A Methodology for Aiding Investment Decision between Assets in Stock Markets Using Artificial Neural Network

PN Kumar[1], Rahul Seshadri.G[2], Hariharan.A[3], VP Mohandas[4] , P Balasubramanian[5]

[1,2,3]Dept. of CSE , [4]Dept. of ECE [5]Amrita School of Business
Amrita Vishwa Vidyapeetham,
Ettimadai, Coimbatore, Tamil Nadu, 641 105, India

Abstract

This paper outlines a methodology for aiding the decision making process for investment between two financial market assets (eg a risky asset versus a risk-free asset), using neural network architecture. A Feed Forward Neural Network (FFNN) and a Radial Basis Function (RBF) Network have been evaluated. The model is employed for arriving at a decision as to where to invest in the next time step, given data from the current time step. The time step could be chosen on daily/weekly/monthly basis, based on the investment requirement. In this study, the FFNN has yielded good results over RBF. Consequently the FFNN developed enable us make a decision on investment in the next time step between a risky asset (eg the BSE Sensex itself or a single share) versus a risk-free asset (eg Securities like Govt Bonds, Public Provident Funds etc).The FFNN is trained with a set of data which helps in understanding the market behaviour. The input parameters or the information set consisting of six items is arrived at by considering important empirical features acting on real markets. These are designed to allow both passive and active, fundamental and technical trading strategies, and combinations of these. Using just six items simplifies the decision making process by extracting potentially useful information from the large quantity of historic data. The prediction made by the FFNN model has been validated from the actual market data. This model can be further extended to choose between any two categories of assets whose historical data is available.

Keywords: *financial forecasting, risky assets, risk free assets, Feed Forward Neural Networks, Radial Basis Function Networks*.

1. Introduction

Artificial Neural Networks (ANN) have earned themselves a unique position as non-linear approximators. In general, of all the AI techniques available, ANN deal best with uncertainty[1]. Like other forms of soft computing, ANN perform well in noisy data environments and has proved to exhibit a high tolerance to imprecision. These characteristics of ANN make them particularly suited to the arena of financial trading. The stock market represents a data source with an abundance of uncertainty and noise.

While ANN have been extensively studied [2,3] to perform a predictive analysis of the security prices, it may not be possible to model one general network that will fit every market and every security, hence models are built specific to markets and asset classes. Risky assets are those which do not have a guaranteed rate of return. An example of risky asset is stocks. Risk-free assets are those which give a return at a constant rate for eg, securities like Govt bonds. Since there is always a particular amount of risk associated with a risky asset, an investor cannot be assured of making a profit. This creates a need for a choice between risky and risk-free assets in order to maximize the profits, while minimizing the risk at the same time. The historical market prices would provide us with a better idea about the market's behavior. This data is incorporated into a Feed-Forward Neural Network in order to predict the behavior of the market in the future. These have been modeled on the Bombay Stock Exchange (BSE) and the results are presented here.

2. Investment Decision : Risky Vs Risk Free Asset

2.1 Artificial Neural Network.

An Artificial Neural Network (ANN) is a mathematical model or computational model that attempts to simulate the structure and/or functional aspects of biological neural networks. There are different types of Neural Networks. There are no standard rules available for determining the appropriate number of hidden layers and hidden neurons per layer. Smaller number of hidden nodes and hidden layers would render better generalization. A pyramid topology, which can be used to infer approximate numbers of hidden layers and hidden neurons has been suggested by Shih [4]. Azoff [5] suggests that a network with one hidden layer and 2N + 1 hidden neurons is sufficient for N inputs, and states that the optimum number of hidden neurons and hidden layers is highly problem dependant. Gately[6] suggests setting the number of hidden nodes to be equal to the total of the number of inputs and outputs. Some researchers suggest training a great number of ANN with different configurations, and then select that configuration that performed best- Kim at al.[7] Finally, another reasonably popular method is used by some researchers such as Kim& Lee[8] and Versace et al[9], whereby genetic algorithms are used to select between possible networks given choices such as network type, architecture, activation functions, input selection and preprocessing. Another method Tan[1], starts with a small number of hidden neurons and increase the number of hidden neurons gradually. A detailed comparative study can be seen at Vanstone[10]. For the purpose of this study, modeling has been attempted using a Feed-Forward Neural Network (FFNN) and a Radial Basis Function Network (RBF).

2.2. Modeling BSE Sensex

BSE Sensex, the most popular Indian stock index has been chosen for the study. The time step considered here is one month. BSE Sensex Index data pertaining to trading months starting from the year 2003 to 2008, is used for training the network. The duration is long enough and covers adequate market fluctuation. The one year monthly closing values of the Sensex for the year 2009 are used as the validation data set.

