Biometrika (1994), **81**, 3, *pp.* 541–53
Printed in Great Britain

On Gibbs sampling for state space models

BY C. K. CARTER AND R. KOHN
Australian Graduate School of Management, University of New South Wales, PO Box 1, Kensington, N.S.W., Australia, 2033

SUMMARY

We show how to use the Gibbs sampler to carry out Bayesian inference on a linear state space model with errors that are a mixture of normals and coefficients that can switch over time. Our approach simultaneously generates the whole of the state vector given the mixture and coefficient indicator variables and simultaneously generates all the indicator variables conditional on the state vectors. The states are generated efficiently using the Kalman filter. We illustrate our approach by several examples and empirically compare its performance to another Gibbs sampler where the states are generated one at a time. The empirical results suggest that our approach is both practical to implement and dominates the Gibbs sampler that generates the states one at a time.

Some key words: Diffuse parameter; Kalman filter; Markov chain Monte Carlo; Mixture of normals; Spline smoothing; Switching regression; Trend plus seasonal model.

1. INTRODUCTION

Consider the linear state space model

$$y(t) = h(t)'x(t) + e(t), \tag{1·1}$$

$$x(t+1) = F(t+1)x(t) + u(t+1), \tag{1·2}$$

where $y(t)$ is a scalar observation and $x(t)$ is an $m \times 1$ state vector. We assume that the error sequences $\{e(t), t \geqslant 1\}$ and $\{u(t), t \geqslant 1\}$ are mixtures of normals. Let θ be a parameter vector whose value determines $h(t)$ and $F(t)$ and also the distributions of $e(t)$ and $u(t)$. Further details of the structure of the model are given in § 2·1. Equation (1·1) is called the observation equation and (1·2) the state transition equation. When $e(t)$ and $u(t)$ are independent Gaussian sequences unknown parameters are usually estimated by maximum likelihood following Schweppe (1965). The Kalman filter and state space smoothing algorithms are used to carry out the computations.

There are a number of applications in the literature where it is necessary to go beyond the Gaussian linear state space model: e.g. Harrison & Stevens (1976), Gordon & Smith (1990), Hamilton (1989) and Shumway & Stoffer (1991). Meinhold & Singpurwalla (1989) robustify the Kalman filter by taking both $e(t)$ and $u(t)$ to be t distributed. A general approach to estimating non-Gaussian and nonlinear state space models is given by Kitagawa (1987). Except when the dimension of the state vector is very small, Kitagawa's approach appears computationally intractable at this stage. Various approximate filtering and smoothing algorithms for nonlinear and non-Gaussian state space models have been given in the literature. See, for example, Anderson & Moore (1979, Ch. 8) and West & Harrison (1989).

Downloaded from https://academic.oup.com/biomet/article-abstract/81/3/541/257082 by University of California, San Diego user on 20 April 2019

Using the Gibbs sampler, Carlin, Polson & Stoffer (1992) provide a general approach to Bayesian statistical inference in state space models allowing the errors $e(t)$ and $u(t)$ to be non-Gaussian and the dependence on $x(t)$ in (1·1) and (1·2) to be nonlinear. They generate the states one at a time utilizing the Markov properties of the state space model to condition on neighbouring states. In this paper we take a different Gibbs sampling approach, generating all the states at once by taking advantage of the time ordering of the state space model. We show how to carry out all the necessary computations using standard Gaussian filtering and smoothing algorithms. Although our approach is less general than that of Carlin et al. (1992), for the class of models considered in this paper our approach will be more efficient than theirs, in the sense that convergence to the posterior distribution will be faster and estimates of the posterior moments will have smaller variances. To quantify the difference between our approach and that of Carlin et al. (1992), we study empirically the performance of both algorithms for two simple and commonly-used trend and seasonal models. For both examples generating the states simultaneously produces Gibbs iterates which converge rapidly to the posterior distribution from arbitrary starting points. In contrast, when the states are generated one at a time there was slow convergence to the posterior distribution for one of the examples and the estimates of the posterior means were far less efficient than the corresponding estimates when generating the states simultaneously. In the second example there is no convergence to the posterior distribution when the states were generated one at a time because the resulting Markov chain is reducible. Our approach is supported theoretically by the results of Liu, Wong & Kong (1994) who show that when measured in some norm generating variables simultaneously produces faster convergence than generating them one at a time.

Section 2 discusses Gibbs sampling and how to generate the states and the indicator variables. Section 3 illustrates the general theory with four examples and empirically compares the performance of our algorithm with that of generating the states one at a time. Appendix 1 shows how to generate the state vector using a state space filtering algorithm and Appendix 2 shows how to generate the indicator variables.

2. The Gibbs sampler

2·1. *General*

Let $Y^n = \{y(1), \ldots, y(n)\}'$ be the vector of observations and $X = \{x(1)', \ldots, x(n)'\}'$ the total state vector. Let $K(t)$ be a vector of indicator variables showing which members of the mixture each of $e(t)$ and $u(t)$ belong to and which values $h(t)$ and $F(t)$ take, and let $K = \{K(1), \ldots, K(n)\}'$. We write the parameter vector $\theta = \{\theta_1, \ldots, \theta_p\}$. We assume that, conditional on K and θ, $e(t)$ and $u(t)$ are independent Gaussian sequences which are also independent of each other. To illustrate our notation we consider the following simple example. Let

$$y(t) = x(t) + e(t), \quad x(t) = x(t-1) + u(t),$$

with $x(t)$ univariate. The errors $e(t)$ are a mixture of two normals with $e(t) \sim N(0, \sigma^2)$ with probability p_1 and $e(t) \sim N(0, C\sigma^2)$ with probability $1 - p_1$, where $C > 1$ and p_1 are assumed known. The disturbance $u(t) \sim N(0, \tau^2)$. Then $\theta = (\sigma^2, \tau^2)$ is the unknown parameter vector. We define the indicator variable $K(t)$ as $K(t) = 0$ if var $\{e(t)\} = \sigma^2$ and $K(t) = 1$ if var $\{e(t)\} = C\sigma^2$.

