Monte Carlo Simulation for Option Pricing

Nikola Petrovski [216051]

1 Introduction

Option pricing is a crucial aspect of financial markets, enabling investors to assess the value and risks associated with derivative contracts. Traditionally, the Black-Scholes model has been a foundation in this domain, providing a solution for European-style options. However, as financial markets evolve and become increasingly complex, alternative methods such as Monte Carlo simulation have gained prominence for their ability to handle a wider range of scenarios and factors.

Monte Carlo simulation is a technique used in modeling and simulations that relies on repeated random sampling to obtain numerical results [1]. In the context of option pricing, Monte Carlo simulation involves simulating multiple possible future price paths of the underlying stock and calculating the option's payoff for each one. By aggregating these payoffs and discounting them back to the present, the expected value of the option can be estimated.

One of the primary advantages of Monte Carlo simulation is its flexibility in modeling various sources of uncertainty, such as volatility clustering, jumps, and stochastic interest rates, which may not be adequately captured by the assumptions underlying the Black-Scholes model. Additionally, Monte Carlo simulation can be applied to a wide range of option types, including path-dependent and exotic options, for which analytical solutions may not exist.

This project presents a thorough examination of option pricing methodologies, contrasting Monte Carlo simulation with the Black-Scholes model. It begins with an introduction discussing what are options and their significance in financial markets. The modeling of stock trajectories using Geometric Brownian Motion (GBM) is explored, alongside Monte Carlo simulation techniques for stock price prediction. The Black-Scholes model is then introduced, outlining its formula and assumptions. A methodology for comparing the two approaches is given, leading to the presentation and analysis of results, which are subsequently discussed. Finally, the project concludes by summarizing key findings and providing insights into the applicability of each method in various scenarios.

2 Options

An option is a financial instrument that gives the holder the right, but not the obligation, to buy or sell an underlying asset, such as a stock, at a predetermined price within a specified time period [2]. The buyer of the option pays a premium to the seller, who takes on the obligation to fulfill the terms of the option if the buyer chooses to exercise it.

Based on the type of the option's right, there are 2 types of options:

- Call Option - Right to buy an asset
- Put Option - Right to sell an asset

In this project we will focusing on European style Call Option, which gives the holder the right (but not the obligation) to buy an asset at a fixed price, the strike price K, at the expiry date T.

2.1 Importance of Option Pricing

Option pricing holds immense importance in financial markets as it enables investors to make informed decisions regarding the valuation and management of their portfolios. Accurately pricing these derivatives is crucial for determining their fair value and assessing the associated risks. Proper option pricing allows investors to hedge against adverse market movements, speculate on future price movements, and construct complex trading strategies. Moreover, option pricing plays a vital role in capital allocation, risk management, and the overall efficiency of financial markets. In essence, understanding and accurately pricing options is essential for maintaining market stability and facilitating efficient capital allocation, making it a fundamental aspect of modern finance.

3 Modeling a Stock Trajectory

Stocks, also known as capital stock or shares, represent ownership stakes in a corporation or company [3]. Each share signifies a fractional ownership of the company, entitling the shareholder to a corresponding portion of earnings, assets upon liquidation, and voting power. The value of a stock can fluctuates based on multiple key factors like company performance, market conditions, and investor sentiment. It's determined by supply and demand dynamics, financial indicators, and external events.

Given the stochastic nature and time-series data associated with stock values, mathematical models are employed to analyze their behavior. One such model is Geometric Brownian Motion (GBM), also known as Exponential Brownian Motion [4]. GBM assumes a constant drift (μ) and volatility (σ), providing a framework to predict stock price movements over time. The value of a stock at any given time (t) can be described using (1), which incorporates Brownian Motion (W_t) to account for randomness in the stock's price evolution.

$$dS_t = \mu S_t dt + \sigma S_t dW_t \tag{1}$$

3.1 Solving the Stochastic Differential Equation (SDE)

In order to solve the SDE presented in (1) we will be using Itô's lemma, which is based a stochastic calculus called Itô calculus. Itô's lemma states the following [5, 6]:

Let $B(t)$ be a Brownian motion and $W(t)$ be an Ito drift-diffusion process which satisfies the stochastic differential equation (2).

$$dW(t) = \mu(W(t), t)dt + \sigma(W(t), t)dB(t) \tag{2}$$

If $f(w, t) \in C^2(R^2, R)$ then $f(W(t), t)$ is also an Ito drift-diffusion process, with its differential given by (3).

$$d(f(W(t), t)) = f'(W(t), t)dW + \frac{1}{2}f"(W(t), t)dW(t)^2 \tag{3}$$

When applying Ito's lemma to the Geometric Brownian Motion S_t, described using (1), and the function $f(S_t) = \ln(S_t)$, we get (4).

