In Proceedings of International Conference on Neural Information Processing, ICONIP 2000, Taejon, Korea, pp 842 - 846, 2000.

Trading Signal Prediction

Siu-Ming CHA

Computer Science and Engineering Department
The Chinese University of Hong Kong,
Shatin, HONG KONG
Email : smcha@cse.cuhk.edu.hk
http://www.cse.cuhk.edu.hk/~smcha

Laiwan CHAN

Computer Science and Engineering Department
The Chinese University of Hong Kong,
Shatin, HONG KONG
Email : lwchan@cse.cuhk.edu.hk
http://www.cse.cuhk.edu.hk/~lwchan

Abstract

We present a trader using feedforward network to output BUY, SELL and HOLD trading signals. We illustrated the effectiveness of the trader on various stocks. We have also applied the trader to stocks which are not the same as the stocks in the training set. It shows that the underlying mechanism learned by the trader can be generalized to any stocks. A major application of this would be on trading of stocks with or without historical records for training purpose.

1 Introduction

Neural networks are powerful modeling tools to find the hidden underlying dynamics of unknown systems. Currently, neural networks have been mainly used as prediction tools to forecast the future price of stocks, financial indexes or exchange rates [4][6][3]. The trading strategy (*i.e.*, when to buy, sell and hold), on the other hand, is some specific pre-defined rules. For examples, Yao and Poh used the rise or fall of predicted prices to determine the trading signals [7]. Pi and Rognaldsson used the slope of the predicted prices as rules in their trading system [5]. In these cases, the trading signals were obtained via a two-steps process; a prediction of future prices and then the generation of the trading signals from the predicted prices using some predefined trading rules. As the ultimate goal of the trader is to give sell signals at the maxima of the price trend and buy signals at the minima, an accurate prediction of the future stock prices which minimizes the mean square error strategy is only an indirect way to estimate the trading signals. A good trading is not required to give an accurate prediction of the future stock prices. Here in this paper, our trading system generates the trading signals directly from the neural network without going through the prediction process.

From the view point of technical analysis, the optimal trading signals directly related to the historical prices. In principle, the underlying mechanism to obtain the signals does not depend on which stocks we are applying [1]. Therefore, a major objective of this paper is to find out if the learned mechanism for trading using neural network can be generalized to any stocks. Section 4 presents the results of this finding. We show that a well-trained trader is able to generalize and apply to other stocks. The major application of this knowledge is that we can train a neural network trader using existing stock data to generate the trading signals. The trader is then applied to trade some recently listed stocks, which do not have much historical records for us to train a separate trader.

2 The target trading signals

The trading signals that we consider in this paper consist of three types; buy, sell and hold. The ideal buy signals are located at the local minima of the historical stock prices. The ideal sell signals are located at the local maxima of the historical stock prices. We have hold signals when the stock price is neither close to the maxima nor minima. This information serves as criterion for us to construct the target signals for the training of our neural network trader. We choose continuous values as the output of the neural networks to denote the strength of the trading signals [8]. The actual trading signals are obtained from thresholding the strength. The thresholding levels can be served as parameters for investors to control the frequency of trading, as some investors prefer to trade frequently and some prefer to trade once in a while. Low risk taking investors trade only when the signals are strong; other investors trade even when the signals are weak.

To construct the target signals to our neural network, we need to local the position of the local extrema of the stock prices. The following strategy would be adopted. Since stock prices are inherently noisy [2], local extrema occur very frequently and the price difference between consecutive extrema may be small. To most investors, small fluctuations of prices are of little interest. Trading would only be considered if the price difference is large enough to cover at least the transaction cost. Therefore, we select only those local extrema with a substantial gap in between. The local extrema we extracted have to exhibit the following characteristics,

- The extrema have to be alternative, *i.e.* a local minimum has be followed by a maximum and vice versa.
- The price at a minimum (maximum) has to be lower (higher) than the prices at its two adjacent maxima (minima).
- Between every consecutive maximum/minimum pair, there exists a daily drop in price larger than Δ_d. Between every consecutive minimum/maximum pair, there exists a daily rise in price larger than Δ_r. In our case, $\Delta_d = \Delta_r = 0.5$.

