Time dependent Directional Profit Model for Financial Time Series Forecasting

Jingtao YAO
Dept of Information Systems
Massey University
Private Bag 11222
Palmerston North
New Zealand

Chew Lim TAN
Dept of Computer Science
National University of Singapore
1 Science Drive 2
Singapore 117543
Singapore
tancl@comp.nus.edu.sg

Abstract

Goodness-of-fit is the most popular criterion for neural network time series forecasting. In the context of financial time series forecasting, we are not only concerned at how good the forecasts fit their targets, but we are more interested in profits. In order to increase the forecastability in terms of profit earning, we propose a profit based adjusted weight factor for backpropagation network training. Instead of using the traditional least squares error, we add a factor which contains the profit, direction, and time information to the error function. The results show that this new approach does improve the forecastability of neural network models, for the financial application domain.

1 Introduction

Traditional time series forecasting uses error functions as the criteria of model performance. This is based on the goodness-of-fit. Neural networks especially backpropagation networks are similar to regression functions except for their nonlinearity. Traditional error functions are used in the same way in regression models. To apply neural network techniques to the financial context, researchers are working on ways to improve the forecastability of neural network based models [2, 3, 6]. Research shows that conventional statistical techniques for forecasting have reached their limitation in applications with nonlinearities in the data set [5]. It is observed that Normalized Mean Square Error (NMSE) and other error functions that are used for financial forecasting models may not make sense in the financial context [1]. Caldwell [1] gives a general review for the performance metrics. Yao *et al.* [9] use the correctness of trend and a paper profit to judge the performance of neural network forecasting model. Caldwell [1] also proposes a new measure, namely, weighted directional symmetry (WDS), which weights errors based on a function of the accuracy of predicted directions. The above works are based on post evaluation of forecasting model performance.

It has been found [4] that methods that give equal weight to all data (ordinary least squares) are less accurate than discounted least squares, which weight the most recent data more heavily. Refenes *et al.* [6] argue that in financial data, especially low frequency data, the structural relationship between an asset price and its determinants changes gradually over time as the economic environment evolves. Thus recent observations should be weighted more heavily than older observations. This is to emphasize the recent information. The Discounted Least Squares neural network model proposed by Refenes *et al.* [6] is to incorporate time factor to neural network forecasting model building.

In the context of financial time series forecasting, we are not only concerned at how good the forecasts fit their targets, but we are more interested in profits. In order to increase the forecastability in terms of profit earning, we propose a profit based adjusted weight factor for backpropagation network training. Instead of using the traditional least squares error, we add a factor which contains the profit, direction, and time information to the error function. The results show that this new approach does improve the forecastability of neural network models, for the financial application domain.

This paper first presents the argument on profit and goodness-of-fit in the next section. A section describes the new approach step by step is followed. In the application section, we apply the new models to five time series, United States Dow Jones Industrials Index, Hong Kong Hang Seng index, Malaysia Kuala Lumpur Composite Index, Japan Nikkei-225 and Singapore Straits Times Industrials Index forecasting and the results are presented in the same section. Finally, the discussion and conclusion section follows.

2 Goodness-of-fit vs. Profit

NMSE is one of the most widely used measurements in neural network training. It represents the fit between the neural network predictions and the actual targets. However, a prediction that follows closely the trend of the actual target would also result in a low NMSE. For pattern recognition, it is a very important signal. We argue that although NMSE is a very important signal for pattern recognition, it may not be the case for trading in the context of time series analysis. We can use the following simple example to explain our argument.

Suppose we have five time series as shown in Table 1. The "Target" is representation of a real financial time series though the figures are rounded for simplicity in illustration. In addition, four forecasting time series are artificially created based on the pattern of the "Target" line according to the following criteria. "Trend" series is a forecasting which accentuates the trend and is thus always correct in terms of trend. "MissBig" series is a forecasting which always fits the targets except for the big changes in the "Target" line. "MissSmall" series is a forecasting which always fits the targets except for the small changes. "Versus" series is a forecasting which is always wrong in terms of trend.

