Bayesian Methods for Hidden Markov Models

Recursive Computing in the 21st Century

Submitted to JASA Reviews

Steven L. Scott[*]

November 10, 2000

Abstract

Markov chain Monte Carlo (MCMC) sampling strategies can be used to simulate hidden Markov model (HMM) parameters from their posterior distribution given observed data. Some MCMC methods (for computing HMM likelihoods, conditional probabilities of hidden states, and the most likely sequence of states) used in practice can be improved by incorporating established recursive algorithms. The most important is a set of forward-backward recursions calculating conditional distributions of the hidden states given observed data and model parameters. We show how to use the recursive algorithms in an MCMC context and demonstrate mathematical and empirical results showing a Gibbs sampler using the forward-backward recursions mixes more rapidly than another sampler often used for HMM's. We introduce an augmented variables technique for obtaining unique state labels in HMM's and finite mixture models. We show how recursive computing allows statistically efficient use of MCMC output when estimating the hidden states. We directly calculate the posterior distribution of the hidden chain's state space size by MCMC, circumventing asymptotic arguments underlying the Bayesian information criterion, which is shown to be inappropriate for a frequently analyzed data set in the HMM literature. The use of log-likelihood for assessing MCMC convergence is illustrated, and posterior predictive checks are used to investigate application specific questions of model adequacy.

Key Words: Forward-Backward, Kalman Filter, MCMC, Gibbs Sampler, Recursion, Local Computation

[*] Assistant Professor of Statistics, Marshall School of Business, University of Southern California. email: sls@usc.edu

A hidden Markov model (HMM) is a mixture model whose mixing distribution is a finite state Markov chain. HMM's have been successfully applied to problems in a variety of fields including signal processing (Juang and Rabiner, 1991; Andrieu and Doucet, 2000), biology (Fredkin and Rice, 1992; Leroux and Puterman, 1992), genetics (Churchill, 1989; Liu *et al.*, 1999), ecology (Guttorp, 1995), image analysis (Romberg *et al.*, 2000), economics (Albert and Chib, 1993; Hamilton, 1989, 1990), and network security (Scott, 1999, 2000). The book by MacDonald and Zucchini (1997) illustrates several applications of HMM's and provides an extended bibliography.

For much of their history, HMM's have been implemented using recursive algorithms developed for parameter estimation (Baum *et al.*, 1970) and for restoring the hidden Markov chain (Viterbi, 1967). Today's Markov chain Monte Carlo (MCMC) techniques allow researchers to implement HMM's without the traditional recursive algorithms, which are viewed as "black boxes" by many statisticians. Contrary to this perception, the recursions have intuitive probabilistic interpretations and can improve many MCMC methods. Through the use of recursive computing, HMM users can take advantage of rapidly mixing MCMC algorithms, use MCMC output to more efficiently estimate the hidden chain, and employ model selection techniques and convergence diagnostics that are otherwise unavailable. The recursive algorithms also appeal to an aesthetic that says relying on simulation to produce directly computable answers is time consuming, inelegant, needlessly inaccurate, and generally not a good idea. This article reviews the most common recursive algorithms for HMM's and explains their role in a modern MCMC environment.

The article is structured around important tasks facing the Bayesian modeler: obtaining the posterior distribution of model parameters, estimating the hidden Markov chain, determining the size of the hidden chain's state space, and using diagnostics to assess model validity and MCMC convergence. Section 1 provides background on HMM's, including two closely linked recursive procedures for evaluating the HMM likelihood (the *likelihood recursion*) and the posterior distribution of each hidden state given observed data and model parameters (the *forward-backward recursions*). Section 2 discusses methods for sampling HMM parameters from their posterior distribution given observed data, with particular emphasis on two Gibbs samplers. The forward-backward Gibbs sampler (FB) capitalizes on recursive techniques from Section 1. The direct Gibbs sampler (DG) samples each state in the hidden Markov chain given the most recent draws of its neighbors. FB requires more computer time per iteration, but it mixes more rapidly than DG and produces useful quantities as byproducts. Section 2 also discusses a label switching issue that HMM's share with finite mixture models. Section 3 describes Bayes and empirical Bayes methods for estimating the hidden Markov chain. Section 4 considers methods for determining the size of the hidden chain's state space. Section 5 uses the likelihood recursion to diagnose MCMC convergence and illustrates the use of posterior predictive checks to determine a model's capacity to produce specific features seen in the data. Section 6 summarizes our conclusions.

1 Background

Hidden Markov models assume the distribution of an observed data point d_t, $t = 1, \ldots, n$, depends on an unobserved (hidden) state $h_t \in \mathcal{S} = \{0, \ldots, S-1\}$. The elements of $\mathbf{h} = (h_1, \ldots, h_n)$ follow a Markov chain with stationary transition matrix $\mathbf{Q} = (q(r,s))$ and initial distribution π_0, often taken to be the stationary distribution of $\mathbf{Q}$. Formally

$$p(h_t | h_1^{t-1}, \mathbf{Q}) = q(h_{t-1}, h_t). \tag{1}$$

where $h_1^{t-1} = (h_1, \ldots, h_{t-1})$. HMM's do not require $\mathbf{h}$ to exist in a physical sense, but the model is more compelling if the hidden states have a physical interpretation. Later sections illustrate how scientific insight about $\mathbf{h}$ can simplify otherwise difficult problems such as choosing S, preventing label switching, and selecting model diagnostics.

The full conditional distribution of d_t is

$$p(d_t | d_{-t}, \mathbf{h}, \theta, \mathbf{Q}, \pi_0) = P_{h_t}(d_t | \theta), \tag{2}$$

where $d_{-t} = \{d_{t'} : t' \neq t\}$, and θ is a parameter vector for the probability distributions $P_0, \ldots, P_{S-1}$. Sometimes $\mathbf{Q}$ is a function of θ, but it is usually treated as a separate parameter. In words (2) says d_t is conditionally independent of all other missing and observed data given h_t and model parameters, a concept illustrated in Figure 1.

HMM's are closely related to other well known classes of models. Finite mixture models (Titterington *et al.*, 1985; Everitt and Hand, 1981) are HMM's where all rows of $\mathbf{Q}$ are equal. State space models (West and Harrison, 1997) assume $\mathbf{h}$ follows a Gaussian process. The Markov modulated

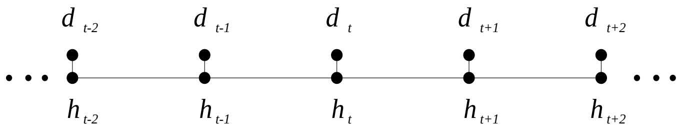


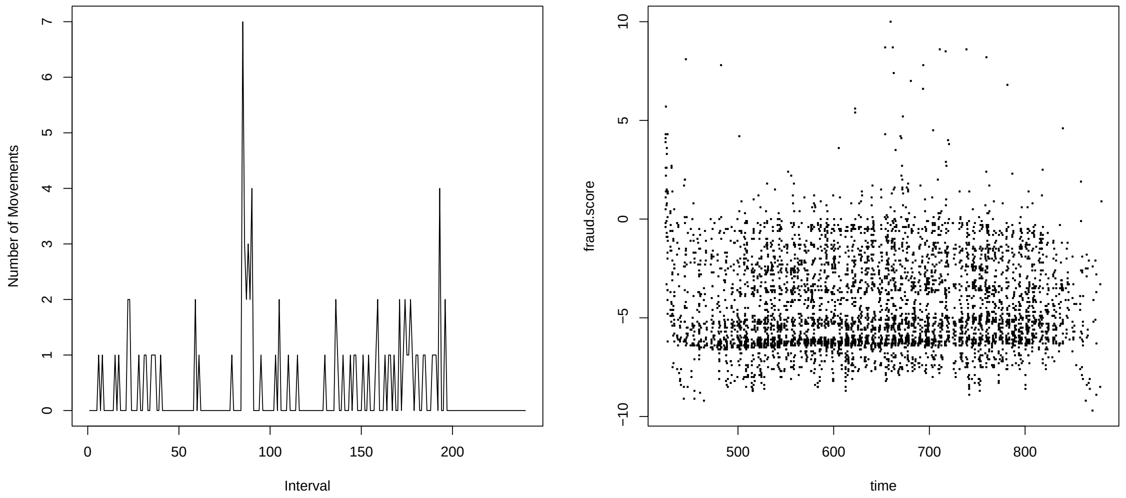
Figure 1: Graphical depiction of HMM dependencies. The conditional distribution of the value any node, given values at all other nodes, depends only on nodes to which it is connected by an edge.

Poisson process (Scott, 1999, 2000) extends HMM's to continuous time. For clarity of exposition we restrict attention to the model defined by (1) and (2) and note generalizations where appropriate. The following examples illustrate our notation and indicate some of the ways (1) and (2) can be usefully expanded.

Example 1 (Fetal Lamb Movements) A popular data set in the HMM literature is the record of fetal lamb movements over 240 consecutive 5 second intervals given by Leroux and Puterman (1992) and replicated here in Figure 2(a). The hidden state h_t describes the fetal activity level during interval t. The activity levels are usually assumed to be $\mathcal{S} = \{\text{passive, active}\}$ or $\mathcal{S} = \{\text{passive, somewhat active, very active}\}$. The fetal activity category evolves according to a stationary Markov chain with transition matrix $\mathbf{Q}$. The number of movements in interval t, d_t, is Poisson with mean θ_{h_t}.

Example 2 (The Business Cycle) Hamilton (1989, 1990) models U.S. GNP using an ARIMA process evolving around means determined by a hidden Markov chain. The hidden state h_t indicates whether the economy is expanding ($h_t = 1$) or contracting ($h_t = 0$) at time t. Here θ contains the ARIMA parameters and the mean GNP growth rates during economic expansions and contractions. The memory in the underlying business cycle is captured by $\mathbf{Q}$. For this example, more links must be added to the top part of Figure 1 to reflect dependence in the observed data even given $\mathbf{h}$.

Example 3 (Network Intrusion) Scott (1999, 2000) uses HMM's to detect criminal intrusion into a telephone account. The observed data d_t is the placement time of the t'th telephone call on the account, which can be augmented by observed call characteristics as in Figure 2(b). The customer responsible for the account generates traffic according to a (possibly non-homogeneous) Poisson process. Criminals break into and leave the account according to a two-state continuous-time Markov process. When a criminal is present he generates additional traffic according to a



(a)

(b)

Figure 2: (a) Number of fetal lamb movements in 240 consecutive five second intervals. (b) Output of a fraud detection system for a telephone account. The vertical axis is a score indicating how closely a telephone call matches the customer's historical calling pattern. Higher scores indicate more unusual calls. Time is in days since Jan. 1 1994.

second Poisson process, independent of the customer. When all processes are homogeneous, data from the account may be represented as a discrete time HMM. The hidden state h_t indicates the fraud status of the t'th call and the criminal's presence or absence during the t'th interval between calls. The parameters θ and $\mathbf{Q}$ are functions of the calling rates for the customer and the criminal, of parameters describing the distribution of call characteristics for the customer and the criminal, and of the rates at which criminals break into and leave the system.

Returning to the general case, extend θ to include $\mathbf{Q}$ and for arbitrary vectors $\mathbf{x} = (x_1, \ldots, x_n)$ write x_j^k for $(x_j, \ldots, x_k)$. The likelihood function for an HMM with observed data d_1^n is

$$p(d_1^n|\theta) = \sum_{\mathbf{h}\in\mathcal{S}^n} \pi_0(h_1) P_{h_1}(d_1|\theta) \prod_{t=2}^{n} q(h_{t-1}, h_t) P_{h_t}(d_t|\theta). \tag{3}$$

The sum in (3) is over S^n elements, so it quickly becomes infeasible to compute even for small values of S as n grows to moderate size. A method for evaluating (3) is needed so the many statistical procedures that depend on likelihood, including likelihood ratio testing, Bayesian model selection, and certain Metropolis-Hastings samplers can be used. The next two sections describe recursive procedures developed to sidestep computational difficulties posed by (3). Section 1.1 explains a technique for quickly evaluating the HMM likelihood. Section 1.2 discusses a related set of forward-backward recursions originally developed to implement an EM algorithm maximizing (3) over θ. Slightly modified forward-backward recursions lead to a Gibbs sampler that is central to the rest of the article.

