International Journal of Computer Engineering and Data Science (ISSN: 2737-8543)
Volume 3– Issue 4, October – December 2023

Artificial Neural Network model for Call Options Pricing Using Market Data

Georgios Rigopoulos [1]

[1] National and Kapodistrian University of Athens, Department of Economics, Athens, Greece

E-mail: rigopgeo@econ.uoa.gr

Article history

Received Aug 24, 2023
Revised Oct 27, 2023
Accepted Dec 30, 2023
Published Nov 18, 2024

ABSTRACT

Accurate option pricing is of key importance for markets and traders. This work explores the feasibility of using artificial neural network model in call option pricing, using the traditional Black-Scholes model as a benchmark. We used a multilayer perceptron model trained to learn Black-Scholes function and tested in real option data from thirty-five S&P100 stocks. In our approach testing data is not oriented from the same distribution as training and this is a unique contribution to existing research. Findings demonstrate that artificial neural networks performs well in actual market data. Although further exploration and experimentation is required to reach required robustness and become less ad hoc and data sensitive, it is a promising approach and can play a substantial role in option pricing.

Keywords: ***option pricing, artificial neural network, multilayer perceptron***

I. INTRODUCTION

Option pricing is a dynamic domain in financial research, and numerous models for option pricing have been introduced during the past decades. Option pricing models can be distinguished in traditional or conventional, that rely on theory, and ones that rely on data and can be considered as relatively latest developments. Some conventional models, like the famous Black-Scholes model, derive closed form solutions for pricing some option types under specific assumptions and parameter values and are very popular due to their speed, flexibility, and accuracy. Alternative models, like Monte Carlo simulation and the Binomial, rely on numerical procedures, following theoretical constructs, and they simulate the underlying asset behavior to estimate the option prices for a wider set of option types compared to Black-Scholes model [1], [2], [3], [4]. Conventional approaches rely on theoretical abstraction and their performance relies on their ability to capture the dynamics and the mechanism of the underlying process, so they have limitations in assumptions or parameters they use, and they do not perform accurately or in a timely manner in every setting.

Following the development of machine learning algorithms and developments in artificial neural networks, many researchers introduced pricing models that rely on the data instead of the theory using either empirical or artificial data. Artificial neural network approaches is the most representative approach from this group of methods, and it can be comparable or competitive to the conventional models in many cases. Machine learning models do not rely on theoretical constructs, so they can model any kind of nonlinear behavior and interaction. However, their approach makes very hard to generate an explainable model, something that is inherent feature of artificial neural networks. So, their black-box behavior, even if the model is successful in pricing, remains an issue for researchers. Also, machine learning models require large volumes of data to be trained accurately, which is not always feasible. Additionally, as they rely on the training approach and dataset, they tend to be domain and data specific, rather than universal as compared to the traditional ones. So, even if machine learning based models are becoming a competitive alternative to traditional pricing methods, further research is needed to offer more robust and widely used approaches [5], [6], [7], [8], [9].

Following the above, this work explores the feasibility of using artificial neural networks in option pricing using the traditional Black-Scholes model as a benchmark. Relevant approaches demonstrate artificial neural networks trained to learn Black-Scholes function, but very few works use market data to test the models. The majority use simulated or generated datasets for both training and prediction. In this work, we train the network using artificially generated data of around 6.5 million instances, and then we apply the testing in real market option data from thirty-five S&P100 stocks. So, testing data are not oriented from the same distribution as training, and we can examine model performance in real market data, adding thus a unique contribution to existing research.

The structure of the paper is as follows. Some key background information for options and their pricing is presented initially, along with the baseline Black-Scholes method. Also, artificial neural networks are introduced, along with some review of key publications for their applicability in pricing. Next, we present the method and the datasets, followed by results and discussion on findings. Findings , support that artificial neural networks can play a substantial role in option pricing, although it still requires further exploration and experimentation to reach required robustness and become less ad hoc and data sensitive.

II. BACKGROUND

A. Options basics

In general, assets' present value is linked to their expected cash flows. However, some assets, called options, depend on underlying assets, derive their value from them, and their cash flows depend on the occurrence of specific events. So, the expected cash flows approach cannot be used to estimate their value. For this reason, alternative methods have been developed to price them in a fair way. Options are financial instruments used either for risk reduction and hedging or as investments following market trends of the underlying assets [1].

