Numerical Methods in Quantitative Finance: A Black-Scholes and Monte Carlo Analysis

Nitin Kumar
Mayank Modi
Ashutosh Ranjan
Aryan Yadav

October 2025

Abstract

This report details the implementation of a quantitative finance framework in Python. The project begins by pricing a European call option using two distinct methods: the analytical Black-Scholes formula and a numerical Monte Carlo simulation. The framework is then extended to solve four practical problems: (1) calculating the implied volatility smile from market data, (2) computing the Value at Risk (VaR) for a mixed portfolio, (3) estimating the option "Greeks" numerically, and (4) verifying the model's no-arbitrage assumption using Put-Call Parity.

Contents

1 Core Project: Black-Scholes vs. Monte Carlo

The project's foundation is the pricing of a European call option.

1.1 Black-Scholes Analytical Model

The Black-Scholes-Merton (BSM) model provides a closed-form solution for a European call option (C) under a set of simplifying assumptions (constant volatility, constant risk-free rate, etc.).

The BSM formula is:

$$C(S_t, t) = N(d_1)S_t - N(d_2)Ke^{-r(T-t)}$$

Where:

$$d_1 = \frac{\ln(S_t/K) + (r + \frac{1}{2}\sigma^2)(T-t)}{\sigma\sqrt{T-t}}, \quad d_2 = d_1 - \sigma\sqrt{T-t}$$

And $N(x)$ is the cumulative distribution function (CDF) for a standard normal variable.

1.2 Monte Carlo Simulation Model

The Monte Carlo method prices the option by simulating a large number of possible stock price paths and averaging their discounted payoffs. It assumes the stock price follows a Geometric Brownian Motion (GBM) under the risk-neutral measure:

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

The solution for the stock price at maturity T is:

$$S_T = S_0 \exp\left(\left(r - \frac{1}{2}\sigma^2\right)T + \sigma\sqrt{T}Z\right), \quad Z \sim N(0,1)$$

The option price is the discounted average of all simulated payoffs:

$$C \approx e^{-rT}\frac{1}{N}\sum_{i=1}^{N}\max(S_T^{(i)} - K, 0)$$

1.3 Results: Convergence

As predicted by the Law of Large Numbers, the Monte Carlo price converges to the analytical Black-Scholes price as the number of simulation paths increases.

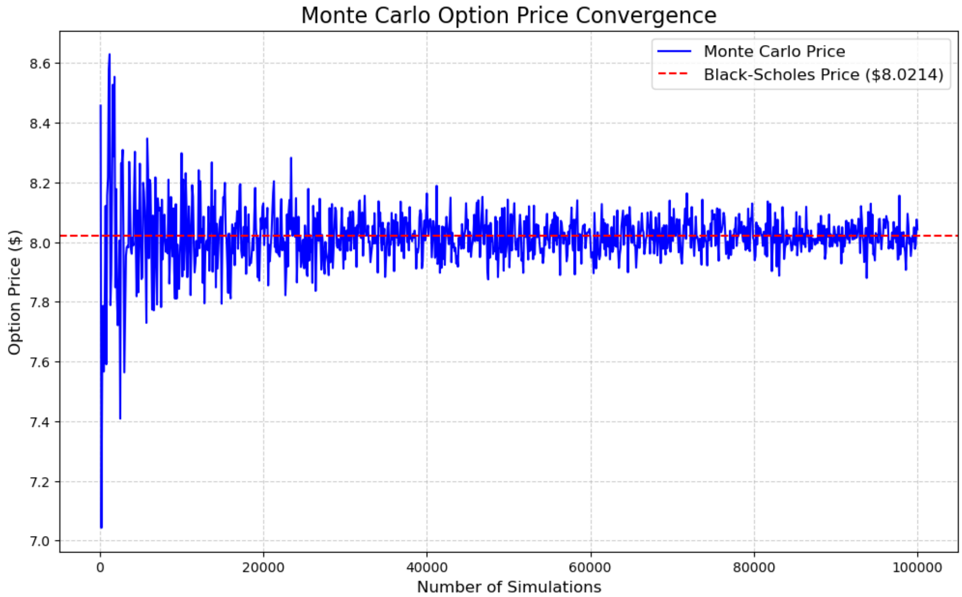

Figure 1: Convergence of the Monte Carlo price (blue) to the analytical Black-Scholes price (red dashed line) as the number of simulations increases.

