Project 1: ACMA830

An unscented Kalman filter and smoother for volatility extraction on Heston Model

Vatsal Desai | vatsal_desai@sfu.ca

December 16 2022

Abstract

This report tries to replicate the results obtained by Junye Li in the article: 'An unscented Kalman smoother for volatility extraction: Evidence from stock prices and options' from the journal Computational Statistics and Data Analysis (2013) which applies a Gaussian smoothing algorithm to scaled uscented trasnformation using a Kalman filter to extract volatility from the Heston Stock Price model. The simulation and real data study shows that both stock prices and option prices are needed for accurately capturing volatility dynamics.

Contents

1. Introduction

In the state-space model framework, Bayesian optimal smoothing, also known as belief inference, refers to statistical methodology that can be used to infer state estimate using all information, which is available not only in the past and at the current, but also in the future. Optimal smoothing is closely related to optimal filtering, which makes inference based on information only available in the past and at the current time.

At the core of financial econometrics is volatility estimation. Volatility pervades almost everywhere in financial markets. For example, it is used in option pricing, in portfolio allocation to control and manage risks, and in computation of risk adjusted returns for comparison of relative performance of various financial investments. Time-varying/stochastic volatility is well documented in empirical studies. There are mainly two modeling approaches for volatility. One is the class of ARCH/GARCH models (Engle, 1982; Bollerslev, 1986), where conditional volatility is a deterministic function of past volatility and return innovations, and the other is the stochastic volatility models (Shephard, 2005), which assume that volatility is unobservable and is driven by a different random process. In the past thirty years, the diffusion process has become a common tool used to model dynamics of financial data. The diffusion stochastic volatility models (Hull and White, 1987; Heston, 1993) may be the most popular ones both in academia and in practice because of their flexibility in pricing derivatives and risk management. These models have the following general form under a given filtered probability space $\{\Omega, \mathcal{F}, \mathbb{F}, \mathbb{P}\}$ where $\mathbb{F} = \{\mathcal{F}_t : t > 0\}$.

$$dS_t = \mu_t S_t dt + \sqrt{V_t} S_t dW_t$$
$$dV_t = \kappa(\theta - V_t)dt + \sigma V_t dZ_t$$

where S_t is the stock price process and V_t is the volatility process. W_t and Z_t are two standard Brownian motions, maybe mutually correlated, with $Corr^{\mathbb{P}}[W_t, Z_t] = \rho$. κ is the speed of mean reversion for volatility, θ is the long run mean of volatility and, σ (at times referred by η or ξ as well) is the volatility of volatility.

However, in the case of stochastic volatility models, there are two main problems that persist while performing any statistical analysis.
1. Variances of stochastic variables are state dependent.
2. The pricing formula for derivatives is not linear.

These problems combined mean that most of the conventional methods such as the standard Kalman filter and smoother are rarely applicable to these models. Even the non-linear Kalman filters, such as the extended Kalman filter and the unscented Kalman filter do not use all the information available in the past and upto current time. Another artifact to note while using these methods is that when the system is highly non-linear and high dimensional (consist of more sources of variability), the extended Kalman filter tends to perform rather poorly.

Here, I have tried to use the unscented transformation approach for non-linear Gaussian system. We first use an unscented Kalman filter to approximate latent states in the model for a discretized version of the Heston stochastic volatility model, and then we smooth the data by assuming Gaussian distribution in the latent variables without making any assumption about the observation states.

This paper implements a simulation study using the Heston stochastic volatility model. I found that the unscented Kalman filter and smoother both perform almost identically when only stock price are supplied as observations, but according to the original paper by Li (2005), the smoother improves significantly when option prices are also taken into account.

I have also applied the same algorithm to the real data on S&P 500 index, and have compared errors when only applying a filter with filter and smoother.

The rest of the paper discusses about the scaled unscented transformation, the unscented Kalman filter as a direct application of scaled unscented transformation and its effectiveness on simulated data as well as real world data.

