行政院國家科學委員會專題研究計劃成果報告
神經網路在選擇權定價上之應用
The pricing of option with Neural Networks

計劃編號：NSC 88-2416-H-004 –013
執行期限：87 年 8 月 1 日至 89 年 4 月 31 日
主持人：陳威光　國立政治大學金融系

1. 摘要

應用類神經網路的技術在財務金融領域是最近幾年來熱門的課題，如同 Kuan and White(1994)所言：神經網路是一種非線性參數模型，而網路系統的學習過程好比進行模型參數之估計。本文採用最常被應用的倒傳遞網路系統，以及理解神經網路兩系統來做為亞式選擇權的評價。隨著特殊選擇權產品的快速發展，如亞式選擇權(或稱平均式選擇權)，這些選擇權的定價公式對理論界或實務界也愈來愈有其迫切性。然而最為市場採用的算術平均選擇權，因其分配不是對數常態分配，因此並沒有標準公式解。大多是以數值的方法逼近。本研究的目的在於利用神經網路非線性的學習特性來評價這些選擇權。本文採用神經網路的學習系統來評價亞式選擇權。本文結果發現網路學習效果具有相當不錯的準確性。

關鍵詞：倒傳遞網路、理解神經網路、亞式選擇權

1. Abstract

It has been a hot issue to utilize the technique of artificial Neural Network (ANN) to the financial markets such as stocks, and futures markets. The purpose of this paper is to apply the Back Propagation Neural Network (BP) and Reasoning Neural Network (RN) to the valuation of Asia option. The option markets have grown dramatically since Black-Scholes (1973) derived their famous option pricing model. The markets for the exotic products especial for Asia options (average rate option) have grown rapidly for the past decade. There is no closed-form solution for the arithmetic average option since the distribution of the sum of the stock price is not lognormal. The propose of the paper is to utilize the BP Neural Network as well as RN Neural Network to price the Asia option. It is shown that artificial neural networks can be successfully employed to approximate formulas for the Asian option, for which analytic foemulas cannot derived.

Keywords：Back Propagation Neural Network, Reasoning Neural Network, Asia options

2. Introduction

Over the past few years, artificial neural networks have become an important tool for the financial analyst, not only for academic but also for practice. The majority of applications focus on forecasting time series, since artificial neural network proved their ability to model high-dimensional non-linear dependencies. Artificial neural networks have been proposed for many applications in the financial area, such as stocks, bonds and futures. For these applications, artificial neural networks are attractive because they have following merits: a nonlinear model, a learning ability, no predefined-functionality, and a generalization ability. Today, most interesting artificial neural networks are nonlinear models. For instance, the multi-layer perceptron (MLP) trained using the Back Propagation (BP) algorithm as described in Rumelhart et al.(1986) is a widely used artificial neural networks. The nonlinear results from the use of nodes with semi-linear activation functions. Instead of extracting explicit rules from domain experts, artificial neural networks employ a learning algorithm to autonomously extract nonlinearity.

Up to now, very few researches exist in option valuation. The option markets have grown dramatically since Black-Scholes (1973) derived their famous option pricing model. The markets for the exotic products especial for Asia options (average rate option) have grown rapidly for the past decade. There is no closed-form solution for the arithmetic average option since the distribution of the sum of the stock price is not lognormal. The purpose of the paper is to utilize the BP Neural Network as well as RN Neural Network to price the Asia option. The results have shown that artificial neural networks can be successfully employed to approximate formulas for the Asian option, for which analytic formulas cannot derived. This paper is organized as following. Section 2 discusses the option pricing model as well as the average rate option valuation. In section 3, the neural networks will be discussed. Section 4 will present the Monte Carlo simulation and the testing results.

