Journal of Finance and Accounting
2023; 11(2): 49-60
http://www.sciencepublishinggroup.com/j/jfa
doi: 10.11648/j.jfa.20231102.12
ISSN: 2330-7331 (Print); ISSN: 2330-7323 (Online)

Optimal Solution for the Gold Bitcoin Portfolio Investment Model

Zihan Yang, Guanhua Zhang, Chuming Liu

Queen Mary University of London Engineering School, Northwestern Polytechnical University, Xi 'an, China

Email address:
dly0707@163.com (Zihan Yang), jp2020304155@qmul.ac.uk (Guanhua Zhang), 2020304073@mail.nwpu.edu.cn (Chuming Liu)

To cite this article:
Zihan Yang, Guanhua Zhang, Chuming Liu. Optimal Solution for the Gold Bitcoin Portfolio Investment Model. *Journal of Finance and Accounting*. Vol. 11, No. 2, 2023, pp. 49-60. doi: 10.11648/j.jfa.20231102.12

Received: March 3, 2023; **Accepted**: March 31, 2023; **Published**: April 13, 2023

Abstract: This topic is a portfolio investment problem with quantitative trading as the background. In order to solve this problem, three types of mathematical models are used in this paper, namely the prediction model, decision model, and risk assessment model. The first is the forecasting model. The paper applies three forecasting models: the grey system Grach (1, 1) forecasting model, the quadratic exponential smoothing forecasting model, and the time series BP-neural network forecasting model. The second is the decision-making model. The decision-making model in the paper is a constrained linear programming model. The objective function is to maximize the total revenue of the day. Finally, there is the risk assessment model. The quantitative investment and multi-factor models are used in the paper to calculate the standard deviation of the rate of return (conventional risk) of gold and Bitcoin respectively in a 30-day cycle, so as to achieve the purpose of quantifying risk, thus reflecting the relationship between gold and Bitcoin. The investment risk index of the two futures products of the currency is provided as a reference for investors. This paper also adjusts the parameters of the prediction model, such as adjusting the value of the number of neurons in the hidden layer of the BP-neural network, to compare the fitting effects corresponding to different parameters, to prove that the prediction model is an optimal solution; Give the decision-making model a certain disturbance, such as changing the definition of the objective function for the total return of the day, to reflect the performance of the forecasting model in dealing with the disturbance of special factors. After that, this paper also conducts a sensitivity analysis of the decision-making model. The specific method is to give a small disturbance to the decision-making model, such as changing its transaction cost, that is, the value of the commission rate a%, recording the final benefit of the decision-making model, and generating a chart to reflect the model. smoothness and sensitivity. Finally, this paper optimizes some models, such as optimizing the BP-neural network model by adaptively adjusting the learning rate and optimizing the linear programming decision model by adding the MACD information factor.

Keywords: Exponential Smoothing, Time Series Model, Gray Model, Constrained Linear Programming Model, ARIMA Model, BP—Neural Network Model, MACD

1. Introduction

1.1. Background

Stock market trading is a way for people to make money. Market traders maximize their profits by frequently buying and selling volatile assets. Nowadays, the stock market changes frequently, and people want to plan their investment direction and investment strategy through a forecasting method. Changes in the market and the ups and downs of the stock market are not absolute, and the story is always changing. Therefore, investors must pay close attention to market trends when investing in stocks, and also pay attention to the impact of the introduction of relevant policies. follow market trends. Some traders are unfamiliar with the trading rules and cannot analyze the trading conditions. To maximize the interests of traders in the investment process and reduce the loss of assets and losses caused by wrong investments, a mathematical model is now established to predict the same type of stock market conditions. The stock market situation of the next day assists investors in investing [1, 2].

1.2. Statement of the Problem

Traders have 1,000 dollars and can maximize their profits by buying gold and bitcoin. We should build an accurate model to predict the price of gold and the price of bitcoin the next day. Specifically, our task is as follows: Build a model to predict what a certain product will do the next day. This model can more accurately predict market prices, and can intuitively send signals to investors to assist them in completing today's investment plans. Fit the data from previous years to determine if the predictive model meets the requirements. Establish a strategy that determines today's investment direction based on tomorrow's forecast data, maximizes the next day's benefits without losing money, and arrives at the most beneficial investment change strategy. Assess what the income of 1,000 dollars on September 11 2016, will be five years later on September 10, 2021.

1.3. General Assumption

1) Assume that the gold bitcoin trading market has no sudden situation.
2) Suppose only the price of this only decision-making factor is only considered.
3) Suppose the regular trading market information indicators still apply under the prediction model.
4) Assume that the cycle of the transaction is performed every day.
5) Defining the changes caused by the price changes caused by gold and Bitcoin during the morning, unity is today.

1.4. Paper Arrangement

Based on a comprehensive understanding of the problem, first, make some analysis of the problem. Then, a series of models are built and their introductions and mathematical derivations are given, and to improve accuracy and persuasion, the models are tested and programmed against the models to produce results and images. The next step is to confirm that the strategy under the established model is the best strategy through comparison, and then the conclusion is drawn. After concluding conclusions, the whole paper is summarized and sublimated, and finally, the advantages and disadvantages of the model are analyzed, the model is evaluated, its sensitivity is tested, and the model is improved.

2. Analysis of the Problem

2.1. Integrated

Before getting this problem, you should first understand that this problem is required to make decisions according to the transaction on the day of the transaction, so the online trading volume, transaction shares, etc., cannot be embodied in the model, which cannot be used as parameters. Planning of the back model. Therefore, this question can be divided into three stages, separately by three categories of mathematical models.

2.2. Prediction Model

The first category is a predictive model, this question should be to use the previously known data, to predict gold, and bitcoin price, only predict the latter price, to plan today's trading strategy, predict how many Typical models are available, and called, such as gray prediction model, GRACH (1, 1) model, LSTM, and BP neural network model, smooth index model, time- series model, etc. [1, 2].

2.3. Decision Model

The second category is a decision model. When the prediction model is established, it can determine today's transaction strategy through tomorrow, the price of gold, which can be used here to use a constrained linear planning model, balance risk, and earning mean-variance. And models such as machine learning.

2.4. Risk Quantization Model

The third category is a risk quantitative model. After a predictive model and decision model, a model that can quantify the risk is an indicator. It is necessary to invest in the risk, and can better make decisions in the case of smaller risk. This is also the supplementation and optimization of decision models, and the risk quantitative model can utilize the VAR risk value model, and CVAR conditional risk value model.

2.5. Model Assessment and Optimization

In the end, we also need to sensitivity analysis of the model and whether it is the optimal strategy, as well as the optimization of the model, such as the prediction model, which can integrate multiple predictive models, see which predictive model will make the benefits, such as Decision models, can reduce investment risks by joining market decision information indicators such as MACD and RSI, and can optimize our decision models through the ant colony algorithm, genetic algorithm, etc.

3. Calculating and Simplifying the Model

3.1. Exponential Smoothing Method

3.1.1. Single Exponential Smoothing

In quantitative trading, investors should use advanced mathematical models to guide investment, use computer technology to reduce the impact of their subjective emotions, and then make rational investment decisions to maximize profits [3]. Specifically, investors should establish prediction models and decision-making models to make more profits in their investment behavior. The Time series model is a common model used to quantify transactions. Through this model, we can use the known data to predict the trend of the stock market to a certain extent. There are many mathematical methods to realize this model. We will first choose the exponential smoothing method as the prediction means [8].

