Option Pricing using Deep Neural Networks

An Experimental Study with B-S Model and RNN Framework

2024Spring GR5293–Prof. Law

Likun Lin, Yiyang Wu

April 29, 2024

Abstract

This report explores two different RNN-based deep learning model architectures namely LSTM and GRU in performing option pricing. Emphasizing on the time-series forecasting task, this report utilizes a simulated time series dataset that simulates market volatility and option values upon the Heston model basis. The traditional B-S model is used as the benchmark model used to compare performance of deep learning models. The performance of the models in option pricing was evaluated using MSE. This study highlights the power of deep learning techniques in the field of time-series option pricing and potential to incorporate large trading data for more accurate predictive models.

1 Introduction

Compared to traditional option pricing techniques based on solving mathematical formulas, deep learning models perform better in capturing non-linear relationships and uncertainties over time. Mathematical models typically make assumptions on the real-time derivatives such as a distribution of asset prices over time with constant volatility in the B-S Model, but this may not account for more real-world situations. The reality is that options and their underlying asset prices fluctuate at no specific nor deterministic pattern influenced by news, economic shifts, politics, etc. Deep learning models have a higher potential in capturing these relationships between input and output variables, suggesting it may outperform traditional methods. However, it is to be determined what effect the nature of simulated time series data will have on model performance compared to in a real-world setting.

This report generate time-series market simulation data with Heston Model by leveraging the Quanlib library in the python ecosystem. Following the Heston Model setting, the simulated data emphasis dynamics on the price volatility. Then by performing a time-series forecasting experiments, this report evaluates the performances of traditional B-S model with modern Deep Neural Network frameworks.

2 Model Description

This section explores the models used in the report: the Black-Scholes (B-S) model, and two deep learning frameworks—Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU).

2.1 Black-Scholes (B-S) model

The Black-Scholes model is a mathematical model for pricing an options contract. Specifically, it calculates the theoretical value of European-style options using current stock prices, expected dividends, the option's strike price, expected interest rates, time to expiration, and volatility. The model is based on several assumptions, including the stock price following a geometric Brownian motion with constant drift and volatility. This report specifically focuses on the European call option pricing.

$$C(S,t) = S_t e^{-q(T-t)} N(d_1) - K e^{-r(T-t)} N(d_2), \tag{1}$$

where:

- $C(S,t)$ is the price of the call option.
- S_t is the current stock price.
- K is the strike price of the option.
- T is the time to maturity.
- r is the risk-free interest rate.
- q is the continuous dividend yield.
- $N(\cdot)$ is the cumulative distribution function of the standard normal distribution.
- $d_1 = \frac{\log\left(\frac{S_t}{K}\right)+\left(r-q+\frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}$.
- $d_2 = d_1 - \sigma\sqrt{T-t}$.
- σ is the volatility of the stock's returns.

2.2 Long Short-Term Memory (LSTM)

LSTM networks are a type of recurrent neural network (RNN) suitable for sequence prediction problems. LSTMs are designed to avoid the long-term dependency problem in traditional RNNs, making them effective for applications where the input data sequence is lengthy or when context is crucial for prediction.

LSTM units include components like input, output, and forget gates that regulate the flow of information. These gates determine what information should be retained or discarded, making LSTMs particularly useful for learning from important events in the past with unknown lags between these events and important future events. An LSTM network can be represented as follows:

$$
\begin{aligned}
f_t &= \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \\
i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\
\tilde{C}_t &= \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\
C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t \\
o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\
h_t &= o_t * \tanh(C_t)
\end{aligned}
$$

where σ represents the sigmoid function, $*$ denotes the Hadamard product (element-wise multiplication), and W and b are the weights and biases of the respective gates.

2.3 Gated Recurrent Units (GRU)

GRU is another variant of RNNs, similar to LSTMs but simpler in structure, often leading to faster computations and requiring fewer parameters. A GRU has two gates, an update gate and a reset gate. The update gate helps the model to decide how much of the past information (from previous time steps) needs to be passed along to the future, while the reset gate determines how much of the past information to forget.A GRU network can be represented as follows:

$$
\begin{aligned}
z_t &= \sigma(W_z \cdot [h_{t-1}, x_t] + b_z) \\
r_t &= \sigma(W_r \cdot [h_{t-1}, x_t] + b_r) \\
\tilde{h}_t &= \tanh(W_h \cdot [r_t * h_{t-1}, x_t] + b_h) \\
h_t &= (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t
\end{aligned}
$$

where z_t is the update gate, r_t is the reset gate, and h_t is the current memory content.