$$\begin{aligned} d(\ln(S_t)) &= (\ln(S_t))'dS_t + \frac{1}{2}(\ln(S_t))"dS_t^2 \\ d(\ln(S_t)) &= \frac{dS_t}{S_t} - \frac{1}{2S_t^2}dS_t^2 \end{aligned} \tag{4}$$

To solve the equation, we first need to simplify the value for dS_t^2. To simplify the expression, we first use (1) as a substitution to get (5).

$$dS_t^2 = (\mu S_t dt + \sigma S_t dW_t)^2 = \mu^2 S_t^2 dt^2 + \mu\sigma S_t^2 dtW_t + \sigma^2 S_t^2 dW_t^2 \tag{5}$$

When $dt \to 0$, dt converges to 0 faster than dW_t, since $dW_t^2 = O(dt)$ [4]. Due to this, we get that the value for dS_t^2 simplifies to (6).

$$dS_t^2 = \sigma^2 S_t^2 dt \tag{6}$$

After simplifying the value for dS_t^2 and applying the substitution from (1) and (6) to (4), we get the expression in (7).

$$\begin{aligned} d(\ln(S_t)) &= \frac{dS_t}{S_t} - \frac{1}{2S_t^2}dS_t^2 \\ d(\ln(S_t)) &= \frac{\mu S_t dt + \sigma S_t dW_t}{S_t} - \frac{\sigma^2 S_t^2 dt}{2S_t^2} \\ d(\ln(S_t)) &= \mu dt + \sigma dW_t - \frac{\sigma^2}{2}dt \\ d(\ln(S_t)) &= (\mu - \frac{\sigma^2}{2})dt + \sigma dW_t \end{aligned} \tag{7}$$

Integrating both sides the final expression in (7), we get the final solution to the SDE illustrated in (8).

$$\begin{aligned} \int d(\ln(S_t)) &= \int (\mu - \frac{\sigma^2}{2})dt + \int \sigma dW_t \\ \ln(\frac{S_t}{S_0}) &= (\mu - \frac{\sigma^2}{2})t + \sigma W_t \end{aligned} \tag{8}$$

Cleaning up the solution in (8) we get the solution for S_t in (9).

$$\begin{aligned} \frac{S_t}{S_0} &= \exp\{(\mu - \frac{\sigma^2}{2})t + \sigma W_t\} \\ S_t &= S_0 \exp\{(\mu - \frac{\sigma^2}{2})t + \sigma W_t\} \end{aligned} \tag{9}$$

Using the solution for S_t in (9), we get the following discretized form in (10) for calculating the change of the value for a given difference of Δt between 2 time itervals, where $z \sim \mathcal{N}(0,1)$.

$$\begin{aligned} S_i &= S_{i-1} \exp\{(\mu - \frac{\sigma^2}{2})\Delta t + \sigma W_t\} \\ S_i &= S_{i-1} \exp\{(\mu - \frac{\sigma^2}{2})\Delta t + \sigma \sqrt{\Delta t} z\} \end{aligned} \tag{10}$$

3.2 Example of a simulated stock path

By employing the discretized form of a stock modeled using Geometric Brownian Motion (GBM), as presented in (10), we can construct a concise script capable of generating random stock paths, as exemplified in Listing 1.

Listing 1: Stock Path Generation Code

```python
import numpy as np

dt = T / 365
price_path = [S0]

for _ in range(356):
    z = np.random.standard_normal()
    price_path.append(
        price_path[-1]
        * np.exp(
            (mu - 0.5 * sigma**2)
            * dt + sigma
            * np.sqrt(dt)
            * z
            )
    )
```

Utilizing the code provided in Listing 1 alongside the specified parameters outlined in Tab 1, we conducted simulations to generate a simulated stock path. The resultant path is visually represented in Fig 1. This visualization offers insights into the stochastic nature of stock movements and illustrates how the GBM model captures the fluctuating behavior of stock prices over time.

Parameter	Value
Initial Stock Price (S_0)	100.00
Risk-free Interest Rate (μ)	0.05
Volatility (σ)	0.20
Time to maturity in years (T)	1.00
Number of time intervals	365.00

