Pricing European-style options using Geometric Fractional Brownian Motion and a Monte Carlo Approach

Esther Yue

Overview

Options are a type of financial instrument that provide the owner with the right, but not the obligation, to purchase or sell the underlying asset at a specified price (known as the strike price K) on or before a certain date (known as the expiration date). There are two main types of options that exist on the market: **call options**, which give the buyer the right to purchase the underlying security at strike price K on or before the expiration date, and **put options**, which give the buyer the right to sell the underlying security. With both types of options, the buyer pays an upfront premium to acquire the right to buy or sell the underlying asset in the future, and the premium represents the market price in which the option is currently valued.

The unpredictability of the option's profitability at expiration contributes to the challenge in pricing an option, given that the upfront premium amount should be dependent on the probability of profitability for the buyer. Many uncertain, random, and changing factors must be taken into consideration when determining the fair market value of an option, such as the current asset price, its intrinsic value, the length of the contract/time to expiration, the potential volatility of the asset, and interest rates. The Black-Scholes model is most frequently used to price European-style options, which are options that can only be exercised at their expiration date. On the other hand, American-style options, which provides the buyer with the additional benefit of the right to exercise their option before the expiration date, use a variety of different models to determine their fair value, such as approximations, lattice and finite difference methods, and Monte Carlo methods.[1]

Monte Carlo Simulations

Monte Carlo simulations are a mathematical technique typically used to predict the range of possible outcomes of a certain phenomenon through repeated random sampling. The simulation factors in uncertainties and the intervention of random variables, and in finance, these simulations can be helpful in predicting the value of an asset given the influence of many different economic variables. In the context of options, Monte Carlo simulations can be used to value options through modeling out the potential movement of an asset with multiple sources of uncertainties.

In this paper, I will focus on the application of Monte Carlo simulations in pricing European-style options, as well as incorporating various features of Geometric Fractional Brownian Motion in the model.

[1] http://www.diva-portal.org/smash/get/diva2:479155/FULLTEXT01.pdf

Geometric Fractional Brownian Motion

The movement of an underlying security tends to follow geometric brownian motion, which is a continuous-time stochastic process characterized by the following stochastic differential equation:

$$dS_t = \mu S_t dt + \sigma S_t dW_t$$

In this equation, S_t represents the stochastic process, and in the context of options pricing, the option price. μ represents the drift, which is the expected return of the asset over time period T. σ represents the stochastic volatility of the stochastic process.

The term $\mu S_t dt$ controls the overall direction or "trend" of the asset's path, while $\sigma S_t dW_t$ represents the random "noise" generated. W_t represents the Wiener process, a random walk with mean 0 and variance t, summarized as $W_t \sim N(0, t) = \sqrt{t}\, N(0, 1)$. Applying Ito's Lemma to the equation, the solution for S_t given by an arbitrary initial value S_0 is:

$$S_t = S_0 e^{(\mu - \frac{\sigma^2}{2})t + \sigma W_t}$$

However, recent research has shown that geometric fractional brownian motion (GFBM) may be more accurate in modeling stock price movements, as GBM modeling assumes successive movements of assets to be mutually independent.[2][3] It has been observed that the price movements of an asset tend to exhibit self-similarity and dependency, meaning that future movements can be somewhat influenced by the trend of existing movements. Several studies have demonstrated that asset returns in financial markets tend to be multifractal in nature, showing power-law scaling instead of demonstrating mutually independent movements.[4]

GFBM incorporates this self-similarity feature through the inclusion of the Hurst exponent. The Hurst exponent indicates the intensity of long-range dependence in the time series and overall captures volatility persistence within the stochastic process, which can be represented through variable $H \in (0, 1)$. Further details about the Hurst exponent are included under the Parameter Estimation section.

The Hurst exponent is incorporated in the Wiener process W_t. As a result, our formula for the option price becomes:

$$S_t^H = S_0 e^{(\mu - \frac{\sigma^2}{2})t + \sigma X_t^H}$$

---

2 https://www.sciencedirect.com/science/article/pii/S111001682030541X

3 https://www.hindawi.com/journals/isrn/2014/791418/

4 https://link.springer.com/chapter/10.1007/978-4-431-66993-7_20

In the GFBM model, if we represent S_0 as S_t and assess the returns over time t + Δt, the logarithmic returns of the asset with price S_t during time [t, t + Δt] can be represented as:

$$R_t^H = \ln \frac{S_{t+\Delta t}^H}{S_t^H} = (\mu - \frac{\sigma^2}{2})\Delta t + \sigma X_t^H$$

where:

$$X_t^H = W_{t+\Delta t}^H - W_t^H,\ t \geq 0$$

Parameter Estimation

In a GBM model, running a Monte Carlo simulation typically requires us to estimate two parameters: the drift μ and stochastic volatility σ. However, incorporating the GFBM model requires us to first estimate the Hurst exponent.

