminant missing factor, we might think that this correction is not necessary. In the experiments we examine the BIC score with and without this correction, and also with and without the prior term included.

6.3.3 Cheeseman-Stutz

The Cheeseman-Stutz approximation uses the following identity for the incomplete-data marginal likelihood:

$$p(\mathbf{y} \,|\, m) = p(\mathbf{z} \,|\, m) \frac{p(\mathbf{y} \,|\, m)}{p(\mathbf{z} \,|\, m)} = p(\mathbf{z} \,|\, m) \frac{\int d\boldsymbol{\theta} \ p(\boldsymbol{\theta} \,|\, m) p(\mathbf{y} \,|\, \boldsymbol{\theta}, m)}{\int d\boldsymbol{\theta} \ p(\boldsymbol{\theta}' \,|\, m) p(\mathbf{z} \,|\, \boldsymbol{\theta}', m)} \tag{6.27}$$

which is true for any completion $\mathbf{z} = \{\hat{\mathbf{s}}, \mathbf{y}\}$ of the data. This form is useful because the complete-data marginal likelihood, $p(\mathbf{z} \,|\, m)$, is tractable to compute for discrete DAGs with

independent Dirichlet priors: it is just a product of Dirichlet integrals (see equation (6.9)). Using the results of section 1.3.2, in particular equation (1.45), we can apply Laplace approximations to both the numerator and denominator of the above fraction to give

$$p(\mathbf{y} \,|\, m) \approx p(\hat{\mathbf{s}}, \mathbf{y} \,|\, m) \frac{p(\hat{\boldsymbol{\theta}} \,|\, m) p(\mathbf{y} \,|\, \hat{\boldsymbol{\theta}}) \, |2\pi A|^{-1}}{p(\hat{\boldsymbol{\theta}}' \,|\, m) p(\hat{\mathbf{s}}, \mathbf{y} \,|\, \hat{\boldsymbol{\theta}}') \, |2\pi A'|^{-1}} \; . \tag{6.28}$$

We assume that $p(\mathbf{y} \,|\, \hat{\boldsymbol{\theta}})$ is computable exactly. If the errors in each of the Laplace approximations are similar, then they should roughly cancel each other out; this will be the case if the shape of the posterior distributions about $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\theta}}'$ are similar. We can ensure that $\hat{\boldsymbol{\theta}}' = \hat{\boldsymbol{\theta}}$ by completing the hidden data $\{\mathbf{s}_i\}_{i=1}^n$ with their expectations under their posterior distributions $p(\mathbf{s}_i \,|\, \mathbf{y}, \hat{\boldsymbol{\theta}})$. That is to say the hidden states are completed as follows:

$$\hat{\mathbf{s}}_{ijk} = \langle \delta(\mathbf{s}_{ij}, k) \rangle_{q_{\mathbf{s}_i}(\mathbf{s}_i)} \; , \tag{6.29}$$

which will generally result in non-integer counts N_{jlk} on application of (6.22). Having computed these counts and re-estimated $\hat{\boldsymbol{\theta}}'$ using equation (6.23), we note that $\hat{\boldsymbol{\theta}}' = \hat{\boldsymbol{\theta}}$. The Cheeseman-Stutz approximation then results from taking the BIC-type asymptotic limit of both Laplace approximations in (6.28),

$$\begin{aligned} \ln p(\mathbf{y} \,|\, m)_{\text{CS}} &= \ln p(\hat{\mathbf{s}}, \mathbf{y} \,|\, m) + \ln p(\hat{\boldsymbol{\theta}} \,|\, m) + \ln p(\mathbf{y} \,|\, \hat{\boldsymbol{\theta}}) - \frac{d}{2} \ln n \\ &\quad - \ln p(\hat{\boldsymbol{\theta}}' \,|\, m) - \ln p(\hat{\mathbf{s}}, \mathbf{y} \,|\, \hat{\boldsymbol{\theta}}) + \frac{d'}{2} \ln n \qquad (6.30) \\ &= \ln p(\hat{\mathbf{s}}, \mathbf{y} \,|\, m) + \ln p(\mathbf{y} \,|\, \hat{\boldsymbol{\theta}}) - \ln p(\hat{\mathbf{s}}, \mathbf{y} \,|\, \hat{\boldsymbol{\theta}}) \; , \qquad (6.31) \end{aligned}$$

where the last line follows from the modes of the Gaussian approximations being at the same point, $\hat{\boldsymbol{\theta}}' = \hat{\boldsymbol{\theta}}$, and also the assumption that the number of parameters in the models for complete and incomplete data are the same, i.e. $d = d'$ (Cheeseman and Stutz, 1996, but also see section 6.5.2). Each term of (6.31) can be evaluated individually:

$$\text{from (6.9)} \qquad p(\hat{\mathbf{s}}, \mathbf{y} \,|\, m) = \prod_{j=1}^{|\mathbf{z}_i|} \prod_{l=1}^{|\mathbf{z}_{i\mathbf{pa}(j)}|} \frac{\Gamma(\lambda_{jl}^0)}{\Gamma(\lambda_{jl} + \hat{N}_{jl})} \prod_{k=1}^{|\mathbf{z}_{ij}|} \frac{\Gamma(\lambda_{jlk} + \hat{N}_{jlk})}{\Gamma(\lambda_{jlk})} \tag{6.32}$$

$$\text{from (6.11)} \qquad p(\mathbf{y} \,|\, \hat{\boldsymbol{\theta}}) = \prod_{i=1}^{n} \sum_{\{\mathbf{z}_{ij}\}_{j \in \mathcal{H}}} \prod_{j=1}^{|\mathbf{z}_i|} \prod_{l=1}^{|\mathbf{z}_{i\mathbf{pa}(j)}|} \prod_{k=1}^{|\mathbf{z}_{ij}|} \hat{\theta}_{jlk}^{\delta(\mathbf{z}_{ij},k)\delta(\mathbf{z}_{i\mathbf{pa}(j)},l)} \tag{6.33}$$

$$\text{from (6.1)} \qquad p(\hat{\mathbf{s}}, \mathbf{y} \,|\, \hat{\boldsymbol{\theta}}) = \prod_{j=1}^{|\mathbf{z}_i|} \prod_{l=1}^{|\mathbf{z}_{i\mathbf{pa}(j)}|} \prod_{k=1}^{|\mathbf{z}_{ij}|} \hat{\theta}_{jlk}^{\hat{N}_{jlk}} \tag{6.34}$$

where the $\hat{N}_{jlk}$ are identical to the N_{jlk} of equation (6.22) if the completion of the data with $\hat{\mathbf{s}}$ is done with the posterior found in the M step of the MAP EM algorithm used to find $\hat{\boldsymbol{\theta}}$. Equation

(6.33) is simply the output of the EM algorithm, equation (6.32) is a function of the counts obtained in the EM algorithm, and equation (6.34) is a simple computation again.

As with BIC, the Cheeseman-Stutz score also needs to be corrected for aliases in the parameter posterior, as described above, and is subject to the same caveat that these corrections are only accurate if the aliases in the posterior are well-separated.

We note that CS is a lower bound on the marginal likelihood, as shown in section 2.6.2 of this thesis. We will return to this point in the discussion of the experimental results.

6.3.4 The VB lower bound

The incomplete-data log marginal likelihood can be written as

$$\ln p(\mathbf{y} \,|\, m) = \ln \int d\boldsymbol{\theta} \; p(\boldsymbol{\theta} \,|\, m) \prod_{i=1}^{n} \sum_{\{\mathbf{z}_{ij}\}_{j \in \mathcal{H}}} \prod_{j=1}^{|\mathbf{z}_i|} p(\mathbf{z}_{ij} \,|\, \mathbf{z}_{i\mathbf{pa}(j)}, \boldsymbol{\theta}) \; . \tag{6.35}$$

We can form the lower bound in the usual fashion using $q_{\boldsymbol{\theta}}(\boldsymbol{\theta})$ and $\{q_{\mathbf{s}_i}(\mathbf{s}_i)\}_{i=1}^{n}$ to yield (see section 2.3.1):

$$\begin{aligned} \ln p(\mathbf{y} \,|\, m) &\geq \int d\boldsymbol{\theta} \; q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \ln \frac{p(\boldsymbol{\theta} \,|\, m)}{q_{\boldsymbol{\theta}}(\boldsymbol{\theta})} \\ &\quad + \sum_{i=1}^{n} \int d\boldsymbol{\theta} q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \sum_{\mathbf{s}_i} q_{\mathbf{s}_i}(\mathbf{s}_i) \ln \frac{p(\mathbf{z}_i \,|\, \boldsymbol{\theta}, m)}{q_{\mathbf{s}_i}(\mathbf{s}_i)} \qquad (6.36) \\ &= \mathcal{F}_m(q_{\boldsymbol{\theta}}(\boldsymbol{\theta}), q(\mathbf{s})) \; . \qquad (6.37) \end{aligned}$$

We now take functional derivatives to write down the variational Bayesian EM algorithm (theorem 2.1, page 54). The VBM step is straightforward:

$$\ln q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \ln p(\boldsymbol{\theta} \,|\, m) + \sum_{i=1}^{n} \sum_{\mathbf{s}_i} q_{\mathbf{s}_i}(\mathbf{s}_i) \ln p(\mathbf{z}_i \,|\, \boldsymbol{\theta}, m) + \mathrm{c} \; , \tag{6.38}$$

with c a constant. Given that the prior over parameters factorises over variables as in (6.4), and the complete-data likelihood factorises over the variables in a DAG as in (6.1), equation (6.38) can be broken down into individual derivatives:

$$\ln q_{\boldsymbol{\theta}_{jl}}(\boldsymbol{\theta}_{jl}) = \ln p(\boldsymbol{\theta}_{jl} \,|\, \boldsymbol{\lambda}_{jl}, m) + \sum_{i=1}^{n} \sum_{\mathbf{s}_i} q_{\mathbf{s}_i}(\mathbf{s}_i) \ln p(\mathbf{z}_{ij} \,|\, \mathbf{z}_{i\mathbf{pa}(j)}, \boldsymbol{\theta}, m) + c_{jl} \; , \tag{6.39}$$

where $\mathbf{z}_{ij}$ may be either a hidden or observed variable, and each c_{jl} is a Lagrange multiplier from which a normalisation constant is obtained. Equation (6.39) has the form of the Dirichlet distribution. We define the expected counts under the posterior hidden variable distribution

$$N_{jlk} = \sum_{i=1}^{n} \left\langle \delta(\mathbf{z}_{ij}, k)\delta(\mathbf{z}_{i\mathbf{pa}(j)}, l) \right\rangle_{q_{\mathbf{s}_i}(\mathbf{s}_i)} \ . \tag{6.40}$$

Therefore N_{jlk} is the expected total number of times the jth variable (hidden or observed) is in state k when its parents (hidden or observed) are in state l, where the expectation is taken with respect to the posterior distribution over the hidden variables for each datum. Then the variational posterior for the parameters is given simply by (see theorem 2.2)

$$q_{\boldsymbol{\theta}_{jl}}(\boldsymbol{\theta}_{jl}) = \mathrm{Dir}\left(\lambda_{jlk} + N_{jlk} \ : \ k = 1, \ldots, |\mathbf{z}_{ij}|\right) \ . \tag{6.41}$$

For the VBE step, taking derivatives of (6.37) with respect to each $q_{\mathbf{s}_i}(\mathbf{s}_i)$ yields

$$\ln q_{\mathbf{s}_i}(\mathbf{s}_i) = \int d\boldsymbol{\theta} \ q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \ln p(\mathbf{z}_i \,|\, \boldsymbol{\theta}, m) + c_i' = \int d\boldsymbol{\theta} \ q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \ln p(\mathbf{s}_i, \mathbf{y}_i \,|\, \boldsymbol{\theta}, m) + c_i' \ , \tag{6.42}$$

where each c_i' is a Lagrange multiplier for normalisation of the posterior. Since the complete-data likelihood $p(\mathbf{z}_i \,|\, \boldsymbol{\theta}, m)$ is in the exponential family and we have placed conjugate Dirichlet priors on the parameters, we can immediately utilise the results of corollary 2.2 (page 74) which gives simple forms for the VBE step:

$$q_{\mathbf{s}_i}(\mathbf{s}_i) \propto q_{\mathbf{z}_i}(\mathbf{z}_i) = \prod_{j=1}^{|\mathbf{z}_i|} p(\mathbf{z}_{ij} \,|\, \mathbf{z}_{i\mathbf{pa}(j)}, \tilde{\boldsymbol{\theta}}) \ . \tag{6.43}$$

Thus the approximate posterior over the hidden variables $\mathbf{s}_i$ resulting from a variational Bayesian approximation is identical to that resulting from exact inference in a model with known point parameters $\tilde{\boldsymbol{\theta}}$. Corollary 2.2 also tells us that $\tilde{\boldsymbol{\theta}}$ should be chosen to satisfy $\boldsymbol{\phi}(\tilde{\boldsymbol{\theta}}) = \overline{\boldsymbol{\phi}}$. The natural parameters for this model are the log probabilities $\{\ln \boldsymbol{\theta}_{jlk}\}$, where j specifies which variable, l indexes the possible configurations of its parents, and k the possible settings of the variable. Thus

$$\ln \tilde{\boldsymbol{\theta}}_{jlk} = \boldsymbol{\phi}(\tilde{\boldsymbol{\theta}}_{jlk}) = \overline{\boldsymbol{\phi}}_{jlk} = \int d\boldsymbol{\theta}_{jl} \ q_{\boldsymbol{\theta}_{jl}}(\boldsymbol{\theta}_{jl}) \ln \boldsymbol{\theta}_{jlk} \ . \tag{6.44}$$

Under a Dirichlet distribution, the expectations are given by differences of digamma functions

$$\ln \tilde{\boldsymbol{\theta}}_{jlk} = \psi(\lambda_{jlk} + N_{jlk}) - \psi(\sum_{k=1}^{|\mathbf{z}_{ij}|} \lambda_{jlk} + N_{jlk}) \quad \forall \, \{j, l, k\} \ . \tag{6.45}$$

where the N_{jlk} are defined in (6.40), and the $\psi(\cdot)$ are digamma functions (see appendix C.1). Since this expectation operation takes the geometric mean of the probabilities, the propagation algorithm in the VBE step is now passed sub-normalised probabilities as parameters

$$\sum_{k=1}^{|\mathbf{z}_{ij}|} \tilde{\boldsymbol{\theta}}_{jlk} \leq 1 \qquad \forall \{j, l\} . \tag{6.46}$$

This use of sub-normalised probabilities also occurred in Chapter 3, which is unsurprising given that both models consist of local multinomial conditional probabilities. In that model, the inference algorithm was the forward-backward algorithm (or its VB analogue), and was restricted to the particular topology of a Hidden Markov Model. Our derivation uses belief propagation (section 1.1.2) for any singly-connected discrete DAG.

