Available online at www.isas.org.in/jisas
JOURNAL OF THE INDIAN SOCIETY OF AGRICULTURAL STATISTICS 68(3) 2014 343-350

Stochastic Volatility Model Fitting using Particle Filter: An Application

Bishal Gurung, Prajneshu and Himadri Ghosh
Indian Agricultural Statistics Research Institute, New Delhi

Received 31 August 2012; Revised 06 January 2014; Accepted 26 March 2014

SUMMARY

In this article, we study the Stochastic volatility (SV) model in which volatility is an unobservable variable following some stochastic process. Procedure for estimation of parameters of this model using Particle Filter (PF), which is a powerful Sequential Monte Carlo technique, is discussed. To this end, relevant computer program is developed in MATLAB, Ver. 7.4 software package. As an illustration, All-India data of month-wise total export of Basmati rice during the period April, 2003 to June, 2013 is considered. Comparative study of the fitted SV model vis-à-vis Exponential Generalized autoregressive conditional heteroscedastic (EGARCH) model is carried out by computing various measures of goodness of fit. Subsequently, forecasting performances of SV and EGARCH models are also compared using several statistical measures. Finally, it is shown that SV model fitted through Particle filter performed better than EGARCH model for the data under consideration.

Keywords: Exponential generalized autoregressive conditional heteroscedastic model, Heteroscedasticity, Particle filter, Stochastic volatility model.

1. INTRODUCTION

The main interest in time-series analysis has been to obtain a model which could explain effectively the mean behaviour of data (Box *et al*. 2008). However, recently concerns about volatility or variance in the data have been raised because changes or patterns in volatility are quite often observed in real datasets. As emphasized by Jaffee (2005), volatility seems to be the norm rather than an exception in International markets due to structure of trade, climatic conditions, and rapidity with which producers can respond to price changes. The export of many agricultural commodities shows a great degree of fluctuations, caused by delays between production decisions and delivery to the market. Forecasting of volatile data is generally carried out by using the Generalized autoregressive conditional heteroscedastic (GARCH) model. However, there are some limitations of this methodology, such as its inability to capture empirical properties, like asymmetry in the conditional variance. To this end, an extension of GARCH model, viz. Exponential GARCH (EGARCH) model may be employed. However, the assumption that volatility is deterministic and is driven only by past observable variables is not appropriate. Alternatively, volatility may also be modelled as an unobservable component following some latent stochastic process, such as an autoregressive model. Models of this kind are called Stochastic volatility (SV) models (Taylor 1994). Bali (2000) demonstrated the superiority of a two-factor SV model over GARCH model by carrying out a study of volatile interest rate changes of U.S.A. treasury bills. Carnero *et al*. (2001) showed that SV models capture, in a more appropriate way, the main empirical properties often observed in volatile time-series data vis-à-vis GARCH models. A thorough review of volatility forecasting by different models is given in Poon and Granger (2003).

Corresponding author: Prajneshu
E-mail address : prajneshu@yahoo.co.in

The SV model can be expressed in terms of a state space model with non-Gaussian errors. Therefore, Kalman filter is not appropriate for efficient estimation of parameters of this model because the filtered states are not the best predictor. As it is difficult to obtain theoretically the conditional density function of state due to intractability of marginal distribution of measurements, next best option is to apply an iterative procedure. Monte Carlo (MC) methods have become one of the standard tools and have allowed the Bayesian paradigm to be applied to sophisticated models. As pointed out by Andrieu *et al*. (2010), although asymptotic convergence of Markov chain MC (MCMC) algorithms is ensured under weak assumptions, their performance is unreliable when the proposal distributions that are used to explore the space are poorly chosen and/or if highly correlated variables are updated independently. The authors have also built efficient high-dimensional proposal distributions using Sequential MC (SMC) methods.

Accordingly, in this article, our aim is to investigate the promising technique of Particle filter (PF), which is capable of assigning stochastic conditional probabilities of unobserved states through importance sampling methodology (Ristic *et al*. 2004). Further, PF method is employed for estimation of parameters of SV model. To this end, relevant computer program is developed in MATLAB, 2007a software package. As an illustration, SV model is applied to describe volatile All-India data of monthly export of Basmati rice during the period April, 2003 to June, 2013. Finally, performances of SV and EGARCH models are compared in respect of their capabilities for modelling as well as forecasting for hold-out data using various measures of goodness of fit.

