Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IMPACT FACTOR: 7.056

IJCSMC, Vol. 12, Issue. 8, August 2023, pg.9 – 15

A Multilayer Perceptron Learning Architecture for Black-Scholes Call Options Pricing

Georgios Rigopoulos
Division of Mathematics-Informatics and Statistics-Econometrics,
Department of Economics
National and Kapodistrian University of Athens, Greece
rigopgeo@econ.uoa.gr

DOI: https://doi.org/10.47760/ijcsmc.2023.v12i08.002

Abstract— This work explores whether an artificial neural network trained with synthetic call option pricing data can perform well on semi-real market data. The approach that is followed is based on a multilayer perceptron using a large synthetic dataset for training and a semi-real dataset from market data simulating top ten S&P 100 stocks. The baseline for errors and price estimations is the Black-Sholes model. Results, demonstrate the overall capability of the network to learn the Black-Scholes function using synthetic data, and estimate call option prices with high level of accuracy, competitively to Black-Sholes model. This is a valuable result for practitioners who might want to use machine learning paradigms for option pricing in various assets and markets. Future work will include feature selection and advanced sampling for training, plus alternative network architectures and further experimentation with hyperparameters. The most important outcome from the work is that machine learning models can be used from practitioners as an alternative method for option pricing.

Keywords— artificial neural network, call option pricing, asset pricing, machine learning

I. INTRODUCTION

Accurate option pricing is a key element of financial markets and traders and pricing models that can estimate arbitrage-free prices for options in real time are of paramount importance for the financial industry. With accurate pricing models, traders speculators or arbitrageurs can avoid selling underpriced options or buying overpriced options. Following this need, numerous models for option pricing have been introduced, but research in the domain is yet very active following technological developments. Option pricing models can be roughly distinguished in traditional, that rely on theory, and ones that rely on data and can be considered as relatively latest developments.

Conventional models, like the very popular Black-Scholes model [1], derive closed form solutions for pricing some option types under specific assumptions and parameter values. They are very popular due to their speed, flexibility, and accuracy. Alternative models, like Monte Carlo simulation and Binomial model [2], rely on numerical procedures, following theoretical constructs, and they simulate the underlying asset behavior to estimate option prices for a wider set of option types compared to Black-Scholes model. Traditional models are very accurate, and they offer pricing solutions for a wide spectrum of underlying assets. Their performance, however, relies on their ability to capture the dynamics and the mechanism of the underlying process and they inevitably have limitations in terms of assumptions or parameters they use.

Following the advent of machine learning algorithms and developments in artificial neural networks, many researchers introduced and benchmarked pricing methods that rely on data instead of theory [3]. Artificial neural network approaches are the most representative methods, and it can be comparable or competitive in some cases to conventional models. Machine learning models do not rely on theoretical constructs or assumptions, so they have the ability to model any kind of nonlinear behavior and interaction, something that is not feasible to express via mathematical formulation using traditional modelling. On the other hand, machine learning models require large volumes of historical data to be trained accurately, which is not always feasible in shallow and less developed markets. Despite their limitations, they are becoming a competitive alternative to traditional pricing methods, but further research is needed to offer more robust and widely used approaches.

Under this perspective, the specific work focuses on exploring the feasibility of machine learning utilization, and especially of artificial neural network model in call option pricing compared to the traditional Black-Scholes model. In this work, we train the network using a large volume of artificially generated data, and then we apply the testing in semi-real market option data simulating the top ten S&P100 companies. In this way, we avoid testing with data oriented from the same distribution as the training ones, contributing thus to the existing research adding an alternate pricing model.

The paper is organized as follows. Initially, some key background information is presented for options and their pricing, referring to the baseline Black-Scholes method. Next, artificial neural networks are introduced, along with some review of key publications for their applicability in pricing. After this, we present details of the method and the dataset and close with results and discussion on findings. In overall, even if it can be supported that artificial neural networks can play a substantial role in option pricing, it is still a method that requires further exploration and experimentation to reach the robustness required and become less ad hoc and data sensitive. However, results are promising, albeit usability in real world is yet to be proven.

