Chapter 4

MONTE CARLO METHODS AND MCMC SIMULATION

In this chapter we treat the theory required for learning about posterior distributions, and we lay the foundation needed in order to learn BNs from incomplete data. We also develop a Markov chain Monte Carlo sampler called MB-MCMC for obtaining DAG models from the posterior model distribution given complete data.

Monte Carlo methods is a broad topic, and in this chapter we only present the essentials. See for instance Robert and Casella, 2002; Gilks et al., 1996 for a thorough treatment of different Monte Carlo techniques, including Markov chain Monte Carlo simulation and practical applications.

1. Monte Carlo methods

We investigate the generic problem of computing the following summation (replace sums with integrals for continuous variables):

$$\mathrm{E}_{\Pr}[h(\boldsymbol{X})] = \sum_{\boldsymbol{x}} h(\boldsymbol{x}) \Pr(\boldsymbol{x}) \tag{4.1}$$

In particular, $h(\boldsymbol{X})$ can be a probability distribution, for instance the conditional distribution $\Pr(\boldsymbol{Y}|\boldsymbol{X})$, in which case the expectation coincides with the distribution $\Pr(\boldsymbol{Y})$.

If eq. 4.1 is difficult to solve analytically or is infeasible to compute because of an extremely large cardinality of $\Omega_{\boldsymbol{X}}$, it can be approximated. Sample $\boldsymbol{X}^{(1)}, \ldots, \boldsymbol{X}^{(n)}$ from $\Pr(\boldsymbol{X})$, i.e., $\boldsymbol{X}^{(t)} \sim \Pr(\boldsymbol{X})$, and compute

the *empirical average* of eq. 4.1:

$$\mathrm{E}_{\Pr}[h(\boldsymbol{X})] \approx \frac{1}{n}\sum_{t=1}^{n} h(\boldsymbol{x}^{(t)}) \tag{4.2}$$

The Law of Large Numbers guarantees that the approximation converges towards the expectation in eq. 4.1 as $n \to \infty$.

For one reason or another it might be the case that sampling from $\Pr(\boldsymbol{X})$ is undesirable. For instance, it may be computationally expensive to produce realisations, or the distribution may be such, that it can be evaluated only up to a normalising term, making it difficult or impossible to sample from. Many computational problems encountered in Bayesian statistics in fact boil down to not being able to determine the normalising term, as we will see in section 2. This means that even solving the approximation of eq. 4.1 in terms of eq. 4.2 may be problematic.

In order to tackle these problems, more sophisticated *Monte Carlo* techniques may help out. We start off by introducing another Monte Carlo method called importance sampling, and then discuss Markov chain Monte Carlo sampling.

1.1 Importance sampling

A slightly more advanced Monte Carlo method than the approximation in eq. 4.2 is the so-called *importance sampler*. Suppose that we don't sample from $\Pr(\boldsymbol{X})$, but instead we sample from another distribution $\Pr'(\boldsymbol{X})$, e.g., it may be computationally cheap to do so.

First off, rewrite the summation in eq. 4.1:

$$\begin{aligned}
\sum_{\boldsymbol{x}} h(\boldsymbol{x})\Pr(\boldsymbol{x}) &= \frac{1}{\sum_{\boldsymbol{x}} \Pr(\boldsymbol{x})} \sum_{\boldsymbol{x}} h(\boldsymbol{x})\Pr(\boldsymbol{x}) \\
&= \frac{1}{\sum_{\boldsymbol{x}} \frac{\Pr(\boldsymbol{x})}{\Pr'(\boldsymbol{x})}\Pr'(\boldsymbol{x})} \sum_{\boldsymbol{x}} h(\boldsymbol{x})\frac{\Pr(\boldsymbol{x})}{\Pr'(\boldsymbol{x})}\Pr'(\boldsymbol{x}) \\
&= \frac{1}{\mathrm{E}_{\Pr'}[\frac{\Pr(\boldsymbol{X})}{\Pr'(\boldsymbol{X})}]} \mathrm{E}_{\Pr'}[h(\boldsymbol{X})\frac{\Pr(\boldsymbol{X})}{\Pr'(\boldsymbol{X})}]
\end{aligned} \tag{4.3}$$

Sample $\boldsymbol{X}^{(1)}, \ldots, \boldsymbol{X}^{(n)}$ from $\Pr'(\boldsymbol{X})$, and the empirical average of eq. 4.3 becomes:

$$\frac{1}{\sum_{t=1}^{n} w_t} \cdot \sum_{t=1}^{n} w_t \cdot h(\boldsymbol{x}^{(t)}) \tag{4.4}$$

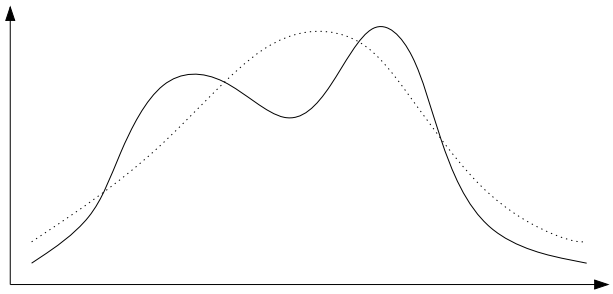

Figure 4.1. The dotted line is $\Pr'(\boldsymbol{X})$, and the solid line is $\Pr(\boldsymbol{X})$.

with weights w_t:

$$w_t = \frac{\Pr(\boldsymbol{x}^{(t)})}{\Pr'(\boldsymbol{x}^{(t)})} \tag{4.5}$$

When $\Pr'(\boldsymbol{X}) > 0$ whenever $\Pr(\boldsymbol{X}) > 0$, it holds that as n increases the approximation becomes more accurate. An essential observation is that the *constant* normalising term of $\Pr(\boldsymbol{X})$ cancels out because the weights as given in eq. 4.5 are normalised in eq. 4.4. This implies that we need only be able to evaluate $\Pr(\boldsymbol{X})$ up to this normalising constant. In fact, the normalising term of $\Pr'(\boldsymbol{X})$ is eliminated as well.

Theoretically speaking, importance sampling puts very little restriction on the choice of sampling distribution; in particular, any strictly positive sampling distribution can be used. When using a uniform sampling distribution, the denominator of w_t is the same for all weights t, and are eliminated by normalisation. Also note that when $\Pr'(\boldsymbol{X})$ and $\Pr(\boldsymbol{X})$ are proportional, the sampler reduces to the empirical average in eq. 4.2.

The weights w_t in eq. 4.5 are called *importance weights* and compensate for the difference between the real distribution and the sampling distribution. Intuitively, for the standard empirical average in eq. 4.2, all samples drawn from $\Pr(\boldsymbol{X})$ have the same weight in the approximation. For importance sampling, any sample from $\Pr'(\boldsymbol{X})$ is, compared to $\Pr(\boldsymbol{X})$, either "over-sampled" (too frequently sampled) and receives a weight less than 1, or "under-sampled" (too infrequently sampled) and receives a weight larger than 1. Figure 4.1 illustrates the principle behind importance sampling. The dotted line is the sampling distribution, and the solid line is from the target distribution, from which we can't sample. Samples drawn at locations where the dotted line lies above the

solid plot, will be drawn more often than necessary, and vice versa. To correct for that mismatch, the importance weights are required.

1.1.1 Choice of the sampling distribution

Although for large n the importance sampling approximation will be good, the sampling distribution has a major impact on the performance of importance sampling. In fact, choosing an inappropriate sampling distribution can have disastrous effects (see for instance Geweke, 1989). The rewrite of the expectation in terms of $\Pr'(\boldsymbol{X})$ results in the variance:

$$\begin{aligned}
\mathrm{Var}_{\Pr'}[h(\boldsymbol{X})\frac{\Pr(\boldsymbol{X})}{\Pr'(\boldsymbol{X})}] &= \mathrm{E}_{\Pr'}[h(\boldsymbol{X})^2\frac{\Pr(\boldsymbol{X})^2}{\Pr'(\boldsymbol{X})^2}] - \mathrm{E}_{\Pr'}[h(\boldsymbol{X})\frac{\Pr(\boldsymbol{X})}{\Pr'(\boldsymbol{X})}]^2 \\
&= \sum_{\boldsymbol{x}} h(\boldsymbol{x})^2\frac{\Pr(\boldsymbol{x})^2}{\Pr'(\boldsymbol{x})^2}\Pr'(\boldsymbol{x}) - \mathrm{E}_{\Pr}[h(\boldsymbol{X})]^2 \\
&= \sum_{\boldsymbol{x}} h(\boldsymbol{x})^2\frac{\Pr(\boldsymbol{x})}{\Pr'(\boldsymbol{x})}\Pr(\boldsymbol{x}) - \mathrm{E}_{\Pr}[h(\boldsymbol{X})]^2 \\
&= \mathrm{E}_{\Pr}[h(\boldsymbol{X})^2\frac{\Pr(\boldsymbol{X})}{\Pr'(\boldsymbol{X})}] - \mathrm{E}_{\Pr}[h(\boldsymbol{X})]^2 \\
&= \sum_{\boldsymbol{x}} h(\boldsymbol{x})^2\frac{\Pr(\boldsymbol{x})^2}{\Pr'(\boldsymbol{x})} - \Big(\sum_{\boldsymbol{x}} h(\boldsymbol{X})\Pr(\boldsymbol{X})\Big)^2 \quad (4.6)
\end{aligned}$$

The second term in eq. 4.6 is independent of $\Pr'(\boldsymbol{X})$, so our choice of $\Pr'(\cdot)$ only affects the first term. Assuming that we want to be able to use a wide range of functions $h(\boldsymbol{X})$ that we don't know a priori, we restrict attention to the effect that the ratio $\Pr(\boldsymbol{X})^2/\Pr'(\boldsymbol{X})$ has on the variance in the first term. When this fraction is unbounded, the variance for many functions is infinite. This leads to general instability and slows convergence. Notice that the ratio becomes extremely large in the tails when $\Pr(\boldsymbol{X})$ is larger than $\Pr'(\boldsymbol{X})$ in that region. A bounded ratio is the best choice, and in particular, in the tails $\Pr'(\boldsymbol{X})$ should dominate $\Pr(\boldsymbol{X})$.