Finally, we construct the target trading signals from the extracted extrema by

$$t_i = \frac{c_i - c_{low}}{c_{high} - c_{low}} \times 0.95 + 0.05 \tag{1}$$

where t_i is the target signal at day i, c_i is the price of the stock at day i, c_{high} and c_{low} are the prices of the adjacent maximum and minimum to day i respectively.

One advantage of this target signal generation over the traditional prediction of stock prices is that the generated target signal sequence is bounded between 0.05 and 0.95. Figure 1 shows an example of our stock price. The stars and circles indicated the extrema we located and they indicated the ideal buy and sell signals respectively. Figure 2 shows the corresponding target signals generated by the described method.

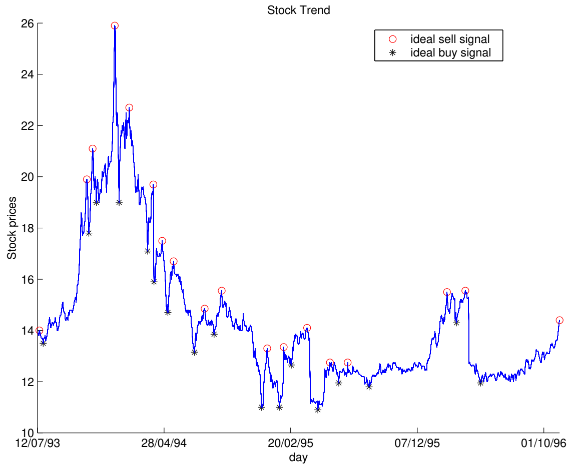

Figure 1: Example of ideal buy and sell locations extracted.

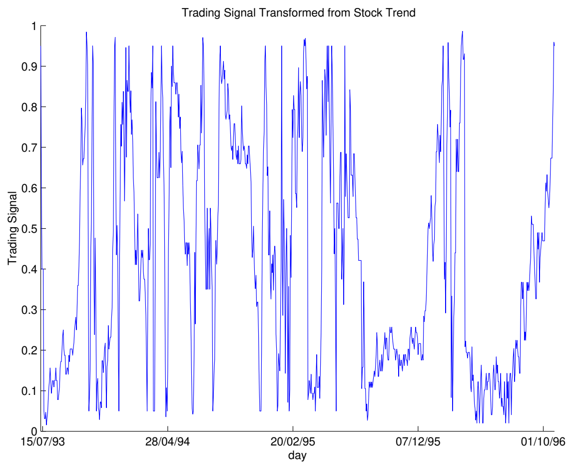

Figure 2: The target signals for the stock in Figure 1.

To map the network output, or target signal, t_i, to the trading signals, S_i, we can simply apply the following thresholding function

$$S_i = \begin{cases} BUY & \text{if } t_i < m_i - \sigma_b \\ SELL & \text{if } t_i > m_i + \sigma_s \\ HOLD & \text{otherwise} \end{cases} \tag{2}$$

where m_i is the mean value of the network output, σ_b and σ_s are two constants. In our case, $\sigma_b = \sigma_s = 0.1$.

3 Trading Stocks

We trained a neural network to predict the stock trading signal. The network is a feedforward model with four input units, three hidden units and one output unit. The four inputs are the rate of return of the stock in the previous four working days. We have also normalized the rate of return by the following equation, which bounds the input signal within the range of -1 and 1.

$$x_i = \frac{c_i - c_{i-j}}{c_i \lambda_j} \quad j = 1, 2, ..., N \tag{3}$$

$$\lambda_j = \frac{2}{P-N} \sum_{i=N+1}^{P} \frac{|c_i - c_{i-j}|}{c_i} \tag{4}$$

where P is the number of stock data in our training set and N is the number of daily lags in the input. In our case, $N = 4$.

