Stochastic Volatility and Time-Varying Risk-Free Rate Model for Pricing European Options

Jamal Timsah

April 4, 2009

Contents

Introduction

Motivation of the Model

Fisher Black and Myron Scholes developed a model in which the price of European-style options could be described by a partial differential equation, assuming that the price follows a geometric Brownian motion with constant drift and volatility parameters. With some manipulation the Black-Scholes PDE can be transformed into a diffusion equation, which can be solved with the Normal distribution as a component of the solution.

Some notable shortcomings to this model lie in the assumptions that were used to create it. Most importantly, it cannot be assumed that the the price of the underlying security in an option follows a geometric Brownian motion with both constant drift and volatility parameters. Instead, it would seem more reasonable that the volatility of a securitiy's price is a stochastic process following a Brownian motion. Additionally, the risk-free rate varies over time, and this would affect the value of options that have long contracts.

A major consequence of relaxing the conditions of the model is that the PDE of the options value will no longer be able to solved using the Normal distribution. The solution will also be very messy. However, the hope is that that Stochastic Volatility with a Time-Varying Risk-Free Rate PDE will better describe the value of options.

Assumptions of the Model

The following assumptions and abbreviations will hold throughout the paper:

- The price of the underlying security is a stochastic process which follows a Brownian motion.
  - The stochastic process is of the form: $dS_t = r_t\, S_t\, dt + \sqrt{v_t} S_t\, dW_t$
  - $r(t)$ is the drift parameter (and also the risk-free rate) and v_t is the stochastic variance
- The variance of the underlying security is a stochastic process which follows a Brownian motion.
  - The stochastic process is of the form: $dv_t = \alpha_{S,t}(v_t)\, dt + \beta_{S,t}(v_t)\, dB_t$
  - $\alpha_{S,t}(v_t) = \theta(\omega - v_t)$
    - ω is the long-term mean volatility.
    - θ is the rate at which v_t tends to ω.
  - $\beta_{S,t}(v_t) = \xi\sqrt{v_t}$
    - ξ is the volatility of the variance process.
  - This choice of α and β is called the Heston model.
- The Brownian motions dW_t and dB_t have a correlation of ρ.
- The risk-free rate is a continuous function of time.
  - r_t = the risk-free rate at time t (may also be written as r(t))
- There are no arbitrage opportunities in the market.
- There are no transaction costs.

- The underlying securities do not yield dividends.
- The underlying securities are completely divisible.
- Short-selling of the underlying securities are permitted.
- Trading on the market occurs continuously.
- The following variables will be in common use:
  - C = the price of a European call option
  - κ = the strike price of a European call option
  - T = the length of the option contract in years
  - S_t = the price of the security at time t (may also be written in lower case)

In reality, many of the above assumptions are violated; however, they greatly simplify derivation of this model. Assuming that securities are divisible and that trading occurs continuously allow the mathematics of the model to be continuous, but the accuracy is sacrificed. The market is not open 24 hours a day, 7 days a week. In fact, the market is open less than one-third of the time that the model assumes. On the other hand, it is reasonable to own a fraction of a security as this occurs in the market during splits. It is also assumed that short-selling is allowed, which is not an unreasonable assumption. Also, it is somewhat reasonable to neglect dividends if the dividend is a small fraction of the price of the stock and they are paid out only a few times a year.

All of these things said, a model can be based entirely on wrong assumptions, but if it works consistently, people will use it. In the next sections, the partial differential equation that describes the price of the security will be derived, and it will be solved numerically in different ways. Then, the model's validity will be tested against historical market data and its performance will be compared to the Black-Scholes model.

Derviation of the Partial Differential Equation

Create a portfolio of securites

Construct a simple portfolio consisting of the following:

- long one option (where f is the value of one option)
- short Δ shares of the underlying security (where s is the price per share)
- short δ shares of a security whose price is volatility-dependent (where h is the price per share)

Let Π denote the value of the portfolio.
The value of the portfolio is then: $\boldsymbol{\Pi} = \mathbf{f} - \mathbf{s} \cdot \boldsymbol{\Delta} - \mathbf{h} \cdot \delta$.

You may be wondering why it is even necessary to contruct a portfolio. After all, we are trying to find the value of the option. This portfolio was selected in such a way that is an immediate extension to the process used in the Black-Scholes derivation. We include an option in the portfolio as that is what we are trying to model. In the Black-Scholes, shares of the underlying security are included because the stock price is a form of randomness that must be taken into account. We include the additional volatility-dependent security because this model has the extra assumption that volatility is a stochastic process. In short, all assumptions of the model must be included in the portfolio as we will eliminate the risk of this portfolio in such a way that exposes the value of the option.

No arbitrage condition

A no arbitrage argument will show that $\mathbf{d\Pi} = \boldsymbol{\Pi}\, \mathbf{r}\, \mathbf{dt}$.

Let us assume that $d\Pi > \Pi\, r(t)\, dt$.
At time t, we can borrow Π at the risk-free rate $r(t)$ from the bank and then invest the Π borrowed into the portfolio. Then at time $t + dt$, the portfolio is now worth $\Pi + d\Pi$ in value and the amount we owe to the bank is $\Pi + \Pi\, r(t)\, dt$. We then make a profit of $d\Pi - \Pi\, r(t)\, dt$ because of our assumption, which is not allowed as this model does not allow arbitrage oppurtunities to exist in the market.

Let us assume that $d\Pi < \Pi\, r(t)\, dt$.
At time t, we short the entire portfolio worth Π and invest Π in the bank at rate $r(t)$. At time $t + dt$, we get $\Pi(1 + r\, dt)$ out of the bank because of the interest accrued over $[t, t + dt]$ and repay $\Pi + d\Pi$ in order to cover what was shorted. We make a profit of $\Pi\, r(t)\, dt - d\Pi$ because of our assumption, which is not allowed as this model does not allow arbitrage oppurtunities to exist in the market.

Thus, $d\Pi = \Pi\, r(t)\, dt$ must hold in order to not violate the no arbitrage condition.

It is important to mention why we only consider $r(t)$ in the above argument when the risk-free rate after dt elapses is $r(t + dt)$. We do not consider the latter rate because of our assumptions. Since $r(t)$ was assumed to be a continuous function, it follows that $r(t) \approx r(t + dt)$.

