

Research Article

Forecasting Computer Products Sales by Integrating Ensemble Empirical Mode Decomposition and Extreme Learning Machine

Chi-Jie Lu¹ and Yuehjen E. Shao²

¹ *Department of Industrial Management, Chien Hsin University of Science and Technology, Taoyuan County 32097, Zhongli, Taiwan*

² *Department of Statistics and Information Science, Fu Jen Catholic University, Xinzhuang District, New Taipei City 24205, Taiwan*

Correspondence should be addressed to Yuehjen E. Shao, stat1003@mail.fju.edu.tw

Received 30 August 2012; Revised 12 November 2012; Accepted 13 November 2012

Academic Editor: Zexuan Zhu

Copyright © 2012 C.-J. Lu and Y. E. Shao. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A hybrid forecasting model that integrates ensemble empirical model decomposition (EEMD), and extreme learning machine (ELM) for computer products sales is proposed. The EEMD is a new piece of signal processing technology. It is based on the local characteristic time scales of a signal and could decompose the complicated signal into intrinsic mode functions (IMFs). The ELM is a novel learning algorithm for single-hidden-layer feedforward networks. In our proposed approach, the initial task is to apply the EEMD method to decompose the original sales data into a number of IMFs. The hidden useful information of the original data could be discovered in those IMFs. The IMFs are then integrated with the ELM method to develop an effective forecasting model for computer products sales. Experimental results from three real computer products sales data, including hard disk, display card, and notebook, showed that the proposed hybrid sales forecasting method outperforms the four comparative models and is an effective alternative for forecasting sales of computer products.

1. Introduction

In an information technology company, forecasting computer products sales is one of the most important and challenging tasks for business financial planning, inventory management, marketing, and customer service. Improving sales forecast accuracy can lead to significant monetary savings, greater competitiveness, enhanced channel relationships, and customer satisfaction. However, developing a proper sale forecast model for computer products is actually a difficult task due to the demand uncertainty, the short lifespan, and quick obsolescence of the products.

The literature demonstrates a great interest in using neural networks for sales forecast [1–6]. It is due to that neural networks can capture subtle functional relationships among the empirical data even though the underlying relationships are unknown or difficult to describe. As neural networks have been successfully applied in sales forecast [1, 2, 7, 8], a neural network model called extreme learning machine (ELM) is used in this study for sales forecast of computer products.

ELM is a novel learning algorithm for single-hidden-layer feedforward networks (SLFN) [9–13]. In ELM, the input weights and hidden biases are randomly chosen, and the output weights are analytically determined by using the Moore-Penrose generalized pseudoinverse [9, 10]. Different from traditional gradient-based learning algorithms for neural networks, ELM not only tends to reach the smallest training error but also the smallest norm of output weights. Accordingly, the ELM algorithm provides much better generalization performance with much faster learning speed and avoids many issues faced in the traditional algorithms such as stopping criterion, learning rate, number of epochs, and local minima and the overtuned problems [10–13]. ELM has attracted a lot of attentions in recent years and has become an important method in sales forecasting [3, 4, 14, 15]. But, to the best of the authors' knowledge, ELM has not been used to forecast computer products sales.

For most existing studies, the observed original values are usually directly used for building sales forecasting models [3, 6, 7, 14, 15]. However, owing to the high-frequency, nonstationary, and chaotic properties of the sales data of computer products, a sales forecasting model utilizing the original sales data often leads to an inability to provide satisfying forecast results. To solve this problem, before constructing a forecasting model, many studies would initially utilize information extraction techniques to extract features contained in data, then use these extracted characteristics to construct the forecasting model [1, 16–19]. That is, the useful or interesting information may not be observed directly from the observed original data, but they can be revealed in the extracted features through suitable feature extractions or signal processing methods. As a result, the prediction performance will be improved by using those features to develop an effective forecasting model.

Empirical mode decomposition (EMD) is a new signal processing technique proposed by Huang et al. [20, 21]. It is based on the local characteristic time scales of a signal and could decompose the complicated signal into intrinsic mode functions (IMFs) [20, 21]. The IMFs represent the natural oscillatory mode embedded in the signal and work as the basis functions, which are determined by the signal itself. Thus, it is a self-adaptive signal processing method and has been employed successfully in time series analysis and forecasting [17, 22, 23]. However, one of the major drawbacks of EMD is the mode mixing problem, which is defined as either a single IMF consisting of components of widely disparate scales, or a component of a similar scale residing in different IMFs. To alleviate the problem of mode mixing in EMD, ensemble empirical mode decomposition (EEMD) was presented [24]. EEMD is a noise-assisted data analysis method, and, by adding finite white noise to the investigated signal, the EEMD method can eliminate the mode mixing problem automatically. Therefore, the EEMD gives a major improvement of EMD. However, there are only few studies using EEMD in time series forecasting and achieved satisfactory results [25]. In addition, there are still no studies that can perform sales forecasting on the basis of EEMD method.

In this study, we propose a hybrid sales forecasting model by integrating EEMD and ELM for computer products. Our proposed methodology consists of two steps: first, the EEMD method is applied to convert original sales data into a number of IMFs of original

data. The IMFs represent underlying information of the original data. That is, the hidden information of the original data could be discovered in those IMFs. Then the IMFs are used in the ELM method to develop an effective sales forecasting model for computer products. In order to evaluate the performance of the proposed hybrid sales forecasting procedure, three computer product sales data, that is, hard disk (HD), display card (DC), and notebook (NB), collected from an IT chain store in Taiwan, are used as the illustrative examples. The superior forecasting capability of the proposed technique can be observed by comparing the results with the four comparison models including single ELM, single support vector regression (SVR), single backpropagation neural network (BPN), and the combined model of EMD and ELM models.