Let $p(X, K, \theta | Y^n)$ be the joint posterior density of X, K and θ. The Gibbs sampler (Gelfand & Smith, 1990) is an iterative Monte Carlo technique that, in our case, success-

Downloaded from https://academic.oup.com/biomet/article-abstract/81/3/541/257082 by University of California, San Diego user on 20 April 2019

ively generates X, K and θ from the conditional densities $p(X|Y^n, K, \theta)$, $p(K|Y^n, X, \theta)$ and $p(\theta_i|Y^n, X, K, \theta_{j\neq i})$ for $i = 1, \ldots, p$ until eventually (X, K, θ) is generated from the joint posterior distribution $p(X, K, \theta|Y^n)$. Tierney (1994) proves the convergence of the Gibbs sampler under appropriate regularity conditions. For any given example it is usually straightforward to check whether these conditions hold.

We will assume that θ_i can be generated from $p(\theta_i|Y^n, X, K, \theta_{j\neq i})$ for $i = 1, \ldots, p$. Efficient ways of doing so will be determined on a case by case basis. Sections 2·2 and 2·3 show how to generate from $p(X|Y^n, K, \theta)$ and $p(K|Y^n, X, \theta)$.

2·2. *Generating the state vector*

We assume that $x(1)$ has a proper distribution and that conditional on K and the parameter vector θ, $h(t)$ and $F(t)$ are known, and $e(t)$ and $u(t)$ are Gaussian with known means and variances. For notational convenience we usually omit dependence on K and θ in this section. For $t = 1, \ldots, n$ let Y^t consist of all $y(j)$ $(j \leqslant t)$. The following lemma shows how to generate the whole of X given Y^n, K and θ. Its proof is straightforward and is omitted.

LEMMA 2·1. *We have*

$$p(X|Y^n) = p\{x(n)|Y^n\} \prod_{t=1}^{n-1} p\{x(t)|Y^t, x(t+1)\}.$$

Thus to generate X from $p(X|Y^n)$ we first generate $x(n)$ from $p\{x(n)|Y^n\}$ and then for $t = n-1, \ldots, 1$ we generate $x(t)$ from $p\{x(t)|Y^t, x(t+1)\}$. Because $p\{x(n)|Y^n\}$ and $p\{x(t)|Y^t, x(t+1)\}$ are Gaussian densities, in order to generate all the $x(t)$ we need to compute $E\{x(n)|Y^n\}$ and $\text{var}\{x(n)|Y^n\}$ and

$$E\{x(t)|Y^t, x(t+1)\}, \quad \text{var}\{x(t)|Y^t, x(t+1)\} \quad (t = n-1, \ldots, 1).$$

Let $x(t|j) = E\{x(t)|Y^j\}$ and $S(t|j) = \text{var}\{x(t)|Y^j\}$. We obtain $x(t|t)$ and $S(t|t)$ for $t = 1, \ldots, n$ using the Kalman filter (Anderson & Moore, 1979, p. 105). To obtain $E\{x(t)|Y^t, x(t+1)\}$ and $\text{var}\{x(t)|Y^t, x(t+1)\}$ we treat the equation

$$x(t+1) = F(t+1)x(t) + u(t+1)$$

as m additional observations on the state vector $x(t)$ and apply the Kalman filter to them. Details are given in Appendix 1.

In many applications the distribution of the initial state vector $x(1)$ is partly unknown and this part is usually taken as a constant to be estimated or equivalently to have a diffuse distribution making $x(1)$ partially diffuse. By this we mean that $x(1) \sim N(0, S_0^{[0]} + kS_0^{[1]})$ with $k \to \infty$. The generation algorithm can be applied as outlined above and in Appendix 1, except that now we use the modified filtering and smoothing algorithms of Ansley & Kohn (1990).

Remark. In specific models it may be possible to use a faster filtering algorithm than the Kalman filter to obtain $x(t|t)$ and $S(t|t)$. See, for example, the fast filtering algorithms of Anderson & Moore (1979, Ch. 6) when $e(t)$ and $u(t)$ are Gaussian and $F(t)$ and $h(t)$ are constant. A referee has suggested the use of a Metropolis step within the Gibbs sampler to speed up the generation of the states (Tierney, 1994). To do so it is necessary to find a candidate density $q(X|K, \theta)$ for generating X which is faster to generate from than $p(X|Y, K, \theta)$ and yet is close enough to it so the rejection rate in the Metropolis step is

Downloaded from https://academic.oup.com/biomet/article-abstract/81/3/541/257082 by University of California, San Diego user on 20 April 2019

not too high. We tried using the prior for X as $q(X|K, \theta)$, but this resulted in huge rejection rates and was therefore not practical.

2·3. *Generating the indicator variables*

Recall that $K(t)$ $(t = 1, \ldots, n)$ is a vector of indicator variables showing which members of the mixture each of $e(t)$ and $u(t)$ belong to and which values $h(t)$ and $F(t)$ take. Let $K^t = \{K(1), \ldots, K(t)\}$ and $X^t = \{x(1), \ldots, x(t)\}$. For notational convenience we omit dependence on θ. Conditionally on K and θ, $e(t)$ and $u(t)$ are independent for $t = 1, \ldots, n$ in (1·1) and (1·2). This implies that

$$p\{y(t)|Y^{t-1}, X^t, K^t\} = p\{y(t)|x(t), K(t)\}, \quad p\{x(t)|X^{t-1}, K^t\} = p\{x(t)|x(t-1), K(t)\}.$$

We assume that the prior distribution of K is Markov. The next lemma shows how to generate the whole of K given Y^n, X and θ. We omit its proof as it is straightforward.

LEMMA 2·2. *We have*

$$p(K|Y^n, X) = p\{K(n)|Y^n, X\} \prod_{t=1}^{n-1} p\{K(t)|Y^t, X^t, K(t+1)\}.$$

Thus to generate K from $p(K|Y^n, X)$ we first generate $K(n)$ from $p\{K(n)|Y^n, X\}$ and then for $t = n-1, \ldots, 1$ we generate $K(t)$ from $p\{K(t)|Y^t, X^t, K(t+1)\}$. Because $p\{K(n)|Y^n, X\}$ and $p\{K(t)|Y^t, X^t, K(t+1)\}$ are discrete valued we can generate from them easily, once we have calculated them. To calculate $p\{K(n)|Y^n, X\}$ and $p\{K(t)|Y^t, X^t, K(t+1)\}$ we use recursive filtering equations (Anderson & Moore, 1979, Ch. 8) in a similar way to our use of the Kalman filter in § 2·2. Details are in Appendix 2. Because $K(t)$ is discrete valued, the filtering equations can be evaluated efficiently.