3 Data-driven Experiment

This section delves into a meticulous data-driven experiment pertinent to option pricing using deep learning. Generally, there are two distinct approaches to data simulation in this domain. The first approach involves simulating a snapshot of the input parameter space alongside the target asset prices, an effective method for approximating a non-linear pricing function. The second approach focuses on time-series forecasting tasks. It entails generating option paths complete with timestamps, thereby enabling the learning of option price dynamics through a sequential learning framework. This report aligns with the latter approach, and will proceed to detail the experimental setup in the subsequent sections.

3.1 Data

The DGP outlined in this experiment models the evolution of option prices over a defined date range from January 1, 2020, to December 31, 2021. This simulation employs the Heston model to capture the stochastic volatility characteristic of financial markets, particularly for 3-months European Call Option Pricing. The process incorporates the following components:

- Use of the New York Stock Exchange (NYSE) calendar for defining trading days and the ActualActual(ISDA) day count convention for calculating time fractions.
- Specification of Heston model parameters to simulate volatility dynamics.
- Employment of a Geometric Brownian Motion (GBM) process for path simulation of the underlying asset's spot price.
- Implementation of flat forward yield curves for risk-free rates and dividend yields.
- Generation of GBM paths using calculated drift and diffusion based on model parameters.
- Pricing of 3-month maturity European call options at-the-money for each simulation date, using the Analytic Heston Engine.

The simulation parameters employed are presented in the table below:

Table 1: Parameter Settings for Data Generating Process

Parameter	**Value**
Risk-Free Rate (r)	0.01
Dividend Yield (q)	0.02
Initial Variance (v_0)	0.01
Mean Reversion Rate (κ)	0.1
Long-Run Variance (θ)	0.01
Volatility of Volatility (σ)	0.1
Correlation (ρ)	-0.5
Drift (μ)	0.05
Volatility (Asset)	0.20
Initial Spot Price (S_0)	100

3.2 Time-Series Modeling

The modeling phase of this report involves a division of the generated data into training and testing datasets, adhering to an 80% training split. This temporal partitioning ensures that all data leading up to the 6th of August, 2021, is utilized for training purposes. Subsequently, the model's forecasting proficiency is evaluated on the test set using the Mean Squared Error (MSE) as the performance metric.

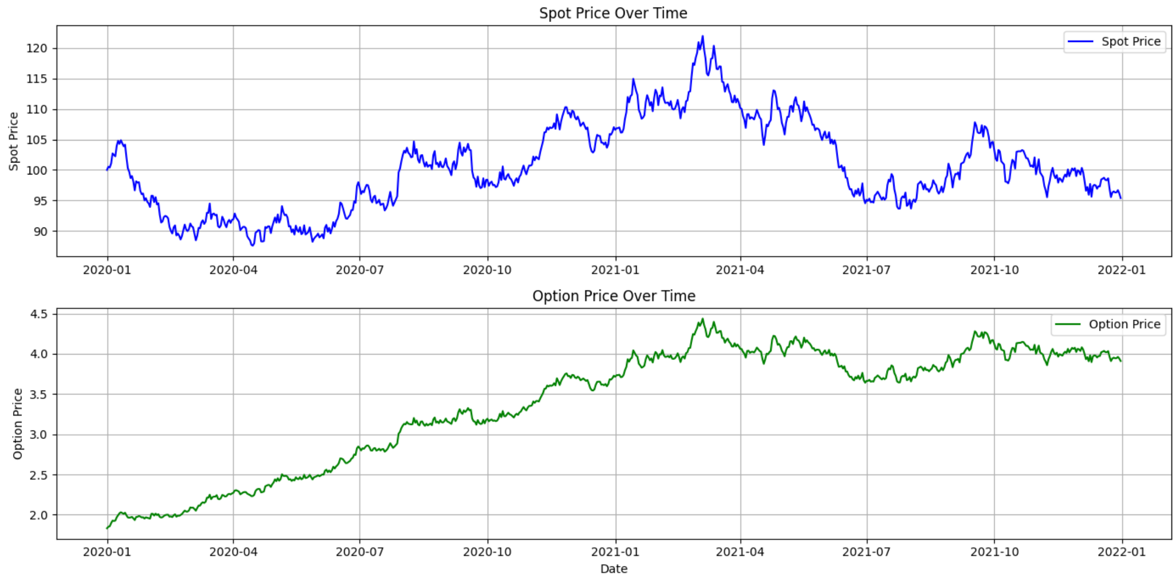

Figure 1: The top panel shows the spot price of the underlying asset over the period from January 2020 to December 2021. The bottom panel displays the corresponding option prices derived from the Heston model. It is evident that the option prices exhibit a distinct trend and volatility pattern in response to the underlying spot price movements.

3.2.1 Black-Scholes Model

For the Black-Scholes model, this report adopts the stock price series from the training dataset to deduce the asset's price volatility. Subsequently, employing the derived mathematical formula (1), the report proceeds to price the options within the test set. This approach enables a direct comparison of theoretical values computed under the Black-Scholes framework against observed market prices.

