Journal of Computational and Applied Mathematics 372 (2020) 112723

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

Blockchain financial investment based on deep learning network algorithm[☆]

Meihua Xie[a], Haiyan Li[b], Yuanjun Zhao[c,d,*]

[a] *Department of Accounting, Anhui Finance and Trade Vocational College, Hefei, Anhui, 230000, China*
[b] *Business college, Shandong Yingcai University, Jinan, Shandong, 250000, China*
[c] *School of Business Administration, Shanghai Lixin University of Accounting and Finance, Shanghai, 201209, China*
[d] *Glorious Sun School of Business and Management, Donghua University, Shanghai, 200051, China*

ARTICLE INFO

Article history:
Received 7 May 2019
Received in revised form 13 November 2019

Keywords:
Deep learning
Stock market investment
Neural network
Financial investment

ABSTRACT

In order to study the use of in-depth learning to process financial data, it is proposed that the related technology of neural network and deep learning can be applied to financial data, and real stock index and futures data can be used to explore the application effect of neural network and in-depth learning. Firstly, the theory and model of in-depth learning and neural network are introduced in detail. Secondly, a simple neural network and in-depth learning model are used in the stock index and futures price forecast. The data used in the input of the model include the price of a stock in the current trading and some data indicators, and the closing price of a stock in the next time. The decline will be reflected in output. If the output is up for 1 and down for 0, new data will be input after the training of the model. Finally, the output data can be compared with the real data to judge the application effect of the model, after comparing e and analyzing the application effect of the model. The results show that the research in this study can help investors to build an automated investment model and the investment strategy of the stock market. The construction can be used for reference to improve investors' investment strategy and return rate.

1. Introduction

The computer has been recognized and applied in all walks of life with its unique advantages. When computer technology is combined with various industries, it has a wide impact on the technology and production mode of traditional industries. With the emergence of Internet finance, cloud computing, and big data, it shows that the traditional financial investment mode in the field of financial investment can also be combined with the traditional financial investment mode. With the integration of computer technology, computer learning has certain advantages in data feature screening and analysis. Comparing with traditional investment, computer learning has infinite energy, discipline, and super data processing ability. At the same time, machine learning has a strong self-adaptability in financial investment strategy. It has the function of a self-adjusting strategy that traditional stock market investment cannot achieve. It can adjust its parameters according to the current situation at high speed, and then help financial investors to obtain greater returns [1].

☆ This work was supported by Fujian Provincial Social Science Planning Project, China in 2019 [grant number: FJ2019B08], the Ministry of education of Humanities and Social Science Project, China [grant number: 19YJC630091], and National Social Science Foundation of China [grant number: 18CGL015].

* Corresponding author at: School of Business Administration, Shanghai Lixin University of Accounting and Finance, Shanghai, 201209, China.
E-mail address: zyj2090834@163.com (Y. Zhao).

https://doi.org/10.1016/j.cam.2020.112723
0377-0427/

The deep learning model is similar to the previous artificial neural network model. The main difference between them is actually in the number of hidden layers. At the beginning, there was only one layer of artificial neural network model, and the number of hidden layers of deep learning model was not only several layers, but even more. From the structure of deep learning model, it can be seen as a multi-layer perceptron with many hidden layers. From the working principle of deep learning model, deep learning model has experienced a feature induction from low-level features to high-level features. Deep learning model is to use multiple hidden layers, one layer by one to further summarize the characteristics of the lower layer, and then transfer to the next hidden layer. After that, it is necessary to continue to extract and summarize advanced features. Finally, more accurate feature extraction can be obtained, so as to more accurate approach to the actual characteristics of the original data. From this point of view, it can be seen that the deep learning model is more accurate than the previous artificial neural network model, that is, the reason why the deep learning model is more accurate than the shallow artificial neural network.

Many scholars have done more in-depth research on in-depth learning in full swing. In the meantime, in-depth learning has consistently shown people its powerful ability. In this upsurge, the use of deep learning process for financial data has also set off a thriving research upsurge. Many scholars apply the neural network model to the financial field after it is put forward. Therefore, in recent years, with the rapid development of in-depth learning, scholars have begun to try to apply the technology of in-depth learning to the financial field. However, some people think that because of the high noise of financial data and the weak ability of in-depth learning to resist noise, in-depth learning technology is not suitable for financial data application [2]. Therefore, the basic theory of neural network and deep learning are introduced and summarized. Neural network and deep learning technology are applied to financial data, and real stock index futures data is used to analyze the application effect of understanding the neural network and deep learning. For the first time, deep learning network is applied to stock market forecasting, and the effect of the model is tested and analyzed. Through experiments, the most structured DBN is determined, and the empirical analysis of different stocks is carried out by using the model. At the same time, it is compared with shallow BP neural network. In order to further verify the prediction ability of the model, the original one-minute data is changed into five-minute data and ten-minute data for comparative analysis. All kinds of experimental results show that the model based on fuzzy theory and deep learning network algorithm is effective and has great research prospects.