Table 1: Values of the Five Time Series

Target	10	11	8	9	6	7	6	12	11	12	11	12	6	7	6	8
Trend	10	16	5	10	4	8	5	18	9	13	10	13	5	8	7	8
MissBig	10	11	8	9	10	7	5	12	11	12	11	12	13	7	6	8
MissSmall	10	11	8	9	6	5	6	12	11	12	13	12	6	7	6	8
Versus	10	9	10	7	8	6	9	8	11	10	11	10	11	7	8	8

The performance of all the four forecasts with respect to the "Target" is shown in Table 2. "MissBig" and "MissSmall" series are the best in terms of goodness-of-fit in all four measures, i.e., Mean Squared Error (MSE), NMSE, Average Error (AE) and Absolute Average Error (AAE). To test the profit, we assume that a 1000 unit seed money is at hand before trading and a strategy based on the comparison of the difference between the predicted and the actual levels is used in trading. The goodness-of-fit happens to be the worst for "Trend" series. But its profit is as good as that of "Target" series which means that the forecast follows exactly its target. The reason for the good performance of this series is that its forecast is always correct in trend and thus the trading based on them will always be right. It shows that *Grad* (gradient) is a better indicator of the quality of forecasting for trading purposes as we can profit in the two "Miss" series. *Grad* can show how good the forecast trend is, which is very useful for trading purposes. Similarly, when the inputs are the changes of levels instead of the actual levels, then *Sign* also points to the accuracy of the forecast trends, and therefore it could be useful for trading. As the aim of financial forecasting is to maximize profit, we suggest that NMSE should not be used as the unique criterion of forecasting performance.

3 Backpropagation Networks with Profit Factor

The backpropagation network [7, 8] is the most popular and most widely implemented neural network especially in the financial forecasting domain. It is based on a multi-layered feedforward topology with supervised learning. The network is fully connected with every node in the lower layer linked to every node

Table 2: Statistics of Target series its forecasts. MSE: Mean Squared Error; AE: Average Error; AAE: Absolute Average Error; After: Seed Money (initially 1000 units) left after all trading.

	MSE	NMSE	AE	AAE	Grad	After	Trading
Target	0.0000	0.000000	0.0000	0.0000	100%	5345.45	100%
Trend	5.4375	1.014577	-0.4375	1.6875	100%	5345.45	100%
MissBig	4.1250	0.769679	-0.6250	0.7500	73.3%	1309.09	63.6%
MissSmall	0.5000	0.093294	0.0000	0.2500	73.3%	3300.00	63.6%
Versus	4.9375	0.921283	-0.0625	1.6875	0%	149.65	0%

in next higher layer. These linkages are attached with weight values. The learning of backpropagation neural network is actually an error minimization procedure. The weights are changed according to an error function which compares the neural network outputs with training targets.

3.1 Learning of Backpropagation Networks

The traditional error function is a Least Squares function, or Ordinary Least Squares function, which is shown in Equation 1.

$$E_{OLS} = \frac{1}{2N}\sum_{p=1}^{N}(t_p - o_p)^2 \tag{1}$$

where N is the number of exemplars in the training set, t_p is the target output, o_p is the neural network output. In particular, our new delta rule for backpropagation is:

$$\Delta w_{hk} = \eta(t_k - o_k)[o_k(1 - o_k)]x_h \tag{2}$$

here η is a user determined learning rate[1], k is the index of an output unit, h is the index of a hidden unit, x_h is the activation of the hidden unit h, o_k is the activation of hidden units k, t_k is the target output unit k, and w_{hk} is the weight from unit h to unit k. Δw_{hk} is the modification to each weight calculated on each backpropagation pass. In practice, a momentum term, determining the effect of previous Δw_{hk} value on the current one, is introduced.

$$\Delta w_{hk}^{(n+1)} = \Delta w_{hk}^{(n+1)} + \alpha \Delta w_{hk}^{(n)} \tag{3}$$