An option is a contract between a seller and a buyer, that offers the holder the right to buy or sell some specified quantity of an underlying asset at a specified price (strike price), at either the expiration of the option (maturity date), or earlier. Options may not be exercised by the holder and let expire, as they holder has the right and not the obligation to execute. Options are distinguished in call and put options with respect to the right to buy or sell the underlying asset.

Call options offer the right to buy a specified quantity of the underlying asset at the strike price, either on maturity, or any time before. If the option is not exercised until the expiration date, it expires without any benefit for the holder. The holder pays a price to purchase the option expecting a benefit if the price of the underlying asset is higher than the strike price. In this case, the holder exercises the option at strike price and buys the underlying asset at this price, instead of the higher market price. The difference is the gross investment profit. If the asset price is lower than the strike price at maturity or earlier, the option is not exercised. So, the net profit is the difference between the gross profit and the call purchase price, if the option is exercised.

Put options offer the right to sell a specified quantity of the underlying asset, at strike price, again either at maturity or earlier. A put option has a price paid by the investor who expects a 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 maturity 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. The net profit again comprises the difference between the gross profit and the put option purchase price [2].

Options can also be classified in terms of the exercise date or the underlying asset types. So, European options do not allow for exercise prior to maturity and the exercise date is defined at the option contract. While, American options allow for exercise at any point of time prior to maturity and are more attractive for trading. 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.

B. Option pricing methods

Options are traded either in futures exchanges or in Over-the-Counter markets (OTC). Some specially organized exchanges exist also 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. Following the relevant theory, the determinants of option price are the following:

- The current value of the underlying asset.
- The value variance of the underlying asset or volatility.
- The dividends of the underlying asset.
- The strike price of the option.
- The expiration date of the option or time to maturity.
- The risk free interest rate during the option life.

Based on the above, a variety of pricing methods and variations have been introduced to price options accurately. Black-Scholes model [3] is the predecessor that set the ground for the domain and since its introduction in 1973 remains the most influential. It offers an analytical method to estimate the theoretical arbitrage-free price of an option provided that some market parameters are known. Another widely used model is the Binomial [4] that was introduced in 1978 and follows a discrete time approach.

Except those two key methods, many variations and novel approaches have been introduced, as the domain is very active and the stakes in the finance industry are very high. However, despite the introduction of more sophisticated methods, the traditional ones seem to outperform in some comparative studies for American options, where analytical solutions cannot be generated [5]. In the following we briefly outline the utilization of artificial neural networks in pricing and use the Black-Scholes model as baseline with some representative works from both approaches along with their basic formulation.

C. Black-Scholes

Black-Scholes model introduced by Fisher Black and Myron Scholes in 1973 [3] is considered as one of the most influential models in finance. It assumes that stock prices moves follow a random walk and that stock prices should not follow a pattern that could be predicted, for a market to be efficient. If this does not hold, stock future prices can be predicted and there could be financial gain. Since its introduction there have been many variations, but in its initial version there is no dividend until the option maturity date, no transaction fees are charged and the risk-free rate and volatility are known constants.

The model is parametric and its famous formula for the arbitrage free price of an option can be used to price options as a function of current stock or underlying asset price, option strike price, option time to maturity or expiration, risk free rate and volatility of the underlying stock return. The formula for call option price is the following:

$$C = S * N(d_1) - K * \left(e^{-r_f/T}\right) * N(d_2) \quad (1)$$

with

$$d_1 = \frac{ln\frac{S}{K} + \left(r_f + \frac{\sigma^2}{2}\right) * T}{\sigma\sqrt{T}} \quad (2)$$

$$d_2 = \frac{ln\frac{S}{K} + \left(r_f - \frac{\sigma^2}{2}\right) * T}{\sigma\sqrt{T}} \quad (3)$$

The formula for put option price is:

$$P = K * \left(e^{-r_f/T}\right) * N(-d_2) - S * N(-d_1) \quad (4)$$

where:

- C: the price of the call option
- P: the price of the put option
- S: the current price of the stock or underlying asset
- N(d): the cumulative normal probability density function
- K: the strike price of the option
- σ: the volatility of the stock or underlying asset
- T: the time to expiration of the option
- r_f: the risk-free interest rate

As the limitations of no dividend, as well as that option exercise does not impact underlying asset values do not hold for all assets and options, several adjustments have been proposed to the initial model. So, for early exercise, allowed for American options, one approach is to use the Black-Scholes unadjusted model and consider the resulting option value as the floor or the conservative estimation of the true value. 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 [6] among others.