2 Extension 1: The Implied Volatility Smile

The BSM model assumes volatility (σ) is constant. This extension tests that assumption by inverting the formula to solve for the σ that market prices imply.

$$\sigma_{\text{implied}} = \text{solve for } \sigma \text{ in: } C_{BS}(S_t, K, T, r, \sigma) - C_{\text{market}} = 0$$

A root-finding algorithm (e.g., 'brentq') is used to find σ_{implied} for various strike prices (K). If the BSM assumption held, the resulting plot would be a flat line.

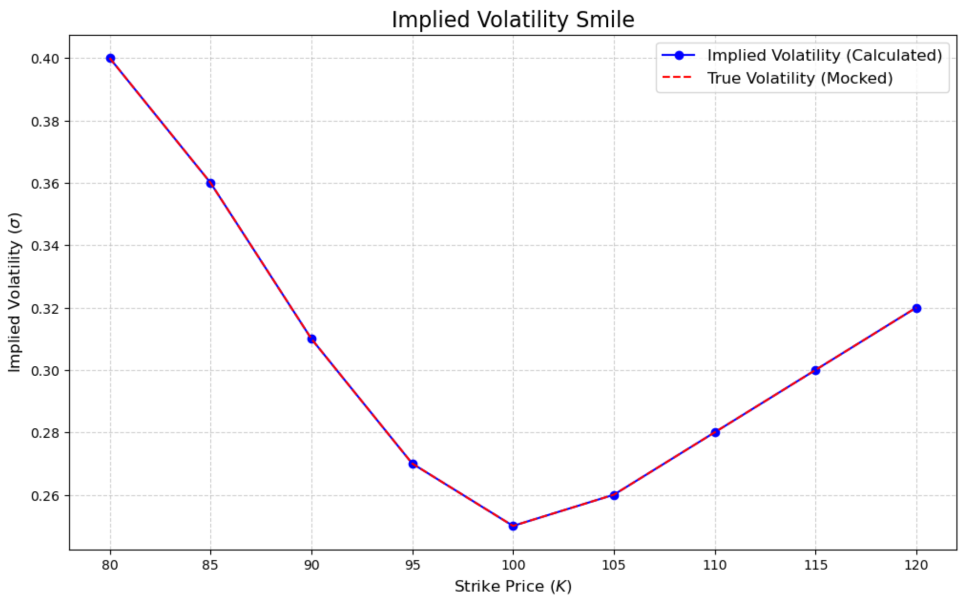

Figure 2: The Implied Volatility Smile. This plot shows that out-of-the-money and in-the-money options have a higher implied volatility than at-the-money options, contradicting the BSM model's assumption.

3 Extension 2: Portfolio Value at Risk (VaR)

This extension uses the Monte Carlo engine to calculate the 10-day 95% and 99% Value at Risk (VaR) for a portfolio of 100 long shares and 2 short call options.

1. Calculate the portfolio's initial value: $P_0 = 100S_0 - 2C(S_0, T)$.
2. Simulate $N = 50,000$ possible stock prices in 10 days: S_{10}.
3. For each S_{10}, re-price the option $C(S_{10}, T - 10 \text{ days})$.
4. Calculate the portfolio's future value: $P_{10} = 100S_{10} - 2C_{10}$.
5. Calculate the Profit & Loss (P&L) distribution: $P\&L = P_{10} - P_0$.
6. The 95% VaR is the 5th percentile of the P&L, and the 99% VaR is the 1st percentile.