1.1 The Likelihood Recursion

The likelihood recursion is a procedure for calculating (3) in $O(S^2n)$ steps, instead of the $O(S^n)$ steps suggested by direct evaluation. Define the *forward variable* $\ell_t(h_t) \equiv p(d_1^t, h_t|\theta)$. In words, $\ell_t(s)$ is the joint probability (density) of d_1^t and the event $\{h_t = s\}$. The likelihood contribution from d_1^t is $\ell_t^* = \sum_{s=0}^{S-1} \ell_t(s)$, yielding (3) when $t = n$. The likelihood recursion computes $\ell_t(s)$ from $\{\ell_{t-1}(r) : r \in \mathcal{S}\}$ as follows.

$$\begin{aligned} \ell_t(s) &= \sum_{r=0}^{S-1} p(d_t, d_1^{t-1}, h_t = s, h_{t-1} = r|\theta) \\ &= P_s(d_t|\theta) \sum_{r=0}^{S-1} q(r,s)\ell_{t-1}(r). \end{aligned} \tag{4}$$

For each s, computing $\ell_t(s)$ requires the sum of S quantities. Thus a single step in the recursion is $O(S^2)$, and evaluating (3) is $O(S^2n)$ as indicated above.

Chib (1996) observes that (4) is unstable because it calculates likelihood rather than log likelihood, but stable modifications of (4) exist. Define $\pi_t(s|\theta) = \ell_t(s)/\ell_t^* = p(h_t = s|d_1^t, \theta)$ and let $M_t = \max_s \log\left\{P_s(d_t|\theta)\sum_r q(r,s)\pi_{t-1}(r|\theta)\right\}$. It is easy to show $\log \ell_t^*$ obeys the recursive relationship

$$\log \ell_t^* = \log \ell_{t-1}^* + M_t + \log\left(\sum_{s=0}^{S-1} \exp\left[\log P_s(d_t|\theta) + \log\left(\sum_{r=0}^{S-1}\pi_{t-1}(r|\theta)q(r,s)\right) - M_t\right]\right). \tag{5}$$

Equation (5) scales each component of $\log \pi_t(s|\theta)$ before exponentiation to prevent computer overflow, then increments $\log \ell_t^*$ by the scaling factor and the log of the normalizing constant for $\pi_t(s|\theta)$.

1.2 The Forward-Backward Recursions

The forward-backward recursions were developed by Baum *et al.* (1970) to implement an EM algorithm maximizing (3) over θ. The recursions produce the *backward variable* $\pi'_t(s|\theta) \equiv Pr(h_t = s|d_1^n, \theta)$, for all s, t, which is needed for the E-step of the EM algorithm. Readers familiar with state space models will note the correspondence between the forward-backward recursions and the prediction and smoothing steps of the Kalman filter (Kalman, 1960). The forward recursion accumulates information about the distribution of $\mathbf{h}$ as it moves down the hidden Markov chain. The backward recursion updates the distribution of $\mathbf{h}$ calculated in the forward step once information has been collected from all observed data.

The forward-backward recursions are traditionally derived using (4) as the forward recursion. An alternative representation generates the matrices $\mathbf{P}_2, \ldots, \mathbf{P}_n$, where $\mathbf{P}_t = (p_{trs})$ and $p_{trs} = p(h_{t-1} = r, h_t = s|d_1^t, \theta)$. In words, $\mathbf{P}_t$ is the joint distribution of (h_{t-1}, h_t) given model parameters and observed data up to time t. One computes $\mathbf{P}_t$ from $\mathbf{P}_{t-1}$ as

$$\begin{aligned} p_{trs} &\propto p(h_{t-1} = r, h_t = s, d_t|d_1^{t-1}, \theta) \\ &= \pi_{t-1}(r|\theta) q(r, s) P_s(d_t|\theta), \end{aligned} \tag{6}$$

where proportionality is reconciled by $\sum_r \sum_s p_{trs} = 1$. Notice that $\pi_t(s|\theta) = \sum_r p_{trs}$ can be computed once $\mathbf{P}_t$ is known, setting up the next step in the recursion. Calculating π_t as a margin of $\mathbf{P}_t$ is equivalent to calculation by (5), so the log-likelihood can be stably computed as a by-product of the forward recursion. There is no need to perform a separate likelihood calculation.

The backward recursion replaces $\mathbf{P}_2, \ldots, \mathbf{P}_n$ with $\mathbf{P}'_2, \ldots, \mathbf{P}'_n$, where $\mathbf{P}'_t = (p'_{trs})$ is the conditional distribution of (h_{t-1}, h_t) given model parameters and d_1^n. Note that $\mathbf{P}'_t$ conditions on all the observed data, while $\mathbf{P}_t$ only conditions on data observed up to time t. Clearly $\mathbf{P}_n = \mathbf{P}'_n$, and one obtains $\mathbf{P}'_t$ from $\mathbf{P}_t$ and $\mathbf{P}'_{t+1}$ using Bayes' rule.

$$\begin{aligned} p'_{trs} &= p(h_{t-1} = r|h_t = s, d_1^n, \theta) p(h_t = s|d_1^n, \theta) \\ &= p(h_{t-1} = r|h_t = s, d_1^t, \theta) \pi'_t(s|\theta) \\ &= p_{trs} \frac{\pi'_t(s|\theta)}{\pi_t(s|\theta)} \end{aligned} \tag{7}$$

where $\pi_t'(s|\theta)$ is computed as the appropriate margin of $\mathbf{P}_{t+1}'$.

Focusing on the forward-backward matrices $\mathbf{P}_t$ and $\mathbf{P}_t'$ rather than the forward-backward variables ℓ_t and π_t' helps strip away some of the mystery of the algorithm. HMM's can be understood using the more general theory of graphical models (Cowell *et al.*, 1999), where *graph* is understood in the sense of Figure 1. Marginal distributions for variables in a graphical model can be efficiently calculated using a local computation algorithm based on a junction tree (Dawid, 1992) describing relationships between cliques on the graph. A clique is a set of variables whose members are all neighbors on the graph, so the cliques in Figure 1 are the transitions $A_t = (h_{t-1}, h_t)$ and the completed data $B_t = (h_t, d_t)$. The local computation algorithm operates by calculating the marginal distributions of cliques in the graph. Marginal distributions of individual variables in the cliques may then be derived using low dimensional sums or integrals. The junction tree for an HMM is a chain $B_1 \rightarrow A_2 \rightarrow B_2 \rightarrow \cdots$, and $\mathbf{P}_t$ and $\mathbf{P}_t'$ represent the A_t cliques being marginalized. A general forward recursion for graphical models calculates the marginal distribution of each clique by recursively averaging over the marginal distribution of its parents in the junction tree. The backward recursion updates the distribution of each clique using information accumulated in the forward step, thus ensuring the marginal distributions cohere with the joint distribution of all variables in the model. See Cowell *et al.* (1999) for details of the general algorithm.

Forward-backward recursions for extensions of HMM's are easily derived using the general local computation method for graphical models. The computational gain from the recursions depends on the size of the cliques. As a simple example, suppose h_t obeys a second order Markov chain $p(h_t|h_1^{t-1}) = p(h_t|h_{t-1}, h_{t-2})$. An $O(S^3 n)$ set of forward-backward recursions is immediately available by expanding $\mathbf{P}_t$ to an array of order 3. See Scott (2000) and Romberg *et al.* (2000) for examples where modified forward-backward recursions are applied to more complicated graphs than Figure 1.

2 Posterior Sampling of θ by MCMC

This section discusses MCMC methods for sampling θ from its posterior distribution given d_1^n. HMM's missing data structure naturally admits posterior samplers that alternate between simulating $\mathbf{h}$ given θ and d_1^n and simulating θ given complete data. An obvious way to sample $\mathbf{h}$ is by direct Gibbs (DG) (Albert and Chib, 1993; Robert and Titterington, 1998; Robert *et al.*, 1993). DG draws each h_t from its full conditional distribution

$$p(h_t = s | h_{-t}, d_1^n, \theta) \propto q(h_{t-1}, s) q(s, h_{t+1}) P_s(d_t|\theta) \tag{8}$$

for $t = 1, \ldots, n$, with appropriate adjustments for end effects. When paired with Gibbs or Metropolis-Hastings steps for sampling θ from $p(\theta|d_1^n, \mathbf{h})$, (8) produces a sequence $\{(\theta, \mathbf{h})^{(j)} : j = 1, 2, \ldots\}$ from a Markov chain whose limiting distribution is $p(\theta, \mathbf{h}|d_1^n)$.

Forward-backward Gibbs (FB) (Chib, 1996; Scott, 1999) is our preferred alternative to DG. FB modifies (8) by using a stochastic version of the forward-backward recursions to sample $\mathbf{h}$ directly from $p(\mathbf{h}|\theta, d_1^n)$. The modification leads to a more rapidly mixing algorithm because fewer components are introduced into the Gibbs Markov chain. The forward recursion for FB produces $\mathbf{P}_2, \ldots, \mathbf{P}_n$ exactly as in Section 1.2. The *stochastic backward recursion* begins by drawing h_n from $\pi_n(\cdot|\theta)$, then recursively draws h_t from the distribution proportional to column h_{t+1} of $\mathbf{P}_{t+1}$. To understand the stochastic backward recursion factor $p(\mathbf{h}|\theta, d_1^n)$ as

$$p(\mathbf{h}|d_1^n, \theta) = p(h_n|d_1^n, \theta) \prod_{t=1}^{n-1} p(h_{n-t}|h_{n-t+1}^n, d_1^n, \theta), \tag{9}$$

and notice (Figure 1 may help) that

$$\begin{aligned} p(h_{n-t} = r|h_{n-t+1}^n, d_1^n, \theta) &= p(h_{n-t} = r|h_{n-t+1}, d_1^{n-t+1}, \theta) \\ &\propto p_{n-t+1, r, h_{t+1}}. \end{aligned} \tag{10}$$

Although the stochastic backward recursion replaces (7), Section 3 explains how combining the two backward recursions leads to improved estimates of $\mathbf{h}$. It is easy to implement the stochastic and non-stochastic backward recursions simultaneously once the forward recursion has been run.

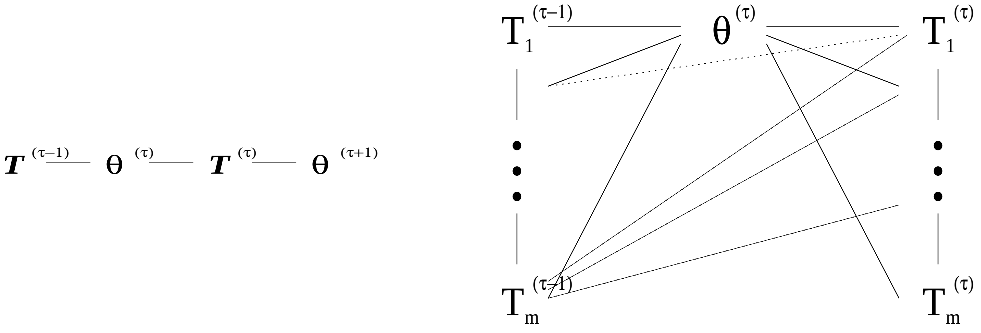
Recursive computing also opens the door for other MCMC procedures that would otherwise be impossible. As an example consider a Metropolis-Hastings sampler (Metropolis *et al.*, 1953; Hastings, 1970) proposing candidate values $\theta_*^{(j+1)}$ from a multivariate normal distribution centered close to the current draw $\theta^{(j)}$. One of $\theta_*^{(j+1)}$ or $\theta^{(j)}$ is promoted to $\theta^{(j+1)}$ depending on a Hastings probability determined by the relative likelihood of $\theta^{(j)}$ and $\theta_*^{(j+1)}$ under $p(\theta|d_1^n)$ and the candidate distribution. This sampler avoids drawing $\mathbf{h}$ altogether by using the likelihood recursion to integrate over $\mathbf{h}$ when calculating Hastings probabilities. Improved candidate distributions may be obtained by modifying the likelihood recursion to produce derivatives of log-likelihood suitable for use in a Langevin diffusion (Gilks and Roberts, 1996).

