Comput Stat (2015) 30:821–843
DOI 10.1007/s00180-014-0543-9

ORIGINAL PAPER

Recurrent support vector regression for a non-linear ARMA model with applications to forecasting financial returns

Shiyi Chen · Kiho Jeong · Wolfgang K. Härdle

Received: 7 September 2013 / Accepted: 10 November 2014 / Published online: 18 November 2014

Abstract Motivated by recurrent neural networks, this paper proposes a recurrent support vector regression (SVR) procedure to forecast nonlinear ARMA model based simulated data and real data of financial returns. The forecasting ability of the recurrent SVR based ARMA model is compared with five competing models (random walk, threshold ARMA model, MLE based ARMA model, recurrent artificial neural network based ARMA model and feed-forward SVR based ARMA model) by using two forecasting accuracy evaluation metrics (NSME and sign) and robust Diebold–Mariano test. The results reveal that for one-step-ahead forecasting, the recurrent SVR model is consistently better than the benchmark models in forecasting both the magnitude and turning points, and statistically improves the forecasting performance as opposed to the usual feed-forward SVR.

Keywords Recurrent support vector regression · Non-linear ARMA · Financial forecasting

S. Chen (✉)
China Center for Economic Studies, School of Economics,
Fudan University, Handan Road 220, Shanghai 200433, China
e-mail: shiyichen@fudan.edu.cn

K. Jeong (✉)
School of Economics and Trade, Kyungpook National University,
Sankyuk-dong 1370, Daegu 702-701, Korea
e-mail: khjeong@knu.ac.kr

W. K. Härdle
Center for Applied Statistics and Economics, Humboldt-Universität zu Berlin,
Spandauer Straße 1, 10178 Berlin, Germany

W. K. Härdle
Lee Kong Chian School of Business, Singapore Management University, Singapore, Singapore

JEL Classification C45 · C53 · F37 · F47 · G17

1 Introduction

This paper considers financial returns forecasting in the framework of a univariate autoregressive moving average (ARMA) model by using the proposed recurrent ε-SVR approach. For more than two decades, the linear ARMA model estimated by maximum likelihood estimation (MLE) has been a popular approach for forecasting non-stationary time series, as opposed to simple moving average methods and the random walk. This implies that the analysed variables should satisfy the normal assumption and have a large sample. However, it has been widely accepted that the returns of a variety of financial variables are not linearly predictable in general and the phenomenon of volatility clustering in it leads to the violation of the normal assumption, as a result of which, the linear ARMA model by MLE usually tends to provide poor forecasting performance (Priestley 1988; Box et al. 1994; Niemira and Klein 1994; Hamilton 1997). Thus, some non-linear alternative ARMA models are proposed and adopted to estimate the time series, including both parametric specifications such as regime-switching or threshold ARMA models and non-parametric ARMA models. The prevailing method to estimate non-parametric time series models is the artificial neural network (ANN). Plenty of studies on the ANN method denote that the ANN approach outperforms traditional MLE in forecasting financial time series and particularly, the recurrent ANN with richer dynamic structure could capture more characteristics of data in the generalisation period than the feed-forward one (Kuan and Liu 1995; Wu 1995; Tian et al. 1997; Lisi and Schiavo 1999; Ashok and Mitra 2002; Gaudart et al. 2004; Kamruzzaman and Sarker 2004), but some indicate mixed or opposite results (Adya and Collopy 1998; Kanas 2003). While the ANN is theoretically better at estimating non-linear finite samples without invoking a probabilistic distribution, it has however been criticized as being vulnerable to the over-fitting problem which usually leads to a local optimum and to empirical risk minimization, the same as the MLE,[1] the latter of which results in good fit and poor out-of-sample forecast. To avoid the theoretical pitfalls of the MLE and ANN in the forecasting area, fortunately, Vapnik (1995, 1997) has successfully developed a novel non-parametric function approximator, the Support Vector Machine (SVM), which is computationally powerful in the sense that it allows for (1) a finite and infinite sample; (2) no prior distribution assumption; and (3) minimising of the structural risk as opposed to empirical risk employed by MLE and ANN, which endows SVM with an excellent generalisation (or forecasting) ability out-of-sample and is the biggest advantage of SVM among all alternatives (We refer to Sect. 2 for a detailed explanation).

SVM was originally developed for classification problems (SVC) and then extended to regression problems (SVR). SVM has recently been successfully applied to financial variable classification and financial time series forecasting, for example, see Trafalis and Ince (2000), Cao and Tay (2001), Gestel et al. (Jul. 2001), Yang et al. (2002),

[1] For MLE, maximizing the joint probability density function amounts to minimizing the sum of residual squares, i.e., minimizing the empirical risk, which is equivalent to the OLS approach.

Härdle et al. (2005, 2006), Espinoza et al. (2006) and Lee et al. (2006), to name a few. As Haykin (1999) argued earlier, the present studies on SVM mostly focus on the feed-forward direction and the previous application literatures of the SVR based time series forecasting only consider the dynamic systems of non-linear Autoregressive (AR) model. In a context of networks, these systems do not have feedback loops from the output or hidden layer to the input layers. It is well known that recurrent ANN, networks with feedbacks, can characterize the behaviour of time series variables with richer dynamic structures and have more potential to significantly reduce the memory requirement than the feed-forward one (Kuan et al. 1994; Kuan 1995; Kuan and Liu 1995). Suykens and Vandewalle (Jul. 2000) and Suykens et al. (2002) extend the recurrent networks to support vector machine and proposed a new recurrent least squares SVM (LS-SVM) procedure. Hong (2011) also introduces a feedback Jordan network into the SVR procedure when forecasting monthly electric loads. Their studies reveal that the recurrent SVR procedure can forecast time series very well.

Also motivated by the recurrent ANN, in this paper, we propose a new ε-insensitive loss based support vector regression (SVR) procedure with the addition of a global feedback connection from the output layer to the input space. In terms of the terminology of the recurrent LS-SVM, we refer to the proposed procedure as a recurrent ε-SVR and to the standard SVR as a feed-forward SVR. To examine the sensitivity of the recurrent ε-SVR with respect to free parameters, we experiment with three free parameters, ε, C and σ^2 by using a cross validation method. The difference between the recurrent LS-SVM and our recurrent ε-SVR is that the different empirical loss functions are used; the former adopts the mean square error (MSE), the latter uses the ε-insensitive error which can lead to sparseness solutions (see Sect. 2 for details). Different from our recurrent design which includes a feedback loop from the output layer directly to input layer, Hong (2011) recurrent SVR applies a Jordan network proposed by Jordan (1987) as a recurrent learning mechanism framework in which a feedback loop is introduced from the output layer to an additional context layer, and then the output values from the context layer are fed back into the hidden layer. Also, different from the cross-validation approach used in this study, Hong (2011) combines the seasonal recurrent SVR with chaotic artificial bee colony algorithm (namely SRSVRCABC) to determine suitable values of the parameters of SVR.

In this paper, the proposed recurrent ε-SVR procedure will be applied to forecasting the ARMA model for the simulated data (linear ARMA series and non-linear Lorenz series) and the real data of financial returns [Canadian Dollar against the U.S. dollar (CAD) exchange rates and New York Stock Exchange^{TM} (NYSE) composite stock index]. The iterative epochs of the recurrent ε-SVR procedure are described in Sect. 2 and illustrated by the simulation data. The forecasting performance among the recurrent and feed-forward SVR ARMA models, recurrent ANN ARMA, MLE ARMA, threshold ARMA and random walk model is compared by using two forecasting evaluation metrics (NMSE and sign) in a one-step-ahead forecasting horizon, and the statistical hypothesis of equal forecasting accuracy between pairwise models is also investigated by using the Diebold and Mariano (1995) test, calculated according to Newey–West Procedure (Newey and West 1987). The Diebold and Mariano (DM) test is one of the most important contributions to the study of out-of-sample forecasting accuracy evaluation over past two decades. This paper is organized as follows.

Section 2 introduces the theory of standard SVR and proposes the recurrent ε-SVR procedure. Section 3 specifies the empirical modelling and forecasting scheme. Section 4 compares the forecasting performance of all candidates by using the simulated and real data, in which the parameters selection and iterative process are illustrated in detail. The conclusion is presented in Sect. 5.

2 Support vector regression (SVR)

2.1 Principle of standard ε-SVR

The support vector machines for regression (SVR) originates from Vapnik's statistical learning theory (Vapnik 1995, 1997), which has the design of a feed-forward network with an input layer, a single hidden layer of non-linear units and an output layer and formulates the regression problem as a quadratic programming (QP) problem (Haykin 1999). SVR estimates a function by non-linearly mapping the input space into a high dimensional hidden space and then running the linear regression in the output space (see Fig. 1). Thus, the linear regression in the output space corresponds to a non-linear regression in the low dimensional input space. And the theory denotes that if the dimensions of feature space (or hidden space) are high enough, SVR may approximate any non-linear mapping relations. As the name implies, the design of the SVR hinges upon the extraction of a subset of the training data that serves as support vectors which represent a stable characteristic of the data.

