Physica A 477 (2017) 161–173

Contents lists available at ScienceDirect

Physica A

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

Multidimensional *k*-nearest neighbor model based on EEMD for financial time series forecasting

Ningning Zhang, Aijing Lin *, Pengjian Shang

School of Science, Beijing Jiaotong University, Beijing 100044, PR China

HIGHLIGHTS

- A new multidimensional *k*-nearest neighbor model based on EEMD method is proposed.
- The new model can be used to forecast the closing price and high price of the four stocks at the same time.
- The experiments show EEMD–MKNN method has high accuracy than other methods in the stock markets forecasting.

ARTICLE INFO

Article history:
Received 9 December 2016
Received in revised form 30 January 2017
Available online 28 February 2017

Keywords:
Ensemble empirical mode decomposition (EEMD)
k-nearest neighbors (KNN)
EEMD–MKNN
Forecasting
Closing price
High price

ABSTRACT

In this paper, we propose a new two-stage methodology that combines the ensemble empirical mode decomposition (EEMD) with multidimensional *k*-nearest neighbor model (MKNN) in order to forecast the closing price and high price of the stocks simultaneously. The modified algorithm of *k*-nearest neighbors (KNN) has an increasingly wide application in the prediction of all fields. Empirical mode decomposition (EMD) decomposes a nonlinear and non-stationary signal into a series of intrinsic mode functions (IMFs), however, it cannot reveal characteristic information of the signal with much accuracy as a result of mode mixing. So ensemble empirical mode decomposition (EEMD), an improved method of EMD, is presented to resolve the weaknesses of EMD by adding white noise to the original data. With EEMD, the components with true physical meaning can be extracted from the time series. Utilizing the advantage of EEMD and MKNN, the new proposed ensemble empirical mode decomposition combined with multidimensional *k*-nearest neighbor model (EEMD–MKNN) has high predictive precision for short-term forecasting. Moreover, we extend this methodology to the case of two-dimensions to forecast the closing price and high price of the four stocks (NAS, S&P500, DJI and STI stock indices) at the same time. The results indicate that the proposed EEMD–MKNN model has a higher forecast precision than EMD–KNN, KNN method and ARIMA.

1. Introduction

Financial time series forecasting has been always a hot research field, yet the non-stationary financial time series makes it challenging. Financial time series is intrinsically non-stationary, noisy and chaotic as stated in [1,2]. This means that the distribution of financial time series is changing over the time. As such, massive efforts have been devoted to improvement of financial time series forecasting. A variety of methods and techniques have been proposed to predict stock market, such as ARIMA, GARCH [3,4], Markov Model [5], SVM [6], Grey Model [7] and Neural Network [8–11], etc. It is undeniable that those

* Corresponding author.
E-mail address: ajlin@bjtu.edu.cn (A. Lin).

http://dx.doi.org/10.1016/j.physa.2017.02.072
0378-4371/

methods have obtained better forecasting results. However, those methods take as input large amounts of numeric time series data to find a model extrapolating the financial markets into the future. Besides, based on the empirical results, it is clear that nonlinear models outperform linear models [12–14]. For example, one of the major limitations of the traditional methods such as ARIMA is that they are essentially linear methods. In order to use them, users must specify the model form without the necessary knowledge about the complex relationship in the data. Therefore, for problems where the underlying data generating process can be well appropriated by linear models, we could consider linear models. However, if the linear models fail to perform well in both in-sample fitting and out-of-sample forecasting, more complex nonlinear models should be considered.

Of all the nonparametric approaches, the *k*-nearest neighbors (KNN) method has proved to be a potential method and widely applied in various forecasting [15–23]. In this approach, K refers to the number of nearest neighbors (NN) in sample space, where distance between data points are qualified by root mean square error (RMSE). Thus the nearest neighbors are data points with the small RMSE and exhibit high similarity. This method has the advantage of the ability of predicting high-dimension and incomplete data. Accordingly, KNN method is suitable for forecasting stock markets. While the great performance of KNN method, a lot of negative problems, such as lower forecasting accuracy for complicated applications [24,25], etc., are emerging. To address these issues, we propose a new method which combines ensemble empirical mode decomposition (EEMD) with MKNN to predict stock index, and it proved to be performing well in forecasting financial time series.

Empirical mode decomposition (EMD) [26], as a new technique for analysis of nonlinear and non-stationary time series, has been proposed and widely applied in various areas recently [27–39]. It decomposes a complicated signal into a collection of intrinsic mode functions (IMFs) and a residue, and those IMFs and the residue reveal the particular characteristic. However, the mode mixing, 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, is one of the major problem of EMD. To resolve the problem of mode mixing in EMD, ensemble empirical mode decomposition (EEMD), an improved method of EMD, is presented by Wu and Huang [40] recently. EEMD is a noise-assisted data analysis approach by adding white noise to the original data, as well as it is effective to alleviate the mode mixing problem.

In the literature, there is a hybrid model, namely EMD–KNN method developed by Lin [41], is a novel algorithm that combines the advantages of KNN and EMD, and has been already successfully used in other fields [42]. The empirical results demonstrate EMD–KNN method outperforms other technologies such as KNN and ARIMA. However, the major drawback of EMD algorithm is mode mixing, and thus this paper provides a new hybrid model which combines EEMD and MKNN, and our proposed model is verified through experimental study in which EMD–KNN, KNN and ARIMA methods are compared and EEMD–MKNN method is applied in predicting closing price and high price of four stock indices (NAS, S&P 500, DJI and STI). The results of the experiments show that EEMD–MKNN method has a higher forecasting accuracy than EMD–KNN, KNN and ARIMA model.

The remainder of this paper is organized as follows: Section 2 introduces the methodologies of the EMD, EEMD, KNN and EEMD–MKNN. Section 3 describes the data of four stock indices used in this article and detailed experimental analysis are given and the results compares the results of ARIMA, KNN, EMD–KNN and the new proposed method EEMD–MKNN by forecasting the daily closing prices and high prices of four stock indices (NAS, S&P 500, DJI and STI). Section 4 gives the conclusions and future studies.

2. Methodology

2.1. Empirical mode decomposition (EMD)

This section we first review empirical mode decomposition (EMD), which was proposed by Huang et al. and deals with nonlinear and non-stationary data by decomposing the signal. The complicated original signal is decomposed into a series of intrinsic mode functions (IMFs) and a residue by EMD. The decomposition should follow the assumptions: (1) the data have more than, or equal to two extrema; (2) the characteristic time scale is defined by the time lapse between the extrema; (3) providing that the data are totally devoid of extrema but contain only inflection points, it can be differentiated once or more times to reveal the extrema. Final results could be gained by integrations of the components. And the IMFs meet the following two conditions: (1) in the whole data set, the number of zero crossings must equal or differ at most by one; (2) at any point, the mean value of the envelope defined by the local maxima and the envelope defined by the local minima is zero. With the above mentioned hypotheses and conditions, any signals can be decomposed into IMF components and a residue.

In the EMD approach, the data $X(t)$ are decomposed into IMFs, c_j,

$$X(t) = \sum_{j=1}^{n} c_j + r_n \tag{1}$$

where r_n is the residue of data $X(t)$.

Therefore, we can obtain a decomposition of the signal into IMFs c_j and a residue r_n, which is the mean trend of $X(t)$. The IMFs $c_1, c_2, \ldots, c_n$ include different frequency bands ranging from high to low.

