Scientific Research and Essays Vol. 5(20), pp. 3164-3174, 18 October, 2010
Available online at http://www.academicjournals.org/SRE
ISSN 1992-2248 ©2010 Academic Journals

Full Length Research Paper

Prediction of engineering equipment support cost using a hybrid method based on Grey neural network and Markov chain

Jun Zhi[1,2]* and Jianyong Liu[1]

[1]Engineering Institute of Engineering Corps, PLA University of Science and Technology, Nanjing 210007, China.
[2]Zhenjiang Watercraft College, Zhenjiang 212003, China.

Accepted 17 August, 2010

On the basis of characteristic analysis of GM (1, 1) prediction model and Markov chain, we adopt the Grey prediction model to forecast the support cost of engineering equipment. It is suitable for time series prediction. Firstly, we improve relevant parameters of the model on the foundation of traditional GM (1, 1) model. Secondly, we map the improved gray prediction model into BP neural network with strong nonlinear processing ability and raise the nonlinear treatment processing ability of the model. Thirdly, we use the Markov chain method to amend the prediction result. The simulation result indicates that compared with traditional GM (1, 1) model, though the laminating degree to known data is not high in the model, the prediction precision improves a lot.

Key words: GM (1, 1), Grey neural network, Markov's chain, support cost.

INTRODUCTION

The support cost of engineering equipment means the sum of expenses relevant to using engineering equipment, including personnel's expenses, training expenses, oil fee, maintenance cost and transportation cost and other relevant expenses.

The support cost of engineering equipment will be accelerated by its increased service life. Though it does not cost much each year, the total cost is extraordinarily large, generally 3 to 20 times of the purchase expenses according to the statistical data. Therefore, the scientific prediction to support cost will be the decisive factor on reducing expenses. It has already become an important problem of the equipment efficiency. Thanks to annual change and supplement, the support cost of engineering equipment, forecast by the grey prediction model, will make very strong time effects. We adopt the Grey prediction model to forecast the support cost of engineering equipment. It is suitable for time series prediction.

*Corresponding author. E-mail: njzhijun@163.com.

ANALYSIS ON ALGORITHM

In the paper, we establish the prediction model based on GM (1, 1) method and revise the result with Markov chain. So we analyze the characteristics of GM (1, 1) model and Markov chain in advance.

Analysis on GM (1, 1) model

GM (1, 1) is the most frequently used grey dynamical prediction model, made up of first-order differential equation which merely includes single variable (Li and Yuan, 2009; Li and Xu, 2009).

Using GM (1, 1) model, we can get accurate prediction result even without complete information. It has characteristics of some necessary sample data and simple operation etc.

We carry on an accumulated generating operation to original series (Deng et al., 2002; Liu, 2004) in order to reduce the randomness and fluctuation of original data sequence $x^{(0)} = \{x^{(0)}(1), x^{(0)}(2), \cdots, x^{(0)}(n)\}$. Set the generated series as $x^{(1)}=\left\{x^{(1)}(1),x^{(1)}(2),\cdots,x^{(1)}(n)\right\}$.

$$x^{(1)}(k)=\sum_{i=1}^{k}x^{(0)}(i) \tag{1}$$

Considering exponential-growth regulation of the sequence $\left\{x^{(1)}(k)\right\}$, we can set up a differential equation of melting form as follows.

$$\frac{dx^{(1)}}{dt}+\alpha x^{(1)}=u \tag{2}$$

Where α and u are unidentified parameters, α is gamma, u is grey action. They respectively reflect the relationship of increase velocity and data change to original series.
Carrying on discrimination to differentiation term $dx^{(1)}/dt$, we get:

$$\begin{aligned}\frac{dx^{(1)}}{dt}&=\frac{\Delta x}{\Delta t}\\&=\frac{x^{(1)}(k+1)-x^{(1)}(k)}{k+1-k}\\&=x^{(1)}(k+1)-x^{(1)}(k)\\&=x^{(0)}(k+1)\end{aligned} \tag{3}$$

α and u can be obtained using least square method:

$$[\alpha,u]^{T}=(B^{T}B)^{-1}B^{T}y_{n} \tag{4}$$

$$B=\begin{bmatrix}\theta x^{(1)}(1)+(1-\theta)x^{(1)}(2) & 1\\ \theta x^{(1)}(2)+(1-\theta)x^{(1)}(3) & 1\\ \vdots & \vdots\\ \theta x^{(1)}(n-1)+(1-\theta)x^{(1)}(n) & 1\end{bmatrix} \tag{5}$$

$$y_{n}=\left[x^{(0)}(2),x^{(0)}(3),\cdots,x^{(0)}(n)\right]^{T} \tag{6}$$

We can get the discrete solution

$$\hat{x}^{(1)}(k+1)=[x^{(0)}(1)-\frac{u}{\alpha}]e^{-\alpha k}+\frac{u}{\alpha} \tag{7}$$

We can get the prediction model of primitive series:

$$\hat{x}^{(0)}(k+1)=(1-e^{a})[x^{(0)}(1)-\frac{u}{\alpha}]e^{-\alpha k} \tag{8}$$

Grey GM (1, 1) prediction model is particularly suitable for forecasting the support cost of engineering equipment which is characteristic of time series. The model learns law from data and predicts the numerical value successively at the next moment by the law. We can learn from the principle of above-mentioned GM (1, 1) model that before continuing calculation, parameter θ should be set at first, therefore, in the paper, we adopt the self-adaptation method to set the parameter. Owing to the model adopts exponential function to predict, the non-linear treatment ability is relatively weak, especially the prediction precision is low to known data with great fluctuation. We need to map the improved grey GM (1, 1) prediction model into BP neural network which has strong nonlinear processing ability and raise the nonlinear treatment processing ability of the model.