Find dΠ from differentiation

The value of the portfolio is:

$\Pi = f - s \cdot \Delta - h \cdot \delta$

Differentiating we get the following:

$d\Pi = df - ds \cdot \Delta - dh \cdot \delta$

We will use the Taylor expansion about the differentials dt, ds, and dv. Assume that everything takes place at time t.

$$\begin{aligned} f(t+dt, s+ds, v+dv) \approx f(t,s,v) &+ \frac{\partial f}{\partial t}(dt) + \frac{\partial f}{\partial s}(ds) + \frac{\partial f}{\partial v}(dv) \\ &+ \frac{1}{2}\left[\frac{\partial^2 f}{\partial t^2}(dt)^2 + \frac{\partial^2 f}{\partial s^2}(ds)^2 + \frac{\partial^2 f}{\partial v^2}(dv)^2 + 2\frac{\partial^2 f}{\partial t \partial s}(dt)(ds) + 2\frac{\partial^2 f}{\partial t \partial v}(dt)(dv) + 2\frac{\partial^2 f}{\partial s \partial v}(ds)(dv)\right] \end{aligned}$$

$$\begin{aligned} h(t+dt, s+ds, v+dv) \approx h(t,s,v) &+ \frac{\partial h}{\partial t}(dt) + \frac{\partial h}{\partial s}(ds) + \frac{\partial h}{\partial v}(dv) \\ &+ \frac{1}{2}\left[\frac{\partial^2 h}{\partial t^2}(dt)^2 + \frac{\partial^2 h}{\partial s^2}(ds)^2 + \frac{\partial^2 h}{\partial v^2}(dv)^2 + 2\frac{\partial^2 h}{\partial t \partial s}(dt)(ds) + 2\frac{\partial^2 h}{\partial t \partial v}(dt)(dv) + 2\frac{\partial^2 h}{\partial s \partial v}(ds)(dv)\right] \end{aligned}$$

The derivatives are approximately the following:

$$\begin{aligned} df \approx\ & \frac{\partial f}{\partial t}(dt) + \frac{\partial f}{\partial s}(ds) + \frac{\partial f}{\partial v}(dv) \\ &+ \frac{1}{2}\left[\frac{\partial^2 f}{\partial t^2}(dt)^2 + \frac{\partial^2 f}{\partial s^2}(ds)^2 + \frac{\partial^2 f}{\partial v^2}(dv)^2 + 2\frac{\partial^2 f}{\partial t \partial s}(dt)(ds) + 2\frac{\partial^2 f}{\partial t \partial v}(dt)(dv) + 2\frac{\partial^2 f}{\partial s \partial v}(ds)(dv)\right] \end{aligned}$$

$$\begin{aligned} dh \approx\ & \frac{\partial h}{\partial t}(dt) + \frac{\partial h}{\partial s}(ds) + \frac{\partial h}{\partial v}(dv) \\ &+ \frac{1}{2}\left[\frac{\partial^2 h}{\partial t^2}(dt)^2 + \frac{\partial^2 h}{\partial s^2}(ds)^2 + \frac{\partial^2 h}{\partial v^2}(dv)^2 + 2\frac{\partial^2 h}{\partial t \partial s}(dt)(ds) + 2\frac{\partial^2 h}{\partial t \partial v}(dt)(dv) + 2\frac{\partial^2 h}{\partial s \partial v}(ds)(dv)\right] \end{aligned}$$

The products of the differentials dt, ds, and dv will be evaluated.

$$\begin{aligned} (dt)^2 &= 0 \\ (ds)^2 &= (r\,s\,dt + \sqrt{v}s\,dW)(r\,s\,dt + \sqrt{v}S\,dW) = r^2\,s^2\,dt^2 + 2\,r\,s^2\,\sqrt{v}\,dW\,dt + v\,s^2\,dW^2 \\ &= v\,s^2\,dt \\ (dv)^2 &= (\alpha_{S,t}(v)\,dt + \beta_{S,t}(v)\,dB)(\alpha_{S,t}(v)\,dt + \beta_{S,t}(v)\,dB) = \alpha_{S,t}^2(v)\,dt^2 + 2\,\alpha_{S,t}(v)\,\beta_{S,t}(v)\,dt\,dB + \beta_{S,t}^2(v)\,dB^2 \\ &= \beta_{S,t}^2(v)\,dt \end{aligned}$$

$$
\begin{aligned}
(dt)(ds) &= 0 \\
(dt)(dv) &= 0 \\
(ds)(dv) &= (r\,s\,dt + \sqrt{v}S\,dW)(\alpha_{S,t}(v)\,dt + \beta_{S,t}(v)\,dB) \\
&= r\,s\,\alpha_{S,t}(v)\,dt^2 + r\,s\,\beta_{S,t}(v)\,dB\,dt + \sqrt{v}S\,\alpha_{S,t}(v)\,dW\,dt + \beta_{S,t}(v)\,\sqrt{v}S\,dW\,dB \\
&= \beta_{S,t}(v)\,\sqrt{v}S\,\rho\,dt
\end{aligned}
$$

Plugging these results into the derivatives gives us the following:

$$
df = \frac{\partial f}{\partial t}(dt) + \frac{\partial f}{\partial s}(ds) + \frac{\partial f}{\partial v}(dv) + \frac{1}{2}\left[\frac{\partial^2 f}{\partial s^2} v\,s^2\,dt + \frac{\partial^2 f}{\partial v^2}\beta_{S,t}^2(v)\,dt + 2\frac{\partial^2 f}{\partial s \partial v}\beta_{S,t}(v)\,\sqrt{v}S\,\rho\,dt\right]
$$

$$
dh = \frac{\partial h}{\partial t}(dt) + \frac{\partial h}{\partial s}(ds) + \frac{\partial h}{\partial v}(dv) + \frac{1}{2}\left[\frac{\partial^2 h}{\partial s^2} v\,s^2\,dt + \frac{\partial^2 h}{\partial v^2}\beta_{S,t}^2(v)\,dt + 2\frac{\partial^2 h}{\partial s \partial v}\beta_{S,t}(v)\,\sqrt{v}S\,\rho\,dt\right]
$$

We can now find $d\Pi$:

$$
\begin{aligned}
d\Pi &= df - ds \cdot \Delta - dh \cdot \delta \\
&= \frac{\partial f}{\partial t}(dt) + \frac{\partial f}{\partial s}(ds) + \frac{\partial f}{\partial v}(dv) + \frac{1}{2}\left[\frac{\partial^2 f}{\partial s^2} v\,s^2\,dt + \frac{\partial^2 f}{\partial v^2}\beta_{S,t}^2(v)\,dt + 2\frac{\partial^2 f}{\partial s \partial v}\beta_{S,t}(v)\,\sqrt{v}S\,\rho\,dt\right] - \Delta\,ds \\
&- \delta \cdot \left(\frac{\partial h}{\partial t}(dt) + \frac{\partial h}{\partial s}(ds) + \frac{\partial h}{\partial v}(dv) + \frac{1}{2}\left[\frac{\partial^2 h}{\partial s^2} v\,s^2\,dt + \frac{\partial^2 h}{\partial v^2}\beta_{S,t}^2(v)\,dt + 2\frac{\partial^2 h}{\partial s \partial v}\beta_{S,t}(v)\,\sqrt{v}S\,\rho\,dt\right]\right)
\end{aligned}
$$

Collecting like terms:

$$
\begin{aligned}
d\Pi = &\left(\frac{\partial f}{\partial t} + \frac{1}{2}\left[\frac{\partial^2 f}{\partial s^2} v\,s^2 + \frac{\partial^2 f}{\partial v^2}\beta_{S,t}^2(v) + 2\frac{\partial^2 f}{\partial s \partial v}\beta_{S,t}(v)\,\sqrt{v}S\,\rho\right]\right) dt \\
&- \delta \cdot \left(\frac{\partial h}{\partial t} + \frac{1}{2}\left[\frac{\partial^2 h}{\partial s^2} v\,s^2 + \frac{\partial^2 h}{\partial v^2}\beta_{S,t}^2(v) + 2\frac{\partial^2 h}{\partial s \partial v}\beta_{S,t}(v)\,\sqrt{v}S\,\rho\right]\right) dt \\
&+ \left(\frac{\partial f}{\partial s} - \delta\frac{\partial h}{\partial s} - \Delta\right) ds + \left(\frac{\partial f}{\partial v} - \delta\frac{\partial h}{\partial v}\right) dv
\end{aligned}
$$

