Neurocomputing 72 (2009) 2815–2823

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Exchange rate prediction using hybrid neural networks and trading indicators

He Ni [a,b], Hujun Yin [a,*]

[a] School of Electrical and Electronic Engineering, The University of Manchester, Manchester, M60 1QD, UK
[b] School of Finance, Zhejiang Gongshang University, HangZhou, PR China

ARTICLE INFO

Available online 16 April 2009

Keywords:
Time series modelling
Forex rate
Neural networks
Hybrid model

ABSTRACT

This paper describes a hybrid model formed by a mixture of various regressive neural network models, such as temporal self-organising maps and support vector regressions, for modelling and prediction of foreign exchange rate time series. A selected set of influential trading indicators, including the moving average convergence/divergence and relative strength index, are also utilised in the proposed method. A genetic algorithm is applied to fuse all the information from the mixture regression models and the economical indicators. Experimental results and comparisons show that the proposed method outperforms the global modelling techniques such as generalised autoregressive conditional heteroscedasticity in terms of profit returns. A virtual trading system is built to examine the performance of the methods under study.

1. Introduction

Prediction of stock prices and foreign exchange (FX) rates has been an active area of research in computational intelligence, signal processing and econometrics, ever since the collapse of the Bretton-Woods system in 1973. FX rates are one of the most important economic indices in the international monetary markets. The FX market is the largest and most lucrative financial markets [1]. Researchers have devoted a great deal of effort in order to find a good explanation (model) of the movement of FX rates between major currencies during the last decade [2,3]. It is widely known that FX rates are affected by many co-integrated, micro-, macro-economics, political and even psychological factors. So far, there is lack of modelling technique that can completely accommodate all the factors and their complex interactions. Therefore, modelling and forecasting FX rate movements poses a great challenge in today's global economy.

Before the emergence of neural network techniques, researchers found that traditional econometric and time series techniques could not reliably outperform the simplest random walk [2,3], which states that the market prices wander in a purely random and unpredictable way. The reason is partly the unrealistic assumptions that are applied to these classical methods. For instance, autoregressive moving average (ARMA) model is subject to the condition of stationarity of the time series. Autoregressive conditional heteroskedastic (ARCH) model introduced by Engle [4] and generalised autoregressive conditional heteroskedasticity (GARCH) model proposed by Bollerslev [5] are proven more effective in modelling dynamic FX rates. However, they have not served as a convincing tool in many practices and real-time tradings.

Artificial neural networks open up a new dimension and offer an alternative approach to FX rate forecasting in term of out-of-sample forecasting performance. Many studies have shown that adaptive neural networks significantly outperform linear models such as ARMA and native random walk model [6–10]. The commonly used techniques so far are multilayer perceptron (MLP), radial basis functions (RBFs) and recurrent networks. As a regressive method, support vector machines (SVMs), or support vector regressions (SVRs) [11], have been proposed as a good alternative to MLP in time series forecasting. SVRs are established on the theory of structural risk minimisation. SVRs can eventually achieve high generalisation performance in time series modelling. Training SVRs is equivalent to solving a linearly constrained quadratic problem; therefore the solution of SVRs is unique and globally optimal; whilst many other networks and learning algorithms always have the pitfall of being trapped to local minima. However, SVRs are computationally intensive, especially when dealing with large data sets, as solving linearly constrained quadratic problem requires large scale, complicated matrix calculations.

The main problem in modelling financial time series is that their dynamics and properties are changing with time. It is particularly true with FX rates due to the amount of inconstant "information flow". Empirical studies [12] show that

* Corresponding author.
E-mail addresses: nihe@mail.zjgsu.edu.cn (H. Ni), h.yin@manchester.ac.uk (H. Yin).

0925-2312/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2008.09.023

the distribution of daily returns[1] is approximately symmetric and leptokurtic (i.e., heavy tailed). One possible explanation for the heavy tailed distribution is that samples are independently distributed as a normal distribution whose mean and variance change over time. Many others argued that observed returns come from a mixture of various normal distributions [13,14]. Further studies on volatility [15] indicate that the average return of high frequency exchange rate data is negligible in comparison to its volatility[2] and the kurtosis is much higher than 3, the kurtosis value of a normal distribution. Therefore, it is not convincing and viable for a single model to capture the dynamics of the entire time series.

A potential solution is to employ the so-called "divide-and-conquer" principle to break down a large, complex problem into several smaller, or simpler ones [9]. The solutions of these simplified problems are then combined to produce the final solution to the original problem. The whole input space is split into several disconnected regions by a group of adaptive reference vectors. The prediction is thus made by the best fit local model in the corresponding region.

In addition to modelling and prediction techniques, there are many influential indicators (or trading rules), which have been extensively used by traders. A trading indicator is a function that returns either a "buy" or a "sell" signal for any given length of price history. The simplest technical trading indicators is moving average. It is also the basis of many other trend-following indicators. A short length moving average fits the raw data closer and implies more sensitive "buy" or "sell" signals. Whilst, a long length moving average fits the raw data looser and implies a more tolerant "buy" or "sell" signal. Though moving average is inherently a follower rather than a leader, it somehow reflects the underlying trend in many cases. Thus moving average can be used as a supplementary information when making predictions. The moving average indicator and time series data can be fed to the neural networks to capture the underlying "rules" of the movement in exchange rates [16]. There are also many more advanced technical indicators such as the moving average convergence/divergence (MACD) [17], relative strength index (RSI) [18], and Larry Williams rule [19], which are used widely in trading. They are all based on moving average and have certain ability to provide early warning of either oversold or overbought. However the predictions based on the time series itself and the information discovered by the technical indicators do not always agree and can be contradictory and inconsistent.

In order to handle the potential conflicts among different signals (time series predication and trading indicators), a genetic algorithm (GA) [20] has been used to integrate the information from these signals at each local model. Through a set of experiments, it has been found that the fusion weights optimised by GA are almost consistent. There are a few reports in the literature on applying GA to take into account trading indicators in analysis of financial time series. In Badawy's work [21], a GA is used to select appropriate trading indicators at a particular trading time. Allen and Karjalainen [22] have reported a indicator combination scheme that can earn excessive returns over a simply buy-and-hold strategy when transaction costs are under certain threshold. Chu et al. [23] have proposed an intelligent trading advisor based on several indicators and historical stock prices.