Before introducing exponential smoothing method, we need to understand the concept of moving average method. Set the observation sequence as $y_1, y_2, y_3 \dots y_T$, $N < T$, The calculation formula is:

$$M_t^{(1)} = \frac{\sum_{k=t-N+1}^{t} y_k}{N} = M_{t-1}^{(1)} + \frac{y_t - y_{t-N}}{N} \tag{1}$$

Similarly, the quadratic moving average formula is:

$$M_t^{(2)} = \frac{M_t^{(1)} + \cdots M_{t-N+1}^{(1)}}{N} = M_{t-1}^{(2)} + \frac{M_t^{(1)} - M_{t-N}^{(1)}}{N} \tag{2}$$

When the data fluctuates up and down a certain value, a prediction model can be established:

$$\hat{y}_{t+1} = M_t^{(1)} = \frac{1}{N}(y_t + \cdots y_{t-N+1}), t = N, N+1, \ldots T \tag{3}$$

The standard error of prediction is:

$$S = \sqrt{\frac{\sum_{t=N+1}^{T} (\hat{y}_t - y_t)^2}{T-N}} \tag{4}$$

In the one-time moving average model, the weight of the influence of the latest n-Period data on the prediction points is the same. However, generally speaking, the influence of historical data on the value of prediction points decreases with the increase of time interval. Therefore, it is more reasonable to give greater weight to the point closer to the prediction point. The exponential smoothing method can meet this demand.

Based on the moving average model, we introduce the weighting coefficient α. The formula of primary exponential smoothing is:

$$S_t^{(1)} = \alpha y_t + (1-\alpha) S_{t-1}^{(1)} = S_{t-1}^{(1)} + \alpha\left(y_t - S_{t-1}^{(1)}\right) \tag{5}$$

Similar to moving average formula:

$$S_t^{(1)} = \alpha y_t + (1-\alpha) S_{t-1}^{(1)} \tag{6}$$

Using this method to predict is the one-time exponential smoothing method. The model is:

$$\hat{y}_{t+1} = \alpha y_t + (1-\alpha)\hat{y}_t \tag{7}$$

It is worth noting that the selection of weighting parameter α will affect the prediction results to a certain extent. The larger the α, the greater the proportion of new data and the smaller the proportion of the original predicted value. In fact, the size of α reflects the magnitude of the correction. The greater the α, the greater the magnitude of the correction; conversely, the smaller the magnitude of the correction. When we choose α as 1, $\hat{y}_{t+1} = y_t$. The predicted value of the next period is equal to the observed value of the current period. When $\alpha = 0$, $\hat{y}_{t+1} = \hat{y}_t$, the observation value of the next period is equal to the observation value of the current period. In these two extreme cases, it is difficult for us to obtain reliable prediction values. Therefore, we need to select the appropriate value of α [3].

When we use the one-time exponential smoothing method for prediction, we should not only select the appropriate weighting coefficient but also determine the initial value s_0. When there are more data in the time series, the initial value has little influence on the future predicted value. However, if there is less time series data, the initial value will have a great impact on the predicted value. At this time, we must carefully study how to determine the initial value. Generally speaking, when there are few data, we can use the average value of the first few data as the initial value [9].

3.1.2. Quadratic Exponential Smoothing Method

The first-order exponential smoothing method introduces the weighting coefficient based on the moving average method, which makes the prediction result more accurate [11]. However, when the change of time series shows a straight-line trend, the one-time exponential smoothing method has a certain lag deviation. Therefore, we use the primary exponential smoothing method for the results of the primary exponential smoothing method to get more accurate prediction results, which is the secondary exponential smoothing method. Its calculation formula is:

$$S_t^{(1)} = \alpha y_t + (1-\alpha) S_{t-1}^{(1)} \tag{8}$$

$$S_t^{(2)} = \alpha S_t^{(1)} + (1-\alpha) S_{t-1}^{(2)} \tag{9}$$

$S_t^{(1)}$ is the smoothing value of the primary exponent, $S_t^{(2)}$ is the smoothing value of quadratic exponent. The prediction model is:

$$\hat{y}_{t+m} = a_t + b_t m, m = 1,2, \ldots \tag{10}$$

$$a_t = 2S_t^{(1)} - S_t^{(2)} \tag{11}$$

$$b_t = \frac{\alpha}{1-\alpha}\left(S_t^{(1)} - S_t^{(2)}\right) \tag{12}$$

3.2. Gray Model

3.2.1. Gray Model Establishment Method

The Gray model, abbreviated as GM model, is to establish a grey differential prediction model through a small amount of incomplete information to make a fuzzy long-term description of the development law of things. Building a grey model requires four steps [10]:

1) Accumulation generation, which generates the irregular original sequence $x^{(0)}$ into a regular sequence $x^{(1)}$.

Primitive sequence is $x^{(0)} = \left(x^{(0)}(1), x^{(0)}(2), \ldots x^{(0)}(n)\right)$

One time accumulation generation sequence (1-AGO) is $x^{(1)} = \left(x^{(1)}(1), x^{(1)}(2), \ldots x^{(1)}(n)\right)$

2) Make a moving average of the $x^{(1)}$ by accumulation to generate a new number sequence $z^{(1)}$.

$x^{(1)}$ have a mean generation sequence $z^{(1)} = (z^{(1)}(2), z^{(1)}(3) \ldots, z^{(1)}(n))$, $z^{(k)} = 0.5x^{(1)} + 0.5x^{(1)}(k-1), l = 2,3 \ldots n$.

3) According to the solution method of differential equation, the model GM (1,1) is obtained.

Establish the grey differential equation model

$$x^{(0)}(k) + az^{(1)}(k) = b, k = 2,3, \ldots n.$$

The corresponding whitening differential equation is

$$\frac{dx^{(1)}}{dt} + ax^{(1)}(t) = b$$

Record μ as $[a,b]^T$, $Y = [x^{(0)}(2), x^{(0)}(3) \ldots x^{(0)}(n)]$,

$$B = \begin{pmatrix} -z^{(1)}(2) & 1 \\ -z^{(1)}(3) & 1 \\ \dots & \dots \\ -z^{(1)}(n) & 1 \end{pmatrix}.$$

According to the least square method, if $J(u) = (Y - Bu)^T (Y - Bu)$ takes the minimum value, μ estimate is

$$\hat{u} = [\hat{a}, \hat{b}]^T = (B^T B)^{-1} B^T Y$$

solve this equation:

$$\hat{x}^{(1)}(k+1) = \left(x^{(0)}(1) - \frac{b}{\hat{a}}\right) e^{-\hat{a}k} + \frac{b}{\hat{a}}, k = 0,1, \dots n-1, \dots \tag{13}$$

Get the estimated value of the original data according to the result of the equation.

3.2.2. Prediction Accuracy Grade Judgment

1) Posterior error ratio C: Concentration degree of difference between predicted value and actual value.
2) Error probability P: It is the ratio of the difference between the residual and the mean value of the residual less than 0.647Sx's number to the total.