Metropolis-Hastings procedures that average over $\mathbf{h}$ are useful (Celeux *et al.*, 2000), but there are reasons to prefer sampling $\mathbf{h}$ instead of integrating it out. First, Metropolis-Hastings algorithms tend to perform poorly when the dimension of θ is large. Most candidate draws in high dimensional problems are rejected unless the variance of the candidate distribution is small, in which case the algorithm moves slowly through the parameter space. The usual remedy of partitioning θ and applying smaller Metropolis-Hastings algorithms to the elements of the partition is unattractive for HMM's because partitioning θ requires a separate run of the likelihood recursion for each Hastings probability that must be computed. Highly correlated elements of θ under $p(\theta|d_1^n)$ are often nearly independent under $p(\theta|d_1^n, \mathbf{h})$, so including $\mathbf{h}$ in the sampling algorithm provides an expanded parameter space that may accelerate mixing with respect to θ if $\mathbf{h}$ can be efficiently drawn. This is the idea behind augmented variables samplers from spatial statistics (Damian *et al.*, 1999; Higdon, 1998; Besag and Green, 1993). Also, some function $g(\mathbf{h})$ is often the object of scientific interest, so sampled $\mathbf{h}$'s may be used for inference and as diagnostics for model adequacy and MCMC convergence (Robert *et al.*, 1999).

Later subsections focus on FB and DG to the exclusion of other sampling schemes. In particular we omit discussion of approximate sampling methods not involving MCMC (e.g. Liu and Chen, 1998). Section 2.1 explains why FB mixes more rapidly than DG when drawing from $p(\mathbf{h}|d_1^n, \theta)$. Section 2.2 discusses a label switching issue that must be dealt with when drawing from $p(\theta|d_1^n, \mathbf{h})$. Label switching is a problem regardless of which sampling scheme is used. Section 2.3 uses the model from Example 1 to illustrate a method for preventing label switching and to compare FB and DG.

2.1 Autocovariance of the FB and DG samplers

FB reduces the dependence of h_t on other hidden states drawn in previous Gibbs iterations by sampling $\mathbf{h}$ directly from $p(\mathbf{h}|d_1^n, \theta)$. Faster mixing for $\mathbf{h}$ translates into faster mixing for θ by analogy with a duality principle introduced by Diebolt and Robert (1994) for finite mixture models.



(a) FB sampler

(b) DG sampler

Figure 3: Conditional independence graphs for the FB and DG samplers. The conditional distribution of a variable given all other quantities depends only on variables to which it is connected by an edge. Each $T_j^{(\tau)}$ in DG depends on θ, $T_{j+1}^{(\tau-1)}, \ldots, T_m^{(\tau-1)}$, and $T_1^{(\tau)}, \ldots, T_{j-1}^{(\tau)}$.

This section shows that the autocovariance of complete data sufficient statistics drawn by DG is the FB autocovariance plus a penalty term. The penalty term tends to be positive and increases with the posterior covariance of the hidden states. The theory presented here is a special case of Liu *et al.* (1994, 1995), in whose terminology FB is a *grouped* and DG is an *ungrouped* sampler.

Assume $p(\theta|\mathbf{h}, d_1^n)$ depends on a vector of complete data sufficient statistics $T = \sum_{j=1}^m T_j$. Dependence on d_1^n will henceforth be suppressed in this section. For interpretation purposes imagine $m = n$ and T_j is an S^2 vector of indicator variables describing (h_{j-1}, h_j), though the notation allows greater generality. Both FB and DG produce sequences $\{(\theta^{(\tau)}, \mathbf{T}^{(\tau)}) : \tau = 1, 2, \ldots\}$, where $\mathbf{T}^{(\tau)} = (T_1^{(\tau)}, \ldots, T_m^{(\tau)})$, from Markov chains whose stationary distribution is $p(\theta, \mathbf{T})$. Under FB, $\mathbf{T}^{(\tau)}$ is conditionally independent of $\mathbf{T}^{(\tau-1)}$ given $\theta^{(\tau)}$. DG satisfies the weaker assumption that $T_j^{(\tau)}$ is conditionally independent of $T_k^{(\tau-1)}$ given $\theta^{(\tau)}$, $\{T_{j'}^{(\tau-1)} : j' > j\}$, and $\{T_{j'}^{(\tau)} : j' < j\}$, for $k \leq j$ (see Figure 3).

Once FB and DG have achieved stationarity it is easily established that $(\mathbf{T}^{(\tau-1)}, \theta^{(\tau)})$ and $(\mathbf{T}^{(\tau)}, \theta^{(\tau)})$ have the same marginal distribution. It follows that

$$\begin{aligned} Cov(T^{(\tau-1)}, T^{(\tau)}) &= E\{Cov(T^{(\tau-1)}, T^{(\tau)}|\theta^{(\tau)})\} + Cov\{E(T^{(\tau-1)}|\theta^{(\tau)}), E(T^{(\tau)}|\theta^{(\tau)})\} \\ &= E\{Cov(T^{(\tau-1)}, T^{(\tau)}|\theta^{(\tau)})\} + Var\{E(T^{(\tau)}|\theta^{(\tau)})\} \end{aligned} \tag{11}$$

with E, Cov, and Var defined under either sampler. The first term in (11) is zero under FB. The second is the same for either sampler because FB and DG have the same stationary distribution. Consequently,

$$Cov_{DG}(T^{(\tau-1)}, T^{(\tau)}) = Cov_{FB}(T^{(\tau-1)}, T^{(\tau)}) + E_{DG}\{Cov_{DG}(T^{(\tau-1)}, T^{(\tau)}|\theta^{(\tau)})\}. \qquad (12)$$

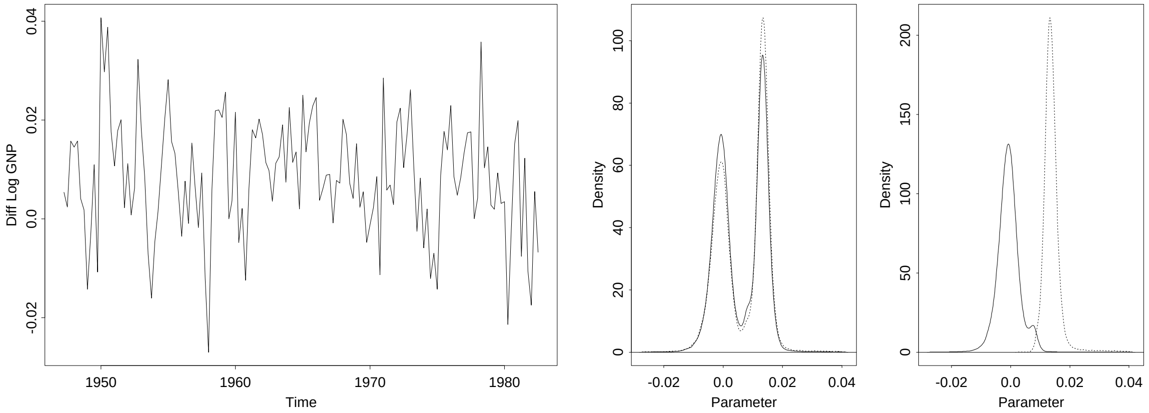
In words, (12) says the autocovariance of the complete data sufficient statistics under direct Gibbs is the FB autocovariance plus a penalty term. To understand the penalty term notice $Cov_{DG}(T^{(\tau-1)}, T^{(\tau)}|\theta^{(\tau)})$ decomposes as

$$\begin{aligned} Cov_{DG}(T^{(\tau-1)}, T^{(\tau)}|\theta^{(\tau)}) &= \sum_{j=1}^{n}\sum_{k=j}^{n} Cov_{DG}(T_j^{(\tau-1)}, T_k^{(\tau)}|\theta^{(\tau)}) + \sum_{j=2}^{n}\sum_{k=1}^{j-1} Cov_{DG}(T_j^{(\tau-1)}, T_k^{(\tau)}|\theta^{(\tau)}) \\ &= UT + LT. \end{aligned}$$

The notation UT and LT indicates sums over the upper and lower triangle of $Cov_{DG}(\mathbf{T}^{(\tau-1)}, \mathbf{T}^{(\tau)}|\theta^{(\tau)})$. It can be shown that UT is nonnegative definite, so the autocovariance under DG tends to be greater than under FB. Furthermore, $Cov_{DG}(T_j^{(\tau-1)}, T_k^{(\tau)}|\theta^{(\tau)}) = Cov_p(T_j, T_k|\theta^{(\tau)})$ for $k < j$, so DG performs worst when the elements of $\mathbf{h}$ are highly related in their posterior distribution. The last observation mathematically expresses a well known feature of DG from spatial statistics (Higdon, 1998; Besag and Green, 1993). Namely, it is hard for DG to move from one configuration of $\mathbf{h}$ to another when there is strong dependence between h_t and its neighbors.

2.2 Label Switching, State Collapsing, and Prior Modeling

Sampling θ from its complete data posterior should be trivial once $\mathbf{h}$ is drawn, but the draw is complicated by an identifiability issue known as *label switching*. The term label switching describes the fact that the HMM likelihood is invariant under arbitrary permutations of the state labels. Most literature on label switching works in the context of finite mixture models, but HMM's face the same issues. Consider the case when $S = 2$. Swap values of θ_0 and θ_1, relabel all points currently in category 0 as category 1, and vice-versa. The complete data likelihood achieves exactly the same value under the new labels as under the old. If θ_0 and θ_1 are exchangeable in their prior distribution then their marginal posterior densities are identical. Label switching is an obstacle



(a)

(b)

Figure 4: (a) Quarterly differences in log U.S. GNP. (b) Posterior distributions of mean parameters in a two state Gaussian HMM applied to the GNP series. Both panels in Figure 4(b) assume prior distribution $p(\mu_s|\tau^2) = \mathcal{N}(\cdot|M_s, \tau^2)$ and $p(1/\tau^2) = \Gamma(DF/2, SS/2)$, where M_s was estimated from EM algorithm, and DF and SS were chosen so the prior mean for τ^2 was comparable to the sample variance. Both panels also truncate $p(\mu_s|\tau^2)$ at the maximum and minimum data values. The prior distribution for the right panel further constrains $\mu_1 > \mu_0$. The unconstrained parameters in the left panel exhibit label switching.

for HMM's regardless of which sampling strategy is used. Its effect can be seen in Figure 4, which shows posterior density estimates for the mean parameters of a two state Gaussian HMM (i.e. $P_s = \mathcal{N}(\cdot|\mu_s, \sigma^2)$) applied to an economic time series. This model is a simplification of the model discussed in Example 2.

Because the data contain no information about the order of state labels, labels may only be identified in the posterior distribution by assumptions in the prior. For example, it is common to assume θ obeys a set of constraints that would be violated by permuting the state labels. Typical constraints involve ordering the means or variances of the mixture components. Parameter constraints can be enforced by explicitly choosing a prior with zero mass on regions where the ordering is violated. Such priors lead to Metropolis-Hastings algorithms that reject draws of θ violating the constraint (Richardson and Green, 1997). Constraints can also be enforced by reparameterizing the model (Robert and Titterington, 1998; Robert, 1998).