Furthermore, the mixture model alone is found to be sensitive to volatility and has poor performance in prediction. An empirical filter can be placed to let only those input vectors with mild volatilities to be used for prediction, while skipping the others (the corresponding action taken on actual trading is therefore hold, neither buy nor sell.)

In order to test the profitability of the proposed method and also for the purpose of comparison with other methods, a virtual trading system is built to stimulate real-time trading actions based on the information generated by the hybrid models with the consideration of volatility filter. The virtual trading system can produce a percent gain of accumulated profit.

The rest of the paper is organised as follows. In Section 2, the related regressive models are described. Section 3 presents a set of trading indicators. The proposed hybrid method is then presented in Section 4, followed by the experiments on exchange rate pair GBP to USD and comparison with other methods in Section 5. Finally, conclusions are given in Section 6.

2. Related work

2.1. Recurrent self-organising map (RSOM)

The self-organising map (SOM) [24] is a vector quantisation method that maps and clusters data of a high dimensional space onto a set of prototypes lying in a lower dimensional space while preserving topological relationships among the nodes. The updating rule for its reference weight m_r is

$$m_r(t+1) = m_r(t) + \alpha(t)h_{rv}(t)(x(t) - m_r(t)), \tag{1}$$

where $x(t)$ is the input, h_{rv} is the so-called neighbourhood function, r is the index of the updating node and v is the index of the winning node, $0<\alpha(t)<1$ is the learning rate, decreasing with time.

The SOM was proposed primarily to handle spatial data or vectors. In order for the SOM to process temporal signals or sequences such as time series, Koskela et al. [10] have extended SOM to a modified version, termed recurrent self-organising map (RSOM), where the previous activity of each neuron plays a part in its current role. The activity of the neuron is then

$$y_i(t) = (1-\beta)y_i(t-1) + \beta(x(t) - m_i(t)), \tag{2}$$

where $y_i(t-1)$ is the temporally leaked activity at node i from time $t-1$; β is the leaking coefficient, $0<\beta<1$. Large β corresponds to short memory while small β corresponds to long memory of past activities.

The previous input values are taken into account in describing the current activity of a node and thus the temporal relationship of the weight of the node between consecutive time points is somewhat captured.

2.2. Support vector regression

Support vector machines [11] can be used as a regressor, termed support vector regression, with three distinct advantages. First, SVR solves a risk minimisation problem by balancing the empirical error and a regularisation term, where the risk is measured by Vapnik's ε-insensitive loss function. Second, SVR usually estimates a set of linear functions defined in a high dimensional feature space. Third, SVR only uses a limited number of support vectors.

The regression function can be estimated by minimising a regularised risk function

$$\text{minimise} \quad \frac{1}{2}\|w\|^2 + C\frac{1}{l}\sum_{i=1}^{l} L_\varepsilon\ , \tag{3}$$

[1] A simple logarithm difference transform, $x'_n = \ln x_{n+1}/x_n$, here the x_n is the scalar value of the time series at time n.

[2] It is most frequently referred to the standard deviation of the change in value of a financial instrument with a specific time horizon.

$$L_\varepsilon = \begin{cases} |y_i - w * \phi(x_i) - b| - \varepsilon, & |y_i - w * \phi(x_i) - b| \geqslant \varepsilon, \\ 0 & \text{otherwise,} \end{cases} \quad (4)$$

where w is a weight vector which is used to determine the maximum margin hyperplane, the term $\|w\|^2$ is called the regularised term, which should be as flat as possible. The second term is the empirical error measured by Vapnik's ε-insensitive loss function. C is the regularisation constant.

2.3. Parametric method: GARCH

The GARCH model [5], loosely speaking, can be thought of heteroscedastic time-varying variance (i.e., volatility). It is conditionally dependent on the observations of the immediate past. Autoregressive describes a feedback mechanism that incorporates past observations into the present. GARCH then is a mechanism that includes past variances in the explanation of future variances. More specifically, GARCH is a time series model that uses past variances to forecast future variances.

GARCH models are generalised ARCH models, where the conditional variance at time t, σ_t^2, depends on earlier variances. That is, a GARCH(g^p, g^q) process is given by

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{g^q} \alpha_i \varepsilon_{t-i}^2 + \sum_{i=1}^{g^p} \beta_i \sigma_{t-i}^2, \quad (5)$$

when $g^p = 0$, GARCH becomes ARCH(g^q), and ε_t is white noise.

GARCH model is one of the most popular methods used in modelling financial time series. Various recent studies show successful applications of the GARCH model in capturing the dynamics of stock indexes [27,28], and exchange rates [29,30].

3. Influential trading indicators

The key for making profit from market trends is to take good advantage of the trend. The first step to profiting from both short- and long-term trends is to understand and identify them. The next step is to employ a disciplined trading strategy that is specific to the trends. There are a number of influential trading indicators widely recognised and used by traders around the world. These rules, though fairly straight forward to obtain,[3] have been proven successful in the trading history.

3.1. Exponential moving average (EMA)

The Dow–Jones theory [25] categories the trend into three different levels, primary, secondary and minor. The three levels of trend have been described as tides, waves and ripples of the sea. The most important indicator, moving average, can smooth out some irregular moves that exist in trading days, however it always lags behind the current moment. According to the Dow–Jones theory, the exponential moving average gives more emphasis on the days closer to the trading day than those further away. In order for the EMA to be used in the temporal modelling by the RSOM, the EMA is also fed into the hybrid network along with the exchange rate. Therefore, the whole time series is divided into small segments, each containing two elements: the movement of the actual exchange rate and 10 days of exponential moving average.[4] If the actual price is denoted as 'ACT' and exponential moving average as 'EMA', the distance can be calculated as

$$d_i(x, q) := w_{ACT} d_i^{ACT}(x^{ACT}, q^{ACT}) + w_{EMA} d_i^{EMA}(x^{EMA}, q^{EMA}), \quad (6)$$

where i is the index of the local model, q is the reference vector, which includes two parts q^{ACT} and q^{EMA} and so does x. w_{EMA} is the weighting on the distance calculated by exponential moving average and the w_{ACT} is the weighting on the distance calculated by the exchange rate. Various experiments point to the best combination weights as $w_{ACT} = 0.6$, $w_{EMA} = 0.4$. $w_{ACT} > w_{EMA}$ is apparently due to that fact that more contribution is from the exchange rate for "recognising" the right patterns. The reason that EMA is not directly used as a trading indicator is that its effect has been already represented by MACD, which focuses on the difference between long-term and short-term moving averages. In addition, EMA is also complementary to the raw FX rate, since it contains more trend information than the rate itself. Therefore a group of FX rate vectors, where a particular local model is built on, not only have the similar distribution but also are in similar trends.