2.3. Model Architecture of FFNN

The functional form used is a FFNN (see Fig 1) with a single hidden unit with restricted inputs giving an output LeBaron[11,12,13]. The output is a simple function $\alpha(z_t, w_j)$. The equations given below define the network,

$$h_k = g_1 (w_{0,k} z_{t,k} + w_{1,k}) \quad (1)$$

$$\alpha(z_t) = g_2 (w_2 + \sum_{k=1}^{6} w_{3,k} h_k) \quad (2)$$

$$g_1(x) = \tanh(x) \quad (3)$$

$$g_2(x) = ½(1+\tanh(x/2)) \quad (4)$$

where z_t is time t information and w_j are parameters. k takes values from 1 to 6 so that the weight array {w} consists of 19 parameters. The output from the intermediate neuron k is denoted h_k. The output from the network, α is a 0 or 1 which would suggest where to invest in the next time-step, a 0 indicating that risk-free asset would give higher returns for the particular time-step and a 1 indicating that investing in an Index Fund tracking the BSE would render higher returns.

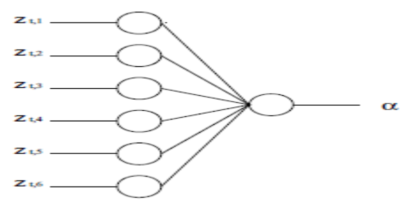

Fig. 1 FFNN: Decision between Risky Vs Risk Free Asset.

2.4 Input Values to the FFNN:The Information Set

The information set consists of six items. These are designed to allow both passive and active, fundamental and technical trading strategies, and combinations of these [11,12]. Using just six items simplifies the decision making process by extracting potentially useful information from the large quantity of historic data The first three inputs are the returns on equity in the previous three time-steps, useful for technical trading. The fourth is a measure of how the current price differs from the rational-expectations price. The last two inputs measure the ratio between the current price and exponentially weighted moving averages of the price. Information set being:

$$z_{t,1} = r_t = \log((p_t + d_t)/p_{t-1})$$
$$z_{t,2} = r_{t-1}$$
$$z_{t,3} = r_{t-2}$$
$$z_{t,4} = \log(r\, p_t / d_t)$$
$$z_{t,5} = \log(p_t / m_{1,t})$$
$$z_{t,6} = \log(p_t / m_{2,t})$$

Where p_t is the share price, d_t is the dividend paid, r is a constant and $m_{i,t}$ is the moving average given by

$$m_{i,t} = \rho_i\, m_{i,t-1} + (1- \rho_i)\, p_t \quad (5)$$

with $\rho_1 = 0.8$ and $\rho_2 = 0.99$.

2.5 Results: Training & Testing FFNN

The MATLAB Neural Network Toolbox has been chosen for creating, training and testing the network. The FFNN was trained with inputs from historical prices of BSE index, calculated taking monthly closing prices of BSE stock index from the year 2003 to 2008. The network was tested with data pertaining to the year 2009 and the results have been found to validate the market scenario. It has been found that the FFNN with one hidden layer with six neurons has produced quite accurate results. The network prediction matched with the test data of 2009 market. The neural network thus establishes the functional dependency between the input parameters and the market behavior.

2.6 Radial Basis Function (RBF) network

Radial Basis Function (RBF) network(Fig 2) is an artificial neural network that uses radial basis functions as activation functions. It is a linear combination of radial basis functions. They are used in function approximation, time series prediction, and control. RBF networks typically have three layers: an input layer, a hidden layer with a non-linear RBF activation function and a linear output layer. The output, $\varphi : \mathbb{R}^n \to \mathbb{R}$, of the network is thus

$$\varphi(\mathbf{x}) = \sum_{i=1}^{N} a_i \rho(||\mathbf{x} - \mathbf{c}_i||)$$

where N is the number of neurons in the hidden layer,

$\mathbf{c}_i$ is the center vector for neuron i, and

a_i are the weights of the linear output neuron.

In the basic form all inputs are connected to each hidden neuron. The norm is typically taken to be the Euclidean distance and the basis function is taken to be Gaussian.

$$\rho\big(\,\|\mathbf{x} - \mathbf{c}_i\|\,\big) = \exp\left[-\beta\,\|\mathbf{x} - \mathbf{c}_i\|^2\right]$$

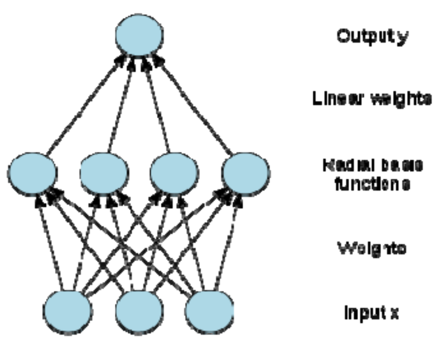

Fig 2 : Architecture of a radial basis function network.

An input vector x is used as input to all radial basis functions, each with different parameters. The output of the network is a linear combination of the outputs from radial basis functions. The Gaussian basis functions are local in the sense that

$$\lim_{||x||\to\infty} \rho(\|\mathbf{x} - \mathbf{c}_i\|) = 0$$

i.e. changing parameters of one neuron has only a small effect for input values that are far away from the center of that neuron. RBF networks are universal approximators on a compact subset of $\mathbb{R}^n$. This means that a RBF network with enough hidden neurons can approximate any continuous function with arbitrary precision. The weights ai, Ci and β are determined in a manner that optimizes the fit between φ and the data.

