

A Novel Weighted Ensemble Technique for Time Series Forecasting

Ratnadip Adhikari and R.K. Agrawal

School of Computer and Systems Sciences,
Jawaharlal Nehru University, New Delhi-110067, India
{adhikari.ratan,rkajnu}@gmail.com

Abstract. Improvement of time series forecasting accuracy is an active research area having significant importance in many practical domains. Extensive works in literature suggest that substantial enhancement in accuracies can be achieved by combining forecasts from different models. However, forecasts combination is a difficult as well as a challenging task due to various reasons and often simple linear methods are used for this purpose. In this paper, we propose a nonlinear weighted ensemble mechanism for combining forecasts from multiple time series models. The proposed method considers the individual forecasts as well as the correlations in pairs of forecasts for creating the ensemble. A successive validation approach is formulated to determine the appropriate combination weights. Three popular models are used to build up the ensemble which is then empirically tested on three real-world time series. Obtained forecasting results, measured through three well-known error statistics demonstrate that the proposed ensemble method provides significantly better accuracies than each individual model.

Keywords: Time Series Forecasting, Ensemble Technique, Box-Jenkins Models, Artificial Neural Networks, Elman Networks.

1 Introduction

Time series forecasting has indispensable importance in many practical data mining applications. It is an ongoing dynamic area of research and over the years various forecasting models have been developed in literature [1,2]. A major concern in this regard is to improve the prediction accuracy of a model without sacrificing its flexibility, robustness, simplicity and efficiency. However, this is not at all an easy task and so far no single model alone can provide best forecasting results for all kinds of time series data [3,4].

Combining forecasts from conceptually different methods is a very effective way to improve the overall forecasting precisions. The earliest use of this practice started in 1969 with the monumental work of Bates and Granger [5]. Till then, numerous forecasts combination methods have been developed in literature [6,7,8]. The precious role of model combination in time series forecasting can be credited to the following facts: (a) by an adequate ensemble technique, the

forecasting strengths of the participated models aggregate and their weaknesses diminish, thus enhancing the overall forecasting accuracy to a great extent, (b) often, there is a large uncertainty about the optimal forecasting model and in such situations combination strategies are the most appropriate alternatives to use, and (c) combining multiple forecasts can efficiently reduce errors arising from faulty assumptions, bias, or mistakes in the data [3].

The simple average is the most widely used forecasts combining technique. It is easy to understand, implement and interpret. However, this method is often criticized because it does not utilize the relative performances of the contributing models and is quite sensitive to the extreme errors [1,3]. As a result, other forms of averaging, e.g. trimmed mean, Winsorized mean, median, etc. have been studied in literature [9]. Another common method is the weighted linear combination of individual forecasts in which the weights are determined from the past forecast errors of the contributing models. But, this method completely ignores the possible relationships between two or more participating models and hence is not so adequate for combining nonstationary and chaotic data. Various modifications of this linear combination technique have also been suggested by researchers [9,10,11].

In this paper, we propose a weighted nonlinear framework for combining multiple time series models. Our approach is partially motivated by the work of Freitas and Rodrigues [12]. The proposed technique considers individual forecasts from different methods as well as the correlations between pairs of forecasts for combining. We consider three models, viz. Autoregressive Integrated Moving Average (ARIMA), Artificial Neural Network (ANN) and Elman ANN to build up the ensemble. An efficient methodology, based on a successive validation approach is formulated for finding the appropriate combination weights. The effectiveness of the proposed technique is tested on three real-world time series (one stationary and two nonstationary financial data). The forecasting accuracies are evaluated in terms of the error measures: Mean Absolute Error (MAE), Mean Squared Error (MSE), and Average Relative Variance (ARV).

The rest of the paper is organized as follows. Section 2 describes a number of common forecasts combination techniques. Our proposed ensemble scheme is presented in Sect. 3. In Sect. 4, we describe the three time series forecasting models, which are used here to build up the ensemble. Experimental results are reported in Sect. 5 and finally Sect. 6 concludes this paper.

