International Journal of Science and Research (IJSR)
ISSN: 2319-7064
Impact Factor 2024: 7.101

Sensitivity Analysis of Input Variables in ANN-Based Option Pricing Models

Hanningtone Simiyu

[1]Bomet University College, Department of Mathematics and Computer Science, P.O. Box 701-20400, Bomet - Kenya
Email: *shanningtone13[at]email.com*

Abstract: ***Parametric models in option pricing have consistently failed to provide results truly consistent with the observed market data, due to the fact that market participants change their option pricing attitudes from time to time. Furthermore, most of these models are based on questionable assumptions such as the assumption of constant volatility, yet empirical evidence strongly suggests otherwise. This has led to the development and the use of alternative non-parametric models such as the ANN models. However, in spite of the well-known significance of the sensitivity indices in option pricing, little attempts have been made to provide a mechanism for computing these indices from the ANN model. Consequently, this study sought to provide a mechanism for and conduct sensitivity analysis from the ANN model outputs and conclude by comparing the performance of the hybrid model and the conventional ANN model developed. All the indices developed lied within their desired range of values and this, being a resultant of the derivatives of the developed ANN output with respect to its various inputs, demonstrated how reliable the model was. Results not only demonstrate how the indices could be computed from the ANN models, but also play a significant role in the validation of the model.***

Keywords: Sensitivity Analysis, Option Pricing, Neural Networks

1. Introduction

Among the non-parametric techniques, perhaps the most fertile area for empirical research has been estimating option pricing formula using neural network [1]. The very first attempt to do this was made by [2]. The study used three different network architectures namely: Radial Basis Function (RBF), Multilayer perceptron (MLP) and Projection Pursuit Regression (PPR) to fit both Monte-Carlo simulated Brownian underlier and Black-Scholes option data and S & P 500 futures thereof. The authors however used a minimalist approach in the selection of their inputs and restricted the network inputs to time maturity $(T - t)$ and Moneyness. Interest rate and volatility was assumed to be constant. The study used financial knowledge in construction, namely the "homogeneity property" of the option price formula which was borrowed from [3], consequently justifying the use of moneyness instead of the underlying price and strike price separately.

[4] compared the option pricing performance of the ANN model to the Black-Scholes and the GARCH pricing models. The study used a MLP with a single layer of hidden nodes. The ANN was trained on the implied volatility rather than the option price and this led to an improved performance in terms of validation errors compared to the competing models. The hedging performance of the neural network, the GARCH option-pricing model and the Black-Scholes were also analyzed. According to [5], one of the limitations of the BSM that prompted the application of the ANN is the controversial assumption that the underlying probability distribution is lognormal. The study thus proposed a couple of hybrid models to reduce these limitations and enhance the ability of option pricing. The key input to their option pricing model was volatility, in which three popular GARCH type models were used in estimating volatility. Two non-parametric models based on the neural networks and the neuro-fuzzy networks were then developed to price call options for S&P 500 index. Results were then compared with those of the BSM and they showed that both the neural network and the neuro-fuzzy network models outperformed the BSM. Furthermore, comparing the neural network and the neuro-fuzzy approaches, [6] observed that for at-the-money options (ATM), the neural network model performed better and for both in-the-money (ITM) and out-of-the money (OTM) options, the neuro-fuzzy model provided better results. [7], using Nifty call option prices, made an attempt to improve the accuracy of option price estimation using ANNs by adjusting all input parameters using a suitable multiplier. The values of these multipliers were determined using known data that minimizes errors in valuation.

Another application involving ANN on option pricing was one done by [8]. The study compared the option pricing ability of Robust ANNs optimized with the Huber function against those optimized with Least Squares. The comparison was in respect to pricing European call options on the S&P 500 using daily data for the period April 1998 to August 2001. In the study, the analysis was augmented with the use of several historical and implied volatility measures. The study also went a step further to include hybrid networks that directly incorporated information from the parametric model in the analysis. It was found that the ANN models with the use of the Huber function outperformed the ones optimized with least squares in terms of the performance errors. [9] applied a hybrid neural network which preprocessed financial input data for improving the estimation of option market prices. The model in this study comprised of two parts. In the first part, a neural network model was developed to estimate volatility, while in the second part an additional neural network was developed to value the difference between the BSM results and the actual option market prices. The resulting option price was then a summation between the BSM and the network response. The study obtained that the hybrid system with a neural network for estimating volatility provided better performance in terms of pricing accuracy than either the BSM with historical volatility, or the BSM with volatility valued by the neural network.

[10], developed an ANN model that processes financial input data to estimate market option prices at closing. The ANNs ability was compared to the BSM, a comparison that revealed that the MSE for the ANN was less than that of the BSM in more than half the cases examined. The ANN model used exactly the same financial data as the BSM. [11] examined whether an MLP ANN, could be used to find a call option pricing formula better corresponding to market prices and the properties of the underlying asset than the Black-Scholes formula. The neural network method was applied to the out-of-sample pricing and delta-hedging of daily Swedish stock index call options from 1997-1999, with the BSM with historical and implied volatility as a benchmark. The findings revealed that the ANN outperformed the benchmarks in both pricing and hedging [11]. [12] applied a non-parametric modular neural network (MNN) model to price the S&P-500 European call options. The modules were based on time to maturity and moneyness of the options. The option price function of interest was homogenous of degree one with respect to the underlying index price and the strike price. The study found that modularity improved the generalization properties of standard feedforward ANN option pricing models (with or without the homogeneity), relative to the Black-Scholes model.