Table 1: Parameters used for stock path model

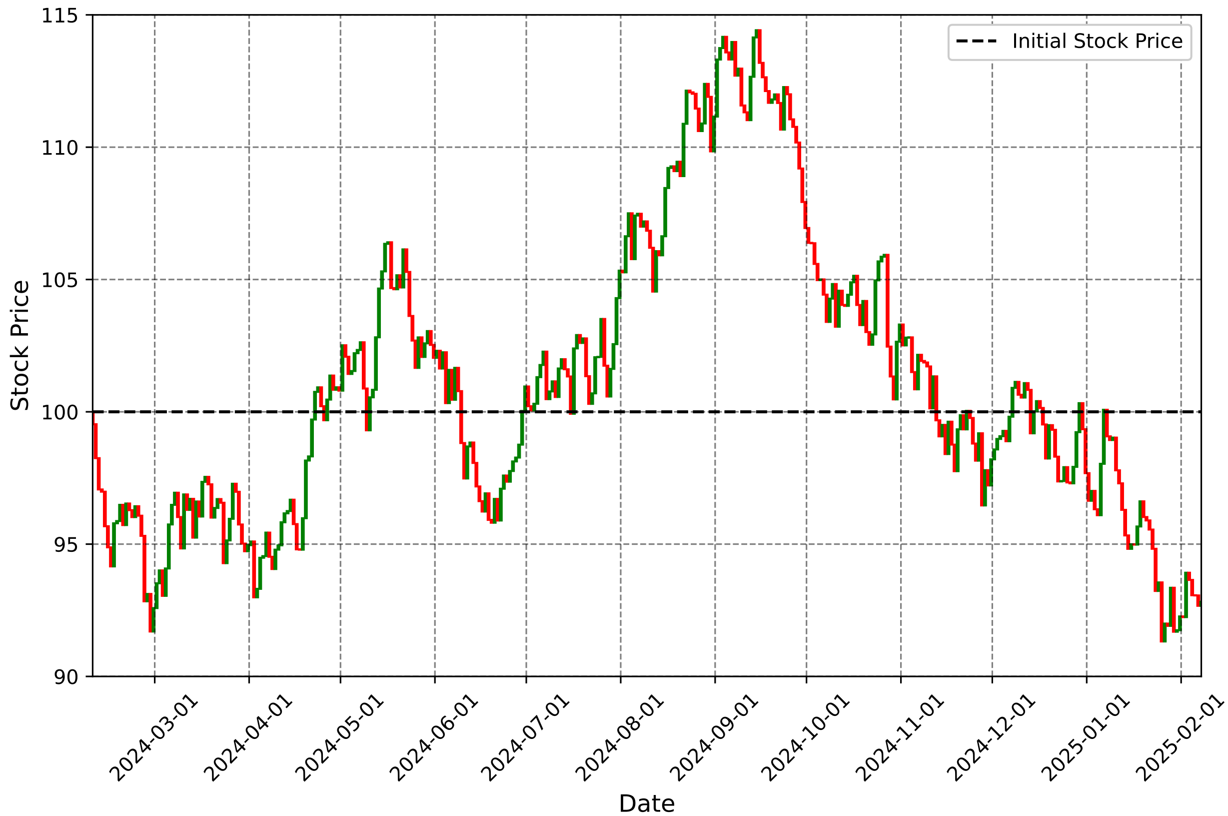

Figure 1: Candlestick Plot of Simulated Stock Path

4 Monte Carlo Simulation for Stock Price

The Monte Carlo simulation, a mathematical method, forecasts potential outcomes of uncertain events. Utilized by computer programs, it scrutinizes historical data to anticipate a spectrum of future outcomes contingent on selected actions [1].

Predicting the price of an option requires a complex computation rooted in the distribution of stochastic processes, where randomness plays a pivotal role. In this context, the Monte Carlo simulation emerges as a powerful tool for analysis and prediction. By simulating numerous possible future scenarios, it provides a robust framework for evaluating the potential value of an option across a spectrum of market conditions.

The initial step in predicting the price of an option involves simulating the price trajectory of the underlying stock at the option's expiration. To accomplish this, we employ the methodology outlined in Section 3 and iteratively apply it using the Monte Carlo simulation technique. This iterative process ensures a comprehensive exploration of potential stock price paths. Furthermore, to obtain a reliable estimate of the stock's value at time T, we iterate the simulation procedure over 1000 iterations, each executed independently to ensure robustness and accuracy in our valuation.

4.1 Optimization with Pandas' DataFrame

Due to the time critical nature of price prediction a simple loop iteration for performing the Monte Carlo simulation may not present a sufficient way of generating the stock's values. To improve the performance of the simulation, we recommend using the Pandas library, primarily the DataFrame data structure, as operation performed on it are vectorized [7].

To determine the optimization gain of Pandas' DataFrame vectorization, we tested the process of performing 10000 simulations using the Monte Carlo simulation method with a basic loop iteration and with DataFrame vectorized operation on a stock with the parameters given in Tab 1. Additionally, the main metrics we used are total execution time needed (seconds) and Improvement Factor (%), calculated based on (11) where T_L is the loop execution time and T_D is the DataFrame vectorized execution time.

$$IF = \frac{T_L - T_D}{T_L} \tag{11}$$

Based on the results presented in Tab 2, we can see that the DataFrame method outperforms the standard loop iteration method and achieves an incredible improvement factor of 70.66%.