Hurst Exponent

The value of the Hurst exponent can characterize price movement in three scenarios:

- If H ∈ (0, ½), each increment is positively correlated, and the movement is characterized as exhibiting short-term memory dependency
- If H = ½, each increment is considered independent from one another, and the movement is characterized as classic Brownian motion
- If H ∈ (½, 1), each increment is negatively correlated, and the movement is characterized as exhibiting long-term memory dependency[5]

There exist numerous methods to estimate the Hurst exponent, namely the rescaled range analysis method (simple, no underlying assumptions needed), the Higuchi method, the periodogram method, and the variance method. For simplicity, we will use the rescaled range method, though this method has its drawbacks with its high sensitivity to short-range dependency.[6]

First, for the time series of length n, $X_1, \ldots, X_n$, we calculate the mean:

$$m = \frac{1}{n}\sum_{i=1}^{n} X_i$$

Then, we create a mean-adjusted series and calculate the resulting cumulative deviation series:

$$Y_t = X_t - m \qquad \text{for } t = 1, 2, \ldots, n$$
$$Z_t = \textstyle\sum_{i=1}^{t} Y_i \qquad \text{for } t = 1, 2, \ldots, n$$

---

[5] https://www.sciencedirect.com/science/article/pii/S111001682030541X

[6] https://mpra.ub.uni-muenchen.de/16424/2/MPRA_paper_16424.pdf

The range deviation series is then characterized as follows:

$$R(n) = \max(Z_1, Z_2, \ldots, Z_n) - \min(Z_1, Z_2, \ldots, Z_n)$$

Utilizing these values, the resulting standard deviation can be calculated:

$$S(n) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(X_i - m)^2}$$

The rescaled range can be calculated as **R(n)/S(n)**.

Given that as the value of n increases, R(n) grows by a factor of n while S(n) converges to some constant value C, the Hurst exponent is estimated by fitting the following equation to the data:

$$E[\frac{R_n}{S_n}] \approx C \cdot n^H$$

Therefore, to derive H, we can take the natural logarithm of R(n)/S(n) and plot it as a function of ln(n), which provides us a linear regression with Hurst exponent H as the slope value.

$$ln\left[\frac{R_n}{S_n}\right] \approx ln(C) \cdot H\ ln(n) \qquad \text{for n = 1, 2, …}$$

Volatility

In a typical GBM model, volatility σ can be calculated by obtaining the standard deviation of the logarithmic returns of the asset over time period T:

$$\sigma = \frac{1}{n-1}\sqrt{\sum_{i=1}^{n}(R_{ti} - \mu)^2}$$

Volatility can also be determined through calculating the asset's implied volatility using the Black-Scholes model. Since this value is not explicitly known, the Newton Raphson method can be used to loop through potential implied volatility numbers much faster than taking a brute force approach.[7]

In the following formula, x represents the implied volatility value, f(x) represents either the difference between the current asset price and the option price (call option) or the difference between the option price and the current price (put option), and the derivative of x represents the option price sensitivity to the implied volatility:

---

[7] https://pythoninoffice.com/calculate-option-implied-volatility-in-python/

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

This equation is looped until an appropriate value x is found: $\sigma = x$.

However, for a GFBM model, the Hurst exponent must be incorporated into the standard deviation of the logarithmic returns. Our volatility parameter therefore becomes:[8]

$$\hat{\sigma} = \sqrt{\frac{\sigma^2}{|\Delta t|^{2H}}}$$

Here, $\Delta t = t_2 - t_1 = \ldots = t_n - t_{n-1}$.

Drift

As the drift parameter μ represents the expected return for the time period, after calculating the Hurst exponent H and volatility parameter $\hat{\sigma}$, we can find the resulting drift for the GFBM model after calculating the mean logarithmic returns:

$$\mu = \frac{1}{n}\sum_{i=1}^{n} R_{ti}$$

$$\hat{\mu} = \frac{\mu}{\Delta t} + \frac{\hat{\sigma}^2}{2}$$

Another option is to assume a risk-neutral environment and approximate the drift parameter to the risk-free rate r.[9] In this simulation, we will assess both scenarios in which the drift μ equals the average historical return over various timeframes, as well as $\mu = r$.