Based on this simple description of EMD, Flandrin et al. [43] and Wu and Huang [44] have shown that, supposing that the data consisted of white noise that has scales populated uniformly through the whole timescale or time–frequency space,

the EMD behaves as a dyadic filter bank: the Fourier spectra of different IMFs collapse to a single shape along the axis of logarithm of period or frequency. And the total number of IMFs of the data set is close to $\log_2 N$ with N the total number of data points. If the data is not pure noise, some scales might be missing; therefore, the total number of the IMFs could be fewer than $\log_2 N$. Additionally, there is a serious mode mixing due to the intermittency of signals in some scales.

Mode mixing appears to be the most significant problem of EMD, which indicates either a single IMF consisting of signals of dramatically disparate scales or a signal of the same scale appearing in different IMF components. It is usually the result of signal intermittency. As discussed by Huang et al. [26], the intermittence could not only cause serious aliasing in the time–frequency distribution, but also make physical meaning of individual IMF unclear. Once the mode mixing phenomenon had happened, an IMF can cease to have physical meaning by itself, suggesting wrongly that there may be different physical processes represented in a mode.

2.2. Ensemble empirical mode decomposition (EEMD)

To overcome the problem of mode mixing in EMD, Flandrin et al. [45] proposed a new noise-assisted data analysis method—EEMD, which defines the true IMFs as the mean of an ensemble of trials. Each trial is composed of the decomposition results of the signal added a white noise [43,44] of finite amplitude [40]. The EEMD algorithm first generates an ensemble of data sets gotten by adding different white noise to the original signal. Then, deal with these new data sets by EMD method. Shortly, for a series $x(t)$, the EEMD algorithm includes the following steps:

Step 1: Generate a new series $y(t)$ by adding white noise into original series $x(t)$.
Step 2: Identify all the local maxima and minima of time series $y(t)$.
Step 3: Generate the upper envelopes $e_u(t)$ and lower envelopes $e_l(t)$ of $y(t)$.
Step 4: Calculate the mean $m(t)$ from the upper and lower envelope.

$$m(t) = \frac{e_u(t) + e_l(t)}{2}. \tag{2}$$

Step 5: Extract the difference between the data and $m(t)$ as the first component $h(t)$:

$$y(t) - m(t) = h(t). \tag{3}$$

Step 6: The sifting process has to be iterated more times. We can repeat this iterative procedure k times, until $h(t)$ is an IMF, and that is the first IMF component c_1.

$$y(t) - c_1 = r_1. \tag{4}$$

Step 7: The residue r_1 is treated as the new series, and repeat Steps 2–6 to gain all r_j, a residue c_n.

By summing up all IMFs and the residue, we finally obtain:

$$y(t) = \sum_{j=1}^{n} c_j + r_n. \tag{5}$$

Fig. 1(a)–(m) shows the original series, all the IMFs and the residue time series of a stock series decomposed by EEMD method.

2.3. MKNN prediction algorithm

The original KNN only considers the Euclidean distance, and then use the nearest values to predict future values, but ignore the changes of pattern vector of the time series. The improved KNN not only considers the Euclidean distance, but also considers the similarity of pattern vectors. Thus the improved KNN has more accurate forecasting results than original KNN. And we extend it to multidimensional KNN algorithm. KNN algorithm is a relatively mature theory algorithm, and a nonparametric method that was used for statistical pattern recognition [46–50]. The key of KNN algorithm is to find the most nearest K samples from the unknown samples and the sample set. Euclidean distance is usually used to determine the distance of the database. In the recent study, the KNN algorithm is constantly being used to predict stock indices. Based on that, this paper develops the MKNN method.

The following presents a simple example of the two dimensional KNN model as an example. Consider two time series $\{X = x_1, x_2, \ldots, x_n\}$ and $\{Y = y_1, y_2, \ldots, y_n\}$, where n is the number of points of the series, and x_n with y_n denote the current state. Firstly, we find out the nearest group, also called the nearest neighbors, of the current state x_n and y_n. Then, we predict x_{n+1} and y_{n+1} on the basis of these nearest values; for instance, if the size of neighborhood is $k = 1$, and the nearest values are x_j and y_j, then we would predict x_{n+1} and y_{n+1} on the basis of x_{j+1} and y_{j+1}.

Taking the two time series $\{X = x_1, x_2, \ldots, x_n\}$ and $\{Y = y_1, y_2, \ldots, y_n\}$, the definition of the difference matrix of the series can be extended to include several consecutive values $Q = \begin{pmatrix} q^x_{n-l} & q^x_{n-l+1} & \cdots & q^x_n \\ q^y_{n-l} & q^x_{n-l+1} & \cdots & q^y_n \end{pmatrix}$ where l $(1 \le l \le n-1)$ is a pattern size, $q^x_i = x_{i+1} - x_i$, $n - l \le i \le n - 1$ and $q^y_j = y_{j+1} - y_j$, $n - l \le j \le n - 1$. We map these differences onto