2. Literature review

Since the beginning of 1980, the artificial intelligence method has been more and more popular with the public. The research and application of artificial intelligence method, which represents the neural network, has developed at an unprecedented speed. These fields, such as commercial prediction, credit rating evaluation, business loss prediction, computer vision, and control system, have been widely used. In 2016, Göçken et al. elaborated the eight-step process of the design of the neural network prediction model, including the trade-off of parameter selection, some common pitfalls and divergence points between practitioners [3]. In 2017, Jie and Wang proposed an improved neural network and fuzzy model for exchange rate forecasting, and discussed the methods of multi-layer perception, radial basis function, dynamic neural network and neuro-fuzzy system [4]. In 2018, Kim and Chang discussed the input variables of exchange rate forecasting, the types of neural network models, the performance comparison of exchange rate forecasting, the stock market index and economic growth. They believed that the economic base was essential to promote exchange rate, stock market index price and economic growth [5]. In 2018, Lahmiri believed that most of the neural network inputs used for exchange rate prediction were univariate, while neural network inputs for stock market index prices and economic growth predictions were multivariate in most cases. There is a mixed comparison of neural network and other models in predicting performance. The reasons may be data differences, prediction levels, neural network model types [6]. In 2018, Das and Padhy proposed a genetic algorithm (GA) method for artificial neural network instance selection for financial data mining. It is considered that artificial neural networks have excellent learning ability, but often exhibit inconsistency and unpredictability for noise data. Performance [7]. In 2017, Song et al. applied a genetic algorithm model to stock market analysis. Experimental results show that the genetic algorithm is a promising artificial neural network instance selection method [8]. In 2018, Pang et al. believed that in financial time series forecasting, the problem of how to use noisy financial data to improve prediction accuracy as much as possible is encountered. Therefore, using supervised neural networks as learning technology to design financial time series prediction is proposed, so as to solve this problem [9]. In 2016, Kaur et al. proposed a new financial time series prediction neural network method based on a periodic component combination. All parameters are estimated in real time in a set of predictors, and then the outputs of these predictors are optimally combined, so as to obtain the final prediction results [10]. In 2018, Nayak and Misra proposed a time series prediction method, which was applied to the prediction of two daily foreign exchange spot exchange rate time series. The results show that the method borrows from chaotic time series analysis, clustering and artificial neural network, and evolutionary computing and other disciplines [11]. In 2017, Chong et al. proposed a multi-scale neural network learning paradigm to predict financial crisis events to achieve early warning purposes. In the proposed multi-scale neural network learning paradigm, the currency exchange rate is first selected as a typical financial indicator reflecting economic fluctuations [12].

A new direction in the field of machine learning research is deep learning, which brought revolution in many applications such as speech recognition and image recognition. In 2016, Meng et al. proposed a new method for applying deep learning to physiological signal analysis, enabling doctors to identify potential risks. In this method, a convolutional

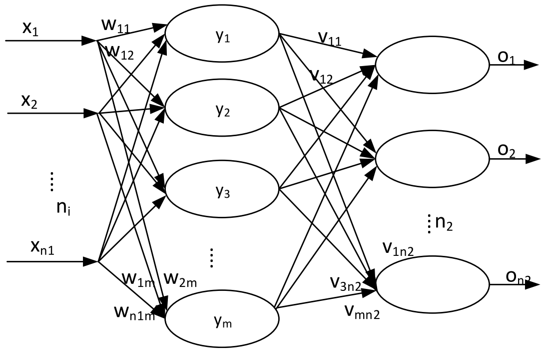

Fig. 1. Structural chart of BP neural network.

neural network-based model is used, which can automatically learn features from the original physiological signals without supervision. Then, based on the learned characteristics, the anomaly data is detected by multivariate Gaussian distribution anomaly detection method. The experimental results show that the method has important application value in physiological signal anomaly detection [13]. In 2017, Mahmud et al. proposed a fault diagnosis model, Deep Intelligent Generalized Regression Neural Network (DGN), which is a supervised deep learning model. Experimental results show that the model has low accuracy and speed [14]. In 2016, Cha et al. proposed a low-intercept probability radar signal recognition algorithm based on radar signal depth learning. The analysis and simulation results show that the algorithm can achieve an average recognition accuracy of 93.4% or higher for FM continuous wave (fmcw), frank, costas and FSK/PSK signals [15]. In 2018, Wen et al. proposed a cascaded pedestrian detection model based on local decorrelation channel characteristics (LDCF) and CNN. The experimental results show that the log average error rate of LDCF+CNN algorithm is lower than the LDCF algorithm and the bit error rate is lower than 13.21% [16]. In 2018, Sato et al. proposed a hybrid space–time saliency feature extraction framework based on deep learning, which is used for the detection of the saliency of video footprints. Compared with traditional manual methods, deep learning networks are very effective for extracting hidden features [17]. In 2018, Wang et al. used an electroencephalogram (EEG) signal based on the Deep Trust Network (DBN) model to identify emotional states (positive, negative, neutral) in an open source deep learning framework and Support vector machine (SVM) for comparison [18]. In 2018, Ubbens et al. proposed to use the deep learning theory to study the candidate actively' s answer characteristics and transform the answer extraction problem into feature learning and classification problems, that is using word vectors to represent the characteristics of the problem [19]. In 2018, Khan and Yari proposed model R, an established neural network model. Model R shows that deep learning can be successfully applied to link weight prediction, which is superior to random block model and its derivative up to 73% in prediction accuracy [20].

3. Proposed method

3.1. Neural networks

BP neural network model: Based on the characteristics of neurons, a typical neuron model is shown in Fig. 1.

A typical BP neural network has three layers, which are, an input layer, an implicit layer, and an output layer. A complete connection is applied among layers. The input layer node is x_j, the hidden layer node is y_j, the output layer node is o_j, w_{ij} is the connection weight of the input layer and the output layer, v_{ij} is the connection weight of the hidden layer and the output layer, and θ_i is the threshold value. The expected output of the network is t_l.