From the literature reviewed, all the studies in which ANN was applied in option pricing attach this resolve to the shortcomings of the BSM with regards to its questionable assumptions such as log-normality of financial data [5], constant volatility [9], among others. However, hardly any study uses sensitivity analysis in the validation of the model with nearly all the studies only using the validation errors in validating their models.

2. Materials and Methods

Data

The study used intraday data for the AAPL stock option for the period between December 2016 and March 2017 with 56,238 data points. After modelling the option prices using ANN, we perform a sensitivity analysis by deriving the Greek letters with respect to the ANN model in a bid to demonstrate the role played by each of the inputs used in determining/ influencing the option prices.

Sensitivity analysis

Sensitivity analysis of model output investigates the relationship between the outputs of a model, possibly implemented in a computer program and its input variables (Saltelli et al., 1991). The relevance of this ranges from quality assurance of the model to identification of critical regions in the parameter space, just to mention but a few. Such sensitivities in option pricing can represent the different dimensions to the risk in an option. Financial institutions selling options to their clients can manage their risks using these sensitivities. Sensitivity analysis techniques are purely model dependent and normally quite easier for linear models than for non-linear models.

Different sensitivity analysis techniques exist ranging from basic correlation analysis to advanced statistics. In option pricing however, sensitivity analysis is performed differently. Sensitivity analysis on option pricing is done using "Greek letters". The author defines "Greek letters" as the sensitivities of the option prices to a single-unit change in the value of either a state variable or a parameter. Financial institutions selling options to their clients can therefore manage their risk by Greek letters analysis. We adopt and explore this definition and line of argument and proceed to compute the sensitivity indices for option prices on our input variables namely: underlying asset price, time to maturity and volatility.

Proposition 1

For an option price P, with a strike price X, underlying asset price S and time to maturity T, the sensitivities (Greeks) of the option price with respect to the underlying asset price, time to maturity and volatility are given by:

$$\begin{aligned}\Delta &= \frac{\partial P}{\partial S}\\ \Theta &= \frac{\partial P}{\partial \tau}\\ v &= \frac{\partial P}{\partial \sigma}\end{aligned} \tag{1}$$

Now, Recall that the output of the ANN model is sigmoid-transformed which is given by:

$$\varphi(y) = \frac{1}{1+e^{-y}}$$

Proposition 2

The derivative (with respect to y) of the sigmoid function is given by:

$$\frac{e^{-y}}{\left(1+e^{-y}\right)^2} = \varphi(y)(1-\varphi(y)) \tag{2}$$

Proof:

$$\begin{aligned}\frac{\partial}{\partial y}\varphi(y) &= \frac{\partial}{\partial y}\left(\frac{1}{1+e^{-y}}\right)\\ &= \frac{\partial}{\partial y}\left(1+e^{-y}\right)^{-1}\\ &= -\left(1+e^{-y}\right)^{-2}\left(-e^{-y}\right)\\ &= \frac{e^{-y}}{\left(1+e^{-y}\right)^2}\end{aligned} \tag{3}$$

In which case we can proceed and re-write the derivative in terms of $\varphi(y)$ as follows:

$$\begin{aligned}\frac{e^{-y}}{\left(1+e^{-y}\right)^2} &= \frac{1}{1+e^{-y}} \cdot \frac{e^{-y}}{1+e^{-y}}\\ &= \frac{1}{1+e^{-y}} \cdot \frac{\left(1+e^{-y}\right)-1}{1+e^{-y}}\\ &= \frac{1}{1+e^{-y}} \cdot \left(\frac{1+e^{-y}}{1+e^{-y}} - \frac{1}{1+e^{-y}}\right)\\ &= \frac{1}{1+e^{-y}} \cdot \left(1 - \frac{1}{1+e^{-y}}\right)\\ &= \varphi(y)(1-\varphi(y))\end{aligned} \tag{4}$$

Completing the proof.

Now, consider the resultant output from the output neuron of the hybrid ANN model which represents the scaled APPL call option price for each α, for $\alpha = 1,2,\ldots,n$ is given by:

$$f_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = \frac{1}{1+\exp\left[-\omega_0 - \sum_{j=1}^{H}\omega_{0j}\left\{\frac{1}{1+\exp\left\{-\omega_{j0} - \omega_{jM}\left(\frac{X}{S_t}\right)^{*}_{\alpha} - \omega_{j\tau}\tau^{*}_{\alpha} - \omega_{j\sigma(AGG)}\left(\sigma^{2*}_{t(AGG)\alpha}\right)\right\}}\right\}\right]}$$

Which can be re-written as:

$$f_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = \frac{1}{1+e^{-I_{j0}}} \tag{5}$$

Where I_{jo}, represents the net input to the output neuron and is given by:

$$I_{jo} = \omega_0 + \sum_{j=1}^{H}\omega_{0j}\left\{\frac{1}{1+\exp\left\{-\omega_{j0} - \omega_{jM}\left(\frac{X}{S_t}\right)^{*}_{\alpha} - \omega_{j\tau}\tau^{*}_{\alpha} - \omega_{j\sigma(AGG)}\left(\sigma^{2*}_{t(AGG)\alpha}\right)\right\}}\right\}$$

The later equation represents the transformation of the input I_{jH} to the hidden neuron and is given by;

$$Q_j\left(\omega_{j0}, \omega_{jM}, \omega_{j\tau}, \omega_{j\sigma(AGG)}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = \frac{1}{1+\exp\left\{-\omega_{j0} - \omega_{jM}\left(\frac{X}{S_t}\right)^{*}_{\alpha} - \omega_{j\tau}\tau^{*}_{\alpha} - \omega_{j\sigma(AGG)}\left(\sigma^{2*}_{t(AGG)\alpha}\right)\right\}}$$