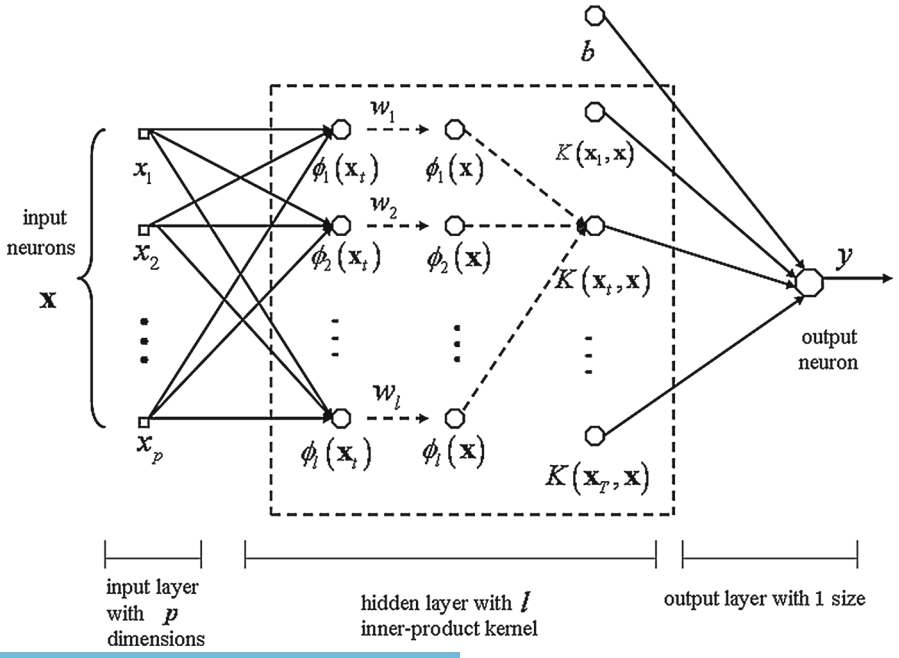

Fig. 1 Architecture of support vector machines

Given a training data set $\{(\mathbf{x}_t, y_t)\}_{t=1}^{T}$, where inputs vector $\mathbf{x}_t \in R^p$ and output scalar $y_t \in R^1$. In classification problem, the variable y only takes two values, -1 and 1; while in regression y can take any real values. Indeed, the desired response y, known as a 'teacher', represents the optimum action to be performed by the SVR. We aim at finding a sample regression function $f(\mathbf{x})$ (or denoted by $\widehat{y}$) as below to approximate the latent, unknown decision function $g(\mathbf{x})$.

$$f(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b \tag{1}$$

where $\phi(\mathbf{x}) = [\phi_1(\mathbf{x}), \ldots, \phi_l(\mathbf{x})]^T$, $\mathbf{w} = [w_1, \ldots, w_l]^T$. The $\phi(\mathbf{x})$ is known as the non-linear transfer function which represents the features of the input space and projects the inputs into the feature space. The dimension of the feature space is l which is directly related to the capacity of the SVR to approximate a smooth input–output mapping; the higher the dimension of the feature space, the more accurate the approximation will be. Parameter $\mathbf{w}$ denotes a set of linear weights connecting the feature space to the output space, and b is the threshold.

To get the function $f(\mathbf{x})$, the optimal $\mathbf{w}^*$ and b^* have to be estimated from the data. Firstly, we define a linear ε-insensitive loss function, L_ε, originally proposed by Vapnik (1995).

$$L_\varepsilon(\mathbf{x}, y, f(\mathbf{x})) = \begin{cases} |y - f(\mathbf{x})| - \varepsilon & for\ |y - f(\mathbf{x})| \geq \varepsilon \\ 0 & otherwise \end{cases} \tag{2}$$

This function indicates the fact that it does not penalize errors below ε. The training points within the ε-tube have no loss and do not provide any information for decision. Therefore, these points do not appear in the decision function $f(\mathbf{x})$. Only those data points located on or outside the ε-tube will serve as the support vectors to be finally used to construct the $f(\mathbf{x})$. The sparseness property of the algorithm results only from the ε-insensitive loss function and greatly simplifies the computation of the SVR. Thus, the SVR based on it is also called ε-SVR, which is different from the other loss functions such as (mean) squared errors (MSE). The non-negative slack variables, ξ and ξ' (below or above the ε-tube, or denoted together by $\xi^{(\prime)}$; see Fig. 2) are employed to describe this kind of ε-insensitive loss, that is, the loss of error on training points out of the ε-tube.

The derivation of SVR follows the principle of structural risk minimization that is rooted in VC dimension theory. Structural risk is the upper boundary of empirical loss, denoted by ε-insensitive loss function, plus the confidence interval (or called margin), which is constructed in Eq. (3). The primal constrained optimization problem of ε-SVR is obtained below:

$$\min_{\mathbf{w}\in R^l, \xi^{(\prime)}\in R^{2T}, b\in R} \mathbf{C}\left(\mathbf{w}, b, \xi_t, \xi_t'\right) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{t=1}^{T}\left(\xi_t + \xi_t'\right) \tag{3}$$

$$s.t. \quad \mathbf{w}^T\phi(\mathbf{x}_t) + b - y_t \leq \varepsilon + \xi_t \quad t = 1, 2, \ldots, T \tag{4}$$

$$y_t - \mathbf{w}^T\phi(\mathbf{x}_t) - b \leq \varepsilon + \xi_t' \quad t = 1, 2, \ldots, T \tag{5}$$

$$\xi_t \geq 0, \xi_t' \geq 0 \quad t = 1, 2, \ldots, T \tag{6}$$

The formulation of the cost function $\mathbf{C}\left(\mathbf{w}, b, \xi_t, \xi_t'\right)$ in Eq. (3) is in perfect accord with the principle of structural risk minimisation, which is illustrated in Fig. 2 (in which the dark circles are data points extracted as support vectors). In Eq. (3), the first term indicates the Euclidean norm of the weight vector $\mathbf{w}(\|\mathbf{w}\|^2 = \mathbf{w}^T\mathbf{w})$ and measures the function flatness; the minimization of it is related to the maximization of the margin of separation $(2/\|\mathbf{w}\|)$, i.e., maximizing the generalisation ability. The second term represents the empirical risk loss determined by the ε-insensitive loss function and is similar to the sum of residual squares in the objective function of MLE and ANN. Finally, SVR obtains the trade-off between the two terms; as a result, it not only well fits the historical data but excellently forecasts the future data. As shown in Fig. 2, both regression lines 1 and 2 can classify the data points correctly and then minimize the empirical loss; however, the margins of generalisation of the two lines are different in which the regression line 1 has the largest margin. It is the special design of minimizing the structural risk that endows SVR with the excellent forecasting ability among all candidates. Evgeniou et al. (2002) also denoted that minimization of an empirical error only is both ill-posed and does not necessarily lead to models with good predictive capabilities, thus, one needs to minimize a structural risk. In addition, the convex quadratic programming and linear restrictions in the above primal problem ensure that SVR can always obtain the global unique optimal solution, which is different from the usual networks that easily get trapped in local minima. The penalty parameter $C > 0$ controls the penalizing extent on the sample which lie out of the ε-tube. Both ε and C must be selected by the user.

The corresponding dual problem of the ε-SVR can be derived from the primal problem by using the Karush–Kuhn–Tucker conditions as follows.

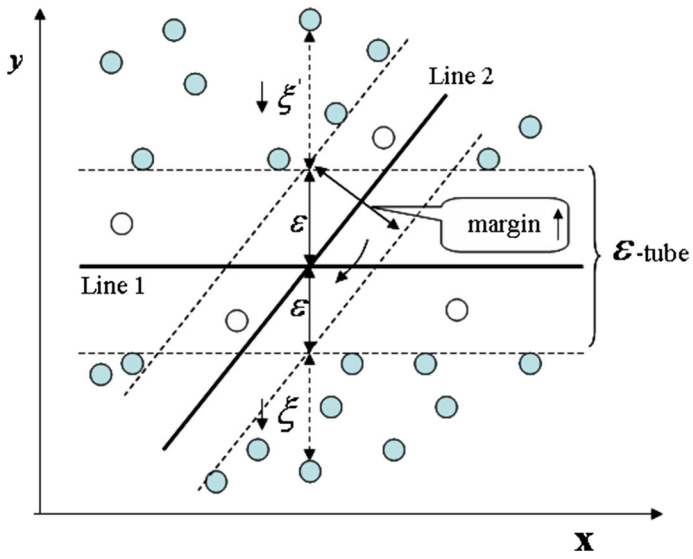

Fig. 2 Principle of structural risk minimization of ε-SVR

$$\min_{\alpha_t^{(\prime)} \in R^{2T}} \quad \frac{1}{2} \sum_{s=1}^{T} \sum_{t=1}^{T} \left(\alpha_s' - \alpha_s\right) \left(\alpha_t' - \alpha_t\right) K\left(x_s \cdot x_t\right)$$

$$+\varepsilon \sum_{t=1}^{T} \left(\alpha_t' + \alpha_t\right) - \sum_{t=1}^{T} y_t \left(\alpha_t' - \alpha_t\right) \tag{7}$$

$$s.t. \quad \sum_{t=1}^{T} \left(\alpha_t - \alpha_t'\right) = 0 \tag{8}$$

$$0 \leq \alpha_t, \alpha_t' \leq C \quad s, t = 1, 2, \ldots, T \tag{9}$$

where, α_t and α_t' (or $\alpha_t^{(\prime)}$) are the Lagrange multipliers. The dual problem can be solved more easily than the primal problem (Scholkopf and Smola 2001; Deng and Tian 2004). Making use of any solution, α_t and α_t', the optimal solutions of primal problem can be calculated, in which, $\mathbf{w}^*$ is unique and expressed as follows:

$$\mathbf{w}^* = \sum_{t=1}^{T} \left(\alpha_t' - \alpha_t\right) \phi\left(\mathbf{x}_t\right) \tag{10}$$