3. Option Pricing Model

Black and Scholes (1973) offer a neat formula enabling investors pricing the call. They derive their option pricing model based on a no arbitrage relationship between the risk-free rate and the return on a portfolio containing the option and the stock. The Black-Scholes model assumes that the underlying asset prices obey a lognormal distribution. Mathematically speaking, the stock price follows a Geometric

Brownian motion:

$$dS = \mu S dt + \sigma S dz \qquad (1)$$

Where μ and σ are the instantaneous mean and standard deviation of stock return, respectively and dz is a Wiener process. Black-Scholes shows that the portfolio can be made risk-free by choosing an appropriate mixture of stock and call option. If the shares of the stock and option are continuously adjusted so that the returns of the hedged portfolio are risk-free all the time, then it must earn the risk-free rate to avoid arbitrage. In process of derivation, they solve for a partial differential equation and yield the formula for call option, which is

$$C_{BS} = SN(d_1) - Ke^{-r(T-T)}N(d_2) \qquad (2)$$

where

$$d_1 = \frac{In(S/K) + (r + 0.5\sigma^2)T}{\sigma\sqrt{T}}$$

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

Where S, K, σ, r and T are the stock price, exercise price, standard deviation of stock return, the constant risk-free rate and time to maturity, respectively. $N(\cdot)$ is cumulative normal density function.

Alternatively, in a risk-neutral world, the option can be priced according to its expected value at maturity, discounted back to the current period at the risk-free rate (e.g. Cox and Ross (1976)), This implies

$$C = e^{-rT} E \max(0, S_T - K) \qquad (3)$$

Where S_T is stock price at maturity time T, and E is the expectation operator. This call price, although is simple in its concept, cannot be implemented unless the data generating process of stock price is known. Boyle (1977) provides an early discussion of the use of Monte Carlo method in pricing option values with variance reduction techniques. Monte-Carlo experiment assumes the stock price, S_t, is first generated by the following data generating process according to the Geometric Brownian motion assumption.

$$S_t = S_{t-1} \exp[(r - 0.5\sigma^2)\Delta t + \sigma\varepsilon_t\sqrt{\Delta t}] \qquad (4)$$

where $(r - 0.5\sigma^2)$ is the annual mean logarithmic stock return and Δt stands for a short period of time (e.g. one day). Once the parameters r and σ are given, the above true stock price can be simulated using a random number generator to generate ε_t. The error term ε_t is drawn from a standard normal distribution with mean zero and standard deviation being equal to one. By use of equation (4) repeatedly, we may obtain the stock price S_T at maturity date and then the call price according to equation (3). Above procedure is repeated many times, say 10,000 time, the average value of these 10,000 calls is taken as the value of the option at time t. As the number of the simulations increases, the average value of calls converges to the Black-Scholes value.

4. Neural Networks

BP represents a convenient learning algorithm that is used by many researchers and practitioners. BP uses the generalized delta rule, which is a generalization of the Least-Mean-Square method. That is, the learning algorithm is defined to minimize an objective function as part of a search for an optimal weight vector (Z) for the network. The objective function is usually defined as

$$E(Z) \equiv \frac{1}{2kq}\sum_{c=1}^{k}\sum_{l=1}^{q}(O(B_c, Y_l, X) - d_{cl})^2 \quad (5)$$

During the learning process, all the training patterns are presented to the network, and their corresponding activation values are calculated. Then, the value of E(Z) is derived by comparing the actual outputs ($O(B_c, Y_l, X)s$) to the desired outputs (d_c/s) for all training patterns. To minimize the value of E(Z), Z is adjusted along the direction of the negative gradient ($-\nabla_Z(Z)$) such that the value for E(Z) is reduced. This process is repeated until either E(Z) or $\|\nabla_Z E(Z)\|$ is sufficiently small.