2. The scaled unscented transformation

The scaled unscented transformation is a method to calculate statistics of any random variable which goes through a non-linear transformation. One of the most direct implementation of this, is an unscented Kalman filter.

a scaled transformation generally takes mean of the original Gaussian (like extended Kalman filter) and transforms it through the non-linear function and approximates a Gaussian around the transformed mean to approximate a linear function. Instead now in case of scaled unscented transformation (in unscented Kalman filter), the mean and a cluster of sigma points around the mean are all transformed via the non-linear function and an approximation is made around those points. Also these sigma points are assigned weights based on how much of the variability present in the transformation is explained by each of the sigma points.

suppose we have a non-liner function f transforming x to y,

$$y = f(x)$$

with mean of $x = \bar{x}$ and co-variance of $x = P_x$, and if x has dimension L, then by SUT we can generate 2L+1 sigma points ($\mathcal{X}_\rangle$) around mean of x and transform them via f, to map new Gaussian around mean of y.

$$\begin{aligned}
\mathcal{X}_0 &= \bar{x} \\
\mathcal{X}_i &= \bar{x} + \left[\sqrt{(L+\lambda)P_x}\right]_i, \quad i = 1, \ldots, L \\
\mathcal{X}_i &= \bar{x} - \left[\sqrt{(L+\lambda)P_x}\right]_i, \quad i = L+1, \ldots, 2L
\end{aligned}$$

where $\lambda = \alpha^2(L + \|) - L$ is a scaling parameter, α determines how spread apart sigma points are around the mean (in our case, it is set to 10^-3), $\| = 3 - L$ is the second scaling parameter.

These sigma points are then propagated through the non-linear function f and mean and co-variance of these transformed sigma points are approximated as weighted sample mean and co-variance of these points.

such as:

$$\begin{aligned}
\mathcal{Y}_i &= f(\mathcal{X}_i) \\
and&, \\
\bar{y} &= \sum_{i=0}^{2L} w_i^{(m)} \mathcal{Y}_i \\
P_y &= \sum_{i=0}^{2L} w_i^{(c)} (\mathcal{Y}_i - \bar{y})(\mathcal{Y}_i - \bar{y})^T
\end{aligned}$$

where the mean weights $w_i^{(m)}$ and covariance weights $w_i^{(c)}$ are given by:

$$\begin{aligned}
w_0^{(m)} &= \frac{\lambda}{L+\lambda} & w_0^{(c)} &= \frac{\lambda}{L+\lambda}(1 - \alpha^2 + \beta) \\
w_i^{(m)} &= w_i^{(c)} = \frac{1}{2(L+\lambda)}, & i &= 1, 2, \ldots, 2L
\end{aligned}$$

here, β is the co-variance correction parameter and for normal distribution, it is taken as 2, and L is taken as 3 for our model (observation state, observation noise and hidden process noise).

Now, in next step, we use these updated mean and co-variance values, and generate new sigma points around that for next time step. Repeating this procedure, at each time step we predict mean and co-variance, and then update them for next time step. This generates a time series allowing us to predict points at all times in future.

therefore, if at time t-1 we have our observed state x_{t-1}, observation noise w_{t-1} and hidden process noise v_{t-1}, we can define,

$$x_{t-1} = [x_{t-1} \; w_{t-1} \; v_{t-1}]^T$$

and $\bar{x}_{t-1|t-1} = \mathbb{E}[x_{t-1}]$, $P^x_{t-1} = diag(P^x_{t-1|t-1} \; R^w_{t-1} \; R^v_{t-1})$,

Then we can generate 2L+1 (=7 in our case) sigma points around this mean as:

$$\begin{aligned}
\mathcal{X}_0 &= \bar{x}_{t-1|t-1} \\
\mathcal{X}_i &= \bar{x}_{t-1|t-1} + \left[\sqrt{(L+\lambda)P_{t-1}}\right]_i \; for \; i = 1, \ldots, L \\
\mathcal{X}_i &= \bar{x}_{t-1|t-1} - \left[\sqrt{(L+\lambda)P_{t-1}}\right]_i \; for \; i = L+1, \ldots, 2L
\end{aligned}$$

And for predicting values of observed process, we can use scaled unscented transformation on these sigma points, for this time prediction step:

$$\begin{aligned}
\mathcal{X}_{t|t-1} &= f(\mathcal{X}_{t-1}) \\
& and, \\
\bar{\S}_{t|t-1} &= \sum_{i=0}^{2L} w_i^{(m)} \mathcal{X}_{i,t|t-1} \\
P^x_{t|t-1} &= \sum_{i=0}^{2L} w_i^{(c)} (\mathcal{X}_{i,t|t-1} - \bar{\mathcal{X}}_{t|t-1})(\mathcal{X}_{i,t|t-1} - \bar{\mathcal{X}}_{t|t-1})^T
\end{aligned}$$

