Adaptive sampling for Bayesian variable selection

DAVID J. NOTT
Department of Statistics, University of New South Wales, Sydney 2052, Australia
D.Nott@unsw.edu.au

ROBERT KOHN
Faculty of Commerce and Economics, University of New South Wales, Sydney 2052, Australia
R.Kohn@unsw.edu.au

SUMMARY
Bayesian methods for variable selection and for dealing with model uncertainty have become increasingly popular in recent years, mostly due to advances in Markov chain Monte Carlo computational algorithms. In this paper we consider adaptive Markov chain Monte Carlo schemes for Bayesian variable selection in Gaussian linear regression models which improve on standard Metropolis-Hastings algorithms by making use of accumulated information about the posterior distribution during sampling for the construction of proposal distributions. Adaptation needs to be done very carefully to ensure that sampling is from the correct ergodic distribution. We give conditions for the validity of an adaptive sampling scheme in this problem, and for simulating from a distribution on a finite state space in general, and suggest a class of adaptive proposal densities which uses best linear prediction to approximate the Gibbs sampler. Our sampling scheme is computationally much faster per iteration than the Gibbs sampler, and when this is taken into account the efficiency gains when using our sampling scheme compared to alternative approaches are substantial in terms of precision of estimation of posterior quantities of interest for a given amount of computation time. We compare our method with other sampling schemes for examples involving both real and simulated data. The methodology developed in the article can be extended to variable selection in more general problems.

Some key words: Best linear prediction, Bayesian model selection, Markov
Markov chain Monte Carlo simulation has proved an invaluable tool for carrying out the calculations required for Bayesian inference in complex statistical models. In this paper we consider adaptive modifications of standard Markov chain Monte Carlo methods for sampling from the posterior distribution in Bayesian variable selection problems for Gaussian linear regression models with a large number of potential predictors. Variable selection in linear regression is an important problem in itself, but is also important as the simplest case of a class of important model selection problems in regression. Problems of model and variable selection arise in many different contexts such as generalized linear models (Raftery, 1996) survival models (Volinsky et al., 1997), graphical models (Madigan and York, 1995) and covariance selection models (Wong, Carter and Kohn, 2003). We wish to stress at the outset that the methods considered in this paper are potentially applicable to all these problems, although we focus on variable selection in Gaussian linear regression in the present paper.

Write \( y = (y_1, \ldots, y_n)^T \) for a collection of \( n \) observed responses and let \( x_i = (x_{i0}, \ldots, x_{ik})^T \) be a \( p \times 1 \) \((p = k + 1)\) vector of predictors for \( y_i, i = 1, \ldots, n \). Also, write \( X \) for the \( n \times p \) design matrix with \( i \)th row \( x_i^T \). A Gaussian linear regression model takes the form

\[
y_i = x_i^T \beta + \epsilon_i
\]

where \( \beta = (\beta_0, \ldots, \beta_k)^T \) is a \( p \times 1 \) vector of unknown parameters and the \( \epsilon_i, i = 1, \ldots, n \) are independent zero mean normal errors with unknown variance \( \sigma^2 \). In matrix form,

\[
y = X\beta + \epsilon
\]

where \( \epsilon \sim N(0, \sigma^2 I) \).

In Bayesian variable selection we place a prior on \( \beta \) which allows some components to be zero. If \( \beta_j = 0 \), this means that the \( j \)th predictor variable does not appear in the expression for the mean of \( y \). When there are a large number of potential predictors it may be helpful to do variable selection in order to find simple and interpretable models and to avoid overfitting the data. The sort of prior distributions used in Bayesian variable selection
are also useful for obtaining estimators of predictive quantities of interest through Bayesian model averaging (Hoeting et al., 1999) where weighted means of predictions from models involving different subsets of the predictors are considered, with weights defined by posterior model probabilities.

In what follows we assume that \( x_{i0} = 1 \) so that \( \beta_0 \) is an intercept term and we assume that this is always included. Priors on \( \beta \) used in Bayesian variable selection are usually defined through a vector of latent binary variables \( \gamma = (\gamma_1, ..., \gamma_k)^T \) (George and McCulloch, 1993) where \( \gamma_j = 1 \) if \( \beta_j \neq 0 \) and \( \gamma_j = 0 \) if \( \beta_j = 0 \). Different values for \( \gamma \) define different models, with the \( j \)th predictor being “active” if \( \gamma_j = 1 \). Making inference about which predictor variables are included is the problem of making inference on \( \gamma \).

For a given value of \( \gamma \), we define \( \beta_\gamma \) as the subset of \( \beta \) obtained by extracting the nonzero components \( \beta_i \) of \( \beta \), the components where \( \gamma_i = 1 \). Similarly, we write \( X_\gamma \) for the design matrix obtained by extracting those columns \( i \) of \( X \) where \( \gamma_i = 1 \). Then if model \( \gamma \) holds, we have

\[
y = X_\gamma \beta_\gamma + \epsilon.
\]

Given \( \gamma \), with a conjugate prior \( p(\beta_\gamma|\sigma^2, \gamma) \) on \( \beta_\gamma \), and with a conjugate prior \( p(\sigma^2|\gamma) \) on \( \sigma^2 \), we can obtain an analytic expression for the marginal likelihood

\[
p(y|\gamma) = \int p(y|\beta_\gamma, \gamma, \sigma^2)p(\beta_\gamma|\gamma, \sigma^2)p(\sigma^2|\gamma)d\beta_\gamma d\sigma^2. \quad (1)
\]

See the next section for details. Then inference about \( \gamma \) is based on the posterior distribution \( p(\gamma|y) \) which is obtained from

\[
p(\gamma|y) \propto p(\gamma)p(y|\gamma) \quad (2)
\]

where \( p(\gamma) \) is the prior distribution on \( \gamma \). The next section gives one possible choice for \( p(\gamma) \). To find posterior probabilities from (2) we need to find a normalizing constant which is the sum of the right hand side of (2) over all possible values of \( \gamma \). Since the number of possible values is \( 2^k \), if the number of predictors \( k \) is large calculating this normalizing constant is difficult. For this reason, Markov chain Monte Carlo methods such as the Metropolis-Hastings algorithm which do not require knowledge of the normalizing constant are used to simulate from \( p(\gamma|y) \) so that inferences can be made from the posterior distribution. Sampling schemes for Bayesian variable selection in Gaussian linear regression models have been considered by George and
McCulloch (1993), Smith and Kohn (1996), Raftery, Madigan and Hoeting (1997) who use the \((MC)^3\) algorithm of Madigan and York (1995), Dennison et al. (1998) and Kohn et al. (2001). George and McCulloch (1997) give a survey of different sampling schemes and prior specifications for Bayesian variable selection in linear models. See also Dellaportas et al. (2002). Another survey focusing on prior specifications is Fernández et al. (2001). Some methods which are particularly well suited to the situation where the posterior distribution is multimodal are also considered by Liang and Wong (2000) and Nott and Green (2004).

The purpose of the present paper is to modify standard Markov chain Monte Carlo algorithms for the Bayesian variable selection problem such as those considered in Kohn, Smith and Chan (2001) to allow adaptation or learning from the past in the construction of proposal distributions. We give a result stating conditions for the validity of an adaptive sampling scheme in this problem, and for simulating from a distribution on a finite state space in general using adaptive methods, and then suggest a particular choice of adaptive proposal distribution for Bayesian variable selection. In particular, we suggest the use of best linear predictors constructed from a running mean and covariance matrix for \(\gamma\) as a computationally cheap method of approximating the Gibbs sampler. Compared to alternative sampling schemes, increased precision of estimation of posterior quantities of interest can be achieved for a given amount of computation time using our suggested proposal distribution.

The structure of the paper is as follows. In Section 2, we outline the prior distributions used. In Section 3, we describe standard Metropolis-Hastings algorithms for the variable selection problem and the adaptive modifications of these algorithms that are the focus of this paper. We state a result giving conditions for the validity of an adaptive sampling scheme in this problem, and in general for sampling from a distribution on a finite state space. The development of adaptive sampling schemes has been the focus of much recent interest (Gilks et al., 1998, Haario et al., 2001, Atchade and Rosenthal, 2003, Pasarica and Gelman, 2003, Andrieu and Moulines, 2003). Development of a general theory for adaptive sampling schemes is a matter of some technical difficulty, and we believe that outlining a theory for adaptation in the finite state space case where proofs of results can be obtained using only elementary methods is of some value. In Section 4, we discuss adaptive sampling using best linear predictor proposal distributions which provides a computationally cheap approximation to the Gibbs sampler. Our adaptive sampler
has a similar efficiency per iteration to the Gibbs sampler, but each iteration takes much less computation time. In Section 5, we compare performance of our adaptive sampling schemes with various non-adaptive schemes such as the Gibbs sampler, the Metropolized Gibbs sampler of Liu (1996) and a Metropolis-Hastings scheme suggested by Kohn et al. (2001). Section 6 gives some discussion and conclusions.

2. PRIOR DISTRIBUTIONS

In the model of Section 1, the unknown parameters are \((\gamma, \sigma^2, \beta_\gamma)\) and for Bayesian inference we need to specify a prior distribution on these quantities. Our prior is specified hierarchically as

\[
p(\gamma, \sigma^2, \beta_\gamma) = p(\gamma)p(\sigma^2|\gamma)p(\beta_\gamma|\sigma^2, \gamma).
\]  

(3)

We specify each of the terms on the right hand side of (3) in turn. First we need some notation. Without loss of generality we assume that the entries in each column of \(X\) have mean zero so that the predictors have been centred, except for the first column which corresponds to the intercept and which has been scaled to have length 1. Then we can write

\[
X^\gamma X_\gamma = \begin{bmatrix}
1 & 0 \\
0 & A_\gamma
\end{bmatrix},
\]

where writing \(q_\gamma\) for the number of columns in \(X_\gamma\), \(A_\gamma\) is a \((q_\gamma - 1) \times (q_\gamma - 1)\) matrix. Our prior \(p(\beta_\gamma|\gamma, \sigma^2)\) is normal, with mean \(\hat{\beta}_\gamma = (X^T_\gamma X_\gamma)^{-1}X^T_\gamma y\) and covariance matrix

\[
\Sigma_\gamma = \begin{bmatrix}
c_1\sigma^2 & 0 \\
0 & c_2\sigma^2 A_\gamma^{-1}
\end{bmatrix},
\]

where \(c_1\) and \(c_2\) are constants. Following Cripps et al. (2003) we choose \(c_1 = n^2\) and \(c_2 = n\). The prior mean for \(\beta_\gamma\) here depends on the data so that we are using an empirical Bayes approach. The rationale for our prior is that it is centred in a way consistent with the likelihood but is diffuse compared to the likelihood. The prior covariance for \(\beta_1, \ldots, \beta_k\) is approximately independent of \(n\), and the prior on \(\beta_0\) is increasingly diffuse as \(n \to \infty\). This prior also has attractive invariance properties – see Kohn et al. (2001) and Cripps et al. (2003) for further discussion of this prior distribution.