Suppose a single $\boldsymbol{x}^{(t)}$ is drawn from $\Pr'(\boldsymbol{X})$ from an area of very low probability (density), and $\Pr(\boldsymbol{x}^{(t)}) \gg \Pr'(\boldsymbol{x}^{(t)})$. Such a sample can have a major impact on the empirical average via importance sampling. The sample is assigned far too much importance compared to the remaining samples because the ratio $\Pr(\boldsymbol{x}^{(t)})/\Pr'(\boldsymbol{x}^{(t)})$ is very large. Now suppose that $\Pr'(\boldsymbol{X})$ is a reasonable approximation of $\Pr(\boldsymbol{X})$ *almost everywhere* except in a few areas, where the importance weights are off-scale. Even though the majority of samples contribute to a reasonable approximation

of the expectation, as soon as a sample is obtained from "a bad area", the approximation seriously deteriorates because the importance weight is so much larger compared to the importance weights associated with the samples from the "good areas". In such a case, it may be better to discard such a sample entirely. This should be done with some caution though. Deletion will generally not introduce bias if the large weight is due to a very small denominator (compared to the denominator of the other weights). If it turns out that the deletion of a large-weight proposal results in a more sensible mass distribution over the remaining sample proposals, then this indeed does indicate that the sample just deleted was "an accidental outlier" and can be deleted without problem. On the other hand, if the numerator in a large-weight sample is large (compared to the numerator of the other weights), one may want to keep such a sample, since it then comes from a "region of relatively high impact" on the empirical approximation. In Hesterberg, 1995 it is suggested to use a mixture distribution as proposal distribution to overcome the problem pertaining to large importance weights. Each component of $\Pr'(\boldsymbol{X})$ is $\Pr'_i(\boldsymbol{X})$ with weight v_i. From an operational point of view, drawing $\boldsymbol{X}^{(t)}$ from $\Pr'(\boldsymbol{X})$ means that with probability v_i we draw from $\Pr'_i(\boldsymbol{X})$. In calculating the importance weights however, we use $\Pr'(\boldsymbol{x}^{(t)})$ rather than $\Pr'_i(\boldsymbol{x}^{(t)})$. This way we may define distributions $\Pr'_i(\boldsymbol{X})$ that cover different areas of $\Pr(\boldsymbol{X})$, such that the importance weights remain bounded.

A general recommendation is to monitor the running variance of the importance weights because it gives a good indication of the mutual differences between the importance weights of the proposals sampled from $\Pr'(\boldsymbol{X})$. When the variance suddenly jumps or is instable, one should at least investigate if the sampling distribution is appropriate, or if deletion of a few samples may correct the problem. Large variance may indicate that the performance of the sampler is poor and that the approximation will be sub-optimal.

2. Markov chain Monte Carlo—MCMC

Importance sampling offers a way of approximating $\mathrm{E}[h(\boldsymbol{X})]$ with respect to a distribution $\Pr(\boldsymbol{X})$ only known up to the normalising term, via some sampling distribution $\Pr'(\boldsymbol{X})$. Markov chain Monte Carlo (MCMC) is a whole range of methods that provide an alternative way of solving the generic problem of approximating the expectation in eq. 4.1.

Importance sampling is a so-called direct sampling method where at least partial knowledge of the distribution from which samples are re-

quired is needed to get a reasonable approximation. These methods are by nature non-iterative, and they do not adapt to the target distribution. MCMC on the other hand is a more flexible and iterative method for handling very awkward distributions. See Neal, 1993 and Andrieu et al., 2003 for a walkthrough of basic MCMC theory.

The problems that we want to solve are usually cast in a Bayesian framework, and we will therefore slightly reformulate the original problem. Consider Bayes' law:

$$\Pr(\boldsymbol{X}|\boldsymbol{Y}) = \frac{\Pr(\boldsymbol{Y}|\boldsymbol{X})\Pr(\boldsymbol{X})}{\sum_{\boldsymbol{x}} \Pr(\boldsymbol{Y}|\boldsymbol{x})\Pr(\boldsymbol{x})} \tag{4.7}$$

The denominator $\Pr(\boldsymbol{Y}) = \sum_{\boldsymbol{x}} \Pr(\boldsymbol{Y}|\boldsymbol{x})\Pr(\boldsymbol{x})$ is responsible for normalisation. Hence, if the numerator can be computed, then sampling from $\Pr(\boldsymbol{X}|\boldsymbol{Y})$, and for instance approximating $\mathrm{E}[h(\boldsymbol{X})]$ with respect to that distribution, can be solved via importance sampling or MCMC. In general $h(\boldsymbol{X})$ may be any function defined on $\Omega_{\boldsymbol{X}}$. Henceforth we leave the function $h(\boldsymbol{X})$ out of the picture, and address the main problem, namely sampling from $\Pr(\boldsymbol{X}|\boldsymbol{Y})$. Sometimes we skip the conditional, and the problem is thus how to sample from $\Pr(\boldsymbol{X})$, when this distribution is known up to a normalising term.

2.1 Markov chains

MCMC is based on the construction of a *Markov chain*, where the $\boldsymbol{X}^{(t)}$'s are produced sequentially, beginning from $\boldsymbol{X}^{(0)}$, such that $\boldsymbol{X}^{(t+1)}$ depends on $\boldsymbol{X}^{(t)}$ only, but is independent of $\boldsymbol{X}^{(t-1)}, \boldsymbol{X}^{(t-2)}, \ldots, \boldsymbol{X}^{(0)}$ (a so-called *one-step memory* sequence):

$$\boldsymbol{X}^{(t+1)} \perp\!\!\!\perp \boldsymbol{X}^{(0)}, \ldots, \boldsymbol{X}^{(t-1)} | \boldsymbol{X}^{(t)}$$

The chain is constructed via *transition probabilities* $T(\boldsymbol{X}^{(t+1)}|\boldsymbol{X}^{(t)})$ corresponding to a conditional distribution for $\boldsymbol{X}^{(t+1)}$ given $\boldsymbol{X}^{(t)}$, and an initial distribution $\Pr_0(\boldsymbol{X}^{(0)})$. The distribution $\Pr_{t+1}(\boldsymbol{X}^{(t+1)})$ is then defined in terms of $\boldsymbol{X}^{(t)}$ via the transition:

$$\Pr_{t+1}(\boldsymbol{X}^{(t+1)}) = \sum_{\boldsymbol{x}^{(t)}} T(\boldsymbol{X}^{(t+1)}|\boldsymbol{x}^{(t)}) \Pr_t(\boldsymbol{x}^{(t)})$$

When the transition probabilities as defined here do not depend on t, the Markov chain is called (time) *homogeneous*.

2.1.1 The invariant target distribution

An *invariant distribution* $\Pr_*(\boldsymbol{X})$ for a Markov chain, is one that persists forever once reached:

$$\Pr_*(\boldsymbol{X}) = \sum_{\boldsymbol{x}'} T(\boldsymbol{X}|\boldsymbol{x}')\Pr_*(\boldsymbol{x}') \tag{4.8}$$

The invariant distribution, is the *target distribution*, i.e., the distribution from which we want to draw samples. An invariant distribution exists if the Markov chain satisfies the *detailed balance* condition. Detailed balance states that the transition probability is symmetric between $\boldsymbol{X}$ and $\boldsymbol{X}'$, when they are sampled from a distribution $\Pr(\boldsymbol{X})$. Formally, it should hold that:

$$T(\boldsymbol{X}'|\boldsymbol{X})\Pr(\boldsymbol{X}) = T(\boldsymbol{X}|\boldsymbol{X}')\Pr(\boldsymbol{X}') \tag{4.9}$$

In fact, this means that $\Pr(\boldsymbol{X}) = \Pr_*(\boldsymbol{X})$, i.e., the distribution must necessarily be an invariant distribution viz. eq. 4.8:

$$\begin{aligned}\sum_{\boldsymbol{x}'} T(\boldsymbol{X}|\boldsymbol{x}')\Pr(\boldsymbol{x}') &= \sum_{\boldsymbol{x}'} T(\boldsymbol{x}'|\boldsymbol{X})\Pr(\boldsymbol{X}) \\ &= \Pr(\boldsymbol{X})\end{aligned}$$

We stress that detailed balance is a sufficient but not necessary condition for ensuring invariance.

2.1.2 Reaching the invariant distribution

To guarantee that the invariant distribution is reached, the Markov chain needs to be *irreducible* and *aperiodic* (Tierney, 1994). The chain is irreducible if starting from $\boldsymbol{X}^{(0)}$ every region of the space from which we need to sample can eventually be reached with a positive probability. In figure 4.2, left, this is illustrated. The outline of the whole figure illustrates the actual state space, and the gray area corresponds to the area explored by an irreducible Markov chain. No matter where $\boldsymbol{X}^{(0)}$ is located, all states *communicate* and can be reached; no region is isolated and the chain is free to move to other states via other intermediate states. To the right, the chain is *reducible*. Again, the whole figure illustrates the state space from which samples are required. The two gray areas are not connected, indicating that only the region from which the initial $\boldsymbol{X}^{(0)}$ is located is explored. The whole state space from which we wish to sample is not covered because there is no way to reach the other region.

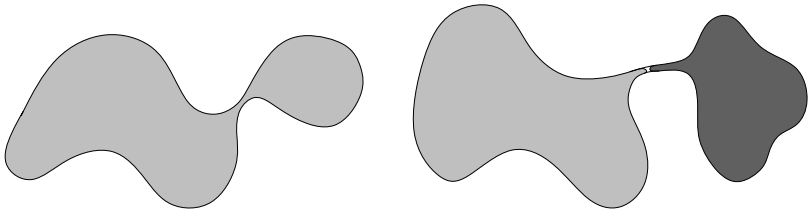

Figure 4.2. State spaces explored by the Markov chain. *Left:* irreducible. *Right:* reducible.

Aperiodicity guarantees that the chain does not get trapped in cycles, preventing the chain from getting the opportunity to reach other areas. It is quite easy to make sure that this does not happen; as long as there is a strictly positive probability of remaining in the current state, i.e., the probability of $\boldsymbol{X}^{(t)}$ and $\boldsymbol{X}^{(t+1)}$ having the same value is > 0, the chain will not get stuck in cycles.

To sum up, if a homogeneous Markov chain...

- has an *invariant* distribution, $\Pr(\boldsymbol{X})$,
- it is *irreducible*, and
- it is *aperiodic*,

the Markov chain will in the limit produce realisations from the invariant distribution, regardless of the starting point $\boldsymbol{X}^{(0)}$:

$$|\Pr_{t+1}(\boldsymbol{X}) - \Pr(\boldsymbol{X})| \to \mathbf{0} \text{ for } t \to \infty$$

Such a Markov chain is said to be *ergodic*.

In the next two sections we discuss two well-known MCMC methods that produce samples from some *desired* invariant distribution by building an ergodic Markov chain: the Metropolis-Hastings sampler, and the Gibbs sampler. Sometimes we say that a MCMC sampler is ergodic, in which case it refers to the Markov chain that is produced by the sampler.

