Journal of Applied Intelligence 3, 193–206 (1993)

A Neural Network Model for Estimating Option Prices

MARY MALLIARIS
Management Science Department, Loyola University Chicago, 820 N. Michigan Ave., Chicago, IL 60611

LINDA SALCHENBERGER
Management Science Department, Loyola University Chicago, 820 N. Michigan Ave., Chicago, IL 60611

Abstract. A neural network model that processes financial input data is developed to estimate the market price of options at closing. The network's ability to estimate closing prices is compared to the Black–Scholes model, the most widely used model for the pricing of options. Comparisons reveal that the mean squared error for the neural network is less than that of the Black–Scholes model in about half of the cases examined. The differences and similarities in the two modeling approaches are discussed. The neural network, which uses the same financial data as the Black–Scholes model, requires no distribution assumptions and learns the relationships between the financial input data and the option price from the historical data. The option-valuation equilibrium model of Black–Scholes determines option prices under the assumptions that prices follow a continuous time path and that the instantaneous volatility is nonstochastic.

Key words: Applied artificial intelligence, neural networks, option pricing, Black-Scholes

1. Introduction

There has recently been considerable interest in the development of artificial neural networks (ANNs) for solving a variety of problems. Neural networks, which are capable of learning relationships from data, represent a class of robust, nonlinear models inspired by the neural architecture of the brain. Theoretical advances, as well as hardware and software innovations, have overcome past deficiencies in implementing machine learning and made neural network methods available to a wide variety of disciplines. Financial applications that require pattern matching, classification, and prediction, such as corporate bond rating [1], trend prediction [2], failure prediction [3], and underwriting [4], have proven to be excellent candidates for this new technology.

In this article, we present a neural network developed to estimate the market prices at closing of OEX options (options on the Standard and Poor's 100) using transactions data for the period January 1, 1990, to June 30, 1990. The neural network is a robust modeling technique that requires no assumptions about price distributions, whereas the Black–Scholes model is based on the assumption that prices follow a lognormal distribution. We compare the performance of the neural network and the Black–Scholes option-pricing model with actual prices as reported by the CBOE (Chicago Board of Options Exchange) in the *Wall Street Journal.*

2. Option-Pricing Models

In 1973, Black and Scholes [5] proposed a model for computing the current market worth of an option. The discovery of the Black–Scholes model was both empirically and theoretically significant. Its theoretical importance came from finding a solution to a longstanding problem that was initially posed by Louis Bachelier in 1900. He assumed that the price of the underlying asset followed a continuous random walk and proceeded to price the call based on such an asset. The problem with this methodology lies with the fact that with probability 1, the price of the asset becomes negative, which is not consistent with the actual behavior of stock prices, which never take negative values. It took an independent discovery of Ito's calculus in the 1940s and 1950s to develop a mathematical theory for the modeling of continuous time processes. This was used by Black and Scholes in their successful formulation and solution of the option-pricing problem.

The empirical significance of the Black–Scholes model lies in its widespread use as a pricing tool on the trading floor. Trading began in 1973, and in less than 20 years, the volume of option trading has increased dramatically. Currently, the trading volume of calls and puts for the OEX is around 760,000 contracts each.

An option is an agreement giving the holder the right to purchase (a call) or sell (a put) some asset at an agreed-upon future time, called the date of expiration. European options cannot be exercised before expiration, whereas American options may be exercised at any time prior to expiration. The price that will be paid at this future date is called the exercise price of the option. The market price of the option is the price paid now for the privilege of buying or selling the underlying asset on or before the expiration date. The Black–Scholes model uses five input variables (exercise price of the option, volatility of the underlying asset, price of the underlying asset, number of days until the option expires, and interest rate) to estimate the price that should be charged for an option. The Black–Scholes option-pricing formula for calculating the equilibrium price of call options is

$$C = S \cdot N(d_1) - Xe^{-rT} \cdot N(d_2) \tag{1}$$

where C is the market price to be charged for the option, N is the cumulative normal distribution, T is the number of days remaining until expiration of the option expressed as a fraction of a year, S is the price of the underlying asset, r is the risk-free interest rate prevailing at period t, X is the exercise price of the option, and d_1 and d_2 are given by

$$d_1 = \frac{\ln(\frac{S}{X}) + (r + \frac{\sigma^2}{2}) \cdot T}{\sigma\sqrt{T}} \tag{2}$$

and

$$d_2 = d_1 - \sigma\sqrt{T} \tag{3}$$

where σ^2 is the variance rate of return for the underlying asset. For any time interval $[0,t]$ of length t, the return on the underlying asset is normally distributed with variance $\sigma^2 t$ [6].

There are seven assumptions underlying this model that assume ideal conditions in the market [5,7]. They are

(a) the interest rate is known and constant through time;
(b) the stock price follows a random walk in continuous time with variance proportional to the square of the stock price; thus the distribution of prices is log-normal and the volatility is constant;
(c) the stock pays no dividends;
(d) the option is European, i.e., it can only be exercised at maturity (expiration date);
(e) there are no transactions costs of buying or selling;
(f) the market operates continuously; and
(g) there are no penalties to short selling.

For a rigorous presentation of the derivation of the Black–Scholes model, see [8].

For the remainder of this article and for the data we have chosen, the exercise price of the option is referred to as exercise price and is denoted by EXER; the annualized square root of the variance of the underlying asset is the volatility, VOL; the price of the underlying asset is the price of the S&P 100 index at closing, or simply, the closing price, CLOSE PRICE; the interest rate used is the prevailing rate on treasury bills, denoted INT; and time to expiration of the option is the number of days to expiration, DAYS. The exercise price, number of days to expiration, and closing price are observable. The volatility cannot be directly observed, so it is computed implicitly. Most observers use the

Implied Standard Deviation of observed option prices as an estimate of volatility [9]. We used at-the-money call options for this estimate and then used that estimate of volatility to calculate the call options for that day.