The rest of this study is organized as follows. Section 2 gives brief overviews of EEMD and ELM. The proposed model is described in Section 3. Section 4 presents the experimental results, and this study is concluded in Section 5.

2. Methodologies

2.1. Ensemble Empirical Mode Decomposition

Ensemble empirical mode decomposition (EEMD) is an improved version of the popular empirical mode decomposition (EMD), the aim of which was to overcome intrinsic drawbacks of mode mixing in EMD [24]. In EEMD method, white noise is introduced to help separate disparate time series scales and improve the decomposition performance of EMD. The added white noise constitutes components of different scale and would uniformly fill the whole time-frequency space. When the uniformly distributed white noise is added to a signal, the different scale components of the signal are automatically projected onto proper scales of reference established by the white noise. Since each decomposed component of noise added contains the signal and the added white noise, each individual trial is certain to get noisy results. As the noise in each trial is different in separate trials, the noise can be almost completely removed by the ensemble mean of entire trials [24]. Then, the ensemble mean can be used to represent the true underlying components. It is applicable for extracting underlying and useful information from the sales data. For more details about EEMD method, please refer to [24].

For a given sales data $x(t)$, the EEMD procedure can be described as follows.

Step 1. Initialize the ensemble number M and amplitude of the added white noise.

Step 2. Execute the m th trial for adding random white noise $w_m(t)$ into $x(t)$ to generate the noise-added data $x_m(t)$, $x_m(t) = x(t) + w_m(t)$.

Step 3. Identify all the local maxima and minima of $x_m(t)$ and generate the upper and lower envelopes using the cubic spline functions.

Step 4. Calculate the mean $m_1(t)$ of the upper and lower envelopes and find the difference $h_1(t)$ between the signal and the mean, that is, $h_1(t) = x_m(t) - m_1(t)$.

Step 5. If h_1 satisfies the properties of IMF, $c_1(t) = h_1(t)$ is the first IMF component from the signal. If not, replace $x_m(t)$ with $h_1(t)$ and go to Step 3. The properties of IMF are (1) the number of extrema and the number of zero crossing must either equal or differ at most by one

in the whole dataset and (2) the mean value of the envelopes defined by the local maximum and the local minimum is zero at any point.

Step 6. Separate $r_1(t)$ from the rest of the data by

$$r_1(t) = x_m(t) - c_1(t). \quad (2.1)$$

Regard the residue $r_1(t)$ as a new signal and repeat Step 3 to Step 6 n times to sift out other IMFs until stopping criteria are satisfied. The stopping criteria can be either when the IMF component, $c_n(t)$, or the residue, $r_n(t)$, is so small that it is less than the predetermined value, or when the residue, $r_n(t)$, becomes a monotonic function from which no more IMF can be extracted. After decomposition, the original signal can be represented as the sum of all IMFs and the residue, $r_n(t)$. Consider

$$x_m(t) = \sum_{i=1}^n c_i(t) + r_n(t), \quad (2.2)$$

where n is the number of IMFs, $r_n(t)$ is the final residue, and $c_i(t)$ is the i th IMF.

Step 7. If $m < M$, let $m = m + 1$ and repeatedly perform from Steps 2 to 6 until $m = M$, but with different white noise each time.

Step 8. Calculate the ensemble mean \bar{c}_i of the M trials for the each IMF, that is, the i th ensemble IMF $\bar{c}_i = (1/M) \sum_{m=1}^M c_{i,m}$, $i = 1, \dots, n$, and the ensemble residue $\bar{r}_n = (1/M) \sum_{m=1}^M r_{n,m}$.

All IMFs are nearly orthogonal to each other, and all have nearly zero means. Thus, the sales data can be decomposed into n ensemble IMFs and one ensemble residue. The IMF components contained in each frequency band are different, and they change with variation of sales data $x(t)$, while ensemble residue represents the central tendency of sales data $x(t)$.

2.2. Extreme Learning Machine

Extreme learning machine (ELM) proposed is a single hidden layer feed-forward networks (SLFNs) which randomly selected the input weights and analytically determines the output weights of SLFNs [9–13]. One key principle of the ELM is that one may randomly choose and fix the hidden node parameters. After the hidden nodes parameters are chosen randomly, SLFN becomes a linear system where the output weights of the network can be analytically determined using simple generalized inverse operation of the hidden layer output matrices.

Consider N arbitrary distinct samples (x_i, t_i) where $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$, and $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in R^m$. SLFNs with \tilde{N} hidden neurons and activation function $g(x)$ can approximate N samples with zero error. This means that

$$H\beta = T, \quad (2.3)$$

where

$$H(w_1, \dots, w_{\tilde{N}}, b_1, \dots, b_{\tilde{N}}, x_1, \dots, x_N) = \begin{bmatrix} g(w_1 \cdot x_1 + b_1) & \cdots & g(w_{\tilde{N}} \cdot x_1 + b_{\tilde{N}}) \\ \vdots & \ddots & \vdots \\ g(w_1 \cdot x_N + b_1) & \cdots & g(w_{\tilde{N}} \cdot x_N + b_{\tilde{N}}) \end{bmatrix}_{N \times \tilde{N}},$$

$$\beta_{\tilde{N} \times m} = (\beta_1^T, \dots, \beta_{\tilde{N}}^T)^t; \quad T_{N \times m} = (T_1^T, \dots, T_N^T)^t, \quad (2.4)$$

where $w_i = [w_{i1}, w_{i2}, \dots, w_{im}]^T$, $i = 1, 2, \dots, \tilde{N}$, is the weight vector connecting the i th hidden node and the input nodes, $\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$ is the weight vector connecting the i th hidden node and the output nodes, and b_i is the threshold of the i th hidden node. $w_i \cdot x_j$ denotes the inner product of w_i and x_j . H is called the hidden layer output matrix of the neural network; the i th column of H is the i th hidden node output with respect to inputs x_1, x_2, \dots, x_N .

