STOCHASTIC VOLATILITY: LIKELIHOOD INFERENCE AND COMPARISON WITH ARCH MODELS

Sangjoon Kim
Salomon Brothers Asia Limited, Tokyo, Japan
Neil Shephard
Nuffield College, Oxford University, Oxford OX1 1NF, UK
and
Siddhartha Chib
John M. Olin School of Business, Washington University, St Louis, MO 63130, USA

October 7, 1996

Abstract

In this paper, Markov chain Monte Carlo sampling methods are exploited to provide a unified, practical likelihood-based framework for the analysis of stochastic volatility models. A highly effective method is developed that samples all the unobserved volatilities at once using an approximating offset mixture model, followed by an importance reweighting procedure. This approach is compared with several alternative methods using real data. The paper also develops simulation-based methods for filtering, likelihood evaluation and model failure diagnostics. The issue of model choice using non-nested likelihood ratios and Bayes factors is also investigated. These methods are used to compare the fit of stochastic volatility and GARCH models. All the procedures are illustrated in detail.

(First version received December 1994)
Some key words: Bayes estimation, Bayes factors, GARCH, Gibbs sampler, Heteroscedasticity, Maximum likelihood, Likelihood ratio, Markov chain Monte Carlo, Quasi-maximum likelihood, Simulation, Stochastic EM algorithm, Stochastic volatility, Stock returns.

1 INTRODUCTION

The variance of returns on assets tends to change over time. One way of modelling this feature of the data is to let the conditional variance be a function of the squares of previous observations and past variances. This leads to the autoregressive conditional heteroscedasticity (ARCH) based models developed by Engle (1982) and surveyed in Bollerslev, Engle, and Nelson (1994).

An alternative to the ARCH framework is a model in which the variance is specified to follow some latent stochastic process. Such models, referred to as stochastic volatility (SV) models, appear in the theoretical finance literature on option pricing (see, for example, Hull and White (1987) in their work generalizing the Black-Scholes option pricing formula to allow for stochastic volatility). For obvious reasons, empirical versions of the SV model are formulated in discrete time. The canonical model in this class for regularly spaced data is:

$$\begin{aligned} y_t &= \beta \epsilon_t \exp(h_t/2), \ \epsilon_t \overset{iid}{\sim} \mathsf{N}(0,1), \\ (h_{t+1} - \mu) &= \phi\,(h_t - \mu) + \eta_t, \ \eta_t \overset{iid}{\sim} \mathsf{N}(0, \sigma_\eta^2), \ t = 1, ..., n, \end{aligned} \tag{1}$$

where y_t is the mean corrected return on holding the asset at time t, h_t is log volatility at time t, $\mathsf{N}(.,.)$ is the normal distribution and iid denotes independent and identically distributed. Here β or $\exp(\mu)$ plays the role of the constant scaling factor and can be thought of as the modal instantaneous volatility, ϕ as the persistence in the volatility and σ_η the volatility of the log-volatility. Either β will have to be set to one or μ to zero to ensure identifiability. In our analysis we will assume that ϵ_t and η_t are uncorrelated, h_t is stationary ($|\phi| < 1$) and that the log-volatility process begins with the stationary initial condition $h_1 \sim \mathsf{N}\left\{\mu, \sigma_\eta^2/(1-\phi^2)\right\}$. All three of these assumptions can be relaxed.

This model has been used as an approximation to the stochastic volatility diffusion by Hull and White (1987) and Chesney and Scott (1989). Its basic econometric properties are discussed in Taylor (1986), the review papers by Taylor (1994), Shephard (1996) and Ghysels, Harvey, and Renault (1996) and the paper by Jacquier, Polson, and Rossi (1994). These papers also review the existing literature on the estimation of SV models.

In this paper we make advances in a number of different directions and provide the first complete Markov chain Monte Carlo simulation-based analysis of the SV model (1) that covers efficient methods for Bayesian and maximum likelihood estimation, likelihood evaluation, computation of filtered volatility estimates, diagnostics for model failure, and computation of statistics for comparing non-nested volatility models. Our study reports on several interesting findings. We consider a very simple Bayesian method for estimating the SV model (based on one-at-a-time updating of the volatilities) which is, unfortunately, inefficient. An improved (multi-move)

method that relies on an offset mixture of normals approximation to a log-chi-square distribution coupled with a importance reweighting procedure is shown to be considerably more effective. Additional refinements of the latter method are developed to reduce the number of blocks in the Markov chain sampling. We also find that the calculation of maximum likelihood estimates (via a Monte Carlo EM algorithm) is computationally less efficient than the Bayesian procedure. We report on useful plots and diagnostics for detecting model failure in a dynamic (filtering) context. Finally, from a formal comparison of alternative models, we find that simple SV models typically fit the data as well as more heavily parameterized GARCH models.

The outline of this paper is as follows. Section 2 contains preliminary discussion of simulation methods and includes a simple, but inefficient, Gibbs sampler for the SV models. Section 3 contains refinements of the basic method that are more efficient. Section 4 considers the computation of the ML estimate via a simulated EM algorithm. Section 5 presents a simulation-based filtering method along with methods for computing the likelihood function and model failure diagnostics. This section also provides non-nested test procedures for comparing the fit of the SV and GARCH class of models. Section 6 considers extensions of the model. Section 7 concludes. Section 8 provides a description of software for fitting these models that is available through the internet. The Appendix details two algorithms used in this paper.

2 PRELIMINARIES

2.1 Quasi-likelihood method

A key feature of the basic SV model in (1) is that it can be transformed into a linear model by taking the logarithm of the squares of the observations

$$\log y_t^2 = h_t + \log \epsilon_t^2, \quad Var(\log \epsilon_t^2) = 4.93. \tag{2}$$

Harvey, Ruiz, and Shephard (1994) have employed Kalman filtering to estimate the unobservable log-volatility, h_t, and used the quasi-likelihood

$$\log L_Q(y|\theta) = -\frac{n}{2}\log 2\pi - \frac{1}{2}\sum_{t=1}^{n}\log F_t - \frac{1}{2}\sum_{t=1}^{n} v_t^2/F_t, \tag{3}$$

to perform parameter estimation. Here $y = (y_1, ..., y_n)$, v_t denotes the one-step-ahead prediction errors for the best linear estimator of $\log y_t^2$, while F_t denotes the corresponding mean square error [1], and θ is the vector of identified unknown parameters $(\phi, \sigma_\eta^2, \beta)$ or $(\phi, \sigma_\eta^2, \mu)$. This quasi-likelihood estimator is suboptimal as $\log \epsilon_t^2$ is poorly approximated by the normal distribution, as

[1] The Kalman filter algorithms for computing v_t and F_t are given in the Appendix.

shown in Figure 1. This means that the quasi-likelihood estimator based on the normal approximation has poor small sample properties, even though the usual quasi-likelihood asymptotic theory is correct.