D. Artificial Neural Networks

Artificial neural networks were initiated back in 1943 by the work of McCulloch and Pitts, where the idea was to use mathematical formulation on the concept of a biological neuron in order to be able to execute computations mimicking brain neurons functionality. In the past decades there has been exponential research developments, driven mainly by big data and computing power evolution in the past decade. Applications were so successful, that we can find numerous domains which utilize the power of artificial neural networks and deep learning models. A groundbreaking work that set the and ground for further developments was the universal approximation theorem, which proves that an artificial neural network can approximate any continuous function in a closed interval based on input variables. The importance of the theorem is quite high as, based on it, we can use an artificial neural network, even with one hidden layer, to model any complex non-linear relationship [9]. This is a key benefit of artificial neural networks models, as the majority of real-world problems cannot be modelled and solved analytically.

A domain where complexity and non-linearity is combined with real time transactions and stochastic processes, and where mathematical modelling is not feasible, is derivatives markets. Under some abstractions and limitations, we can model analytically, but again not all problems can be solved. Option pricing is an example of a key problem for financial industry, that can partially be modelled by parametric methods and solved analytically. Black-Scholes variations and Monte Carlo simulation are the key parametric traditional methods. However, following the advent of data and artificial neural networks, researchers in the nineties proposed alternative non parametric approaches based on machine learning. Hutchinson et al. [10] were among the first researchers to study a data-driven approach and propose the utilization artificial neural networks for option pricing. Those initial approaches opened a new research direction for financial derivatives pricing using machine learning methods. Some early works reported a quite high level of accuracy [11], [12], followed by recent works with rich analysis and benchmarking [7], [8]. As works in machine learning based option pricing is increasing, not all researchers agree to positive results. Controversy on whether artificial neural networks outperform compared to traditional methods and in what settings is still under research. Usually, works were based on plain neural network architectures and limited data, so reported weak results for neural networks compared to Black-Scholes, something expected as neural networks require large training datasets. Also, some researchers claimed that results differ if we examine options in the money or out of the money or other factors. However, an increasing corpus of papers agree on the high level of neural networks accuracy for option pricing compared to traditional models.

It is important to mention though that in all works the Black-Scholes model is still used as a benchmark to test for the errors in pricing. So, despite the promising performance of artificial neural networks, theoretical models are still dominant. However, research direction is towards developing neural network architectures that can offer increased accuracy.

III. Data and Methods

The aim of this work is to explore the accuracy level of neural network architecture for call option pricing estimation using empirical data for testing. The approach we followed is to:

- use a multilayer perceptron network architecture,
- train it using artificial data generated from the Black-Scholes formula, that is considered as a benchmark method for all option pricing methods, and next,
- test the network in real call options market data for a portfolio of thirty-five stocks randomly selected from the S&P100.

The key research question that we explore is how well an artificial network performs in real market data, when trained with synthetic data. This work builds on some related works [13], [14], however it examines real testing data, instead of artificially generated. Our approach examines the case that testing data, that a neural network is going to use for prediction, do not follow the same distribution with the training data. In similar works, we see that the performance of multilayer perceptron network in option pricing is quite high, using artificial data for both training and testing, something that is expected, in general. So, we learn the Black-Scholes with artificial data and test the accuracy in real data.

The approach we followed comprises the phases below:

- Generate artificial call options data using Black-Scholes formula for a range of realistic values.
- Define a multilayer perceptron model with initial parameters.
- Train the model with the artificial dataset.
- Validate the model with a subset of the artificial dataset.
- Collect real market data for call options for thirty-five randomly selected S&P100 stocks.
- Test the model with the real market dataset.
- Evaluate the model using the real market data as the benchmark.

The entire work for the data, both generation of artificial data and collection of market data, was executed by specific modules developed in Python 3.11 [15]. The multilayer perceptron was implemented in Python, using Keras library and Tensorflow as the computational engine. For the computations, a typical desktop computer was used with Intel Core i5-at 2.90 GHz and 8GB RAM.