The BP neural network formula is as follows:

Firstly, the output formula of the hidden layer node is

$$y_j = f\left(\sum w_{ij}x_j + \theta_i\right) = f\left(net_i\right) \tag{1}$$

In the formula, $net_i = \sum_j w_{ij}x_{ij} + \theta_i$

Secondly, the output layer node formula is

$$o_l = f\left(\sum_i v_{ti}y_i + \theta_l\right) = f\left(net_l\right) \tag{2}$$

In the formula, $net_i = \sum_j w_{ij}x_{ij} + \theta_i$

Thirdly, the error output formula is

$$\begin{aligned} E &= \frac{1}{2}\sum_l (t_l - o_l)^2 = \frac{1}{2}\sum_l \left(t_l - f\left(\sum_l v_{1i} y_i + \theta_l\right)\right)^2 \\ &= \frac{1}{2}\sum_l \left(t_l - f\left(\sum v_{1i} f\left(\sum_j w_{ij} + \theta_t\right) + \theta_l\right)\right)^2 \end{aligned} \tag{3}$$

Firstly, the formula for the output layer node is

$$\frac{\partial E}{\partial v_{1i}} = -\left(t_l - o_l\right) f\left(net_l\right) y_i \tag{4}$$

Secondly, the formula for the hidden layer node is

$$\frac{\partial E_l}{\partial w_{ij}} = -\sum \left(t_l - o_l\right) f\left(net_l\right) v_{1i} f\left(net_l\right) x_j = -\sum_t \delta_l v_{1i} f\left(net_l\right) . \bullet x_j$$

Due to the correction of the weight Δv_{ij}, Δw_{ij} is proportional to the error function along the gradient, so

$$\Delta v_{1i} = -\eta \frac{\partial E}{\partial v_{1i}} = \eta \delta_l y_i \tag{5}$$

$$\delta_l = \left(t_l - o_l\right) f'\left(net_l\right) \tag{6}$$

$$\Delta w_{1i} = -\eta' \frac{\partial E}{\partial w_{1i}} = \eta' {\delta_i}' y_j \tag{7}$$

$$\delta_i{}' = f'\left(net_l\right) \sum_l \delta_i v_{1i} \tag{8}$$

Fourthly, the output layer correction formula is
Error control, all sample errors are:

$$E = \sum_{k=1}^{p} e_k < \varepsilon \tag{9}$$

Single sample error is

$$e_k = \sum_{i=1}^{n} \left| t_l{}^{(k)} - o_1{}^{(k)} \right| \tag{10}$$

In the formula, p is the number of samples, and n is the number of output nodes.
Error formula is

$$\delta_l = \left(t_l - o_l\right) o_l \left(1 - o_l\right) \tag{11}$$

Weight correction formula is

$$v_{li}\left(k+1\right) = v_{li}\left(k\right) + \eta \delta_l \tag{12}$$

Fifthly, the implicit layer node correction formula is
Error formula is

$$\delta_l{}' = y_i\left(1 - y_i\right) \sum_l \delta_l v_{li} \tag{13}$$

Weight correction formula is

$$w_{ij}\left(k+1\right) = w_{ij}\left(k\right) + \eta' \delta' x_j \tag{14}$$

The threshold correction formula is

$$\theta_i\left(k+1\right) = \theta\left(k\right) + \eta' \delta'{}_l \tag{15}$$

3.2. BP neural network algorithm flow

Firstly, the model is initialized. Secondly, the output vector X is randomly selected, and the node output of the output layer is calculated $o_i = f\left(net_i\right) = f\left(w_i g\left(net_{i=1}\right)\right)$. Thirdly, the net input and actual output of each neuron in the hidden layer and the output layer are calculated. Fourthly, the error of each output layer is calculated $E = \sum E_K = \frac{1}{2}\sum\sum e^2{}_{ik}$.

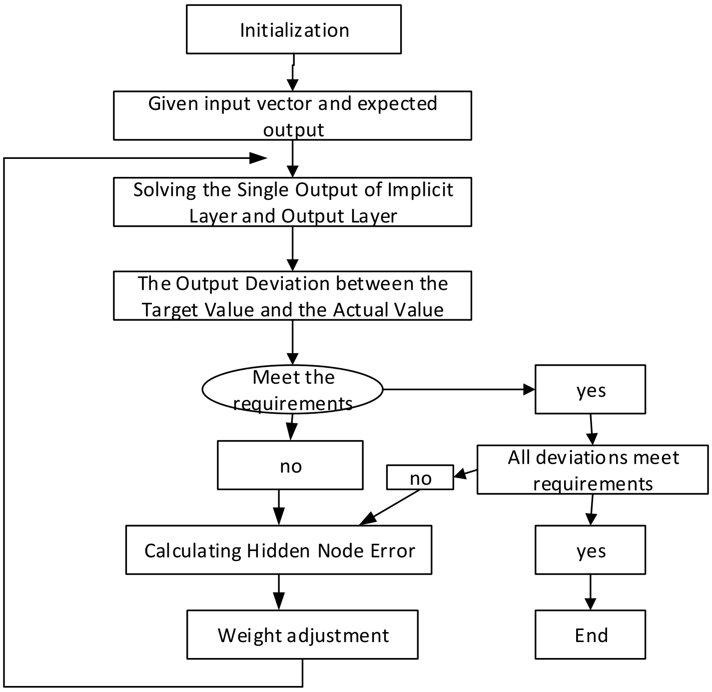

Fig. 2. Flow chart of BP neural network.