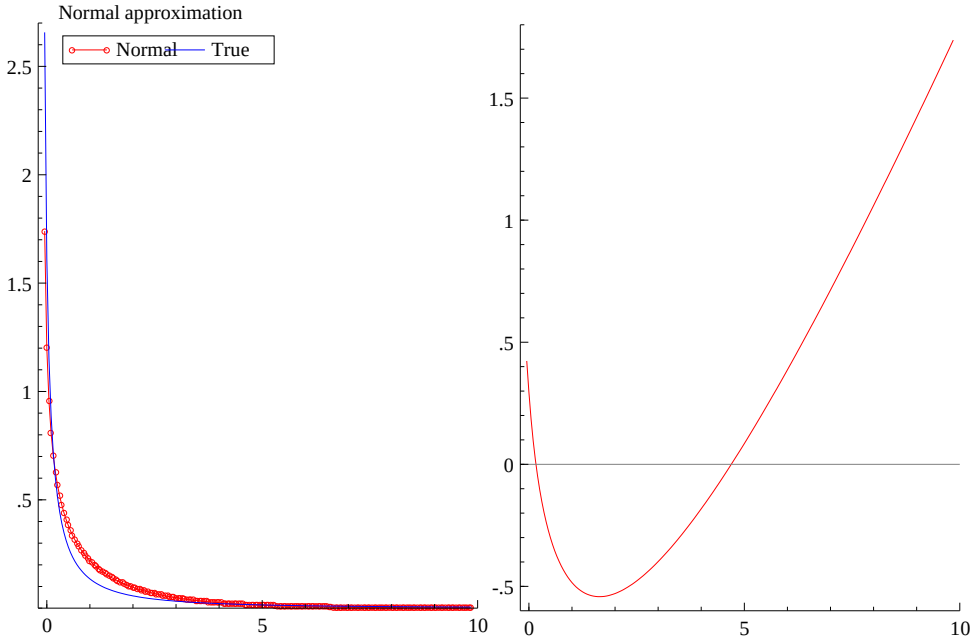

Figure 1: *Log-Normal approximation to* χ^2_1 *density. Left is the* χ^2_1 *density and approximation. Right is the log of the ratio of the* χ^2_1 *density to the approximation.*

2.2 Markov chain Monte Carlo

An alternative, exact approach to inference in the SV model is based on Markov chain Monte Carlo (MCMC) methods, namely the Metropolis-Hastings and Gibbs sampling algorithms. These methods have virtually revolutionized the practice of Bayesian statistics and have found a number of applications in the recent statistical literature. Early work on these methods appears in Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller (1953), Hastings (1970), Ripley (1977) and Geman and Geman (1984) while the more recent developments spurred by Gelfand and Smith (1990) are summarized in Gilks, Richardson, and Spiegelhalter (1996) and Tanner (1996, Ch. 6). The econometrics work on this topic is reviewed in Chib and Greenberg (1996). A tutorial introduction to the Metropolis-Hastings algorithm including its derivation from the logic of reversibility, is given by Chib and Greenberg (1995).

In our application, following the general principle outlined by Tanner and Wong (1987), we will develop MCMC methods to sample the augmented posterior density $(h, \theta)\,|y$ that includes the latent volatilities $h = (h_1, ..., h_n)$ as unknown parameters. This augmentation of

the parameter space bypasses the calculation of the intractable likelihood function $f(y|\theta) = \int f(y|h,\theta)f(h|\theta)dh$ and makes posterior simulation feasible. The great virtue of sampling based methods is that they can be used to estimate various summaries of the posterior distribution without numerical integration. For example, posterior moments and marginal densities can be obtained by averaging the relevant function of interest over the sampled variates. The posterior mean of θ is simply estimated by the sample mean of the simulated θ values. The accuracy of the resulting estimates can be assessed statistically and the estimates can be made arbitrarily accurate by increasing the simulation sample size.

2.2.1 An initial Gibbs sampling algorithm for the SV model

For the problem of simulating a multivariate density x, the Gibbs sampler with each component blocked is defined by the distributions $x_i|x_{\backslash i}$, where $x_{\backslash i}$ denotes the elements of x excluding x_i. The algorithm proceeds by sampling each component of x from these (so called) full conditional distributions where the most recent values of the conditioning variables are used in the simulation. One cycle of the algorithm is called a sweep or a scan. Under regularity conditions, as the sampler is repeatedly swept, the draws from the sampler converge to draws from the multivariate density x at a geometric rate. For the SV model the x vector becomes $(h,\theta)|y$. To sample this posterior density, one possibility (suggested by Jacquier, Polson, and Rossi (1994) and Shephard (1993)) is to update each of the elements of the x vector one at a time.

1. Initialize h and θ.
2. Sample with replacement h_t from $h_t|h_{\backslash t}, y, \theta$, $t = 1, ..., n$.
3. Sample $\sigma_\eta^2|y, h, \phi, \mu, \beta$.
4. Sample $\phi|h, \mu, \beta, \sigma_\eta^2$.
5. Sample $\beta|y, h$ or from $\mu|h, \phi, \sigma_\eta^2$.
6. Goto 2.

Cycling through 2 to 5 is a complete sweep of this (single move) sampler. The Gibbs sampler will require us to perform many thousands of sweeps to generate samples from $\theta, h|y$.

The most difficult part of this sampler is to effectively sample from $h_t|h_{\backslash t}, y_t, \theta$ as this operation has to be carried out n times for each sweep. However,

$$f(h_t|h_{\backslash t}, \theta, y) \propto f(h_t|h_{\backslash t}, \theta)f(y_t|h_t, \theta), \quad t = 1, ..., n.$$

We can sample from this density using a simple accept/reject algorithm. [2] We write, ignoring end conditions to save space,

$$h_t|h_{\setminus t},\theta = h_t|h_{t-1},h_{t+1},\theta \sim \mathsf{N}(h_t^*,v_t^2),$$

where

$$h_t^* = \mu + \frac{\phi\left\{(h_{t-1}-\mu)+(h_{t+1}-\mu)\right\}}{(1+\phi^2)} \qquad \text{and} \qquad v_t^2 = \frac{\sigma_\eta^2}{(1+\phi^2)}$$

and $\log f(y_t|h_t,\theta) = const + \log f^*$ where

$$\begin{aligned}\log f^* &= -\frac{1}{2}h_t - \frac{y_t^2}{2\beta^2}\left\{\exp(-h_t)\right\} \\ &\leq -\frac{1}{2}h_t - \frac{y_t^2}{2\beta^2}\left\{\exp(-h_t^*)(1+h_t^*) - h_t\exp(-h_t^*)\right\} \\ &= \log g^*.\end{aligned}$$

Hence we can draw from f using an accept/reject algorithm (see Ripley (1987)). The proposal value is drawn from the normalized version of g^*, a normal density with mean

$$\mu_t = h_t^* + \frac{v_t^2}{2}\left[\frac{y_t^2}{\beta^2}\exp(-h_t^*) - 1\right], \tag{4}$$

and variance v_t^2. Thus we can sample from $h_t|h_{\setminus t},\theta,y$ by proposing $h_t \sim \mathsf{N}(\mu_t,v_t^2)$ and accepting with probability f^*/g^*.[3]

Sampling σ_η^2 and ϕ Sampling the σ_η^2 and ϕ one at a time is straightforward. If we assume a conjugate prior $\sigma_\eta^2|\phi,\beta \sim S_\sigma/\chi^2_{\sigma_r}$, then σ_η^2 is sampled from

$$\sigma_\eta^2|y,h,\phi,\mu \sim \chi^{-2}_{n+\sigma_r}\left\{S_\sigma + (h_1-\mu)^2(1-\phi^2) + \sum_{t=1}^{n-1}\left\{(h_{t+1}-\mu)-\phi(h_t-\mu)\right\}^2\right\}. \tag{5}$$

[2] Five previous MCMC algorithms for simulating from $h_t|h_{t-1},h_{t+1},y_t;\theta$ have been given in the literature by Shephard (1993), Jacquier, Polson, and Rossi (1994), Shephard and Kim (1994), Geweke (1994) and Shephard and Pitt (1997). The closest to our suggestion is Geweke (1994) who also bounded $\log f^*$, but by $-0.5h_t$. This suffers from the property of having a high rejection rate for slightly unusual observations (for example, 0.9 for $|y_t|/\beta\exp(h_t/2) > 3$). Shephard and Pitt (1997), on the other hand, used a quadratic expansion of $\log f^*$ about h_t^*. This increases the generality of the procedure but it involves a Metropolis rejection step and so is more involved. Shephard (1993) approximated f^* by a normal distribution with the same moments as $\log\epsilon_t^2$.

Geweke (1994) and Shephard and Kim (1994) independently suggested the use of the Gilks and Wild (1992) procedure for sampling from log concave densities such as $\log f(h_t|h_{\setminus t},\theta,y)$. This is generalizable to non-concave densities using the Gilks, Best, and Tan (1995) sampler. Typically these routines need about 10 to 12 evaluations of $\log f(h_t|h_{\setminus t},\theta,y)$ to draw a single random variable. Hence they are about 10 times less efficient than the simple accept/reject algorithm given above.

