





HPC-QuantLib

FINITE DIFFERENCE METHODS / MONTE-CARLO / QUANTLIB

Probability Distribution of the Heston Model, Part I

 FEBRUARY 4, 2014  SEPTEMBER 27, 2014  HPCQUANTLIB  1 COMMENT

The Heston model is defined by the following stochastic differential equation of the log spot $x_t = \ln S_t$

$$\begin{aligned} dx_t &= \left(r_t - q_t - \frac{\nu_t}{2} \right) dt + \sqrt{\nu_t} dW_t^x \\ d\nu_t &= \kappa (\theta - \nu_t) dt + \sigma \sqrt{\nu_t} dW_t^\nu \\ \rho dt &= dW_t^x dW_t^\nu \end{aligned}$$

To a significant extent the popularity of the Heston model is based on the fact that semi-closed formulas for vanilla European options exist using the characteristic function of the model. The time evolution of the probability density function $p(x_t, \nu_t, t)$ is given by the corresponding Fokker-Planck equation [1]

$$\frac{\partial p}{\partial t} = \frac{1}{2} \frac{\partial^2}{\partial x^2} (\nu p) + \frac{\partial^2}{\partial x \partial \nu} (\rho \sigma \nu p) + \left(\frac{\nu}{2} - r_t + q_t \right) \frac{\partial}{\partial x} p + \frac{\sigma^2}{2} \frac{\partial^2}{\partial \nu^2} (\nu p) - \frac{\partial}{\partial \nu} (\kappa (\theta - \nu) p)$$

with the initial condition

$$p(x, \nu, t = 0) = \delta(x - x_0) \delta(\nu - \nu_0)$$

The reduced probability density function

$$p(x_t, t \mid x_0, \nu_0) = \int_0^\infty p(x_t, \nu_t, t) d\nu$$

for this initial value problem can be calculated using a semi-closed integral formula [2]

$$\begin{aligned} \Gamma &= \kappa + i \rho \sigma p_x \\ \Omega &= \sqrt{\Gamma^2 + \sigma^2 (p_x^2 - i p_x)} \\ p(x_t, t \mid \nu_0) &= \int_{-\infty}^{\infty} \frac{dp_x}{2\pi} \exp \left(i p_x (x_t - x_0 - (r - q)t) - \nu_0 \frac{p_x^2 - i p_x}{\Gamma + \Omega \coth(\Omega t/2)} \right) \\ &\quad \times \exp \left(-\frac{2\theta\kappa}{\sigma^2} \ln \left(\cosh \frac{\Omega t}{2} + \frac{\Gamma}{\Omega} \sinh \frac{\Omega t}{2} \right) + \frac{\kappa \Gamma \theta t}{\sigma^2} \right) \\ &= \int_{-\infty}^{\infty} \frac{dp_x}{2\pi} \tilde{p}(p_x, t \mid \nu_0) \end{aligned}$$



This gives the opportunity to write a pricing engine for arbitrary European payoffs. The value of an European option with payoff function $P(S_t) \in \mathbb{L}^2$ at maturity t is given by

$$\begin{aligned}\text{npv} &= \int_{-\infty}^{\infty} dx_t \int_0^{\infty} d\nu_t P(S_0 e^{x_t + (r_t - q_t)t}) e^{-r_t t} p(x_t, \nu_t, t) \\ &= e^{-r_t t} \int_{-\infty}^{\infty} dx_t P(S_0 e^{x_t + (r_t - q_t)t}) p(x_t, t \mid x_0, \nu_0)\end{aligned}$$

The calculation needs two nested integrations which can be carried out efficiently using e. g. the Gauss-Lobatto algorithm. The solution of the equation

$$|\tilde{p}(p_x, t \mid \nu_0)| = \text{QL_EPSILON}$$

determines the upper boundary for the integration over p_x . The boundaries $[-x_{min}, x_{max}]$ for the integration over x_t are chosen such that the interval covers ten times the expected variance

$$-x_{min} = x_{max} = 10 \sqrt{\int_0^t E[\nu_t] dt} = 10 \sqrt{\theta t + \frac{1}{\kappa} (\nu_0 - \theta) (1 - e^{-\kappa t})}$$