However, b^* is not unique and formulated in terms of different cases. If $i \in \{t | \alpha_t \in (0, C)\}$,

$$b^* = y_i - \sum_{t=1}^{T} \left(\alpha_t' - \alpha_t\right) K\left(\mathbf{x}_t \cdot \mathbf{x}_i\right) + \varepsilon \tag{11}$$

If $j \in \left\{t | \alpha_t' \in (0, C)\right\}$,

$$b^* = y_j - \sum_{t=1}^{T} \left(\alpha_t' - \alpha_t\right) K\left(\mathbf{x}_t \cdot \mathbf{x}_j\right) - \varepsilon \tag{12}$$

The cases of both $i, j \in \left\{t | \alpha_t^{(\prime)} = 0\right\}$ and $i, j \in \left\{t | \alpha_t^{(\prime)} = C\right\}$ rarely occur in reality.

Thus, the regression decision function $f\left(\mathbf{x}\right)$ will be computed by the use of $\mathbf{w}^*$ and b^* in the following forms:

$$\begin{aligned} f\left(\mathbf{x}\right) &= \mathbf{w}^{*T} \phi\left(\mathbf{x}\right) + b^* \\ &= \sum_{t=1}^{T} \left(\alpha_t' - \alpha_t\right) \phi^T\left(\mathbf{x}_t\right) \phi\left(\mathbf{x}\right) + b^* \\ &= \sum_{t=1}^{T} \left(\alpha_t' - \alpha_t\right) K\left(\mathbf{x}_t, \mathbf{x}\right) + b^* \end{aligned} \tag{13}$$

where $K\left(\mathbf{x}_t, \mathbf{x}\right) = \phi^T\left(\mathbf{x}_t\right) \phi\left(\mathbf{x}\right)$ is the inner-product kernel function. In fact, the SVR theory considers only the form of $K\left(\mathbf{x}_t, \mathbf{x}\right)$ in the feature space without specifying

$\phi(\mathbf{x})$ explicitly and without computing all corresponding inner products. Therefore, the kernel function greatly reduces the computational complexity of high dimensional hidden space and becomes the crucial part of SVR. The function which satisfies Mercer's theorem can be chosen as the SVR kernel. In this paper the chosen kernel is the widely-used Gaussian kernel, or called the radial based function (RBF) kernel which offers a way to measure proximity between two data points and is expressed as below.

$$K(\mathbf{x}_t, \mathbf{x}) = \exp\left(\frac{-\|\mathbf{x} - \mathbf{x}_t\|^2}{2\sigma^2}\right) \tag{14}$$

where σ^2 is the kernel width which implicitly controls the complexity of the feature space and the solution (the higher the σ^2 is, the lower the complexity is). For the Gaussian kernel, the explicit expression of non-linear transformation function $\phi(\mathbf{x})$ is unknown, and the corresponding feature dimension l is infinite.

2.2 Algorithm of recurrent ε-SVR

As Haykin (1999) said, the SVR described in Sect. 2.1 usually appears in the design of a simple feed-forward network in which an input layer of source nodes projects onto an output layer of computation node, but not vice versa, see Fig. 3a. This process is known as feed-forward SVR. If the in-sample fitting errors are white noise, or do not display auto-correlation, the feed-forward SVR is efficient in the sense that they can be utilized to estimate AR(p) model directly. Let $\mathbf{O}_t$ and $\mathbf{H}_t$ represent the single-output and l hidden unit activations. Symbolically, we have

$$\mathbf{O}_t = \psi\left(\mathbf{w}^T\mathbf{H}_t + b\right);\ \mathbf{H}_t = \phi(\mathbf{x}_t) \tag{15}$$

where $\mathbf{x}_t = \left\{\mathbf{x}_{t,i}\right\}_{i=1}^{p} = \{y_{t-i}\}_{i=1}^{p}$. Note that ψ and ϕ are vector-valued functions and represent the identity function and the transfer function to produce Gaussian kernel, respectively.

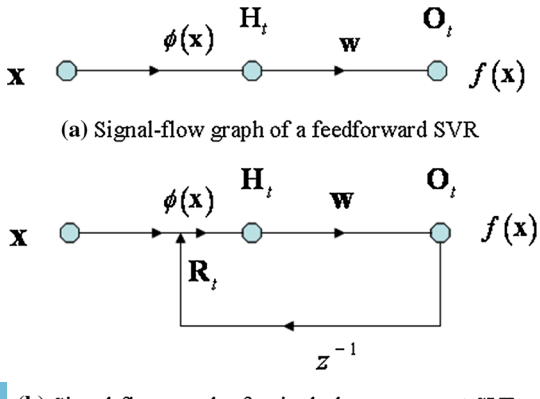

Fig. 3 Signal-flow graphs of feed-forward and recurrent SVR

If it is not the case, the information reflected behind the errors should be utilized to improve the estimating power of the model, thus, the ARMA model, i.e., introducing the error terms (MA part) into the AR model, becomes reasonable. To estimate the ARMA model, a feedback process of ε-SVR with an unobservable MA part as an input has to be described—which distinguishes itself from a feed-forward SVR in that it has at least one feedback loop (see Fig. 3b). In this paper, we abuse the terminology and refer to this process as "recurrent ε-SVR". The feedback loops involve the use of particular branches composed of *one-delay operator*, z^{-1}, which result in non-linear dynamical behaviour and have a profound impact on the learning capability of SVR. Thus, the recurrent ε-SVR will capture more dynamic characteristics of y_t than a feed-forward SVR does.

Let $\mathbf{R}_t$ denote one-delayed internal feedbacks. Then, the recurrent ε-SVR can be represented in the following generic form

$$\mathbf{O}_t = \psi\left(\mathbf{w}^T\mathbf{H}_t + b\right);\ \mathbf{H}_t = \phi\left(\mathbf{x}_t, \mathbf{R}_t\right) \tag{16}$$

where, $\mathbf{x}_t = \{y_{t-i}\}_{i=1}^{p}$. $\mathbf{R}_t$ is chosen to be $\mathbf{O}_{t-1}$; that is, the recurrent process has output feedbacks rather than hidden unit activations feedbacks. Thus, $\mathbf{R}_t$ can be expressed as

$$\mathbf{R}_t = \tau\left(\mathbf{x}_{t-1}, \mathbf{R}_{t-1}; \mathbf{w}, b\right) \tag{17}$$

with τ also a vector-valued function.

If $\mathbf{R}_t = \mathbf{0}$, the process simply reduces to a feed-forward SVR, in which the finite lagged responses are used as inputs to capture dynamics. This approach manifests the drawback that the correct lag length needed is typically unknown and somewhat difficult to determine. On the one hand, the finite lagged dependent variables may not be enough to capture certain temporal structures, especially those dependent on a long history of targets. On the other hand, storing all the past information in memory is practically implausible. The case is similar to building a linear AR model with finite p lags. This deficiency could be circumvented by our device of recurrent SVR. The feedback variable $\mathbf{R}_t$ will serve as a memory device to store past information compactly. That is,

$$\mathbf{R}_t = \tau\left(\mathbf{x}_{t-1}, \tau\left(\mathbf{x}_{t-2}, \mathbf{R}_{t-2}; \mathbf{w}, b\right); \mathbf{w}, b\right) = \cdots = \upsilon\left(\mathbf{x}_{t-1}, \mathbf{x}_{t-2}, \ldots, \mathbf{x}_1; \mathbf{w}, b\right) \tag{18}$$

Thus, the output of recurrent SVR can be written in the following feed-forward form

$$\begin{aligned}\mathbf{O}_t &= \psi\left(\mathbf{w}^T\phi\left(\mathbf{x}_t, \mathbf{R}_t\right) + b\right)\\ &= \kappa\left(\mathbf{x}_t, \mathbf{R}_t\left(\mathbf{w}, b\right); \mathbf{w}, b\right)\\ &= f\left(\mathbf{x}_t, \mathbf{x}_{t-1}, \mathbf{x}_{t-2}, \ldots, \mathbf{x}_1; \mathbf{w}, b\right)\end{aligned} \tag{19}$$

As $\mathbf{R}_t$ depends on the entire history of inputs, introducing recurrent variable $\mathbf{R}_t$ with the contraction mapping requirement of τ to a feed-forward SVR is similar to adding invertible moving average term to an AR model. Therefore, a recurrent SVR

may be interpreted as a parsimonious model which incorporates all the past inputs without storing all of them in memory. That is, in our device, $\mathbf{R}_t$ can be set to just one-delayed error term $u_{t-1}, u_{t-1} = z^{-1}[y_t - \mathbf{O}_t]$, so as to avoid the difficulty in determining the lag length of recurrent input. Very small number of lag p in $\mathbf{x}_t$ is also appropriate for this recurrent SVR, for instance, $p = 2$ in our application. Thus, the specification of recurrent SVR based non-linear ARMA model used in this study is just simply ARMA (2, 1) model. It is the richer dynamic structure and specification convenience that make the recurrent SVR attractive in dynamic applications.