2 Forecasts Combination Methods

Let the actual time series be $\mathbf{Y} = \{y_1, y_2, \dots, y_N\}$ and $\hat{\mathbf{Y}}^{(i)} = \{\hat{y}_1^{(i)}, \hat{y}_2^{(i)}, \dots, \hat{y}_N^{(i)}\}$ be its forecast obtained from the i^{th} method ($i = 1, 2, \dots, N$). Then the series obtained from linearly combining these n forecasted series is given by:

$$\begin{cases} \hat{\mathbf{Y}}^{(c)} = \{\hat{y}_1^{(c)}, \hat{y}_2^{(c)}, \dots, \hat{y}_N^{(c)}\}, \\ \hat{y}_k^{(c)} = w_1 \hat{y}_k^{(1)} + w_2 \hat{y}_k^{(2)} + \dots + w_n \hat{y}_k^{(n)} = \sum_{i=1}^n w_i \hat{y}_k^{(i)} \\ \forall k = 1, 2, \dots, N \end{cases} \quad (1)$$

where, w_i is the weight assigned to the i^{th} forecasting method. To ensure unbiasedness, sometimes it is assumed that the weights add up to unity. Different combination techniques are developed in literature which are based on different weight assignment schemes; some important among them are discussed here:

- In the *simple average*, all models are assigned equal weights, i.e. $w_i = 1/n$ ($i = 1, 2, \dots, n$) [9,10].
- In the *trimmed average*, individual forecasts are combined by a simple arithmetic mean, excluding the worst performing $k\%$ of the models. Usually, the value of k is selected from the range of 10 to 30. This method is sensible only when $n \geq 3$ [9,10].
- In the *Winsorized average*, the i smallest and largest forecasts are selected and set to the $(i + 1)^{\text{th}}$ smallest and largest forecasts, respectively [9].
- In the *error-based* combining, an individual weight is chosen to be inversely proportional to the past forecast error of the corresponding model [3].
- In the *outperformance* method, the weight assignments are based on the number of times a method performed best in the past [11].
- In the *variance-based* method, the optimal weights are determined by minimizing the total Sum of Squared Error (SSE) [7,10].

All the combination techniques, discussed above are linear in nature. The literature on nonlinear forecast combination methods is very limited and further research works are required in this area [10].

3 The Proposed Ensemble Technique

Our ensemble technique is an extension of the usual linear combination method in order to deal with the possible correlations between pairs of forecasts and is partially motivated from the work of Freitas and Rodrigues [12].

3.1 Mathematical Description

For simplicity, here we describe our ensemble technique for combining forecasts from three methods; but, it can be easily generalized. Let, the actual test dataset of a time series be $\mathbf{Y} = [y_1, y_2, \dots, y_N]^T$ with $\hat{\mathbf{Y}}^{(i)} = [\hat{y}_1^{(i)}, \hat{y}_2^{(i)}, \dots, \hat{y}_N^{(i)}]^T$ being its forecast obtained from the i^{th} method ($i = 1, 2, 3$). Let, $\mu^{(i)}$ and $\sigma^{(i)}$ be the mean and standard deviation of $\hat{\mathbf{Y}}^{(i)}$ respectively. Then the combined forecast of \mathbf{Y} is defined as: $\hat{\mathbf{Y}}^{(c)} = [\hat{y}_1^{(c)}, \hat{y}_2^{(c)}, \dots, \hat{y}_N^{(c)}]^T$, where

$$\begin{aligned} \hat{y}_k^{(c)} = & w_0 + w_1 \hat{y}_k^{(1)} + w_2 \hat{y}_k^{(2)} + w_3 \hat{y}_k^{(3)} \\ & + \theta_1 v_k^{(1)} v_k^{(2)} + \theta_2 v_k^{(2)} v_k^{(3)} + \theta_3 v_k^{(3)} v_k^{(1)} \end{aligned} \quad (2)$$

$$v_k^{(i)} = \left(y_k^{(i)} - \mu^{(i)} \right) / \left(\sigma^{(i)} \right)^2, \quad \forall i = 1, 2, 3; k = 1, 2, \dots, N.$$

In (2), the nonlinear terms are included in calculating $\hat{y}_k^{(c)}$ to take into account the correlation effects between two forecasts. It should be noted that for combining n methods, there will be $\binom{n}{2}$ nonlinear terms in (2).

