



Analysis of LEACH Algorithm in Wireless Sensor Network

Suman (M.Tech Scholar), **Ms. Savita Bishnoi** (Associate Professor)

Department of Computer Sciences & Engineering

jangra.suman13@gmail.com

Rohtak Institute of Engineering & Management

Abstract— Neural networks are good at classification, forecasting and recognition. They are also good candidates of financial forecasting tools. Forecasting is often used in the decision making process. Neural network training is an art. Trading based on neural network outputs, or trading strategy is also an art. We will discuss a seven-step neural network forecasting model building approach in this article. Pre and post data processing/analysis skills, data sampling, training criteria and model recommendation will also be covered in this article.

Keywords: Financial Trading, Neural Networks, Genetic Algorithms.

1. Introduction

Approaches to forecasting the future direction of share market prices fall broadly into two categories—those that rely on *technical analysis*, and those that rely on *fundamental analysis*. While technical analysis uses only historical data (past prices, volume of trading, volatility, etc.) to determine the movement in the price of some financial asset, fundamental analysis is based on external information; that is, information that comes from the economic system surrounding the market. Such information includes interest rates, prices and returns of other assets, and many other macro- or micro-economic variables. The use of technical analysis goes against the grain of conservative academic opinion, which regards this behaviour as irrational given the *efficient markets hypothesis* (Malkiel 1996). The efficient markets hypothesis asserts that the price of an asset reflects all of the information that can be obtained from past prices of the asset. The argument is that any opportunity for a profit will be exploited immediately, and hence disappear. That is, the market is so efficient that no one can buy or sell quickly enough to consistently benefit. A consequence of the efficient markets hypothesis is that stock prices follow a random walk and are unpredictable based on any amount of historical data. The most appropriate investment strategy is thus a buy-and-hold strategy. Despite the implications of the efficient markets hypothesis, many traders continue to make buy and sell decisions based on historical data. These decisions are made under the premise that patterns exist in that data, and that these patterns provide an indication of future movements. If such patterns exist, then it is possible in principle to apply automated pattern recognition techniques such as neural networks to the discovery of these patterns.

Several sources have reported on the simulation of trading agents based on Artificial Neural Networks (ANNs) (White 1988; Kimoto *et al* 1990; Yoon & Swales 1991; Weigend & Gershenfeld 1994). While the traditional approach to supervised neural network weight optimisation is the well-known backpropagation algorithm (Rumelhart & McClelland 1986), Beltratti,

Margarita and Terna (1996) report on the use of genetic search for neural network weight optimisation in this domain. One of the advantages of genetic search as a weight-optimisation technique is that it allows flexibility in the choice of criteria that can be used as an objective function to guide search through the space of weight configurations. Thus, rather than making buy/sell decisions on the basis of a numerical prediction of the next day's price, genetic weight optimization allows a trading regime to be discovered that optimises the financial *return* over some training period.

In this paper we describe the methodology by which neural networks can be trained indirectly, using a genetic algorithm based weight optimisation procedure, to determine buy and sell points for financial commodities traded on a stock exchange. In order to test the significance of the returns achieved using this methodology, we compare the returns on four financial time series with returns achieved on random walk data derived from each of these time series using a bootstrapping procedure.

The bootstrapped samples contain exactly the same distribution of daily returns as the original series, but lack any serial dependence present in the original. Our results indicate that on some price series the return achieved is significantly greater than that which can be achieved on the bootstrapped samples. This lends support to the claim that some financial time series are not entirely random, and that—contrary to the predictions of the efficient markets hypothesis—a trading strategy based solely on historical price data can be used to achieve returns better than those achieved using a buy-and-hold strategy.

2. Neural Networks for Automated Trading

One approach to developing neural network trading models is to first train the neural network to predict the value of the closing price of some asset one or more days into the future. An entry/exit (i.e. *buy* or *sell*) decision can then be made on the basis of this prediction. This section describes an alternative approach that does not attempt exact numeric prediction of the asset value, but rather, attempts to recognize patterns in the input data that can provide clues as to the optimal points to make buy or sell decisions.

The neural network buying and selling agent we use consists of an input layer, one hidden layer of sigmoidally activated units, and a single sigmoidally activated output that is thresholded such that output values above 0.5 are interpreted as a *buy* signal, and all other values are interpreted as a signal to *sell*. The inputs to the network are typically the price of the asset at the close of trade on the previous trading day, and variables derived from this.

These could include moving averages, various delayed inputs (price two days prior, etc.).¹ The network is shown schematically in Figure 1.