Table 1. *Accuracy evaluation criteria of GM (1,1) model.*

Degree of predicted precision	P	C
Level 1 (good)	> 0.95	< 0.35
Level 2 (eligible)	> 0.80	< 0.50
Level 3 (basic eligibility)	> 0.70	< 0.65
Level 4 (disqualification)	≤ 0.70	≥ 0.65

3.3. Time Series Forecasting Model

3.3.1. Introduction to Time Series

The basic characteristics of time series include trend, serial correlation, and randomness. Trend refers to the overall monotonicity of the series, such as stable, rising, or falling; serial correlation refers to the linear correlation between the current series value and one or some previous series values; Deterministic since the model cannot capture all the features in the real world, there will always be some noise, which we call white noise [2].

The Mathematical expression of time series: For example, the time series value at time T can be expressed as: XT a time series can be expressed as $\{X_T | t = 1,2,3, \dots, n\}$, the sequence value at time T-1 can be expressed as X_{T-1} or $X[t-1]$. The difference of the time series can be expressed as first order difference $X_T - X_{T-1}$ second order difference $\nabla X_T - \nabla X_{T-1} = (X_T - X_{T-1}) - (X_{T-1} - X_{T-2})$ k-order difference $X_T - X_{T-K}$. Of course, a time series must also have a measure of serial correlation. Here, the autocorrelation coefficient and the partial autocorrelation coefficient are used to measure the correlation inertia between the series. The autocorrelation coefficient (ACF), also known as the total correlation coefficient, is used to measure the degree of correlation of the same event in different periods [2]. The calculation formula of the autocorrelation coefficient is $\rho_h = \frac{r(h)}{r(0)}$, where h is the period set by a time series, $r(h)$ is the covariance function of the h period, and $r(0)$ is the variance. Partial autocorrelation coefficient (PACF), also known as conditional correlation coefficient, is used to measure the degree of correlation after removing the influence of intermediate variables. Assuming X_{t-2} *and* X_t is related by X_{t-1} PACF is the degree of correlation between the two after removing X_{t-1} the correlation. The lag K-order partial autocorrelation coefficient $X_{t-1}, X_{t-2}, \dots, X_{t-k+1}$ refers to the correlation measure of the influence of given the intermediate K-1 random variables (after removing the interference of the intermediate K-1 random variables) [12]. Correlation measure of the effect of X_{t-k} on X_t. The derivation formula is:

$$\varphi_{11} = \rho_1 \varphi_{kk} = \frac{\rho_k - \sum_{j=1}^{k-1} \varphi_{k-1,j} \rho_{k-j}}{1 - \sum_{j=1}^{k-1} \varphi_{k-1,j} \rho_j}$$

Commonly used traditional time series models include white noise model, autoregressive model, moving average model, autoregressive moving average model, etc. Here we use the autoregressive moving average model ARIMA.

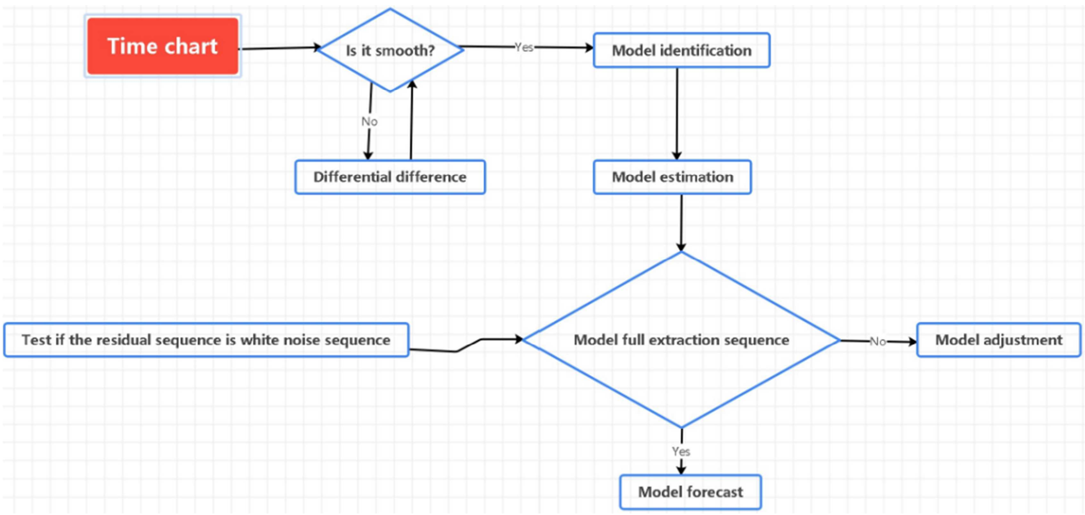

Figure 1. *Time Series Forecasting Model (ARIMA) modeling steps.*

3.3.2. Traditional Time Series Forecasting Models

The evaluation indicators of the ARIMA prediction model are ME (error), MAE (absolute error), MAPE (percentage error), MSE (mean squared error sum of squares, and RMSE (standard deviation). The model is also known as the Box-Jenkins model, the following are the modeling steps:

1) Stationarity test, here the unit root test (ADF) is used, and the null hypothesis is that there is at least one unit root in the sequence. The alternative hypothesis is that there is no unit root. Here, multiple parameters can be tested, such as T-test result t, significance p, and AIC value, etc. Analyze the t-value to see if it can significantly reject the null hypothesis that the series is not stationary (p_i 0.01 or 0.05).
2) If it is stable here, continue the model identification, otherwise enter the difference, and find that our model ACF index is decreasing, PACF first-order truncation, so it is the ARMA (1,0) model, and the original sequence is ARMA (1,1,0) Model.
3) Finally, estimate and predict the model.

3.3.3. Time Series-BP Neural Network Model

The learning process of the BP neural network consists of two processes: forward propagation of signals and backpropagation of errors. During forward propagation, the input samples are passed in from the input layer, processed layer by layer in the hidden layer, and then transmitted to the output layer. If the actual output of the output layer does not match the expected output, turn to the back-propagation stage of the error. The backpropagation of the error is to pass the output error back to the input layer by layer through the hidden layer in a certain form, and apportion the error to all units of each layer, so as to obtain the error signal of each layer unit, and this error signal is used as a correction [13]. The basis for the weight of each unit. BP network consists of the input layer, and output layer, and hidden layer. The structure diagram of the BP neural network is as follows:

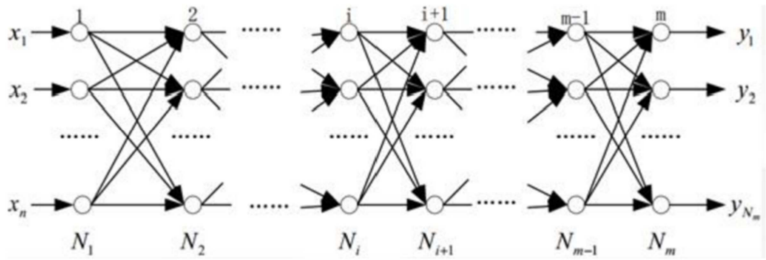

Figure 2. *The structure of BP neural network.*

The following is the mathematical derivation of BP neural network.