The expected natural parameters become normalised only if the distribution over parameters is a delta function, in which case this reduces to the MAP inference scenario of section 6.3.1. In fact, if we look at the limit of the digamma function for large arguments (see appendix C.1), we find

$$\lim_{x \to \infty} \psi(x) = \ln x \, , \tag{6.47}$$

and equation (6.45) becomes

$$\lim_{n \to \infty} \ln \tilde{\boldsymbol{\theta}}_{jlk} = \ln(\lambda_{jlk} + N_{jlk}) - \ln(\sum_{k=1}^{|\mathbf{z}_{ij}|} \lambda_{jlk} + N_{jlk}) \tag{6.48}$$

which has recovered the MAP estimator for $\boldsymbol{\theta}$ (6.23), up to the -1 entries in numerator and denominator which become vanishingly small for large data, and vanish completely if MAP is performed in the softmax basis. Thus in the limit of large data VB recovers the MAP parameter estimate.

To summarise, the VBEM implementation for discrete DAGs consists of iterating between the VBE step (6.43) which infers distributions over the hidden variables given a distribution over the parameters, and a VBM step (6.41) which finds a variational posterior distribution over parameters based on the hidden variables' sufficient statistics from the VBE step. Each step monotonically increases a lower bound on the marginal likelihood of the data, and the algorithm is guaranteed to converge to a local maximum of the lower bound.

The VBEM algorithm uses as a subroutine the algorithm used in the E step of the corresponding EM algorithm for MAP estimation, and so the VBE step's computational complexity is the same — there is some overhead in calculating differences of digamma functions instead of ratios of expected counts, but this is presumed to be minimal and fixed.

As with BIC and Cheeseman-Stutz, the lower bound does not take into account aliasing in the parameter posterior, and needs to be corrected as described in section 6.3.2.

6.3.5 Annealed Importance Sampling (AIS)

AIS (Neal, 2001) is a state-of-the-art technique for estimating marginal likelihoods, which breaks a difficult integral into a series of easier ones. It combines techniques from importance sampling, Markov chain Monte Carlo, and simulated annealing (Kirkpatrick et al., 1983). It builds on work in the Physics community for estimating the free energy of systems at different temperatures, for example: thermodynamic integration (Neal, 1993), *tempered transitions* (Neal, 1996), and the similarly inspired *umbrella sampling* (Torrie and Valleau, 1977). Most of these, as well as other related methods, are reviewed in Gelman and Meng (1998).

Obtaining samples from the posterior distribution over parameters, with a view to forming a Monte Carlo estimate of the marginal likelihood of the model, is usually a very challenging problem. This is because, even with small data sets and models with just a few parameters, the distribution is likely to be very peaky and have its mass concentrated in tiny volumes of space. This makes simple approaches such as sampling parameters directly from the prior or using simple importance sampling infeasible. The basic idea behind annealed importance sampling is to move in a *chain* from an easy-to-sample-from distribution, via a series of intermediate distributions, through to the complicated posterior distribution. By annealing the distributions in this way the parameter samples should hopefully come from representative areas of probability mass in the posterior. The key to the annealed importance sampling procedure is to make use of the importance weights gathered at all the distributions up to and including the final posterior distribution, in such a way that the final estimate of the marginal likelihood is unbiased. A brief description of the AIS procedure follows:

We define a series of inverse-temperatures $\{\tau(k)\}_{k=0}^{K}$ satisfying

$$0 = \tau(0) < \tau(1) < \cdots < \tau(K-1) < \tau(K) = 1 \,. \tag{6.49}$$

We refer to temperatures and inverse-temperatures interchangeably throughout this section. We define the function:

$$f_k(\boldsymbol{\theta}) \equiv p(\boldsymbol{\theta} \,|\, m) p(\mathbf{y} \,|\, \boldsymbol{\theta}, m)^{\tau(k)} \,, \qquad k \in \{0, \ldots, K\} \,. \tag{6.50}$$

Thus the set of functions $\{f_k(\boldsymbol{\theta})\}_{k=0}^{K}$ form a series of unnormalised distributions which *interpolate* between the prior and posterior, parameterised by τ. We also define the normalisation constants

$$\mathcal{Z}_k \equiv \int d\boldsymbol{\theta} \; f_k(\boldsymbol{\theta}) = \int d\boldsymbol{\theta} \; p(\boldsymbol{\theta} \,|\, m) p(\mathbf{y} \,|\, \boldsymbol{\theta}, m)^{\tau(k)} \,, \qquad k \in \{0, \ldots, K\} \,. \tag{6.51}$$

We note the following:

$$\mathcal{Z}_0 = \int d\boldsymbol{\theta}\; p(\boldsymbol{\theta} \,|\, m) = 1 \tag{6.52}$$

from normalisation of the prior, and

$$\mathcal{Z}_K = \int d\boldsymbol{\theta}\; p(\boldsymbol{\theta} \,|\, m) p(\mathbf{y} \,|\, \boldsymbol{\theta}, m) = p(\mathbf{y} \,|\, m)\;, \tag{6.53}$$

which is exactly the marginal likelihood that we wish to estimate. We can estimate $\mathcal{Z}_K$, or equivalently $\frac{\mathcal{Z}_K}{\mathcal{Z}_0}$, using the identity

$$p(\mathbf{y} \,|\, m) = \frac{\mathcal{Z}_K}{\mathcal{Z}_0} \equiv \frac{\mathcal{Z}_1}{\mathcal{Z}_0}\,\frac{\mathcal{Z}_2}{\mathcal{Z}_1}\,\cdots\,\frac{\mathcal{Z}_K}{\mathcal{Z}_{K-1}} = \prod_{k=1}^{K} \mathcal{R}_k\;, \tag{6.54}$$

Each of the K ratios in this expression can be individually estimated using importance sampling (see section 1.3.6). The kth ratio, denoted $\mathcal{R}_k$, can be estimated from a set of (not necessarily independent) samples of parameters $\{\boldsymbol{\theta}^{(k,c)}\}_{c\in\mathcal{C}_k}$ which are drawn from the higher temperature $\tau(k-1)$ distribution (the importance distribution), i.e. each $\boldsymbol{\theta}^{(k,c)} \sim f_{k-1}(\boldsymbol{\theta})$, and the importance weights are computed at the lower temperature $\tau(k)$. These samples are used to construct the Monte Carlo estimate for $\mathcal{R}_k$:

$$\mathcal{R}_k \equiv \frac{\mathcal{Z}_k}{\mathcal{Z}_{k-1}} = \int d\boldsymbol{\theta}\; \frac{f_k(\boldsymbol{\theta})}{f_{k-1}(\boldsymbol{\theta})}\frac{f_{k-1}(\boldsymbol{\theta})}{\mathcal{Z}_{k-1}} \tag{6.55}$$

$$\approx \frac{1}{C_k}\sum_{c\in\mathcal{C}_k} \frac{f_k(\boldsymbol{\theta}^{(k,c)})}{f_{k-1}(\boldsymbol{\theta}^{(k,c)})}\;, \qquad \text{with } \boldsymbol{\theta}^{(k,c)} \sim f_{k-1}(\boldsymbol{\theta}) \tag{6.56}$$

$$= \frac{1}{C_k}\sum_{c\in\mathcal{C}_k} p(\mathbf{y} \,|\, \boldsymbol{\theta}^{(k,c)}, m)^{\tau(k)-\tau(k-1)}\;. \tag{6.57}$$

Here, the importance weights are the summands in (6.56). The accuracy of each $\mathcal{R}_k$ depends on the constituent distributions $\{f_k(\boldsymbol{\theta}), f_{k-1}(\boldsymbol{\theta})\}$ being sufficiently close so as to produce low-variance weights. The estimate of $\mathcal{Z}_K$ in (6.54) is unbiased if the samples used to compute each ratio $\mathcal{R}_k$ are drawn from the equilibrium distribution at each temperature $\tau(k)$. In general we expect it to be difficult to sample directly from the forms $f_k(\boldsymbol{\theta})$ in (6.50), and so Metropolis-Hastings (Metropolis et al., 1953; Hastings, 1970) steps are used at each temperature to generate the set of C_k samples required for each importance calculation in (6.57).

Metropolis-Hastings for discrete-variable models

In the discrete-variable graphical models covered in this chapter, the parameters are multinomial probabilities, hence the support of the Metropolis proposal distributions is restricted to the

simplex of probabilities summing to 1. At first thought one might suggest using a Gaussian proposal distribution in the softmax basis of the current parameters $\boldsymbol{\theta}$:

$$\theta_i \equiv \frac{e^{b_i}}{\sum_j^{|\boldsymbol{\theta}|} e^{b_j}} \,. \tag{6.58}$$

Unfortunately an invariance exists: with β a scalar, the transformation $b_i' \leftarrow b_i + \beta \,\forall i$ leaves the parameter $\boldsymbol{\theta}$ unchanged. Therefore the determinant of the Jacobian of the transformation (6.58) from the vector $\mathbf{b}$ to the vector $\boldsymbol{\theta}$ is zero, and it is hard to construct a reversible Markov chain.

A different and intuitively appealing idea is to use a Dirichlet distribution as the proposal distribution, with its mean positioned at the current parameter. The precision of the Dirichlet proposal distribution at inverse-temperature $\tau(k)$ is governed by its *strength*, $\alpha(k)$, which is a free variable to be set as we wish, provided it is not in any way a function of the sampled parameters. A Metropolis-Hastings acceptance function is required to maintain detailed balance: if $\boldsymbol{\theta}'$ is the sample under the proposal distribution centered around the current parameter $\boldsymbol{\theta}^{(k,c)}$, then the acceptance function is:

$$a(\boldsymbol{\theta}', \boldsymbol{\theta}^{(k,c)}) = \min\left(\frac{f_k(\boldsymbol{\theta}')}{f_k(\boldsymbol{\theta}^{(k,c)})} \frac{\mathrm{Dir}(\boldsymbol{\theta}^{(k,c)} \,|\, \boldsymbol{\theta}', \alpha(k))}{\mathrm{Dir}(\boldsymbol{\theta}' \,|\, \boldsymbol{\theta}^{(k,c)}, \alpha(k))} \,,\, 1\right) \,, \tag{6.59}$$

where $\mathrm{Dir}(\boldsymbol{\theta} \,|\, \overline{\boldsymbol{\theta}}, \alpha)$ is the probability density of a Dirichlet distribution with mean $\overline{\boldsymbol{\theta}}$ and strength α, evaluated at $\boldsymbol{\theta}$. The next sample is instantiated as follows:

$$\boldsymbol{\theta}^{(k,c+1)} = \begin{cases} \boldsymbol{\theta}' & \text{if } w < a(\boldsymbol{\theta}', \boldsymbol{\theta}^{(k,c)}) \qquad \text{(accept)} \\ \boldsymbol{\theta}^{(k,c)} & \text{otherwise} \qquad \text{(reject)} \,, \end{cases} \tag{6.60}$$

where $w \sim \mathrm{U}(0,1)$ is a random variable sampled from a uniform distribution on $[0,1]$. By repeating this procedure of accepting or rejecting $C_k' \geq C_k$ times at the temperature $\tau(k)$, the MCMC sampler generates a set of (dependent) samples $\{\boldsymbol{\theta}^{(k,c)}\}_{c=1}^{C_k'}$. A subset of these $\{\boldsymbol{\theta}^{(k,c)}\}_{c \in \mathcal{C}_k}$, with $|\mathcal{C}_k| = C_k \leq C_k'$, is then used as the importance samples in the computation above (6.57). This subset will generally not include the first few samples, as these samples are likely not yet samples from the equilibrium distribution at that temperature.

An algorithm to compute all ratios

The entire algorithm for computing all K marginal likelihood ratios is given in algorithm 6.1. It has several parameters, in particular: the number of annealing steps, K; their inverse-temperatures (the annealing schedule), $\{\tau(k)\}_{k=1}^K$; the parameters of the MCMC importance sampler at each temperature $\{C_k', C_k, \alpha(k)\}_{k=1}^K$, which are the number of proposed samples,

Algorithm 6.1: **AIS**. To compute all ratios $\{\mathcal{R}_k\}_{k=1}^K$ for the marginal likelihood estimate.