	Results Value
Loop Time (seconds)	14.22
DataFrame Time (seconds)	4.17
Improvement Factor (%)	**70.66**

Table 2: Results achieved from DataFrame method testing

4.2 Option Payoff and Price Calculation

To calculated the option payoff of a call option based on a given strike price K (without an option premium) we use (12). The reason for using max of the difference of the stock price $S(T)$ at time T and the strike price K and 0 is that if the option will make us lose money, we choose not to exercise the right of the option.

$$O_p = \max\left(0, S(T) - K\right) \tag{12}$$

Once we calculate the option payoff for each simulated stock path, we need to calculate the value of the option price. To extract the price information we use (13), where $\bar{O}_p$ is the mean option payoff for each simulated trajectory.

$$OP = e^{-\mu T}\bar{O}_p \tag{13}$$

4.3 Monte Carlo Simulation Results

Utilizing the Monte Carlo Simulation method with the stock parameters detailed in Tab 1, we generated simulated stock trajectories, as depicted in Fig 2. Analysis of these results reveals an increase in the mean stock value, ascending from 100.00$ to 105.18$ over time, accompanied by a corresponding escalation in the standard deviation, indicative of heightened randomness in the stock's behavior.

Subsequent to evaluating the simulated stock paths, we derived the option payoff distribution, illustrated in Fig 3, employing a strike price of 105.00$. Based on this results, our analysis yielded an option price of 7.92$.