3.2 Directional Profit Model

In financial applications, profit gain is the major goal. To reflect this point in evaluating a forecasting models performance, Yao *et al.* [9] use the correctness of trend and a paper profit. Based on the directional symmetry, Caldwell [1] proposes a Weighted Directional Symmetry (WDS) function. Comparing with the targets, incorrectly predicted directions are penalized more heavily than the correct predictions. The definition of WDS is as follows,

$$\mathrm{WDS} = \frac{100\sum_{p=1}^{N} d_p|t_p - o_p|}{n} \tag{4}$$

where

$$d_p = \begin{cases} g & \text{if}(t_p - t_{p-1})(o_p - o_{p-1}) \leq 0 \\ h & \text{otherwise and} \end{cases}$$

the g and h are constants or some function of t_p, $g = 1.5$ and $h = 0.5$ as suggested by Caldwell for instance.

WDS is a post measure of the neural network performance in the financial context. For financial trading, the correctness of trend or direction is sometime more important than the goodness-of-fit. If the correctness

[1] learning rate η determines the speed of learning. The second factor has two effects: first, near-target output leads to smaller change; second, when output is less than target, $\Delta w_{hk} > 0$ thus forcing it to be higher and vise versa. The third factor has its larger values when the output is near 0.5.

is only based on how the forecasts fit the targets, we might not earn what we expected according to our forecasts. This is the main idea of WDS.

To further consider the profit driven procedure of financial trading, we understand that not only the direction but also the amount of change is important. The WDS weights should be adjusted more if a wrong direction is forecasted for a big change. Weights will be adjusted less if a right direction is forecasted for a big change. We then propose another new profit adjust factor which is a function of changes and direction as follows,

$$f_{DP}(p) = F(|t_p - t_{p-1}|, sign(\Delta t_p, \Delta o_p)) \tag{5}$$

The Directional Profit adjustment factor is defined as

$$f_{DP}(p) = \begin{cases} a_1 & \text{if } \Delta t_p * \Delta o_p > 0 \text{ and } |\Delta t_p| \leq \sigma \\ a_2 & \text{if } \Delta t_p * \Delta o_p > 0 \text{ and } |\Delta t_p| > \sigma \\ a_3 & \text{if } \Delta t_p * \Delta o_p < 0 \text{ and } |\Delta t_p| \leq \sigma \\ a_4 & \text{if } \Delta t_p * \Delta o_p < 0 \text{ and } |\Delta t_p| > \sigma \end{cases}$$

where σ is a threshold of the changes of sample data. We use the standard deviation of the training data set (including validation set).

$$\sigma = \frac{1}{n} \sum_{p=1}^{N} t_p - \mu \tag{6}$$

where μ is the mean of the target series. In the case of our first experiment, $a_1 = 0.5, a_2 = 0.8, a_3 = 1.2$ and $a_4 = 1.5$. [2]

The new error function will be,

$$E_{DP} = \frac{1}{2N} \sum_{p=1}^{N} f_{DP}(p)(t_p - o_p)^2 \tag{7}$$

3.3 Time Dependent Directional Profit Model

The Discounted Least Squares function proposed by Refenes *et al.* [6] is given by

$$E_{DLS} = \frac{1}{2N} \sum_{p=1}^{N} w(p)(t_p - o_p)^2 \tag{8}$$

where N, t_p and o_p are as same as in Equation 1 and $w(p)$ is an adjustment of the contribution of observation i to the overall error

$$w(p) = \frac{1}{1 + e^{\left(a - \frac{2ap}{N}\right)}} \tag{9}$$

The discount rate a will decide the function which actually is a function of the recency of the observation. We use a=6 as suggested by Refenes *et al.* [6] in this study.

