International Engineering Journal For Research & Development

Vol.3
IOCARDET

E-ISSN NO:-2349-0721

Impact factor : 2.25

PREDICTION OF STOCK VALUES IN STOCK MARKET USING ARTIFICIAL NEURAL NETWORK: A REVIEW

[1]Mr.D.A.Mahalle, Prof.V.N.Nikam, Ms. Kalyani Gawande
mahalledatta@gmail.com[1],virendranikam@gmail.com[2]
M.E. Scholar Dr.SAU. KGIET, Darapur, Assistant professor Dr.SAU. KGIET, Darapur

ABSTRACT—

This paper proposes a novel constraint bagging forecasting method for stock price prediction. In the proposed approach, each of predictors is firstly constructed by training on a set of samples produced by bootstrapping using neural networks. The goal of this project is thus to experiment with ANNs and to evaluate performance of ANN models in studying stock price patterns in time by attempting to predict future results of a time-series by simply studying patterns in the time series of stock prices. In this paper, two kinds of neural networks, a feed forward multilayer Perceptron (MLP) and an Elman recurrent network, are used to predict a company's stock value based on its stock share value history. The experimental results show that the application of MLP neural network is more promising in predicting stock value changes rather than Elman recurrent network and linear regression method. However, based on the standard measures that will be presented in the paper we find that the Elman recurrent network and linear regression can predict the direction of the changes of the stock value better than the MLP.

Index Terms— Stock Market; Genetic Algorithm; Artificial Neural Network (ANN); Prediction; Forecasting; Data; Autoregressive (AR)

INTRODUCTION

It is nowadays a common notion that vast amounts of capital are traded through the Stock Markets all around the world. National economies are strongly linked and heavily influenced by the performance of their Stock Markets [1]. The characteristic that all Stock Markets have in common is the uncertainty, which is related with their short and long term future state. This feature is undesirable for the investor but it is also unavoidable whenever the Stock Market is selected as the investment tool. The best that one can do is to try to reduce this uncertainty. Stock Market Prediction (or Forecasting) is one instrument in this process [3]. The Stock Market prediction task divides researchers and academics into two groups those who believe that we can devise mechanisms to predict the market and those who believe that the market is efficient and whenever new information comes up the market absorbs it by correcting itself, thus there is no space for prediction [4]. Furthermore they believe that the Stock Market follows a Random Walk, which infers that the best prediction you can have about tomorrow's value is today's value.

LITERATURE REVIEW

When individuals or organizations lack sufficient knowledge or information to enable them to plan future activities, analyses of expectations or forecasting are frequently involved. When it comes to financial matters, investors must develop plans in the face of several uncertainty factors, so that accuracy in forecasting becomes a very important issue. This chapter reviews relevant literature in the financial forecasting area and classifies it according to the forecasting techniques used: genetic algorithms, neural networks and the combination of genetic algorithms and neural networks. Related information about the SET, the application domain of this study, is examined [5].

STOCK

Encyclopedia of Britannica Online (2006) defines the term Stock as: The subscribed capital of a corporation or limited-liability company usually divided into shares and represented by transferable certificates. The certificates may detail the contractual relationship between the company and its stockholders, or shareholders, and set forth the division of the risk, income, and control of the business. A stock owner owns a part of the company and by buying or selling the shares; the rate of ownership might be increased or decreased. If the decision making power decides to distribute a portion of the company's profits rather than reusing them, an owner is paid a profit share, called dividend [10].

STOCK PRICING

A stock is priced at a static price when issued; then, it may be traded at any rate. For publicly traded companies, the stocks are traded in the stock market, where prices are determined by the supply/demand equilibrium. As stated by Brown (2002), a company that misses profit estimations can experience severe stock price drops due to the fear of investors

STOCK PRICING

Predicting stock has become more sophisticated since Wuthrich Etal (1998) presented a study on daily stock market forecasting through the use of textual web data. They used a variety of sources of financial information such as The Wall Street Journal, The Financial Times, Reuters, Dow Jones and Bloomberg. These textual information sources contain news, financial analysis reports, and information about the situation in the world's stock, currency and bond markets. This textual information was weighted for use of specific keywords, the weights being used to generate probabilistic rules for a prediction model [6]. These authors predicted five stock market indices but the results were inaccurate. Other researchers have also used qualitative data for forecasting market trends. Peramunetilleke and Wong (2002) forecast intra-day currency exchange rate movements by weighting keywords in money market news headlines. Their prediction rules, when applied to the weighted data, produced a better outcome than random guessing [8]. However, the data processing was challenging as not all sources were reliable and the meaning of text sections, containing the same keywords, may have differences. The authors suggested that their technique might be incorporated into other numeric time series analyses to improve the accuracy of predictions [3].

