Original Article

IJEEE

International Journal of Electrical Engineering & Education
0(0) 1–14
© The Author(s) 2021
Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/0020720920984675
journals.sagepub.com/home/ije
SAGE

Extreme learning machine for stock price prediction

Fangzhao Zhang

Abstract

Stock market performance prediction has always been a hit research topic and is attractive due to its strong potential to generate financial profit. Being able to predict future stock price in a relatively accurate way forms a significant task of stock market analysis. Different mechanisms from fundamental analysis to statistical modeling have been deployed to study stock market performance and various factors from fundamental factors, technical factors to market sentiments are also incorporated in the stock price prediction task. However, due to the chaotic stock market performance, which is close to random walk, and the difficulty in discerning influential factors, predicting stock price faces a lot of challenges. In recent years, fast development in fields such as machine learning has offered new ways to look at this task. In this paper, we employ Extreme Learning Machine (ELM) algorithm, a recent modification of traditional feed-forward neural network with single hidden layer, whose learning speed is greatly improved based on solid mathematical background and capability to circumvent problems such as local minimum is also enhanced, to construct an ELM combination model to study stock market performance and predict stock price. A comparison between the predicted output and the real data is carried out to test the feasibility of applying ELM model to stock market analysis. The result indicates that ELM model is desirable for predicting stock price variation trend while some inaccuracy exists in the prediction of peak values, which may require further model modification. Overall, by applying the machine learning model ELM to predict stock price and generating desirable outcome, this paper both contributes to offering a new way to investigate stock market performance and enlarging the field deployment of ELM model as well.

Math & CS Ddepartment, University of British Columbia, Vancouver, Canada

Corresponding author:
Fangzhao Zhang, Math & CS Department, University of British Columbia, Vancouver, BC V6T 1Z4, Canada.
Email: fangzhao.zh@gmail.com

Keywords
Extreme learning machine, stock price prediction, artificial neural network, features generation

Introduction

Due to the significant potential profit space and the quickness of gaining interest of the stock exchange market, it is unsurprisingly tempting for people to look for ways to accurately forecast the trend of stock price variation and investment strategies to maximize one's earnings from stock market. Being able to foretell the performance of future stock market not only does good to single investors by generating abnormal returns out of their initial capitols, but also benefits corporates by sending appropriate indications for future business adjustments. Further, accurate stock market forecasting also assists government to look into upcoming economic situation and become better prepared for any possible economic recessions. With respect to all these advantages, huge number of researches have gone into figuring out distinguishable subtle patterns of stock market performance which backs up stock price forecasting and coming up with specific investment strategies which are likely to generate above-average returns. Whereas the task faced by researchers is never easy due to the stock market's chaotic nature and the numerous variants with complex interacting relationships between each other that are responsible for stock market performance hidden behind, (indeed, the random-walk time series behavior of stock market and inefficiency of both fundamental and technical analysis for stock market performance analysis give a rough taste of how intractable the task might be), appreciable observations and conclusions do come up thanks to vast and continuous researches dedicated to this field.

Starting with fundamental and technical analysis techniques for the study of stock market performance, the behavior of which is witnessed to be no better than random walk, neural network, which was introduced as an approach to model financial and economic time series, becomes widely studied later on. With the swift development of fields including machine learning and statistical analysis, various analyzing approaches become available and are extensively adapted to financial analysis including study of stock market performance and price trend prediction. Recently, method of artificial neural network (ANN) appears as an intriguing analysis approach due to its ability to unravel hidden patterns behind seemingly unrelated data attributed to its affinity to nonparametric, nonlinear regression models, which fits the analysis of stock market well given that one of the most significant problems faced by stock market analysis is to deal with the tremendous parameter set. However, the imperfection of ANN still exists in its potential adoption of local minimum. Exposed to this problem, technique of support vector machine (SVM), which computes globally maximum instead, seems to make up the discrepancy while other problems appear including its intractable parameter choice and inefficiency with increasing data set. A lot of current research focuses on further developing this technology by either subtly modify its implementation details or hybridizing it with other methods. For example, Yongjun, Li, et al.[1] manage to improve traditional support vector machine's performance for increment learning by decomposing its sample space with different weight attached, Guo, Z., et al.[2] exploit support vector machine optimized by PSO to financial time series forecasting, and Honghai, Y. and Haifei, L.[3] attempt to predict stock market assisted by a combination of support vector machine and empirical model decomposition.