1. Initialise $\boldsymbol{\theta}_{\text{ini}} \sim f_0(\boldsymbol{\theta})$ i.e. from the prior $p(\boldsymbol{\theta} \,|\, m)$
2. For $k = 1$ to K annealing steps
   (a) Run MCMC at temperature $\tau(k-1)$ as follows:
      i. Initialise $\boldsymbol{\theta}^{(k,0)} \leftarrow \boldsymbol{\theta}_{\text{ini}}$ from previous temp.
      ii. Generate the set $\{\boldsymbol{\theta}^{(k,c)}\}_{c=1}^{C'_k} \sim f_{k-1}(\boldsymbol{\theta})$ as follows:
         A. For $c = 1$ to C'_k
            Propose $\boldsymbol{\theta}' \sim \text{Dir}(\boldsymbol{\theta}' \,|\, \boldsymbol{\theta}^{(k,c-1)}, \alpha(k))$
            Accept $\boldsymbol{\theta}^{(k,c)} \leftarrow \boldsymbol{\theta}'$ according to (6.59) and (6.60)
            End For
         B. Store $\boldsymbol{\theta}_{\text{ini}} \leftarrow \boldsymbol{\theta}^{(k,C'_k)}$
      iii. Store a subset of these $\{\boldsymbol{\theta}^{(k,c)}\}_{c \in \mathcal{C}_k}$ with $|\mathcal{C}_k| = C_k \leq C'_k$
   (b) Calculate $\mathcal{R}_k \equiv \frac{\mathcal{Z}_k}{\mathcal{Z}_{k-1}} \approxeq \frac{1}{C_k} \sum_{c=1}^{C_k} \frac{f_k(\boldsymbol{\theta}^{(k,c)})}{f_{k-1}(\boldsymbol{\theta}^{(k,c)})}$

   End For
3. Output $\{\ln \mathcal{R}_k\}_{k=1}^K$ and $\ln \hat{\mathcal{Z}}_K = \sum_{k=1}^K \ln \mathcal{R}_k$ as the approximation to $\ln \mathcal{Z}_K$

the number used for the importance estimate, and the precision of the proposal distribution, respectively.

Nota bene: In the presentation of AIS thus far, we have shown how to compute estimates of $\mathcal{R}_k$ using a set, $\mathcal{C}_k$, of importance samples (see equation (6.56)), chosen from the larger set, $\mathcal{C}'_k$, drawn using a Metropolis-Hastings sampling scheme. In the original paper by Neal (2001), the size of the set $\mathcal{C}_k$ is *exactly one*, and it is only for this case that the validity of AIS as an unbiased estimate has been proved. Because the experiments carried out in this chapter do in fact only use $C_k = |\mathcal{C}_k| = 1$ (as described in section 6.4.1), we stay in the realm of the proven result. It is open research question to show that algorithm 6.1 is unbiased for $C_k = |\mathcal{C}_k| > 1$ (personal communication with R. Neal).

Algorithm 6.1 produces only a single estimate of the marginal likelihood; the variance of this estimate can be obtained from the results of several annealed importance samplers run in parallel. Indeed a particular attraction of AIS is that one can take averages of the marginal likelihood estimates from a set of G annealed importance sampling runs to form a better (unbiased) estimate:

$$\left[\frac{\mathcal{Z}_K}{\mathcal{Z}_0}\right]^{(G)} = \frac{1}{G} \sum_{g=1}^{G} \prod_{k=1}^{K^{(g)}} \mathcal{R}_k^{(g)} \,. \tag{6.61}$$

However this computational resource might be better spent simulating a single chain with a more finely-grained annealing schedule, since for each k we require each pair of distributions $\{f_k(\boldsymbol{\theta}), f_{k-1}(\boldsymbol{\theta})\}$ to be sufficiently close that the importance weights have low variance. Or perhaps the computation is better invested by having a coarser annealing schedule and taking more samples at each temperature to ensure the Metropolis-Hastings sampler has reached equilibrium. In Neal (2001) an in-depth analysis is presented for these and other similar concerns for estimating the marginal likelihoods in some very simple models, using functions of the variance of the importance weights (i.e. the summands in (6.56)) as guides to the reliability of the estimates.

In section 6.5.1 we discuss the performance of AIS for estimating the marginal likelihood of the graphical models used in this chapter, addressing the specific choices of proposal widths, number of samples, and annealing schedules used in the experiments.

6.3.6 Upper bounds on the marginal likelihood

This section is included to justify comparing the marginal likelihood to scores such as MAP and ML. The following estimates based on the ML parameters and the posterior distribution over parameters represent strict bounds on the true marginal likelihood of a model, $p(\mathbf{y})$,

$$p(\mathbf{y}) = \int d\boldsymbol{\theta}\; p(\boldsymbol{\theta}) p(\mathbf{y} \,|\, \boldsymbol{\theta}) \;. \tag{6.62}$$

(where we have omitted the dependence on m for clarity).

We begin with the ML estimate:

$$p(\mathbf{y})_{\mathrm{ML}} = \int d\boldsymbol{\theta}\; \boldsymbol{\delta}(\boldsymbol{\theta} - \boldsymbol{\theta}_{\mathrm{ML}}) p(\mathbf{y} \,|\, \boldsymbol{\theta}) \tag{6.63}$$

which is the expectation of the data likelihood under a delta function about the ML parameter setting. This is a strict upper bound only if $\boldsymbol{\theta}_{\mathrm{ML}}$ has found the global maximum of the likelihood. This may not happen due to local maxima in the optimisation process, for example if the model contains hidden variables and an EM-type optimisation is being employed.

The second estimate is that arising from the MAP estimate,

$$p(\mathbf{y})_{\mathrm{MAP}} = \int d\boldsymbol{\theta}\; \boldsymbol{\delta}(\boldsymbol{\theta} - \boldsymbol{\theta}_{\mathrm{MAP}}) p(\mathbf{y} \,|\, \boldsymbol{\theta}) \tag{6.64}$$

which is the expectation of the data likelihood under a delta function about the MAP parameter setting. However is not a strict upper or lower bound on the marginal likelihood, since this depends on how the prior term acts to position the MAP estimate.

The last estimate, based on the posterior distribution over parameters, is for academic interest only, since we would expect its calculation to be intractable:

$$p(\mathbf{y})_{post.} = \int d\boldsymbol{\theta}\; p(\boldsymbol{\theta} \,|\, \mathbf{y}) p(\mathbf{y} \,|\, \boldsymbol{\theta}) \;. \tag{6.65}$$

This is the expected likelihood under the posterior distribution over parameters. To prove that (6.65) is an upper bound on the marginal likelihood, we use a simple convexity bound as follows:

$$\begin{aligned}
p(\mathbf{y})_{post.} &= \int d\boldsymbol{\theta}\; p(\boldsymbol{\theta} \,|\, \mathbf{y}) p(\mathbf{y} \,|\, \boldsymbol{\theta}) && (6.66)\\
&= \int d\boldsymbol{\theta}\; \frac{p(\boldsymbol{\theta}) p(\mathbf{y} \,|\, \boldsymbol{\theta})}{p(\mathbf{y})} p(\mathbf{y} \,|\, \boldsymbol{\theta}) \qquad \text{by Bayes' rule} && (6.67)\\
&= \frac{1}{p(\mathbf{y})} \int d\boldsymbol{\theta}\; p(\boldsymbol{\theta}) \left[p(\mathbf{y} \,|\, \boldsymbol{\theta}) \right]^2 && (6.68)\\
&\geq \frac{1}{p(\mathbf{y})} \left[\int d\boldsymbol{\theta}\; p(\boldsymbol{\theta}) p(\mathbf{y} \,|\, \boldsymbol{\theta}) \right]^2 \qquad \text{by convexity of } x^2 && (6.69)\\
&= \frac{1}{p(\mathbf{y})} \left[p(\mathbf{y}) \right]^2 = p(\mathbf{y}) \;. && (6.70)
\end{aligned}$$

As we would expect the integral (6.65) to be intractable, we could instead estimate it by taking samples from the posterior distribution over parameters and forming the Monte Carlo estimate:

$$\begin{aligned}
p(\mathbf{y}) \leq p(\mathbf{y})_{post.} &= \int d\boldsymbol{\theta}\; p(\boldsymbol{\theta} \,|\, \mathbf{y}) p(\mathbf{y} \,|\, \boldsymbol{\theta}) && (6.71)\\
&\approx \frac{1}{C} \sum_{c=1}^{C} p(\mathbf{y} \,|\, \boldsymbol{\theta}^{(c)}) && (6.72)
\end{aligned}$$

where $\boldsymbol{\theta}^{(c)} \sim p(\boldsymbol{\theta} \,|\, \mathbf{y})$, the exact posterior. Had we taken samples from the prior $p(\boldsymbol{\theta})$, this would have yielded the true marginal likelihood, so it makes sense that by concentrating samples in areas which give rise to high likelihoods we are over-estimating the marginal likelihood; for this reason we would only expect this upper bound to be close for small amounts of data. An interesting direction of thought would be to investigate the mathematical implications of drawing samples from an approximate posterior instead of the exact posterior, such as that obtained in a variational optimisation, which itself is arrived at from a lower bound on the marginal likelihood; this could well give an even higher upper bound since the approximate variational posterior is likely to neglect regions of low posterior density.

6.4 Experiments

In this section we experimentally examine the performance of the variational Bayesian procedure in approximating the marginal likelihood for all the models in a particular class. We first describe the class defining our space of hypothesised structures, then chose a particular mem-

ber of the class as the "true" structure, generate a set of parameters for that structure, and then generate varying-sized data sets from that structure with those parameters. The task is then to estimate the marginal likelihood of every data set under each member of the class, including the true structure, using each of the scores described in the previous section. The hope is that the VB lower bound will be able to find the true model, based on its scoring, as reliably as the gold standard AIS does. We would ideally like the VB method to perform well even with little available data.

Later experiments take the true structure and analyse the performance of the scoring methods under many different settings of the parameters drawn from the parameter prior for the true structure. Unfortunately this analysis does not include AIS, as sampling runs for each and every combination of the structures, data sets, and parameter settings would take a prohibitively large amount of compute time.

A specific class of graphical model. We look at the specific class of discrete directed *bipartite* graphical models, i.e. those graphs in which only hidden variables can be parents of observed variables, and the hidden variables themselves have no parents. We further restrict ourselves to those graphs which have just $k = |\mathcal{H}| = 2$ hidden variables, and $p = |\mathcal{V}| = 4$ observed variables; both hidden variables are binary i.e. $|\mathbf{s}_{ij}| = 2$ for $j \in \mathcal{H}$, and each observed variable has cardinality $|\mathbf{y}_{ij}| = 5$ for $j \in \mathcal{V}$.

The number of distinct graphs. In the class of bipartite graphs described above, with k distinct hidden variables and p observed variables, there are 2^{kp} possible structures, corresponding to the presence or absence of a directed link between each hidden and each conditionally independent observed variable. If the hidden variables are unidentifiable, which is the case in our example model where they have the same cardinality, then the number of possible graphs is reduced. It is straightforward to show in this example that the number of graphs is reduced from $2^{2\times 4} = 256$ down to 136.

The specific model and generating data. We chose the particular structure shown in figure 6.1, which we call the "true" structure. We chose this structure because it contains enough links to induce non-trivial correlations amongst the observed variables, whilst the class as a whole has few enough nodes to allow us to examine exhaustively every possible structure of the class. There are only three other structures in the class which have more parameters than our chosen structure; these are: two structures in which either the left- or right-most visible node has both hidden variables as parents instead of just one, and one structure which is fully connected. As a caveat, one should note that our chosen true structure is at the higher end of complexity in this class, and so we might find that scoring methods that do not penalise complexity do seemingly better than naively expected.

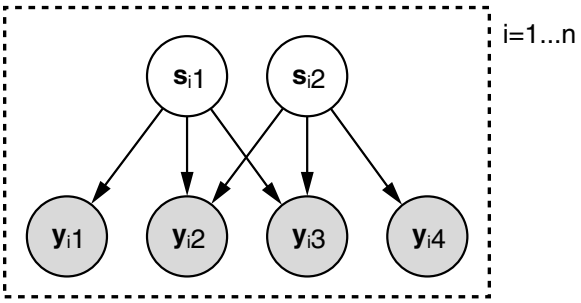

Figure 6.1: The true structure that was used to generate all the data sets used in the experiments. The hidden variables (top) are each binary, and the observed variables (bottom) are each five-valued. This structure has 50 parameters, and is two links away from the fully-connected structure. In total there are 136 possible distinct structures with two (identical) hidden variables and four observed variables.

Evaluation of the marginal likelihood of all possible alternative structures in the class is done for academic interest only; in practice one would embed different structure scoring methods in a greedy model search outer loop (Friedman, 1998) to find probable structures. Here, we are not so much concerned with structure *search* per se, since a prerequisite for a good structure search algorithm is an efficient and accurate method for evaluating any particular structure. Our aim in these experiments is to establish the reliability of the variational bound as a score, compared to annealed importance sampling, and the currently employed asymptotic scores such as BIC and Cheeseman-Stutz.

