

Research on Financial Data Prediction Based on Deep Autoencoder

Ziming Gao¹, Jinchang Zhang², Boyang Nan³

¹Xi'an Jiaotong-liverpool University, Jiangsu 261041

²College of Information Control, China University of Mining and Technology, Xuzhou, Jiangsu, 221116

³College of Economics and Management, Three Gorges University, Yichang, Hubei, 443002

Keywords: Deep Learning; Stock Index Futures; Trend Forecasting; Trading Strategy; Deep Autoencoder

Abstract: As a revolutionary achievement in machine learning in the era of big data, deep learning has developed rapidly since its introduction, and it has created an upsurge of research and application in the Internet field. From the perspective of market microstructure, the formation and change of stock prices are determined by the trading behavior of buyers and sellers. Therefore, the mining of high-frequency market data may have a predictive ability for future stock price movements. In this paper, the deep learning prediction model is trained by a large amount of historical data in the sample, and the price index of the stock index futures is predicted at 1 sec. The accuracy of the model outside the sample is over 73%. At the same time, based on the prediction of stock price changes by deep learning stock price forecasting model, this paper proposes the intraday trading strategy of stock index futures. The trading strategy has accumulated 99.6% yield since 2013, with an annualized rate of return of 77.6% and a maximum retracement of -5.86%. Through the empirical study of the high-frequency price forecasting model of stock index futures, this paper verifies the effectiveness of the machine learning tool in the big data era of deep learning in stock price forecasting, and proposes a stock index futures trading strategy based on the forecasting model, which has achieved good results.

1. Introduction

Along with the development of machine learning, quantitative hedge funds such as Renaissance and DE Shaw must continue to absorb new machine learning models and explore new trading strategies. With the rapid development of computer science and technology, the reduction of storage costs and the improvement of computing speed, people are paying more and more attention to the popular terms such as “cloud computing” and “big data”. There is more and more data in the financial market – there are nearly 2,000 stocks in the A-share market alone. Considering the high-frequency data in seconds, there will be more than 20 million new data samples per trading day. Using deep learning to extract useful information from the “data ocean” can be an effective means to help investors get excess returns.

2. Deep Learning Model

2.1 Basic structure

The deep learning model is actually a neural network model with multiple hidden layers. The raw data is abstracted layer by layer and finally classified. However, the common multi-layer neural network model is highly non-convex. In training, due to the existence of a large number of local best advantages and poor convergence, it is difficult to obtain good learning results, and the over-fitting phenomenon is too serious in practical applications. In the training of the deep learning model, a large amount of data is used to learn layer by layer on the unsupervised network, and then the parameters after training are used as the initial values of the parameter learning of the supervised neural network. The overall structure of the model is shown in Figure 1.

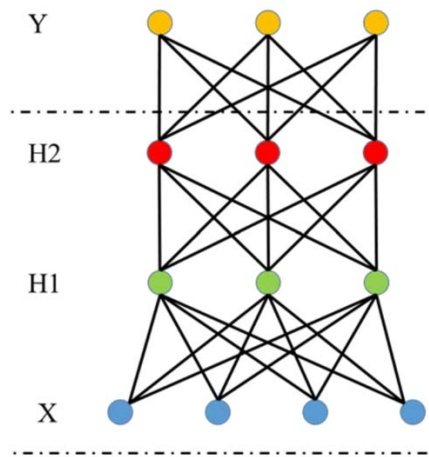


Figure 1 Basic structure of deep learning network

The deep learning network contains two hidden layers h1 and h2. In the layer-by-layer unsupervised learning, the first hidden layer h1 is obtained through the original data x, and then the second hidden layer h2 is obtained through the first hidden layer h1. After the unsupervised network is trained, the labeled samples are generally used for supervised learning by backpropagation algorithms.

