The Prediction of Haze Based on BP Neural Network and Matlab

Ma Limei [1,2,3] , Wang Fangwei [1,2]

[1] College Of Information Technology, Hebei Normal University, Shijiazhuang 050024, Email: malimei@bupt.edu.cn

[2] Key Laboratory of Network and Information Security in Hebei Province, Shijiazhuang 050024

[3] Visiting scholar of Beijing University of Posts and Telecommunications 100876

Abstract. In this paper, the neural network theory is used to establish the BP neural network prediction system for the occurrence of haze. The corresponding parameters are determined by MATLAB language, and the effect of the model is tested by the prediction of Shijiazhuang area. The result shows the feasibility of the predictive model. So it' s valuable and has a bright future.

Keywords: BP neural network, haze, Matlab

1. Introductiong

In recent years, the haze has been becoming more and more serious harm to people' s daily life and health, the Beijing Tianjin Hebei region is particularly significant, the most serious area is Hebei Shijiazhuang, haze weather accounted for almost the whole winter, many factors causing haze, from the angle of the influence of climatic factors for the occurrence of haze weather, mainly includes the following 4 factors:

1) SO2: the most common sulfur oxides. A colorless, poisonous gas with a strong irritant.One of the major pollutants in the atmosphere. The volcano emits gas when it erupts, and sulfur dioxide is produced in many industrial processes. Due to coal and oil usually contain sulfur compounds, so the combustion generates sulfur dioxide, so the car tail gas and sulfur dioxide, into the winter heating period in North China, using a large amount of coal, it will produce large amounts of sulfur dioxide, causing haze aggravated.

2) NO2: nitrogen dioxide is a reddish brown, highly active gaseous substance, also known as nitrogen oxide. Nitrogen dioxide plays an important role in the formation of ozone. In addition to natural sources, nitrogen dioxide mainly comes from the release of high temperature combustion processes, such as motor vehicle exhaust and boiler exhaust emissions. In addition, the industrial process also produces some nitrogen dioxide. Worldwide anthropogenic pollution is estimated to be about 53 million tons of nitrogen oxides per year.

3) CO: carbon monoxide is the most widely distributed and the largest amount of pollutants in the atmosphere, and it is also one of the important pollutants produced during combustion. The main

source of CO in the atmosphere is internal combustion engine exhaust, followed by combustion of fossil fuels in boilers. Any carbon containing substance may produce carbon monoxide when combustion is incomplete.

4) O3: the ozone layer in the atmosphere is now widely known for its protection of the living things of the earth. It absorbs most of the ultraviolet radiation released by the sun and protects animals and plants from such rays. This gives people the impression that it is protected by ozone should better, this is not the case, if the ozone in the atmosphere, especially ozone near the ground in the atmosphere gathered too much for humans but is a scourge of high concentration ozone. Ozone is also a greenhouse gas, which can cause more serious greenhouse effect. High concentrations of ozone near the ground can stimulate and damage the mucous tissues of the eyes, respiratory system, and have a negative effect on human health.

2. Based on Neural Network Haze Forecast Principle

There are many causes of haze, as we all know, there is a close relationship between haze and weather factors, it is also affected by temperature, wind and rainfall and other factors. There is a complex interaction between the factors that affect the haze, and it is difficult to establish a precise and perfect prediction model using traditional methods. The BP neural network has good characteristics of predicting nonlinear complex systems, and can effectively describe its complex nonlinear characteristics such as uncertainty and multi input.

Prediction is to estimate the size of future unknown data by some known historical data, with time series { X_i }, where historical data $X_n, X_{n+1}, \cdots\cdots, X_{n+m}$ To predict the size of the future n+m+k (k>0) moment, that is, to predict the size X_{n+m+k}, the method is to find the historical data, $X_n, X_{n+1}, \cdots\cdots$, X_{n+m} And X_{n+m+k} some nonlinear function relation:

$$X_{n+m+k} = F(X_n, X_{n+1}, \cdots\cdots, X_{n+m})$$

Here, x_1, $x_2 \cdots\cdots$, x_n, N represents the output of haze factors, X_{n+m+k} which means the output haze size.

Neural network is used to predict, that is, the neural network is used to fit the function F (y), and the future data is obtained. Commonly used are the following three types of forecasts:

1) single step prediction, when k=1, network input X_n, X_{n+1}, $\cdots\cdots$, X_{n+m}, m historical data, output X_{n+m+k} sizes. Such forecasts are clearly not applicable to weather haze forecasts.