Jacquier, Polson, and Rossi (1994)'s Metropolis algorithm uses a very different approach. They approximate the density of $h_t|h_{\setminus t}$ and so use a non-Gaussian proposal. Typically this procedure is considerably slower than the use of the Gilks and Wild (1992) methods suggested above.

[3] This proposal has an average acceptance rate of approximately $1-y_t^2\exp(-h_t^*)v_t^2/\left(4\beta^2\right)$. A typical situation is where $v_t^2 = 0.01$. Usually $y_t^2\exp(-h_t^*)v_t^2/\beta^2$ will not be very large as h_t^* is the smoothed log-volatility of y_t and so reflects the variation in y_t. An extreme case is where $y_t^2\exp(-h_t^*)\sigma_t^2/\beta^2 = 100$, which leads to an average acceptance rate of approximately 0.75 . In our experience an average acceptance rate of over 0.995 seems usual for real financial datasets.

throughout we set $\sigma_r = 5$ and $S_\sigma = 0.01 \times \sigma_r$.

For ϕ, sampling from the full conditional density is also easy. Suppose that the prior distribution on ϕ (concentrated on the stationary interval $|\phi| < 1$) is given by

$$\log f(\phi|\sigma^2, \beta) = const + \left(\phi^{(1)} - 1\right) \log\left\{\frac{(1+\phi)}{2}\right\} + \left(\phi^{(2)} - 1\right) \log\left\{\frac{(1-\phi)}{2}\right\}, \quad \phi^{(1)}, \phi^{(2)} > \frac{1}{2}. \tag{6}$$

This implies a mean of $\left\{2\phi^{(1)}/\left(\phi^{(1)} + \phi^{(2)}\right) - 1\right\}$. In our work we will select $\phi^{(1)} = 20$ and $\phi^{(2)} = 1.5$, giving a prior mean of 0.86. The likelihood of ϕ is

$$\log f(h|\phi, \sigma^2) = const - \sum_{t=1}^{n-1} \frac{\{(h_{t+1} - \mu) - \phi(h_t - \mu)\}^2}{2\sigma_\eta^2} - \frac{(h_1 - \mu)^2 (1 - \phi^2)}{2\sigma_\eta^2} + \frac{1}{2} \log\left(1 - \phi^2\right), \tag{7}$$

implying that the full conditional density of ϕ is concave in ϕ for all values of $\phi^{(1)}, \phi^{(2)}$. This posterior can now be sampled by a variety of means including the accept/reject or Metropolis-Hastings methods. Of course, if necessary alternative priors on ϕ can also be used. For example, the flat prior $f(\phi|\sigma_\eta^2, \beta) \propto 1$ has some attractive properties, in particular allowing a conjugate analysis (Sims and Uhlig (1991) and Jacquier, Polson, and Rossi (1994)). But this prior can cause problems when the data are close to being non-stationary (Phillips (1991) and Schotman and Van Dijk (1991)). Other priors (restricted to the stationary region) for autoregressive models have been discussed by Chib and Greenberg (1994) and Marriott and Smith (1992). We feel that it is important to impose stationarity in the SV model so as to control the implied risk premium. Further, if $\phi = 1$ then the μ terms cancel in (1) and so μ becomes unidentified from the data. The prior we select avoids these two problems rather well.

Sampling β or μ The model has been parameterized to allow two possible ways of modelling the modal volatility, either through β or μ. Whichever is chosen, the other will have to be fixed at some known value. In the case of β this is unity, in the case of μ this is 0. Here we detail the full conditional distributions that are used to sample either parameter.

Suppose $\beta^2 \sim S_\beta/\chi^2_{\beta_r}$, then

$$\beta^2|y, h \sim \chi^{-2}_{n+\beta_r}\left\{S_\beta + \sum_{t=1}^{n} y_t^2 \exp(-h_t)\right\}. \tag{8}$$

On the other hand, if we work with a diffuse prior on μ, then it is sampled using

$$\mu|h, \phi, \sigma_\eta^2 \sim N\left[\frac{\sigma_\mu^2}{\sigma_\eta^2}\left\{\mu_h (1-n)(1-\phi)^2 + h_1\left(1 - \phi^2\right)\right\}, \sigma_\mu^2\right],$$

where

$$\sigma_\mu^2 = \sigma_\eta^2\left\{(1-n)(1-\phi)^2 + \left(1-\phi^2\right)\right\}^{-1} \qquad \text{and} \qquad \mu_h = \frac{1}{(n-1)(1-\phi)} \sum_{t=2}^{n} (h_t - \phi h_{t-1}).$$

By selecting $\beta_r = 1$ and $S_\beta = 0$, the prior for β and μ are identical and draws from one sampler can be used to make inference about the other parameter. Indeed in most of our work we will sample μ and record $\exp(\mu/2)$.

Illustration To illustrate this algorithm we analyze the daily returns on the UK Sterling/US Dollar exchange rate from 1/10/81 to 28/6/85. Later in the paper we will also use the corresponding series for the German Deutschemark (DM), Japanese Yen and Swiss Franc (SwizF), all against the US Dollar. This dataset has been previously analysed using quasi-likelihood methods in Harvey, Ruiz, and Shephard (1994). The returns will be computed as

$$y_t = 100 \times \left\{ (\log r_t - \log r_{t-1}) - \frac{1}{n} \sum_{i=1}^{n} (\log r_i - \log r_{i-1}) \right\},$$

where r_t denotes the exchange rate at time t. The MCMC sampler was initialized by setting all

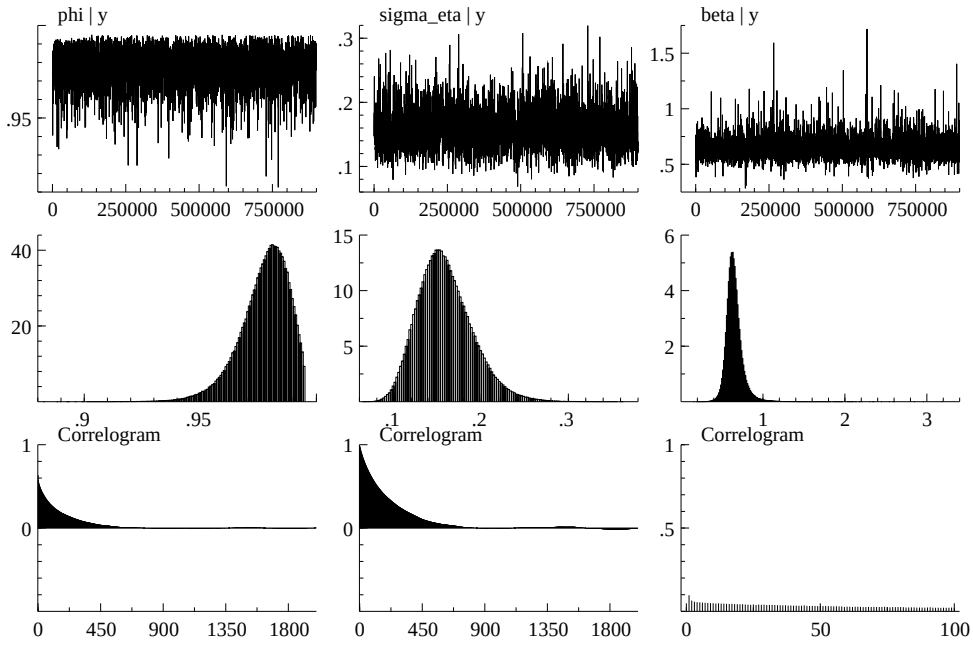

Figure 2: *Single move Gibbs sampler for the Sterling series. Top graph: parameters against iteration number. Bottom: the correlograms from the simulations. In total 1,000,000 iterations were drawn, discarding the first 50,000.*

the $h_t = 0$ and $\phi = 0.95$, $\sigma_\eta^2 = 0.02$ and $\mu = 0$. We iterated the algorithm on the log-volatilities for $1,000$ iterations and then the parameters and log-volatilities for $50,000$ more iterations, before recording any answers. The complete $1,000,000$ iterations[4] are graphed in Figure 2 and