In Chapter 2, we present explanation and details about the origination and process of extreme learning machine algorithm, which will be used to analyze stock market performance later. In Chapter 3, we include some general description about stock market performance and a list of significant indicators which are commonly used to characterize stock prices. We also explain how to integrate stock market analysis with ELM by exploiting such indicators in Chapter 3. In Chapter 4, we set up ELM algorithm to actually analyze a set of stock price data and display the process and result of running the algorithm. Chapter 5 concludes our research of applying ELM to stock market prediction with some further research-worthy topics proposed.

Extreme learning machine

With the development of researches in fields of statistics and data mining, a lot of techniques and models used for data classification, pattern analysis and trend prediction have been proposed and tested these years. One famous algorithm among those is the artificial neural network, which was first introduced by mathematician Pitts and psychologist McCulloch[4] as neurons mathematical model (MP model) by investigating the structure of real biological neural networks. After then, though there might be troughs and peaks, researchers have never stopped the way to exploit and extend this model for further applications.

Nowadays, as an extension and modification of the simple MP model, which was formerly attacked for its low-level approximation ability, researches on feedforward neural networks with single hidden layer (SLFN) have drawn a lot of attention since the time it was proposed due to its strong nonlinear mapping ability. The continuous mapping ability of SLNF is verified step by step by a lot of recent works including Hornik's[5] and Leshno's.[6] In later literature published in 1998, Huang and Babri[7] showed that a N-hidden-node SLFN acquired the ability to learn N distinct samples with nearly any nonlinear activation functions. Till this point, all parameters in SLFN model including input weights and biases in all layers needed to be tuned. However, as has been proved by Tamura and Tateishi[8] in more recent literature, SLFNs with N randomly generated sigmoidal hidden nodes can exactly learn N distinct samples. Here, randomly generated nodes stand for nodes with randomly picked input weights and hidden biases. In Extreme learning machine: Theory and applications, Huang et al.[9] further proved that as long as the activation function was infinitely differentiable, the mapping ability of SLNF is not influenced by whether or not the input weights and all biases have been tuned. That is, there may be an opportunity to save training time of SLFN by ignoring the training process of parameters including input weights and hidden biases.

In real applications, although SLFN has been recognized as a useful tool in a lot of fields from artificial intelligence to data mining, shortcomings such as the influence of local minimum, overfitting problems and low training speed of traditional SLFN training methods which mainly rely on gradient descent for parameter tuning have limited the usage of SLFN. To further address these problems and enhance the generalization ability of the model, one training technique for SLFN called extreme learning machine (ELM) has been proposed by Huang et al.[9] that avoids the potential influence of local minimum and raises the training speed to hundreds of times faster than previous gradient-descent based training methods.

Given N arbitrary distinct samples $(x_i,\ t_i)$ where $x_i = [x_{i1}, \ldots, x_{in}]^T \in \boldsymbol{R}^n$ and $t_i = [t_{i1}, \ldots, t_{im}]^T \in \boldsymbol{R}^m$, the standard SLFN model can be mathematically represented by ($\sim N$ denotes the number of hidden nodes and $g(x)$ denotes the activation function)

$$\sum_{i=1}^{\sim N} \beta_i g_i(x_j) = \sum_{i=1}^{\sim N} \beta_i g(\omega_i \cdot x_j + b_i) = \mathrm{o}_j, \ \ (j = 1, \ldots, N) \tag{1}$$

where $\omega_i = [\omega_{i1}, \ldots, \omega_{in}]^T$ is the weight vector connecting the ith hidden neuron and the input neuron, $\beta_i = [\beta_{i1}, \ldots, \beta_{im}]^T$ is the weight vector connecting the ith hidden neuron and the output neurons, and b_i is the bias of the ith hidden neuron. Here $\omega_i \cdot x_j$ represents the inner product of ω_i and x_j. The activation function $g(x)$ can be chosen as Sigmoid function, Sine function, RBF function etc. The output neurons are chosen as linear here.

When the approximation has no error, then o_j is just t_j and the above equation can be written in a more compact way as follows:

$$H\beta = T \tag{2}$$

where

$$H(\omega_i, \ldots, \omega_{\sim N}, b_i, \ldots, b_{\sim N}, x_i, \ldots, x_N) = \begin{bmatrix} g(\omega_1 \cdot x_1 + b_1) & \cdots & g(\omega_{\sim N} \cdot x_1 + b_{\sim N}) \\ \vdots & \ddots & \vdots \\ g(\omega_1 \cdot x_N + b_1) & \cdots & g(\omega_{\sim N} \cdot x_N + b_{\sim N}) \end{bmatrix}_{N \times \sim N} \tag{3}$$

$$\beta = \begin{bmatrix} {\beta_1}^T \\ \vdots \\ {\beta_{\sim N}}^T \end{bmatrix}_{\sim N \times m} \quad \text{and} \quad T = \begin{bmatrix} {t_1}^T \\ \vdots \\ {t_N}^T \end{bmatrix}_{N \times m} \tag{4}$$

H is called the hidden layer output matrix and the ith column of H is the ith hidden neuron's output with respect to inputs $x_i, \ldots, x_N$.