2) multi step prediction, when k>1, that is, the network input m historical data, output X_{n+m+1}, X_{n+m+2}, $\cdots\cdots$, X_{n+m+k}, The experimental results show that the prediction error of fog is larger.

3) rolling prediction, also known as iterative step by step prediction

A single step prediction is then carried out, and then the output is fed back to the input as part of the network input, as shown below (to predict the size of the next q moments):

Execution steps neural networksinputs and outputs (prediction results)

(1) $x_a, x_{a.1}, \cdots, x_{a.m} \quad x_{a.m.1}$

(2) $x_{a.1}, x_{a.2}, \cdots, x_{a.m.1} \quad x_{a.m.2}$

…………

q $x_{a.q-1}, x_{a.q}, \cdots, x_{a.m.q-1} \; x_{a.m.q}$

Prediction of haze weather rolling forecasts, forecast of haze weather is to establish a prediction function according to the collected historical data and its impact, so the prediction results and the actual results of error as small as possible, determined between the amount and the weather factors, the future weather forecast the change trend of haze size. The fog and haze, reaches the aim of early warning, the haze hazard is reduced to the minimum.

3. Establishment of Fog and Haze Prediction Model

3.1 BP algorithm principle

BP algorithm is a common algorithm for training multilayer feedforward networks. It has three or more than three layers of neural networks, including input layer, output layer and intermediate layer. When a learning mode for the network, the neuronal activation sizes from the input layer of the middle layer (hidden layer) transmitted to the output layer, the output layer neurons corresponding to the input mode of the network response. Then, according to the principle of reducing the desired output and the actual output error, the connection rights are corrected from the output layer, the intermediate layer, and finally back to the input layer. Since this correction process takes place from the output layer to the input layer, it is called error propagation algorithm" . With the error back-propagation training continuously, the network also continue to improve the correct rate of the input response pattern, as a kind of suitable for non-linear object analysis and prediction tools are widely used in various fields.

Figure 1 shows the usual BP network model with an intermediate layer (hidden layer). Where O_k represents the output unit, V_j represents the implicit unit, and ξ_j represents the input unit. The connection power from the input unit to the implied unit is w_{jk} , and the connection power from the implicit unit j to the output unit i is w_{ij} , and $\omega = (w_{jk}, w_{ij})$ represents all the connection rights.

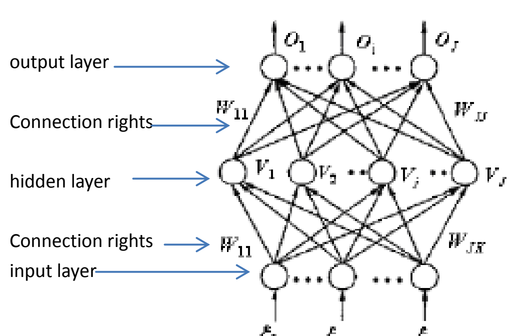


Figure 1 a BP network model with

Figure.1 BP network model with an implicit layer

3.2 BP network learning process specific algorithm steps

According to the theory of BP, three layer BP neural network input layer neuron number is L and the number of neurons in the hidden layer is M, the number of output layer neurons of N, P on the training mode, here are the specific steps of the algorithm of the BP network learning process:

1) Initialize the weights and thresholds of each neuron, and assign the random sizes (-1, +1) to $\{w_{ij}\}$、$\{w_{jk}\}$、$\{\theta_j\}$、$\{\theta_k\}$

2) A sample set of data is sampled randomly from the training sample into the network, for example, $B_n = (b_{n1}, \cdots, b_{nN})$, $D_n = (d_{n1}, \cdots, d_{nN})$ is provided to the network;

3) Use input mode $B_n = (b_{n1}, \cdots, b_{nN})$, connection power { w_{ij} }, threshold { θ_j } to calculate the input of the units in the middle layer S_j ;

According to the principle of neuron model, the input S_j (activation size) of each neuron in middle layer is calculated.

$$S_j = \sum_{i=1}^{L} W_{ij} \cdot b_{in} - \theta_j \quad j=1,2,\cdots,M \tag{1}$$