Options with an exercise price equal to the closing price of the index are said to be at-the-money. In the pricing of calls, exercise prices less than the closing price are in-the-money, and exercise prices greater than the closing price are out-of-the-money. If the ratio of the closing price to the exercise price is less than (greater than) .85 (1.15), then the option is said to be deep-out-of- (deep-in-) the-money.

Since its introduction in 1973, the Black–Scholes options-pricing model has performed better overall than any model [9,10]. The major alternative models have been Cox and Ross' pure jump model, Merton's mixed diffusion-jump model (both these models relax the continuous time assumption), Cox and Ross' constant elasticity of variance diffusion model, Geske's compound option diffusion model, and Rubinstein's displaced diffusion model (these last three relax the assumption of constant volatility).

Option trading has also been considered an appropriate domain for expert-system applications. A constraint logic programming model [11] has been developed as an expert system that uses the Black-Scholes model to evaluate strategies and to compute option values. Constraint satisfaction has been used for other approaches to option-price modeling [12]; however, no strong measures of the effectiveness of these models have been reported.

Empirical tests show that Black–Scholes remains superior among option-pricing equilibrium models, with the possible exception of cases in which trades are made deep-in- and deep-out-of-the-money. The volume of research related to the Black–Scholes model, which continues to proliferate even 20 years after the model's introduction, indicates that there is considerable interest and value in developing a model that is more robust than Black–Scholes. In addition, there is some reason to believe that the trading process itself may reveal underlying strategies as well as analytical models, and there is information to be gained from historical pricing data. Neural networks have been shown to be useful in modeling nonstationary processes and nonlinear dependencies and thus may represent a channel of investigation in the search for another type of option-pricing model.

3. Methodology

3.1. The Data Set

The data set used for this research was developed using option-price transactions data published in the *Wall Street Journal* during the period from January 1, 1990, to June 30, 1990. The data set selected for testing includes pricing data from April 23 to June 29, 1990, and includes in-the-money options and out-of-the-money options with time to expiration between 30 and 60 days. Typically, 12 different call prices per day are quoted.

The five variables selected to estimate the market price of the option are those used in the Black–Scholes model; exercise price, time to expiration, closing price, volatility, and interest rate. A sample data set used to calculate the Black–Scholes model prices is shown in table 1. For the neural network, we added two lagged variables: yesterday's closing price, LAG CLOSE PRICE, and yesterday's market price of the option, LAG MARKET PRICE.

Preliminary data analysis revealed dependencies and relationships between the variables; these were used to partition the data sets for the neural network. Figure 1 shows a graph of exercise prices versus market prices. From deep-in-the-money to at-the-money, there is a sharp and steady decrease of prices. From at-the-money through out-of-the-money, the prices have a gentle asymptotic approach to the x-axis. Experimentation with different training sets showed that better results could be obtained in the neural networks when the data were separated into in-the-money and out-of-the-money groups. Prices in-the-money vary from \$60.00 to \$0.75; prices out-of-the-money vary from \$15.50 to \$0.0625. A larger proportion of observations exist for out-of-the-money prices than for in-the-money prices. Correlations were also found between time to expiration and market price of the option, and be-

Table 1. Sample data set

Date			EXER	Days	Close price	INT Rate	VOL	Market price
M	D	Y						
4	23	90	280	26	315.58	7.71	0.161633	39.5
4	23	90	290	26	315.58	7.71	0.161633	28
4	23	90	295	26	315.58	7.71	0.161633	22.5
4	23	90	300	26	315.58	7.71	0.161633	17.875
4	23	90	305	26	315.58	7.71	0.161633	14.5
4	23	90	310	26	315.58	7.71	0.161633	10
4	23	90	315	26	315.58	7.71	0.161633	6.625
4	24	90	280	25	313.96	7.77	0.166284	38
4	24	90	290	25	313.96	7.77	0.166284	27.25
4	24	90	295	25	313.96	7.77	0.166284	22.5
4	24	90	300	25	313.96	7.77	0.166284	17
4	24	90	305	25	313.96	7.77	0.166284	12.75
4	24	90	310	25	313.96	7.77	0.166284	8.875
4	25	90	280	24	315.06	7.77	0.159941	36
4	25	90	300	24	315.06	7.77	0.159941	17
4	25	90	305	24	315.06	7.77	0.159941	13.5
4	25	90	310	24	315.06	7.77	0.159941	9.125
4	25	90	315	24	315.06	7.77	0.159941	6
4	26	90	280	23	315.82	7.77	0.158642	37
4	26	90	290	23	315.82	7.77	0.158642	25.375
4	26	90	295	23	315.82	7.77	0.158642	21.5
4	26	90	300	23	315.82	7.77	0.158642	17.625
4	26	90	305	23	315.82	7.77	0.158642	13.5
4	26	90	310	23	315.82	7.77	0.158642	9.5
4	26	90	315	23	315.82	7.77	0.158642	6.25
4	27	90	280	22	312.48	7.77	0.136054	32.375
4	27	90	290	22	312.48	7.77	0.136054	23
4	27	90	295	22	312.48	7.77	0.136054	18.5
4	27	90	300	22	312.48	7.77	0.136054	14.25
4	27	90	305	22	312.48	7.77	0.136054	10
4	27	90	310	22	312.48	7.77	0.136054	6.375

tween the closing price and the market price of the option.