2. SOME PRELIMINARIES

In this section, Stochastic volatility (SV) model along with its estimation procedure using Particle Filtering (PF) technique is briefly discussed.

2.1 Stochastic Volatility (SV) Model

SV model is more realistic and flexible than GARCH model, since it essentially involves two random processes, one for the observations and the other for volatility. For a discrete univariate time-series data $\{y_t, t = 1, ..., T\}$, it is given by

$$y_t = \varepsilon_t \sigma_t, \; t = 1, ..., T, \tag{1}$$

where y_t are observations, ε_t is a white noise process with variance σ_ε^2 and σ_t^2 is corresponding volatility. Further, $h_t^* = log\,\sigma_t^2$ follows an AR(1) process with Gaussian white noise and is unobserved. In terms of h_t^*, Eq.(1) can be written as

$$y_t = \varepsilon_t exp\,(h_t^*/2), \;\; h_{t+1}^* = \alpha + \varphi h_t^* + \eta_t,$$

$$\eta_t \sim IID(0, \sigma_\eta^2), \;\; |\varphi| < 1, \tag{2}$$

where $|\varphi| < 1$ implies stationarity of h_t^*. The parameter φ measures persistence of shocks to volatility. When φ is close to unity and σ_η^2 is close to 0, evolution of volatility over time is very smooth. The variance of log-volatility process σ_η^2 measures uncertainty about future volatility. From Eq.(2), we get

$$(h_{t+1}^* - \alpha^*) = \varphi(h_t^* - \alpha^*) + \eta_t, \tag{3}$$

where $\alpha^* = \alpha/(1 - \varphi)$. So, from Eq.(3):

$$y_t = exp(h_t^* - \alpha^*)/2\} exp(\alpha^*/2)\varepsilon_t, \tag{4}$$

which can be written as

$$y_t = \sigma_* exp(h_t/2)\varepsilon_t, \; h_t = h_t^* - \alpha^*, \;\; \sigma_* = exp\,(\alpha^*/2). \tag{5}$$

Estimate of σ_* may be used to estimate α^*, which along with estimate of φ yields estimate of α. Thus, Eq.(1) can be written as

$$y_t = \sigma_* \; exp(h_t/2)\varepsilon_t, \; h_{t+1} = \varphi h_t + \eta_t. \tag{6}$$

2.2 Particle Filtering

A general state-space model is defined by the following state and measurement equations:

$$\alpha_{t+1} = \boldsymbol{f}_t\,(\alpha_t, \boldsymbol{w}_t) \tag{7}$$

$$X_t = \boldsymbol{h}_t(\alpha_t, \boldsymbol{v}_t), \tag{8}$$

where $\boldsymbol{f}_t$ is a function describing evolution of the state and $\boldsymbol{h}_t$ is a function mapping state vector to the observations. The quantities $\{\boldsymbol{w}_t\}$ and $\{\boldsymbol{v}_t\}$ are zero mean noise with

$$E[\boldsymbol{v}_t\, \boldsymbol{v}'_t] = R_t \text{ and } E[\boldsymbol{w}_t\, \boldsymbol{w}'_t] = \boldsymbol{Q}_t,$$

α_t are the unobservable state variables with initial density $p(\alpha_0)$ and X_t are the observations. The above model can be characterised in terms of its probabilistic description via the state transition density $p(\alpha_t|\alpha_{t-1})$ and the observation density $p(x_t|\alpha_t)$. The main statistical problem related to this type of state space model is to estimate state of the dynamic system α_t in some optimal manner from all the noisy observations. Complete solution to this problem can be given by the conditional density $p(\alpha_0|X_T)$, $T = 1, 2, ..., N$. For linear Gaussian state-space models, Kalman filter (KF) is one such method of getting the predicted, filtered and smoothed value of α_t. But, in the case of nonlinear or non-Gaussian state space models, KF does not work well. One plausible way is to use the Extended KF (EKF), which linearizes original nonlinear filter dynamics around current estimated state and then proceeds as in the linear case. However, EKF has two drawbacks, viz. Linearization can produce highly unstable filters, and derivation of Jacobian matrices often leads to significant implementation difficulties. Further, as the SV model can be transformed to a linear state space form with non-Gaussian errors, therefore EKF is also not appropriate for estimating the parameters of this model. Therefore, the promising new methodology called Particle filtering (For details, see *e.g.* Polson *et al.* 2008 and Ristic *et al.* 2004), which approximates the posterior probability of states using a large number of particles with associated weights, is employed.