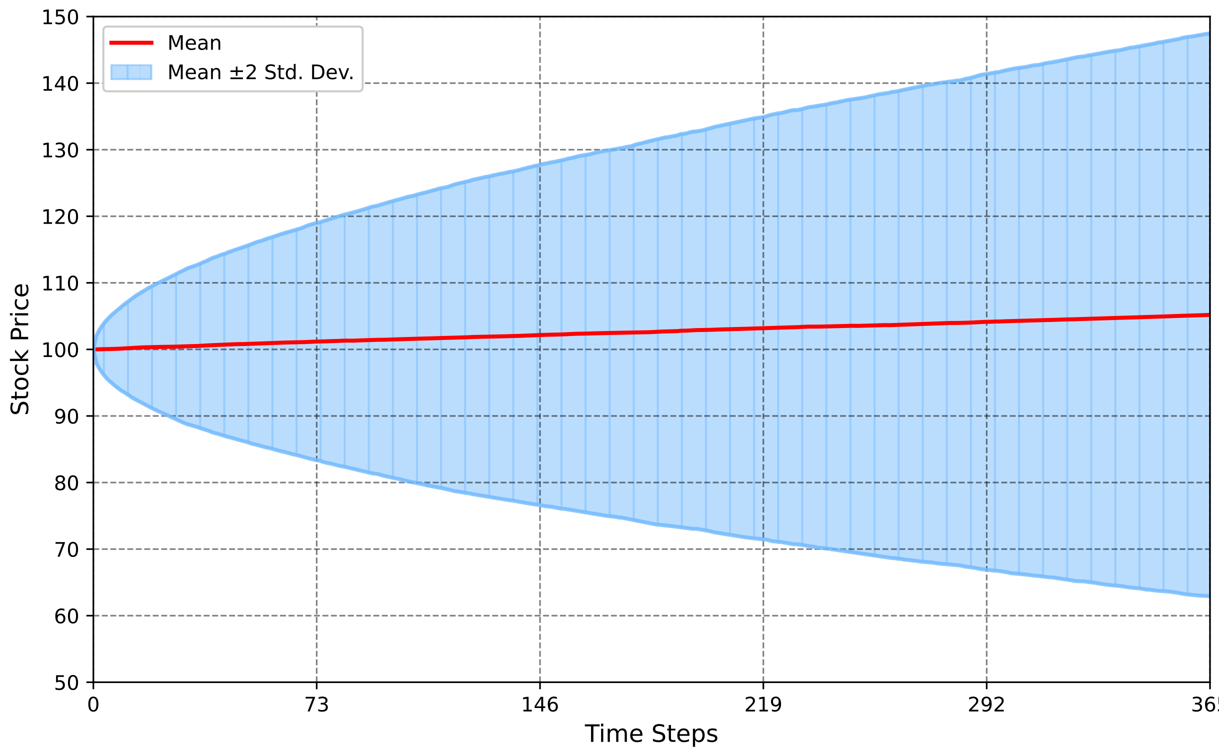

Figure 2: Aggregate Information for Simulated Stock Trajectories

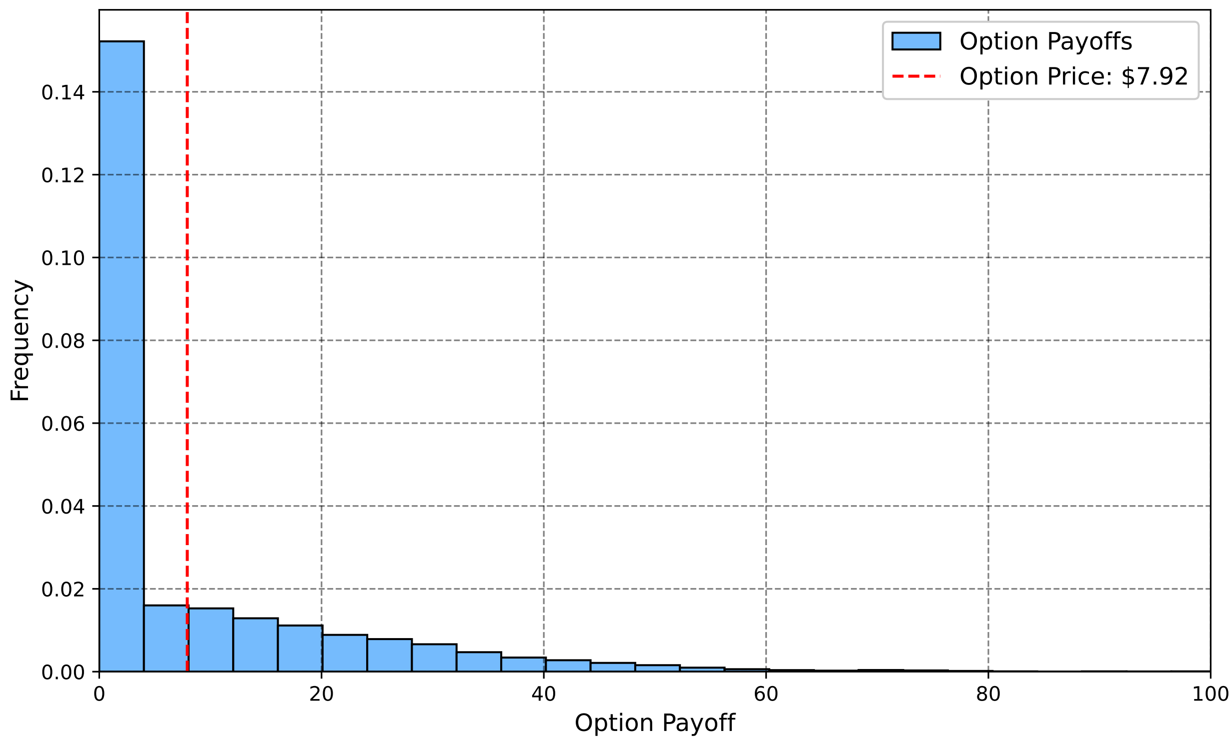

Figure 3: Distribution of Option Payoffs

5 Black-Scholes Model

The Black–Scholes model [8] serves as a mathematical framework to comprehend the dynamics of derivative investments in financial markets. Utilizing the Black–Scholes equation, a parabolic partial differential equation, the model derives the Black–Scholes formula, given in (14) (with the meaning of the parameters explained in Tab 3). This formula offers a theoretical valuation of European-style options, illustrating that each option possesses a distinct price determined by the security's risk. The expected return is replaced by the risk-neutral rate in calculations, highlighting the unique pricing of options within this framework.

$$\begin{aligned} C &= N(d_1)S_0 - N(d_2)Ke^{-rt} \\ d_1 &= \frac{ln(\frac{S_0}{K}) + (r + \frac{\sigma^2}{2})t}{\sigma\sqrt{t}} \\ d_2 &= d_1 - \sigma\sqrt{t} \end{aligned} \tag{14}$$

6 Comparison Methodology

To evaluate the similarity between the two option pricing methods, we will conduct a series of 200 experiments using generated input parameters. After option price generation for each

Parameter	Meaning
C	Option Price
N	CDF of Normal Distribution
S_0	Initial Stock Price
K	Strike Price
r	Risk-Free Interest Rate
t	Time to Maturity
σ	Volatility

Table 3: Explanation of Black-Scholes Formula's Parameters

input array, we will be comparing the prices obtained with both methods using multiple metrics.

The metrics selected for the comparison of the two option pricing methods include:

- Coefficient of determination (R^2)
- Mean squared difference (MSD)
- Mean absolute difference (MAD)
- Root mean squared difference (RMSD)

6.1 Simulation of Input Data

Because we were unable to find publicly available datasets which incorporate the parameters utilized by the methods, we resort to generating simulated data points for the purpose of comparing the presented option pricing models. The parameters that we will need to simulate are presented in Tab 4.