The output is the strength of the trading signal and sigmoid function was used as the activation function.

To evaluate the effectiveness of the trading signals, we derived the *investment return* function, IR, as follows,

$$IR = \sum_{i \in D(S_i)} \frac{\bar{c}_{pi} - c_i}{c_i} \quad (5)$$

where $\bar{c}_{pi}$ is the p-days average price of the stock starting from day $i+1$, i.e.

$$\bar{c}_{pi} = \frac{1}{p} \sum_{j=1}^{p} c_{i+j} \quad (6)$$

$\bar{c}_{pi}$ is used to evaluate the average expected return as we are assuming that we would take the profit within a short period of time. $D(S_i)$ in equation 6 is the set containing those days which give a particular trading signal. In specific, we are interested in buy and sell signals. So, we use IR_S to denote the IR for SELL signals and IR_B to denote the IR for the BUY signals. For the BUY signals, a large positive IR_B is desirable. On the other hand, for the SELL signals, a negative IR_S is desirable. Taking into the account of both signals, the overall return during the period should be $IR_B - IR_S$.

We apply the neural network trader to the following three stocks in the Hong Kong market.

S1 : Hong Kong Telecommunications Ltd.

S2 : HSBC Holdings Plc.

S3 : The Hong Kong and China Gas Co. Ltd.

These stocks are major stocks in Hong Kong and also constituent stocks of the Hang Seng Index. The daily closing prices from 12/7/93 to 7/11/96, with a total of 826 days, were used. S1 is a sample stock showing many ups and downs on the historical prices (Figure 3). It has a slight drop during the testing period (-9.4 %). S2 is a stock with a raising trend (Figure 4). It has a substantial gain of 98.8 % during the testing period. S3 is a stock with low volatility during the testing period. In each stock, the first half of the historical daily closing prices were used as training set and the remainings were used as testing set.

Tables 1, 2 and 3 show the investment return of these stocks for BUY and SELL trading signals. Each entry on the table is the average result of five runs, and the average IR of $p = 7, 5, 3$ were listed on separated columns. "Gain" indicates the change in the price during the testing period for reference. The trader performed well on S1 as expected. It found the appropriate time to buy the stock before the price raised and to sell the stock just before the price dropped. This is demonstrated by the fact that the larger p is, the better is the gain. S2 had a very large IR_B, partly due to the rising trend of the stock. For short selling, the IR_S of S2 is not satisfactory when p is small. Only IR_S for $p = 7$ is negative. But overall, it achieved an expected

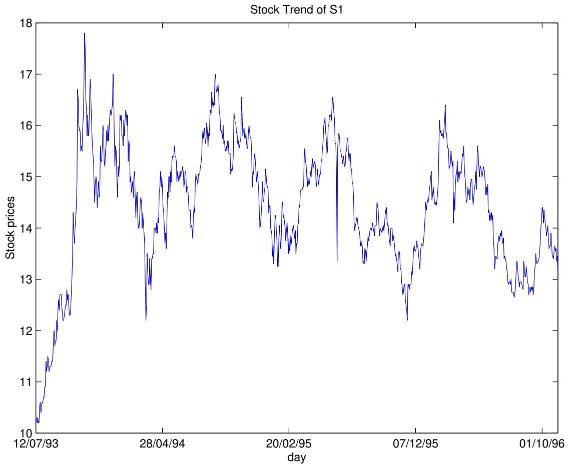

Figure 3: The historical prices of S1.