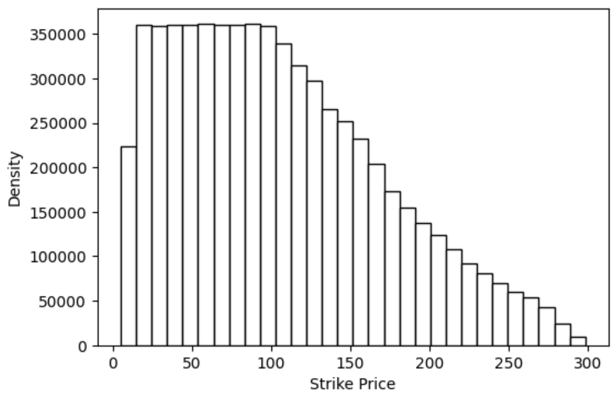
A. Training dataset

The training dataset was generated with artificial data using the Black-Scholes formula for a range of values, replicating a large number of call options. Even though in real trading call option prices deviate from the Black-Scholes formula ones, it is a baseline method to calculate option prices formally. For this work we generated around 6.5 million call option prices in total, with the process taking 19 minutes of cpu time. For the generation, we used stock price values ranging between 10 and 200 USD, with strike prices as a multiple of stock prices to avoid extreme values. So, strike prices range between 10 and 300 USD. The volatility was selected between 10% and 60% with a step of 5%, and the risk-free rate was ranging between 1% and 2%. Finally, the time to maturity was selected between 0.1 and 1 year. In Table 1 the values for the training set are summarized.

TABLE I. Table 1. Parameter range for the 6.5M 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 distributions of the generated call prices and the strike prices are depicted below for reference (Fig. 1, Fig 2). The stock prices follow a uniform distribution, while the strike prices and the call prices are right skewed. Even if the dataset is artificial, as soon as the objective is to learn the Black-Scholes formula, the call price is generated in the dataset by the actual Black-Scholes formula and this is used in the training phase from the artificial neural network to learn the formula.



International Journal of Computer Engineering and Data Science (ISSN: 2737-8543)
Volume 3– Issue 4, October – December 2023

Figure 1. Generated strike prices

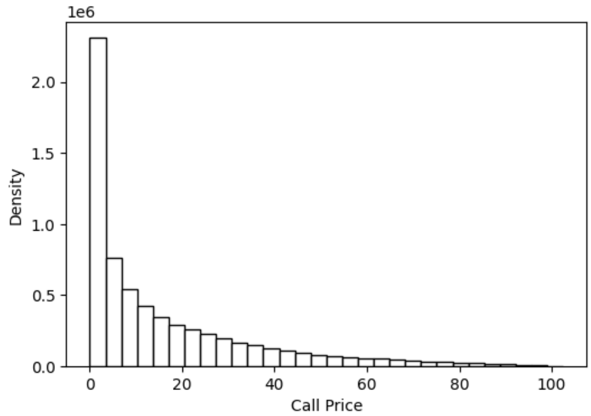


Figure 2. Generated call prices

B. *Testing dataset*

For the testing phase we utilized data originated from publicly available market data for thirty-five randomly selected S&P 100 stocks. As market data do not strictly follow the theoretical calculations, and on the other hand include some extreme values that are not met in practical trading, we performed a number of adjustments. In total a number of 3500 records were collected. The decisions taken for data preparation are the following:

- The stocks selected are a random subset of S&P 100 stocks, in order to include more diverse data, instead of picking based on some criterion, like revenues or market capitalization.
- Stock prices refer to the closing price of the previous day. We compared the closing prices with the ask and the average of bid and ask, and we did not see deviations, so we kept the previous closing price.
- Dividend was collected from market data as provided (forward dividend and yield).
- For the implied volatility we used the market provided volatility, as the mean value over the last 30-day period, derived from the average of the put and call implied volatilities for options with the relevant expiration date, based on market data.
- We focused on call options, but the same analysis can be applied on put options.
- We selected in-the-money call options.
- We filtered the data, excluding not realistic and non-representative observations from the data and to obtain more meaningful results.

Some further filtering was applied, as followed in other works [13], [14], to exclude nonrealistic and non-representative observations. Some very high option prices were excluded to avoid large deviations between theoretical and observed option prices. The distributions of the call and strike prices for the dataset are depicted below (Fig. 3, Fig. 4, Fig. 5). The stock prices vary while the strike prices and the call prices are right skewed, in a similar way to the artificial dataset. Also, we calculated the theoretical call option prices using the Black-Scholes formula in order to use them as a benchmark. In general, it seems that even if the market data can be considered that they depart from the theory, the distributions are not substantially different from the artificial data.