Thus, the determination of the output weights (linking the hidden layer to the output layer) is as simple as finding the least-square solution to the given linear system. The minimum norm least-square (LS) solution to the linear system (i.e., (2.3)) is

$$\hat{\beta} = H^\Psi T, \quad (2.5)$$

where H^Ψ is the Moore-Penrose generalized inverse of matrix H . The minimum norm LS solution is unique and has the smallest norm among all the LS solutions.

Step of ELM algorithm can be summarized as follows.

Step 1. Randomly assign input weight w_i , and bias b_i .

Step 2. Calculate the hidden layer output matrix H .

Step 3. Calculate the output weight β , $\hat{\beta} = H^\Psi T$, where $T = (t_1, \dots, t_N)^t$.

3. The Proposed EEMD-ELM Sales Forecasting Method

This paper proposed a novel sales forecasting model for computer products by integrating EEMD and ELM. The research scheme of the proposed methodology is presented in Figure 1.

As shown in Figure 1, the proposed methodology consists of two steps. In the first step, we use EEMD to decompose original data into a number of IMFs to represent underlying information of the original data.

In EEMD, the number of trials in the ensemble and the amplitude of added white noise are two key parameters needed to be carefully selected. The standard deviation of error is $e = a/\sqrt{N}$, where a is the amplitude of noise added and N the number of trials.

To reduce the error, it seems that the smaller noise amplitude is better. But if the noise amplitude is too small, it may not introduce the change of extrema that the EMD relies on. Hence the noise amplitude should not be too small. Under this condition, the noise effect can be reduced to be negligible by increasing the number of trials. To make EEMD effective, the amplitude of noise is suggested to be 0.2 times the standard deviation of the signal, and the

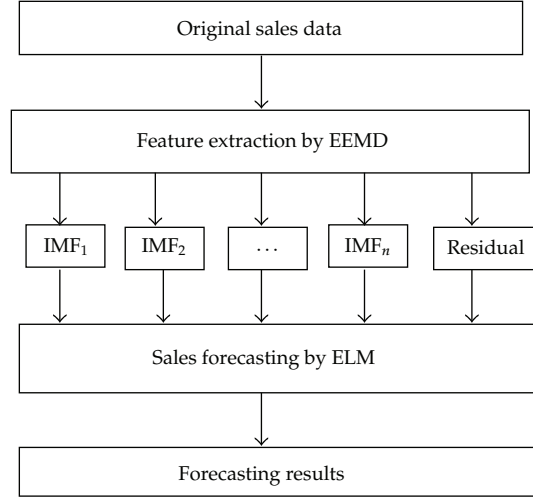


Figure 1: Proposed hybrid EEMD-ELM sales forecasting model.

number of trials be a few hundred. In reality, the number of ensemble is often set to 100; the standard deviation of the added white noise is set to 0.1 to 0.3 [26].

In the second step, the IMFs are integrated into the ELM approach to develop sales forecasting model for computer products. As discussed in Section 2.2, it is known that the most important and critical ELM parameter is the number of hidden nodes and that ELM tends to be unstable in single run forecasting [3, 4, 10]. Therefore, the ELM models with different numbers of hidden nodes varying from 1 to 15 are constructed. For each number of nodes, an ELM model is repeated 30 times and the average RMSE of each node is calculated. The number of hidden nodes that gives the smallest average RMSE value is selected as the best parameter of ELM model.

4. Experimental Results

4.1. Datasets and Performance Criteria

For evaluating the performance of the proposed EEMD-ELM forecasting model, the daily sales data of three computer products, that is, hard disk (HD), notebook (NB), and display card (DC), are used in this study. Figures 2(a)–2(c) show, respectively, the daily sales of HD, NB, and DC from 2006/1/1 to 2009/9/12. For each product, there are totally 1351 data points in the dataset. The first 1000 data points (74% of the total sample points) are used as the training sample, and the remaining 351 data points (26% of the total sample points) are holdout and used as the testing sample for out of sample forecasting. The moving (or rolling) window technique is used to forecasting the training and testing data.

To demonstrate the effectiveness of the proposed model, the performance of the proposed EEMD-ELM method is compared to the single ELM, single SVR, single BPN, and the integrated EMD and ELM models (called EMD-ELM model) in this section. The SVR based on statistical learning theory is a novel neural network algorithm [27] and has been successfully applied in sales forecasting [1, 6]. The BPN is the most popular neural network training algorithm for time series forecasting problems [28, 29]. The single ELM/BPN/SVR

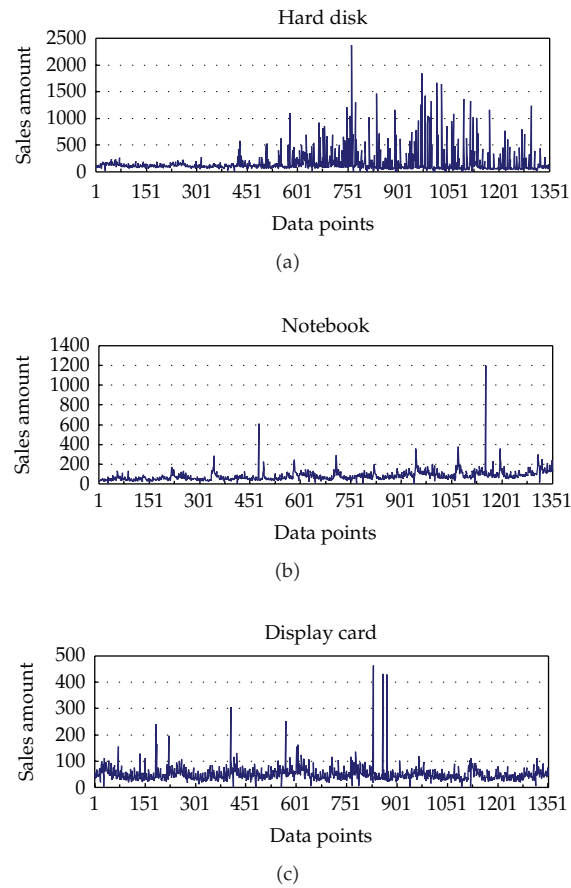


Figure 2: Daily sales amount of three computer products from 2006/1/1 to 2009/9/12 (a) hard disk, (b) notebook, and (c) display card.