Fifthly, the error of each layer is calculated as $\varepsilon^{(r)}{}_{p,k}$ $(r = 0, 1, 2)$. Sixthly, the hidden layer is adjusted to the weight of the output layer $w_k = w_{k-1} + \eta \Delta w_k$. Seventhly, the input layer is adjusted to the weight of the hidden layer $v_j = v_{j-1} + \eta \Delta v_j$. Eighthly, the total network error is calculated to see if the maximum error requirement is met. Ninth, if not, it is necessary to return to step (2). Otherwise, the training ends.

The network training process is shown in Fig. 2.

The advantage of using BP neural network model to predict stocks: the neural network has a mapping function and can deal with many nonlinear problems, which is more suitable for solving those internal complicated problems. BP neural network parallel processing makes it have good fault-tolerant and faster processing speed. BP neural network has good self-learning ability and promotion and generalization ability.

3.3. Deep learning model

Deep learning has become more and more popular with the current large number of applications and the promotion of news media. Scholars and researchers are increasingly researching deep learning and deep learning model-based applications. When it comes to the deep learning model, it is inevitable to associate with the concept of artificial intelligence [21–25]. Therefore, it is necessary first to analyze the concepts of this series and understand the connections and differences. From a subordinate perspective, artificial intelligence is a broad concept that encompasses all relevant technologies in the field. A category that is slightly smaller than artificial intelligence is machine learning, and machine learning uses computers to learn features in raw data. Through the observed data set, the development law of the research object can be mastered, and then the rules of mastering to combine the newly appeared data or the given possible data can be used to speculate the possible situation of the future research object [26–30].

Deep learning is a Hubel weissel model simulating the cerebral cortex, which uses layers of "abstraction" to express data or signals, similar to the cerebral cortex's resolution deep learning model for images. First of all, from the original signal (similar to the pixel in the face recognition model), the lower level feature is isolated (similar to the edge of the object in the face recognition system), and then the higher level feature (similar to the contour formed by the edge in the face recognition system) is obtained from the lower level feature. Then, a higher level of expression is obtained (similar to the face in face recognition). Finally, a classifier is established on the high-level features to obtain the prediction output of the model, as shown in Figs. 3 and 4.

The discriminant constrained Boltzmann machine can be used for classification, image recognition and so on. It can also be used as the last module of the depth confidence network to build a deeper network. In this study, a single discriminant constrained Boltzmann machine is used as the prediction tool, and principal component analysis is used to reduce the dimension of the selected stock factors, and a stock prediction model based on PCA-DRBM is constructed. The previous day's data are used to predict the stock's up and down trend the next day

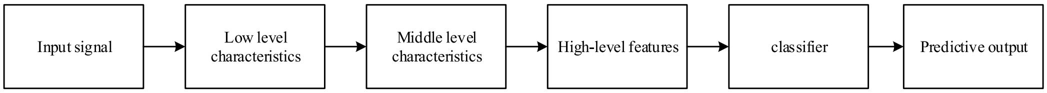

Fig. 3. The hierarchical structure of deep learning.

Fig. 4. The model operation flow chart.

Conceptually, it is obvious that artificial neural networks are included in machine learning. Artificial neural networks use mathematical methods to simulate neural networks composed of contacts, neurons, and cells in a biological organism, so as to achieve the goal of building the model. The deep learning model belongs to the category of the artificial neural network model. From the perspective of the construction of the two models, the foundation of the deep learning model is the artificial neural network model, but a large amount of hidden is added on the basis of the artificial neural network model. The inclusion layer enhances the level of feature learning and improves the accuracy of the final feature extraction. The deep learning model also includes two ways of learning features. One is supervised learning, and the other is unsupervised learning. Supervised learning conducts supervised learning by training deep learning models with tagged data sets, while unsupervised learning uses unlabeled data sets to train deep learning models.

In the application of the deep learning model, the situation of the problem and whether the existing data has a label, which training method to apply can be decided. In this study, because the deep learning model is used to learn the characteristics of the securities price change, the two-day securities price is used as a label to train the deep learning model required, so a supervised learning approach will be used in the study.

4. Experiments

In this study, the price data of stock index futures through the neural network and deep learning model are introduced in the previous study, and a predictive model is established to predict the short-term price of stock index futures, so as to capture short-term trading opportunities in actual transactions. In order to better observe the application effect of neural network and deep learning, in this study, the actual quantitative trading is simplified, and only the direction of stock index futures data is predicted, that is, only the ups and downs are predicted. The empirical goal of this study is l focused on predicting the rise and fall of the closing price of the stock index futures at the next moment by the data available at the current moment of stock index futures. The data obtained at the current moment can be used as the independent variable X, as the input of the model. If the closing price of the next time rises, the dependent variable is $Y = 1$. If the closing price of the next time falls, then the dependent variable is $Y = 0$.

Because the financial data has the characteristics of time sequence, the front data must be taken as training take and the back data must be taken as testing and forecasting, so the accuracy cannot be tested by cross validation. In order to simulate the actual investment process, the research method of sliding window is adopted in this study. The sliding window takes the data at the beginning as the research object for training and prediction. After the prediction output result is completed, the whole window slides forward to continue the training and prediction. Compared with the usual methods of data partition, such as random set aside method, the sliding window improves the utilization rate of data as much as possible, and it is more rigorous, scientific and in line with the actual situation. In each data window, data is divided into training interval, detection interval, space and test interval. Among them, the training interval is used to train the model and get the parameters. The detection interval is used to test and select the parameters of the model. Space interval is used to prevent the rate of return vector of training interval from using the data of test interval, so as to affect the prediction accuracy. The test interval is used to output the predicted value and compare it with the actual value.