3. EXAMPLES

3·1. *General*

We illustrate the results in § 2 and Appendices 1 and 2 by applying them to four examples. The first is a stochastic trend model giving a cubic spline smoothing estimate of the signal. The second example is a trend plus seasonal model. In the third example the errors $e(t)$ are a discrete mixture of normals with Markov dependence. The fourth example discusses switching regression. The first two examples compare empirically the performance of the approach that generates all the states simultaneously with the approach that generates the states one at a time.

3·2. *Example* 1: *Cubic smoothing spline*

Our first example is a continuous time stochastic trend model for which the signal estimate is a cubic smoothing spline. We implement the Gibbs sampler using the first element of the state vector and make the important point that in many applications only a subset of the elements of the state vector is needed.

Suppose we have observations on the signal plus noise model

$$y(i) = g(t_i) + e(i) \quad (i = 1, \ldots, n), \tag{3·1}$$

with the $e(i)$ independent $N(0, \sigma^2)$ with the signal $g(t)$ generated by the stochastic differential equation

$$d^2g(t)/dt^2 = \tau dW(t)/dt; \tag{3·2}$$

Downloaded from https://academic.oup.com/biomet/article-abstract/81/3/541/257082 by University of California, San Diego user on 20 April 2019

$W(t)$ is a Wiener process with $W(0)=0$ and var $\{W(t)\}=1$ and τ is a scale parameter. We assume that the initial conditions on $g(t)$ and $dg(t)/dt$ are diffuse; that is, with $k\to\infty$,

$$\{g(t_1), dg(t_1)/dt\}' \sim N(0, kI_2). \tag{3·3}$$

We take $0\leqslant t_1<t_2<\ldots<t_n$. Following Kohn & Ansley (1987) we can write (3·1) and (3·2) in state space form as

$$y(i)=h'x(t_i)+e(i), \quad x(t_i)=F(\delta_i)x(t_{i-1})+u(i),$$

where the state vector $x(t)=\{g(t), dg(t)/dt\}'$, the increments $\delta_i=t_i-t_{i-1}$ ($t_0=0$), and the $u(i)$ are independent $N\{0, \tau^2U(\delta_i)\}$. The vector $h=(1,0)'$ and the 2×2 matrices $F(\delta)$ and $U(\delta)$ are given by

$$F(\delta)=\begin{pmatrix}1 & \delta\\ 0 & 1\end{pmatrix}, \quad U(\delta)=\begin{pmatrix}\delta^3/3 & \delta^2/2\\ \delta^2/2 & \delta\end{pmatrix}.$$

The vector of unknown parameters is $\theta=(\sigma^2, \tau^2)'$ and from (3·3) the initial state vector $x(t_1)$ has a diffuse distribution. For a further discussion of this model and its connection with spline smoothing see Wahba (1983) and Kohn & Ansley (1987).

To complete the Bayesian specification of the model we impose the improper priors $p(\sigma^2)\propto 1/\sigma^2\exp(-\beta_\sigma/\sigma^2)$, with β_σ small, and $p(\tau^2)\propto 1$. As we only use the first element of $x(t)$ in the Gibbs sampler let $G=\{g(t_1),\ldots,g(t_n)\}$. The vectors G and θ are generated as follows. For given θ, X is generated as explained in § 2·2 and Appendix 1. We then extract G. To generate σ^2, we can show that

$$p(\sigma^2|Y^n, G, \tau^2)\propto(\sigma^2)^{-n/2-1}\times\exp\left[-\frac{1}{\sigma^2}\left\{\frac{1}{2}\sum_{i=1}^{n}e(i)^2+\beta_\sigma\right\}\right],$$

where $e(i)=y(i)-h'x(t_i)$. Hence σ^2 is generated from an inverse gamma distribution with parameters $n/2$ and $\frac{1}{2}\sum e(i)^2+\beta_\sigma$. To generate τ^2, note that for given $k>0$

$$p(\tau^2|Y^n, G, \sigma^2; k)=p(\tau^2|G; k)\propto p(G|\tau^2; k)p(\tau^2)\propto p(G|\tau^2; k).$$

It follows from Ansley & Kohn (1985) that

$$\lim_{k\to\infty}p(\tau^2|G; k)\propto(\tau^2)^{-n/2+1}\times\exp\left\{-\frac{1}{2\tau^2}\sum_{i=3}^{n}\varepsilon(i)^2/R(i)\right\},$$

where $\varepsilon(i)$ and $R(i)$ are the innovations and innovation variances respectively obtained from running the modified Kalman filter on the state space model

$$g(t_i)=h'x(t_i), \quad x(t_i)=F(\delta_i)x(t_{i-1})+u(i).$$

Hence τ^2 is generated from an inverse gamma distribution with parameters $n/2-2$ and $\frac{1}{2}\sum\varepsilon(i)^2/R(i)$.

For this model we now describe the Gibbs sampler approach of Carlin et al. (1992). The state vector $x(t)$ is generated from $p\{x(t)|y(t), x(t-1), x(t+1), \sigma^2, \tau^2\}$. The error variance σ^2 is generated as above by noting that $p(\sigma^2|Y^n, X, \tau^2)=p(\sigma^2|Y^n, G, \tau^2)$. To generate τ^2 note that for given $k>0$

$$p(\tau^2|Y^n, X, \sigma^2; k)=p(\tau^2|X; k)\propto p(X|\tau^2; k)p(\tau^2)\propto p(X|\tau^2; k).$$

It follows from Ansley & Kohn (1985) that

$$\lim_{k\to\infty}p(\tau^2|X; k)\propto(\tau^2)^{-n+1}\times\exp\left\{-\frac{1}{2\tau^2}\sum_{i=2}^{n}u(i)'U(\delta_i)^{-1}u(i)\right\},$$

Downloaded from https://academic.oup.com/biomet/article-abstract/81/3/541/257082 by University of California, San Diego user on 20 April 2019

where $u(i) = x(t_i) - F(\delta_i)x(t_{i-1})$. Hence we generate τ^2 given Y^n, X and σ^2 from an inverse gamma distribution with parameters $n-2$ and $\frac{1}{2}\sum u(i)'U(\delta_i)^{-1}u(i)$.