2.2 Autoencoder model

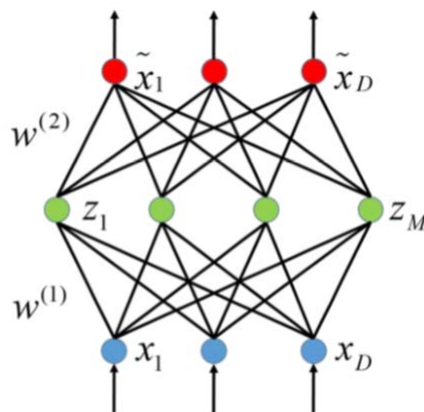


Figure 2 Schematic diagram of the Autoencoder structure

Autoencoder is a commonly used unsupervised network learning method and a special neural network model. The output of the network model is its input, as shown in Figure 2. After the hidden layer z is obtained by the encoder "encoding", the hidden layer can be "decoded" by the decoder to restore the original data. The optimization objective function of the model is:

$$L_{AE} = \frac{1}{N} \sum_{i=1}^N \left\| \tilde{\mathbf{x}}^{(i)} - \mathbf{x}^{(i)} \right\|_2^2$$

That is, make the predicted output of the model as equal as possible to the input. In terms of the transmission of information, this is a coding method that makes information loss as small as possible. The idea of the deep learning model based on the self-encoder is: for the original data, the coding layer h1 is obtained by learning from the encoder; then the self-encoder learning is performed on the code h1 to obtain the coding layer h2; thus obtaining the deep neural network layer by layer Each hidden layer. Then, the learned encoder coefficients and results are used as the initial values of the deep neural network model to learn the deep neural network.

2.3 Model training

In this paper, a robust Autoencoder is used to divide the training of the deep learning model into two parts. The first step is to use the noise-free Autoencoder to learn the hidden layer coefficients of the deep network layer by layer using the data without labels or removed labels. The second step is to use the coefficient obtained by the noise reduction Autoencoder learning as the initial value of the network coefficient, and use the data containing the label to perform supervised learning on the deep network through the back propagation algorithm.

The Mini-Batch descent method commonly used in deep learning model training is an optimized iterative algorithm based on the stochastic gradient descent method. The algorithm is that each parameter update does not follow a single sample, but instead calculates a gradient of parameter updates in a batch consisting of several samples:

$$w_{ij}^{(n)} := w_{ij}^{(n-1)} - \alpha \frac{\partial}{\partial w_{ij}} \sum_{n_k \in \text{Batch}(n)} E_{n_k}(\mathbf{w}^{(n-1)})$$

Therefore, the mini-batch method can be considered as a compromise between the ordinary gradient descent method and the stochastic gradient descent method. On the one hand, the efficiency of the algorithm iteration is guaranteed; on the other hand, in the case of equalization of various samples in the batch, the parameter update at each iteration is approximately toward the "optimal" direction. At the same time, compared with the stochastic gradient descent method, each iteration of the mini batch descent method uses multiple samples for calculation, making full use of the high efficiency of scientific computing software such as Matlab in vectorization calculation. In the mini-batch method, the number of samples in each batch is typically between 2 and 200.

3. Trading Strategy

In the short-term, it is possible to make directional predictions in the short-term, and capture short-term arbitrage opportunities. The deep learning-based trading strategy is to learn a large amount of historical transaction data by means of deep learning, and establish a predictive model to capture short-term trading opportunities in actual transactions. The deep-learning stock index futures trading strategy proposed in this paper first uses the deep learning model to predict the rise and fall of stock index futures in the short term, and then determines the trading signals of stock index futures based on the forecast results.

The first is the deep learning prediction model. The shorter the general prediction interval, the stronger the predictive power of the machine learning model. This article considers the 1-second high-frequency data of stock index futures. In the choice of predictive model input, the stock price in the short term, the range of price changes, the order price, etc. are selected. The output of the forecasting model is the direction of the ups and downs in the short term. Considering the rise and fall associated with the transaction, it is recommended to select samples with larger future fluctuations as samples of interest to the machine learning model. After the deep learning prediction model is trained, the future rise and fall can be judged based on the output of the prediction model.