And for our hidden process, the measurement update step would be:

$$\begin{aligned}
\mathcal{Y}_{t|t-1} &= g(\mathcal{Y}_{t-1}) \\
& and, \\
\bar{\dagger}_{t|t-1} &= \sum_{i=0}^{2L} w_i^{(m)} \mathcal{Y}_{i,t|t-1} \\
P^y_{t|t-1} &= \sum_{i=0}^{2L} w_i^{(c)} (\mathcal{Y}_{i,t|t-1} - \bar{\mathcal{Y}}_{t|t-1})(\mathcal{Y}_{i,t|t-1} - \bar{\mathcal{Y}}_{t|t-1})^T \\
P^{xy}_{t|t-1} &= w_i^{(c)} (\mathcal{X}_{i,t|t-1} - \bar{\mathcal{X}}_{t|t-1})(\mathcal{Y}_{i,t|t-1} - \bar{\mathcal{Y}}_{t|t-1})^T \\
\bar{x}_{t|t} &= \bar{x}_{t|t-1} + P^{xy}_{t|t-1}(P^y_{t|t-1})^{-1}(y_t - \bar{\dagger}_{t|t-1}) \\
P^x_{t|t} &= P^x_{t|t-1} - \left[P^{xy}_{t|t-1}(P^y_{t|t-1})^{-1}\right] \cdot P^y_{t|t-1} \cdot \left[P^{xy}_{t|t-1}(P^y_{t|t-1})^{-1}\right]^T
\end{aligned}$$

3. Simulation study

In this section, I shall demonstrate a simulation study carried out to implement the above mentioned techniques, to apply SUT via a Kalman filter to the Heston stochastic volatility model and then applying a Gaussian smoother to the hidden volatility process without making any assumptions about the observed stock price process. Afterwards, there are some comparisons between the hidden volatility predicted by Kalman filter and smoother with the actual simulated values.

To run this simulation, the continuous time Heston volatility model was discretized as below:

$$V_{t+1} = V_t + \kappa(\theta - V_t)\Delta t + \sigma\sqrt{V_t \Delta t}\mathcal{Z}_t^1$$
$$logS_{t+1} = logS_t + \left(r - \frac{V_t}{2}\right)\Delta t + \sqrt{V_t \Delta t}\mathcal{Z}_t^2$$

where, V_t is the volatility at time t and S_t is the stock price at time t. κ is the speed of mean reversion of volatility, θ is the long run average value (mean level) of volatility, σ is the variance of volatility, Δt is the uniform time difference between each time steps $(t_i - t_{i-1})$, r is the risk free interest rate, $\mathcal{Z}_t^1$ is a standard normal variable and if correlation between volatility and stock price process is given by ρ then, $\mathcal{Z}_t^2 = \rho\mathcal{Z}_t^1 + \sqrt{1-\rho^2}\mathcal{Z}_t^3$, and $\mathcal{Z}_t^3$ is also a standard normal variable.

In this simulation study, we have first taken some default values for each of these parameters, which are taken from their long term average values as seen from empirical studies.

Initial stock price $S_0 = 100$, initial volatility $V_0 = 0.04$, speed of mean reversion $\kappa = 3$, long run average value of volatility $\theta = 0.04$, variance of volatility $\sigma = 0.3$, correlation between volatility and stock price $\rho = -0.6$,time horizon $\tau = 1$ *year* total number of time steps $Nsteps = 252$, and time difference between each time step $\Delta t = 1/252$ are supplied, and simulated stock price and volatility values are generated. I have also assumed a constant risk free interest rate $r = 0.05$ over this entire time horizon.

The parameters for Kalman filter used in the simulation study are: dimensions $L = 3$, scaling parameter $\alpha = 10^{-3}$, scaling parameter $k = 3 - L = 0$, covariance correction parameter $\beta = 2$.

Then we apply SUT via a Kalman filter and smoother and results of this are shown as below. (Blue lines show simulated values and red lines show predicted values.)

As we can see from this, the smoothed values fit a lot better to the real values for volatility process, although, without considering options, just stock price alone as the observation state is not enough to truly capture the nature of the volatility as also stated by Junye Li (2013).