Parameter	Meaning
S_0	Initial Stock Price
K	Strike Price
T	Time to maturity (in years)
r	Risk-free Interest Rate
σ	Volatility

Table 4: Models' Parameters that need to be generated

As the main parameter for a stock trajectory is the starting point (the initial stock price S_0), we first begin with its simulation. To simulate a range of "different" stocks we choose to generate the initial stock price by sampling a Normal distribution with parameters $\mu = 100$ and $\sigma = 10$. Afterwards, we move onto to simulating the Strike price K which presents the fixed price at which the option needs to be exercised. As the strike price needs to be calculated using option pricing methods we used the assumption that the strike price is calculated using (15), where $\alpha \sim \mathcal{N}(5, 2.5)$.

$$K = S_0 + |\alpha| \tag{15}$$

Following the initial and strike price, we will generate the Time to maturity T and the Risk-free Interest Rate r together. As risk-free interest rates are usually calculated based on government treasury notes [9] will be using US Treasury Yield for T years. Additionally, due to the link between the Risk-free Interest rate and the parameter T, we will be sampling them in pairs from uniform distribution. The distribution will consist of the values presented in Tab 5, which were extracted from [10] on 11th of January.

Time to Maturity (years)	**Risk-Free Interest Rate (%)**
0.5	5.21
1.0	4.74
2.0	4.25
5.0	3.88
10.0	3.97

Table 5: Combinations of Parameters T and r

The last parameter that needs to be simulated is the volatility (σ) of the underlying stocks. The generation process for σ lies in sampling from a Lognormal distribution with parameters $\mu = 0$ and $\sigma = 1$. The reason from sampling from a Lognormal is based on the work of Pierre Cizeau et al. [11], which found that the volatility of the S&P 500 (from 1984 to 1996) follows a Lognormal distribution.

6.2 Visualization of Input Data

Using the parameters' simulation methods previously outlined, we achieved the following value distribution, presented in Fig 4, for the Initial Price, Strike Price and Volatility. Based on the extracted results we can see that we achieved the parameters to follow the previously presented distributions.

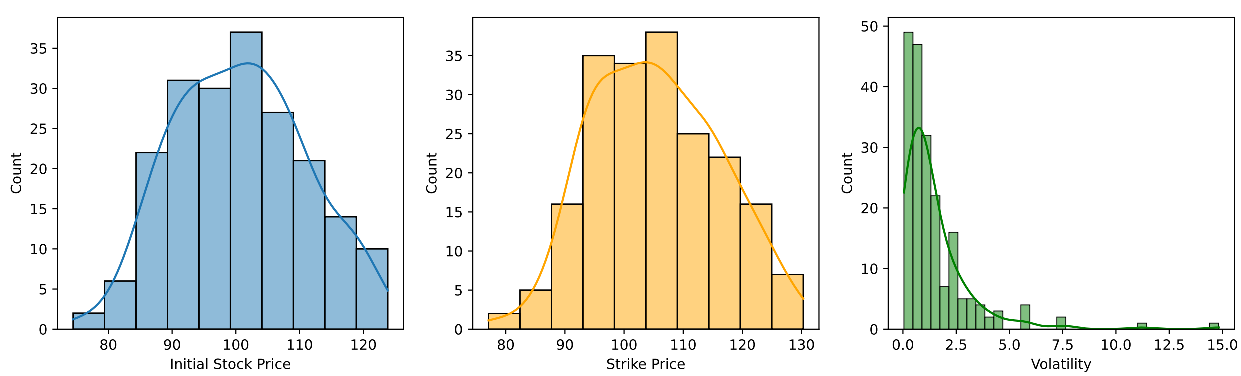

Figure 4: Distribution of Continuous Parameters

7 Results

Using both methods we achieved the resulting option prices presented in Fig 5. Based on these results we can see that the Monte Carlo Simulation model generates lower evaluations for the option prices, when compared to the prices generated for the same stocks using the Black-Scholes model.

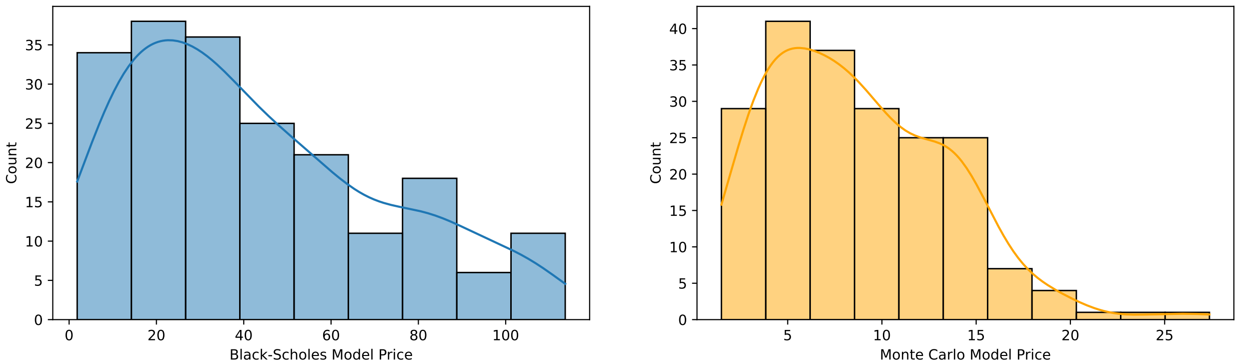

Figure 5: Option Price Distribution generated with both methods

Furthermore, from Fig 6 and Fig 7, we can see that the distinction in prices can be attributed to the models' difference in correlation between their evaluations and the parameters used. Additionally, based on these results, we can see that the Monte Carlo simulations are highly correlated with the Initial Stock Price and the Strike Price, while the Black-Scholes Model's prices are mostly correlated with the underlying stock's volatility.