Which implies that equation I_{j0} can also be re-written as

$$\begin{aligned} I_{jo} &= \omega_0 + \sum_{j=1}^{H}\omega_{0j}\left\{\frac{1}{1+\exp\left\{-\omega_{j0} - \omega_{jM}\left(\frac{X}{S_t}\right)^{*}_{\alpha} - \omega_{j\tau}\tau^{*}_{\alpha} - \omega_{j\sigma(AGG)}\left(\sigma^{2*}_{t(AGG)\alpha}\right)\right\}}\right\} \\ &= \omega_0 + \sum_{j=1}^{H}\omega_{0j}Q_j\left(\omega_{j0}, \omega_{jM}, \omega_{j\tau}, \omega_{j\sigma(AGG)}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) \end{aligned} \tag{6}$$

It should be noted that equation I_{j0} is a net representation of the net input to the j^{th} hidden node which was given by:

$$I_{jH} = \omega_{j0} + \omega_{jM}\left(\frac{X}{S_t}\right)^{*}_{\alpha} + \omega_{j\tau}\tau^{*}_{\alpha} + \omega_{j\sigma(AGG)}\left(\sigma^{2*}_{t(AGG)\alpha}\right)$$

This leads to a transformation of the input I_{jH} to the j^{th} hidden neuron as follows:

$$\begin{aligned} Q_j\left(\omega_{j0}, \omega_{jM}, \omega_{j\tau}, \omega_{j\sigma(AGG)}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) &= \frac{1}{1+\exp\left\{-\omega_{j0} - \omega_{jM}\left(\frac{X}{S_t}\right)^{*}_{\alpha} - \omega_{j\tau}\tau^{*}_{\alpha} - \omega_{j\sigma(AGG)}\left(\sigma^{2*}_{t(AGG)\alpha}\right)\right\}} \\ &= \frac{1}{1+e^{-I_{jH}}} \end{aligned} \tag{7}$$

From here, we proceed to obtain the different sensitivities i.e. Delta, Theta and Vega as follows:

(a) Delta ANN (Δ_{ANN}): Option prices versus the underlying asset price

Let, the ultimate option price be denoted and given for each α, for $\alpha = 1,2,\ldots,n$. by.

$$\gamma_{3\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = f_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\begin{bmatrix} Max\left\{f\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \\ -Min\left\{f\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \end{bmatrix} + Min\left\{f\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\}$$

Which is a descaled representation of scaled output. Then, the rate of change of the option price with respect to the rate of change of the underlying asset price, is denoted by Δ_{ANN} (Delta ANN) and is obtained as follows:

$$\Delta_{ANN} = \frac{\partial}{\partial S_t}\gamma_{3\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = \gamma'^{S}_{3\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)$$
$$= f'^{S}_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\begin{bmatrix} Max\left\{f'^{S}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \\ -Min\left\{f'^{s}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\}\end{bmatrix}$$
$$+ Min\left\{f'^{S}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \tag{8}$$

This is obtained by working backwards on I_{jH} as follows: I_{jH} is a representation of the net input to the j^{th} hidden node is given by:

$$I_{jH} = \omega_{j0} + \omega_{jM}\left(\frac{X}{S_t}\right)^*_{\alpha} + \omega_{j\tau}\tau^*_{\alpha} + \omega_{j\sigma(AGG)}\left(\sigma^{2*}_{t(AGG)\alpha}\right)$$

Now, the derivative of this equation I_{jH} above with respect to the underlying asset price S_t is given by:

$$I'^{S}_{jH} = \frac{\partial}{\partial S_t} I_{jH} = \omega_{jM}\frac{\partial}{\partial S_t}\left(\frac{X}{S_t}\right)^*_{\alpha}$$
$$= \omega_{jM}\frac{\partial}{\partial S_t}\left\{X(S_t)^{-1}\right\}^*_{\alpha}$$
$$= \omega_{jM}(-1)\left\{X(S_t)^{-2}\right\}^*_{\alpha}$$
$$= -\omega_{jM}\left\{\frac{X}{(S_t)^2}\right\}^*_{\alpha} \tag{9}$$

Next, note that the sigmoid transformation of the input I_{jH} to the j^{th} hidden neuron is given by equation 7. Thus, using proposition 2, the derivative with respect to S_t of this sigmoid transformation (equation 7) is given:

$$\frac{\partial}{\partial S_t}Q_j\left(\omega_3, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = \frac{\partial}{\partial S_t}\left(\frac{1}{1+e^{-I_{jH}}}\right)$$
$$= \frac{\partial}{\partial S_t}\left(1+e^{-I_{jH}}\right)^{-1}$$
$$= -\left(1+e^{-I_{jH}}\right)^{-2}\left(-I'^{S}_{jH}e^{-I_{jH}}\right)$$
$$= \frac{I'^{S}_{jH}e^{-I_{jH}}}{\left(1+e^{-I_{jH}}\right)^2} \tag{10}$$

Next, we find the derivative with respect to S_t of the net input to the output neuron given by equation 6 as:

$$I'^{S}_{j0} = \frac{\partial}{\partial S_t}I_{jo} = \frac{\partial}{\partial S_t}\left[\omega_0 + \sum_{j=1}^{H}\omega_{0j}Q_j\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right]$$
$$= \sum_{j=1}^{H}\omega_{0j}\frac{\partial}{\partial S_t}Q_j\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)$$
$$= \sum_{j=1}^{H}\omega_{0j}\frac{I'^{S}_{jH}e^{-I_{jH}}}{\left(1+e^{-I_{jH}}\right)^2} \tag{11}$$