[4]We have employed a 32 bit version of the modified Park and Miller (1988) uniform random number as the basis of all our random numbers. This has a period of $2^{31} - 1$, which allows us to draw around 2.1 billion random

summarized in Table 1. [5]

The summary statistics of Table 1 report the inefficiency factors of the sampler. These are estimated as the variance of the sample mean from the MCMC sampling scheme relative to a hypothetical sampler which draws independent random variables from the posterior. This ratio is estimated using a Parzen window (see, for example, Priestley (1981, Ch. 6)) with

$$\widehat{R}_{B_M} = 1 + \frac{2B_M}{B_M - 1} \sum_{i=1}^{B_M} K\left(\frac{i}{B_M}\right) \widehat{\rho}(i),$$

where $\widehat{\rho}(i)$ is an estimate of the autocorrelation at lag i of the MCMC sampler, B_M represents the bandwidth, and K the Parzen kernel given by

$$\begin{aligned} K(z) &= 1 - 6z^2 + 6z^3, && z \in [0, \tfrac{1}{2}], \\ &= 2(1-z)^3, && z \in [\tfrac{1}{2}, 1], \\ &= 0, && \text{elsewhere.} \end{aligned}$$

The correlogram indicates important autocorrelations for ϕ and σ_η at large lag lengths. If we require the Monte Carlo error in estimating the mean to be no more than one percentage of the variation of the error due to the data, then this Gibbs sampler would have to be run for around $40,000$ iterations. This seems a reasonably typical result: see Table 2.

	Mean	MC S.E.	Inefficiency	Covariance & *Correlation*		
$\phi \vert y$	0.97762	0.00013754	163.55	0.00011062	*-0.684*	*0.203*
$\sigma_\eta \vert y$	0.15820	0.00063273	386.80	-0.00022570	0.00098303	*-0.129*
$\beta \vert y$	0.64884	0.00036464	12.764	0.00021196	-0.00040183	0.0098569
Time	5829.5	0.58295				

Table 1: *Daily returns for Sterling: summaries of Figure 2. The Monte Carlo S.E. of simulation is computed using a bandwidth of 2,000, 4,000 and 2,000 respectively. Italics are correlations rather than covariances of the posterior. Computer time is seconds on a Pentium Pro/200. The other time is the number of seconds to perform 100 sweeps of the sampler.*

Parameterization An alternative to this sampler is to replace the draw for $\mu|h, \phi, \sigma_\eta^2$ with $\beta|y, h$. Such a move would be a mistake. Table 3 reports the inefficiency factor for this sampler using 1,000,000 draws of this sampler. There is a small deterioration in the sampler for $\phi|y$ and a very significant reduction in efficiency for $\beta|y$. The theoretical explanation for the inadequacies of the β parameterization is provided by Pitt and Shephard (1996).

numbers. In these experiments we are drawing approximately $n \times 2 \times 1.05$ random numbers per sweep of the sampler, where 5% is a very conservative estimate of the overall rejection rate. For this dataset this is 1984 draws per sweep. Given that we employ $1,000,000$ sweeps, we are close, but not beyond, the period of our random number generator.

[5] Timings will be given for all the computations given in this paper. These are made using the authors C++ code which has been linked to Ox. The single move algorithm is optimised to this special case and so is about as fast as it is possible to make it. The latter algorithms are much more general and so it is not fair to compare the computed time reported here to their times.

	$\phi\|y$		$\sigma_\eta\|y$		$\beta\|y$	
Series	Mean	Inefficiency	Mean	Inefficiency	Mean	Inefficiency
DM	0.96496	122.77	0.15906	292.81	0.65041	15.762
Yen	0.98010	313.03	0.12412	676.35	0.53597	14.192
SwizF	0.95294	145.48.	0.20728	231.15	0.70693	13.700

Table 2: *Bandwidth was 2,000, 4,000 and 2,000, respectively for the parameters, for all series. In all cases 1,000,000 sweeps were used.*

	$\phi\|y$		$\sigma_\eta\|y$		$\beta\|y$	
Series	Mean	Inefficiency	Mean	Inefficiency	Mean	Inefficiency
Sterling	0.97793	465.30	0.15744	439.73	0.64280	5079.6

Table 3: *Bandwidth was 4,000, 4,000 and 15,000, respectively for the parameters, for all series. 1,000,000 sweeps were used.*

Reason for slow convergence The reason for the slow convergence reported in Table 1 can be discerned by thinking of the Gaussian equivalent of this problem where analytic convergence results exist. Under the Gaussian assumption and the linear approximation (2) and (1), the sampler in the simulation of h from $h|y;\theta$ has an analytic convergence rate of (Pitt and Shephard (1996, Theorem 1))

$$4\phi^2 / \left\{1 + \phi^2 + \sigma_\eta^2 / \mathsf{Var}(\log \varepsilon_t^2)\right\}^2,$$

where θ is taken as fixed at the expected values given in the results for the Sterling series. If $\mathsf{Var}(\log \varepsilon_t^2)$ is set equal to 4.93, then this result implies a geometric convergence rate of 0.9943 and an inefficiency factor of 175 which is in the range reported in Table 1.

In order to improve the above sampler it is necessary to try to sample the log-volatilities in a different way. One method is to sample groups of consecutive log volatilities using a Metropolis algorithm. This is investigated in Shephard and Pitt (1997). In this paper we detail a more ambitious model specific approach. This approach is described next.

3 OFFSET MIXTURE METHOD

In this section we design an offset mixture of normals distribution (defined below) to accurately approximate the exact likelihood. This approximation helps in the production of an efficient (adapted Gibbs sampler) Monte Carlo procedure that allows us to sample all the log-volatilities at once. We then show how one can make the analysis exact by correcting for the (minor) approximation error by reweighting the posterior output.

3.1 The model

Our approximating parametric model for the linear approximation (2) will be an offset mixture time series model

$$\log(y_t^2 + c) = h_t + z_t, \qquad z_t|\omega = i \sim \mathsf{N}(\mu_i - 1.2704, \sigma_i^2),$$

where z_t is a mixture of K normals which are selected to closely approximate the true density of $\log \epsilon_t^2$. The "offset" c was introduced into the SV literature by Fuller (1996, pp. 494-7) in order to robustify the QML estimator of the SV to y_t^2 being very small. Throughout we will set $c = 0.001$.

If we denote by $k_Z(z)$ the mixture density approximation to $\log \epsilon_t^2$, we have

$$k_Z(z) = \sum_{i=1}^{K} \pi_i f_{z|\omega=i}(z|\omega = i), \quad z|\omega = i \sim \mathsf{N}(\mu_i - 1.2704, \sigma_i^2). \tag{9}$$

We are now in a position to select K, $\mu_1, ..., \mu_K$, $\pi_1, ..., \pi_K$ and $\sigma_1^2, ..., \sigma_K^2$ to make the approximation "sufficiently good". In our work, following for instance Titterington, Smith, and Makov (1985, p. 133), we matched the first four moments of $f_{\exp(Z)}(r)$ and $k_Z(r)$ to those of a χ_1^2 and $\log \chi_1^2$ random variable respectively, and required that the approximating densities lie within a small distance of the true density. This was carried out by using a non-linear least squares program to move the weights, means and variances around until the answers were satisfactory. It is worth noting that this nonlinear optimisation incurs only a one-time cost, as there are no model-dependent parameters involved. We found what we judged to be satisfactory answers by setting $K = 7$. The implied weights, means and variances are given in Table 4, while the approximating and the true density are drawn in Figure 3. It would be easy to improve the fit by increasing the value of K, however further experiments that we have conducted suggest that increasing K has little discernible effect on our main results.

ω	$Pr(\omega = i)$	μ_i	σ_i^2
1	0.00730	-10.12999	5.79596
2	0.10556	-3.97281	2.61369
3	0.00002	-8.56686	5.17950
4	0.04395	2.77786	0.16735
5	0.34001	0.61942	0.64009
6	0.24566	1.79518	0.34023
7	0.25750	-1.08819	1.26261

Table 4: *Selection of the Mixing Distribution to be* $\log \chi_1^2$.