The forecast results cannot be directly used to establish a trading strategy. There are two reasons for this: 1. the ups and downs are only a directional forecast. They do not indicate how big the probability of a rise or fall is, and whether the magnitude is large enough. 2. The fluctuation of stocks at high frequencies Limited, short-term trading is difficult to bring excessive returns. Based on these two considerations, the trading strategy in this paper has two characteristics. First, given a threshold, only the predicted score that meets or exceeds the threshold triggers the buy and sell signal. Second, the opening and closing positions in the transaction are triggered by the buying and selling signals. Although the deep learning model is a classification model, it is also possible to give score values for which the samples belong to each category. For the stock price change prediction model,

the rising predicted score Score1 and the falling predicted score Score2 can be obtained. In order to judge whether the stock price rises in the future, the sample with the score Score1 is larger, and the probability of future increase is greater. Correspondingly, the sample with the larger score Score2 has a higher probability of falling in the future. The threshold may be set according to the size of the score, so that when the score exceeds the threshold, the buying and selling signal is triggered. The multi-signal trigger threshold is BuyTrigger, and the short-signal trigger threshold is SellTrigger, then the buy and sell signal is as follows:

$$Signal = \begin{cases} 1 & Score1 > BuyTrigger \\ -1 & Score2 > SellTrigger \\ 0 & \end{cases}$$

The trading strategy is triggered by the buying and selling signals. Every day when opening, it is empty. Multiple positions and short positions are created by the trigger of the buy/sell signal, and the holding time is uncertain. If there is a short position trigger when holding multiple positions, immediately close the position and create a short position in reverse, otherwise does not change the holding position. If there is a long signal trigger when holding a short position, immediately close the position and create multiple positions in reverse, otherwise (if there is a short signal trigger), the position is not changed. Due to market noise and sudden changes, the predictive model pair can only accurately predict a part of the stock price change. Therefore, when the position is held, the stop loss strategy should be set. When the real-time price breaks through the stop loss line r , the stop loss is immediately closed. The stop loss price is calculated based on the stock price at the time of the most recent trading signal.

4. Empirical Analysis

4.1 Data selection

This paper selects the one-second high-frequency market of stock index futures in 2012 as the training set (in-sample data), and the stock index futures market from January 1, 2013 to April 18, 2014 as sample data. In order to introduce the dynamic change of the market price index over time in the input data, we add the samples of the first 4 moments of the sample considered at time t to the input variable sequence to form an extended input vector:

$$\bar{\mathbf{X}}_t = [\mathbf{X}_t, \mathbf{X}_{t-1}, \mathbf{X}_{t-2}, \mathbf{X}_{t-3}, \mathbf{X}_{t-4}]$$

The closing price of the previous trading day is the same as the data preprocessing process including the standardization of stock price data, and the smooth normalization of extreme data:

$$x_t^i := \frac{x_t^i - \min x^i}{\max x^i - \min x^i}$$

There are more than 3.9 million data in the sample, and the next second share price of about 1.1 million samples fell, the next second share price of about 1.1 million samples rose, and the next second price of the other 1.7 million samples did not change. The bilateral transaction costs considered in this paper are 20,000. In order to increase the difference of different types of data during model training, the data we selected is a sample with a stock price change of 0.04% or more at the next moment, with more than 11,000 samples. In addition, more than 11,000 samples with the same stock price in the next second were selected and added to the training set. Therefore, in the empirical analysis, there are 21,700 samples and 50 variables in the training set.

4.2 Price ups and downs forecast

The first hidden layer of the deep learning network has 200 nodes, the second hidden layer has 100 nodes, and the output layer has 2 nodes (outputs Score1 and Score2 in turn indicate that the predicted price is a rising or falling score). The number of hidden layers of unsupervised learning is 50, and the number of iterations with supervised learning is 400. On the Intel Xeon E5620, with a 2.4GHz processor, the model's training time is 15 minutes. When the model is applied, the prediction time of a single sample is about 1ms.

Table 1 Out-of-sample prediction results

	Y>0	Y=0	Y<0
Score1>Score2	1, 126, 825 (73. 9%)	950, 895	403, 163 (26. 3%)
Score1<Score2	397, 325 (26. 1%)	971, 699	1, 128, 936 (73. 7%)

The results of the out-of-sample prediction are shown in Table 1, where Y represents the actual stock price change. When the actual stock price is rising (Y>0), a 73.9% chance model will give a signal of price increase; when the actual stock price falls (Y<0), the model also has a 73.7% chance to give a signal of price decline. Therefore, the accuracy of model prediction is over 73%. If it is not as good as the transaction cost, it will be traded according to the output of the deep learning model (Score1>Score2 is long, Score1<Score2 is short, and the next second is immediately closed), then the average single transaction has a gain of about 0.0039%, far from Cover transaction costs.