Hedge the Portfolio

Now we will hedge the risk by setting the ds and dv coefficients to zero. We will find how many shares of the stock and the other security are needed for the portfolio to be risk-free.

$$
\begin{aligned}
0 &= \frac{\partial f}{\partial v} - \delta\frac{\partial h}{\partial v} \\
\delta\frac{\partial h}{\partial v} &= \frac{\partial f}{\partial v} \\
\delta &= \frac{\left(\frac{\partial f}{\partial v}\right)}{\left(\frac{\partial h}{\partial v}\right)} = \frac{\partial f}{\partial h}
\end{aligned}
$$

$$
\begin{aligned}
0 &= \frac{\partial f}{\partial s} - \delta \frac{\partial h}{\partial s} - \Delta \\
0 &= \frac{\partial f}{\partial s} - \frac{\partial f}{\partial h}\frac{\partial h}{\partial s} - \Delta \\
\Delta &= 0
\end{aligned}
$$

At this point we have a portfolio that acts as a bond. By combining the results from the no-arbitrage argument and the $d\Pi$ found by differentiation we get the following relationship:

$$
\begin{aligned}
\Pi\, r\, dt = d\Pi = &\left(\frac{\partial f}{\partial t} + \frac{1}{2}\left[\frac{\partial^2 f}{\partial s^2} v\, s^2 + \frac{\partial^2 f}{\partial v^2}\beta_{S,t}^2(v) + 2\frac{\partial^2 f}{\partial s \partial v}\beta_{S,t}(v)\sqrt{v} S\, \rho \right]\right) dt \\
&- \delta \cdot \left(\frac{\partial h}{\partial t} + \frac{1}{2}\left[\frac{\partial^2 h}{\partial s^2} v\, s^2 + \frac{\partial^2 h}{\partial v^2}\beta_{S,t}^2(v) + 2\frac{\partial^2 h}{\partial s \partial v}\beta_{S,t}(v)\sqrt{v} S\, \rho \right]\right) dt
\end{aligned}
$$

Dividing everything by dt and substituting in the value of the portfolio yields:

$$
\begin{aligned}
(f - s\cdot\Delta - h\cdot\delta)\, r = &\left(\frac{\partial f}{\partial t} + \frac{1}{2}\left[\frac{\partial^2 f}{\partial s^2} v\, s^2 + \frac{\partial^2 f}{\partial v^2}\beta_{S,t}^2(v) + 2\frac{\partial^2 f}{\partial s \partial v}\beta_{S,t}(v)\sqrt{v} S\, \rho \right]\right) \\
&- \delta \cdot \left(\frac{\partial h}{\partial t} + \frac{1}{2}\left[\frac{\partial^2 h}{\partial s^2} v\, s^2 + \frac{\partial^2 h}{\partial v^2}\beta_{S,t}^2(v) + 2\frac{\partial^2 h}{\partial s \partial v}\beta_{S,t}(v)\sqrt{v} S\, \rho \right]\right)
\end{aligned}
$$

Substituting in the values of Δ and δ yields:

$$
\begin{aligned}
\left(f - h \cdot \frac{\left(\frac{\partial f}{\partial v}\right)}{\left(\frac{\partial h}{\partial v}\right)} \right) r = &\left(\frac{\partial f}{\partial t} + \frac{1}{2}\left[\frac{\partial^2 f}{\partial s^2} v\, s^2 + \frac{\partial^2 f}{\partial v^2}\beta_{S,t}^2(v) + 2\frac{\partial^2 f}{\partial s \partial v}\beta_{S,t}(v)\sqrt{v} S\, \rho \right]\right) \\
&- \frac{\left(\frac{\partial f}{\partial v}\right)}{\left(\frac{\partial h}{\partial v}\right)} \cdot \left(\frac{\partial h}{\partial t} + \frac{1}{2}\left[\frac{\partial^2 h}{\partial s^2} v\, s^2 + \frac{\partial^2 h}{\partial v^2}\beta_{S,t}^2(v) + 2\frac{\partial^2 h}{\partial s \partial v}\beta_{S,t}(v)\sqrt{v} S\, \rho \right]\right)
\end{aligned}
$$

Rearranging we get:

$$
\begin{aligned}
&\frac{\partial f}{\partial t} + \frac{1}{2}\left[\frac{\partial^2 f}{\partial s^2} v\, s^2 + \frac{\partial^2 f}{\partial v^2}\beta_{S,t}^2(v) + 2\frac{\partial^2 f}{\partial s \partial v}\beta_{S,t}(v)\sqrt{v} S\, \rho \right] - r\, f \\
&\quad = \frac{\left(\frac{\partial f}{\partial v}\right)}{\left(\frac{\partial h}{\partial v}\right)} \cdot \left(\frac{\partial h}{\partial t} + \frac{1}{2}\left[\frac{\partial^2 h}{\partial s^2} v\, s^2 + \frac{\partial^2 h}{\partial v^2}\beta_{S,t}^2(v) + 2\frac{\partial^2 h}{\partial s \partial v}\beta_{S,t}(v)\sqrt{v} S\, \rho \right] - r\, h \right)
\end{aligned}
$$

$$
\begin{aligned}
&\frac{1}{\left(\frac{\partial f}{\partial v}\right)} \cdot \left(\frac{\partial f}{\partial t} + \frac{1}{2}\left[\frac{\partial^2 f}{\partial s^2} v\, s^2 + \frac{\partial^2 f}{\partial v^2}\beta_{S,t}^2(v) + 2\frac{\partial^2 f}{\partial s \partial v}\beta_{S,t}(v)\sqrt{v} S\, \rho \right] - r\, f \right) \\
&\quad = \frac{1}{\left(\frac{\partial h}{\partial v}\right)} \cdot \left(\frac{\partial h}{\partial t} + \frac{1}{2}\left[\frac{\partial^2 h}{\partial s^2} v\, s^2 + \frac{\partial^2 h}{\partial v^2}\beta_{S,t}^2(v) + 2\frac{\partial^2 h}{\partial s \partial v}\beta_{S,t}(v)\sqrt{v} S\, \rho \right] - r\, h \right)
\end{aligned}
$$

Gatheral reasons that since both sides of the equation are functions of f and h respectively, then they must both be equal to some function ϕ of the independent variables t, s, and v. In the following equation, the right side denotes the market price of volatility risk, which is the ϕ (Gatheral 6).

$$\frac{\partial f}{\partial t} + \frac{1}{2}\left[\frac{\partial^2 f}{\partial s^2} v\, s^2 + \frac{\partial^2 f}{\partial v^2}\beta^2_{S,t}(v) + 2\frac{\partial^2 f}{\partial s \partial v}\beta_{S,t}(v)\sqrt{v} S\,\rho\right] - r\,f = -(\alpha_{S,t}(v) - \phi\,\beta_{S,t}(v)\sqrt{v})\left(\frac{\partial f}{\partial v}\right)$$