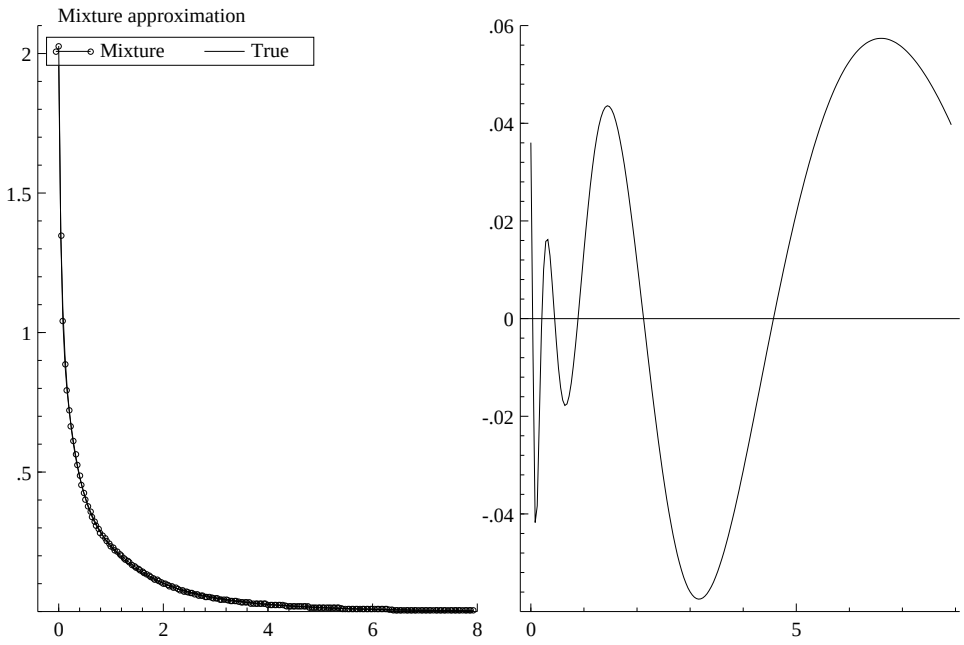

Figure 3: Left: $\log \chi_1^2$ density and mixture approximation. Right: the log of the ratio of the χ_1^2 density to the approximation.

3.2 Mixture simulator

Mixture models were, until recently, statistically intractable, but recent MCMC work has overcome this problem. The general algorithm was suggested independently by Shephard (1994) and Carter and Kohn (1994). Define $x = \omega, h, \phi, \sigma_\eta, \mu | y$. Then the sampler has the form

1. Initialize ω, ϕ, σ_η and μ.
2. Sample $h|y, \omega,\ \phi,\ \sigma_\eta,\ \mu$.
3. Sample $\omega|y, h$.
4. Update ϕ, σ_η, μ according to (7), (8) and (5).
5. Goto 2.

The important improvement over the methods in section 2 is that it is now possible to efficiently sample from the highly multivariate Gaussian distribution $h|y, \omega, \phi, \sigma_\eta, \mu$ because $y|\omega, \phi, \sigma_\eta, \mu$ is a Gaussian time series which can be placed into the state-space form associated with the Kalman filter. The time series literature calls such models partially non-Gaussian or conditionally Gaussian. This particular model structure means we can sample from the entire $h|y, \omega, \phi, \sigma_\eta, \mu$ using the Gaussian simulation signal smoother detailed in the Appendix.

The results from 750,000 sweeps of this mixture sampler are given in Table 5 and Figure 4. This sampler has less correlation than the single move sampler and suggests that generating 20,000 simulations from this sampler would probably be sufficient for inferential purposes.

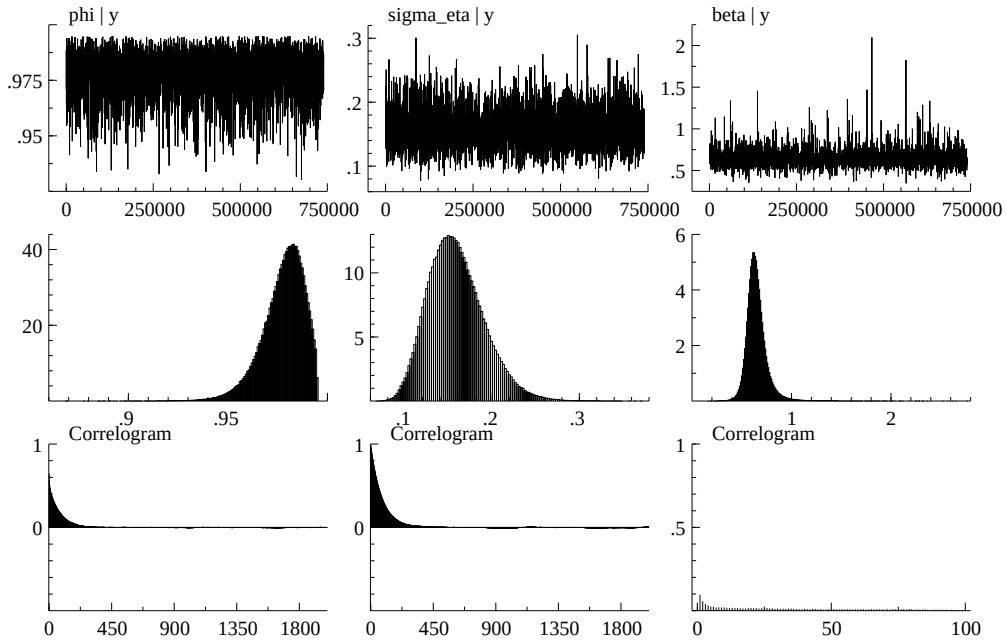

Figure 4: *Mixture sampler for Sterling series. Top: parameters against iteration number. Bottom: correlograms from the simulations. In total 750,000 iterations were drawn, discarding the first 10,000.*

	Mean	MC S.E.	Inefficiency	Covariance & *Correlation*		
$\phi\|y$	0.97779	6.6811e-005	29.776	0.00011093	*-0.690*	*0.203*
$\sigma_\eta\|y$	0.15850	0.00046128	155.42	-0.00023141	0.0010131	*-0.127*
$\beta\|y$	0.64733	0.00024217	4.3264	0.00021441	-0.00040659	0.010031
Time	15374	2.0498				

Table 5: *Daily returns for Sterling against Dollar. Summaries of Figure 2. The Monte Carlo S.E. of simulation is computed using a bandwidth of 2000, 2000 and 100 respectively. Italics are correlations rather than covariances of the posterior. Computer time is seconds on a Pentium Pro/200. The other time is the number of seconds to perform 100 complete passes of the sampler.*

3.3 Integrating out the log-volatilities

Although this mixture sampler improves the correlation behaviour of the simulations, the gain is not very big as there is a great deal of correlation between the volatilities and parameters. However, we can use the Gaussian structure of $y|\omega, \sigma_\eta, \phi$ to overcome this. We can define

$x = \omega, \sigma_\eta, \phi | y$ and then update these in blocks. This can be carried out as we can evaluate $y|\omega, \sigma_\eta, \phi$ using the Kalman filter. Then up to a constant of proportionality, the posterior density of $\sigma_\eta, \phi|y, \omega$ is

$$\log f(\sigma_\eta, \phi|y, \omega) = const + \log f(\phi) + \log f(\sigma_\eta) - \frac{1}{2}\sum_{t=1}^{n} \log F_t - \frac{1}{2}\sum_{t=1}^{n} v_t^2/F_t,$$

where v_t denotes the one-step-ahead prediction errors for the best mean square estimator of $\log(y_t^2 + c)$, and F_t denotes the corresponding mean square error. Both of these quantities are calculated conditional on ω.