models simply apply the ELM/BPN/SVR methods to input variables to forecast the index closing price without using EEMD/EMD as a preprocessing tool. The EMD-ELM method first applies EMD to the original sales data for decomposing IMFs and then develops ELM model using the IMFs as inputs.

The three input variables used for single ELM, single SVR, and single BPN models are the previous day's sales volume ($t-1$), sales volume 2 days previous ($t-2$), and sales volume 3 days previous ($t-3$).

To build a single BPN model, the neural network toolbox of MATLAB software is adapted in this study. The LIBSVM package (LIBSVM package: <http://www.csie.ntu.edu.tw/~cjlin/libsvm/index.html>) is adapted for developing single SVR model. For constructing single ELM, EMD-ELM, and EEMD-ELM models, the ELM package (ELM package: <http://www.ntu.edu.sg/home/egbhuang/ELM.Codes.htm>) is used in this study. The forecasting models constructed in this study are carried out in MATLAB 7.4 environment running in an Intel Core 2, 2.5 GHZ CPU. Note that the default settings of neural network toolbox, LIBSVM package, and ELM package are used.

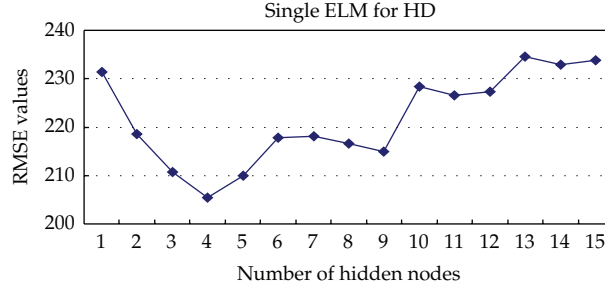


Figure 3: The average RMSE values of the single ELM model with different numbers of hidden nodes for HD.

The prediction performance is evaluated using the root mean square error (RMSE), the mean absolute difference (MAD), and the mean absolute percentage error (MAPE). The definitions of these criteria are as below:

$$\begin{aligned}
 \text{RMSE} &= \sqrt{\frac{\sum_{i=1}^n (T_i - P_i)^2}{n}}, \\
 \text{MAD} &= \frac{\sum_{i=1}^n |T_i - P_i|}{n}, \\
 \text{MAPE} &= \frac{\sum_{i=1}^n |(T_i - P_i)/T_i|}{n},
 \end{aligned} \tag{4.1}$$

where T and P represent the actual and predicted value, respectively; n is the total number of data points.

4.2. Forecasting Results of HD

For forecasting sales of HD, the single ELM is initially constructed. Figure 3 shows the average RMSE values of the single ELM model with different numbers of hidden nodes. From Figure 3, it can be seen that the ELM model with four hidden nodes has the smallest average RMSE values and is therefore the best ELM model for forecasting sales of HD.

For the proposed EEMD-ELM model, first, the original sales of HD (as shown in Figure 2(a)) are decomposed to ten IMFs using EEMD. Using the same process, the EMD-ELM model uses the EMD model to estimate ten IMFs from the original sales data of HD. Figures 4 and 5, respectively, show the ten IMFs of the sales data of HD using EEMD and EMD models. It can be seen from Figures 4 and 5 that the IMFs, regardless of using EEMD or EMD for decomposition, can be used to represent different underlying factors like trends and seasonal variations that affect the sales of HD simultaneously. However, it can also be observed from the figures that the IMFs of EEMD model can enhance/discover more trend (or major) information than that of the IMFs decomposed by EMD. That is, the IMFs extracted by EMD contain more detail information of the original data and may negatively affect the forecasting performance since the original daily sales is high-frequency and nonstationary data. After obtaining the IMFs, then, the IMFs in Figures 4 and 5 are, respectively, used for building ELM prediction models of the EEMD-ELM and EMD-ELM models. In building