Recall that the resultant output from the output neuron was given in equation by equation 5. Consequently, an application of proposition 2 on this expression yields the following derivative with respect to the underlying asset price.

$$f'^{S}_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = \frac{\partial}{\partial S_t}f_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = \frac{\partial}{\partial S_t}\left[\frac{1}{1+e^{-I_{j0}}}\right]$$
$$= \frac{\partial}{\partial S_t}\left(1+e^{-I_{j0}}\right)^{-1}$$
$$= -\left(1+e^{-I_{j0}}\right)^{-2}\left(-I'^{S}_{j0}e^{-I_{j0}}\right)$$
$$= \frac{I'^{S}_{j0}e^{-I_{j0}}}{\left(1+e^{-I_{j0}}\right)^2}$$
$$= \frac{e^{-I_{j0}}}{\left(1+e^{-I_{j0}}\right)^2}\sum_{j=1}^{H}\omega_{0j}\frac{I'^{S}_{jH}e^{-I_{jH}}}{\left(1+e^{-I_{jH}}\right)^2}$$
$$= \frac{e^{-I_{j0}}}{\left(1+e^{-I_{j0}}\right)^2}\sum_{j=1}^{H}\omega_{0j}\frac{-\omega_{jM}\left\{\frac{X}{(S_t)^2}\right\}^*_{\alpha}e^{-I_{jH}}}{\left(1+e^{-I_{jH}}\right)^2} \tag{12}$$

In which case the last two expressions in equation 12 are obtained by substituting I'^{s}_{j0} and I'^{s}_{jH} from equation 11 and 9 respectively.

Ultimately, our ANN delta will be given by:

$$\Delta_{ANN} = \gamma'^{S}_{3\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)$$
$$= f'^{S}_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\begin{bmatrix} Max\left\{f'^{S}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \\ -Min\left\{f'^{S}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\}\end{bmatrix}$$
$$+ Min\left\{f'^{S}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \tag{13}$$

(b) Theta ANN ($\boldsymbol{\Theta_{ANN}}$): Option prices versus time to maturity

The rate of change of an option price with respect to the passage of time is usually denoted by Θ and is defined as:

$$\Theta = \frac{\partial P}{\partial \tau} \tag{14}$$

In which case P is the option price and τ is the passage in time. This rate can be defined in terms of time to maturity $\tau = T - t$ so that:

$$\Theta = \frac{\partial P}{\partial \tau} = \frac{\partial P}{\partial \tau}\frac{\partial \tau}{\partial t} = (-1)\frac{\partial P}{\partial \tau} \tag{15}$$

This implies that Θ can be expressed as minus one times the rate of change of the option price with respect to time to maturity.

Recall that the ultimate option price be denoted and given for each α, for $\alpha = 1,2,\ldots,n$. by:

$$\gamma_{3\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = f_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\begin{bmatrix} Max\left\{f\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \\ -Min\left\{f\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\}\end{bmatrix}$$
$$+ Min\left\{f\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\}$$

Which is a descaled representation of the scaled output. Then, the rate of change of the option price with respect to the passage in time, is denoted by Θ_{ANN} (Theta ANN) and is obtained as follows:

$$\begin{aligned}\Theta_{ANN} &= \frac{\partial}{\partial\tau}\gamma_{3\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = \gamma'^{\tau}_{3\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) \\ &= f'^{\tau}_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\begin{bmatrix} Max\left\{f'^{\tau}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \\ -Min\left\{f'^{\tau}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\}\end{bmatrix} \\ &\quad + Min\left\{f'^{\tau}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\}\end{aligned} \quad (16)$$

This is obtained by working backwards on I_{jH} as follows: I_{jH} is a representation of the net input to the j^{th} hidden node was given by:

$$I_{jH} = \omega_{j0} + \omega_{jM}\left(\frac{X}{S_t}\right)^*_{\alpha} + \omega_{j\tau}\tau^*_{\alpha} + \omega_{j\sigma(AGG)}\left(\sigma^{2*}_{t(AGG)\alpha}\right) \quad (17)$$

Now, the derivative of I_{jH} above with respect to τ is given by:

$$I'^{\tau}_{jH} = \frac{\partial}{\partial\tau} I_{jH} = \omega_{j\tau} \quad (18)$$

Recall that the sigmoid transformation of the input I_{jH} to the j^{th} hidden neuron was given by equation 7. Thus, using proposition 2, the derivative with respect to τ of this sigmoid transformation is obtained as follows:

$$\begin{aligned}\frac{\partial}{\partial\tau}Q_j\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) &= \frac{\partial}{\partial\tau}\left(\frac{1}{1+e^{-I_{jH}}}\right) \\ &= \frac{\partial}{\partial\tau}\left(1+e^{-I_{jH}}\right)^{-1} \\ &= -\left(1+e^{-I_{jH}}\right)^{-2}\left(-I'^{\tau}_{jH}e^{-I_{jH}}\right) \\ &= \frac{I'^{\tau}_{jH}e^{-I_{jH}}}{\left(1+e^{-I_{jH}}\right)^2}\end{aligned} \quad (19)$$

Next, we find the derivative with respect to τ of the net input to the output neuron given by equation 6 as:

$$\begin{aligned}I'^{\tau}_{j0} = \frac{\partial}{\partial\tau} I_{jo} &= \frac{\partial}{\partial\tau}\left[\omega_0 + \sum_{j=1}^{H}\omega_{0j}Q_j\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right] \\ &= \sum_{j=1}^{H}\omega_{0j}\frac{\partial}{\partial\tau}Q_j\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) \\ &= \sum_{j=1}^{H}\omega_{0j}\frac{I'^{\tau}_{jH}e^{-I_{jH}}}{\left(1+e^{-I_{jH}}\right)^2}\end{aligned} \quad (20)$$