First define the variables as follows

Input vector: $x = (x_1, x_2,\cdots, x_n)$

Hidden layer input vector: $h_i = (h_{i1}, h_{i2}, \ldots, h_{ip})$

Hidden layer output vector: $h_o = (h_{o1}, h_{o2,\ldots,}h_{op})$

Output layer input vector: $y_i = (y_{i1}, y_{i2}, \ldots, y_{iq})$

Output layer output vector: $yO = (yO_1, yO_2, \ldots, yO_q)$

Expected output vector: $d = d_1, d_2, \ldots, d_q$

The connection weight between the input layer and the intermediate layer: W_{ih}

Continuous weights of hidden layer and output layer: w_{ho}

Threshold of each neuron in the hidden layer: b_h

Threshold of each neuron in the output layer: b_o

Number of sample data: $k = 1,2, \ldots, m$

Activation function: $f(.)$

Error function: $e = \frac{1}{2}\sum_{o=1}^{q}[d_o(k) - yo_o(k)]^2$

The first step, network initialization - assign each connection value separately, in random numbers in the interval $(-1, 1)$, set the error function e, given the calculation.

The precision value and the maximum number of learning times M.

The second step is to randomly select the Kth input sample and the corresponding expectation output:

$$x(k) = (x_1(k), x_2(k), \cdots, x_n(k))$$
$$d(k) = \left(d_1(k), d_2(k), \cdots, d_q(k)\right)$$

The third step is to calculate the input and output of each neuron in the hidden layer:

$$\mathrm{hi_h(k)} = \sum_{i=1}^{n} \mathrm{W_{ih}x_i(k) - b_h, h = 1,2, \cdots, p}$$
$$\mathrm{ho_h(k) = f(hi_h(k)), h = 1,2, \cdots, p}$$
$$\mathrm{yi_o(k)} = \sum_{h=1}^{p} \mathrm{w_{ho_o}ho_h(k) - b_o, o = 1,2, \cdots, q}$$
$$\mathrm{yo_o(k) = f(yi_o(k)), o = 1,2, \cdots, q}$$

The fourth step is to use the expected output and actual output of the network to calculate the error. The partial derivative of the difference function to each neuron in the output layer, according to Derivative rule of compound function:

$$\frac{\partial e}{\partial w_{ho}} = \frac{\partial e}{\partial yi_o}\frac{\partial yi_o}{\partial w_{ho}}$$

$$\frac{\partial yi_o}{\partial w_{ho}} = \frac{\partial(\sum_h^p w_{ho}ho_h(k)-b_o)}{\partial w_{ho}} = ho_h(k)$$

$$\frac{\partial e}{\partial yi_o} = \frac{\partial\left(\frac{1}{2}\sum_{o=1}^{q}[d_o(k)-yo_o(k)]^2\right)}{\partial yi_o}$$

$$= -(\mathrm{d_o(k)} - \mathrm{yo_o(k)})\mathrm{yo_o'(k)}$$
$$= -(\mathrm{d_o(k)} - \mathrm{yo_o(k)})\mathrm{f'(yi_o(k))}$$

$$\delta_\mathrm{o}(\mathrm{k}) = (\mathrm{d_o(k)} - \mathrm{yo_o(k)})\mathrm{f'(yi_o(k))}$$

$$\frac{\partial e}{\partial yi_o} = -\delta_\mathrm{o}(\mathrm{k})$$

$$\frac{\partial e}{\partial hi_h(k)} = \frac{\partial e}{\partial ho_h(k)}\frac{\partial ho_h(k)}{\partial hi_h(k)}$$

$$= -\sum_{h=1}^{p} \mathrm{w_{ho}}(\mathrm{d_o(k)} - \mathrm{yo_o(k)})\mathrm{f'(yi_o(k))}$$

$$= -\left(\sum_{h=1}^{p} \delta_o(k) - yo_o(k)\right)f'(yi_o(k))\sum_{h=1}^{q} w_{ho}\frac{\partial ho_h(k)}{\partial hi_h(k)}$$

$$\delta_\mathrm{h}(\mathrm{k}) = \left(\sum_{h=1}^{q}\frac{\partial}{\partial hi_h(k)} = -\delta_\mathrm{h}(\mathrm{k}), \frac{\partial e}{\partial w_{ho}} = -\delta_\mathrm{h}(\mathrm{k})\mathrm{ho_h(k)}\right.$$

The fifth step is to use the o(k) of each neuron in the output layer and the hidden layer.

The output of each neuron is used to modify the connection weight $w_{ho}(k)$:

$$\Delta \mathrm{w_{ho}(k)} = -\mu\frac{\partial e}{\partial w_{ho}} = \mu\delta_\mathrm{h}(\mathrm{k})\mathrm{ho_h(k)}$$

$$w_{ho}^{N+1} = w_{ho}^{N} + \eta\delta_\mathrm{o}(\mathrm{k})\mathrm{ho_h(k)}$$

The sixth step is to use the h(k) of each neuron in the hidden layer and the input layer. The input of each neuron modifies the connection weights, and in the final formula represent the learning rate, which is to adjust the step size, to prevent the value from being too large to cause non-convergence.

$$\Delta w_{ih}(k) = -\mu\frac{\partial e}{\partial w_{ih}} = -\mu\frac{\partial e}{\partial hi_h(k)}\frac{\partial hi_h(k)}{\partial w_{ih}} = \delta_h(k)x_i(k)$$

$$w_{ih}^{N+1} = w_{ih}^{N} + \eta\delta_h(k)x_i(k)$$

The seventh step, calculate the global error:

$$\mathrm{E} = \frac{1}{2m}\sum_{k=1}^{m}\sum_{o=1}^{q}[d_o(k) - yo_o(k)]^2$$

The last step, judge whether the network error meets the requirements, when the error When the preset accuracy is reached or the number of learning times is greater than the set maximum number of times, then ends the algorithm. Otherwise, select the next learning sample and the corresponding period. Looking forward to the output, returning to the third step, and entering the next round of learning.

For this BP neural network model, some parameters need to be set. Our team chose to set the hidden layer neurons to 20, the maximum number of training steps to 500, the maximum training iteration process to 50, and the training accuracy and learning rate to be set. Set to 0.01, Setup Division of Data for Training, Validation, and Testing is set to 70%, 15%, and 15% respectively.

3.4. Linear Programming Model with Constraints

In the above contents of this section, we mainly introduce three prediction models. Through the information provided by these prediction models, we can make more informed choices in the decision-making process. Next, we will introduce a programming model, that is, the linear programming model with constraints [14].

In order to achieve the planning goal of maximizing investment interests from 16 to 21 years, we will pursue the maximization of Daily interests.

$$maxZ = a_{i+1}(x_i + \Delta x) + c_{i+1}$$

$$c_{i+1} = c_i - \alpha_{\text{gold}}\, a_i|\Delta x| - \alpha_{\text{bitcoin}}\, b_i|\Delta y| - a_i\Delta x - b_i\Delta y$$

$$\begin{cases} c_{i+1} \geq 0 \\ \Delta x \geq -x_i, \Delta y \geq -y_i \end{cases} \tag{14}$$

Define the investor's daily status as $[c_i, a_i, b_i]$, Among them c_i is cash holdings, a_i is gold holdings and b_i is bitcoin holdings. Define $a_i x_i + b_i y_i + c_i$ as the total assets of investors. $\Delta x, \Delta y$ are the changes in the number of gold and bitcoin. When they are positive, they buy, and when they are negative, they sell. Assuming that the prediction results are more accurate, we can determine the amount of bitcoin or gold bought or sold by listing the objective function and conditions of linear programming [4, 5].