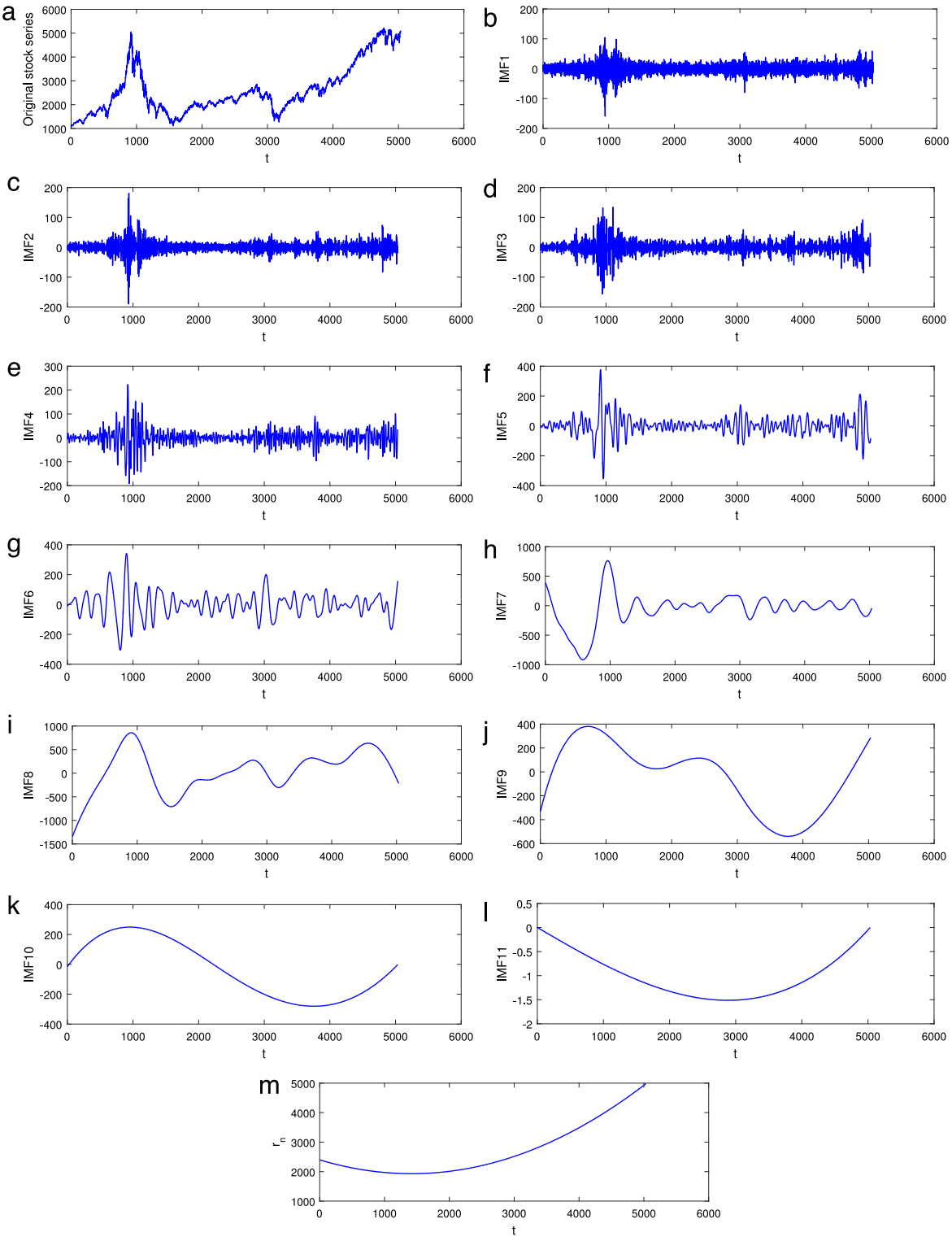

Fig. 1. Ensemble empirical mode decomposition (EEMD) of a stock time series. Fig. 1(a) the original stock time series are plotted versus time t. Fig. 1(b)–(l) Intrinsic mode functions (IMF) for a stock time series are plotted versus time t, Fig. 1(m) residue r_n ($n = 11$) is plotted as a function of time. Each IMF captures fluctuations of original time series occurred on distinct time scales—higher order IMF correspond to larger characteristic time scale.

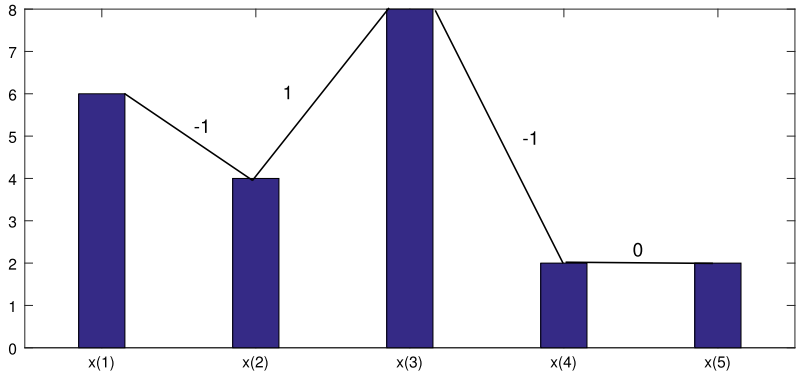

Fig. 2. An example of a pattern vector obtained from the MKNN algorithm. The state vector is [6, 4, 8, 2, 2], corresponding pattern vector is [−1, 1, −1, 0], with the pattern size $l = 4$. The MKNN prediction algorithm involves searching all the pattern vector to find the nearest matches and predict the next value based on those values of the nearest matches.

trinary variable d_i to define the direction by encoding each of them as a −1, 0 or 1 where $d_i = -1$ if $q_i < 0$, 1 if $q_i > 0$ and 0 if $q_i = 0$. Hence, a pattern in time series can be represented as $P_d = \begin{pmatrix} d^x_{n-l} & \cdots & d^x_{n-1} \\ d^y_{n-l} & \cdots & d^y_{n-1} \end{pmatrix}$, a vector of −1, 0, 1. Supposing $X = [x_{n-4}, x_{n-3}, x_{n-2}, x_{n-1}, x_n] = [6, 4, 8, 2, 2]$ and $Y = [y_{n-4}, y_{n-3}, y_{n-2}, y_{n-1}, y_n] = [9, 2, 5, 5, 1]$ are two state vectors with pattern size $l = 4$. So taking X as a example, the difference vector is $P^x_d = [-1, 1, -1, 0]$, as shown in Fig. 2. And the difference matrix is $P_d = \begin{pmatrix} -1 & 1 & -1 & 0 \\ -1 & 1 & 0 & -1 \end{pmatrix}$. The pattern size for matching has a vital effect on minimizing the error and accurately predicting the direction of series change. Accordingly, it is important to optimize the size of pattern in order to obtain the best results.

In short, the procedure of the algorithm is shown as follows:

Step 1: Start from a minimal neighborhood size k.
Step 2: Start from a minimal pattern size l.
Step 3: Form the pattern of size l describing the current state, i.e.,

$$P_d = \begin{pmatrix} d^x_{n-l} & \cdots & d^x_{n-1} \\ d^y_{n-l} & \cdots & d^y_{n-1} \end{pmatrix} \tag{6}$$

$(d^x_{n-l}, \ldots, d^x_{n-1})$ and $(d^y_{n-l}, \ldots, d^y_{n-1})$ respectively denote the current states of $\{X = x_1, x_2, \ldots, x_n\}$ and $\{Y = y_1, y_2, \ldots, y_n\}$.

Step 4: Search the time series $\begin{pmatrix} d^x_1 & \cdots & d^x_{n} \\ d^y_1 & \cdots & d^y_n \end{pmatrix}$ to find the nearest matches by Euclidean distance, and sort them in ascending order and choose the first Ns matches, and each nearest match corresponds to an index j. For which the matching pattern is $P'_d = \begin{pmatrix} d^x_{j-l} & \cdots & d^x_{j-1} \\ d^y_{j-l} & \cdots & d^y_{j-1} \end{pmatrix}$, the difference matrix associated with P'_d is $Q'_d = \begin{pmatrix} q^x_{j-l} & \cdots & q^x_{j-1} \\ q^y_{j-l} & \cdots & q^y_{j-1} \end{pmatrix}$, and the final differences associated with match number h are $\hat{Q}^h_j$ (the time series X)and $\tilde{Q}^h_j$ (the time series Y).

Step 5: Estimate the value $\hat{x}_{n+1}$ and $\hat{y}_{n+1}$ based on the final differences for all nearest neighbors:

$$\begin{aligned} \hat{x}_{n+1} &= x_n + \hat{Q}_m \quad \text{where } \hat{Q}_m = \sum_{h=1}^{k} \frac{\hat{Q}^h_j}{k} \\ \hat{y}_{n+1} &= y_n + \tilde{Q}_m \quad \text{where } \tilde{Q}_m = \sum_{h=1}^{k} \frac{\tilde{Q}^h_j}{k}. \end{aligned} \tag{7}$$

Step 6: Calculate the root mean squared error (RMSE) between the actual and predicted values for selecting appropriate neighborhood size k and pattern size l for the whole set:

$$\begin{aligned} RMSE_x &= \sqrt{\frac{1}{N}\sum_{i=1}^{N}[x(i) - \hat{x}(i)]^2} \\ RMSE_y &= \sqrt{\frac{1}{N}\sum_{j=1}^{N}[y(j) - \hat{y}(j)]^2}. \end{aligned} \tag{8}$$

Step 7: Repeat Steps 3–6 for pattern sizes $l + 1, l + 2, \ldots, l$ max.

