IDENTIFICATION OF BATES STOCHASTIC VOLATILITY MODEL BY USING NON-CENTRAL CHI-SQUARE RANDOM GENERATION METHOD

ShinIchi AIHARA, Arunabha BAGCH and Saikat SAHA

Tokyo University of Science,Suwa , 5000-1 Toyohira Chino,Nagano,Japan
FELab and Department of Applied Mathematics, University of Twente, Enschede, The Netherlands
Department of Electrical Engineering, Linköping University, SE-58183 Linköping, Sweden

ABSTRACT

We study the identification problem for Bates stochastic volatility model, which is widely used as the model of a stock in finance. By using the exact simulation method, a particle filter for estimating stochastic volatility and its systems parameters is constructed. Simulation studies for checking the feasibility of the developed scheme are demonstrated.

***Index Terms*—** Nonlinear filter, Particle filter, Stochastic volatility, Parameter estimation,Chi-square distribution

1. INTRODUCTION

In this paper, we estimate stochastic volatility and unknown system parameters in general stochastic volatility models with jumps as proposed by Bates [4] as given below

$$dS_t = (\mu_S + \lambda_S v_t)S_t dt + \sqrt{v_t} S_t dB_t + S_t dZ_t^J - \lambda m^J S_t dt$$
$$dv_t = \{\kappa(\theta - v_t) + \lambda_v v_t\}dt + \xi\sqrt{v_t}dZ_t \quad (1)$$

where B_t and Z_t are standard Brownian motion processes with correlation ρ, Z_t^J denotes the pure-jump process which contains two components: random-event times and random jump sizes, and is independent of B_t and Z_t. Denoting the intensity of the jump event time as λ and the mean relative jump size as m^J, and by applying Ito's formula to $y_t = \log S_t/S_0$, we have

$$dy_t = (\mu_S - \lambda m^J + (\lambda_s - \frac{1}{2})v_t)dt + \sqrt{v_t}dB_t + dq_t^J, \quad (2)$$

where q_t^J is a compound Poisson process with intensity λ and Gaussian distribution of jump size,i.e., $N(\mu_J, \sigma_J^2)$. Introducing the new Brownian motion

$$\tilde{Z}_t = \frac{1}{\sqrt{1-\rho^2}}(Z_t - \rho B_t), \quad (3)$$

(1) becomes

$$dv_t = \kappa(\theta - v_t)dt + \xi\sqrt{v_t}\sqrt{1-\rho^2}d\tilde{Z}_t$$
$$+\xi\rho(dy_t - (\mu_S - \lambda m^J - (\frac{1}{2} - \lambda_S)v_t)dt - dq_t^J). \quad (4)$$

This work is partially supported by MECSST of Japan under Grant-in-Aid for Research(c) 17560402.

The systems parameters $\mu_S, \kappa, \theta, \xi, \lambda_S$ and λ_v need to be calibrated from the historical data.

Our first problem is to estimate volatility v_t based on observed data y_t. This is the usual filtering problem in signal processing. But the traditional extended Kalman filtering technique does not work in this situation, because a) the model is highly nonlinear, b) model has jumps, c) observation noise contains the system state.

The increasingly popular particle filtering technique method works extremely well in this situation [1]. One common problem of using particle filter in general is to obtain an appropriate importance function. In our model we can actually sample from the optimal importance function which is in itself a remarkable fact [2, 3]. One serious difficulty of using particle filtering is the generation of the systems particles. Discrete approximations may lead to negative samples. To circumvent this problem, the exact simulation method has been recently proposed in [5] . Using this algorithm we can now formulate the particle filter where the transition and optimal importance functions are given by non-central chi square density functions.

Estimating unknown parameters in stochastic volatility models is known to be very difficult. The MLE does contain the usual difficult problem of multiple local maxima. Even the outputs from EM algorithm do not converge well, because of the shape of the chi-square probability density.

Although the usual augmented state approach does not always behave properly as shown in [1], we use this augmented state approach, because the bounds for unknown parameters are easily set *a-priori* and hence the estimate of the degrees of freedom of the non-central chi-square probability is well controlled in the required region for fitting commodity data.