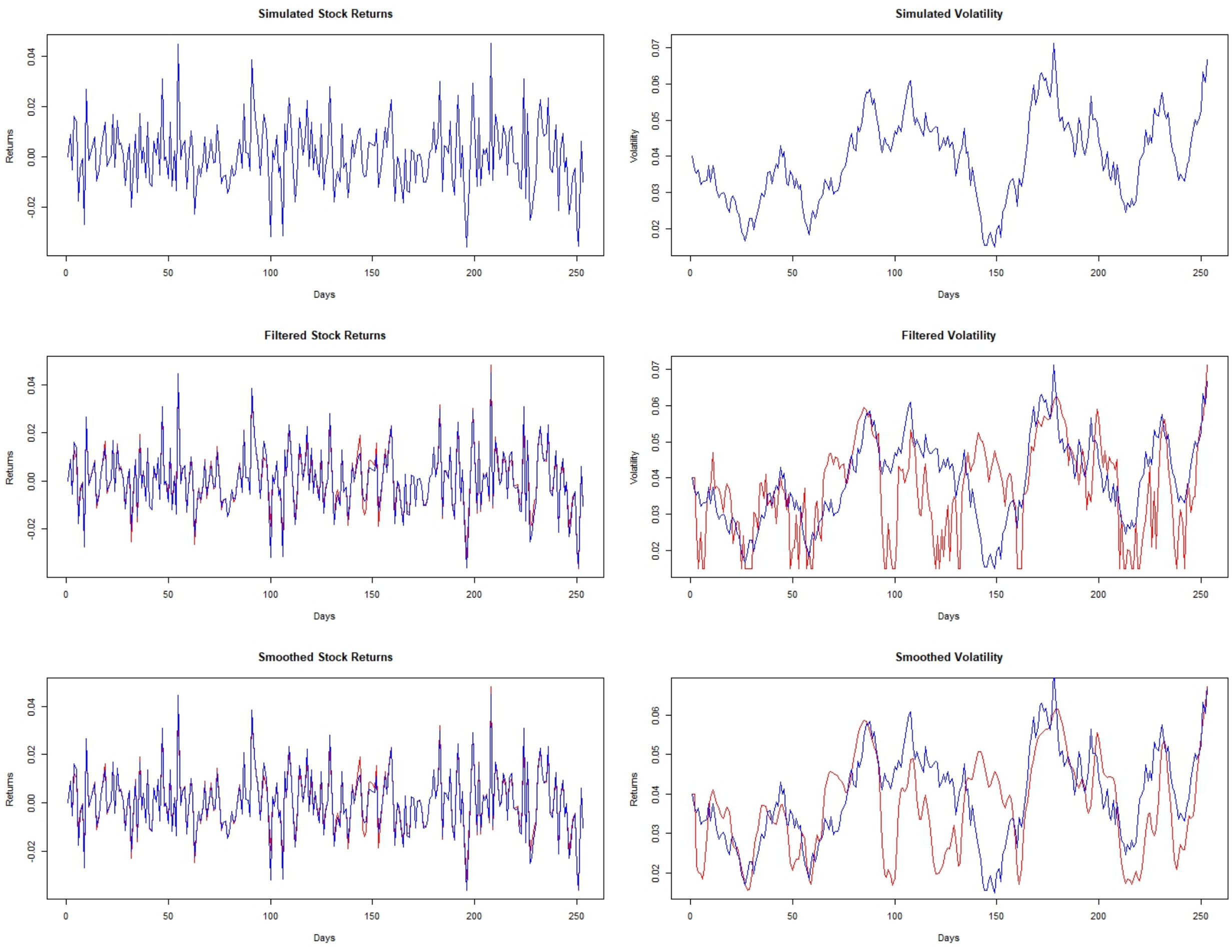

Figure 1: Simulated, filtered and smoothed volatilities for $\Theta = (\kappa, \theta, \sigma, \rho) = (3, 0.04, 0.3, -0.6)$

Next, we shall try to see the effect of speed of mean reversion and variance of volatility on our filter and see the difference at different values of these parameter to see how close our fit is to the model.