Analysis on Markov chain

Markov chain method is an approach of probability prediction. In accordance with this probability and immediate preceding term state, the system reckons a Markov's course whose time and state are dispersed through calculating transfer probability among the states, which is called Markov chain.

When the system state is at the moment of *t*, the system state at the moment *t*+1 is only related to the state at the moment of *t*, and has nothing to do with the previous state of *t*. The method is available to changing rule of system state from the known data. It adjusts predicted value by means of calculating the probabilities of state transition (Glenn, 1998; Simmons, 2005; Takagi and Furukawa, 2004; Prowell, 2004). Consequently, on the basis of studying the changing rule of the known data, we could adopt the way to adjust the prediction result of GNN prediction model aiming at improving the prediction precision.

PREDICTION MODEL CONSTRUCTION

Improvement of adaptability parameters

The traditional GM (1, 1) model acquiescently set the parameter θ of matrix *B* as 0.5. The model is suited to series prediction problem which is strictly monotonic. However, not all kinds of engineering equipment support cost are monotonic, it even changes remarkably. We introduce the improved method of adaptability parameters to improve the precision of prediction model. Link parameters θ and α together, while θ is changed

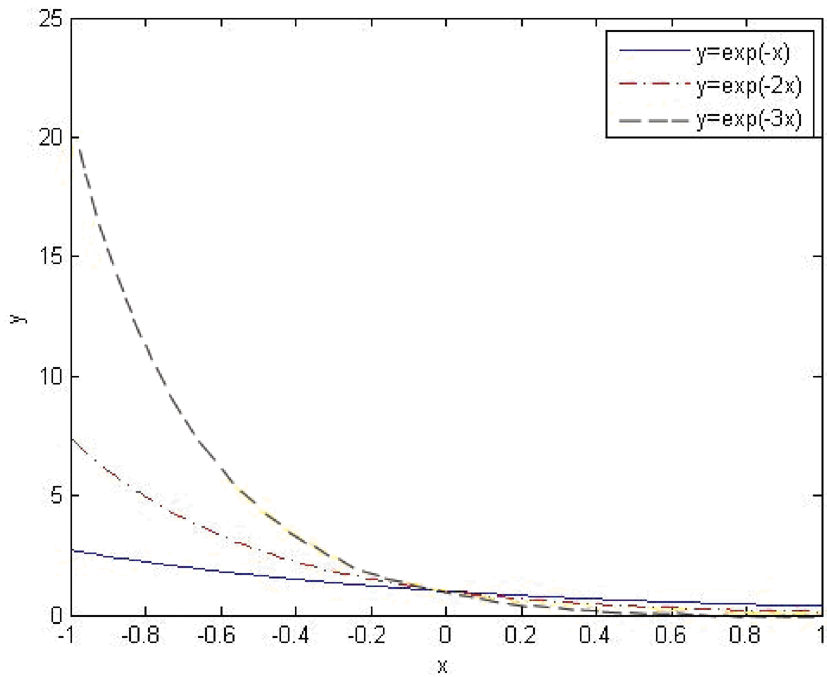

Figure 1. Change trend of exponential function.

with α. In Figure1, in line with the corresponding exponential function $x = Ce^{-\alpha x}$, we find that when $|\alpha|$ is smaller or larger, so is the curve distortion, as well as the prediction error.

The method selects different θ calculation matrix B according to different α, and we also use a personal judgment method.

$$\theta = \frac{1}{\alpha} - \frac{1}{e^{\alpha} - 1} \tag{9}$$

Set θ as 0.5 in advance, and set the initial value as $\theta(1)$. Getting the estimated values of model parameter α, we can take α into Equation 9, calculate the value of parameter θ and set it as $\theta(k+1)$, comparing with the size of $\theta(k+1)$ and $\theta(k)$. If the difference is larger than the threshold value given in advance, it means that there may be greater improvement in the prediction precision. We use $\theta(k+1)$ instead of $\theta(k)$ to calculate matrix B and parameter α; again construct the model and carry on the predicting operation until the change of two values is smaller than threshold value by iteration. At the moment, output the prediction result. If difference is smaller than threshold value at the beginning, we still use initial value to calculate the matrix B, construct the model and output the predicted value. In fact, compared with the size of $\theta(k+1)$, $\theta(k)$ is to check the deformation of curve indirectly. Due to large difference and much deformation, we need to adjust model parameters. Even with small difference and little deformation, the original parameters are still suitable.

Improve of initial data

In Equation 8, we find that, besides relating to u/a, the prediction result of traditional GM (1, 1) model depends on the initial data of time series as well. Selection of initial data may cause certain influence on the prediction result.

Prerequisite of choosing $x^{(0)}(1)$ as initial data is to suppose that prediction value of $x^{(0)}(1)$ is equal to original value. But actual conditions are not always like this. Prediction curve and curve of original series do not always cross at starting point. In light with this question, we can improve the initial data of traditional grey prediction model, and construct new initial data in order to change the systematic error of the model.