It can be seen that the training set accuracy and test set accuracy are significantly higher than the reference value, indicating that the model is effective. The prediction accuracy of all characteristic data is summarized as follows:

It can be seen from Table 1 that the prediction accuracy of PCA60 and AE80 is relatively higher in their respective models, and RBM40 and RBM80 are very close. Compared with the other two models, the prediction accuracy of PCA model is higher than that of the other two models. However, the prediction accuracy of Raw Data is higher than that of other feature data, which indicates that the direct feature extraction of financial data is not good.

The data has been taken from Shanghai and Shenzhen 300 stock index futures. Since the futures adopts the $T + 0$ trading system, it is a very suitable trading type for traders who do short-term trading. The goal is to make a directional forecast for the price in the short term, so it is appropriate to select the CSI 300 stock index futures data. The data source

Table 1
Summary of accuracy rate.

Characteristic data	Characteristic number	Training set	Test set
Raw Data	60	0.5731	0.5723
PCA20	20	0.5591	0.5421
PCA60	60	0.5754	0.5706
AE40	80	0.5417	0.5362
AE80	160	0.5619	0.5482
RBM40	80	0.5637	0.5512
RBM80	160	0.5627	0.5518

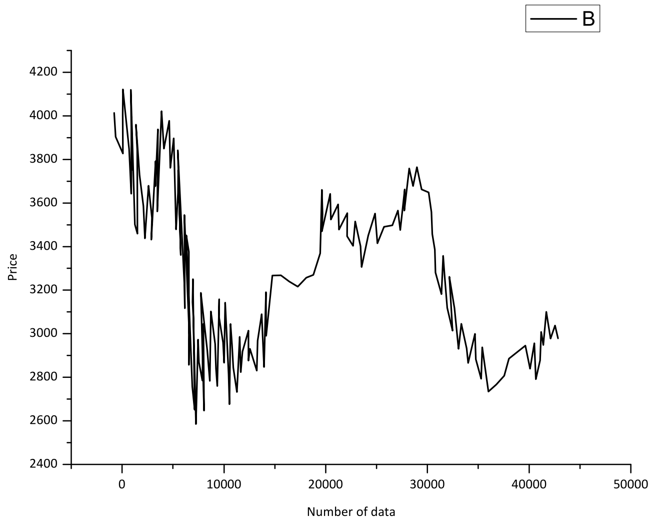

Fig. 5. Minute price volatility chart.

in the text is the wind database. The data selection interval is from the xxxx year x month x number to the xxxx year x month x day minute data. The total data volume is 42 275. Each data contains the price data, which consist of the opening price, the highest price, the lowest price, and the closing price. Each data also contains the trading volume, turnover, and technical indicators such as KDJ, RSI, DMI, BIAS, and MA. The closing price data for this period is shown in Fig. 5. It can be seen that the fluctuations of these minutes in this period are still relatively large. If the forecast is accurate, there will be an arbitrage opportunity.

In this study, the opening price, closing price, highest price, lowest price, trading volume and technical indicators KDJ, RSI, DMI, BIAS, and MA are used for the input data of the model. $Xt = (X_t{}^1, X_t{}^2, X_t{}^3, X_t{}^4, X_t{}^5, X_t{}^6, X_t{}^7, X_t{}^8, X_t{}^9, X_t{}^{10},)$ represent these data at time t.

In order to eliminate the influence of different units of measurement on the results, the data is first standardized, and all data are normalized in the following way.

$$\overline{x_t^i} = \frac{x_t^i - \min x_t^i}{\max x_t^i - \min x_t^i} \tag{16}$$

The processed data lies between 0 and 1. If the price of the closing price at the next moment rises, the dependent variable $Y = 1$, and if it falls, $Y = 0$. Thus, the input data of our model is a vector with 10 elements, and the output is a binary variable. Since the main concern is to rise or fall, and it is only profitable when it is up or down, in reality, it is possible that the closing price is remained unchanged at the next moment, which is unprofitable. Therefore, the measure taken in the model is to remove the data directly and train the model with only the data that has risen or fallen at the next moment. After these treatments, the amount of data becomes 26,846, and the distribution of rising and falling data is shown in Fig. 6. Among them, there are 13,645 falling data, accounting for 51.8%, while the rising data is 13,201, accounting for 48.2. %, the two types of data are basically balanced.

5. Discussion

Firstly, a simple single hidden layer neural network is used for training. Since the input data has ten components, the input layer of the network has ten neurons, and the output layer has one neuron. Since the output Y is a binary variable and the output result is a value between 0 and 1, a threshold (a) is selected, and 0.5 is selected as the threshold. The

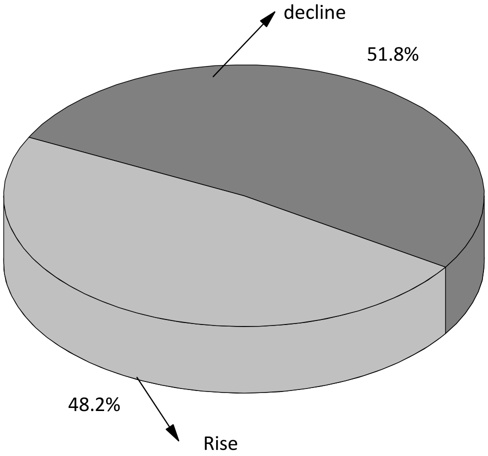

Fig. 6. Increase-decline data ratio.