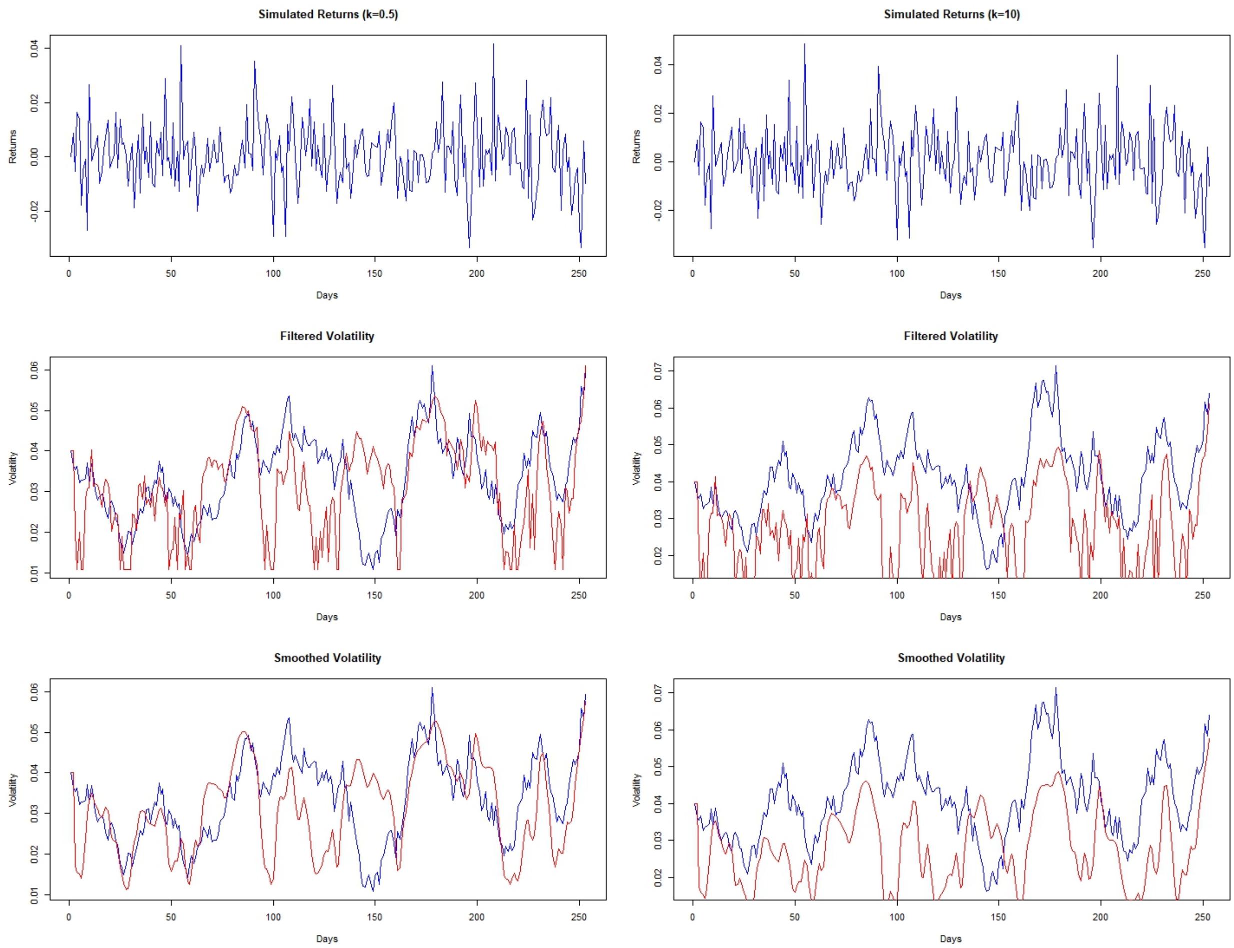

Figure 2: Filtered and smoother volatilities for $\kappa = 0.5$ and $\kappa = 10$

We can see that when the speed of mean reversion is very slow ($\kappa = 0.5$), the model performs much better, due to the slow movement in volatility process, the transformation is able to approximate a better Gaussian around mean, but when speed of mean reversion is high ($\kappa = 10$), the model performs poorly. Even though, our filtered and smoothed volatility values capture the shape of the simulated volatility values, the scale is much lower, due to the filtered values taking longer to revert back to mean as compared to simulated ones.