Now, according to Eqs. (3)–(6), we can rewrite the primal problem of recurrent ε-SVR for non-linear ARMA (2,1) model as follows:

$$\min_{\mathbf{w},b,\xi^{(\prime)}} \quad \mathbf{C}\left(\mathbf{w}, b, \xi_t^{(\prime)}\right) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{t=1}^{T}\left(\xi_t + \xi_t'\right) \tag{20}$$

$$s.t. \quad \mathbf{w}^T\phi\left(y_{t-1}, y_{t-2}, u_{t-1}\right) + b - y_t \le \varepsilon + \xi_t \tag{21}$$

$$y_t - \mathbf{w}^T\phi\left(y_{t-1}, y_{t-2}, u_{t-1}\right) - b \le \varepsilon + \xi_t' \tag{22}$$

$$\xi_t \ge 0, \xi_t' \ge 0 \quad t = 1, 2, \ldots, T \tag{23}$$

Also, the convex quadratic programming and linear restrictions ensure that the recurrent ε-SVR can always obtain the global unique optimal solution $\mathbf{w}^*$. By using the Karush–Kuhn–Tucker conditions, we can construct its dual problem, obtain the corresponding solution, α_t and α_t', and compute $\mathbf{w}^*$ and b^*. Because the inner-product kernel is a Gaussian kernel, the regression decision function $f(\mathbf{x})$ of recurrent ε-SVR is formulated as

$$\begin{aligned} f(\mathbf{x}) &= f\left(y_{s-1}, y_{s-2}, u_{s-1}\right) = \mathbf{w}^{*T}\phi\left(y_{s-1}, y_{s-2}, u_{s-1}\right) + b^* \\ &= \sum_{t=1}^{T}\left(\alpha_t' - \alpha_t\right)\exp\left(-\frac{1}{2\sigma^2}\left|\left|\left(y_{s-1}, y_{s-2}, u_{s-1}\right) - \left(y_{t-1}, y_{t-2}, u_{t-1}\right)\right|\right|^2\right) + b^* \end{aligned} \tag{24}$$

where s is any time point within or outside of the training period. And the MA part, u_{s-1}, can be skipped over and only the AR part is used for forecasting during the test period. The real constant coefficient σ^2 is also chosen by the users. Using the estimated decision function (24), we can achieve the best generalisation capability in forecasting y on new inputs.

The difficulty of estimating the recurrent ε-SVR lies in the fact that the error term is unobservable. To overcome such difficulty, we employ the model residuals as estimates of the errors in an iterative way, which is similar to the way that the linear ARMA model is iteratively estimated by MLE (Box et al. 1994; Hamilton 1997). Likewise, we initially set the error term to be its expectation, 0. In the following, the empirical procedure of the recurrent ε-SVR executed during the training phase is described. As denoted above, the empirical procedure is illustrated for the case of the non-linear stochastic ARMA (2, 1) model, $y_t = g(y_{t-1}, y_{t-2}, e_{t-1}) + e_t$. The letter i indicates the iterative epoch and t denotes the period.

Step 1: *Set* $i = 1$ *and start with all residuals at zero:* $e_t^{(1)} = 0$.
Step 2: *Run a SVR procedure to get the decision function* $f^{(i)}$ *to the points* $\{x_t, y_t\}$ *with all inputs* $x_t = \left\{y_{t-1}, y_{t-2}, e_{t-1}^{(i)}\right\}$.
Step 3: *Compute the new residuals* $e_t^{(i+1)} = y_t - f^{(i)}$.
Step 4: *Terminate the computational process when the stopping criterion is satisfied; otherwise, set* $i = i + 1$ *and go back to Step 2.*

Note that the first iterative epoch is in fact a feed-forward SVR process and results in a AR (2) model and that the following epochs provide results of the ARMA (2,1) model, being estimated by the recurrent ε-SVR.

In general, the procedure cannot be shown to converge, and there are no well-defined criteria for stopping its operation. Rather, some reasonable criteria can be found, although with its own practical drawback, which may be used to terminate the computational process. To formulate such a criterion, it is logical to think in terms of the properties of the estimated residual series. After enough long iterative steps, the auto-correlation displayed behind the residuals during the first AR epoch should disappear, and the information in the residual behavior has been used out and the final residual series should be white noise. Accordingly, we may suggest a sensible convergence criterion for the recurrent ε-SVR procedure as follows:

> The recurrent ε-SVR procedure is considered to have converged when the corresponding residuals become white noise, or have no auto-correlation.

To quantify the measurement of white noise, we use the formal hypothesis test, Ljung–Box–Pierce Q-test to investigate a departure from randomness based on the ACF of the residuals. Under the null hypothesis of no auto-correlation in residuals, the Q-test statistic is asymptotically Chi square distributed. Concretely, we just need to check the actual p values of Q-test of lag 1. It's reasonable to think there is no higher order auto-correlation if no one-order auto-correlation is in the residuals. Only if the p values of Q-test are simultaneously higher than 0.1 for consecutive five epochs, should the iterative computational process be stopped. To overcome the drawback of this convergence criterion, we use cross validation to avoid the possibility of an over-fitting problem; see Sect. 4.1 for detailed information.

3 Empirical modelling and forecasting scheme

3.1 Empirical models and their specification

As denoted in Sect. 2.2, very few lag numbers of non-linear ARMA model are enough for recurrent ε-SVR and ANN approaches to capture the dynamic characteristics of data sets. Therefore, the basic forecasting framework in this study is the ARMA (2, 1) model. For the convenience of comparison, we make use of the ARMA model with the same lag orders for its linear form. The linear ARMA (2, 1) model estimated by MLE is described below:

$$y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + e_t + \theta_1 e_{t-1} \tag{25}$$

The empirical models for the recurrent ε-SVR and the recurrent ANN are specified as the non-linear ARMA (2, 1) process, expressed below:

$$y_t = g(y_{t-1}, y_{t-2}, e_{t-1}) + e_t. \tag{26}$$

Then, the feed-forward ε-SVR corresponds to the nonlinear AR (2) model,

$$y_t = g\,(y_{t-1}, y_{t-2}) + e_t \tag{27}$$

In this paper, the non-linear function $g\,(\cdot)$ specified for recurrent ε-SVR is a radial basis function because only the Gaussian kernel is chosen for the SVR in this study. Of course, other functions such as polynomial, spline, hyperbolic tangent kernel also satisfy Mercer's conditions and can be adopted as the non-linear function of SVR. Before the implementation of the recurrent ε-SVR, their free parameters, ε (or denoted epsilon), C and Gaussian kernel width σ^2 (or sigma2) must be determined in advance through cross validation. The process of sensitivity analysis will be illustrated by using simulation in Sect. 4.1.

The benchmark recurrent ANN used in this study is the feedback multilayer perceptrons (MLP) network, denoted recurrent MLP. We specify this kind of recurrent back-propagation network with the following architecture: one non-linear hidden layer with four neurons, each using a tan-sigmoid differentiable transfer function to generate the output, and one linear output layer with one neuron. Thus, the non-linear function $g\,(\cdot)$ specified for recurrent MLP is a tan-sigmoid function. As a training algorithm, the fast training Levenberg–Marquardt algorithm is chosen. The value of the learning rate parameter used in the training process is set to be 0.05. These specifications and choices are standard in neural network literature.

In addition to the ARMA (2, 1) framework specified above, this paper will also compare the forecasting of recurrent SVR approach with another two benchmark models: one is the random walk model, the simplest time series model; another is a non-linear two-regime threshold ARMA model, TARMA(2; 2, 2; 0, 1), specified as below:

$$y_t = \begin{cases} \mu^{(1)} + \phi_1^{(1)} y_{t-1} + \phi_2{}^{(1)} y_{t-2} + e_t & if\ y_{t-2} \geq 0 \\ \mu^{(2)} + \phi_1^{(2)} y_{t-1} + \phi_2^{(2)} y_{t-2} + e_t + \theta_1 e_{t-1} & if\ y_{t-2} < 0 \end{cases} \tag{28}$$

in which if the market return is non-negative the model is specified as AR (2) framework; if the return is negative the model is specified as an ARMA (2, 1) model, the introduced moving average term also serving as a memory device because the financial market is normally asymmetric in which the bad news often has a more long-term influence on the financial return and its volatility.