We now compare empirically the approach generating all the states at once to the approach that generates the states one at a time. The data are generated by (3·1) with the function

$$g(t) = \tfrac{1}{3}\beta_{10,5}(t) + \tfrac{1}{3}\beta_{\tau,\tau}(t) + \tfrac{1}{3}\beta_{5,10}(t),$$

where $\beta_{p,q}$ is a beta density with parameters p and q, and $0 \leqslant t \leqslant 1$. This function was used by Wahba (1983) in her simulations. The error standard deviation is $\sigma = 0{\cdot}2$, the sample size is $n = 50$ and the design is equally spaced with $t_i = i/50$ $(i = 1, \ldots, 50)$. For both algorithms we first ran the Gibbs sampler with the starting values

$$(\sigma^2)^{[0]} = 1, \quad x(t)^{[0]} = E\{x(t) | \sigma^2 = 1, \tau^2 = 1\}.$$

The value of τ^2 was generated by the Gibbs sampler. Figure 1(a) is a plot of the iterates of σ^2 and Fig. 1(b) a plot of the iterates of τ^2 when the states are generated one at a time. The horizontal axis is the iterate number. It appears that for this approach the Gibbs sampler takes about 15 000 iterations to converge. Figures 1(c) and (d) are plots of the first 2000 iterates of the Gibbs sampler of σ^2 and τ^2 respectively when the states are generated simultaneously. The same starting values are used for both approaches. The Gibbs sampler appears to converge after 100 iterations. Similar results are obtained for other arbitrary starting values.

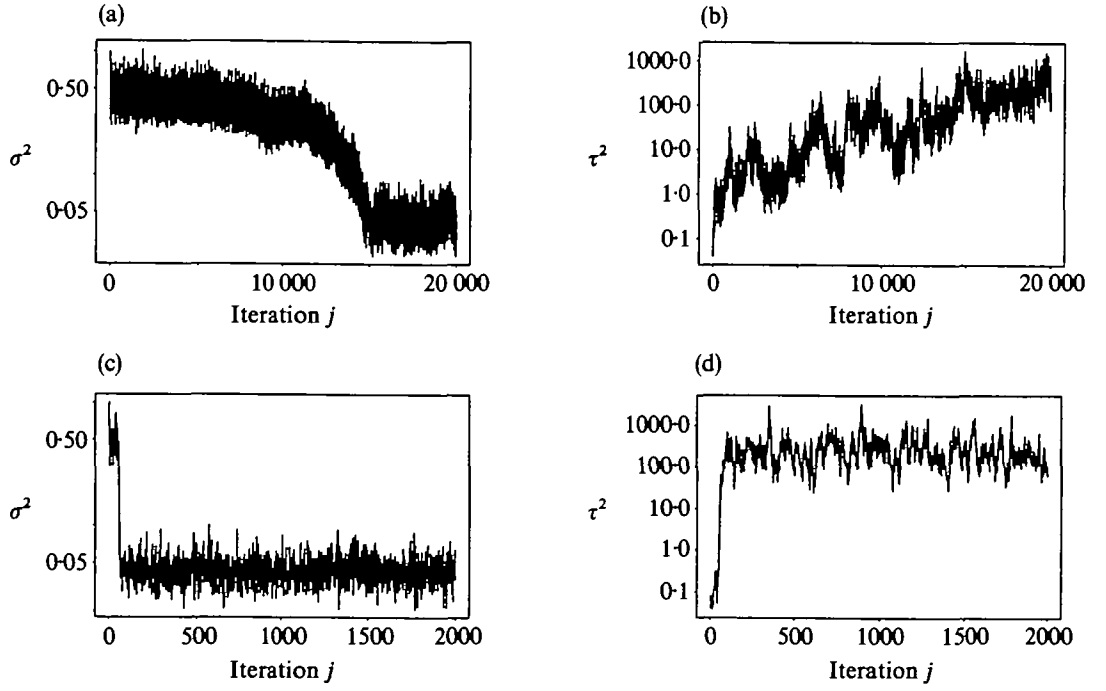

Fig. 1. Example 1: generated values of σ^2 and τ^2 with starting values $\sigma^2 = 1$ and $X = E(X | \sigma^2 = 1, \tau^2 = 1)$. In (a) and (b) the states are generated one at a time and in (c) and (d) they are generated simultaneously.

To study the relative efficiencies of the two algorithms once the Gibbs sampler has converged we use the marginal likelihood estimates of σ^2 and τ^2 as starting values. For a

Downloaded from https://academic.oup.com/biomet/article-abstract/81/3/541/257082 by University of California, San Diego user on 20 April 2019

definition and discussion of the marginal likelihood estimates see Kohn & Ansley (1987). For both algorithms we ran the Gibbs sampler for a warm-up period of 1000 iterations followed by a sampling run of 10 000 iterates. Using the final 10 000 iterates we computed the first 300 autocorrelations of the signal estimate at the abscissa $t = 0{\cdot}4$, which we call $g(0{\cdot}4)$, and also for τ^2. Figures 2(a) and (b) are plots of the autocorrelations of the iterates of $g(0{\cdot}25)$ and τ^2 respectively when the states are generated one at a time and Fig. 2(c) and (d) are the corresponding plots for the algorithm when the states are generated simultaneously. Clearly the autocorrelations for the first algorithm are much higher than for the second.

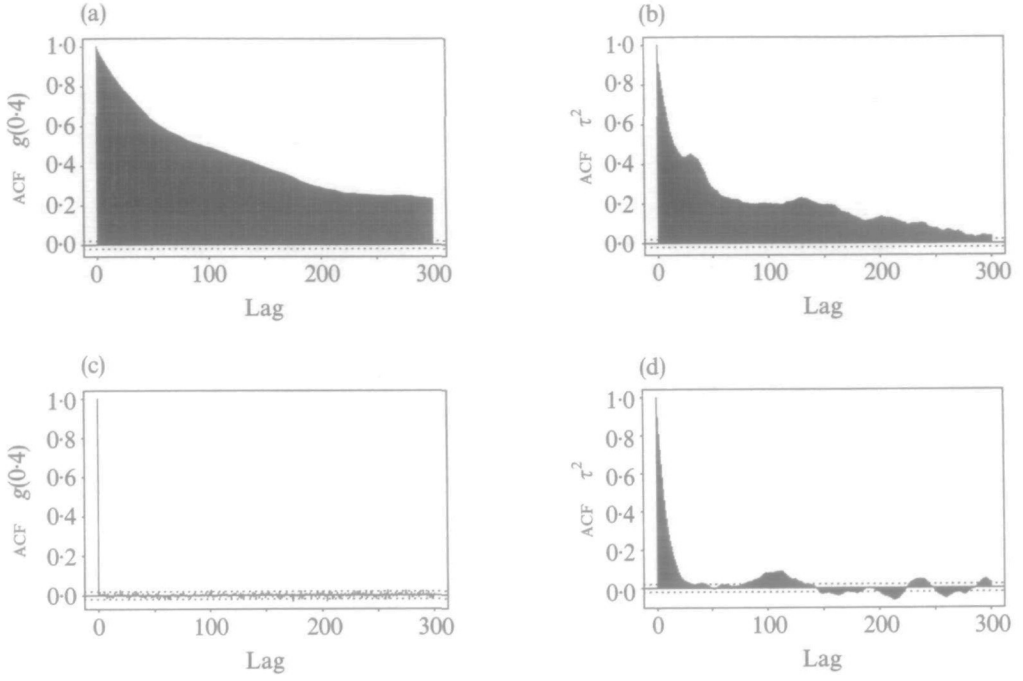

Fig. 2. Example 1: sample autocorrelation function (ACF) for $g(0{\cdot}25)$ and τ^2. In (a) and (b) the states are generated one at a time and in (c) and (d) they are generated simultaneously.

