Comput Econ (2015) 45:49–89
DOI 10.1007/s10614-013-9407-6

Hybrid Method of Multiple Kernel Learning and Genetic Algorithm for Forecasting Short-Term Foreign Exchange Rates

Shangkun Deng · Kazuki Yoshiyama · Takashi Mitsubuchi · Akito Sakurai

Accepted: 7 November 2013 / Published online: 20 November 2013

Abstract Our proposed prediction and learning method is a hybrid referred to as MKL-GA, which combines multiple kernel learning (MKL) for regression (MKR) and a genetic algorithm (GA) to construct the trading rules. In this study, we demonstrate that the evaluation criteria used to examine the effectiveness of a financial market price forecasting method should be the profit and profit-risk ratio, rather than errors in prediction. Thus, it is necessary to use a price prediction method and a trading rules learning method. We tested the proposed method on the foreign exchange market for the USD/JPY currency pair, where the features used for prediction were extracted from the trading history of the three main currency pairs with three different short-term horizons. MKR is essential for utilizing the information contained in many of the

Electronic supplementary material The online version of this article (doi:10.1007/s10614-013-9407-6) contains supplementary material, which is available to authorized users.

This research has been partially supported by a fund "Keio Leading-edge Laboratory of Science & Technology Research Grant" and by the Global-COE program (Symbiotic, Safe and Secure System Design) of Keio University, Japan. This journal is an extended version of conference paper (Deng and Sakurai 2011) in the Proceedings of Artificial Intelligence and Applications 2011 (AIA 2011).

S. Deng · K. Yoshiyama · T. Mitsubuchi · A. Sakurai (✉)
Graduate School of Science and Technology, Keio University,
3-14-1 Hiyoshi, Kohoku-ku, Yokohama 223-8522, Japan
e-mail: sakurai@ae.keio.ac.jp
URL: http://www.sakurai.comp.ae.keio.ac.jp/

S. Deng
e-mail: dsk8672@gmail.com

K. Yoshiyama
e-mail: cloud-cluster@z8.keio.jp

T. Mitsubuchi
e-mail: mitsubuchi@ae.keio.ac.jp

features derived from different information sources and for various representations of the same information source. The GA is essential for generating trading rules, which are described using a mixture of discrete structures and continuous parameters. First, the MKR predicts the change in the exchange rate based on technical indicators such as the moving average convergence and divergence of the three currency pairs. Next, the GA generates a trading rule by combining the results of the MKR with several commonly used overbought/oversold technical indicators. The experimental results show that the proposed hybrid method outperforms other baseline methods in terms of the returns and return-risk ratio. In addition, the kernel weights employed for different currency pairs and the different time horizons used in the MKR step, as well as the trading strategy generated in the GA step, should be beneficial during actual trading.

Keywords FX trading · Financial prediction · Multiple kernel learning · Genetic algorithm · MKL-GA hybrid method · Technical indicators

Abbreviations

FX	Foreign exchange
SVM	Support vector machine
SVR	Support vector regression
MKL	Multiple kernel learning
MKR	Multiple kernel regression
GA	Genetic algorithm
SMA	Simple moving average
EMA	Exponential moving average
RSI	Relative strength index
WPR	Williams percent range
BIAS	Bias ratio
MACD	Moving average convergence and divergence
RMSE	Root mean square error
ANN	Artificial neural network

1 Introduction

The foreign exchange (FX) market is the most influential of the financial markets, especially in the current era of a rapidly expanding global economy, because of the vast number and values of its transactions, and the range of currencies that can be traded. The FX market is the one whose influence cannot be avoided in life, not only for people those engaged in the financial filed, but also for entrepreneurs and policy-makers. If it was possible to predict the market to a certain degree, investors could hedge their assets or policy-makers could amend their policies. Thus, prediction is an important research area.

A wide variety of information affects price movements in the market, but not all of these to be considered if the market is efficient. The market or market participants will combine all of the information instantly if the market is efficient, so prices will be fixed to their proper values with only probabilistic fluctuations. Opinions differ about whether the financial markets of economically developed countries are really efficient. If a market is efficient, it is not possible to make correct predictions of the prices in the market based on the price history. The efficiency is realized only if rational traders, instantaneously and correctly react to a change in a manner that optimizes the outcomes, which adjusts the prices instantaneously to their proper positions. However, real humans are not necessarily rational and have limited capacities; hence, they contribute to inefficiencies in the markets in which they participate. An abundance of research suggests that there is inefficiency in FX markets. A previous study concluded that profits based on the moving average crossover rule and daily data declined from 1971 into the 2000s (Olson 2004), whereas studies of intraday data have produced mixed results (Schulmeister 2007; Dempster and Jones 2002; Gencay et al. 2003; Riccardo et al. 1997; Neely and Weller 2003). Thus, if the market has not been efficient since 2000, trading rules based on moving averages and Alexander's filter have not been able to exploit the inefficiency. In this study, we focused on simulated intraday trading in the 2000s using more sophisticated trading rules and a more versatile prediction function with more endogenous feature variables (explanatory variables) to determine whether trading based on these rules would have yielded excess profits. We assume that the market is largely efficient so exogenous information is absorbed into market behavior, despite the slight inefficiencies attributable to the irrational behavior of traders. Therefore, the rates themselves and the indexes calculated from these rates were the only values that we needed to consider as explanatory variables.

Many researchers have failed to obtain excess profit from the markets since 2000, possibly because of the long period required for the optimization or estimation of the parameters required by the trading rules. A longer period of optimization (in-sample estimation) produces better trading rules, but only if the time series is stationary. However, information creation, transmission, and sharing are changing more rapidly than ever because of the rapid development of the internet, proprietary networks of markets, and powerful computers, which means that any market trends have short periods and they are harder than ever to find.

We used moving windows in this study, but they were much narrower in width than those used in other studies (Meese and Rogoff 1983; Pesaran and Allan 1995; Hann and Steurer 1996; Yeh et al. 2011). The windows are the periods used for in-sample estimations and out-of-sample tests. Unlike rolling regression, the in-sample estimation period rolls forward along with the out-of-sample test period. The lengths of the in-sample estimation period and out-of-sample test period were fixed throughout our experiments. In this study, the learning period was 1000 trading hours (around 8 weeks) and the out-of-sample test period was 500 trading hours (around 4 weeks).

We adopt the genetic algorithm (Goldberg 1989), a class of meta-heuristic algorithms, to optimize the trading rules. The trading rules used by many researchers are based on moving averages or Alexander's filter, where the parameters are estimated during the in-sample estimation period. Our trading rules were more complex and there was no closed form for estimating the parameters, so we had no option other than "training" our rules using a machine learning technique. Thus, we had to optimize specific objective functions by varying the parameters. In this study, we focused on the maximization of the profits and the profit-risk ratio. These objective functions (i.e., the profits or profit risk ratio) were discontinuous functions of a series of exchange rates, so we could not develop an iterative procedure to optimize the parameters by assuming gradual or continuous changes in the parameters due to gradual or continuous changes in the objective function. Therefore, we had to apply meta-heuristic methods, such as genetic algorithms or evolutionary algorithms.

Let us consider how we can make a trading rule more sophisticated. A trading rule is a rule that mimics how traders decide whether to buy or sell by observing and recognizing any patterns that emerge in the market behavior. We assume that a rule comprises the following: predictions of price changes; when and how to open a position based on the predicted, current, and historical values; and the condition where the position is closed by a limit or stop order based on the current price. In our study, prediction was independent of other factors, including leverage, limit orders, and stop orders, so the prediction function was identified initially before the trading rule was optimized.

The position taken by a trader is either long or short. A long (or short) position means that a trader buys (or sells) one currency (the base currency of a pair) and expects to sell (or buy) the currency in the near future at a higher (or lower) rate. A trader who buys a currency expects that the exchange rate of that currency will rise relative to a quoted currency in the near future. During simulated trading, the condition of the trading rule is examined when a fixed time period has passed after the position was taken. If the position to be taken at the time is different from the one taken at the previous time, the previous position is closed and a new position is opened. If not, the previous position is maintained. This periodic examination is a model for real traders who check the market behavior at fixed time intervals. If the market moves rapidly in an adverse direction, the trader may lose capital over a short period. To minimize losses, we can place a stop or stop-loss order. If the market moves rapidly in the expected direction but then recedes before the check time arrives, we miss a chance to take profits. To minimize the loss of opportunities, we can place a take profit order.

Leverage is another method used by the trading rule. A trader wants to invest variable amounts of capital depending on the fluctuating conditions. The trader will use leverage to invest more when they are more certain of a prediction than that when they uncertain. In the FX market, traders use a maximum leverage of 10–1 or 100–1, depending on the brokers and regulations. It should be noted that the effectiveness of leverage has been proven empirically (Sermpinis et al. 2012).

We reasoned that more explanatory variables (or feature variables) than those used currently by many researchers should be incorporated into the modeling functions, so the information underlying these variable would be integrated to facilitate reliable predictions and trading rules.

In current studies of prediction, the prices or their moving averages are recognized widely as useful information for making predictions. Some studies also incorporate information from the stock market or macroeconomic variables for FX prediction, such as the GDP or interest rate, but in this study, we focused on utilizing information from other currency pairs and other time horizons.

Researchers around the world have used technical indicators for many years to predict stock prices or FX rates (Gehrig and Menkhoff 2006; Park and Irwin 2004).

The results obtained using various adaptations of technical indicators to predict stock prices and FX rates are highly variable, but we decided to include some of these technical indicators in our proposed model because many traders who build the market use them.

We also incorporated variables related to different time horizons because different trading behaviors may appear over different time horizons. We assumed that some traders would receive information from outside at the same time as the other traders but their reaction times to the information may be variable. Observing and integrating the outside information and reacting to it over different time scales corresponds to observing the market behavior and combining its descriptions over different time horizons. Traders often watch many different charts on displays while trading but the charts may have different time scales. Traders flip the time scales and watch the movements of prices over various time scales. The resulting decision may depend on charts with various time scales. In previous studies where the prediction horizon was set to 1 month, the moving average was calculated with a unit of 1 month, but we assumed that a price change in 1 month might be affected by an emerging trend at a 1-month scale but also by emerging trends at 2-month or half-month scales. Thus, to predict 1 h ahead, we calculated a moving average every 2 h or 30 min, in addition to a 1 h moving average. Other time horizons could be used, e.g., 4 h or 15 min trends, but we could not search for a wide range of alternatives so we had to limit ourselves to a small range of choices.

We also used the rates of currency pairs other than the target currency pair as explanatory variables. For example, EUR/USD was used to predict USD/JPY. This is because a currency pair in a FX market is like a security on a stock market because all of the main currency pairs are correlated due to the fact that traders watch the movements of the rates between these currencies when opening or closing positions. Various Internet sites show real-time and historical correlations (Online Source 1, Online Source 2, Online Source 3).

Many researchers have studied whether there are correlations among different currency exchange rates (Wu 2007; Lee 2003; Drożdż et al. 2007; Mizuno et al. 2006; Kwapień et al. 2009). However, the existence of a correlation does not necessarily mean that the exchange rates of another currency pair can be used to predict the target currency pair. Nonetheless, we decided to include variables calculated from other currency pairs because we could not exclude the possibility of their predictability.

The prediction function we used, which is explained below, was support vector regression (SVR) where the feature variables were derived from historical exchange rates. Artificial neural networks (ANN) are the most popular nonlinear prediction functions. In the machine learning field, large margin classifiers such as support vector machines (SVMs) are also known to obtain better out-of-sample classification accuracies in many cases.

SVR, an extension of SVM to regression problems, is known to deliver comparable or better out-of-sample accuracies than ANN. We decided to use SVR in our study because it allowed us to apply different nonlinear functions to different sets of explanatory variables, possibly by using different kernel functions in the SVM and SVR, i.e., by extending these to multi-kernel SVM and SVR.

In particular, we recognized the versatility of the multi-kernel method and the importance of the sparsity of representation produced by large-margin classifiers such as SVM. In general, potentially useful variables may increase the prediction accuracy when they are added to explanatory variables. If we add more than necessary, however, the mutual dependences between variables may have adverse effects. Thus, variable selection or feature selection are necessary. However, large margin classifiers are known not to decrease the accuracy even when there are many relevant and irrelevant variables (Joachims 1998). In other words, it was not necessary to select variables for large margin classifiers and regressions.

Large margin classifiers solve nonlinear cases using nonlinear kernels. If the feature variables have their own distinct characteristics, it is not reasonable to apply one kernel function to the features and create a classifier or a predictor.

A multiple kernel method was proposed to solve this problem, which uses linearly combined kernel functions instead of a kernel function in the SVM and SVR settings. In our study, a moving average and a technical indicator relative strength index had different natures, so the best kernels for these diverse variables should be also different. Therefore, we formed groups of similar features, each of which corresponded to a kernel function, before we applied multiple kernel learning.