Care is needed when choosing parameter constraints because priors that truncate the support of θ can be informative, and enforcing constraints by reparameterizing θ can produce a different model. Celeux *et al.* (2000) provide examples where ordering different sets of model parameters produces noticeably different posterior means of θ. Also note from Figure 4(b) that a weakly informative prior on θ does little to prevent label switching. Scientific insight about $\mathbf{h}$ may therefore provide useful prior information by suggesting an order for θ. "Scientific insight" and "prior information" are often discussed in abstract terms, but here we mean something concrete. For example, the model for U.S. GNP in Figure 4 is intended to associate $h_t = 0$ with recessions and $h_t = 1$ with economic expansions. Thus $\mu_1 < \mu_0$ is nonsensical and should be removed from consideration. Similar logic can often be applied in other circumstances as long $\mathbf{h}$ has a physical interpretation.

Selecting appropriate prior distributions can also help curb state collapsing, a special case of label switching where no observations are allocated to a particular state during a Gibbs iteration. Parameters for a collapsed state are drawn from their prior distribution, so weak priors produce draws of θ far from values justified by data. Hierarchical priors allowing elements of θ to borrow strength from one another can help the sampler recover from a collapsed state. Priors constraining mean parameters to lie within the range of the data are also helpful. Frequent state collapsing is evidence that the model is over parameterized. Thus, variable dimension Monte Carlo methods discussed in Section 4.2 can help deal with collapsed states by allowing a state to "die" and later be reborn in a sensible location.

Scientific insight about $\mathbf{h}$ might not be available in all cases. For example, an HMM might be fit to "nonparametrically" obtain the predictive distribution of d_{n+1} given d_1^n or to cluster the observations of a time series when no intuition exists about the source of the clustering. Stephens (1997, 2000b) proposes a method for identifying mixture components when prior information about $\mathbf{h}$ is unavailable. Stephens' method runs MCMC with no identifiability constraints to sample from a posterior distribution containing $S!$ symmetric modes. At the end of the MCMC run the method searches for a permutation of the state labels leading to marginal distributions of θ that minimize a specified loss function. Loss functions have been developed to promote unimodal marginal distributions of θ and efficient clustering of d_1^n.

2.3 Example: Constrained Markov-Poisson HMM

This section uses a constrained Markov-Poisson HMM $P_s = \text{Poisson}(\theta_s)$ with $\theta_0 < \cdots < \theta_{S-1}$ to illustrate a general method for ordering the variance parameters of hidden Markov and finite mixture models. The method works when $P_0, \ldots, P_{S-1}$ are from a distributional form which is invariant under addition, such as (multivariate) Gaussian, Poisson, gamma with common scale parameter, or binomial with common p. The key is to view d_t as the sum of contributions from various "regimes." Regimes $0, \ldots, h_t$ are active, but regimes $h_t + 1, \ldots, S-1$ are passive and do not contribute to the sum. The result is an ordered variance model because adding two non-degenerate random variables increases their variance. Let $\mathbf{d} = (\mathbf{d}_1, \ldots, \mathbf{d}_n)$ where $\mathbf{d}_t = (d_{0t}, \ldots, d_{S-1,t})$ contains the contribution of each regime to $d_t = \sum_s d_{st}$.

For the Poisson case define $\lambda_s = \theta_s - \theta_{s-1}$, where $\lambda_0 = \theta_0$. If $\boldsymbol{\lambda} = (\lambda_0, \ldots, \lambda_{S-1})$ then

$$p(\mathbf{d}, \mathbf{h}|\boldsymbol{\lambda}, \mathbf{Q}) \propto \pi_0(h_1) \left(\prod_{r=0}^{S-1} \prod_{s=0}^{S-1} q(r,s)^{n_{rs}} \right) \left(\prod_{s=0}^{S-1} \lambda_s^{\sum_t d_{st}} \exp(-n_s \lambda_s) \right), \tag{13}$$

where $n_s = \sum_{t=1}^n I(h_t \geq s)$ is the number of times regime s was active and n_{rs} is the number of transitions from r to s. Assume prior density $p(\boldsymbol{\lambda}, \mathbf{Q}) = \prod_{r=0}^{S-1} \Gamma(\lambda_r | a_r, b_r) \mathcal{D}(Q_r|\boldsymbol{\nu}_r)$, where Γ and $\mathcal{D}$ are the gamma and Dirichlet densities, and Q_r is the r'th row of $\mathbf{Q}$. The rows of $\mathbf{Q}$ and elements of $\boldsymbol{\lambda}$ are independent in their posterior distribution given $\mathbf{d}$ and $\mathbf{h}$, with

$$p(\lambda_s|\mathbf{d}, \mathbf{h}) = \Gamma(a_s + \sum_t d_{st}, b_s + n_s), \qquad p(Q_r|\mathbf{d}, \mathbf{h}) = \mathcal{D}(\boldsymbol{\nu}_r + \mathbf{n}_r),$$

where $\mathbf{n}_r = (n_{rs})$ is the vector of transitions out of state r.

To apply FB, note that $p(\mathbf{d}, \mathbf{h}|\boldsymbol{\lambda}, \mathbf{Q}, d_1^n) = p(\mathbf{h}|\boldsymbol{\lambda}, \mathbf{Q}, d_1^n) p(\mathbf{d}|\mathbf{h}, \boldsymbol{\lambda}, \mathbf{Q}, d_1^n)$. FB simulates $\mathbf{h}$ from $p(\mathbf{h}|\boldsymbol{\lambda}, \mathbf{Q}, d_1^n)$ using the stochastic forward-backward recursions. The detailed regime contributions $(\mathbf{d}_1, \ldots, \mathbf{d}_n)$ are conditionally independent given $\mathbf{h}$ and $\boldsymbol{\lambda}$. If $h_t = r$ then no regime higher than r can contribute to d_t, so $d_{st} = 0$ for all $s > r$. The full conditional distribution of $(d_{0t}, \ldots, d_{rt})$ is multinomial with total d_t and probability vector proportional to $(\lambda_0, \ldots, \lambda_r)$.

Reckless use of DG leads to trouble because the full conditional of h_t depends on $\mathbf{d}_t$.

$$p(h_t = s|\boldsymbol{\lambda}, \mathbf{Q}, h_{-t}, \mathbf{d}) \propto q(h_{t-1}, s) q(s, h_{t+1}) \prod_{r=0}^{s} \lambda_r^{d_{rt}} \exp(-\lambda_r) \prod_{r=s+1}^{S-1} I(d_{rt} = 0). \tag{14}$$

If $d_{st} > 0$ for any $s > r$ then (14) places probability 0 on the events $\{h_t = r\}$, making it difficult for the sampler to move from larger to smaller values of h_t. A compromise defines the missing data for time t as $(h_t, \mathbf{d}_t)$ and samples $(h_t, \mathbf{d}_t)$ by nested conditioning. The full conditional factors as

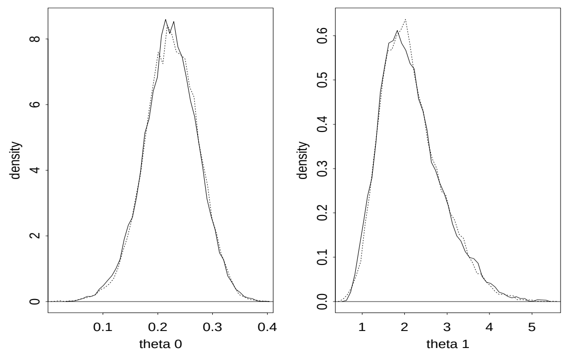


Figure 5: Marginal distributions for θ_0 and θ_1 computed by the FB (dashed) and DG (solid) samplers for the two state Markov-Poisson HMM applied to the fetal lamb data. Density estimates are based on Gibbs runs of 10,000 iterations with the first 100 dropped. There is little difference between the FB and DG density estimates because FB and DG sample from the same target density.

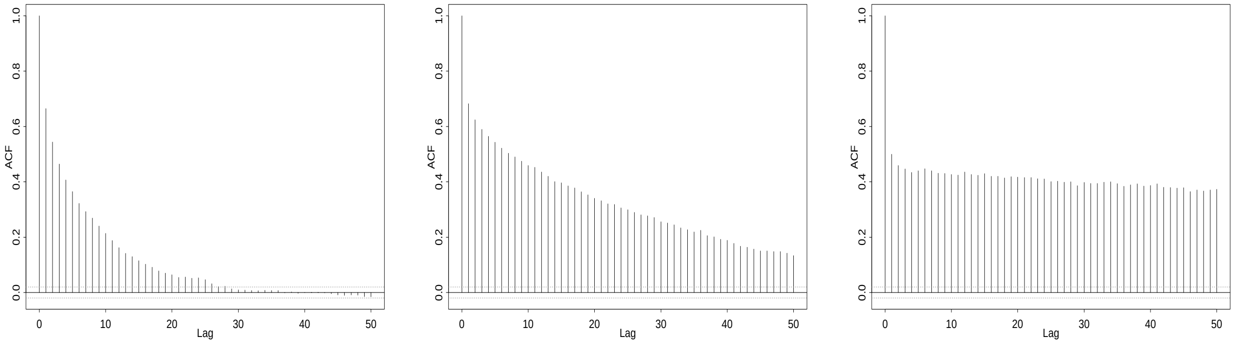
$p(h_t, \mathbf{d}_t|\boldsymbol{\lambda}, \mathbf{Q}, h_{-t}, \mathbf{d}_{-t}, d_1^n) = p(h_t|\boldsymbol{\lambda}, \mathbf{Q}, h_{-t}, \mathbf{d}_{-t}, d_1^n)p(\mathbf{d}_t|\boldsymbol{\lambda}, \mathbf{Q}, \mathbf{h}, \mathbf{d}_{-t}, d_1^n)$. First draw h_t from

$$p(h_t = s|\boldsymbol{\lambda}, \mathbf{Q}, h_{-t}, \mathbf{d}_{-t}, d_1^n) \propto q(h_{t-1}, s)q(s, h_{t+1})P_s(d_t|\theta).$$

Then draw $\mathbf{d}_t$ as in FB.

Figure 5 displays marginal posterior density estimates calculated from FB and DG for the two state Markov-Poisson HMM applied to the fetal lamb data. There is little difference between the densities because both samplers have the same stationary distribution. Their difference is highlighted in Figure 6, which shows empirical autocorrelation functions for Gibbs draws of θ_0 under FB, the DG sampler drawing h_t from (14), and the DG sampler drawing $(h_t, \mathbf{d}_t)$ by nested conditioning. Figure 6 illustrates the cost of introducing unnecessary Gibbs components into the sampling algorithm. Figure 6(a) is from a rapidly mixing sampler with only two components. Mixing suffers as Figure 6(b) increases the number of Gibbs components to $n + 1$. Figure 6(c) includes n additional Gibbs components that are highly correlated with other missing data drawn by the sampler, causing a terrible waste of the computer's resources.

Computationally, DG is an $O(Sn)$ algorithm, while FB is $O(S^2n)$. Thus DG's speed can be an advantage as S increases. Table 1 shows the CPU time needed for FB and DG to generate 1,000 draws from the posterior distribution of θ given data simulated from Markov-Poisson HMM's with different sample sizes and state spaces. Directly comparing CPU time for FB and DG is not



(a) (b) (c)

Figure 6: Autocorrelation functions for Gibbs draws of θ_0 under (a) the FB recursions, (b) direct Gibbs sampling where $\mathbf{d}_t$ is drawn conditional on h_t, and (c) direct Gibbs. All are based on 10,000 draws from the constrained two state Markov-Poisson HMM applied to the fetal lamb data. Direct Gibbs mixes much slower than the other two algorithms because the additional missing data.

completely fair because FB calculates the likelihood associated with each drawn θ, a quantity needed in sections 4 and 5. Therefore simulation times are shown for the DG algorithm with and without the likelihood recursion. Table 1 illustrates the increasing toll exacted by the forward-backward recursions as S increases. When $S = 2$, FB takes about 30% longer than the direct Gibbs algorithm not computing the likelihood. When $S = 8$, FB takes more than twice as long. Pairing DG with the likelihood recursion eliminates DG's speed advantage when $S = 2$, but when $S = 8$ the algorithm is comfortably between FB and raw DG. Unless the number of states is exceptionally large we prefer FB because of its superior mixing properties, because it calculates likelihood "for free" as part of the forward recursion, and because it allows more efficient estimation of $\mathbf{h}$ as discussed in Section 3.