[3] Most of the widely used trading indicators are built upon the exponential moving average with some further, simple mathematical operations.

[4] The 10 trading days (equivalent to two trading weeks) moving average is proven to be the best length by experiments and also has been accepted as the most popular or default indicator.

Table 1
Success rates of various trading indicators based on the average prediction performance on daily closing price of USD vs GBP.

Indicators	Oversold signal (%)	Overbought signal (%)
MACD	55.67	49.74
RSI	62.29	65.14
Williams %R	54.81	56.66

3.2. Other trading indicators: MACD, RSI, Williams %R

The other trading rules or empirical rules are also used for identifying overbought and oversold signals. The adopted rules include three widely used trading indicators: moving average convergence/divergence, relative strength index and Larry Williams (Williams %R). MACD shows the difference between a fast and slow exponential moving average of closing prices. During the 1980s MACD was proven to be a valuable tool for traders [17]. It is also used as a good monitoring tool in the modern era. RSI is another popular momentum oscillator developed by J. Welles Wilder and detailed in his book "New Concepts in Technical Trading Systems" [18]. The principle is that when there is a high proportion of daily movement in one direction it suggests an extreme, and prices are likely to reverse. Developed by Larry Williams, Williams %R is a technical analysis oscillator showing the current closing price in relation to the high and low of a certain period of time n, usually 14 days [19]. It is used to determine market entry and exit points and is normally used in the stock market.

When applied to the closing prices of daily exchange rates between USD and GBP, these three indicators in general outperform other indicators. Their individual successful rates are listed in Table 1.

4. Proposed hybrid model

In this section the proposed hybrid method is described. It consists of a group of local regressive models, each having different probability densities. The mixture of local models can be interpreted as a hybrid model that is contributed by several underlying sources (local models), each generating the data separately according to its own distribution. When building such a group of local models, each data item is assigned to a source that

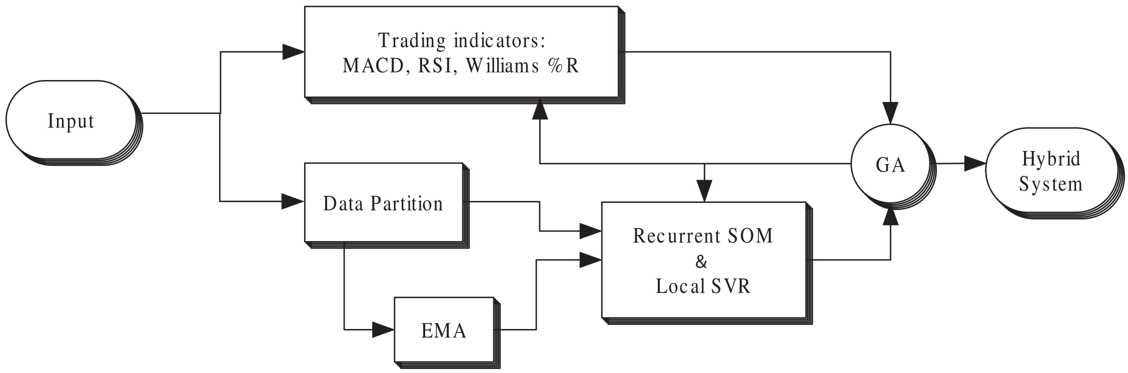


Fig. 1. The operation diagram of the proposed hybrid method.

is most likely to have generated it. A GA is then applied to further integrate trading indicators with the local model output.

The RSOM and local SVR form a two-stage architecture. The idea is to firstly partition the whole input space into several partially- or non-overlapping regions with the RSOM and then SVRs are used to fit local data and to produce forecasting results on those partitioned regions. These regression models are the local models.

Since the signals generated by various indicators may not always agree, it is necessary to have a mechanism to handle the conflicts when they inevitably occur. Here we use a weighting factor for each indicator, initially set to their average performance as shown in Table 1. The precise contributions of these indicators are then learnt by the GA to search from a small area around the initial weights.

The trading decisions shown by a set of N rules are defined as $e_i,\ i = 1, 2, \ldots, N$ taking values within $[0, 1]$ for "buy" or $[-1, 0]$ for "sell" signals, respectively. When $e_i = 1$, the rule i signals a strong buy; $e_i = -1$ means a strong sell, and $e_i = 0$ indicates "hold" or do nothing. The weighting factors learnt by the GA represent the strength or reliability of "buy" or "sell" signals generated by the indicators. The overall decision T is made by combining the signals from these indicators, as well as the prediction, p, generated by the difference between predicted one-step ahead price and the observed price

$$p = sign(\hat{x}_{t+1} - x_t), \tag{7}$$

$$T = w_0 p + \sum_{i=1}^{N} w_i e_i, \tag{8}$$

where w_i is the weighting factor learnt by the GA on indicator i; and w_0, whose value is proportional to the confidence level[5] of a local regressive model, is the weighting on predication. The operation diagram of the proposed prediction method is shown in Fig. 1.

GA is a class of search, adaptation, and optimisation techniques based on the principle of natural evolution [26]. It evolves a population of candidate solutions and evaluates the quality of each solution by pre-defined problem-specific fitness function. Candidate solutions are represented by character strings (often binary). The fitness function measures the quality of solutions.

GA is particularly useful for the problems with a non-differentiable or discontinuous objective function, to which gradient-based methods are not applicable. It also does not require much prior knowledge. Most traders rely on experience and empirical knowledge or strategies for selecting and using trading rules. Most trading rules signal "buy" or "sell" in discrete forms. Therefore GA turns out to be an ideal optimisation tool to use, though it has limitations such as excessive computation.