II. OPTIONS BASICS

Options are popular financial instruments that are used, along with other financial derivatives, either for risk reduction [4] and hedging or as investments following market trends of the underlying assets. An option is a contract between a seller and a buyer, that offers holder the right to buy or sell some specified quantity of an underlying asset at a specified price, known as the strike or exercise price, at a date which is either the expiration of the option, or maturity date, or earlier. Options may not be exercised by the holder and expire, as they provide to holder the right to execute and not the obligation and they are distinguished in call and put options.

- A call option provides holder the right to buy a specified quantity of an underlying asset at a price, named as strike or exercise price, either on the expiration date, or at any time before. If the option is not exercised until the expiration date, it expires without any benefit for the holder. Option holder pays a price to get the option expecting a benefit in case the value of the underlying asset is higher than the strike price. In this case, the holder exercises the option at the strike price and buys the underlying asset at this price, instead of the market value that is higher. In case the price of the asset is lower than the strike price at the expiration date or earlier, the option is not exercised.
- On the other hand, a put option provides holder the right to sell a specified quantity of the underlying asset, at a specified price, called strike or exercise price, either at the expiration date or earlier. A put option has a price paid by the investor and there is expectation of profit in case the price of the underlying asset is less than the strike price of the option. If the underlying asset has a price lower than the strike price of the put option on the expiration date or before, the option is exercised and the option holder sells the underlying asset at a higher price compared to the market value, which comprises the gross profit of the investment. In case the underlying asset has a price higher than the strike price, the option is left to expire.

Options can also be classified according to the exercise date or the underlying asset types. So, the options that do not allow for exercise prior to the expiration date are called European options, where the exercise date is defined at the option contract. The options that allow for exercise at any point of time prior to the expiration date are called American options and are more attractive for trading. Further option classification can be done in terms of the underlying asset. So, considering some fundamental asset types, options can be either stock options, stock index options, future options, or product options. Many more option types exist, but in this work, we focus on the two most known, the American and European ones.

III. OPTION PRICING METHODS

Options are traded in futures exchanges or in Over-the-Counter markets (OTC), although there exist specially organized exchanges mainly in less developed markets. In any market, the option buyer pays at the initiation of an option contract the option price or premium to the option seller, named as writer. The premium is the benefit for the seller, and it is the maximum profit the seller might gain from the transaction. Thus, the accurate determination of option prices is very important aspect for the efficiency of option markets. For an option, the determinants of its value are the following:

- Current value of the underlying asset. As an option is linked to an underlying asset, every change in the asset value will be reflected in the call or put option. For stock options this is reflected by stock price.

- Value variance of the underlying asset or volatility. The variance in the asset value affects the option value, as the higher the variance of the underlying asset value, the higher the potential profit for the option, as the options have the potential to benefit from large price changes.
- Dividends of the underlying asset. The dividend paid by the underlying asset decreases the asset value, so it also changes the option value.
- Strike price of the option. The higher the strike the lower the call option value or the higher the put option value.
- Expiration date of the option or time to maturity. The longer the expiration date, the more valuable the option is.
- Risk free interest rate during the option life. It expresses the opportunity cost and an increase in the rate increases the call option values, while reduces the put option values.

Based on the above key elements, a number of pricing methods and variations have been introduced to price options accurately. Black-Scholes model [1] is considered as the predecessor that set the ground for the entire domain, and since its introduction in 1973 remains one of the most influential ones. Another widely used model is the Binomial model [2], introduced in 1978 and following a discrete time approach to estimate option prices. Many variations and novel approaches have been introduced since then, as the domain is very active and stakes in the finance industry are very high. A detailed review of pricing methods is not within our scope, as it spans across decades and can be very extended. In this work we focus on the utilization of artificial neural networks in pricing and use the Black-Scholes model as baseline, so we present in the next some representative works from both approaches along with their basic formulation.