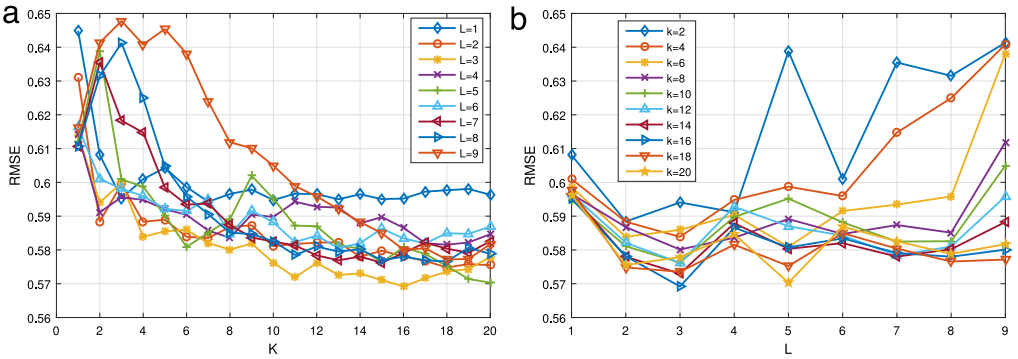

Fig. 3. Identifying the optimal k and l for the EEMD–MKNN algorithm. (a) Root mean square error (RMSE) for models with different L are plotted versus the number of nearest neighbor k. (b) RMSE for models with different k are plotted versus the pattern size l. Fig. 3 just shows the changes of RMSE in the range of $1 \leq k \leq 20$, $1 \leq l \leq 9$ without $k > 20$ and $l > 9$, but this does not affect the result.

Step 8: Repeat Steps 2–7 for neighborhood sizes $k+1, k+2, \ldots, k$ max.
Step 9: Choose the optimal pattern recognition model which yields minimal RMSE by optimizing the neighborhood size k and pattern size l.

The selection of appropriate k and l is very important. Fig. 3 shows RMSE performance of different neighborhood size k and pattern size l. RMSE are computed using KNN model for values of k from 1 to 20, and pattern sizes l from 1 to 9. With the gradual increments of neighborhood size k and l start from 1, it will improve the precision of forecasting. While k and l arrive at a certain value, the increment of k and l will reduce the accuracy due to choosing neighbors that is less relative to the current state vector to generate prediction.

2.4. EEMD–MKNN

We discuss the details of the EEMD and MKNN algorithm in the previous part. As mentioned in the introduction, there are some limitations in the original KNN algorithm. Therefore, the hybrid method, called EEMD–MKNN, is developed to take the advantages of EEMD and MKNN. The EEMD–MKNN process is as follows: First, the given time series $F(t)$ is decomposed into limited number of IMFs, and a residue r_n by EEMD method:

$$F(t) = \sum_{i=1}^{n} F_i + r_n. \tag{9}$$

Then, calculate the forecasting values $P_i (i = 1, 2, \ldots, n+1)$ of $F_i (i = 1, 2, \ldots, n)$ and r_n separately using the MKNN algorithm. The final predict value P is computed by:

$$P = \sum_{i=1}^{n+1} P_i. \tag{10}$$

3. Analysis and results

3.1. Data

Most of the articles about financial time series prediction are forecasting closing price of the stocks. But according to Dow Jones theory, the highest price of the stocks is also important. Therefore, we predict closing price and high price of the stocks simultaneously. In an effort to illustrate the effectiveness of the EEMD–MKNN method for real-world dynamics, we consider the daily closing price and high price of NAS, S&P 500, DJI and STI over a period of 20 years, from July 25, 1996 to July 25, 2016. To avoid predicted number impact on predictive accuracy, we predict the last 100, 120, 150 and 200 stock trading days of NAS, S&P 500, DJI and STI stock indices based on the past data. The data are downloaded from the website http://finance.yahoo.com/.

3.2. Predicting results comparisons

In this section, we predict the closing price and high price of four stock indices (NAS, S&P 500, DJI and STI) using EEMD–MKNN method. Technical analysis of the closing price and the highest price, the two price represents the day of the price fluctuations in the 2 extremes: the time of an extreme (closing price) and the price of an extreme (the highest price).

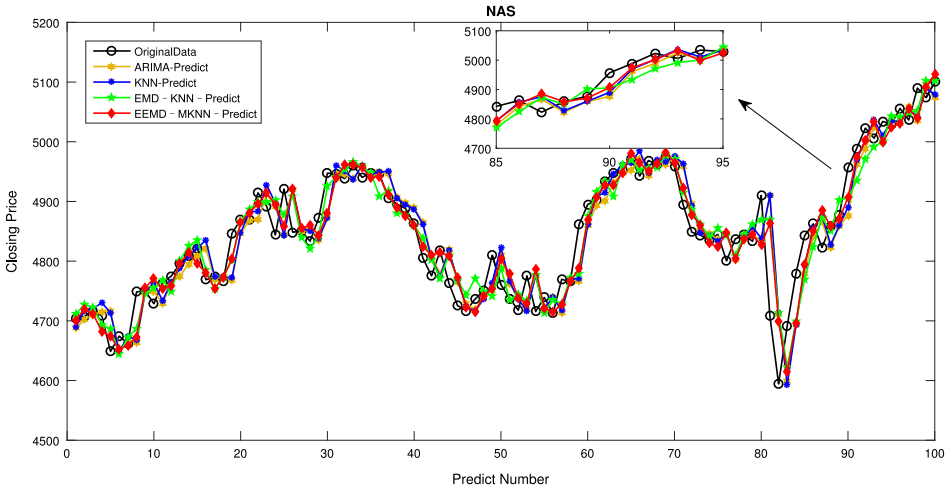

Fig. 4. Comparison of the predicted results of closing price of NAS using ARIMA, KNN, EMD–KNN and EEMD–MKNN methods. The closing price of NAS and different predicted values by using those four methods are plotted versus predict number of the last 100 stock trading days of NAS. In order to identify the differences of predicted values with those four methods, we magnify the part of predicted results of predict number 10–20 stock trading days of NAS. The results indicate the predicted values by EEMD–MKNN are closer to actual values than those by other three methods.

Table 1
Comparison of NMSE, MASE and MAPE values for closing price of NAS by the four methods. The results demonstrate that the EEMD–MKNN method provides a higher forecasting accuracy in contrast with the EMD–KNN, KNN and ARIMA method.

	MAPE	MASE	NMSE
EEMD–MKNN	0.5497	0.7964	0.3474
EMD–KNN	0.5673	0.8214	0.3477
KNN	0.6715	0.9726	0.4158
ARIMA	0.6902	1.0008	0.4190

Closing price is the basis of making money or losing money. To show the good performance of EEMD–MKNN model, we compare it with three other models: the EMD–KNN model, KNN model, autoregressive integrated moving average (ARIMA) model. With the optimal value of k and l, the predicted results were attained in Figs. 4–11, which present real value and predictive value of the four models. In contrast with three other methods, the EEMD–MKNN method relatively performs well according to the predicting results.

To evaluate the forecasting performance of different models, three error measures are used in the experiment: mean absolute percentage Error (MAPE), mean absolute scaled error (MASE) and root mean squared error (NMSE).

$$MAPE = mean\left(\left|\frac{100(x_t - \hat{x}_t)}{x_t}\right|\right) \tag{11}$$

$$MASE = mean\left(\left|\frac{x_t - \hat{x}_t}{\frac{1}{n-1}\sum_{i=2}^{n}|x_i - x_{i-1}|}\right|\right) \tag{12}$$

$$NMSE = \sqrt{\frac{\sum_{t=1}^{n} e_t{}^2}{\sum_{t=1}^{n}(x_t - mean(x))^2}} \tag{13}$$

where x_t denotes the actual value of the time series $X(t)$ at time t and $\hat{x}_t$ denotes the predicted value of x_t. Tables 1–8 shows the results of comparison of NMSE, MASE and MAPE values for closing price and high price of NAS, S&P 500, DJI and STI by ARIMA, KNN, EMD–KNN and EEMD–MKNN methods. And those confirm the series predicted by EEMD–MKNN method are closer to actual values than the three other methods. Additionally, multidimensional KNN model considers two factors (closing price and high price of stock) simultaneously, so it performs better than ARIMA model in predicting financial time series.