Instead of looking for new trading rules as it was the case in [22], the GA is applied to obtain a combinations of various existing trading rules. That is, the signals generated by trading rules are weighted according to their contribution learnt by the GA, while the structure of each trading rule is intact. The advantage of this approach is to retain the original mechanism of trading rules, most of which are indeed summaries of several decades of successful experiences. The contribution for each candidate ranges from -1 to 1, where the value is proportional to the volume should be invested. Promising solutions in a generation are chosen by "roulette wheel" (selection) based on their fitness. More sensible solutions are constructed (crossover) from their chosen parents. Some random changes are arbitrarily made on a certain proportion of new solutions (mutation) in order to prevent the optimisation from being trapped to local minima.

The fitness function is defined on the profit made over a pre-determined trading strategy. The strategy counts the effects of FX predication, signals from various trading rules, spread cost, and leverage. Signals from regression models and indicators are quantified with value proportional to its learnt weighting factor and the sign determining whether it is a "sell" or a "buy". In case of non-confliction situation, where all signals give the same "buy" or "sell" sign, the sum of the weighted signals is the overall decision. Otherwise the dominating signals, which have large weightings, may determine the overall sell or buy, however the overall signal is weakened by the opposite signals. The value of fitness function is defined by the accumulated profits after a given number of trades. Therefore, the weights found by the GA are optimal in this specific sense, that is, to maximise a particular profit measure. The training procedure is outlined as follows:

Training procedure of the proposed hybrid system

1. Window the time series (FX rate and its EMA) into segments. Separate the entire data set into training (regression model training and hybrid model training), validation, testing parts.
2. Recursively train the recurrent SOMs with the training data. Use the trained prototypes to partition the input space into different regions. Repeat step 2 until validation set confirms the optimal partition.
3. Train SVR-based one-step predictive models in the partitioned regions by obtaining the most adequate SVR based on the training segments in each partitioned region.
4. Apply the trading rules on the training data and collect the signals from the rules as well as the prediction on the same training data for the GA.
5. Create a set of random weights (chromosomes) for the prediction and indicator signals as the first generation.
6. Compute the accumulative profits as the fitness. Create new chromosomes by crossover operation on selected chromosomes (probabilistically based on their fitness values). Repeat the crossover until a new generation is generated. Randomly mutate or flip some bits of the chromosomes.

[5] It is defined as the standard deviation of training vectors of the local model.

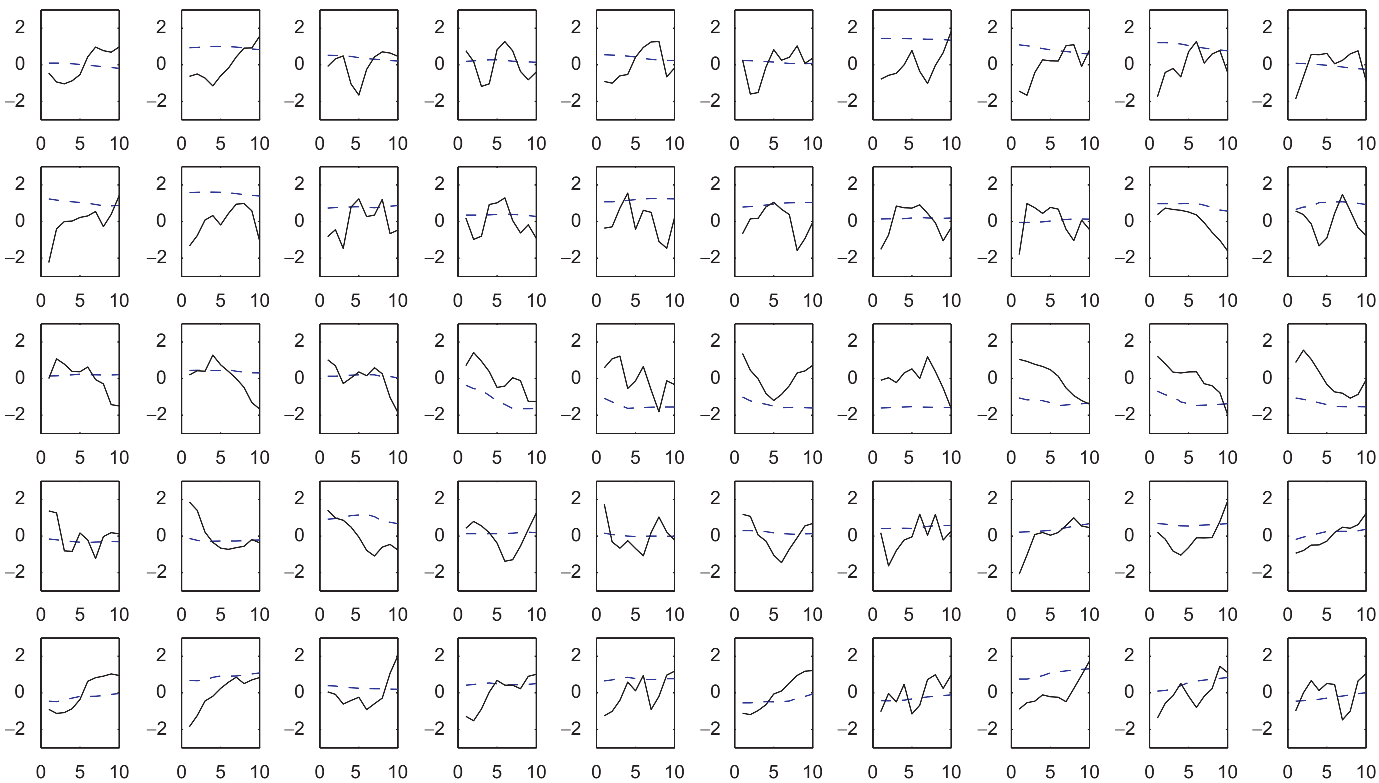


Fig. 2. An illustration of reference vectors at SOM units. Solid line represents normalised price curve and dashed line represents corresponding exponential moving average curve.