Obviously the nested integration makes this algorithm more tricky than the standard ways to price plain vanilla European options but it is not limited to vanilla payoffs. The implementation of this algorithm can be found [here](https://github.com/lballabio/quantlib/blob/master/QuantLib/ql/experimental/exoticoptions/analyticpdfhestonengine.cpp) (<https://github.com/lballabio/quantlib/blob/master/QuantLib/ql/experimental/exoticoptions/analyticpdfhestonengine.cpp>) within the [QuantLib trunk on Github](https://github.com/lballabio/quantlib). (<https://github.com/lballabio/quantlib>).

Broadie and Kaya [1] have outlined an algorithm to sample from the full probability density function $p(x_t, \nu_t, t \mid x_0, \nu_0)$ instead of the reduced density function $p(x_t, t \mid x_0, \nu_0)$. Starting point for this algorithm is the exact solution of the Heston stochastic differential equation

$$\begin{aligned}x_t &= x_0 + (r_t - q_t)t - \frac{1}{2} \int_0^t \nu_s ds + \rho \int_0^t \sqrt{\nu_s} dW_s^{(1)} + \sqrt{1 - \rho^2} \int_0^t \sqrt{\nu_s} dW_s^{(2)} \\ \nu_t &= \nu_0 + \kappa \theta t - \kappa \int_0^t \nu_s ds + \sigma \int_0^t \sqrt{\nu_s} dW_s^{(1)}\end{aligned}$$

The probability density function of the variance process ν_t is given by a noncentral chi-squared distribution

$$\nu_t = \frac{\sigma^2(1 - e^{-\kappa t})}{4\kappa} \chi_d'^2 \left(\frac{4\kappa e^{-\kappa t}}{\sigma^2(1 - e^{-\kappa t})} \nu_0 \right), d = \frac{4\theta\kappa}{\sigma^2}$$

The distribution $\Psi(0, t)$ of the integral $\int_0^t \nu_s ds$ conditional on ν_t and ν_0 can be calculated via the characteristic function

$$\text{Pr}(\Psi(t) \leq x) = \frac{2}{\pi} \int_0^{\infty} \frac{\sin ux}{u} \text{Re}(\Phi(u)) du$$

$$\begin{aligned}\Phi(a) &= \frac{\gamma(a) e^{-\frac{1}{2}(\gamma(a) - \kappa)t} (1 - e^{-\kappa t})}{\kappa (1 - e^{-\gamma(a)t})} \exp \left(\frac{\nu_t + \nu_0}{\sigma^2} \left[\frac{\kappa (1 + e^{-\kappa t})}{1 - e^{-\kappa t}} - \frac{\gamma(a) (1 + e^{-\gamma(a)t})}{1 - e^{-\gamma(a)t}} \right] \right) \\ &\times \frac{I_{0.5d-1} \left(\sqrt{\nu_0 \nu_t} \frac{4\gamma(a) e^{-0.5\gamma(a)t}}{\sigma^2 (1 - e^{-\gamma(a)t})} \right)}{I_{0.5d-1} \left(\sqrt{\nu_0 \nu_t} \frac{4\kappa e^{-0.5\kappa t}}{\sigma^2 (1 - e^{-\kappa t})} \right)} \\ &\times \frac{\exp \left((0.5d - 1) \left[-\frac{1}{2} \gamma(a)t + \ln \frac{\gamma(a)}{1 - e^{-\gamma(a)t}} \right] \right)}{\left(\frac{\gamma(a) e^{-0.5\gamma(a)t}}{1 - e^{-\gamma(a)t}} \right)^{0.5d-1}}\end{aligned}$$

$$\gamma(a) = \sqrt{\kappa^2 - 2i\sigma^2 a}$$

The modified Bessel function of first kind $I_{0.5d-1}(z)$ can be evaluated using series expansion for small and medium $|z|$ or asymptotic approximation for large $|z|$ [5]. Unfortunately Boost provides only real versions of the Bessel functions and the copyright status of older complex valued Fortran77 routine is vague. Therefore QuantLib comes with its own implementation (<https://github.com/lballabio/quantlib/blob/master/QuantLib/ql/math/modifiedbessel.cpp>).