The activation size is inserted into the activation function and the output of the intermediate layer j is V_j

$$V_j = f(s_j) = \frac{1}{1 + \exp(-\sum_{i=1}^{L} W_{ij} b_{in} + \theta_j)} \quad n = 1,2,\cdots,p \tag{2}$$

4) Use the output V_j of the intermediate layer, the connection power { w_{ij} }, and the threshold { θ_k } to calculate the input l_k of each unit of the output layer;

According to the principle of neuron model, the input l_k (activation size) of each neuron in the output layer is calculated.

$$l_k = \sum_{j=1}^{p} W_{jk} \cdot V_j - \theta_k \quad j=1,2,\cdots,M \tag{3}$$

When the activation size is substituted into the activation function, the response O_k of the units in the output layer can be obtained

$$O_k = f(\sum_{j=1}^{p} W_{jk} V_j - \theta_k) = \frac{1}{1 + \exp(-\sum_{j=1}^{p} W_{jk} V_j + \theta_k)} \quad k = 1,2,\cdots,N \tag{4}$$

5) The correction error of each node in the output layer is calculated by using the expected output mode

$D_n = (d_{n1}, \cdots, d_{nN})$ and the actual output O_k of the network δ_k.

$$\delta_k = (d_k - O_k) \cdot O_k (1 - O_k) \quad k = 1,2,\cdots,N \tag{5}$$

6) The correction error of each node in the middle layer is calculated by using the connection error { w_{ij} }, the output layer' s general error δ_k and the output V_j of the middle layer.

$$e_j = V_j (1 - V_j) \cdot \sum_{i=1}^{L} \delta_k \cdot W_j \quad \text{j=1,2,}\cdots\text{,M} \tag{6}$$

7) Error back propagation: according to the calculation results in 5, the output of δ_k and the units of the middle layer V_j , { w_{ij} } and { θ_k } are corrected to calculate the new connection weight and threshold between the next layer and the output layer.

$$w_{jk}(N+1) = w_{jk}(N) + \mu \delta_k V_j + \alpha \Delta w_{jk} \tag{7}$$

j=1,2,⋯,M $k = 1,2,\cdots,N$ (0< α <1)

$$\theta_k (N+1) = \theta_k (N) + \mu \cdot \delta_k \quad k = 1,2,\cdots,N \tag{8}$$

8) Error back propagation: according to the calculation results in e_j 6, the input b_n^k of each input layer of the input layer, the connection weights { w_{ij} } and threshold { θ_j } are corrected, and the new connection weight and threshold between the input layer and the intermediate layer are calculated.

$$w_j (N+1) = w_j (N) + \beta e_j b_n^k + \alpha \Delta w_j \tag{9}$$

i=1,2,⋯,L $j = 1,2,\cdots,M$ (0< α <1)

$$\theta_j (N+1) = \theta_j (N) + \beta \cdot e_j \quad j = 1,2,\cdots,M \tag{10}$$

9) Then select a sample model randomly from the sample space and return to 3, until the training of all samples is finished.

10) Select a pattern pair from the N learning model again, and return to step 3 until the network global error E is less than a minimum size, i.e., the network convergence. The network cannot converge when the number of learning returns is greater than the pre-set size.

11) And finally, the end of study. In the above steps, 3-6 is the forward propagation process of the input learning mode, 7 and 8 are the inverse propagation processes of network errors, and 9 and 10 complete the training and convergence process.

12) The global error function of the network is defined as:

$$E = \sum_P E_P \tag{11}$$

$$E_p = \frac{1}{2}\sum_j (d_{jp} - o_{jp})^2 \quad (12)$$

E_p is the error of a set. For all input modes, E is the global error of the network. The purpose of the training network is to find a set of weights to minimize the error function. From the above we can see that the amount of adjustment of each connection is proportional to the error function E_p of each learning model, which is called standard error back-propagation algorithm. Compared with the global error function, the connection weights of E should be unified after all the learning patterns are supplied to the network. This algorithm is called the cumulative error back-propagation algorithm. When the learning model is not too large, i.e., the learning mode is relatively small, the cumulative error inverse propagation algorithm converges faster than the standard error backpropagation algorithm.

The learning rule of BP nets is to achieve a gradient descent of the sum of squares and errors of E_p (or E) on the set of learning patterns rather than the gradient of the absolute error $\delta_k = (d_k - O_k)$ of a particular pattern component. Therefore, after each calibration, the network output error may also increase for some neurons, but after repeated calculations, the error should be smaller.