However, there are some undesirable characteristics of BP, such as the unknown optimal number of hidden-layer notes, the relatively optimal learning result, and the sluggish learning process. To address these undesirable characteristics of BP, Tsaih has developed Reasoning Neural Networks (RN). Although RN is applied to MLP networks, it belongs to the weight-and-structure-change category of learning algorithms. Initially, only one hidden-layer node is considered. Then, the algorithm seeks to recruit (add) or prune (remove) hidden-layer notes during the learning process. The ability to recruit or prune hidden-layer notes during the learning process helps to overcome the undesired characteristics of the BP algorithm.

There are four distinctive aspects to the RN lerning algorithm：(1) the Linearly Separating Condition (LSC), (2) the Thinking Mechanism, (3) the Cramming Mechanism, and (4) the Reasoning Mechanism. Each of these will now be addressed.

The RN is used for classification applications where our interest is in learning to determine if an input pattern is a number of one class of patterns, called class 1, or of a different class, called class 2. The LSC is used to improve the learning process. During the learning process, the training patterns are presented one by one. Let $K(k) \equiv \{1,...,k\}$ and $K(k) \equiv K_{l1}(k) \cup K_{l2}(k)$, where $K_{l1}(k)$ and $K_{l2}(k)$ are, respectively, the set of indices of the first k training pattern in class 1 and 2, with respect to the l th output node. Let the desired output values of classes 1 and 2 be 1.0 and −1.0, respectively, and the LSC(k, l) be

$$\min_{c \in K_{11}(k)} net(B_c, Y_l, X) > \max_{c \in K_{12}(k)} net(B_c, Y_1, X) \qquad (6)$$

When LSC(k, l) is satisfied, the network uses the following threshold value for correctly classifying :

$$v_1 = \frac{\min_{c \in K_{11}(k)} net(B_c, Y_l, X) + \max_{c \in K_{12}(k)} net(B_c, Y_1, X)}{2} \qquad (7)$$

That is,

$$B \in \begin{cases} Class_{11} & if \quad net(B, Y_1, X) \geq v_1 \\ Class_{12} & if \quad net(B, Y_1, X) < -v_1 \end{cases} \qquad (8)$$

Weights are adjusted via the Thinking Mechanism. Currently, the Thinking Mechanism is implemented using the momentum version of the generalized delta rule and a decay mechanism. However, this does not guarantee non-optimal results. One technique for escaping from non-optimal solutions (local minimum) involves recruiting (adding) hidden-layer nodes. By recruiting hidden-layer nodes, the dimension of the weight space increases and changes such that the non-optimal attractor is removed. This is implemented via the Cramming Mechanism.

However, by recruiting more hidden-layer nodes using the Cramming Mechanism, generalization ability of the network may degrade. Therefore, the RN learning procedure needs to be able to prune irrelevant hidden nodes during the learning process. In a Z, the i th hidden node is irrelevant with respect to the l th output node if LSC(k, l) is still satisfied with the same Z except $r_{li} = 0$, and a hidden node is irrelevant if it is irrelevant with respect to all output nodes. The Reasoning Mechanism is designed for the purpose of pruning hidden nodes. Thus, it involves not only the Thinking Mechanism but also a capability for pruning irrelevant hidden-layer nodes.

In summary, RN adopts a learning procedure that guarantees an optimal solution for 2-class categorization learning problems. At this point, however, RN is designed to deal only with binary output patterns. When working with non-binary outputs, real numbers can first be converted to binary digits. However, this will increase the number of output-layer nodes and the learning complexity, requiring longer learning times.

We use RN to enhance the performance of the BP algorithm. With respect to application problems with non-binary outputs, we propose the following learning procedure for MLP networks：(1)the RN learning procedure is first used to generate the network's structure (the number of hidden-layer nodes and the weights), and (2)the resulting network is then trained via BP. We call this the RNBP learning procedure.