Using the sampling run of 10 000 iterates we now present the relative efficiencies of the two algorithms in estimating the posterior mean $E\{g(t)|Y\}$ of the signal. There are two ways to estimate the posterior mean. The first is to use the sample moments of the Gibbs iterates to form what is called a histogram estimate. The second way is to form a mixture estimate. When generating all the states simultaneously the histogram and mixture estimates of the posterior mean of $g(t)$ are respectively

$$\frac{1}{N}\sum_{j=1}^{N} g(t)^{[j]}, \quad \frac{1}{N}\sum_{j=1}^{N} E\{g(t)|Y, \theta^{[j]}\}, \tag{3·4}$$

where $N = 10\,000$ and $g^{[j]}(t)$ is the jth Gibbs iterate of $g(t)$ during the sampling period. The smoothed values $E\{g(t)|Y, \theta^{[j]}\}$ in (3·4) are obtained using the smoothing algorithm of Ansley & Kohn (1990). For the algorithm generating the states one at a time the histogram estimates are as in (3·4) while the mixture estimates are computed as in § 2 of Carlin et al. (1992). The results of Gelfand & Smith (1990) and Liu et al. (1994) suggest that mixture estimates will usually have smaller variance than histogram estimates. We

Downloaded from https://academic.oup.com/biomet/article-abstract/81/3/541/257082 by University of California, San Diego user on 20 April 2019

first consider the efficiency of the histogram estimates of the signal by estimating the posterior mean of the signal at the abscissae $t = 0{\cdot}02$, $0{\cdot}25$ and $0{\cdot}5$ and calling $\hat{g}(t)$ the estimate at t. We assume that in the sampling period the Gibbs sampler has converged so the $g(t)^{[J]}$ form a stationary sequence for each t. For a given t let $\gamma_{it} = \text{cov}\{g(t)^{[J]}, g(t)^{[J+i]}\}$ be the ith autocovariance of $g(t)^{[J]}$ with corresponding sample autocovariance $\hat{\gamma}_{it}$. We estimate N var $\{\hat{g}(t)\}$ by

$$\sum_{|i| \leqslant 1000} (1 - |i|/N)\hat{\gamma}_{it}$$

using the first 1000 sample autocovariances. For a discussion of variance estimation from simulation experiments see Moran (1975).

Table 1 presents the results for the histogram estimates. The first column gives the abscissa t, the second column the sample variance estimate $\hat{\gamma}_{0t}$, and the third column the variance estimate of N var $\{\hat{g}(t)\}$ when the states are generated simultaneously. The fourth and fifth columns have the same interpretation as the second and third columns except that now the states are generated one at a time. The sixth column is the ratio of the fifth and third columns and is an estimate of the factor by which the number of Gibbs iterates for the approach which generates one state at a time would have to increase in order to have the same accuracy as the approach which generates all the states at once. We take it to be the measure of the efficiency of the two algorithms. Table 1 shows that the efficiencies range from 91 to 358 so that the number of iterates of the algorithm that generates the states one at a time would need to increase by a factor of about 350 to achieve the same accuracy as that which generates the states simultaneously. We also note from the table that the sample variances $\hat{\gamma}_{0t}$ are approximately the same for both algorithms suggesting that we are generating from the correct distribution for the algorithm that generates the states one at a time.

Table 2 has the same interpretation as Table 1 but we now deal with the iterates generated by the mixture estimates. The efficiencies now range from 498 to 178 000.

Table 1. *Histogram estimates of* $E\{g(t)|Y^n\}$

	Simultaneous		One at a time		
t	$\hat{\gamma}_{0,t}$	N var $\{\hat{g}(t)\}$	$\hat{\gamma}_{0,t}$	N var $\{\hat{g}(t)\}$	Ratio
0·02	$2{\cdot}3 \times 10^{-2}$	$4{\cdot}4 \times 10^{-2}$	$2{\cdot}2 \times 10^{-2}$	4	91
0·25	$6{\cdot}8 \times 10^{-3}$	$7{\cdot}7 \times 10^{-2}$	$7{\cdot}6 \times 10^{-3}$	2·5	318
0·5	$7{\cdot}0 \times 10^{-3}$	$7{\cdot}0 \times 10^{-3}$	$9{\cdot}3 \times 10^{-3}$	2·5	358

Table 2. *Mixture estimates of* $E\{g(t)|Y^n\}$

	Simultaneous		One at a time		
t	$\hat{\gamma}_{0,t}$	N var $\{\hat{g}(t)\}$	$\hat{\gamma}_{0,t}$	N var $\{\hat{g}(t)\}$	Ratio
0·02	$2{\cdot}9 \times 10^{-3}$	$7{\cdot}8 \times 10^{-3}$	$2{\cdot}1 \times 10^{-2}$	3·9	498
0·25	$5{\cdot}8 \times 10^{-6}$	$1{\cdot}4 \times 10^{-5}$	$7{\cdot}5 \times 10^{-3}$	2·5	178731
0·5	$1{\cdot}1 \times 10^{-4}$	$1{\cdot}4 \times 10^{-4}$	$9{\cdot}2 \times 10^{-3}$	2·5	17455

We repeated this study with different functions $g(t)$, different sample sizes and different values of error standard deviation and obtained similar results to those reported above. We conclude that for this simple model the approach that generates the states one at a time is far slower to converge and is far less efficient than the approach that generates all the states simultaneously.