Since we are using the Heston model, he describes market risk in such a way that yields the following:

The Stochastic Volatility PDE

$$\frac{\partial f}{\partial t} + \frac{1}{2} v\, s^2 \frac{\partial^2 f}{\partial s^2} + \rho\,\xi\, v\, s\,\frac{\partial^2 f}{\partial v \partial s} + \frac{1}{2}\xi^2 v \frac{\partial f^2}{\partial v^2} + r\,s\,\frac{\partial f}{\partial s} - \theta(v - \omega)\frac{\partial f}{\partial v} - r\,f = 0$$

Numerical Solution of the PDE

Now that we have a partial differential equation that models the value of an option, we must now find a way to solve for the value of the option. Since the Heston model of volatility was selected, there exists a near-analytical solution in that the value equals the integral of a function which has no antiderivative. Even though this solution exists, we will not even consider it because it involves integration of the real part of a complex function, and the derivation is beyond the scope of this paper. Instead, we will solve the PDE numerically while taking into account the stability of the method that we choose.

Boundary Conditions of the Model

We will solve the PDE:

$$\frac{\partial f}{\partial t} + \frac{1}{2} v\, s^2 \frac{\partial^2 f}{\partial s^2} + \rho\, \xi\, v\, s\, \frac{\partial^2 f}{\partial v \partial s} + \frac{1}{2}\xi^2 v \frac{\partial f^2}{\partial v^2} + r\, s\, \frac{\partial f}{\partial s} - \theta(v - \omega)\frac{\partial f}{\partial v} - r_t\, f = 0$$

With the following boundary conditions:

$$\begin{aligned} C(0,t) &= 0 \\ C(S_T, T) &= \max(S_T - K, 0) \\ \lim_{t\to\infty} C(S_t, t) &= S \end{aligned}$$

From now on, f denotes the value of a European call option.

Finite Difference Approach - Explicit Method

Discretization of the Partial Derivatives

We will use a backward difference formula for the time partial (the reason why we do this is at the end of the section), and the centered difference formulas for the other partials:

$$\begin{aligned}
\frac{\partial f}{\partial t} &\approx \frac{f(t,s,v) - f(t-\Delta t, s, v)}{\Delta t} \\
\frac{\partial f}{\partial s} &\approx \frac{f(t,s+\Delta s,v) - f(t, s-\Delta s, v)}{2\Delta s} \\
\frac{\partial f}{\partial v} &\approx \frac{f(t,s,v+\Delta v) - f(t, s, v-\Delta v)}{2\Delta v} \\
\frac{\partial^2 f}{\partial s^2} &= \frac{\partial}{\partial s}\left[\frac{\partial f}{\partial s}\right] \approx \frac{\partial}{\partial s}\left[\frac{f(t,s,v) - f(t,s-\Delta s,v)}{\Delta s}\right] \approx \frac{\frac{f(t,s+\Delta s,v)-f(t,s,v)}{\Delta s} - \frac{f(t,s,v)-f(t,s-\Delta s,v)}{\Delta s}}{\Delta s} \\
&= \frac{f(t,s+\Delta s,v) - f(t,s,v)}{\Delta s^2} - \frac{f(t,s,v) - f(t,s-\Delta s,v)}{\Delta s^2} \\
&= \frac{f(t,s+\Delta s,v) - 2f(t,s,v) + f(t,s-\Delta s,v)}{\Delta s^2}
\end{aligned}$$

$$
\begin{aligned}
\frac{\partial^2 f}{\partial v^2} &= \frac{\partial}{\partial v}\left[\frac{\partial f}{\partial v}\right] \approx \frac{\partial}{\partial v}\left[\frac{f(t,s,v)-f(t,s,v-\Delta v)}{\Delta v}\right] \approx \frac{\frac{f(t,s,v+\Delta v)-f(t,s,v)}{\Delta v} - \frac{f(t,s,v)-f(t,s,v-\Delta v)}{\Delta v}}{\Delta v} \\
&= \frac{f(t,s,v+\Delta v)-f(t,s,v)}{\Delta v^2} - \frac{f(t,s,v)-f(t,s,v-\Delta v)}{\Delta v^2} \\
&= \frac{f(t,s,v+\Delta v)-2f(t,s,v)+f(t,s,v-\Delta v)}{\Delta v^2} \\
\frac{\partial^2 f}{\partial v \partial s} &= \frac{\partial}{\partial v}\left[\frac{\partial f}{\partial s}\right] \approx \frac{\partial}{\partial v}\left[\frac{f(t,s+\Delta s,v)-f(t,s-\Delta s,v)}{2\Delta s}\right] \\
&\approx \frac{\frac{f(t,s+\Delta s,v+\Delta v)-f(t,s-\Delta s,v+\Delta v)}{2\Delta s} - \frac{f(t,s+\Delta s,v-\Delta v)-f(t,s-\Delta s,v-\Delta v)}{2\Delta s}}{2\Delta v} \\
&= \frac{f(t,s+\Delta s,v+\Delta v)-f(t,s-\Delta s,v+\Delta v)}{4\Delta s\Delta v} - \frac{f(t,s+\Delta s,v-\Delta v)-f(t,s-\Delta s,v-\Delta v)}{4\Delta s\Delta v} \\
&= \frac{f(t,s+\Delta s,v+\Delta v)-f(t,s-\Delta s,v+\Delta v)-f(t,s+\Delta s,v-\Delta v)+f(t,s-\Delta s,v-\Delta v)}{4\Delta s\Delta v}
\end{aligned}
$$

Substitution and Simplification

Let the following intervals be defined to form the mesh that will be used:

$$
\begin{aligned}
&t \in [0,T] \quad \text{where the step size is } \Delta t \quad (L \text{ elements in the interval}) \\
&s \in [0,S_{max}] \quad \text{where the step size is } \Delta s \quad (M \text{ elements in the interval}) \\
&v \in [0,V_{max}] \quad \text{where the step size is } \Delta v \quad (N \text{ elements in the interval})
\end{aligned}
$$

We will then define indices of the intervals in the following way: Let the following intervals be defined:

$$
\begin{aligned}
i &\in [1,2,\cdots,L-1,L] \\
j &\in [1,2,\cdots,M-1,M] \\
k &\in [1,2,\cdots,N-1,N]
\end{aligned}
$$

The approximations to the partial derivatives will be plugged into the discretized PDE and we will try to get something useful out of it.

$$
\frac{\partial f}{\partial t} + \frac{1}{2}(k\Delta v)\,(j^2\Delta s^2)\frac{\partial^2 f}{\partial s^2} + \rho\,\xi\,(k\Delta v)\,(j\Delta s)\,\frac{\partial^2 f}{\partial v\partial s} + \frac{1}{2}\xi^2(k\Delta v)\frac{\partial f^2}{\partial v^2} + r\,(j\Delta s)\,\frac{\partial f}{\partial s} - \theta((k\Delta v)-\omega)\frac{\partial f}{\partial v} - r_t\,f = 0
$$