4. Input and Output Data Design

The number of neurons in the input and output layers of the BP network is entirely determined by the user' s requirements. The number of neurons in the input layer is determined by the influence factors, consider various factors, thus produce haze, combined with the meteorological data in February 2016, the SO2, NO2, CO several factors and O3 mainly influence the BP neural network model is established to determine the relationship between occurrence quantity and climatic factors of haze. According to the February 2016 air index real-time monitoring data, forecast the haze of March 2016 weather. We selected SO2, NO2, CO, O3 and other four factors as the input data, so the number of input nodes is 4, the number of output nodes is 1, namely SO2, NO2, CO, O3, BP neural network model is established to determine the relationship between the size and climate factors in haze, to predict the future the fog haze size.

The range of fluctuations in the data collected is large, will affect the neural network learning speed and prediction accuracy, therefore, pretreatment to the input and output data, the so-called pre-processing is normalized to the data processing, the input and output data into the parameters can receive the activation function, transform the input data in practice in order to meet the requirements of the network, the input and output of the data, before learning first to normalize the input data according to the following formula

$$y_i = \frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)} * 0.8 + 0.1 \quad i = 1, 2, \cdots\cdots,$$

In the formula y_i is the normalized data, x_i collected real data, the maximum size of $\max(x_i)$ represents a set of data, the minimum size of $\min(x_i)$ represents a set of data, the data after normalization are located in [0.1, 0.9].

5. Using MATLAB To Realize The Determination of The Number of Hidden Layer Neurons and Transfer Function

The number of hidden layer neurons selection is a complex problem, in the application of neural network in the fact that the application of artificial neural network are often transformed into how to determine the parameters of the network structure and connection weights for each. The number of hidden layer neurons may be too little training a network or network is not strong, cannot identify previously never seen samples of fault tolerance; but the number of hidden units too much, and can make the learning time is too long, the error is not necessarily the best, so there is a proper hidden layer unit the problem of determining the number of.

General or by empirical formula to determine the number of hidden layer neurons, but this approach is not accurate, this paper used MATLAB language program to realize, the number of neurons in the input layer of the network has been determined to be 4, the number of neurons in the output layer is 1, according to the hidden layer design formula, and considering the actual situation forecast, to solve the problem of network hidden layer neuron number should be between 6~12. Therefore, we design a BP network with a variable number of neurons in the hidden layer, and determine the optimal number of neurons in the hidden layer by comparing the errors, and test the influence of the transfer function on the network performance.

5.1 Determination of the number of neurons in the hidden layer

The network design and procedures are shown in figure 2:

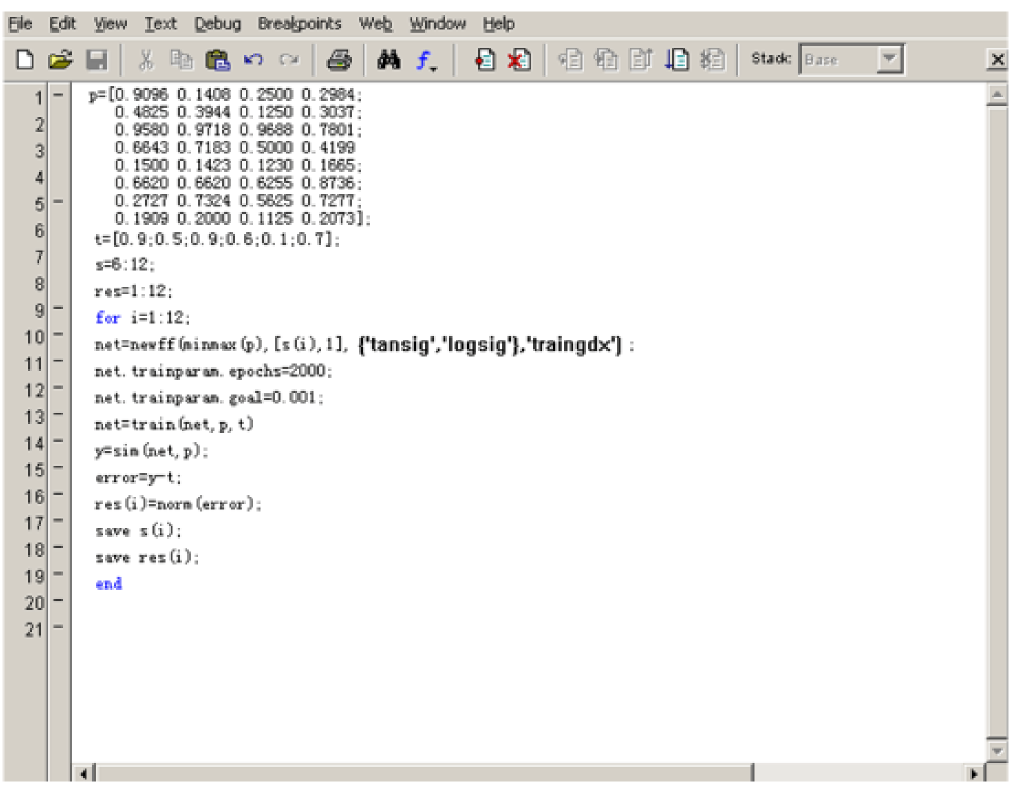


Figure.2 The Procedure for Determining The Number of Neurons In the Hidden Laye

Thus, the transfer function of the hidden layer neurons in the network is Tansig, and the transfer function of the output layer neurons is logsig. The results of the above procedures are shown in table 1.

Table 1 The Number of Neurons and Network Errors

Numberof neurons	6	7	8	9	10	11	12
Network error	0.2642	0.6437	0.1446	0.1442	0.1448	0.1856	0.3584

Table 1 shows that, after 2000 training, the BP network with 9 neurons in the hidden layer is the best approximation to the function, because it has the least error, and the network achieves the target error after 500 training. The network errors of the hidden layer are 8 and 10, but the training time is long. Considering the training speed of network performance, the number of neurons in the network hidden layer is determined to be 9.

It can be seen from table 1, the number of neurons in the hidden layer is not the more the better performance of the network, in the program, and no error obviously with the increase of the number of hidden layer neurons decreased, when the number of neurons increased from 6 to 7, but the error increases. This phenomenon was observed from 9 to 10 and from 11 to 12.

5.2 Predictive model topology

Bp network the number of hidden units are constantly adjusted, we through the results of MATLAB programs, analysis of the hidden nodes is 9, 1 is the number of nodes in the output layer, the prediction model structure as shown in Figure 3 Bp network.

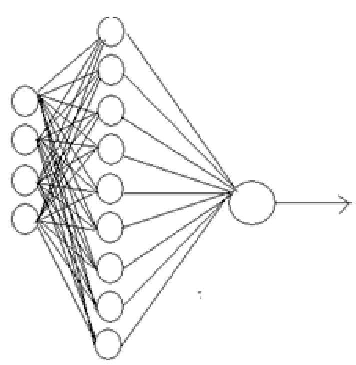


Figure.3 Topological Structure of Prediction Model

5.3 Determination of transfer function

The adoption of different transfer functions also has an impact on the performance of the network, such as convergence speed. The following uses different transfer functions to train the network and observe the results.

The transfer function of the hidden layer and the output layer is Tansig, which is trained by the gradient descent momentum method, and the learning rate is adaptive. When the number of neurons in the hidden layer is 9 and the approximation error of the network is 0.1442, the training results of the network are shown in Figure 4. From Figure 4, we can see that the network achieves the target error after 500 training.