Construct new initial data $\beta x^{(0)}(1)$ and the corresponding prediction model as Equation 10 shows.

$$\hat{x}^{(1)}(k+1) = [\beta x^{(0)}(1) - \frac{u}{\alpha}]e^{-\alpha k} + \frac{u}{\alpha} \tag{10}$$

The optimization target is to make the variance of prediction value and actual value minimum. Define the variance of prediction result as follows.

$$R=\sum_{k=1}^{n}\left[\hat{x}^{(1)}(k)-x^{(1)}(k)\right]^2$$
$$=\sum_{k=1}^{n}\left[\left[\beta x^{(0)}(1)-\frac{u}{\alpha}\right]e^{-\alpha(k-1)}+\frac{u}{\alpha}-x^{(1)}(k)\right]^2 \quad (11)$$

We can get the analytical formula of β through the derivation of variance. The concrete derivation process is as follows.

$$\frac{dR}{d\beta}=\frac{d\left\{\sum_{k=1}^{n}\left[\left[\beta x^{(0)}(1)-\frac{u}{\alpha}\right]e^{-\alpha(k-1)}+\frac{u}{\alpha}-x^{(1)}(k)\right]^2\right\}}{d\beta}$$
$$=2\cdot\sum_{k=1}^{n}\left[\left[\beta x^{(0)}(1)-\frac{u}{\alpha}\right]e^{-\alpha(k-1)}+\frac{u}{\alpha}-x^{(1)}(k)\right]x^{(0)}(1)e^{-\alpha(k-1)}$$
$$=2\cdot[\beta\left(x^{(0)}(1)\right)^2\sum_{k=1}^{n}e^{-2\alpha(k-1)}-\frac{u}{\alpha}x^{(0)}(1)\sum_{k=1}^{n}e^{-2\alpha(k-1)}$$
$$+\frac{u}{\alpha}x^{(0)}(1)\sum_{k=1}^{n}e^{-\alpha(k-1)}-x^{(0)}(1)\sum_{k=1}^{n}x^{(1)}(k)e^{-\alpha(k-1)}]$$
$$=0$$

Both sides of the equation are divided by $2x^{(0)}(1)$, we can get:

$$\beta x^{(0)}(1)\sum_{k=1}^{n}e^{-2\alpha(k-1)}-\frac{u}{\alpha}\sum_{k=1}^{n}e^{-2\alpha(k-1)}+\frac{u}{\alpha}\sum_{k=1}^{n}e^{-\alpha(k-1)}-\sum_{k=1}^{n}x^{(1)}(k)e^{-\alpha(k-1)}$$
$$=\beta x^{(0)}(1)\sum_{k=1}^{n}e^{-2\alpha(k-1)}+\frac{u}{\alpha}\left(\sum_{k=1}^{n}e^{-\alpha(k-1)}-\sum_{k=1}^{n}e^{-2\alpha(k-1)}\right)-\sum_{k=1}^{n}x^{(1)}(k)e^{-\alpha(k-1)}$$

Let

$$b=\sum_{k=1}^{n}e^{-2\alpha(k-1)},\quad c=\sum_{k=1}^{n}e^{-\alpha(k-1)},\quad d=\sum_{k=1}^{n}x^{(1)}(k)e^{-\alpha(k-1)}$$

And

$$\frac{dR}{d\beta}=\beta x^{(0)}(1)\cdot b+\frac{u}{\alpha}(c-b)-d=0$$

We can get

$$\beta=\frac{\frac{u}{\alpha}(b-c)+d}{x^{(0)}(1)\cdot b} \quad (12)$$

When the parameter β is obtained, we can use improved initial data to predict the data of time series.

Mean value modification tactics

When the sample data curve is not smooth or the fluctuation is great, the prediction precision of Grey prediction model is relatively low. We must carry on the pretreatment to sample data and reduce the fluctuation of initial data (Trajkovic et al., 2003; Luo and Cui, 2005). In the paper, we adopt the method of changing original series into mean value series to reduce the randomness of series.

Therefore, it is necessary to prove the theoretical foundation of mean value modification model. We also need to explain that mean value of original series is helpful to reduce fluctuation of series, so we provide the following definitions and theorems, and prove the theorem.

Definition

Let original series be

$$x^{(0)}=\left\{x^{(0)}(1),x^{(0)}(2),\cdots,x^{(0)}(n)\right\};$$

Corresponding prediction value series is

$$\hat{x}^{(0)}=\left\{\hat{x}^{(0)}(1),\hat{x}^{(0)}(2),\cdots,\hat{x}^{(0)}(n)\right\};$$

Mean value series is defined as

$$y^{(0)}=\left\{y^{(0)}(1),y^{(0)}(2),\cdots,y^{(0)}(n)\right\},$$

Where $y^{(0)}(k)=\sum_{i=1}^{k}x^{(0)}(i)/k$, $k=1,2\cdots,n$;

Prediction value of mean value series is $\hat{y}^{(0)}=\left\{\hat{y}^{(0)}(1),\hat{y}^{(0)}(2),\cdots,\hat{y}^{(0)}(n)\right\}$

Theorem

By means of making average processing to original series, the randomness of generated series will be weakened and the fluctuation is reduced.

Prove: Known original series is

$$x^{(0)} = \left\{x^{(0)}(1), x^{(0)}(2), \cdots, x^{(0)}(n)\right\},$$

Corresponding actual value series is

$$x'^{(0)} = \left\{x'^{(0)}(1), x'^{(0)}(2), \cdots, x'^{(0)}(n)\right\},$$

Random error series is

$$\varepsilon = \left\{\varepsilon(1), \varepsilon(2), \cdots, \varepsilon(n)\right\},$$

$$\varepsilon(k) = \left|x^{(0)}(k) - x'^{(0)}(k)\right|.$$

Obviously $x^{(0)}(k) = x'^{(0)}(k) + \varepsilon(k)$

When we make average processing to original series, we can get:

$$y^{(0)}(k) = \sum_{i=1}^{k} x^{(0)}(i)/k = \sum_{i=1}^{k} x'^{(0)}(i)/k + \sum_{i=1}^{k} \varepsilon(i)/k \quad (13)$$

Where, we find that the fluctuation of original series will be weakened after average processing.

The concrete steps of mean value modification method are as follows.