The parameters of the true model

Conjugate uniform symmetric Dirichlet priors were placed over all the parameters of the model, that is to say in equation (6.4), $\lambda_{jlk} = 1 \; \forall \{jlk\}$. This particular prior was arbitrarily chosen for the purposes of the experiments, and we do not expect it to influence our conclusions much. For the network shown in figure 6.1 parameters were sampled from the prior, once and for all, to instantiate a true underlying model, from which data was then generated. The sampled parameters are shown below (their sizes are functions of each node's and its parents' cardinalities):

$$\boldsymbol{\theta}_1 = \begin{bmatrix} .12 & .88 \end{bmatrix} \quad \boldsymbol{\theta}_3 = \begin{bmatrix} .03 & .03. & .64 & .02 & .27 \\ .18 & .15 & .22 & .19 & .27 \end{bmatrix} \quad \boldsymbol{\theta}_6 = \begin{bmatrix} .10 & .08 & .43 & .03 & .36 \\ .30 & .14 & .07 & .04 & .45 \end{bmatrix}$$

$$\boldsymbol{\theta}_2 = \begin{bmatrix} .08 & .92 \end{bmatrix} \quad \boldsymbol{\theta}_4 = \begin{bmatrix} .10 & .54 & .07 & .14 & .15 \\ .04 & .15 & .59 & .05 & .16 \\ .20 & .08 & .36 & .17 & .18 \\ .19 & .45 & .10 & .09 & .17 \end{bmatrix} \quad \boldsymbol{\theta}_5 = \begin{bmatrix} .11 & .47 & .12 & .30 & .01 \\ .27 & .07 & .16 & .25 & .25 \\ .52 & .14 & .15 & .02 & .17 \\ .04 & .00 & .37 & .33 & .25 \end{bmatrix}$$

where $\{\boldsymbol{\theta}_j\}_{j=1}^{2}$ are the parameters for the hidden variables, and $\{\boldsymbol{\theta}_j\}_{j=3}^{6}$ are the parameters for the remaining four observed variables. Recall that each row of each matrix denotes the

probability of each multinomial setting for a particular configuration of the parents. Each row of each matrix sums to one (up to rounding error). Note that there are only two rows for $\boldsymbol{\theta}_3$ and $\boldsymbol{\theta}_6$ as both these observed variables have just a single binary parent. For variables 4 and 5, the four rows correspond to the parent configurations (in order): $\{[1\ 1], [1\ 2], [2\ 1], [2\ 2]\}$.

Also note that for this particular instantiation of the parameters, both the hidden variable priors are close to deterministic, causing approximately 80% of the data to originate from the $[2\ 2]$ setting of the hidden variables. This means that we may need many data points before the evidence for two hidden variables outweighs that for one.

Incrementally larger and larger data sets were generated with these parameter settings, with

$$n \in \{10,20,40,80,110,160,230,320,400,430,\\ 480,560,640,800,960,1120,1280,2560,5120,10240\} .$$

The items in the $n = 10$ data set are a subset of the $n = 20$ and subsequent data sets, etc. The particular values of n were chosen from an initially exponentially increasing data set size, followed by inclusion of some intermediate data sizes to concentrate on interesting regions of behaviour.

6.4.1 Comparison of scores to AIS

All 136 possible distinct structures were scored for each of the 20 data set sizes given above, using MAP, BIC, CS, VB and AIS scores. Strictly speaking, MAP is not an approximation to the marginal likelihood, but it is an upper bound (see section 6.3.6) and so is nevertheless interesting for comparison.

We ran EM on each structure to compute the MAP estimate of the parameters, and from it computed the BIC score as described in section 6.3.2. We also computed the BIC score including the parameter prior, denoted BICp, which was obtained by including a term $\ln p(\hat{\boldsymbol{\theta}} \,|\, m)$ in equation (6.26). From the same EM optimisation we computed the CS score according to section 6.3.3. We then ran the variational Bayesian EM algorithm with the same initial conditions to give a lower bound on the marginal likelihood. For both these optimisations, random parameter initialisations were used in an attempt to avoid local maxima — the highest score over three random initialisations was taken for each algorithm; empirically this heuristic appeared to avoid local maxima problems. The EM and VBEM algorithms were terminated after either 1000 iterations had been reached, or the change in log likelihood (or lower bound on the log marginal likelihood, in the case of VBEM) became less than 10^{-6} per datum.

For comparison, the AIS sampler was used to estimate the marginal likelihood (see section 6.3.5), annealing from the prior to the posterior in $K = 16384$ steps. A nonlinear anneal-

ing schedule was employed, tuned to reduce the variance in the estimate, and the Metropolis proposal width was tuned to give reasonable acceptance rates. We chose to have just a single sampling step at each temperature (i.e. $C'_k = C_k = 1$), for which AIS has been proven to give unbiased estimates, and initialised the sampler at each temperature with the parameter sample from the previous temperature. These particular choices are explained and discussed in detail in section 6.5.1. Initial marginal likelihood estimates from single runs of AIS were quite variable, and for this reason several more batches of AIS runs were undertaken, each using a different random initialisation (and random numbers thereafter); the total of G batches of scores were averaged according to the procedure given in section 6.3.5, equation (6.61), to give the AIS$^{(G)}$ score. In total, $G = 5$ batches of AIS runs were carried out.

Scoring all possible structures

Figure 6.2 shows the MAP, BIC, BICp, CS, VB and AIS$^{(5)}$ scores obtained for each of the 136 possible structures against the number of parameters in the structure. Score is measured on the vertical axis, with each scoring method (columns) sharing the same vertical axis range for a particular data set size (rows).

The horizontal axis of each plot corresponds to the number of parameters in the structure (as described in section 6.3.2). For example, at the extremes there is one structure with 66 parameters which is the fully connected structure, and one structure with 18 parameters which is the fully unconnected structure. The structure that generated the data has exactly 50 parameters. In each plot we can see that several structures can occupy the same column, having the same number of parameters. This means that, at least visually, it is not always possible to unambiguously assign each point in the column to a particular structure.

The scores shown here are those corrected for aliases — the difference between the uncorrected and corrected versions is only just perceptible as a slight downward movement of the low parameter structures (those with just one or zero hidden variables), as these have a smaller number of aliases S (see equation (6.26)).

In each plot, the true structure is highlighted by a '$\circ$' symbol, and the structure currently ranked highest by that scoring method is marked with a '$\times$'. We can see the general upward trend for the MAP score which prefers more complicated structures, and the pronounced downward trend for the BIC and BICp scores which (over-)penalise structure complexity. In addition one can see that neither upward or downward trends are apparent for VB or AIS scores. Moreover, the CS score does tend to show a downward trend similar to BIC and BICp, and while this trend weakens with increasing data, it is still present at $n = 10240$ (bottom row). Although not verifiable from these plots, we should note that for the vast majority of the scored structures

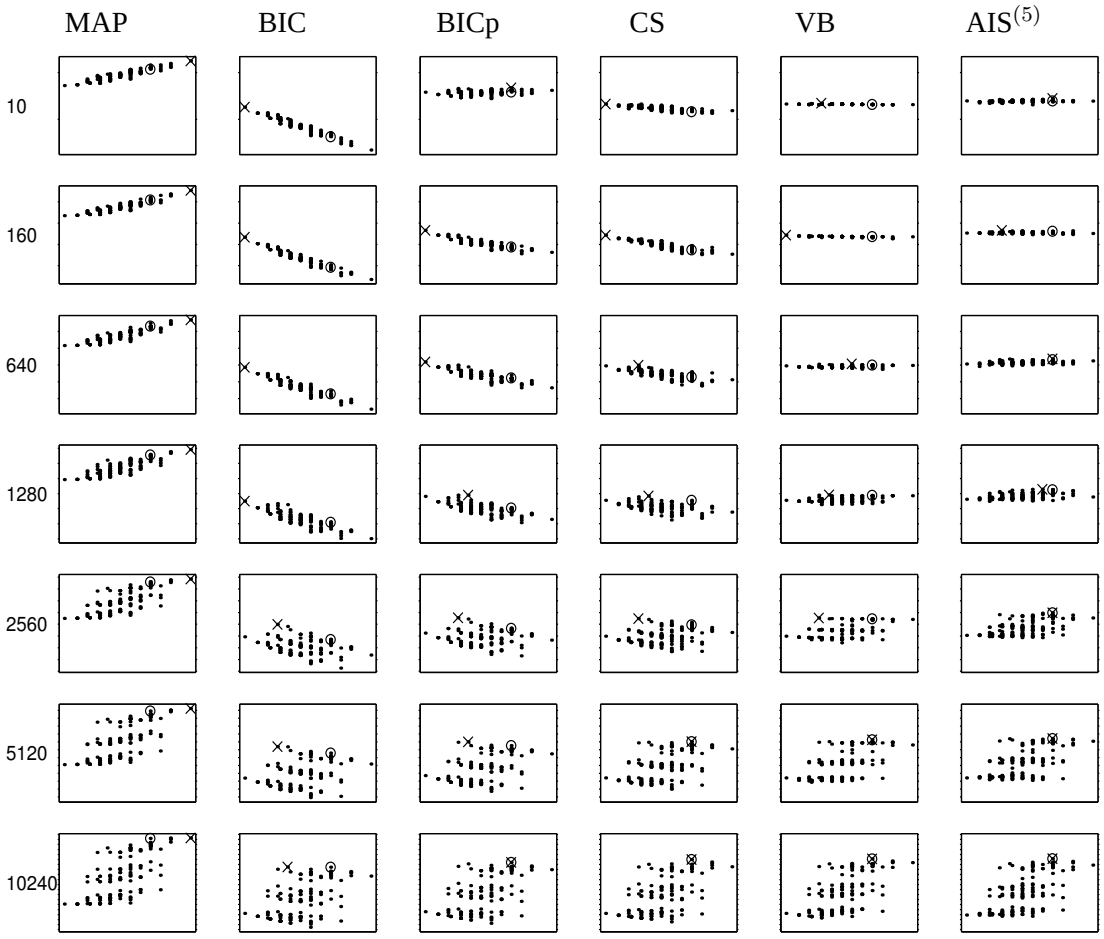

Figure 6.2: Scores for all 136 of the structures in the model class, by each of six scoring methods. Each plot has the score (approximation to the log marginal likelihood) on the vertical axis, with tick marks every 40 nats, and the number of parameters on the horizontal axis (ranging from 18 to 66). The middle four scores have been corrected for aliases (see section 6.3.2). Each row corresponds to a data set of a different size, n: from top to bottom we have $n = 10, 160, 640, 1280, 2560, 5120, 10240$. The true structure is denoted with a '$\circ$' symbol, and the highest scoring structure in each plot marked by the '$\times$' symbol. Every plot in the same row has the same scaling for the vertical score axis, set to encapsulate every structure for all scores. For a description of how these scores were obtained see section 6.4.1.

and data set sizes, the AIS$^{(5)}$ score is higher than the VB lower bound, as we would expect (see section 6.5.1 for exceptions to this observation).

The horizontal bands observed in the plots is an interesting artifact of the particular model used to generate the data. For example, we find on closer inspection some strictly followed trends: all those model structures residing in the upper band have the first three observable variables ($j = 3, 4, 5$) governed by at least one of the hidden variables; and all those structures in the middle band have the third observable ($j = 4$) connected to at least one hidden variable.

In this particular example, AIS finds the correct structure at $n = 960$ data points, but unfortunately does not retain this result reliably until $n = 2560$. At $n = 10240$ data points, BICp, CS, VB and AIS all report the true structure as being the one with the highest score amongst the other contending structures. Interestingly, BIC still does not select the correct structure, and MAP has given a structure with sub-maximal parameters the highest score. The latter observation may well be due to local maxima in the EM optimisation, since for previous slightly smaller data sets MAP chooses the fully-connected structure as expected. Note that as we did not have intermediate data sets it may well be that, for example, AIS reliably found the structure after 1281 data points, but we cannot know this without performing more experiments.

Ranking of the true structure

A somewhat more telling comparison of the scoring methods is given by how they rank the true structure amongst the alternative structures. Thus a ranking of 1 means that the scoring method has given the highest marginal likelihood to the true structure.

Note that a performance measure based on ranking makes several assumptions about our choice of loss function. This performance measure disregards information in the posterior about the structures with lower scores, reports only the number of structures that have higher scores, and not the amount by which the true structure is beaten. Ideally, we would compare a quantity that measured the divergence of all structures' posterior probabilities from the true posterior.

Moreover, we should keep in mind that at least for small data set sizes, there is no reason to assume that the actual posterior over structures has the true structure at its mode. Therefore it is slightly misleading to ask for high rankings at small data set sizes.

Table 6.1 shows the ranking of the true structure, as it sits amongst all the possible structures, as measured by each of the scoring methods MAP, BIC, BICp, CS, VB and AIS$^{(5)}$; this is also plotted in figure 6.3 where the MAP ranking is not included for clarity. Higher positions in the plot correspond to better rankings.

n	MAP	BIC*	BICp*	CS*	VB*	BIC	BICp	CS	VB	AIS$^{(5)}$
10	21	127	55	129	122	127	50	129	115	59
20	12	118	64	111	124	118	64	111	124	135
40	28	127	124	107	113	127	124	107	113	15
80	8	114	99	78	116	114	99	78	116	44
110	8	109	103	98	114	109	103	98	113	2
160	13	119	111	114	83	119	111	114	81	6
230	8	105	93	88	54	105	93	88	54	54
320	8	111	101	90	44	111	101	90	33	78
400	6	101	72	77	15	101	72	77	15	8
430	7	104	78	68	15	104	78	68	14	18
480	7	102	92	80	55	102	92	80	44	2
560	9	108	98	96	34	108	98	96	31	11
640	7	104	97	105	19	104	97	105	17	7
800	9	107	102	108	35	107	102	108	26	23
960	13	112	107	76	16	112	107	76	13	1
1120	8	105	96	103	12	105	96	103	12	4
1280	7	90	59	8	3	90	59	6	3	5
2560	6	25	17	11	11	25	15	11	11	1
5120	5	6	5	1	1	6	5	1	1	1
10240	3	2	1	1	1	2	1	1	1	1

Table 6.1: Ranking of the true structure by each of the scoring methods, as the size of the data set is increased. Asterisks (*) denote scores uncorrected for parameter aliasing in the posterior. Strictly speaking, the MAP score is not an estimate of the marginal likelihood. Note that these results are from data generated from only one instance of parameters under the true structure's prior over parameters.