The PF is a SMC algorithm grounded in particle representation, and it can be considered as a generalization of KF for general state-space models. The posterior distribution is derived from Bayes theorem, and the underlying volatility can be estimated. Instead of giving a single estimate for the filter or the smoother as in KF, PF method provides particles with associated weights to approximate the conditional density. Particles and weights are updated sequentially along with the state evolution, when new observations become available. Some salient aspects of PF method are included in Annexure-1 and relevant codes, developed in MATLAB, Ver. 7.4 software package, are appended as Annexure-2.

3. AN ILLUSTRATION

Month-wise total export of Basmati rice from India during the period April, 2003 to June, 2013, obtained from the website (*www.indiastat.com*), is considered. Out of total 123 data points, first 106 data points corresponding to the period April, 2003 to January, 2012 are used for model building and the remaining 17 data points, *i.e.* from February, 2012 to June, 2013 are used for validation purpose. A perusal of data indicates presence of volatility at several time-epochs. EViews, Ver. 4 software package is employed for fitting ARIMA and GARCH family of models. In the first instance, best ARIMA model is selected on the basis of minimum Akaike information criterion (AIC) and Bayesian information criterion (BIC) values given by

$$AIC = Tlog(\sigma^2) + 2(p + q + 1) \quad (9)$$

$$BIC = Tlog(\sigma^2) + (p + q + 1)logT, \quad (10)$$

and the results are reported in Table 1.

Table 1. Parameter estimates along with their standard errors for fitted ARIMA(2,1,0) model

Parameter	Estimate	Standard error
Intercept	14.00	8.75
AR1	–0.36	0.10
AR2	–0.16	0.09

However, the ACF of squared residuals of fitted ARIMA (2, 1, 0) model, reported in Table 2, are found to be quite high. Further, ARCH-LM test statistic value of 3.69 at lag 13 is found to be significant at 5% level, thereby indicating presence of volatility in the data. But it is not reasonable to apply ARCH model of such a high order. So, in the second instance, GARCH family of models, viz. GARCH, EGARCH, and Integrated GARCH (IGARCH) are fitted. On the basis of minimum AIC and BIC values, selected AR(1)-EGARCH(1,1) model is obtained as

$$\underset{(52.87)}{Y_t = 316.71} + \underset{(0.05)}{0.91}Y_{t-1} + \varepsilon_t,\ \varepsilon_t = h_t^{1/2}\eta_t,$$

where the figures within brackets ( ) indicate corresponding standard errors and h_t satisfies the variance equation

$$\ln(h_t) = \underset{(0.05)}{-0.17} + \underset{(0.13)}{0.48}\ln(h_{t-1}) + \underset{(0.03)}{0.08}|\varepsilon_{t-1}/\sqrt{h_{t-1}}| + \underset{(0.18)}{0.98}\,\varepsilon_{t-1}/\sqrt{h_{t-1}}.$$

To study the appropriateness of fitted EGARCH model, ACF of standardized residuals and squared standardized residuals are computed. It is found that, in both the situations, ACF is significant at 5% level, thereby reflecting that the mean and variance equations are not correctly specified. Moreover, fitted model is not able to capture properly the volatility present at various time-epochs.

