Intelligent Trading Using Support Vector Regression and Multilayer Perceptrons Optimized with Genetic Algorithms

Ming Zhu and Lipo Wang

Abstract—**This paper proposes an intelligent trading system using support vector regression optimized by genetic algorithms (SVR-GA) and multilayer perceptron optimized with GA (MLP-GA). Experimental results show that both approaches outperform conventional trading systems without prediction and a recent fuzzy trading system in terms of final equity and maximum drawdown for Hong Kong Hang Seng stock index.**

I. Introduction

The fast changing nature of financial market has always been a puzzle to investors and researchers. Some previous works have showed that financial market is highly noisy and non-linear [1]. Price prediction has been one of main research foci, while others concentrated on prediction of trends in the financial market to achieve better profits (e.g., [10]).

In recent years, neural networks, especially multilayer perceptron (MLP) [2], have been extensively implemented in financial market for price prediction. Among the many MLP training algorithms, genetic algorithms (GA) can be used to determine the initial weights of the MLP [3]. Support vector regression (SVR) has become a popular tool for time series prediction [4]. In this paper, both MLP and SVR optimized with GA are used for financial prediction.

Mainly two types of input data have been used in training. One type is price or technical indicators, while the other type includes macroeconomic indices, such as the interest rate and gold price. In order to keep the trading system simple, in this paper, only price and technical indicators are used as input. Patel et al [5] used the MLP as well as radial basis function neural networks to forecast closing prices of Dow Jones Industrial Average and Nikkei 225 Stock Average indices. They claimed that the best forecasting classification accuracies were 72%. Lee et al [6] used neural networks with weighted fuzzy membership functions to forecast daily Korea composite stock price index. Their classification accuracy was 59.0361%. Doeksen et al. [7] used MLP trained with back propagation and Mamdani and Takagi-Sugeno fuzzy inference systems trained with gradient descent algorithms and genetic algorithms to predict the price of Microsoft and Intel. They claimed that the best network could provide 103% return on Microsoft stock in 252 trading days. Chen et al. [8] used a probabilistic neural network to predict the future price and implemented the model on Taiwan Stock Index to generate buy and sell signals. They claimed that this probabilistic neural network based investment strategy could obtain 362.27% from September 1987 to August 1992, which is higher than other strategies discussed in the paper. Ang et al. [9], who used rough set-based neuro-fuzzy stock trading decision model to trade on Singapore exchange, reported a profit of 16.0362 times of initial capital invested in NOL during 1917 trading days and 2.8582 times of initial capital invested in DBS during 2222 trading days. Huang et al. [10] uses hierarchical coevolutionary fuzzy system (HiCEFS) to predict a technical indicator and hence build a prudent trading strategy. Furthermore, by testing this model with real world data of Hong Kong Hang Seng Index and NOL stock in Singapore Exchange, they achieved a final return of 14.251 times of original capital on NOL stock in 2329 trading days and 5.781 times of original capital on Hang Seng Index in 2461 trading days.

In summary, many pioneer researches have focused on the minimizing the mean square error (MSE) in price direction prediction as well as providing paper profits in trading financial market. In this paper, an intelligent trading system is presented and evaluated in terms of equity curve and maximum drawdown.

The rest of the paper is organized as follows. In section II, the conventional trading system and trading system adopted SVR-GA and MLP-GA are presented. In section III, experiment settings and trading simulation results are given and evaluated. Finally, section IV provides conclusions of this paper.