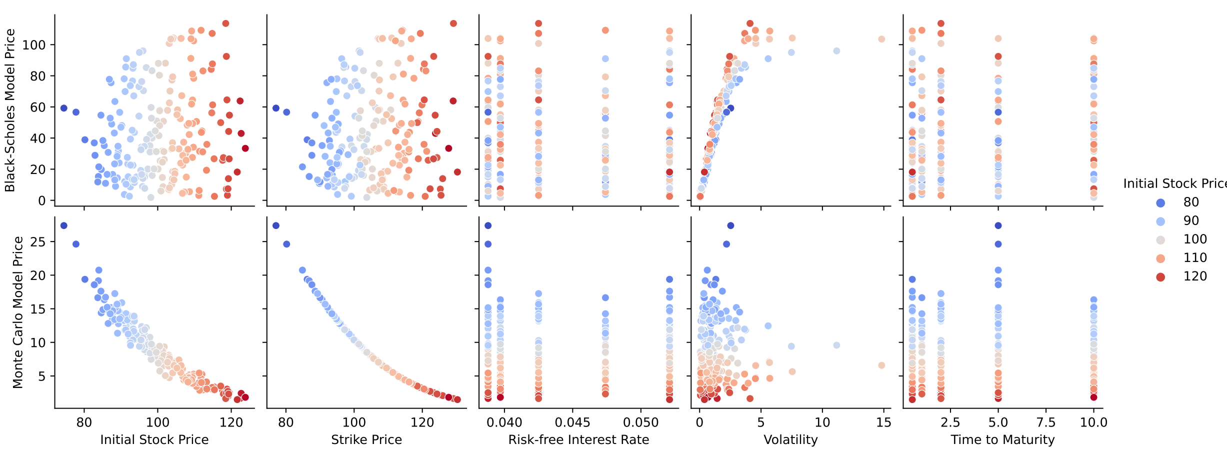

Figure 6: Pairplot for the Parameters and the Option Prices generated with both models

Finally, from the results displayed in Tab 6, we can see that the two models are not correlated, as the R^2 score is 0.02. Also, we can see that the differences between the models are very high, with the lowest being the Mean Absolute Difference, with a value of 34.74$.