7. Repeat step 6 until the maximum number of generations is reached or the process is stopped by the validation.
8. Apply the learned weights to each sub-regressive model. The hybrid model is now ready for testing its profitability.

5. Experiments and comparison

This section presents experiments of the proposed method on the FX rate forecasting. The proposed hybrid method is shown to be able to capture the dynamics of highly non-linear, non-stationary time series, as well as to explore the supplemental effects of trend indicators.

A set of 4000, 10-dimensional data samples is used in this experiment. Each data sample presents 10 consecutive trading days' closing price of USD vs GBP. This window length, 10, is chosen by extensive experiments. We chose this currency pair because it was found as one of the most efficient exchange pairs [16,31]. All the data samples are normalised as well as paired with 10-day exponential moving averages, which are also normalised by the mean and the standard deviation of original data samples. The traditional price-to-return conversion is not applied because it cannot always make the time series stationary [32] and it will also lose the useful information carried by the exponential moving average, which has been proven valuable in comparative studies. 50% of the data is used for training the RSOM and SVR based local regression model, 30% of data is used for training the hybrid system based on GA, 10% of data is used for validation and the rest 10% is used for out-of-sample performance test.

A tree alike growing lattice is adopted for the RSOM for partitioning the data to fit by local regression models. Such structure can fit better the dynamics and would not be highly restricted by the pre-fixed topology of a lattice [33]. Every local model is represented by a reference vector (prototype) that contains a normalised closing price curve and a corresponding exponential moving average curve, as shown in Fig. 2.

Every training vector is allocated to a particular local model by the closest reference vector. Fig. 3 shows all the reference curves and data segments mapped to them. In Fig. 3, there is a clear "diversity" among local models. The area of shadows for a local model is proportional to the value of the standard deviation of the training vectors belonging to that local model. It is reasonable to believe that the smaller the shadow is, the more reliable the prediction will be, provided that each local model contains similar number of training vectors. Accordingly, more weighting should be given to the prediction by the regression than the signals by the trading indicators.

Fig. 4 further illustrates that the confidence level which has been made proportional to the variance of the corresponding local model has effectively extended the accuracy of the prediction. From the upper plot of Fig. 4, it is shown that the actual prices are mostly enclosed by the prediction with upper lower confidence boundaries. The lower plot shows the variance and the number of training vectors in a number of local models. The confidence level is calculated by those two values: small variance and large number of vectors corresponding to high confidence level of the local model, while big variance and small number of vectors corresponding to low confidence level.

Fig. 5 shows the signals generated by different indicators. The signal can be read by the shape of the triangle, △ is a oversold signal and ▽ is an overbought signal.

The indications made by three rules are integrated into the resulting curve (prediction) by the hybrid method trained by GA. In each local model, a population of 20 candidates (weighting factors) are set as chromosomes and the maximum number of generations is set to 400, which is proven to be more than enough by random tests. The GA training can be stopped either by a

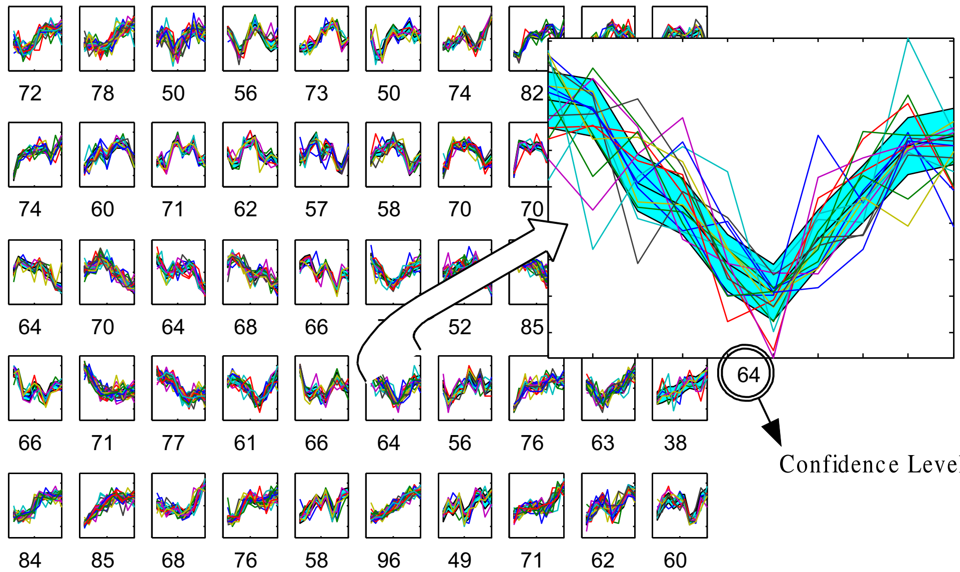


Fig. 3. Reference vectors and data segments (price curves) mapped to them. The shadows represent the area within one standard deviation away from the reference vectors. The number under each sub-plot is proportional to the confidence level.