Please notice that $\Phi(a)$ is already a continuous version of the characteristic function and therefore the integration does not need to track the branches of $\arg(z)$ when calculating the complex valued Bessel function [4].

The integration over the characteristic function is best done using either Gauss-Laguerre, Gauss-Lobatto or trapezoidal rule. The two later algorithms need to truncate the integration at some upper bound. First guess for a truncation limit can be taken from the Cornish-Fisher expansion (http://en.wikipedia.org/wiki/Cornish%E2%80%93Fisher_expansion) for some very small ϵ . The moment-generating function $\Phi(a)$ can be used to get the first, second and third moment of the distribution via finite difference quotient.

$$m_n = E(X^n) = \left. \frac{d^n}{dy^n} \Phi(x + iy) \right|_{x=y=0}$$

The next term is now fairly easy to calculate

$$\int_0^t \sqrt{\nu_s} dW_s^{(1)} = \frac{1}{\sigma} \left(\nu_t - \nu_0 - \kappa \theta t + \kappa \int_0^t \nu_s ds \right)$$

The log spot process can now be sampled using a standard normal random variable Z and

$$\begin{aligned} x_t &= x_0 + m(t) + \sigma(t)Z \\ \sigma^2(t) &= (1 - \rho^2) \int_0^t \nu_s ds \\ m(t) &= (r_t - q_t)t - \frac{1}{2} \int_0^t \nu_s ds + \rho \int_0^t \sqrt{\nu_s} dW_s^{(1)} \end{aligned}$$

This sampling algorithm is exact even for very large time steps and therefore gives some advantages for quasi random Monte-Carlo methods but the inversion of the integration of the characteristic function is also very slow. The algorithm is implemented within the HestonProcess (<https://github.com/lballabio/quantlib/blob/master/QuantLib/ql/processes/hestonprocess.cpp>) class.

[1] I. Clark, Foreign Exchange Option Pricing: A Practitioners Guide (http://books.google.de/books?id=7vua-0-2sgMC&pg=PT130&lpg=PT130&dq=heston+model+fokker+planck+equation&source=bl&ots=nHutgX1qSu&sig=2FFDK3WVot5LSxBldfE5VoYCKc&hl=de&sa=X&ei=R_3kUvDpBoWEtAb0xIHYAQ&ved=0CEwQ6AEwAg#v=onepage&q=heston%20model%20fokker%20planck%20equation&f=false), p. 113

[2] A. Dragulescu, V. Yakovenko, Probability distribution of returns in the Heston model with stochastic volatility (<http://arxiv.org/pdf/cond-mat/0203046.pdf>)

[3] M. Broadie, Ö. Kaya, Exact Simulation of Stochastic Volatility and other Affine Jump Diffusion Processes (<http://finmath.stanford.edu/seminars/documents/Broadie.pdf>)

[4] R. Lord, Efficient pricing algorithms for exotic derivatives (<http://www.google.de/url?sa=t&rct=j&q=&esrc=s&source=web&cd=3&ved=0CDwQFjAC&url=http%3A%2F%2Frepub.eur.nl%2Fpub%2F13917%2FLordR-Thesis.pdf&ei=49nwUsreLYjnsbwZ2IHIDQ&>)

[usg=AFQjCNFWjTcnOft7cgo_0q-VDDhUaS1YUA&bvm=bv.60444564,d.Yms](#)), p. 40

[5] J.R. Culham, Bessel Functions of the First and Second Kind (http://www.mhtlab.uwaterloo.ca/courses/me755/web_chap4.pdf).

One thought on “Probability Distribution of the Heston Model, Part I”

1. **Probability Distribution of the Heston Model, Part II | HPC-QuantLib** says:
FEBRUARY 22, 2014 AT 5:43 PM
[...] ← Probability Distribution of the Heston Model, Part I [...]

REPLY 

BLOG AT WORDPRESS.COM. DO NOT SELL OR SHARE MY PERSONAL INFORMATION