For our prior \(p(\sigma^2|\gamma)\) on \(\sigma^2\), we use an inverse gamma prior

\[
p(\sigma^2|\gamma) \sim IG \left( \frac{\kappa}{2} - 1, \frac{\kappa SSE(\gamma)}{2(n - q_\gamma)} \right)
\]

5
where \( SSE(\gamma) = y^T y - y^T X_\gamma (X_\gamma^T X_\gamma)^{-1} X_\gamma^T y \) is the residual sum of squares for model \( \gamma \) and \( \kappa \) is a constant which following Cripps et al. (2003) we choose as \( \kappa = 7 \). The rationale for this prior is that it is centred in a way consistent which the likelihood, having a mode at the unbiased estimator \( SSE(\gamma)/(n - q_\gamma) \) of \( \sigma^2 \), and is diffuse compared to the marginal likelihood for \( \sigma^2 \). Again see Cripps et al. (2003) for further discussion. Finally, for our prior on \( \gamma \) we use a prior on the number of active terms that is beta-binomial,

\[
p(\gamma) = \frac{B(q_\gamma - 1 + a_\pi, p - q_\gamma + b_\pi)}{B(a_\pi, b_\pi)}
\]

where \( B(\cdot, \cdot) \) is the beta distribution. This prior arises from assuming each predictor has probability \( \pi \) of inclusion independently,

\[
p(\gamma|\pi) = \pi^{q_\gamma - 1}(1 - \pi)^{p - q_\gamma}
\]

and putting a beta prior on \( \pi \):

\[
p(\pi) = \frac{\pi^{a_\pi - 1}(1 - \pi)^{b_\pi - 1}}{B(a_\pi, b_\pi)}
\]

\( 0 \leq \pi \leq 1 \).

The parameters \( a_\pi \) and \( b_\pi \) can be chosen to achieve a certain mean and standard deviation for the number of active terms (Kohn et al., 2001). Integrating out \( \pi \) gives the prior \( p(\gamma) \) above. This beta-binomial prior was introduced by Kohn et al. (2001).

With the priors we have chosen on \( \beta_\gamma \) and \( \sigma^2 \) it is possible to find the marginal likelihood \( p(y|\gamma) \) (equation (1)) analytically. After integrating out \( \beta_\gamma \) as a normal integral and \( \sigma^2 \) as an inverse gamma integral we obtain (Cripps et al., 2003)

\[
p(y|\gamma) = (c_1 + 1)^{-1/2}(c_2 + 1)^{-\frac{2 q_\gamma - 1}{2}}\left(\frac{SSE(\gamma)}{2(n - q_\gamma)}\right)^{-\frac{q_\gamma}{2}}(n + \kappa - q_\gamma)^{-\frac{n + p - 2}{2}} \times K(\kappa)
\]

where

\[
K(\kappa) = (2\pi)^{-n/2}\Gamma(\kappa/2 - 1)^{-1}\Gamma\left(\frac{n + \kappa - 2}{2}\right)^{\kappa(\kappa - 1)},
\]

where \( \Gamma(\cdot) \) is the gamma function. Note that it is possible to obtain an analytic expression for \( p(y|\gamma) \) for alternative conjugate prior specifications to the one we have used here, but the focus of this paper is on sampling schemes for exploring the posterior distribution, and the prior specifications
given above are the ones used throughout the rest of this paper. If an analytic expression for \( p(y|\gamma) \) is not available then reversible jump Markov chain Monte Carlo (Green, 1995) can be used for the Bayesian variable selection problem (Denison et al. 1998).

3. METROPOLIS-HASTINGS AND ADAPTIVE
METROPOLIS-HASTINGS ALGORITHMS

3.1 Standard Metropolis-Hastings Schemes

Markov chain Monte Carlo methods for simulating from \( p(\gamma|y) \) proceed by construction of a Markov chain \( \Gamma = \{\gamma^{(n)}; n \geq 0\} \) with stationary distribution \( p(\gamma|y) \). Then starting with some arbitrary initial value \( \gamma^{(0)} \) we can simulate the chain and for large \( n \), \( \gamma^{(n)} \) will have approximately the posterior distribution \( p(\gamma|y) \). Under certain conditions posterior expectations of interest can be estimated by averaging over a sample path of \( \Gamma \):

\[
E(h(\gamma)|y) \approx \frac{1}{s} \sum_{i=b+1}^{b+s} h(\gamma^{(i)})
\]

where in taking the average the first \( b \) iterates have been discarded as a “burn in” period where the distribution of the state of the chain is affected by the initial value and may not be typical of the stationary distribution. For further introductory background on Markov chain Monte Carlo see Gilks et al. (1996), Liu (2001) or Gelman et al. (2003).

Mostly in Bayesian variable selection some kind of Metropolis-Hastings algorithm will be used to construct the Markov chain \( \Gamma \). We do not describe the Metropolis-Hastings algorithm in general here but refer the reader to the introductory references above. In this paper we consider three different Metropolis-Hastings schemes for sampling from \( p(\gamma|y) \). The first is the Gibbs sampler. We introduce some more notation before describing it. Write \( p(\gamma_l|\gamma_{\neq l}, y) \) for the posterior conditional distribution of \( \gamma_l \) given all other components of \( \gamma \). If \( j = (j_1, ..., j_k) \) is a permutation of \( (1, ..., k) \), then we write \( \gamma_{<l}(j) = (\gamma_{j_1}, ..., \gamma_{j_{l-1}})^T \) and \( \gamma_{>l}(j) = (\gamma_{j_{l+1}}, ..., \gamma_{j_k})^T \). In the Gibbs sampler to generate iterates \( \gamma^{(0)}, ..., \gamma^{(b+s)} \) we proceed as follows.

1. Initialize \( \gamma^{(0)} \).
2. For \(i = 1, \ldots, b + s\)
   
   (a) Generate a random permutation \(j = (j_1, \ldots, j_k)\) of \((1, \ldots, k)\).
   
   (b) For \(l = 1, \ldots, k\), draw \(\gamma_{jl}^{(i)}\) from \(p(\gamma_{jl} | \gamma_{\neq jl} = (\gamma_{<l}^{(i+1)}(j), \gamma_{>l}^{(i)}(j)), y)\)

Sampling from one of the full conditional distributions in step 2, \(p(\gamma_l | \gamma_{\neq l}, y)\) say, is easily done. We have

\[
p(\gamma_l | \gamma_{\neq l}, y) \propto p(\gamma)p(y | \gamma)
\]

so since there are only two possible values for \(\gamma_l\) we simply evaluate the right hand side for \(\gamma_l = 0\) and 1 with \(\gamma_{\neq l}\) fixed and normalize. Writing \(\gamma^1 = (\gamma_1, \ldots, \gamma_{l-1}, 1, \gamma_{l+1}, \ldots, \gamma_k)^T\) and \(\gamma^0 = (\gamma_1, \ldots, \gamma_{l-1}, 0, \gamma_{l+1}, \ldots, \gamma_k)^T\) we obtain

\[
p(\gamma_l = 1 | \gamma_{\neq l}, y) = \frac{p(\gamma^1)p(y | \gamma^1)}{p(\gamma^0)p(y | \gamma^0) + p(\gamma^1)p(y | \gamma^1)}
\]

and of course \(p(\gamma_l = 0 | \gamma_{\neq l}, y) = 1 - p(\gamma_l = 1 | \gamma_{\neq l}, y)\).

The second sampling scheme we consider is the Metropolized Gibbs sampler of Liu (1996). In this sampling scheme proposal values for \(\gamma_j\) are generated from the posterior conditional distribution for \(\gamma_j\) under the constraint that the proposal does not equal the current state. In this case, where \(\gamma_j\) is binary, this involves always proposing to switch from the current state. Liu (1996) shows that this sampling scheme is more efficient than an implementation of the Gibbs sampler in a certain sense applying a general result of Peskun (1973). The Metropolized Gibbs sampling algorithm proceeds in exactly the same way as for the Gibbs sampler above except that in step 2 b) if we write \(p\) for \(p(\gamma_{jl} = \gamma_{jl}^{(i)} | \gamma_{\neq jl} = (\gamma_{<l}^{(i+1)}(j), \gamma_{>l}^{(i)}(j)), y)\) then we flip to the opposite value for \(\gamma_{jl}\) rather than staying with the current with probability

\[
\min \left(1, \frac{1 - p}{p}\right).
\]

The last of the Metropolis-Hastings sampling schemes we consider, which we call the Kohn-Smith-Chan scheme, was proposed by Kohn et al. (2001) and is particularly efficient in problems with a large number of predictors and many useless predictors. The Kohn-Smith-Chan scheme simulates proposal values from the conditional prior distribution. In Kohn et al., 2001, the prior \(p(\gamma)\) of Section 2 was used, and simulations from the conditional prior are
computationally inexpensive. The whole idea is not to waste too much time on updating useless predictors. When most predictors are useless, the current state for $\gamma_j$ corresponding to a useless predictor will most likely be zero, and a proposal simulated from the conditional prior for the prior of Section 2 will also most likely be zero since the conditional prior probability of a predictor being active is based on the fraction of other predictors currently in the model. See Kohn et al. (2001) for details. This means that in a situation where most predictors are useless, in updating the indicator for a useless predictor proposed and current states are likely to be the same and there is no need to compute the Metropolis-Hastings acceptance probability. The acceptance probability will involve the marginal likelihood $p(y|\gamma)$ for proposed and current states, which is expensive to compute, unlike the conditional prior values used in the Metropolis-Hastings proposal, which are inexpensive to compute. In simulation studies Kohn et al. (2001) show that while the Gibbs sampler is more efficient per iteration than their scheme, when computation time is taken into account a greater precision of estimation of posterior quantities of interest can be achieved for a given amount of computation time because their scheme is much faster per iteration. The Kohn-Smith-Chan scheme proceeds as follows.

1. Initialize $\gamma^{(0)}$.

2. For $i = 1, ..., s$,
   (a) Generate a random permutation $j = (j_1, ..., j_k)$ of $(1, ..., k)$.
   (b) For $l = 1, ..., k$,
      i. Generate a proposal value $\gamma_{jl}^P$ for $\gamma_{jl}$ from the conditional prior distribution for $\gamma_{jl}$, $p(\gamma_{jl} | \gamma_{\neq jl}) = (\gamma_{l\leq i+1}(j), \gamma_{(i+1)\leq l}(j))$.
      ii. With $\gamma^C = (\gamma_{l\leq i+1}(j), \gamma^{(i)}_{jl}, \gamma_{(i+1)\leq l}(j))$ and $\gamma^P = (\gamma_{l\leq i+1}(j), \gamma^P_{jl}, \gamma^{(i)}_{(i+1)\leq l}(j))$ set $\gamma^{(i+1)}_{jl} = \gamma^P_{jl}$ with probability $\min(1, \alpha)$ where
         \[ \alpha = \frac{p(y|\gamma^P)}{p(y|\gamma^C)} \]
         and set $\gamma^{(i+1)}_{jl} = \gamma^{(i)}_{jl}$ otherwise.

Note that in the acceptance probability $\alpha$ of the Kohn-Smith-Chan scheme the usual prior and proposal ratios in the Metropolis-Hastings acceptance
probability cancel, so that the acceptance probability is just a ratio of marginal likelihood values. If the proposed and current values are the same, then the acceptance probability is 1, and there is no need to compute any marginal likelihood values.