This study can be summarized as follows: (1) we focused on predicting exchange rate changes over a short time horizon and we simulated trading using a trading rule, which was estimated and optimized with historical intraday data; (2) we employed a rolling window with a relatively short period of in-sample learning and of out-sample testing to follow rapid changes in trends, (3) we improved the sophistication of the trading rule, by integrating information from different sources using the predicted rate changes, a combination of order types, and some technical indicators; (4) we used many potentially useful variables to integrate information from different sources, including technical indicators for the different time horizons of different currency pairs; (5) we used multiple kernel learning for the large margin classifiers, without selecting variables manually; and (6) we used a genetic algorithm (GA) to optimize the sophisticated trading rules.

The use of a GA to search for the best parameters corresponds to finding the best models in a probabilistic manner. However, the search is not exhaustive, so it is possible that a much better model still exists.

2 Background

The FX market is considered to be the largest financial market in the world. This market has the advantage that its traders can trade practically around the clock on business days. Therefore, compared with trading in stock market, traders have more chances to trade in FX markets. However, compared with stock markets, the FX market is also a high-risk market.

To predict FX rates directly, researchers have used methods from statistics and signal processing, such as the autoregressive (AR) model (Champernowne 1948), the autoregressive moving average (ARMA) model (Box and Jenkins 1994), and the autoregressive integrated moving average (ARIMA) model (Box and Jenkins 1994).

ARMA model is a special case of ARIMA model where the order of integration is 0. These models are not sufficiently powerful for stock or FX market forecasting (Ravindran et al. 2008), probably because stock time series and FX time series have different characteristics than these models assume. In addition, autoregressive conditional heteroskedasticity (ARCH) for modeling volatility was proposed by Bollerslev (1986), which is, when combined with ARMA/ARIMA, considered to be more effective for modeling the dynamics of FX rates. However, some experimental results obtained using these methods are highly variable (Bonilla et al. 2011; Ravindran et al. 2008). Thus, we decided to pursue the use of nonlinearity for expressing exchange rate. From a methodological point of view, ANNs are alternative methods for describing regression functions, where the space of the regression function is difficult to assume. Many studies have shown that ANNs significantly outperform linear models such as ARMA and a naïve random walk model (Hann and Steurer 1996; Zhao et al. 2009; Chen and Leung 2004; Kodogiannis and Lolis 2002; De Faria et al. 2009; Koskela et al. 1997; Mehdi and Mehdi 2011), while there are reports of good FX rate forecasting performance with ANN (Wong and Selvi 1998; Azoff 1994). The most commonly used types of neural networks are feedforward networks with sigmoid functions or radial basis functions (RBFs).

SVMs are alternatives to ANN, which sometimes perform better (Kwok 2000; Cao 2003). SVMs are renowned for its ability to perform well when there are many relevant features (Joachims 1998). Pai and Lin (2005) designed a hybrid method, which combined ARIMA with SVM. SVMs are known to be robust to the overlearning caused by many relevant or irrelevant features, so they can improve predictions that use many features. We verified the validity of using a SVM with many features but we concluded that it would be better to try different learning kernels for different sets of features after we analyzed the results of preliminary experiments where we used different kinds of features, as described below.

In recent years, many researchers have used the multiple kernel learning (MKL) (Bach et al. 2004; Sonnenburg et al. 2006) method to address the problem of selecting suitable kernels for different feature sets. This technique mitigates the risk of erroneous kernel selection to some degree by using a set of kernels, deriving a weight for each kernel, and making better predictions based on the weighted sum of the kernels.

One of the major advantages of MKL is that it can combine different kernels for different input features. Several researchers have applied MKL in their research fields. For example, Joutou and Yanai (2009) to applied MKL to food image recognition. Foresti et al. (2009) applied MK regression to wind speed prediction and their results outperformed those of some conventional methods. Recently, some researchers have applied MKL to predicting the FX and stock markets. For example, Fletcher et al. (2010) applied MKL to predicting the FX market based on the limit order book. Luss and d'Aspremont (2012) applied MKL to the prediction of abnormal returns from news using text classification. Yeh et al. (2011) applied MKL to predicting the stock prices on the Taiwan stock market.

A technical indicator for FX rates is a function that returns a value for given FX rates over a given length of time in the past. These technical indicators might provide traders with guidance on whether a currency pair is being oversold or overbought, whether a trend will continue, and so on. The moving average is the simplest and best-known technical indicator. It is also the basis of many other trend-following or overbought/oversold indicators. The moving average is inherently a follower rather than a leader, but it reflects the underlying trends in many cases.

There are many well-known advanced technical indicators such as moving average convergence/divergence (MACD) (Stawicki 2007), the relative strength index (RSI) (Wilder 1978), Williams %R (Williams 2005), and the bias ratio (BIAS). Williams %R and BIAS are both based on a moving average and they have the specific abilities to provide an early warning of overselling or overbuying. We extracted features such as these technical indicators from the time series data of currency pairs other than the target pairs. Previous studies of the prediction of FX rates generally considered only the target currency pair. However, there should be various correlations between the target currency pair and other currency pairs in the financial markets because these currencies are traded against each other in the global market.

In addition, we extracted features from the same time series data, but over different time frames. Features with a longer or shorter time horizon can also be useful for predicting a trend or a rate with a 1-h time horizon. Therefore, features at 30-min and 2-h scales are extracted from three or five FX currency pairs. Most previous studies have only used the features from the target trading time horizon, e.g., only the daily time series data are used to predict the rate one day in advance.

Evaluation measures are very important for gauging the accuracy and efficiency of prediction methods. The hit ratio and RMSE are often used to evaluate predictions of FX rates and stock prices. The hit ratio is the proportion of correct predictions about changes in direction made by a predictor. However, even if our predictions have a large hit ratio, we might not make a profit if we fail to predict the magnitudes of changes accurately. Similarly, even if the RMSE of the proposed model is small, we might not make a profit without predicting the directions of changes accurately.

Although ARMA-GARCH models are reported to be superior in predicting stock prices than ARMA models, i.e., volatility is useful for prediction of its original sequence, in our experiments for hourly foreign exchange rate predictions, we could not observe a considerable difference in RMSE of the two. Therefore we did not consider volatility in our proposed model. In addition, traders often consider volatility of market to hedge the risk. In this study, we did not use volatility indicator but we considered stop-loss when designed the GA chromosome, which has the ability to close the position when FX market is in high volatility.

The action taken when someone can predict the FX rate accurately is to buy or sell, so the appropriate measure of the correctness of the predictions should be the profit. In any simulation, there should be a trading strategy or rule that simulates the FX transactions on which the profit depends.

In this study, we measured the quality of predictions made with trading strategies for buying and selling foreign currencies over short periods (our target time interval was 1 h), where we assuming that we started trading with a certain amount of Japanese yen. We applied MK regression to predicting the exchange rate changes using features such as MACD indicators, before we applied the GA to search for a trading strategy with an optimal combination of the MK regression results and the values of some overbought/oversold technical indicators. We include leveraging (a trading tool) when we place an order. We evaluated the profit with the proposed method and the performance in terms of the return-risk ratio.

In our trading strategy, we include the predicted FX rate changes and the overbought/oversold technical indicators. We also included thresholds that prompted specific actions. We bought if the combined value from the prediction and the overbought/oversold indicators was higher than the buying threshold, whereas we sold if the combined value was lower than the selling threshold. Furthermore, to control the risk from an unexpected jump in the FX rate, we also use take-profit and stop-loss orders, and we include their levels in the trading strategy.

We used a GA to learn the trading strategies, where the resulting profit was the fitness value of the chromosomes, because the profit from transactions comprised many discrete events. In addition, our GA design considered different leverages for buying and selling and it had settings for profit-taking and loss-stopping. Thus, are model was more complex than those used in previous studies so it was expected to perform better.

GAs have been used successfully for economic and financial prediction (Frick et al. 1996). A specific feature of the GA is that its hypothesis space is discrete and structured. The GA uses techniques inspired by evolution, i.e., inheritance, mutation, selection, and crossover (Banzhaf et al. 1998). Many researchers have made predictions about the stock market or FX market using GAs and they have reported good performance. Fuente et al. (2006) used a GA method for trading on stock markets, while Allen and Karjalainen (1999) and Hirabayashi et al. (2009) employed GAs to generate trading rules. Badawy et al. (2005) used a GA to select appropriate technical indicators at a particular trading time. Allen and Karjalainen (1999) reported an indicator combination scheme that delivered greater returns than a simple buy-and-hold strategy when the transaction costs were below a certain threshold. Chu et al. (2000) proposed an intelligent trading advisor based on several indicators and historical stock prices.

In summary, we first predicted the target FX rate changes using features extracted by multiple kernel regression from a time series of the exchange rates of the target currency pair and other currency pairs. Next, we used the GA to learn the trading strategy where the parameters were related to the predicted FX rate changes, overbought/oversold technical indicators, buying and selling thresholds, and the levels of take-profit and stop-loss orders. Our evaluation measures were the returns and return-risk ratios of the trades based on the trading strategy.

In this study, we assumed that the dynamics of the FX market changed slowly and they were considered to be relatively stable for at least 1500 trading hours (around 3 months or 12 weeks). However, this property is not supported by some previous studies. Indeed, there is evidence that the money market can change abruptly. However, our experimental results demonstrated that the proposed system obtained steady profits so this assumption was not unrealistic.

The remainder of this paper is organized as follows. Section 3 describes some well-known indicators, which are used widely by traders for technical analysis. It also describes the evaluation measures used in this research, as well as providing background information for SVR, MKL, and GA. Section 4 explains our proposed hybrid method in detail. Section 5 presents the design of the experiments. Section 6

Table 1 List of technical indicators used in this research

Indicator	Mathematical formula	Parameters
SMA	$SMA_n(t) = \frac{\sum_{k=t-n+1}^{t} R(k)}{n}$	n is the length of timeframe
EMA	$EMA_n(t) = R(t) * k + (1-k) * EMA_n(t-1)$	Usually $a = 2/(n+1)$, n is the length of timeframe
MACD	$MACDValue(t) = EMA_{12}(t) - EMA_{26}(t)$ $MACDSignal(t) = EMA_9(MACD(t))$	
BIAS	$BIAS_n(t) = \frac{R(t)-SMA_n(t)}{SMA_n(t)}$	n is the length of timeframe
WPR	$\%R(t) = \frac{R(t)-high_{Nperiods}}{high_{Nperiods}-low_{Nperiods}} \times 100$	n is the length of timeframe
RSI	$RSI_n(t) = 100 - \frac{100}{1+RS_n(t)}$	$RS_n(t) = \frac{\text{Average of positive price changes in } n \text{ days}}{\text{Average of negative price changes in } n \text{ days}}$ n is the length of timeframe

provides the experimental results and a discussion. Section 7 concludes this study, by describing the problems we encountered and making some suggestions for future work.

3 Preliminaries

3.1 Technical Indicators

Numerous influential trading technical indicators are widely recognized and used by traders around the world. Some technical indicators are fairly straightforward to obtain and have proven successful in trading history, such as the moving average (MA) and MACD, which help traders to spot or follow trends, and BIAS, WPR, and RSI, which are used for identifying the overbought and oversold conditions of a stock.

Table 1 shows the technical indicators used in this study, where $R(k)$ refers to the closing rate at time period k and n is the length of the time frame used to calculate the value of an indicator.

MA is used to understand the present trend so it referred to as a trend-following index. It is used to emphasize the direction of a trend and to smooth out random price fluctuations. SMA is a simple mean value with identical weights to those used for past rates and the EMA is the average value of the rates of a currency pair for a given length of, which attributes greater weights to newer changes and lower weights to older ones.

MACD is used to predict changes in the market tendency and it provides two indicators: MACD and the MACD signal. MACD represents the difference between a fast and slow EMA of closing prices. "Fast" refers to a short-period average and "slow" is a long-period average. When MACD (t) is greater than 0, the short and steep uptrend is more influential than the long and gentle uptrend, which means that the stock price is likely to go up in the near future. Based on the default parameters, MACD is the difference between the 12-period and 26-period EMAs. The default values (12, 26,

Table 2 Summary of evaluation measures

Evaluation measure	Calculation	Description
RMSE	$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(P_i - R_i)^2}{n}}$	P_i is the predicted price at time i, and R_i is the real price at time $(i+1)$, and n is the number of prediction times.
Accumulated return	$AC = \sum_{i=1}^{m} \mathrm{Re}_i$	Re_i is the return in testing period i, and m is the number of testing periods.
Sharpe ratio	$S = \frac{R-R_f}{\sigma} = \frac{E[R-R_f]}{\sqrt{var[R-R_f]}}$	R is the asset return, R_f is the return on a benchmark asset, $E\left[R - R_f\right]$ is the expected value of the excess of the asset return over the benchmark return, and $var\left[R - R_f\right]$ is the variances of the asset return. In our experiments, we used the Sharpe ratio as an evaluation criterion to evaluate the return-risk ratio performance of the models.

and 9) of MACD parameters can be changed based on the needs of the traders. In our study, we simply used the default values of the MACD parameters because this value set is recognized widely and used throughout the world.