Unlike the traditional gradient descent-based training algorithm, the speed of which has been significantly lowered due to the multiple iterations involved and the requirement for training all parameters including the input weights ω_i as well as all hidden biases b_i $(i = 1, \ldots, \sim N)$ and the result of which is also exposed to influence of local minimum problem inherent in gradient descent-based computation, the ELM training algorithm offers a much faster way of training SLFN with better generalization performance.

Huang et al. have shown that once the number of hidden nodes $\sim N$ has been specified, there exists no necessity to tune the input weights ω_i and biases b_i $(i = 1, \ldots, \sim N)$ in training of SLFN as long as the activation function is infinitely differentiable. That is, randomly generated input weights and biases do no harm to the mapping ability of SLFN. According to these observations, the traditional gradient descent-based training algorithm can be replaced by ELM, a SLFN training algorithm that is way faster than the traditional ones with user-specified number of hidden nodes and randomly generated input weights and biases. The whole training process requires only one iteration and once after the number of hidden nodes is given and the input weights and biases have been randomly assigned, the hidden layer output matrix H is fixed, thus the training process of SLFN is converted to solve $H\beta = T$ least squares solution $\hat{\beta}$ with fixed H and T

$$\| H(\omega_1, \ldots, \omega_{\sim N}, b_1, \ldots, b_{\sim N})\hat{\beta} - T \| = \min_{\beta} \| H(\omega_1, \ldots, \omega_{\sim N}, b_1, \ldots, b_{\sim N})\beta - T \| \tag{5}$$

The solution to (5) has the form of $\hat{\beta} = H^{\dagger}\mathrm{T}$ where $H^{\dagger}$ denotes the Moore-Penrose generalized inverse of matrix H.

The whole training process of the algorithm can thus be concluded into the following three steps:

Given a training set $= \{(x_i,\ t_i) x_i \in \boldsymbol{R}^n,\ t_i \in \boldsymbol{R}^m, i = 1, \ldots, N\}$, activation function $g(x)$ and hidden node number $\sim N$,

Step 1: Randomly assign input weights ω_i and bias b_i, $i = 1, \ldots, \sim N$.
Step 2: Calculate the hidden layer output matrix H.
Step 3: Calculate the output weight $\beta = H^{\dagger}\mathrm{T}$ where $T = [t_1, \ldots, t_N]^T$.

With these unique advantages, ELM is of tremendous usage. In real application, for example, Mao[10] combined ELM with signal processing to achieve higher processing speed, Mohammed[11] exploited ELM for a new human face recognition algorithm with better recognition accuracy and Chang et al.[12] applied ELM to build up a new soft sensing model. Some improved algorithms based on ELM have also been proposed in recent literature, including but not limited to an online sequential fuzzy extreme learning machine (OS-Fuzzy-ELM) proposed by Rong et al.[13] in 2009 and an error minimized extreme learning machine (EM-ELM) proposed by Romero and Alquezar[14] in 2012.

The application of ELM for stock price prediction

The features generation of stock price data

Predicting stock price can be a tricky task due to the multiple volatile potential influence factors that might be involved, which make stock price prediction become complex and multi-dimensional. In fact, it is not even clear whether it is possible or not to predict the trend of stock market at the very beginning. By the Random Walk Theo rem proposed by Horne and Parker[15] in 1967, the future stock price is considered to be independent from historical data, which indicates the random change of stock price and the impossibility to predict the behavior of stock market behavior. In 1970, Fama[16] further suggests the Efficient Market Hypothesis (EMH), claiming that the stock price depends on market information and traders' profit of investing stock market is in exact synchronization with the risk they take. By classifying the market price into three forms from weak form, semi-strong form to strong form by taking into consideration the information involved, the author claims that there is no way to successfully predict the stock market behavior.