The Black-Scholes model's predictive accuracy was assessed by comparing the predicted option prices against the actual market prices. The following plot illustrates this comparison for the test set period, which extends from August 2021 to January 2022.

3.2.2 Deep Learning Frameworks Implementation

In implementing the deep learning frameworks, this report ensures consistency across control factors of the experiments, including data inputs, training configurations, and model architectures. Both deep learning models utilize an identical training dataset and are subject to the same number of training iterations, specifically set at 1000, without any division into batches. The architectures for both models are constructed uniformly, featuring dual hidden layers each with a dimensionality of 200, followed by a singular linear output layer to predict option prices. This standardized approach facilitates a fair comparison of the models' predictive capabilities.

3.3 Model Evaluation and Performance Comparison

The efficacy of each model within the scope of this report is assessed through a rigorous evaluation framework. The cornerstone of this assessment is the Mean Squared Error (MSE) metric, which quantifies the average squared difference between the predicted and actual option prices. A comparative analysis is conducted to appraise the predictive performance of the Black-Scholes (B-S) model, the LSTM network, and the GRU network.

3.3.1 MSE Comparative Analysis

The Mean Squared Error (MSE) is utilized to quantitatively assess the prediction accuracy of the Black-Scholes model, LSTM network, and GRU network. Table 2 consolidates the MSE values derived

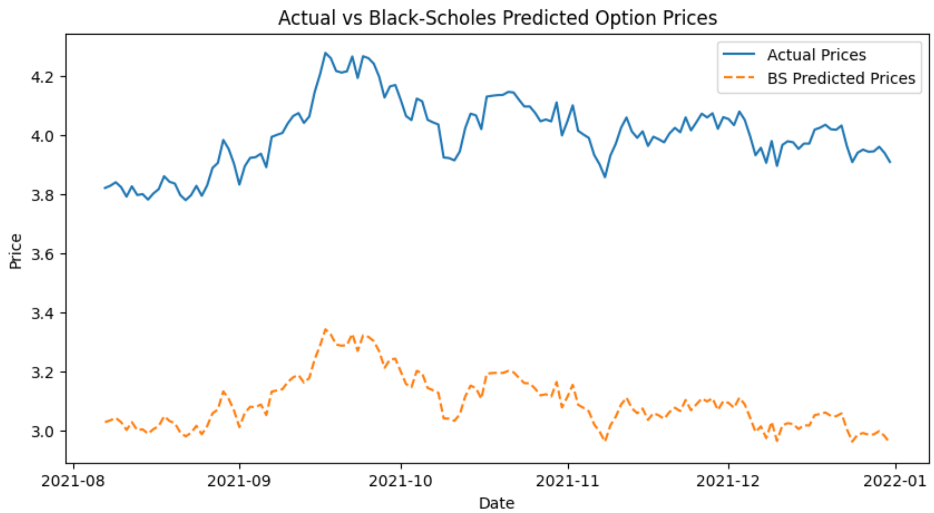

Figure 2: Comparison of actual market prices and Black-Scholes model predicted prices for options. The solid line represents the actual market prices observed, while the dashed line depicts the option prices as forecasted by the Black-Scholes model.

from the models' performance, offering an objective measure of their forecasting precision.

Table 2: MSE Comparison of Black-Scholes Model, LSTM, and GRU

Model	**MSE**
Black-Scholes Model	0.83
LSTM Network	0.004
GRU Network	0.1

The results highlight a stark contrast in predictive accuracy between the models. The LSTM network outperforms the others with an MSE of just 0.004, indicating a highly precise model that aligns closely with the actual price movements. The GRU network, while not as precise as the LSTM, also surpasses the Black-Scholes model with an MSE of 0.1. These outcomes signify not only a superior predictive capability of the deep learning models over traditional methodologies but also underscore the remarkable precision of the LSTM network in the context of option pricing.

3.3.2 GRU Model Performance

Figure 3 illustrates the GRU model's predictions in relation to the actual prices. The model demonstrates a substantial alignment with the market, particularly capturing the overall trend and periodic movements of the option prices with a high degree of fidelity.

3.3.3 LSTM Model Performance

Similarly, Figure 4 presents the LSTM model's performance. The model exhibits proficiency in tracking the actual price series, although with occasional deviations. Nonetheless, the LSTM predicted prices largely resonate with the price patterns observed in the market, indicating the model's effectiveness in grasping complex price dynamics.

The visual analyses suggest that both deep learning models, LSTM and GRU, are adept at forecasting option prices. They each show distinct capabilities in learning from historical data and translating it into accurate future price predictions, outperforming the traditional Black-Scholes model as evidenced by their respective MSE scores.