We get a mess:

$$
\begin{aligned}
&\frac{f(t,s,v)-f(t-\Delta t,s,v)}{\Delta t} + \frac{1}{2}(k\Delta v)\,(j^2\Delta s^2)\frac{f(t,s+\Delta s,v)-2f(t,s,v)+f(t,s-\Delta s,v)}{\Delta s^2} \\
&+ \rho\,\xi\,(k\Delta v)\,(j\Delta s)\,\frac{f(t,s+\Delta s,v+\Delta v)-f(t,s-\Delta s,v+\Delta v)-f(t,s+\Delta s,v-\Delta v)+f(t,s-\Delta s,v-\Delta v)}{4\Delta s\Delta v} \\
&+ \frac{1}{2}\xi^2(k\Delta v)\frac{f(t,s,v+\Delta v)-2f(t,s,v)+f(t,s,v-\Delta v)}{\Delta v^2} + r_t\,(j\Delta s)\,\frac{f(t,s+\Delta s,v)-f(t,s-\Delta s,v)}{2\Delta s} \\
&- \theta((k\Delta v)-\omega)\frac{f(t,s,v+\Delta v)-f(t,s,v-\Delta v)}{2\Delta v} - r_t\,f(t,s,v) = 0
\end{aligned}
$$

At this point, we will introduce new notation so that we can make this equation more readable. The following substitution will be made:

$$f_{t+1}^{(s+1,v+1)} = f(t+\Delta t, s+\Delta s, v+\Delta v)$$

$$\begin{aligned}
\frac{f_t^{(s,v)} - f_{t-1}^{(s,v)}}{\Delta t} &+ \frac{1}{2}(k\Delta v)(j^2\Delta s^2)\frac{f_t^{(s+1,v)} - 2f_t^{(s,v)} + f_t^{(s-1,v)}}{\Delta s^2} \\
&+ \rho\xi(j\Delta s)(k\Delta v)\frac{f_t^{(s+1,v+1)} - f_t^{(s-1,v+1)} - f_t^{(s+1,v-1)} + f_t^{(s-1,v-1)}}{4\Delta s\Delta v} \\
&+ \frac{1}{2}\xi^2(k\Delta v)\frac{f_t^{(s,v+1)} - 2f_t^{(s,v)} + f_t^{(s,v-1)}}{\Delta v^2} + r_t(j\Delta s)\frac{f_t^{(s+1,v)} - f_t^{(s-1,v)}}{2\Delta s} \\
&- \theta((k\Delta v) - \omega)\frac{f_t^{(s,v+1)} - f_t^{(s,v-1)}}{2\Delta v} - r_t f_t^{(s,v)} = 0
\end{aligned}$$

Eliminating in the numerator and denominator, we get:

$$\begin{aligned}
\frac{f_t^{(s,v)} - f_{t-1}^{(s,v)}}{\Delta t} &+ \frac{1}{2}(k\Delta v)j^2\left(f_t^{(s+1,v)} - 2f_t^{(s,v)} + f_t^{(s-1,v)}\right) \\
&+ \rho\xi jk\frac{f_t^{(s+1,v+1)} - f_t^{(s-1,v+1)} - f_t^{(s+1,v-1)} + f_t^{(s-1,v-1)}}{4} \\
&+ \frac{1}{2}\xi^2 k\frac{f_t^{(s,v+1)} - 2f_t^{(s,v)} + f_t^{(s,v-1)}}{\Delta v} + r_t j\frac{f_t^{(s+1,v)} - f_t^{(s-1,v)}}{2} \\
&- \theta((k\Delta v) - \omega)\frac{f_t^{(s,v+1)} - f_t^{(s,v-1)}}{2\Delta v} - r_t f_t^{(s,v)} = 0
\end{aligned}$$

Grouping the terms by time, we get:

$$\begin{aligned}
f_{t-1}^{(s,v)} = f_t^{(s,v)} + \Delta t\Big(&\frac{1}{2}(k\Delta v)j^2\left(f_t^{(s+1,v)} - 2f_t^{(s,v)} + f_t^{(s-1,v)}\right) \\
&+ \rho\xi jk\frac{f_t^{(s+1,v+1)} - f_t^{(s-1,v+1)} - f_t^{(s+1,v-1)} + f_t^{(s-1,v-1)}}{4} \\
&+ \frac{1}{2}\xi^2 k\frac{f_t^{(s,v+1)} - 2f_t^{(s,v)} + f_t^{(s,v-1)}}{\Delta v} + r_t j\frac{f_t^{(s+1,v)} - f_t^{(s-1,v)}}{2} \\
&- \theta((k\Delta v) - \omega)\frac{f_t^{(s,v+1)} - f_t^{(s,v-1)}}{2\Delta v} - r_t f_t^{(s,v)}\Big)
\end{aligned}$$

The following explicit equation is in a form that is suitable to be used in MATLAB.

$$f_{t-1}^{(s,v)} = \hat{a}f_t^{(s-1,v-1)} + \hat{b}f_t^{(s,v-1)} + \hat{c}f_t^{(s+1,v-1)} + \hat{d}f_t^{(s-1,v)} + \hat{e}f_t^{(s,v)} + \hat{f}f_t^{(s+1,v)} + \hat{g}f_t^{(s-1,v+1)} + \hat{h}f_t^{(s,v+1)} + \hat{i}f_t^{(s+1,v+1)}$$

Where:

$$\hat{a} = \tfrac{1}{4}\rho\xi jk\Delta t \qquad \hat{b} = \left(k\tfrac{\xi^2}{2\Delta v} + \tfrac{\theta(k\Delta v - \omega)}{2\Delta v}\right)\Delta t \qquad \hat{c} = -\tfrac{1}{4}\rho\xi jk\Delta t$$

$$\hat{d} = \left(kj^2\tfrac{\Delta v}{2} - j\tfrac{r_t}{2}\right)\Delta t \qquad \hat{e} = 1 - \left(kj^2\Delta v + k\tfrac{\xi^2}{\Delta v} + r_t\right)\Delta t \qquad \hat{f} = \left(kj^2\tfrac{\Delta v}{2} + j\tfrac{r_t}{2}\right)\Delta t$$

$$\hat{g} = -\tfrac{1}{4}\rho\xi jk\Delta t \qquad \hat{h} = \left(k\tfrac{\xi^2}{2\Delta v} - \tfrac{\theta(k\Delta v - \omega)}{2\Delta v}\right)\Delta t \qquad \hat{i} = \tfrac{1}{4}\rho\xi jk\Delta t$$

Finite Difference Approach - Implicit Method

Discretization of the Partial Derivatives

All of the above difference equations will hold except for the approximation of $\frac{\partial f}{\partial t}$. Instead, we will use a forward difference approximation.

$$\frac{\partial f}{\partial t} \approx \frac{f(t+\Delta t, s, v) - f(t, s, v)}{\Delta t}$$