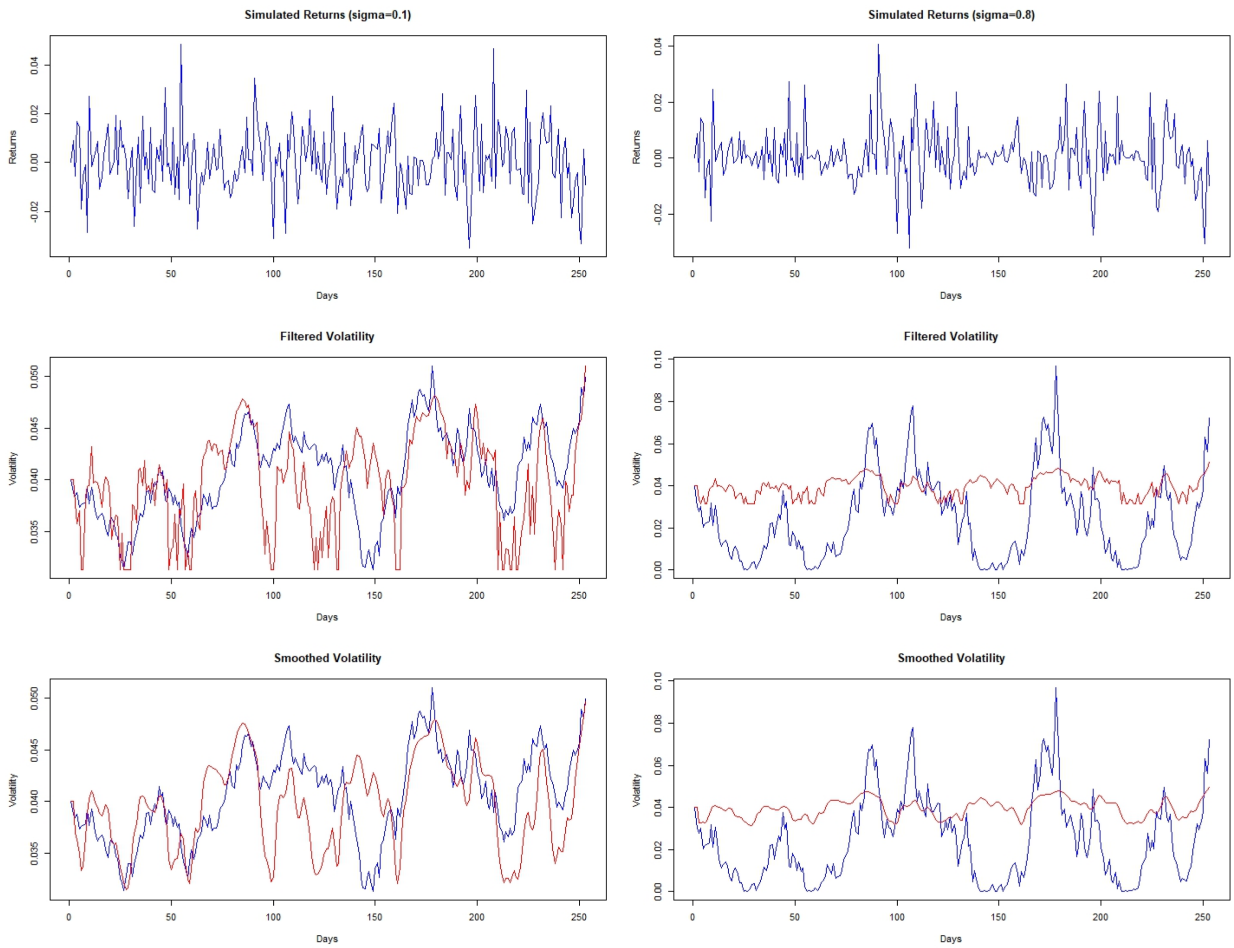

Figure 3: Filtered and smoother volatilities for $\sigma = 0.1$ and $\sigma = 0.8$

As we can see, when the variance of the volatility process is very low ($\sigma = 0.1$), the model performance is much better, but when the variance of the volatility is very high ($\sigma = 0.8$), the model performs poorly, and gives a rather mean value of the process and does not capture the true nature of volatility process.

4. Empirical applications

Here, I have used the same algorithm to filter volatility values for S&P 500 index. The real B.S implied volatility values are taken from CBOE (Chicago Board Options Exchange) and S&P 500 data is taken from actual index values on a daily frequency for 252 days starting from 3rd Jan to 31st December 1996 (252 days).

for this period from the data, correlation between volatility and stock price processes was $\rho = -0.3062$, risk free interest rate $r = 5.3$ were supplied to the model. All other parameter being same as the simulation study, below is the comparison given on the real world data.

As we can see from the graph above, the model is somewhat able to capture the nature of Black-Scholes implied volatility, but without incorporating options, the returns seem all over the place as well as whenever the direction of the volatility changes rapidly, the model lags behind, and then overshoots at next time-step.

As mentioned by Junye Li (2013) in his paper, this can be improved by incorporating option prices into the model and extra observation states.