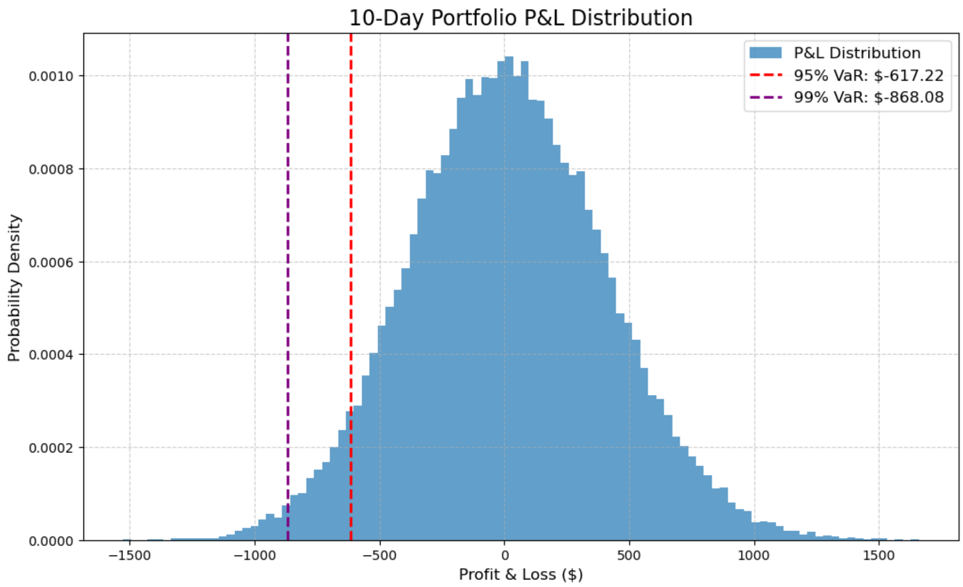

Figure 3: Histogram of 10-day P&L outcomes. The red and purple dashed lines indicate the 95% and 99% VaR, respectively, showing the portfolio's downside risk.

4 Extension 3: Calculating the Greeks

This extension compares the analytical BSM "Greeks" to numerical estimates from the Monte Carlo simulator using the finite difference ("bump-and-reprice") method.

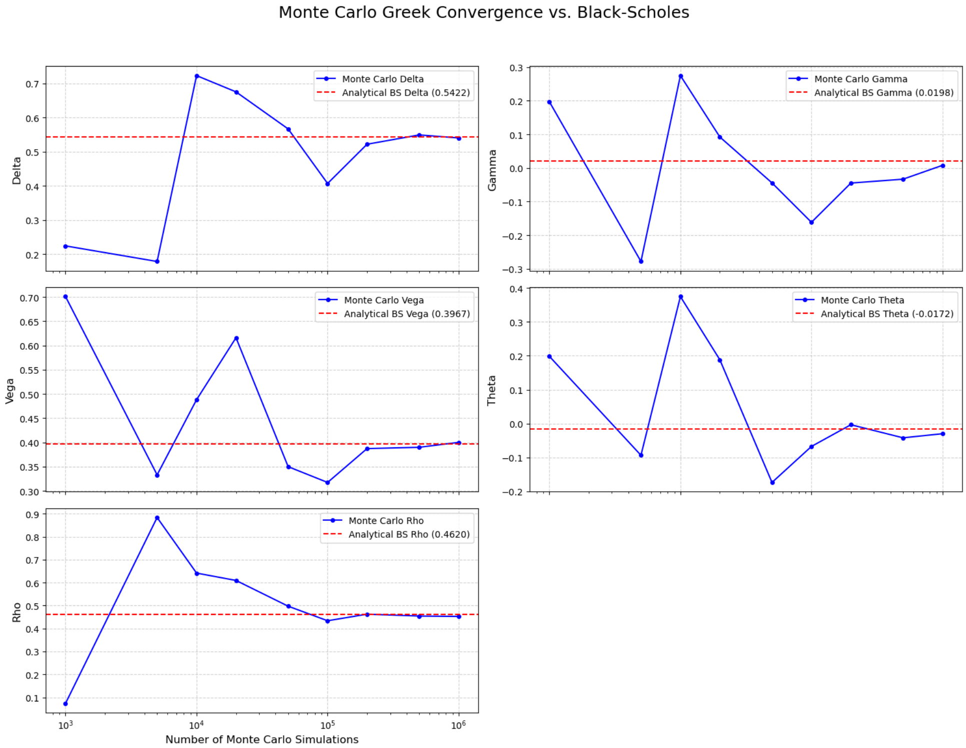

Figure 4: Convergence of numerical (Monte Carlo) Greek estimates to their analytical (BSM) values. First-order Greeks converge, while the second-order Gamma remains noisy.