This implies that we can sample from $\sigma_\eta, \phi|y, \omega$ directly by making the proposal $\left\{\sigma_\eta^{(i)}, \phi^{(i)}\right\}$, given the current value $\left\{\sigma_\eta^{(i-1)}, \phi^{(i-1)}\right\}$, by drawing from some density $g(\sigma_\eta, \phi)$ and then accepting them using the Metropolis-Hastings probability of move

$$\exp\left[\min\left\{0, \log f(\sigma_\eta^{(i)}, \phi^{(i)}|y, \omega) - \log f(\sigma_\eta^{(i-1)}, \phi^{(i-1)}|y, \omega) + \log g(\sigma_\eta^{(i-1)}, \phi^{(i-1)}) - \log g(\sigma_\eta^{(i)}, \phi^{(i)})\right\}\right]. \tag{10}$$

If this suggestion is rejected then we write $\left\{\sigma_\eta^{(i)}, \phi^{(i)}\right\} = \left\{\sigma_\eta^{(i-1)}, \phi^{(i-1)}\right\}$. We call this an 'integration sampler' as it integrates out the log-volatilities.

The structure of this sampler is then generically:

1. Initialize ω, ϕ, σ_η and μ.
2. Update ϕ and σ_η from ϕ, $\sigma_\eta|y, \omega$ using a Metropolis-Hastings suggestion based on $g(\sigma_\eta, \phi)$ accepting with probability (10).
3. Sample $h, \mu|y, \omega, \phi, \sigma_\eta$.
4. Sample $\omega|y, h$.
5. Goto 2.

An important characteristic of this sampler is that the simulation smoother can jointly draw h and μ. The scheme allows a free choice of the proposal density $g(\sigma_\eta, \phi)$. We have employed a composite method which first draws 200 samples (discarding the first ten samples) from the posterior density of $\sigma_\eta, \phi|y$ using a robust Metropolis-Hastings sampler based on Gilks, Best, and Tan (1995) which only requires the coding of the function $y|\omega, \phi, \sigma_\eta$ and the prior. We then used these 200 draws to estimate the posterior mean and covariance. The mean and twice the covariance are then used to form a Gaussian proposal density $g(\sigma_\eta, \phi)$ for the above Metropolis-Hastings algorithm. As an alternative, one could also use a multivariate Student t proposal

distribution instead of the Gaussian. See Chib and Greenberg (1995) for further discussion on the issues involved in choosing a proposal density for the Metropolis-Hastings algorithm.

The resulting sampler is reported in Figure 5 and Table 6. These suggest that 2,000 samples from this generator would be sufficient for this problem. This result seems reasonably robust to the dataset.

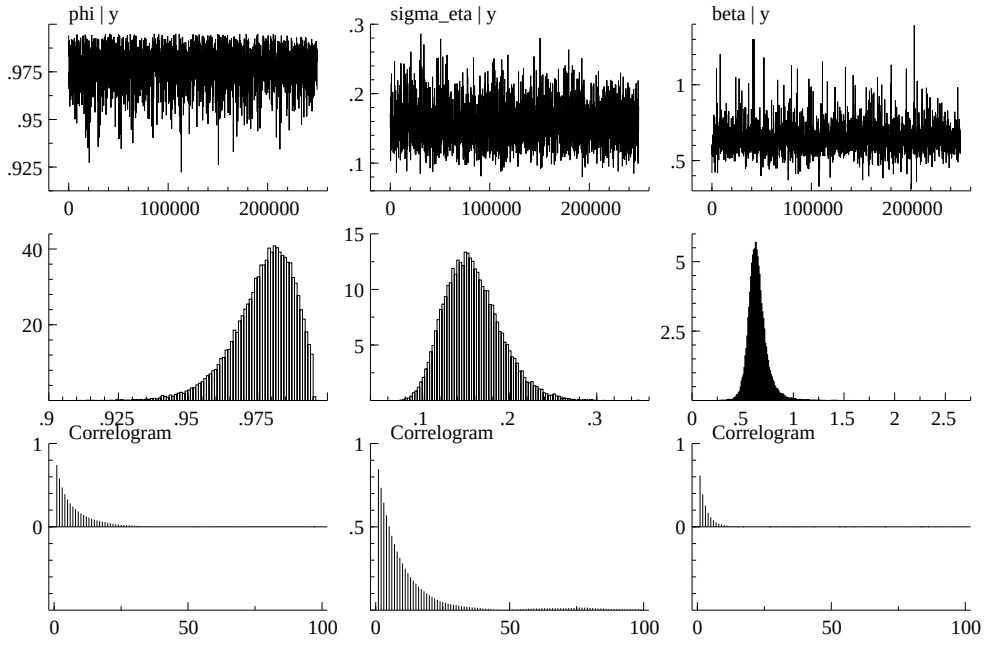

Figure 5: *The integration sampler for Sterling series. Top: parameters against iteration number. Bottom: correlograms from the simulations. In total 250,000 iterations were drawn, discarding the first 250.*

	Mean	MC S.E.	Inefficiency	Covariance & *Correlation*		
$\phi \vert y$	0.97780	6.7031e-005	9.9396	0.00011297	*-0.699*	*0.205*
$\sigma_\eta \vert y$	0.15832	0.00025965	16.160	-0.00023990	0.0010426	*-0.131*
$\beta \vert y$	0.64767	0.00023753	1.4072	0.00021840	-0.00042465	0.010020
Time	8635.2	3.4541				

Table 6: *Daily returns for Sterling against Dollar. Summaries of Figure 5. The Monte Carlo S.E. of simulation is computed using a bandwidth of 100, 100 and 100 respectively. Italics are correlations rather than covariances of the posterior. Computer time is seconds on a Pentium Pro/200. The other time is the number of seconds to perform 100 complete passes of the sampler.*

3.4 Reweighting

The approach based on our (very accurate) offset mixture approximation provides a neat connection to conditionally Gaussian state space models and leads to elegant and efficient sampling procedures, as shown above. Another virtue of the offset mixture approach is that it can be made to produce a sample from the exact posterior density of the parameters and volatilites by appending a straightforward step at the conclusion of the above procedures. First write the mixture approximation as making draws from $k(\theta, h|y)$, then define

$$r(\theta, h) = \log f(\theta, h|y) - \log k(\theta, h|y) = const + \log f(y|h) - \log \sum_{j=1}^{K} k(y|h, \omega_j) \Pr(\omega_j),$$

yielding

$$\begin{aligned} \mathsf{E}g(\theta)|y &= \int g(\theta) f(\theta|y) d\theta \\ &= \int g(\theta) \exp\{r(\theta, h)\} k(\theta, h|y) d\theta dh / \int \exp\{r(\theta, h)\} k(\theta, h|y) d\theta dh. \end{aligned}$$

Thus we can estimate functionials of the posterior by reweighting the MCMC draws according to

$$\widehat{\mathsf{E}g(\theta)|y} = \sum_j g(\theta^j) c^j,$$

where the weights are

$$c^j = \exp\left\{r(\theta^j, h^j)\right\} / \sum_i \exp\left\{r(\theta^i, h^i)\right\}. \tag{11}$$

As the mixture approximation is very good, we would expect that the weights c^j would have a small variance.

To see the dispersion of the weights, we recorded the weights from the sampler which generated Figure 5 and plotted the resulting log-weights in Figure 6. The log-weights are close to being normally distributed with a standard deviation of around one.

To see the effect of the weights on the parameters estimates, we reweighted the 250,000 samples displayed in Figure 5. This produced the estimates which are given in Table 7. These Monte Carlo estimates of the posterior means are statistically insignificantly different from Monte Carlo estimated values given in Table 1. However, the Monte Carlo precision has improved dramatically. Further, the Monte Carlo standard errors indicate that this data set could be routinely analysed using around 1,500 sweeps.

This conclusion seems to hold up for some other exchange rate series. Table 8 reports the estimates of the parameters and inefficiency models for the DM, Yen and Swiss Franc series. This table is the exact analog of Table 2 for the single move algorithm.