BIAS, WPR and RSI are generally used to decide whether a stock is possibly in an oversold, overbought, or normal condition. An extremely high or low value is a signal, which tells the trader to buy when the currency pair is oversold and to sell when it is overbought. The parameter n of these indicators can be set by the traders. In this study, we used a GA to obtain the best parameter n for these indicators in the training period.

3.2 Evaluation Measures

To evaluate the performance of the models, we used the following three measures: RMSE to evaluate the goodness of fit of the model predictions of rate changes, the accumulated return to evaluate the profit-making ability, and the Sharpe ratio to evaluate the ability to control risk while yielding good profits.

RMSE is a frequently used measure of the differences between the values predicted by a model or an estimator and the values actually generate by the entity being modeled or estimated.

In addition, we executed trading based on the trading signals output by each model and evaluated the returns (loss or profit). In general, high returns are accompanied by the potential for high risk. Therefore, we attempt to find a method to reduce the risk and increase profits. The Sharpe ratio is used to measure the excess return per unit of risk in an investment asset or a trading strategy, which is named after William Forsyth Sharpe (Sharpe 1994).

Summaries of these three evaluation measures are provided in Table 2.

3.3 Support Vector Regression, Multiple Kernel Learning, and Genetic Algorithm

3.3.1 Support Vector Regression

SVR is a version of a SVM (Vapnik 1995) with several distinct advantages. For example, SVR is used to solve problems by balancing the empirical error and a regularization term, where the risk is measured using Vapnik's ε-insensitive loss function. In addition, SVR usually estimates a set of linear functions, which are defined in a feature space with a high dimension. SVR is also renowned for its ability to perform well when there are many relevant features. The regression function can be estimated by minimizing a regularized risk function as follows:

$$\min_{w} \frac{1}{2}\|w\| + C*\frac{1}{l}\sum_{i=i}^{l} L_{\varepsilon} \tag{1}$$

$$L_{\varepsilon} = \begin{cases} |y_i - w*\Phi(x_i) - b| - \varepsilon, & if |y_i - w*\Phi(x_i) - b| \geq \varepsilon \\ 0, & otherwise \end{cases} \tag{2}$$

where w and b are a weight vector and an offset, respectively, which define the maximum margin hyper-plane. Note that $\Phi(\cdot)$ is a possibly nonlinear mapping from the input space to a feature space.

3.3.2 Multiple Kernel Learning

In general, the normal SVM is applied to a single feature type. In this study, we used MKL to integrate the features of different time horizons and different currency pairs. Using MKL, we trained an SVM with an adaptively weighted combination of kernels, which combined different types of features. The kernel combined was as follows:

$$K_{comb}(x, y) = \sum_{j=1}^{K} \beta_j K_j(x, y) \tag{3}$$

$$\beta_j \geq 0, \sum_{j=1}^{K} \beta_j = 1 \tag{4}$$

where β_j are the weights used to combine sub-kernels . The optimal weights were estimated by MKL based on the training data and we obtained an optimal combined kernel by preparing one sub-kernel for each feature set and estimating the weights using MKL. An efficient MKL algorithm was proposed by Sonnenburg et al. (2006) for estimating the optimal weights and SVM parameters simultaneously by iterating the training steps of a normal SVM. In our experiments, we used the Shogun toolbox, which includes the MKL library.

3.3.3 Genetic Algorithm (GA)

Goldberg (1989) provides an excellent discussion of the use of GAs for solving optimization problems. GAs start with an initial set of solutions, known as population.

Fig. 1 Process used by the genetic algorithm

```
Begin:
    t← 0;
    initialize population P(t);
    evaluate P(t);
    While (not termination condition) do
        Begin
            t← t+1;
            select P(t) from P(t−1);
```

The individual solutions in the population are known as chromosomes. Each chromosome is made up of a number of genes that encode representations of part of the solution. During each iteration (referred to as a generation in GA terminology), the current population evolves using reproduction strategies such as crossover, mutation, and immigration. A fitness function is used to evaluate the chromosomes and the survival of chromosomes from one generation to the next is biased in favor of the fittest chromosomes. In addition to reproduction strategies, an elitist strategy can be used to propagate the fittest chromosomes to the next generation. A combination of these strategies helps the population improve from generation to generation until the fittest member of the population represents a nearly optimal solution. Figure 1 and steps 1–6 show the GA procedures used to design the chromosomes employed in our study, which were based on those described by Goldberg (1989).

Step 1: Initialization
Generate the initial population.

Step 2: Evaluation
After initialization, each chromosome is evaluated using a fitness function.

Step 3: Selection
Selection is a process were suitable chromosomes derived from the parent population are selected for the next generation. In this step, the selection model is tournament selection. This step is repeated until the number of chromosomes selected is equal to the number in the population. To ensure the propagation of the elite chromosome, the GA uses the elitism mechanism. This mechanism selects P% individuals, which have the best relative fitness values, as the offspring of the next generation, whereas the remaining individuals experience genetic operations.

Step 4: Crossover and Mutation

Crossover operates by swapping the corresponding segments of a string representation of the parents, which extends the search for a new solution. Mutation is a genetic mechanism, which randomly selects a member of the population and changes one randomly chosen bit in its bit string representation.

Step 5: Evaluation

Each chromosome is evaluated using the designed fitness function.

Step 6: Check termination criteria

The process described in steps 2–5 are repeated until the termination criteria are satisfied. The proposed algorithm is terminated if the maximum number of generations is reached or the solution with the highest fitness has not changed for several generations.

Previous studies, such as Allen and Karjalainen (1999), have also used genetic programming to find trading rules. However, the rules obtained by Allen and Karjalainen (1999) were very simple and they lacked sufficient expressiveness to be as profitable as real traders, while even a simple combination of a technical indicator and a stop-loss was impossible. By contrast, our GA settings considered technical indicators and their parameters, while we also used the take-profit and stop-loss levels as genes in our chromosome design.

We used the Sharpe ratio as the basis for making comparisons. We aimed to control the risk while achieving good profits because the FX market is full of risk. The design of the take-profit and stop-loss orders was expected to allow our GA to adjust to rapid dynamic changes in the FX market. Furthermore, the fluctuations in the FX market are small compared with those in the stock market, so our GA chromosome also considered leveraging for buying and selling. Therefore, our GA chromosome design considered a great deal of information related to real FX trading, so the trading rules generated by our GA were expected to be more powerful than those produced in previous studies.

4 Method

4.1 Trading Strategy of Our Proposed Method

The signals generated by various indicators might not always be in agreement, so it is necessary to develop a mechanism that resolves any conflicts that might occur.

The final decision, D, is a linear combination of the overbought/oversold indicators and the FX rate changes predicted from the MK regression:

$$D = \sum_{i=1}^{N} w_i e_i \tag{5}$$

where w_i are the weights learned by the GA and e_i are the values of the MK regression, as well as the values of the overbought/oversold technical indicators we considered (RSI, William %R, and BIAS). Note that the indicators we used were in ratio forms, i.e., RSI/100, WPR/100, and BIAS, as shown in Table 1. Furthermore, it should be noted that the MK regression outputs are exchange rate changes, which are shown in (6). According to these conventions, e_i in (5) are all dimensionless so they are consistent.

Exchange rates do not change as much as stock prices. Therefore, it is too difficult to obtain high returns from the FX market without leverage. In our experiments, we used leverage in the same way as traders. However, this makes the risk of FX trading very high, so we had to implement a mechanism to control the risk. Thus, we set the stop-loss level and take-profit level relative to the contracted price and implemented these as features in the GA, i.e., as genes in the GA chromosomes.

After the weights and other parameters were learned by the GA, we obtained the decision value, D, which was related to the hours covered by the testing period. If the value of D was less than or equal to the threshold value for buying, we bought with the buying leverage. If the value of D was greater than or equal to the threshold value for selling, we sold with the selling leverage. If the value of D is between the threshold values for buying and selling, we did not open a new position. In addition, because the target prediction horizon was 1-h, our trading strategy closed the position 1 h after we opened the position. However, if a trading signal was the same as the trading signal from 1 h earlier, our trading strategy held the position until the next hour.

4.2 Features Used for Multiple Kernel Regression

USD/JPY was selected as the target currency pair in our experiments. When traders trade currencies, they consider the target currency pair and changes in other important currency pairs. In addition, traders watch different time horizons for indicators, rather than the target trading period alone.

Therefore, we extracted features from different currency pairs and different time horizons. The historical FX data we used were list of deals with time stamps in seconds in ICAP data or 1-min interval data, which contained the opening, high, low, and closing values for the interval in Forexite data. We processed these data and obtained the values of indicators with different time horizons at the same time point. Note that the calendar dates (date, hour, and minute) of the ICAP data and Forexite data are consistent. For USD/JPY, EUR/USD, GBP/USD, we used the data of ICAP for MKL-3. Since we did not have ICAP data for AUD/USD and USD/CHF at the time of experiments, we used those of Forexite for experiments relating to MKL-5. ICAP and Forexite data correspond to different markets but traders could access to both so that exchange rates are the same on average.

In our experiments, the target trading currency pair was USD/JPY and the trading period was 1 h. First, we used three main currency pairs as the input feature sources: GBP/USD, EUR/USD, and USD/JPY. This was simply because GBP/USD and EUR/USD are the other two most widely traded currency pairs in FX markets. In addition, AUD/USD and USD/CHF are two direct currency pairs traded commonly in FX markets. For comparison, we considered AUD/USD and USD/CHF, as well as the three main currency pairs, as input feature sources in the experiments using our proposed method.

We calculated the 1-h, 30-min, and 2-h indicators for each time point of the 1-h time horizon for the USD/JPY currency pair. Table 3 shows the MK regression input

Table 3 Input features for each currency pair and time horizon

No.	Feature	No.	Feature
1	MACD-value at time t	5	MACD-value at time $(t-2)$
2	MACD-signal at time t	6	MACD-signal at time $(t-2)$
3	MACD-value at time $(t-1)$	7	MACD-value at time $(t-3)$
4	MACD-signal at time $(t-1)$	8	MACD-signal at time $(t-3)$

features for each currency pair with each time horizon. The number of dimensions is eight.

For each time point over the USD/JPY 1-h time horizon, we calculated the MACD indicators for three different currency pairs with three different time horizons. Finally, we obtained eight input features for the three different currency pairs with the three different time horizons, and we had one target variable: the exchange rate for the USD/JPY 1-h time horizon. The output of the MK regression was the change rate per 1 h for USD/JPY, as follows.

$$Change\_Rate = [Price(t+1) - Price(t)]/Price(t) \tag{6}$$

For each input feature set of a currency pair and a time horizon, we assigned one Gaussian kernel and one linear kernel, and we assigned the default values to the parameters of the Gaussian kernel.

4.3 Design of GA Chromosomes

We designed the chromosome shown in Table 4 for the trading strategy described in Sect. 4.1.

The genes were represented as follows.

(1) Numbers 1–4 (20 bits) represent the weights of the indicators and the MK regression results. The range of the weights for all of the indicators and the MK regression value was from -1 to $+1$, where the least significant bit represents $2/32 = 0.0625$.
(2) Numbers 5 and 6 (10 bits) represent the buy and sell leverages. The type was integer and the values ranged from 1 to 32.
(3) Numbers 7 and 8 (10 bits) represent the take-profit and stop-loss levels for the contracted price in terms of the ratio relative to the current rate. The percentage for the take-profit and stop-loss relative to the contracted price ranged from 1 to 5 %.
(4) Numbers 9 and 10 (10 bits) represent the threshold values for buying and selling. Each threshold ranged from -3.3 to $+3.3$. In our GA design, we set the constraint that the buying threshold was less than the selling threshold.
(5) Numbers 11–13 (15 bits) were used for the RSI, WPR, and BIAS parameters. The values ranged from 1 to 32.

Table 4 GA chromosome design

No	Length (bits)	Value range	Meaning
1	5	-1 to 1	RSI weight
2	5	-1 to 1	WPR weight
3	5	-1 to 1	BIAS weight
4	5	-1 to 1	MKL weight
5	5	1–32	Leverage ratio for buying
6	5	1–32	Leverage ratio for selling
7	5	1–5 %	Take-profit point in percentage
8	5	1–5 %	Stop-loss point in percentage
9	5	-3.3 to 3.3	Threshold of D for buying
10	5	-3.3 to 3.3	Threshold of D for selling
11	5	1 to 32	Parameter of RSI
12	5	1 to 32	Parameter of WPR
13	5	1 to 32	Parameter of BIAS

In the GA training step, we set the population size to 200 and the maximum number of generations to 100. In preliminary experiments, we varied these numbers. The results are not conclusive, but we have chosen these numbers considering computation time and resulted performance. We implemented another termination criterion, i.e., the process terminates when fitness value of the best individual did not improve for 10 successive generations. We initialized these individuals with random chromosomes according to the gene structure shown in Table 4. To keep very fit individuals, the elite 10 % (the top 10 % of individuals in terms of fitness) were reserved automatically at each generation. The fitness value was the profit accumulated during GA learning.