Downloaded from https://academic.oup.com/biomet/article-abstract/81/3/541/257082 by University of California, San Diego user on 20 April 2019

3·3. *Example 2: Trend plus seasonal components time series model*

A popular model for quarterly economic time series is

$$y(t) = g(t) + T(t) + e(t) \quad (t = 1, \ldots, n), \tag{3.5}$$

with the errors $e(t)$ independent $N(0, \sigma^2)$ and with the seasonal $g(t)$ and trend $T(t)$ generated by the stochastic difference equations

$$\sum_{j=0}^{3} g(t-j) = v(t), \tag{3·6}$$

$$T(t) - 2T(t-1) + T(t-2) = w(t), \tag{3·7}$$

where the $v(t)$ are independent $N(0, \tau^2)$ and the $w(t)$ are independent $N(0, \omega^2)$. This model is proposed by Kitagawa & Gersch (1984) who regard (3·6) and (3·7) as priors for the seasonal and trend and who express (3·5)–(3·7) in state space form with the state vector

$$x(t) = \{g(t), g(t-1), g(t-2), T(t), T(t-1)\}'.$$

For the approach generating the states simultaneously estimation of the model using the Gibbs sampler can be done as in § 3·2. For the approach generating the states one at a time the state vector $x(t)$ is known, for $1 < t < n$, if we condition on $x(t-1)$ and $x(t+1)$ and so new variation is only introduced when generating $x(1)$ and $x(n)$. Thus the resulting Gibbs sampler does not converge to the posterior distribution.

To study empirically the approach that generates the states simultaneously we consider for simplicity the pure seasonal model

$$y(t) = g(t) + e(t).$$

We generated 50 observations using $g(t) = \sin(2\pi t/4{\cdot}1)$ and $\sigma = 0{\cdot}2$. The priors for σ^2 and τ^2 are the same as in § 3·2. Figure 3 shows the output from the Gibbs sampler with the starting values $(\sigma^2)^{[0]} = 1$ and $x(t)^{[0]} = E\{x(t) | \sigma^2 = 1, \tau^2 = 1\}$. Figure 3(a) shows the data together with the function $g(t)$ and the mixture estimates of $g(t)$. Fig. 3(b) and (c) show the generated values of σ^2 and τ^2 respectively. The warm-up period was 1000 iterations and the sampling period was 1000 iterations. From Fig. 3 the algorithm to generate the states simultaneously appears to converge within a hundred iterations. Similar results were obtained for other abitrary starting values.

To understand the difference in performance of the two algorithms we view the state transition equation (1·2) as a prior for the state vector. If this prior is tight then generating the states one at a time will produce Gibbs iterates of the states which are highly dependent and so will tend to move very little in successive iterations. An extreme case is the seasonal plus trend model (3·5)–(3·7).

3·4. *Normal mixture errors with Markov dependence*

We illustrate our algorithm to generate the indicator variables by considering the case where $e(t)$ and $u(t)$ are normal mixtures. It is sufficient to discuss the case where $e(t)$ is a mixture of two normals and $u(t)$ is normal as the general case can be handled similarly. We assume that we have a linear state space model as given in (1·1) and (1·2), and that $e(t)$ has mean zero and variance equal to either σ^2 or $\kappa^2\sigma^2$. Such a normal mixture model for $e(t)$ with $\kappa > 1$ has been used by Box & Tiao (1968) to handle outliers. Let $K(t) = 1$ if $e(t)$ has variance σ^2, and let $K(t) = 2$ otherwise, and let $K = \{K(1), \ldots, K(n)\}'$.

We note that, conditionally on K, the $e(t)$ and $u(t)$ are Gaussian so that the results in

Downloaded from https://academic.oup.com/biomet/article-abstract/81/3/541/257082 by University of California, San Diego user on 20 April 2019

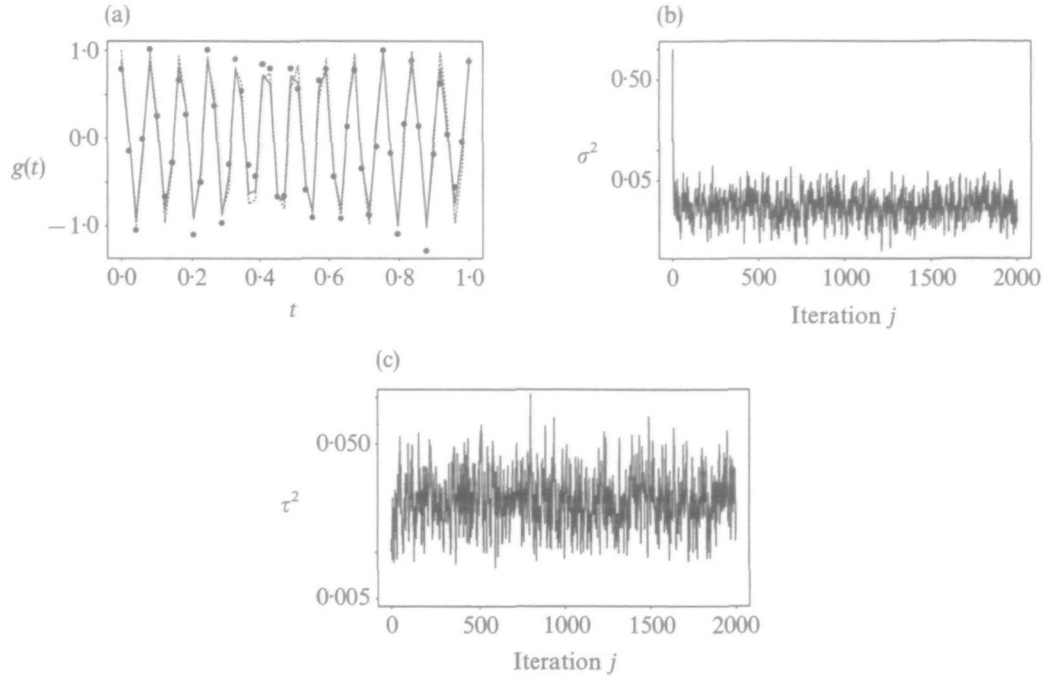

Fig. 3. Example 2: (a) shows the data (dots) together with the function $g(t)$ (dashes) and the mixture estimates of $g(t)$ (solid); (b) and (c) show the generated values of σ^2 and τ^2. The states are generated simultaneously.