4. The Model Results

Using the quadratic exponential smoothing method, we get a good fitting effect:

Table 2. *Golden Unit Root Test (ADF) Test Results.*

ADF inspection form							
variable	Differential order	t	p	AIC	critical value		
					1%	5%	10%
USD (PM)	0	-0.434	0.904	9957.828	-3.436	-2.864	-2.568
	1	-8.159	0.000***	9948.534	-3.436	-2.864	-2.568
	2	-12.877	0.000***	9993.297	-3.436	-2.864	-2.568

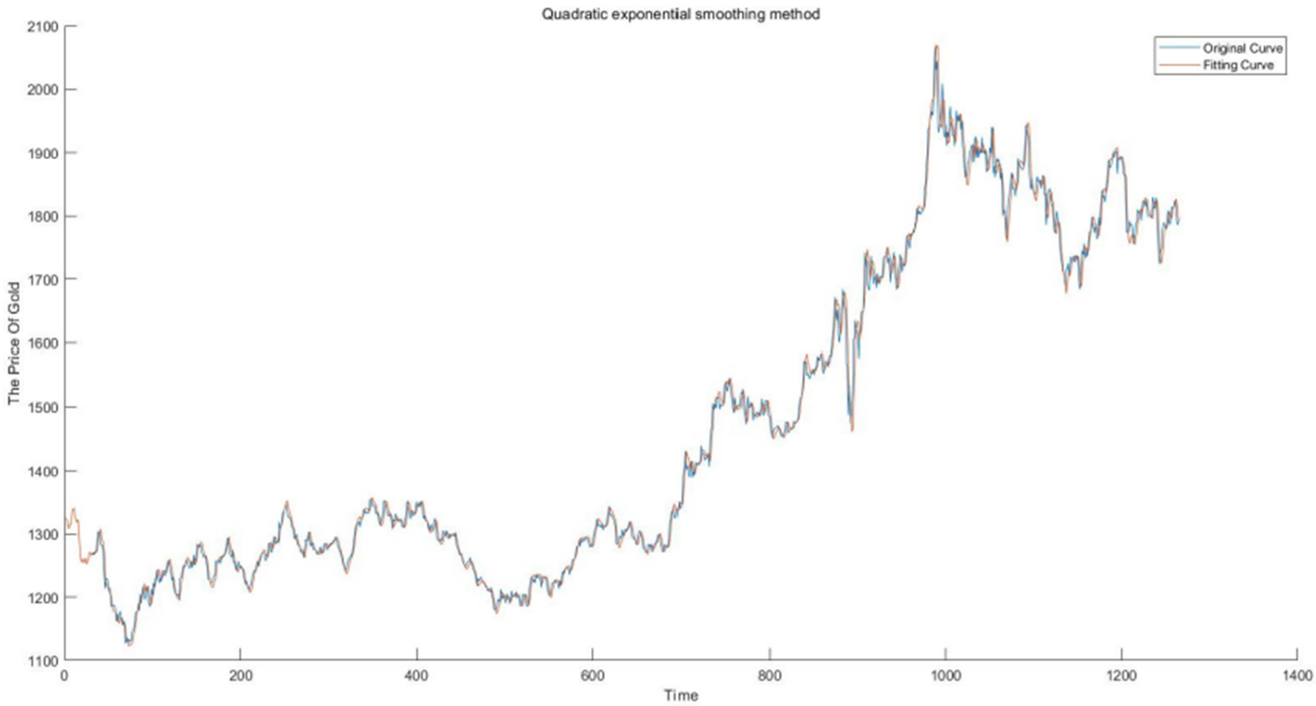

Figure 3. *Gold price fitting.*

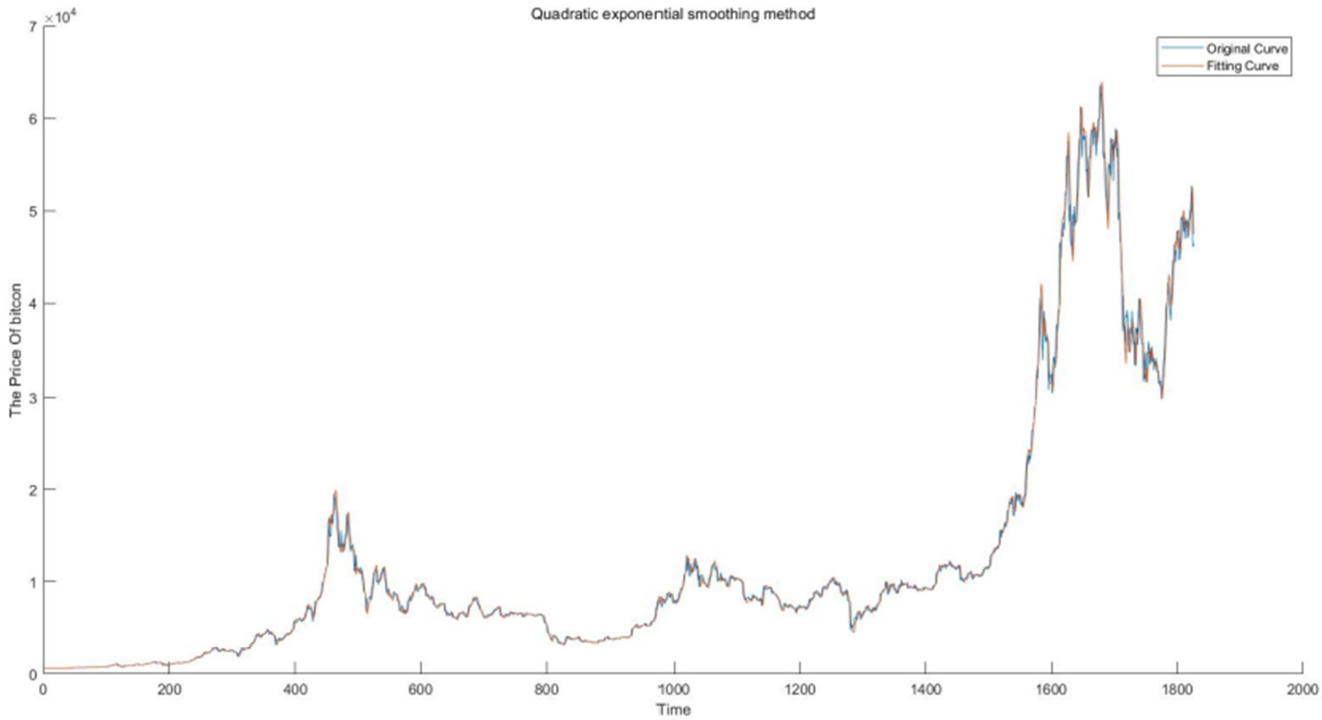

Figure 4. *Bitcoin price fitting.*

Table 3. *Bitcoin Unit Root Test (ADF) Test Results.*

ADF inspection form							
variable	**Differential order**	***t***	***p***	**AIC**	**critical value**		
					1%	**586**	**10%**
Value	0	-0.238	0.934	29168.936	-3.434	-2.863	-2.568
	1	-8.535	0.000***	29151.816	-3.434	-2.863	-2.568
	2	-15.723	0.000***	29187.267	-3.434	-2.863	-2.568