4.4 Design of the Proposed Method

Figure 2 shows our proposed hybrid method, while Fig. 3 is a flowchart of the MKL-GA hybrid method. The following are the training and testing procedures used by the proposed hybrid method.

(1) We obtained the exchange rate change predictions for the 1-h USD/JPY one step ahead by applying multiple kernel regression.
(2) We created a set of random values for the chromosomes as the first generation.
(3) For each chromosome, we applied the trading strategy to the training data at each specified time during the training period by calculating the overbought/oversold technical indicators, computing the decision value D, and making decisions.
(4) We computed the profits accumulated during the trading period as the fitness value. Tournament selection was used as the selection method for the GA in our system. This method selected a number of individuals from the population. A "tournament" was performed and the winner was selected to perform the crossover. In addition, the top 10 % chromosomes (those that reached the top 10 % in terms of profit)

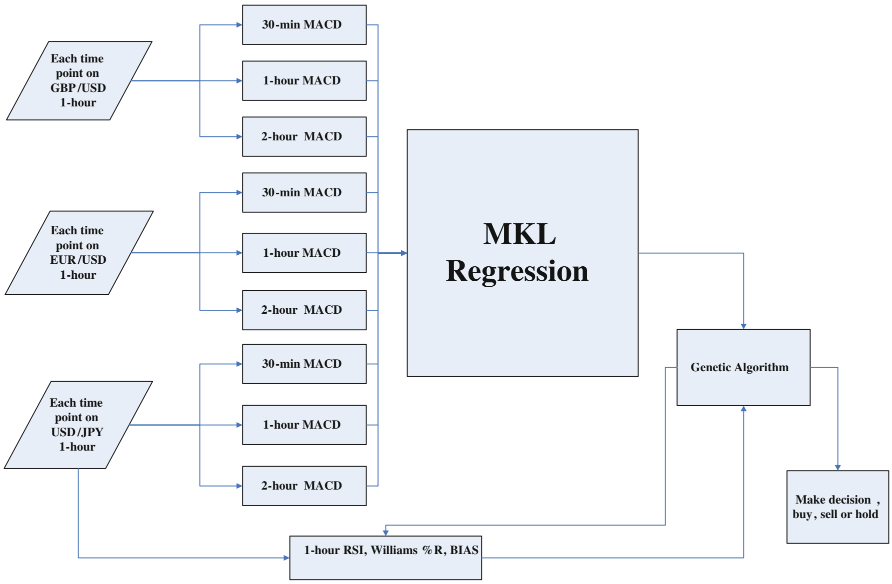

Fig. 2 Diagram of proposed hybrid method

were retained directly for the next generations. New chromosomes were created by applying a crossover operation (we used two-point crossover as our crossover method) to the chromosomes selected from the current generation. Mutation was performed by converting chromosome's 0 to 1, or 1 to 0. Crossover was repeated until a new generation was produced. Portions of the chromosomes were mutated or flipped randomly. The probabilities of crossover and mutation were 60 and 1 %, respectively.

(5) Steps 3 and 4 were repeated until the maximum number of generations (100) was reached or the fitness of the best individual did not improve for 10 successive generations. The best chromosome was then selected to represent the optimized trading strategy.

(6) We calculated the profit and loss by applying the resulting trading strategy to the testing data.

5 Experiments

5.1 FX Data

The exchange rates used in our study were obtained from ICAP and Forexite. The ICAP data comprised a list of best offers, best bids, and dealt prices, if any, for every second. These were very reliable but were limited to USD/JPY, EUR/USD, and GBP/USD from 2008 to 2011. The Forexite data were freely downloadable from their website and have been used by many researchers. For 2008–2011, we used ICAP data for USD/JPY and EUR/USD, and Forexite data for GBP/USD, because the number of

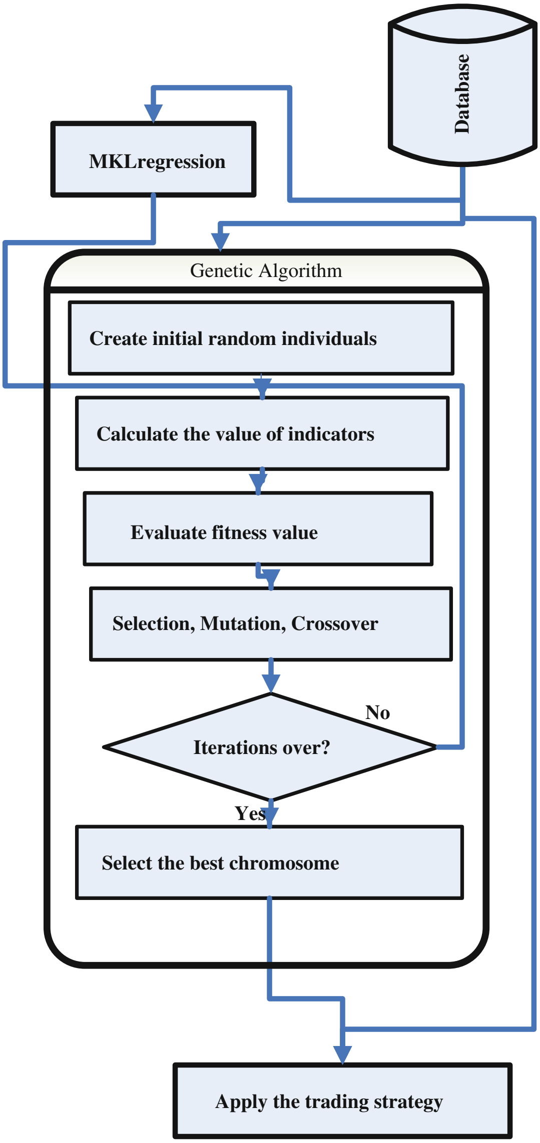

Fig. 3 Flowchart of the MKL-GA hybrid model

deals in the ICAP GBP/USD data was lower than we needed and the correlations between the ICAP data and Forexite data for GBP/USD were very high. The calendar dates (daily, hourly, and minute frequencies) of the ICAP data and Forexite data were consistent. We also checked the correlations between the ICAP data and Forexite data for USD/JPY and EUR/USD. Moreover, we compared the results of our experiments using both datasets and we concluded that we would be able to obtain equivalent results from the ICAP data and the Forexite data.

Table 5 Training and testing periods

Period	Process	Length of period
A	MK regression training	1000 trading hours (around 8 weeks)
B	MK regression testing (prediction)	500 trading hours (around 4 weeks)
B-1	MKL-GA training	250 trading hours (around 2 weeks)
B-2	MKL-GA testing (trading)	250 trading hours (around 2 weeks)

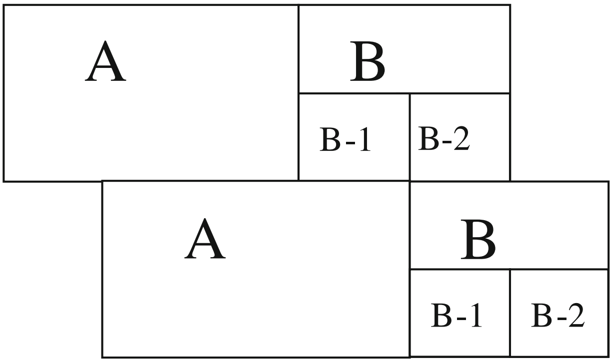

Fig. 4 Rolling windows used for training and testing (prediction)

We constructed 1-h and 30-min data from the ICAP data for USD/JPY and EUR/USD from 2008 to 2011, and we transformed the Forexite 1-min data for the five main currency pairs (USD/JPY, GBP/USD, EUR/USD, AUD/USD, and USD/CHF) into three different sets of time horizon data for other years. We also calculated the indicators at each time point for each time horizon.

To separate the training and testing data, we used a rolling window method. We performed MK regression with 1000 trading hours (around 8 weeks) of data and we obtained the predicted values for the subsequent 500 trading hours (around 4 weeks). Based on the values predicted by the multiple kernel regression, the first 250 trading hours (around 2 weeks) of predictions were used as GA training and the remaining 250 trading hours (around 2 weeks) of predictions were used for GA testing (Fig. 4; Table 3), i.e., for testing the overall MKL-GA procedure. For each successive experiment, we moved the training and testing periods forward by 250 trading hours (around 2 weeks), which was the same as the MKL-GA trading period and this made the trading period continuous. Table 6 shows the combined periods for the training and testing datasets.

In addition, the spread (i.e., the bid/asked difference) was fixed in the experiments at 0.03 JPY per USD trading. According to EBS data for the years from 2008 to 2011, the average spreads of USD/JPY were 0.0235, 0.0246, 0.0174, and 0.0117 JPY per USD, respectively. These averages were calculated as the duration-weighted averages of the differences between the best offers and the best asking prices at the end of every second reported by ICAP. In FX trading, the spread is usually the transaction cost, which was considered in the trading simulations in the present study.

Table 6 Datasets used in the experiments

Dataset	Training periods	Testing (prediction and trading) periods
Dataset 1	2008/01–2008/11	2008/03–2008/12
Dataset 2	2009/01–2009/11	2009/03–2009/12
Dataset 3	2010/01–2010/11	2010/03–2010/12
Dataset 4	2011/01–2011/11	2011/03–2011/12

In this experiment, we treated the swap point as negligible because the swap points of our target USD/JPY were quite small. Theoretically, the swap point is the difference between the bank rates of two currencies. For JPY, the bank rate is the uncollateralized overnight call rate, whereas that for USD is the federal funds rate. The JPY rate was 0.51 % at its highest in March 2008 but less than 0.1 % from 2009 to 2011. The USD rate was 3.94 in January 2008, 0.16 % in December 2008, and 0.21 to 0.07 % from 2009 to 2011 (Online Source 4). Thus, the difference was very small, except during 2008. The long positions and short positions tended to have almost equal lengths, so the profits and losses from interest that we received and paid were almost balanced. It means that the swap points were almost negligible overall.

We tested if the foreign currency exchange rate time series that we target is well approximated by linear models. The BDS test was proposed by Broock et al. (1996) and is now widely known for its power to test against a wide range of nonlinear models (Barnett et al. 1997). We apply the BDS test to the residuals from the ARIMA-type model fitted to the foreign currency exchange rate series, a possibly nonlinear time series. Theoretically we need to filter out all possible linearity which is difficult at best, but filtering by ARIMA models is accepted for the linear whitening (Barnett et al. 1997).

In the tests we used the embedding dimensions from 2 to 5, and ε (the distance threshold) from 0.5, 1, 1.5 and 2 standard deviations of the data set, which are widely used and are default values for BDS test in tseries package in R which we used for the analysis in the current study. The degrees of the ARIMA model fitted are from (1, 0, 0) to (7, 1, 7). As is explained below, we ran BDS tests for each testing period with varying degrees and obtained 16 p-values, each corresponding to a combination of embedding dimension and an epsilon.

When investigated yearly, the p-values in the test results are very small, i.e., even the largest one was as large as 2.42×10^{-20}. The null hypothesis that the residuals are i.i.d. is rejected. When tested every 1500 h, which is the length of the period when our proposed method learns and predicts, majority of the periods exhibit small p-values, i.e., in 64 periods among the total of 76 periods, all the p-values are small than 0.01 and in remaining 12 periods, although some p-values are greater than 0.01, p-values in at most four cases (among 16 cases of combination of four kinds of embedding dimensions and four kinds of epsilons) are greater than 0.01. The 12 cases distribute as one in 2008, eight in 2009, three in 2010, and none in 2011. We therefore concluded that in hourly time series of USD/JPY exchange rate from 2008 to 2011, there might be nonlinearity dependence.

5.2 Initial Capital and Trading Amount

We set the trading amount as the base amount multiplied by the leverage. We set the maximum leverage as high as 32, so the ratio of the base amount relative to the initial capital was set to 1/20 for safety.

5.3 Proposed Method and Baseline Methods

The proposed and baseline methods are shown in Table 7, while Table 8 provides a summary of the combinations of trading strategies and predictions.

Table 7 List of the methods used in the experiments

No	Method	Description
1	SVR-1-STS	SVR and a simple trading strategy. One currency pair is used
2	SVR-3-STS	SVR and a simple trading strategy. Three currency pairs are used
3	SVR-5-STS	SVR and a simple trading strategy. Five currency pairs are used
4	SVR-1-GA	Hybrid of SVR and GA. One currency pair is used
5	SVR-3-GA	Hybrid of SVR and GA. Three currency pairs are used
6	SVR-5-GA	Hybrid of SVR and GA. Five currency pairs are used
7	MKL-3-STS	MK regression and a simple trading strategy. Three currency pairs are used
8	MKL-5-STS	MK regression and a simple trading strategy. Five currency pairs are used
9	MKL-3-GA	Hybrid of MK regression and GA (proposed method). Three currency pairs are used
10	MKL-5-GA	Hybrid of MK regression and GA. Five currency pairs are used
11	Buy and hold	Buy a currency pair and hold until the last time
12	Sell and hold	Sell a currency pair and hold until the last time
13	ANN	ANN and a simple trading strategy
14	ARIMA	ARIMA and a simple trading strategy

Table 8 Summary of the combinations of trading strategies and predictions (method numbers are shown)

		Trading strategy		
		Simple	Trained by GA with SVR prediction	Trained by GA with MKL prediction
Prediction	SVR	1, 2, 3	4, 5, 6	
	MKR	7, 8		9, 10

We also considered a simple trading strategy (STS) in our baseline methods to compare with the trading strategy generated by our proposed method. In this simple trading strategy, we opened a position based solely on the prediction of the result from the SVR or the MK regression. We bought the base currency of the target currency pair if the prediction was going up and sold if the prediction was going down. We closed the position after a specified period, i.e., 1 h in our experiments.