2.1.3 Metropolis-Hastings sampling

Metropolis-Hastings MCMC (Metropolis et al., 1953; Hastings, 1970) by construction produces samples from the distribution, $\Pr(\boldsymbol{X})$, which is the invariant distribution for the Markov chain. Quite similar to importance sampling, a *proposal* distribution, $\Pr'(\boldsymbol{Y}|\boldsymbol{X})$, from which

we can sample exists. The Metropolis-Hastings algorithm produces a Markov chain $\boldsymbol{X}^{(0)}, \boldsymbol{X}^{(1)}, \ldots$ in the following way:

---

1 Draw $\boldsymbol{Y} \sim \Pr'(\boldsymbol{Y}|\boldsymbol{X}^{(t)})$

2 Take

$$\boldsymbol{X}^{(t+1)} = \begin{cases} \boldsymbol{y} & \text{with probability } \rho(\boldsymbol{x}^{(t)}, \boldsymbol{y}) \\ \boldsymbol{x}^{(t)} & \text{with probability } 1 - \rho(\boldsymbol{x}^{(t)}, \boldsymbol{y}) \end{cases}$$

where

$$\rho(\boldsymbol{X}, \boldsymbol{Y}) = \min\left\{ \frac{\Pr(\boldsymbol{Y}) \Pr'(\boldsymbol{X}|\boldsymbol{Y})}{\Pr(\boldsymbol{X}) \Pr'(\boldsymbol{Y}|\boldsymbol{X})}, 1 \right\}$$

The transition probability is the probability of proposing $\boldsymbol{Y}$ times the probability of accepting this candidate as $\boldsymbol{X}^{(t+1)}$; additionally, it includes the probability of proposing any state $\boldsymbol{X}'$ and rejecting it, remaining in the current state, $\boldsymbol{Y} = \boldsymbol{X}^{(t)}$. Thus the transition $T(\boldsymbol{Y}|\boldsymbol{X}^{(t)})$ becomes:

$$\rho(\boldsymbol{X}^{(t)}, \boldsymbol{Y}) \Pr'(\boldsymbol{Y}|\boldsymbol{X}^{(t)}) + I(\boldsymbol{Y} = \boldsymbol{X}^{(t)}) \sum_{\boldsymbol{x}'} (1 - \rho(\boldsymbol{X}^{(t)}, \boldsymbol{x}')) \Pr'(\boldsymbol{x}'|\boldsymbol{X}^{(t)})$$

where $I(\cdot)$ is the indicator function. We indeed have that $\Pr(\boldsymbol{X})$ is the invariant distribution, because detailed balance holds. For the second term this is easy to see. For the first term we distinguish cases. In the case $\frac{\Pr(\boldsymbol{Y}) \Pr'(\boldsymbol{X}|\boldsymbol{Y})}{\Pr(\boldsymbol{X}) \Pr'(\boldsymbol{Y}|\boldsymbol{X})} > 1$, we have that $\rho(\boldsymbol{X}, \boldsymbol{Y}) = 1$, and by applying eq. 4.9 it follows:

$$\begin{aligned} \Pr(\boldsymbol{X}) \rho(\boldsymbol{X}, \boldsymbol{Y}) \Pr'(\boldsymbol{Y}|\boldsymbol{X}) &= \Pr(\boldsymbol{X}) \Pr'(\boldsymbol{Y}|\boldsymbol{X}) \\ &= \Pr(\boldsymbol{X}) \Pr'(\boldsymbol{Y}|\boldsymbol{X}) \frac{\Pr(\boldsymbol{Y}) \Pr(\boldsymbol{X}|\boldsymbol{Y})}{\Pr(\boldsymbol{Y}) \Pr(\boldsymbol{X}|\boldsymbol{Y})} \\ &= \Pr(\boldsymbol{Y}) \Pr'(\boldsymbol{X}|\boldsymbol{Y}) \frac{\Pr(\boldsymbol{X}) \Pr(\boldsymbol{Y}|\boldsymbol{X})}{\Pr(\boldsymbol{Y}) \Pr(\boldsymbol{X}|\boldsymbol{Y})} \\ &= \Pr(\boldsymbol{Y}) \rho(\boldsymbol{Y}, \boldsymbol{X}) \Pr'(\boldsymbol{X}|\boldsymbol{Y}) \end{aligned}$$

In case $\frac{\Pr(\boldsymbol{Y}) \Pr'(\boldsymbol{X}|\boldsymbol{Y})}{\Pr(\boldsymbol{X}) \Pr'(\boldsymbol{Y}|\boldsymbol{X})} < 1$ we have $\rho(\boldsymbol{X}, \boldsymbol{Y}) = \frac{\Pr(\boldsymbol{Y}) \Pr'(\boldsymbol{X}|\boldsymbol{Y})}{\Pr(\boldsymbol{X}) \Pr'(\boldsymbol{Y}|\boldsymbol{X})}$ and $\rho(\boldsymbol{Y}, \boldsymbol{X}) = 1$, and it follows:

$$\Pr(\boldsymbol{X}) \rho(\boldsymbol{X}, \boldsymbol{Y}) \Pr'(\boldsymbol{Y}|\boldsymbol{X}) = \Pr(\boldsymbol{X}) \Pr'(\boldsymbol{Y}|\boldsymbol{X}) \frac{\Pr(\boldsymbol{Y}) \Pr'(\boldsymbol{X}|\boldsymbol{Y})}{\Pr(\boldsymbol{X}) \Pr'(\boldsymbol{Y}|\boldsymbol{X})}$$

$$\begin{aligned} &= \Pr(\boldsymbol{Y})\Pr'(\boldsymbol{X}|\boldsymbol{Y}) \\ &= \Pr(\boldsymbol{Y})\Pr'(\boldsymbol{X}|\boldsymbol{Y})\rho(\boldsymbol{Y},\boldsymbol{X}) \end{aligned}$$

Hence, the Markov chain has invariant distribution $\Pr(\boldsymbol{X})$.

The Metropolis-Hastings sampler depends on the ratios $\Pr(\boldsymbol{Y})/\Pr(\boldsymbol{X})$ and $\Pr'(\boldsymbol{X}|\boldsymbol{Y})/\Pr'(\boldsymbol{Y}|\boldsymbol{X})$, implying that the normalising term for both the proposal distribution and, more importantly, for $\Pr(\cdot)$ cancels out.

Irreducibility is guaranteed when $\Pr'(\boldsymbol{Y}|\boldsymbol{X})$ covers the entire sampling area on which the invariant distribution $\Pr(\boldsymbol{X})$ is defined, since then any point can be reached via a proposal. Aperiodicity is also guaranteed, since there is a non-zero probability of remaining in the current state. If the proposal distribution is strictly positive this means that the invariant distribution will be reached in the limit. Notice that this is quite similar to importance sampling, where a sufficient condition for convergence was that the sampling distribution was positive.

A special case of the Metropolis-Hastings MCMC occurs when we have that $\Pr'(\boldsymbol{Y}|\boldsymbol{X}) = \Pr'(\boldsymbol{Y})$, i.e., the proposal distribution is independent of the last state like in importance sampling. This leads to:

$$\rho(\boldsymbol{X},\boldsymbol{Y}) = \min\Big\{\frac{\Pr(\boldsymbol{Y})\Pr'(\boldsymbol{X})}{\Pr(\boldsymbol{X})\Pr'(\boldsymbol{Y})},1\Big\} = \min\Big\{\frac{\Pr(\boldsymbol{Y})/\Pr'(\boldsymbol{Y})}{\Pr(\boldsymbol{X})/\Pr'(\boldsymbol{X})},1\Big\} \quad (4.10)$$

The numerator and the denominator also arise in importance sampling as the importance weights. With importance sampling the samples are weighted in order to correct for the difference between $\Pr(\boldsymbol{X})$ and $\Pr'(\boldsymbol{X})$, whereas for Metropolis-Hastings MCMC, the unweighted samples can be used, as they provably do come from $\Pr(\boldsymbol{X})$ (at least in the limit), and the weights are not required in the direct sense. However, indirectly the weights are still needed for the ratio given in eq. 4.10.

2.1.4 Gibbs sampling

Gibbs sampling (Geman and Geman, 1984) can be seen as special case of Metropolis-Hastings sampling, where the transition probabilities are defined in terms of the conditionals of the invariant distribution. Often we are able to draw from these conditionals, without being able to draw from the joint invariant distribution.

The Gibbs sampler we present is defined in terms of *blocks* of variables. Each block is drawn conditional on the variables not part of the current block. Our presentation deviates from the customary treatment of Gibbs sampling. The reason is that the "normal" Gibbs sampler can

be regarded as a special case of block Gibbs sampling, and that the added value of blocking is beneficial in many respects (Liu, 1994; Liu et al., 1994). We illustrate this in Section 2.1.6, and in Section 3.3 we exploit blocking for learning DAG models.

By construction, the Gibbs sampler needs at least two blocks. We write $\boldsymbol{X} = \boldsymbol{Y} \cup \boldsymbol{Z}$, indicating that we partition $\boldsymbol{X}$ into two blocks: block one, $\boldsymbol{Y}$ and block two, $\boldsymbol{Z}$. For ease of exposition, we assume that the blocks are disjoint, but in general they do not need to be. The proposal only changes block $\boldsymbol{Y}$:

$$\boldsymbol{Y} \sim \Pr'(\boldsymbol{Y}|\boldsymbol{Z}) = \Pr(\boldsymbol{Y}|\boldsymbol{Z})$$

where the conditional side $\boldsymbol{Z}$ is unchanged between moves, and unconditional on the last state of block $\boldsymbol{Y}$. Again, this is similar to drawing from the sampling distribution in importance sampling.

The relationship between the proposal distribution and the invariant distribution is constant through the ratio:

$$\Pr(\boldsymbol{Z}) = \frac{\Pr(\boldsymbol{Y}, \boldsymbol{Z})}{\Pr(\boldsymbol{Y}|\boldsymbol{Z})} = \frac{\Pr(\boldsymbol{Y}, \boldsymbol{Z})}{\Pr'(\boldsymbol{Y}|\boldsymbol{Z})}$$

Similar to eq. 4.10, the acceptance ratio becomes:

$$\rho(\boldsymbol{U}, \boldsymbol{Y}) = \min\left\{\frac{\Pr(\boldsymbol{Y}, \boldsymbol{Z})/\Pr'(\boldsymbol{Y}|\boldsymbol{Z})}{\Pr(\boldsymbol{U}, \boldsymbol{Z})/\Pr'(\boldsymbol{U}|\boldsymbol{Z})}, 1\right\} = 1$$

yielding an acceptance rate of 1, meaning that all proposals are accepted.

Obviously only sampling $\boldsymbol{Y}$ means that the Markov chain can't be irreducible, since the proposal distribution only proposes changes to one block. By only sampling in this lower-dimensional space, it immediately follows that any point not in that dimension will remain fixed—it can't be reached. A minimal conditional for ensuring irreducibility is to propose changes to block $\boldsymbol{Z}$ as well, i.e., sample $\boldsymbol{Z}$ from the conditional. Formally speaking, this is achieved by combining several Gibbs samplers, one per block. This amounts to applying transitions in turn, one transition per block. The chain remains invariant because each separate block transition leaves the chain invariant. To see why this is, suppose that we start the sampler from the invariant distribution. Each block is now sampled from the conditional of the invariant distribution. This transition leaves the marginal distribution (that coincides with the marginal of the invariant distribution) of the other blocks intact. For the block that is sampled, the transition obviously also leaves the chain invariant.

This argument holds for all the blocks and therefore we may conclude that once the invariant distribution has been reached, a combination of block transitions leaves the chain invariant.