The buy and sell signals that are generated by the network, in conjunction with the particular trading strategy that is adopted, determines the trading position. The trading strategy that we adopt is a *one-point* buying and selling strategy. This means that all available capital is invested in shares, or all capital is invested in some low-risk fixed interest security. On the basis of the trading signal issued by the network, either the low-risk security is sold and shares are bought (*buy* signal), or vice-versa (*sell* signal). Note that shares can only be sold if the investor is currently 'in the market', and bought if the investor is "not in the market".

The most common approach to neural network weight optimisation is backpropagation training (Rumelhart & McClelland 1986). Backpropagation is a supervised training algorithm that relies on the availability of a set of labelled training data. However, direct (i.e. *supervised*) training of the network is not possible in this case, since we are not supplied with labelled training data. That is, we do not know *a priori* what are the optimal buy and sell

3. Towards a Better Robust Financial Forecasting Model

In working towards a more robust financial forecasting model, the following issues are worth examining.

First, instead of emphasizing on the forecasting accuracy only, other financial criteria should be considered. Current researchers tend to use goodness of fit or similar criteria to judge or train their models in financial domain. In terms of mathematical calculation this approach is a correct way in theory. As we understand that a perfect forecasting is impossible in reality. No model can achieve such an ideal goal.

Under this constraint, seeking a perfect forecasting is not our aim. We can only try to optimize our imperfect forecasts and use other yardsticks to give the most realistic measure.

Second, there should be adequate organization and processing of forecasting data. Preprocessing and proper sampling of input data can have impact on the forecasting performance. Choice of indicators as inputs through sensitivity analysis could help to eliminate redundant inputs. Furthermore, NN forecasting results should be used wisely and effectively. For example, as the forecast is not perfect, should we compare the NN output with the previous forecast or with the real data especially when price levels are used as the forecasting targets?

Third, a trading system should be used to decide on the best tool to use. NN is not the single tool that can be used for financial forecasting. We also cannot claim that it is the best forecasting tool. In fact, people are still not aware of which kind of time series is the most suitable for NN applications. To conduct post forecasting analysis will allow us to find out the suitability of models and series. We may then conclude that a certain kind of models should be used for a certain kind of time series. Training or

building NN models is a trial and error procedure. Some researchers are not willing to test more on their data set [14]. If there is a system that can help us to formalize these tedious exploratory procedures, it will certainly be of great value to financial forecasting.

Instead of just presenting one successful experiment, possibility or confidence level can be applied to the outputs. Data are partitioned into several sets to find out the particular knowledge of this time series. As stated by David Wolpert and William Macready about their No-Free-Lunch theorems [28], averaged over all problems, all search algorithms perform equally. Just experimenting on a single data set, a NN model which outperforms other models can be found. However, for another data set one model which outperforms NN model can also be found according to No-Free-Lunch theorems. To avoid such a case of one model outperforming others, we partition the data set into several sub data sets. The recommended NN models are those that outperform other models for all sub time horizons. In other words, only those models incorporated with enough local knowledge can be used for future forecasting.

It is very important and necessary to emphasize these three issues here. Different criteria exist for the academics and the industry. In academics, sometime people seek for the accuracy towards 100%. While in industry a guaranteed 60% accuracy is typically aimed for. In addition, profit is the eventual goal of practitioners, so a profit oriented forecasting model may fit their needs.

Cohen [5] surveyed 150 papers in the proceedings of the 8th National Conference on artificial intelligence. He discovered that only 42% of the papers reported that a program had run on more than one example; just 30% demonstrated performance in some way; a mere 21% framed hypotheses or made predictions. He then concluded that the methodologies used were incomplete with respect to the goals of designing and analyzing AI system.

Tichy [20] showed that in a very large study of over 400 research articles in computer science. Over 40% of the articles are about new designs and the models completely lack experimental data. In a recent IEEE computer journal, he also points out 16 excuses to avoid experimentation for computer scientists [21].

What he is talking is true and not a joke.

Experimental Design

As described in the introduction, a consequence of the efficient markets hypothesis is that price series follow a random walk, and hence any trading strategy based on timing or predicting the market will never consistently outperform a simple buy-and-hold strategy. However the trading strategy described above has been observed to outperform a buy-and-hold strategy on some financial price series (Skabar & Cloete 2001; Cloete and Skabar 2001). How might we determine whether the observed returns achieved by using the neural network trading agent to time buy and sell points based on historical data are real or anomalous?