3.2 Forecasting scheme

In this paper, a recursive forecasting scheme is employed with an updating sample window; the estimating and forecasting process is carried out recursively by updating

the sample with one observation each time, re-running the recurrent ε-SVR procedure and recalculating the model parameters and corresponding forecasts. The notations used in this study are as follows; The total number of series y_t is denoted as T and the number of observations used for the first in-sample estimation is T_1 (or called training sample). Then, $T - T_1$ observations are retained as a forecasting or test sample. Let the actual series at period $t + j$ and the j-step-ahead forecast of the series made at period t be written as y_{t+j} and $\widehat{y}_{t+j}$, respectively. Then, we can write

$$\widehat{y}_{t+j|t} = \widehat{E}\left(y_{t+j}|y_t, y_{t-1}, \ldots, y_1\right) \tag{29}$$

so that the j-step-ahead forecast of the series made at time t is the expected value of the series j periods in the future, given all information available at time t. In Eq. (29), $t = T_1, \ldots, T - j$. Thus, the forecast horizon is fixed at j steps ahead, and the starting point t is varied. Therefore, we can estimate and forecast the recurrent ε-SVR based ARMA (2, 1) model for $n = T - j - T_1 + 1$ times.

In this paper, only one-step-ahead forecasts are used for out-of-sample forecasting evaluation which indicates $j = 1$. We set $n = 100$ for linear ARMA simulation and $n = 400$ for non-linear Lorenz simulated series. The forecasting sample for the real data of CAD and NYSE returns is from 2 January 2014 to 30 May 2014. Thus, $n = 104$ is for CAD and $n = 103$ for NYSE returns.

3.3 Evaluation metrics and pairwise comparison of competing models

Two evaluation metrics are employed to compare the forecasting performance among the recurrent ε-SVR and the competing methods: normalized mean square error (NMSE) and correct sign predictions (sign) (Pesaran and Timmerman 1990; Moosa 2000). The NMSE measures the magnitude of the forecasting error and the sign measures correctness in predicted directions, i.e., the turning point correctness. Their formulas are

$$NMSE\,(\%) = 100 \times \frac{MSE}{Var\,(y)} = 100 \times \frac{\sum_{i=1}^{n}\left(y_i - \widehat{y}_i\right)^2/n}{\sum_{i=1}^{n}\left(y_i - \overline{y}_i\right)^2/(n-1)} \tag{30}$$

$$sign\,(\%) = \frac{100}{n}\sum_{i=1}^{n} a_i, \quad \text{where } a_i = \begin{cases} 1 & (y_{i+1} - y_i)\left(\widehat{y}_{i+1} - \widehat{y}_i\right) > 0 \\ 0 & otherwise \end{cases} \tag{31}$$

To test for equal forecasting accuracy of two competing models, we use the two-sided DM test statistic proposed by Diebold and Mariano (1995) for the difference of NMSE metric. The null hypothesis is H0: NMSE1–NMSE0 = 0, where the subscript 0 denotes the benchmark model and 1 the target model. The DM tests in this study are investigated in a robust form, by simply scaling the numerator by a heteroscedasticity and autocorrelation consistent (HAC) (co)variance matrix calculated according to Newey–West procedures (Newey and West 1987). We use Andrews (1991) approximation rule to automatically select the number of lags for HAC matrix. In the case of a large sample, the DM statistic converges in distribution to a standard normal.

4 Forecasting application with simulated and real data

4.1 Simulations

4.1.1 Data generating process

To evaluate the forecasting performance of recurrent ε-SVR approach, we first conduct the following simulation. The target variable $y_t, t = 1, \ldots, T$ is randomly generated from two models: (1) a linear ARMA (2, 1) model:

$$y_t - 0.9y_{t-1} + 0.3y_{t-2} = e_t - 0.7e_{t-1} \tag{32}$$

where the noise inputs, e_t, are generated from the standard normal distribution and the simulated y_t are discrete; (2) a non-linear Lorenz feedback system:

$$\begin{aligned} dy/dt &= 16\,(x - y) \\ dx/dt &= -yz + 45.92y - x \\ dz/dt &= yx - 4z \end{aligned} \tag{33}$$

where the step size is 0.01. The Student's t noise is included in the simulated continuous y_t series (see Lorenz 1963 for more). We include both linear and non-linear simulations to see how the recurrent ε-SVR procedure performs when a linear series is not really applicable. In the simulations, the sample size T is 1,000, and the number of replications is 200. The reported results are the mean values of 200 independent replications.

4.1.2 Parameters selection and iterative epochs of recurrent SVR

The use of cross-validation is appealing particularly when we have to design a somewhat complex approach with a good generalisation as the goal. For example, here, we may use cross-validation to determine the values of free parameters with the best performance, and when it is best to stop training, as described in the following. The first training data, that is, the former 900 observations for the linear ARMA series (briefly denoted LARMA) and 600 for non-linear Lorenz series, are exemplified. The training data are further randomly partitioned into two disjoint subsets: estimating sample and validating sample (700 and 200 observations for LARMA; 500 and 100 for Lorenz).

As shown in Sect. 2, two free parameters (ε and C) and kernel width σ^2 have to be determined by us before running the recurrent ε-SVR procedure. The motivation of using cross validation here is to validate the model on a data set different from the one used for parameter estimation. In this way we may use the training set to assess the performance of various values of parameters, and thereby choose the best one. The sensitivity analysis of recurrent ε-SVR (represented by the generalisation error NMSE) with respect to three parameters are illustrated in Fig. 4.

Figure 4a–c describes the sensitivity analysis for one of 200 simulated linear ARMA series. Parameter ε varies between the range [0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.3, 0.5, 0.7, 0.9, 1.0] with C being fixed at 0.1 and σ^2 at 1. The values of ε before the

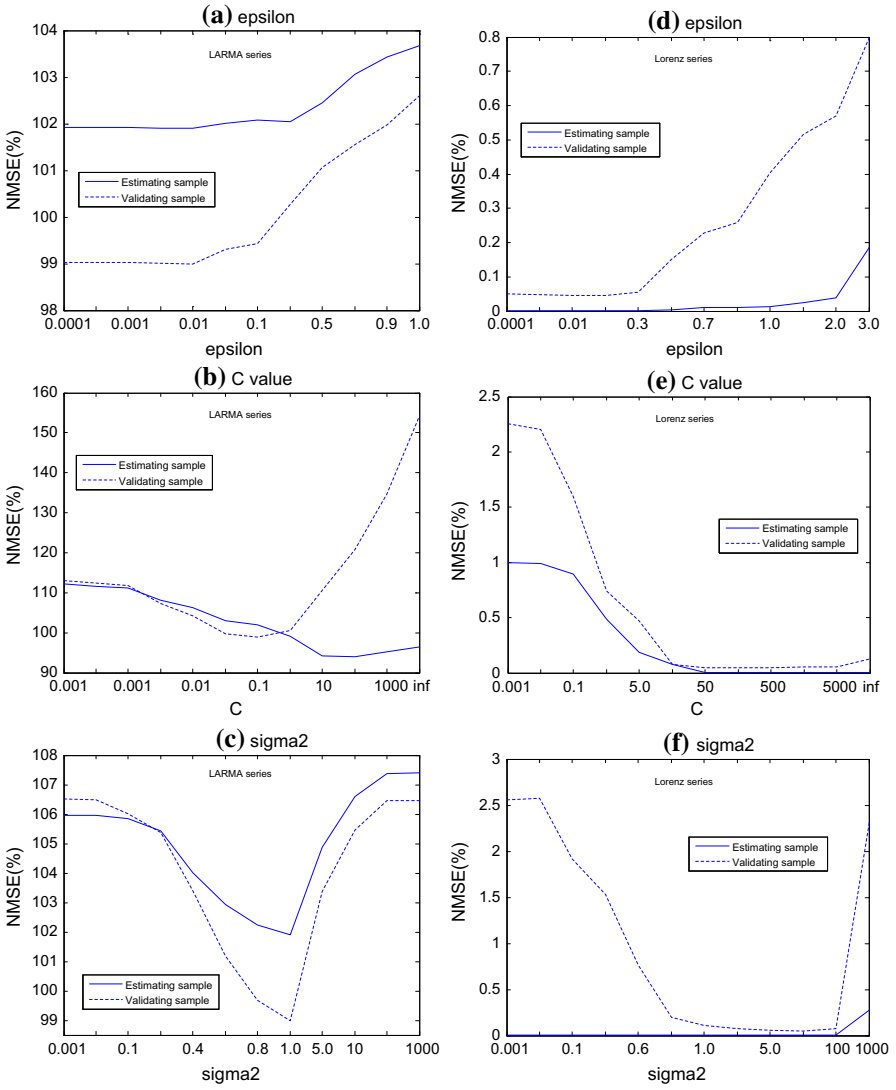

Fig. 4 Sensitivity analysis of the recurrent ε-SVR for simulation data

point of ε at 0.01 have no influence on the performance of our recurrent SVR, which is considerably stable. Parameter C varies from very small value 0.0001 to infinity with ε being fixed at 0.01 and σ^2 at 1. Clearly, when $C = 0.1$, NMSE of validation sample obtains the lowest value, 99 %; after that, over-fitting the training set occurs. Coefficient σ^2 varies between values of 0.01 and 0.1 with C being fixed at 0.1 and ε at 0.01. Both values of NMSE attain the minima when $\sigma^2 = 1.0$. Thus, the appropriate parameters of recurrent SVR for linear ARMA series are: $\varepsilon = 0.01, C = 0.1$ and $\sigma^2 = 1.0$. Figure 4d–f describes the parameter selection process for the non-linear Lorenz series. Similar to LARMA, the performance of recurrent SVR is very stable