(1) Carry on average processing to original series according to above-mentioned methods; and get the following mean value series:

$$y^{(0)} = \left\{y^{(0)}(1), y^{(0)}(2), \cdots, y^{(0)}(n)\right\}$$

(2) Carry on an accumulation to mean value series and get the generated series

$$y^{(1)} = \left\{y^{(1)}(1), y^{(1)}(2), \cdots, y^{(1)}(n)\right\}.$$

(3) Calculate $[\alpha, u]$, and get the prediction value of mean value series.

(4) Revert the prediction value of mean value series back to the prediction value of original series. The method is as Equation (14) shows.

$$\hat{x}^{(0)}(k+1) = (1-e^{\alpha})e^{-\alpha k}(y^{(0)}(1) - \frac{u}{a})(k+1-ke^{\alpha}) \quad (14)$$

The concrete derivation process is as follows:

We adopt mean value method to carry on the pretreatment to initial array and get

$$\sum_{i=1}^{k} x^{(0)}(i) = k \cdot y^{(0)}(k) \quad (15)$$

$$\hat{x}^{(0)}(k+1) = \sum_{i=1}^{k+1} \hat{x}^{(0)}(i) - \sum_{i=1}^{k} \hat{x}^{(0)}(i) = (k+1) \cdot \hat{y}^{(0)}(k+1) - k \cdot \hat{y}^{(0)}(k) \quad (16)$$

From the view of prediction model, we can get

$$\hat{y}^{(0)}(k+1) = (1-e^{\alpha})\left(\beta y^{(0)}(1) - \frac{u}{\alpha}\right)e^{-\alpha k} \quad (17)$$

Where, $\beta = \dfrac{\frac{u}{\alpha}(b-c)+d}{y^{(0)}(1) \cdot b} \quad k = 1, 2 \cdots, n$

Substituted in Equation (16)

$$\hat{x}^{(0)}(k+1) = (k+1) \cdot \left[(1-e^{\alpha})(\beta y^{(0)}(1) - \frac{u}{\alpha})e^{-ak}\right] - k \cdot \left[(1-e^{\alpha})(\beta y^{(0)}(1) - \frac{u}{a})e^{-a(k-1)}\right]$$

Simplification:

$$\hat{x}^{(0)}(k+1) = (1-e^{\alpha})\left[\beta y^{(0)}(1)e^{-\alpha k}(k+1-ke^{\alpha}) - \frac{u}{\alpha}e^{-\alpha k}(k+1-ke^{\alpha})\right]$$

$$\hat{x}^{(0)}(k+1) = (1-e^{\alpha})e^{-\alpha k}(\beta y^{(0)}(1) - \frac{u}{\alpha})(k+1-ke^{\alpha})$$

$$y^{(0)}(1) = x^{(0)}(1)$$

So

$$\hat{x}^{(0)}(k+1) = (1-e^{\alpha})e^{-\alpha k}(\beta x^{(0)}(1) - \frac{u}{\alpha})(k+1-ke^{\alpha})$$

Where, $\beta = \dfrac{\frac{u}{\alpha}(b-c)+d}{x^{(0)}(1) \cdot b} \quad k = 1, 2 \cdots, n$

Similarly

$$\hat{x}^{(1)}(k+1) = e^{-\alpha k}[\beta x^{(0)}(1) - \frac{u}{\alpha}](k+1-ke^{\alpha}) + \frac{u}{\alpha}$$

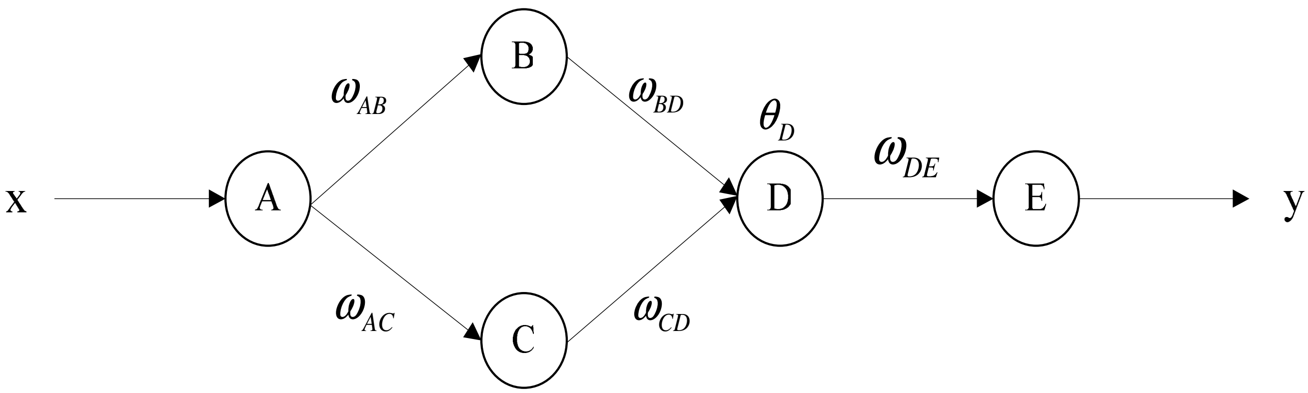

Figure 2. Neural network mapping chart of improved gray prediction model.

Where, $\beta = \dfrac{\frac{u}{\alpha}(b-c)+d}{x^{(0)}(1)\cdot b} \quad k = 1, 2\cdots, n$

Method of mapping improved gray model into neural network

In the paper, we map Grey prediction model into artificial neural network, and combine both structures together. It not only can improve the final prediction precision but also can reduce the uncertainty that people bring for setting the parameters of network (Zhou et al., 2007).