The Gibbs sampler is usually presented in the following way:

Assign $X_i \in \boldsymbol{X}$ to some block $\boldsymbol{B}_j, j = 1, \ldots, k$, such that X_i is *part of at least one block*. Let $\Pr(\boldsymbol{B}_j | \boldsymbol{B}_{r \neq j})$ be the conditional invariant distribution:

$$
\begin{aligned}
\boldsymbol{B}_1^{(t+1)} &\sim \Pr(\boldsymbol{B}_1 | \boldsymbol{b}_2^{(t)}, \ldots, \boldsymbol{b}_k^{(t)}) \\
\boldsymbol{B}_2^{(t+1)} &\sim \Pr(\boldsymbol{B}_2 | \boldsymbol{b}_1^{(t+1)}, \boldsymbol{b}_3^{(t)}, \ldots, \boldsymbol{b}_k^{(t)}) \\
&\vdots \\
\boldsymbol{B}_k^{(t+1)} &\sim \Pr(\boldsymbol{B}_2 | \boldsymbol{b}_1^{(t+1)}, \ldots, \boldsymbol{b}_{k-1}^{(t+1)}) \\
\boldsymbol{B}_1^{(t+2)} &\sim \Pr(\boldsymbol{B}_1 | \boldsymbol{b}_2^{(t+1)}, \ldots, \boldsymbol{b}_k^{(t+1)}) \\
&\vdots
\end{aligned}
$$

The realisations of $\boldsymbol{X}$ thus obtained, are coming from the invariant distribution, $\Pr(\boldsymbol{X})$. In particular if X_i is assigned to the singleton set $\boldsymbol{B}_i$ and $k = p$ (number of variables in $\boldsymbol{X}$), then the Gibbs sampler reduces to drawing from the so-called *full conditionals*; each draw is univariate conditional on $\boldsymbol{X} \setminus \{X_i\}$. This is also referred to as a *single-site Gibbs sampler*.

The visitation scheme as suggested above is not crucial for convergence. Random visitation, a systematic sweep or any other combination is possible. Depending on the problem at hand, one scheme may be better than the other. As long as all X_i of $\boldsymbol{X}$ are sampled "infinitely" often, the invariant distribution will be reached.

The Markov chain is also aperiodic, because there is a probability > 0 of remaining in the current state (of a particular block). All dimensions of the state space are considered by sampling from the corresponding conditional, providing a minimal condition for irreducibility. Together with the so-called *positivity requirement*, this provides a sufficient condition for irreducibility. The positivity requirement says that all the conditionals must be strictly positive. Hence, not only are all dimensions visited, but all values along those dimensions can be reached as well.

We illustrate an instance of Gibbs sampling in the context of sampling from a BN. Suppose we are given the BN $(m, \boldsymbol{\theta})$ representing the joint distribution $\Pr(\boldsymbol{X}|m, \boldsymbol{\theta})$, and that the distribution required is $\Pr(\boldsymbol{Z}|m, \boldsymbol{\theta})$ for only a subset of the variables, $\boldsymbol{Z} \subseteq \boldsymbol{X}$. Since Gibbs sampling returns realisations from $\Pr(\boldsymbol{X}|m, \boldsymbol{\theta})$, any marginal distribution of any subset can be estimated by way of counting the realisations. That is, estimate $\Pr(\boldsymbol{Z}|m, \boldsymbol{\theta})$ by using the empirical average of the realisations of $\boldsymbol{Z}$, i.e.:

$$\Pr(\boldsymbol{z}|m, \boldsymbol{\theta}) \approx \frac{1}{n} \sum_{t=1}^{n} I(\boldsymbol{z} \subseteq \boldsymbol{x}^{(t)})$$

By employing a univariate Gibbs sampler drawing from the full conditionals, the Markov blanket makes the sampling process easy. The full conditional distribution reduces to $\Pr(X_j|X_{j-1}, X_{j+1}, \ldots, X_p, m, \boldsymbol{\theta}) = \Pr(X_j|\boldsymbol{X}_{mb(j)}, m, \boldsymbol{\theta})$ because the Markov blanket shields off all influence from variables outside the Markov blanket. Following the BN decomposition, the univariate distribution is:

$$\Pr(X_j|\boldsymbol{X}_{mb(j)}, m, \boldsymbol{\theta}) = \frac{\theta_{X_j|\boldsymbol{X}_{pa(j)}} \prod_{i \in ch(j)} \theta_{X_i|\boldsymbol{X}_{pa(i)}}}{\sum_{x_j} \theta_{x_j|\boldsymbol{X}_{pa(j)}} \prod_{i \in ch(j)} \theta_{X_i|\boldsymbol{X}_{pa(i)}}}$$

from which each variable in $\boldsymbol{Z}$ is drawn in turn according to the normal Gibbs procedure. Notice, that any conditional distribution over a set of variables from the BN can be calculated similarly. In that case simply fix the variables on the conditional side (the so-called *evidence*), and proceed with the Gibbs sampling procedure by updating the other variables. In that respect, Gibbs sampling is a BN inference method, albeit an approximate one.

2.1.5 Mixing, burn-in and convergence of MCMC

From a computational point of view, the most important properties of MCMC samplers pertain to the following:

- the *mixing* of the chain,
- the *burn-in* time, and
- the *convergence*.

It is crucial to understand that MCMC produces *correlated* samples. In this regard it may be beneficial to see MCMC as a way of "walking

around" some state space, such that locations with high probability (density) are passed relatively often. At every step, the current location is returned, and this corresponds to a draw. MCMC is adaptive in the sense that it will have a tendency to seek areas of "mass" or "interest" rather than just walk around aimlessly.

Mixing refers to the long-term correlations between the states of the chain. It refers to how fast the states "forget" about the previous states, i.e., how far from an iid sample the state of the chain is. This captures a notion of how large the "steps" are when traversing the state space. In general we want consecutive realisations to be as close to iid as possible. Slow mixing implies long-term drifts or trends. The terms *mobility* or *acceleration* of a chain, refer to the mixing properties.

When starting an ergodic chain at time $t = 0$ with a realisation of $\boldsymbol{X}^{(0)}$ not sampled from the invariant distribution, the time it takes to reach the invariant distribution is referred to as the *burn-in*, i.e., the time it takes before samples can be regarded as coming from the target distribution. After the burn-in, we say that the chain has *converged*; the realisations from then on may be considered samples from the invariant distribution.

The question is how long the chain needs to be before the realisations are sufficiently close to the invariant distribution. Obviously, we would like to reach the invariant distribution with a minimum amount of computational effort usually meaning that we want to keep the "wasteful" burn-in realisations at a minimum. At the same time we also would like to be certain that the invariant distribution indeed *has* been reached. Prematurely assuming that samples come from the invariant distribution when in fact they are just from a burn-in is less than useful.

With poor mobility of the chain, the sampler explores the state space in an inefficient way, which in effect means that the burn-in period increases. Moreover, assuming that the chain indeed has converged, poor mixing means that more realisations are required in order to gather enough empirical information to get an impression of the invariant distribution. In that respect, the mixing properties of a chain is the most significant factor that determines the performance of MCMC.

Unless we can provide an initial $\boldsymbol{X}^{(0)}$ from the invariant distribution (no burn-in is required at all), diagnosing convergence of MCMC is necessary for a trustworthy result. Unfortunately this is a non-trivial problem. In practice diagnosing convergence using an automated procedure is quite difficult and problem dependent. This means that in general such automated methods are unsafe on their own. Many diag-

nostics are very technical, and for many problems they do not directly apply, or are difficult to implement; see for instance Gelman and Rubin, 1992.

Monitoring the realisations against t gives a reasonable impression of the range of values that are "possible" for the invariant distribution. When realisations are multivariate, scalar functions of the realisations may be monitored. A burn-in will lie outside this range and stand out in comparison to the remaining realisations. With poor mixing, a burn-in may be difficult to discern, since a certain trend may last for many iterations. We may be lead to think that the chain has reached the invariant distribution when it in fact has not. However, if we are able accelerate the chain, then we may monitor the realisations over a shorter time frame, and trends are easier to discern.

Admittedly, a graphical criterion for assessing convergence may be deceptive because it does obviously not say anything about "where the chain will be in a few moments from now". This is especially so if only a single chain is used. Running MCMC with different initial so-called overdispersed values with respect to the invariant distribution, and comparing the plots by visual inspection provides a better way of checking if the chain is stuck "by accident" or if all runs result in similar bands of realisations. Of course similar plots is no guarantee for convergence, but it indicates a certain robustness, because the different chains eventually all reach the same region meaning that minor non-deterministic disturbances have not resulted in "off-course" behaviour of the chains. However, running several parallel chains is from a computational point of view not the preferred choice for assessing convergence.

2.1.6 The importance of blocking

In the previous section the importance of the Markov chain mixing properly was stressed. In this section we address the issue of mixing in the context of the Gibbs sampler. Blocking plays an important role in this regard.

Suppose that we apply (block) Gibbs sampling to obtain samples from the invariant distribution $\Pr(\boldsymbol{Y}, \boldsymbol{Z})$, $\boldsymbol{X} = \boldsymbol{Y} \cup \boldsymbol{Z}$. We first sample from $\Pr(\boldsymbol{Y}|\boldsymbol{Z})$ and then from $\Pr(\boldsymbol{Z}|\boldsymbol{Y})$ etc. The values of the variables behind the conditional constrain the probability of sampling certain values from the distribution, e.g., when sampling from $\Pr(\boldsymbol{Z}|\boldsymbol{Y})$, the value that $\boldsymbol{Z}$ can take on, is constrained by the value of the last realisation of $\boldsymbol{Y}$. Although irreducibility is guaranteed when the conditionals are strictly positive, an unfortunate value on the conditional side may imply that

that the variance of the conditional distribution is very small. It may be highly improbable (though not impossible) to sample certain values "in the tail", effectively having a negative impact on the mobility of the chain. Theoretically the chain will reach the entire sampling space, in practice this may take very long, because the chain has maneuvered itself into a region that is comparable to a local optimum.

When dependence between $\boldsymbol{Z}$ and $\boldsymbol{Y}$ is strong, then the situation sketched may very well occur. If the dependence is absent, thus when $\Pr(\boldsymbol{Y}, \boldsymbol{Z}) = \Pr(\boldsymbol{Y}) \cdot \Pr(\boldsymbol{Z})$, we have that the conditionals reduce to $\Pr(\boldsymbol{Y})$ and $\Pr(\boldsymbol{Z})$, and the interaction effects are gone. For instance, when sampling from $\Pr(\boldsymbol{Z}|\boldsymbol{Y}) = \Pr(\boldsymbol{Z})$, a previous value assigned to $\boldsymbol{Y}$ does not influence the probability of drawing a particular value $\boldsymbol{z}$.

The above discussion suggests that we should look for blocks that are only weakly dependent since this prevents the chain from getting trapped (see also Roberts and Sahu, 1997). It may be difficult to devise such blocks where this is the case for all values one can assign to the blocks. In this respect a notion of *context* (in)dependence also plays a role. Two blocks may be strongly dependent for *some* values, yet for other values less. Hence, we seek blocks that are context independent on each other for *many* block assignments.

Additionally, Gibbs sampling allows for dynamic creation of blocks while running. The blocks need not *a priori* be defined before starting the Gibbs sampler, but the individual blocks may change. Also, the blocks need not be disjoint. We may profit from this dynamic way of defining blocks since it offers a large degree of freedom in order to accommodate the desire to sample blocks that don't constrain each other strongly.