4.3 Intraday trading strategy

In fact, only a very small number of trading opportunities, stock prices will fluctuate greatly. Among the more than 390,000 samples in the sample, only 11,000 samples have a stock price change of 0.04% or more, less than 0.3%. Therefore, the trading opportunity α selected in our strategy is around 0.1% (the long short position adds up to about 0.2%). Based on the predicted scores of all the data of the training samples, define the multi-signal threshold BuyTrigger and the short signal threshold SellTrigger as follows:

$$BuyTrigger := p(\text{Score1} > BuyTrigger) = \alpha$$

$$SellTrigger := p(\text{Score2} > SellTrigger) = \alpha$$

That is, there is a chance of α , the rising prediction score Score1 will be greater than the BuyTrigger, triggering the long signal; if there is a chance, the falling prediction score Score2 will be larger than the SellTrigger, triggering the short signal. In the case of $\alpha=0.1\%$, with $r=0.2\%$ loss line, take April 1st, 2014 as an example, a total of 9 times to do more and short signals, as shown in Figure 3, the red up arrow indicates Multiple signals, green down arrow indicates a short signal. One second after the 5 times of long signal is sent, the stock index futures price has risen 4 times and once fell; one second after the short signal is issued, the stock index futures price has fallen 3 times, one time unchanged. The long-short signal given by the prediction model is used for trading. Since the long-short signals are sent out at intervals, the number of transactions actually executed is also 9 times, of which 5 transactions have losses and 4 transactions have positive returns. After deducting transaction costs, the cumulative gain on the day was 0.26%.

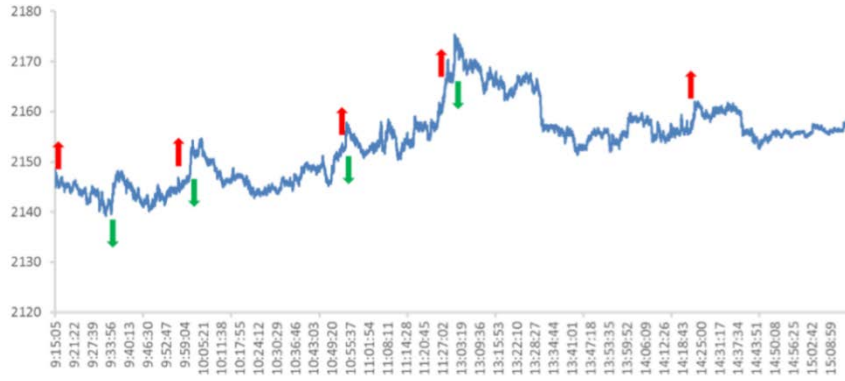


Figure 3 Daily stock index futures market and long short signal

Similarly, Figure 4 shows the long short signal given by the deep learning model on April 3, 2014. There are a total of 12 signals to do more short. One second after the 5 times of multi-signal transmission, the stock index futures price has risen 4 times, once unchanged; after 7 short-selling signals, the stock index futures price has dropped 6 times, one time unchanged. The trading is performed by the short-selling signal given by the prediction model. Since 9:21:04, 9:45:12 and 11:02:18 are short-selling when they are short-selling, they do not trade. 9:32:41 when the long signal is sent at the moment, it is already in the long position, so no operation is performed. The actual number of transactions executed is 8 times. The cumulative yield curve outside the sample is shown in Figure 5. The cumulative rate of return is 99.6%, the annualized rate of return is 77.6%, and the maximum retracement is -5.86%. Due to the limited trading capacity of high frequencies, the cumulative rate of return here is calculated on a simple basis. The average yield for a single transaction is 0.018%.