3.2. *The Estimation Process*

Under supervised learning, the feedforward, back-propagation neural network learns relationships between input and output variables during a training process, as data are presented to the network. One approach to testing the performance of the network is to check its accuracy in estimating values for a holdout sample generated from the training set. For evaluating the performance of the option-pricing neural network, we selected a more realistic and more difficult performance measure. The network was trained using historical data, and option-price estimations for a future period were developed with the trained network and compared to actual prices.

To capture the volatile nature of the options market, a relatively short time frame was used for the training sets and testing sets. The testing sets were developed using a two-week time frame; this was a convenient choice because interest rate and volatility changed weekly and were relatively stable over a two-week period. Five two-week periods were selected for price estimation: the weeks beginning April 23, May 7, May 21, June 4, and June 18. To provide the neural network models with a variety of examples, each training set included as many observations as necessary to provide at least one full cycle (30 days prior to the estimation period) of pricing data.

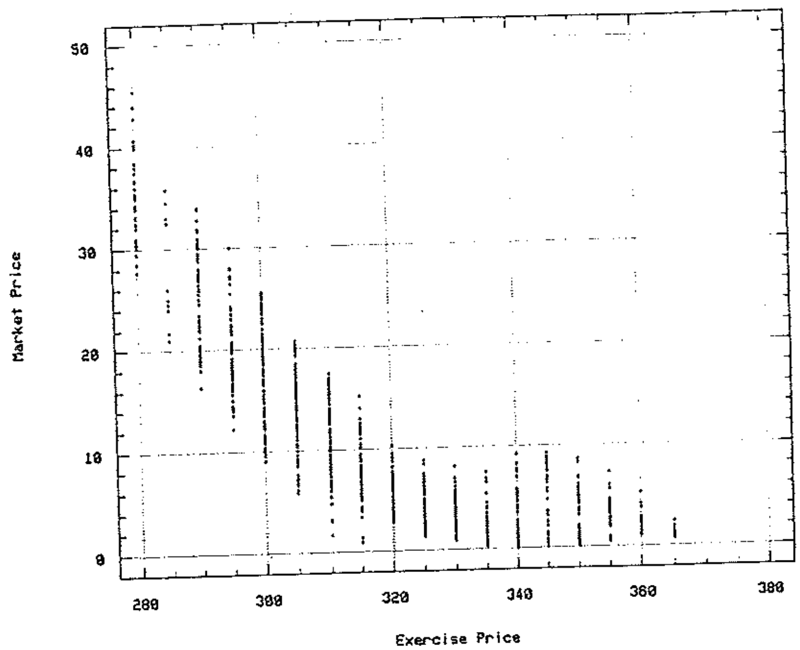

Fig. 1. Exercise price versus market price.

4. The Neural Network Model for Option Pricing

4.1. Neural Networks and Backpropagation

Inspired by studies of the brain and the nervous system, neural networks are composed of neurons or processing elements and connections, organized in layers. These layers can be structured hierarchically, and the first layer is called the input layer, the last layer is called the output layer, and the interior layers are called the middle or hidden layers. Feedforward networks map inputs into outputs with signals flowing in one direction only, from the input layer to the output layer. Each connection between neurons has a numerical weight associated with it that models the influence of an input cell on an output cell. Positive weights indicate reinforcement; negative weights are associated with inhibition. With supervised learning, connection weights are learned by the network through a training process, as examples from a training set are presented repeatedly to the network.

Each processing element has an activation level, specified by continuous or discrete values. If the neuron is in the input layer, its activation level is determined in response to input signals it receives from the environment. For cells in the middle or output layers, the activation level is computed as a function of the activation levels on the cells connected to it and the associated connection weights. This function is called the transfer function or activation function and may be a linear discriminant function, i.e., a positive signal is output if the value of this function exceeds a threshold level, and zero otherwise. It may also be a continuous, nondecreasing function. The most commonly used for backpropagation is the sigmoidal or logistic function

$$f(x) = \frac{1}{1 + e^{-\gamma x}} \qquad (4)$$

where γ is a constant which controls the slope.

While basically an information processing technology, neural networks differ from traditional modeling techniques in a fundamental way. Parametric models require that the developer specify the nature of the functional relationship between the dependent variable and the independent variables, e.g., linear or logistic. Neural networks with at least one middle layer use the data to develop an internal representation of the relationship between the variables so that a priori assumptions about underlying parameter distributions are not required [13]. As a consequence, we might expect better results with neural networks when the relationship between the variables does not fit the assumed model. Nevertheless, many decisions regarding model parameters and network topology can affect the performance of the network.

In a feedforward neural network, the connection weights can be determined during a two-step training process that presents examples $\{(x_p,y_p) : p = 1,\ldots,P\}$ where x_p is the input vector and y_p is the output vector. In the first step, for each layer of nodes, the network computes the output vector o_p as a function of the input vector and the associated connection weights. The values for the output layer nodes are compared to the actual output vector, and a performance criteria, such as the sum of the squared error, is used to determine the error for the output layer. In the second step, the error is backpropagated through the network and the weights w_{ij} are modified, according to their contribution to the network error F. For further details, see [14].

4.2. *The Development of the Neural Network for Option Pricing*

Since feedforward, single-hidden-layer neural networks have been successfully used for classification and prediction [15–17], we selected this network model for our initial experiments and used the backpropagation training algorithm. A neural network consisting of 7 input nodes, 4 middle-layer nodes, and 1 output node was developed (see figure 2). The input nodes represent the five financial variables used in the Black–Scholes model (EXER, DAYS, CLOSE PRICE, VOL, and INT) and two lag variables (LAG CLOSE PRICE and LAG MARKET PRICE), and the output node (MARKET PRICE) represents the market price of the option.