Sigmoid function is used as the inspirit function of gray neural network. We can carry on the following deformation:

$$\hat{x}^{(0)}(k+1) = (1-e^{\alpha})e^{-\alpha k}(\beta x^{(0)}(1)-\frac{u}{\alpha})(k+1-ke^{\alpha})$$
$$=(1-e^{\alpha})(k+1-ke^{\alpha})(\beta x^{(0)}(1)-\frac{u}{\alpha})\frac{e^{-\alpha k}}{1+e^{-\alpha k}}\times(1+e^{-\alpha k})$$
$$=(1-e^{\alpha})(k+1-ke^{\alpha})(\beta x^{(0)}(1)-\frac{u}{\alpha})\frac{1}{1+e^{\alpha k}}\times(1+e^{-\alpha k})$$

Notice that

$$\frac{1}{1+e^{\alpha k}} = 1 - \frac{1}{1+e^{-\alpha k}}$$

The deformation is as follows.

$$\hat{x}^{(0)}(k+1) = (1-e^{\alpha})(k+1-ke^{\alpha})(\beta x^{(0)}(1)-\frac{u}{a})\left[1-\frac{1}{1+e^{-\alpha k}}\right]\times(1+e^{-\alpha k})$$
$$=\left[(\beta x^{(0)}(1)-\frac{u}{\alpha})-\beta x^{(0)}(1)\frac{1}{1+e^{-\alpha k}}+\frac{u}{\alpha}\frac{1}{1+e^{-\alpha k}}\right]\times\cdots$$
$$(1-e^{\alpha})(k+1-ke^{\alpha})(1+e^{-\alpha k})$$

We map the improved gray prediction model into neural network after changing; the concrete structure is as Figure 2 shows. The form of sigmoid function is

$$f(x) = \frac{1}{1+\exp(-kx)}$$

, reference above-mentioned model expressions, we can set the inspirit function of improved gray neural network as

$$f(x) = \frac{1}{1+\exp(-x)}.$$

Inspirit function acts on node *B* and *C* in the network, other nodes do not have inspirit function, the threshold value of node *D* is θ_D, threshold value of other nodes are 0, the function of node *E* is just to carry the zooming operation on node *D*. The concrete network parameters are set as follows:

$$x = k$$

$$w_{AB} = w_{AC} = \alpha$$

$$w_{BD} = -\beta x^{(0)}(1)$$

$$w_{CD} = \frac{u}{\alpha}$$

$$\theta_D = \frac{u}{\alpha} - \beta x^{(0)}(1)$$
$$w_{DE} = (1-e^{\alpha})(k+1-ke^{\alpha})(1+e^{-\alpha k})$$

Analyzing the corresponding relation between BP neural network and improved gray prediction model, we can find that the weight and threshold value of BP network influence each other (Lu and Zhong, 2000). While designing the algorithm of BP network, we mainly hold the following several points:

1. w_{AB} and w_{AC} are always equal, to study w_{AB} is to study α, updated w_{AB} is the new value of α.
2. The connection between node *D* and *E* plays a zooming role only to the output, and w_{DE} is only related to input and α. So w_{DE} does not participate in the training course, we just need to take the adjusted α value into the formula of w_{DE}.
3. Weight w_{BD} is only related to the input and α, learning of w_{BD} is just to take the adjusted value into the formula.
4. Only inspirit function of node *B* and *C* are Sigmoid functions, inspirit function of other nodes are linear functions. We use linear function with derivation. There is only threshold value on node *D*, and other nodes not.
5. We only consider the first three layers when network trains, and performance function of network is varied correspondingly.

The formula of network training course is derived as follows.

Positive propagation

Calculation of hidden layers

Input of node *B*:

$$net_B = k \times w_{AB} \tag{18}$$

Input of node *C*:

$$net_C = k \times w_{AC} \tag{19}$$

Output of node *B*:

$$a_B = f(net_B) = \frac{1}{1+\exp(-k \times w_{AB})} \tag{20}$$

Output of node *C*:

$$a_c = f(net_C) = \frac{1}{1+\exp(-k \times w_{AC})} \tag{21}$$

Calculation of output layer

Input of node *D*:

$$net_D = a_B \times w_{BD} + a_C \times w_{CD} - \theta_D \tag{22}$$

Output of node *D*:

$$a_D = f(net_D) = net_D \tag{23}$$

Reverse weight modification

The evaluation function of network performance is adjusted as follows.

$$E = \frac{1}{2}\sum_{i=1}^{p}\sum_{j=1}^{l}\left(\frac{t-y}{w_{DE}}\right)^2 = \frac{1}{2}\sum_{i=1}^{p}\sum_{j=1}^{l}\left(\frac{t}{w_{DE}} - a_D\right)^2 \tag{24}$$

Where, *p* denotes sample quantity, *l* denotes nodes of output layer

The formula derivation of weight w_{CD} change:

$$\begin{aligned}\Delta w_{CD} &= -\eta \frac{\partial E}{\partial w_{CD}} \\ &= -\eta \frac{\partial E}{\partial a_D} \cdot \frac{\partial a_D}{\partial net_D} \cdot \frac{\partial net_D}{\partial w_{CD}} \\ &= -\eta \left(\frac{t}{w_{DE}} - a_D\right) \cdot (-1) \cdot a_C \\ &= \eta a_C \left(\frac{t}{w_{DE}} - a_D\right)\end{aligned} \tag{25}$$

The formula derivation of weight w_{AC} change:

$$\begin{aligned}\Delta w_{AC} &= -\eta \frac{\partial E}{\partial w_{AC}} \\ &= -\eta \frac{\partial E}{\partial a_D} \cdot \frac{\partial a_D}{\partial net_D} \cdot \frac{\partial net_D}{\partial a_C} \cdot \frac{\partial a_C}{\partial net_C} \cdot \frac{\partial net_C}{\partial w_{AC}} \\ &= -\eta \left(\frac{t}{w_{DE}} - a_D\right) \cdot (-1) \cdot w_{CD} \cdot a_C \cdot (1-a_C) \cdot k \\ &= \eta k w_{CD} a_C (1-a_C)\left(\frac{t}{w_{DE}} - a_D\right)\end{aligned}$$