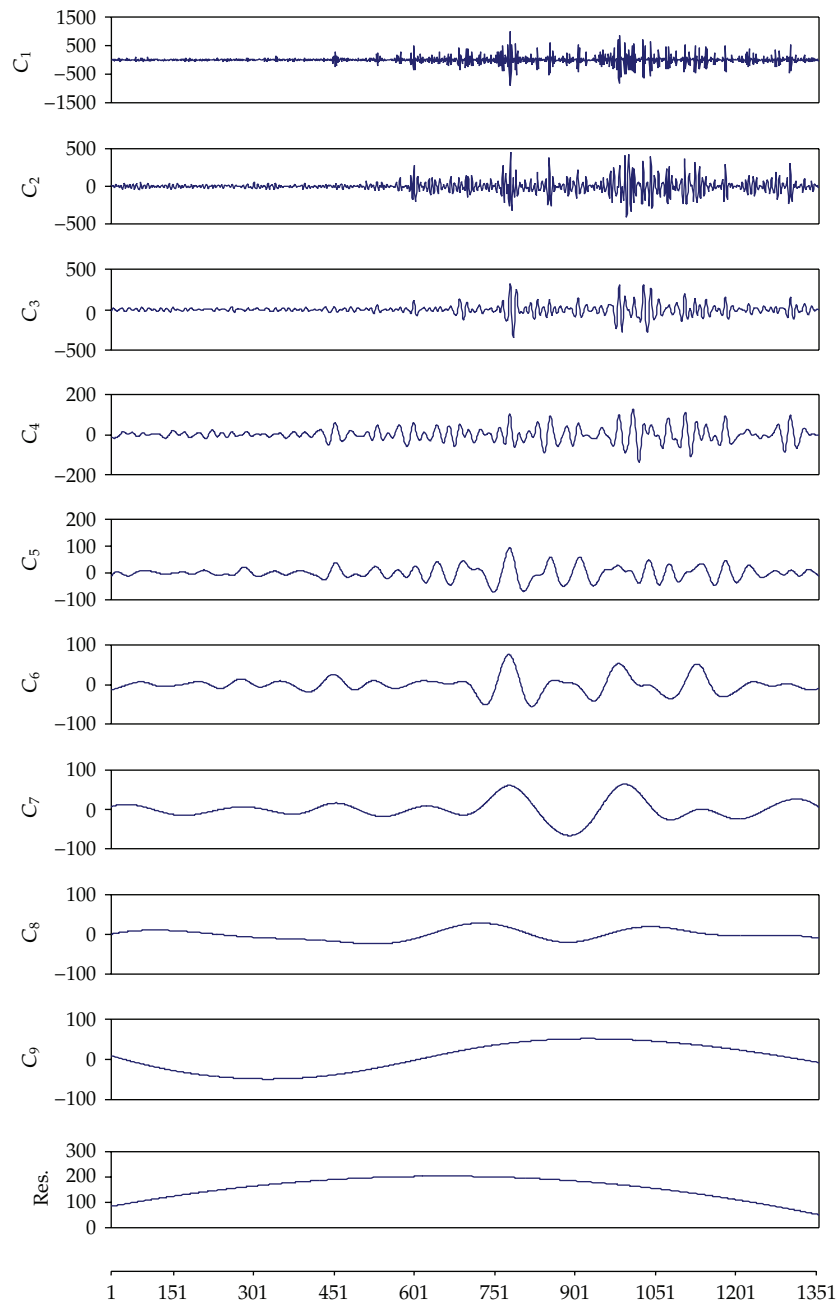


Figure 4: Decomposition results of sales data of HD via EEMD.

ELM prediction models of EEMD-ELM and EMD-ELM, like the single ELM model, different numbers of hidden nodes are tested.

From Figures 6 and 7, it can be seen that the EEMD-ELM model with ten hidden nodes and the EMD-ELM model with seven hidden nodes have the smallest average RMSE values and are therefore the best EEMD-ELM and EMD-ELM models considered in this study.