Determining the proper number of middle-layer nodes requires validation techniques to avoid underfitting (too few nodes) and overfitting (too many nodes). Generally speaking, too many nodes in the middle layer, and, hence, too many connections, produces a neural network that memorizes the data and lacks the ability to generalize. One approach that can be used to

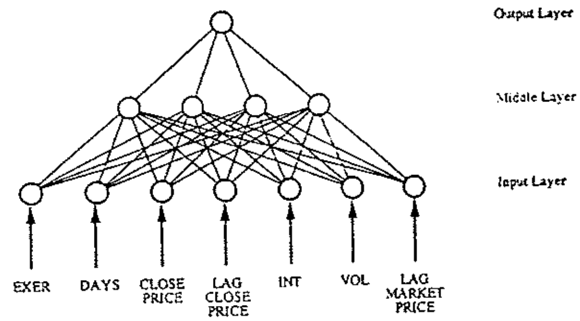

Fig. 2. Neural network for option pricing.

avoid overfitting is v-fold cross-validation [18], a variation on leave-one-out cross-validation [19]. We employed a special case of cross-validation in which the data are divided into two subsets, a training set and a validation set [20]. The training set is used to estimate the parameter, and the second set, which is a holdout sample, is used to estimate performance on new data. The cross-validation mean squared error is computed for networks of increasing size, where the number of middle-layer nodes varies from 3 to 5. The results show that the optimal number of middle layer nodes is 4 (see table 2).

The network is fully connected, with a direct connection from exercise price to the output node. Better results were achieved with this additional connection because of the linear dependence between EXER and MARKET PRICE observed in the data set and verified with a series of regression models. All the connection weights were initially randomized, and were then determined during the training process.

The generalized Delta rule was used with the backpropagation of error to transfer values from internal nodes. (For a more detailed explanation of backpropagation learning and the generalized Delta rule, see [14].) The sigmoidal function is

Table 2. A comparison of neural network models for the week beginning April 23

	Network mean squared error	
Model	Training set	Testing set
3 middle nodes	0.154464	0.204471
4 middle nodes	0.074409	0.135781
5 middle nodes	0.116703	0.178432

the activation function specified in this neural network and is used to adjust weights associated with each input node.

Supervised learning was conducted with training sets consisting of the seven predictor variables and the corresponding market price of the option for each exercise price, for each trading day. For the input nodes in which the data were not in ratio form, the values were scaled to be within a range of 0 to 1. This minimizes the effect of magnitude among the inputs and increases the effectiveness of the learning algorithm. The selection of the examples for the training set focused on the degree to which the data set represented the population. The size of the training set is important, since a larger training set may take longer to process computationally but may accelerate the rate of learning and reduce the number of iterations required for convergence.

The learning rate and momentum were set initially at 0.9 and 0.6, respectively, and the learning rate was adjusted downward and the momentum was adjusted upward to improve performance. The training examples were presented to the network in random order to maximize performance and to minimize the introduction of bias. Training was performed until convergence was achieved using the network MSE, which is graphically displayed during training. For most cases, the MSE was essentially stable after 40,000 iterations. The network was implemented using the software package Neuralworks Professional II Plus® running on a 386-based microcomputer with a math co-processor.

4.3. Experimental Design

To compare the estimations made by each model, we report the mean absolute deviation (MAD), mean absolute percent error (MAPE), and mean squared error (MSE) for each of the five two-week periods for both in-the-money and out-of-the-money prices. While MAD and MSE are meaningful measures of error for this application, we also report MAPE. Since prices vary from $60.00 to $0.0625, it is important to compare the amount of the error with the corresponding base price, i.e., to measure the relative pricing error. Option prices were estimated from the Black–Scholes model using a computer program based on equations (1) to (3). Neural network estimations were developed by inputting the estimation sets into a trained network.

5. Results

The initial results showed that, compared to the actual prices, the neural network estimations had a lower MAPE and MSE than Black–Scholes for 4 out of 5 two-week periods for the out-of-the-money case, but Black–Scholes was superior for 3 of 5 two-week periods for in-the-money trades. These results are reported in tables 3 and 4. A bias commonly reported in the literature is that Black–Scholes tends to underprice in-the-money calls [9]. To examine pricing bias, we plotted the percent pricing error versus the percent the option is in-the-money or out-of-the-money. Pricing error is calculated as the difference in the model price and actual market price, divided by the model price. Pricing error is negative when the model underestimates the actual market price and is positive when the model overestimates the market price. The percent in-the-money or out-of-the-money is found by calculating the difference in the exercise price and the closing price and then dividing by the exercise price.

Pricing bias was investigated for both the Black–Scholes and the neural network models. Figures 3 and 4 show percent error (values

Table 3. Comparative analysis, actual prices with estimated prices, out-of-the-money

Week beginning	MSE	MAD	MAPE
April 23			
Black–Scholes	0.435342	0.598932	30.81731
Neural network	0.074409	0.207702	12.74440
May 7			
Black–Scholes	0.160047	0.340729	16.23661
Neural network	0.151516	0.321083	13.44321
May 21			
Black–Scholes	0.204219	0.378636	9.43207
Neural network	0.253676	0.422369	12.30240
June 4			
Black–Scholes	0.245477	0.286645	9.104615
Neural network	0.231779	0.312945	9.097162
June 18			
Black–Scholes	1.250466	0.660788	17.45452
Neural network	0.455352	0.447812	10.94668