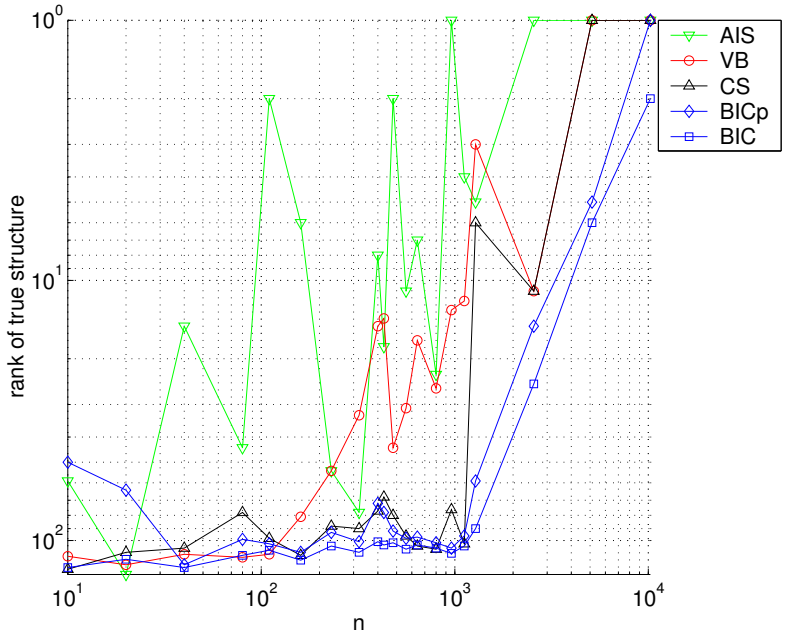

Figure 6.3: Ranking given to the true structure by each scoring method for varying data set sizes (higher in plot is better), by BIC, BICp, CS, VB and AIS$^{(5)}$ methods.

For small n, the AIS score produces a better ranking for the true structure than any of the other scoring methods, which suggests that the AIS sampler is managing to perform the Bayesian parameter averaging process more accurately than other approximations. For almost all n, VB outperforms BIC, BICp and CS, consistently giving a higher ranking to the true structure. Of particular note is the stability of the VB score ranking with respect to increasing amounts of data as compared to AIS (and to some extent CS).

Columns in table 6.1 with asterisks (*) correspond to scores that are not corrected for aliases, and are omitted from the figure. These corrections assume that the posterior aliases are well separated, and are valid only for large amounts of data and/or strongly-determined parameters. In this experiment, structures with two hidden states acting as parents are given a greater correction than those structures with only a single hidden variable, which in turn receive corrections greater than the one structure having no hidden variables. Of interest is that the correction nowhere degrades the rankings of any score, and in fact improves them very slightly for CS, and especially so for the VB score.

Score discrepancies between the true and top-ranked structures

Figure 6.4 plots the differences in score between the true structure and the score of the structure ranked top by BIC, BICp, CS, VB and AIS methods. The convention used means that all the differences are exactly zero or negative, measured from the score of the top-ranked structure — if the true structure is ranked top then the difference is zero, otherwise the true structure's score must be less than the top-ranked one. The true structure has a score that is close to the top-ranked structure in the AIS method; the VB method produces approximately similar-sized differences, and these are much less on the average than the CS, BICp, and BIC scores. For a better comparison of the non-sampling-based scores, see section 6.4.2, and figure 6.6.

Computation Time

Scoring all 136 structures at 480 data points on a 1GHz Pentium III processor took: 200 seconds for the MAP EM algorithms required for BIC/BICp/CS, 575 seconds for the VBEM algorithm required for VB, and 55000 seconds (15 hours) for a single run of the AIS algorithm (using 16384 samples as in the main experiments). All implementations were in MATLAB. Given the massive computational burden of the sampling method (approx 75 hours), which still produces fairly variable scores when averaging over five runs, it does seem as though CS and VB are proving very useful indeed. Can we justify the mild overall computational increase for VB? This increase results from both computing differences between digamma functions as opposed to ratios, and also from an empirically-observed slower convergence rate of the VBEM algorithm as compared to the EM algorithm.

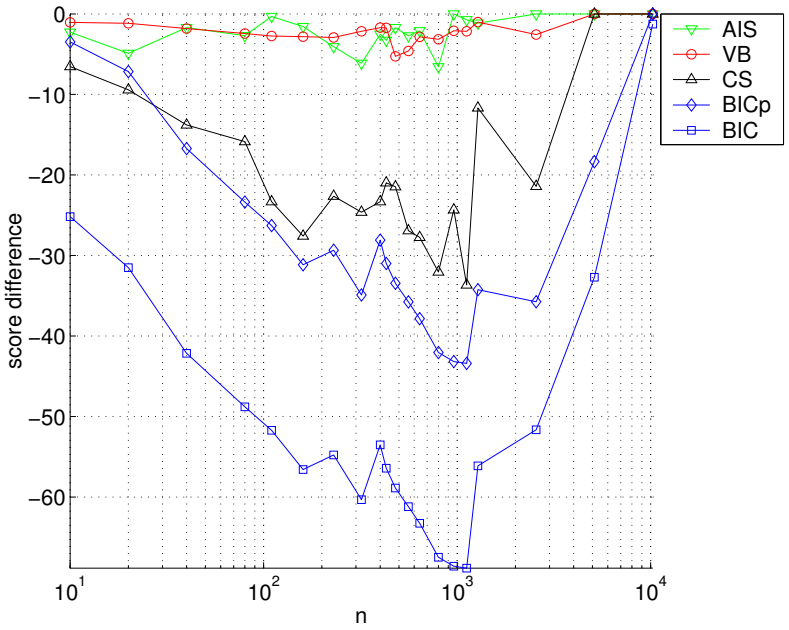

Figure 6.4: Differences in log marginal likelihood estimates (scores) between the top-ranked structure and the true structure, as reported by BIC, BICp, CS, VB and AIS$^{(5)}$ methods. All differences are exactly zero or negative: if the true structure is ranked top then the difference is zero, otherwise the score of the true structure must be less than the top-ranked structure. Note that these score differences are not per-datum scores, and therefore are not normalised for the data n.

6.4.2 Performance averaged over the parameter prior

The experiments in the previous section used a single instance of sampled parameters for the true structure, and generated data from this particular model. The reason for this was that, even for a single experiment, computing an exhaustive set of AIS scores covering all data set sizes and possible model structures takes in excess of 15 CPU days.

In this section we compare the performance of the scores over many different sampled parameters of the true structure (shown in figure 6.1). 106 parameters were sampled from the prior (as done once for the single model in the previous section), and incremental data sets generated for each of these instances as the true model. MAP EM and VBEM algorithms were employed to calculate the scores as described in section 6.4.1. For each instance of the true model, calculating scores for all data set sizes used and all possible structures, using three random restarts, for BIC/BICp/CS and VB took approximately 2.4 and 4.2 hours respectively on an Athlon 1800 Processor machine, which corresponds to about 1.1 and 1.9 seconds for each individual score.

The results are plotted in figure 6.5, which shows the median ranking given to the true structure by each scoring method, computed over 106 randomly sampled parameter settings. This plot corresponds to a smoothed version of figure 6.3, but unfortunately cannot contain AIS averages

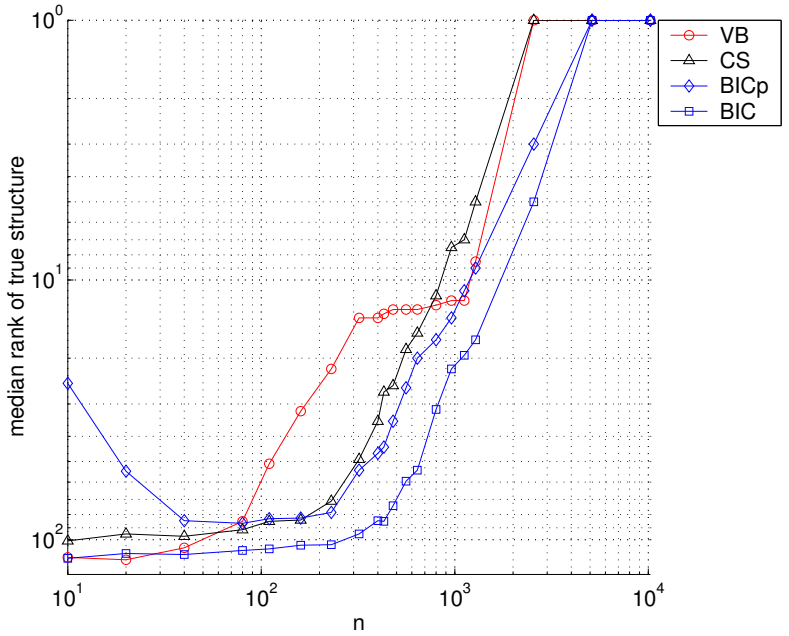

Figure 6.5: Median ranking of the true structure as reported by BIC, BICp, CS and VB methods, against the size of the data set n, taken over 106 instances of the true structure.

% times that \ than	BIC*	BICp*	CS*	CS*†	BIC	BICp	CS	CS†
VB ranks worse	16.9	30.2	31.8	32.8	15.1	29.6	30.9	31.9
same	11.1	15.0	20.2	22.1	11.7	15.5	20.9	22.2
better	72.0	54.8	48.0	45.1	73.2	55.0	48.2	45.9

Table 6.2: Comparison of the VB score to its competitors, using the ranking of the true structure as a measure of performance. The table gives the percentage fraction of times that the true structure was ranked lower, the same, and higher by VB than by the other methods (rounded to nearest .1%). The ranks were collected from all 106 generated parameters and all 20 data set sizes. Note that VB outperforms all competing scores, whether we base our comparison on the alias-corrected or uncorrected (*) versions of the scores. The CS score annotated with † is an improvement on the original CS score, and is explained in section 6.5.2.

for the computational reasons mentioned above. The results clearly show that for the most part VB outperforms all other scores on this task by this measure although there is a region in which VB seems to underperform CS, as measured by the median score.

Table 6.2 shows in more detail the performance of VB and its alias-uncorrected counterpart VB* in terms of the number of times the score correctly selects the true model (i.e. ranks it top). The data was collated from all 106 sampled true model structures, and all 20 data set sizes, giving a total of 288320 structures that needed to be scored by each approximate method. We see that VB outperforms the other scores convincingly, whether we compare the uncorrected (left hand side of table) or corrected (right hand side) scores. The results are more persuasive for the alias-corrected scores, suggesting that VB is benefitting more from this modification — it is not obvious why this should be so.

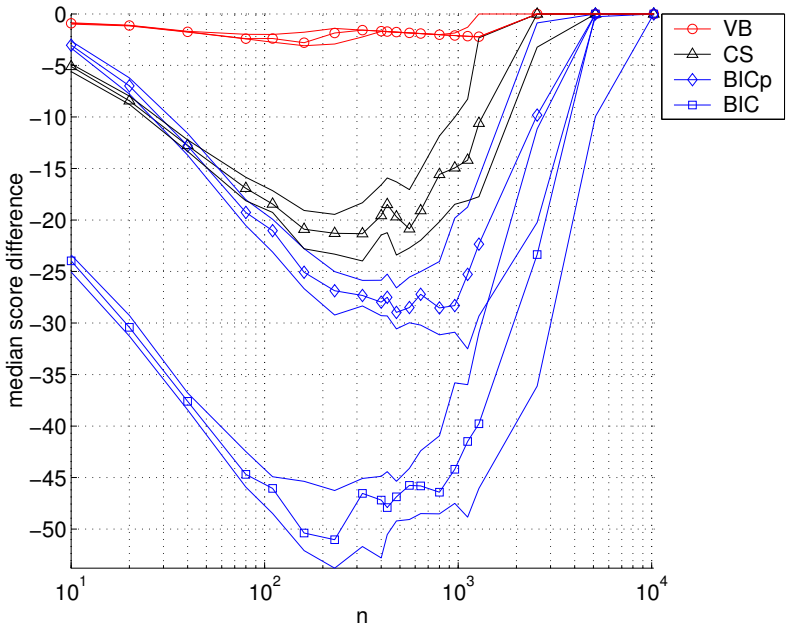

Figure 6.6: Median difference in score between the true and top-ranked structures, under BIC, BICp, CS and VB scoring methods, against the size of the data set n, taken over 106 instances of the true structure. Also plotted are the 40-60% intervals about the medians.

These percentages are likely to be an underestimate of the success of VB, since on close examination of the individual EM and VBEM optimisations, it was revealed that for several cases the VBEM optimisation reached the maximum number of allowed iterations before it had converged, whereas EM always converged. Generally speaking the VBEM algorithm was found to require more iterations to reach convergence than EM, which would be considered a disadvantage if it were not for the considerable performance improvement of VB over BIC, BICp and CS.