Table 2. Autocorrelation functions (ACF) and Partial autocorrelation functions (PACF) of the squared residuals for fitted ARIMA model

Lag	ACF	PACF	Q-Stat	Probability
1	–0.005	–0.005	0.003	
2	–0.035	–0.035	0.131	
3	–0.076	–0.077	0.757	0.384
4	–0.155	–0.158	3.376	0.185
5	–0.028	–0.039	3.463	0.326
6	–0.164	–0.190	6.476	0.166
7	–0.200	–0.255	11.000	0.051
8	0.078	0.001	11.698	0.069
9	0.044	–0.033	11.919	0.103
10	0.006	–0.111	11.924	0.155
11	0.096	0.007	13.004	0.162
12	0.137	0.119	15.248	0.123
13	0.458	0.437	30.237	0.001
14	–0.076	–0.081	18.933	0.090
15	–0.069	0.022	19.527	0.108
16	–0.015	0.065	19.555	0.145
17	–0.052	–0.012	19.896	0.176
18	0.024	0.087	19.970	0.222
19	–0.066	0.041	20.533	0.248
20	–0.176	–0.185	24.551	0.138

So, finally SV model is fitted to the data through PF method by using 1000 particles. Further, h_t is estimated and one-step ahead prediction error along with its mean squared error, as well as prediction error decomposition form of the likelihood are obtained. The hyperparameters are denoted by θ. Subsequently, parameters are estimated to construct optimal solution for θ that maximizes the likelihood. The fitted model, using the code given in Annexure-2, is obtained as

$$\log(y_t^2) = 2.87 + h_t + \xi_t,\ \ h_t = 0.85 + \eta_t, Var\{\eta_t\} = 0.66.$$

To study the appropriateness of fitted SV model, ACF of standardized residuals and squared standardized residuals are again computed. It is found that now, in both the situations, these are non-significant at 5% level, thereby implying that the mean and variance equations are correctly specified. The AIC and BIC values for fitted SV model are respectively computed as 683.78 and 691.68, which are lower than the corresponding values, viz. 707.34 and 713.65 for fitted AR(1)-EGARCH(1,1) model. Further, relative performance of fitted SV model vis-à-vis GARCH model is evaluated using the Mean square error (MSE) criterion, defined as

$$MSE = \sum_{i=1}^{N} \{Y_i - \hat{Y}_i\}^2 / N.$$

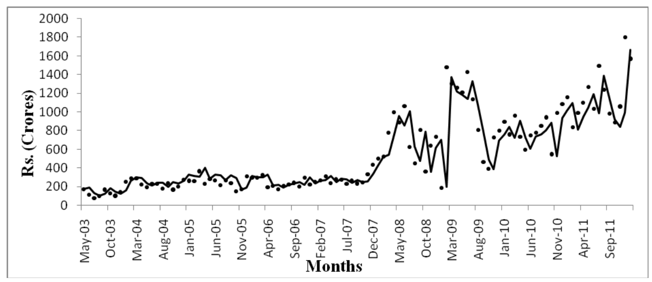

Fig. 1(a). Fitted EGARCH model along with data points

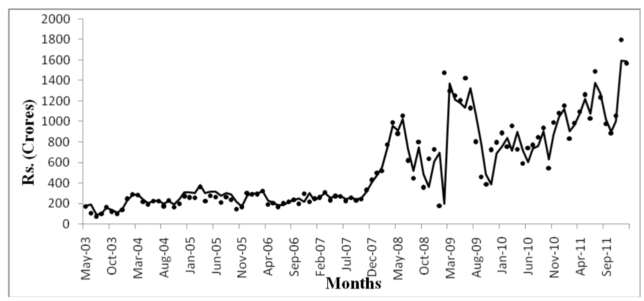

Fig. 1(b). Fitted Stochastic volatility model along with data points

MSE values for the fitted SV and EGARCH models are respectively computed as 960126.65 and 968926.87. All these imply that SV model has performed better than AR(1)-EGARCH(1,1) model for describing the volatile data under consideration. The graphs of fitted SV and EGARCH models along with data points are exhibited in Figs. 1(a) and 1(b), which indicate that fitted SV model is able to capture the volatilities present in the data in a better way than EGARCH model.

3.1 Forecasting Performance

In this sub-section, forecasting abilities of the fitted EGARCH and SV models are evaluated. For 17 hold-out data points corresponding to All-India monthly export of Basmati rice from February, 2012 to June, 2013, one-step ahead forecasts are computed and the same are reported in Table 3.