Substitution and Simplification

We get the following discretized PDE:

$$\begin{aligned}
\frac{f_{t+1}^{(s,v)} - f_t^{(s,v)}}{\Delta t} &+ \frac{1}{2}(k\Delta v)j^2 \left(f_t^{(s+1,v)} - 2f_t^{(s,v)} + f_t^{(s-1,v)}\right) \\
&+ \rho\xi jk \frac{f_t^{(s+1,v+1)} - f_t^{(s-1,v+1)} - f_t^{(s+1,v-1)} + f_t^{(s-1,v-1)}}{4} \\
&+ \frac{1}{2}\xi^2 k \frac{f_t^{(s,v+1)} - 2f_t^{(s,v)} + f_t^{(s,v-1)}}{\Delta v} + r_t j \frac{f_t^{(s+1,v)} - f_t^{(s-1,v)}}{2} \\
&- \theta((k\Delta v) - \omega)\frac{f_t^{(s,v+1)} - f_t^{(s,v-1)}}{2\Delta v} - r_t f_t^{(s,v)} = 0
\end{aligned}$$

Separating the terms by time, we get:

$$\begin{aligned}
f_{t+1}^{(s,v)} = f_t^{(s,v)} - \Delta t\Big(&\frac{1}{2}(k\Delta v)j^2 \left(f_t^{(s+1,v)} - 2f_t^{(s,v)} + f_t^{(s-1,v)}\right) \\
&+ \rho\xi jk \frac{f_t^{(s+1,v+1)} - f_t^{(s-1,v+1)} - f_t^{(s+1,v-1)} + f_t^{(s-1,v-1)}}{4} \\
&+ \frac{1}{2}\xi^2 k \frac{f_t^{(s,v+1)} - 2f_t^{(s,v)} + f_t^{(s,v-1)}}{\Delta v} + r_t j \frac{f_t^{(s+1,v)} - f_t^{(s-1,v)}}{2} \\
&- \theta((k\Delta v) - \omega)\frac{f_t^{(s,v+1)} - f_t^{(s,v-1)}}{2\Delta v} - r_t f_t^{(s,v)}\Big)
\end{aligned}$$

Representing the above in a simpler form yields the following implicit equation.

$$f_{t-1}^{(s,v)} = \hat{a} f_t^{(s-1,v-1)} + \hat{b} f_t^{(s,v-1)} + \hat{c} f_t^{(s+1,v-1)} + \hat{d} f_t^{(s-1,v)} + \hat{e} f_t^{(s,v)} + \hat{f} f_t^{(s+1,v)} + \hat{g} f_t^{(s-1,v+1)} + \hat{h} f_t^{(s,v+1)} + \hat{i} f_t^{(s+1,v+1)}$$

Where:

$$\begin{aligned}
&\hat{a} = -\tfrac{1}{4}\rho\xi jk\Delta t & &\hat{b} = -\left(k\tfrac{\xi^2}{2\Delta v} + \tfrac{\theta(k\Delta v - \omega)}{2\Delta v}\right)\Delta t & &\hat{c} = \tfrac{1}{4}\rho\xi jk\Delta t \\
&\hat{d} = -\left(kj^2\tfrac{\Delta v}{2} - j\tfrac{r_t}{2}\right)\Delta t & &\hat{e} = 1 + \left(kj^2\Delta v + k\tfrac{\xi^2}{\Delta v} + r_t\right)\Delta t & &\hat{f} = -\left(kj^2\tfrac{\Delta v}{2} + j\tfrac{r_t}{2}\right)\Delta t \\
&\hat{g} = \tfrac{1}{4}\rho\xi jk\Delta t & &\hat{h} = -\left(k\tfrac{\xi^2}{2\Delta v} - \tfrac{\theta(k\Delta v - \omega)}{2\Delta v}\right)\Delta t & &\hat{i} = -\tfrac{1}{4}\rho\xi jk\Delta t
\end{aligned}$$

Solving this System of Coupled ODEs

Conceptually understanding how to represent a system of coupled ODEs where the stock price and the volatility must be determined completely on their domains for each time step proved to be very difficult. I decided to solve the system using a massive sparse martix. This method will be quite inefficient because the matrix will mostly have zeros. However, I do not know of another way to represent this problem so we will proceed with this idea.

We are trying to solve $Ax = b$, where A is a (M x N) x (M x N) matrix of the coefficients that is multiplied with x which is a vector with (M x N) entries for every stock price and volatility combination at the current step. b is the vector of every stock and volatility combination at the next time step.

We will now have a change of notation:

$$f_t^{(s,v)} = f(t, s, v)$$

This is how the vector x is structured:

$$x = \begin{bmatrix} f_t^{(0,0)} \\ f_t^{(\Delta s,0)} \\ \vdots \\ f_t^{(S_{max}-\Delta s,0)} \\ f_t^{(S_{max},0)} \\ f_t^{(0,\Delta V)} \\ \vdots \\ f_t^{(S_{max},\Delta v)} \\ \vdots \\ f_t^{(S_{max},V_{max})} \end{bmatrix}$$

Note how moving down the vector the option price reflects a changing stock price with a fixed volatility until S_{max} is reached. Then, volatility is incremented and we move along the domain of the stock price until the vector is filled with all M x N entries.

We finally populate A such that each element in b is defined by a linear combination of 9 elements in x with coefficients $\hat{a}$ through $\hat{\imath}$. This will be evident in the MATLAB code for this method.

This system of equations is solved at each time step, which causes the method to run slowly. Fortunately, the increased freedom we having in picking the step size makes using this method somewhat worthwhile.

Final Remarks on these Finite Difference Methods

I never explained why the methods we have developed are explicit or implicit. Let us first consider the explicit method. Since the time derivative uses the Backward Euler method, you may be wondering why exactly the method is explicit. We are trying to solve an initial value problem but the only value we know is the value of the option at its expiration (This is the second boundary condition). So we start at the end and work backwards until we reach the point when the contract was written. By using backward Euler in a "backward" way, our method is explicit and convergence will be an issue when it is applied in the later section.

The same reasoning causes our implicit formula to be implicit even though we are using forward Euler for the time derivative. We are starting at a time step where the value of the option is unknown and iterating through time to the value of the option at expiration. This means that our implicit method will require simeltaneously solving a system of coupled ODEs.

Earlier, I mentioned that there exists S_{max} and V_{max} when defining the domain of the variables. While in theory it is possible for the intervals on which the stock price and volatility are definied to be unbounded, more often than not it is perfectly resasonble to set some maximum value that you think is unlikely to be surpassed over the duration of the option contract. Upper bounds on the stock price and volatility are necessary to discretize the space, and without them we cannot get anything useful from the methods. In my opinion in is reasonable to assume that the stock price will rarely increase fivefold and that the volatility will not surpass the upper bound on the price of the stock. With this in mind, we continue to the last section where these are applied with market data.

Application of the Derived Methods to the Market

In this section, we will show how to test the the Stochastic Volatility Method and the Black-Scholes Option Pricing Formula using market data against the market price of an option.

Calibration of the Heston Volatility Model

The calibration of this method for use with market data is non-trivial, and in fact, it rivals the work needed to iterate over time. To calibrate, we will use a nonlienear least squares regression numerical method. This portion of the paper lies beyond my understanding of numerical methods applied to option pricing. So, we will borrow heavily from Nimalin Moodley's paper, "The Heston Model: A Practical Approach."

To continue, the following data is required: (See Market Data under Appendices.)