Another way of comparing the forecasting power is to use correlation coefficient for measuring similarity between actual values and predicted values. Tables 9 and 10 present the results of the comparisons between the actual values and predicted

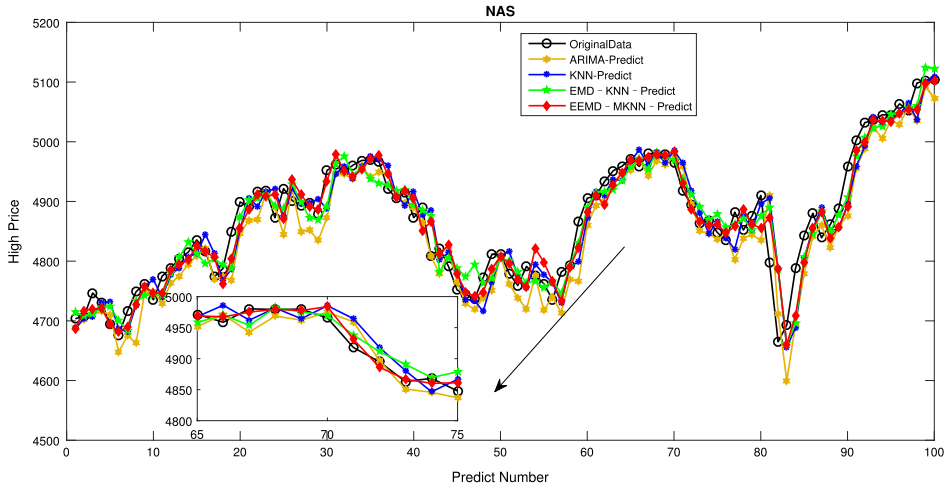

Fig. 5. Comparison of the predicted results of high price of NAS using ARIMA, KNN, EMD–KNN and EEMD–MKNN methods. The high price of NAS and different predicted values by using those four methods are plotted versus predict number of the last 100 stock trading days of NAS. In order to identify the differences of predicted values with those four methods, we magnify the part of predicted results of predict number 65–75 stock trading days of NAS. The results indicate the predicted values by EEMD–MKNN are closer to actual values than those by other three methods.

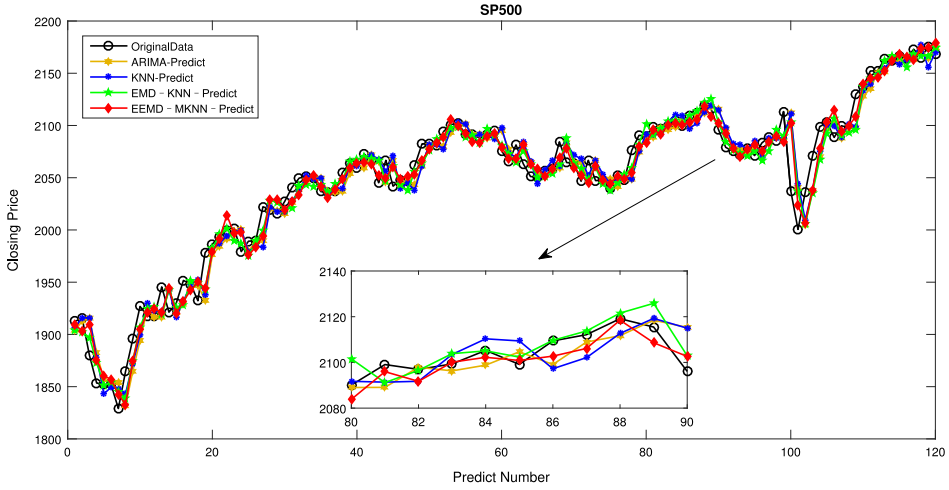

Fig. 6. Comparison of the predicted results of closing price of S&P 500 using ARIMA, KNN, EMD–KNN and EEMD–MKNN methods. The closing price of S&P 500 and different predicted values by using those four methods are plotted versus predict number of the last 100 stock trading days of S&P 500. In order to identify the differences of predicted values with those four methods, we magnify the part of predicted results of predict number 10–20 stock trading days of S&P 500. The results indicate the predicted values by EEMD–MKNN are closer to actual values than those by other three methods.

Table 2
Comparison of NMSE, MASE and MAPE values for high price of NAS by the four methods. The results demonstrate that the EEMD–MKNN method provides a higher forecasting accuracy in contrast with the EMD–KNN, KNN and ARIMA method.

Methods	MAPE	MASE	NMSE
EEMD–MKNN	0.4468	0.7978	0.2883
EMD–KNN	0.4893	0.8741	0.3013
KNN	0.5578	0.9969	0.3465
ARIMA	0.6519	1.1683	0.3960

values of NAS, S&P 500, DJI and STI using EEMD–MKNN, EMD–KNN, KNN and ARIMA models. It is clear that the bigger the values of correlation coefficient between actual values and predicted values, the higher is forecast precision. The results prove again that the new proposed EEMD–MKNN method is more successful than the three other methods in predicting either closing price or high price for the four stock indices.

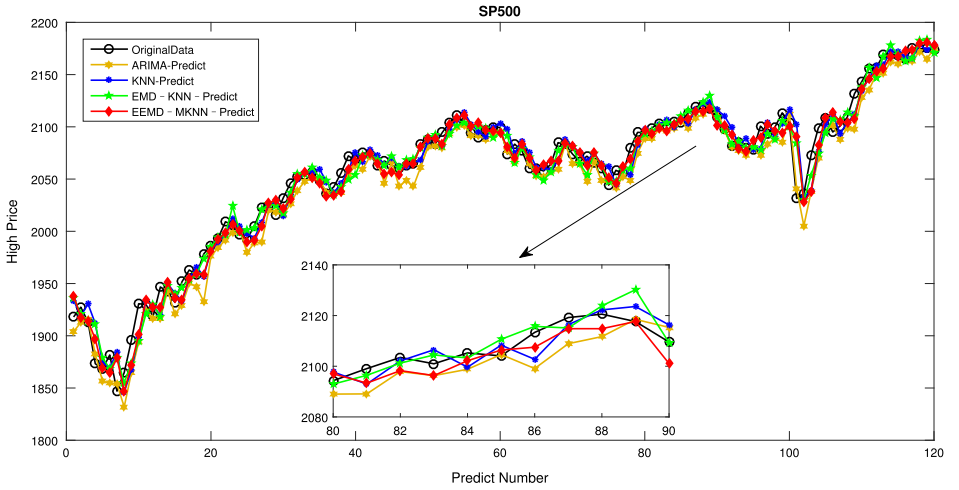

Fig. 7. Comparison of the predicted results of high price of S&P 500 using ARIMA, KNN, EMD–KNN and EEMD–MKNN methods. The high price of S&P 500 and different predicted values by using those four methods are plotted versus predict number of the last 100 stock trading days of S&P 500. In order to identify the differences of predicted values with those four methods, we magnify the part of predicted results of predict number 65–75 stock trading days of S&P 500. The results indicate the predicted values by EEMD–MKNN are closer to actual values than those by other three methods.