### 3.2 Adaptive Markov chain Monte Carlo algorithms

In this paper we are interested in improving on the Metropolis-Hastings schemes above by devising sampling schemes where proposal distributions are based on sampled values at previous iterations. We want to use what we have learned about the posterior distribution so far to construct good proposal distributions which are fast to compute. Adaptation needs to be done with considerable care since we are abandoning a Markovian framework in allowing proposal distributions to possibly depend on the whole past history of the chain. We now outline a general theory for adaptation for sampling distributions on a finite state space such as those encountered in the Bayesian variable selection problem, before going on to discuss some specific adaptive proposal distributions for Bayesian variable selection. We state a result now that uses similar conditions to those proposed by Atchadé and Rosenthal (2003). In fact, the results we give in this section can be deduced as a corollary to those of Atchadé and Rosenthal (2003). However, we consider the case of a finite state space and give an elementary and transparent proof that does not rely on the mixingale arguments given by Atchadé and Rosenthal (2003) for the general state space case. We believe this is of considerable value in explaining and popularizing the important idea of adaptive sampling.

Let \( \mathcal{Z} = \{Z_n : n \geq 0\} \) be a process on a finite state space \( \mathcal{Z} \) evolving according to a collection of transition probabilities

\[
T_n(z, z') = \text{pr}(Z_{n+1} = z'|Z_n = z, Z_{n-1} = z_{n-1}, ..., Z_0 = z_0).
\]

It is important to remember that \( T_n(z, z') \) depends on \( z_{n-1}, ..., z_0 \) although for notational simplicity we have not shown this explicitly. Writing \( p(z_n) \), \( n \geq 0 \), for the distribution of \( Z_n \), we can state the following theorem.
Theorem 1:

Suppose that for every $n$ and $z_0, ..., z_{n-1} \in \mathcal{Z}$ and for some distribution $\pi(z)$ on $\mathcal{Z}$

$$
\sum_{\mathcal{Z}} \pi(z_n)T_n(z_n, z_{n+1}) = \pi(z_{n+1})
$$

(4)

and that there exists an $\epsilon > 0$ (not depending on $n, z_0, ..., z_{n-1}$) with

$$
T_n(z, z') \geq \epsilon \pi(z').
$$

(5)

If also

$$
|T_n(z, z') - T_{n+k}(z, z')| \leq a_n c_k
$$

(6)

where $a_n = O(n^{-r_1})$ and $c_k = O(k^{r_2})$ for some $r_1, r_2 > 0$ then for any initial distribution $p(z_0)$ for $Z_0$

$$
\sup_{z_n} |p(z_n) - \pi(z_n)| \to 0
$$

as $n \to \infty$.

**Proof:**

See Appendix A.

Theorem 1 says that if condition (4) holds (which is a kind of balance condition) and if the uniform ergodicity condition (5) holds, and if the kernels $T_n(z, z')$ behave for large $n$ like a fixed kernel (stability condition (6)) then the distribution of $Z_n$ converges to $\pi(z)$. For our applications we require an ergodic theorem which will allow estimation of expectations with respect to $\pi(z)$ by averages over a sample path.

Theorem 2:

Under the conditions of Theorem 1, for any measurable function $h : \mathcal{Z} \to R$,

$$
\frac{1}{n+1} \sum_{i=0}^{n} h(Z_i) \to E_\pi(h(Z))
$$
almost surely.

**Proof:**

See Appendix A.

The two results above give a fairly general framework for adaptation in the case of sampling from a distribution on a finite state space. If we are modifying the conventional Metropolis-Hastings algorithm for sampling a distribution $\pi(z)$ by allowing the proposal distribution at step $n$ to depend on the past, then we can do this under some natural conditions. In general, a Metropolis-Hastings kernel where any tuning parameters in the proposal are chosen depending only on $z_0, ..., z_{n-1}$, but not on $z_n$, will satisfy the balance condition (4). In giving results on adaptive sampling for a general state space Atchadé and Rosenthal (2003) introduce a stability condition similar to our condition (6) for the transition kernels following similar reasoning to that given in Haario et al. (2001). Both Haario et al. (2001) and Atchadé and Rosenthal (2003) use mixingale theory to prove an ergodic theorem for the adaptive process. Here we have specialized to the case of a finite state space and given a proof that avoids mixingale arguments and uses only elementary methods. See Appendix A. The proof also extends readily to the case of a compact state space.

### 3.3 Adaptive Metropolis-Hastings Schemes for Variable Selection

We now describe an adaptive Metropolis-Hastings scheme for Bayesian variable selection. Our proposal distributions make use of the estimated posterior mean and covariance matrix of $\gamma$ from previously sampled iterates. We assume that the sampling scheme is run for $ad$ iterations, where adaptation of the proposal distribution occurs after every $d$ iterations. Adaptation is only done after every $d$ iterations for computational reasons. We state the algorithm first before giving some further remarks.

1. Initialize $\gamma^{(0)}$.

2. For $m = 1, ..., d$, generate $\gamma^{(m)}$ using a Markov chain Monte Carlo scheme such as the Kohn-Smith-Chan scheme or Metropolized Gibbs sampler.
3. For $i = 1, \ldots, a - 1$, Calculate $\bar{\gamma}(i) = \text{mean}(\gamma^{(1)}, \ldots, \gamma^{(id-1)})$ and $\hat{W}(i) = (\hat{\Sigma}(i) + \lambda I)^{-1}$ where $\hat{\Sigma}(i) = \text{cov}(\gamma^{(1)}, \ldots, \gamma^{(id-1)})$ and $\lambda$ is a non-negative constant. For $m = 1, \ldots, d$,

(a) Generate a random permutation $j = (j_1, \ldots, j_k)$ of $(1, \ldots, k)$.

(b) For $l = 1, \ldots, k$,

i. Calculate

$$\psi = \hat{\gamma}(i)_{j_l} - d(i, j_l)^{-1} w(i, j_l) \hat{w}^T((\gamma_{\neq j_l} - \hat{\gamma}(i)_{\neq j_l})$$

where as before $\gamma_{\neq l}$ denotes the vector obtained by deleting $\gamma_l$ from $\gamma$, and we have written $d(i, j)$ for the $j$th diagonal element of $\hat{W}(i)$ and $w(i, j)$ for the vector of length $k - 1$ obtained by deleting the $j$th entry from the $j$th column of $\hat{W}(i)$. Write $\gamma^C = (\gamma^{(id+m)}, \gamma^{(id+m-1)}_{j_l}, \gamma^{(id+m-1)}_{\neq j_l})$. Generate a proposal value $\gamma^P_{j_l}$ for $\gamma_{j_l}$ from a proposal distribution $q(\gamma^P_{j_l}|\gamma^C)$ where

$$q(\gamma^P_{j_l} = 1|\gamma^C) = \begin{cases} 
\psi & \text{if } \delta \leq \psi \leq 1 - \delta \\
\delta & \text{if } \psi < \delta \\
1 - \delta & \text{if } \psi > 1 - \delta 
\end{cases}$$

where $0 < \delta < 0.5$ is a user specified parameter.

ii. Let $\gamma^P = (\gamma^{(id+m)}_{\neq j_l}, \gamma^P_{j_l}, \gamma^{(id+m-1)}_{\neq j_l})$. Set $\gamma^{(id+m)}_{j_l} = \gamma^P_{j_l}$ with probability $\min\{1, \alpha\}$ where

$$\alpha = \frac{p(\gamma^P)p(y|\gamma^P)q(\gamma^P_{j_l}|\gamma^P)}{p(\gamma^C)p(y|\gamma^C)q(\gamma^P_{j_l}|\gamma^C)}.$$ 

Otherwise, $\gamma^{(id+m)}_{j_l} = \gamma^{(id+m-1)}_{j_l}$.

Some of the steps in the above algorithm require some further explanation. In step 2, we run a non-adaptive algorithm for an initial period of $d$ iterations so that we can learn about the posterior in order to apply our adaptive proposal distribution. Then in step 3 we run our adaptive sampling scheme for $a - 1$ further blocks of $d$ iterations, changing our adaptive proposal at the end of each block. If the parameter $\lambda$ is set to 0 in step 3, then $\psi$ is the best linear predictor of $\gamma_{j_l}$ given $\gamma_{\neq j_l} = \gamma^C_{\neq j_l}$ with the posterior mean and covariance estimated from previous iterations. The best linear predictor can
be used as an estimate of $E(\gamma_{jl}|\gamma^C_{jl}, y)$. For a Gaussian random vector the best linear predictor and conditional mean are the same which gives some motivation for this. Since $\gamma_{jl}$ is binary the above conditional expectation is equal to $\text{pr}(\gamma_{jl} = 1|\gamma^C_{jl}, y)$, which defines the posterior full conditional distribution in the Gibbs sampler. Truncating our estimate of this conditional probability gives us the adaptive proposal distribution used: we truncate small probabilities to 0 and large probabilities to 1 where in the examples below we chose $\delta = 0.01$. The truncation bounds the proposal ratio in the Metropolis-Hastings acceptance probability which improves mixing and ensures the validity of the adaptive scheme according to the theorems of the last section.

Usually we set the parameter $\lambda$ in step 3 to be some very small but positive constant to ensure that the inverse of $\Sigma(i) + \lambda I$ exists. The sample covariance matrix of past iterates for $\gamma$ can be singular if, for instance, one of the $\gamma_{jl}$’s is constant for all iterates. If we set $\lambda$ to be very large, this effectively means that $\psi$ is the estimated marginal posterior probability of inclusion $\hat{\gamma}(i)_{jl}$. Using this adaptive proposal distribution corresponding to a large $\lambda$ value is of interest in itself and we comment on this later. Our adaptive proposal distribution has some of the same advantages as the Kohn-Smith-Chan scheme described in the last section. Our proposal distribution is inexpensive to compute (since $\hat{\gamma}(i)_{jl}$, $d(i, jl)$ and $w(i, jl)$ in the calculation of $\psi$ at step 3 b) don’t depend on the value of $\gamma_{jl}$) and for predictors where the decision about whether or not to include them in the model is clear, it is likely that our proposal and current values are the same at each iteration and we do not waste much time on updating these predictors. In many Bayesian variable selection problems with large numbers of predictors many of the components of $\gamma$ never change their value and we want to focus computational effort on those predictors where there is some doubt about their inclusion. Unlike the Kohn-Smith-Chan scheme, our proposal does not treat all predictors in the same way and it makes intelligent use of the dependence between variables through the use of the best linear predictor for constructing proposals. Effectively we are using best linear prediction to approximate the Gibbs sampler but at much less computational expense, since the Gibbs sampler requires computation of the marginal likelihood each time a component of $\gamma$ is updated. Our sampling scheme can also be useful for doing block updating of predictors. That is, instead of updating each component of $\gamma$ singly with the other components held fixed we might consider changing two or more components of $\gamma$ simultaneously. Block updating is discussed in Appendix.
4. EXAMPLES

In this section we compare performance of our adaptive sampling schemes with alternative approaches for three simulated data sets and one real data set.