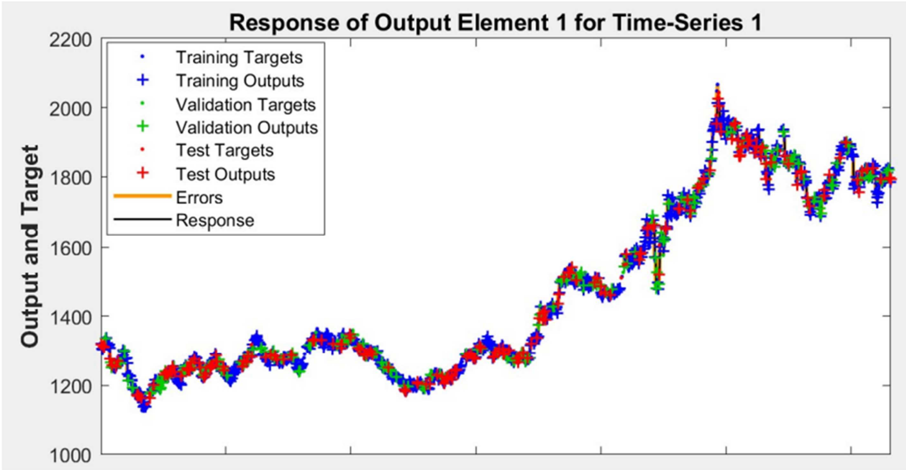

Figure 5. The output of the gold forecast result response of Output Element 1 for Time-Series1.

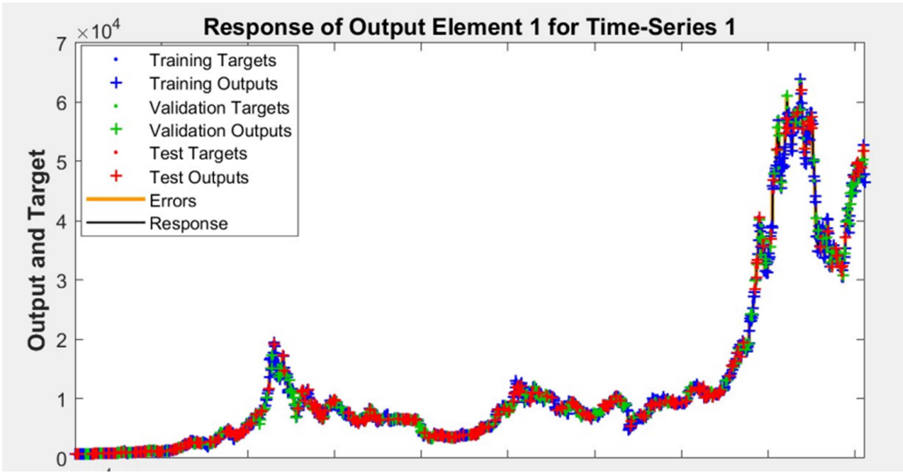

Figure 6. The output of the bitcoin forecast result response of Output Element 1 for Time-Series.

5. Validating the Model

In order to confirm that our model is a relatively optimal model, taking the time series-BP neural network prediction model as an example, we can adjust the corresponding parameters, such as the setting of the number of neurons in the hidden layer or the training, testing, validating the distribution's respective proportions, and then record the difference to see which set of parameters will make the difference the smallest. Under this model, the most intuitive way to reflect the difference is to observe the R^2 value of the training data and the MSE of the training data (Root mean square error) [4]. Taking the data of gold as an example, the following table is obtained by adjusting the parameters through experiments:

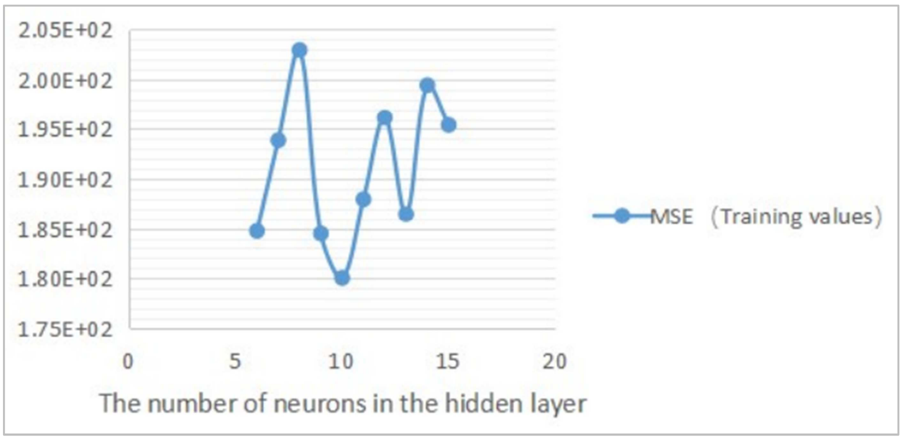

Figure 7. The effect of number of neurons in the hidden layer on MSE.

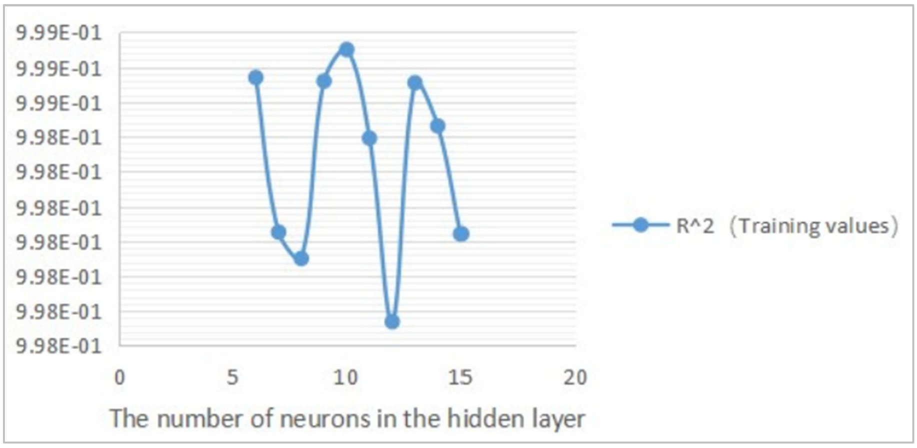

Figure 8. *The effect of number of hidden neurons on* R^2.

From the above picture, we can see that when the number of neurons in the hidden layer is 10 and when testing, validating the distribution's respective proportions is 70:15:15, it corresponds to the smallest MSE value and the closest R^2 to 1 value, MSE stands for Root mean square error, the smaller the value, the better the fitting effect, and the R^2 is the closest to 1, which means that the difference with the original data is the smallest, that is, the more accurate the prediction.