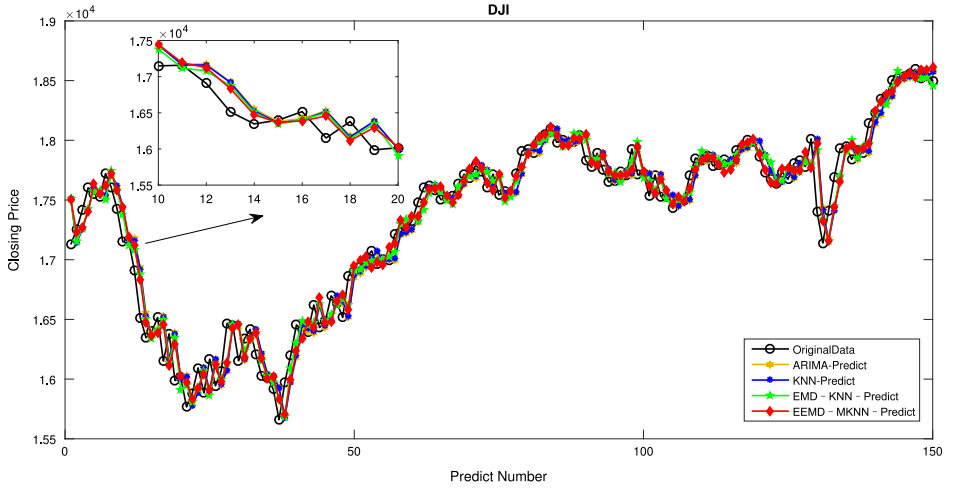

Fig. 8. Comparison of the predicted results of closing price of DJI using ARIMA, KNN, EMD–KNN and EEMD–MKNN methods. The closing price of DJI and different predicted values by using those four methods are plotted versus predict number of the last 100 stock trading days of DJI. In order to identify the differences of predicted values with those four methods, we magnify the part of predicted results of predict number 10–20 stock trading days of DJI. The results indicate the predicted values by EEMD–MKNN are closer to actual values than those by other three methods.

Table 3
Comparison of NMSE, MASE and MAPE values for closing price of S&P 500 by the four methods. The results demonstrate that the EEMD–MKNN method provides a higher forecasting accuracy in contrast with the EMD–KNN, KNN and ARIMA method.

	MAPE	MASE	NMSE
EEMD–MKNN	0.4997	0.8178	0.1861
EMD–KNN	0.5121	0.8407	0.1920
KNN	0.5955	0.9783	0.2194
ARIMA	0.6134	1.0064	0.2269

4. Conclusions

The EEMD–MKNN method, which is combined EEMD method with MKNN model, is proposed to forecast financial time series in this paper, and it is extended to two dimensions to predict the closing price and high price of the stocks simultaneously. Experiments on the data of four stock indices (NAS, S&P 500, DJI and STI) demonstrate the validity of the method. Furthermore, three error measures (NMSE, MASE and MAPE) and correlation coefficient are used to evaluate

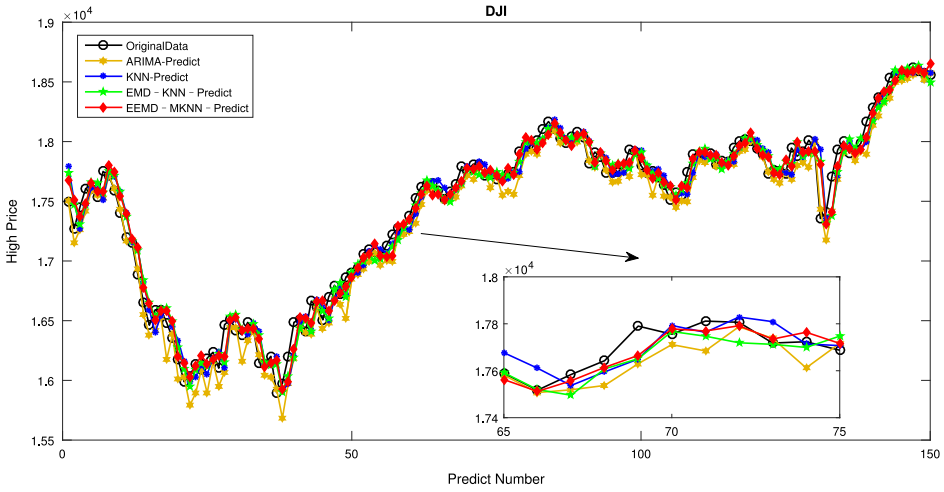

Fig. 9. Comparison of the predicted results of high price of DJI using ARIMA, KNN, EMD–KNN and EEMD–MKNN methods. The high price of DJI and different predicted values by using those four methods are plotted versus predict number of the last 100 stock trading days of DJI. In order to identify the differences of predicted values with those four methods, we magnify the part of predicted results of predict number 65–75 stock trading days of DJI. The results indicate the predicted values by EEMD–MKNN are closer to actual values than those by other three methods.

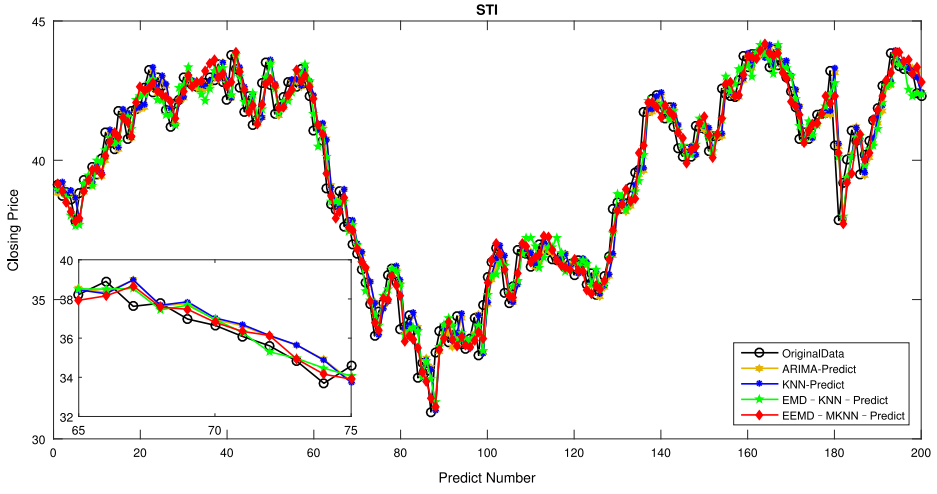

Fig. 10. Comparison of the predicted results of closing price of STI using ARIMA, KNN, EMD–KNN and EEMD–MKNN methods. The closing price of STI and different predicted values by using those four methods are plotted versus predict number of the last 100 stock trading days of STI. In order to identify the differences of predicted values with those four methods, we magnify the part of predicted results of predict number 10–20 stock trading days of STI. The results indicate the predicted values by EEMD–MKNN are closer to actual values than those by other three methods.

Table 4
Comparison of NMSE, MASE and MAPE values for high price of S&P 500 by the four methods. The results demonstrate that the EEMD–MKNN method provides a higher forecasting accuracy in contrast with the EMD–KNN, KNN and ARIMA method.

Methods	MAPE	MASE	NMSE
EEMD–MKNN	0.4201	0.8336	0.1634
EMD–KNN	0.4171	0.8273	0.1649
KNN	0.5009	0.9929	0.1907
ARIMA	0.5732	1.1433	0.2079

the forecasting performance of different models. The comparison among EEMD–MKNN method and EMD–KNN method, KNN method, ARIMA model shows EEMD–MKNN method has higher accuracy than the three other methods in the stock markets forecasting. On the whole, the results of experiments suggest EEMD–MKNN method is robust for predicting financial time series. For future work, the new proposed method can be applied to forecast other time series. Besides that, future improvement could be focusing on predicting the IMFs and residue by different approaches to get better forecast effect and higher accuracy.