We can also plot the smoothed version of figure 6.4 over instances of parameters of the true structure drawn from the prior; this is plotted in figure 6.6, which shows the median difference between the score of the true structure and the structure scoring highest under BIC, BICp, CS and VB. Also plotted is the 40-60% interval around the median. Again, the AIS experiments would have taken an unfeasibly large amount of computation time, and were not carried out.

We can see quite clearly here that the VB score of the true structure is generally much closer to that of the top-ranked structure than is the case for any of the other scores. This observation in itself is not particularly satisfying, since we are comparing scores to scores rather than scores to exact marginal likelihoods; nevertheless it can at least be said that the dynamic range between true and top-ranked structure scores by the VB method is much smaller than the range for the other methods. This observation is also apparent (qualitatively) across structures in the various plots in figure 6.2. We should be wary about the conclusions drawn from this graph comparing VB to the other methods: a completely ignorant algorithm which gives the same score to all

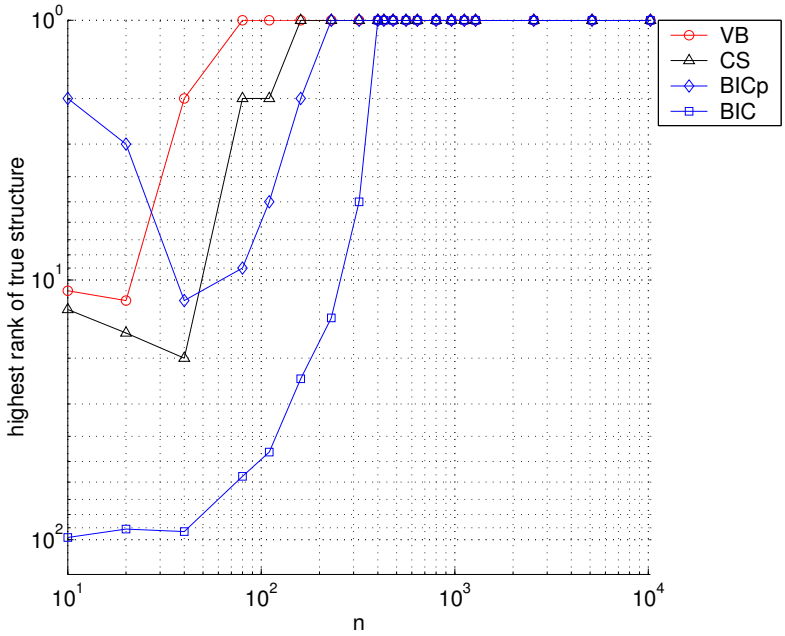

Figure 6.7: The highest ranking given to the true structure under BIC, BICp, CS and VB methods, against the size of the data set n, taken over 106 instances of the true structure. These two traces can be considered as the results of the $\min$ operation on the rankings of all the 106 instances for each n in figure 6.5.

possible structures would look impressive on this plot, giving a score difference of zero for all data set sizes.

Figures 6.7 and 6.8 show the best performance of the BIC, BICp, CS and VB methods over the 106 parameter instances, in terms of the rankings and score differences. These plots can be considered as the extrema of the median ranking and median score difference plots, and reflect the bias in the score.

Figure 6.7 shows the best ranking given to the true structure by all the scoring methods, and it is clear that for small data set sizes the VB and CS scores can perform quite well indeed, whereas the BIC scores do not manage a ranking even close to these. This result is echoed in figure 6.8 for the score differences, although we should bear in mind the caveat mentioned above (that the completely ignorant algorithm can do well by this measure).

We can analyse the expected performance of a naive algorithm which simply picks any structure at random as the guess for the true structure: the best ranking given to the true model in a set of 106 trials where a structure is chosen at random from the 136 structures is, on the average, roughly 1.8. We can see in figure 6.7 that CS and VB surpass this for $n > 30$ and $n > 40$ data points respectively, but that BICp and BIC do so only after 300 and 400 data points. However we should remember that, for small data set sizes, the true posterior over structures may well not have the true model at its mode.

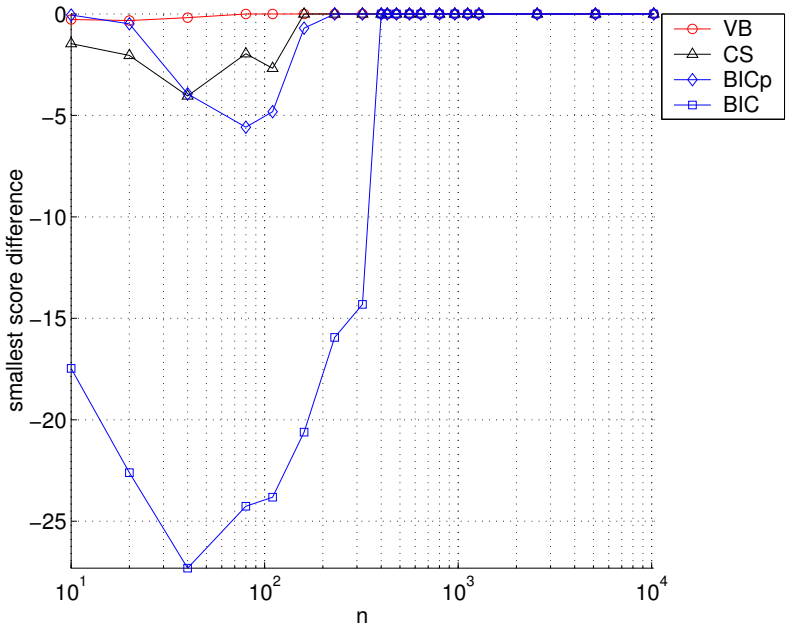

Figure 6.8: The smallest difference in score between the true and top-ranked structures, under BIC, BICp, CS and VB methods, against the size of the data set n, taken over 106 instances of the true structure. These two traces can be considered as the results of the $\max$ operation on the all the 106 differences for each n in figure 6.6.

Lastly, we can examine the success rate of each score at picking the correct structure. Figure 6.9 shows the fraction of times that the true structure is ranked top by the different scoring methods. This plot echoes those results in table 6.2.

6.5 Open questions and directions

This section is split into two parts which discuss some related issues arising from the work in this chapter. In section 6.5.1 we discuss some of the problems experienced when using the AIS approach, and suggest possible ways to improve the methods used in our experiments. In section 6.5.2 we more thoroughly revise the parameter-counting arguments used for the BIC and CS scores, and provide a method for estimating the complete and incomplete-data dimensionalities in arbitrary models, and as a result form a modified score CS†.

6.5.1 AIS analysis, limitations, and extensions

The technique of annealed importance sampling is currently regarded as a state-of-the-art method for estimating the marginal likelihood in discrete-variable directed acyclic graphical models (personal communication with R. Neal, Z. Ghahramani and C. Rasmussen). In this section the

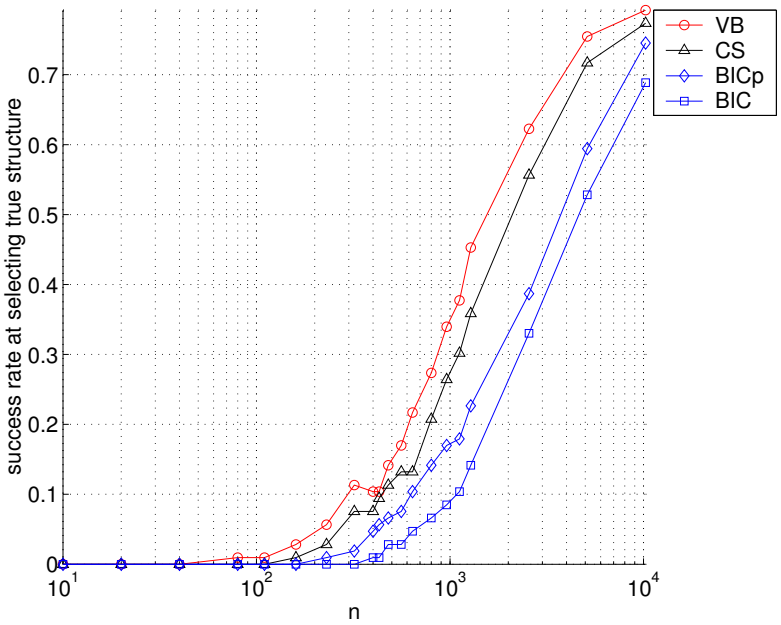

Figure 6.9: The success rate of the scoring methods BIC, BICp, CS and VB, as measured by the fraction of 106 trials in which the true structure was given ranking 1 amongst the 136 candidate structures, plotted as a function of the data set size. See also table 6.2 which presents softer performance rates (measured in terms of relative rankings) pooled from all the data set sizes and 106 parameter samples.

AIS method is critically examined as a reliable tool to judge the performance of the BIC, CS and VB scores.

The implementation of AIS has considerable flexibility; for example the user is left to specify the length, granularity and shape of the annealing schedules, the form of the Metropolis-Hastings sampling procedure, the number of samples taken at each temperature, etc. These and other parameters were described in section 6.3.5; here we clarify our choices of settings and discuss some further ways in which the sampler could be improved. Throughout this subsection we use AIS to refer to the algorithm which provides a single estimate of the marginal likelihood, i.e. $\text{AIS}^{(1)}$.

First off, how can we be sure that the AIS sampler is reporting the correct answer for the marginal likelihood of each structure? To be sure of a correct answer one should use as long and gradual an annealing schedule as possible, containing as many samples at each temperature as is computationally viable (or compare to a very long simple importance sampler). In the AIS experiments in this chapter we always opted for a single sample at each step of the annealing schedule, initialising the parameter at the next temperature at the last accepted sample, and ensured that the schedule itself was as finely grained as we could afford. This reduces the variables at our disposal to a single parameter, namely the total number of samples taken in each run of AIS, which is then directly related to the schedule granularity. Without yet discussing the shape

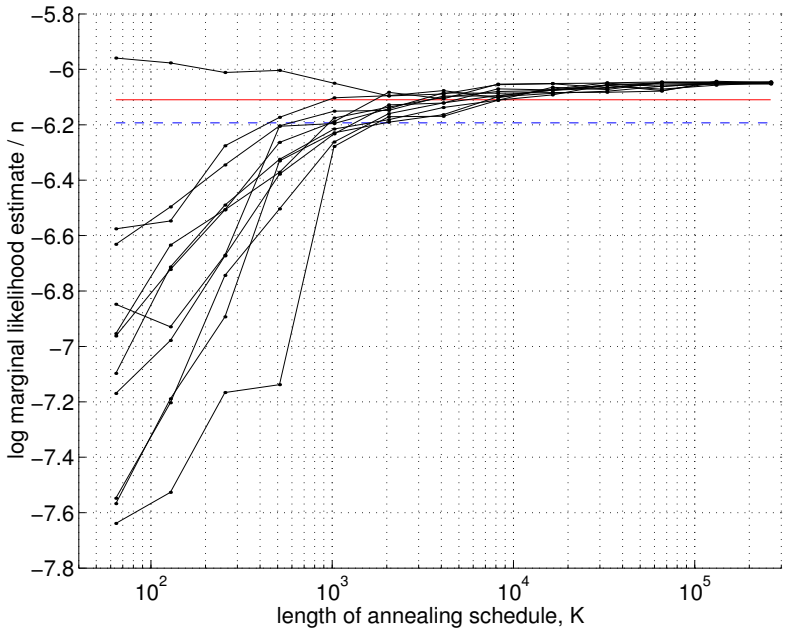

Figure 6.10: Logarithm of AIS estimates (vertical) of the marginal likelihood for different initial conditions of the sampler (different traces) and different duration of annealing schedules (horizontal), for the true structure with $n = 480$ data points. The top-most trace is that corresponding to setting the initial parameters to the true values that generated the data. Shown are also the BIC score (dashed) and the VB lower bound (solid).

of the annealing schedule, we can already examine the performance of the AIS sampler as a function of the number of samples.

Figure 6.10 shows several AIS estimates of the marginal likelihood for the data set of size $n = 480$ under the model having the true structure. Each trace is a result of initialising the AIS sampler at a different position in parameter space sampled from the prior (6.4), except for the top-most trace which is the result of initialising the AIS algorithm at the exact parameters that were used to generate the data (which as the experimenter we have access to). It is important to understand the abscissa of the plot: it is the number of samples in the AIS run and, given the above comments, relates to the granularity of the schedule; thus the points on a particular trace do *not* correspond to progress through the annealing schedule, but in fact constitute the results of runs that are completely different other than in their common parameter initialisation.

Also plotted for reference are the VB and BIC estimates of the log marginal likelihood for this data set under the true structure, which are not functions of the annealing duration. We know that the VB score is a strict lower bound on the log marginal likelihood, and so those estimates from AIS that consistently fall below this score must be indicative of an inadequate annealing schedule shape or duration.

For short annealing schedules, which are necessarily coarse to satisfy the boundary requirements on τ (see equation (6.49)), it is clear that the AIS sampling is badly under-estimating the log marginal likelihood. This can be explained simply because the rapid annealing schedule does not give the sampler time to locate and exploit regions of high posterior probability, forcing it to neglect representative volumes of the posterior mass; this conclusion is further substantiated since the AIS run started from the true parameters (which if the data is representative of the model should lie in a region of high posterior probability) over-estimates the marginal likelihood, because it is prevented from exploring regions of low probability. Thus for coarse schedules of less than about $K = 1000$ samples, the AIS estimate of the log marginal likelihood seems biased and has very high variance. Note that the construction of the AIS algorithm guarantees that the estimates of the marginal likelihood are unbiased, but not necessarily the log marginal likelihood.