Table 2
The first five results with a low error rate.

h	l	epo	Trainer	Tester
100	0.55	2200	0.11627	0.23023
80	0.77	2200	0.1243	0.23496
200	1.1	2200	0.11627	0.24013
150	0.44	2200	0.15686	0.24475
50	5.5	3300	0.14344	0.24618

value whose output value is greater than 0.5 is judged as predy = 1, and the value whose output value is less than 0.5 is judged as predy = 0, which means

$$\text{predy} = \begin{cases} 0 & \text{ouput} \geq 0.5 \\ 1 & \text{ouput} < 0.5 \end{cases}$$

The parameters are usually referred to as parameters obtained through learning in the model, and the so-called hyperparameters are parameters that point to the model before the training model, such as the number of neurons in the hidden layer, the learning rate, and the number of learning. It determines the structure of the model and some of the characteristics of the model as it learns. The selection of hyperparameters in neural networks is very important. If the super parameters are not selected suitably, the results can be unsatisfied. Therefore, the training of the model will first focus on the selection of hyperparameters.

In this study, a four-layer deep neural network structure with two hidden layers is constructed. The number of neurons in the second and third layers is 120 and 60, respectively. Relu function is selected as the activation function in the model, because compared with Sigmoid activation function, Relu activation function provides faster learning speed, and it has better learning effect when applied to deep neural network model. At the same time, in this study, the artificial neural network is used as the control group, and the usual sigmoid function is used as its activation function, and the settings of other parameters are the same.

20,000 data are randomly extracted from all 26 846-training data as training samples, and the remaining 6846 data are used as test samples. Then, the previously selected combination of parameters is used to train, and the training error rate and the test error rate are recorded separately as before. After completing the training, the final result is concluded as an output, recording the training error rate and the test error rate under each parameter combination. The data is then sorted according to the test error rate from low to high, and the first five results with lower error rates are checked after sorting, as shown in Table 2.

The parameter with the lowest test error rate in the selected result is the parameter with the highest prediction accuracy for the test sample as the parameter of the final model. Therefore, the number of hidden layers of the neural network is selected to be 100, the learning rate is 0.5, and the trained epoch is 2000. Under these parameters, the Y value predicted by the neural network is compared to the true Y value of the test sample. The results show that among the 6846 test samples, 3972 are predicted, with an accuracy rate of 58.01%. Among the 6846 test samples, there are 3491 rising data and 3355 falling data. Of the rising data, 2060 are predicted, and 1912 of the falling data are predicted, as shown in Table 3.

It can be seen from the model results that when the real price rises, the model has a 59% chance to give a signal of the price increase. When the real price falls, the model also has a 56.98% chance to give a down signal. In theory, there

Table 3
The model results.

	True $y = 0$	True $y = 1$
Pred $y = 0$	1912	1431
Pred $y = 1$	1443	2060

Table 4
The sample results.

	True $y = 0$	True $y = 1$
Pred $y = 0$	1985	1356
Pred $y = 1$	1349	2156

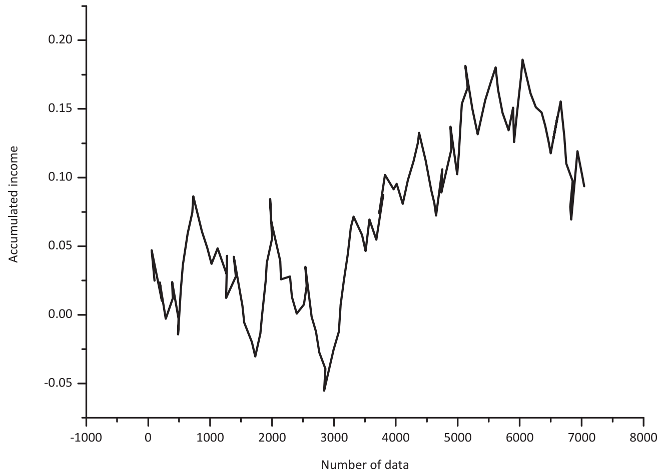

Fig. 7. Accumulated return rate map.

is a chance of profit. Considering the handling fee of two-tenths of a second for each transaction, the cumulative yield on the 6846 data is calculated by calculation. The relationship between the cumulative yield and the number of data can be seen in Fig. 7. From the figure, although the cumulative rate of return fluctuates, it shows an inevitable upward trend as a whole, indicating that as the number of data increases, the overall profit is also increasing.