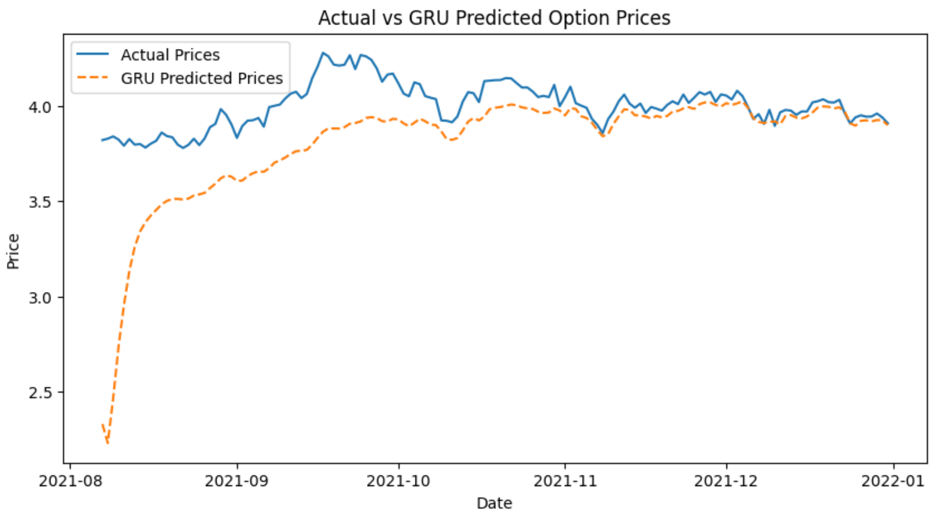

Figure 3: Comparison between the actual option prices and those predicted by the GRU model. The continuous line represents the actual market prices, while the dashed line corresponds to the GRU model's predictions.

4 Conclusion

This study has successfully explored the capabilities of LSTM and GRU deep learning models in the context of time-series option pricing, demonstrating the potential of these advanced RNN architectures to outperform the traditional Black Scholes model. The empirical results show that both LSTM and GRU models, with their ability to capture complex, non-linear dependencies in time-series data, provide a significant improvement in predicting option prices as indicated by lower mean squared errors. Specifically, the LSTM model exhibited the best performance with the lowest MSE of 0.004 , suggesting a slightly better ability to manage temporal dependencies and volatility patterns compared to the GRU model.

The use of a simulated dataset based on the Heston model has provided a controlled environment to assess the models' predictive capabilities. However, it also highlighted the limitations inherent in synthetic datasets, which may not encapsulate the multifaceted and stochastic nature of real-world financial markets.

4.1 Future Research

While simulated data is sufficient to test the accuracy and performance of different option pricing models and serves the purpose of revealing the model that achieves relatively better results, real-world trading data is needed for these models to be evaluated more practically and accurately. The fluctuations and randomness created in the generated data could merely mimic that of a real-life scenario. It is also needless to say that no predictive model generated with option prices of the past will perfectly predict prices of the future. The implementation of trading data will allow, in particular, DL models to have a better grasp of recent trends and reach a better result after combining historical data as well. This further research will determine the practicality of the implementation of these models.

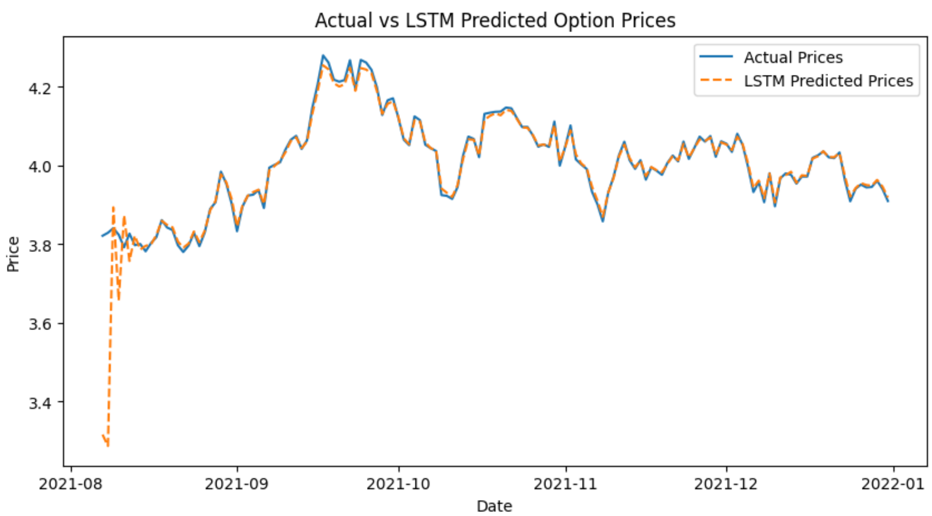

Figure 4: Comparison of actual option prices with the LSTM model's predictions. The actual prices are shown by the solid line, and the LSTM's predictions are depicted with a dashed line.