However, based on the assumption that stock price does indeed follow some predictable patterns, different techniques, including statistical methods and machine learning methods, are applied to financial time series analysis.[17] The prediction task mainly falls into two fields, regression task and classification task namely. Depending on the data type included in analysis, stock price prediction can be classified as either technical analysis, which focuses only on the time series problem involving historical prices and trends in stock market and is better at short-term price prediction of turning points, or as fundamental analysis, which concerns also internal/external factors of companies such as interest rate and product innovation and is better at predicting the trend of stock market movement.[18]

To predict future stock price based on historical stock prices, we can first apply time series analysis and represent the desired stock price as a time series of length N, which, as validated by Token's theorem, can be considered as a function of historical stock prices $P_0, P_1, P_2, \cdots, P_{N-1}$ where P_i denotes the close price on day i $(0 \leq i < N)$. However, one flawed point in this representation may lie in the limitation of choosing historical stock prices at each single time, which has little representative ability to map the stock market performance during given time stamp. Thus, to resolve the problem and use historical data sources in a more reliable way, we preprocess historical stock prices by splitting those tick data using a fixed window with length L, then the data representing the ith window is taken by considering stock prices $P_{iL}, \cdots, P_{(i+1)L-1}$. Specifically, we will extract the following indicators in further stock price prediction:

1. Maximum: we take the maximum for the ith window as $\mathrm{Max}(P_{iL}, \cdots, P_{(i+1)L-1})$.
2. Minimum: we take the minimum for the ith window as $\mathrm{Min}(P_{iL}, \cdots, P_{(i+1)L-1})$.

3. Average: we take the average value for the ith window as $Sum(P_{iL}, \ldots, P_{(i+1)L-1})/L$.
4. Standard Deviation (SD): we take the standard deviation for the ith window as $\sqrt{\frac{1}{L-1}\sum_{j=0}^{L-1}(P_{iL+j}-\bar{P})^2}$ where $\bar{P}$ is the mean value of $P_{iL}, \ldots, P_{(i+1)L-1}$.
5. Pseudo Log Return (PLR): the logarithmic difference between average prices of consecutive windows.
6. Trend Indicator (TI): we take the trend indicator for the ith window as a linear model applied on $P_{iL}, \ldots, P_{(i+1)L-1}$ to produce a linear equation with certain slope where a negative slope suggests a decrease in the stock price, a positive slope suggests an increase in the stock price and a zero slope suggests an invariant stock price.

The stock price prediction with ELM

Structure of ELM. As an algorithm for training single hidden-layer feedforward neural network, Extreme Learning Machine achieves high training speed and good generalization ability by simplifying the training process of single hidden-layer feedforward neural network to a large extent compared to the traditional BP algorithm. As already explained above, based on rigid mathematical deduction, training of single hidden-layer feedforward neural network requires only the number of hidden nodes being given, and the input weights $[w_{11}, \ldots, w_{nh}]$ and biases $[b_1, \ldots, b_h,\ b_o]$ can then be randomly assigned without influencing the final approximation ability of the neural network. In this way, the whole training process concerns only about the output weights $[w_{1o}, \ldots, w_{ho}]$, which saves the time used for training all other parameters. Further, with this observation, the task of training single hidden-layer feedforward neural network is reduced to find a least square solution of a derived equation, indicating that the whole training process can be completed in a single iteration, enhancing the training speed significantly by getting rid of numerous iterations necessary in traditional back propagation method and thus characterizing the superior behavior of ELM.

Integrate stock price prediction with ELM combination model. To use ELM to predict stock price, we construct an ELM model as described above with hidden nodes. We take indicators mentioned in 3.1 as input layer, the Integrate stock price prediction with ELM structure can be shown as Figure 1 and Figure 2.

In this paper we proposed a combined prediction model for the stock price prediction, which is to make full use of the model advantages provided by various single forecasting methods and improving the accuracy of prediction results. It is suitable for small sample data and has good prediction accuracy for fitting historical data of exponential law but low prediction accuracy for medium and long term especially in stock price sequence. Extreme Learning Machine (ELM) has strong ability to process non-linear information and high accuracy