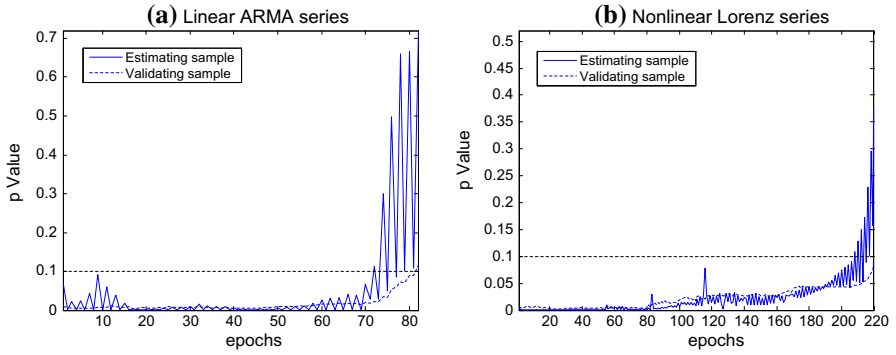

Fig. 5 Iterative epochs of recurrent ε-SVR for simulation data

and not influenced by any value of ε before the point $\varepsilon = 0.3$. And when $C = 50$ and $\sigma^2 = 10$, the values of NMSE for validation subsets all reach to their minima, 0.046 %. Therefore, the correct parameters chosen for Lorenz series are $\varepsilon = 0.1$, $C = 50$ and $\sigma^2 = 10$, respectively.

With good forecasting performance as the goal, it is very difficult to figure out when it is best to stop training only in terms of fitting performance. It is possible for the procedure to end up over-fitting the training data if the training session is not stopped at the right point. We can identify the onset of over-fitting and the stopping point through the use of cross-validation. Figure 5a, b describes the iterative epochs for one of 200 linear ARMA and non-linear Lorenz series, respectively. For the former series, the iterative process of recurrent ε-SVR is stopped at the 82th epoch; while, for the latter series, the iterative process is longer and stopped after 220 iterative steps, maybe due to the non-linearity and noise of the series. Now, we can say, at about the 10 percent significance level, the final residuals obtained from the recurrent SVR procedure have no autocorrelation. In addition, the p value curves of both estimating and validating samples exhibit a similar pattern (increase for an increasing number of epochs) and point to the almost same stopping point. That is to say, there is no over-fitting phenomenon for the examples illustrated here, the recurrent ε-SVR model does as well on the validating subset as it does on the estimating subset, on which its design is based.

4.1.3 Comparing forecasting performance

There is still the possibility of over-fitting after training. Therefore, the generalisation performance of the competed models is further measured and evaluated on the test set, which is different from the validation subset. For the simulated data, the forecasting sample is the latter 100 observations for the LARMA series and latter 400 for Lorenz series. Thus, the recurrent ε-SVR and the benchmark models should be recursively trained and forecast 100 and 400 times for respective series to obtain the corresponding one-step-ahead forecasts for evaluation.

The average NMSE and the proportion of correct sign predictions of 200 replicable simulations for each method are reported in Table 1, in which, a smaller NMSE and

Table 1 Measures of forecasting performance for simulation data

Series	Metrics	Random walk	Threshold ARMA	MLEARMA	Recurrent ANN ARMA	Feedforward SVR ARMA	Recurrent SVR ARMA
LARMA	NMSE	101.88	102.42	101.23	101.05	100.97	100.96
	sign	34.34	40.40	42.31	40.37	40.16	41.76
Lorenz	NMSE	0.00853	0.00265	0.00624	0.00082	0.00092	0.00074
	sign	96.24	98.25	98.77	96.41	98.99	99.48

Table 2 Diebold–Mariano test for the NMSE difference on simulation data

Models	LARMA						Lorenz					
	DM1	DM2	DM3	DM4	DM5	DM6	DM1	DM2	DM3	DM4	DM5	DM6
Random walk		0.428	0.661	0.524	0.991	0.958		0.894	0.604	1.000	1.000	1.000
Threshold ARMA	0.572		0.816	0.902	0.993	0.984	0.106		0.206	0.998	0.997	0.999
MLE ARMA	0.339	0.184		0.753	0.953	0.928	0.396	0.794		1.000	1.000	1.000
Recurrent ANN ARMA	0.476	0.098	0.247		0.873	0.846	0.000	0.002	0.000		0.359	0.904
Feedforward SVR ARMA	0.009	0.007	0.047	0.127		0.584	0.000	0.003	0.000	0.641		0.916
Recurrent SVR ARMA	0.042	0.016	0.072	0.154	0.416		0.000	0.001	0.000	0.096	0.084	

DM1–DM6 are the robust Diebold and Mariano (1995) test by using Newey–West procedures (Newey and West 1987) when the benchark models are the random walk, threshold ARMA, MLE ARMA, recurrent ANN ARMA, feedforward SVR ARMA and recurrent SVR ARMA models, respectively. For each test we consider the NMSE metrics

a higher sign value indicate the better forecasting performance. As shown in Table 1, the recurrent SVR ARMA model almost outperforms the benchmarks in one-step-ahead forecasting, except for the sign metric of MLE model. The overall superiority of the recurrent SVR over the feed-forward one reveals that the proposed recurrent ε-SVR in this study really improves the forecasting performance of the standard SVR because the recurrent networks have a higher ability to capture the dynamic feature of series than does the feed-forward one. The fact that the recurrent SVR behaves better than the recurrent ANN confirms that the structural risk minimizing principle endows SVR with stronger forecasting ability as opposed to ANN model. The sign value of MLE model for LARMA series is 42.91 %, higher than those of recurrent ANN, feed-forward and recurrent SVR (40.37, 40.16 and 41.76 %), indicating that the non-linear ARMA model is not suitable to the data with linearity with respect to non-linear Lorenz series. The evidence that the values of NMSE for Lorenz series are far lower than those for LARMA may have resulted from the continuous nature for the former and discrete the latter.

Table 2 presents the p values of Diebold–Mariano (DM) test for the NMSE difference, which are defined as the significance levels at which the null hypothesis under investigation can be rejected. We report the results of the DM test, say DM1, in the second and eighth column for two simulated series, respectively, under the null hypothesis that the NMSE metric produced by the random walk model equals to those obtained by the use of the other models. Concretely, a p value no greater than 0.05 indicates that the random walk model yields a higher forecasting error (in terms of NMSE) relative to the competing model at a 5 % significance level, a p value no smaller than 0.95

means that random walk produces a lower forecasting error at 5 % level, while a p value between 0.05 and 0.95 implies that the benchmark and competing models have the equivalent forecasting accuracy from the viewpoint of statistics. DM2–DM6 are organized in the same manner and illustrative of the test results when the benchmark models are respectively, the threshold ARMA, MLE based ARMA, recurrent ANN ARMA, feed-forward and recurrent SVR models. The same interpretation applies to the p values reported for DM2–DM6.

For LARMA series, DM1 tests reveal that both feed-forward and recurrent SVR ARMA models have a statistically stronger forecasting power than random walk at least at 5 % significance level, while the forecasting ability between random walk and threshold ARMA, MLE based ARMA and recurrent ANN ARMA models is similar. According to DM2 statistic, the non-linear threshold ARMA model and linear MLE based ARMA model have a similar forecasting ability, while the forecasting power of threshold ARMA model is statistically lower than the non-linear ARMA model estimated by recurrent ANN, feed-forward and recurrent SVR at least at 10 % significance level. DM3 statistic tells us that the linear MLE based ARMA model is statistically inferior to both feed-forward and recurrent SVR models but equivalent to the recurrent ANN model in forecasting the LARMA series. The only exception is suggested by the statistic of DM4 and DM5; that is, the recurrent ANN model is not inferior to feed-forward and recurrent SVR model and the recurrent SVR model is also not better than feed-forward SVR model. For Lorenz series, the DM tests only reveal three cases of similar forecasting performance, random walk and MLE-based ARMA model, threshold ARMA and MLE ARMA model, recurrent ANN and feed-forward SVR ARMA model. Other cases all show the statistical forecasting difference between the benchmark and target models. Especially, the recurrent SVR ARMA model consistently and statistically outperforms all other benchmark models at least at a 10 % significance level. Obviously, the forecasting performance measured by NMSE metrics in Table 1 is mostly supported by the DM tests reported in Table 2.