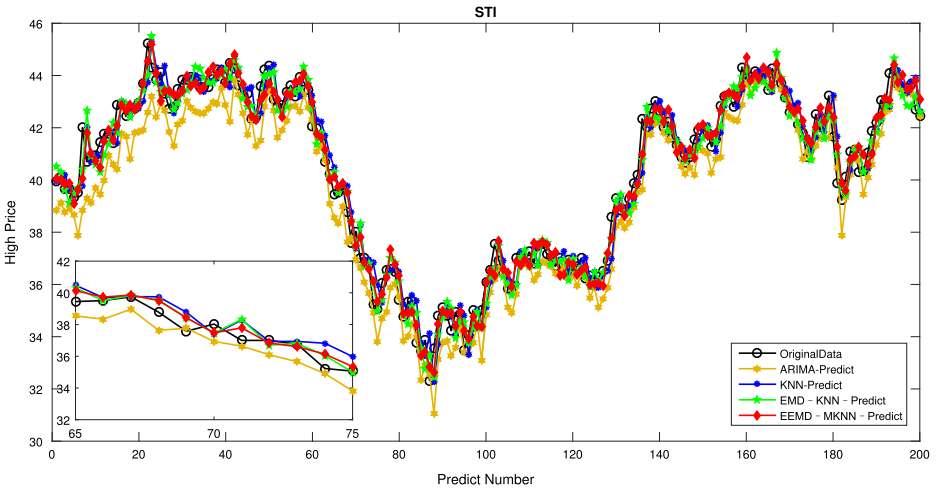

Fig. 11. Comparison of the predicted results of high price of STI using ARIMA, KNN, EMD–KNN and EEMD–MKNN methods. The high price of STI and different predicted values by using those four methods are plotted versus predict number of the last 100 stock trading days of STI. In order to identify the differences of predicted values with those four methods, we magnify the part of predicted results of predict number 65–75 stock trading days of STI. The results indicate the predicted values by EEMD–MKNN are closer to actual values than those by other three methods.

Table 5
Comparison of NMSE, MASE and MAPE values for closing price of DJI by the four methods. The results demonstrate that the EEMD–MKNN method provides a higher forecasting accuracy in contrast with the EMD–KNN, KNN and ARIMA method.

	MAPE	MASE	NMSE
EEMD–MKNN	0.6054	0.8646	0.1948
EMD–KNN	0.6053	0.8652	0.1914
KNN	0.7020	1.0033	0.2202
ARIMA	0.7065	1.0107	0.2211

Table 6
Comparison of NMSE, MASE and MAPE values for high price of DJI by the four methods. The results demonstrate that the EEMD–MKNN method provides a higher forecasting accuracy in contrast with the EMD–KNN, KNN and ARIMA method.

Methods	MAPE	MASE	NMSE
EEMD–MKNN	0.4400	0.8309	0.1495
EMD–KNN	0.4744	0.8952	0.1592
KNN	0.5246	0.9894	0.1792
ARIMA	0.5759	1.0883	0.1914

Table 7
Comparison of NMSE, MASE and MAPE values for closing price of STI by the four methods. The results demonstrate that the EEMD–MKNN method provides a higher forecasting accuracy in contrast with the EMD–KNN, KNN and ARIMA method.

	MAPE	MASE	NMSE
EEMD–MKNN	1.1928	0.8088	0.1812
EMD–KNN	1.3101	0.8847	0.1962
KNN	1.4884	1.0019	0.2288
ARIMA	1.4891	1.0032	0.2282

Table 8
Comparison of NMSE, MASE and MAPE values for high price of STI by the four methods. The results demonstrate that the EEMD–MKNN method provides a higher forecasting accuracy in contrast with the EMD–KNN, KNN and ARIMA method.

Methods	MAPE	MASE	NMSE
EEMD–MKNN	0.9807	0.7437	0.1545
EMD–KNN	1.1858	0.8956	0.1789
KNN	1.3374	1.0049	0.2098
ARIMA	2.1513	1.6414	0.3186

Table 9
Comparison of correlation coefficient values for closing price of NAS, S&P 500, DJI and STI by ARIMA, KNN, EMD–KNN and EEMD–MKNN method. The results demonstrate that the EEMD–MKNN method provides a higher forecasting accuracy in contrast with the EMD–KNN, KNN and ARIMA method.

Stocks	ARIMA	KNN	EMD–KNN	EEMD–MKNN
NAS	0.9109	0.9128	0.9380	0.9390
S&P 500	0.9748	0.9761	0.9820	0.9829
DJI	0.9753	0.9756	0.9816	0.9810
STI	0.9739	0.9739	0.9806	0.9841

Table 10
Comparison of correlation coefficient values for high price of NAS, S&P 500, DJI and STI by ARIMA, KNN, EMD–KNN and EEMD–MKNN method. The results demonstrate that the EEMD–MKNN method provides a higher forecasting accuracy in contrast with the EMD–KNN, KNN and ARIMA method.

Stocks	ARIMA	KNN	EMD–KNN	EEMD–MKNN
NAS	0.9409	0.9491	0.9543	0.9587
S&P 500	0.9819	0.9890	0.9865	0.9872
DJI	0.9840	0.9911	0.9873	0.9888
STI	0.9780	0.9797	0.9841	0.9882

Acknowledgments

We acknowledge support from the National Natural Science Foundation of China (61673005, 61304145 and 61371130), the Research Fund for the Doctoral Program of Higher Education (20130009120016).