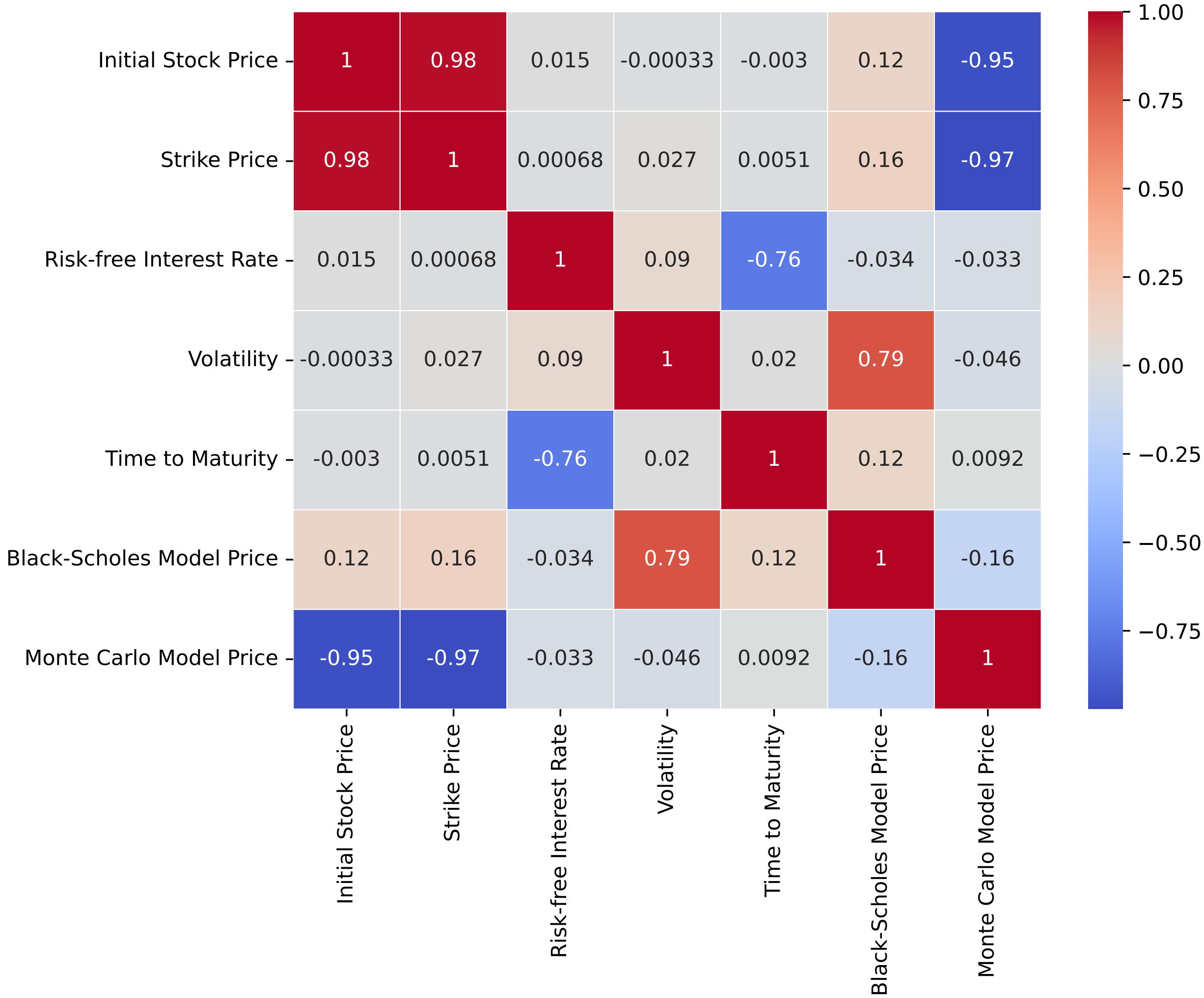

Figure 7: Parameters and Generated Option Prices Correlation Heatmap

Metric	**Result**
Coefficient of Determination (R^2)	0.02
Mean Square Difference	2060.08
Mean Absolute Difference	34.74
Root Mean Squared Difference	45.39

Table 6: Comparation Metrics' Results

8 Discussion

The observed results reveal significant disparities among the models in terms of the option prices they produce for identical stock value parameters. Upon scrutinizing the prices depicted in Fig 5, it becomes evident that the Monte Carlo Simulation tends to yield more conservative estimates compared to its Black-Scholes counterpart.

Moreover, the discrepancies between the models' outputs can be traced back to the differences in how their prices correlate with the input parameters. This implies that variations in the input parameters have a pronounced impact on the resulting option prices, with each model reacting differently to changes in these variables. Such differences underscore the importance of understanding the underlying assumptions and methodologies employed by each model in order to make informed decisions when evaluating financial instruments.

9 Conclusion

In conclusion, in this project we successfully presented and analysed the capabilities of Monte Carlo Simulation methods for use in Option Prices. Afterwards, we present the Black-Scholes model and compare its results to simulated data parameters. Finally, we presented our findings and conclude that the models differ in their estimations due to their differences in correlations with the input parameters.

For future work, we recommend analysing the accuracy of the model with real data parameters and comparing them again to see which approach is better.

References

[1] AWS, "What is The Monte Carlo Simulation?." https://aws.amazon.com/what-is/monte-carlo-simulation/. [Accessed 06-02-2024].

[2] Wikipedia, "Option (finance)." https://en.wikipedia.org/wiki/Option_(finance). [Accessed 05-02-2024].

[3] Wikipedia, "Stock." https://en.wikipedia.org/wiki/Stock. [Accessed 05-02-2024].

[4] Wikipedia, "Geometric Brownian motion." https://en.wikipedia.org/wiki/Geometric_Brownian_motion. [Accessed 06-02-2024].

[5] QuantStart, "Ito's Lemma." https://www.quantstart.com/articles/Itos-Lemma/. [Accessed 08-02-2024].

[6] Vlad Gheorghiu, "Ito calculus in a nutshell." https://quantum.phys.cmu.edu/QIP/ito_calculus.pdf, 2011. [Accessed 08-02-2024].

[7] Vegibit, "What Is A Vectorized Operation In Pandas." https://vegibit.com/what-is-a-vectorized-operation-in-pandas/. [Accessed 06-02-2024].

[8] F. Black and M. Scholes, "The pricing of options and corporate liabilities," *Journal of political economy*, vol. 81, no. 3, pp. 637–654, 1973.

[9] corporatefinanceinstitute, "Risk-Free Rate." https://corporatefinanceinstitute.com/resources/valuation/risk-free-rate/. [Accessed 09-02-2024].

[10] Bloomberg, "US Treasury Yields." https://www.bloomberg.com/markets/rates-bonds/government-bonds/us. [Accessed 11-01-2024].

[11] P. Cizeau, Y. Liu, M. Meyer, C.-K. Peng, and H. E. Stanley, "Volatility distribution in the s&p500 stock index," *Physica A: Statistical Mechanics and its Applications*, vol. 245, no. 3-4, pp. 441–445, 1997.