NEURAL NETWORK

Neural networks are computer programs consisting of computing nodes and interconnections between nodes (Yao et al., 1999). They are recognized as effective tools for financial forecasting (Yao Tan, 2001a) and can learn? From experience as do humans, cope with non-linear data, and deal with partially understood application domains, such as stock market behaviors. Moreover, the fundamental stock market indicators, gross domestic product, interest rate, gold prices and exchange rates and technical indicators, including closing prices, opening prices, highest prices and lowest prices, can be incorporated into neural networks to help improve predictive outputs (Yao et al., 1999).

BP METHOD

The most common method used for training neural networks is back propagation, first introduced by John Hopfield (Berry Linoff, 1997, p.303). The back propagation concept generally follows three steps: [3]

- The network gets a training example and, using the existing weights in the network, it calculates the output or outputs for the example.
- In the Back-propagation algorithm, the errors is then calculated by taking the difference between the calculated result and the expected result (actual result).
- The error is fed back through the network and the weights adjusted to minimize the error (Berry Linoff, 1997, p.304). Neural networks with back-propagation method or back-propagation neural networks provide reasonable speed (Franklin, 2003), are straightforward (Yusof, 2005) and tend to deliver reasonable outputs or results for unseen data (MathWorks,2009). Many forecasting investigations used neural networks which included back-propagation methods. Examples include Schwaerzel and Rosen (1997), Tkacz (2001), Jaruszewicz and Mandziuk (2004), Luu and Kennedy (2006), Mjalli et al. (2006), Abdelmouez et al. (2007) and Chaigusin et al. (2008b). However the main drawback of back-propagation neural networks is the likelihood of being trap in local optima (Berry Linoff, 1997, p.305).

artificial neural network

The artificial neural network is simplified models of the biological neuron system, is a massively parallel distributed processing system made up of highly interconnected neural computing elements that can learn and thereby acquire knowledge and make it available for use. The artificial neural network learns by example. They can therefore be trained with known examples of a problem to acquire knowledge about it. Once appropriately trained, the network can be put to effective use in solving unknown or untrained instances of the problem [11].

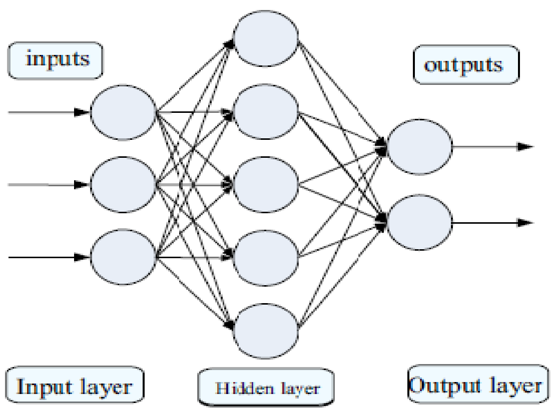

Figure 1: Artificial Neural network with layers.

NEURON

A neuron is a processing unit that takes several inputs and gives a distinct output. The **Figure 1** below depicts a single neuron with R inputs $p1$, $p2$,…, pR, each input is weighted with a value $w11$, $wl2$,…, wlR and the output of the neuron an equal to f ($w11 \cdot p1 + w12 \cdot p2 + \ldots + w1R \cdot pR$) [8].

Each neuron apart from the number of its input is characterized by the function f known as a transfer function. The most commonly used transfer functions are the hard-limit, the pure linear, the sigmoid and the tansigmoid function. The preferences on these functions derive from their characteristics

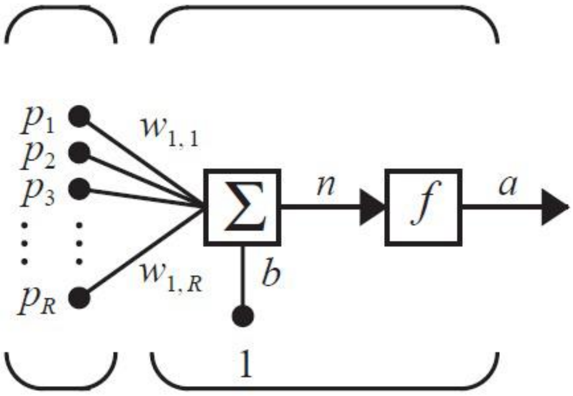

Figure 2: A simple neuron with R input.