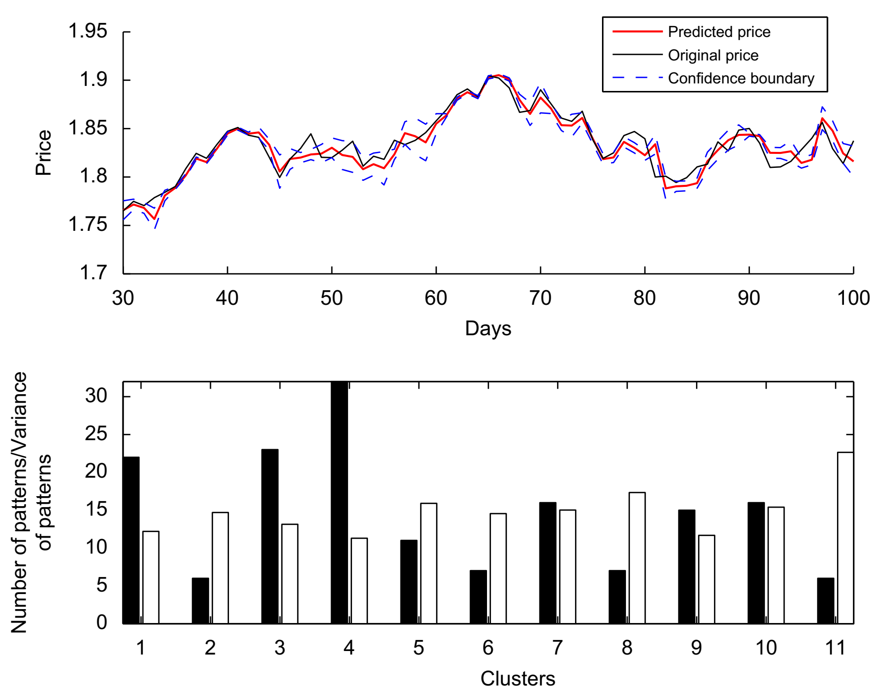


Fig. 4. The predication of RSOM based local SVR models (for illustrate purpose only part of results are plotted here): upper plot is the predication with confidence level; lower figure is the histogram of training vectors (black column) and variance (white column) of each cluster.

validation process or by reaching the maximum generations. The effects of the training is shown in Fig. 6. The prediction is conducted by firstly identifying a local hybrid model and then by summing up the weighted effects from regression models and indicators. The average rates of correct prediction made by SOM+SVR, SOM+MLP, and GARCH and the proposed hybrid method are 62.23%, 59.81% and 61.90% and 68.53%, respectively.

A virtual trading system has been designed to measure the performance of the proposed hybrid system against other models and methods in terms of percentage gain over a period. The flowchart of the system is shown in Fig. 7.

Inputs to the system are spot FX rate, total holding assets and the trading volume, which is initialised to zero. Firstly the volatility of input FX rate vector is measured and compared to its historical average. High volatility (e.g. five times of its historical average), which may be caused by, for instance, unexpected crisis or sharp interest rates changes, is deemed as unsuitable for trading. Therefore the trading volume is set to zero and consequently no profit will be made on those inputs. The next step is to calculate the difference between the FX price and its predication. Then the trading volume is looked up from an action table, which consists of a range of possible differences between price and its predication, and corresponding trading volume. For

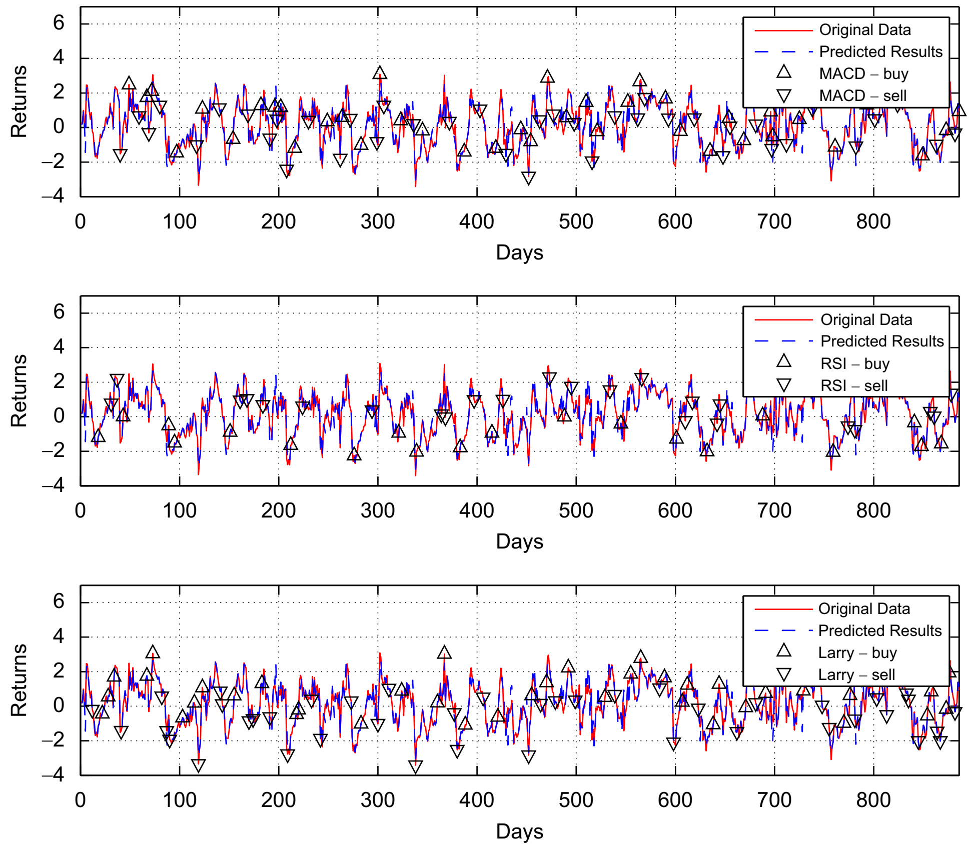


Fig. 5. Effects of various indicators in prediction on returns: △ indicates an upward trend, while ▽ indicates a downward move in the future.

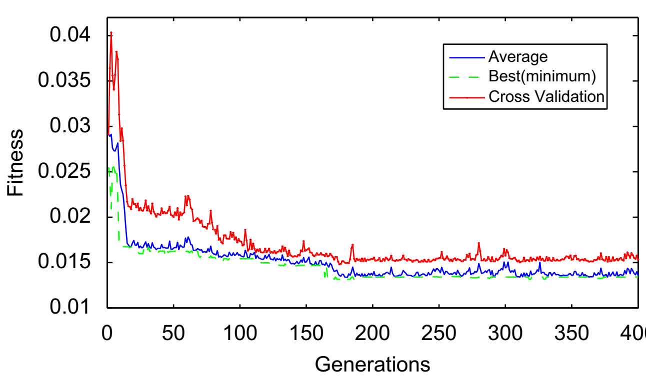


Fig. 6. GA training process. The fitness function is defined by the profitability of a certain trading strategy with learned weights. The dashed line is the unit with the best performance in the population and the dot line is the cross validation.

any price and its predication pair, there is a specific trading volume in the action table. The trading volume in the table is shown as the percentage of the total tradable assets. For example, in case of predicted sharp increase of FX rate the corresponding action is to invest 30% of the trading assets[6] to buy in. The trading volume is further tuned, as shown in Eq. (8), by trading indicators weighted by the GA. Positive volume presents a buy action (sell base currency), negative volume presents a sell action (buy base currency). At the final stage, the profit or lost is added to, or subtracted from, the current holding assets and the measurable profit is representable by a percentage gain, which is the percentage change of holding assets along a number of consecutive trading days.