Table 3. One-step ahead forecasts of export data (in Rs. Crore)

Months	Actual	EGARCH model	SV model
Feb.,'12	1291.68	1630.94 (353.71)*	1604.70 (350.22)
Mar.,'12	1661.60	1384.57 (476.33)	1319.89 (375.66)
Apr.,'12	1240.95	1468.77 (548.71)	1697.25 (498.44)
May,'12	1566.36	1394.81 (434.67)	1268.14 (360.01)
Jun.,'12	1818.17	1396.14 (397.32)	1600.09 (305.22)
Jul.,'12	1644.50	1667.67 (328.54)	1856.97 (250.22)
Aug.,'12	1711.10	1678.31 (376.09)	1679.80 (298.72)
Sep.,'12	1326.78	1642.18 (554.91)	1747.74 (430.55)
Oct.,'12	1060.29	1461.84 (432.87)	1355.69 (390.96)
Nov.,'12	1160.88	1157.88 (345.88)	1083.84 (276.31)
Dec.,'12	1718.17	1101.59 (323.87)	1486.45 (289.12)
Jan.,'13	1875.72	1445.04 (396.86)	1754.96 (250.52)
Feb.,'13	2075.62	1761.90 (259.18)	1915.68 (227.87)
Mar.,'13	2530.55	1935.58 (487.10)	2119.60 (443.28)
Apr.,'13	2590.12	2264.89 (391.73)	2583.68 (272.31)
May,'13	2417.03	2486.19 (379.16)	2644.45 (302.77)
Jun.,'13	2749.79	2417.25 (381.69)	2467.88 (329.80)

*Figures within brackets ( ) indicate corresponding standard errors.

It may be noted from Table 3 that, for fitted SV model, an attractive feature is that all the actual values lie within the prediction intervals corresponding to Estimates ± Standard errors, whereas for fitted AR(1)-EGARCH(1, 1) model, as many as 5 actual values corresponding to the time-epochs, viz. June, '12, December, '12, January, '13, February, '13, and March, '13 lie outside the prediction intervals. Another point worth noting is that widths of all the prediction intervals for fitted SV model are less than those for fitted AR(1)-EGARCH(1, 1) model. Further, performance of fitted models is also compared on the basis of one-step ahead Mean square prediction error (MSPE), Mean absolute prediction error (MAPE) and Relative mean absolute prediction error (RMAPE), given respectively as

$$MSPE = \sum_{i=0}^{N-1} \{Y_{T+i+1} - \hat{Y}_{T+i+1}\}^2 / N$$

$$MAPE = \sum_{i=0}^{N-1} \{| Y_{T+i+1} - \hat{Y}_{T+i+1}\} / N$$

$$RMAPE = \sum_{i=0}^{N-1} \{| Y_{T+i+1} - \hat{Y}_{T+i+1} | / Y_{T+i+1}\} \times 100 / N.$$

The MSPE, MAPE and RMAPE values for fitted SV model are respectively computed as 74702.96, 241.39 and 15.14, which are found to be much lower than the corresponding ones for fitted AR(1)-EGARCH(1, 1) model, viz. 114871.50, 288.03 and 16.81 respectively. Thus, SV model has performed much better than EGARCH model for forecasting purpose also.

To sum up, SV model is appropriate for modelling as well as forecasting the volatile data under consideration.

4. CONCLUDING REMARKS

In this article, utility of Particle filter for fitting Stochastic volatility model is highlighted. As an illustration, the proposed procedure is applied for modelling and forecasting of volatile All-India monthly Basmati rice export data. These types of studies would go a long way in assisting planners to take appropriate policy decisions as they are based on sound statistical footing. Attempts are being made towards development of efficient procedures of more advanced Stochastic volatility models through Particle filter method and will be reported separately in due course of time.

ACKNOWLEDGEMENTS

Authors are grateful to the referee for valuable comments.

REFERENCES

Andrieu, C., Doucet, A. and Holenstein, R. (2010). Particle Markov chain Monte Carlo methods. *J. Roy. Statist. Soc.*, **B72,** 269-342.

Angelidis, T., Benos, A. and Degiannakis, S. (2004). The use of GARCH models in VaR estimation. *Stat. Meth.*, **1**, 105-128.