3.2 Optimization of the Combination Weights

The combined forecast defined in (2) can be written in vector form as follows:

$$\hat{\mathbf{Y}}_k^{(c)} = \mathbf{F}\mathbf{w} + \mathbf{G}\boldsymbol{\theta} \quad (3)$$

where,

$$\mathbf{w} = [w_0, w_1, w_2, w_3]^T, \boldsymbol{\theta} = [\theta_1, \theta_2, \theta_3]^T.$$

$$\mathbf{F} = \left[\mathbf{1} | \hat{\mathbf{Y}}^{(1)} | \hat{\mathbf{Y}}^{(2)} | \hat{\mathbf{Y}}^{(3)} \right]_{N \times 4}.$$

$$\mathbf{1} = [1, 1, \dots, 1]^T.$$

$$\mathbf{G} = \begin{bmatrix} v_1^{(1)} v_1^{(2)} & v_1^{(2)} v_1^{(3)} & v_1^{(3)} v_1^{(1)} \\ \vdots & \vdots & \vdots \\ v_N^{(1)} v_N^{(2)} & v_N^{(2)} v_N^{(3)} & v_N^{(3)} v_N^{(1)} \end{bmatrix}_{N \times 3}.$$

The weights are to be optimized by minimizing the forecast SSE, given by:

$$\begin{aligned} \text{SSE} &= \sum_{k=1}^N \left(y_k - \hat{y}_k^{(c)} \right)^2 \\ &= (\mathbf{Y} - \mathbf{F}\mathbf{w} - \mathbf{G}\boldsymbol{\theta})^T (\mathbf{Y} - \mathbf{F}\mathbf{w} - \mathbf{G}\boldsymbol{\theta}) \\ &= \mathbf{Y}^T \mathbf{Y} - 2\mathbf{w}^T \mathbf{b} + \mathbf{w}^T \mathbf{V} \mathbf{w} + \\ &\quad 2\mathbf{w}^T \mathbf{Z} \boldsymbol{\theta} - 2\boldsymbol{\theta}^T \mathbf{d} + \boldsymbol{\theta}^T \mathbf{U} \boldsymbol{\theta} \end{aligned} \quad (4)$$

where,

$$\begin{aligned} \mathbf{V} &= [\mathbf{F}^T \mathbf{F}]_{4 \times 4}, \mathbf{b} = [\mathbf{F}^T \mathbf{Y}]_{4 \times 1}, \mathbf{Z} = [\mathbf{F}^T \mathbf{G}]_{4 \times 3}, \\ \mathbf{d} &= [\mathbf{G}^T \mathbf{Y}]_{3 \times 1}, \mathbf{U} = [\mathbf{G}^T \mathbf{G}]_{3 \times 3}. \end{aligned}$$

Now from $(\partial/\partial \mathbf{w}) (\text{SSE}) = 0$ and $(\partial/\partial \boldsymbol{\theta}) (\text{SSE}) = 0$, we get the following system of linear equations:

$$\begin{cases} \mathbf{V}\mathbf{w} + \mathbf{Z}\boldsymbol{\theta} = \mathbf{b} \\ \mathbf{Z}^T \mathbf{w} + \mathbf{U}\boldsymbol{\theta} = \mathbf{d} \end{cases} \quad (5)$$

By solving (5), the optimal combination weights can be obtained as:

$$\begin{cases} \boldsymbol{\theta}_{\text{opt}} = (\mathbf{U} - \mathbf{Z}^T \mathbf{V}^{-1} \mathbf{Z})^{-1} (\mathbf{d} - \mathbf{Z}^T \mathbf{V}^{-1} \mathbf{b}) \\ \mathbf{w}_{\text{opt}} = \mathbf{V}^{-1} (\mathbf{b} - \mathbf{Z} \boldsymbol{\theta}_{\text{opt}}) \end{cases} \quad (6)$$

These optimal weights are determinable if and only if all the matrix inverses, involved in (6) are well-defined.