A. *Black-Scholes*

Black-Scholes model was introduced in 1973 by Fisher Black and Myron Scholes, and is considered as the most influential model in the domain. It assumes random walk for stock prices and that they should not follow pattern that could be predicted, for a market to be efficient. If this assumption does not hold, there could be financial gain from predicting stock future prices. The model assumes also in its initial version, that there is no dividend paid until the maturity date of the option, there are no transaction fees, the risk free rate and the volatility are known and remain constant [1].

The model results in a formula for estimating the arbitrage free price of an option, as a function of current stock or underlying asset price, strike price of the option, time to maturity or expiration of the option, risk free rate and volatility of the underlying stock return. In its initial formulation the Black-Scholes model was based on the assumption that the underlying asset pays no dividends and the option will be exercised only at the maturity date. Also, that the option exercise does not impact the underlying asset value.

As those limitations do not hold in practice for all assets and options, several adjustments have been proposed. So, to support early exercise, that is allowed for American options, an approach is to use Black-Scholes unadjusted model and consider the resulting option value as the floor or the conservative estimation of the true value. Alternatively, the model can be adjusted by using it to calculate the price for every possible exercise date and then select the maximum of the estimated call values. While for the dividend, a discounted stock price can be used as the result of dividend payments. The model is referenced in almost every work related to pricing options, and the interested reader can find enough details on the model at the work of Hull [5] among others [6].

B. *Artificial Neural Networks*

The universal approximation theorem is probably the most important work related to artificial neural networks as is suggests that an artificial neural network can approximate any continuous function in a closed interval based on input variables. The importance of the theorem lies on the fact that it allows possible to use an artificial neural network, even a simple one, to model and capture any kind of complex non-linear relationships [7]. As the majority of real world problems cannot be modelled analytically, this is a key advantage of artificial neural networks.

Derivatives market is such a domain, where complexity and non-linearity is combined with real time transactions and stochastic processes, and mathematical modelling is not feasible, unless we abstract and limit the model at high extent. Researchers in the nineties proposed non parametric approaches based on data and utilizing artificial neural networks. [8], [9]. After those initial approaches, a new direction of research was initiated for the pricing of financial derivatives using machine learning based methods. Some early period indicative works, that report a high level of accuracy, include the works of Bennell and Sutcliffe [10] and Gradojevic et al. [11]. Some recent works with rich analysis and benchmarking include the works of Gu [3] and Chen [12]. As the corpus of works in option pricing via machine learning is increasing, not all researchers derive a positive result. There is still controversy on whether artificial neural networks outperform compared to traditional methods and in what settings. Some early works, which were based on simple neural network architectures and limited data, reported weak results for neural networks compared to Black-Scholes, while others reported that the results differ on various factors like if we examine options in the money or out of the money. On the other hand, numerous papers report outstanding performance of neural networks for option pricing compared to traditional models.

In all the above works, it is important to mention that the traditional methods are still used as a benchmark, and especially the Black-Scholes model to test for the errors in pricing. So, even if the artificial neural networks are promising, there is still a need to benchmark using a theoretical model. However, with the advent of computational power and data it is feasible to build more complex neural network architectures that can offer increased accuracy.

IV. DATA AND METHODS

In this work, we explore the utilization of artificial neural network for call option pricing, using the Black-Sholes model as benchmark. We follow the path of some related works, like the work of Culkin and Das [13], but we use semi-real data for the test, instead of fully artificially generated. The main research question that we explore is the performance of an artificial neural network trained with option pricing artificial data on semi-real market data. The rationale behind this question, and work in general, is that relevant works use mainly synthetic option datasets for training and testing, usually derived from some generator function version of a traditional model, like Black-Scholes. But, if both training and testing data is coming from the same distribution, it is more or less expected that the accuracy of the artificial neural network will be superior. Even if the concept is to use an artificial neural network to learn the Black-Scholes function accurately, something that is feasible with artificial datasets, testing again in artificial data will be misleading, as real-world data do not follow the same distribution as the generated. So, in this work a mixed approach is followed. Namely:

1. Train and validate an artificial neural network with synthetic call option and call option pricing data, derived by some random process based on the Black-Scholes formulation.
2. Test the artificial neural network with semi-real world call option data, generated to mimic the top 10 S&P100 stocks.