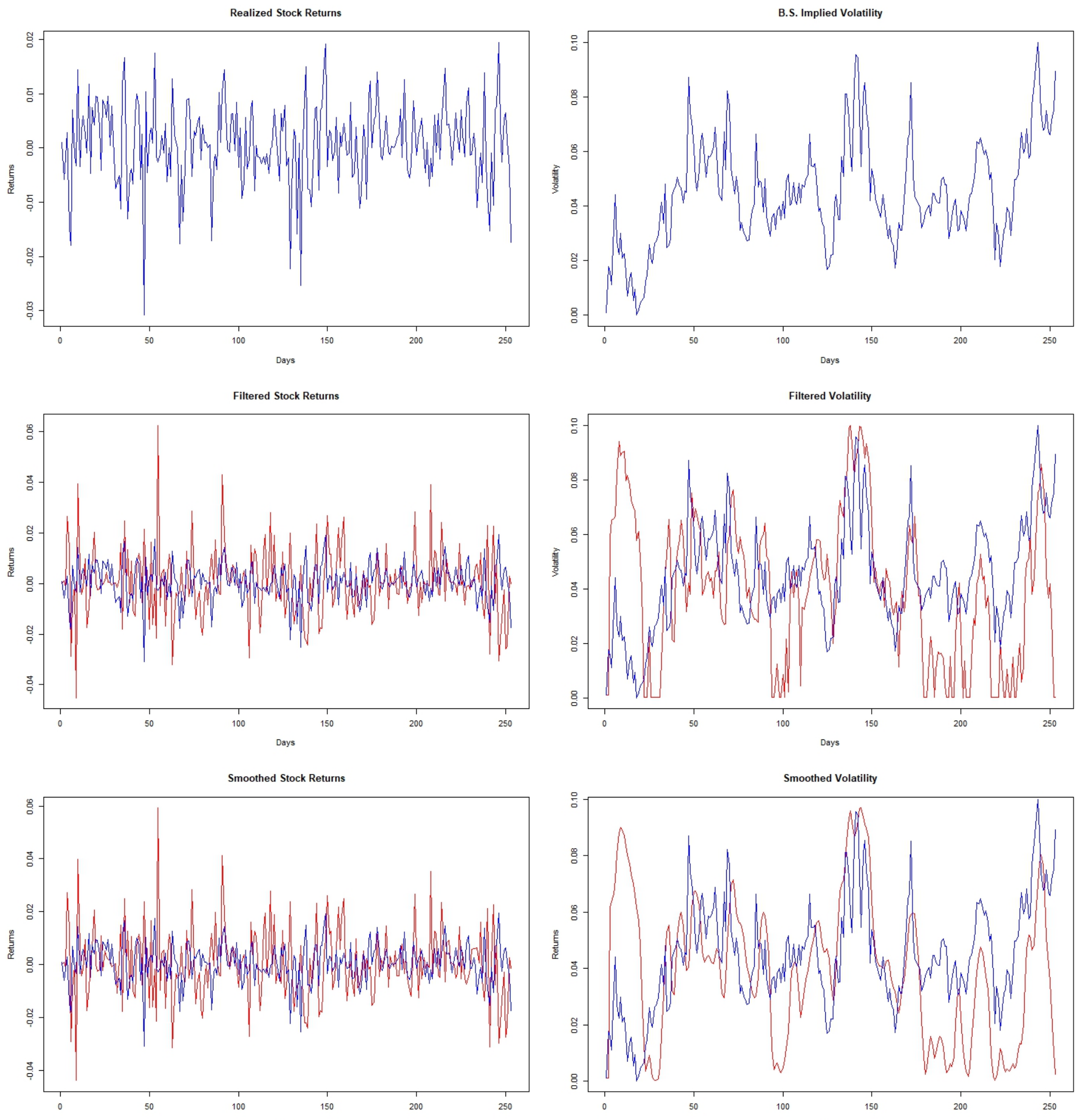

Figure 4: Black–Scholes Implied, filtered and smoothed volatilities for S&P 500 index for the year of 1996

We can also compare the difference between implied volatility and predicted values from the filter and smoother. Which would show the amplitude and directions of the error terms in our model.