The adjusting method of weight and threshold value:

$$w'_{AC} = w'_{AB} = \alpha' = w_{AC} + \Delta w_{AC};$$

$$w'_{CD} = \frac{u'}{\alpha'} = w_{CD} + \Delta w_{CD};$$

$w'_{DE} = (1-e^{\alpha'})(k+1-ke^{\alpha'})(1+e^{k\alpha'})$;

$w'_{BD} = \beta' x^{(0)}(1)$;

$$\theta'_D = \frac{u'}{\alpha'} - x^{(0)}(1)$$

$$\beta' = \frac{\frac{u'}{a'}(b'-c')+d'}{x^{(0)}(1)\cdot b'}$$

$$b' = \sum_{k=1}^{n} \exp(-2\alpha'(k-1))$$

$$c' = \sum_{k=1}^{n} \exp(-\alpha'(k-1))$$

$$d' = \sum_{k=1}^{n} x^{(1)}(i)\exp(-\alpha'(k-1))$$

Where, w'_{AC}, w'_{AB}, w'_{BD}, w'_{CD}, w'_{DE}, θ'_D are respectively adjusted weight and threshold value. α', u', β', b', c', d' are adjusted parameters.

The improved gray prediction model uses the training of neural network to adjust models and final prediction result. While the network trains, node *E* only plays a role in scaling, so we do not consider/neglect the influence (Gao and Feng, 2004). The training of neural network is to adjust and study parameters *u*, *a*, initial data $\beta x^{(0)}(1)$ again and again.

Markov chain to amend prediction result

Markov chain method carries on trend prediction to the data sequence of grey neural network, and utilizes the prediction method of Markov's state transfer matrix to carry on the second fitting to the gray prediction value of support expenditure. Namely we use Markov's chain method to obtain the deviation law of GNN model prediction results and adjust them according to the law. It increases the prediction credibility.

Suppose that the original time series and gray prediction result of support cost of engineering equipment with pretreatment are respectively:

$$x^{(0)} = \left\{x^{(0)}(1), x^{(0)}(2), \cdots, x^{(0)}(n)\right\}$$

$$\hat{x}^{(0)} = \left\{\hat{x}^{(0)}(1), \hat{x}^{(0)}(2), \cdots, \hat{x}^{(0)}(n)\right\}$$

They are referred to as original data and prediction result, the modification approach of final prediction value is as follows.

1. Choose former *k* group data and calculate the ratio of original data and prediction result that is $x^{(0)}(k)/\hat{x}^{(0)}(k)$.
2. Divide the ratio into *m* kinds of state according to the ratio range. $E=\{E_1, E_2, \cdots E_m\}$, there are interval ranges in each state, the interval range of E_i is $e_i(e_{il}, e_{iu})$.
3. Confirm the state grade of each pair of data according to the division of state.

(4)Construct state transfer matrix P^t

$$P^t = \begin{bmatrix} p^t_{11} & p^t_{12} & \cdots & p^t_{1m} \\ p^t_{21} & p^t_{22} & \cdots & p^t_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ p^t_{m1} & p^t_{m2} & \cdots & p^t_{mm} \end{bmatrix}$$

Where, $p^t_{ij} = M^t_{ij}/M_i$, p^t_{ij} is the probability of state i transferred to state j by *t* steps, M^t_{ij} is the times of state i transferred to state j in the direction of prediction value through *t* steps and M_i is the appearing times of state i.

5. Compile prediction table. Select *m* nearest data couple which is the least distance to the value to be predicted. The steps in compiling the prediction table are given in 6.
6. Make sum of probability in each column of the Table 1, and select the state whose sum is the greatest as the final transfer state.
7. Calculate the final prediction value according to Equation (26)

$$x'^{(0)}(k+1) = \hat{x}^{(0)}(k+1)\cdot\frac{(e_l+e_u)}{2} \quad (26)$$

Where e_u and e_l are respectively the upper and lower bounds of selected state.

RESULTS AND DISCUSSION

Pretreatment of sample data

Obtain the support cost data of certain engineering

Table 1. Statistics of state transfer probability.

State	Steps	E_1	E_2	…	E_m
E_k	m	p_{k1}^{m}	p_{k2}^{m}	…	p_{km}^{m}
…	…	…	…	…	…
E_j	2	p_{j1}^{2}	p_{j2}^{2}	p_{j3}^{2}	p_{jm}^{2}
E_i	1	p_{i1}^{1}	p_{i2}^{1}	p_{i3}^{1}	p_{im}^{1}
Sum		$\sum$	$\sum$	$\sum$	Σ

Table 2. State division.

State 1	State 2	State 3	State 4
[0.9278□0.9992)	[0.9992□1.0707)	[1.0707□1.1421)	[1.1421□1.2136)

Table 3. Corresponding state of BP network calculation result.

Year	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008
Cost	1.434	1.717	1.975	2.365	2.558	2.590	2.915	3.045	3.309	3.410	3.520
Result of BP	1.434	1.886	2.038	2.201	2.373	2.556	2.751	2.958	3.179	3.413	3.664
Error		1.213	1.098	1.032	0.931	0.928	0.987	0.944	0.972	0.961	0.999
State		4	3	2	1	1	1	1	1	1	1

equipment from 1998 to 2008 through surveying.