In our experiment, the MKL-GA with features extracted from three main currency pairs (USD/JPY, EUR/USD, and GBP/USD) was our proposed method (Method 9), while our proposed method with features extracted from five currency pairs (Method 10) was also used in the experiments for comparison. In addition, we compared the results of a method trained in a similar way to our proposed method but trading was conducted using FX rates predicted by SVR instead of MK regression. For the simple trading strategy with SVR, we used SVR to predict the change in the rate at the succeeding time (Methods 1–3). In the simple trading strategy with SVR, if the predicted rate change was greater than zero, we opened a long position for our target currency pair (USD/JPY), whereas if the predicted rate change was less than zero, we opened a short position for our target currency pair.

Methods 4–6 were used to determine whether MK regression was necessary for our proposed framework. The SVR was trained and its predictions were used to train the GA. Methods 7 and 8 were used to determine whether a trading strategy trained using the GA was superior to a simple trading strategy if the MK regression predictions were used.

Methods 11 and 12 used the simplest trading strategy: simply buy or sell the target currency pair (USD/JPY) and wait until the end time. The holding time was 250 trading hours (around 2 weeks), like that shown in Part D of Table 5. Methods 13 and 14 used the simple trading strategy to trade based on the predictions of an ANN method and an ARIMA(1,0,1) method, respectively. ARMA-GARCH model was chosen so that we could compare the results of proposed method with that of conventional linear and nonlinear models. For ANN, we tried varying number of units in the hidden layer and we show the results with a highest average profit in Sect. 6.

6 Results and Discussion

First, we evaluated the results of the MK regression part of our proposed method. We used RMSE as the evaluation measure and we compare the results of Simple Random Walk (SRW), MKL-3, MKL-5, SVR-1, SVR-3, SVR-5 and ARMA-GARCH. For ARMA-GARCH model, we tried several combinations of parameters as ARMA, and we show the results of ARMA(1,1) with GARCH(1,1) since there are very small differences in the results.

6.1 RMSE Results

Based on the average RMSE results for the testing periods from 2008 to 2011, which are shown in Tables 9 and 10, we found that the error (RMSE) with MK regression was much smaller than that with SVR. Thus, MK regression outperformed SVR, which

Table 9 Average RMSE for each method in each year

Year	ARMA-GARCH	SRW	SVR-1	SVR-3	SVR-5	MKL-3	MKL-5
2008	0.212021	0.207860	0.297933	0.822765	0.699106	0.216617	0.206236
2009	0.155264	0.150891	0.193786	0.648358	0.775471	0.157167	0.154017
2010	0.126683	0.124943	0.172585	0.619003	0.664844	0.131836	0.126947
2011	0.108685	0.101184	0.123165	0.420466	0.432392	0.108152	0.102514

Table 10 Standard deviation of the RMSE for each method in each year

Year	ARMA-GARCH	SRW	SVR-1	SVR-3	SVR-5	MKL-3	MKL-5
2008	0.07575	0.076772	0.196957	0.432763	0.215905	0.071405	0.076052
2009	0.02295	0.023337	0.082710	0.168785	0.155797	0.021742	0.023879
2010	0.03595	0.040646	0.103058	0.156235	0.174371	0.038881	0.037304
2011	0.02947	0.032658	0.058568	0.139091	0.101686	0.031470	0.031909

was very important for the success of the subsequent GA learning. We also found that the RMSE results with MK regression are better than it with ARMA-GARCH. Many empirical studies have reported that SRW is better than or competitive with other models used to predict FX rates and the average RMSE results with MKL-3 and MKL-5 were very close to that with SRW, but we could not use SRW for trading because SRW gave us no indication of the action to take.

6.2 Profit and Loss Ratios with the Proposed Method and Benchmark Methods

In one experiment set, we consider 1000 trading hours (around 8 weeks) for MKL learning and 500 trading hours (around 4 weeks) for MKL testing, which includes 250 trading hours (around 2 weeks) for GA training and 250 trading hours (around 2 weeks) for GA testing. Thus, for each year we conducted 19 experiments. We calculate the average and standard deviation of the profit of each experimental set for each method and each year. The profits were measured relative to the initial investment and were not scaled for years or weeks. The results are shown in Tables 11 and 12.

We then calculated the annual returns with each method, where the annual return was the sum of profits made in 19 experiments during a specific year. The returns were measured relative to the initial investment. Table 13 shows the returns obtained with different methods. First, we focus on the methods based on the simple trading strategy (SVR-3-STS, SVR-5-STS, MKL-3-STS, and MKL-5-STS).

In some years, the methods based on the simple trading strategy obtained good returns, such as SVR-5-STS and MKL-5-STS in 2008. However, the methods with a simple trading strategy did not yield good returns stably, which is demonstrated by the results in Table 13. They also incurred huge losses in some years, such as SVR-1-STS from 2008 to 2011, or SVR-3-STS in 2008 and 2009. There were large differences in the returns during each experiment and the huge overall losses in some testing years

Table 11 Average profits with each method during 250 trading hours (around 2 weeks) in each year (19 experiments per year). The profits were measured relative to the initial investment

Method	2008	2009	2010	2011
SVR-1-STS	−0.01337	−0.02367	−0.01644	−0.01702
SVR-3-STS	−0.01725	−0.00659	−0.00416	−0.00433
SVR-5-STS	0.01644	0.00397	−0.00297	0.00242
MKL-3-STS	−0.00672	0.00308	−0.00232	0.00066
MKL-5-STS	0.00992	0.00417	0.00153	0.00201
SVR-1-GA	0.00103	0.00192	0.00260	0.00389
SVR-3-GA	0.00144	0.00173	0.00111	0.00299
SVR-5-GA	0.00411	0.00248	−0.00202	0.00157
MKL-3-GA	0.00754	0.01084	0.00762	0.00660
MKL-5-GA	0.00510	0.00376	0.00952	0.00648
Buy and hold	−0.00432	−0.00221	−0.00429	−0.00194
Sell and hold	0.00432	0.00221	0.00429	0.00194
ANN	−0.00680	0.00042	−0.00287	−0.00477
ARIMA	−0.00478	0.00091	−0.00641	−0.00317

Table 12 Standard deviation of the profit for each method during 250 trading hours (around 2 weeks) in each year (19 experiments per year)

Method	2008	2009	2010	2011
SVR-1-STS	0.02647	0.01635	0.01865	0.01792
SVR-3-STS	0.03431	0.01738	0.01838	0.02068
SVR-5-STS	0.02934	0.02938	0.01752	0.01422
MKL-3-STS	0.04304	0.02040	0.02014	0.01728
MKL-5-STS	0.03323	0.02477	0.02375	0.01976
SVR-1-GA	0.00750	0.01191	0.01173	0.01081
SVR-3-GA	0.00911	0.00573	0.00689	0.00857
SVR-5-GA	0.00896	0.00605	0.01610	0.01023
MKL-3-GA	0.01631	0.01299	0.01502	0.01631
MKL-5-GA	0.01152	0.00908	0.01522	0.01168
Buy and hold	0.03257	0.02416	0.01876	0.01556
Sell and hold	0.03257	0.02416	0.01876	0.01556
ANN	0.02234	0.01200	0.00872	0.01582
ARIMA	0.05104	0.01542	0.02120	0.03152

were caused by very high losses in specific experiments. For example, Fig. 5 shows the distribution of the returns with SVR-3-STS during the testing periods in 2011, which shows that there was a large overall loss in 2010 and it was attributable mainly to the huge losses in the first and tenth trading periods of 2011, although the number of periods that yielded profits was almost the same as the number of periods that incurred losses (9 and 10, respectively).

Based on the results with the SVR-GA hybrid method, which combined SVR with GA and was a simplified version of our proposed hybrid method, we found that the overall profits with our proposed methods were better in each year than those with

Table 13 Profits with the proposed method and baseline methods between 2008 and 2011

Method	2008	2009	2010	2011
SVR-1-STS	−0.25415	−0.44980	−0.31232	−0.32329
SVR-3-STS	−0.32785	−0.12513	−0.07902	−0.08219
SVR-5-STS	0.31239	0.07539	−0.05648	0.04592
MKL-3-STS	−0.12785	0.05858	−0.04415	0.01267
MKL-5-STS	0.18839	0.07919	0.02915	0.03819
SVR-1-GA	0.01957	0.03648	0.04952	0.07405
SVR-3-GA	0.02747	0.03287	0.02126	0.05689
SVR-5-GA	0.07804	0.04710	−0.03844	0.02988
MKL-3-GA	0.14338	0.20603	0.14478	0.12541
MKL-5-GA	0.09688	0.07150	0.18083	0.12310
Buy and hold	−0.08221	−0.04199	−0.08157	−0.03682
Sell and hold	0.08221	0.04199	0.08157	0.03682
ANN	−0.12912	0.00811	−0.05460	−0.09059
ARIMA	−0.09086	0.01731	−0.12184	−0.06032

The profits were measured relative to the initial investment per year

SVR-GA. This may have been because of the relatively large errors (RMSE) of the SVR, as shown in Tables 9 and 10. We also found that ANN achieved losses in 3 years and a profit close to zero in 2009, while ARIMA suffered losses in 3 years.

The results with our proposed methods showed that MKL-3-GA and MKL-5-GA delivered returns that ranged from 7.1 to 18.08 % per year and they incurred no losses for 1 year. A comparison of these methods over each period showed that our proposed method performed consistently better than the other methods. Figures 6, 7, 8 and 9 show the accumulated profits (relative to the initial investments) in the experiments with proposed method MKL-3-GA method.

Figures 6, 7, 8 and 9 show that how the large returns were made had interesting characteristics: in 2008 most of the profits were made during August to September but in 2010, most of the profits were made during March to May, whereas most of the profits were made in July to August in 2011. Note that those profits are made by several moderately large jumps of profits in their accumulation, and also note that, although the most profit comes from these short periods, small but steady profits come from the other periods. In the other year, these small but steady profits accumulate to large profits without moderately large jumps. For example, profits were made in most of the year in 2009, almost from the start to the finish, and the final profit was 20 %. Similar results were reported in a previous study Fletcher et al. (2010), where the model lacked high directional predictability but it yielded good profits because some transactions had enhanced profitability by trading more when in confidence of predicted direction and trading less when in less confidence.

6.3 Statistical Test of Returns with Proposed Method MKL-3-GA

We conducted a statistical analysis of the returns from the experiments for 2008–2011 (76 experiments in total, 19 in each year). We tested whether the average returns from the 76 experiments was positive. The sample size was 76, so we applied a large sample

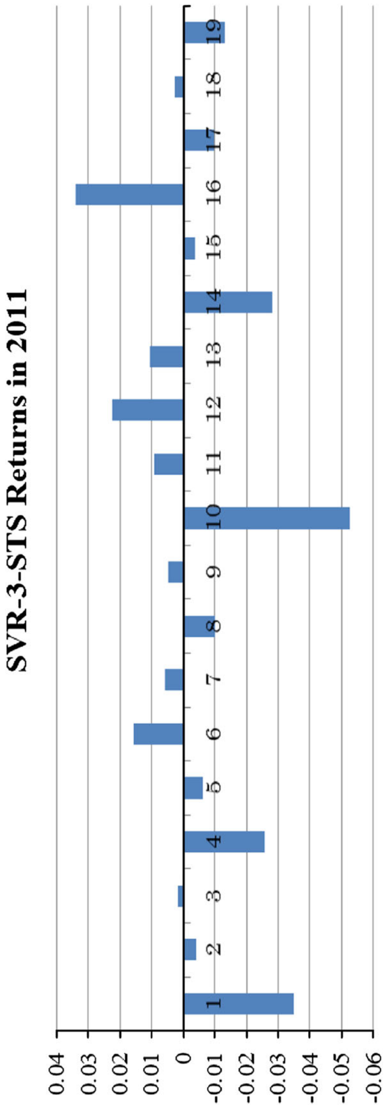

Fig. 5 Distribution of returns in each trading period during 2010 (SVR-3-STS)