Recall that the resultant output from the output neuron was given in equation 5. Consequently, an application of proposition 2 on this expression yields the following derivative with respect to the time to maturity.

$$\begin{aligned}f'^{\tau}_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) &= \frac{\partial}{\partial\tau}f_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = \frac{\partial}{\partial\tau}\left[\frac{1}{1+e^{-I_{j0}}}\right] \\ &= \frac{\partial}{\partial\tau}\left(1+e^{-I_{j0}}\right)^{-1} \\ &= -\left(1+e^{-I_{j0}}\right)^{-2}\left(-I'^{\tau}_{j0}e^{-I_{j0}}\right) \\ &= \frac{I'^{\tau}_{j0}e^{-I_{j0}}}{\left(1+e^{-I_{j0}}\right)^2} \\ &= \frac{e^{-I_{j0}}}{\left(1+e^{-I_{j0}}\right)^2}\sum_{j=1}^{H}\omega_{0j}\frac{I'^{\tau}_{jH}e^{-I_{jH}}}{\left(1+e^{-I_{jH}}\right)^2} \\ &= \frac{e^{-I_{j0}}}{\left(1+e^{-I_{j0}}\right)^2}\sum_{j=1}^{H}\omega_{0j}\frac{-\omega_{j\tau}e^{-I_{jH}}}{\left(1+e^{-I_{jH}}\right)^2}\end{aligned} \quad (21)$$

In which case the last two expressions in equation 13 are obtained by substituting I'^{τ}_{j0} and I'^{τ}_{jH} from equation 20 and 18 respectively.

Ultimately, our ANN theta will be given by:

$$\Theta_{ANN} = -\frac{\partial}{\partial \tau}\gamma_{3\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = -\gamma'^{\tau}_{3\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)$$
$$= -\left[\begin{array}{l} f'^{\tau}_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\left[\begin{array}{l} Max\left\{f'^{\tau}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \\ - Min\left\{f'^{\tau}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\}\end{array}\right] \\ + Min\left\{f'^{\tau}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\}\end{array}\right] \quad (22)$$

(c) Vega ANN (Δ_{ANN}): Option prices versus the underlying asset price

In this case, we denote the sensitivity index by Vega (ν) and define it by:

$$\nu = \frac{\partial P}{\partial \sigma} \quad (23)$$

This index gives the rate of change of our option price with respect to volatility.

As before, let the ultimate option price be denoted and given for each α, for $\alpha = 1,2,\ldots,n$. as.

$$\gamma_{3\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = f_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\left[\begin{array}{l} Max\left\{f\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \\ - Min\left\{f\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\}\end{array}\right] + Min\left\{f\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\}$$

Which is a descaled representation scaled output. Then, the rate of change of the option price with respect to the rate of change of volatility, is denoted by ν_{ANN} (Vega ANN) and is obtained as follows:

$$\nu_{ANN} = \frac{\partial}{\partial \sigma^{2*}_{t(AGG)}}\gamma_{3\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = \gamma'^{\sigma^{2*}_{t(AGG)}}_{3\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)$$
$$= f'^{\sigma^{2*}_{t(AGG)}}_{\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\left[\begin{array}{l} Max\left\{f'^{\sigma^{2*}_{t(AGG)}}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \\ - Min\left\{f'^{\sigma^{2*}_{t(AGG)}}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\}\end{array}\right] + Min\left\{f'^{\sigma^{2*}_{t(AGG)}}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \quad (24)$$

This is again obtained by working backwards from equation I_{jH} as follows:

I_{jH} is a representation of the net input to the j^{th} hidden node was given by:

$$I_{jH} = \omega_{j0} + \omega_{jM}\left(\frac{X}{S_t}\right)^{*}_{\alpha} + \omega_{j\tau}\tau^{*}_{\alpha} + \omega_{j\sigma(AGG)}\left(\sigma^{2*}_{t(AGG)\alpha}\right)$$

Now, the derivative of I_{jH} with respect to $\sigma^2_{t(AGG)}$ is given by:

$$I'^{\sigma^{2*}_{t(AGG)}}_{jH} = \frac{\partial}{\partial \sigma^{2*}_{t(AGG)}} I_{jH} = \omega_{j\sigma(AGG)} \quad (25)$$

Next, recall that the sigmoid transformation of the input I_{jH} to the j^{th} hidden neuron was given by equation 7. Thus, using proposition 2, the derivative with respect to $\sigma^2_{t(AGG)}$ of this sigmoid transformation (equation 7) is given:

$$\frac{\partial}{\partial \sigma^{2*}_{t(AGG)}} Q_j\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) =$$
$$= \frac{\partial}{\partial \sigma^{2*}_{t(AGG)}}\left(\frac{1}{1+e^{-I_{jH}}}\right)$$
$$= \frac{\partial}{\partial \sigma^{2*}_{t(AGG)}}\left(1+e^{-I_{jH}}\right)^{-1}$$
$$= -\left(1+e^{-I_{jH}}\right)^{-2}\left(-I'^{\sigma^{2*}_{t(AGG)}}_{jH} e^{-I_{jH}}\right)$$
$$= \frac{I'^{\sigma^{2*}_{t(AGG)}}_{jH} e^{-I_{jH}}}{\left(1+e^{-I_{jH}}\right)^2} \quad (26)$$