A block we may regard as a set of variables that inherently belong together, or alternatively, as something that should be sampled together, because the individual variables strongly depend on each other, and splitting them has a negative impact on the mixing of the chain (Jensen et al., 1995). It may not always be easy to determine these blocks, but for some problems there is a natural decomposition which follows almost immediately from the problem description. Although a different branch of computer science, in evolutionary computation the importance of the right decomposition has also been acknowledged as being very important, and deceptive decompositions have been investigated (Pelikan et al., 2003). In many other ways there are actually striking similarities between Monte Carlo methods and the population based paradigm on which evolutionary computation is based. In Section 3.3 we will sam-

ple edges from the posterior model distribution, and we investigate how edges of a DAG model are best blocked to improve mixing.

As an example of blocking, suppose that samples are required from some distribution $\Pr(\boldsymbol{X}), \boldsymbol{X} = \{X_1, X_2, X_3, X_4\}$, and assume that at some iteration the values of these variables are such that X_1 and X_2 naturally belong together and should not be split. Due to this dependence, they are consequently joined in a block (X_1, X_2). We draw the values for (X_1, X_2) (jointly) from the conditional $\Pr(X_1, X_2|x_3, x_4)$, where (x_3, x_4) are values from a previous iteration. In the next iteration, the situation may be such that the variables (X_2, X_3, X_4) inherently belong together, and consequently should be joined, and sampled as a block from $\Pr(X_2, X_3, X_4|x_1)$, and so on. As long as each X_i is sampled "infinitely often", jointly or individually, samples from $\Pr(\boldsymbol{X})$ are obtained in the limit.

3. Learning models via MCMC

MCMC offers a feasible way of learning BN models from a Bayesian perspective. In Chapter 3, Section 3.2, model learning was treated from a model selection point of view. To a Bayesian, a single model is a rather poor summary statistic of the posterior model distribution. A Bayesian is interested in the entire distribution $\Pr(M|\boldsymbol{d})$. Also, if we are interested in some feature over models quantified by Δ, we can average the feature with respect to the model posterior:

$$\mathrm{E}[\Delta(M)|\boldsymbol{d}] = \sum_m \Delta(m) \cdot \Pr(m|\boldsymbol{d})$$

In this section we discuss two methods for learning models via MCMC using the marginal likelihood scoring criterion; the discussion is based on Riggelsen, 2005. Other MCMC approaches for learning BN models exist, see for instance Friedman and Koller, 2003. An MCMC approach for model learning which does not employ the marginal likelihood criterion is given in Green, 1998. An alternative to MCMC is described in Madigan and Raftery, 1994.

The posterior distribution $\Pr(M|\boldsymbol{d})$ is obtained via Bayes' law, hence has the form of eq. 4.7, thus is precisely an instance of what can be solved via MCMC. The denominator $\sum_m \Pr(\boldsymbol{d}|m) \Pr(m)$ in eq. 3.12 is due to the large number of models impossible to evaluate.

3.1 Sampling models

MCMC Metropolis-Hastings sampling of models is discussed in Madigan and York, 1995; Kocka and Castelo, 2001. Here the *proposal distribution*, guides the incremental changes of the models by proposing to somehow change the current model. This proposal is produced by drawing $M' \sim \Pr'(M|M^{(t)})$. With probability:

$$\rho(M^{(t)}, M') = \min\left\{1, \frac{\Pr'(M^{(t)}|M')\Pr(M'|\boldsymbol{d})}{\Pr'(M'|M^{(t)})\Pr(M^{(t)}|\boldsymbol{d})}\right\}$$

the proposal is accepted and $M^{(t+1)} = m'$, otherwise $M^{(t+1)} = m^{(t)}$. For $t \to \infty$ models from the invariant distribution are obtained.

The usual proposal distribution changes the current model in a single adjacency, by selecting two vertices at random, and either adds, reverses or removes the edge (arc) between the vertices. Since these proposals are uniform, the proposal fraction $\Pr'(M^{(t)}|M')/\Pr'(M'|M^{(t)})$ cancels out, and the sampler is driven by the marginal likelihood ratio. Furthermore, since models consecutive differ in a single edge, it follows from the decomposition of the marginal likelihood, that all terms cancel out except those terms for which the parent set changes.

3.2 Sampling edges

In the following we apply a single-site Gibbs MCMC for sampling models from the posterior model distribution. Instead of considering M as a single random variable, we suggest to split the model into separate edges, each of which we regard as a random variable. This is the natural decomposition of a DAG model for applying Gibbs sampling, and is equivalent to applying the single adjacency Metropolis-Hastings sampler discussed in the previous section. However, the Gibbs sampling technique explicitly gives rise to the question if single edges is the best decomposition from an efficiency point of view: how does it influence the mixing of the chain? This is investigated in the sections to come, beginning from Section 3.3.

Define for all $r = 1, \ldots, \frac{p\cdot(p-1)}{2}$ possible edges of DAG model M the random variables E_r with state space $\Omega_{E_r} = \{\leftarrow, \rightarrow, \not\leftrightarrow\}$, i.e., every edge of the graph can take on a direction, or can be absent. If the configuration of all edges forms a DAG, the posterior joint distribution, $\Pr(E_1, \ldots, E_{\frac{p\cdot(p-1)}{2}}|\boldsymbol{d})$ is well-defined.

Via Gibbs sampling, models from the posterior are obtained as joint realisations of the edge assignments. The process goes as follows: draw

edges at iteration t from the full conditional given the data:

$$\begin{aligned}
E_1^{(t)} &\sim \Pr(E_1 | e_2^{(t-1)}, \ldots, e_{\frac{p\cdot(p-1)}{2}}^{(t-1)}, \boldsymbol{d}) \\
&\vdots \\
E_{\frac{p\cdot(p-1)}{2}}^{(t)} &\sim \Pr(E_{\frac{p\cdot(p-1)}{2}} | e_1^{(t)}, \ldots, e_{\frac{p\cdot(p-1)}{2}-1}^{(t)}, \boldsymbol{d}) \\
E_1^{(t+1)} &\sim \Pr(E_1 | e_2^{(t)}, \ldots, e_{\frac{p\cdot(p-1)}{2}}^{(t)}, \boldsymbol{d}) \\
&\vdots
\end{aligned}$$

Each draw is subject to the constraint that all edges together must form an acyclic graph. There are always at least two possible edge assignments. Removal is always an option since it can't introduce a cycle. Also, it is always possible to add an arc in at least one direction between any two vertices that are not adjacent.

The fact that not all 3 edge assignments are allowed at every drawing stage, means that irreducibility can't be guaranteed by using the positivity argument, i.e., that $\Pr(E_l | \boldsymbol{E} \setminus \{E_l\}, \boldsymbol{d}) > 0$. However, the Markov chain defined here *is* irreducible, because at every draw the state $\neq$ is a possibility and this state never introduces a cycle. Hence, there is a non-zero probability of removing arcs that obstruct the addition of other edges in any direction in the graph (obstruct in the sense that the graph would become cyclic). Consequently all DAGs can be reached by "breaking down" the current DAG (a kind of backtracking) and rebuilding another one thereby reaching another DAG.

In order to draw edge E_l from the full conditional given the data we calculate:

$$\begin{aligned}
\Pr(E_l | \boldsymbol{E} \setminus \{E_l\}, \boldsymbol{d}) &= \frac{\Pr(E_1, \ldots, E_l, \ldots, E_{\frac{p\cdot(p-1)}{2}} | \boldsymbol{d})}{\sum_{e_l} \Pr(E_1, \ldots, e_l, \ldots, E_{\frac{p\cdot(p-1)}{2}} | \boldsymbol{d})} \\
&= \frac{\Pr(\boldsymbol{d} | E_1, \ldots, E_l, \ldots, E_{\frac{p\cdot(p-1)}{2}})}{\sum_{e_l} \Pr(\boldsymbol{d} | E_1, \ldots, e_l, \ldots, E_{\frac{p\cdot(p-1)}{2}})} \qquad (4.11)
\end{aligned}$$

where a uniform model prior $\Pr(M)$ on the model space, and the denominator $\Pr(\boldsymbol{d}) = \sum_m \Pr(\boldsymbol{d}|m) \Pr(m)$ both cancel out. When drawing

an edge from the Gibbs sampler, say E_l, at most two terms are affected, namely the terms pertaining to the vertices of edge E_l; denote those vertices by X_i and X_j. It follows from the marginal likelihood decomposition given in eq. 3.18 that:

$$\Pr(\boldsymbol{d}|\boldsymbol{E}) = \prod_{r \neq \{i,j\}} \Pr(\boldsymbol{d}_r|\boldsymbol{d}_{pa(r)}, \boldsymbol{E}) \cdot \Pr(\boldsymbol{d}_i|\boldsymbol{d}_{pa(i)}, \boldsymbol{E}) \Pr(\boldsymbol{d}_j|\boldsymbol{d}_{pa(j)}, \boldsymbol{E})$$

such that all factors $\prod_{r \neq \{i,j\}} \Pr(\boldsymbol{d}_r|\boldsymbol{d}_{pa(r)}, \boldsymbol{E})$ cancel out in eq. 4.11 because the parent sets for the corresponding vertices are unchanged.

To approximate the expected value of model features, we use the empirical average:

$$\mathrm{E}[\Delta(M)|\boldsymbol{d}] \approx \frac{1}{n} \sum_{t=1}^{n} \Delta(m^{(t)}) \tag{4.12}$$

where n denotes the total number of samples from the Markov chain. Often this kind of averaging is done over features one can read directly off a model, e.g., Markov blanket features of vertices, but theoretically any statement that the model entails can be averaged.

3.3 Blocking edges

In Section 2.1.6, the impact of blocking on the performance of MCMC was treated. Therefore, we need to analyse if there are edges that inherently belong together, and should be considered jointly when sampling. Although the univariate edge sampler theoretically does what it is supposed to do, it is from a practical point of view advantageous to seek blocks that can improve the mixing of the chain and prevent it from getting trapped for long periods of time.

The state of a single edge E_l between X_i and X_j is determined by those terms in the marginal likelihood that correspond to X_i and X_j, that is, in eq. 3.18 the terms $\Pr(\boldsymbol{d}_i|\boldsymbol{d}_{pa(i)}, \boldsymbol{E})$ and $\Pr(\boldsymbol{d}_j|\boldsymbol{d}_{pa(j)}, \boldsymbol{E})$. These terms indirectly depend on $\boldsymbol{X}_{pa(i)}$ and $\boldsymbol{X}_{pa(j)}$ in terms of their realisations in the (fixed) data sample. The marginal likelihood does not decompose into independent terms of parent set variables, and every time the parent set changes for a particular vertex, the corresponding term in the marginal likelihood needs to be recomputed. This means that when drawing an edge state for E_l, the current parent sets of X_i and X_j constrain the distribution of E_l. It follows that E_l depends on all incoming edges (arcs) to X_i and X_j; this is a form of context dependence, because when those same edges are outgoing or absent, the dependence is not

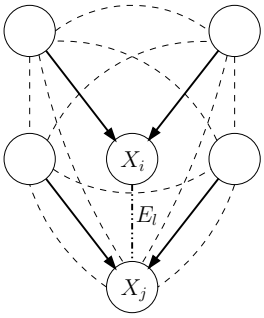

Figure 4.3. Dashed-dotted edge is the centre edge E_l. Solid edges (arcs) are the edges on which E_l currently depends; they "span" the block. All edges, including the dashed ones, potentially constrain each other (though in a varying degree) when assigned directions.

present. From the perspective of E_l, the dependence just described is a direct one, but the dependency goes even further, because the edges to X_i and X_j in turn depend on edges in the same way—this is an indirect dependence. When the parent sets $\boldsymbol{X}_{pa(i)}$ and $\boldsymbol{X}_{pa(j)}$ are large, E_l is influenced by many edges. This also means that in relatively dense areas of a DAG, the edges strongly depend on each other. Edge E_l can be considered the "centre" of a dependence region.