3.3 Approach for Weights Determination

The optimal weights in the proposed ensemble technique solely depend on the knowledge of the forecast SSE value. But, in practical applications it is unknown in advance, since the dataset \mathbf{Y} to be forecasted is unknown. Due to this reason, we suggest a robust mechanism for estimating the combination weights from the training data. Here, we divide the available time series into a suitable number of pairs of training and validation subsets and determine the optimal weights for each pairs; the desired combination weights are then calculated as the mean of all these pairwise optimal weights. In this way, the past forecasting performances of the participating models are effectively utilized for weights determination.

The necessary steps of our ensemble scheme are outlined in Alg. 1.

Algorithm 1. Weighted nonlinear ensemble of multiple forecasts

Inputs: The training data: $\mathbf{Y} = [y_1, y_2, \dots, y_N]^T$ of the associated time series and its n forecasts, obtained from the models: M_i ($i = 1, 2, \dots, n$).

Output: The combined forecast vector $\hat{\mathbf{Y}}^{(c)} = [\hat{y}_1^{(c)}, \hat{y}_2^{(c)}, \dots, \hat{y}_N^{(c)}]^T$.

Steps:

1. Select *base_size*, *validation_window* and the positive integer k , such that:

$$base_size + k \times validation_window = N.$$

2. Set $j \leftarrow 1$.
3. $\mathbf{W} \leftarrow \text{empty}$, $\Theta \leftarrow \text{empty}$
 // initially set both the final weight matrices \mathbf{W} and Θ as the empty matrices of orders $n \times 1$ and $\binom{n}{2} \times 1$, respectively.
4. **while** $j \leq k$ **do**
5. Define:

$$\alpha = base_size + (j - 1) \times validation_window.$$

$$\mathbf{Y}_{\text{train}} = [y_1, y_2, \dots, y_\alpha]^T.$$

$$\mathbf{Y}_{\text{validation}} = [y_{\alpha+1}, y_{\alpha+2}, \dots, y_{\alpha+validation_window}]^T.$$

// this step selects a pair of training and validation subsets of \mathbf{Y} .

6. Train each model M_i ($i = 1, 2, \dots, n$) on $\mathbf{Y}_{\text{train}}$ to forecast the corresponding $\mathbf{Y}_{\text{validation}}$ dataset.
7. Determine the optimal combination weight vectors \mathbf{w}_k and θ_k using (6).
8. $\mathbf{W} = [\mathbf{W} | \mathbf{w}_k]$, $\Theta = [\Theta | \theta_k]$ // augment the currently found weight vectors to the corresponding weight matrices.
9. $j \leftarrow j + 1$.
10. **end while**
11. Calculate the final weight vectors \mathbf{w}_{comb} and θ_{comb} as:

$$\mathbf{w}_{\text{comb}} = \text{mean}(\mathbf{W}, \text{row-wise}).$$

$$\theta_{\text{comb}} = \text{mean}(\Theta, \text{row-wise}).$$

12. Use \mathbf{w}_{comb} and θ_{comb} to calculate the combined forecast vector according to (3).
-

4 Three Time Series Forecasting Models

In this paper, we consider three popular time series forecasting methods to build up our proposed ensemble. These three methods are briefly described here.

4.1 Autoregressive Integrated Moving Average (ARIMA)

The ARIMA models are the most widely used methods for time series forecasting, which are developed by Box and Jenkins in 1970 [2]. These models are based on the assumption that the successive observations of a time series are linearly generated from the past values and a random noise process. Mathematically, an ARIMA(p, d, q) model is represented as follows:

$$\phi(L)(1-L)^d y_t = \theta(L)\epsilon_t \quad (7)$$

where,

$$\phi(L) = 1 - \sum_{i=1}^p \phi_i L^i, \quad \theta(L) = 1 + \sum_{j=1}^q \theta_j L^j, \quad \text{and } Ly_t = y_{t-1}.$$

The terms p, d, q are the model orders, which respectively refer to the *autoregressive*, *degree of differencing* and *moving average processes*; y_t is the actual time series and ϵ_t is a white noise process. In this model, a nonstationary time series is transformed to a stationary one by successively (d times) differencing it [2,4]. A single differencing is often sufficient for practical applications. The ARIMA(0, 1, 0), i.e. $y_t - y_{t-1} = \epsilon_t$ is the popular Random Walk (RW) model which is frequently used in forecasting financial and stock-market data [4].