Regard the support cost of engineering equipment in the past ten years as known time series data. At the same time we take them into the traditional GM (1, 1) prediction model and improved GNN prediction model to predict the cost of 2008 so as to examine the usability of GNN prediction model to support cost of engineering equipment. We get $\alpha = -0.0768$, $u = 1.7334$ in the traditional GM (1, 1) prediction model through calculation. Prediction model of first-order accumulated series:

$$\begin{cases} \hat{x}^{(1)}(k+1) = 23.9949 \times e^{0.0768k} - 22.5609, k = 1,2,\cdots,n \\ \hat{x}^{(1)}(1) = x^{(1)}(1) \end{cases}$$

Prediction model of original series:

$$\begin{cases} \hat{x}^{(0)}(k+1) = 1.7746 \times e^{0.0768k} \\ \hat{x}^{(0)}(1) = x^{(0)}(1) \end{cases}$$

In the improved GNN prediction model, the adaptive parameter $\theta = 0.5047$, corresponding $\alpha = -0.0561$, $u = 1.5249$ initial data parameter $\beta = -4.0369$, $\alpha = -0.0512$ $u = 1.3923$ with adjustment of BP network. The original series prediction model is:

$$\hat{x}^{(0)}(k+1) = (1-e^{\alpha})e^{-\alpha k}(\beta x^{(0)}(1) - \frac{u}{\alpha})(k+1-ke^{\alpha})$$

$$\begin{cases} \hat{x}^{(0)}(k+1) = 1.4587 \times e^{0.0512k} \times (0.0499k+1), k = 1,2\cdots,n \\ \hat{x}^{(0)}(1) = x^{(0)}(1) \end{cases}$$

Use Markov's chain to adjust the final prediction result of BP network. Firstly divide the ratio of prediction value to actual value into three states. The range of each state is as shown in Table 2.

Corresponding state of each prediction data is as shown in Table.3.

State transfer matrix is as follows.

$$p(1) = \begin{bmatrix} 0.8571 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad p(2) = \begin{bmatrix} 0.7143 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Table 4. Prediction.

State	Step	1	2	3	4
1	4	0.4286	0	0	0
1	3	0.5714	0	0	0
1	2	0.7143	0	0	0
1	1	0.8571	0	0	0
Sum		2.5714	0	0	0

Table 5. Contrast of prediction result.

Year	Actual value	Prediction value of GM (1,1)	Prediction value of Improved GNN	Prediction error of GM(1,1)	Prediction error of improved GNN
1998	1.434	1.4340	1.4340	0	0
1999	1.717	1.9163	1.8856	0.1161	0.0982
2000	1.975	2.0693	2.0386	0.0478	0.0322
2001	2.365	2.2346	2.2009	-0.0551	-0.0694
2002	2.558	2.4131	2.3733	-0.0567	-0.0722
2003	2.590	2.6058	2.5564	0.0061	-0.0130
2004	2.915	2.8139	2.7510	-0.0347	-0.0563
2005	3.045	3.0386	2.9581	-0.0021	-0.0285
2006	3.309	3.2813	3.1786	-0.0084	-0.0394
2007	3.410	3.5433	3.4135	0.0370	-0.0010
2008	3.520	3.8263	3.5301	0.0870	0.0029

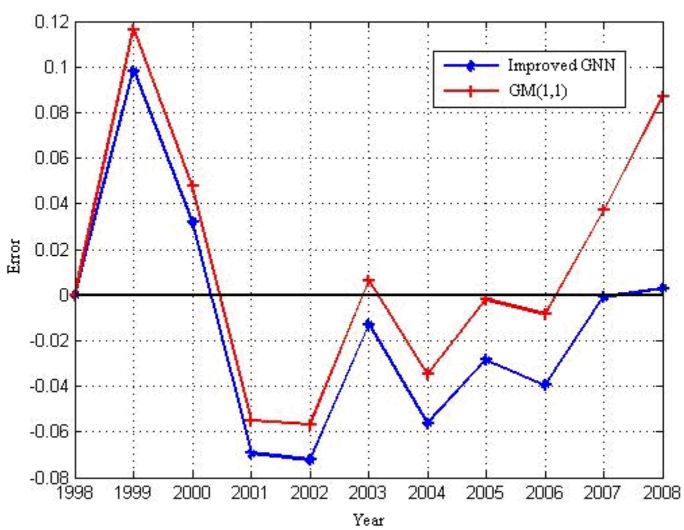

Figure 3. Contrast of prediction error.

$$p(3)=\begin{bmatrix}0.5714 & 0 & 0 & 0\\ 1 & 0 & 0 & 0\\ 1 & 0 & 0 & 0\\ 1 & 0 & 0 & 0\end{bmatrix} p(4)=\begin{bmatrix}0.4286 & 0 & 0 & 0\\ 1 & 0 & 0 & 0\\ 1 & 0 & 0 & 0\\ 1 & 0 & 0 & 0\end{bmatrix}$$

The prediction table is compiled as shown in Table 4.

In the table, we find that the final calculation result is 3.5301.

Test and analysis of prediction precision

The prediction result of GM (1, 1) and GNN prediction model, together with the corresponding error, make up the prediction result contrast table that is as shown in Table 5 Through comparing the prediction results in Table 7 and prediction error in Figure 3, we find that traditional GM (1, 1) model and improved GNN prediction model have similar fitting precision in known series. But finally the improved GNN prediction model is obviously better than traditional GM (1, 1) prediction model in the prediction precision of unknown data. Analyzing the training curve of improved GNN prediction model is as Figure 4 shows. We know that the training of BP network is quick in the improved GNN prediction model. It finishes the second learning of related parameters promptly through 191 generations. Though its prediction result is not better than the final result in precision, it has improved greatly compared to traditional GM (1, 1) model. The fitting of known data mainly relies on the setting of BP network training target. The smaller the training target setting is, the more accurate the prediction of known data is. The phenomena of over-learning may appear. We set the training target relatively large to avoid the appearance of this phenomenon. So the laminating degree of improved GNN prediction model and known data is not high.The final prediction result contrast is as Figure 5 shows. The prediction curve shape of improved GNN and