2. EXACT PARTICLE FILTERING

2.1. Exact sampling

In order to implement the particle filter, the original system is usually approximated to the discrete-time one by using the Euler method. This approximation easily causes bias from the original continuous system. For example, the discrete-time volatility process v_k often becomes negative. To avoid this bias, we propose the exact sampling method which is developed by Broadie and Kaya [5] for simulating the Heston process. In particle filter we generate samples from the optimal importance function $p(v_{t_2}|v_{t_1}, y_{t_2}, y_{t_1})$. Now we shall present the exact sampling procedure. For simplicity we consider the time interval $t_1 < t_2$ and set the following assumption: At most one jump occurs in this time interval and we observe y_{t_2} and y_{t_1}.

1. Exact sampling from $p(v_{t_2}|v_{t_1}, y_{t_2}, y_{t_1})$
From (2), the volatility process v_{t_2} is represented by

$$v_{t_2} = \tilde{v}_{t_1} + \int_{t_1}^{t_2} \tilde{\kappa}(\tilde{\theta} - v_s)ds + \int_{t_1}^{t_2} \xi\sqrt{v_s}\sqrt{1-\rho^2}d\tilde{Z}_s, \quad (5)$$

where

$$\begin{aligned} \tilde{v}_{t_1} &= v_{t_1} + \rho\xi\{y_{t_2} - y_{t_1} \\ &\quad -(\mu_S - \lambda m^J)(t_2 - t_1) - \Delta q_{t_1}^i\} \\ \tilde{\kappa} &= \kappa - \frac{\rho\xi}{2} + \xi(\rho\lambda_S - \lambda_v) \\ \tilde{\theta} &= \frac{\kappa\theta}{\tilde{\kappa}}. \\ \Delta q_{t_1}^i &= \text{jump sample from } q_{t_1}^J \text{ for } t_1 < t < t_2. \end{aligned}$$

Now assuming that $\tilde{v}_{t_1} \geq 0$, we find that the transition law of v_{t_2} given v_{t_1}, y_{t_1} and y_{t_2} is expressed as the non-central chi-square random variable $\chi_d^2(\lambda_\chi)$ with d degree of freedom and non-centrality parameter λ_χ,

$$\frac{\xi^2(1-\rho^2)(1-e^{-\tilde{\kappa}(t_2-t_1)})}{4\tilde{\kappa}}\chi_d^2(\lambda_\chi), \quad (6)$$

where

$$d = \frac{4\tilde{\theta}\tilde{\kappa}}{\xi^2(1-\rho^2)}$$

and

$$\lambda_\chi = \frac{4\tilde{\kappa}e^{-\tilde{\kappa}(t_2-t_1)}}{\xi^2(1-\rho^2)(1-e^{-\tilde{\kappa}(t_2-t_1)})}\tilde{v}_{t_1}.$$

Hence by using MATLAB code ”ncx2rnd.m”, we can get a sample v_{t_2} .

For the case that $\tilde{v}_{t_1} < 0$, which may occur when v_{t_1} is very small, we need to adjust above procedure as described in the next step.

2. $\tilde{v}_{t_1} < 0$ case
We reconstruct the data for $t_1 < \tau_1 \leq t_2$ such that

$$\begin{aligned} &\Delta y(\tau_1 - t_1) \\ &= \frac{\rho\xi\{y_{t_2} - y_{t_1} - (\mu_S - \lambda m^J)(t_2 - t_1) - \Delta q_{t_1}^j\}}{t_2 - t_1} \\ &\quad \times(\tau_1 - t_1) \end{aligned}$$

and τ_1 satisfies

$$v_{t_1} + \Delta y(\tau_1 - t_1) \geq 0.$$

where it is always possible to find τ_1, because

$$\lim_{\tau_1 \to 0}\{v_{t_1} + \Delta y(\tau_1 - t_1)\} = v_{t_1} > 0.$$

By using the step 1., we obtain $v_{\tau_1} > 0$. Now we check whether $v_{\tau_1} + \Delta y(t_2 - \tau_1)$ is non-negative or not. If this is non-negative, we repeat the step 1. again and obtain $v_{t_2} > 0$.If not we need to find τ_2:

$$v_{\tau_1} + \Delta y(\tau_2 - \tau_1) \geq 0.$$

Repeat above procedure, we finally obtain $v_{t_2} > 0$. For $\tilde{\tilde{v}}_{t_2} < 0$ case, we should use the same procedure mentioned here.