4.2 Artificial Neural Networks (ANNs)

ANNs are the most efficient computational intelligence models for time series forecasting [10]. Their outstanding characteristic is the nonlinear, nonparametric, data-driven and self-adaptive nature [4,13]. The *Multilayer Perceptrons (MLPs)* are the most popular ANN architectures in time series forecasting. MLPs are characterized by a feedforward network of an input layer, one or more hidden layers and an output layer, as depicted in Fig. 1. Each layer contains a number of nodes which are connected to those in the immediate next layer by acyclic links. In practical applications, usually a single hidden layer is used [4,10,13].

The notation (p, h, q) is commonly used to refer an ANN with p input, h hidden and q output nodes. The forecasting performance of an ANN model depends on a number of factors, e.g. the selection of a proper network architecture, training algorithm, activation functions, significant time lags, etc. However, no rigorous theoretical framework is available in this regard and often some experimental guidelines are followed [13]. In this paper, we use popular model selection criteria, e.g. *Akaike Information Criterion (AIC)* and *Bayesian Information Criterion (BIC)* [13,14] for selecting suitable ANN structures. The *Resilient Propagation*

(*RP*) [15,16] is applied as the network training algorithm and the logistic and identity functions are used as the hidden and output layer activation functions, respectively.

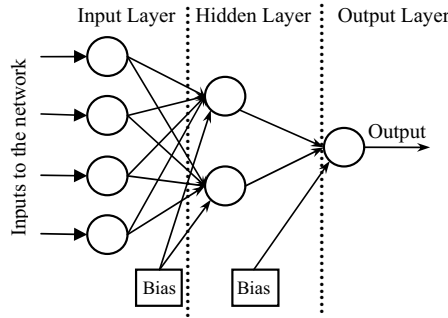


Fig. 1. Example of a multilayer feedforward ANN

4.3 Elman Artificial Neural Networks (EANNs)

Elman networks belong to the class of recurrent neural networks in which one extra layer, known as the context layer is introduced to recognize the spatial and temporal patterns in the input data [17]. The Elman networks contain two types of connections: feedforward and feedback. At every step, the outputs of the hidden layer are again fed back to the context layer, as shown in Fig. 2. This recurrence makes the network dynamic, so that it can perform non linear time-varying mappings of the associated nodes [16,17]. Unlike MLPs, there seems to be no general model selection guidelines in literature for the Elman ANNs [10]. However, it is a well-known fact that EANNs require much more hidden nodes than the simple feedforward ANNs in order to adequately model the temporal relationships [10,16]. In this paper, we use 24 hidden nodes and the training algorithm *trainrnn* [16] for fitting EANNs.

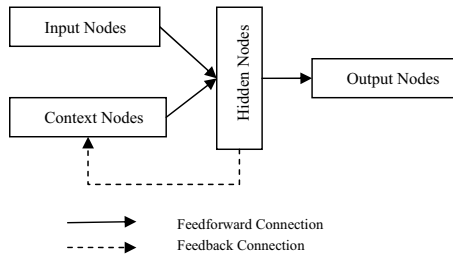


Fig. 2. Architecture of an Elman network

5 Experiments and Discussions

To empirically examine the performances of our proposed ensemble technique, three important real-world time series are used in this paper. These are the Wolfs sunspots, the daily closing price of S & P 500 index and the exchange rates between US Dollar (USD) and Indian Rupee (INR) time series. These time series are obtained from the Time Series Data Library (TSDL) [18], the Yahoo! Finance [19] and the Pacific FX database [20], respectively and are described in Table 1. The natural logarithms of the S & P data are used in our analysis. All three time series are divided into suitable training and testing sets. The training sets are used for fitting the three forecasting models as well as to build up the proposed ensemble; the testing sets are used to evaluate the out-of-sample forecasting performances of the fitted models and the ensemble.