5 Extension 4: Verifying No-Arbitrage

The final extension verifies the model's internal consistency by testing the fundamental no-arbitrage condition of Put-Call Parity:

$$C - P = S_0 - Ke^{-rT}$$

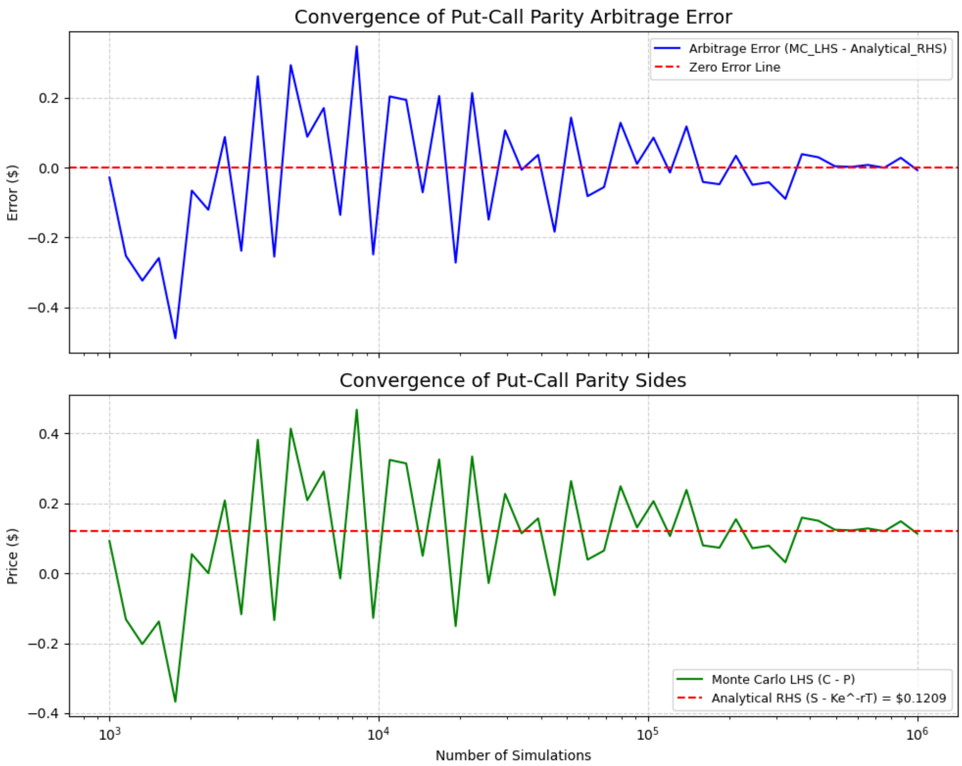

Figure 5: The top plot shows the arbitrage error approaching zero. The bottom plot shows the simulated (MC) and analytical sides of the parity equation converging.

6 Conclusion

This project successfully built and validated a Monte Carlo pricing engine against the analytical Black-Scholes model. The framework was then extended to perform practical financial analysis, including testing model assumptions against market data (Implied Volatility), quantifying portfolio risk (VaR), estimating risk sensitivities (Greeks), and verifying the model's core no-arbitrage principle (Put-Call Parity).

References

[1] Sheldon M. Ross, *An Elementary Introduction to Mathematical Finance*, 3rd Edition, Cambridge University Press, 2019.

[2] K. Burkov, *Understanding Financial Mathematics: A Practical Guide for Practitioners*, Wiley, 2018.

[3] Steve Roman, *Introduction to Financial Mathematics*, Springer, 2015.