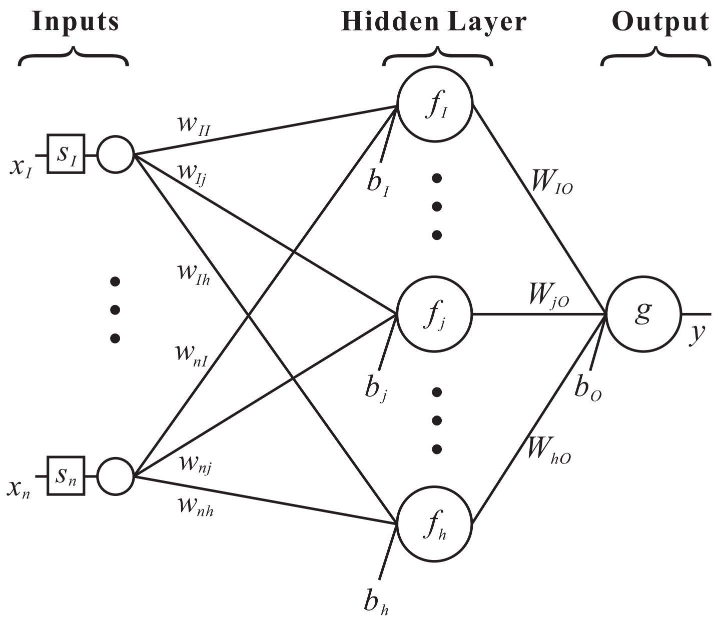

Figure 1. Structure of ELM.

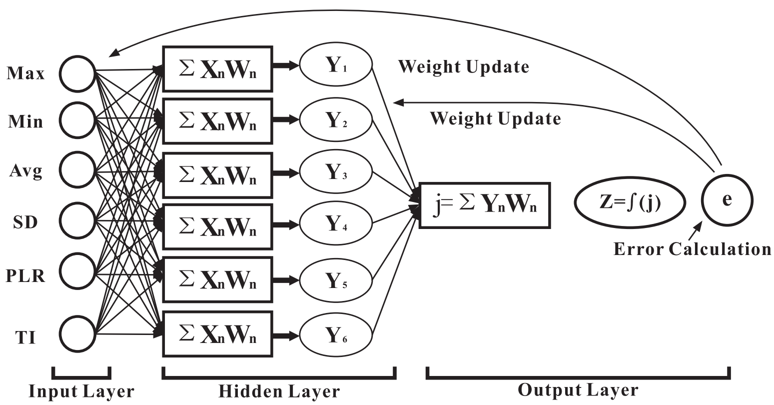

Figure 2. The Integrate stock price prediction with ELM structure.

for medium- and long-term prediction of prediction objects, but it needs a large number of training samples. The flow chart of the algorithm is as follow Figure 3.

In this paper, we adopt the stock time series data to analyze the stock market performance. Suppose the ith stock has time series data S^i:

$$S^i = \{s_1^i, s_2^i, \ldots, s_T^i\} \tag{6}$$

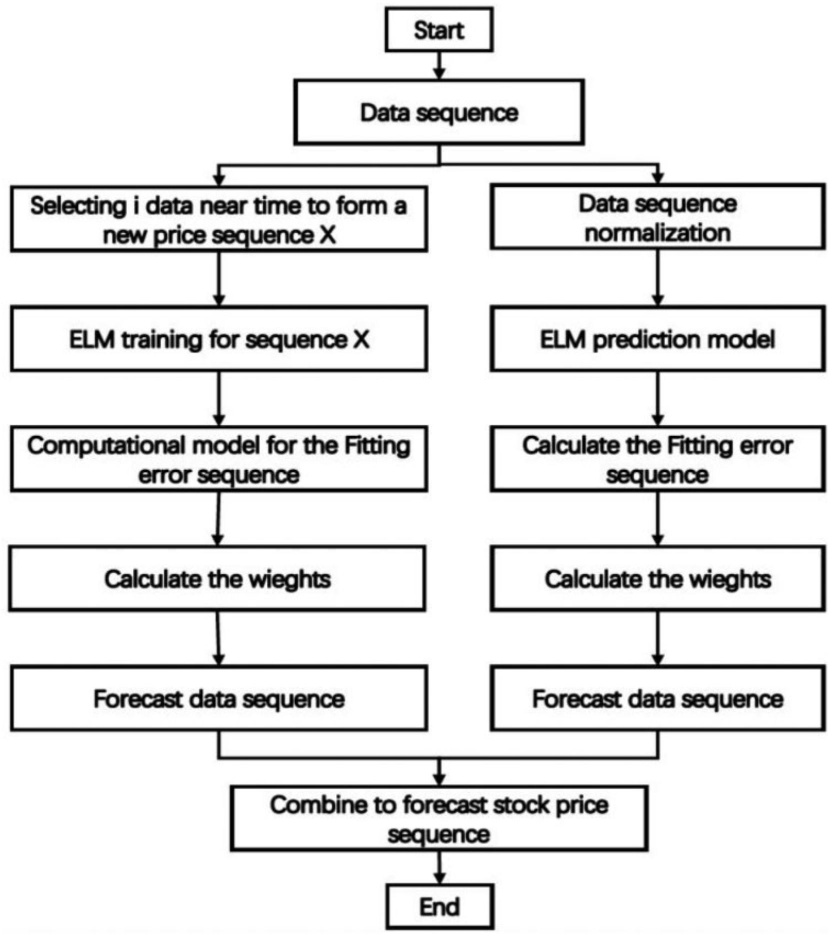

Figure 3. ELM combination model for stock price prediction.