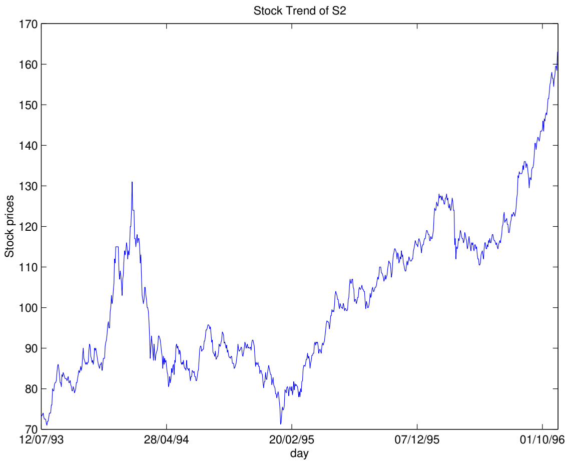

Figure 4: The historical prices of S2.

return of about 250 % compared to about 100% raise in stock price. S3 showed a small gain using the trader as expected, because of its low volatility. In the experiments, we only quoted the result of average expected return, which we assumed that the investment in each trading is a fixed amount. Higher return may be yielded if we invested more if the strength of trading signals are strong. Since our objective is the demonstration of the effectiveness of our trader instead of building a complicated trading strategy to optimize the profit, we prefer to use a simple measurement to evaluate the performance.

	$p = 7$	$p = 5$	$p = 3$	average
IR_B	147.4 %	138.7 %	56.3 %	114.1 %
IR_S	-190.2 %	-93.6 %	-53.8 %	-112.5 %
gain	-9.4 %			

Table 1: Testing result (IR) of S1.

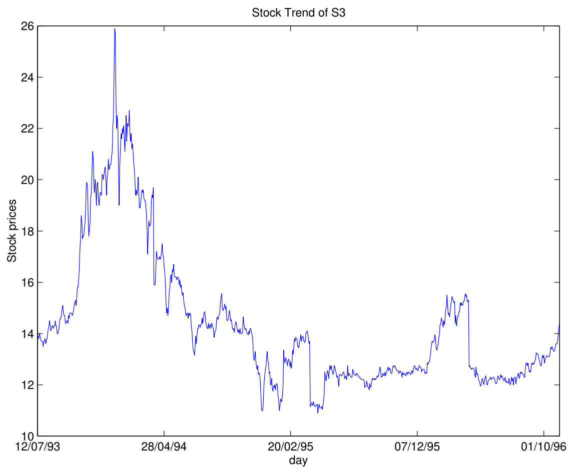

Figure 5: The historical prices of S3.

	$p = 7$	$p = 5$	$p = 3$	average
IR_B	491.6 %	260.1 %	108.7 %	286.8 %
IR_S	-6.01 %	28.6 %	41.4 %	21.3 %
gain	98.8 %			

Table 2: Testing result (IR) of S2.

4 Applying the trader to other testing sets

As explained in the introduction, we want to verify if the knowledge learned by the trader can be generalized to any stocks. Similar to what we did in the last section, we trained the trader using the training sets. But this time, we tested the trader using the testing sets from other stocks. Table 4 summarizes the average result of the cases with $p = 7, 5$, and 3, each with five runs. The data in brackets were the ones which both training sets and testing sets come from the same stock. It indicates that the trader trained by S1 gave the best performance. Using it to trade S3 gave better performance than the trader using S3 as training set. Another interesting observation is that the trader using S1 and S3 as training sets can make profits using selling signals when applied to testing set S2, which has an overall rising trend.

5 Applying the trader to new stocks

The previous section showed that the trader we developed can be applied to other stocks. In particular, we can generate satisfactory results even if the training set and the testing set are data from two unrelated stocks. One main application of this result is on the trading of stocks without much history. As nowadays, we have many new stocks appearing on the market. For examples, many Chinese Enterprises have been listing on the Hong Kong market for only a couple of years. Recently, we have many IT related stocks appearing on the market. These stocks have insufficient data for us to train the neural network. As we have learned that our trader works well on stocks other than the one used for training, this opens a solution for the trading of these stocks with limited history. S4 (Shanghai Petrochemical Co. Ltd.) and S5 (Pearl Oriental Holdings Ltd.) are two recently listed stocks and we use them as examples in this paper. We found that our traders, trained by S1, S2 and S3, performed well (Tables 5 and 6). The testing period was from 10/3/95 to 07/11/96, a total of 409 trading days. The average number of buying and selling signals, N_B and N_S respectively, are also tabulated for reference. This indicates that we have about 73.1 trading signals (including buy and sell) generated during the period, and on average we have either a buy or sell signal on every 5 or 6 days.