These considerations suggest that when we have to determine E_l, all edges in a region around E_l should be joined in a block, and be drawn "as one". First of all we notice that the dependence is a context dependence and therefore a block dynamically changes over time depending on the states of the edges in the vicinity of E_l. Also, the blocks overlap because the state of many edges is constrained by the same set of edges; this is especially so in dense regions where edges are close to each other. We suggest to apply a Gibbs sampler that *focuses per draw on one dependence region with one particular edge at the centre* instead of focusing on a single edge. By considering a region around E_l, we acknowledge that these edges may constrain the assignment of the state of the centre edge strongly, and to a lesser extent constrain each other.

The question that arises is then how large, or, alternatively, how far from the centre edge E_l, edges should be joined in a block. We suggest to join all edges on which E_l *directly* depends (along with E_l itself) into block l. However, this set of edges only determines the *current* situation but not necessarily "the right one" which is exactly what is object to learning. Therefore we also include in the block all edge variables *between*

the vertices spanned by the currently dependent edges, i.e., not only those pointing to X_i and X_j; see figure 4.3 for an illustration. All the edges in that block are "close" and either currently constrain each other (and most strongly constrain E_l), or potentially will constrain each other if assigned a direction.

A consequence of this dynamic way of blocking, is that areas that are currently dense, receive more attention and are sampled more frequently than less dense regions. In dense regions edges constrain each other, hence spending more time in these regions helps solving unfortunate edge assignments that negatively impact the edge configurations of dependent edges.

3.3.1 Blocks and Markov blankets

The suggested way of blocking coincides with the Markov blanket: the solid edges (arcs) in figure 4.3 connect X_i with the vertices in the Markov blanket of X_i. Hence, initially a block is defined by the edges in the Markov blanket of X_i.

The Markov blanket contains exactly those variables are the "most relevant" vertices for determining X_i. For that reason, it is reasonable to think that these variables potentially are related (adjacent in the graph) as well, since they represent concepts that are "close" from a domain perspective. Of course, this argument only is valid in the heuristic sense. The dashed edges in figure 4.3 can be thought of as representing those edges that potentially connect the Markov blanket vertices. In that respect including those edges in the block a priori is not a bad idea. We refer to the edges within a block as the *currently relevant* edges, and those not within the block as *currently irrelevant* edges.

Formally, these sets of edges are defined in the following way:

- The set of *relevant edges* of X_i is:

$$\boldsymbol{\mathcal{E}}'_i = \{E = (X_s, X_l) | X_s, X_l \in \boldsymbol{X}_{mb(i)} \cup \{X_i\}\}$$

- The set of *irrelevant edges* of X_i is:

$$\boldsymbol{\mathcal{E}}''_i = \{E = (X_i, X_l) | X_l \in \boldsymbol{X}\} \setminus \boldsymbol{\mathcal{E}}'_i$$

- A *block* of edges for X_i is: $\boldsymbol{\mathcal{E}}_i = \boldsymbol{\mathcal{E}}'_i \cup \boldsymbol{\mathcal{E}}''_i$

In figure 4.4 the 3 sets are illustrated.

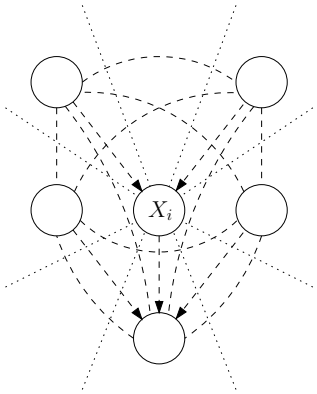

Figure 4.4. A block, $\boldsymbol{\mathcal{E}}_i$. Dashed edges belong to the relevant edges, $\boldsymbol{\mathcal{E}}_i'$. Dotted edges belong to the irrelevant edges, $\boldsymbol{\mathcal{E}}_i''$.

The general approach is now as follows: either shrink or grow the Markov blanket of a vertex by adding or removing vertices via currently irrelevant edges, *or* change the internal relationships between the vertices of the Markov blanket via the currently relevant edges. Observe that $\boldsymbol{\mathcal{E}}_i'$ actually captures the notion of a sub-graph, and that the proposed approach boils down to learning sub-graphs that, when combined, form the overall DAG model.

3.3.2 Sampling blocks

Block Gibbs sampling in terms of $\boldsymbol{\mathcal{E}}_i$, follows the sampling scheme:

$$\begin{aligned}
\boldsymbol{\mathcal{E}}_1^{(t)} &\sim \Pr(\boldsymbol{\mathcal{E}}_1 | \bar{\boldsymbol{\varepsilon}}_1, \boldsymbol{d}) \\
&\vdots \\
\boldsymbol{\mathcal{E}}_p^{(t)} &\sim \Pr(\boldsymbol{\mathcal{E}}_p | \bar{\boldsymbol{\varepsilon}}_p, \boldsymbol{d}) \\
\boldsymbol{\mathcal{E}}_1^{(t+1)} &\sim \Pr(\boldsymbol{\mathcal{E}}_1 | \bar{\boldsymbol{\varepsilon}}_1, \boldsymbol{d}) \\
&\vdots
\end{aligned}$$

Here $\bar{\boldsymbol{\varepsilon}}_j$ is the *current* configuration of the edges in the *complement* of the set $\boldsymbol{\mathcal{E}}_j$. As previously remarked in Section 2.1.4, the visitation scheme is not crucial, and random selection of blocks may be beneficial for computational reasons.

Sampling a block of edges may seem difficult at first sight. In contrast to single edges, entire blocks have many different configurations. The probability of each possible configuration of $\boldsymbol{\mathcal{E}}_i$ needs to be determined before we can draw one. Additionally, the relevant and irrelevant edges play different roles. Adding a vertex not currently belonging to the Markov blanket via the irrelevant edges should have a lower priority than changing the internal relationships between those vertices that already belong to the Markov blanket via the relevant edges.

Metropolis-Hastings sampling provides a way to deal with these issues. We use a "Metropolis-within-Gibbs" MCMC sampler, where Gibbs sampling and Metropolis-Hastings sampling are combined in order to sample models from the invariant model posterior. Each draw from the Gibbs sampler is performed by a Metropolis-Hastings sampler. The proposal distribution is defined as a mixture distribution, where one component, $f(\cdot)$, deals with the edges in the relevant edge set, and the other component, $g(\cdot)$, deals with the edges in the irrelevant edge set, that is:

$$\Pr'(\boldsymbol{\mathcal{E}}^{(t+1)}|\boldsymbol{\mathcal{E}}^{(t)}) = w \cdot f(\boldsymbol{\mathcal{E}}^{(t+1)}|\boldsymbol{\mathcal{E}}'^{(t)}) + (1-w) \cdot g(\boldsymbol{\mathcal{E}}^{(t+1)}|\boldsymbol{\mathcal{E}}''^{(t)}) \quad (4.13)$$

where $0 < w < 1$ determines the mixture weights.

When $f(\cdot)$ is applied, values for the edges in the relevant set are drawn, i.e., MCMC is run in order to obtain the posterior distribution of those edges. The cardinality of the set of relevant edges is kept fixed once the Metropolis-Hastings sampler is applied—we merely assign (new) values to the edge variables. Hence, the set of relevant edges is determined before entering the Metropolis-Hastings sampler, and does not change until control is given back to the overall Gibbs sampler. For $g(\cdot)$ the same holds, but here assignments are considered to the variables in the irrelevant set.

The distribution $f(\cdot)$ produces uniform proposals such that each edge $E_r \in \boldsymbol{\mathcal{E}}'$ has probability $1/|\boldsymbol{\mathcal{E}}'|$ of being drawn. Depending on the current value e_r, a state change is proposed to one of the (at most) two alternatives with probability 0.5. For example, if $E_r^{(t)} = \neq$ then either $E_r^{(t+1)} = \rightarrow$ or $E_r^{(t+1)} = \leftarrow$ is proposed. For the distribution $g(\cdot)$ the same holds, but here we have $E_r \in \boldsymbol{\mathcal{E}}''$ with probability $1/|\boldsymbol{\mathcal{E}}''|$. Notice that edges not in either of these two sets remain unchanged.

The weight w in the mixture varies how much "attention to pay" to the configuration of the edges in the Markov blanket of the current vertex. We want to try out several different configurations of the edges in the relevant edge set before deciding to grow or shrink it via the

irrelevant edge set. This implies that $f(\cdot)$ should be applied more often than $g(\cdot)$.

Uniform proposals are not the only option, but it makes the Metropolis-Hastings acceptance ratio easy to compute. For Metropolis-Hastings, the acceptance probability depends on the following fraction, with the proposal distribution defined in eq. 4.13:

$$\frac{\Pr'(\boldsymbol{\mathcal{E}}^{(t)}|\boldsymbol{\mathcal{E}}^{(t+1)})}{\Pr'(\boldsymbol{\mathcal{E}}^{(t+1)}|\boldsymbol{\mathcal{E}}^{(t)})}$$

For both the mixture in the numerator and in the denominator, the weights are the same, the conditional distributions select edges with equal probability and there is always the same number of alternative edge assignments, i.e., the distributions are uniform, hence, the ratio cancels out.

If non-uniform proposals are used, then the proposal ratio does not cancel out, and it has to be computed explicitly. It is questionable if one is able to define such non-uniform proposals unless prior knowledge is available. Moreover, prior knowledge is rarely expressed in terms of relevant and irrelevant edge sets, so specifying a non-uniform proposal distribution is in no way trivial.

There seems to be something counterintuitive as to why the combined MCMC sampler approach works: why sample single edges via Metropolis-Hastings when the performance gain lies in the fact that all edges in a block should be considered "one entity"? One could argue that the problem with the edge dependences has just been pushed down one level to another MCMC sub-process, and that this process suffers from exactly the same thing as the original single edge Gibbs MCMC. The reason why the proposed "Metropolis-within-Gibbs" approach works is that edges that are part of several Markov blankets are sampled relatively often. This entails that edges in dense regions of the DAG are sampled more often. More time is spent in regions where edges constrain each other strongly, meaning that potential obstructions are "solved" simply because improbable yet not impossible alternatives edge assignments are eventually accepted because they have been proposed many times. In doing so we take into account that sampling models is more difficult in some parts of the model space.