We see that all runs converge for sufficiently long annealing schedules, with AIS passing the BIC score at about 1000 samples, and the VB lower bound at about 5000 samples. Thus, loosely speaking, where the AIS and VB scores intersect we can consider their estimates to be roughly equally reliable. We can then compare their computational burdens and make some statement about the advantage of one over the other in terms of compute time. At $n = 480$ the VB scoring method requires about $1.5s$ to score the structure, whereas AIS at $n = 480$ and $K = 2^{13}$ requires about $100s$; thus for this scenario VB is 70 times more efficient at scoring the structures (at its own reliability).

In this chapter's main experiments a value of $K = 2^{14} = 16384$ steps was used, and it is clear from figure 6.10 that we can be fairly sure of the AIS method reporting a reasonably accurate result at this value of K, at least for $n = 480$. However, how would we expect these plots to look for larger data sets in which the posterior over parameters is more peaky and potentially more difficult to navigate during the annealing?

A good indicator of the mobility of the Metropolis-Hastings sampler is the acceptance rate of proposed samples, from which the representative set of importance weights are computed (see (6.60)). Figure 6.11 shows the fraction of accepted proposals during the annealing run, averaged over AIS scoring of all 136 possible structures, plotted against the size of the data set, n; the error bars are the standard errors of the mean acceptance rate across scoring all structures. We can see that at $n = 480$ the acceptance rate is rarely below 60%, and so one would indeed expect to see the sort of convergence shown in figure 6.10. However for the larger data sets the acceptance rate drops to 20%, implying that the sampler is having considerable difficulty obtaining representative samples from the posterior distributions in the annealing schedule. Fortunately this drop is only linear in the logarithm of the data size. For the moment, we defer discussing the temperature dependence of the acceptance rate, and first consider combining AIS sampling runs to reduce the variance of the estimates.

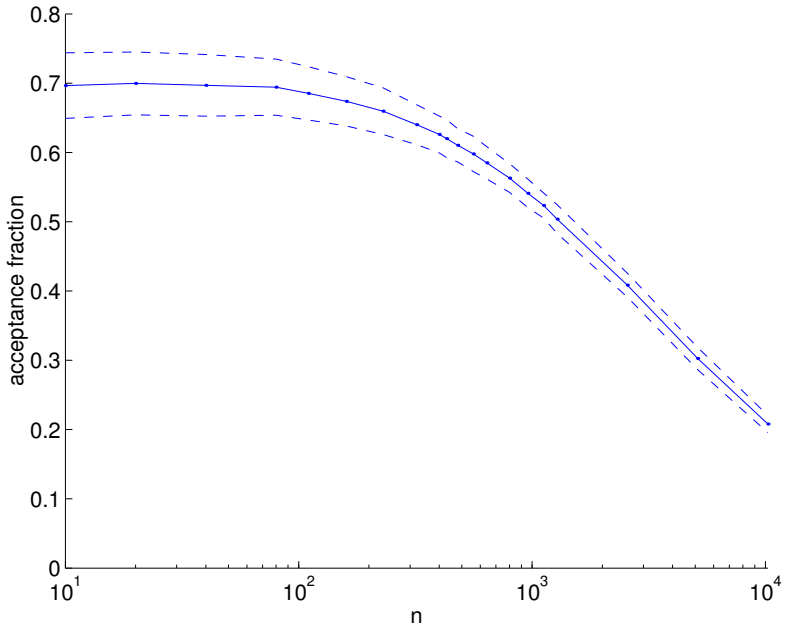

Figure 6.11: Acceptance rates of the Metropolis-Hastings proposals along the entire annealing schedule, for one batch of AIS scoring of all structures, against the size of the data set, n. The dotted lines are the sample standard deviations across all structures for each n.

One way of reducing the variance in our estimate of the marginal likelihood is to pool the results of several AIS samplers run in parallel according to the averaging in equation (6.61). Returning to the specific experiments reported in section 6.4, table 6.3 shows the results of running five AIS samplers in parallel with different random seeds on the entire class of structures and data set sizes, and then using the resulting averaged AIS estimate, $\text{AIS}^{(5)}$, as a score for ranking the structures. In the experiments it is the performance of these averaged scores that are compared to the other scoring methods: BIC, CS and VB. To perform five runs took at least 40 CPU days on an Athlon 1800 Processor machine.

By examining the reported AIS scores, both for single and pooled runs, over the 136 structures and 20 data set sizes, and comparing them to the VB lower bound, we can see how often AIS violates the lower bound. Table 6.4 shows the number of times the reported AIS score is below the VB lower bound, along with the rejection rates of the Metropolis-Hastings sampler that was used in the experiments (which are also plotted in figure 6.11). From the table we see that for small data sets the AIS method reports "valid" results and the Metropolis-Hastings sampler is accepting a reasonable proportion of proposed parameter samples. However at and beyond $n = 560$ the AIS sampler degrades to the point where it reports "invalid" results for more than half the 136 structures it scores. However, since the AIS estimate is noisy and we know that the tightness of the VB lower bound increases with n, this criticism could be considered too harsh — indeed if the bound were tight, we would expect the AIS score to violate the bound on roughly 50% of the runs anyway. The lower half of the table shows that, by combining AIS estimates from separate runs, we obtain an estimate that violates the VB lower bound far less

n	$AIS^{(1)}$ #1	$AIS^{(1)}$ #2	$AIS^{(1)}$ #3	$AIS^{(1)}$ #4	$AIS^{(1)}$ #5	$AIS^{(5)}$
10	27	38	26	89	129	59
20	100	113	88	79	123	135
40	45	88	77	5	11	15
80	10	47	110	41	95	44
110	1	50	8	2	62	2
160	33	2	119	31	94	6
230	103	25	23	119	32	54
320	22	65	51	44	42	78
400	89	21	1	67	10	8
430	29	94	21	97	9	18
480	2	42	14	126	18	2
560	47	41	7	59	7	11
640	12	10	23	2	23	7
800	7	3	126	101	22	23
960	1	4	1	128	8	1
1120	3	53	3	37	133	4
1280	76	2	50	7	12	5
2560	1	1	4	1	1	1
5120	12	1	24	2	16	1
10240	1	1	2	12	1	1

Table 6.3: Rankings resulting from averaging batches of AIS scores. Each one of the five columns correspond to a different initialisation of the sampler, and gives the rankings resulting from a single run of AIS for each of the 136 structures and 20 data set size combinations. The last column is the ranking of the true structure based on the mean of the AIS marginal likelihood estimates from all five runs of AIS of each structure and data set size (see section 6.3.5 for averaging details).

n	10 . . . 560	640	800	960	1120	1280	2560	5120	10240
single									
#$AIS^{(1)}$<VB*	≤5.7	12.3	8.5	12.3	10.4	17.0	25.5	53.8	71.7
#$AIS^{(1)}$<VB	≤7.5	15.1	9.4	14.2	12.3	20.8	31.1	59.4	74.5
% M-H rej.	<40.3	41.5	43.7	45.9	47.7	49.6	59.2	69.7	79.2
averaged									
#$AIS^{(5)}$<VB*	0	0.0	0.0	0.0	0.0	0.7	3.7	13.2	50.0
#$AIS^{(5)}$<VB	≤1.9	0.0	0.0	0.0	1.5	2.2	5.1	19.9	52.9

Table 6.4: AIS violations: for each size data set, n, we show the percentage of times, over the 136 structures, that a particular *single* AIS run reports marginal likelihoods below the VB lower bound. These are given for the VB scores that are uncorrected (*) and corrected for aliases. Also shown are the average percentage rejection rates of the Metropolis-Hastings sampler used to gather samples for the AIS estimates. The bottom half of the table shows the similar violations by the AIS score that are made from averaging the estimates of marginal likelihoods from five separate runs of AIS (see section 6.3.5). Note that the Metropolis-Hastings rejection rates are still just as high for each of the individual runs (not given here).

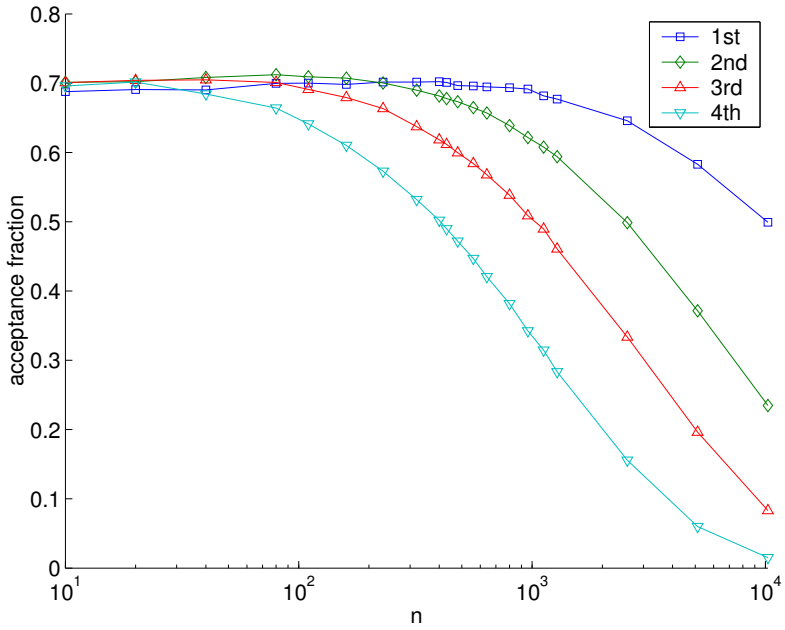

Figure 6.12: Acceptance rates of the Metropolis-Hastings proposals for each of four quarters of the annealing schedule, for one batch of AIS scoring of all structures, against the size of the data set, n. Standard errors of the means are omitted for clarity.

often, and as expected we see the 50% violation rate for large amounts of data. This is a very useful result, and obviates to some extent the Metropolis-Hastings sampler's deficiency in all five runs.

However, considering for the moment a single AIS run, for large data set sizes the VB bound is still violated an unacceptable number of times, suggesting that the Metropolis-Hastings proposals are simply not adequate for these posterior landscapes. This suggests a modification to the proposal mechanism, outlined below. Diagnostically speaking, this hopefully has served as a good example of the use of readily-computable VB lower bounds for evaluating the reliability of the AIS method *post hoc*.

Let us return to examining why the sampler is troubled for large data set sizes. Figure 6.12 shows the fraction of accepted Metropolis-Hastings proposals during each of four quarters of the annealing schedule used in the experiments. The rejection rate tends to increase moving from the beginning of the schedule (the prior) to the end (the posterior), the degradation becoming more pronounced for large data sets. This is most probably due to the proposal width remaining unchanged throughout all the AIS implementations: ideally one would use a predetermined sequence of proposal widths which would be a function of the amount of data, n, and the position along the schedule. This would hopefully eliminate or at least alleviate the pronounced decrease in acceptance rate across the four quarters, but would also cause each individual trace to not drop so severely with n.

We can use a heuristic argument to roughly predict the optimal proposal width to use for the AIS method. From mathematical arguments outlined in sections 1.3.2 and 1.3.4, the precision of the posterior distribution over parameters is approximately proportional to the size of the data set n. Furthermore, the distribution being sampled from at step k of the AIS schedule is effectively that resulting from a fraction $\tau(k)$ of the data. Therefore these two factors imply that the width of the Metropolis-Hastings proposal distribution should be inversely proportional to $\sqrt{n\tau(k)}$. In the case of multinomial variables, since the variance of a Dirichlet distribution is approximately inversely proportional to the strength, α, (see appendix A), then the optimal strength of the proposal distribution should be $\alpha_{opt} \propto n\tau(k)$ if its precision is to match the posterior precision. Note that we are at liberty to set these proposal precisions arbitrarily beforehand without causing the sampler to become biased.

We have not yet discussed the shape of the annealing schedule: should the inverse-temperatures $\{\tau(k)\}_{k=1}^{K}$ change linearly from 0 to 1, or follow some other function? The particular annealing schedule in these experiments was chosen to be nonlinear, lingering at higher temperatures for longer than at lower temperatures, following the relationship

$$\tau(k) = \frac{e_\tau k/K}{1 - k/K + e_\tau} \qquad k \in \{0, \ldots, K\} \,, \tag{6.73}$$

with e_τ set to 0.2 . For any setting of $e_\tau > 0$, the series of temperatures is monotonic and the initial and final temperatures satisfy (6.49):

$$\tau(0) = 0 \,, \qquad \text{and} \qquad \tau(K) = 1 \,. \tag{6.74}$$

For large e_τ, the schedule becomes linear. This is plotted for different values of e_τ in figure 6.13. The particular value of e_τ was chosen to reduce the degree of hysteresis in the annealing ratios, as discussed below.

Hysteresis in the annealing ratios

As presented in section 6.3.5 and algorithm 6.1, the algorithm for computing each and every marginal likelihood ratio in (6.54) did so in a forward manner, carrying over the parameter setting $\boldsymbol{\theta}_{\text{ini}}$ from the calculation of the previous ratio to initialise the sampling procedure for calculating the next ratio. However, whilst it makes sense to move from higher to lower temperatures to avoid local maxima in the posterior in theory, the final estimate of the marginal likelihood is unbiased regardless of the order in which the ratios are tackled. In particular, we can run the AIS algorithm in the *reverse* direction, starting from the posterior and warming the system to the prior, calculating each ratio exactly as before but using the last sample from the lower temperature as an initialisation for the sampling at the next higher temperature in the schedule (note that by doing this we are *not* inverting the fractions appearing in equation (6.54)).