The profit curves, plotted in Fig. 8, show the profits made by different methods under study. The proposed method consistently outperforms the rest, especially in medium and long terms.[7]

6. Conclusions

In this paper a hybrid model is proposed for exchange rate modelling and forecasting, based on recurrent SOMs, support vector regressions and several common trading rules. RSOM is used to partition the nonstationary time series into coherent groups. Then SVRs are employed to model grouped samples and prediction is made by the best fitting local models. A genetic algorithm is adopted to further integrate the trading rules with the local regressive models. The final hybrid model consists of a number of local models, each having a set of learnt weighting factors that are used to combine its prediction and indicator signals. The performance of the hybrid system is measured by its profitability. The results of the proposed method are improved

[6] Trading assets present the tradable assets, including the total sum of the base currency in case of buy and the term currency in case of sell.

[7] Commission charges and other administrative costs are not taken into account.

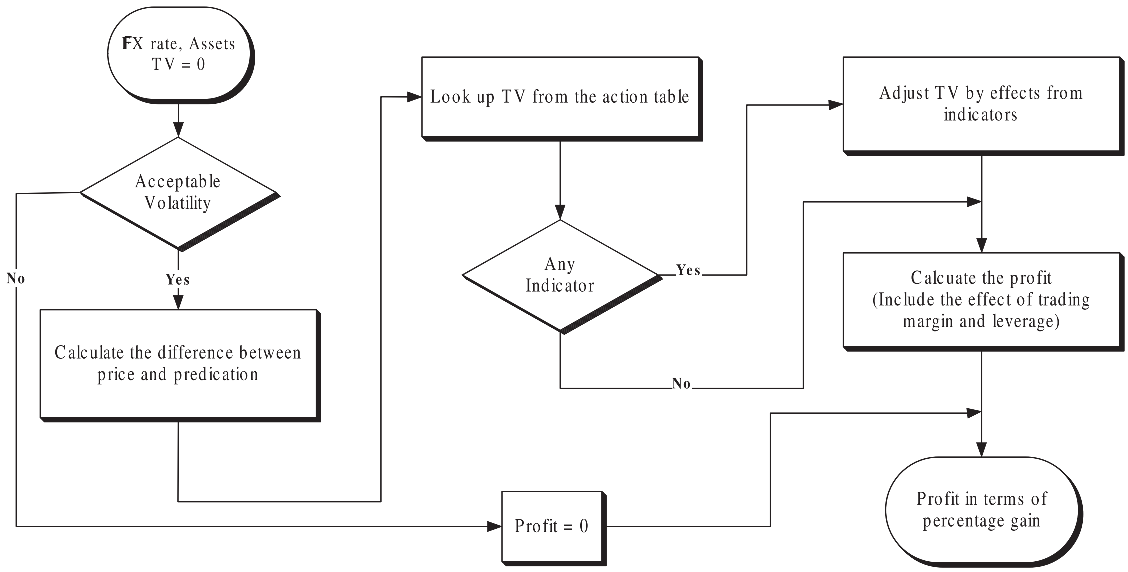


Fig. 7. The flow chart of the virtual trading system. TV stands for trading volume.

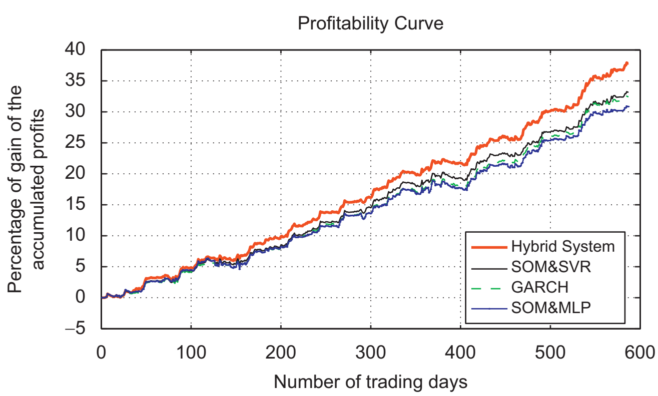


Fig. 8. Profits made by various approaches. The x axis represents the time span and the y axis represents the percentage gain of the accumulated profits.

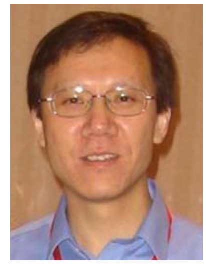
over global models such as GARCH and other methods. The optimal performance of the proposed hybrid system is dependent on the specific fitness measure and the quality/quantity of trading rules adopted. The proposed hybrid method is effective in combining both quantitative and qualitative data in forecasting financial time series movement.