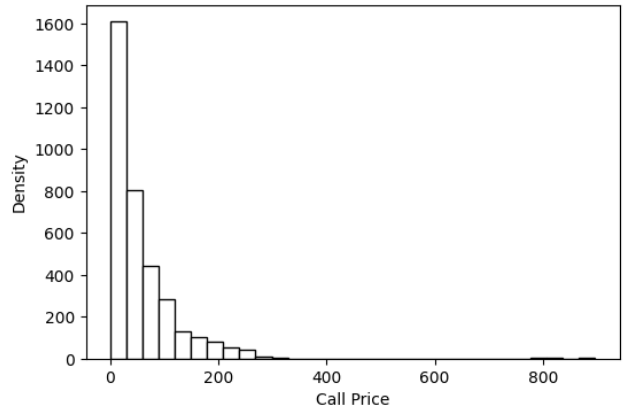


Figure 3. Call prices

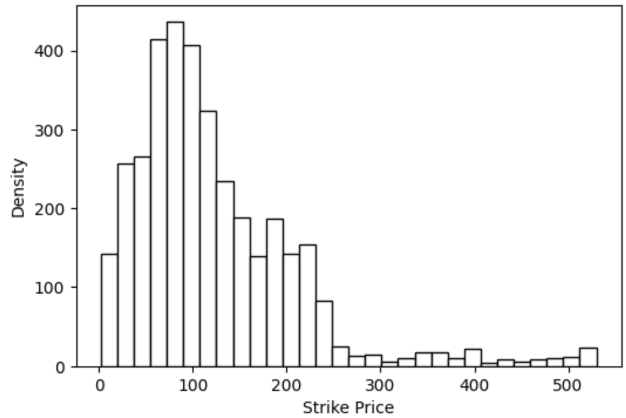


Figure 4. Strike prices

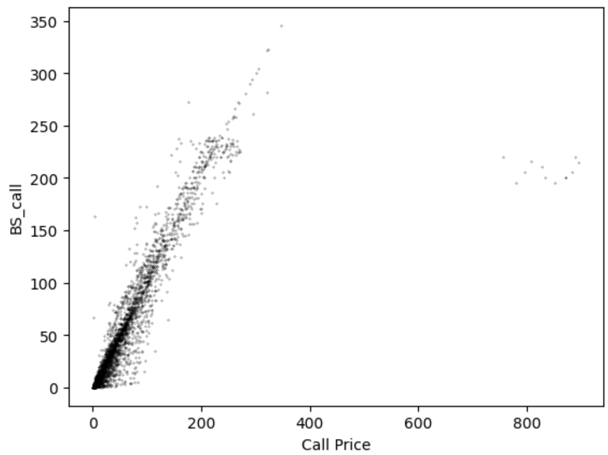


Figure 5. Actual call prices versus theoretical BS calculated

C. *Artificial neural network*

For the artificial neural network, we selected the architecture of a multilayer perceptron (MLP), as a commonly used approach for such works and it also fits well in finance settings.

The model was trained using the entire set of parameters as input features:

- strike price,
- stock price,
- risk free interest rate,
- time to maturity and
- volatility
- and the call price as the output.

Following similar approaches (Hutchinson et al. 1994), the input variables were normalized. After some experimentation, we used a network with one input layer, three hidden layers of 120 neurons each, and one output layer for the call option price output. The first and the third hidden layers utilize Elu activation function and the second the Relu activation function. The model was trained using the artificial dataset of 6.5m instances, split into 80% subset used as training sample and remaining 20% used as validation sample. The test was performed on the real market dataset and not the synthetic, where the performance of the model was evaluated. The model hyperparameters were tuned to 25% dropout rate, a rule of thumb dropout rate to prevent overfitting (Srivastava et al. 2014). 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).

IV. Results and Discussion

The key research question in this work is to explore the accuracy of an artificial neural network in estimating option prices on real market data, having been trained with artificial data originated from the Black-Scholes formula. The approach we followed was to test the model on a random set of S&P 100 stocks and their market based option data, after we had trained it and tuned the hyperparameters with synthetic data. For both training and testing we utilized Python tailor made libraries [15] along with Tensorflow module. The set of tuned parameters was exported into a file that was used in all testing scenarios using the market dataset. We focused at in-the-money call options, but the same approach can fit at out-of-the-money options.