- Risk-free rate
- Contract Length of Option in Years
- Spot Price of Option
- Strike Price of Option
- Ask Price of Option
- Bid Price of Option
- Mid Price of Option

Moodley provides a calibration method in his paper that we will use directly. It utilizes MATLAB's lsqnonlin to optimize the following function:

$$\min_{\Omega} S(\Omega) = \min_{\Omega} \sum_{i=1}^{N} w_i \left[C_i^{\Omega}(K_i, T_i) - C_i^{M}(K_i, T_i) \right]^2$$

Where Ω is a vector of parameter values, C_i^{Ω} is the i-th option price from the model, C_M^{Ω} is the i-th option price from the market, K_i denotes the strike price, T_i denotes the contract length, N is the number of options, and w_i denotes a weight which is $\frac{1}{|bid-ask|}$. He manipulates the above equation in a way that makes calculating market risk unnecessary and somewhat makes it somewhat easier to minimize.

The motivation behind optimization of this function is simple to understand, but hard to implement. We take market data and work backwards to determine what θ, ω, v_0, ρ and ξ (the constants of the Heston model) cause the model to be correct. We then use the model with those values to predict future values with the hope that we have provided enough data to for the model to be reasonably useful. However, minimizing a five dimensional structure which has numerous non-differentiable points is difficult, and the fact that there may not be a unique solution to the optimization problems makes it even trickier.

In short, the optimization problem is ill-posed and the nonlinear least squares allows us to get the data we need to use it.

Conclusion

In the first section of the paper, we have derived a partial differential equation which models the price of an option assuming that the volaility of the option price is a stochastic process while also taking into account that the risk free rate changes over time. In the second section we derived two finite difference methods - an explicit one which is relatively fast but has stability issues if we are not careful with our step sizes and an implicit method which is very slow, but does not diverge under any step sizes. Finally, we mentioned how to put this model into practice by using a least squares method to solve the calibration problem.

Though the methods mentioned in the paper are functional, they are by no means the most efficient or accurate. One of the most promising ways to approximate the solution of the PDE involves Fast Fourier Transform. Gaussian quadrature is probably the most effective way to approximate the solution of the Heston model. The Heston model, however, is one of the few stochastic volatility models for which a closed-form solution integral even exists. For other choices of the α and β of the variance process, only numerical methods can solve them.

I would have liked to go into greater detail with the applications, but the matlab code that Moodley supplied does not work, so once I find the error I will include the application of the method.

Appendices

Explicit Finite Difference Code

```
% explicitdifference.m - UNTESTED Explicit Finite Difference method to evaluate the PDE
function c = explicitdifference(Smax,K,Vmax,r,T,dt,ds,dv,rho,xi,theta,omega)
% Smax = stock price ceiling
% K = option strike price
% Vmax = Volatility ceiling
% r = risk-free rate function
% t = ime contract expires
% dt = time step size
% ds = stock price step size
% dv = volatility step size
% rho = rho
% xi = xi
% theta = theta
% omega = omega

%Number of steps
L = round(T/dt)+1;
M = round(Smax/ds)+1;
N = round(Vmax/dv)+1;

%The f matrix is the value of the option.
f = zeros(L,N,M);

%Populate the matrix with initial values based on the boundary conditions.
stockval = 0:ds:Smax;
for i=1:M
    for j=1:N
        if stockval(i)>K
            %The Last Time step gets the value
            f(L,i,j) = stockval(i) - K;
        end
    end
end

%Main loop through the mesh, Indices and lengths correspond with the paper
for i=L-1:-1:1
    for j=1:M
        for k=1:N

           a = 1/4 * rho * xi * j * k * dt;
           b = (k*xi^2/(2*dv) + (theta*(k*dv-omega))/(2*dv))*dt;
           c = -1/4 * rho * xi * j * k * dt;
           d = (k*j^2*dv/2 - j*r(i)/2)* dt;
           e = 1 - (dv + xi^2/dv +r(i))*dt;
           f = (k*j^2*dv/2 + j*r(i)/2)* dt;
```

```
            g = -1/4 * rho * xi * j * k * dt;
            h = (k*xi^2/(2*dv) - (theta*(k*dv-omega))/(2*dv))*dt;
            I = 1/4 * rho * xi * j * k * dt;

            if j==1
                 f(i,j,k) = 0;
            else
                if j==M
                    %Time Value of Money
                    f(i,j,k) =  Smax-K*exp(-r*(L-i)*dT);
                else
                    f(i,j,k) = a*f(i+1,j-1,k-1) + b*f(i+1,j,k-1) + c*f(i+1,j+1,k-1) + ...
                        d*f(i+1,j-1,k) + e*f(i+1,j,k) + f*f(i+1,j+1,k) + g*f(i+1,j-1,k+1) + ...
                        h*f(i+1,j,k+1) + I*f(i+1,j+1,k+1);
                end
            end
         end
    end
end
c = f;
```

Implicit Finite Difference Code

```
% implicitdifference.m - UNTESTED Implicit Finite Difference method to evaluate the PDE
function c = implicitdifference(Smax,K,Vmax,r,T,dt,ds,dv,rho,xi,theta,omega)
% Smax = stock price ceiling
% K = option strike price
% Vmax = Volatility ceiling
% r = risk-free rate function
% t = ime contract expires
% dt = time step size
% ds = stock price step size
% dv = volatility step size
% rho = rho
% xi = xi
% theta = theta
% omega = omega

L = round(T/dt)+1;
M = round(Smax/ds)+1;
N = round(Vmax/dv)+1;

%The f matrix is the value of the option.
f = zeros(L,N,M);

%The A matrix is the matrix of coefficients
A = zeros(M*N,M*N);

%The x vector is the vector of all stock/volatility combos at t
```

```
x = zeros(M*N,1);

%The b vector is the vector of all stock/volatility combos at t + dt
b = zeros(M*N,1);

%Populate the b vector
for k = 1:N
    for j = 1:M
        if j*ds > K
            f(i,j,k) = j*ds - K;
            b(k*(M-1)+j,1) = f(i,j,k);
        end
    end
end

%Place the coefficients in the correct spot of the matrix
for k = 1:N
    for j = 1:M

       a = -1/4 * rho * xi * j * k * dt;
       b = -(k*xi^2/(2*dv) + (theta*(k*dv-omega))/(2*dv))*dt;
       c = 1/4 * rho * xi * j * k * dt;
       d = -(k*j^2*dv/2 - j*r/2)* dt;
       e = 1 + (dv + xi^2/dv +r)*dt;
       f = -(k*j^2*dv/2 + j*r/2)* dt;
       g = 1/4 * rho * xi * j * k * dt;
       h = -(k*xi^2/(2*dv) - (theta*(k*dv-omega))/(2*dv))*dt;
       I = -1/4 * rho * xi * j * k * dt;

       %INCLUDE TESTS AT BOUNDARIES
       A((k-1)*(M-1)+(j-1)) = a;
       A((k)*(M-1)+(j-1)) = b;
       A((k+1)*(M-1)+(j-1)) = c;
       A((k-1)*(M-1)+(j)) = d;
       A((k)*(M-1)+(j)) = e;
       A((k+1)*(M-1)+(j)) = f;
       A((k-1)*(M-1)+(j+1)) = g;
       A((k)*(M-1)+(j+1)) = h;
       A((k+1)*(M-1)+(j+1)) = I;

    end
end

%Begin time loop
%ADD CODE FOR TIME VALUE OF MONEY
%The solution
x = A\b;
%Put the solution in the f matrix
for k = 1:N
    for j = 1:M
        f(i,j,k) = x(k*(M-1)+j,1);
    end
```

```
end
%end time loop
```