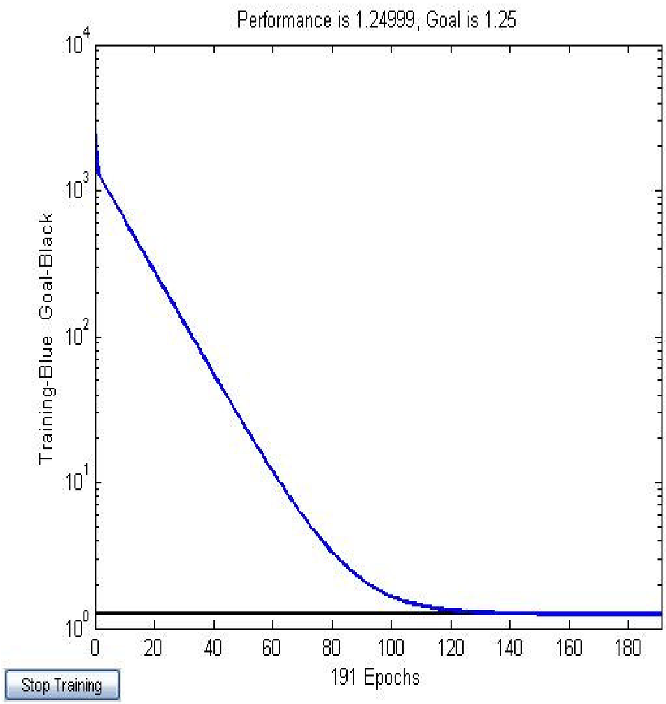

Figure 4. Training curve of BP network cost.

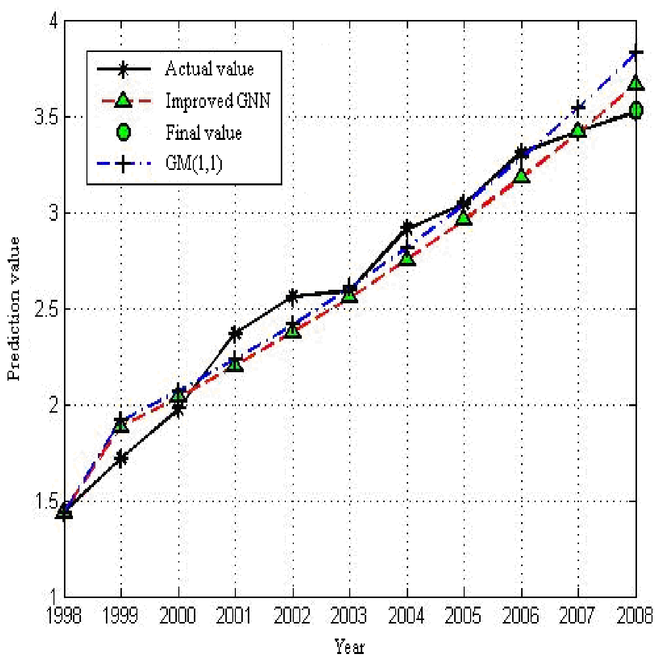

Figure 5. Prediction contrast of support.

traditional GM (1, 1) prediction model are basically the same, but they separate gradually in the prediction course. Traditional GM (1, 1) prediction model is comparatively close to known data, and the prediction result of improved GNN prediction model is obviously better than traditional GM (1, 1) prediction model. At last, use Markov's chain to adjust the prediction result again. In that case, the prediction precision could improve further.

Conclusions

GNN prediction model is improved on the basis of traditional GM (1, 1) model. It has characteristic of traditional GM (1, 1) model, so the prediction curve shapes of two methods are very similar. However the aim of adopting mean value modification tactics is to reduce the fluctuation sensitivity of model to known data. So in the face of known data with great fluctuation, improved GNN prediction model can reduce the interference of fluctuation, and receive better precision.

REFERENCES

Gao W, Feng XT (2004). Study on displacement prediction of landslide based on grey system and evolutionary neural network. Rock and Soil Mechanics, 25(4): 514-517.

Glenn DGL (1998). Determining optimal testing times for Markov chain usage models. Proc. 1998 IEEE Southeastcon Conference, pp. 1-4.

Li DW, Xu HJ (2009).Improved grey Markov model and its application in prediction of flight accident rate. China Safety Sci. J., 19(9): 53-57.

Li HW, Yuan XW (2009). Application of GM (1, 1) model in atmospheric quality forecast. J. Liaoning Tech. Univ., 28(S): 134-136.

Liu SF (2004). Grey system theory and applications. pp. 155-158.

Lu HH, Zhong L (2000). Exploring in fusion technology of grey system and neural networks. Microcomput. Dev., 10(3): 3-5.

Luo YF, Cui XL (2005). A Fourier series model for forecasting reference crop evapotranspiration, 38(6): 45-52.

Prowell SJ (2004). A cost-benefit stopping criterion for statistical testing. Proc.37th International Conference on System Science, pp.304-309.

Simmons E (2005). The usage model: a structure for richly describing product usage during design and development. Proc.13th IEEE International Conference on Requirements Engineering, pp. 403-407.

Takagi T, Furukawa Z (2004). Constructing a usage model for statistical testing with source code generation methods.Proc.11th Asia-Pacific Software Engineering Conference, pp. 448-454.

Trajkovic S, Todorovic B, Stankovic M (2003). Forecasting of reference evapotranspiration by artificial neural networks. J. Irrigation Drainage Eng., 129(6): 454-457.

Zhou ZG, Guo K, Chen LB (2007). Grey neural network technology in prediction for time sequence. Stat. Decision 7(1): 128-129.