Next, we find the derivative with respect to $\sigma^2_{t(AGG)}$ of the net input to the output neuron given by equation 6 as:

$$I'^{\sigma^{2*}_{t(AGG)}}_{j0} = \frac{\partial}{\partial \sigma^{2*}_{t(AGG)}} I_{jo} = \frac{\partial}{\partial \sigma^{2*}_{t(AGG)}}\left[\omega_0 + \sum_{j=1}^{H}\omega_{0j}Q_j\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right]$$
$$= \sum_{j=1}^{H}\omega_{0j}\frac{\partial}{\partial \sigma^{2*}_{t(AGG)}} Q_j\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)$$
$$= \sum_{j=1}^{H}\omega_{0j}\frac{I'^{\sigma^{2*}_{t(AGG)}}_{jH} e^{-I_{jH}}}{\left(1+e^{-I_{jH}}\right)^2} \quad (27)$$

Paper ID: SR25608164927 DOI: https://dx.doi.org/10.21275/SR25608164927

Consider the resultant output from the output neuron was given by equation 5. Consequently, an application of proposition 2 on this expression yields the following derivative with respect to volatility.

$$
\begin{aligned}
f_\alpha^{'\sigma^{2*}_{t(AGG)}}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) &= \frac{\partial}{\partial \sigma^{2*}_{t(AGG)}} f_\alpha\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = \frac{\partial}{\partial \sigma^{2*}_{t(AGG)}}\left[\frac{1}{1+e^{-I_{j0}}}\right] \\
&= \frac{\partial}{\partial \sigma^{2*}_{t(AGG)}}\left(1+e^{-I_{j0}}\right)^{-1} \\
&= -\left(1+e^{-I_{j0}}\right)^{-2}\left(-I_{j0}^{'\sigma^{2*}_{t(AGG)}} e^{-I_{j0}}\right) \\
&= \frac{I_{j0}^{'\sigma^{2*}_{t(AGG)}} e^{-I_{j0}}}{\left(1+e^{-I_{j0}}\right)^2} \\
&= \frac{e^{-I_{j0}}}{\left(1+e^{-I_{j0}}\right)^2} \sum_{j=1}^{H} \omega_{0j} \frac{I_{jH}^{'\sigma^{2*}_{t(AGG)}} e^{-I_{jH}}}{\left(1+e^{-I_{jH}}\right)^2} \\
&= \frac{e^{-I_{j0}}}{\left(1+e^{-I_{j0}}\right)^2} \sum_{j=1}^{H} \omega_{0j} \frac{\omega_{j\sigma(AGG)} e^{-I_{jH}}}{\left(1+e^{-I_{jH}}\right)^2}
\end{aligned}
\tag{28}
$$

In which case the last two expressions in equation 28 are obtained by substituting $I_{j0}^{'\sigma^2_{t(AGG)}}$ and $I_{jH}^{'\sigma^2_{t(AGG)}}$ from equation 27 and 25 respectively.

Ultimately, our Vega ANN will be given by:

$$
\begin{aligned}
v_{ANN} &= -\frac{\partial}{\partial \sigma^{2*}_{t(AGG)}} \gamma_{3\alpha}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) = -\gamma_{3\alpha}^{'\sigma^{2*}_{t(AGG)}}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right) \\
&= f_\alpha^{'\sigma^{2*}_{t(AGG)}}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\begin{bmatrix} Max\left\{f^{'\sigma^{2*}_{t(AGG)}}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \\ -\,Min\left\{f^{'\sigma^{2*}_{t(AGG)}}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\} \end{bmatrix} \\
&\quad + Min\left\{f^{'\sigma^{2*}_{t(AGG)}}\left(\boldsymbol{\omega_3}, X, S_t, \tau, \sigma^{2*}_{t(AGG)}\right)\right\}
\end{aligned}
\tag{29}
$$

3. Results and Discussion

In validating the developed ANN model, the study used intraday data for the AAPL stock option for the period between December 2016 and March 2017 with 56,238 data points. Of these options values, 59.15% were in the money (ITM) while 40.85% were out of the money (OTM). The data was divided into three sets with 50% (28,119) used for training the model, 25% (14,160) used in testing and the remaining 25% (14,059) used in the validation of the model. Basic preliminary analysis on the AAPL stock price showed a stock in which prices were generally characterized by an increasing trend over the 82 trading days between December 2016 and March 2017 with the prices ranging from slightly above USD 100 to USD 140 as shown in Figure 1.

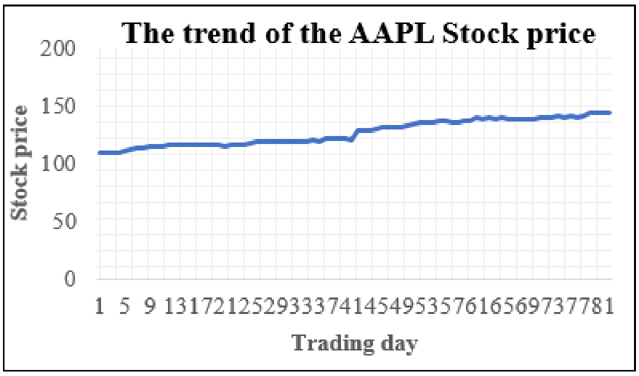
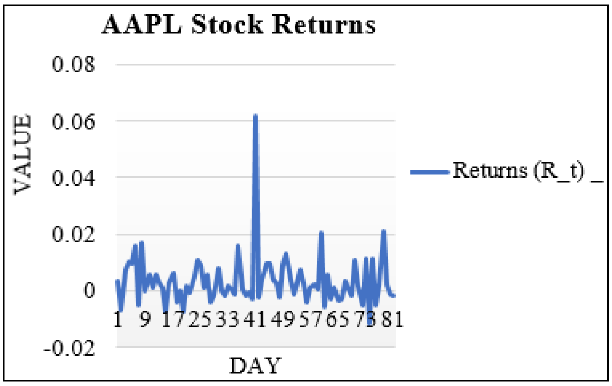


Figure 1: The trend of the AAPL Stock Price

Important in calculating option prices is volatility. This is defined in terms of the returns of the underlying asset, thus prompting an analysis of the AAPL stock returns. A basic analysis on the AAPL stock returns revealed averagely constant variation/movement in the stock prices except for around the 40th trading day where a sharp rise is experienced as can clearly be seen in figure 2 which is a graph of the AAPL stock squared returns. The squared returns are mainly confined to between 0 to 0.0005 with a single outlier of about 0.004. This is further confirmed by the calculated 10-day historical volatility which exhibits a fairly constant trend as shown in figure 2.