The experiments in this paper are performed using MATLAB. For fitting ANN and EANN models, the neural network toolbox [16] is used. Forecasting efficacies of the models are evaluated through three well-known error statistics, viz. Mean Absolute Error (MAE), Mean Squared Error (MSE), and Average Relative Variance (ARV), which are defined below:

$$\text{MAE} = \frac{1}{n} \sum_{t=1}^n |y_t - \hat{y}_t|, \quad \text{MSE} = \frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2,$$

$$\text{ARV} = \left(\sum_{t=1}^n (y_t - \hat{y}_t)^2 \right) / \left(\sum_{t=1}^n (\mu - \hat{y}_t)^2 \right),$$

where, y_t and \hat{y}_t are the actual and forecasted observations, respectively; N is the size and μ is the mean of the test set. For an efficient forecasting model, the values of these error measures are expected to be as less as possible.

The sunspots series is stationary with an approximate cycle of 11 years, as can be seen from Fig. 3(a). Following Zhang [4], the ARIMA(9, 0, 0) (i.e. AR(9))

Table 1. Description of the time series datasets

Time Series	Description	Size
Sunspots	The annual number of observed sunspots (1700–1987).	Total size: 288 Training: 171 Testing: 67
S & P 500	Daily closing price of S & P 500 index (2 Jan. 2004–31 Dec. 2007)	Total size: 1006 Training: 800 Testing: 206
Exchange Rate	USD to INR exchange rates (1 July 2009–16 Sept. 2011)	Total size: 681 Training: 565 Testing: 116

and the (7, 5, 1) ANN models are fitted to this time series. The EANN model is fitted with same numbers of input and output nodes as the ANN, but with 24 hidden nodes. For combining, we take *base_size* = 41, *validation_window* = 20 and the number of iterations *k* = 9.

The S & P and exchange rate are nonstationary financial series and both exhibit quite irregular patterns which can be observed from their respective time plots in Fig. 4(a) and Fig. 5(a). The RW-ARIMA model is most suitable for these type of time series¹. For ANN modeling, the (8, 6, 1) and (6, 6, 1) network structures are used for S & P and exchange rate, respectively. As usual, the fitted EANN models have the same numbers of input and output nodes as the corresponding ANN models, but 24 hidden nodes. For combining, we take *base_size* = 200, *validation_window* = 50, *k* = 12 for the S & P data and *base_size* = 165, *validation_window* = 40, *k* = 10 for the exchange rate data.

In Table 2, we present the forecasting performances of ARIMA, ANN, EANN, simple average and the proposed ensemble scheme for all three time series.

Table 2. Forecasting results for the three time series

Error Measures		ARIMA	ANN	EANN	Average	Ensemble
Sunspots	MAE	17.63	15.58	14.71	13.59	12.50
	MSE	483.5	494.9	492.7	384.3	274.7
	ARV	0.216	0.308	0.280	0.218	0.120
S & P 500 ²	MAE	12.68	12.33	13.27	10.43	9.368
	MSE	28.27	21.58	24.61	15.69	13.59
	ARV	0.344	0.378	0.397	0.271	0.230
Exchange Rate	MAE	0.255	0.140	0.137	0.134	0.133
	MSE	0.105	0.032	0.030	0.029	0.028
	ARV	0.188	0.070	0.064	0.064	0.053

From Table 2, it can be seen that our ensemble technique has provided lowest forecast errors among all methods. Moreover, the proposed technique has also achieved considerably better forecasting accuracies than the simple average combination method, for all three time series. However, we have empirically observed that like the simple average, the performance of our ensemble method is also quite sensitive to the extreme errors of the component models.