2.2. Construction of probability density function

If we use the Euler scheme, the generated sample becomes conditionally Gaussian. However in the exact sampling scheme , the processes generated are governed by the non-central chi-square distribution. Although the explicit function form of this distribution is not possible, we can numerically evaluate the pdf by using the MATLAB code, ”ncx2pdf.m”.

- $p(v_{t_2}|v_{t_1}, y_{t_2}, y_{t_1})$ form
Noting that the jump size $U^s_.$ is Gaussian with mean μ_J and variance σ_J^2, we have

$$\begin{aligned} &p(v_{t_2}|v_{t_1}, y_{t_2}, y_{t_1}) \\ &\quad \times \text{pdf of } \left\{\frac{\xi^2(1-\rho^2)(1-e^{-\tilde{\kappa}(t_2-t_1)})}{4\tilde{\kappa}}\chi_d^2(\tilde{\lambda}_\chi)\right\} \\ &\quad + e^{-\lambda(t_2-t_1)}\lambda(t_2 - t_1) \\ &\quad \times \int_{-\infty}^{\infty} \text{pdf of } \left\{\frac{\xi^2(1-\rho^2)(1-e^{-\tilde{\kappa}(t_2-t_1)})}{4\tilde{\kappa}}\right. \\ &\quad\quad \left. \times\chi_d^2(\tilde{\lambda}_\chi - \frac{4\tilde{\kappa}e^{-\tilde{\kappa}(t_2-t_1)}\rho}{\xi(1-\rho^2)(1-e^{-\tilde{\kappa}(t_2-t_1)})}U^s)\right\} \\ &\quad \times \frac{1}{\sqrt{2\pi\sigma_J^2}}\exp(-\frac{(U^s - \mu_J)^2}{2\sigma_J^2})dU^s \end{aligned} \quad (7)$$

where

$$\tilde{\lambda}_\chi = \frac{4\tilde{\kappa}e^{-\tilde{\kappa}(t_2-t_1)}}{\xi^2(1-\rho^2)(1-e^{-\tilde{\kappa}(t_2-t_1)})}\{v_{t_1} + \rho\xi\{y_{t_2} - y_{t_1} - (\mu_S - \lambda m^J)(t_2 - t_1)\}\}$$

In (7), the first term implies that we have no jump and the second term is caused by the jump size $U^s \in N(\mu^J, \sigma_J^2)$. Furthermore in the second term we need to calculate the Gaussian integral. We may use some numerical procedure to calculate this but the best choise is still an open problem.

- $p(v_{t_2}|v_{t_1}, y_{t_1})$ form
  It follows from (5) that

$$p(v_{t_2}|v_{t_1}, y_{t_1}) = \text{pdf of } \frac{\xi^2(1-e^{-(\kappa-\xi\lambda_v)(t_2-t_1)})}{4(\kappa-\xi\lambda_v)}\chi_d^2(\lambda_\chi^v), \quad (8)$$

  where

$$d = \frac{4\theta\kappa}{\xi^2},$$

  and

$$\lambda_\chi^v = \frac{4(\kappa-\xi\lambda_v)e^{-(\kappa-\xi\lambda_v)(t_2-t_1)}}{\xi^2(1-e^{-(\kappa-\xi\lambda_v)(t_2-t_1)})}v_{t_1}.$$

- $p(y_{t_2}|y_{t_1}, \int_{t_1}^{t_2} v_s ds)$ form
  In this case, we easily get

$$\begin{aligned} p(y_{t_2}|y_{t_1}, \int_{t_1}^{t_2} v_s ds) &= \frac{1-e^{-\lambda(t_2-t_1)}\lambda(t_2-t_1)}{\sqrt{2\pi(1-\rho^2)\int_{t_1}^{t_2} v_s ds}} \\ &\times \exp[-\frac{1}{2(1-\rho^2)\int_{t_1}^{t_2} v_s ds}\{y_{t_2} - (y_{t_1} \\ &+(\mu_S - \lambda m^J - \frac{\kappa\theta\rho}{\xi})(t_2-t_1) \\ &\quad -(\frac{1}{2} - \frac{\kappa\rho}{\xi} + \rho\lambda_v - \lambda_S)\int_{t_1}^{t_2} v_s ds \\ &+\frac{\rho}{\xi}(v_{t_2} - v_{t_1}))\}^2] \\ &+\frac{e^{-\lambda(t_2-t_1)}\lambda(t_2-t_1)}{\sqrt{2\pi((1-\rho^2)\int_{t_1}^{t_2} v_s ds + \sigma_J^2)}} \\ &\times \exp[-\frac{1}{2((1-\rho^2)\int_{t_1}^{t_2} v_s ds + \sigma_J^2)}\{y_{t_2} \\ &-(y_{t_1} + (\mu_S - \lambda m^J - \frac{\kappa\theta\rho}{\xi})(t_2-t_1) \\ &\quad -(\frac{1}{2} - \frac{\kappa\rho}{\xi} + \rho\lambda_v - \lambda_S)\int_{t_1}^{t_2} v_s ds + \mu^J \\ &+\frac{\rho}{\xi}(v_{t_2} - v_{t_1}))\}^2] \quad (9) \end{aligned}$$