The training and testing datasets were generated using tailor made Python 3.11 code, and for the neural network model Keras library was utilized in Python with Tensorflow as the computational engine. As the computations were not very demanding, a typical desktop computer was used with Intel Core i5-at 2.90 GHz and 8GB RAM.

A. Training dataset

For the training dataset, as the target is to learn the Black-Scholes function, we generated around 1.6 million call options using the Black-Scholes function for call price, as we focus on calls only. The stock price ranges between 10 and 200 (the currency is not important here, but USD can be assumed) and the strike prices are a multiple of the stock prices to avoid extreme values. The risk-free interest rate is ranging from 1% to 2% and there is no dividend. Maturity is between 0.1 and 1, in years and the volatility ranges from 10% to 60%. In total a number of around 1.6 million call prices were generated in around 3 minutes processing time. The table below summarizes the value zones for the training dataset.

TABLE I. PARAMETER RANGE FOR THE 1.6M CALL OPTION PRICES DATASET.

Parameter	Range
Strike price (K)	0-290 (USD)
Dividend rate (q)	0%
Volatility (a)	10%–60%
Stock price (S)	10–200 (USD)
Maturity (T)	0.1 to 1 year
Risk free rate (r)	1%–3%

The generated stock price data follow a uniform distribution, while the strike prices and the call prices (Fig. 1) are right skewed. Even if the dataset is not realistic, call prices were generated by the Black-Scholes formula used from the artificial neural network to learn the formula.

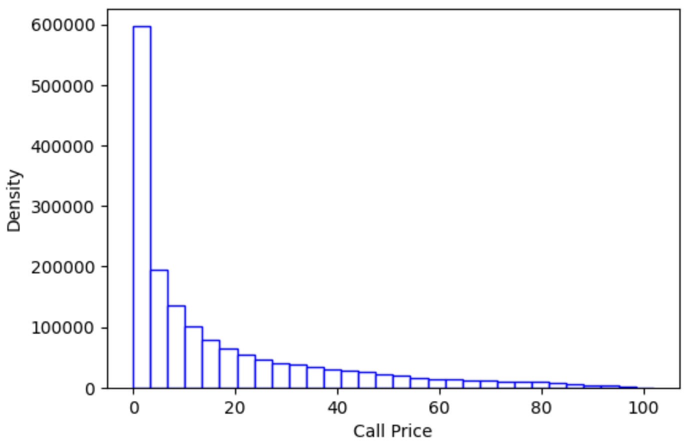

Fig. 1. Distribution of training call prices

B. Testing dataset

In order to test the trained network, we generated a semi-real dataset simulating top ten S&P100 stocks call options data using custom Python based implementation. The approach was to test accuracy of the trained network in data not derived from identical distribution with the training set, combining parameters and collect valuable results to use for further experimentation. To generate the test dataset, original stock prices were collected from market data closing prices for a 30-day period and we used the mean value over the last 30-day period for the implied volatility. We simulated call options, generated a dataset of around three thousand values and split the set into two subsets, in-the-money, and out-of-the-money to compare both. After filtering and cleaning with Python code, we generated the theoretical BS call price for them using Black-Scholes formula, to use as a benchmark. The distribution of call prices is depicted below (Fig. 2) and in general it is not dramatically different from the synthetic training data.