	$p = 7$	$p = 5$	$p = 3$	average
IR_B	19.9 %	6.9 %	4.1 %	10.3 %
IR_S	-16.9 %	-5.6 %	-4.8 %	-9.1%
gain	5.1 %			

Table 3: Testing result (IR) of S3.

BUYING	Training		
Testing	S1	S2	S3
S1	(114.1%)	121.4%	-37.65%
S2	71.3%	(286.8%)	23.5%
S3	134.7%	76.9%	(10.3%)
SELLING	Training		
Testing	S1	S2	S3
S1	(-112.5 %)	9.2%	-11.0 %
S2	-120.9 %	(21.3 %)	-47.6 %
S3	-34.2 %	6.65 %	(-9.07 %)

Table 4: IR for different combinations of training sets and testing sets.

	IR_B	N_B	IR_S	N_S
S1	160.8%	20.7	-232.8 %	23.5
S2	122.7 %	38.8	-254.5 %	62.2
S3	116.3 %	30.8	-104.8 %	39.5
gain	-8.9 %			

Table 5: IR of S4 and average number of signals using traders trained on other stocks.

6 Conclusion

In this paper, we construct a trader using neural networks. This trader uses historical prices and generates BUY, SELL and HOLD signals for the trading of stocks. We discovered that the knowl-

	IR_B	N_B	IR_S	N_S
S1	62.7 %	19.3	-171.64 %	26.0
S2	70.0 %	50.7	-81.0 %	58.5
S3	29.7 %	27.0	-31.1 %	41.8
gain	103.3 %			

Table 6: IR of S5 and average number of signals using traders trained on other stocks.

edge of trader can be generalized to perform trading on other stocks. This is particularly useful to trade on stocks with relatively few historical data as the training set. We had also found that better result is obtained if the training set comes from stocks with plenty of ups and downs. This shows that we should select stocks with rises and falls as training set to our trader.

Acknowledgement The authors would like to thank The Research Grants Council, HK for support.

References

[1] Joumana Ghosn and Yoshua Bengio. Multi-task learning for stock selection. In Michael C. Mozer, Michael I. Jordan, and Thomas Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, page 946. The MIT Press, 1997.

[2] M. Magdon-Ismail, A. Nicholson, and Y. Abu-Mostafa. Financial markets: very noisy information processing. *Proceedings of the IEEE*, Vol 86(11):2184 – 2195, Nov 1998.

[3] M.E. Malliaris. Modeling the behavior of the s&p 500 index: a neural network approach. In *Proceedings of the Tenth Conference on Artificial Intelligence for Applications*, pages 86 – 90, 1994.

[4] John Moody. Forecasting the economy with neural nets: A survey of challenges and solutions. In Klaus-Robert Muller Genevieve B. Orr, editor, *Neural networks : tricks of the trade*, pages 347 – 371. Berlin ; New York : Springer, 1998.

[5] Hong Pi and Thorsteinn S. Rgnvaldsson. A neural network approach to futures trading. In *Neural Networks in Financial Engineering (Proc. 3rd International Conf. on Neural Networks in the Capital Markets*, pages 17 – 25. World Scientific, Singapore, 1996.

[6] A. S. Weigend, B. A. Huberman, and D. E. Rumelhart. Predicting sunspots and exchange rates with connectionist networks. In M. Casdagli and S. Eubank, editors, *Nonlinear Modeling and Forecasting*, pages 395 – 432. Addison-Wesley, 1992.

[7] Jingtao Yao and Hean-Lee Poh. Forecasting the klse index using neural networks. In *ICNN'95(1995 IEEE International Conference on Neural Networks), Perth, Western Australia*, volume 2, pages 1012 – 1017, 1995.

[8] Joseph Zirilli. *Financial Prediction Using Neural Networks*. International Thomson Publishing Services Ltd., London, UK, first edition, 1997.