3 Estimating the Hidden States

Estimating $\mathbf{h}$ is often the central question in applied problems, such as the fraud detection problem from Example 3. All Bayes estimates of $\mathbf{h}$ derive from its posterior distribution $p(\mathbf{h}|d_1^n)$, a high dimensional distribution that must be summarized to be understood. For most applications summarizing $p(\mathbf{h}|d_1^n)$ with its marginal distributions $\pi_t'(s) = Pr(h_t = s|d_1^n)$ is sufficient. Sometimes the overall configuration of $\mathbf{h}$ is more interesting than the value of an individual state. For example,

	S= 2			S= 4			S= 8		
	DG	DGL	FB	DG	DGL	FB	DG	DGL	FB
n=500	4.7	6.5	6.2	7.9	12.2	13.8	24.4	32.1	50.1
n=1000	8.6	12.2	11.5	15.7	22.6	26.6	48.8	62.5	99.8
n=1500	13.0	17.4	17.9	22.9	33.2	40.3	63.2	84.9	144.4
n=2000	16.7	23.0	21.9	31.3	44.2	53.6	82.5	111.9	188.2

Table 1: CPU time (in seconds) required to generate 1,000 Gibbs draws for the FB and DG samplers applied to data simulated from Markov-Poisson mixtures of length n. DGL is the DG algorithm running the likelihood recursion. The true parameter values in the simulation are $\theta_0 = 1$, $\theta_1 = 2$ and $\theta_s = \theta_{s-1} + \theta_{s-2}$ for $s = 2, \ldots, S-1$, $q(s,s) = 0.6$ and $q(s, s+1) = q(s, s-1) = 0.2$ for $s = 1, \ldots, S-2$, $q(0,0) = q(S-1, S-1) = 0.8$, and $q(0,1) = q(S, S-1) = 0.2$.

in gene sequencing (Liu *et al.*, 1999) h_t might represent a DNA base element where $\mathbf{h}$ represents a gene, or in speech recognition (Juang and Rabiner, 1991) h_t may be a phoneme in the word $\mathbf{h}$. Maximum *a posteriori* (MAP) estimation selects $\hat{\mathbf{h}}$ to maximize $p(\mathbf{h}|d_1^n)$, ensuring a coherent reconstruction of the hidden chain that may differ from a corresponding reconstruction from its marginals. Recursive computing improves both marginal and MAP summaries of $p(\mathbf{h}|d_1^n)$, which are discussed in Sections 3.1 and 3.2 respectively.

3.1 Marginal Distribution of h_t

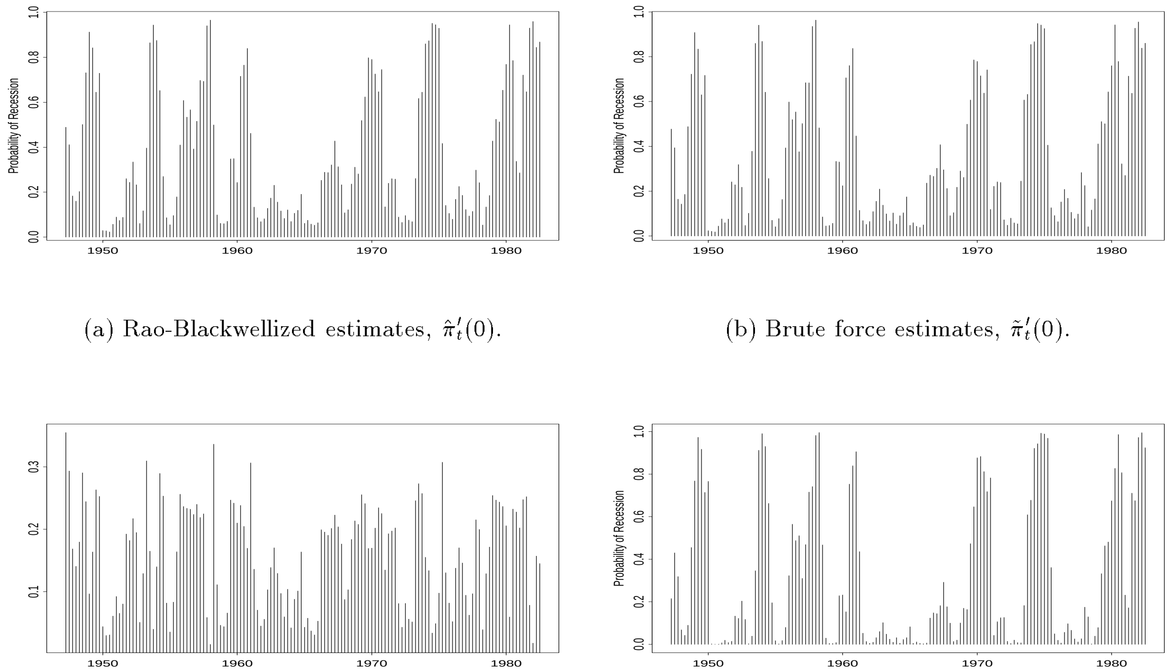
Let $\theta^{(1)}, \ldots, \theta^{(m)}$ be Gibbs draws of θ and let $\mathbf{h}^{(1)}, \ldots, \mathbf{h}^{(m)}$ be Gibbs draws of $\mathbf{h}$ with $\mathbf{h}^{(j)} = (h_1^{(j)}, \ldots, h_n^{(j)})$. An obvious estimate of $\pi'_t(s)$ is

$$\tilde{\pi}'_t(s) = 1/m \sum_{j=1}^{m} I(h_t^{(j)} = s). \tag{15}$$

The forward-backward recursions improve (15) through Rao-Blackwellization (Gelfand and Smith, 1990; Casella and Robert, 1996). The Rao-Blackwellized estimate

$$\hat{\pi}'_t(s) = 1/m \sum_{j=1}^{m} \pi'_t(s|\theta^{(j)}) \tag{16}$$

sheds a layer of Monte Carlo variability by averaging probabilities rather than events simulated with those probabilities. Calculating (16) requires the non-stochastic backward recursion to produce $\pi'_t(s|\theta^{(j)})$, but simultaneously running both the stochastic and non-stochastic backward recursions



(a) Rao-Blackwellized estimates, $\hat{\pi}'_t(0)$.

(b) Brute force estimates, $\tilde{\pi}'_t(0)$.

(c) Difference: $SD(\tilde{\pi}'_t(0|\theta)) - SD(I(h_t = 0))$.

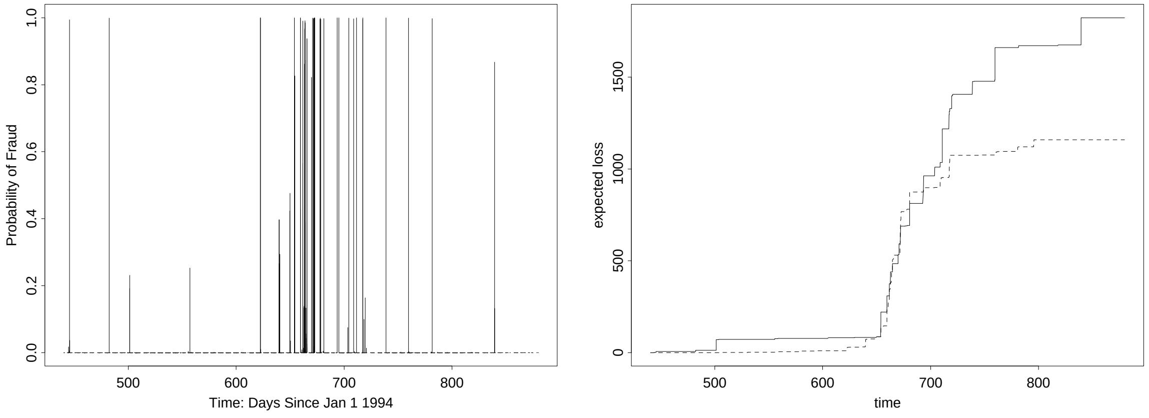
(d) Empirical Bayes estimates, $\pi'_t(0|\hat{\theta})$.

Figure 7: Marginal probabilities of recession based on 100,000 Gibbs iterations for the two state Gaussian HMM applied to the U.S. GNP data. These estimates agree closely with official business cycle data from http://www.nber.org/cycles.html.

requires little effort once the forward recursion has been implemented.

Figure 7 compares $\hat{\pi}_t$ to $\tilde{\pi}_t$ and to the empirical Bayes probability of economic recession based on the two state Gaussian model for the U.S. GNP data from Figure 4. Figure 7(c) shows the difference in posterior standard deviations for Gibbs draws of $\pi'_t(0|\theta^{(j)})$ and $I(h_t^{(j)} = 0)$. If the MCMC output had been independently sampled then the standard deviations represented by Figure 7(c) would be divided by $\sqrt{m}$ to produce standard errors for the estimates in Figures 7(a) and (b). The extra variability in $\tilde{\pi}'_t$ can clearly be overcome by running the sampler longer, but $\hat{\pi}'_t$ produces more information content per iteration and is a virtually costless addition to the FB sampler.

The two Bayes estimates in Figure 7 are more similar to one another than to the empirical Bayes estimate $\pi'_t(0|\hat{\theta})$, which can only be calculated by the forward-backward recursions. Because the empirical Bayes probabilities ignore uncertainty about θ, they tend be closer to zero and one. Empirical Bayes probabilities are inferior to full Bayes, but they are useful when the cost of



(a)

(b)

Figure 8: (a) Probability of fraud for telephone calls on the account presented in Figure 2(b). (b) Expected loss due to fraud based on the HMM (dotted line) and a mixture model not considering time (solid line).

MCMC calculations is prohibitive. For example, the fraud detection model described in Example 3 is designed to monitor real time data from several million accounts like the one in Figure 2(b). MCMC may be used sparingly, perhaps once per month, to update estimates of θ for each account, but MCMC is far too slow to monitor the entire network of accounts in real time. Figure 8(a) presents the empirical Bayes probability of fraud for each telephone call on the account, which may be quickly computed by a single run of the forward-backward recursions based on a past estimate of θ. Without the forward-backward recursions empirical Bayes calculations for an HMM would be intractable. Instead we might ignore the effect of time and screen for fraud using a finite mixture model. Figure 8(b) compares the cumulative expected financial loss due to fraud computed from an HMM and from a finite mixture model. The HMM is clearly superior to the finite mixture model because it allows information about fraud to be shared across time. The HMM understands that occasional calls with large fraud scores represent weaker evidence of fraud than a cluster of such calls, so it generate fewer "false alarms" than the finite mixture model.