where T denotes time length. Then, the time series of the jth parameter with respect to the ith stock can be represented as x^j:

$$x^{i,j} = \left\{ x_1^{i,j}, x_2^{i,j}, \ldots, x_T^{i,j} \right\} \quad \mathrm{j} = 1,\ 2, \ldots,\ \mathrm{N} \tag{7}$$

here, $x^{i,j}$ stands for the eigen vector of input data in ELM neural network. The stock type y_i, trained by the ELM model, can thus be assessed based on the variation of the stock price in the following M days, which we denote here as Δs_t^i:

$$\Delta s_t^i = \left| \frac{\frac{1}{M} \sum_{j=0}^{M-1} s_{t+j}^i}{s_t^i} - 1 \right| \tag{8}$$

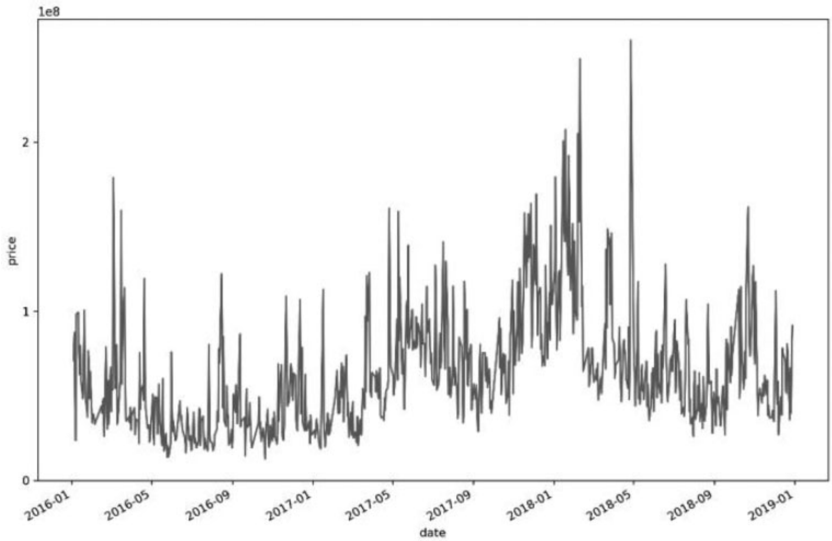

Figure 4. Trend of stock price.

where Δs_t^i denotes the absolute value of stock price change in the future M days with M being period of investigation. Thus, the stock derived from ELM model possesses the following characteristics:

$$y_i = \begin{cases} 1, \Delta s_t^i > \mu \\ 0, \Delta s_t^i \leq \mu \end{cases} \tag{9}$$

which means when the price change becomes greater than the manually set threshold value μ, we view the stock performance in this time window as significant to stock price.

Simulation

Due to the incapability to quantify and model the real outside data of stock, we mainly adopt raw data including the opening price, highest price, lowest price, and closing price to describe the trend of the stock price. Part of the data set we adopt can be visualized as in Figure 4.

In the following we construct the training data set and training parameters that will be used in the ELM model. Here, we treat 10% of the data set as testing data which will be used to better modify and correct the derived ELM model. We record the loss in our model training process, which is displayed in Figure 5. From Figure 5, It can be witnessed that the model becomes stable when the training process reaches 10 epochs.

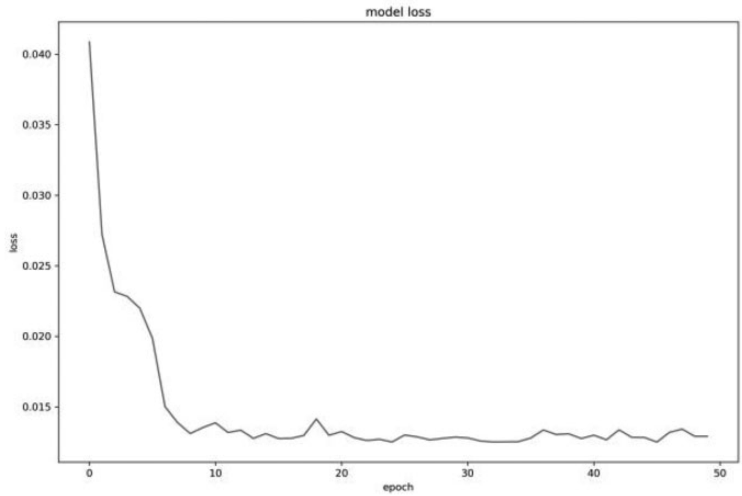

Figure 5. Model training parameters.