Black-Scholes Code

```
% blackscholes.m — Black—Scholes formula for option pricing
function c = blackscholes(S,t,K,s,r)
% S = current stock price
% t = time until option expiration
% K = option strike price
% s = volatility
% r = risk—free rate
d1 = (log(S/K) + (r+s^2/2)*t)/(s*t^(.5));
d2 = d1 — s*t^(.5);
c = S*normcdf(d1)—K*exp(—r*t)*normcdf(d2);
```

Moodley's Calibration Code

```
clear;
global OptionData;
global NoOfOptions;
global NoOfIterations;
global PriceDifference;
NoOfIterations = 0;
load OptionData.m ;
%OptionData = [r,T,S0,K,Option Value,bid,offer]
Size = size(OptionData);
NoOfOptions = Size(1);
%input sequence in initial vectors [2*kappa*theta — sigma^2,...
% theta,sigma,rho,v0]
x0 = [6.5482 0.0731 2.3012 —0.4176 0.1838];
lb = [0 0 0 —1 0];
ub = [20 1 5 0 1];
options = optimset('MaxFunEvals',20000);
%sets the max no. of iteration to 20000 so that termination
%doesn?t take place early.
tic;
Calibration = lsqnonlin(@HestonDifferences,x0,lb,ub);
toc;
Solution = [(Calibration(1)+Calibration(3)^2)/ ...
(2*Calibration(2)), Calibration(2:5)];
```

```
function ret = HestonDifferences(input)
global NoOfOptions;
```

```
global OptionData;
global NoOfIterations;
global PriceDifference;
NoOfIterations = NoOfIterations + 1;
%counts the no of iterations run to calibrate model
for i = 1:NoOfOptions
PriceDifference(i) = (OptionData(i,5)-HestonCallQuad( ...
(input(1)+input(3)^2)/(2*input(2)),input(2), ...
input(3),input(4),input(5), ...
OptionData(i,1),OptionData(i,2),OptionData(i,3), ...
OptionData(i,4)))/sqrt((abs(OptionData(i,6)- ...
OptionData(i,7))));
%input matrix = [kappa theta sigma rho v0]
end
ret = PriceDifference';
```

Market Data

r	Term	Spot	Strike	Mid Price	Bid	Offer
2.27%	0.126027	$ 1,544.50	$ 1,000.00	$ 559.00	$ 553.00	$ 565.00
2.27%	0.126027	$ 1,544.50	$ 1,050.00	$ 509.50	$ 503.50	$ 515.50
2.27%	0.126027	$ 1,544.50	$ 1,100.00	$ 460.00	$ 454.00	$ 466.00
2.27%	0.126027	$ 1,544.50	$ 1,150.00	$ 411.00	$ 405.00	$ 417.00
2.27%	0.126027	$ 1,544.50	$ 1,200.00	$ 362.50	$ 356.50	$ 368.50
2.23%	0.375342	$ 1,544.50	$ 1,200.00	$ 386.00	$ 378.50	$ 393.50
2.27%	0.126027	$ 1,544.50	$ 1,250.00	$ 315.00	$ 309.00	$ 321.00
2.23%	0.375342	$ 1,544.50	$ 1,250.00	$ 345.50	$ 338.00	$ 353.00
2.27%	0.126027	$ 1,544.50	$ 1,300.00	$ 269.50	$ 263.50	$ 275.50
2.23%	0.375342	$ 1,544.50	$ 1,300.00	$ 300.50	$ 293.00	$ 308.00
2.27%	0.126027	$ 1,544.50	$ 1,350.00	$ 223.00	$ 217.00	$ 229.00
2.23%	0.375342	$ 1,544.50	$ 1,350.00	$ 259.00	$ 251.50	$ 266.50
2.19%	0.627397	$ 1,544.50	$ 1,350.00	$ 281.00	$ 272.00	$ 290.00
2.27%	0.126027	$ 1,544.50	$ 1,400.00	$ 179.00	$ 176.00	$ 182.00
2.23%	0.375342	$ 1,544.50	$ 1,400.00	$ 221.00	$ 213.50	$ 228.50
2.19%	0.627397	$ 1,544.50	$ 1,400.00	$ 244.00	$ 235.00	$ 253.00
2.27%	0.126027	$ 1,544.50	$ 1,450.00	$ 140.00	$ 136.00	$ 144.00
2.23%	0.375342	$ 1,544.50	$ 1,450.00	$ 180.00	$ 174.00	$ 186.00
2.19%	0.627397	$ 1,544.50	$ 1,450.00	$ 207.50	$ 198.50	$ 216.50
2.27%	0.126027	$ 1,544.50	$ 1,500.00	$ 105.00	$ 102.00	$ 108.00
2.23%	0.375342	$ 1,544.50	$ 1,500.00	$ 149.50	$ 145.00	$ 154.00
2.19%	0.627397	$ 1,544.50	$ 1,500.00	$ 173.00	$ 166.00	$ 180.00
2.27%	0.126027	$ 1,544.50	$ 1,600.00	$ 56.50	$ 51.50	$ 61.50
2.23%	0.375342	$ 1,544.50	$ 1,600.00	$ 96.00	$ 92.00	$ 100.00
2.19%	0.627397	$ 1,544.50	$ 1,600.00	$ 121.00	$ 114.00	$ 128.00
2.27%	0.126027	$ 1,544.50	$ 1,700.00	$ 23.50	$ 20.50	$ 26.50
2.23%	0.375342	$ 1,544.50	$ 1,700.00	$ 57.25	$ 51.00	$ 63.50
2.19%	0.627397	$ 1,544.50	$ 1,700.00	$ 81.50	$ 77.00	$ 86.00
2.27%	0.126027	$ 1,544.50	$ 1,800.00	$ 10.00	$ 7.00	$ 13.00
2.23%	0.375342	$ 1,544.50	$ 1,800.00	$ 32.50	$ 28.00	$ 37.00
2.19%	0.627397	$ 1,544.50	$ 1,800.00	$ 50.50	$ 44.50	$ 56.50
2.27%	0.126027	$ 1,544.50	$ 1,900.00	$ 4.50	$ 2.00	$ 7.00
2.23%	0.375342	$ 1,544.50	$ 1,900.00	$ 18.25	$ 14.50	$ 22.00
2.19%	0.627397	$ 1,544.50	$ 1,900.00	$ 35.50	$ 29.50	$ 41.50

References

[1] F. Black and M. Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, (81):631–659, 1973.

[2] Gang Dong. Numerical methods of pde in computational finance. Rutgers University.

[3] Jim Gatheral. *The Volatility Surface: A Practitioner's Guide*. Jon Wiley & Sons, Inc., 2006.

[4] Steven L. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *The Review of Financial Studies*, 6(2):327–343, 1993.

[5] Nimalin Moodley. *The Heston Model: A Practical Approach with Matlab Code*. University of the Witwatersrand, 2005.