Bali, T.G. (2000). Testing the empirical performance of stochastic volatility models of the short term interest rate. *J. Finan. Quant. Anal.,* **35**, 191-215.

Barndorff-Nielsen, O.E., Nicolato, E. and Shephard, N.G. (2002). Some recent developments in stochastic volatility modelling. *Quant. Fin.*, **2**, 11-23.

Bollerslev, T. (1986). Generalized autoregressive conditional heteroscedasticity. *J. Econ.*, **31**, 307-327.

Box, G.E.P., Jenkins, G.M. and Reinsel, G.C. (2007). *Time-Series Analysis: Forecasting and Control.* 3rd Ed. Dorling Kindersley, New Delhi.

Broto, C. and Ruiz, E. (2004). Estimation methods for stochastic volatility models: A survey. *J. Econ. Sur.*, **18**, 613-649.

Durbin, J. and Koopman, S.J. (2001). *Time Series Analysis by State Space Models*. Oxford University Press.

Engle, R.F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica*, **50**, 987-1007.

Ghosh, H., Iquebal, M.A. and Prajneshu (2006). On mixture nonlinear time-series modelling and forecasting for ARCH effects. *Sankhya*, **68**, 111-129.

Harvey, A.C., Ruiz, E. and Shephard, N.G. (1994). Multivariate stochastic variance models. *Rev. Econ. Stud.*, **61**, 247-264.

Polson, N.G., Stroud, J.R. and Muller, P. (2008). Practical filtering with sequential parameter learning. *J. Roy. Statist. Soc.*, **B 70**, 413-428.

Poon, S.H. and Granger, C.W.J. (2003). Forecasting volatility in financial markets: A review. *J. Econ. Lit.,* **41**, 478-539.

Ristic, B., Arulampalam, S. and Gordon, N. (2004). *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, Boston, U.S.A.

Taylor, S.J. (1994). Modelling stochastic volatility: A review and comprehensive study. *Math. Fin.,* **4**, 183-204.

ANNEXURE-1

Some Salient Aspects of PF Methodology

To approximate the conditional density of α_t, viz. $p(\alpha_t|\alpha_{T-1}, X_T)$ given previous states α_{T-1}, $T = 1, 2, .., N$ along with past and present data X_T, $T = 1, 2, ..., N$, particle filtering (PF) introduces a sequential importance density $q(\alpha_t|\alpha_{T-1}, X_T)$, where it is easier to sample from $q(\alpha_t|\alpha_{T-1}, X_T)$ rather than from $p(\alpha_t|\alpha_{T-1}, X_T)$. The joint conditional density of α_T given X_T is

$$p(\alpha_T|X_T) = \frac{p(x_t \mid \alpha_T, X_{T-i})\, p(\alpha_T \mid X_{T-1})}{p(x_t \mid X_{T-1})} \quad \text{(A.1)}$$

$$= \frac{p(x_t \mid \alpha_T)\, p(\alpha_t \mid \alpha_{t-1})}{p(x_t \mid X_{T-1})}\, p(\alpha_{T-1} \mid X_{T-1})$$

$$\propto \frac{p(x_t \mid \alpha_T)\, p(\alpha_t \mid \alpha_{t-1})}{q(\alpha_t \mid \alpha_{T-1} X_t)}\, q(\alpha_t \mid \alpha_{T-1}, X_t)$$

$$\times\ p(\alpha_{T-1} \mid X_{T-1}).$$

Suppose that particle approximation of $p(\alpha_{T-1}|X_{T-1})$ is given as $\sum_{j=1}^{M} w_{t-1}^{(j)} \delta(\alpha_{T-1} - \alpha_{T-1}^{(j)})$ and $\alpha_t^{(j)}$ is a sample from $q(\alpha_t \mid \alpha_{T-1}^{(j)}, X_t)$, for $j = 1, 2, ..., M$. Then the particle approximation of $p(\alpha_T|X_T)$ is

$$p(\alpha_T \mid X_T) \approx \sum_{j=1}^{M} \frac{p(x_t \mid \alpha_t^{(j)})\, p(\alpha_t^{(j)} \mid \alpha_{t-1}^{(j)})}{q(\alpha_t^{(j)} \mid \alpha_{T-1}^{(j)}, X_T)}$$