The results from the testing phase of the model are presented in Table 1, and the normalized predicted call prices against the actual ones are depicted in Fig. 1. As we can see, the Root mean square error is 6.5%. Although the value looks high, given the fact that we followed an experimental approach, it can be considered within acceptable limits. Also, from the histogram (Fig. 2) of the differences between actual and predicted call prices, we can see that the error is very small in general.

TABLE II. Testing Error Results with market data

Mean Squared Error:	0.004304196575715178
Root Mean Squared Error:	0.0656063760294316
Mean Absolute Error:	0.04031098310438521
Mean Percent Error:	0.22251207590972308

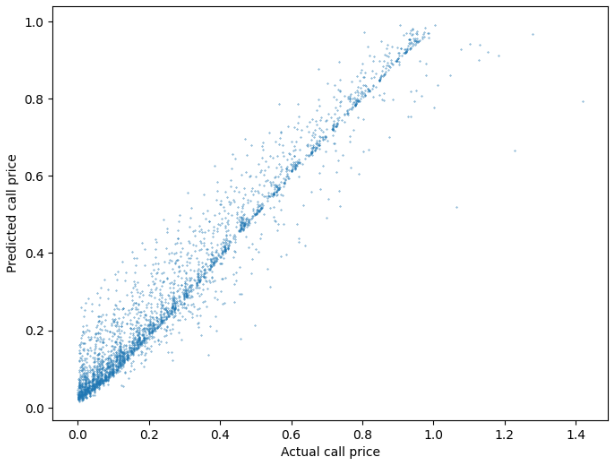


Figure 6. Predicted call prices against actual ones

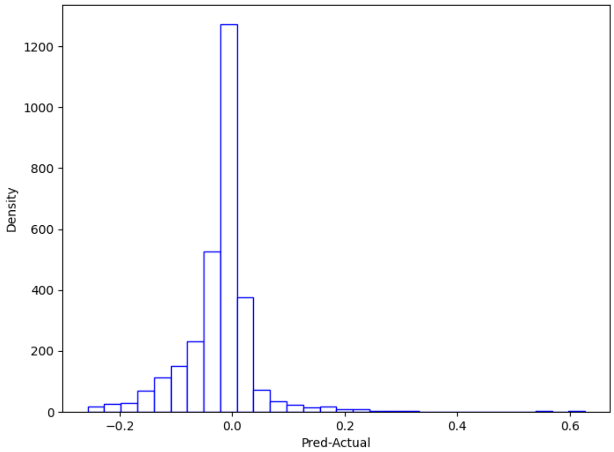


Figure 7. Difference between predicted and actual prices

To check model accuracy, we performed an additional testing with the market dataset, but we replaced call option price, as output variable, with the value calculated from the Black-Scholes formula. Even if the market price is not identical to the calculated one, as shown previously, it can be used as a benchmark. So, for the in-the-money call options, the results are presented in Table 2, and the normalized predicted call prices against the real ones are depicted in Fig. 3. As we can see, the Root mean square error is 2.8%, that is comparable to the market dataset, and from the histogram (Fig. 4) of the differences between actual and predicted call prices, we can also see that the error follows the same pattern as in the network.

TABLE III. TESTING ERROR RESULTS WITH BS PRICES

Mean Squared Error:	0.0008069991156531251
Root Mean Squared Error:	0.02840772985743713
Mean Absolute Error:	0.02018331088989721
Mean Percent Error:	0.06679347660794391