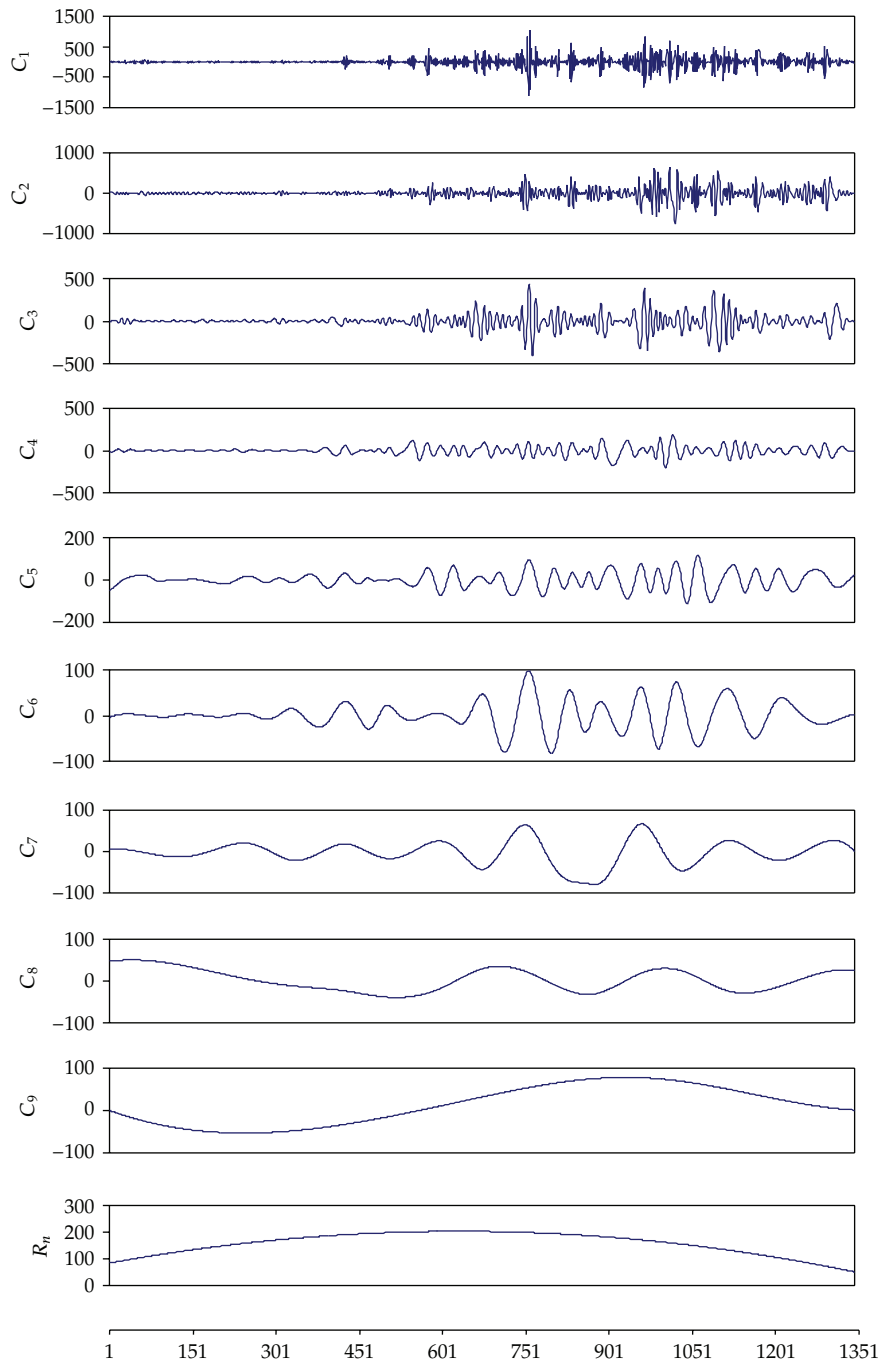


Figure 5: Decomposition results of sales data of HD via EMD.

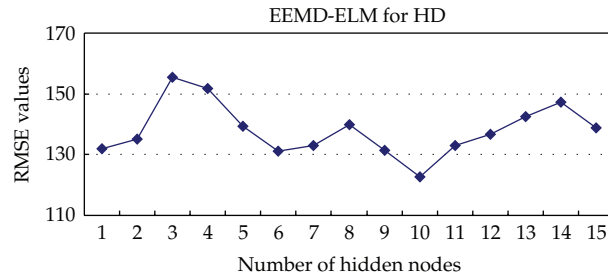


Figure 6: The average RMSE values of the EEMD-ELM models with different numbers of hidden nodes for HD.

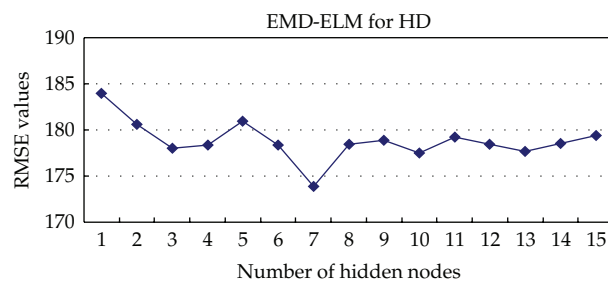


Figure 7: The average RMSE values of the EMD-ELM models with different numbers of hidden nodes for HD.

In building the single SVR model, the first step is to select the kernel function. As radial basis function (RBF) is one of the most used kernel function for SVR [30], it is adapted in this study. The selection of three parameters, regularization constant C , and loss function ε and σ (the width of the RBF) of an SVR model is important to the accuracy of forecasting [30, 31]. In this study, the grid search method proposed by Hsu et al. [31] is used. The testing results of the single SVR model with combinations of different parameter sets are summarized in Table 1. From Table 1, it can be found that the parameter set ($C = 2^{11}$, $\varepsilon = 2^3$, $\sigma = 2^5$) gives the best forecasting result (minimum testing RMSE) and is the best parameter set for single SVR model in forecasting HD sales.

In the BPN model, the input layer has 3 nodes as the 3 forecasting variables are adopted. As there are no general rules for determining the appropriate number of nodes in the hidden layer, the number of hidden nodes to be tested is set at 5, 6, and 7. The network has only one output node: the forecast HD sales. As lower learning rates tend to give the best network results [32, 33], learning rates of 0.01, 0.05, and 0.10 are tested during the training process. The network topology with the minimum testing RMSE is considered the optimal network. Table 2 reports the results of testing the single BPN model with different combinations of hidden nodes and learning rates. From Table 2, it can be observed that the 6 hidden nodes with a learning rate of 0.05 give the minimum testing RMSE and hence is the best topology setup for the single BPN model.

The forecast results of HD using the single ELM, single SVR, single BPN, and EMD-ELM and the proposed EEMD-ELM models are computed and listed in Table 3. Table 3 depicts that the RMSE, MAPE, and MAD of the proposed EEMD-ELM model are, respectively, 132.94, 9.21%, and 84.11. It can be observed that these values are smaller than

Table 1: The model selection results of single SVR model HD.

C	ϵ	σ	Testing RMSE
2^7	2^1	2^3	211.85
	2^3	2^5	211.82
	2^5	2^7	211.77
2^9	2^1	2^3	211.56
	2^3	2^5	211.52
	2^5	2^7	211.55
2^{11}	2^1	2^3	211.80
	2^3	2^5	211.82
	2^5	2^7	211.90

Table 2: The model selection results of the single BPN model HD.

Hidden nodes	Learning rate	Testing MSE
5	0.01	216.60
	0.05	218.81
	0.10	219.39
6	0.01	214.21
	0.05	213.87
	0.10	216.03
7	0.01	218.53
	0.05	217.16
	0.10	217.92

those of the single ELM, single SVR, single BPN, and EMD-ELM models. It indicates that there is a smaller deviation between the actual and predicted values using the proposed EEMD-ELM model. Thus, the proposed EEMD-ELM model provides a better forecasting result for HD sales than the four comparison models in terms of prediction error. Besides, it also can be observed from Table 3 that the training time of single ELM, EMD-ELM, and EEMD-ELM models is, respectively, 0.0373 s, 0.0701 s, and 0.0779 CPU times. However, the CPU times for training single SVR and single BPN models are 0.0998 s and 0.1221, respectively. Thus, it can be found that the ELM algorithm is more efficient than BPN and SVR algorithms.