LAYER

Figure 3 presents the Artificial Neural network is de- fined as data processing system consisting of many of simple highly interconnected processing elements (artificial neurons) is an architecture inspired by the structure of the cerebral cortex of the brain. Each network has got exactly one input layer, zero or more hidden layers and one output layer. All of them apart from the input layer consist of neuron. The number of inputs to the Artificial Neural Networks equal to the dimension of our input samples **Figure 3** shows, while the number of the outputs we want from the Artificial Neural Networks define the number of neurons in the output layer [12].

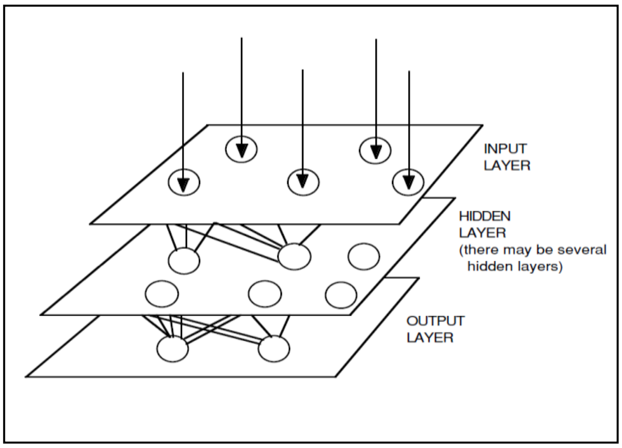

Figure 3: Neural Network Diagram

In our case the output layer will have exactly one neuron since the only output we want from the network is the prediction of tomorrow's excess return.

The mass of hidden layers as well as the mass of neurons in each hidden layer is proportional to the ability of the network to approximate more complicated functions. Of course this does not infer by any means that networks with complicated structures will always perform better. The reason for this is that the more complicated a net- work is the more sensitive it becomes to noise or else, it is easier to learn apart from the underlying function the noise that exists in the input data. Therefore clearly there is a tradeoff between the representational power of a net- work and the noise it will incorporate.

GENERIC ALGORITHM

Genetic Algorithm proposed in 1975 by Johan Holland. Genetic Algorithms are computerized search and optimization algorithms based on the mechanics of natural genetics and natural selection. Genetic algorithms per- formed directed random searches through a given set of alternative with the aim of finding the best alternative with respect to the given criteria of goodness as illustrated in **Figure 3 [1]**. The criteria are required to be ex- pressed in terms of an objective function which is usually called a fitness function. Fitness is defined as a figure of merit, which is to be either maximized or minimized [3].

The biological organism to be specified is defined by one or by a set of chromosomes. The overall set of chromosomes is called genotype, and the resulting organism is called the phenotype. Every chromosome consists of genes. The gene position within the chromosome refers to the type of organism characteristic, and the coded content of each gene refers to an attribute within the organism type. In Genetic Algorithms terminology, the set of strings (chromosomes) forms a structure (genotype). Each string consists of characters (genes). Genetic Algorithms are a method for moving from one population of "chromosomes" [13] (strings of 1, 0 bits) to a new population by using a kind of "natural selection" together with the genetics inspired operators of crossover, mutation, and inversion. Each chromosome consists of "genes" (bits), each gene being an instance of a particular "allele" (0 or 1). The selection operator chooses those chromosomes in the population that will be allowed to reproduce, and on average the fitter chromosomes produce more offspring than the less fit ones. Crossover exchanges sub- parts of two chromosomes, roughly mimicking biological recombination among two single chromosomes (haploid) organisms; mutation randomly changes the allele values of some locations in the chromosome; and the inversion reverses the order of a contiguous section of the chromosome, thus rearranging the order in which genes are arrayed [4].

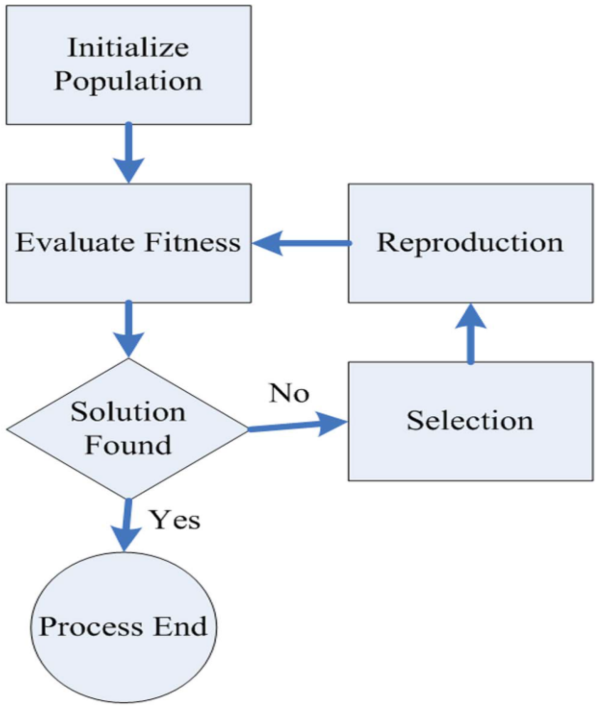

Figure 4: Flow chart of Genetic algorithm.