4.1 Example: Simulated data set with multicollinearity

Our first example involves a simulated data set and is similar to an example considered in Raftery et al. (1997) and in Fernández et al. (2001). We generate a design matrix \( X \) of dimensions 50 \( \times \) 16 by generating the first column \( x_0 \) as an intercept term, the next ten columns \( x_1 - x_{10} \) as independent standard normal variates and then the last 5 columns \( x_{11} - x_{15} \) are generated from \( x_1 - x_5 \) by

\[
[x_{11}, ..., x_{15}] = [x_1, ..., x_5] \times [0.3, 0.5, 0.7, 0.9, 1.1]' \times [1, 1, 1, 1, 1] + E
\]

where \( E \) is a 50 \( \times \) 5 matrix of independent standard normal variates. In this design matrix, there is strong correlation amongst the last 5 predictors \( x_{11} - x_{15} \) and moderate correlation between \( x_1 - x_5 \) and \( x_{11} - x_{15} \). The responses \( y \) were then generated as

\[
y = 4 + 2x_1 - x_5 + 1.5x_7 + x_{11} + 0.5x_{13} + \epsilon
\]

where \( \epsilon \sim N(0, 2.5^2 I) \).

We compare a number of different sampling schemes for this problem. The sampling schemes are the Gibbs sampler, the Metropolized Gibbs sampler, the Kohn-Smith-Chan scheme, the Kohn-Smith-Chan scheme with predictors updated in randomly chosen blocks of size 2, our one at a time adaptive scheme with \( \lambda = 0.001 \), our adaptive scheme with blocks of size 2 and \( \lambda = 0.001 \) and our adaptive scheme with blocks of size 3 and \( \lambda = 0.001 \). See Kohn et al. (2001) for further details of the block Kohn-Smith-Chan scheme.

We have also experimented with our adaptive scheme using \( \lambda = \infty \) where proposals are based on estimated marginal posterior probabilities of inclusion but these sampling schemes are generally inferior to our adaptive schemes.
with a small value for \( \lambda \). Other schemes that we might have considered such as the Gibbs sampler or Metropolized Gibbs sampler with blocking are not considered here because implementation of these samplers with blocking is very computationally expensive and in general these schemes are not competitive, in terms of precision of estimation achieved for a given amount of computational effort, with the other block updating schemes considered here.

We need to outline how comparisons are made between the different sampling schemes. Following George and McCulloch (1997), we consider the estimated marginal probability of inclusion for each of the fifteen predictors \( X_1 - X_{15} \):

\[
\tilde{\gamma}_j = \frac{1}{s} \sum_{i=b+1}^{b+s} \gamma_{j(i)}
\]

for \( j = 1, \ldots, 15 \) where as before \( s \) is the number of sampling iterates and \( b \) is the number of burn-in iterates. An estimate of the variance of \( \tilde{\gamma}_j \) is

\[
\frac{\hat{\sigma}^2}{s} \left( 1 + 2 \sum_{i=1}^{M} \left( 1 - \frac{i}{s} \right) \hat{\rho}(i) \right)
\]

where \( \hat{\sigma}^2 \) is the sample variance of the \( \gamma_j \) iterates after burn in, \( \hat{\rho}(i) \) is the estimated autocorrelation function of the \( \gamma_j \) iterates after burn in, and \( M \) is some truncation point where the autocorrelation of the \( \gamma_j \) sequence is thought to be negligible. The above expression is an estimator of the variance of \( \tilde{\gamma}_j \) based on a general expression for an estimator of the variance of an ergodic average for a stationary time series. If the variance estimate is smaller for one sampling scheme compared to another then this means that the sampling scheme achieving the smaller standard error achieves a greater estimated precision of estimation for the marginal probability of inclusion for that variable. When comparing different sampling schemes it is helpful to use the so-called estimated inefficiency factor

\[
\left( 1 + 2 \sum_{i=1}^{M} \left( 1 - \frac{i}{s} \right) \hat{\rho}(i) \right)
\]

which divides the variance estimate above by the variance estimate for an independent sequence. It measures the estimated factor by which the number of iterates must increase to achieve the same precision of estimation as estimation based on an independent sequence.
Table 1 reports for the predictor variables $X_2 - X_{15}$ and each of the sampling schemes the estimated inefficiency factors of the $\gamma_j$. We do not report the inefficiency factor for $X_1$ since the probability of inclusion for this predictor is almost 1. Also reported are the acceptance rates of proposals for the sampling schemes and the time taken for sampling. Programs were written in MATLAB and run on a Pentium IV 2.3 GHz PC. The number of burn-in iterates $b$ was 5000 for each sampling scheme, and the number of sampling iterates $s$ was 50000. In the adaptive schemes, adaptation was performed every 5000 iterations. The prior parameters $a_\pi$ and $b_\pi$ in $p(\gamma)$ were chosen so that the expected number of active terms was 5 with a standard deviation of 5. As can be seen from the table, the Gibbs and Metropolized Gibbs schemes are generally most efficient per iteration, but the computation time is much higher than for the other schemes. The adaptive schemes are almost as efficient per iteration as the Gibbs and Metropolized Gibbs schemes but are much less computationally expensive to implement, being 2 to 3 times faster. For the blocked adaptive schemes the acceptance rate remains high even for blocks of size 3 suggesting that our best linear predictor proposal is doing a good job of approximating the Gibbs sampler. This is confirmed by a comparison of our adaptive proposals with the block conditional distributions. Surprisingly, blocking doesn’t seem to help much in this example for increasing the efficiency per iteration, although we have experimented with other examples where the multicollinearity is more severe and in these cases blocking does provide an improvement.

4.2 Example: Simulated data set with many predictors

In our second example we generate a $50 \times 31$ design matrix $X$ where the first column $X_0$ is an intercept and the remaining 30 columns are independent standard normals. We generated the first 16 elements of $\beta$ as independent uniform random variables on the interval $[-1, 1]$. The reason for looking at this example is that we expect the Kohn-Smith-Chan scheme to perform poorly here. Essentially in the Kohn-Smith-Chan scheme where proposals are simulated from the conditional prior with the prior of Section 2, we are likely to propose inclusion if most other variables are included and are likely to propose that a variable not be included if most other variables are not currently included. In situations where most predictors are useless or where most predictors are included the conditional prior proposal is likely
Table 1: Estimated inefficiency factors for $\varpi_j$ for first simulated data set. Estimates are based on 50,000 iterations with 5,000 iterations burn in. In the table method A is the Gibbs sampler, method B the Metropolized Gibbs sampler, methods C and D the Kohn-Smith-Chan schemes with blocks of size 1 and 2 respectively, and methods E, F and G are the adaptive schemes with blocks of size 1, 2 and 3 respectively.

<table>
<thead>
<tr>
<th>Predictor</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X_2$</td>
<td>1.01</td>
<td>0.86</td>
<td>1.83</td>
<td>1.97</td>
<td>0.81</td>
<td>1.09</td>
<td>1.23</td>
</tr>
<tr>
<td>$X_3$</td>
<td>0.98</td>
<td>0.81</td>
<td>2.11</td>
<td>2.37</td>
<td>1.28</td>
<td>1.39</td>
<td>1.35</td>
</tr>
<tr>
<td>$X_4$</td>
<td>1.20</td>
<td>1.08</td>
<td>2.03</td>
<td>3.13</td>
<td>1.27</td>
<td>1.27</td>
<td>1.34</td>
</tr>
<tr>
<td>$X_5$</td>
<td>1.01</td>
<td>0.83</td>
<td>1.46</td>
<td>1.99</td>
<td>1.43</td>
<td>1.47</td>
<td>1.60</td>
</tr>
<tr>
<td>$X_6$</td>
<td>1.39</td>
<td>0.86</td>
<td>3.23</td>
<td>3.71</td>
<td>1.82</td>
<td>2.23</td>
<td>1.83</td>
</tr>
<tr>
<td>$X_7$</td>
<td>1.94</td>
<td>1.10</td>
<td>9.94</td>
<td>10.57</td>
<td>2.97</td>
<td>3.02</td>
<td>2.53</td>
</tr>
<tr>
<td>$X_8$</td>
<td>0.87</td>
<td>0.86</td>
<td>1.79</td>
<td>2.17</td>
<td>1.10</td>
<td>1.12</td>
<td>1.15</td>
</tr>
<tr>
<td>$X_9$</td>
<td>1.05</td>
<td>0.76</td>
<td>1.65</td>
<td>2.32</td>
<td>1.10</td>
<td>1.28</td>
<td>1.34</td>
</tr>
<tr>
<td>$X_{10}$</td>
<td>1.07</td>
<td>0.94</td>
<td>1.71</td>
<td>2.35</td>
<td>1.39</td>
<td>1.12</td>
<td>1.34</td>
</tr>
<tr>
<td>$X_{11}$</td>
<td>3.21</td>
<td>1.91</td>
<td>9.23</td>
<td>12.97</td>
<td>4.66</td>
<td>5.89</td>
<td>4.71</td>
</tr>
<tr>
<td>$X_{12}$</td>
<td>2.70</td>
<td>1.94</td>
<td>5.70</td>
<td>6.78</td>
<td>2.98</td>
<td>3.56</td>
<td>3.42</td>
</tr>
<tr>
<td>$X_{13}$</td>
<td>2.43</td>
<td>1.78</td>
<td>4.99</td>
<td>7.15</td>
<td>3.42</td>
<td>3.46</td>
<td>3.87</td>
</tr>
<tr>
<td>$X_{14}$</td>
<td>1.11</td>
<td>1.03</td>
<td>1.85</td>
<td>2.65</td>
<td>1.31</td>
<td>1.56</td>
<td>1.40</td>
</tr>
<tr>
<td>$X_{15}$</td>
<td>1.26</td>
<td>0.97</td>
<td>1.82</td>
<td>2.59</td>
<td>1.72</td>
<td>1.60</td>
<td>1.42</td>
</tr>
<tr>
<td>Time (Seconds)</td>
<td>444.0</td>
<td>443.0</td>
<td>169.0</td>
<td>166.9</td>
<td>129.5</td>
<td>163.5</td>
<td>213.7</td>
</tr>
</tbody>
</table>

| Acceptance Rate | 1.00 | 0.16 | 0.70 | 0.53 | 0.95 | 0.91 | 0.88 |
to be helpful, but when roughly half the predictors should be included in
the model as in this example we expect that the Kohn-Smith-Chan proposal
might not work well. Our adaptive proposal based on best linear prediction,
however, should continue to work well. We generated the responses as

\[ y = X\beta + \epsilon \]

where \( \epsilon \sim N(0, 2.5^2 I) \) and comparisons between the sampling schemes are
made as in the last example. The number of burn-in and sampling iterates
and the adaptation period for the adaptive schemes are the same as before.
The prior parameters \( a_\pi \) and \( b_\pi \) were again chosen so that the prior mean
number of active predictors was 5, with a standard deviation of 5. We have
reported our results in the same way as for the first example. Because there
are 30 predictors here we report estimated inefficiency factors only for the first
15 predictors, which are those with nonzero coefficients. The results are given
in Table 2. The best sampling schemes in terms of efficiency per iteration are
the Gibbs sampler and Metropolized Gibbs sampler. However, the adaptive
schemes approximate the performance of the Gibbs sampler very well and
are about three times faster. As expected, the Kohn-Smith-Chan schemes
perform poorly in this example. They are slower than the adaptive schemes,
have a low acceptance rate and are the worst of the sampling schemes in
terms of efficiency per iteration.