4.3. Forecasting Results of NB and DC

In this section, we forecast the sales of NB and DC. The modeling process described in Section 4.2 was applied. After carefully selection, for forecasting NB, the best numbers of hidden nodes for the EEMD-ELM, EMD-ELM, and single ELM models are, respectively, 5, 8, and 12. The EEMD-ELM, EMD-ELM and single ELM models with, respectively, 3, 7, and 4 hidden nodes are the best models and used for forecasting sales of DC. The parameter sets of the single SVR models for NB and DC are ($C = 2^9, \epsilon = 2^{-1}, \sigma = 2^3$) and ($C = 2^7, \epsilon = 2^1, \sigma = 2^{-3}$), respectively. The parameter sets (hidden nodes = 5, learning = 0.10) and (hidden nodes = 6, learning = 0.05) are, respectively, the best parameter sets of the single BPN models for forecasting NB and DC sales.

Table 3: The forecasting results of HD using the single ELM, single SVR, single BPN, EMD-ELM, and the proposed EEMD-ELM models.

Computer products	Models	RMSE	MAPE	MAD	Training time (s)
HD	Single ELM	207.47	15.21%	98.44	0.0373
	Single SVR	211.52	15.40%	103.14	0.0998
	Single BPN	213.87	16.03%	105.87	0.1221
	EMD-ELM	173.87	12.22%	92.22	0.0701
	EEMD-ELM	132.94	9.21%	84.11	0.0779

Table 4: The forecasting results of NB and DC using the single ELM, single SVR, single BPN, EMD-ELM, and the proposed EEMD-ELM models.

Computer products	Models	RMSE	MAPE	MAD	Training time (s)
NB	Single ELM	76.87	3.84%	31.21	0.0382
	Single SVR	77.02	3.91%	31.31	0.0903
	Single BPN	78.11	3.95%	32.05	0.1103
	EMD-ELM	62.09	5.37%	40.45	0.0713
	EEMD-ELM	56.33	3.53%	28.01	0.0801
DC	Single ELM	22.94	5.53%	16.31	0.0365
	Single SVR	23.02	5.59%	16.65	0.0872
	Single BPN	22.96	5.54%	16.31	0.0981
	EMD-ELM	22.38	4.53%	15.88	0.0695
	EEMD-ELM	11.99	4.39%	9.22	0.0797

Table 4 shows the forecasting results of NB and DC using EEMD-ELM, EMD-ELM, and single ELM models. It can be seen from Table 4 that the proposed EEMD-ELM method also performs well in forecasting sales of NB and SC. From Table 4, it can be found that the proposed EEMD-ELM model has the smallest RMSE, MAPE, and MAD values in comparison with single ELM, single SVR, single BPN, and EMD-ELM models for NB and DC products. From Table 4, it also can be seen that training time of ELM algorithm is shorter than that of BPN and SVR algorithms.

Based on the findings in Tables 3 and 4, it can be observed that the proposed EEMD-ELM method outperforms the single ELM, single SVR, single BPN and EMD-ELM models under all three computer products. It indicates that the proposed EEMD-ELM approach indeed provides better forecasting accuracy than the other four comparison approaches. Therefore, it can be concluded that the EEMD is a promising tool for extracting hidden/interesting features from computer products, and the proposed EEMD-ELM model can be a good alternative for forecasting sales of computer products. Moreover, it can also be concluded from Tables 3 and 4 that the training speed of ELM algorithm is faster than BPN and SVR algorithms in all three datasets, which verifies the conclusion made in the literature [10].

5. Conclusions

Sales forecasting is a crucial aspect of marketing and inventory management in managing computer products. This paper proposes a hybrid sales forecasting scheme by integrating EEMD and ELM for computer products. The proposed EEMD-ELM method first uses EEMD

to extract features, that is, IMFs, from the original sales data. The IMFs containing hidden information are then used in ELM for constructing the prediction model. This study compares the proposed method with EMD-ELM and single ELM models by using the sales data of three computer products including HD, NB, and DC. Experimental results showed that the proposed model can produce lower prediction error and outperforms the single ELM, single SVR, single BPN, and EMD-ELM models under all three computer products. According to the experiments, it can be concluded that the EEMD can effectively extract underlying information from original sales data and improve the forecasting performance of ELM. The proposed EEMD-ELM model can be a good alternative for forecasting computer products sales.

Acknowledgment

This research was partially supported by the National Science Council of the Republic of China under Grant Nos. NSC 101-2221-E-231-006- and NSC 99-2221-E-030-014-MY3.