References

[1] R. Baillie, P. McMahon, The Foreign Exchange Market: Theory and Econometric Evidence, Cambridge, UK, 1989.
[2] R.A. Meese, K. Rogoff, Empirical exchange rate models of the seventies: do they fit out of sample?, Journal of International Economics 14 (1983) 3–24.
[3] L. Kilian, M.P. Taylor, Why is it so difficult to beat random walk forecast of exchange rates, Journal of International Economics 60 (2003) 85–107.
[4] R. Engle, Autoregressive conditional heteroskedasticity with estimates of the variance of United Kingdom inflation, Econometrics 50 (1982) 987–1007.
[5] T. Bollerslev, Generalized autoregressive conditional heteroskedasticity, Journal of Econometrics 31 (1986) 307–327.
[6] T.H. Hann, E. Steurer, Much ado about nothing? exchange rate forecasting: neural networks vs. linear models using monthly and weekly data, Neurocomputing 10 (1996) 323–339.
[7] A.-S. Chen, M.T. Leung, Regression neural network for error correction in foreign exchange forecasting and trading, Computers and Operations Research 31 (2004) 1049–1068.
[8] V. Kodogiannis, A. Lolis, Forecasting financial time series using neural network and fuzzy system-based techniques, Neural Computing and Applications 11 (2002) 90–102.
[9] L.J. Cao, Support vector machines experts for time series forecasting, Neurocomputing 51 (2002) 321–339.
[10] T. Koskela, M. Varsta, J. Heikkonen, K. Kaski, Time series prediction using recurrent SOM with local linear models, Research reports B15, Helsinki University of Technology, Laboratory of Computational Engineering, 1997.
[11] V. Vapnik, The Nature of Statistical Learning Theory, Springer, NY, 1995.
[12] F. Diebold, Empirical Modeling of Exchange Rate Dynamics, in: Lecture notes in economics and mathematical systems, Springer, Berlin, 1988.
[13] P.K. Clark, A subordinate stochastic process model with finite variance for speculative price, Econometrica 41 (1973) 135–155.
[14] B. Mandelbrot, H. Taylor, On the distribution of stock price differences, Operations Research 15 (1969) 1057–1062.
[15] B. Zhou, High-frequency data and volatility on foreign-exchange rates, Journal of Business and Economic Statistics, American Statistical Association 14 (1996) 45–52.
[16] J.T. Yao, C.L. Tan, A case study on using neural networks to perform technical forecasting of FOREX, Neurocomputing 34 (2000) 79–98.
[17] S.P. Stawicki, Application of financial analysis techniques to vital sign data: a novel method of trend interpretation in the intensive care unit, OPUS 12 Scientist 1 (2007) 14–16.
[18] J.W. Wilder, New Concepts in Technical Trading Systems, Trend Research, 1978.
[19] L. Williams, Trade Stocks Commodities with the Insiders: Secrets of the COT Report, Wiley Trading, 2005.
[20] D.B. Fogel, Evolutionary Computation: The Fossil Record, IEEE Press, New York, 1998.
[21] F.A. Badawy, H.Y. Abdelazim, M.G. Darwish, Genetic algorithms for predicting the Egyptian stock market, in: Proceedings of ITI Third International Conference on Information and Communication Technology, Enabling Technologies for the New Knowledge Society, 2005, pp. 109–122.
[22] F. Allen, R. Karjalainen, Using genetic algorithms to find technical trading rules, Journal of Financial Economics 51 (1999) 245–271.
[23] S.C.W. Chu, H.S. Ng, K.P. Lam, Intelligent trading advisor, in: Proceedings of 2000 IEEE International Conference on management of innovation and technology, vol. 1, 2000, pp. 53–58.
[24] T. Kohonen, Self-Organizing Maps, Springer, Berlin, 1997.
[25] J. Murpy, Technical Analysis of the Financial Markets, Prentice-Hall, London, 1998.
[26] J.H. Holland, Outline for a logical theory of adaptive systems, Journal of the Association for Computing Machinery 3 (1975) 297–314.
[27] E. Jondeau, M. Rockinger, The Copula-GARCH model of conditional dependencies: an international stock market application, Journal of International Money and Finance 25 (2006) 827–853.

[28] T. Ane, L. Ureche-Rangau, Stock market dynamics in a regime-switching asymmetric power GARCH model, International Review of Financial Analysis 15 (2006) 109–129.
[29] C.M. Hafner, H. Herwartz, Volatility impulse responses for multivariate GARCH models: an exchange rate illustration, Journal of International Money and Finance 25 (2006) 719–740.
[30] C. Hartz, S. Mittnik, M. Paolella, Accurate value-at-risk forecasting based on the normal-GARCH model, Computational Statistics and Data Analysis 51 (2006) 2295–2312.
[31] E.E. Peters, Chaos and Order in the Capital markets: A New View of Cycles, Prices, and Market Volatility, Wiley, NY, 1991.
[32] C.W.J. Granger, Time series analysis, cointegration, and applications, The American Economic Review 94 (2004) 421–425.
[33] H. Ni, H. Yin, Recurrent self-organising maps and local support vector machine models for exchange rate prediction, in: Lecture Note on Computer Science, vol. 3973, 2006, pp. 504–511.

He Ni has recently been awarded Ph.D. degree, for which he worked during 2004–2008 in the School of Electrical and Electronic Engineering, the University of Manchester. His research interests include financial time-series forecasting, neural networks and machine learning. He obtained a BEng degree in Electronic Engineering from Xidian University and a M.Sc. degree in Automatic Control and System Engineering from the University of Sheffield in 2001 and 2003, respectively. He has taken up a lectureship at Zhejiang Gongshang University since September 2008.



Hujun Yin is a Senior Lecturer (Associate Professor) at The University of Manchester, School of Electrical and Electronic Engineering. He received BEng and MSc degrees from Southeast University and PhD degree from University of York in 1983, 1986 and 1996, respectively. His main research interests include neural networks, self-organising systems in particular, pattern recognition, and bio-/neuro-informatics. He has studied, extended and applied the self-organising map (SOM) and related topics (principal manifolds and data visualisation) extensively in the past ten years and proposed a number of extensions including Bayesian SOM and ViSOM, a principled data visualisation method. He has published over 100 peer-reviewed articles in a range of topics from density modelling, text mining and knowledge management, gene expression analysis and peptide sequencing, novelty detection, to financial time series modelling, and recently decoding neuronal responses.

He is a senior member of the IEEE and a member of the UK EPSRC College. He is an Associate Editor of the IEEE Transactions on Neural Networks and a member of the Editorial Board of the International Journal of Neural Systems. He has served on the Programme Committee for more than thirty international conferences. He has been the Organising Chair, Programme Committee Chair, and General Chair for a number of conferences, such as 2001 International Workshop on Self-Organising Maps (WSOM'01), International Conference on Intelligent Data Engineering and Automated Learning (IDEAL) (2002–2008), 2006 International Symposium on Neural Networks (ISNN'06). He sits on the Steering Committee of the WSOM series. He has also guest-edited a number of special issues on several leading international journals. He has received research funding from the EPSRC, BBSRC and DTI. He has also been a regular assessor of the EPSRC, the BBSRC, the Royal Society, the Hong Kong Research Grant Council, Netherlands Organisation for Scientific Research, and Slovakia Research and Development Council.