2.3. Exact particle filter algorithm

Now we can perform the exact particle filter. The weight $w_\cdot^{(i)}$ is given by the following recursive form:

$$w_{t_k}^{(i)} = w_{t_{k-1}}^{(i)} \frac{p(y_{t_k}|y_{t_{k-1}}, \int_{t_{k-1}}^{t_k} v_s^{(i)} ds)p(v_{t_k}^{(i)}|v_{t_{k-1}}^{(i)}, y_{t_{k-1}})}{p(v_{t_k}^{(i)}|v_{t_{k-1}}^{(i)}, y_{t_k}, y_{t_{k-1}})}. \quad (10)$$

The algorithm steps are:

- At each time t_k, using $y_{t_k}, y_{t_{k-1}}$, we generate particles $v_{t_k}^{(i)}$ from the algorithm (1) and calculate $p(v_{t_k}^{(i)}|v_{t_{k-1}}^{(i)}, y_{t_k}, y_{t_{k-1}})$ given by (7).
- Using $v_{t_k}^{(i)}, v_{t_{k-1}}^{(i)}$, we calculate $p(v_{t_k}^{(i)}|v_{t_{k-1}}^{(i)}, y_{t_{k-1}})$ given by (8).
- Using the above generated $\int_{t_{k-1}}^{t_k} v_s^{(i)} ds$ in the algorithm (2) and the observation data $y_{t_k}, y_{t_{k-1}}$, we calculate $p(y_{t_k}|y_{t_{k-1}}, \int_{t_{k-1}}^{t_k} v_s^{(i)} ds)$ from (9).
- Update the weight $w_{t_k}^{(i)}$ given by (10).
- In the above steps, we may use the resampling method, if needed.

3. PARAMETER IDENTIFICATION

Before constructing the parameter identification procedure, we will discuss about the noncentral chi-square probability density function. If the degrees of freedom d defined by

$$d = \frac{4\kappa\theta}{\xi^2}$$

takes its value greater than 2, the log likelihood function is negative and convex and "zero" is not attainable. However if $d \le 2$, the point "zero" is attainable and the function form of the log likelihood function becomes concave and its value becomes positive near zero point. Since in most practical applications in finance $d/2 << 1$, the log likelihood functional is not easy to be selected as the optimal cost. Furthermore if we use the EM-algorithm for finding the MLE for model parameters, in the maximization step we seek the next step optimal parameter value for fixing the state as the smoothed value with some fixed parameters and then the cost function tends to move to the infinity. This implies that we need the strict upper and lower bounds for unknown parameters for using EM-algorithm. That is, the value of the degrees of freedom should be fixed in proper region. However in practice, it is not possible to guess its value in advance. For avoiding these difficulties, we propose the usual filtering algorithm for identifying the system parameters.

3.1. Parallel filtering

For identifying the system parameters, we construct the parallel filtering algorithm for fixing the unknown parameter defined as

$$\alpha = [\kappa\ \theta\ \xi\ \mu_S\ \rho\ \lambda\ \mu_J\ \sigma_J].$$

To perform the particle filter for v_k with the fixed α, we assume that

$$\alpha \in U(\text{uniform distribution with known upper and lower bounds}).$$

To start our particle filter, we generate the initial pair particles

$$(v_0^{(i)}, \alpha^{(i)}) \in \mathcal{N} \times U,\ i = 1, 2, \cdots, N.$$

At each time step t_k we generate $v_{k+1}^{(i)}$ with the fixed $\alpha^{(i)}$ from (5) and we can calculate (10). We then construct the estimates

$$\hat{\alpha}_{t_k} = \sum_{i=1}^{N} \alpha^{(i)} w_{t_k}^{(i)}, \text{ and } \quad \hat{v}_{t_k} = \sum_{i=1}^{N} v_{t_k}^{(i)} w_{t_k}^{(i)}.$$