Table 4. Comparative analysis, actual prices with estimated prices, in-the-money

Week beginning	MSE	MAD	MAPE
April 23			
Black–Scholes	1.055732	0.676936	3.8057
Neural network	1.175115	0.82434	5.1689
May 7			
Black–Scholes	1.459734	0.670291	2.7142
Neural network	3.127410	1.289340	7.4727
May 21			
Black–Scholes	1.386018	0.766019	2.8867
Neural network	1.006885	0.832762	4.6876
June 4			
Black–Scholes	1.771367	0.784969	2.8864
Neural network	2.397112	1.053282	5.2136
June 18			
Black–Scholes	3.945318	1.391258	7.2002
Neural network	1.407175	0.987918	6.6399

greater than 0 on the y-axis indicate overpricing and less than 0 indicate underpricing) relative to percent in- or out-of-the-money (negative values indicate out-of-the-money, and positive values are in-the-money). In the Black–Scholes model (see figure 3), underpricing is more prevalent than overpricing for in-the-money and overpricing is predominant for out-of-the-money. The neural network model (see figure 4) underprices options more than it overprices them, for both in- and out-of-the-money. For both models, the most serious errors occur out-of-the-money; however, the overpricing errors are more significant for the Black–Scholes model as prices move deep out-of-the-money.

Since the Black–Scholes model overprices options out-of-the-money and the neural network model tends to underprice these options, we examined the results obtained when the model prices are averaged. The MSE, MAPE, and MAD are reported, for each of the five two-week periods, in table 5. The MSE, MAPE, and MAD were each significantly lower for three (different) two-week periods. This indicates that there may be some benefit in combining the estimates provided by the two models for out-of-the-money prices. Since both models underprice in-the-money options, averaging for in-the-money data would compound rather than improve the error. Thus, only out-of-the-money options benefit from this combination.

Paired sample comparisons tests were run on the Black–Scholes estimates and actual market prices and on the neural network estimates and actual market prices. In table 6, we report the results for out-of-the-money prices. The means, variances, and standard deviations for each sam-

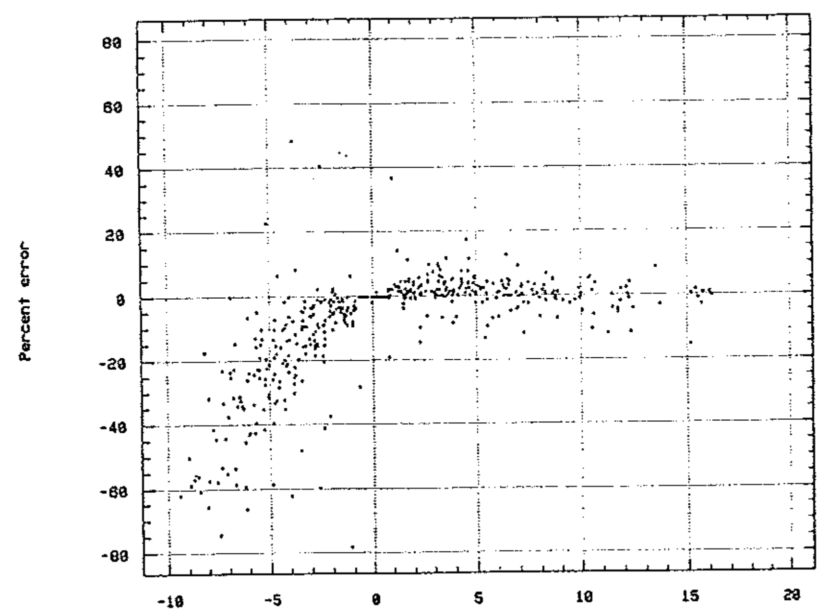

Fig. 3. Black–Scholes pricing bias.

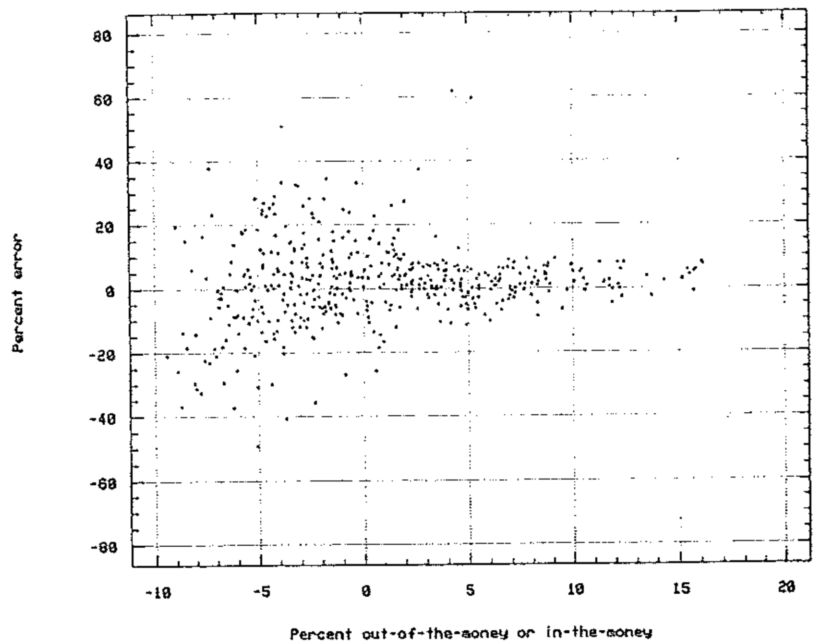

Fig. 4. Neural network pricing bias.

ple and for the differences between the model price and the actual price are reported. The null hypothesis of no difference in the means is rejected at the 5% significance level for each model. The 95% confidence intervals for the mean differences show that the Black–Scholes consistently overprices the options, while the neural network underprices them. We also observe that the standard deviation of the differences is smaller in the neural network prices.