II. Trading System Design

A. Technical Analysis

There are two main types of analysis in the market: technical analysis and fundamental analysis. Fundamental analysis is based on the premise that a stock, bond, fund, commodity, or a market as a whole has an underlying intrinsic value. By analyzing the fundamental characteristics, such as assets, liabilities, income, supply or demand, values can be determined [11]. On the other hand, technical analysis believes that the market's price reflects all relevant information, such as news and events. In addition, it believes that history will repeat itself in such a way that we could trade for profits. Therefore, technical analysis uses only historical data to build the model for future investment. Among all the technical indicators, moving average is considered as the simplest and most useful one. It is popular because moving average could discover the trends by smoothing the prices. Most importantly, moving average could be a useful tool

The authors are with School of Electrical & Electronic Engineering, Nanyang Technological University, Block S1, 50 Nanyang Avenue, Singapore 639709. zhuming0828@gmail.com, elpwang@ntu.edu.sg

978-1-4244-8126-2/10/$26.00 ©2010 IEEE

since investors can make profits through trends. Exponential moving average (EMA), being one of the moving average indicators, is considered as more adaptive since it puts more weights on recent prices, e.g., today's closing price, while putting less weights on earlier days. Equation (1) shows the calculation of EMA:

$$\mathrm{EMA_N(t)} = \mathrm{EMA_N(t-1)} + \frac{2}{\mathrm{N+1}}[\mathrm{p(t)} - \mathrm{EMA_N(t-1)}]$$

where $\mathrm{p(t)}$ is the close price at time t. (1)

However, this technical indicator would not be useful unless it could be used to form a trading system with a trading rule. Hence, in this paper, a trading system proposed by Huang et al. [10] is adopted. This trading system uses a technical indicator named Percentage Price Oscillator (PPO), PPO is calculated as equation (2):

$$\mathrm{PPO_{S,L}(t)} = \frac{\mathrm{EMA_S(t)} - \mathrm{EMA_L(t)}}{\mathrm{EMA_L(t)}} \quad (2)$$

A buy signal is triggered if PPO is greater than 0, in other words, when short term EMA crosses over with long term EMA. A sell signal is triggered if PPO is less than 0, which means long term EMA is above short term EMA. This trading system is a typical trend following system which could catch every major trend to make promising profit, while suffering minuscule losses when significant trends are absent in the market.

B. Intelligent Trading Systems

When using PPO trading system, there would be a lag between the time when the trend starts and the time when the trading system detects it. Failing to compensate the lag has been a dominant disadvantage of traditional trading systems (without prediction). An intelligent trading system attempts to predict PPO in the near future, so as to enter the market before the trend while closing the position before the market falls. The input for our intelligence trading system studied in this paper is PPO of the last 8 days and the output is PPO in the future 5 days. Our intelligent model is either an MLP optimized by GA or an SVM optimized by GA. 0.2% of transaction cost and slippages are counted in the process of calculating profits (Fig.1).

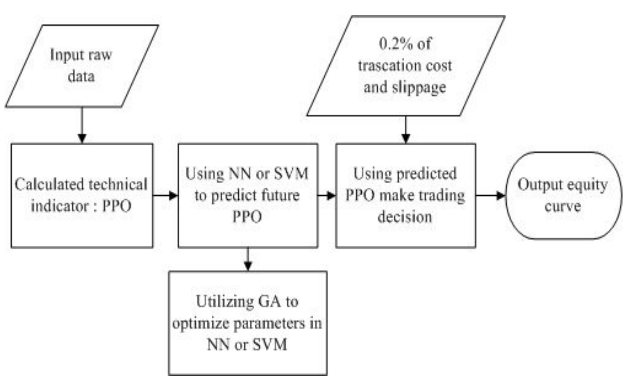

Fig. 1. A predictive trading system.

III. Experimental Results and Analysis

A. Experimental Settings

A feed-forward MLP with one hidden layer is used. The number of hidden neurons is determined to be 30 by the trial and error [19][20][21]. The Levenberg-Marquardt algorithm is used to train the MLP. Initial weights and bias of the neural network are determined by GA. The weights and bias in each layer are coded in sequence in GA. Initial population is selected randomly. The maximum generation is set as 800 based on trial and error, and mutation probability is 0.02. All the settings are summarized in Table I. The fitness in GA is based on the error of predicted output and desired output, expressed as equation (3).

$$fitness = \frac{1}{\sum(Y_{desire} - Y_{predict})^2} \quad (3)$$

where Y_{desire} is the desired output and $Y_{predict}$ is the predicted output.