In this paper, we use the term *Forecast Diagram* to refer the graph which shows the actual and forecasted observations of a time series. In each forecast diagram, the solid and dotted line respectively represents the test and forecasted time series. The forecast diagrams, obtained through our proposed ensemble method for sunspots, S & P and exchange rate series are depicted in Fig. 3(b), Fig. 4(b) and Fig. 5(b), respectively.

¹ In RW-ARIMA, the preceding observation is the best guide for the next prediction.

² Original MAE=Obtained MAE $\times 10^{-3}$; Original MSE=Obtained MSE $\times 10^{-5}$.

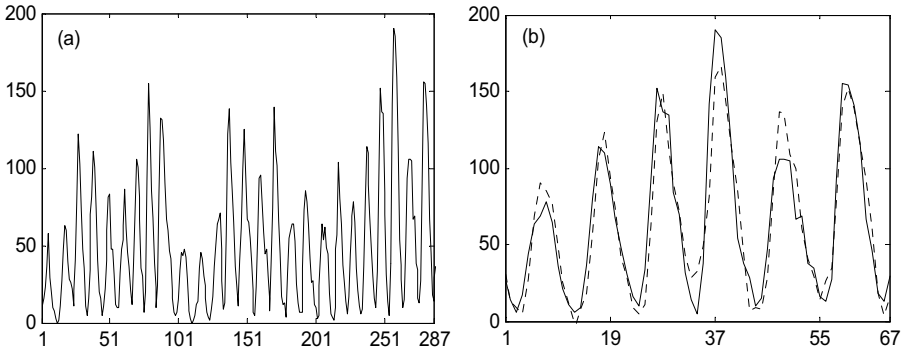


Fig. 3. (a) The sunspots series, (b) Ensemble forecast diagram for the sunspot series

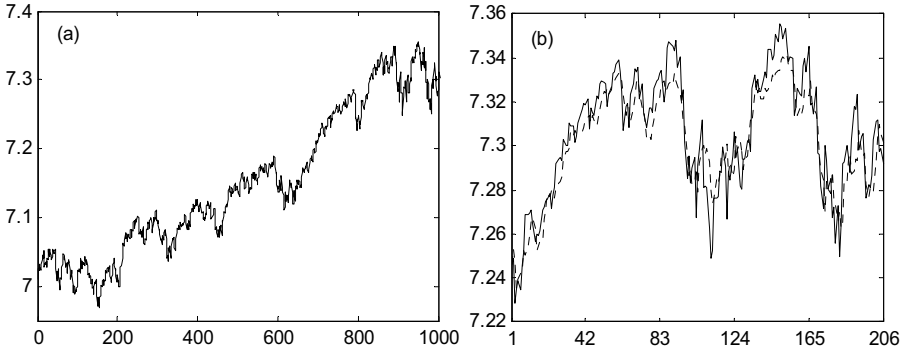


Fig. 4. (a) The S & P series, (b) Ensemble forecast diagram for the S & P series

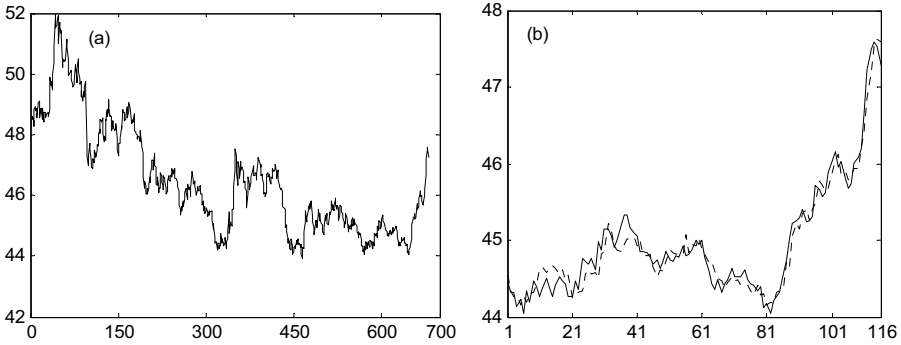


Fig. 5. (a) Exchange rate series, (b) Ensemble forecast diagram for exchange rate series

6 Conclusions

Improving the accuracy of time series forecasting is a major area of concern in many practical applications. Although numerous forecasting methods have been developed during the past few decades, but it is often quite difficult to select the best among them. It has been observed by many researchers that combining multiple forecasts effectively reduces the prediction errors and hence provides considerably increased accuracy.