4.2 Real data analysis

In this sub-section, we investigate the forecasting performance of all candidates by using real data for two kinds of financial variables: CAD/USD exchange rates and NYSE average index.

4.2.1 Data description

The first data set consists of the daily nominal bilateral exchange rates of the Canadian Dollar (CAD) against the US dollar for the period between 2 January 2004 and 30 May 2014. The data are obtained from a database of Policy Analysis Computing and Information Facility in Commerce (PACIFIC) at the University of British Columbia. The second data set consists of daily closing price of New York Stock Exchange™ (NYSE) composite stock index for the period of 2 January 2004 to 30 May 2014. The data are downloaded directly from the Market Information section of the NYSE™ web page.

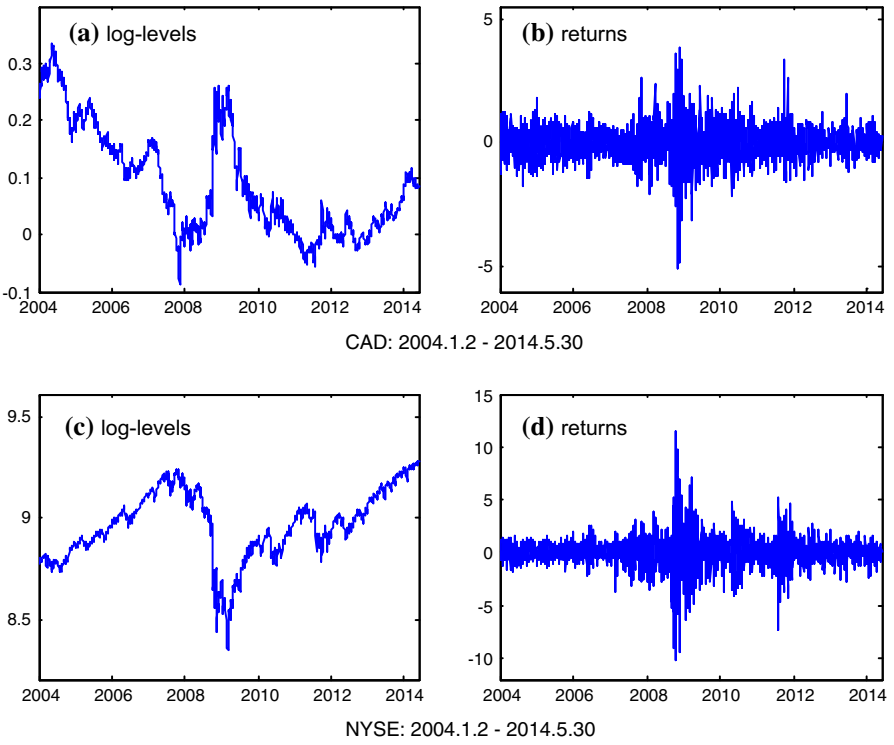

Fig. 6 Log levels and returns of CAD exchange rates and NYSE stock index

It has been widely accepted that a variety of financial variables including foreign exchange rates and stock prices are integrated at an order of one. To avoid the issue of possible non-stationarity, this paper considers the financial returns, y_t, which are converted from corresponding levels (price or index), I_t, by using continuous compounding transforms as

$$y_t = 100 \times (\log I_{t+1} - \log I_t) \tag{34}$$

Both data are transformed into daily returns via Eq. (33), providing a return series of 2,611 observations for CAD and 2,619 observations for NYSE. For CAD returns, the recursive training is used with updating window data starting from the former 2,507 observations through the former 2,610 observations; the 104 one-day-ahead forecasts of returns will be obtained. For the NYSE, the recursive training is from the former 2,516 observations through the former 2,618 observations; the 103 one-day-ahead forecasts of returns are achieved. Both the forecasting returns correspond to the period of 2 January 2014 to 30 May 2014. The daily series for the log-levels of price and the returns of the CAD and NYSE are depicted in Fig. 6. The figure shows that the price indices are obviously non-stationary, and the return series are mean-stationary, and exhibit the typical volatility clustering phenomenon with periods of unusually large volatility followed by periods of relative tranquility.

Table 3 Measures of forecasting performance for real data

Models	Metrics	Random walk	Threshold ARMA	MLE ARMA	Recurrent ANN ARMA	Feedforward SVR ARMA	Recurrent SYR ARMA
CAD	NMSE	99.34	99.47	99.42	101.11	96.89	90.75
	sign	46.15	50.00	52.88	54.81	53.85	55.77
NYSE	NMSE	99.68	99.46	99.20	98.40	98.62	91.68
	sign	52.43	56.31	55.34	54.37	57.28	60.19

Table 4 Diebold–Mariano test for the NMSE difference on real data

Models	CAD						NYSE					
	DM1	DM2	DM3	DM4	DM5	DM6	DM1	DM2	DM3	DM4	DM5	DM6
Random walk		0.481	0.337	0.094	0.973	1.000		0.576	0.591	0.913	0.904	0.998
Threshold ARMA	0.519		0.594	0.459	0.987	1.000	0.424		0.563	0.938	0.909	0.984
MLE ARMA	0.663	0.406		0.076	0.999	1.000	0.409	0.437		0.531	0.525	0.999
Recurrent ANN ARMA	0.906	0.541	0.924		1.000	1.000	0.087	0.062	0.469		0.484	0.978
Feedforward SVR ARMA	0.027	0.013	0.001	0.000		0.997	0.096	0.091	0.475	0.516		1.000
Recurrent SVR ARMA	0.000	0.000	0.000	0.000	0.003		0.002	0.016	0.001	0.022	0.000	

DM1–DM6 are the robust Diebold and Mariano (1995) test by using Newey–West procedures (Newey and West 1987) when the benchark models are the random walk, threshold ARMA, MLE ARMA, recurrent ANN ARMA, feedforward SVR ARMA and recurrent SVR ARMA models, respectively. For each test we consider the NMSE metrics

4.2.2 Comparing forecasting performance

The implementation of parameter selection and iterative process of recurrent ε-SVR for real data are the same as the simulations and are skipped over here to save space. Based on such kind of sensitivity analysis, the appropriate parameters are $\varepsilon = 0.005, \mathrm{C} = 0.001$ and $\sigma^2 = 1$ for CAD returns and $\varepsilon = 0.3, \mathrm{C} = 0.01$ and $\sigma^2 = 0.2$ for NYSE returns.

The results of out-of-sample return forecasting accuracy based on two quantitative metrics (NMSE and sign) are presented in Table 3. Table 4 reports the p values of Diebold–Mariano (DM) test for the difference of NMSE metric in a robust HAC form from Newey–West procedures. The NMSE metrics reveal that the recurrent SVR ARMA model has the strongest forecasting ability as opposed to other five benchmark models for both CAD and NYSE returns (with least NMSE value of 90.75 and 91.68), which is statistically confirmed by DM tests reported in Table 4 at least at 5% significance level. Except for recurrent SVR model, the feed-forward SVR statistically outperforms all other models in forecasting one-day-ahead financial returns of CAD and the NYSE, with two exceptions that it has similar forecasting performance to MLE and recurrent ANN ARMA model when forecasting NYSE returns. All other DM statistics show that four models of random walk, threshold ARMA, MLE ARMA and recurrent ANN model have similar forecasting performance based on NMSE metric when compared with each other, except for the random walk and

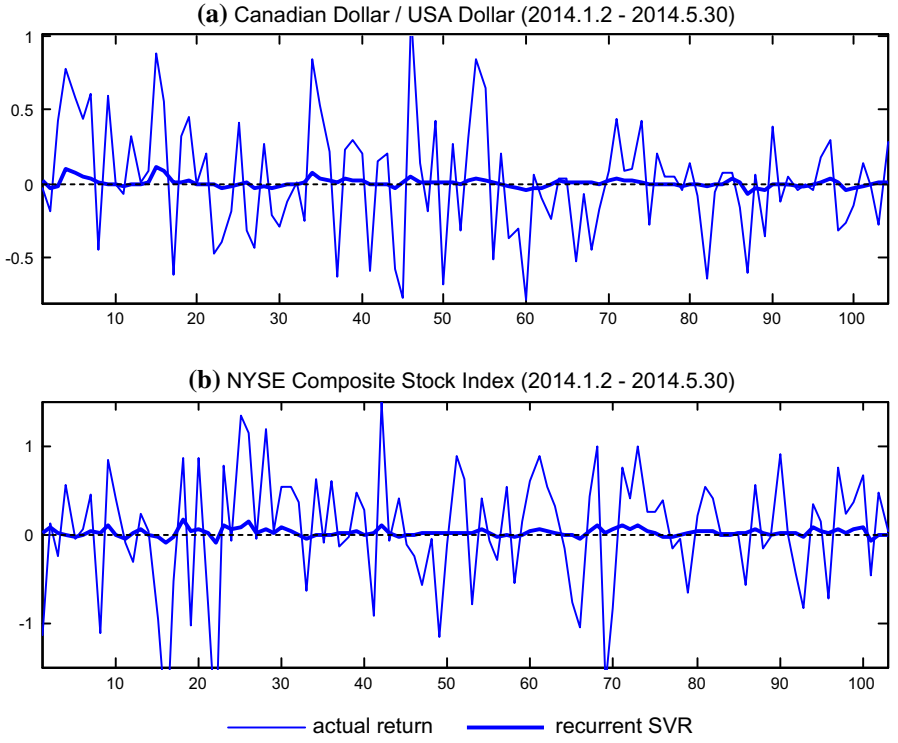

Fig. 7 Actual and forecasted financial returns

recurrent ANN, threshold and recurrent ANN model in forecasting NYSE returns. The sign metrics reported in Table 3 show that recurrent SVR ARMA model also behaves best in forecasting turning points of both CAD and NYSE returns, 55.77 and 60.19 %, respectively. The empirical evidence of real data also confirms the conclusion obtained in the Monte Carlo Simulation and does favour the theoretical the advantage of recurrent SVR model.