The settings for the NN+GA model are selected as Table I.

TABLE I
The NN+GA Model Settings

GA settings	
Population size	300
Maximum Generation	800
Stop criteria	maximum generation reached
Probability of mutation	0.02
Crossover method	Arithmetical crossover
Neural Network Settings	
Layers	Single hidden layer with 30 neurons
MATLAB Transfer function	Transig, purelin
Training	Levenberg-Marquardt
Performance Measure	MSE (mean square error)

Given that a training set (x_i, y_i), i = 1…l, the support vector machine requires the minimum value of following equation (4) [13][18].

$$min_{w,b,\varepsilon}\ \frac{1}{2}W^TW + C\sum_{i=1}^{l}\varepsilon_i \quad (4)$$

$$subject\ to\ y_i(W^T\emptyset(x_i)+b) \geq 1-\varepsilon_i, \varepsilon_i \geq 0$$

The kernel function in this experiment is radial basis function (RBF):

$$K(x_i, x_j) = \exp\left(-\gamma\left|\left|x_i - x_j\right|\right|^2\right), \gamma > 0 \quad (4)$$

Main parameters in SVR are error insensitive tube around the regression function [14] and the balance of training errors with model complexity. In this paper, GA is used to determine these two SVR parameters. The Fitness function used to optimize SVR is the same as optimizing NN, which is described in equation(3).

The settings for SVM+GA model are summarized in table II.

TABLE II
THE SVM+GA MODEL SETTINGS

GA settings	
Population size	30
Maximum Generation	200
Stop criteria	maximum generation reached
Probability of mutation	0.05
Probability of crossover	0.4
SVM Settings	
Kernel function	radial basis function

The training results of neural networks could be evaluated by linear regression. The best network is indicated by the correlation coefficient, r closed to unity (r ≈ 1) [15]. In this experiment, the linear regression R=0.96342, which could be considered as well trained network. Figure 2 shows this overall linear regression.

PPO of future 5 days is selected to be target after prudent consideration. As a matter of fact, larger time horizon would definitely produce more profit, which is made by early entry and early exit. On the other hand, the larger the time horizon, the harder it is to predict. This would increase the chance of wrong prediction, which decreases the profit. Table 3 is total return of investing 1 dollar, with different prediction time horizon.

In this experiment, reinvesting all capital is selected as the money management strategy, in which the trading system would re-invest all the profit and initial capital for next buy or sell decision.

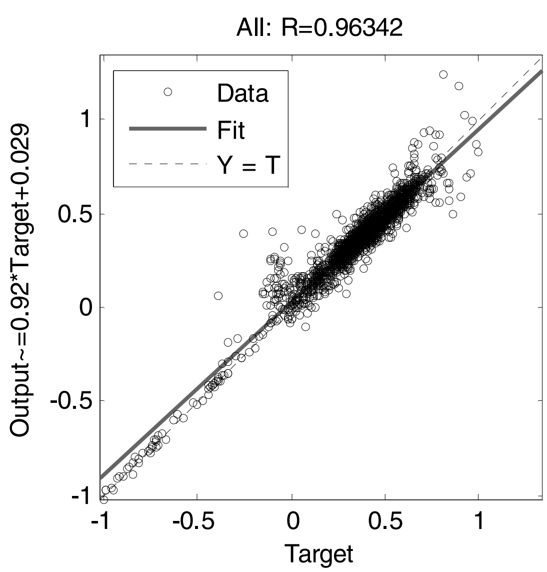

Fig.2. The overall linear regression of trained neural network

TABLE III
EXPECTED RETURN FOR DIFFERENT PREDICTION TIME

	Total return
No prediction	2.064
Predict future 3 days PPO	4.357
Predict future 5 days PPO	6.910
Predict future 7 days PPO	5.464