5. Experiment Results and Conclusions

Though there is a closed form solution for the geometric average, there is no generally accepted closed-form solution for the arithmetic average rate option, since the distribution for the arithmetic average rate is not lognormal. Numerical approaches are most common in the valuation of these options. Kemna and Vorst(1990) propose a Monte Carlo approach to valuate arithmetic average rate options, which employing the corresponding geometric average rate option as a control variate. Turnbull and Wakeman (1991) apply the Edgeworth series expansion around the lognormal distribution to value the arithmetic average rate option. They provide an algorithm to compute moments for the arithmetic average. Levy (1992), however, develops a closed-form analytical approximations for valuing the arithmetic average rate option on foreign exchange rates based on a defined Wilkinson approximation. The above reviewed methods all adopt some regularity restrictions to approximate the true values of the average rate options. These restrictions more or less save computation time but involve drawbacks. For example, Levy's (1992) approach relies on the assumption that the distribution of the sums of log normal variates is itself well approximated at least to a first and second order by the log normal. However, the accuracy of the pricing formula deteriorates as volatility increases as Levy indicates. The values of the Asian option are computed through Monte Carlo simulation with control variate method.

Under the risk-neutral assumptions, the Black-Scholes type option valuation formula can be obtains as long as the expected final stock return and variance are known. The general formula as followings:

$$C=e^{M+0.5V-rT}\,N(d_1) - Ke^{-rt}N(d_2) \tag{9}$$

where, $d_1=\dfrac{M-\ln K+V}{\sqrt{V}}$

$d_2=d_1-\sqrt{V}$

The geometric average $\overline{S_G}=\sqrt[n]{S_1\cdot S_2\cdot S_i\cdots S_n}$，the expected value of $\ln\overline{S_G}$ is M, the variance of $\ln\overline{S_G}$ is V, then

$$M = \ln S + \frac{1}{n}\sum_{i=1}^{n}(r - 0.5\sigma^2)T_i$$

$$= \ln S + \frac{n+1}{2n}(r - 0.5\sigma^2)T$$

$$V = \frac{\sigma^2}{n^2}\sum_{i=1}^{n} = \frac{(n+1)(2n+1)}{6n^2}\sigma^2 T$$

And for the arithmetic average

$$E(\overline{S_A}) = \frac{S}{n}\sum_{i=1}^{n} e^{rT_i}$$

By using the arithmetic average and the variance from geometric average, we obtain the modified geometric average, which is closer to the true value of arithmetic average. The geometric average in this experiment is as control variate to speed the converge.

The training data contains 180 observations, and the testing data also contains 140 observations. The stock volatility are from 0.2 to 1.0 annual, the moneyness, defined as the strike price divided by the stock price are from 0.6 to 1.5. The average periods are quarterly average, monthly average, weekly average, and daily average.

The prices of Asian option used to train the network are obtained using Monte Carlo Simulations. In order to speed the converge, the Control Variate method are used. The used of variance reduction technique leads to more efficient estimates of option prices. The simulation runs 10,000 times. The simulation results are showed in Table 1 and Figure one. The data shows that the error between the network and the true value are small. There is a interesting result that the mispricing is negative for the out-of-the-money option, but is positive for the in-the-money option. In this paper, it is shown that artificial neural networks can be successfully employed to approximate formulas for the Asian option, for which analytic formulas cannot derived.

6. References

1.Black, F., and M. Scholes (1973) "The Pricing of Options and Corporate Liabilities," Journal of Political Economy, Vol.81 pp. 637-659.

2.Boyle, P.P., 1977, Options: A Monte Carlo Approach", Journal of Financial Economics, 323-338

3.Chen, S.-H and W.-C. Lee (1977), "Option Pricing with Genetic Algorithms : A Second Report," Proceedings of 1997 International Conference on Neural Networks, IEEE Press.

4.Chen, W.K., R.W Tsaih and Y.Lin "Application of Neural Network to Financial Swaps", Journal of Computational Intelligence in Finance, pp. 27-37, May 1998.

5.Cox, J.C. and S.A. Ross, 1976, The Valuation of Options for Alternative Stochastic Processes, Journal of Financial Economics, 145-166.