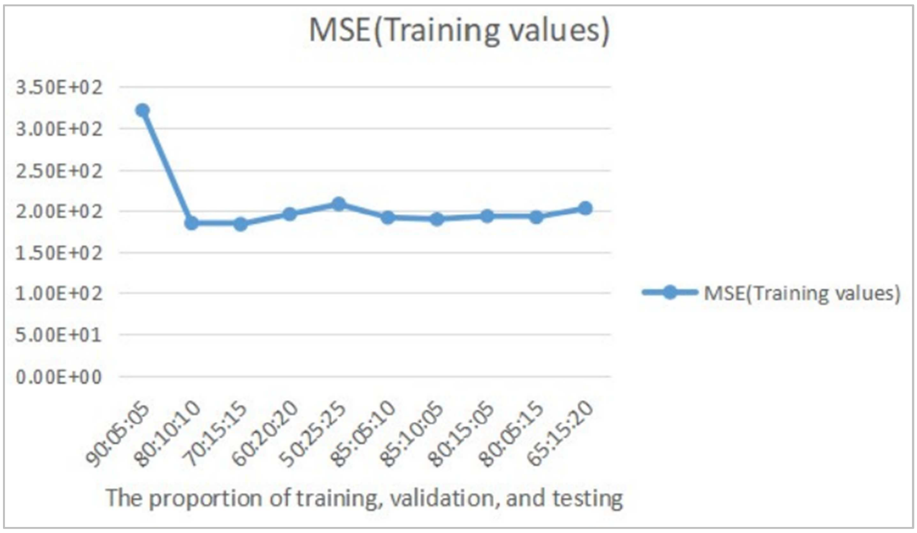

Figure 9. *The effect of the proportion of training, validation, and testing on MSE.*

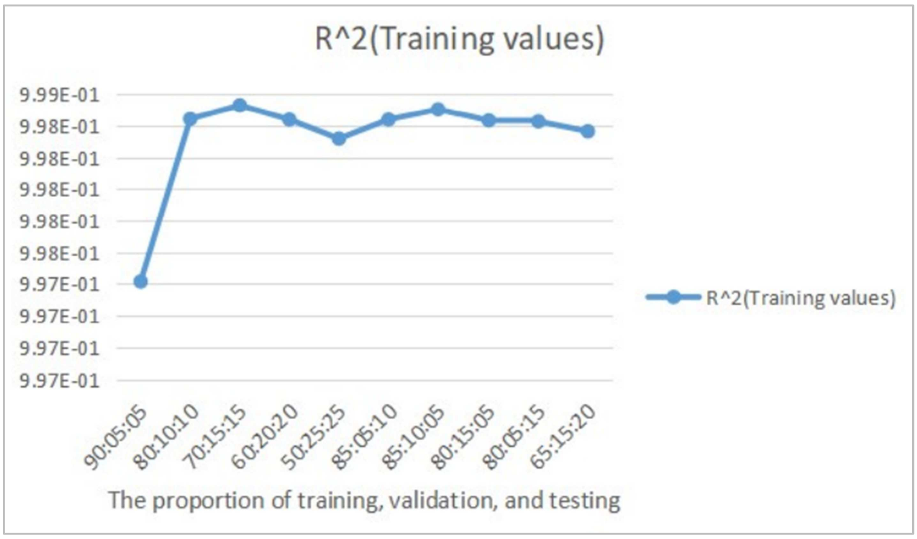

Figure 10. *The effect of the proportion of training, validation, and testing on* R^2.

For our decision model, we use a constrained linear programming model, and our objective function is the maximum return, where we define the total return obtained through trading on the day and the change in the price of gold and Bitcoin on the next day minus the Today's commission is today's pure profit. This strategy allowed us to earn 15 million US dollars in the final investment and profit. When we define the objective function as the total income obtained through trading on the day minus today's commission, the final profit The target function is only 705,000 dollars, and we try to define the objective function as the total income through the next day's change in the price of gold and bitcoin minus today's commission. This strategy finally earned a total of 312,200 dollars. Therefore, our decision-making model is relatively perfect.

6. Evaluate of the Mode

6.1. Sensitivity Analysis

By adjusting the commission rate $\alpha\%$ of gold and Bitcoin in the decision-making model, the following table is obtained. From this, it can be concluded that the sensitivity of the model is high, and the decision-making will change the frequency of buying and selling according to the adjustment of $\alpha\%$ [5].

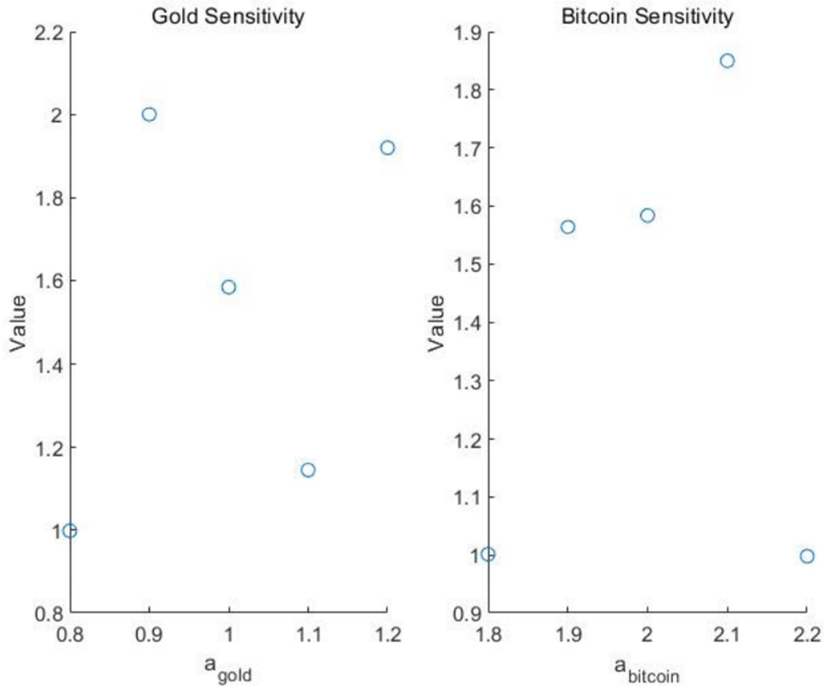

Figure 11. *The relationship between commission rate $\alpha\%$ and sensitivity in the decision-making model (Gold: left, Bitcoin: right).*

Note: the value unit of ordinate is 10000000. Through the sensitivity analysis of the model, it is found that when the $\alpha\%$ value is slightly changed, the final investment value changes greatly, indicating that the investment decision model has better sensitivity [5].

6.2. Risk Assessment Model

Here, quantitative investment and multi-factor models are used to define and quantify risk. Markowitz describes risk as the standard deviation of the rate of return. The formula for risk definition is: conventional risk $\sigma_i = \sqrt{E[(r_i - \mu_i)^2]}$ where r_i and u_i represent the return and return of asset i, respectively mean value of i. Semivariance: ${\sigma_h}^2 = E[(\min(r_i - u_i), 0)^2]$, downside volatility: $\sigma_h = \sqrt{E[(\min(r_i - u_i), 0)^2}$. Here, the conventional risk, that is, the standard deviation of the rate of return, is used to estimate the risk of the gold and bitcoin trading markets, respectively. We take 30 days as a cycle and get the standard deviation of the rate of return of the original data as shown below:

Analysis of the graph shows that the standard deviation of the rate of return of Bitcoin is much larger than that of gold, so risks should be appropriately considered when conducting Bitcoin transactions under the decision-making model [7].