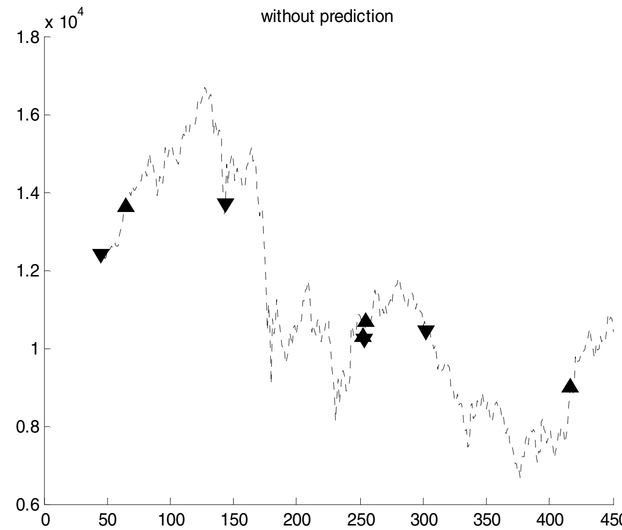

Fig.3. Trading signal of conventional trading system

B. *Results and Analysis*

This intelligent trading system uses Hong Kong Hang Seng Stock Index (HSI) from 1986-12-31 to 1997-1-28, total 2500 daily closing price as training session, and uses HSI from 1997-1-29 to 2007-3-8, total 2500 daily price as out of sample testing data. All the HSI index data was obtained from Yahoo Finance (http://finance.yahoo.com/q/hp?s=^HSI). The proposed trading system assumes that it is possible to enter the market using the close price on the same day which triggers the trading signal. In addition, it assumes that the initial capital is 1 dollar and it is valid to buy or sell fraction number of the HSI. The PPO is calculated using parameters that short term of 15 days EMA and long term of 45 days EMA.