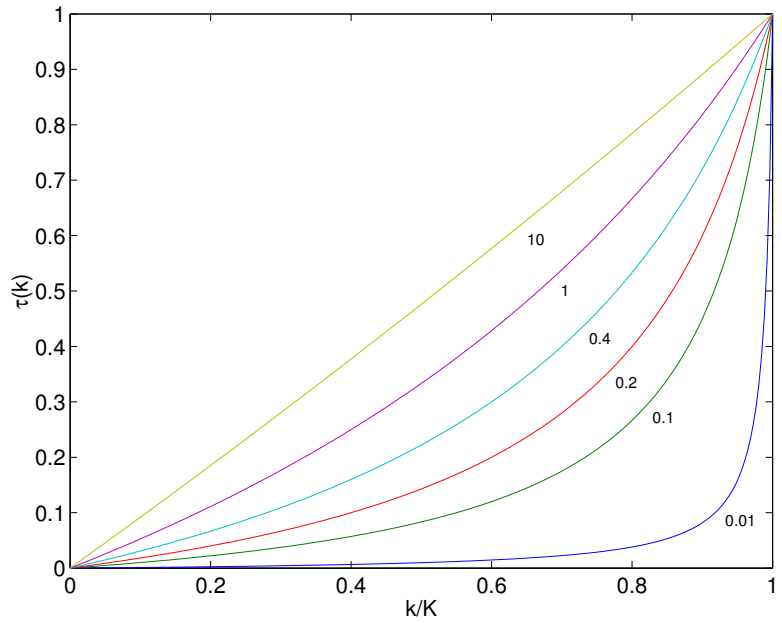

Figure 6.13: Non-linear AIS annealing schedules, plotted for six different values of e_τ. In the experiments performed in this chapter, $e_\tau = 0.2$.

What can this reverse procedure do for us? If we look at figure 6.10 again, we can see that for any random parameter initialisation the reported marginal likelihood is much more often than not an underestimate of the true value. This is because for coarse annealing schedules we are unlikely to locate regions of high posterior probability by the time the system is quenched. If we were then to run the AIS algorithm in a reverse direction, starting from where we had finished the forward pass, we would expect on average to report a higher marginal likelihood than that just reported by the forward pass, simply because the sampler has had longer to explore the high probability regions.

A logical conclusion is that if the forward and reverse passes yield very different values for the marginal likelihood, then we have most likely used too short an annealing schedule. And furthermore, since the marginal likelihood estimates are constructed from the product of many ratios of marginal likelihoods, we can use the discrepancies between the ratios calculated on the forward and reverse passes to choose temperature regions where more sampling is required, and dilate the annealing schedules in these regions accordingly. Of course we should remember that these discrepancies are stochastic quantities, and so we should modify the schedule based on averaged discrepancies over several runs.

This heuristic analysis was used when designing the shape and granularity of the annealing schedule, and we found that more time was required at higher and intermediate temperatures at the expense of lower temperatures. An area of future research is to formalise and more fully investigate this and related arguments. For example, it would be useful to characterise the dependence of the degree of hysteresis along the schedule for different settings of e_τ.

6.5.2 Estimating dimensionalities of the incomplete and complete-data models

The BICp, BIC and CS approximations take the limit of the Laplace approximation as the amount of data tends to infinity, and result in scores that depend on the dimensionalities of the incomplete and complete models, d and d' respectively. In the experiments in this chapter, for BIC d was calculated using a simple counting argument (see equation (6.24) in section 6.3.2), and for CS d and d' were assumed to be equal, which is the assumption made in the original implementation of Cheeseman and Stutz (1996).

In models that have no hidden variables, the value of d required for the BIC approximation can usually be arrived at by adding together the degrees of freedom in each parameter, taking care to take into consideration any parameter degeneracies. However, in models that do have hidden variables the number of free parameters in the incomplete model is much less than that in the complete model. This is because the full effect of each hidden variable cannot always be fully manifest in the functions produced on the observed variables. This situation can be seen in the following discrete example: imagine the model consisting of a single k-valued hidden variable which is the (only) parent of a p-valued observed variable. The naive counting argument would return the complete dimensionality as $d' = (k-1) + (p-1) \times k$. However, the incomplete dimensionality can be no more than $d = (p-1)$, as a model with this many degrees of freedom can exactly model any observed set of counts of the observed variable.

In a general setting, deducing the complete and incomplete model dimensionalities can be complicated (see, for example, Settimi and Smith, 1998; Kočka and Zhang, 2002), since it involves computing the rank of the Jacobian of the transformation for parameters from incomplete to complete models. Geiger et al. (1996) describe a method by which d can be computed in discrete DAGs, by diagonalising the Jacobian symbolically; they also present a theorem that guarantees that a randomised version of the symbolic operation is viable as well. Unfortunately their approach seems difficult to implement efficiently on an arbitrary topology discrete DAG, since both symbolic and randomised versions require diagonalisation. Furthermore it is not clear how, if at all, it can be transferred to DAGs containing continuous variables with arbitrary mappings between the complete and incomplete data models.

For the purposes of this chapter, we have used a simple method to estimate the dimensionalities of each model in our class. It is based on analysing the effect of random perturbations to the model's parameters on the complete and incomplete-data likelihoods. The procedure is presented in algorithm 6.2, and estimates the number of effective dimensions, d and d', by computing the rank of a perturbation matrix. Since the rank operation attempts to find the number of linearly independent rows of the matrices C and C', the random $\boldsymbol{\epsilon}$-perturbations must be small enough such that the change in the log likelihoods are linear with $\boldsymbol{\epsilon}$. Also, the number of samples n should be chosen to be at least as large as the total number of parameters possible in

Algorithm 6.2: $d(m), d'(m)$: To estimate incomplete and complete model parameter dimensionalities.

---

1. For each structure m

   (b) Obtain $\boldsymbol{\theta}_{\text{MAP}}$ using the MAP EM algorithm (section 6.3.1).

   (a) Obtain a representative set of all possible observed data $\{\mathbf{y}_i\}_{i=1}^n$.

   (d) Randomly (spherically) ϵ-perturb $\hat{\boldsymbol{\theta}}_{\text{MAP}}$ R times, to form $\{\hat{\boldsymbol{\theta}}_1, \ldots, \hat{\boldsymbol{\theta}}_R\}$.

   (e) Compute the matrix $C(n \times R) : C_{ir} = \ln p(\mathbf{y}_i \,|\, \hat{\boldsymbol{\theta}}_r)$ for all (i, r).
   Estimate $d(m) = \text{rank}(C) - 1$.

   (f) Compute the matrix $C'(n \times R) : C'_{ir} = \ln p(\mathbf{s}_i, \mathbf{y}_i \,|\, \hat{\boldsymbol{\theta}}_r)$ for all (i, r), where $\mathbf{s}_i$ is a randomly instantiated hidden state.
   Estimate $d'(m) = \text{rank}(C') - 1$.

   End For

the model (as the rank of a matrix can be no more than the smaller of the number of rows or columns), and preferably several times this for reliable estimates.

This procedure was found to give reasonable results when carried out on all the model structures used in this chapter, with a randomly generated data set of size $n = 1000$ and $R = 100$. Without listing all the results, it suffices to say that: for all structures $d \leq d' \leq d+2$, and for the majority of structures $d' = d+|\mathcal{H}|$ — that is to say a further degree of freedom is provided for each binary hidden variable (of which there are at most 2) on top of the incomplete dimensionality. There are some structures for which the discrepancy $d' - d$ is smaller than 2, which is not as we would expect.

There may be several reasons for this discrepancy. First the random perturbations may not have explored certain directions from the MAP estimate, and thus the algorithm could have reported a lower dimensionality than true (unlikely). Second, the data $\mathbf{y}$ only represented a subset of all possible configurations (almost certainly since there are 5^4 possible realisations and 1000 data points are generated randomly), and therefore the effective dimensionality drops.

These results support the use of a more accurate CS† score — see equation (6.30), which modifies the score by adding a term $(d' - d)/2 \cdot \ln n$. The effect of this is to raise the scores for models with 2 hidden variables by $\ln n$, raise those with just 1 hidden variable by $1/2 \cdot \ln n$, and leave unchanged the single model with no hidden states.

Table 6.5 shows the improvement (in terms of ranking) of the more accurate CS† over the original CS approximation, bringing it closer to the performance of the VB score. The table shows the number of times in the 106 samples (see experiments in section 6.4 above) that the

n	BIC	BICp	CS	CS†	VB
10	0	0	0	0	0
20	0	0	0	0	0
40	0	0	0	0	0
80	0	0	0	1	1
110	0	0	0	0	1
160	0	0	1	2	3
230	0	1	3	5	6
320	0	2	8	10	12
400	1	5	8	9	11
430	1	6	10	10	11
480	3	7	12	12	15
560	3	8	14	16	18
640	5	11	14	17	23
800	7	15	22	23	29
960	9	18	28	33	36
1120	11	19	32	33	40
1280	15	24	38	41	48
2560	35	41	59	62	66
5120	56	63	76	76	80
10240	73	79	82	83	84

Table 6.5: Number of times (out of 106) that each score selects the true structure. Shown are the performance of the original BIC, BICp, CS and VB scores, all corrected for aliasing, and also shown is the CS† score, resulting from (further) correcting CS for the difference between complete and incomplete data model dimensionalities.

score successfully selected the true model structure. Is it clear that CS† is an improvement over CS, suggesting that the assumption made above is true. However, we should interpret this experiment with some care, because our original choice of the true model having two hidden variables may be masking a bias in the altered score; it would make sense to perform similar experiments choosing a much simpler model to generate the data.

The improvement in performance of the CS† score, averaged over all data set sizes and all 106 generated parameter sets can be see in table 6.2 (page 233), where it is compared alongside BIC, CS and VB. It can be seen that VB still performs better. Further verification of this result will be left to future work.

6.6 Summary

In this chapter we have presented various scoring methods for approximating the marginal likelihood of discrete directed graphical models with hidden variables. We presented EM algorithms for ML and MAP parameter estimation, showed how to calculate the asymptotic criteria of BIC and Cheeseman-Stutz, derived the VBEM algorithm for approximate Bayesian learning which

maintains distributions over the parameters of the model and has the same complexity as the EM algorithm, and presented a (somewhat impoverished) AIS method designed for discrete-variable DAGs.

We have shown that VB consistently outperforms BIC and CS, and that VB performs respectively as well as and more reliably than AIS for intermediate and large sizes of data. The AIS method has very many parameters to tune and requires extensive knowledge of the model domain to design efficient and reliable sampling schemes and annealing schedules. VB on the other hand has not a single parameter to set or tune, and can be applied without any expert knowledge, at least in the class of singly-connected discrete-variable DAGs with Dirichlet priors which we have considered in this chapter. Section 6.5.1 discussed several ways in which the AIS method could be improved, for example by better matching the Metropolis-Hastings proposal distributions to the annealed posterior; in fact a method based on slice sampling should be able to adapt better to the annealing posterior with little or no expert knowledge of the shape of the annealed posterior (Neal, 2003).

It may be that there exists a better AIS scheme than sampling in parameter space. To be more specific, for any completion of the data the parameters of the model can be integrated out tractably (at least for the class of models examined in this chapter); thus an AIS scheme which anneals in the space of completions of the data may be more efficient than the current scheme which anneals in the space of parameters (personal communication with R. Neal). However, this latter scheme may only be efficient for models with little data compared to the number of parameters, as the sampling space of all completions increases linearly with the amount of data. This avenue of research is left to further work.

This chapter has presented a novel application of variational Bayesian methods to discrete DAGs. In the literature there have been other attempts to solve this long-standing model selection problem. For example the *structural EM* algorithm of Friedman (1998) uses a structure search algorithm which uses a scoring algorithm very similar to the VBEM algorithm presented here, except that for tractability the distribution over $\boldsymbol{\theta}$ is replaced by the MAP estimate, $\boldsymbol{\theta}_{\text{MAP}}$. We have shown here how the VB framework enables us to use the entire distribution over $\boldsymbol{\theta}$ for inference of the hidden variables.

In chapter 2 we proved that the Cheeseman-Stutz score is in fact a lower bound on the marginal likelihood and, more importantly, we proved that there exists a construction which is guaranteed to produce a variational Bayesian lower bound that is *at least as tight* as the Cheeseman-Stutz score (corollary 2.5 to theorem 2.3, page 79). This construction builds a variational Bayesian approximation using the same MAP parameter estimate used to obtain the CS score. However, we did not use this construction in our experiments, and ran both the MAP EM and VB optimisations independently of each other. As a result we cannot guarantee that the VB bound is in all runs tighter than the CS bound, as the dynamics of the optimisations for MAP learning

and VB learning may in general lead even identically initialised algorithms to different optima in parameter space (or parameter distribution space). Nevertheless we have still seen improvement in terms of ranking of the true structure by VB as compared to CS. A tighter bound on the marginal likelihood does not necessarily directly imply that we should have better structure determination, although it certainly suggests this and is supported by the experimental results. Empirically, the reader may be interested to know that the VB lower bound was observed to be *lower* than the CS score in only 173 of the 288320 total scores calculated (about 0.06%). If the construction derived in corollary 2.5 had been used then this number of times would of course be exactly zero.