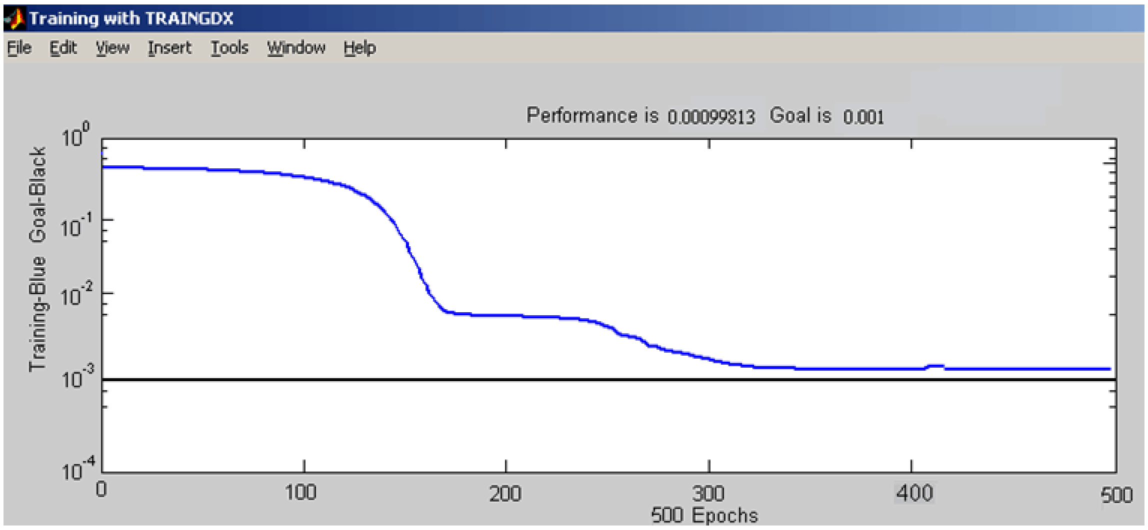


Figure.4 Training Results of Transfer Function Tansig

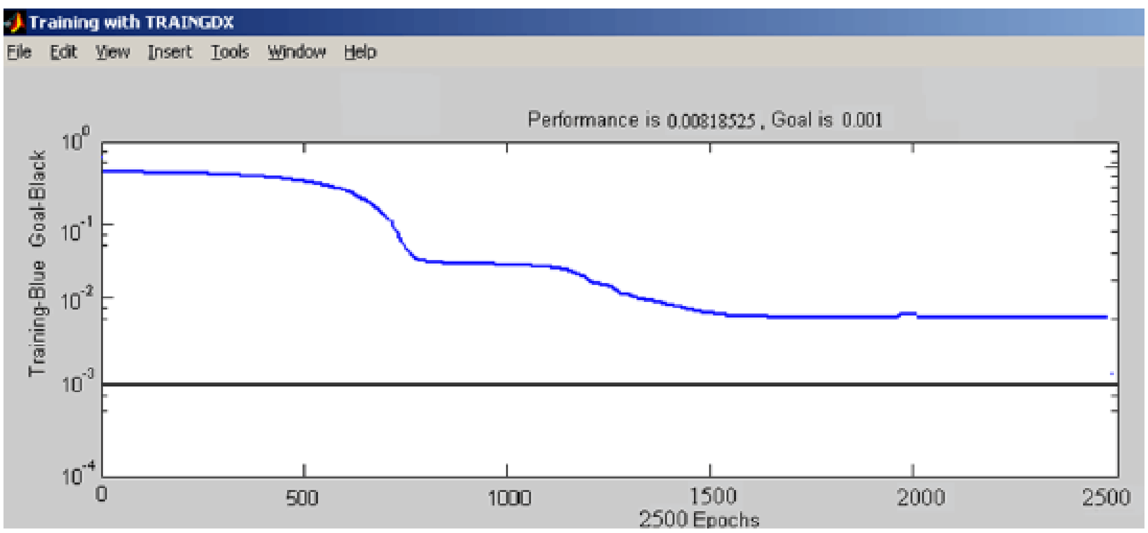


Figure.5 Training Results of Transfer Function Logsig

Finally, the transfer function of network hidden layer neurons using Tansig transfer function, the output layer neurons by logsig, with the rest of the code in Figure 1 shows that after the 2000 training, the network still did not meet the target error, but also through the training curve of the network, as shown in Figure 5, we see the training process of the network convergence very slowly, the training error of res=0.4146 is large.

At this point, we identified the final BP network structure for this prediction, as shown in table 2.

Table 2 BP network structure

Network structure	**hidden layer**	**neuron transfer function**
Single hidden layer BPnetwork	9	tansig

5.4 Determination of learning rate

After the network structure is determined, it is necessary to use the sample data to train through some learning rules, so as to improve the adaptability of the network. Learning speed is an important factor in the training process. It determines the weight change in each cycle. In general, it tends to select smaller learning rates to ensure system stability. The range of learning speed is usually between 0.01-0.07, and the learning rate is 0.05.

6. Simulation

6.1 Simulation example

In this case the data collected from the National City air quality real-time publishing platform, through 1 months of Hebei city in Shijiazhuang Province 7 haze monitoring points collected data, as shown in Table 3, due to space limitations in Table 3 for a day of data, main factors affecting the haze is SO2, NO2, CO, O3, in order to deal with convenient, real-time monitoring of the data (unit: g/m3, Co mg/m3) were normalized after finishing finishing, all the data are shown in table 4.