References

[1] G. Deboeck, Trading on the Edge: Neural, Genetic, and Fuzzy Systems for Chaotic Financial Markets, Vol. 39, John Wiley & Sons, 1994.
[2] N.R. Swanson, H. White, Forecasting economic time series using flexible versus fixed specification and linear versus nonlinear econometric models, Int. J. Forecast. 13 (4) (1997) 439–461.
[3] R.G. Donaldson, M. Kamstra, An artificial neural network garch model for international stock return volatility, J. Empir. Finance 4 (1) (1997) 17–46.
[4] P.H. Franses, D. Van Dijk, Forecasting stock market volatility using (nonlinear) garch models, J. Forecast. (1996) 229–235.
[5] F. Hao, The applications of markov prediction method in stock market, Friends Sci. 6 (1) (2006) 78–81.
[6] L. Cao, F.E. Tay, Financial forecasting using support vector machines, Neural Comput. Appl. 10 (2) (2001) 184–192.
[7] Y. Wang, Predicting stock price using fuzzy grey prediction system, Expert Syst. Appl. 22 (1) (2002) 33–38.
[8] F. Castiglione, Forecasting price increments using an artificial neural network, Adv. Complex Syst. 4 (01) (2001) 45–56.
[9] A.S. Chen, M.T. Leung, H. Daouk, Application of neural networks to an emerging financial market: forecasting and trading the Taiwan stock index, Comput. Oper. Res. 30 (6) (2003) 901–923.
[10] G. Grudnitski, L. Osburn, Forecasting s&p and gold futures prices: An application of neural networks, J. Futures Mark. 13 (6) (1993) 631–643.
[11] X. Zhu, H. Wang, L. Xu, H. Li, Predicting stock index increments by neural networks: The role of trading volume under different horizons, Expert Syst. Appl. 34 (4) (2008) 3043–3054.
[12] G.P. Zhang, Time series forecasting using a hybrid arima and neural network model, Neurocomputing 50 (2003) 159–175.
[13] J.H. Stock, M.W. Watson, A comparison of linear and nonlinear univariate models for forecasting macroeconomic time series, Tech. rep., National Bureau of Economic Research, 1998.
[14] C.-W. Chu, G.P. Zhang, A comparative study of linear and nonlinear models for aggregate retail sales forecasting, Int. J. Prod. Econ. 86 (3) (2003) 217–231.
[15] G.A. Davis, N.L. Nihan, Nonparametric regression and short-term freeway traffic forecasting, J. Transp. Eng.-ASCE 117 (2) (1991) 178–188.
[16] I. Turkoglu, E.D. Kaymaz, A hybrid method based on artificial immune system and *k*-nn algorithm for better prediction of protein cellular localization sites, Appl. Soft Comput. 9 (2) (2009) 497–502.
[17] R. Mehrotra, A. Sharma, Conditional resampling of hydrologic time series using multiple predictor variables: A *k*-nearest neighbour approach, Water Resour. Res. 29 (7) (2006) 987–999.
[18] M. Bannayan, G. Hoogenboom, Weather analogue: a tool for real-time prediction of daily weather data realizations based on a modified *k*-nearest neighbor approach, Environ. Modell. Softw. 23 (6) (2008) 703–713.
[19] R.E. McRoberts, Estimating forest attribute parameters for small areas using nearest neighbors techniques, Forest. Ecol. Manag. 272 (2012) 3–12.
[20] R.E. McRoberts, E.O. Tomppo, Remote sensing support for national forest inventories, Remote Sens. Environ. 110 (4) (2007) 412–419.
[21] A.T. Hudak, N.L. Crookston, J.S. Evans, D.E. Hall, M.J. Falkowski, Nearest neighbor imputation of species level, plot-scale forest structure attributes from lidar data, Remote Sens. Environ. 112 (5) (2008) 2232–2245.
[22] A. Nothdurft, J. Saborowski, J. Breidenbach, Spatial prediction of forest stand variables, J. Forest Res.-Jpn. 128 (3) (2009) 241–251.
[23] P. Packalén, M. Maltamo, Predicting the plot volume by tree species using airborne laser scanning and aerial photographs, For. Sci. 52 (6) (2006) 611–622.
[24] S. Magnussen, E. Tomppo, R.E. McRoberts, A model assisted *k*-nearest neighbour approach to remove extrapolation bias, Scand. J. For. Res. 25 (2) (2010) 174–184.
[25] R.E. McRoberts, Diagnostic tools for nearest neighbors techniques when used with satellite imagery, Remote Sens. Environ. 113 (3) (2009) 489–499.
[26] N.E. Huang, Z. Shen, S.R. Long, M.C. Wu, H.H. Shih, Q. Zheng, N.C. Yen, C.C. Tung, H.H. Liu, The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis, in: Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, Vol. 454, The Royal Society, 1998, pp. 903–995.
[27] X. Zhang, K.K. Lai, S.Y. Wang, A new approach for crude oil price analysis based on empirical mode decomposition, Energy Econ. 30 (3) (2008) 905–918.
[28] S. Loutridis, Damage detection in gear systems using empirical mode decomposition, Eng. Struct. 26 (12) (2004) 1833–1841.
[29] K. Guhathakurta, I. Mukherjee, A.R. Chowdhury, Empirical mode decomposition analysis of two different financial time series and their comparison, Chaos Solitons Fractals 37 (4) (2008) 1214–1227.

[30] J.R. Yeh, S.-Z. Fan, J.S. Shieh, Human heart beat analysis using a modified algorithm of detrended fluctuation analysis based on empirical mode decomposition, Med. Eng. Phys. 31 (1) (2009) 92–100.
[31] C.H. Loh, T.C. Wu, N.E. Huang, Application of the empirical mode decomposition-Hilbert spectrum method to identify near fault ground-motion characteristics and structural responses, Bull. Seismol. Soc. Amer. 91 (5) (2001) 1339–1357.
[32] Z.K. Gao, N.D. Jin, Scaling analysis of phase fluctuations in experimental three phase flows, Physica A 390 (20) (2011) 3541–3550.
[33] B.M. Battista, C. Knapp, T. McGee, V. Goebel, Application of the empirical mode decomposition and Hilbert-Huang transform to seismic reflection data, Geophysics 72 (2) (2007) H29–H37.
[34] T. Wang, M. Zhang, Q. Yu, H. Zhang, Comparing the applications of emd and eemd on time–frequency analysis of seismic signal, J. Appl. Geophys. 83 (2012) 29–34.
[35] K. Coughlin, K.K. Tung, 11-year solar cycle in the stratosphere extracted by the empirical mode decomposition method, Adv. Space Res. 34 (2) (2004) 323–329.
[36] R.B. Pachori, V. Bajaj, Analysis of normal and epileptic seizure eeg signals using empirical mode decomposition, Comput. Methods Programs Biomed. 104 (3) (2011) 373–381.
[37] E.S. Neto, M. Custaud, J. Cejka, P. Abry, J. Frutoso, D. Gharib, P. Flandrin, et al., Assessment of cardiovascular autonomic control by the empirical mode decomposition, Methods Inf. Med. 43 (1) (2004) 60–65.
[38] H. Liang, Z. Lin, R. McCallum, Artifact reduction in electrogastrogram based on empirical mode decomposition method, Med. Biol. Eng. Comput. 38 (1) (2000) 35–41.
[39] J. Echeverria, J. Crowe, M. Woolfson, B. Hayes Gill, Application of empirical mode decomposition to heart rate variability analysis, Med. Biol. Eng. Comput. 39 (4) (2001) 471–479.
[40] Z. Wu, N.E. Huang, Ensemble empirical mode decomposition: a noise assisted data analysis method, Adv. Adapt. Data Anal. 1 (01) (2009) 1–41.
[41] A. Lin, P. Shang, G. Feng, B. Zhong, Application of empirical mode decomposition combined with *k*-nearest neighbors approach in financial time series forecasting, Fluct. Noise Lett. 11 (02) (2012) 1250018.
[42] Y. Ren, P. Suganthan, Empirical mode decomposition-k nearest neighbor models for wind speed forecasting, J. Power Energy Eng. 2 (04) (2014) 176.
[43] P. Flandrin, G. Rilling, P. Goncalves, Empirical mode decomposition as a filter bank, IEEE Signal Proc. Lett. 11 (2) (2004) 112–114.
[44] Z. Wu, N.E. Huang, A study of the characteristics of white noise using the empirical mode decomposition method, in: Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, Vol. 460, The Royal Society, 2004, pp. 1597–1611.
[45] P. Flandrin, P. Gonçalves, G. Rilling, Emd equivalent filter banks, from interpretation to applications, World Sci. (2005) 57–74.
[46] T. Cover, P. Hart, Nearest neighbor pattern classification, IEEE Trans. Inform. Theory 13 (1) (1967) 21–27.
[47] T. Brandsma, T.A. Buishand, Simulation of extreme precipitation in the rhine basin by nearest neighbour resampling, Hydrol. Earth Syst. Sci. 2 (2/3) (1998) 195–209.
[48] S. Gangopadhyay, M. Clark, B. Rajagopalan, Statistical downscaling using *k*-nearest neighbors, Water Resour. Res. 41 (2005).
[49] N. Meade, A comparison of the accuracy of short term foreign exchange forecasting methods, Int. J. Forecast. 18 (1) (2002) 67–83.
[50] F. Fernandez-Rodrıguez, S. Sosvilla-Rivero, J. Andrada-Felix, Exchange rate forecasts with simultaneous nearest neighbour methods: Evidence from the ems, Int. J. Forecast. 15 (4) (1999) 383–392.