4.3 Example: Nonparametric regression with variable selection on basis
functions

In this example we look at an application to nonparametric regression,
following an example in Kohn et al. (2001). Kohn et al. (2001) consider
approaching the problem of flexible regression by doing Bayesian variable
selection on a set of predictors derived from a flexible basis function expan-
sion which is able to approximate any suitably smooth mean function. This
approach to nonparametric regression was introduced in Smith and Kohn
(1996). In this example we consider a simulated data set where \( n = 600 \)
observations are generated from the model

\[ y_i = f(z_i) + \epsilon_i \]

\( i = 1, \ldots, 600 \) where the errors \( \epsilon_i \) are independent \( N(0, 0.5^2) \), \( f(z_i) \) is the mean
function and the predictors \( z_i = (z_{i1}, z_{i2}) \) are generated uniformly on the unit
Table 2: Estimated inefficiency factors for $\gamma_j$ for second simulated data set. Estimates are based on 50,000 iterations with 5,000 iterations burn in. In the table method A is the Gibbs sampler, method B the Metropolized Gibbs sampler, methods C and D the Kohn-Smith-Chan schemes with blocks of size 1 and 2 respectively, and methods E, F and G are the adaptive schemes with blocks of size 1, 2 and 3 respectively.

<table>
<thead>
<tr>
<th>Predictor</th>
<th>Method</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X_1$</td>
<td></td>
<td>1.11</td>
<td>0.92</td>
<td>3.29</td>
<td>4.32</td>
<td>1.53</td>
<td>1.71</td>
<td>2.06</td>
</tr>
<tr>
<td>$X_2$</td>
<td></td>
<td>8.29</td>
<td>3.53</td>
<td>20.66</td>
<td>18.84</td>
<td>7.87</td>
<td>8.19</td>
<td>9.38</td>
</tr>
<tr>
<td>$X_3$</td>
<td></td>
<td>1.45</td>
<td>0.84</td>
<td>2.74</td>
<td>3.80</td>
<td>1.28</td>
<td>1.80</td>
<td>1.86</td>
</tr>
<tr>
<td>$X_4$</td>
<td></td>
<td>4.46</td>
<td>2.25</td>
<td>11.16</td>
<td>9.79</td>
<td>5.32</td>
<td>5.27</td>
<td>6.22</td>
</tr>
<tr>
<td>$X_5$</td>
<td></td>
<td>8.31</td>
<td>4.29</td>
<td>17.48</td>
<td>16.45</td>
<td>7.91</td>
<td>8.02</td>
<td>10.32</td>
</tr>
<tr>
<td>$X_6$</td>
<td></td>
<td>1.40</td>
<td>0.95</td>
<td>2.84</td>
<td>3.39</td>
<td>1.35</td>
<td>1.41</td>
<td>1.75</td>
</tr>
<tr>
<td>$X_7$</td>
<td></td>
<td>1.97</td>
<td>0.82</td>
<td>3.77</td>
<td>4.43</td>
<td>2.38</td>
<td>2.13</td>
<td>2.94</td>
</tr>
<tr>
<td>$X_8$</td>
<td></td>
<td>7.50</td>
<td>3.42</td>
<td>15.44</td>
<td>17.20</td>
<td>7.90</td>
<td>8.83</td>
<td>8.69</td>
</tr>
<tr>
<td>$X_9$</td>
<td></td>
<td>11.40</td>
<td>5.79</td>
<td>24.71</td>
<td>25.65</td>
<td>12.48</td>
<td>12.41</td>
<td>12.89</td>
</tr>
<tr>
<td>$X_{10}$</td>
<td></td>
<td>2.95</td>
<td>1.30</td>
<td>28.65</td>
<td>10.37</td>
<td>4.74</td>
<td>6.75</td>
<td>4.55</td>
</tr>
<tr>
<td>$X_{11}$</td>
<td></td>
<td>7.11</td>
<td>3.56</td>
<td>14.10</td>
<td>16.30</td>
<td>6.69</td>
<td>8.04</td>
<td>9.78</td>
</tr>
<tr>
<td>$X_{12}$</td>
<td></td>
<td>1.73</td>
<td>1.16</td>
<td>3.55</td>
<td>4.43</td>
<td>1.70</td>
<td>2.06</td>
<td>3.05</td>
</tr>
<tr>
<td>$X_{13}$</td>
<td></td>
<td>1.93</td>
<td>0.88</td>
<td>5.03</td>
<td>6.33</td>
<td>2.38</td>
<td>2.83</td>
<td>3.19</td>
</tr>
<tr>
<td>$X_{14}$</td>
<td></td>
<td>13.87</td>
<td>7.30</td>
<td>27.82</td>
<td>31.59</td>
<td>14.65</td>
<td>15.30</td>
<td>15.90</td>
</tr>
<tr>
<td>$X_{15}$</td>
<td></td>
<td>2.54</td>
<td>0.97</td>
<td>4.10</td>
<td>5.30</td>
<td>2.62</td>
<td>3.19</td>
<td>3.74</td>
</tr>
</tbody>
</table>

| Time (seconds) |         | 2 113.9 | 2 111.8 | 1 051.6 | 1 051.6 | 701.6 | 804.9 | 931.4 |
| Acceptance Rate |        | 1.00 | 0.30 | 0.67 | 0.46 | 0.93 | 0.87 | 0.76 |
square $[0, 1]^2$. The mean function $f(z) = f(z_1, z_2)$ is chosen here as

$$f(z) = 1 + N(\mu_1, \Sigma_1, z) + N(\mu_2, \Sigma_2, z)$$

where $N(\mu, \Sigma, z)$ denotes a bivariate normal density with mean $\mu$ and covariance matrix $\Sigma$ and we have $\mu_1 = (0.25, 0.75)^T$, $\mu_2 = (0.75, 0.25)^T$, $\Sigma_{1,11} = \Sigma_{1,22} = 0.05$, $\Sigma_{1,12} = 0.01$, $\Sigma_{2,11} = \Sigma_{2,22} = 0.1$ and $\Sigma_{2,12} = 0.01$.

For this simulated data set, our objective is to estimate the mean function $f(z)$ in the situation where we know nothing about it except that it is smooth. The approach taken to this problem in Kohn et al. (2001) is to consider a collection of basis functions

$$(7)$$

$$\{1, z_1, z_2, \|z - \rho_1\|^2 \log(\|z - \rho_1\|), \ldots, \|z - \rho_s\| \log(\|z - \rho_s\|)\}$$

where $\rho_i$, $i = 1, \ldots, s$ are a collection of so-called knot points. Provided these knot points are carefully chosen we can approximate any smooth function as a linear combination of the basis functions. A different set of basis functions could be used: see Kohn et al. (2001) for a discussion of some of the possibilities. Writing $h_i(z)$, $i = 1, \ldots, s + 3$ for the basis terms in (7) we can approximate the mean function $f(z)$ as

$$f(z) \approx \sum_{j=1}^{s+3} \beta_j h_j(z).$$

Thus to estimate $f(z)$ we can consider the linear model

$$y_i = \sum_{j=1}^{s+3} \beta_j h_j(z_i) + \epsilon_i$$

with transformed predictors $h_j(z_i)$, $j = 1, \ldots, s + 3$ and estimate $f(z)$ by estimating the coefficients $\beta_j$. Since the number of knots $s$ needs to be chosen large in (7) to get a set of basis functions capable of approximating any suitable smooth function fairly well, there is a large number of coefficients and a danger of overfitting. To avoid this, Smith and Kohn (1996) suggest the use of Bayesian variable selection methods and Bayesian model averaging in estimating the $\beta_j$.

We have not yet specified our method for choosing the knot points. We describe a method for doing this introduced by Cripps et al. (2003). We
partition the unit square into cells of side length $\delta = 1/r$. For each of these cells, if one of the observed predictors lies in the cell then the mid-point of the cell is taken as one of the knot points. This is a much simpler method for choosing knots than alternative methods based on applying clustering algorithms to the observed knots. We have chosen $r = 10$ in our example so that there are 100 potential knot points.

We compared the same sampling schemes in the same way as for the previous examples, except that in this case, following Kohn et al. (2001), instead of reporting estimated inefficiency factors for estimated marginal probabilities of inclusion we report estimated inefficiency factors for fitted mean values at 10 points chosen uniformly at random from an equally spaced $21 \times 21$ grid on the unit square. The fitted mean at a point $z^*$ is estimated by model averaging as

$$\hat{f}(z^*) = \frac{1}{s} \sum_{i=b+1}^{b+s} x^*_{\gamma(i)}^T (X^T\gamma(i)X_{\gamma(i)})^{-1}(X^T\gamma(i)y)$$

where $b$ is the number of “burn in” iterates, $s$ is the number of sampling iterates, $\gamma(i)$ is the value of $\gamma$ at iteration $i$ and $x^*_{\gamma(i)}$ is the vector obtained by extracting from the vector $x^* = (h_1(z^*), ..., h_{s+3}(z^*))^T$ those components $j$ where $\gamma_j(i) = 1$. Results of the comparison between sampling schemes are shown in Table 3. Again the adaptive schemes approximate fairly well the performance of the Gibbs sampler, but are more than 5 times faster. The use of blocking seems helpful in this example. The Kohn-Smith-Chan schemes also work well here, where the high posterior probability models contain relatively few terms.

4.4 Example: US crime data set

As an example involving real data we consider the US crime rates data set discussed by Ehrlich (1973). See also Raftery, Madigan and Hoeting (1997). The response here is the rate of crimes in a particular category per head of population, which is measured for 47 states of the U.S. The predictors include measures related to the sentencing regime in each state. The full set of predictors used is shown in Table 4. There is very strong correlation between the predictors Po1 and Po2, police expenditure in 1960 and 1959 respectively, and U1 and U2, unemployment rate of urban males aged 13-24 and aged 35-39 respectively. Results for the different sampling schemes in
Table 3: Estimated inefficiency factors for fits at 10 randomly chosen points from an equally spaced $21 \times 21$ grid on the unit square for nonparametric regression example. Estimates are based on 50,000 iterations with 5,000 iterations burn in. In the table method A is the Gibbs sampler, method B the Metropolized Gibbs sampler, methods C and D the Kohn-Smith-Chan schemes with blocks of size 1 and 2 respectively, and methods E, F and G are the adaptive schemes with blocks of size 1, 2 and 3 respectively.