DP Model emphasizes on the profit while DLS model emphasizes on time. To include emphasis on the time concept, we incorporate Refenes's approach to our DP model. We name it Time dependent Directional Profit model (TDP) which is proposed as follows,

$$E_{TDP} = \frac{1}{2N} \sum_{p=1}^{N} f_{TDP}(p) * (t_p - o_p)^2 \tag{10}$$

where $f_{TDP}(p) = f_{DP}(p) * w(p)$. The weight changing is the same as the approach used in the traditional OLS backpropagation network except that an adjustment factor is introduced for both DP and TDP models.

[2] We use $a_1 = 0.5, a_2 = 1, a_3 = 1$ and $a_4 = 1.5$ at first(a_1 and a_2 following Caldwell). We then use h=0.1 as the step and find out that $a_2 = 0.8$ and $a_3 = 1.2$ perform the best among our experiments. Further optimization may be conducted for the a_i in future.

4 Experiments with Financial Time Series

Four major Asian stock market indices, Hong Kong Hang Seng index (HS), Malaysia Kuala Lumpur Composite Index (KL), Japan Nikkei-225 (NK) and Singapore Straits Times Industrials Index (ST) together with the world economic benchmark United States Dow Jones Industrials Index (DJ) are applied to these four models, namely benchmark Ordinary Least Squares (OLS) model, Refenes's Discounted Least Squares (DLS) model, Directional Profit (DP) model and Time dependent Directional Profit (TDP) model.

We will examine seven time series for each stock index in this paper. There are 1960 data for each stock under study. Each time series contains 280 trading days stock indices. For example, the seventh time period of HS indices, chosen for our experiment, ranges from 3 August 1995 to 17 September 1996. The first 260 days are used as training data which include 80 days for validation purpose. The remaining 20 days are used for testing the forecastability of the neural network model. The time period and the statistic results of the data under study are shown in Table 3.

A one hidden layer network is selected as the neural network model in this study. To simplify our experiment, the learning rate, η, is set to 0.25 and the momentum rate, α, is set to 0.9. The structure of the neural network is 30-10-1, i.e. 30 inputs, 10 nodes in the hidden layer and one output. Thus 30 consecutive days of data are fed to the network to forecast the index of the following day.

Table 3: Statistic Results for Five Time Series

	DJ	**HS**	**KL**	**NK**	**ST**
Minimum	2243.040	2093.000	327.340	14309.409	993.160
Mean	3524.442	6260.082	738.201	23011.814	1735.970
Maximum	5889.200	12201.089	1314.460	38915.859	2493.710
St. Derivation	902.099	2996.524	253.020	6113.686	417.006
Variance('000)	813.782	8979.157	64.019	37377.965	173.893
Skew	0.955	0.255	0.343	0.990	0.181
First Day	06 Feb 89	08 Nov 88	19 Oct 88	25 Jan 89	06 Dec 88
Last Day	16 Sept 96	17 Sept 96	17 Sept 96	17 Sept 96	17 Sept 96

As we focus on the profit, instead of using NMSE only, we introduce a paper profit measure for the model performance. To test the profit, we assume that a 1000 unit seed money is at hand before trading. The seed money is used to buy a certain amount of indexed stocks in the right proportion when the prediction shows a rise in the stock price. To calculate the profit, the major blue chips in the stock baskets are bought or sold at the same time. To simplify the calculation, assume that the aggregate price of the major blue chips is the same as the index. A method used is to go long when the neural network model predicts that the stock price will rise. After that the basket of stocks will be held at hand till the next turning point that the neural network predicts. The formula used to calculate the annualized profits is defined as follows:

$$Return = \left(\frac{After}{Seed}\right)^{\frac{m}{n}} - 1 \tag{11}$$

where $Seed = 100$ (The seed money used on the first testing day), $After =$ The money left on the last testing day, $m = 260$(No. of trading days per year), $n = 20$ (No. of trading days of testing period).

Table 4 compares the paper profit achieved by different models for the whole data set of five markets under study.

5 Concluding Remarks

Most applications of financial time series forecasting rely on the goodness-of-fit as their performance criterion. However, a well fitted prediction times series with target series does not necessarily mean a good profit achieved by the forecasts. As the ultimate goal of using financial forecasts is to make profit, the financial

Table 4: Comparison of Profits Achieved for Whole Data Set (on Average). Excess: Excess Return over benchmark (OLS).