Paper ID: SR25608164927 DOI: https://dx.doi.org/10.21275/SR25608164927

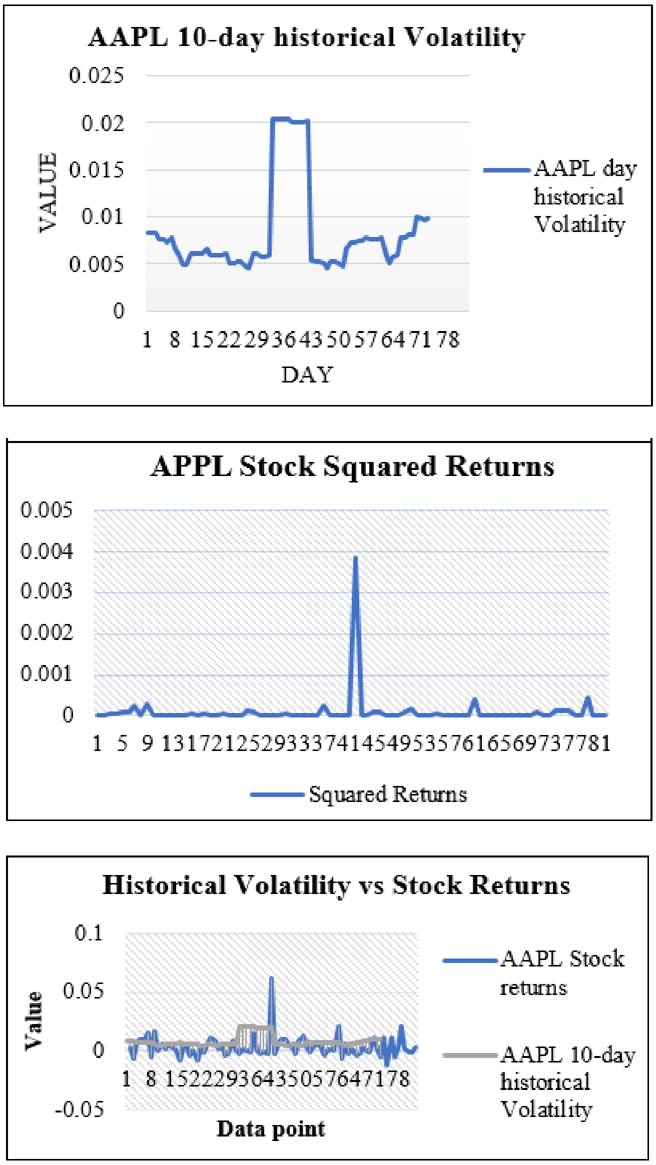


Figure 2-5: AAPL Stock Returns and Volatility

Sensitivity analysis

The delta of a call option is the slope of the option price curve at a point corresponding to a price of the underlying asset. In this study, since option pricing was done using the ANN model, the delta corresponds to the slope of the ANN model outputs at various AAPL stock prices. Figure 6 shows the relationship between our call option prices and the AAPL stock prices.

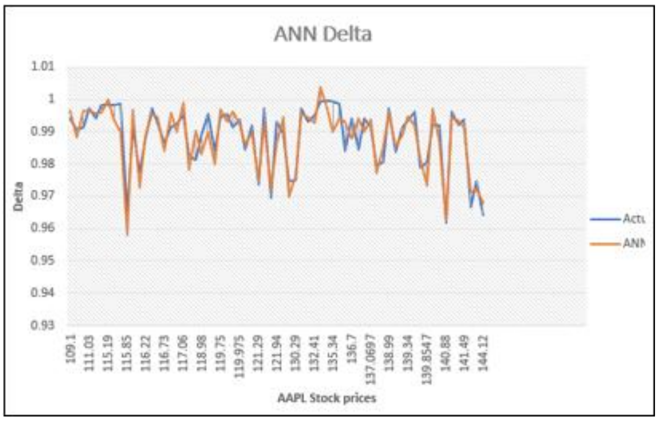


Figure 6: ANN Delta

By computing the delta ratio, a financial institution that sells Options to a client can make a delta neutral position to hedge the risk of changes of the underlying asset price. In our figure 6 for instance, suppose we consider the AAPL stock price at $100, the call option price at this point is $10, and the delta of the call option is 0.4. This implies that the financial institution involved would vend 10 call option to its client, so that the client has right to buy 1,000 shares at the maturity time. The implication of this is that: to construct a delta hedge position, the institution should buy 0.4 x 1,000 = 400 shares of the AAPL stock. If the AAPL stock price goes up to $1, the option price will go up by $0.4. Under this circumstances, the financial institution has a $400 ($1 x 400 shares) gain in its AAPL stock position, and a $400 ($0.4 x 1,000 shares) loss in its option position. The total payoff of the financial institution is zero. On the other hand, if the stock price goes down by $1, the option price will go down by $0.4. The total payoff of the financial institution is also zero.