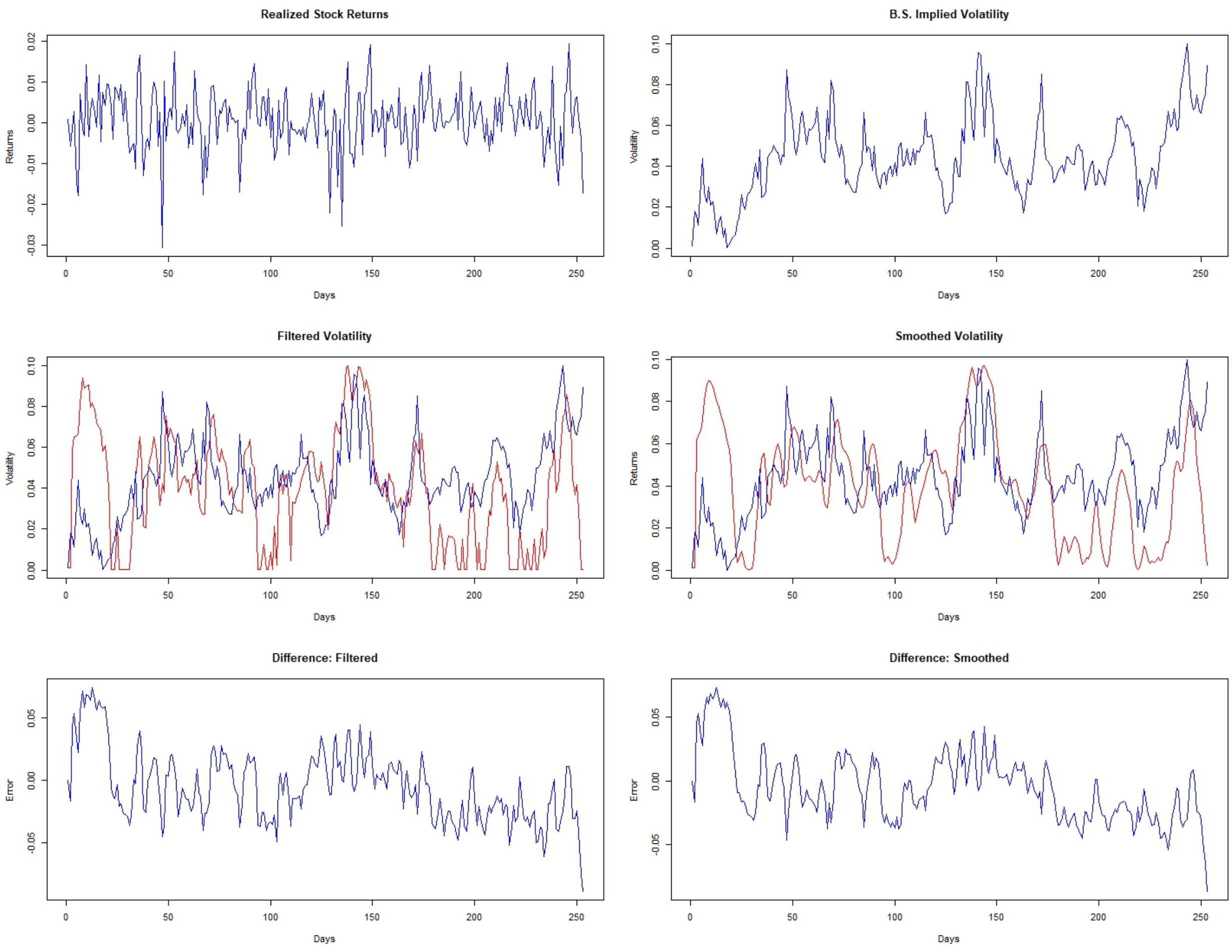

Figure 5: Panel 1: Realized stock returns and B.S. implied volatility for 1996, Panel 2: Filtered and Smoothed volatilities generated from our model, Panel 3: Absolute errors between BS implied volatility vs filtered and smoother volatility from our model

As we can clearly see, the error terms get slightly reduced in amplitude in case of UKS as compared to UKF. We can also see that in low volatility regions, the error is positive and in high volatility regions, the error terms are negative, which is consistent with our mean reverting asymptotic normality assumption.

5. Conclusion

This paper shows a smoothing algorithm for volatility extraction based on the scaled unscented transformation for the Heston stochastic volatility model using an unscented Kalman filter and smoother. Simulation study and empirical application both suggest that in order to truly capture the dynamics of volatility both stock prices and option prices are required. With the rise in use of sequential Monte Carlo methods (particle filter and smoother), this method provides a middle ground in terms of the computational resource requirements and accuracy. Unlike particle filter, this algorithm doesn't consider all the particles present in the Gaussian around the mean, yet it provides a faster run time with acceptable accuracy if both stock prices and option prices are used. The paper provides a practically efficient method to analyze the diffusion stochastic volatility models.