3.2 Maximum *a Posteriori* Estimation of h

The *Viterbi algorithm* (Viterbi, 1967; Fredkin and Rice, 1992) is a method of finding the empirical Bayes *most likely trajectory* $\hat{\mathbf{h}} = (\hat{h}_1, \ldots, \hat{h}_n)$ maximizing $p(\mathbf{h}|d_1^n, \theta)$. The algorithm uses a forward-backward strategy similar to Section 1.2, but with maximizations replacing averages. Begin by observing that the $\mathbf{h}$ optimizing $p(\mathbf{h}|d_1^n, \theta)$ also optimizes $p(\mathbf{h}, d_1^n|\theta)$, so the problem reduces to maximizing the complete data likelihood. Define $L_1(s) = \pi_0(s)P_s(d_1|\theta)$ and $L_t(h_t) = \max_{h_1,\ldots,h_{t-1}} p(h_1^t, d_1^t|\theta)$. In words, $L_t(s)$ is the largest contribution to the complete data likelihood that can be obtained if $h_t = s$. The quantities $L_t(\cdot)$ are computed recursively through

$$L_t(s) = \max_r \left[L_{t-1}(r)q(r,s)\right] P_s(d_t|\theta). \tag{17}$$

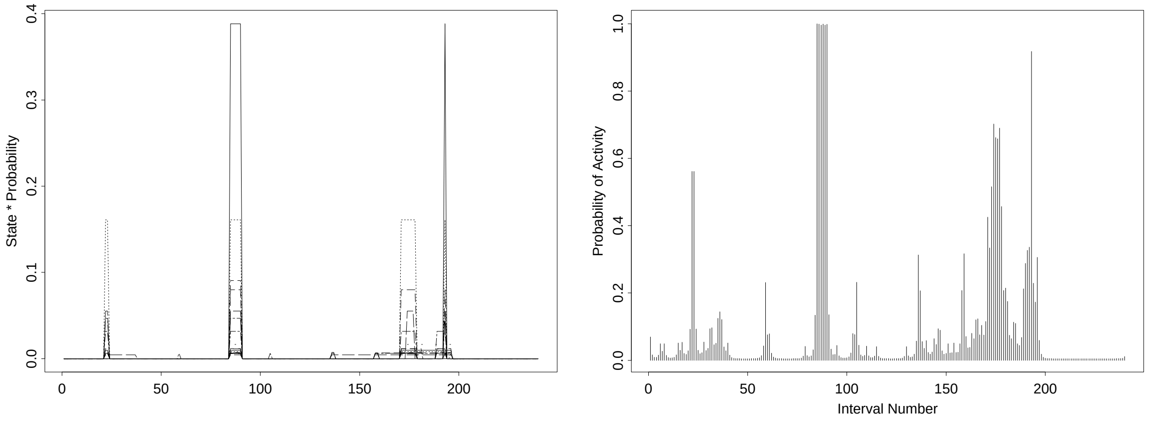
To find $\hat{\mathbf{h}}$, run (17) for $t = 1, \ldots, n$, storing each $L_t(\cdot)$ as the algorithm progresses. Choose h_t to maximize $L_n(s)$, and for $t = n-1, \ldots, 1$ choose

$$\hat{h}_t = \arg\max_{r \in \mathcal{S}} L_t(r) q(r, \hat{h}_{t+1}) \tag{18}$$

to maximize the complete data likelihood conditional on $\hat{h}_{t+1}$. Note that $\hat{h}_1, \ldots, \hat{h}_t$ are determined when $\arg\max_r L_t(r)q(r,s)$ is the same for all s, a frequent event when S is small. When this occurs, we say the estimation has *converged.* At each convergence, select $\hat{h}_t$ to maximize $L_t(r)q(r,s)$ over r, and use (18) to compute all previously undetermined elements of $(\hat{h}_1, \ldots, \hat{h}_{t-1})$. Earlier values of $L_t(\cdot)$ are no longer necessary and may be discarded. Updating $\hat{\mathbf{h}}$ at each convergence limits the memory demands of the Viterbi algorithm and allows the algorithm to run in nearly real time.

The Viterbi algorithm suffers from the same stability issues as the forward-backward recursions, with the same remedies. Renormalizing L_t at each iteration and transforming the algorithm to the log scale may be accomplished by analogy with (5). The algorithm may be expanded to accommodate more general HMM's by modifying the general local computation method for marginalization discussed in Section 1.2. See Cowell *et al.* (1999) for details.

There is no Viterbi-style algorithm for maximizing $p(\mathbf{h}|d_1^n)$ because averaging over θ destroys



(a)

(b)

Figure 9: Estimating $\mathbf{h}$ for the fetal lamb data. (a) Most likely trajectories times their MCMC frequency. (b) Marginal probabilities $\hat{\pi}'_t(1)$.

the model's Markov structure. Simply evaluating the objective function requires Monte Carlo.

$$
\begin{aligned}
p(\mathbf{h}|d_1^n) &= \int p(\mathbf{h}|d_1^n, \theta)p(\theta|d_1^n)\ d\theta \\
&\approx 1/m \sum_{j=1}^{m} p(\mathbf{h}|d_1^n, \theta^{(j)}).
\end{aligned}
\tag{19}
$$

Equation (19) is difficult to optimize because the optimal $\mathbf{h}$ need not be the most likely trajectory for any particular θ. A primitive solution is to select the most commonly occurring $\mathbf{h}^{(j)}$ in the Gibbs run. Random MCMC fluctuations in $\mathbf{h}^{(j)}$ will require extremely long MCMC runs, along with ample memory and hard disk storage, for this approach to be effective. An approximate but preferable solution can be obtained by viewing $\hat{\mathbf{h}}$ as a function of θ and selecting $\hat{\mathbf{h}}(\theta^{(j)})$ from the MCMC run that is "most often most likely." The set of distinct $\hat{\mathbf{h}}(\theta^{(j)})$'s produced by the sampler will be much smaller than the set of distinct $\mathbf{h}^{(j)}$'s. Figure 9(a) shows the most likely trajectories for the fetal lamb data multiplied by their frequency in the MCMC output. Marginal probabilities are included for reference in Figure 9(b). Note that some states with high marginal probability of fetal activity, like intervals 22 and 23, may not be represented when $\mathbf{h}$ is estimated by MAP.

4 Selecting the State Space Size

Up to now S has been assumed known from the context of the application, but sometimes deciding S is a question of scientific interest. Absent firm *a priori* theories, choosing S is a model selection problem that views S as a random variable with a posterior distribution given d_1^n. Recursive computing is important for model selection because standard Bayesian model selection and model averaging techniques require the ability to compute likelihoods for the models being considered. Section 4.1 explains how to calculate $p(S|d_1^n)$ by MCMC. Section 4.2 discusses recently developed variable dimension Monte Carlo methods that may be used instead of standard model selection theory.

4.1 Calculating $p(S|d_1^n)$ by MCMC

For practical reasons assume $S \in \{1, \ldots, S_{\max}\}$. Bounding S is a mild restriction, because choosing $S_{\max}$ to be the number of distinct values in d_1^n allows each data value to have its own mixture component. Smaller values of $S_{\max}$ can be chosen, but no larger state space is identifiable.

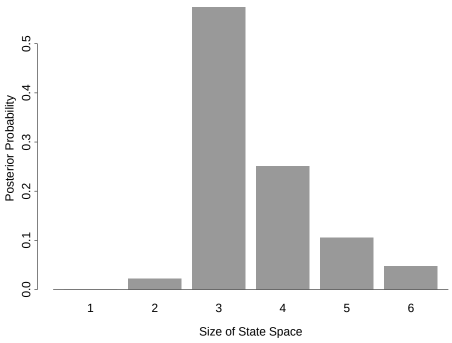
Redefine $\theta = (\theta_1, \ldots, \theta_{S_{\max}})$, where θ_S is the parameter of an HMM with state space $\mathcal{S} = \{0, \ldots, S-1\}$. Formally, $p(d_1^n|\theta, S) = p(d_1^n|\theta_S, S)$. The dimension of θ is $\sum_{S=1}^{S_{\max}} k_S$ where k_S is the number of free parameters in θ_S. Assume

$$p(\theta, S) = p(S) \prod_{r=1}^{S_{\max}} p(\theta_r). \tag{20}$$

It follows from (20) that $\theta_1, \ldots, \theta_{S_{\max}}$ are conditionally independent in their posterior distribution given d_1^n and S, and may be independently sampled by $S_{\max}$ parallel Gibbs samplers. Let $\theta^{(1)}, \ldots, \theta^{(m)}$ represent m draws of θ from the samplers, where $\theta^{(j)} = (\theta_1^{(j)}, \ldots, \theta_{S_{\max}}^{(j)})$. A Monte Carlo calculation of $p(S|d_1^n)$ is

$$\begin{aligned} p(S|d_1^n) &= \int p(S|d_1^n, \theta) p(\theta|d_1^n)\, d\theta \\ &\approx 1/m \sum_{j=1}^{m} p(S|d_1^n, \theta^{(j)}), \end{aligned} \tag{21}$$

where $p(S|d_1^n, \theta^{(j)}) \propto p(d_1^n|\theta_S^{(j)}, S)p(S)$. The normalizing constant for $p(S|d_1^n, \theta^{(j)})$ is easily com-



(a) Posterior probability of S based on an MCMC run of 30,000 iterations.

S	maximized log-posterior	k_S	$C(S)$	BIC
1	-174.3	1	-177.0	354.0
2	-150.7	4	-161.6	323.2
3	-140.7	9	-165.3	330.6
4	-139.2	16	-183.1	366.2
5	-139.5	25	-208.0	416.0
6	-139.8	36	-238.4	476.8

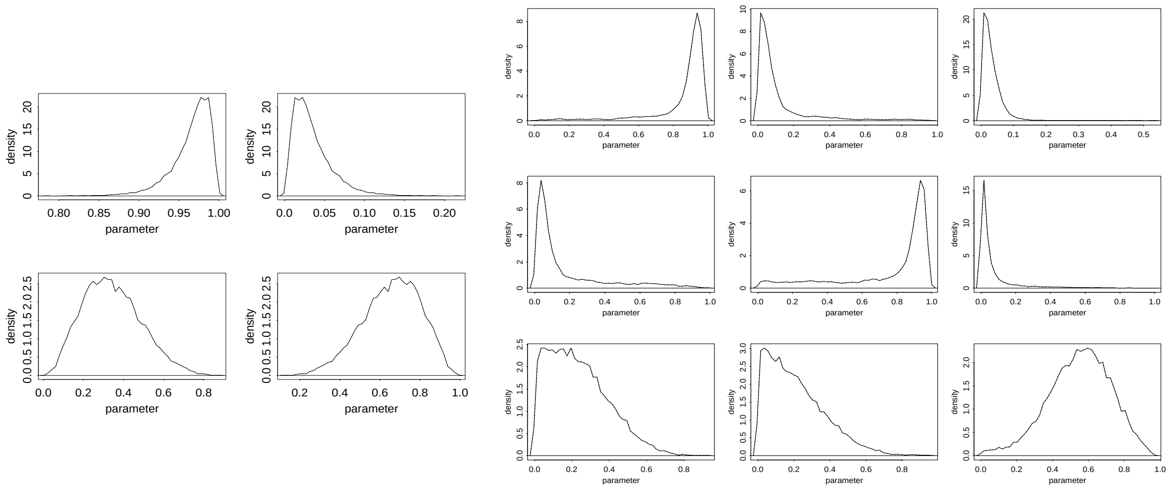
(b) Maximized log posterior, number of free parameters, Schwarz criterion, and BIC for each S.

Figure 10: Comparing $P(S|D)_1^n)$ to BIC for Markov-Poisson mixtures applied to the fetal lamb data. The two methods give different conclusions.

puted because S takes values on a finite set. Up to proportionality, (21) averages the $S_{\max}$ likelihoods corresponding to each $\theta^{(j)}$ over the life of the Gibbs sampler.

Calculating $p(S|d_1^n)$ by MCMC is an improvement over the Schwarz criterion (Schwarz, 1978; Kass and Raftery, 1995), an asymptotic approximation to $\log p(S|d_1^n)$. Minus twice the Schwarz criterion is the Bayesian information criterion (BIC) which has been used to select S for hidden Markov models (e.g. Leroux and Puterman, 1992, and others). The Schwarz criterion is $C(S) = \log \hat{\ell} - k_S \log(n)/2$, where $\hat{\ell}$ is the likelihood $p(d_1^n|\theta_S, S)$ maximized over θ_S, n is the sample size, and k_S is the number of free parameters in θ_S. The Schwarz criterion (equivalently BIC) assumes $p(S)$ is uniform and approximates $p(\theta_S|d_1^n, S)$ with a multivariate normal to be integrated over using Laplace's approximation (Tierney and Kadane, 1986).