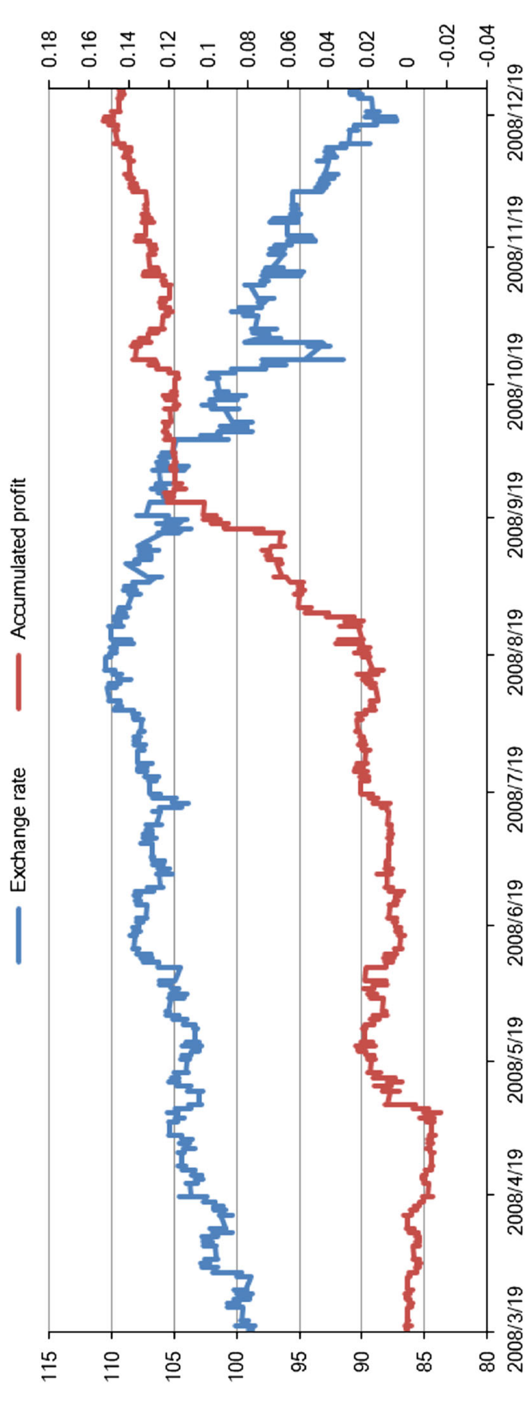

Fig. 6 Chart of USD/JPY profit accumulated by MKL-3-GA from March 2008 to December 2008 (in proportion to initial investment)

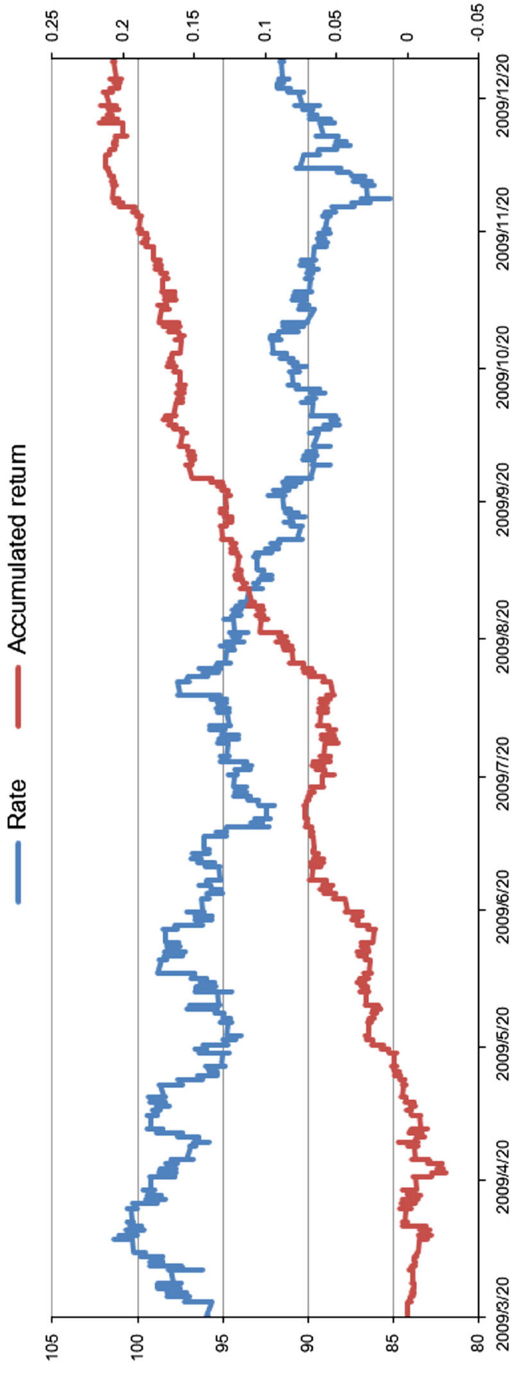

Fig. 7 Chart of USD/JPY profit accumulated by MKL-3-GA from March 2009 to December 2009 (in proportion to initial investment)

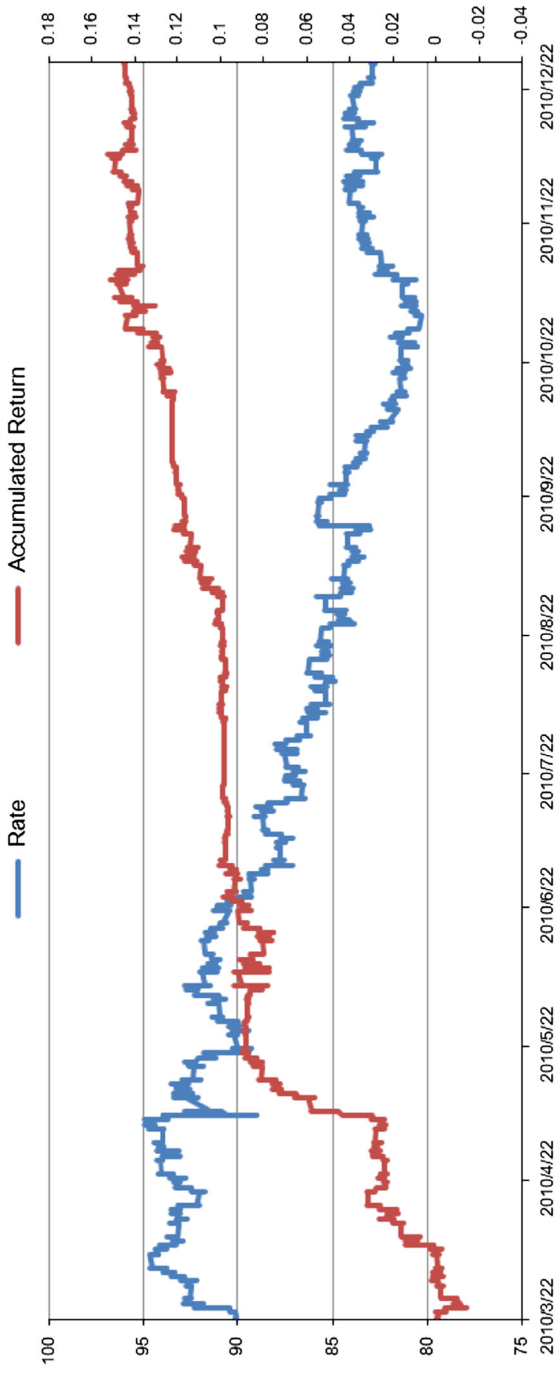

Fig. 8 Chart of USD/JPY profit accumulated by MKL-3-GA from March 2010 to December 2010 (in proportion to initial investment)

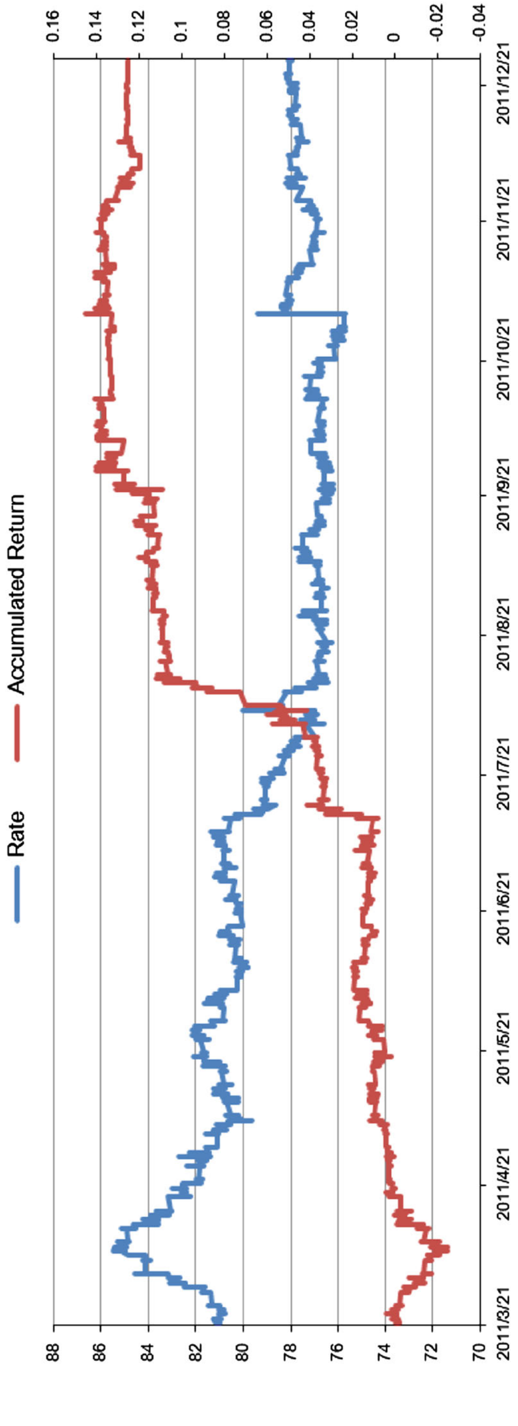

Fig. 9 Chart of USD/JPY profit accumulated by MKL-3-GA from March 2011 to December 2011 (in proportion to initial investment)

Table 14 Sharpe ratios with the proposed method and the baseline methods

Model	Average return per year (2008–2011)	Std. dev.	Sharpe ratio
SVR-1-STS	−0.33488	0.08239	−4.07751
SVR-3-STS	−0.15354	0.11809	−1.30945
SVR-5-STS	−0.09430	0.15599	0.59759
MKL-3-STS	−0.02518	0.08031	−0.32718
MKL-5-STS	0.08373	0.07309	1.13066
SVR-1-GA	0.04490	0.02297	1.90724
SVR-3-GA	0.03462	0.01558	2.15199
SVR-5-GA	0.01033	0.00422	2.18863
MKL-3-GA	0.15490	0.03521	4.36833
MKL-5-GA	0.11808	0.04684	2.49758
Buy and hold	−0.06064	0.02462	−2.50714
Sell and hold	0.06064	0.02462	2.41877
ANN	−0.06665	0.05833	−1.15942
ARIMA	−0.06393	0.05969	−1.08906

method. Thus, we assumed that the distribution of the sample mean was approximated by a normal distribution and the one-sided p-value for the null hypothesis that "the average return is 0" was 1.077×10^{-6} because the sample mean and standard deviation were 0.008153 and 0.01500, respectively. Therefore, the null hypothesis was rejected with a high confidence level and we had a high level of confidence that the average profits with our proposed method MKL-3-GA were positive.

6.4 Sharpe Ratios Values with the Proposed and Benchmark Methods

We evaluated the Sharpe ratio for our proposed method and the other baseline methods over the testing periods. The interest rate set by the Bank of Japan from 2008 to 2011 was considered the risk-free return, which was 0.25 % in 2008 and 0.1 % from 2009 to 2011. We calculated the average returns of interest from 2008 to 2011 and the average risk-free return of each testing year (10 months in each year) was 0.1088 %. Table 14 shows the average returns, standard deviations, and Sharpe ratios for each method.

A higher Sharpe ratio indicates a higher return or lower volatility. In some testing periods, several baseline methods obtained higher returns than our proposed method, but in the longer term our proposed method had a significantly higher Sharpe ratio than the baseline methods throughout the testing periods. We might expect improvements of the returns and the Sharpe ratios with the other methods if we introduced the additional tools used in our proposed methods. For example, we did not implement risk management in the simple trading strategy (STS) (i.e., we did not use stop-loss and take-profit orders). Indeed, the huge losses with SVR-1-STS in 2008, SVR-3-STS in 2009, and MKL-3-STS in 2010 could have been reduced if we had implemented risk management. However, we could not estimate the appropriate levels for the take-profit and stop-loss points in percentage terms without learning.