	DJ	**HS**	**KL**	**NK**	**ST**
OLS	16.812	17.812	4.274	16.658	19.389
DLS	7.255	16.322	16.373	22.799	37.870
DP	20.213	1.120	8.359	25.323	29.517
TDP	19.618	24.672	31.897	32.642	37.951
Best	DP/TDP	TDP	TDP	TDP	TDP
Excess(TDP)	2.806	6.861	27.623	15.984	18.562
Excess(DP)	3.401	-16.692	4.085	8.665	10.128

profitability becomes the major concern of selecting a forecasting model. We propose a profit model to meet the real goal of financial forecasting. We understand that success in one instance does not necessarily mean the same for other cases. Therefore five markets and seven data sets are experimented in our study. The results show that this proposed new criterion does improve the forecastability of neural network forecasting models. It provides another way to improve neural network model in order to suit the application domain requirement.

To emphasize time or profit alone would not be good for financial forecasting. A time dependent profit model combining these two factors has been shown to achieve about 50% improvement about 3% excess annual return above benchmark OLS model even for the worst case. We have gained insights from this study in that emphasizing time or profit alone appears not sufficient. The combination of these two factors in our model has been shown to be promising model with up to 27.6% excess annual return above benchmark OLS model or at least 2.8% excess annual return in the worst case.

There are still rooms for further improvement. Instead of using just time delayed indices, other input parameters [6, 9] could be introduced for better forecasting in the future. While the proposed model will be further fine tuned. the present research has been worthwhile in demonstrating the need to consider time dependency and direction of profit change in formulating a new criterion for financial forecasting with neural networks.

References

[1] Caldwell, R.B., “Performances Metrics for Neural Network-based Trading System Development”, *NeuroVe$t Journal*, Vol. 3, No. 2. 1995, pp22-26.

[2] Hutchinson, J.M., A.W. Lo and T. Poggio, “A Nonparametric Approach to Pricing and Hedging Derivative Securities via Learning Networks”, *The Journal of Finance*, Vol. 49, No. 3, July 1994, pp851-889.

[3] LeBaron, B. and A.S. Weigend, “A Bootstrap Evaluation of the Effect of Data Splitting on Financial Time Series”, *IEEE Transactions on Neural Networks*, Vol. 9, No. 1, 1998, pp213-220.

[4] Makridakis, S., “The Accuracy of Extrapolation (Time Series) Methods: Results of a Forecasting Competition”, *Journal of Forecasting*, Vol. 1, 1982, pp111-153.

[5] Refenes, A. N., A. Zapranis, and G. Francis, “Stock Performance Modeling Using Neural Networks: A Comparative Study with Regression Models”, *Neural Network*, Vol 5, 1994, pp961-970.

[6] Refenes, A.N., Y. Bentz, D.W. Bunn, A.N. Burgess, A.D. Zapranis, “Financial Time Series Modeling with Discounted Least Squares Backpropagation”, *Neurocomputing*, Vol. 14, No. 2, 1997, pp123-138.

[7] Rumelhart, D.E., G.E. Hinton, R.J. Williams, “Learning Internal Representations by Error Propagation”, in: *Parallel Distributed Processing*, Volume 1 D.E. Rumelhart, J.L. McClelland (Eds.), MIT Press, Cambridge, MA, 1986, pp318-362.

[8] Werbos, P.J., *Beyond Regression: New Tools for prediction and analysis in the behavioral Science*, PhD Thesis, Harvard University, 1974.

[9] Yao, J.T., H.L. Poh, “Equity Forecasting: a Case Study on the KLSE Index”, *Neural Networks in Financial Engineering, Proceedings of 3rd International Conference On Neural Networks in the Capital Markets, Oct 1995, London*, A.-P N. Refenes, Y. Abu-Mostafa, J. Moody and A. Weigend (Eds.), World Scientific , 1996, pp341-353.