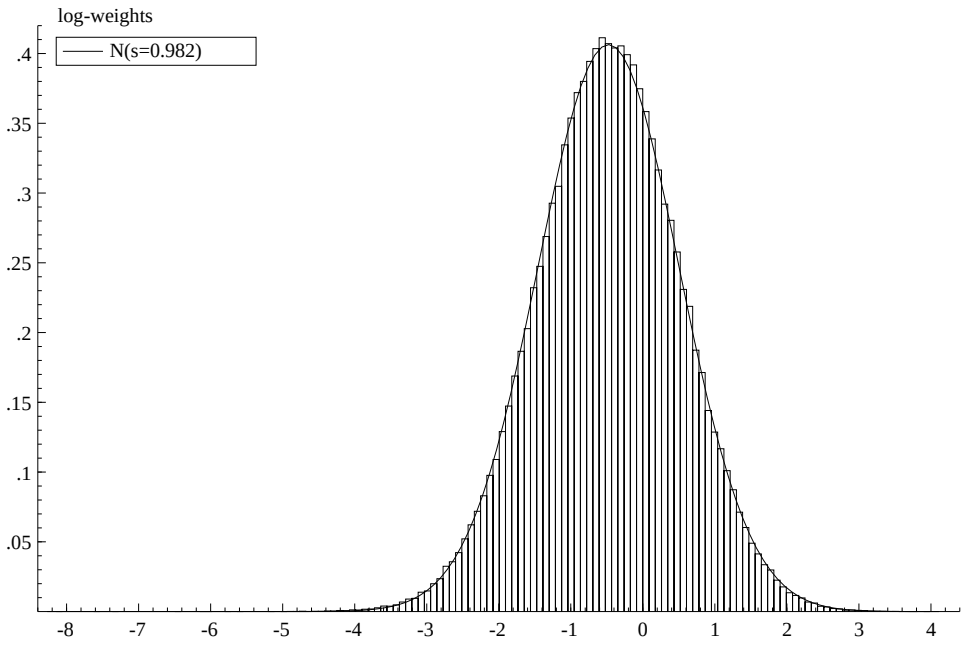

Figure 6: *Log of the $M \times c^j$ for 250,000 sweeps for the integration sampler. All the weights around zero would indicate a perfect sampler.*

4 APPROXIMATE MAXIMUM LIKELIHOOD ESTIMATION

In this section we consider the computation of the maximum likelihood estimate without utilizing the (intractable) likelihood function of the SV model. Our approach is based on a simulated version of the EM algorithm. The theory behind this approach is straightforward and is discussed in detail in Qian and Titterington (1991) and reviewed by Tanner (1996, Ch.4).

The EM algorithm exploits the decomposition:

$$\log f(y;\theta) = \log \Pr(\omega) + \log f(y|\omega;\theta) - \log \Pr(\omega|y;\theta). \tag{12}$$

	Mean	MC S.E.	Inefficiency	Covariance & *Correlation*		
$\phi\|y$	0.97752	7.0324e-005	11.20	0.00010973	*-0.685*	*0.204*
$\sigma_\eta\|y$	0.15815	0.00024573	14.81	-0.00022232	0.00096037	*-0.129*
$\beta\|y$	0.64909	0.00025713	1.64	0.00021181	-0.00039768	0.0098312
Time	10105	4.0423				

Table 7: *Daily returns for Sterling against Dollar. Summaries of reweighted sample of 250,000 sweeps of the integration sampler. The Monte Carlo S.E. of simulation is computed using a block one tenth of the size of the simulation. Italics are correlations rather than covariances of the posterior. Computer time is seconds on a Pentium Pro/200. The other time is the number of seconds to perform 100 complete passes of the sampler.*

	$\phi\|y$		$\sigma_\eta\|y$		$\beta\|y$	
Series	Mean	Inefficiency	Mean	Inefficiency	Mean	Inefficiency
DM	0.96529	8.31	0.15812	11.99	0.65071	9.73
Yen	0.97998	23.10	0.12503	35.66	0.53534	2.71
SwizF	0.95276	13.52	0.20738	15.33	0.70675	8.38

Table 8: *Bandwidth for each parameter was 100 on all series. In all cases 250,000 sweeps were used.*

Since $y|\omega, \theta$ is a normal, the likelihood

$$\log f(y|\omega;\theta) = const - \frac{1}{2}\sum_{t=1}^{n} \log F_t - \frac{1}{2}\sum_{t=1}^{n} v_t^2/F_t, \tag{13}$$

can be computed using the Kalman filter, where v_t is the one-step-ahead prediction error for $\log(y_t^2 + c)$ given ω and θ and F_t is the corresponding mean square error. The EM algorithm iterates:

$$\theta^{(i+1)} = \underset{\theta}{\operatorname{argmax}}\, Q(\theta, \theta^i; y) = \underset{\theta}{\operatorname{argmax}} \sum_{\omega} \log f(y|\omega;\theta) \Pr(\omega|y;\theta^i), \tag{14}$$

which guarantees that the likelihood will not decrease. The objective function for this optimisation problem can be estimated as

$$\frac{1}{M^i}\sum_{j=1}^{M^i} \log f(y|\omega^j;\theta), \tag{15}$$

where ω^j is a simulated value from the MCMC chain $\omega|y;\theta^i$. The simulation is carried out in the same way as the mixture sampler but keeping the parameters fixed. Hence it has the structure

1. Initialize ω.
2. Sample $h|y, \omega, \theta^i$
3. Sample $\omega|y, h$
4. Goto 2.

Then (15) is maximized with respect to the parameters θ using a numerical optimisation routine. Each numerical maximization step keeps the ω^j fixed until convergence is obtained, then a new ω vector is obtained through another simulation step. Between maximizations, it is useful to store the computed Hessian matrix, so that the old values can be used as the starting points for the next optimisation. This trick can reduce the computational cost of the procedure by a large factor (5 to 15-fold reductions are not uncommon).

It is clear that the simulated EM algorithm is not as clean as the Bayesian analysis for it involves two stages of 'approximation' (running the MCMC and then using only M^i replications),

rather than one in the Bayesian case (just running the MCMC). This is more of a problem in simulating the estimators for the Monte Carlo experiments than it is for carrying out estimation using actual data. As a practical matter the convergence of this algorithm is very easy to monitor. One can easily design a visual, interactive estimation procedure in which one increases M as the process converges to less variable parameter estimates. While this type of approach has its merits, the construction of elegant methods for obtaining a simple maximum likelihood estimator that does not require such monitoring and control is still an open area of research.[6]

5 FILTERING VIA SIMULATION

5.1 Introduction

There has been considerable recent work on the development of methods to perform filtering, that is computing features of $h_t|Y_{t-1},\theta$, by simulation. Here $Y_t = (y_1, ..., y_t)$. Leading papers in this field include Gordon, Salmond, and Smith (1993), West (1993) and Muller (1991). We work with a simpler approach which is a special case of a suggestion made by Pitt (1995). Throughout we will assume θ is known. In practice θ will be set to some estimated value, such as the maximum likelihood estimator or the Monte Carlo estimator of the posterior mean.