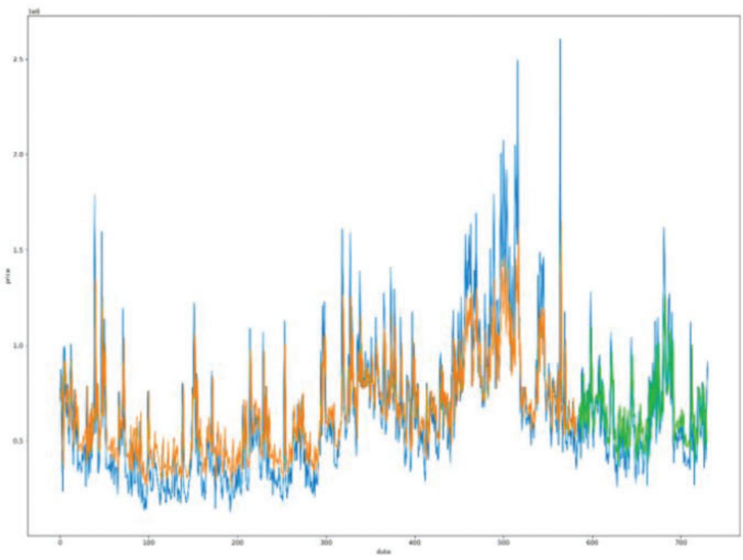

Figure 6. Comparison based on ELM-derived stock price prediction.

We then use the derived ELM model to train and predict the raw stock market data, the process of which is showed in Figure 6, where the blue line depicts the raw data set of stock price, the yellow line represents the prediction output of ELM model constructed from training data set, and the green line shows the predicted

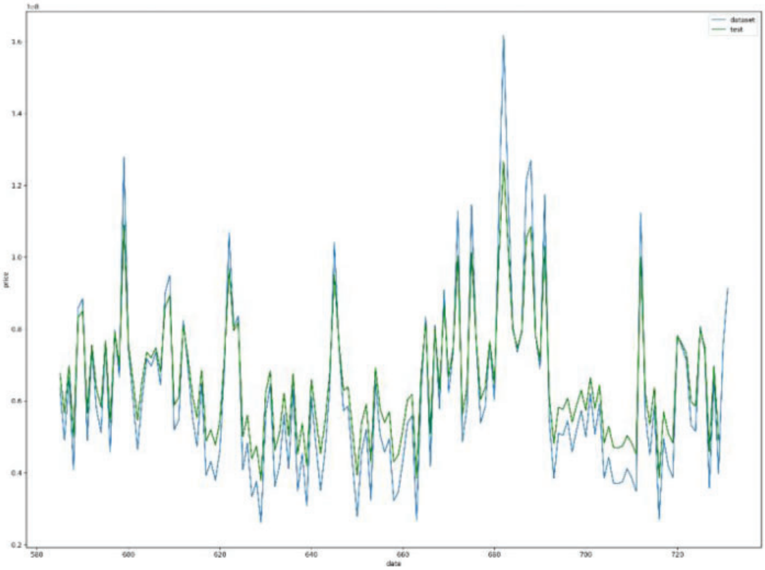

Figure 7. Comparison between predicted stock price trend and real data.

stock price trend derived from well-trained ELM model. From the overall stock price trend, the prediction output of our ELM model is persistent with the real stock price change trend while the predicted peak values are not satisfactory enough at some data points. The overall performance is as expected.

Zoom into the real testing data comparison, we can see that the predicted stock price trend is close to the real situation while flaws still exist in the prediction of peak values (Figure 7).