It is worth noting that the relationship between option prices and stock prices is not always linear hence the delta fluctuations over different AAPL stock price. Any investor interested in maintaining a delta neutral portfolio ought to adjust his hedged ratio periodically. The more frequent the adjustment are done the better the delta-hedging. Delta measure can be combined with other risk measures to yield better risk mitigation measures.

When it comes to theta, it is also the slope of the option price curve at a point corresponding to a change in time. In this study, since option pricing was done using the ANN model, the theta corresponds to the slope of the ANN model outputs at various points in time. Figure 7 shows the relationship between our call option prices and time.

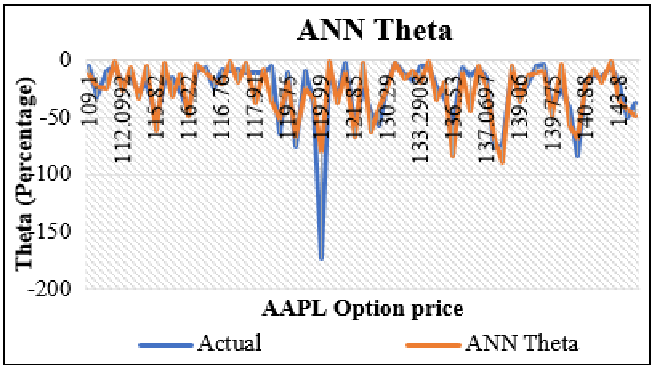


Figure 7: Hybrid ANN Theta

It is however worth noting that the value of an option is the combination of time value and stock value, and when time passes, the time value of the option decreases. Thus, the rate of change of the option price with respect to the passage of time, theta, is usually negative. Since the passage of time on an option is not uncertain, we do not need to make a theta hedge portfolio against the effect of the passage of time. However, we still regard theta as a useful parameter, because it is a proxy of gamma in the delta neutral portfolio.

Paper ID: SR25608164927 DOI: https://dx.doi.org/10.21275/SR25608164927

ISSN: 2319-7064
Impact Factor 2024: 7.101

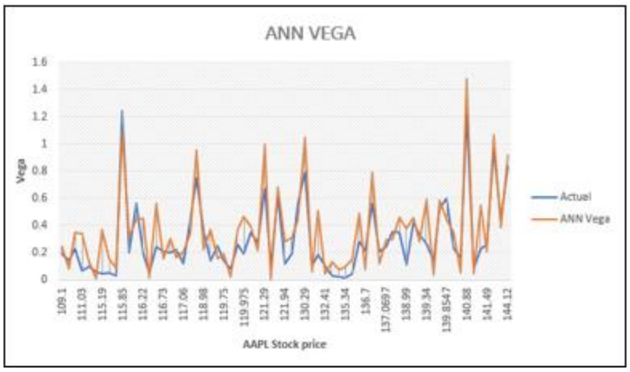


Figure 8: Hybrid ANN Vega

Suppose a delta-neutral and gamma-neutral portfolio has a vega equal to and the vega of a particular option is similar to gamma, we can add a position of in option to make a vega-neutral portfolio. To maintain delta-neutral, we should change the underlying asset position. However, when we change the option position, the new portfolio is not gamma-neutral. Generally, a portfolio with one option cannot maintain its gamma-neutral and vega-neutral at the same time. If we want a portfolio to be both gamma-neutral and vega-neutral, we should include at least two kind of option on the same underlying asset in our portfolio.

4. Conclusion

Sensitivity analysis on ANN not only aides in demonstrating how the indices can be computed from the ANN models, but also plays a significant role in the validation of the model. All the indices developed were within their desired range of values and this, being a resultant of the derivatives of the developed hybrid ANN output with respect to its various inputs, demonstrated how reliable the model was.

References

[1] Fogarasi, N. (2004). Option Pricing using Neural Networks. Technical Report.

[2] Hutchnison, 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.

[3] Merton, R.C. (1990). Theory of rational option pricing. *Bell J. Econ. Manage. Sci.*, 4: 141-183. DOI: 10.2307/3003143

[4] Mostafa, F., & Dillon, T. (2008). A neural network approach to option pricing. *Computational Finance and its Applications* III (71-84)

[5] Hajizadeha, E., and Seifia, A. (2011). A hybrid modeling approach for option pricing. *Researchgate*. DOI: 10.1063/1.3663498

[6] Mitra S.K. (2012). An Option Pricing Model That Combines Neural Network Approach and Black Scholes Formula. *Global Journals of Computer Science and Technology*. Volume 12 Issue 4.

[7] Andreou, P. C., Charalambous, C., & Martzoukos, H. S. (2006). Robust artificial neural networks for pricing of European options. *Computational Economics* 27, 329-351

[8] Enke, D., & Dagli, C.H. (2007). A hybrid option pricing model using a neural network for estimating volatility. *International Journal of General Systems*. DOI: 10.1080/03081070701210303

[9] Enke, D., & Monfared, S.A. (2014). Volatility Forecasting using a Hybrid GJR-GARCH Neural Network Model. *Procedia Computer Science 36* (2014) 246 – 253

[10] Malliaris, M., & Salchenberg, L. (1993). A neural network model for estimating option prices. *Journal of applied intelligence* 3, 193-206 (1993)

[11] Amilon, H. (2003). A Neural Network Versus Black-Scholes: A Comparison of Pricing and Hedging Performances. *Journal of Forecasting,* 2003, 22, 317–335.

[12] Gradojevic, N., Ramazan, G., & Kukolj, D. (2007). Option Pricing with Modular Neural Networks. *IEEE Transactions on Neural Networks* 20(4):626-37