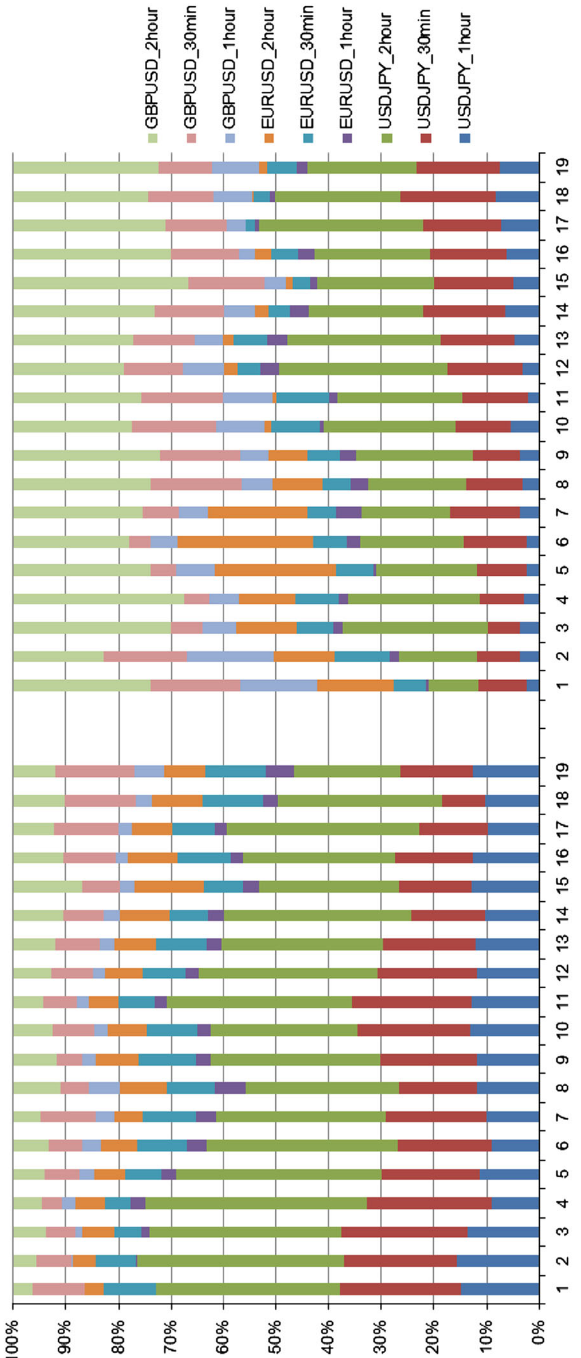

Fig. 10 Coefficients of MK regression (three currency pairs) for 2008 and 2009

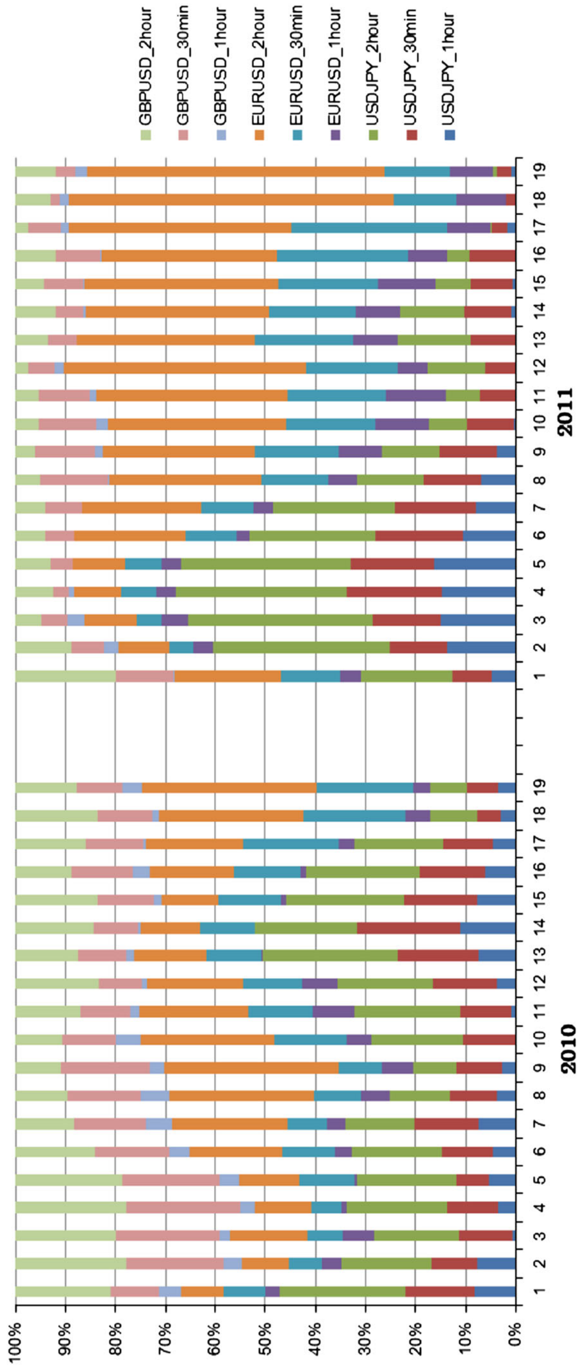

Fig. 11 Coefficients of MK regression (three currency pairs) for 2010 and 2011

Table 15 Summary of the properties of weightings for specific currency pairs and time horizons (three currency pairs)

Year	Consistently small weight	Consistently large weight	Weight changes rapidly
2008	1-h GBP/USD, 1-h EUR/USD	2-h USD/JPY	
2009	1-h EUR/USD	2-h GBP/USD	2-h EUR/USD
2010	1-h GBP/USD, 1-h EUR/USD	2-h USD/JPY	
2011	1-h GBP/USD	2-h EUR/USD	2-h USD/JPY

6.5 Weight Results for Multiple Kernel Learning

6.5.1 Weight Results for MKL-3

There were 19 training and testing sequences in each year. The x-axis values (1–19) indicate the indexes of the testing periods, which were numbered as our rolling windows proceeded. We had 19 learning and testing pairs in each year. The y-axis represents the relative weights of the currency pairs and the time horizons.

Figures 10 and 11 show the weights for three currency pairs with three time horizons during the training periods of 2008 to 2011. These figures show that the relative weights of the 2-h USD/JPY were higher than the others in most of the MKL training periods. Thus, the 2-h USD/JPY could be important references for identifying the trends in our target USD/JPY with a 1-h time horizon. In addition, the weight of the 30-min USD/JPY was more stable than the others in the 2008–2011 training periods.

However, some currency pairs with specific time horizons had weights that were consistently smaller than others during the training periods of 2008–2011. For example, the weight of the 1-h EUR/USD ranged from 0 to 2.5 % from 2008 to 2010. Table 15 provides a summary of the properties of the weights for specific currency pairs and their time horizons.

6.5.2 Weight Results for MKL-5

Figures 12 and 13 show the weights for five currency pairs with three time horizons in the training periods of 2008 to 2011. These weighting results show that the 2-h USD/CHF and 2-h EUR/USD changed greatly over time. For example, the 2-h USD/CHF had a relatively high weight in 2011 but a very low weight in 2009. In addition, the relative weight of the 2-h AUD/USD was larger than the others in most of the MKL training periods from 2008 to 2011. Thus, the 2-h AUD/USD could be an important reference for identifying the trend in our target USD/JPY with a 1-h time horizon.

However, some currency pairs with specific time horizons had consistently lower weights than others during the training periods of 2008–2011. For example, the weight of the 1-h EUR/USD ranged from 0 to 2 %. Table 16 provides a summary of the properties of the weights of specific currency pairs with their time horizons.

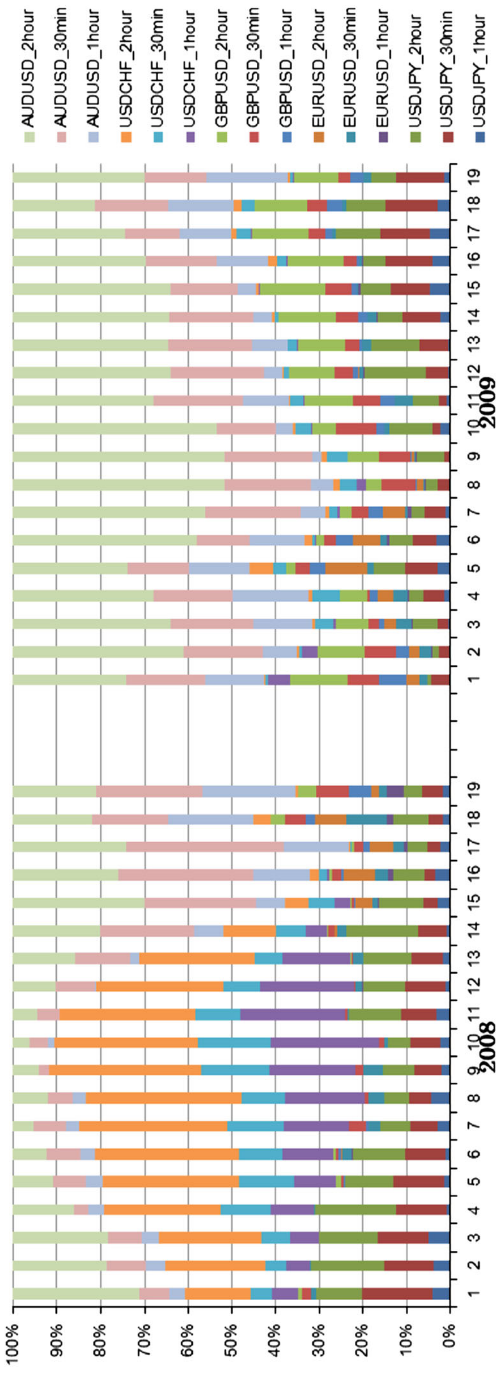

Fig. 12 Coefficients of MK regression (five currency pairs) for 2008 and 2009

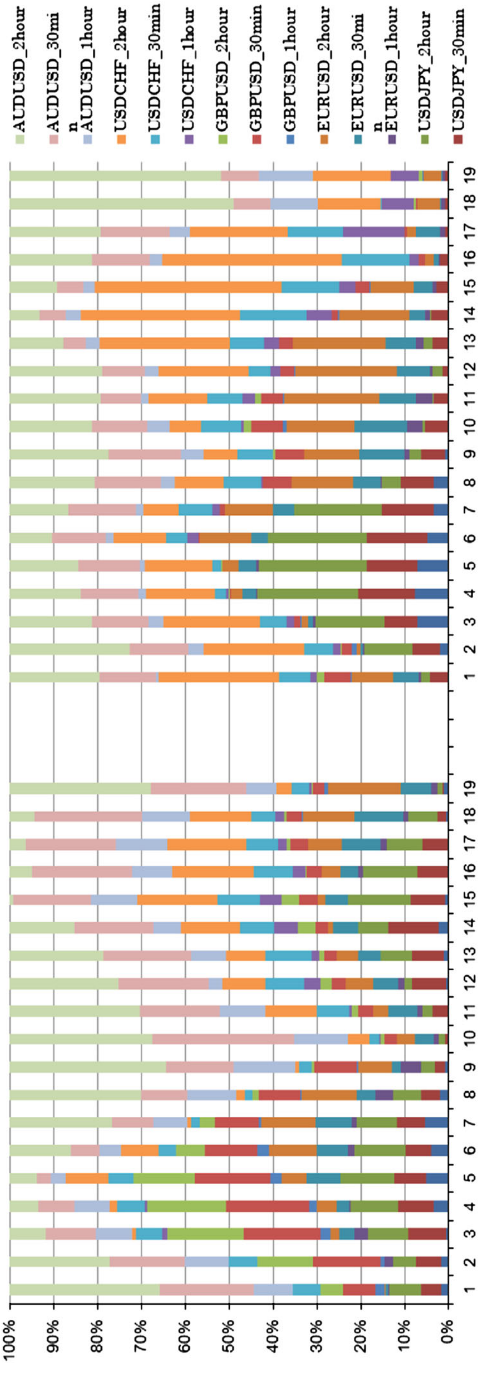

Fig. 13 Coefficients of MK regression (five currency pairs) for 2010 and 2011

Table 16 Summary of the properties of weightings for specific currency pairs and time horizons (five currency pairs)

Year	Consistently small weight	Consistently large weight	Weight changes rapidly
2008	30-min, 1-h, 2-h GBP/USD;		1-h AUD/USD;
	30-min, 1-h, 2-h EUR/USD;		2-h USD/JPY
	1-h USD/JPY		
2009	30-min, 1-h, 2-h EUR/USD;	30-min, 2-h AUD/USD	
	30-min, 1-h, 2-h USD/CHF;		
	1-h GBP/USD		
2010	1-h USD/CHF;		2-h AUD/USD;
	1-h GBP/USD;		2-h USD/CHF
	1-h EUR/USD		
2011	30-min, 1-h, 2-h GBP/USD;	2-h USD/CHF	30-min, 1-h USD/CHF;
	1-h EUR/USD		2-h AUD/USD;
			2-h EUR/USD

Initially, we expected that GBP/USD and EUR/USD would be more important than USD/CHF and AUD/USD for our target USD/JPY prediction. However, we found that the weights of three time horizons were consistently smaller for EUR/USD and GBP/USD over the years than for USD/CHF and AUD/USD, which contradicted our expectation. However, we also found that the weights of AUD/USD and USD/CHF usually changed rapidly, which may suggest that these are important for our predictions, although the importance might not be stable over different periods.