Conclusion

Stock market analysis and stock price prediction have always been a tough task due to the chaotic data set and the large number of potential influential factors. Historically, various models and techniques have been applied to this field and some methods are verified to produce significant output, discrediting the former consideration of stock price prediction as impossible. Recently, fields of computer intelligence and machine learning are experiencing fast development, which provide stock market analysis with more available models and techniques. Specifically, this paper introduces Extreme Learning Machine (ELM) algorithm, a modification of feedforward neural network with single hidden layer proposed in 2006, to the field of stock price prediction. By incorporating factors such as maximum price and trend indicator in time series data of stock price as inputs, an ELM combination model appropriate for stock price prediction is constructed. To test the validity of the proposed model, a comparison between model-predicted stock price and real stock price is carried out on testing data set, the output of which reveals that the ELM combine model generates satisfactory output for predicting stock price variation trend, where the model-derived output is well confirmed by real data. Meanwhile, inaccuracy still exists in prediction of peak values by the derived ELM combination model. The contribution of this paper lies in trailing with ELM algorithm to predict stock price and producing significant result in predicting stock price change trend and in enlarging range of valid field applications of Extreme Learning Machine algorithm.[19] Further work might focus on how to modify this algorithm to get better prediction result for peak values in stock market price prediction.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

ORCID iD

Fangzhao Zhang https://orcid.org/0000-0003-1633-6265

References

1. Yongjun LI, Guohe F, Deyu QI, et al. Fast incremental weighted support vector machines for predicating stock index. *Control Theory Appl* 2006; 23(5): 805–809.
2. Guo Z, Hui W and Yuan Y. PSO-SVM applied to SWASV studies for accurate detection of cd (II) based on disposable electrode. *Int J Agricult Biol Eng* 2017; 10: 251–261.
3. Yu H and Liu H. Improved Stock Market Prediction by Combining Support Vector Machine and Empirical Mode Decomposition. *Proceedings of the 2012 Fifth International Symposium on Computational Intelligence and Design* Volume 1, 2012. pp.531–534. New York, NY, IEEE.
4. McCulloch SW and Pitts W. A logical calculus of the ideas immanent in nervous activity. *Bull Math Biophys* 1943; 5: 115–133.
5. Hornik K. Approximation capabilities of multilayer feedforward networks. *Neural Networks* 1991; 4: 251–257.
6. Leshno M, Lin VY, Pinkus A, et al. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks* 1993; 6: 861–867.
7. Huang G and Babri HA. Upper bonds on the number of hidden neurons in feedforward networks with arbitrary bounded nonlinear activation functions. *IEEE Trans Neural Netw* 1998; 9: 224–229.
8. Tamura S and Tateishi M. Capabilities of a four-layered feedforward neural network: four layers versus three. *IEEE Trans Neural Netw* 1997; 8: 251–255.
9. Huang G-B, Zhu Q-Y, Siew C-K, et al. Extreme learning machine: theory and applications. *Neurocomputing* 2006; 70: 489–501.

10. Mao KZ. RBF neural network center selection based on fisher ratio class separability measure. *IEEE Trans Neural Netw* 2002; 13: 1211–1217.
11. Mohammed AA, Minhas R, Jonathan Wu QM, et al. Human face recognition based on multidimensional PCA and extreme learning machine. *Pattern Recognition* 2011; 44: 2588–2597.
12. Chang Y, et al. Soft sensing modeling based on extreme learning machine for biochemical process. *Jiliang Xuebao* 2009; 30: 324–327.
13. Rong H-J, Huang G-B, Sundararajan N, et al. Online sequential fuzzy extreme learning machine for function approximation and classification problems. *IEEE Trans Syst Man Cybern B Cybern* 2009; 39: 1067–1072.
14. Romero E and Alquézar R. Comparing error minimized extreme learning machines and support vector sequential feed-forward neural networks. *Neural Netw* 2012; 25: 122–129.
15. Horne C, Van J and Parker GGC. The random-walk theory: an empirical test. *Financ Anal J* 1967; 23: 87–92.
16. Fama FE. Efficient capital markets: a review of theory and empirical work. *J Finance* 1970; 25: 383–417.
17. Kumar A and Shankar G. Priority based optimization of PID controller for automatic voltage regulator system using gravitational search algorithm. In: *2015 international conference on recent developments in controlling and automation power engineering*, 2015, pp.292–297. USA: IEEE.
18. Gandoman FH, Sharaf AM, Abdel Aleem SH, et al. Distributed FACTS stabilization scheme for efficient utilization of distributed wind energy systems. *Int Trans Electr Energ Syst* 2017; 27: e2391.
19. Hanh NV and Hop NH. The effectiveness of the industrial field trip in introduction to engineering: a case study at Hung Yen University of Technology and Education, Vietnam. *Int J Electric Eng Educ* 2018; 55: 273–289.