$$\times\, \delta(\alpha_t - \alpha_t^{(j)})\, w_{t-1}^{(j)}\, \delta(\alpha_{T-1} - \alpha_{T-1}^{(j)}) \quad \text{(A.2)}$$

$$\approx \sum_{j=1}^{M} w_t^{(j)} \delta(\alpha_T - \alpha_T^{(j)}),$$

where

$$w_t^{(j)} = \frac{p(x_t \mid \alpha_t^{(j)})\, p(\alpha_t^{(j)} \mid \alpha_{t-1}^{(j)})}{q(\alpha_t^{(j)} \mid \alpha_{T-1}^{(j)} X_T)}\, w_{t-1}^{(j)} \quad \text{and}$$

$$\alpha_T^{(j)} = \{\alpha_{T-1}^{(j)}, \alpha_t^{(j)}\}.$$

Using Sequential importance sampling (SIS) particle filter, particles and associated weights $\{\alpha_t^{(j)}, w_t^{(j)}\}_{j=1}^{M}$ can be obtained sequentially. However, complication with SIS is degeneracy, where after a few iterations, only few particles have consequential weights. Thus, considerable computational effort would be spent updating particles whose contribution to the approximation of $p(\alpha_T|X_T)$ is negligible. Resampling is designed to solve this problem by removing particles with small weights, focussing instead on particles with large weights. This resampling step involves generating a new particle set $\{\alpha_t^{(j*)}\}_{j=1}^{M}$ by sampling with replacement M times from the original set $\{\alpha_t^{(j)}\}_{j=1}^{M}$, so that $p\{\alpha_t^{(j*)} = \alpha_t^{(j)}\} = w_t^{(j)}$. A noteworthy issue is the choice of importance density. An appropriate choice is to use the prior $p(\alpha_t \mid \alpha_{t-1}^{j})$. A general particle filter draws $\{\alpha_t^{(j)}\}_{j=1}^{M}$ using SIS filter, and resample $\{\alpha_t^{(j*)}\}_{j=1}^{M}$ when degeneracy occurs.

ANNEXURE-2

Code for Estimation of Parameters of SV Model using PF Method in MATLAB, Ver. 7.4 Software Package

```
function [Pparms,RLIK]=mainchis(obs,INTPARM,Maxit,Nopart)
Pparms=zeros(Maxit,3);
RLIK=zeros(Maxit,1);
NOITER=0;
for i=1:Maxit
obs_full=obs;
f=PartFilt(INTPARM, obs_full, Nopart);
Param=Estim(s, obs_full, Nopart)
Pparms(i,:)=Param;
 function f=PartFilt(INTPARM, obs, Nopart)
phi=INTPARM(1);
Q=INTPARM(2);
meann=INTPARM(3);
mu=0;
SIGMA=Q/(1-phi^2);
f(:,1)=randn(Nopart, 1)*sqrt(SIGMA)+mu;
wt=randn(Nopart, n)*sqrt(Q);
for t=1:n
p=phi*f(:,t)+wt(:,t);
w=exp((obs(t)-a(t)*p-meann)/2).*exp(-1*exp(obs(t)-a(t)*p-meann)/2);
f(:,t+1)=RESAM(p,w,Nopart);
end
function RESAMDATA=RESAM(data,weight,NofSample)
n=max(size(data));
re_ind=rand(1,NofSample);
cmwt=cumsum(weight)/sum(weight);
for k=1:NofSample
st=(re_ind(k)>cmwt(1:n-1));
RESAMDATA(k)=data(sum(st)+1);
end
RESAMDATA=RESAMDATA';
function loglike=comp_loglike(x,obs,parmvec,n)
phi=parmvec(1);
Q=parmvec(2);
meann=parmvec(3);
mu=0;
SIGMA=Q/(1-phi^2);
loglike=-1*(log(SIGMA)+(x(1)-mu)^2/SIGMA+n*log(Q)+sum((x(2:n+1)-phi*x(1:n)).^2)/Q+sum(exp(obs-
x(2:n+1)-meann)-(obs-x(2:n+1)-meann)))/2;
```