<table>
<thead>
<tr>
<th>Method</th>
<th>Point</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>47.73</td>
<td>26.93</td>
<td>50.34</td>
<td>54.25</td>
<td>60.35</td>
<td>44.71</td>
<td>42.84</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>128.93</td>
<td>85.70</td>
<td>195.63</td>
<td>168.78</td>
<td>169.03</td>
<td>142.81</td>
<td>126.95</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>61.21</td>
<td>63.18</td>
<td>75.86</td>
<td>85.42</td>
<td>97.05</td>
<td>75.55</td>
<td>79.94</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>80.20</td>
<td>70.52</td>
<td>107.33</td>
<td>82.36</td>
<td>130.79</td>
<td>108.90</td>
<td>97.54</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>39.95</td>
<td>44.60</td>
<td>90.67</td>
<td>128.98</td>
<td>101.37</td>
<td>69.66</td>
<td>90.48</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>87.80</td>
<td>53.51</td>
<td>78.96</td>
<td>108.23</td>
<td>145.08</td>
<td>85.65</td>
<td>85.31</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>128.66</td>
<td>81.74</td>
<td>155.85</td>
<td>163.11</td>
<td>174.45</td>
<td>166.22</td>
<td>125.09</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>78.26</td>
<td>64.09</td>
<td>132.49</td>
<td>111.70</td>
<td>115.59</td>
<td>110.06</td>
<td>76.11</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>56.19</td>
<td>51.82</td>
<td>90.61</td>
<td>108.11</td>
<td>84.23</td>
<td>78.40</td>
<td>60.39</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>66.58</td>
<td>58.52</td>
<td>114.14</td>
<td>106.41</td>
<td>105.54</td>
<td>47.04</td>
<td>89.09</td>
</tr>
<tr>
<td>Time</td>
<td></td>
<td>5 904.2</td>
<td>5 874.2</td>
<td>640.9</td>
<td>772.7</td>
<td>1 023.7</td>
<td>1 120.8</td>
<td>1 316.9</td>
</tr>
<tr>
<td>Acceptance Rate</td>
<td>1.00</td>
<td>0.02</td>
<td>0.67</td>
<td>0.78</td>
<td>0.94</td>
<td>0.93</td>
<td>0.93</td>
<td></td>
</tr>
</tbody>
</table>
this problem are reported in the same way as for the first two examples, and are shown in Table 5. Estimated inefficiency factors for the adaptive schemes are slightly higher than for the Gibbs sampler and Metropolized Gibbs sampler, but these schemes are about twice as fast per iteration. The Kohn-Smith-Chan schemes do not perform well in this example.

Table 4: Predictors for US crime data set

<table>
<thead>
<tr>
<th>Predictor</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>percentage of males aged 14-24</td>
</tr>
<tr>
<td>So</td>
<td>indicator variable for a southern state</td>
</tr>
<tr>
<td>Ed</td>
<td>mean years of schooling</td>
</tr>
<tr>
<td>Po1</td>
<td>police expenditure in 1960</td>
</tr>
<tr>
<td>Po2</td>
<td>police expenditure in 1959</td>
</tr>
<tr>
<td>LF</td>
<td>labour force participation rate</td>
</tr>
<tr>
<td>M.F</td>
<td>number of males per 1000 females</td>
</tr>
<tr>
<td>Pop</td>
<td>state population</td>
</tr>
<tr>
<td>NW</td>
<td>number of nonwhites per 1000 people</td>
</tr>
<tr>
<td>U1</td>
<td>unemployment rate of urban males 14-24</td>
</tr>
<tr>
<td>U2</td>
<td>unemployment rate of urban males 35-39</td>
</tr>
<tr>
<td>GDP</td>
<td>gross domestic product per head</td>
</tr>
<tr>
<td>Ineq</td>
<td>income inequality</td>
</tr>
<tr>
<td>Prob</td>
<td>probability of imprisonment</td>
</tr>
<tr>
<td>Time</td>
<td>average time served in state prisons</td>
</tr>
</tbody>
</table>

5. DISCUSSION AND CONCLUSIONS

In this paper we have considered adaptive modifications of standard Markov chain Monte Carlo sampling schemes for Bayesian variable selection in Gaussian linear regression models. The key idea of our sampling scheme is to use best linear prediction for approximating the Gibbs sampler based on a running mean and covariance matrix computed from previously sampled iterations. We have also outlined some general results concerning adaptation in Markov chain Monte Carlo with elementary proofs in the case of distributions on a finite state space. However, as well as having a theory showing that adaptive sampling is possible it is important to have practical
Table 5: Estimated inefficiency factors for $\hat{\gamma}_j$ for US crimes example. Estimates are based on 50,000 iterations with 5,000 iterations burn in. In the table method A is the Gibbs sampler, method B the Metropolized Gibbs sampler, methods C and D the Kohn-Smith-Chan schemes with blocks of size 1 and 2 respectively, and methods E, F and G are the adaptive schemes with blocks of size 1, 2 and 3 respectively.

<table>
<thead>
<tr>
<th>Predictor</th>
<th>Method</th>
<th>(\hat{\gamma}_j)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>A</td>
<td>2.75</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>0.92</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>4.30</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>5.91</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>3.19</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>3.31</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>3.14</td>
</tr>
<tr>
<td>So</td>
<td>A</td>
<td>0.92</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>0.65</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>2.89</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>2.78</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>1.83</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>1.21</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>1.45</td>
</tr>
<tr>
<td>Ed</td>
<td>A</td>
<td>2.93</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>1.46</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>6.58</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>7.39</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>3.76</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>3.35</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>3.79</td>
</tr>
<tr>
<td>Po1</td>
<td>A</td>
<td>4.05</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>2.13</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>5.61</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>8.29</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>4.05</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>4.62</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>5.21</td>
</tr>
<tr>
<td>Po2</td>
<td>A</td>
<td>3.86</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>2.12</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>5.69</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>8.76</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>3.71</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>3.80</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>4.00</td>
</tr>
<tr>
<td>LF</td>
<td>A</td>
<td>0.90</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>0.97</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>2.62</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>3.40</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>1.80</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>1.97</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>1.33</td>
</tr>
<tr>
<td>M.F</td>
<td>A</td>
<td>1.38</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>0.96</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>2.96</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>3.81</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>2.01</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>2.16</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>2.12</td>
</tr>
<tr>
<td>Pop</td>
<td>A</td>
<td>0.99</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>0.77</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>2.06</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>2.58</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>1.25</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>1.27</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>1.92</td>
</tr>
<tr>
<td>NW</td>
<td>A</td>
<td>1.22</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>0.86</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>2.58</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>2.52</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>1.23</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>1.64</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>1.74</td>
</tr>
<tr>
<td>U1</td>
<td>A</td>
<td>1.54</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>0.76</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>2.87</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>3.15</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>2.20</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>1.78</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>2.13</td>
</tr>
<tr>
<td>U2</td>
<td>A</td>
<td>2.46</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>1.02</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>3.63</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>5.54</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>3.31</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>2.88</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>3.76</td>
</tr>
<tr>
<td>GDP</td>
<td>A</td>
<td>1.21</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>0.57</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>2.27</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>3.15</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>1.55</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>1.69</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>1.30</td>
</tr>
<tr>
<td>Ineq</td>
<td>A</td>
<td>2.43</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>1.61</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>9.45</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>10.27</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>4.23</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>3.95</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>3.57</td>
</tr>
<tr>
<td>Prob</td>
<td>A</td>
<td>1.70</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>0.70</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>3.44</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>4.28</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>4.28</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>2.01</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>2.66</td>
</tr>
<tr>
<td>Time</td>
<td>A</td>
<td>1.18</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>0.57</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>2.36</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>3.05</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>1.42</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>1.68</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>1.61</td>
</tr>
<tr>
<td>Time (seconds)</td>
<td>A</td>
<td>542.6</td>
</tr>
<tr>
<td></td>
<td>B</td>
<td>541.5</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>260.4</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>749.8</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>204.3</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>249.8</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>311.0</td>
</tr>
</tbody>
</table>

| Acceptance Rate | A | 1.00 |
|                | B | 0.32 |
|                | C | 0.78 |
|                | D | 0.51 |
|                | E | 0.93 |
|                | F | 0.88 |
|                | G | 0.78 |
suggestions for how adaptation should be implemented in practice and in particular how to construct proposals. Our best linear predictor proposals based on a running mean and covariance matrix provide one fairly general idea about how to construct adaptive proposals in general problems. Adapting the ideas described in this paper for doing variable selection in regression with non-Gaussian responses and where regression coefficients can’t be integrated out analytically seems a fruitful topic for further research. Here there may be no very easily implemented Markov chain Monte Carlo scheme which can serve as a gold standard for comparison, and it may be hard to come up with good Metropolis-Hastings proposals without some preliminary experimentation. It is in these kind of problems where the ideas of adaptive sampling are potentially even more valuable than for the problems considered in the current paper. This is a topic of continuing research.

APPENDIX A

Proof of Theorem 1. Write $Z_{\leq k} = (Z_1, \ldots, Z_k)$, $Z_{j \leq k} = (Z_j, \ldots, Z_k)^T$ for $j \leq k$ and define $\Delta_{\leq k}$ and $\Delta_{j \leq k}$ similarly. First observe that we can embed the process $Z = \{Z_n; n \geq 0\}$ into a process $Z^+ = \{(Z_n, \Delta_n); n \geq 0\}$ where the $\Delta_n$ are iid Bernoulli($\epsilon$) and

\[
\text{pr}(Z_{n+1} = z_{n+1}|Z_{\leq n} = z_{\leq n}, \Delta_{\leq n} = \delta_{\leq n}) = \begin{cases} 
\pi(z_{n+1}) & \text{if } \Delta_n = 1 \\
R_n(z_n, z_{n+1}) & \text{if } \Delta_n = 0
\end{cases}
\]

where

\[
R_n(z_n, z_{n+1}) = \frac{T_n(z_n, z_{n+1}) - \epsilon \pi(z_{n+1})}{1 - \epsilon}
\]

which is a proper transition kernel since $T_n(z_n, z_{n+1}) \geq \epsilon \pi(z_{n+1})$ for every $n$ and $z_0, \ldots, z_n$. This is the split chain construction of Athreya and Ney (1978). Clearly

\[
\text{pr}(Z_{n+1} = z_{n+1}|Z_{\leq n} = z_{\leq n}) = \epsilon \pi(z_{n+1}) + (1 - \epsilon)R_n(z_n, z_{n+1})
\]

Equation (8) shows that we can think of simulating from $T_n(z_n, z_{n+1})$ as simulating from a mixture kernel. We flip a coin that lands heads with probability $\epsilon$ and for heads we draw $Z_{n+1}$ from $\pi(z_{n+1})$, otherwise draw $Z_{n+1}$ from $R_n(z_n, z_{n+1})$. Note that $\sum_z \pi(z_n)R_n(z_n, z_{n+1}) = \pi(z_{n+1})$ which holds because of the balance condition (4) for $T_n(z_n, z_{n+1})$. Now we show $p(z_n) =$
\[ \text{pr}(Z_n = z_n) \rightarrow \pi(z_n) \text{ as } n \rightarrow \infty. \] The idea of the proof is to write \( p(z_n) \) in terms of probabilities conditional on the last value of 1 for \( \Delta \) being \( j \) steps in the past. If \( \Delta_{n-j} = 1 \), then at the next step we draw the state for \( Z \) from \( \pi(\cdot) \). Then the \( Z \) process evolves according to the kernels \( R_k(z_k, z_{k+1}) \) which by the stability condition for large \( k \) behave like a fixed kernel with invariant distribution \( \pi(\cdot) \). Precisely, let

\[ A_{n,j} = \{\Delta_{n-1} = 0, \ldots, \Delta_{n-j} = 1\} \]