§§ 2 and 3·2 apply to the generation of X and θ. Thus we will only consider the generation of K. We first assume that a priori the $K(t)$ come from a Markov chain with

$$p\{K(t+1)=2|K(t)=i\}=p_i \quad (i=1,2).$$

For simplicity we take the probabilities p_1 and p_2 as fixed, for example we could take $p_1=0{\cdot}05$ and $p_2=0{\cdot}5$ for detecting outliers that cause a temporary increase in variance. Alternatively we could place a prior on p_1 and p_2, for example a beta prior. To simplify notation we omit dependence on θ. From Lemma 2·2 the distribution of K given Y^n and X is

$$p(K|Y^n,X)=p\{K(n)|Y^n,X\}\prod_{t=1}^{n-1}p\{K(t)|Y^t,X^t,K(t+1)\}.$$

We show how to calculate each term in Appendix 2.

We now consider the simpler case where a priori the $K(t)$ are independent. In this case Lemma 2·2 becomes

$$p(K|Y^n,X)=\prod_{t=1}^{n}p\{K(t)|y(t),x(t)\}\propto\prod_{t=1}^{n}p\{y(t)|x(t),K(t)\}p\{K(t)\},$$

so that the $K(t)$ are independent and binomial and it is straightforward to generate them.

Our approach can also handle errors that are general normal scale mixtures, for example t distributed errors. Some further details and examples are given by Carlin et al. (1992).

Downloaded from https://academic.oup.com/biomet/article-abstract/81/3/541/257082 by University of California, San Diego user on 20 April 2019

3·5. *Switching regression model*

In the switching regression model the coefficients $\{h(t), F(t), t = 1, \ldots, n\}$ take on a small number of different values determined by some probabilistic distribution. Shumway & Stoffer (1991) use a switching regression model to identify targets when a large number of targets with unknown identities is observed. To show how our results apply it is sufficient to discuss the simplest case in which $F(t)$ is constant for all t and $h(t)$ takes on just two values, h_1 and h_2 say. Let $K(t) = 1$ if $h(t) = h_1$, and let $K(t) = 2$ otherwise. As in § 3·4 we assume that a priori the $K(t)$ come from a Markov chain with parameters p_1 and p_2. If p_1 and p_2 are unknown we would place a beta prior on them. Given $K = \{K(1), \ldots, K(n)\}$ we generate X as in § 2·1. Generating K given Y^n, X and θ is very similar to the way we generated it in § 3·4 and we omit details.

Acknowledgement

We would like to thank the Division of Mathematics and Statistics, CSIRO, and the Australian Research Council for partial support. We would also like to thank David Wong for help with the computations.

Appendix 1

Algorithm to generate state vector

We show how to generate X conditional on Y^n, K and θ. We omit dependence on K and θ, and, as in § 2·2, let

$$x(t|j) = E\{x(t)|Y^j\}, \quad S(t|j) = \text{var}\{x(t)|Y^j\}.$$

For $t = 1, \ldots, n$ the conditional mean $x(t|t)$ and the conditional variance $S(t|t)$ are obtained using the Kalman filter (Anderson & Moore, 1979, p. 105).

Using Lemma 2·1 we show how to generate $x(n), \ldots, x(1)$ in that order conditioning on Y^n. First, $p\{x(n)|Y^n\}$ is normal with mean $x(n|n)$ and variance $S(n|n)$. To generate $x(t)$ conditional on Y^t and $x(t+1)$ we note that we can regard the equation

$$x(t+1) = F(t+1)x(t) + u(t+1)$$

as m additional observations on $x(t)$. If $U(t+1)$ is diagonal then the $u_i(t+1)$ $(i = 1, \ldots, m)$ are independent and we can apply the observation update step of the Kalman filter m times as shown below to obtain

$$E\{x(t)|Y^t, x(t+1)\}, \quad \text{var}\{x(t)|Y^t, x(t+1)\}.$$

More generally we can factorize $U(t+1) = L(t+1)\Delta(t+1)L(t+1)'$ using the Cholesky decomposition with $L(t+1)$ a lower triangular matrix with ones on the diagonal and $\Delta(t+1)$ a diagonal matrix. Let

$$\tilde{x}(t+1) = L(t+1)^{-1}x(t+1), \quad \tilde{F}(t+1) = L(t+1)^{-1}F(t+1), \quad \tilde{u}(t+1) = L(t+1)^{-1}u(t+1).$$

We can then write

$$\tilde{x}(t+1) = \tilde{F}(t+1)x(t) + \tilde{u}(t+1)$$

so that, for $i = 1, \ldots, m$,

$$\tilde{x}_i(t+1) = \tilde{F}_i(t+1)'x(t) + \tilde{u}_i(t+1), \tag{A1·1}$$

where $\tilde{F}_i(t+1)'$ is the ith row of $\tilde{F}(t+1)$ and $\tilde{x}_i(t+1)$ and $\tilde{u}_i(t+1)$ are the ith elements of $\tilde{x}(t+1)$

Downloaded from https://academic.oup.com/biomet/article-abstract/81/3/541/257082 by University of California, San Diego user on 20 April 2019

and $\tilde{u}(t+1)$. The elements $\tilde{u}_i(t+1)$ are independent $N\{0, \Delta_i(t+1)\}$, where $\Delta_i(t+1)$ is the ith diagonal element of $\Delta(t+1)$. For $i = 1, \ldots, m$ let

$$x(t|t, i) = E\{x(t)|Y^t, x_1(t+1), \ldots, x_i(t+1)\},$$

$$S(t|t, i) = \text{var}\{x(t)|Y^t, x_1(t+1), \ldots, x_i(t+1)\},$$

and define $x(t|t, 0) = x(t|t)$ and $S(t|t, 0) = S(t|t)$. We now apply the observation update step of the Kalman filter m times to (A1·1) as follows.
For $i = 1, \ldots, m$ let

$$\varepsilon(t, i) = \tilde{x}_i(t+1) - \tilde{F}_i(t+1)'x(t|t, i-1),$$

$$R(t, i) = \tilde{F}_i(t+1)'S(t|t, i-1)\tilde{F}_i(t+1) + \Delta_i(t+1).$$

Then

$$x(t|t, i) = x(t|t, i-1) + S(t|t, i-1)\tilde{F}_i(t+1)\varepsilon(t, i)/R(t, i),$$

$$S(t|t, i) = S(t|t, i-1) - S(t|t, i-1)\tilde{F}_i(t+1)\tilde{F}_i(t+1)'S(t|t, i-1)/R(t, i).$$

We therefore obtain

$$x(t|t, m) = E\{x(t)|Y^t, x(t+1)\}, \quad S(t|t, m) = \text{var}\{x(t)|Y^t, x(t+1)\}.$$

It is now straightforward to generate $x(t)$ conditionally on Y^t and $x(t+1)$, as it is normally distributed with mean $x(t|t, m)$ and variance $S(t|t, m)$.