PROPOSED WORK

One of the primary objectives of this project is to also provide financial analysts with a tool with which they can predict values of stocks. The ability of the network to keep updating it with historical information will allow it to predict the data even more accurately and thus aim to increase the profitability for the investors and the users of this application. Some aspects that we aim at achieving is this regard are to:

- To aid the users of the software to make valuable decisions.
- Help the investors to generate and maximize their profits out of the investments they make in the stock market.
- Reduce the risk involved when investing in stocks.
- Support the users in an efficient portfolio management.

PROPAGATION ALGORITHM

Most people would consider the Back Propagation network to be the quintessential Neural Net. Actually, Back Propagation is the training or learning algorithm rather than the network itself. The network used is generally of the simple type shown in figure 5, in chapter 1 and in the examples up until now. These are called *Feed-Forward* Networks (we'll see why in chapter 7 on Hopfield Networks) or occasionally *Multi-Layer Perceptron's (MLPs)* [7].

The network operates in exactly the same way as the others we've seen (if you need to remind yourself, look at worked example 1). Now, let's consider what Back Propagation is and how to use it.

A Back Propagation network learns by example. You give the algorithm examples of what you want the network to do and it changes the network's weights so that, when training is finished, it will give you the required output for a particular input. Back Propagation networks are ideal for simple Pattern Recognition and Mapping Tasks. As just mentioned, to train the network you need to give it examples of what you want the output you want (called the *Target*) for a particular input as shown in Figure 5.

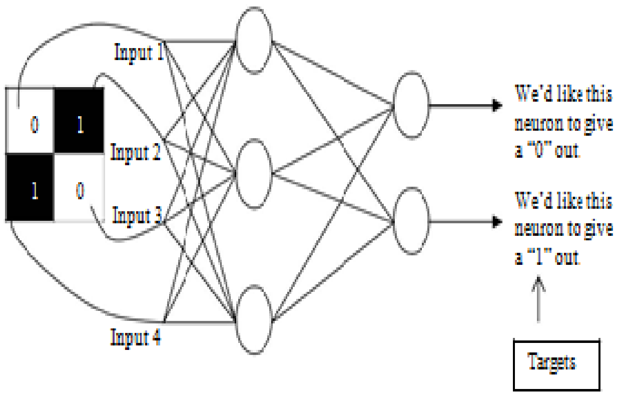

Figure 5: Input and target for second pattern

1. Algorithm: First apply the inputs to the network and work out the output – remember this initial output could be anything, as the initial weights were random numbers.
2. Next work out the error for neuron B. The error is *What you want – What you actually get*, in other words:

$Error_B = Output_B\ (1\text{-}Output_B)\ (Target_B – Output_B)$

The "*Output (1-Output)*" term is necessary in the equation because of the Sigmoid Function – if we were only using a threshold neuron it would just be *(Target – Output)*.

3. Change the weight. Let W^+_{AB} be the new (trained) weight and W_{AB} be the initial weight.

$W^+_{AB} = W_{AB} + (Error_B \times Output_A)$

Notice that it is the output of the connecting neuron (neuron A) we use (not B). We update all the weights in the output layer in this way.

4. Calculate the Errors for the hidden layer neurons. Unlike the output layer we can't calculate these directly (because we don't have a Target), so we *Back Propagate* them from the output layer (hence the name of the algorithm). This is done by taking the Errors from the output neurons and running them back through the weights to get the hidden layer errors. For example if neuron A is connected as shown to B and C then we take the errors from B and C to generate an error for A.

$Error_A = Output_A\ (1 - Output_A)\ (Error_B\ W_{AB} + Error_C\ W_{AC})$

Again, the factor "*Output (1 - Output)*" is present because of the sigmoid squashing function.

5. Having obtained the Error for the hidden layer neurons now proceed as in stage 3 to change the hidden layer weights. By repeating this method we can train a network of any number of layers.