We plot the actual and one-day-ahead forecasting returns by the recurrent ε-SVR in Fig. 7. The 104 one-day-ahead forecasting returns correspond to the out-of-sample period between 2 January 2014 and 30 May 2014 for the CAD and the 103 one-day-ahead forecasting returns correspond to the same out-of-sample period for the NYSE. As anticipated, the recurrent ε-SVR very accurately captures the actual returns.

5 Conclusions

In this paper we propose a recurrent ε-SVR procedure for non-linear ARMA models which has a global feedback loop from the output layer to the input space and examine the empirical forecasting performance of the proposed procedure. Empirical applications are made for forecasting the simulated data and the real data of the Canadian Dollar (CAD) against US Dollar daily exchange rates and the New York Stock

Exchange^{TM} (NYSE) composite stock index. The forecasting ability of the recurrent ε-SVR is also compared with those of random walk, threshold ARMA, MLE-based ARMA, the recurrent ANN ARMA and the feed-forward SVR ARMA with regard to two quantitative evaluation metrics and robust Diebold–Mariano tests following Newey–West procedure.

The NMSE and sign evidence from both the simulated and real data analysis obviously shows that the proposed recurrent ε-SVR statistically improves the forecasting performance of the standard feed-forward one. And it also consistently outperforms the benchmark models in forecasting the return magnitude and the turning points, just with two exceptions revealed by DM4 and DM5 when forecasting the linear ARMA simulation series. Empirical analysis is in favour of the theoretical advantage of the recurrent SVR. The sensitivity to free parameters of the recurrent ε-SVR results and its iterative process are also examined in detail by using the cross-validation method, which can be implemented very easily. In conclusion, the proposed recurrent ε-SVR method can be used as another standard SVR construction procedure in other applications.

Acknowledgments The authors thank the editor, Stefan Trueck, and three anonymous referees for their constructive comments. Kiho Jeong's research was supported by Kyungpook National University Research Fund, 2011. Shiyi Chen appreciates the supports from Shanghai Leading Talent Project, Fudan Zhuo-Shi Talent Plan and Fudan 985 Project. The work was also sponsored by Deutsche Forschungsgemeinschaft through SFB 649 "Economic Risk".

References

Adya M, Collopy F (1998) How effective are neural networks at forecasting and prediction? A review and evaluation. J Forecast 17:481–495

Andrews DWK (1991) Heteroskedasticity and autocorrelation consistent covariance matrix estimation. Econometrica 59:817–858

Ashok KN, Mitra A (2002) Forecasting daily foreign exchange rates using genetically optimized neural networks. J Forecast 21:501–511

Box GEP, Jenkins GM, Reinsel GC (1994) Time series analysis: forecasting and control. Prentice Hall, Englewood Cliffs

Cao L, Tay F (2001) Financial forecasting using support vector machines. Neural Comput Appl 10:184–192

Database of Exchange Rates: http://pacific.commerce.ubc.ca/xr Policy Analysis Computing and Information Facility in Commerce (PACIFIC) at University of British Columbia

Database of NYSE stock index: http://www.nyse.com/marketinfo/datalib/ the Market Information section of the NYSE^{TM} web page

Deng NY, Tian YJ (2004) New methods in data mining: support vector machine. Science Press, Beijing

Diebold FX, Mariano RS (1995) Comparing predictive accuracy. J Bus Econ Stat 13:253–265

Espinoza M, Suykens JAK, De Moor B (2006) LS-SVM regression with autocorrelated errors. In: Proceedings of the 14th IFAC symposium on system identification (SYSID), Newcastle, Australia, pp 582–587

Evgeniou T, Poggio T, Pontil M, Verri A (2002) Regularization and statistical learning theory for data analysis. Comput Stat Data Anal 38(4):421–432

Gaudart J, Giusiano B, Huiart L (2004) Comparison of the performance of multi-layer perceptron and linear regression for epidemiological data. Comput Stat Data Anal 44(4):547–570

Hamilton JD (1997) Time series analysis. princeton University Press, Princeton

Härdle WK, Moro RA, Schäfer D (2005) Predicting bankruptcy with support vector machines. In: Statistical tools for finance and insurance. Springer, Berlin

Härdle WK, Moro RA, Schäfer D (2006) Graphical data representation in bankruptcy analysis. In: Handbook for data visualization. Springer, Berlin

Haykin S (1999) Neural networks: a comprehensive foundations. Prentice Hall, New Jersey
Hong Wei-Chiang (2011) Electric load forecasting by seasonal recurrent SVR (support vector regression) with chaotic artificial bee colony algorithm. Energy 36(9):5568–5578
Jordan MI (1987) Attractor dynamics and parallelism in a connectionist sequential machine. In: Proceeding of 8th annual conference of the cognitive science society, Hillsdale, pp 531–546
Kamruzzaman J, Sarker R (2004) ANN-based forecasting of foreign currency exchange rate. Neural Inf Process Lett Rev 3(2):49–58
Kanas A (2003) Non-linear forecasts of stock returns. J Forecast 22(4):299–315
Kuan C-M (1995) A recurrent Newton algorithm and its convergence properties. IEEE Trans Neural Netw 6:779–783
Kuan C-M, Hornik K, White H (1994) A convergence result for learning in recurrent neural networks. Neural Comput 6:420–440
Kuan C-M, Liu T (1995) Forecasting exchange rates using feedforward and recurrent neural networks. J Appl Economet 10:347–364
Lee TS, Chiu CC, Chou YC, Lu CJ (2006) Mining the customer credit using classification and regression tree and multivariate adaptive regression splines. Comput Stat Data Anal 50(4):1113–1130
Lisi F, Schiavo RA (1999) A comparison between neural networks and chaotic models for exchange rate prediction. Comput Stat Data Anal 30(1):87–102
Lorenz EN (1963) Deterministic non-periodic flow. J Atmos Sci 20:130–141
Moosa I (2000) Exchange rate forecasting: techniques and applications. Macmillan Press LTD, Lonton
Newey WK, West KD (1987) A simple positive, semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. Econometrica 55(3):703–708
Niemira MP, Klein PA (1994) Forecasting financial and economic cycles. Wiley, New York
Pesaran M, Timmerman A (1990) The statistical and economic significance of the predictability of excess returns on common stocks. Department of Applied Economics, University of Cambridge, working paper no. 9022
Priestley MB (1988) Nonlinear and non-stationary time series analysis. Academic Press, London
Scholkopf B, Smola A (2001) Learning with kernels. MIT Press, Cambridge
Suykens JAK, Vandewalle J (2000) Recurrent least squares support vector machines. IEEE Trans Circuits Syst I 47(7):1109–1114
Suykens JAK, Van Gestel T, De Brabanter J, De Moor B, Vandewalle J (2002) Least squares support vector machines. World Scientific, Singapore
Tian J, Juhola M, Grönfors T (1997) AR parameter estimation by a feedback neural network. Comput Stat Data Anal 25(1):17–24
Trafalis T, Ince H (2000) Support vector machine for regression and applications to financial forecasting. In: International joint conference on neural networks, pp 348–353
Van Gestel T, Suykens J, Baestaens D, Lambrechts A, Lanckriet G, Vandaele B, De Moor B, Vandewalle J (2001) Financial time series prediction using least squares support vector machines within the evidence framework. IEEE Trans Neural Netw 12(4):809–821
Vapnik VN (1995) The nature of statistical learning theory. Springer, New York
Vapnik VN (1997) Statistical learning theory. Wiley, New York
Wu B (1995) Model-free forecasting for nonlinear time series (with application to exchange rates). Comput Stat Data Anal 19(4):433–459
Yang H, Chan L, King I (2002) Support vector machine regression for volatile stock market prediction. In: Proceedings of the third international conference on intelligent data engineering and automated learning, pp 391–396