Results of the paired sample comparisons test for the in-the-money cases are shown in table 7. There is a statistically significant difference between the means of the sample of neural network predictions and the sample of actual market prices. This is not surprising, since the bias tests indicated the tendency of the neural network to consistently underestimate prices. The Black–Scholes, however, did not show a significant difference from zero; hence it provides a better model for in-the-money, for this data set.

Scatterplots were developed showing the market prices versus the Black–Scholes prices (see figures 5 and 7) and the market prices versus neural network predictions (see figures 6 and 8). From figures 5 and 6, which show out-of-the-money prices, we observe more outliers in the Black–Scholes estimates than in the neural network estimates. This is consistent with the higher standard deviation found in the paired comparisons test. For the neural network estimates, prices furthest from at-the-money are more clustered than for the Black–Scholes. While a strong linear relationship is indicated in each, more variation is observed in the Black–Scholes as the market prices become larger, i.e., as prices move further from at-the-money. Figures 7 and 8, which show in-the-money prices, show more consistent spread for the neural network prices, while the Black–Scholes prices are more clustered near at-the-money, which is the expected result.

A few observations about the results can be made. First, although we have only presented summary statistics, one can observe similarities between the individual price estimates made by

Table 5. Results of averaging the Black–Scholes and neural network estimates, out-of-the-money

Week beginning	MSE	MAD	MAPE
April 23	0.108835	0.299466	25.21678
May 7	0.040011	0.170364	9.703556
May 21	0.855436	0.430711	12.11847
June 4	0.061369	0.143322	4.920285
June 18	0.318628	0.330394	16.56730

Table 6. Out-of-the-money, paired samples comparison

A. Paired samples comparison with Black–Scholes

	Black–Scholes	Market price	Differences
Mean	3.96412	3.45731	0.506807
Variance	5.88913	5.57354	0.72131
Std. deviation	2.42675	2.36084	0.8493
95% confidence intervals for differences:			
Mean:	(0.394979, 0.618635)		
Variance:	(0.604129, 0.876435)		
Std. deviation:	(0.777257, 0.936181)		
Sample size	$N = 224$		

B. Paired samples comparison with neural networks

	Network	Market price	Differences
Mean	3.33894	3.45731	−0.118374
Variance	4.84811	5.57354	0.23783
Std. deviation	2.20184	2.36084	0.487678
95% confidence intervals for differences:			
Mean:	(−0.182587, −0.0541612)		
Variance:	(0.199193, 0.288978)		
Std. deviation:	(0.44631, 0.537566)		
Sample size	$N = 224$		

the two models. Each model has difficulty computing prices when the trades are deep in-the-money. This is expected for the neural network because the majority of trades are close to at-the-money, and thus there are insufficient examples to present to the network for these cases. Secondly, we would not expect to achieve results with the neural network that are significantly different than those of Black–Scholes if many traders are using the Black–Scholes model and the market prices reflect their strategies. The neural network is only capable of learning the relationships that are embedded in the observations. The neural network exhibited a bias of underpricing the options and in fact may be best utilized as input into another pricing mechanism, or when averaged with the overpriced Black–Scholes estimate.

6. Summary and Conclusions

This empirical examination of the Black–Scholes option-valuation model and the neural network option-pricing model leads to some interesting

Table 7. In-the-money, paired samples comparison

A. Paired samples comparison with Black–Scholes

	Black–Scholes	Market price	Differences
Mean	21.4778	21.58	−0.102209
Variance	104.888	101.118	1.41015
Std. deviation	10.2415	10.0557	1.1875
95% confidence intervals for differences:			
Mean:	(−0.253529, 0.0491108)		
Variance:	(1.18749, 1.70225)		
Std. deviation:	(1.08972, 1.3047)		
Sample size	$N = 239$		

B. Paired samples comparison with neural networks

	Network	Market price	Differences
Mean	21.0799	21.5785	−0.498506
Variance	95.9599	100.656	1.78591
Std. deviation	9.79591	10.0328	1.33638
95% confidence intervals for differences:			
Mean:	(−0.668798, −0.328215)		
Variance:	(1.50392, 2.15585)		
Std. deviation:	(1.22634, 1.46828)		
Sample size	$N = 239$		

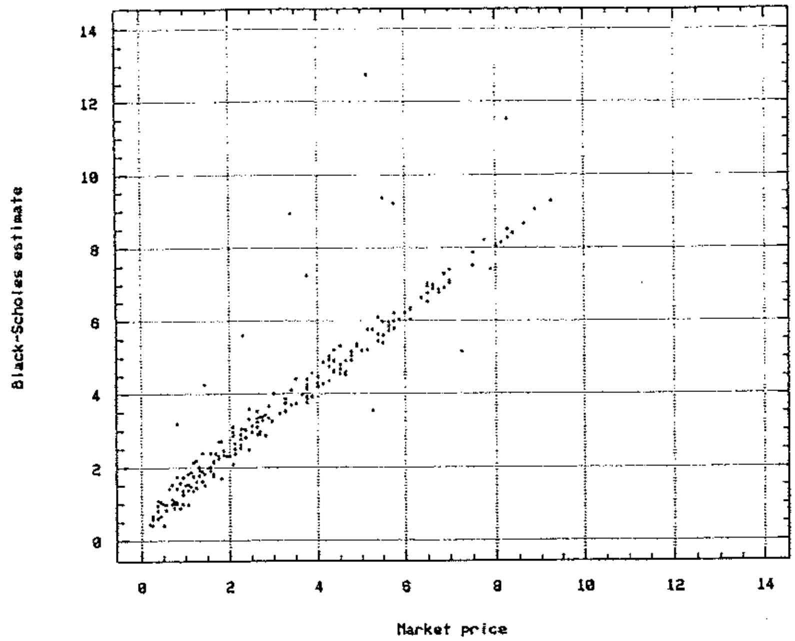

Fig. 5. Black–Scholes estimates versus market prices out-of-the-money.