3.3.3 Validity of the sampler

In using such a "Metropolis-within-Gibbs" sampler, the question arises if the convergence towards the invariant model distribution is still guar-

anteed. Combining MCMC samplers boils down to combining several transition probabilities via a mixture or product of transitions. It is well-known (Tierney, 1994) that such combinations leave the overall chain ergodic with the desired invariant distribution, provided the sub-MCMC samplers created via Metropolis-Hastings sampling in the separate Gibbs sampling steps are ergodic with the invariant distributions $\Pr(\boldsymbol{\mathcal{E}}_i|\bar{\boldsymbol{\mathcal{E}}}_i, \boldsymbol{d})$.

THEOREM 4.1 *The* MB-MCMC *model sampler produces an ergodic Markov chain with the invariant distribution* $\Pr(M|\boldsymbol{d})$.

It is not difficult to see that this is the case, because it follows from the fact that Metropolis-Hastings MCMC produces an ergodic Markov chain: The proposal distribution of the Metropolis-Hastings sampler will with a non-zero probability propose a state change to any edge in $\boldsymbol{\mathcal{E}}_i$, which guarantees irreducibility. With a non-zero probability it will remain in the current state for any edge implying aperiodicity. We may thus conclude that the Metropolis-Hastings sampler, for all $i = 1, \ldots, p$, in the limit returns realisations from the invariant distribution $\Pr(\boldsymbol{\mathcal{E}}_i|\bar{\boldsymbol{\mathcal{E}}}_i, \boldsymbol{d})$, i.e., realisations for the edges in $\boldsymbol{\mathcal{E}}_i$ given all other edges. We note that formally it is not even an requirement that Markov chains created by the Metropolis-Hastings samplers converges for every i before going on to the next Gibbs sampling step.

Since blocks dynamically change over time, we need to check if all edges are sampled. The Gibbs sampler makes a call to a Metropolis-Hastings sampler for every vertex, and it trivially follows that every edge eventually will be part of an irrelevant edge set. We can't guarantee that the edges will be part of a relevant edge set however, but as long as all edges *are* considered (and sampled via Metropolis-Hastings MCMC), we have that the realisations drawn come from $\Pr(E_1, \ldots, E_{\frac{p\cdot(p-1)}{2}}|\boldsymbol{d})$ as $t \to \infty$.

3.3.4 The MB-MCMC model sampler

Algorithm 1 contains the pseudocode of the Markov blanket MCMC (MB-MCMC) sampler. Line 3 determines the block to pay attention to; here a systematic sweep is shown. Line 4 calls the algorithm for reversing covered arcs; we refer to Kocka and Castelo, 2001 for the implementation of this step. Lines 5–6 determines the edges to consider, and in lines 8–12 the edges are drawn from the sets of relevant edges. The proposals are accepted or rejected in line 13–16. In line 17 the configuration of all edges is recorded, i.e., here the actual models from the posterior are

saved. One may decide to sub-sample the Markov chain of models by only recording the draws once in a while.

Algorithm 1: MB-MCMC

Input : k, MH-steps; w, prop. expand/change block conf.
Output: Edges from $\Pr(\boldsymbol{E}|\boldsymbol{d})$ (requires burn-in).

$M \leftarrow G = (\boldsymbol{X} = \{X_1, \ldots, X_p\}, \boldsymbol{E} = \{E_1 =\not\sim, \ldots, E_{\frac{p\cdot(p-1)}{2}} =\not\sim\})$
for $r \leftarrow 0$ **to** ∞ **do**
  $i \leftarrow (r \bmod p) + 1$
  `rcar(`*10*`)`
  `/* Define relevant and irrelevant sets/blocks */`
  $\boldsymbol{\mathcal{E}}'_i \leftarrow \{E = (X_s, X_l) | X_s, X_l \in \boldsymbol{X}_{mb(i)} \cup \{X_i\}\}$
  $\boldsymbol{\mathcal{E}}''_i \leftarrow \{E = (X_i, X_l) | X_l \in \boldsymbol{X}\} \setminus \boldsymbol{\mathcal{E}}'_i$
  for $t \leftarrow 0$ **to** k **do**
    draw $U \sim \mathcal{U}[0,1]$ `/* Uniform draw */`
    `/* Draw edge values for the block */`
    if $u < w$ **and** $\boldsymbol{\mathcal{E}}'_i \neq \varnothing$ **then**
      draw $\boldsymbol{\mathcal{E}}_i^{(t+1)} \sim f(\boldsymbol{\mathcal{E}}_i | \boldsymbol{\varepsilon}_i'^{(t)})$
    else
      draw $\boldsymbol{\mathcal{E}}_i^{(t+1)} \sim g(\boldsymbol{\mathcal{E}}_i | \boldsymbol{\varepsilon}_i''^{(t)})$
    $\rho \leftarrow \min\{1, \Pr(\boldsymbol{d}|\boldsymbol{\varepsilon}_i^{(t+1)}) / \Pr(\boldsymbol{d}|\boldsymbol{\varepsilon}_i^{(t)})\}$
    draw $U \sim \mathcal{U}[0,1]$
    if $u \geq \rho$ **then**
      $\boldsymbol{\mathcal{E}}_i^{(t+1)} \leftarrow \boldsymbol{\varepsilon}_i^{(t)}$
  `record(`$e_1, \ldots, e_{\frac{p\cdot(p-1)}{2}}$`)`

The algorithm takes two arguments: k determines the number of times the Metropolis-Hastings sampler is run, and w determines the probability of changing the internal configuration of a component vs. adding or removing new vertices. Parameter k need not be large for the overall invariant model distribution to be reached, i.e., the Metropolis-Hastings sampler need not converge at every call. In fact we have found it to be beneficial for the convergence rate to assign k a small value; too large a value may lead to slow mixing and convergence. In our experiments we have set $k = 5$, and $w = 0.95$. Letting the Metropolis-Hastings sampler converge before the overall Gibbs sampler reaches its invariant distribution, means that the invariant distributions of the Metropolis-Hastings

samplers are reached and they now coincide with the marginal (at that time still) sub-optimal distribution of the Gibbs sampler. There is no reason to exhaustively explore the state space of the marginals of the Gibbs sampler, unless we are close to its invariant distribution. Doing so has a negative impact on the acceleration of the chain.

When every vertex is assigned a cache that keeps the sufficient statistics indexed by the parent set, we may drastically improve the speed of MCMC by querying the cache before querying the data. We have implemented the Markov blanket sampler in C++ using STL, and for the experiments in the next section we were able to reach what we believe are the invariant distributions in less than 10 minutes on a 2 GHz machine.

3.3.5 Evaluation

We considered two BNs for the experiments: The ALARM network with 37 vertices and 46 arcs (Beinlich et al., 1989) about intensive care patient monitoring, and the Insurance network with 27 vertices and 52 arcs (Binder et al., 1997) classifying car insurance applications. We used the BDeu metric for the counts $\boldsymbol{\alpha}$ with an ESS of 1. All experiments were run for 1,000,000 iterations. As convergence diagnostic we monitored the number of edges as suggested in for instance Giudici and Green, 1999. We compared the Markov blanket MCMC with $e\text{MC}^3$, a single edge MCMC sampler that also employs the RCAR algorithm (Kocka and Castelo, 2001; Castelo, 2002).

In figure 4.5 the results of the ALARM network are illustrated for 1000 and 5000 samples. With 1000 samples, we see that two independent runs of the MB-MCMC both converge towards models with about 50–53 edges. There is no significant difference in the convergence behaviour. For $e\text{MC}^3$ two runs produce different behaviour and result in models with 68–77 edges. For 5000 records similar observations hold, but overall the number of edges is lower: 45–51 for MB-MCMC and 57–70 for $e\text{MC}^3$. We notice that $e\text{MC}^3$ seems sensitive to the starting point of the chain. To show this more clearly, we ran both samplers starting from the empty graph, and from the actual ALARM graph for 7000 samples. The results are illustrated in figure 4.6. For the 7000 records we would expect that the number of edges on average should converge to 46, i.e. there is enough data to support the data generating model. For MB-MCMC, both chains converge towards models with 44–50 edges. The most frequently sampled model is similar to the ALARM network ± 2 arcs. For $e\text{MC}^3$ there is a big difference.

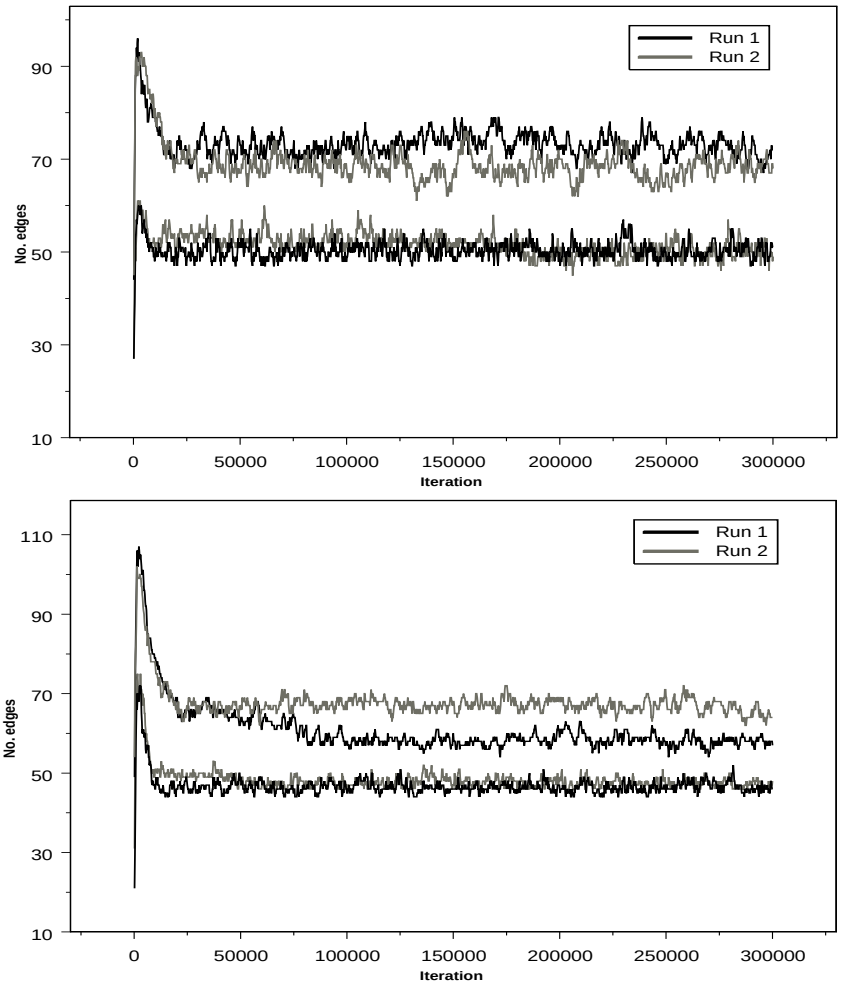

Figure 4.5. ALARM network. Convergence behaviour given 1000 (*top*) and 5000 (*bottom*) records for two independent runs. The lower lines are from the Markov blanket MCMC, and the upper lines from $e\text{MC}^3$.