\( j = 1, \ldots, n \) and let \( B_n = \{\Delta_{n-1} = 0, \ldots, \Delta_0 = 0\} \). Then

\[ p(z_n) = \sum_{j=1}^{n} \text{pr}(Z_n = z_n|A_{n,j})\text{pr}(A_{n,j}) + \text{pr}(B_n)\text{pr}(Z_n = z_n|B_n) \]

\[ = \sum_{j=1}^{n} \epsilon(1 - \epsilon)^{j-1}\text{pr}(Z_n = z_n|A_{n,j}) + (1 - \epsilon)^{n}\text{pr}(Z_n = z_n|B_n). \] (9)

Now,

\[ \text{pr}(Z_n = z_n|A_{n,j}) = \sum_{z \leq n-1} \text{pr}(Z_{\leq n} = z_{\leq n}|A_{n,j}) \]

\[ = \sum_{z \leq n-1} \text{pr}(Z_{\leq n-j} = z_{\leq n-j}|A_{n,j}) \]

\[ \text{pr}(Z_{n-j+1 \leq n} = z_{n-j+1 \leq n}|A_{n,j}, Z_{\leq n-j} = z_{\leq n-j}) \]

\[ = \sum_{z \leq n-1} \text{pr}(Z_{\leq n-j} = z_{\leq n-j}) \]

\[ \pi(z_{n-j+1}) \prod_{k=1}^{j-1} R_{n-k}(z_{n-k}, z_{n-k+1}) \] (10)

Now note that for any sequences \( \{a_k\}_{k=1}^{m} \) and \( \{b_k\}_{k=1}^{m} \) we can write

\[ \prod_{k=1}^{m} a_k = b_1 \ldots b_m + (a_1 b_2 \ldots b_m - b_1 \ldots b_m) + \cdots + (a_1 \ldots a_m - a_1 \ldots a_{m-1} b_m) \]

\[ = \prod_{k=1}^{m} b_k + \sum_{l=1}^{m} \left( \prod_{k=1}^{l-1} a_k \right) \left( a_l - b_l \right) \left( \prod_{k=l+1}^{m} b_k \right). \]
Applying this result to the sequences \( \{R_{n-k}(z_{n-k}, z_{n-k+1})\}_{k=1}^{j-1} \) and \( \{R_{n-j}(z_{n-k}, z_{n-k+1})\}_{k=1}^{j-1} \) and substituting into (10) gives

\[
\begin{align*}
\text{pr}(Z_n = z_n | A_{n,j}) &= \sum_{z_{n-j+1} \leq n} \text{pr}(Z_{\leq n-j} = z_{\leq n-j}) \pi(z_{n-j+1}) \prod_{k=1}^{j-1} R_{n-j}(z_{n-k}, z_{n-k+1}) \\
&\quad + \sum_{z_{n-j} \leq n} \text{pr}(Z_{\leq n-j} = z_{\leq n-j}) \pi(z_{n-j+1}) \\
&\quad \sum_{l=1}^{j-1} \left( \prod_{k=1}^{l-1} R_{n-k}(z_{n-k}, z_{n-k+1}) \right) (R_{n-l}(z_{n-l}, z_{n-l+1}) - R_{n-j}(z_{n-l}, z_{n-l+1})) \\
&\quad \sum_{l=1}^{j-1} R_{n-j}(z_{n-l}, z_{n-l+1}) \left( \prod_{k=l+1}^{j-1} R_{n-j}(z_{n-k}, z_{n-k+1}) \right).
\end{align*}
\]

Note that

\[
\sum_{z_{n-j+1} \leq n} \pi(z_{n-j+1}) \prod_{k=1}^{j-1} R_{n-j}(z_{n-k}, z_{n-k+1}) = \pi(z_n)
\]

by the balance condition for the kernel \( R_{n-j}(\cdot, \cdot) \). Also, write

\[
Q_{j,l}(z_{\leq n}) = \text{pr}(Z_{\leq n-j} = z_{\leq n-j}) \pi(z_{n-j+1}) \left( \prod_{k=1}^{l-1} R_{n-k}(z_{n-k}, z_{n-k+1}) \right) n_Z^{-1} \left( \prod_{k=l+1}^{j-1} R_{n-j}(z_{n-k}, z_{n-k+1}) \right)
\]

where \( n_Z \) is the number of elements in the finite state space for \( Z \). Note that \( Q_{j,l}(z_{\leq n}) \) is a probability function. We have

\[
\text{pr}(Z_n = z_n | A_{n,j}) = \pi(z_n) + \sum_{z_{n-j} \leq n} \sum_{l=1}^{j-1} n_Z Q_{j,l}(z_{\leq n})(R_{n-l}(z_{n-l}, z_{n-l+1}) - R_{n-j}(z_{n-l}, z_{n-l+1}))
\]
and hence substituting into equation (9)

\[ |p(z_n) - \pi(z_n)| = \frac{1}{n+1} \sum_{i=0}^{n} h(Z_i) \to E_\pi(h(Z)) \]

in mean square. Now,

\[ E \left( \left( \frac{1}{n+1} \sum_{i=0}^{n} h(Z_i) - E_\pi(h(Z)) \right)^2 \right) = T_1 + T_2 \]
where
\[ T_1 = \frac{1}{(n+1)^2} \sum_{i,j=0}^{n} (E(h(Z_i)) - E_\pi(h(Z)))(E(h(Z_j)) - E_\pi(h(Z))) \]

and
\[ T_2 = \frac{1}{(n+1)^2} \sum_{i,j=0}^{n} \text{cov}(h(Z_i), h(Z_j)). \]

We have shown (proof of Theorem 1) that
\[ |p(z_i) - \pi(z_i)| \leq K_1(1 - \epsilon)^{i-1} + K_2 i^{-r_1} \quad (11) \]
for constants $K_1$ and $K_2$. Hence
\[ |E(h(Z_i)) - E_\pi(h(Z))| \leq \sup_{z,z'} |h(z) - h(z')| (K_1(1 - \epsilon)^{i-1} + K_2 i^{-r_1}) \]
\[ \leq C_1(1 - \epsilon)^{i-1} + C_2 i^{-r_1}. \]

Hence
\[ \lim_{n \to \infty} |T_1| \leq \lim_{n \to \infty} \frac{1}{(n+1)^2} \sum_{i=0}^{n} \sum_{j=0}^{n} |(E(h(Z_i)) - E_\pi(h(Z)))(E(h(Z_j)) - E_\pi(h(Z)))| \]
\[ = \lim_{n \to \infty} \frac{1}{(n+1)^2} \sum_{i=0}^{n} \sum_{j=0}^{n} |(E(h(Z_i)) - E_\pi(h(Z)))(E(h(Z_j)) - E_\pi(h(Z)))| \]
\[ \leq \frac{1}{(n+1)^2} \sum_{i=0}^{n} \sum_{j=0}^{n} (C_1(1 - \epsilon)^{i-1} + C_2 i^{-r_1})(C_1(1 - \epsilon)^{j-1} + C_2 j^{-r_1}) \]

which approaches 0 as $n \to \infty$. The equality in the second line above follows from the boundedness of $h(\cdot)$ since the state space is finite. It remains to show $T_2 \to 0$ as $n \to \infty$. Without loss of generality suppose $i < j$ and write
\[ p(z_i, z_j) = \text{pr}(Z_i = z_i, Z_j = z_j) \]
\[ = \text{pr}(Z_i = z_i)\text{pr}(Z_j = z_j | Z_i = z_i) \]
\[ = p(z_i) p(z_j | z_i) \]
Now,

\[ p(z_j|z_i) = \sum_{z_{\leq i-1}} p(z_j, z_{\leq i-1}|z_i) \]
\[ = \sum_{z_{\leq i-1}} p(z_{\leq i-1}|z_i)p(z_j|z_{\leq i}). \]

Hence

\[ |\pi(z_j) - p(z_j|z_i)| \leq |\pi(z_j) - \sum_{z_{\leq i-1}} p(z_{\leq i-1}|z_i)p(z_j|z_{\leq i})| \]
\[ \leq \sum_{z_{\leq i-1}} p(z_{\leq i-1}|z_i)|\pi(z_j) - p(z_j|z_{\leq i})| \]

Applying the reasoning of theorem 1 to the process \( \{Z_{n+i}; n \geq 0\} \) we have

\[ |\pi(z_j) - p(z_j|z_{\leq i})| \leq K_1(1 - \epsilon)^{j-i-1} + K_2(j-i)^{-r_1} \]

where the constants \( K_1 \) and \( K_2 \) can be taken to be the same constants as in (11). Hence

\[ |\pi(z_j) - p(z_j|z_i)| \leq K_1(1 - \epsilon)^{j-i-1} + K_2(j-i)^{-r_1} \]

which gives

\[ |p(z_i, z_j) - \pi(z_i)\pi(z_j)| = |p(z_i)p(z_j|z_i) - p(z_i)\pi(z_j) + p(z_i)\pi(z_j) - \pi(z_i)\pi(z_j)| \]
\[ \leq K_1(1 - \epsilon)^{j-i-1} + K_2(j-i)^{-r_1} + K_1(1 - \epsilon)^{i-1} + K_2i^{-r_1} \]

and

\[ \text{cov}(h(Z_i), h(Z_j)) = \sum_{z_i, z_j}(h(z_i) - E(h(Z_i)))(h(z_j) - E(h(Z_j)))\pi(z_i)\pi(z_j) \]
\[ + \sum_{z_i, z_j}(h(z_i) - E(h(Z_i)))(h(z_j) - E(h(Z_j)))(p(z_i, z_j) - \pi(z_i)\pi(z_j)) \]

so that

\[ |\text{cov}(h(Z_i), h(Z_j))| \leq |(E_\pi(h(Z_i)) - E(h(Z_i)))(E_\pi(h(Z_j)) - E(h(Z_j)))| \]
\[ + C_h(K_1(1 - \epsilon)^{j-i-1} + K_2(j-i)^{-\lambda_1} + K_1(1 - \epsilon)^{i-1} + K_2i^{-\lambda_1}) \]
where $C_h$ is a constant depending only on $h$. Hence

$$|\text{cov}(h(Z_i), h(Z_j))| \leq (C_1(1 - \epsilon)i^{-1} + C_2i^{-r_1})(C_1(1 - \epsilon)j^{-1} + C_2j^{-r_1}) + C_h(K_1(1 - \epsilon)i^{-1} + K_2(j - i)^{-r_1} + K_1(1 - \epsilon)j^{-1} + K_2j^{-r_1}).$$

Now

$$\lim_{n \to \infty} \frac{1}{(n + 1)^2} \sum_{i=0}^{n} \sum_{j=0}^{n} \text{cov}(h(Z_i), h(Z_j)) = \lim_{n \to \infty} \frac{1}{(n + 1)^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \text{cov}(h(Z_i), h(Z_j))$$

and using the above bound for $|\text{cov}(h(Z_i), h(Z_j))|$ this approaches 0 as $n \to \infty$. This proves mean square convergence. To show almost sure convergence, write $S_n = \sum_{i=1}^{n} h(Z_i)$ and let $k$ be an integer such that $kr_1 > 1$. We want to show $n^{-1}S_n \to E_\pi(h(Z))$ almost surely. By Chebyshev’s inequality

$$Pr(|S_{ik} - E(S_{ik})| \geq \epsilon i^k) \leq \frac{\text{var}(S_{ik})}{\epsilon^2 i^{2k}}$$

from which we have since $\text{var}(S_{ik}) = O(i^{2k-kr_1})$ from the proof of mean square convergence

$$\sum_{i} Pr(|S_{ik} - E(S_{ik})| \geq \epsilon i^k) < \infty$$

which shows $i^{-k}S_{ik} \to E_\pi(h(Z))$ almost surely since $i^{-k}E(S_{ik}) \to E_\pi(h(Z))$. Then from almost sure convergence of this subsequence using similar arguments to those in the proof of the strong law of large numbers (Grimmett and Stirzaker, 1992, p. 295) and exploiting the boundedness of $h(\cdot)$, we can obtain almost sure convergence.