Next, the deep learning method is used to train and predict using the deep learning model composed of the deep belief network mentioned in the previous section. The training method of the deep belief network is not the same as the single hidden layer neural network training method. It firstly uses the restricted Boltzmann machine to perform pre-training layer by layer, and then uses the feedback method to fine-tune, like a simple neural network, the input is still 10 neurons, but the hidden layer has two layers. At the time of training, 20,000 data are still used as training samples, and 6846 data are used as test samples. Due to the increase in the number of network layers, the pre-training time and tuning time are much higher than the single hidden layer neural network. After the pre-training and tuning are completed, the test sample data is input into the network, and an output is generated. At this time, the output is still a value between 0 and 1. As with the above processing method, the value is still 0.5 as the closed value. After comparing with the actual value, the results are as follows in Table 4:

It can be seen from the results that among the 6846 test samples, the number of accurate predictions is 4141, and the accuracy rate is 60.4%. Among the 3491 data with the real price increase, 2156 are predicted to be rising, and the accuracy rate is 61.7%. Among the 3355 data with the real price drop, the forecasted decline is 1985, and the accuracy rate is 59.2%. Compared with the single hidden layer neural network, whether it is rising data or falling data, the prediction accuracy of deep learning is improved, and the final total prediction accuracy is correspondingly improved. Similarly, the cumulative yield is calculated for the data under the predicted results, and the cumulative yield on the 6846 test samples is 12.9%. The graph of the cumulative yield on the sample is shown in Fig. 8. As it can be seen from the figure, although there are times, when the rate of return is negative, in the end, it is still profitable overall. Moreover, compared with the simple neural network, the profit rate is also slightly improved, indicating that the higher the forecast accuracy rate is, the higher the profit will be (see Table 5).

Deep reinforcement learning strategy cannot achieve good results for all individual stocks, but it is effective for most of them. The effectiveness of comprehensive evaluation and the effectiveness of individual stocks show that the strategy has limitations. In this case, deep reinforcement learning strategy shows the characteristics of approximate rate of profit, but not the characteristics of complete profit. Therefore, in the actual investment, it is necessary to reduce the strategy's

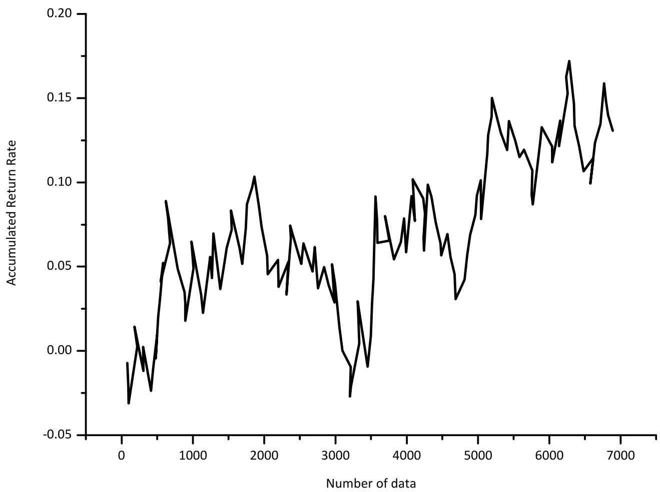

Fig. 8. Accumulated return rate map.

Table 5
Running time of five algorithms to test function.

Test function	Running time (unit: second)				
	BAS	PSO	BA	FRA	MBAS
1	1.458	1.712	2.523	2.011	4.642
2	1.216	1.500	2.761	1.151	5.571
3	1.256	1.723	2.606	1.234	4.916
4	1.283	1.954	2.568	1.231	6.096
5	1.241	1.938	2.589	1.186	6.093
6	1.253	1.789	2.531	1.238	5.026
7	1.235	1.774	2.583	1.101	4.851
8	1.316	1.925	2.562	1.304	6.249

inadaptability to individual stocks and reduce the strategy's risk through the method of decentralized investment. To a certain extent, this empirical study verifies the role of decentralized investment in risk diversification, and also shows that deep reinforcement learning strategies need to diversify risk in order to make profits.

Through the comparison of model results, it can be seen that the prediction effect of shallow BP neural network is slightly lower than that of deep learning neural network, but the difference is not very obvious The average absolute error and square root mean square error of BP neural network in prediction effect are obviously different from those of deep learning neural network. Through the comparison of the prediction results of the stock, it can be seen that the deep learning neural network has better prediction effect than the shallow neural network.

6. Conclusions

This study uses the single hidden layer neural network and deep learning model to model the stock index futures data, and predicts the rise and fall of the closing price at $t+1$ according to the price, transaction data and technical indicators of the previous t time. The predicted result is shown in the following table.

When using a single hidden layer neural network, the overall prediction accuracy is 58.01%, the prediction accuracy when rising is 59%, and the prediction accuracy when falling is 56.98%. When using the deep learning model, the overall prediction accuracy rate and the rising and falling prediction accuracy rate are improved compared with the single hidden layer neural network. The overall prediction accuracy rate is 60.4%, which is 2.39 percentage points higher. The prediction accuracy rate when the price rises are 61.7%, which is improved 2.7 percentage points. The forecast accuracy rate, when the price falls, is 59.2%, an increase of 2.22 percentage points.

In general, whether it is a single hidden layer neural network or a deep learning model, there is a certain predictive power for stock index futures, and deep learning is more predictive than a single hidden layer neural network. According to the results of the model, there are certain arbitrage opportunities. However, since the forecast is only a directional forecast, it only predicts the ups and downs but does not give a specific forecast of the rise and fall. However, in practice, considering the problem of transaction costs and the factors such as the stop-loss point and the training time that may occur in a minute of large fluctuations, the model is not guaranteed to be absolutely effective in practice, and is only used as a reference method in actual quantitative investment.