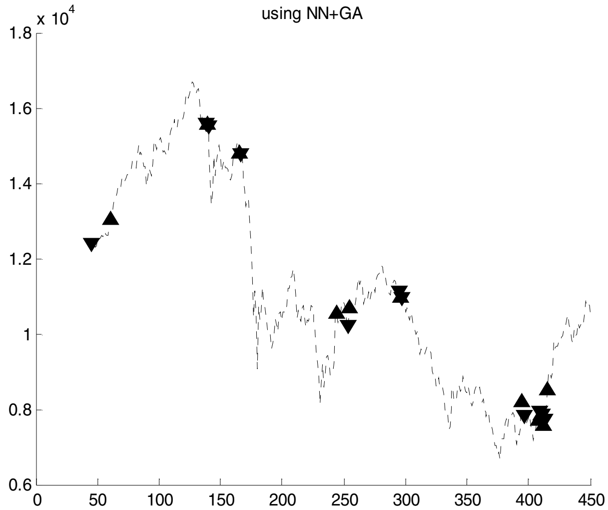

Fig.4. Trading signal of NN+GA trading system

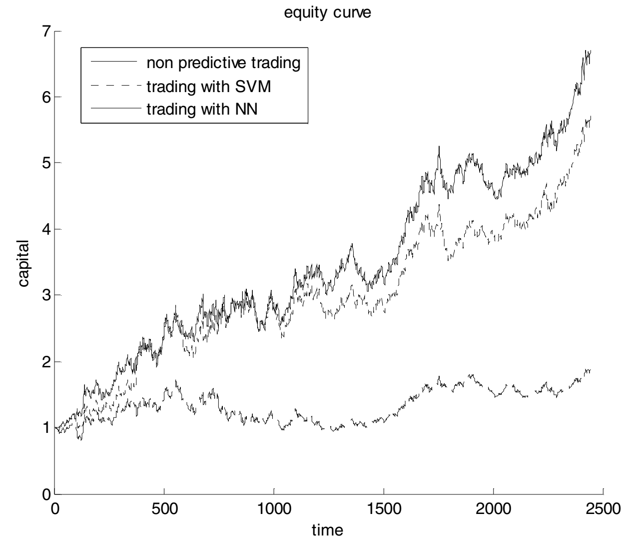

Fig.5. the equity curves of AI trading systems and conventional trading system

The testing data is shown in Fig 3 and Fig. 4 with buy and sell signals on the corresponding day. The equity curves of proposed intelligent trading systems are shown in Fig 5 with conventional trading system equity curve in contrast. The predictive NN+GA model achieves 6.91 times of original capital from 1997-1-28 to 2007-3-8, and SVM+GA model achieves 5.705 times of original capital, while in the mean time, a non-predictive trading system only achieves 2.064 for 1 dollar investment. In comparison, Huang et al. [10] uses hierarchical coevolutionary fuzzy system (HiCEFS) to achieve 5.781 times of original capital on Hang Seng Index in 2461 trading days.

In addition, through Fig 3 and Fig 4, it is obvious that prediction trading system would enter the market and exit the market earlier compared with trading system without prediction. However, using prediction has certain disadvantage. During non-trendy time, the proposed trading system may make wrong prediction and hence suffer some losses. For example, NN+GA trading system enters the market at day 61 at price 13030 and exit on the day 139 at price 15560, takes profit of 2530 points. On the other hand, for trading system without prediction, it enters the market at day 65 at price 13630 and exit at day 144 at price 13710, takes a profit of 80 points. That is the reason why the predictive model performs better than trading system without prediction. But during non-trendy market, such as around day 400, the trading system without prediction holds the position while the intelligent model made a wrong prediction. In this case, the investment incurred some losses.

Moreover, another important criterion to evaluate the trading system is the maximum drawdown (MDD). MDD is defined as the maximum cumulative loss from a market peak to the following trough [16]. A trading system with larger MDD will usually be difficult to execute when we count the human nature into consideration. Furthermore, larger MDD usually lead to fund redemptions since people would lose confidence in the performance of the trading system. Therefore, a smaller MDD is desirable for certain trading system and easier for people to execute the trading signal. MDD is calculated as equation(5) [17] :

$$\mathrm{D(T)} = \mathrm{SUP}_{t\in[0,T]}\left[\mathrm{SUP}_{s\in[0,t]}X(s) - X(t)\right] \quad (5)$$

The trading system using NN+GA suffers a MDD from 3.039 dollars to 2.191 dollars, which is 27.9% of the highest capital. Trading system with SVM+GA suffers a MDD from 2.843 dollars to 2.032 dollars, which is 28.5% of highest capital. In contrast, the trading system without prediction would have a MDD from 1.705 dollars to 1 dollar, which is 41.34% of the highest capital. Thus the NN+GA trading system reduced the risk involved. As it is shown in the previously mentioned example regarding the conventional trading system without prediction, the capital is back to original 1 dollar after 1276 trading days. This may shake people's will to follow this system.

IV. Conclusion and Future Work

In this paper, a predictive trading system is proposed to trade on real market data of Hong Kong Hang Seng Index. Neural network optimized by GA and support vector regression optimized by GA are implemented as predictive model in the trading system. The trading system mainly uses technical indicator price percentage oscillator (PPO) as trading rules. Hence the predictive model uses last 8 days PPO as input to predict future 5 days PPO, and based on predicted PPO to make trading decisions.

The testing period is 10 years, which is long enough to reduce the possibility of curve fitting. The proposed predictive trading system produces around 3 times more profits compared with conventional trading system without prediction.

Despite promising profits generated by the trading system, further improvements such as applying the system to other markets, or applying a better money management strategy can be considered as future research area. Furthermore, due to the randomness introduced by GA, neural network may not always be trained well enough every time. We shall study effective ways to assure reasonable performance for each training session.

REFERENCES

[1] E. F. Fama, "The Behavior of Stock Market Prices," *Business*, vol. 38, pp. 34-105, 1965.

[2] A. P. N. Refenes, A. N. Burgess, and Y. Bentz, "Neural networks in financial engineering: A study in methodology," *IEEE Transactions on Neural Networks*, vol. 8, no. 6, pp. 1222 - 1267, 1997.