**APPENDIX B**

Suppose that in step 3 b) i) of our adaptive algorithm instead of updating a single variable $\gamma_{ji}$ we wish to update a block of size 2, which in what follows we write as $(\gamma_u, \gamma_v)$. We can construct an approximation to the joint posterior conditional distribution of $(\gamma_u, \gamma_v)$ given the other components of $\gamma$ (which we write as $p(\gamma_u, \gamma_v|\gamma \neq \{u,v\}, y)$) in terms of the conditional mean and covariance.
matrix of \((\gamma_u, \gamma_v)\). To see how to do this, observe that if \(\eta = (\eta_1, \eta_2)\) is any binary random vector with \(E(\eta_i) = \mu_i\ i = 1, 2\) and \(\text{cov}(\eta_1, \eta_2) = \sigma_{12}\), then

\[
\begin{align*}
\text{pr}(\eta_1 = 1, \eta_2 = 1) &= \sigma_{12} + \mu_1\mu_2 \\
\text{pr}(\eta_1 = 1, \eta_2 = 0) &= \mu_1 - \sigma_{12} - \mu_1\mu_2 \\
\text{pr}(\eta_1 = 0, \eta_2 = 1) &= \mu_2 - \sigma_{12} - \mu_1\mu_2 \\
\text{pr}(\eta_1 = 0, \eta_2 = 0) &= 1 + \sigma_{12} + \mu_1\mu_2 - \mu_1 - \mu_2.
\end{align*}
\]

Representing \(p(\gamma_u, \gamma_v|\gamma \neq \{u,v\}, y)\) in this way in terms of conditional means and covariances of \(\gamma_u, \gamma_v\) given \(\gamma \neq \{u,v\}, y\), we generalize our previous sampling scheme by approximating the means and covariances in this representation of the block conditional distribution using best linear prediction to obtain a proposal distribution \(q(\gamma_u, \gamma_v|\gamma \neq \{u,v\}, y)\).

More precisely, we can assume without loss of generality that \(u = 1, v = 2\) by relabelling predictors if necessary. We write the matrix \(W(i)\) in step 3 of our algorithm in partitioned form as

\[
W(i) = (\hat{\Sigma} + \lambda I)^{-1} = \begin{bmatrix} S_{11} & S_{12} \\ S_{12}^T & S_{22} \end{bmatrix}
\]

where \(S_{11}\) is a \(2 \times 2\) matrix, \(S_{22}\) is \((k - 2) \times (k - 2)\) and \(S_{12}\) is \(2 \times (k - 2)\). If \(\lambda = 0\), then taking the best linear predictor of \((\gamma_1, \gamma_2)\) given \(\gamma \neq \{1,2\}, y\) and replacing means and covariances with sample versions gives

\[
(\hat{\gamma}(i)_1, \hat{\gamma}(i)_2)^T - S_{11}^{-1}S_{12}(\gamma_{\neq \{1,2\}}^C - \hat{\gamma}(i)_{\neq \{1,2\}})
\]

with the covariance matrix of the prediction error being \(S_{11}^{-1}\). For a Gaussian random vector, the best linear predictor is the conditional mean, and the associated prediction error covariance matrix is the conditional covariance. This motivates using (13) as an approximation of the conditional mean and \(S_{11}^{-1}\) as an approximation to the conditional covariance and then substituting into the representation (12) gives an approximation to the block conditional distribution, \(\hat{p}(\gamma_u, \gamma_v|\gamma \neq \{u,v\}, y)\) say. We construct a proposal distribution as

\[
q(\gamma_u, \gamma_v|\gamma \neq \{u,v\}, y) \propto \hat{p}(\gamma_u, \gamma_v|\gamma \neq \{u,v\}, y) + C
\]

where \(C\) is chosen as

\[
C = \max \left\{ 0, \frac{\max \hat{p}(\gamma_u, \gamma_v|\gamma \neq \{u,v\}, y) - (1 - \delta)}{4(1 - \delta) - 1}, \frac{\delta - \min \hat{p}(\gamma_u, \gamma_v|\gamma \neq \{u,v\}, y)}{1 - 4\delta} \right\}
\]
which ensures that the values for \( q(\cdot) \) are bounded between \( \delta \) and \( 1 - \delta \). After generating a proposal value for \( \gamma_u, \gamma_v \) from this proposal distribution the proposal is accepted with the usual Metropolis-Hastings acceptance probability of \( \min\{1, \alpha\} \) where

\[
\alpha = \frac{p(\gamma^P)p(y|\gamma^P)q(\gamma_u^C, \gamma_v^C|\gamma_{P \cup \{u,v\}}, y)}{p(\gamma^C)p(y|\gamma^C)q(\gamma_u^P, \gamma_v^P|\gamma_{C \cup \{u,v\}}, y)}
\]

and \( \gamma^C \) and \( \gamma^P \) denote the current and proposed values.

In modifying our adaptive algorithm we have discussed how to update in step 3 b) i) using blocks of size 2, but we have not said how we choose the blocks. The way we have done this in the examples is as follows. First, based on the first \( d \) iterations, we estimate which predictors have marginal probability of inclusion between \( \delta \) and \( 1 - \delta \). For the examples below, \( \delta = 0.01 \). Then at each iteration we randomly split these predictors into blocks of size 2. If there is an odd number of predictors, there will be one remaining predictor updated as a block of size one. All remaining predictors are updated as blocks of size one. This specifies the procedure for generating the blocks. Then in updating \( \gamma \), we use these blocks, visiting them in random order.

We have only discussed so far updating using blocks of size 2 and size 1. We can use similar ideas to the above to update using blocks of size 3 or more. We outline briefly how this is done for blocks of size 3. The idea is to generalize the representation (12) to express the joint distribution for a binary random vector \( \eta = (\eta_1, \eta_2, \eta_3) \) in terms of means and covariances. In general, means and covariances are not enough to determine the joint distribution of \( \eta \) when \( \eta \) has length 3 or more, but if we assume all moments of higher than second order are zero then we can determine a distribution from means and covariances. In particular, writing \( \mu_i = E(\eta_i) \) and \( \sigma_{ij} = \text{cov}(\eta_i, \eta_j) \),
1 ≤ i, j ≤ 3 and assuming \( E((\eta_1 - \mu_1)(\eta_2 - \mu_2)(\eta_3 - \mu_3)) = 0 \) we have

\[
\begin{align*}
\text{pr}(\eta_1 = 1, \eta_2 = 1, \eta_3 = 1) &= \mu_1 \mu_2 \mu_3 + \mu_1 \sigma_{23} + \mu_2 \sigma_{13} + \mu_3 \sigma_{12} \\
\text{pr}(\eta_1 = 1, \eta_2 = 1, \eta_3 = 0) &= \sigma_{12} + \mu_1 \mu_2 - \mu_1 \mu_2 \mu_3 - \mu_1 \sigma_{23} - \mu_2 \sigma_{13} - \mu_3 \sigma_{12} \\
\text{pr}(\eta_1 = 1, \eta_2 = 0, \eta_3 = 1) &= \sigma_{13} + \mu_1 \mu_3 - \mu_1 \mu_2 \mu_3 - \mu_1 \sigma_{23} - \mu_2 \sigma_{13} - \mu_3 \sigma_{12} \\
\text{pr}(\eta_1 = 0, \eta_2 = 1, \eta_3 = 1) &= \sigma_{23} + \mu_2 \mu_3 - \mu_1 \mu_2 \mu_3 - \mu_1 \sigma_{23} - \mu_2 \sigma_{13} - \mu_3 \sigma_{12} \\
\text{pr}(\eta_1 = 1, \eta_2 = 0, \eta_3 = 0) &= \mu_1 - \sigma_{13} - \sigma_{12} - \mu_1 \mu_3 - \mu_1 \mu_2 + \mu_1 \mu_2 \mu_3 \\
&+ \mu_1 \sigma_{23} + \mu_2 \sigma_{13} + \mu_3 \sigma_{12} \\
\text{pr}(\eta_1 = 0, \eta_2 = 1, \eta_3 = 0) &= \mu_2 - \sigma_{12} - \sigma_{23} - \mu_1 \mu_2 - \mu_2 \mu_3 + \mu_1 \mu_2 \mu_3 \\
&+ \mu_1 \sigma_{23} + \mu_2 \sigma_{13} + \mu_3 \sigma_{12} \\
\text{pr}(\eta_1 = 0, \eta_2 = 0, \eta_3 = 1) &= \mu_3 - \sigma_{13} - \sigma_{23} - \mu_1 \mu_3 - \mu_2 \mu_3 + \mu_1 \mu_2 \mu_3 \\
&+ \mu_1 \sigma_{23} + \mu_2 \sigma_{13} + \mu_3 \sigma_{12} \\
\text{pr}(\eta_1 = 0, \eta_2 = 0, \eta_3 = 0) &= 1 - \mu_1 - \mu_2 - \mu_3 + \mu_1 \mu_2 + \mu_1 \mu_3 + \mu_2 \mu_3 \\
&- \mu_1 \mu_2 \mu_3 + \sigma_{12} + \sigma_{13} + \sigma_{23} \\
&- \mu_1 \sigma_{23} - \mu_2 \sigma_{13} - \mu_3 \sigma_{12}
\end{align*}
\]

Using this representation and best linear prediction to approximate conditional means and covariances in a block conditional distribution for a block of size three gives a proposal distribution similar to before after truncation. Blocks of order higher than three can be updated in a similar way. Choosing how to block variables is done in the same way as for blocks of size 2. Based on the first \( d \) iterations, we get the variables with estimated marginal probability of inclusion between \( \delta \) and \( 1 - \delta \). At each iteration we randomly split these variables into blocks of size three. If the number of variables to block is not a multiple of three, then there are one or two variables left over which are updated as blocks of size 1. All remaining variables are updated as blocks of size 1. The blocks chosen at each iteration are updated in random order.

ACKNOWLEDGEMENT

This work was partially done while the first author was visiting the Institute for Mathematical Sciences, National University of Singapore, 2004. The visit was supported by the Institute. We thank Wilfrid Kendall for useful discussions about adaptive sampling and for providing several instructive examples.
REFERENCES


Chain Monte Carlo in Practice, London: Chapman and Hall.


methods. Technical Report, Department of Computer Science, University of Toronto.