2.7 Results: Training & Testing: RBF Network

The MATLAB Neural Network Toolbox has been chosen for creating, training and testing the RBF network as well. The network was trained with inputs from historical prices of BSE index, as was done above, taking monthly closing prices during the period 2003 to 2008. The network was tested with data pertaining to the year 2009. However, the RBF network did not yield good results. This perhaps is attributable to the fact that the data presented to the network was close to each other, thereby resulting in improper clustering and inaccurate results.

3. Future Extension

The model presented above can be extended to choose between any two risky assets. This strategy can be effectively employed to compare between two mutual funds, two index funds or to arrive at an investment decision between two stock exchanges even. Further studies can explore the strategy for comparison of more than two risky assets. Whereas in FFNN(in the case of Risky Vs Risk-free asset), the model only suggests a choice of where to invest, a possible extension might be to find out the proportion of wealth to be invested in Risky and Risk-free assets respectively.

4. Conclusion

Two neural network models, FFNN and RBF have been designed, tested and validated from the BSE data to enable an investor make a decision at different time steps. FFNN with one hidden layer with six neurons has produced quite accurate results. However, the analysis carried out on a RBF network did not yield good results. The model of FFNN pertains to arriving at a decision between investment in a risky and a risk-free asset. Hence the models suggested by this paper can be used as a tool for an informed investment decision in the share markets. It is hoped that this can bring about a better investment strategy and help in achieving greater profits to investors.

References

[1].Tan, C.N.W., Artificial Neural Networks: Applications in Financial Distress Prediction and Foreign Exchange Trading (Gold Coast,QLD: Wilberto Press, 2001).
[2].White, H., Economic Prediction using Neural Networks: The case of IBM Daily Stock Returns. Second Annual IEEE Conference on Neural Networks 1988, 451-458.
[3].Oppenheimer, H.R. and G.G. Schlarbaum, Investing with Ben Graham: An Ex Ante Test of the Efficient Markets

Hypothesis, Journal of Financial and Quantitative Analysis, XVI(3), 1981, 341-360.

[4].Shih, Y., Neuralyst Users Guide: Cheshire Engineering Corporation, 1994.

[5].Azoff, M.E., Neural Network Time Series Forecasting of Financial Markets (Chichester: Wiley, 1994).

[6].Gately, E., Neural Networks for Financial Forecasting (New York: Wiley, 1996).

[7].Kim, J.-H., et al., Stock Price Prediction using Back propagation Neural Network in KOSPI. International

[8].Kim, K.-J. and W.B. Lee, Stock market prediction using artificial neural networks with optimal feature transformation, Neural Computing and Applications, 13(3), 2004, 255-260.

[9].Versace, M., et al., Predicting the exchange traded fund DIA with a combination of genetic algorithms and neural networks, Expert Systems with Applications, 27(3), 2005, 417-425.

[10].Vanstone Bruce and Gavin Finnie, An empirical methodology for developing stock market trading systems using artificial neural networks, Expert Systems with Applications: An International Journal Volume 36 , Issue 3 (April 2009)

[11].B. LeBaron. Empirical regularities from interacting long- and short-memory investors in an agent based stock market. IEEE Transactions on Evolutionary Computation, 5(5):442–455, October 2001.

[12].B. LeBaron. Evolution and time horizons in an agent based stock market. Macroeconomic Dynamics,

[13.B. LeBaron. A builder's guide to agent based financial markets. Quantitative Finance, 1(2):254–261,2001.

[14].Harry Markowitz, "Portfolio Selection", Journal of Finance, March 1952, pp. 77-91.

Prof PN Kumar[1], MTech(IITK),FIE,FIETE is Chairman CSE Dept, Amrita School of Engineering, Ettimadai (Amrita Vishwa Vidyapeetham). He is a visiting professor at Amrita School of Business and is an adjunct faculty of State University of New York, Buffalo. Interest areas are Agent Based Modeling, Soft Computing and Software Project Management.

Rahul Seshadri.G[2] is a Graduate in Computer Science & Engg from Amrita School of Engineering, Ettimadai (Amrita Vishwa Vidyapeetham). Interest areas are Stock Market Prediction and Soft Computing. Presently he is pursuing his MBA.

Hariharan.A[3]. is a Graduate in Computer Science & Engg from Amrita School of Engineering, Ettimadai (Amrita Vishwa Vidyapeetham). Interest areas are Stock Market Prediction and Software Engineering. Presently he is a Software Developer in Cognizant Technologies at Coimbatore.

Dr VP Mohandas[4], , PhD(IITB),FIETE is Chairman ECE Dept, Amrita School of Engineering, Ettimadai (Amrita Vishwa Vidyapeetham). His areas of interests include Dynamic System Theory, Signal Processing, Soft Computing and their application to socio-techno-economic and financial systems.

Dr P Balasubramanian[5],Ph.D (JNU) is Associate Professor, Finance at Amrita School of Business. He has thirteen years of experience in teaching and research and two years of industrial experience. Interest areas are Micro Economics, Macro Economics, Options, Futures and other Derivatives.