7 Conclusions and Future Works

In this study, we developed a hybrid method, which combined MK regression with a GA, where MK regression was used to build a prediction model and the GA was used to formulate a trading strategy. The MACDs of the main (three or five) currency pairs and three different short time horizons were used to make predictions. We used one Gaussian kernel and one linear kernel for each currency pair, and we used the MK method to combine the features of different currency pairs.

First, we applied MK regression to FX data for a specific training period to estimate the optimal parameters and weights. The results showed that the FX rate changes predicted by MK regression were much better than those predicted by SVR during our testing period, in terms of the RMSEs. Next, we used the GA to optimize the trading strategy using the predicted FX rate changes and the overbought/oversold indicators for the training period.

We traded USD/JPY based on the trading strategies generated using the GA, and calculating the returns and Sharpe ratios for the testing periods from March to December between 2008 and 2011. Each testing period was 250 h (around 2 weeks). In some testing periods, several baseline methods outperformed our proposed methods in terms of profit, but our proposed methods obtained consistently good profits and Sharpe ratios without experiencing losses in any year. The profits ranged from 7.1 to 20.60 % per year from 2008 to 2011, and the Sharpe ratios for MKL-3-GA and MKL-5-GA were 4.36 and 2.49, respectively. The average profit obtained during the testing period was positive with a statistically high confidence (higher than 99 % confidence). In short, our proposed method obtained consistently favorable returns with low volatility over a 4-year period.

The weights obtained using multiple kernel learning, which were applied to the regression function to determine the FX rate changes with different currency pairs and different time horizons, showed that there was a possible correlation between our target pair (USD/JPY), its target trading time interval (1 h), and other currency pairs, or other time horizons. The relative weights of the kernels calculated from the results of multiple kernel learning could be utilized by traders to identify possible correlations between reference currency pairs with reference time horizons and the target trading currency pair with the target time horizon.

The problems that remain include the determination of the best time horizon lengths for use as the references in our target trading time horizon. The selection of a very long or very short time horizon may also have negative effect on the predictions.

We only determined the relative weights among currency pairs with certain time horizons, because we did not consider all of the possible time horizons for all of the different pairs during MKL training. In addition, changing the input features might produce different weights for different currency pairs and different time horizons (we used the MACD indicator as the feature for MK regression). Changing the input features (using other indicators or other transforms from raw data) will be a future direction in our research.

Acknowledgments This research was partially supported by the fund "Keio Leading-edge Laboratory of Science & Technology Research Grant" and by the Global-COE program (Symbiotic, Safe and Secure System Design) of Keio University, Japan. The authors would like to thank ICAP and Forexite for making their foreign exchange data available for this research.

References

Allen, F., & Karjalainen, R. (1999). Using genetic algorithms to find technical trading rules. *Journal of Financial Economics*, *51*, 245–271.

Azoff, E. M. (1994). *Neural networks time series forecasting of financial markets*. Chichester: Wiley.

Bach, F. R., Lanckriet, G. R. G., & Jordan, M. I. (2004). Multiple kernel learning, conic duality, and the SMO algorithm. *Proceedings of the 21st International Conference on, Machine Learning* (pp. 6–13).

Badawy, F. A., Abdelazim, H. Y., & Darwish, M. G. (2005). Genetic algorithms for predicting the Egyptian stock market. *Proceedings of ITI Third International Conference on Information and Communication Technology Enabling Technologies for the New Knowledge Society* (pp. 109–122).

Banzhaf, W., Nordin, P., Keller, R. E., & Francone, F. D. (1998). *Genetic Programming—An introduction: On the automatic evolution of computer programs and its applications*. San Francisco: Kaufmann.

Barnett, W. A., Gallant, A. R., Hinich, M. J., Jungeilges, J. A., Kaplan, D. T., & Jensen, M. J. (1997). A single-blind controlled competition among tests for nonlinearity and chaos. *Journal of econometrics*, *82*(1), 157–192.

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, *31*, 307–327.
Bonilla, C. A., Romero-Meza, R., & Maquieira, C. (2011). Nonlinearities and GARCH inadequacy for modeling stock market returns: Empirical evidence from Latin America. *Macroeconomic Dynamics*, *15*(5), 713–724.
Box, G. E. P., & Jenkins, G. M. (1994). *Time series analysis: Forecasting and control* (3rd ed.). Englewood Cliffs: Prentice Hall.
Broock, W. A., Scheinkman, J. A., Dechert, W. D., & LeBaron, B. (1996). A test for independence based on the correlation dimension. *Econometric Reviews*, *15*(3), 197–235.
Cao, L. J. (2003). Support vector machine experts for time series prediction. *Neurocomputing*, *51*, 321–339.
Champernowne, D. G. (1948). Sampling theory applied to autoregressive schemes. *Journal of the Royal Statistical Society: Series B*, *10*, 204–231.
Chen, A.-S., & Leung, M. T. (2004). Regression neural network for error correction in foreign exchange forecasting and trading. *Computers and Operations Research*, *31*, 1049–1068.
Chu, S. C. W., Ng, H. S., & Lam, K. P. (2000). Intelligent trading advisor. *Proceedings of 2000 IEEE International Conference on Management of Innovation and Technology* (Vol. 1, pp. 53–58).
De Faria, E. L., Albuquerque, M. P., Gonzalez, J. L., Cavalcante, J. T. P., & Albuquerque, M. P. (2009). Predicting the Brazilian stock market through neural networks and adaptive exponential smoothing methods. *Expert Systems with Applications*, *36*(10), 12506–12509.
Dempster, M. A. H., & Jones, C. M. (2002). Can channel pattern trading be profitably automated? *The European Journal of Finance*, *8*(3), 275–301.
Deng, S., & Sakurai, A. (2011). Combining multiple kernel learning and genetic algorithm for forecasting short time foreign exchange rate. *Proceedings of AIA*, *2011*. doi:10.2316/P.2011.717-111.
Drożdż, S., Górski, A. Z., & Kwapień, J. (2007). World currency exchange rate cross-correlations. *The European Physical Journal B*, *58*(4), 499–502.
Fletcher, T., Hussain, Z., & Shawe-Taylor, J. (2010). Multiple kernel learning on the limit order book. *JMLR: Workshop and Conference Proceedings*, *11*, 167–174.
Foresti, L., Tuia, D., Pozdnoukhov, A., & Kanevski, M. (2009). Multiple kernel learning of environmental data. Case study: Analysis and mapping of wind fields. In C. Alippi, M. Polycarpou, C. Panayiotou, & G. Ellinas (Eds.), *Artificial neural networks-ICANN. Lecture notes in computer science* (pp. 933–943). Berlin: Springer.
Frick, A., Herrmann, R., Kreidler, M., Narr, A., & Seese, D. (1996). Genetic based trading rules: A new tool to beat the market with? *Proceedings of 6th International AFIR Colloquium* (Vol. 1/2, pp. 997–1018.)
Fuente, D., Garrido, A., Laviada, J., & Gomez, A. (2006). Genetic algorithms to optimize the time to make stock market investment. In *GECCO 2006 Conference Proceedings* (Vol. 2, pp. 1857–1858).
Gencay, R., Dacarogna, M., Olsen, R., & Pictet, O. (2003). Foreign exchange trading models and market behavior. *Journal of Economic Dynamics and Control*, *27*, 909–935.
Gehrig, T., & Menkhoff, L. (2006). Extended evidence on the use of technical analysis in foreign exchange. *International Journal of Finance and Economics*, *11*(4), 327–338.
Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading, MA: Addison Wesley.
Hann, T. H., & Steurer, E. (1996). Much ado about nothing? Exchange rate forecasting: Neural networks vs. linear models using monthly and weekly data. *Neurocomputing*, *10*, 323–339.
Hirabayashi, A., Aranha, C., & Iba, H. (2009). Optimization of the trading rule in foreign exchange using genetic algorithm. In *GECCO 2009 Conference Proceedings* (pp. 1529–1536).
Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. *Proceedings of the European Conference on Machine Learning (ECML)* (Vol. 1398, pp. 137–142).
Joutou, T., & Yanai, K. (2009). A food image recognition system with multiple kernel learning. *Proceedings of International Conference on Image Processing* (pp. 285–288).
Kodogiannis, V., & Lolis, A. (2002). Forecasting financial time series using neural network and fuzzy system-based techniques. *Neural Computing and Applications*, *11*, 90–102.
Koskela, T., Varsta, M., Heikkonen, J., & Kaski, K. (1997). Time series prediction using recurrent SOM with local linear models. *International Journal of Knowledge-Based Intelligent Engineering Systems*, *2*, 60–68.
Kwapień, J., Gworek, S., Drożdż, S., & Górski, A. (2009). Analysis of a network structure of the foreign currency exchange market. *Journal of Economic Interaction and Coordination*, *4*(1), 55–72.

Kwok, J. T.-Y. (2000). The evidence framework applied to support vector machines. *IEEE Transactions on Neural Networks*, *11*(5), 1162–1173.

Lee, M. (2003). Common trend and common currency: Australia and New Zealand. *International Journal of Business and Economics*, *2*, 155–165.

Luss, R., & d'Aspremont, A. (2012). Predicting abnormal returns from news using text classification. Quantitative Finance, 1–14. doi:10.1080/14697688.2012.672762.

Meese, R. A., & Rogoff, K. (1983). Empirical exchange rate models of the seventies: Do they fit out of sample? *Journal of International Economics*, *14*, 3–24.

Mehdi, K., & Mehdi, B. (2011). A novel hybridization of artificial neural networks and ARIMA models for time series forecasting. *Applied Soft Computing*, *11*, 2664–2675.

Mizuno, T., Takayasu, H., & Takayasu, M. (2006). Correlation networks among currencies. *Physica A*, *364*, 336–342.

Neely, C. J., & Weller, P. A. (2003). Intraday technical trading in the foreign exchange market. *Journal of International Money and Finance*, *22*, 223–237.

Olson, D. (2004). Have trading rule profits in the currency market declined over time? *Journal of Banking and Finance*, *28*, 85–105.

Pai, P.-F., & Lin, C.-S. (2005). A hybrid ARIMA and support vector machines model in stock price forecasting. *Omega: The International Journal of Management Science*, *33*(6), 497–505.

Park, C.-H., & Irwin, S. H. (2004). *The profitability of technical analysis: A review, AgMAS project research report no. 2004-04*.

Pesaran, M. H., & Allan, T. (1995). Predictability of stock returns: Robustness and economic significance. *The Journal of Finance*, *50*(4), 1201–1228.

Ravindran, R., Ganisen, S., & Roslan, M. R. (2008). Exchange rate forecasting. *Annual London Conference on Money, Economy and Management : An Application of Higher Order ARIMA and GARCH models*.

Riccardo, C., Charles, G., Dominique, G., & Richard, P. (1997). Do technical trading rules generate profits? Conclusions from the intra-day foreign exchange market. *International Journal of Finance & Economics*, *2*(4), 267–280.

Schulmeister, S. (2007). The profitability of technical stock trading has moved from daily to intraday data. *WIFO working papers*.

Sermpinis, G., Jason, L., Andreas, K., & Christian, L. D. (2012). Forecasting and trading the EUR/USD exchange rate with gene expression and psi sigma neural networks. *Expert Systems with Applications*, *39*(10), 8865–8877.

Sharpe, W. F. (1994). The Sharpe ratio. *Journal of Portfolio Management*, *21*(1), 49–58.

Sonnenburg, S., Rätsch, G., Schäfer, C., & Schölkopf, B. (2006). Large scale multiple kernel learning. *Journal of Machine Learning Research*, *7*, 1531–1565.

Stawicki, S. P. (2007). Application of financial analysis techniques to vital sign data: A novel method of trend interpretation in the intensive care unit. OPUS 12. *Scientist*, *1*(1), 14–16.

Vapnik, V. (1995). *The nature of statistical learning theory*. New York: Springer.

Wilder, J. W. (1978). *New concepts in technical trading systems* (1st ed.). Greensboro, NC: Trend Research.

Wong, B. K., & Selvi, Y. (1998). Neural network applications in finance: A review and analysis of literatures (1990–1996). *Information & Management*, *34*, 129–139.

Williams, L. (2005). *Trade stocks commodities with the insiders: Secrets of the COT report*. Hoboken, NJ: Wiley.

Wu, M-Ch. (2007). Phase correlation of foreign exchange time series. *Physica A: Statistical Mechanics and its Applications*, *375*(2), 633–642.

Yeh, C.-Y., Huang, C.-W., & Lee, S.-J. (2011). A multiple-kernel support vector regression approach for stock market price forecasting. *Expert Systems with Applications*, *38*(3), 2177–2186.

Zhao, Q.-Y., Zhao, X.-Y., & Duan, F. (2009). Prediction model of stock prices based on correlative analysis and neural networks. *Proceedings of the 2009 Second International Conference on Information and Computing Science* (Vol. 3, pp. 189–192).