Figure 4 Intraday stock index futures market and long short signal

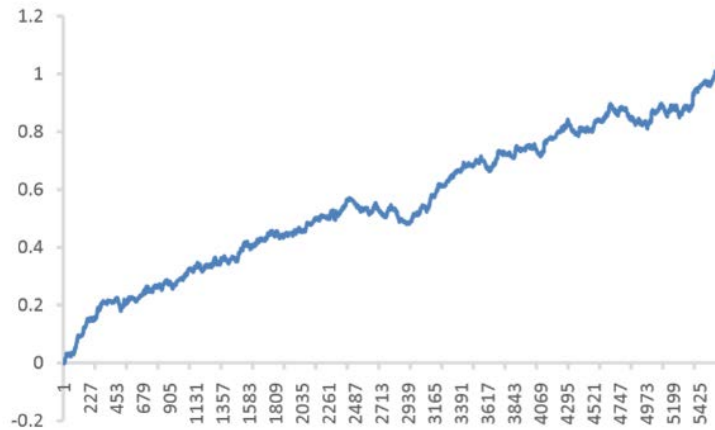


Figure 5 Cumulative yield curve outside the sample

It can be seen that this strategy is profitable at around 0.1%. Due to the existence of strict stop-loss measures, the winning rate of the transaction generally does not exceed 50%, and the odds are generally between 1 and 2. Therefore, the single-transaction yield is relatively low, relying on large trading frequencies to obtain profits. Moreover, it can be seen that after deducting the transaction cost of bilateral two thousand, the average single transaction yield is between 0.01% and 0.02%, which is greatly affected by the impact cost. In actual transactions, it is necessary to strictly control the impact cost.

5. Conclusion

As an effective means of big data modeling and analysis, deep learning has made remarkable achievements in the field of machine learning in recent years. This paper first proposes a high-frequency forecasting model of stock price based on deep learning. In the empirical calculation of stock index futures, the accuracy of 1 second high-frequency stock price forecast exceeds 73%. However, due to the small change in stock prices at high frequencies; it is not possible to directly obtain excess profits from a single one-second stock price forecast. Based on the deep learning prediction model, this paper then proposes an intraday trading strategy for high-frequency market data of stock index futures. Under the transaction cost of the second day, the annualized rate of return is 77.6%, and the maximum retracement is -5.86%.

References

- [1] A machine learning based stock forecasting and quantitative investment system [D]. Fang Puyi. Zhejiang University, 2018.
- [2] Application Research of Artificial Neural Network in Stock Forecasting [D]. Xiao Qi. South China University of Technology, 2017.
- [3] Research on stock price forecasting model based on correlation [D]. Ning Yuhe. Beijing University of Posts and Telecommunications, 2018.
- [4] Prediction of Shanghai and Shenzhen 300 Index Based on Convolutional Neural Network [D]. Chen Xiangyi. Beijing University of Posts and Telecommunications 2018.
- [5] Research on stock intelligent investment strategy based on support vector machine parameter optimization algorithm [D]. Gao Wen. Shanghai Normal University, 2018.
- [6] Predicting stock and stock price index movement using Trend Deterministic Data Preparation and machine learning techniques [J]. Jigar Patel, Sahil Shah, Priyank Thakkar, K Kotecha. Expert Systems with Applications. 2015 (1).
- [7] Integrating piecewise linear representation and weighted support vector machine for stock trading signal prediction [J]. Linkai Luo, Xi Chen. Applied Soft Computing Journal. 2013 (2).
- [8] Stock trading with cycles: A financial application of ANFIS and reinforcement learning [J]. Zhiyong Tan, Chai Quek, Philip Y.K. Cheng. Expert Systems with Applications. 2010 (5).
- [9] Textual analysis of stock market prediction using breaking financial news [J]. Robert P. Schumaker, Hsinchun Chen. ACM Transactions on Information Systems (TOIS). 2009 (2).
- [10] Thou shalt buy and hold [J]. Albert Shiryaev, Zuoquan Xu, Xun Yu Zhou. Quantitative Finance. 2008 (8).
- [11] A neural network with a case based dynamic window for stock trading prediction [J]. Pei-Chann Chang, Chen-Hao Liu, Jun-Lin Lin, Chin-Yuan Fan, Celeste S.P. Ng. Expert Systems with Applications. 2008 (3).