[3] Kuan-Yu Chen and Chia-Hui Ho, "An Improved Support Vector Regression Modeling for Taiwan Stock Exchange Market Weighted Index Forecasting", *ICNN&B '05: International Conference on Neural Networks and Brain,* Volume 3, , 2005

[4] L. Cao and F. Tay, "Support Vector Machine with adaptive parameters in financial time series forecasting," *IEEE Transactions on Neural Networks*, vol. 14, no. 6, pp. 1506–1518, 2003.

[5] P.B. Patel and T. Marwala, "forecasting closing price indices using neural networks." In *International Conference on Systems, Man and Cybernetics*, pp. 2351-2356, Oct 8-11, 2006, Taipei, Taiwan.

[6] S.H. Lee, H.J. Kim and J.S. Lim, "forecasting short term KOSPI time series based on NEWFM," in *Advance Language Processing and Web Information Technology (ALPIT)*, pp. 303-307, July, 2007.

[7] B. Doeksen, A. Abraham, J. Thomas, and M. Paprzycki, "Real stock trading using soft computing models," in *Information Technology: Coding and Computing (ITCC)*, 2005, vol. 2, pp. 162-167.

[8] A.S. Chen, M. T. Leung, and H. Daouk, "Application of neural networks to an emerging financial market: Forecasting and trading the Taiwan Stock Index," *Computers and Operations Research*, vol. 30, no. 6, pp. 901–923, May 2003.

[9] K.K. Ang and C. Quek, "Stock Trading Using RSPOP: A Novel Rough Set-Based Neuro-Fuzzy Approach," *IEEE Transactions on Neural Networks*, vol. 17, no.5, pp. 1301 - 1315, 2006.

[10] H.M. Huang, M. Pasquier, and C. Quek, "Financial Market Trading System With a Hierarchical Coevolutionary Fuzzy Predictive Model," *IEEE Transactions on Evolutionary Computation*, vol. 13, no.1, pp. 56 - 70, 2009.

[11] H. Bandy, *Quantitative Trading Systems*, Blue Owl Press, 2007.

[12] Lipo Wang (ed.), *Support Vector Machines: Theory and Applications,* Berlin, Springer, 2005.

[13] C. W. Hsu, C. C. Chang, and C. J. Lin, *A practical guide to support vector classification,* Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, 2003. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf

[14] A. J. Smola and B. Scholkopf, "A tutorial on support vector regression," *NeuroCOLT2 Technical Report* NC2-TR-1998-030, 2003.

[15] P.N. Bahrun and M.N. Taib, "Selected Malaysia Stock Predictions using Artificial Neural Network," in *International Colloquium on Signal Processing & Its Applications (CSPA)*, 2009, pp. 428 – 431.

[16] M. Magdon-Ismail, A. Atiya, "Maximum Drawdown," *Risk Magazine*, Volume 17, Number 10, pp. 99-102, October, 2004.

[17] M. Magdon-Ismail, A. Atiya, A. Pratap, Y. Abu-Mostafa, "On the Maximum Drawdown of a Brownian Motion", *Journal of Applied Probability*, Vol. 41, no. 1, PP. 147-161, March, 2004.

[18] Chih-Chung Chang and Chih-Jen Lin, "LIBSVM : a library for support vector machines," 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

[19] Lipo Wang and Xiuju Fu, *Data Mining with Computational Intelligence,* Berlin, Springer, 2005.

[20] K.K. Teo, L. Wang, and Z. Lin, "Wavelet packet multi-layer perceptron for chaotic time series prediction: effects of weight initialization," Proceedings of International Conference of Computational Science ICCS 2001 Part II, San Francisco, CA, USA, May 28-30, 2001, LNCS 2074, pp. 310 -317, 2001.

[21] Kok Keong Teo, Lipo Wang and Zhiping Lin, "Wavelet multi-layer perceptron neural network for timeseries prediction," Proc. 10th International Conference on Computing and Information (ICCI 2000, Kuwait, 18-21 November 2000).