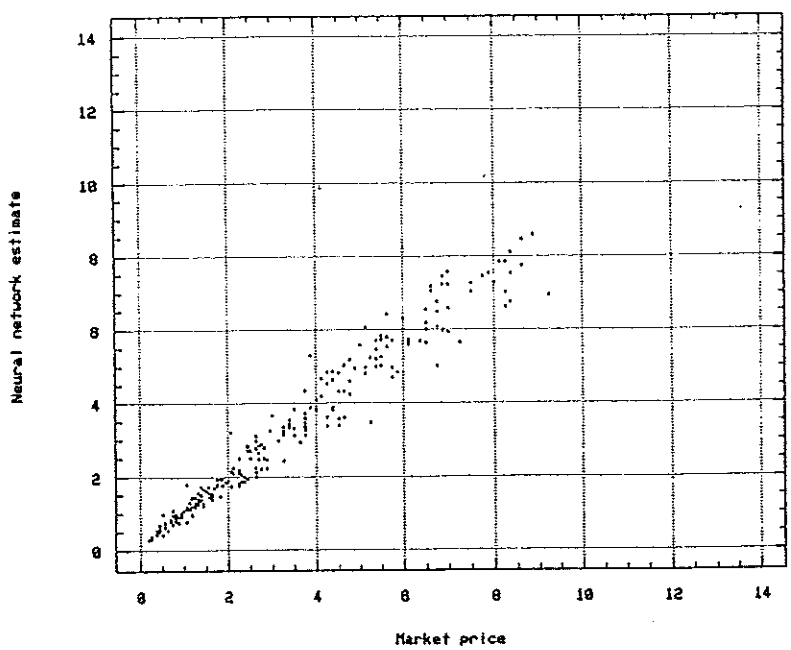

Fig. 6. Neural network estimates versus market prices out-of-the-money.

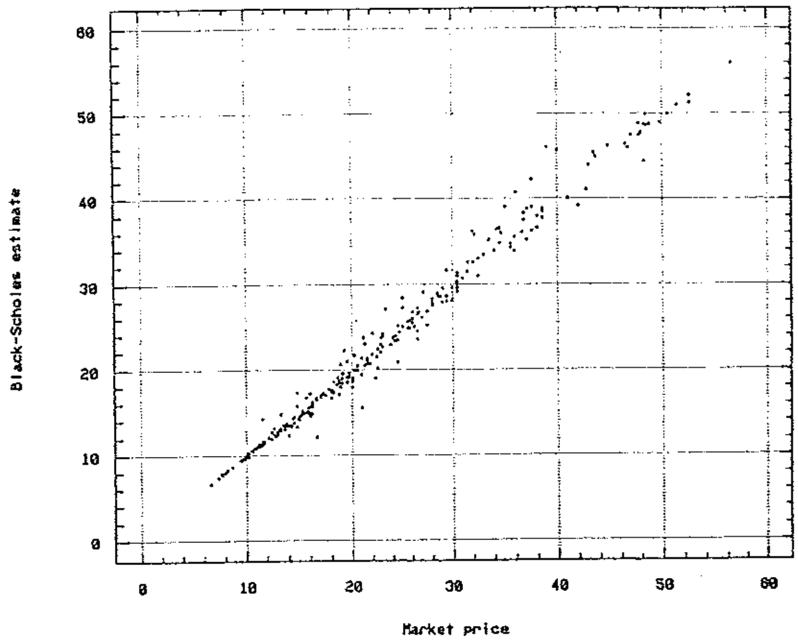

Fig. 7. Black–Scholes estimates versus market prices in-the-money.

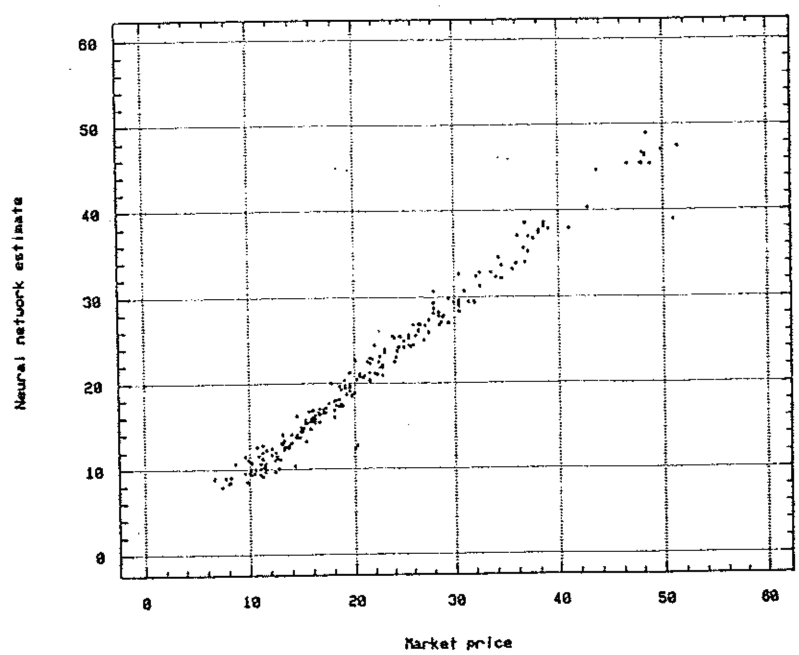

Fig. 8. Neural network estimates versus market prices in-the-money.

conclusions. While both models perform best when estimating prices close to at-the-money, the Black–Scholes makes greater overpricing errors deep-out-of-the-money, showing many more outliers. A common result emerges for in-the-money cases, with both models consistently underpricing options. However, for both in- and out-of the money prices, the neural network outperforms the Black–Scholes model in about 50% of the cases examined, as measured by the mean squared error.