APPENDIX 2

Algorithm to generate indicator variables

We show how to generate K conditional on Y^n, X and θ. We omit dependence on θ. Let $k_1, \ldots, k_m$ be the possible values assumed by $K(t)$ $(t = 1, \ldots, n)$ and suppose that the transition matrices specifying $p\{K(t)|K(t-1)\}$ $(t = 2, \ldots, n)$ are known. We note that if $y(t)$ is observed then

$$p\{K(t)|Y^t, X^t\} \propto p\{y(t)|x(t), K(t)\}p\{x(t)|x(t-1), K(t)\}p\{K(t)|Y^{t-1}, X^{t-1}\},$$

and if $y(t)$ is not observed then

$$p\{K(t)|Y^t, X^t\} \propto p\{x(t)|x(t-1), K(t)\}p\{K(t)|Y^{t-1}, X^{t-1}\}.$$

The following algorithm uses recursive filtering equations following Anderson & Moore (1979, Ch. 8) to calculate $p\{K(t)|Y^t, X^t\}$.

Discrete filter: For $t = 1, \ldots, n$.

Step 1. Performed for $t > 1$,

$$p\{K(t)|Y^{t-1}, X^{t-1}\} = \sum_{j=1}^{m} p\{K(t)|K(t-1) = k_j\}p\{K(t-1) = k_j|Y^{t-1}, X^{t-1}\}.$$

Step 2a. If $y(t)$ is observed set

$$p^*\{K(t)|Y^t, X^t\} = p\{y(t)|x(t), K(t)\}p\{x(t)|x(t-1), K(t)\}p\{K(t)|Y^{t-1}, X^{t-1}\}.$$

Step 2b. If $y(t)$ is not observed set

$$p^*\{K(t)|Y^t, X^t\} = p\{x(t)|x(t-1), K(t)\}p\{K(t)|Y^{t-1}, X^{t-1}\}.$$

Step 3. Obtain $p\{K(t)|Y^t, X^t\}$ using

$$p\{K(t)|Y^t, X^t\} = p^*\{K(t)|Y^t, X^t\} \Big/ \sum_{j=1}^{m} p^*\{K(t) = k_j|Y^t, X^t\}.$$

Downloaded from https://academic.oup.com/biomet/article-abstract/81/3/541/257082 by University of California, San Diego user on 20 April 2019

We note that $p\{y(t)|x(t), K(t)\}$ and $p\{x(t)|x(t-1), K(t)\}$ for $t = 1, \ldots, n$ are known from the specification of the state space model.

Using Lemma 2·2 we show how to generate $K(n), \ldots, K(1)$ in that order conditioning only on Y^n and X. First, we calculate $p\{K(t)|Y^t, X^t\}$ for $t = 1, \ldots, n$ using the discrete filter shown above. To generate $K(t)$ conditional on Y^t, X^t and $K(t+1)$ we use the following result for $t = n-1, \ldots, 1$:

$$p\{K(t)|Y^t, X^t, K(t+1)\} = \frac{p\{K(t+1)|K(t)\}p\{K(t)|Y^t, X^t\}}{p\{K(t+1)|Y^t, X^t\}}.$$

References

Anderson, B. D. O. & Moore, J. B. (1979). *Optimal Filtering*. Englewood Cliffs, New Jersey: Prentice Hall.

Ansley, C. F. & Kohn. R. (1985). Estimation filtering and smoothing in state space models with partially diffuse initial conditions. *Ann Statist.* **13**, 1286–316.

Ansley, C. F. & Kohn, R. (1990). Filtering and smoothing in state space models with partially diffuse initial conditions. *J. Time Ser. Anal.* **11**, 277–93.

Box, G. E. P. & Tiao, G. C. (1968). A Bayesian approach to some outlier problems. *Biometrika* **55**, 119–29.

Carlin, B. P., Polson, N. G. & Stoffer, D. S. (1992). A Monte Carlo approach to nonnormal and nonlinear state space modeling. *J. Am. Statist. Assoc.* **87**, 493–500.

Gelfand, A. E. & Smith, A. F. M. (1990). Sampling-based approaches to calculating marginal densities. *J. Am. Statist. Assoc.* **85**, 398–409.

Gordon, K. & Smith, A. F. M. (1990). Monitoring and modeling biomedical time series. *J. Am. Statist. Assoc.* **85**, 328–37.

Hamilton, J. D. (1989). A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica* **57**, 357–84.

Harrison, P. J. & Stevens, C. F. (1976). Bayesian forecasting (with discussion). *J. R. Statist. Soc.* B **38**, 205–47.

Kitagawa, G. (1987). Non-Gaussian state space modeling of nonstationary time-series (with discussion). *J. Am. Statist. Assoc.* **82**, 1032–63.

Kitagawa, G. & Gersch, W. (1984). A smoothness priors-state space approach to time series with trend and seasonalities. *J. Am. Statist. Assoc.* **79**, 378–89.

Kohn, R. & Ansley, C. F. (1987). A new algorithm for spline smoothing based on smoothing a stochastic process. *SIAM J. Sci. Statist. Comput.* **8**, 33–48.

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

Meinhold, R. J. & Singpurwalla, N. D. (1989). Robustification of Kalman filter models. *J. Am. Statist. Assoc.* **84**, 479–86.

Moran, P. A. P. (1975). The estimation of standard errors in Monte Carlo simulation experiments. *Biometrika* **62**, 1–4.

Schweppe, C. F. (1965). Evaluation of likelihood functions for Gaussian signals. *IEEE Trans. Info. Theory* **11**, 61–70.

Shumway, R. H. & Stoffer, D. S. (1991). Dynamic linear models with switching. *J. Am. Statist. Assoc.* **86**, 763–9.

Tierney, L. (1994). Markov chains for exploring posterior distributions. *Ann. Statist.* To appear.

Wahba, G. (1983). Bayesian 'confidence intervals' for the cross-validated smoothing spline. *J. R. Statist. Soc.* B **45**, 133–50.

West, M. & Harrison, J. (1989). *Bayesian Forecasting and Dynamic Models*, Springer Series in Statistics. New York: Springer-Verlag.

[*Received June* 1992. *Revised July* 1993]

Downloaded from https://academic.oup.com/biomet/article-abstract/81/3/541/257082 by University of California, San Diego user on 20 April 2019