4. SIMULATION STUDIES

We set the following parameters in Table 1 with their estimates: Here we set $dt = 1/252$ and $M = 2000$. The lower

Table 1. Model parameters and their estimates($T = 1$)

	κ	θ	μ_S	ρ	ξ
True	0.864	1.100	0.060	-0.150	2.100
Estimated	0.830	1.037	0.057	-0.147	2.059
	λ	μ^J	σ^J	λ_v	λ_S
True	0.100	-0.020	0.250	0.188	0.372
Estimated	0.096	-0.19	0.239	0.166	0.344

and upper bounds for parameters are set as given in Table.2.

Table 2. Lower and upper bounds of model parameters

	κ	θ	μ_S	ρ	ξ
Upper bound	0.993	1.265	0.069	-0.113	2.415
Lower bound	0.648	0.825	0.045	-0.173	1.575
	λ	μ^J	σ^J	λ_v	λ_S
Upper bound	0.115	-0.015	0.287	0.216	0.428
Lower bound	0.075	-0.023	0.188	0.141	0.279

Now we show the observation data y_t and its log price .

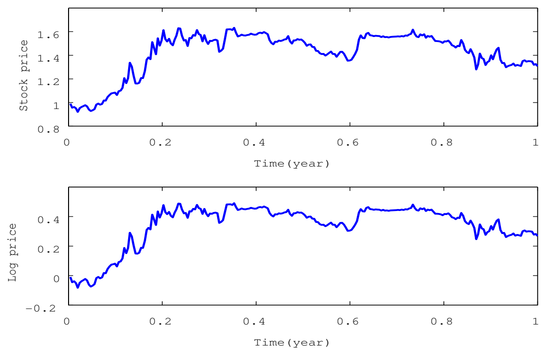

Fig. 1. Observation data y_t

In Fig.2, the true and estimated v_t is demonstrated.

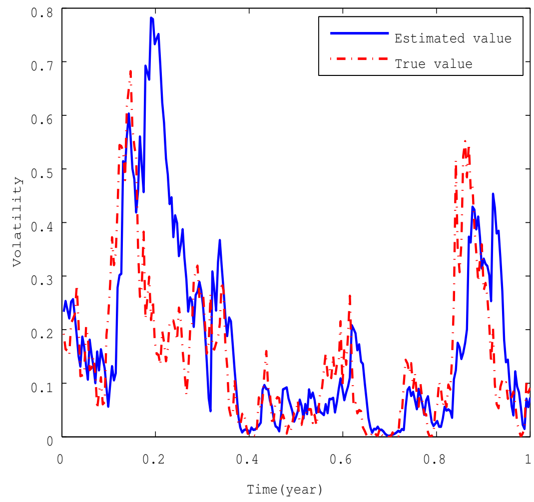

Fig. 2. True and estimated v_t

5. CONCLUSIONS

We developed a particle filter algorithm for estimating a stochastic volatility and its model parameters. It has been shown that the exact simulation technique used here works well even for the case that the degrees of freedom of a non-central chi-square distribution is less that 2. Noting that the estimation procedure developed here is an on-line scheme, it is also possible to apply this scheme to a mean-variance hedging problem in finance.

6. REFERENCES

[1] S. Aihara, A. Bagchi, and S. Saha. Estimating volatility and model parameters of stochastic volatility models with jumps using particle filter. *Proc. of 17th IFAC World Congress*, 17, 2008.

[2] A. Doucet and S. Godsill and C. Andrieu On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistic and Computing*, 10, p.197-208,2000.

[3] RO. Cappé and E. Moulines and T .Rydén. *Inference in Hidden Markov Models*. Springer Science+Busimess Media Inc.NY, 2005.

[4] R. Cont and P. Tankov. *Financial Modelling With Jump Processes*. Chapman & Hall/CRC, Boca Raton, 2004.

[5] M.Broadie and O. Kaya Exact simulation od Stochastic Volatility and Other Affine Jump Diffine Processes. *Operations Research*, 54, p.217-231,2006.