One way of testing this is to compare the performance of the procedure on real data with performance on one or more sets of random walk data. If performance on the random data does not differ significantly with that on real data, then we cannot claim to have discovered any real *predictability*. We first address the problem of generating random walk data.

Hypothesis Testing

We are interested in determining whether the return achieved by applying the procedure of Section 2 to a real price series differs significantly to that achieved by applying it to the pseudo price series. Thus, the null hypothesis can be expressed as follows:

H₀: There is no significant difference between the return achieved when the procedure is applied to the real time series and the return achieved when the procedure is applied to the pseudo time series.

The corresponding alternative hypothesis is thus:

H₁: There is a significant difference between the return achieved when the procedure is applied to the real time series and the return achieved when the procedure is applied to the pseudo time series.

The null hypotheses can be tested by applying the procedure we have described in Section 2 to each of the pseudo time series that have been produced using the bootstrapping procedure. This will result in some empirical distribution of overall returns. The return on the *original* series can then be compared with this distribution of returns and a *p*-value obtained. The *p*-value simply provides the probability of observing a result as extreme, or more extreme, than that which would be expected if the null hypothesis were true; the smaller the *p*-value, the less likely the null hypothesis is true.⁴ Rejection of a null hypothesis would allow us to accept the alternative hypothesis that the return achieved on the original price series is significantly different to that which we would be expected if the series was random. And this, in turn, would imply that there is some serial dependence in the original time series (which is not present in the pseudo time series), thus providing support against the efficient markets hypothesis.

4. Discussion

As mentioned in Section 2.4, one of the main dangers in attempting to automate trading strategies is that of *datasnooping*.

We believe that our methodology is reasonably free from criticisms of *data-snooping* for the following reasons. Firstly, every experiment that we conducted—across all four indices—used exactly the same network structure, the same inputs, the same learning parameters, and the same training/test set samplings. The only difference was that between the actual time series. Secondly, by performing many trials on the original price series using different test set windows, we can be quite confident that the average returns we achieved are not anomalous.

A second criticism often directed at research which purports to have discovered a trading strategy that outperforms a buy-and-hold strategy is that the costs associated with trading have not been accounted for. Our experiments have been performed incorporating a trading cost of 0.1% per trade, which is currently the approximate commission for online trading. It is interesting to note the relatively low frequency of trading performed by the network, which ranges from a maximum of approximately 10 trades per year on the Dow Jones data to a minimum of approximately 5 trades per year on the Australian All Ordinaries. This trading frequency is significantly less than that of network traders based on *forecasting* numerical price movements.

What might be the cause of the differences in being able to successfully trade using these four indices. One possible explanation for this could be the fact that the Dow Jones and Australian All Ordinaries indices are *blue chip* indices. That is, they represent the averaged values of a large number of large, established, stable, and relatively secure companies. In contrast, the S&P500 and the NASDAQ include a significant proportion of *tech.* stocks, whose prices are known to have been much more volatile than blue chip companies, especially in recent years (recall the bursting of the technology stocks bubble). The inclusion of such volatile stocks in the makeup of these indices may make these series more chaotic, thus reducing the capacity to time trading decisions on past prices of these indices. However, this is highly speculative and experiments would need to be designed to test these ideas formally.

A rather obvious question that arises out of the results of this research is that if it is possible to exploit historical prices on the Dow Jones data to achieve a return better than a buy-and-hold strategy, how might we identify other series with this same property. That is, how might we determine *a priori* whether some given price series possesses such a desirable quality? One approach to this would be to perform the same experiments described in this paper over very many different price series and, on the basis of the results, assign each of these series some measure of what may cautiously be called 'predictability'. Patterns could then be sought between these so-called *predictability* values and measurable statistical properties of the price series (autocorrelation, Box-Pierce Q statistics, etc.). We leave this exploration for future work.

5. Conclusions

This paper has described a methodology by which neural networks can be trained indirectly, using a genetic algorithm based weight optimisation procedure, to determine buy and sell points for financial commodities traded on a stock exchange. In order to test the significance of the returns achieved using this methodology, the returns on four financial price series were compared with returns achieved on random walk data derived from each of these series using a bootstrapping procedure. These bootstrapped samples contain the same distribution of daily returns as the original series, but lack any serial dependence present in the original. The results indicate that on the Dow Jones Industrial Average Index, the return achieved over a four year out of sample period are significantly greater than that which would be expected had the price series been random. This lends support to the claim that some financial time series are not entirely random, and that—contrary to the predictions of the efficient markets hypothesis—a trading strategy based solely on historical price data can be used to achieve returns better than those achieved using a buy-and-hold strategy.