The chain started from the actual network stays at around 50–55 edges, but the chain started from the empty graph gets stuck at 63–70. The most frequently sampled model is in both situations less similar to the actual ALARM network than in the MB-MCMC case (excess of ± 10 and ± 25 arcs).

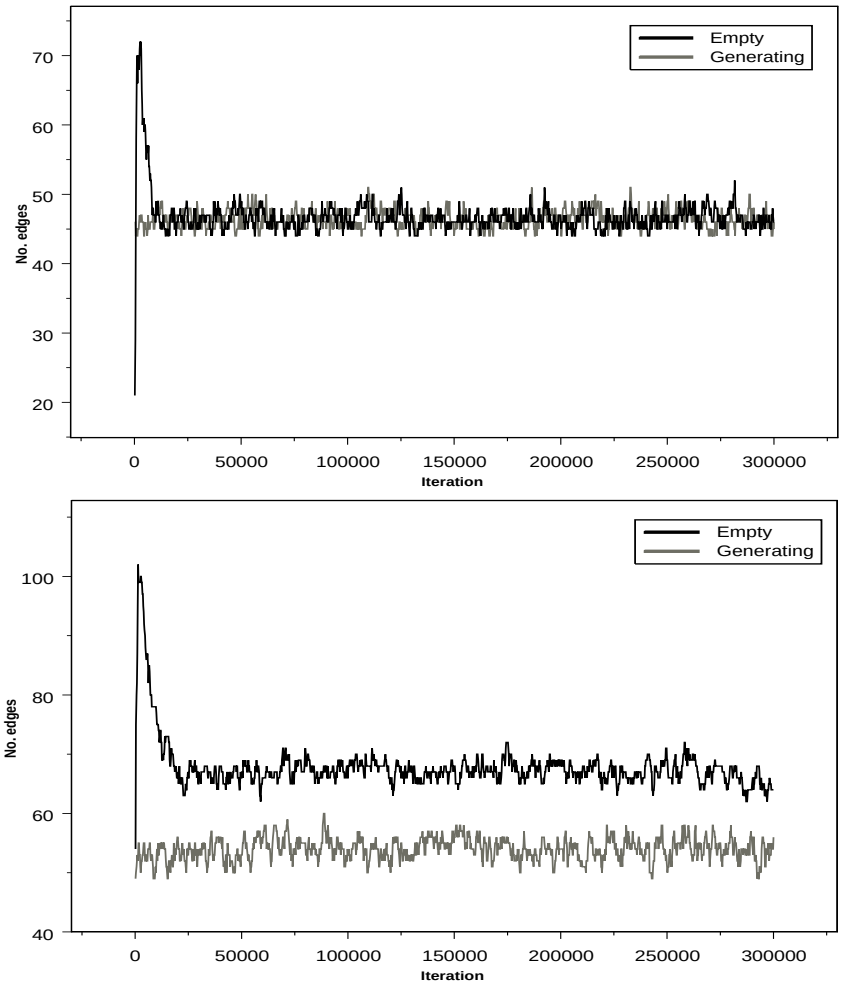

Figure 4.6. ALARM network. Convergence behaviour of the Markov blanket MCMC (*top*) and $e\text{MC}^3$ (*bottom*) given 7000 records starting from the empty and the data generating model.

Next we consider results of the Insurance network in figure 4.7 for 500 samples. We would like to note that the association between several parent-child variables in the Insurance network is rather weak and that even for large data sets these associations will be deemed absent by the marginal likelihood score. For 500 records the MB-MCMC converges to an invariant distribution where models are sampled with 36–40 edges.

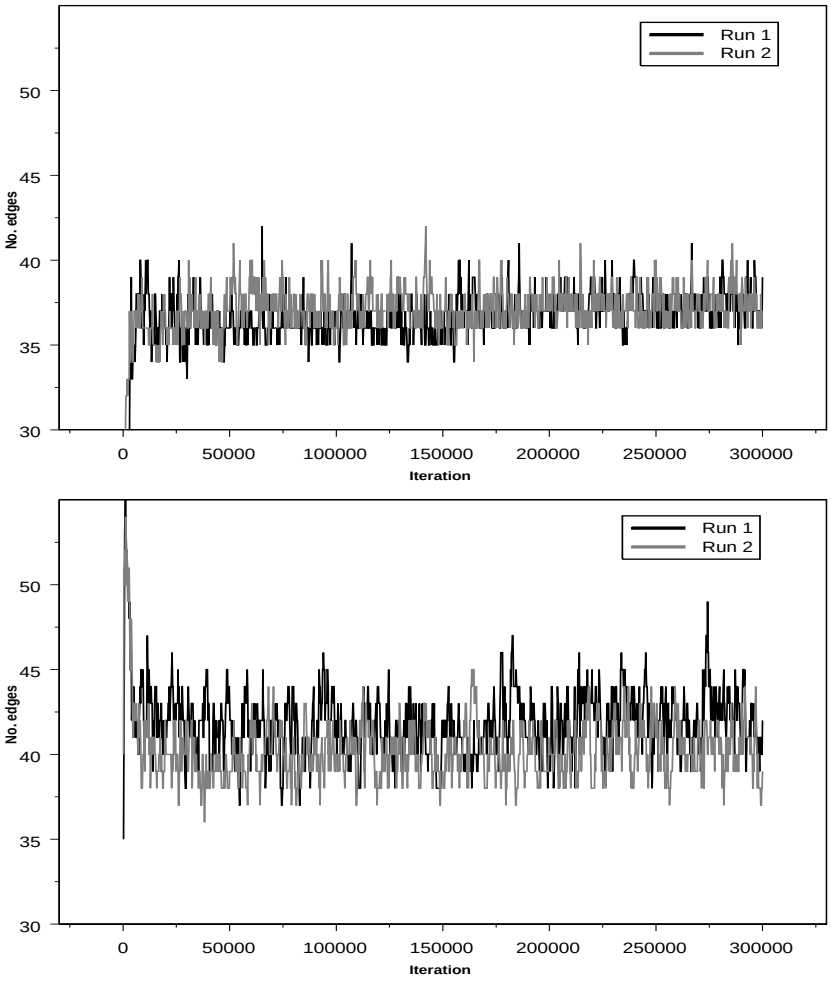

Figure 4.7. Insurance network. Convergence behaviour given 500 records for two independent runs of the Markov blanket MCMC (*top*) and eMC^3 (*bottom*).

The two runs meet at around 150,000 iterations. For eMC^3 however, the two chains don't quite agree in the number of edges: somewhere between 37–46. We also ran both samplers beginning from the empty and the actual Insurance graph for which the results are illustrated in figure 4.8. For MB-MCMC both starting points produce models with 45–47 edges. Also here we see that eMC^3 is sensitive to the initial model. Starting

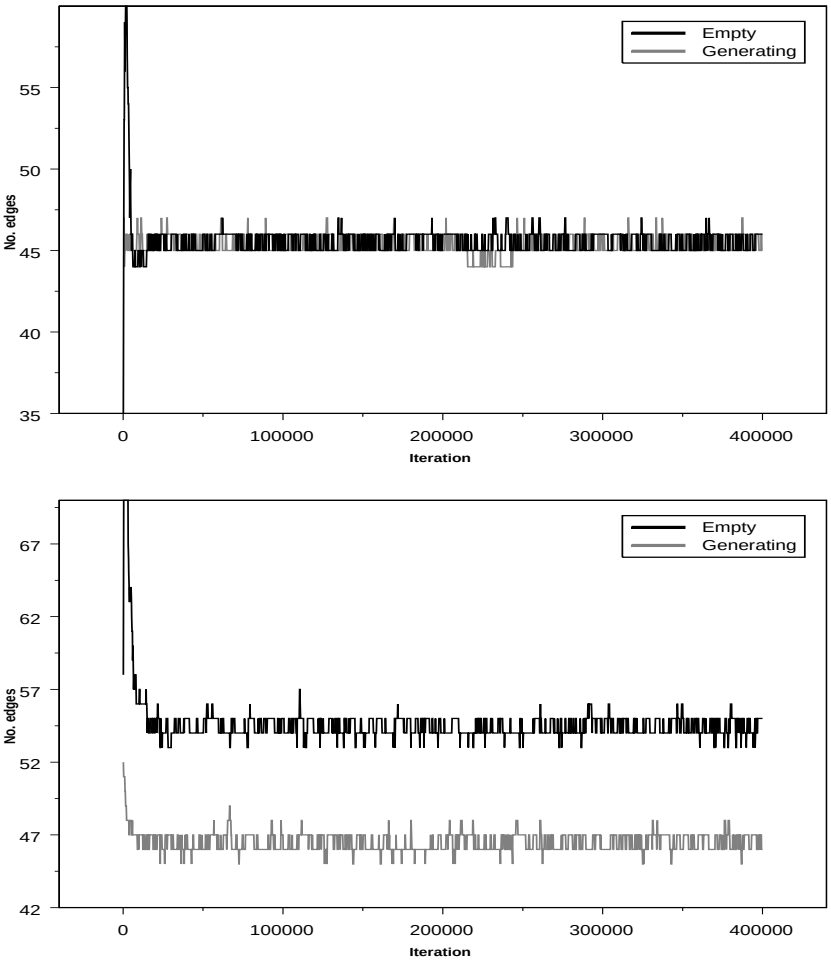

Figure 4.8. Insurance network. Convergence behaviour of the Markov blanket MCMC (*top*) and $e\text{MC}^3$ (*bottom*) given 10,000 records starting from the empty and the data generating model.

from the data generating model, the sampler converges to an invariant distribution where models with 45–47 edges are sampled. Starting from the empty graph, models with 54–56 edges are sampled. We see that even with 10,000 records, there is not enough information in the data sample to support the 52 arcs in the data generating Insurance network.

Observe that for large amounts of data, the fluctuation in the number of edges is larger. This is especially noticeable when comparing the plots in figures 4.7 and 4.8; the variability of the plots for 500 records is larger than for 10,000 records. This is to be expected because there is no pronounced "best" model with merely 500 records. For large amounts of data, the models do not differ a lot, and model selection may in that case be an efficient alternative to MCMC. After all, MB-MCMC is from a computational point of view generally more demanding than most algorithms developed for model selection.

3.3.6 Conclusion

MB-MCMC is a Bayesian approach to learning Bayesian network models. Being Bayesian is beneficial when model learning is based on a relatively small amount of data; in practice this may often be the case. When data is scarce, there may be several models that are structurally quite different from each other, yet are almost equally likely given the data. In other words, the data supports several models that differ widely, yet from a scoring perspective are very close. Model selection methods will only return "the best" model, but give no clue as to how and in what respect models differ that score almost equally well.

For large data sets—where "large" of course is related to the number of variables of our domain—the best model is much more likely than any other model, and model selection may be adequate.

By employing MB-MCMC there is no added computational burden compared to existing MCMC approaches. The improvement in mixing and convergence is only due to a wiser decomposition of the joint distribution. From a local perspective, the edges in the Markov blanket of the vertices form a natural dependence relationship in the sense that they constrain each other. This local dependence also from a more global perspective makes sense: dense areas of the DAG is "tougher" to alter than less dense regions.