DESIRED IMPLICATIONS

This study makes contributions in terms of the application of artificial intelligence techniques for SET prediction: first, the design of appropriate neural network architectures for the SET; second, the development of an ensemble system of NNs for prediction of the movements of the SET index and third, the list of interrelated indicators associated with the unique characteristics of the SET that can be used for its prediction. This study extends the knowledge about appropriate neural network configurations for capturing information associated with predicting the SET [7]. It investigated the use of different neural network architectures for predicting the SET, resulting in an increased knowledge about neural network configurations specifically addressing the characteristics of the SET. The outcomes of this study provide a guide to addressing issues such as the number of hidden layers and the number of hidden nodes as well as the activation function that can be used [11]. Work in this study is important to other researchers working with the SET as only rules of thumb exist in the literature for determining suitable neural network configurations. The study also investigated the use of two training methods for NNs, back-propagation and genetic algorithms, and found that there is no significance differences in the performances of NNs trained via either methods. Given that training via genetic algorithm is a much slower process that training via back-propagation, it may be appropriate that training via back-propagation be a preferred option in future investigations [4]. In contrast to most existing work in predicting the SET using single neural networks, this study also developed a gating network to be used as an ensemble mechanism which combines the results of the three best neural networks for predicting the movement of the SET index.

The gating network is composed of two layers with voting and dynamic gates in the first layer and an adaptive gate in the last layer. Experiments and analysis showed that the gating network is a better predictor of the directions of the movement of the SET index than any single neural network. The study provided insights as to the suitability of such gating system for predicting the movements of the SET index. The results and documented approach can be used as guidelines to future applications of neural networks for stock market prediction generally and the Thai stock market specifically [9].

CONCLUSION:

Use of ANN is a well-known technique for stock market forecasting. In this paper we have proposed optimal neural network architecture for stock market prediction based on empirical observations. The generalized form of this architecture can be written as $[m - m/2 - m/10 - 1]$. This architecture holds good for the financial stock of a company. We have acquired financial stock data of companies and expressed it temporally as a time series, upon which the analytics have been performed. The final system is able to compare the actual market value of a stock with the predicted value. The system further correctly predicts whether to buy or withhold the stock based on a mean based binary classification. So, this paper provides a measure to select the optimized parameters in terms of neurons and epochs to provide an accurate forecasting.

REFERENCES:

[1] Chen An-Sing, Leung Mark, Daouk Hazem, “Application of Neural Networks to an Emerging Financial Market: Forecasting and Trading the Taiwan Stock Index,” Computers & Operations Research, vol. 30, 2003, pp. 901–923.

[2] G. Armano, M. Marchesi, and A. Murru, “A Hybrid Genetic-Neural Architecture for Stock Indexes Forecasting,” Information Sciences, vol. 170, 2005, pp 3-33.

[3] O. B. Antwerpen, “6de Eeuwse Traditionele Bak-en Zan- dsteenarchitectuur,” 2010.

[4] B. G. Malkei, “A Random Walk Down Wall Street,” 7th Edition, W. W. Norton & Company, New York, 1999.

[5] T. Kimoto, K. Asakawa, and M. Yoda, “Trading s & p 500 stock index futures using a neural network,” in *Proceedings of the Annual International Conference on Artificial Intelligence Applications*, Wall Street, New York, June 1995, pp. 63–72.

[6] G. Zhang, B. Patuwo, and M. Hu, “Forecasting with artificial neural networks: the state of the art,” *International Journal of Economics and Finance*, vol. 4, pp. 139–151, March 2012.

[7] T. Kamijo and T. Tanigawa, “Stock price pattern recognition: a recurrent neural network approach,” in *Proceedings of the International Joint Conference on Neural Networks*, San Diego, CA, June 1990, pp. 215–221.

[8] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic Game Theory*. London: Cambridge University Press, 2007.

[9] Q. Cao, K. B. Leggio, and M. J. Schniederjans, "A comparison of Fama, French's model and artificial neural network in predicting the Chinese stock market," Computers & Operations Research, vol. 32, pp.2499-2512, 2005.

[10] A. P. N. Refenes, and W P Holt, "Forecasting volatility with neural regression: a contribution to model adequacy," IEEE Trans. Neural Networks, vol. 12, no. 4, pp. 850-864, July, 2001.

[11] Z. Liao, and J. Wang, "Forecasting model of global stock index by stochastic time effective neural network," Expert Systems with Applications, vol. 37, pp. 834-841, 2010.

[12] G. S. Atsalakis, and K. P. Valavanis, "Surveying stock market forecasting techniques - part II: soft computing methods," Expert Systems with Applications, vol. 36, pp. 5932-5941, 2009.