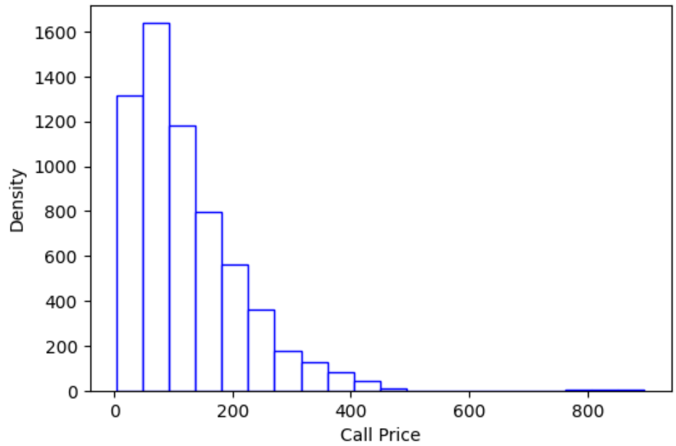

Fig. 2. Distribution of testing call prices

C. Artificial neural network

The neural network that was selected was a multilayer perceptron (MLP), a common approach that fits pretty well in such settings. For the training, we used the entire set of parameters as input features (strike price, stock price, risk free interest rate, time to maturity and volatility) and set the call price as the output. Following similar approaches [9], the input variables were normalized. The network we finally derived after experimenting, comprises one input layer, three hidden layers with 120 neurons each, and one output layer for the estimated call option price. The first and third hidden layers utilize Elu activation function and the second Relu activation function. For the training phase, the 1.6m dataset was split into 80% subset used as training sample and into a remaining 20%, that was used as a validation sample. As mentioned, the actual test was performed on the semi-real data, where the performance of the model was evaluated. The hyperparameters were tuned to a 25% dropout rate, a rule of thumb dropout rate to prevent overfitting. The number of epochs used for training was 100 and the batch size (the number of samples processed before updating the model) was set to 64. Finally, the loss function was optimized using mean square error (MSE).

V. RESULTS AND DISCUSSION

A mentioned earlier, the research question in this work was to explore the performance of an artificial neural network, that is trained with synthetic call option pricing data, on semi-real market data,. The process we followed is exploratory and not exhaustive. So, we built a typical network architecture with hyperparameters that are widely followed, trained it using synthetic data, derived by Black-Scholes function and then tested it in semi-real data. The model was trained using Tensorflow in Python and the set of parameters were exported into a file that we used in all testing scenarios using the test dataset. The results were split into two scenarios, the one for in-the-money options and the other for the out-of-the-money ones.

For the out-of-the-money options, the results are presented in the table (Table II). As we can see, the Root mean square error is 5.5%, that is relatively within acceptable limits, provided the experimental approach.

TABLE II. RESULTS FOR OUT-OF-THE-MONEY PRICING PERFORMANCE

Mean Squared Error:	0.003029
Mean Absolute Error:	0.040295
Root Mean Squared Error:	0.055039
Mean Percent Error:	0.124738

For the in-the-money options, the results are presented in the table below (Table III). As we can see, the Root mean square error is 4.8%, that is relatively within acceptable limits, provided the experimental approach.

TABLE III. RESULTS FOR IN-THE-MONEY PRICING PERFORMANCE

Mean Squared Error:	0.002329
Root Mean Squared Error:	0.048260
Mean Absolute Error:	0.035826
Mean Percent Error:	0.109374

In the figure below (Fig. 3) the differences between predicted call prices and the BS calculated for the test dataset are depicted for all call option prices and we can also see that the error is very small in general.

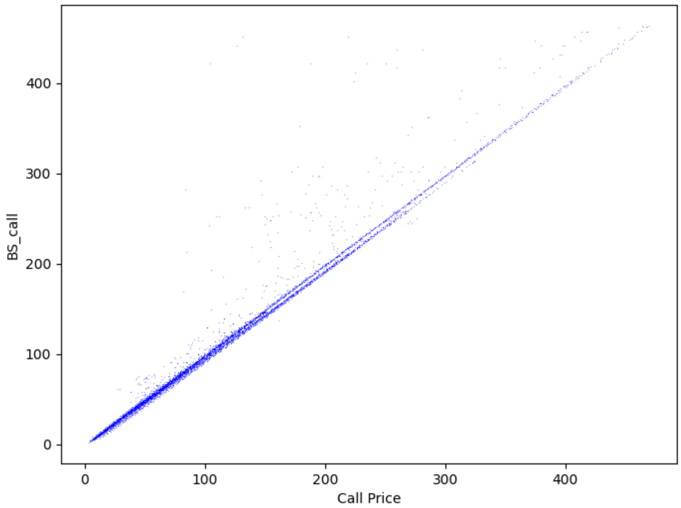

Fig. 3. Predicted vs. Black-Scholes call prices.