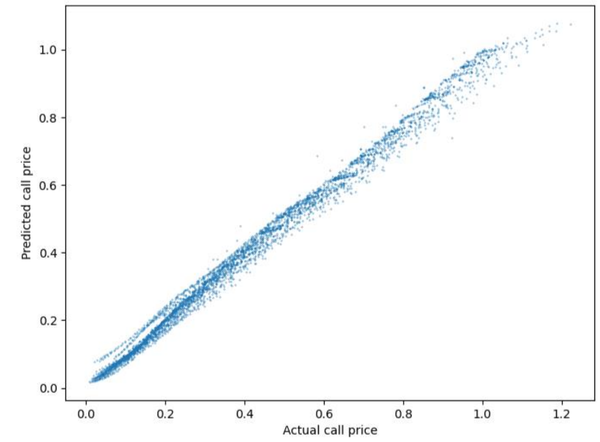


Figure 8. Predicted call prices against actual ones

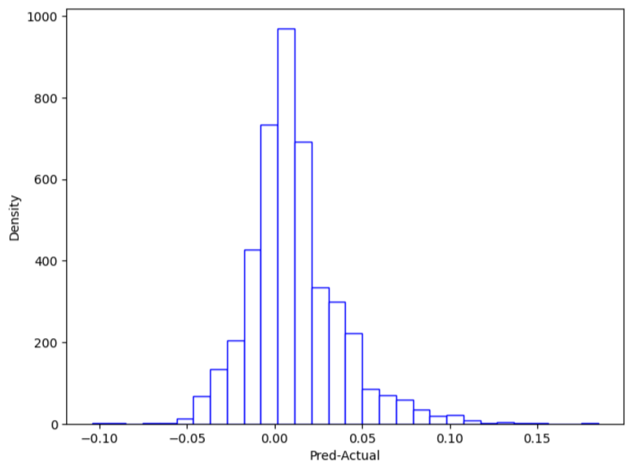


Figure 9. Difference between predicted and actual prices

So, in overall we can claim that the model, even at a preliminary setting, is comparable to the Black-Scholes formula for calculating option prices at market data, and can valuate options in acceptable accuracy. Provided that market prices are not strictly derived from Black-Scholes formula, it is reasonable to assert that the aforementioned level of error is within reasonable limits. Some additional experiments can be performed including put options and combining variations of volatility estimations for both in and out-of-the money options. Also, the trained model can be benchmarked to various alternative machine learning models in terms of accuracy and computational performance.

V. CONCLUSION

In this work we explored the accuracy of an artificial neural network on call option pricing using real market data for testing and artificial option pricing data for training. We used a multilayer perceptron model, a large synthetic dataset for training, generated from Black-Scholes formula and a real market dataset, comprising thirty fine randomly selected S&P 100 stocks. As the baseline for pricing errors and estimations in the study we used Black-Sholes model. From the results, we can see that a multilayer perceptron is capable to learn the BS function accurately using synthetic data, and estimate prices for options with high level of accuracy, competitive to Black-Scholes formula.

Other relevant works using artificial neural networks conclude in similar results, however this work adds the experimentation of using actual market data. Provided that the model is not static, but it can be retrained using additional data, including mixed artificial and actual data, its accuracy can be increased and it can become more valuable for practitioners, who might select machine learning paradigms for option pricing in various assets and markets. Some limitations in this work include the training sample, the specific network architecture and the limited focus on S&P 100 stock options. As artificial neural networks are data driven, developing appropriate training datasets is critical for their performance, so there is need for diverse training datasets. Also, in this work we did not proceed to feature engineering or advanced sampling, for the training, something that can be examined further in subsequent works. In addition, alternative network architectures can be tested or further experimentation with hyperparameters can be performed and focus can be expanded to additional assets. In future we plan to develop training processes using market data from a variety of sources.

Despite the limitations, it is evident that machine learning models can be used from practitioners as main or alternative methods for option pricing, however it is necessary to build appropriate user friendly software solutions to deploy similar machine learning models on web environment or mobile phone settings. This work, and any future contributions, aim on the development of this fats evolving area.

REFERENCES

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

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

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

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

[5] Zhao, J. (2018). American option valuation methods. International Journal of Economics and Finnance 10(5), 1–13.

[6] Hull, J. (1997) Options, Futures and Other Derivative Securities. Prince Hall, Englewood Cliffs.

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

[8] Chen, Luyang, Markus Pelger, and Jason Zhu. 2019. “Deep learning in asset pricing.” Management Science

[9] Ruf, J. & Wang, W. (2020), ‘Neural networks for option pricing and hedging: a literature review’, Journal of Computational Finance, Forthcoming . Malliaris, M. & Salchenberger, L. (1993), ‘A neural network model for estimating option prices’, Applied Intelligence 3(3), 193–206.

[10] Hutchinson, J. M., Lo, A. W. & Poggio, T. (1994), ‘A nonparametric approach to pricing and hedging derivative securities via learning networks’, The journal of Finance 49(3), 851–889.

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

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

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

[14] Rigopoulos, G., (2023), A Multilayer Perceptron Learning Architecture for Black-Scholes Call Options Pricing International Journal of Computer Science and Mobile Computing 12 (8), 9-15.

[15] Rigopoulos, G., (2023), PyOpt: A Python library for option pricing, Technical report, National and Kapodisrtian University of Athens.