[8] http://www.diva-portal.org/smash/get/diva2:1257290/FULLTEXT01.pdf

[9] https://www.hindawi.com/journals/jps/2011/595741/

Fractional Gaussian Noise

To run the GFBM model, we need to be able to simulate the incremental "noise" process, called fractional Gaussian noise (FGN), which will determine potential paths that the asset price will travel. This is represented by the X_t^H term in the GFBM stochastic equation. Through simulating a finite sequence of FGN dependent on the value of the derived Hurst exponent of historical returns within a variety of time frames, we can extract a set of fractional Brownian motion values that can be inputted into the stochastic equation. Simulating FGN values can be obtained through a decomposition of the covariance matrix C of an n-dimensional Gaussian variable, then obtaining a n-row vector with independent and identically distributed entries:

$$C = \begin{pmatrix} a_0 & a_1 & \cdots & a_{n-2} & a_{n-1} \\ a_1 & a_0 & \cdots & a_{n-3} & a_{n-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n-2} & a_{n-3} & \cdots & a_0 & a_1 \\ a_{n-1} & a_{n-2} & \cdots & a_1 & a_0 \end{pmatrix}$$

where $a_k = \frac{1}{2}(|k-1|^{2H} - 2|k|^{2H} + |k+1|^{2H})$[10]

The Cholesky decomposition and the Davies and Harte decomposition methods are both used to simulate FGN values, however we will be using the Cholesky method in this simulation.

Payoff Model

After parameter estimation, we can implement a simple function to model out the payoff of both call and put options. For a call option, the payoff equals the **spot price - strike price**, where the spot price refers to the call option's price at the time of expiration; however, if the spot price is lower than the strike price, the payoff is 0. We can write this out as:

$$C_P = \max(S_0 - K, 0)$$

For the payoff equals the maximum of **strike price - spot price**, 0:

$$P_P = \max(K - S_0, 0)$$

After obtaining the payoff for each simulation result, we simply discount the value back to the present, assuming continuous compounding, by multiplying each value by $\mathbf{e^{-rT}}$. By feeding the possible paths through the payoff model, we can simulate the different possible values of the option by its expiration date.

Black-Scholes Model

---

10 https://raw.githubusercontent.com/732jhy/Fractional-Brownian-Motion/master/Cholesky.py

The Black-Scholes model can be an effective way to determine the accuracy of our Monte Carlo model for a call option. We can input our parameter estimations into the model:

$$C = N(d_1)S_t - N(d_2)Ke^{-rt}$$

where $d_1 = \frac{\ln\frac{S_t}{K}+(r+\frac{\sigma^2}{2})t}{\sigma\sqrt{t}}$, $d_2 = d_1 - \sigma\sqrt{t}$

C = call option price
N = CDF of the normal distribution
S_T = spot price
K = strike price
r = risk-free rate
t = time to maturity
σ = asset volatility

[to be updated]

Sources:

http://www.diva-portal.org/smash/get/diva2:479155/FULLTEXT01.pdf
https://mpra.ub.uni-muenchen.de/16424/2/MPRA_paper_16424.pdf
https://www.hindawi.com/journals/jps/2011/595741/
https://link.springer.com/chapter/10.1007/978-4-431-66993-7_20
https://www.hindawi.com/journals/isrn/2014/791418/
https://www.research.manchester.ac.uk/portal/files/54503314/FULL_TEXT.PDF
https://pythoninoffice.com/calculate-option-implied-volatility-in-python/
http://www.diva-portal.org/smash/get/diva2:1257290/FULLTEXT01.pdf

Tutorials:
https://www.sciencedirect.com/science/article/pii/S111001682030541X
http://www.diva-portal.org/smash/get/diva2:1257290/FULLTEXT01.pdf
https://towardsdatascience.com/monte-carlo-pricing-in-python-eafc29e3b6c9
https://github.com/732jhy/Fractional-Brownian-Motion

American Options:
https://www.diva-portal.org/smash/get/diva2:301061/FULLTEXT01.pdf
http://www.diva-portal.org/smash/get/diva2:479155/FULLTEXT01.pdf

Greeks:
http://www.smileofthales.com/wp-content/uploads/2015/04/Greeks1.pdf
https://www.investopedia.com/trading/using-the-greeks-to-understand-options/