References

- [1] C. J. Lu and Y. W. Wang, "Combining independent component analysis and growing hierarchical self-organizing maps with support vector regression in product demand forecasting," *International Journal of Production Economics*, vol. 128, no. 2, pp. 603–613, 2010.
- [2] S. Thomassey and M. Happiette, "A neural clustering and classification system for sales forecasting of new apparel items," *Applied Soft Computing Journal*, vol. 7, no. 4, pp. 1177–1187, 2007.
- [3] Z. L. Sun, T. M. Choi, K. F. Au, and Y. Yu, "Sales forecasting using extreme learning machine with applications in fashion retailing," *Decision Support Systems*, vol. 46, no. 1, pp. 411–419, 2008.
- [4] W. K. Wong and Z. X. Guo, "A hybrid intelligent model for medium-term sales forecasting in fashion retail supply chains using extreme learning machine and harmony search algorithm," *International Journal of Production Economics*, vol. 128, no. 2, pp. 614–624, 2010.
- [5] P. C. Chang, C. H. Liu, and C. Y. Fan, "Data clustering and fuzzy neural network for sales forecasting: a case study in printed circuit board industry," *Knowledge-Based Systems*, vol. 22, no. 5, pp. 344–355, 2009.
- [6] C. J. Lu, T. S. Lee, and C. M. Lian, "Sales forecasting for computer wholesalers: a comparison of multivariate adaptive regression splines and artificial neural networks," *Decision Support Systems*, vol. 54, no. 1, pp. 584–596, 2012.
- [7] R. Fildes, K. Nikolopoulos, S. F. Crone, and A. A. Syntetos, "Forecasting and operational research: a review," *Journal of the Operational Research Society*, vol. 59, no. 9, pp. 1150–1172, 2008.
- [8] S. Thomassey and A. Fiordaliso, "A hybrid sales forecasting system based on clustering and decision trees," *Decision Support Systems*, vol. 42, no. 1, pp. 408–421, 2006.
- [9] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks," in *Proceedings of the IEEE International Joint Conference on Neural Networks*, pp. 985–990, Budapest, Hungary, July 2004.
- [10] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006.
- [11] G. B. Huang and L. Chen, "Convex incremental extreme learning machine," *Neurocomputing*, vol. 70, no. 16–18, pp. 3056–3062, 2007.
- [12] G. B. Huang and L. Chen, "Enhanced random search based incremental extreme learning machine," *Neurocomputing*, vol. 71, no. 16–18, pp. 3460–3468, 2008.
- [13] G. B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 42, pp. 513–529, 2012.
- [14] F. L. Chen and T. Y. Ou, "Sales forecasting system based on Gray extreme learning machine with Taguchi method in retail industry," *Expert Systems with Applications*, vol. 38, no. 3, pp. 1336–1345, 2011.

- [15] M. Xia, Y. Zhang, L. Weng, and X. Ye, "Fashion retailing forecasting based on extreme learning machine with adaptive metrics of inputs," *Knowledge-Based Systems*, vol. 36, pp. 253–259, 2012.
- [16] C. J. Lu, T. S. Lee, and C. C. Chiu, "Financial time series forecasting using independent component analysis and support vector regression," *Decision Support Systems*, vol. 47, no. 2, pp. 115–125, 2009.
- [17] K. L. Chen, C. C. Yeh, and T. L. Lu, "A hybrid demand forecasting model based on empirical mode decomposition and neural network in TFT-LCD industry," *Cybernetics and Systems*, vol. 43, no. 5, pp. 426–441, 2012.
- [18] H. Liu and J. Wang, "Integrating independent component analysis and principal component analysis with neural network to predict Chinese Stock Market," *Mathematical Problems in Engineering*, vol. 2011, Article ID 382659, 15 pages, 2011.
- [19] C. J. Lu, "Integrating independent component analysis-based denoising scheme with neural network for stock price prediction," *Expert Systems with Applications*, vol. 37, no. 10, pp. 7056–7064, 2010.
- [20] N. E. Huang, Z. Shen, S. R. Long et al., "The empirical mode decomposition and the Hubert spectrum for nonlinear and non-stationary time series analysis," *Proceedings of the Royal Society A*, vol. 454, no. 1971, pp. 903–995, 1998.
- [21] N. E. Huang, M. L. Wu, W. Qu, S. R. Long, and S. S. P. Shen, "Applications of Hilbert-Huang transform to non-stationary financial time series analysis," *Applied Stochastic Models in Business and Industry*, vol. 19, no. 3, pp. 245–268, 2003.
- [22] Z. Guo, W. Zhao, H. Lu, and J. Wang, "Multi-step forecasting for wind speed using a modified EMD-based artificial neural network model," *Renewable Energy*, vol. 37, no. 1, pp. 241–249, 2012.
- [23] Y. Wei and M. C. Chen, "Forecasting the short-term metro passenger flow with empirical mode decomposition and neural networks," *Transportation Research Part C*, vol. 21, no. 1, pp. 148–162, 2012.
- [24] Z. H. Wu and N. E. Huang, "Ensemble empirical mode decomposition: a noise assisted data analysis method," *Advances in Adaptive Data Analysis*, vol. 1, no. 1, pp. 1–41, 2009.
- [25] L. Tang, L. Yu, S. Wang, J. Li, and S. Wang, "A novel hybrid ensemble learning paradigm for nuclear energy consumption forecasting," *Applied Energy*, vol. 93, pp. 432–443, 2012.
- [26] R. Zhang, Y. Bao, and J. Zhang, "Forecasting erratic demand by support vector machines with ensemble empirical mode decomposition," in *Proceedings of the 3rd International Conference on Information Sciences and Interaction Sciences (ICIS '10)*, pp. 567–571, June 2010.
- [27] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, NY, USA, 2000.
- [28] G. Zhang, B. Eddy Patuwo, and M. Y. Hu, "Forecasting with artificial neural networks: the state of the art," *International Journal of Forecasting*, vol. 14, no. 1, pp. 35–62, 1998.
- [29] A. Vellido, P. J. G. Lisboa, and J. Vaughan, "Neural networks in business: a survey of applications (1992–1998)," *Expert Systems with Applications*, vol. 17, no. 1, pp. 51–70, 1999.
- [30] V. Cherkassky and Y. Ma, "Practical selection of SVM parameters and noise estimation for SVM regression," *Neural Networks*, vol. 17, no. 1, pp. 113–126, 2004.
- [31] C. W. Hsu, C. C. Chang, and C. J. Lin, "A practical guide to support vector classification," Tech. Rep., Department of Computer Science and Information Engineering, National Taiwan University, 2012.
- [32] S. Haykin, *Neural Network: A Comprehensive Foundation*, Prentice Hall, New Jersey, NJ, USA, 1999.
- [33] Y. Chauvin and D. E. Rumelhart, *Backpropagation: Theory, Architectures, and Applications*, Lawrence Erlbaum Associates, New Jersey, NJ, USA, 1995.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