VI. CONCLUSIONS

This work explored the question of an artificial neural network performance on semi-real market data, provided that it has been trained with synthetic option pricing data. The approach that was followed was based on experimenting with a multilayer perceptron using a large synthetic dataset for training and a semi-real dataset from market data simulating the top ten S&P100 stocks. The baseline for the experiment in terms of price errors and price estimations was the Black-Sholes model. Results, demonstrate the overall capability of the network to learn the Black-Scholes function using synthetic data, and estimate call option prices with high level of accuracy, competitive to Black-Sholes model.

Although relevant works conclude in a similar result, the present work offers the additional element of experimentation and using semi-real market data. This is a valuable result for practitioners who might want to use machine learning paradigms for option pricing in various assets and markets. Limitations in this study have to do with the test sample, the training sample, the network architecture, and the focus. As mentioned, artificial neural networks are data driven, so training dataset is critical in terms of quantity and features. In this work we did not proceed to feature selection or advanced sampling for the training, something that could be examined further. Also, alternative network architectures can be tested or further experimentation with hyperparameters can be performed.

Probably the most important outcome from the above discussion is that machine learning models can be used from practitioners as an alternative method for option pricing, as developments in software allows to deploy such machine learning models on the web very easily and this work contributes to the development of this area.

REFERENCES

[1]. Black, F., & Scholes, M., "The Pricing of Options and Corporate Liabilities". *Journal of Political Economy,* 81(3), 637–654, 1973.

[2]. Cox, J., Ross, S., & Rubinstein, M., Option Pricing: A Simplified Approach. *Journal of Financial Economics*, 7, 44-50, 1979.

[3]. Gu, Shihao, Bryan Kelly, and Dacheng Xiu., "Empirical asset pricing via machine learning." Edited by Andrew Karolyi. Review of Financial Studies 33 (5): 2223–73, 2020.

[4]. Whaley, R. E.,. *Derivatives: markets, valuation, and risk management* (Vol. 345). John Wiley & Sons, 2007.

[5]. Hull, J., *Options, Futures and Other Derivative Securities*. Prince Hall, Englewood Cliffs, 1997.

[6]. Brealey, R. A., Myers, S. C., Allen, F., & Mohanty, P., *Principles of corporate finance*. Tata McGraw-Hill Education, 2020.

[7]. Ruf, J. & Wang, W., "Neural networks for option pricing and hedging: a literature review", *Journal of Computational Finance*, 2020.

[8]. Malliaris, M. & Salchenberger, L., "A neural network model for estimating option prices", *Applied Intelligence* 3(3), 193–206, 1993.

[9]. Hutchinson, J. M., Lo, A. W. & Poggio, T., "A nonparametric approach to pricing and hedging derivative securities via learning networks", *The journal of Finance* 49(3), 851–889, 1994.

[10]. Bennell, J. & Sutcliffe, C., "Black–scholes versus artificial neural networks in pricing ftse 100 options", *Intelligent Systems in Accounting, Finance & Management: International Journal* 12(4), 243–260, 2004.

[11]. Gradojevic, N., Gen¸cay, R. & Kukolj, D., "Option pricing with modular neural networks", *IEEE transactions on neural networks* 20(4), 626–637, 2009.

[12]. Chen, Luyang, Markus Pelger, and Jason Zhu., "Deep learning in asset pricing." Management Science, 2019.

[13]. Culkin R, Das S R., "Machine learning in finance: the case of deep learning for option pricing". *Journal of Investment Management*, 15(4): 92-100, 2017.