In this paper, we propose a novel nonlinear weighted ensemble technique for forecasts combination. It is an extension of the common linear combination scheme in order to include possible correlation effects between the participating forecasts. An efficient successive validation mechanism is suggested for determining the appropriate combination weights. The empirical results with three real-world time series and three forecasting methods demonstrate that our proposed technique significantly outperforms each individual method in terms of obtained forecast accuracies. Moreover, it also provides considerably better results than the classic simple average combining technique. In future works, our ensemble mechanism can be further explored with other diverse forecasting models as well as other varieties of time series data.

Acknowledgments. The first author sincerely expresses his profound gratitude to the Council of Scientific and Industrial Research (CSIR) for the obtained financial support, which helped a lot in performing this research work.

References

1. Gooijer, J.G., Hyndman, R.J.: 25 Years of time series forecasting. *J. Forecasting* 22(3), 443–473 (2006)
2. Box, G.E.P., Jenkins, G.M.: *Time Series Analysis: Forecasting and Control*, 3rd edn. Holden-Day, California (1970)
3. Armstrong, J.S.: Combining Forecasts. In: Armstrong, J.S. (ed.) *Principles of Forecasting: A Handbook for Researchers and Practitioners*. Kluwer Academic Publishers, Norwell (2001)
4. Zhang, G.P.: Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing* 50, 159–175 (2003)
5. Bates, J.M., Granger, C.W.J.: Combination of forecasts. *Operational Research Quarterly* 20(4), 451–468 (1969)
6. Clemen, R.T.: Combining forecasts: A review and annotated bibliography. *J. Forecasting* 5(4), 559–583 (1989)
7. Aksu, C., Gunter, S.: An empirical analysis of the accuracy of SA, OLS, ERLS and NRLS combination forecasts. *J. Forecasting* 8(1), 27–43 (1992)
8. Zou, H., Yang, Y.: Combining time series models for forecasting. *J. Forecasting* 20(1), 69–84 (2004)
9. Jose, V.R.R., Winkler, R.L.: Simple robust averages of forecasts: Some empirical results. *International Journal of Forecasting* 24(1), 163–169 (2008)
10. Lemke, C., Gabrys, B.: Meta-learning for time series forecasting and forecast combination. *Neurocomputing* 73, 2006–2016 (2010)

11. Bunn, D.: A Bayesian approach to the linear combination of forecasts. *Operational Research Quarterly* 26(2), 325–329 (1975)
12. Frietas, P.S., Rodrigues, A.J.: Model combination in neural-based forecasting. *European Journal of Operational Research* 173(3), 801–814 (2006)
13. Zhang, G., Patuwo, B.E., Hu, M.Y.: Forecasting with artificial neural networks: The state of the art. *J. Forecasting* 14, 35–62 (1998)
14. Faraway, J., Chatfield, C.: Time series forecasting with neural networks: a comparative study using the airline data. *J. Applied Statistics* 47(2), 231–250 (1998)
15. Reidmiller, M., Braun, H.: A direct adaptive method for faster backpropagation learning: The rprop algorithm. In: *Proceedings of the IEEE Int. Conference on Neural Networks (ICNN)*, San Francisco, pp. 586–591 (1993)
16. Demuth, M., Beale, M., Hagan, M.: *Neural Network Toolbox User's Guide*. The MathWorks, Natic (2010)
17. Lim, C.P., Goh, W.Y.: The application of an ensemble of boosted Elman networks to time series prediction: A benchmark study. *J. of Computational Intelligence* 3, 119–126 (2005)
18. Time Series Data Library, <http://robjhyndman.com/TSDL>
19. Yahoo! Finance, <http://finance.yahoo.com>
20. Pacific FX database, <http://fx.sauder.ubc.ca/data.html>