6.3. Time Series-BP Neural Network Model Optimization

For the improvement of the neural network model, we can start from two aspects: The first point is to increase the momentum term, that is, add the previous change after the update formula. The formula is as follows: $\Delta w(t) = \eta \cdot \delta \cdot o + \alpha \cdot \Delta w(t-1)\, \Delta w(t-1)$, is the amount of change in the last iteration, as is the update of the bias value. The second point is that we can optimize the model through adaptive adjustment of the learning rate [3, 4]. The learning rate determines the accuracy of the model to a certain extent. If the learning rate is too large, the accuracy will be poor, and it will not even converge. If the learning rate is too small, the convergence will be slow. If this iteration makes the value E of the error function decrease, increase the learning rate

$\eta = \beta \cdot \eta, \beta > 1$, if this iteration makes the value E of the error function increase, then this adjustment Invalid, and reduce the learning rate $\eta = \alpha \cdot \eta, (0 < \alpha < 1)$.

6.4. Linear Programming Decision Model Optimization

Our decision models are mathematically idealized models that do not adequately account for contingencies or risks that may occur in financial markets. Therefore, to optimize the decision-making model, an information indicator MACD is added for optimization. The MACD indicator is a trend indicator based on the principle of moving average construction, and the closing price is smoothed.

When MACD is 0, that is, when DIF crosses DEA, it means that the trend of DIF changes and becomes stronger. At this time, it is displayed as a red column, which is a buy signal [6].

When MACD is 0, that is, when DIF breaks below DEA, it means that the trend change of DIF is weakening. At this time, it is displayed as a green column, which is a sell signal [6].

7. Strengths and Weaknesses

7.1. Predictive Model

7.1.1. Grey System Prediction GM (1,1) Model

1) Strengths

The matrix is mainly used in the calculation process, and he solved the problems encountered in the calculation. The gray prediction program written is simple and practical, easy to operate, and has high prediction accuracy. Does not require typical probability distributions; reduces the randomness of time series; can calculate small samples; easy to calculate.

2) Weaknesses

There is a strong dependence on historical data, the relationship between various factors is not considered, and the error is too large.

7.1.2. Quadratic Exponential Smoothing

1) Strengths

The calculation is simple, the sample requirements are small, the adaptability is strong, and the results are relatively stable.

2) Weaknesses

Given a smaller proportion in the long-term and a larger proportion in the near term, only short-term forecasts can be made.

7.1.3. Time Series ARIMA Model

1) Strengths

The characteristics, trends, and development laws of variable changes can be found from the time series, so as to effectively predict the future changes of variables [15].

2) Weaknesses

When using the time series analysis method for market forecasting, attention should be paid to future development and change laws and development levels of market phenomena, which may not be completely consistent with their historical and current development and changes.

7.1.4. Time Series BP Neural Network Model

1) Strengths

The BP neural network is a global approximation, so the overall performance is better. In addition, the number of hidden layers and the number of neurons in each layer can be increased infinitely, which makes the ability of the BP neural network unlimited.

2) Weaknesses

The convergence is slow, the accuracy is not high enough, and it may fall into a local minimum.

7.2. Decision Model

Constrained Linear Programming Model

1) Strengths

There is a unified algorithm, any linear programming problem can be solved, and a method for solving multivariate optimal decision-making.

2) Weaknesses

Transactions are carried out in units of days, lack of consideration for long-term investment, more inclined to mathematics, not considering economic benefits and less consideration for risks.

8. Conclusions

The data predicted by the three established prediction models, ARIMA, BP-neural network, quadratic exponential smoothing model, based on the linear programming decision model, and the multi-factor risk quantify. The matrix is mainly used in the calculation process, and he solved the problems encountered in the calculation. The gray prediction program written is simple and practical, easy to operate, and has high prediction accuracy. Does not require typical probability distributions; reduces the randomness of time series; can calculate small samples; easy to calculate.

References

[1] Wen Jing, Yin Chengyu, Liao Guowei, Wang Yuan, Zhang Jie. Application of GM (1,1) grey model to predict the incidence trend of thyroid cancer in China, Modern Oncology, mar. 2022.

[2] GDP prediction and analysis of Jiangsu Province Based on ARIMA model and BP neural network model [D]. Xu Mingyan Shandong University. 2020.

[3] Discussion on calculation method of regression coefficient of quadratic exponential smoothing prediction model [J] Chen Wu, Zhang Shanjiang, Hou Chunhua, Chen Chen, Zeng lichen Statistics and decision making 2016 (19).

[4] Mathematical modeling algorithm and application [M] National Defense Industry Press, Si Shoukui, 2011.

[5] Sensitivity analysis of changing constraints in linear programming [J] Liu Qijia, Xiao Yu Industry and Technology Forum 2021 (19).

[6] Application skills, defects and remedies of MACD index [J] Nie Shuyun Accounting of Chinese township enterprises. 2007 (04).

[7] Theoretical and empirical research on risk measurement method and financial asset allocation model [J] Wu Shi Nong, Chen Bin Economic research 1999 (09).

[8] 6.4.3.2. forecasting with single exponential smoothing. Available at: https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc432.htm (Accessed: March 20, 2023).

[9] Frost, J. (2021) Exponential smoothing for time series forecasting, Statistics By Jim. Available at: https://statisticsbyjim.com/time-series/exponential-smoothing-time-series-forecasting/ (Accessed: March 20, 2023).

[10] Author links open overlay panelJie Cui a b et al. (2012) A novel grey forecasting model and its optimization, Applied Mathematical Modelling. Elsevier. Available at: https://www.sciencedirect.com/science/article/pii/S0307904X12005835 (Accessed: March 20, 2023).

[11] Application of quadratic exponential smoothing and Markov chain in... (no date). Available at: https://www.researchgate.net/publication/355098269_Application_of_Quadratic_Exponential_Smoothing_and_Markov_Chain_in_Computer_Predicting_Total_Amount (Accessed: March 20, 2023).

[12] Time series forecasting: Definition, applications, and examples (no date) Tableau. Available at: https://www.tableau.com/learn/articles/time-series-forecasting (Accessed: March 20, 2023).

[13] Author links open overlay panelLihong Yu and AbstractTaking kidney bean as the research object (2022) Optimization of BP neural network model by chaotic krill herd algorithm, Alexandria Engineering Journal. Elsevier. Available at: https://www.sciencedirect.com/science/article/pii/S1110016822001223#:~:text=The%20BP%20neural%20network%20model%20%28Back%20Propagation%29%20is,for%20information%20processing%20and%20has%20good%20predictive%20effect. (Accessed: March 20, 2023).

[14] Linear programming - definition, formula, problem, examples (no date) Cuemath. Available at: https://www.cuemath.com/algebra/linear-programming/ (Accessed: March 20, 2023).

[15] Hayes, A. (2023) Autoregressive integrated moving average (ARIMA) prediction model, Investopedia. Investopedia. Available at: https://www.investopedia.com/terms/a/autoregressive-integrated-moving-average-arima.asp (Accessed: March 20, 2023).