Figure 10 compares MCMC estimates of $p(S|d_1^n)$ with BIC for Markov-Poisson HMM's applied to the fetal lamb data. The two methods lead to different conclusions. BIC suggests a two state model, but the three state model is most likely under $p(S|d_1^n)$. BIC penalizes the larger models more than it should based on Figure 10(a), suggesting the sample size of 240 is not large enough to justify BIC's asymptotics. Figure 11 shows the marginal posterior densities of $q(r, s)$ under the two and three state Markov-Poisson models applied to the fetal lamb data. Indeed, BIC's normality assumptions are violated by asymmetry, heavy tails, and proximity to boundaries. The posterior



(a) Two State Model

(b) Three State Model

Figure 11: Marginal posterior densities for elements of $\mathbf{Q}$ under the two and three state Markov-Poisson HMM's applied to the fetal lamb data. Figures are arranged in the same order as elements of $\mathbf{Q}$, so that the first row, second column shows the posterior density of $q(0,1)$. Results are based on MCMC runs of 10,000 iterations with the first 1,000 deleted. The densities are far from normal, especially in Figure (b).

density of $q(1,1)$ when $S = 3$, shown in the central plot of Figure 11(b), is a specific example. It is highly probable that $q(1,1)$ lies in a narrow interval centered around 0.93, although there is a small but appreciable probability it falls somewhere between 0 and 0.8. Roughly speaking, BIC approximates the posterior density of $q(1,1)$ by a normal distribution centered at 0.93 with variance chosen to cover the "narrow interval." BIC integrates over the approximating normal density, ignoring the probability mass between 0 and 0.8.

4.2 Variable Dimension Monte Carlo

Variable dimension Monte Carlo (VDMC) methods are sampling schemes that allow S, and thus the dimension of θ, to vary within the sampling algorithm so that the Monte Carlo distribution of S is $p(S|d_1^n)$. As with label switching more attention has been paid to finite mixture models than HMM's in the VDMC literature. The growing list of VDMC algorithms that have been applied to finite mixture models includes jump diffusions (Grenander and Miller, 1994; Phillips and Smith, 1996), reversible jump MCMC (Green, 1995; Richardson and Green, 1997), and birth-death sampling (Stephens, 2000a). Reversible jump MCMC has been applied to HMM's by Robert *et al.* (2000), and the other methods could be similarly extended. Reversible jump MCMC is distinct from the other two methods because it accommodates sampling $\mathbf{h}$ and θ together. For reversible jump MCMC the forward-backward recursions can improve mixing for θ_S and $\mathbf{h}$ during periods of the algorithm where S remains constant. Jump diffusions and birth-death sampling integrate over $\mathbf{h}$ much like the Metropolis-Hastings algorithm discussed in Section 2. Recursive computing is a requirement for these methods because they must be able to evaluate the likelihood of proposed jumps, births, and deaths.

A key difference between (21) and variable dimension Monte Carlo is the definition of the parameter space. If $\theta_s \in \Theta_s$ for $s \in \{1, \ldots, S_{\max}\}$ then (21) simulates θ from $\Theta = \Theta_1 \times \cdots \times \Theta_{S_{\max}}$. Variable dimension Monte Carlo samples from $\Theta^* = \Theta_1 \cup \cdots \cup \Theta_{S_{\max}}$. The distinction between Θ and Θ^* is important because the scientific interpretation of θ often changes with its dimension, in much the same way as regression coefficients are interpreted differently when new variables are added to a regression model. As $S_{\max}$ grows the need to run $S_{\max}$ Gibbs samplers in (21) becomes a burden. Variable dimension Monte Carlo methods require careful tuning and clever choices of potential moves between parameter spaces. In return, they escape the computational burden of (21) for large $S_{\max}$ by rarely sampling (S, θ) from regions of small posterior probability. The advantages of (21) are that it is easily implemented when $S_{\max}$ is small, and it computes $p(S|d_1^n, \theta^{(j)})$ directly at each iteration rather than by simulation, much in the spirit of Section 3.1.

5 Diagnostics

This section briefly discusses diagnostics for MCMC convergence and for model adequacy. Cowles and Carlin (1996) and Mengersen *et al.* (1999) review the large body of work on MCMC convergence diagnostics. We call special attention to Robert *et al.* (1999) who deal specifically with HMM's. Most MCMC convergence diagnostics seek to determine whether the distribution of some function $g(\theta)$ has stabilized under the long run frequency properties of the MCMC chain. The function g is typically left unspecified. The user may either apply the diagnostic to each component of θ or select g to be some global summary of the model parameters. In many ways, the most natural global summary of a multidimensional parameter θ is its value under its posterior distribution $p(\theta|d_1^n)$.

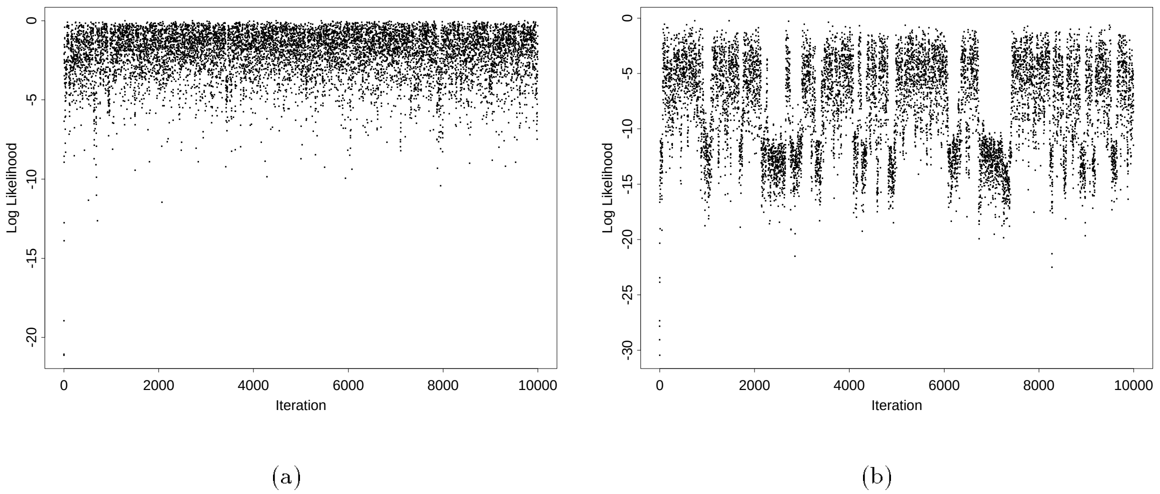
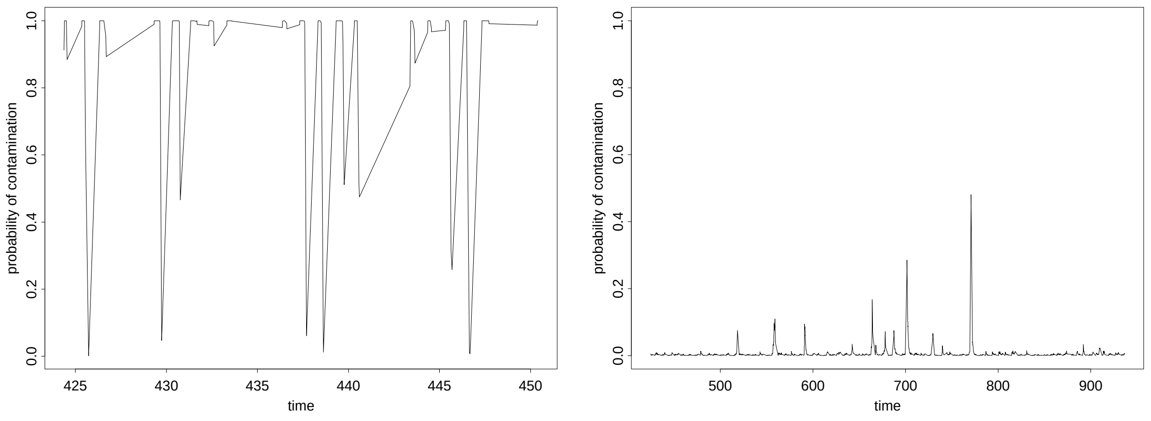


Figure 12: Log-posteriors from Gibbs runs of two (a) and three (b) state Markov-Poisson HMM's on fetal lamb data. Only the first 10,000 values are shown, but log-posterior values are scaled by the maximum achieved in the full run of 100,000 iterations. The chains for both models converge very quickly.

Thus, a final benefit of the likelihood and forward-backward recursions is that they provide a global summary of the model that may be monitored to deduce MCMC convergence. Figure 12 shows log-posterior values from the FB sampler applied to the two and three state Markov-Poisson HMM's for the fetal lamb data. Both samplers rapidly achieve regions of high posterior density. The second mode apparent in Figure 12(b) is responsible for the non-normal posterior distribution of $\mathbf{Q}$ noted in Figure 11. Frequent switching between modes in Figure 12(b) suggests the mixing weights for the modes are well estimated.

The posterior distribution of $\mathbf{h}$ can be an excellent diagnostic for model adequacy when the hidden states are interpretable. If the hidden states are not behaving as expected then the model probably needs more structure. Figure 13(a) shows empirical Bayes probabilities $\pi_t'(h_t = 1|\hat{\theta})$ for the fraud detection model from Example 3 applied to an account known to not contain fraud. The hidden states are supposed to represent a criminal's presence or absence. Instead, Figure 13(a) suggests they describe whether the company's office is closed or open for business. An expanded model that considers the company's hourly and daily calling behavior produces the more believable estimates shown in Figure 13(b).

Appropriate diagnostics for model adequacy will depend on the application at hand, but two

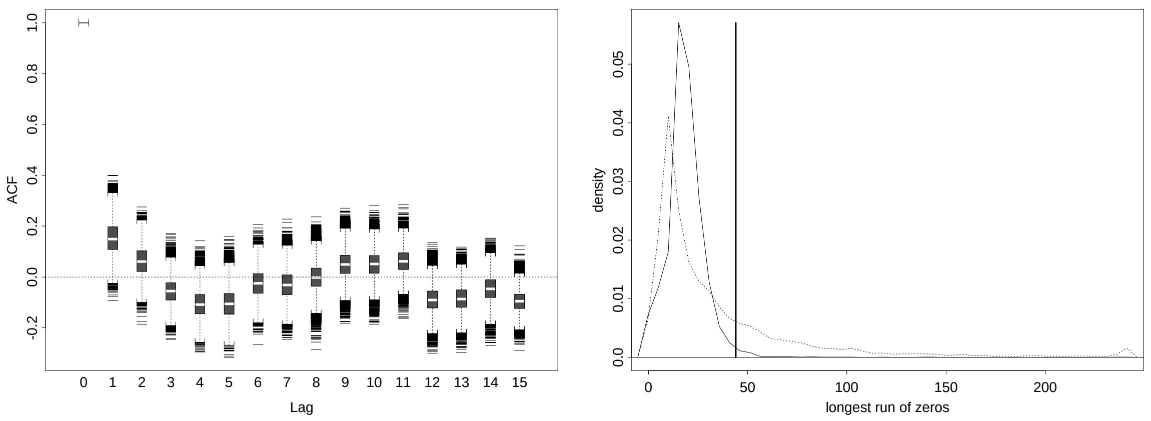


(a) (b)

Figure 13: Empirical Bayes probability of criminal activity as a function of time for an account known to not contain fraud. (a) Based on the HMM discussed in Example 3. (b) A more elaborate model considering the accounts hourly and daily calling patterns. Figure 13(b) more closely matches the intended behavior of the model. Time is measured in days since January 1, 1994. Figure 13(a) is shown on a restricted time scale to prevent overplotting.

generally useful diagnostics are the autocorrelation function (advocated by MacDonald and Zucchini, 1997) and the predictive distribution of the data (advocated by Robert *et al.*, 1999). The ACF can be used to determine whether more structure must be added to Figure 1. For example, the model we fit to the U.S. GNP data in Sections 2 and 3 ignores possible dependence in the residuals $(d_t - \mu_{h_t})$, which are functions of θ. Figure 14(a) shows the posterior distribution of the autocorrelation function for the residuals. All marginal distributions of the ACF easily cover zero, suggesting that dependence in the residuals is weak enough to be safely ignored.