References

[1] T.D. Chaudhuri, I. Ghosh, Forecasting volatility in Indian stock market using artificial neural network with multiple inputs and outputs, Int. J. Comput. Appl. 120 (8) (2016) 7–15.
[2] A.H. Moghaddam, M.H. Moghaddam, M. Esfandyari, Stock market index prediction using artificial neural network, J. Econ. Finance Adm. Sci. 21 (41) (2016) 89–93.
[3] M. Göçken, Özçalıcı, Mehmet, A. Boru, et al., Integrating metaheuristics and Artificial Neural Networks for improved stock price prediction, Expert Syst. Appl. 44 (C) (2016) 320–331.
[4] W. Jie, J. Wang, Forecasting stochastic neural network based on financial empirical mode decomposition, Neural Netw. 90 (2017) 8–20.
[5] H.Y. Kim, H.W. Chang, Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-type models, Expert Syst. Appl. 103 (2018) S0957417418301416.
[6] S. Lahmiri, A technical analysis information fusion approach for stock price analysis and modeling, Fluct. Noise Lett. 17 (1) (2018) 1850007.
[7] S.P. Das, S. Padhy, A novel hybrid model using teaching–learning-based optimization and a support vector machine for commodity futures index forecasting, Int. J. Mach. Learn. Cybern. 9 (1) (2018) 97–111.
[8] W. Song, W. Song, W. Song, et al., A double-layer neural network framework for high-frequency forecasting, Acm Trans. Manage. Inf. Syst. 7 (4) (2017) 1–17.
[9] X. Pang, Y. Zhou, W. Pan, et al., An innovative neural network approach for stock market prediction, J. Supercomput. (1) (2018) 1–21.
[10] G. Kaur, J. Dhar, R.K. Guha, Minimal variability OWA operator combining ANFIS and fuzzy c-means for forecasting BSE index, Math. Comput. Simul. 122 (2016) 69–80.
[11] S.C. Nayak, B.B. Misra, Estimating stock closing indices using a GA-weighted condensed polynomial neural network, Financ. Innov. 4 (1) (2018) 21.
[12] E. Chong, C. Han, F.C. Park, Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies, Expert Syst. Appl. 83 (2017) 187–205.
[13] G. Meng, W. Cong, C. Zhu, Construction of hierarchical diagnosis network based on deep learning and its application in the fault pattern recognition of rolling element bearings, Mech. Syst. Signal Process. 72–73 (2) (2016) 92–104.
[14] M. Mahmud, M.S. Kaiser, A. Hussain, et al., Applications of deep learning and reinforcement learning to biological data, IEEE Trans. Neural Netw. Learn. Syst. 29 (6) (2017) 2063–2079.
[15] K.H. Cha, L.M. Hadjiiski, R.K. Samala, et al., Bladder cancer segmentation in CT for treatment response assessment: Application of deep-learning convolution neural network—A pilot study, Tomogr. A J. Imaging Res. 2 (4) (2016) 421–429.
[16] D. Wen, Z. Wei, Y. Zhou, et al., Deep learning methods to process fMRI data and their application in the diagnosis of cognitive impairment: A brief overview and our opinion, Front. Neuroinform. 12 (23) (2018) 23.
[17] M. Sato, K. Horie, A. Hara, et al., Application of deep learning to the classification of images from colposcopy, Oncol. Lett. 15 (3) (2018) 3518–3523.
[18] R. Wang, L. Jie, Y. Duan, et al., Study on the combined application of CFAR and deep learning in ship detection, J. Indian Soc. Remote Sens. (4) (2018) 1–9.
[19] J. Ubbens, M. Cieslak, P. Prusinkiewicz, et al., The use of plant models in deep learning: an application to leaf counting in rosette plants:, Plant Methods 14 (1) (2018) 6.
[20] S. Khan, T. Yairi, A review on the application of deep learning in system health management, Mech. Syst. Signal Process. 107 (1) (2018) 241–265.
[21] Liu Feng, Mao Suzhen, L-P bounds for nonisotropic marcinkiewicz integrals associated to surfaces, J. Aust. Math. Soc. 99 (3) (2015) 380–398.
[22] Tramontana Fabio, A.A. Elsadany, Xin Baogui, H.N. Agiza, Local stability of the cournot solution with increasing heterogeneous competitors, Nonlinear Anal. RWA 26 (2015) 150–160.
[23] Meng Xinzhu, Liu Rui, Zhang Tonghua, Adaptive dynamics for a non-autonomous Lotka–Volterra model with size-selective disturbance, Nonlinear Anal. RWA 16 (2014) 202–213.
[24] Liu Xikui, Li Yan, Zhang Weihai, Stochastic linear quadratic optimal control with constraint for discrete-time systems, Appl. Math. Comput. 228 (2014) 264–270.
[25] M. Wang, G. Wang, Four properties for complex potentials in power series form, Appl. Math. Comput. 219 (9) (2013) 4185–4193.
[26] X. Zhou, X. Shi, H. Cheng, Modelling and stability analysis for a tuberculosis model with healthy education and treatment, Comput. Appl. Math. 32 (2) (2013) 245–260.
[27] S. Pang, T. Li, F. Dai, M. Yu, Particle swarm optimization algorithm for multi-salesman problem with time and capacity constraints, Appl. Math. Inf. Sci. 7 (6) (2013) 2439.
[28] Z. Bai, W. Sun, Existence and multiplicity of positive solutions for singular fractional boundary value problems, Comput. Math. Appl. 63 (9) (2012) 1369–1381.
[29] Z. Zhang, H. Shao, Z. Wang, H. Shen, Reduced-order observer design for the synchronization of the generalized Lorenz chaotic systems, Appl. Math. Comput. 218 (14) (2012) 7614–7621.
[30] Z. Bai, Eigenvalue intervals for a class of fractional boundary value problem, Comput. Math. Appl. 64 (10) (2012) 3253–3257.