Our results demonstrate that the neural network methodology offers a valuable alternative to estimating option prices to the traditional Black–Scholes model. For out-of-the-money estimates, better results were obtained by averaging the prices from the two models. The evidence reported here is encouraging, particularly in view of the essentially undisputed superiority of the Black–Scholes model. Analytically, it is interesting that the well-developed methodology of Black–Scholes, with its explicit formula for pricing options, derived using sophisticated financial arbitrage arguments and advanced stochastic calculus techniques, can actually be approximated by neural networks.

There are several limitations that may restrict the use of neural network models for estimation. There is no formal theory for determining optimal network topology, and therefore, decisions like the appropriate number of layers and middle-layer nodes must be determined using experimentation. The development and interpretation of neural network models requires more expertise from the user than traditional analytical models. Training a neural network can be computationally intensive, and the results are sensitive to the selection of learning parameters, activation function, topology of the network, and the composition of the data set.

References

1. S. Dutta and S. Shekhar, "Bond-rating: A non-conservative application of neural networks," in *Proc. IEEE Int. Conf. Neural Networks*, IEEE Computer Press: Los Alamitos, CA, 1988, pp. 142–150.
2. G. Jang, F. Lai, B. Jiang, and L. Chien, "An intelligent trend prediction and reversal recognition system using dual-module neural networks", in *Proc. First Int. Conf. Artif. Intell. Appl. on Wall Street*, IEEE Computer Press: New York, 1991, pp. 42–51.
3. L. Salchenberger, E.M. Cinar, and N. Lash, "Neural networks: A new tool for predicting thrift failures," *Decision Sci.*, vol. 23, no. 4, pp. 899–916, 1992.
4. E. Collins, S. Ghosh, and C. Scofield, "An application of a multiple neural-network learning system to emulation of mortgage underwriting judgments," in *Proc. IEEE Int. Conf. Neural Networks*, IEEE Computer Press: Los Alamitos, CA, 1988, pp. 459–466.
5. F. Black and M. Scholes, "The pricing of options and corporate liabilities," *J. Pol. Econ.*, vol. 81, pp. 637–654, 1973.
6. J. Macbeth and L. Merville, "An empirical examination of the Black–Scholes call option pricing model," *J. Fin.*, vol. 34, no. 5, pp. 1173–1186, 1979.
7. D. Chiras and S. Manaster, "The information content of option prices and a test of market efficiency," *J. Finl. Econ.*, vol. 6, pp. 213–234, 1978.
8. A. Malliaris and W. Brock, *Stochastic Methods in Economics and Finance*, North-Holland: Amsterdam, 1982.
9. M. Chesney and L. Scott, "Pricing European currency options: a comparison of the modified Black–Scholes model and a random variance model," *J. Fin. Quant. Anal.*, vol. 24, pp. 267–284, 1989.
10. D. Galai, "A survey of empirical tests of option-pricing models," in *Option Pricing*, edited by M. Brenner, Lexington Books: Lexington, MA, 1983.
11. R. Yap, "Constraint logic programming and option trading," *IEEE Expert*, vol. 2, no. 3, pp. 42–50, 1987.
12. F. Mizoguchi and L. Ohwada H, "A constraint-oriented decision support system for option-based investment strategies," *Proc. First Int. Conf. Artif. Intell. Appl. on Wall Street*, IEEE Computer Press: New York, 1991, pp. 160–167.
13. H. White, "Some asymtotic result for learning in single hidden layer feedforward network models," *J. Am. Statist. Assoc.*, vol. 84, pp. 1003–1013, 1989.
14. D.E. Rumelhart and J.L. McClelland, *Parallel Distributed Processing*, MIT Press: Cambridge, MA, 1986.
15. K. Tam and M. Kiang, "Managerial applications of neural networks: The case of bank failure predictions," *Management Sci.*, vol. 38, no. 7, pp. 926–947, July 1992.
16. D. Hawley, J. Johnson, and D. Raina, "Artificial neural systems: A new tool for decision making", *Finl. Analyst*, vol. 46, pp. 63–72, November 1990.
17. G. Swales and Y. Yoon, "Applying artificial neural networks to investment analysis," *Finl. Analyst*, vol. 69, pp. 78–80, October 1992.
18. S. Geisser, "The predictive sample reuse method with applications," *J. Am. Statist. Assoc.*, vol. 70, pp. 350–360, June 1975.
19. F. Mosteller and J.W. Tukey, "Data analysis, including statistics," in *Handbook of Social Psychology*, edited by G. Lindzey and E. Aronson, Addison Wesley: Reading, MA, 1968.
20. A. Weigend, B. Huberman, and D. Rumelhart, "Predicting sunspots and exchange rates with connectionist networks," in *Nonlinear Modeling and Forecasting*, Proc. vol. XII, Sante Fe. Addison Wesley: Reading, MA, 1990, pp. 395–432.

Mary E. Malliaris is an Assistant Professor of Management Science at Loyola University of Chicago. Her research interests include artificial intelligence, neural networks, and applications in economics and finance. Malliaris holds an undergraduate degree from Louisiana State University in mathematics, an M.S. from Northwestern University in mathematics, and a Ph.D. from Loyola University in research methodology.

Linda M. Salchenberger is an Associate Professor of Management Science at Loyola University of Chicago. She received her Ph.D. from Northwestern University in Managerial Economics and Decision Sciences. Her research interests include applications of artificial intelligence, including neural networks and expert systems, and she has published in *Information and Management* and *Decision Sciences*.