The predictive distribution of the data, $p(d|d_1^n)$ is often used as a diagnostic for finite mixture models. HMM's have a more complicated predictive distribution because of the added time dimension. As with finite mixture models, the marginal distribution of d_t may be obtained by weighting $P_0, \dots, P_{S-1}$ according to the stationary distribution of $\mathbf{Q}$. More complicated summaries of $p(d|d_1^n)$ may be evaluated using posterior predictive checks (Rubin, 1984; Meng, 1994; Gelman *et al.*, 1995). That is, for each $\theta^{(j)}$ drawn by the MCMC algorithm, simulate a data set $d^{(j)}$ from $p(d_1^n|\theta^{(j)})$ and compute the summary for $d^{(j)}$. For example, the fetal lamb data ends with a run of zeros much longer than other such runs in the data set. Figure 14(b) calculates the posterior



(a) (b)

Figure 14: (a) Posterior distribution of the autocorrelation function for the residuals from the two parameter Gaussian model applied to the U.S. GNP data from Figure 4. (b) Posterior predictive distribution for the longest run of zeros for the two (solid) and three (dotted) state Poisson models applied to the fetal lamb data. The vertical line indicates the longest run of zeros in the data.

predictive distribution of the longest run of zeros for the two and three state Poisson models. The three state model is much more capable of producing long runs of zeros than the two state model.

6 Conclusion

MCMC allows hidden Markov models to be implemented without using recursive computing, but the likelihood, forward-backward, and Viterbi recursions bring a richness to the models that would not otherwise exist. The forward-backward recursions lead to a Gibbs sampler that mixes faster than its natural competitor, and the likelihood recursion opens the door to more general samplers that would be impossible without a tractable method for computing HMM likelihoods. The non-stochastic backward recursion allows Bayes estimates of **h** to be Rao-Blackwellized and empirical Bayes estimates of **h** to be used when MCMC is not an option. The Viterbi algorithm allows approximate MAP reconstructions of **h** to be constructed much more efficiently than by naive MCMC sampling. The likelihood recursion is indispensable for classical Bayesian model selection, either by MCMC or by approximate methods, and it allows the use of variable dimension Monte Carlo techniques that depend on likelihood. Likelihood also provides an automatically computed summary of model parameters that can be monitored to deduce MCMC convergence. Finally

Section 2.3 illustrates that the ability to draw **h** directly from $p(\mathbf{h}|d_1^n, \theta)$ can simplify the task of drawing missing data beyond the hidden Markov chain.

A second theme of the article, which has nothing to do with recursive computing, is that there are advantages to understanding the physical meaning of **h**. Selecting appropriate diagnostics and preventing label switching and are both simplified through scientific understanding of **h**. Informative prior assumptions are needed to defeat the intrinsic identifiability problems that cause label switching. The assumptions may be applied directly through the prior distribution, or indirectly by reparameterizing the model or choosing loss functions for estimation after an unidentified MCMC run. Intuition about how **h** should behave allows the posterior distribution of **h** to be used as a model diagnostic.

We conclude with a note about the historical significance of the paper by Baum *et al.* (1970). The paper contains two innovations developed to implement HMM's: an inequality and a recursion. Both foreshadowed much more general techniques. Dempster *et al.* (1977) showed that the inequality could be applied to a much broader set of problems when they introduced the general EM algorithm. The recursion is a special case of a local computation algorithm which has helped drive the recent success of graphical models, a promising area of current research. The computing landscape has changed considerably since Baum *et al.* (1970), but the ideas therein remain relevant in today's MCMC world.

References

Albert, J. H. and Chib, S. (1993). Bayes inference via Gibbs sampling of autoregressive time series subject to Markov mean and variance shifts. *Journal of Business and Economic Statistics* **11**, 1–15.

Andrieu, C. and Doucet, A. (2000). Joint bayesian model selection and estimation of noisy sinusoids via reversible jump mcmc. *MCMC Preprint Service* .

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics* **41**, 164–171.

Besag, J. and Green, P. (1993). Spatial statistics and Bayesian computation. *Journal of the Royal Statistical Society, Series B, Methodological* **55**, 1, 25–37.

Casella, G. and Robert, C. P. (1996). Rao-Blackwellisation of sampling schemes. *Biometrika* **83**, 81–94.

Celeux, G., Hurn, M., and Robert, C. P. (2000). Computational and inferential difficulties with mixture prior distributions. *Journal of the American Statistical Association* **95**, 451, 957–970.

Chib, S. (1996). Calculating posterior distributions and modal estimates in Markov mixture models. *Journal of Econometrics* **75**, 79–97.

Churchill, G. A. (1989). Stochastic models for heterogeneous dna sequences. *Bulletin of Mathematical Biology* **51**, 1, 79–94.

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

Cowles, M. K. and Carlin, B. P. (1996). Markov chain Monte Carlo convergence diagnostics: A comparative review. *Journal of the American Statistical Association* **91**, 883–904.

Damian, P., Wakefield, J., and Walker, S. (1999). Gibbs sampling for Bayesian non-conjugate and hierarchical models by using auxiliary variables. *Journal of the Royal Statistical Society, Series B, Methodological* **61**, 331–344.

Dawid, A. P. (1992). Applications of a general propagation algorithm for probabilistic expert systems. *Statistics and Computing* **2**, 25–36.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm (C/R: p22-37). *Journal of the Royal Statistical Society, Series B, Methodological* **39**, 1–22.

Diebolt, J. and Robert, C. P. (1994). Estimation of finite mixture distributions through Bayesian sampling. *Journal of the Royal Statistical Society, Series B, Methodological* **56**, 363–375.

Everitt, B. S. and Hand, D. J. (1981). *Finite Mixture Distributions*. Chapman & Hall.

Fredkin, D. R. and Rice, J. A. (1992). Bayesian restoration of single-channel patch clamp recordings. *Biometrics* **48**, 427–448.

Gelfand, A. E. and Smith, A. F. M. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association* **85**, 398–409.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (1995). *Bayesian Data Analysis.* Chapman & Hall.

Gilks, W. R. and Roberts, G. O. (1996). Strategies for improving MCMC. In W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, eds., *Markov chain Monte Carlo in Practice*, chap. 6, 89–114. Chapman & Hall.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* **82**, 711–732.

Grenander, U. and Miller, M. I. (1994). Representations of knowledge in complex systems (disc: P581-603). *Journal of the Royal Statistical Society, Series B, Methodological* **56**, 549–581.

Guttorp, P. (1995). *Stochastic Modeling of Scientific Data.* Chapman & Hall.

Hamilton, J. D. (1989). A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica* **57**, 2, 357–384.

Hamilton, J. D. (1990). Analysis of time series subject to changes in regime. *Journal of Econometrics* **45**, 39–70.

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

Higdon, D. M. (1998). Auxiliary variable methods for Markov chain Monte Carlo with applications. *Journal of the American Statistical Association* **93**, 422, 585–595.

Juang, B. H. and Rabiner, L. R. (1991). Hidden Markov models for speech recognition. *Technometrics* **33**, 251–272.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Trans. ASME. J. Basic Eng.* **83D**, 35–45.

Kass, R. E. and Raftery, A. E. (1995). Bayes factors. *Journal of the American Statistical Association* **90**, 773–795.

Leroux, B. G. and Puterman, M. L. (1992). Maximum-penalized-likelihood estimation for independent and Markov-dependent mixture models. *Biometrics* **48**, 545–558.

Liu, J. S. and Chen, R. (1998). Sequential monte carlo methods for dynamic systems. *Journal of the American Statistical Association* **93**, 443, 1032–1044.

Liu, J. S., Neuwald, A. F., and Lawrence, C. E. (1999). Markovian structures in biological sequence alignments. *Journal of the American Statistical Association* **94**, 1–15.

Liu, J. S., Wong, W. H., and Kong, A. (1994). Covariance structure of the Gibbs sampler with applications to the comparisons of estimators and augmentation schemes. *Biometrika* **81**, 27–40.

Liu, J. S., Wong, W. H., and Kong, A. (1995). Covariance structure and convergence rate of the Gibbs sampler with various scans. *Journal of the Royal Statistical Society, Series B, Methodological* **57**, 157–169.

MacDonald, I. L. and Zucchini, W. (1997). *Hidden Markov and other models for discrete-valued time series*. Chapman & Hall.

Meng, X.-L. (1994). Posterior predictive p-values. *The Annals of Statistics* **22**, 1142–1160.

Mengersen, K. L., Robert, C. P., and Guihenneuc-Jouyaux, C. (1999). MCMC convergence diagnostics: a reviewww. In J. Berger, J. Bernardo, A. Dawid, D. Lindley, and A. Smith, eds., *Bayesian Statistics 6*, 415–440. Oxford University Press.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *Journal of Chemical Physics* **21**, 1087–1092.

Phillips, D. B. and Smith, A. F. (1996). Bayesian model comparison via jump diffusions. In W. R. Gilks, S. Richardson, and D. J. Speigelhalter, eds., *Markov Chain Monte Carlo in Practice*, 215–239. Chapman & Hall.

Richardson, S. and Green, P. (1997). On Bayesian analysis of mixtures with an unknown number of components. *Journal of the Royal Statistical Society, Series B, Methodological* **59**, 4, 731–792. with discussion.

Robert, C. P. (1998). MCMC specificities of latent variable models. In *CompStat '98*, Bristol, England.

Robert, C. P., Celeux, G., and Diebolt, J. (1993). Bayesian estimation of hidden Markov chains: A stochastic implementation. *Statistics & Probability Letters* **16**, 77–83.

Robert, C. P., Rydén, T., and Titterington, D. M. (1999). Convergence controls for MCMC algorithms, with applications to hidden Markov chains. *Journal of Statistical Computation and Simulation* **64**, 327–355.

Robert, C. P., Rydén, T., and Titterington, D. M. (2000). Bayesian inference in hidden Markov models through the reversible jump Markov chain Monte Carlo method. *Journal of the Royal Statistical Society, Series B, Methodological* **62**, 1, 57–76.

Robert, C. P. and Titterington, D. M. (1998). Reparametrisation strategies for hidden Markov models and Bayesian approaches to maximum likelihood estimation. *Statistical Computing* .

Romberg, J. K., Choi, H., and Baraniuk, R. G. (2000). Bayesian tree-structured image modeling using wavelet-domain hidden Markov models. *IEEE Transactions on Image Processing* (submitted).

Rubin, D. B. (1984). Bayesianly justifiable and relevant frequency calculations for the applied statistician. *The Annals of Statistics* **12**, 1151–1172.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics* **6**, 2, 461–464.

Scott, S. L. (1999). Bayesian analysis of a two state Markov modulated Poisson process. *Journal of Computational and Graphical Statistics* **8**, 3, 662–670.

Scott, S. L. (2000). Detecting network intrusion using the Markov modulated nonhomogeneous Poisson process. *Journal of the American Statistical Association* (submitted).

Stephens, M. (1997). Discussion of the paper by Richardson and Green. *Journal of the Royal Statistical Society, Series B, Methodological* 768–769.

Stephens, M. (2000a). Bayesian analysis of mixtures with an unknwon number of components– an alternative to reversible jump methods. *Annals of Statistics* to appear.

Stephens, M. (2000b). Dealing with label switching in mixture models. *Journal of the Royal Statistical Society, Series B, Methodological* **62**, 795–810.

Tierney, L. and Kadane, J. B. (1986). Accurate approximations for posterior moments and marginal densities. *Journal of the American Statistical Association* **81**, 393, 82–86.

Titterington, D. M., Smith, A. F. M., and Makov, U. E. (1985). *Statistical Analysis of Finite Mixture Distributions.* Wiley.

Viterbi, A. J. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory* **13**, 2, 260–269.

West, M. and Harrison, J. (1997). *Bayesian Forecasting and Dynamic Models.* Springer.