Working recursively, suppose we possess a random sample from $h_{t-1}|Y_{t-1},\theta$. These will be written as $h_{t-1}^1, ..., h_{t-1}^M$. Then

$$f(h_t|Y_t,\theta) \propto f(y_t|h_t,\theta) f(h_t|Y_{t-1},\theta). \tag{16}$$

Clearly we do not know the solution of the integral $f(h_t|Y_{t-1},\theta) = \int \phi(h_t|h_{t-1},\theta) f(h_{t-1}|Y_{t-1},\theta) dh_{t-1}$, however it can be estimated unbiasedly and consistently, as $M \to \infty$, by

$$\frac{1}{M}\sum_{j=1}^{M} \phi(h_t|h_{t-1}^j,\theta),$$

where $\phi()$ is the density function of $h_t|h_{t-1}^j,\theta$. For the SV model this is Gaussian. This suggests approximating (16) by

$$\widehat{f}(h_t|Y_t,\theta) \propto \sum_{j=1}^{M} f(y_t|h_t,\theta)\phi(h_t|h_{t-1}^j,\theta),$$

which is analytically tractable. Here we will use a first order Taylor expansion of $\log f(y_t|h_t,\theta)$ about $h_{t|t-1} = \mu + \phi\left(\overline{h_{t-1}^j} - \mu\right)$, the average of the future simulations of the log-volatility. Then

$$\log f(y_t|h_t,\theta) \quad = \quad -\frac{1}{2}h_t - \frac{y_t^2}{2}\{\exp(-h_t)\}$$

[6] A recent paper which discusses the asymptotic theory of using the MCMC method in conjunction with the EM algorithm is Chan and Ledolter (1995). They suggest stopping rules for the procedure. Although this type of approach is in its infancy, it seems to hold some promise.

$$\begin{aligned} &\leq -\frac{1}{2}h_t - \frac{y_t^2}{2}\left\{\exp(-h_{t|t-1})(1+h_{t|t-1}) - h_t\exp(-h_{t|t-1})\right\} \\ &= \log g(h_t, h_{t|t-1}, \theta). \end{aligned}$$

Thus we can write

$$\begin{aligned} f(y_t|h_t,\theta)\phi(h_t|h_{t-1}^j,\theta) &\leq g(h_t, h_{t|t-1},\theta)\phi(h_t|h_{t-1}^j,\theta) \\ &= \pi_j\phi(h_t, h_{t-1}^j, y_t, h_{t|t-1},\theta), \end{aligned}$$

where $\phi(h_t, h_{t-1}^j, y_t, h_{t|t-1}, \theta)$ denotes a Gaussian density with

$$h_t \sim \mathsf{N}\left[\mu + \phi h_{t-1}^j + \frac{\sigma_\eta^2}{2}\left\{y_t^2\exp(-h_{t|t-1}) - 1\right\}, \sigma_\eta^2\right] = \mathsf{N}\left[h_{t|t-1}^j, \sigma_\eta^2\right],$$

and the weights are proportional to

$$\pi_j = \exp\left[\frac{1}{2\sigma_\eta^2}\left\{\left(\mu + \phi h_{t-1}^j\right)^2 - h_{t|t-1}^{j2}\right\}\right].$$

The implication of this is that we can sample from $\widehat{f}(h_t|Y_t,\theta)$ by first sampling a h_{t-1}^j with probability proportional to π_j and then sampling a new h_t by proposing from $\mathsf{N}\left(h_{t|t-1}^j, \sigma_\eta^2\right)$ which is then accepted with probability $f(y_t|h_t,\theta)/g(h_t, h_{t|t-1},\theta)$.

Of course this type of procedure is only an approximation, however by selecting a large M this sampler will become arbitrarily accurate.

5.1.1 Application

To illustrate this, we apply these methods to the Sterling/Dollar series, filtering the volatility. Throughout we will employ $M = 2,500$. Similar results were obtained when M fell to $1,000$, although reducing M below that figure created important biases. The results are made conditional of the estimated parameters, which are taken from Table 9 and based on 2,500 sweeps of the integration sampler.

	Mean	MC S.E.	Inefficiency	Covariance & *Correlation*		
$\phi\|y$	0.97611	0.0018015	11.636	0.00014783	*-0.765*	*0.277*
$\sigma_\eta\|y$	0.16571	0.0065029	17.657	-0.00033148	0.0012693	*-0.232*
$\beta\|y$	0.64979	0.0047495	1.4563	0.00030503	-0.00074971	0.008209
Time	97.230	3.8892				

Table 9: *Daily returns for Sterling series. Summaries of reweighted sample of 2,500 sweeps of the integration sampler. The Monte Carlo S.E. of simulation is computed using a block one tenth of the size of the simulation. Italics are correlations rather than covariances of the posterior. Computer time is seconds on a Pentium Pro/200. The other time is the number of seconds to perform 100 complete passes of the sampler.*

The resulting filtered and smoothed estimates of the volatility are given in Figure 7, together with a graph of the absolute values of the returns. The graph shows the expected feature of the

filtered volatility lagging the smoothed volatility. Throughout the sample, the filtered volatility is slightly lower than the smoothed values due to the gradual fall in volatility observed for these series during this period.

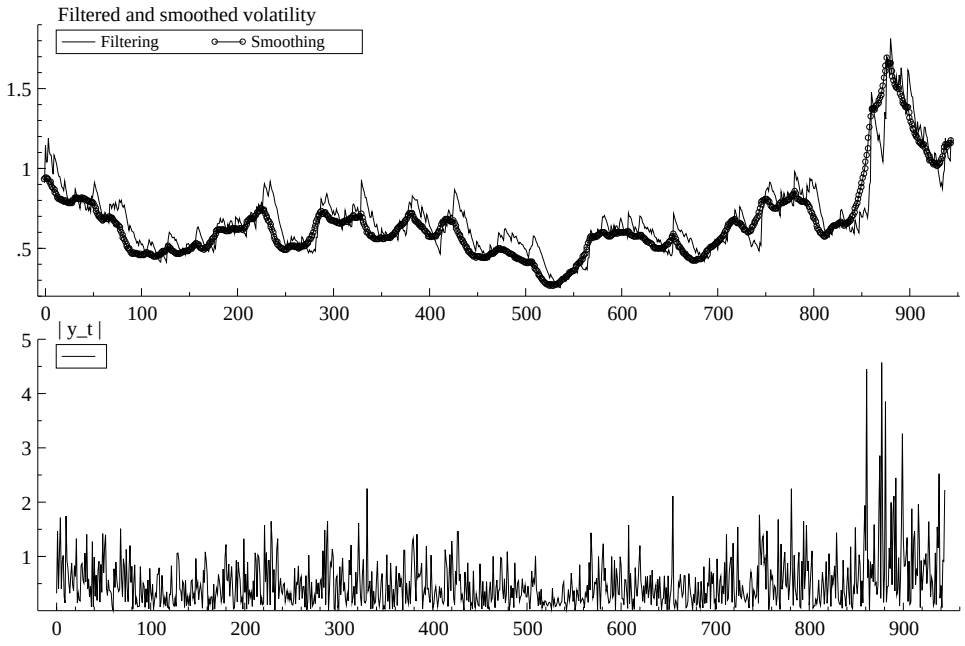

Figure 7: *Top: filtered and smoothed estimate of the volatility* $\exp(h_t/2)$*, computed using* $M = 2000$*. Bottom:* $|y_t|$*, the absolute values of the returns.*

5.2 Diagnostics

Having designed a filtering algorithm it is a simple matter to sample from the one-step-ahead prediction density and distribution function. By definition the prediction density is

$$f(y_{t+1}|Y_t,\theta) = \int f(y_{t+1}|Y_t,h_{t+1},\theta)\, f(h_{t+1}|Y_t,h_t,\theta)\, f(h_t|Y_t,\theta)\, dh_{t+1}dh_t$$

which can be sampled by the method of composition as follows. For each value h_t^j $(j = 1, 2, ..., M)$ from the filtering algorithm, one samples h_{t+1}^j from

$$h_{t+1}^j \sim \mathsf{N}\left\{\mu + \phi\left(h_t^j - \mu\right), \sigma_\eta^2\right\}.$$

Based on these M draws on h_{t+1} from the prediction density, we can estimate the probability that y_{t+1}^2 will exceed the observed y_{t+1}^{o2} :

$$\Pr(y_{t+1}^2 \widehat{\leq} y_{t+1}^{o2}|Y_t) = u_{t+1}^M = \frac{1}{M}\sum_{j=1}^{M} \Pr(y_{t+1}^2 \leq y_{t+1}^{o2}|h_{t+1}^j)\,. \tag{17}$$

For each $t = 1, \ldots, n$, under the null of a correctly specified model u_t^M converges in distribution to independent and identically distributed uniform random variables as $M \to \infty$. This provides a valid basis for diagnostic checking. These variables can be mapped into the normal distribution, by using the inverse of the normal distribution function $n_t^M = F^{-1}(u_t^M)$ to give a standard sequence of independent and identically distributed normal variables, which are then transformed one-step-ahead forecasts normed by their correct standard errors. These can be used to carry out Box-Ljung, normality, and heteroscedasticity tests, among others.