Table 3 Data on Urban Air Quality in Real Time in China

Real–time monitoring data(ug/m2,co units mg/m2)						
site	So2	No2	co	O3	PM10	PM2.5
Great Hall of the People	50	59	2.1	58	204	118
Staff hospital	52	46	2.1	114	204	112
High-tech Zone	68	73	2.4	80	258	143
22 Zhongnan Campus	75	82	2.4	76	241	137
Century Park	56	64	2.4	69	235	151
Southwest high diocese	39	36	1.9	70	198	134
Yongsan	26	64	1.3	52	205	42

The example of a complete MATLAB program as shown in Figure 6, run the program can get the network training results as shown in Figure 7, the network can be seen after the 10 training can meet the error requirement of network output results as shown in Figure 8, QuXianru is shown in figure 9.

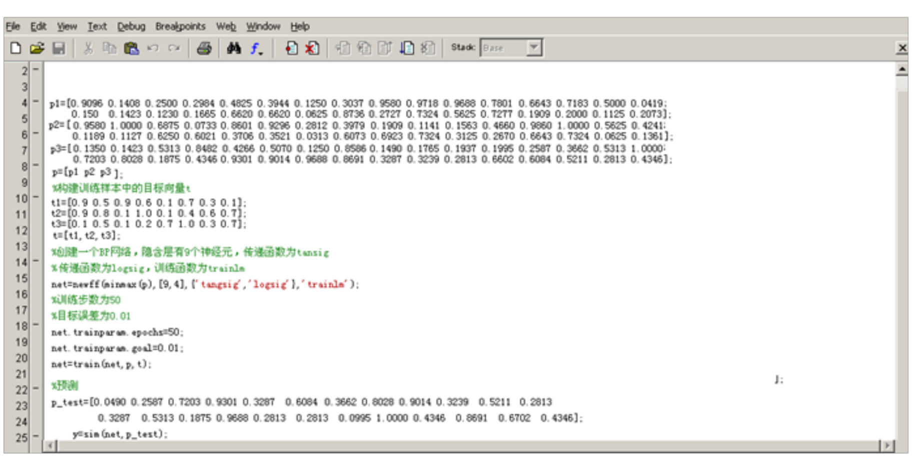


Figure.6 MATLAB Program

Table 4 All Data After Normalization

Years	date	SO2	NO2	CO	O3	Haze size
February 2016	1	0. 9096	0. 1408	0. 2500	0. 2984	0. 9
	2	0. 4825	0. 3944	0. 1250	0. 3037	0. 5
	3	0. 9580	0. 9718	0. 9688	0. 7801	0. 9
	4	0. 6643	0. 7183	0. 5000	0. 4199	0. 6
	5	0. 150	0. 1423	0. 1230	0. 1665	0. 1
	6	0. 6620	0. 6620	0. 6255	0. 8736	0. 7
	7	0. 2727	0. 7324	0. 5625	0. 7277	0. 3
	8	0. 1909	0. 2000	0. 1125	0. 2073	0. 1
	9	0. 9580	1. 0000	0. 6875	0. 0733	0. 9
	10	0. 8601	0. 9296	0. 2812	0. 3979	0. 8
	11	0. 1909	0. 1141	0. 1563	0. 4660	0. 1
	12	0. 9860	1. 0000	0. 5625	0. 4241	1. 0
	13	0. 1189	0. 1127	0. 6250	0. 6021	0. 1
	14	0. 3706	0. 3521	0. 0313	0. 6073	0. 4
	15	0. 6923	0. 7324	0. 3125	0. 2670	0. 6
	16	0. 6643	0. 7324	0. 0625	0. 1361	0. 7
	17	0. 1350	0. 1423	0. 5313	0. 8482	0. 1
	18	0. 4266	0. 5070	0. 1250	0. 8586	0. 5
	19	0. 1490	0. 1765	0. 1937	0. 1995	0. 1
	20	0. 2587	0. 3662	0. 5313	1. 0000	0. 2
	21	0. 7203	0. 8028	0. 1875	0. 4346	0. 7
	22	0. 9301	0. 9014	0. 9688	0. 8691	1. 0
	23	0. 3287	0. 3239	0. 2813	0. 6702	0. 3
	24	0. 6084	0. 5211	0. 2813	0. 4346	0. 7

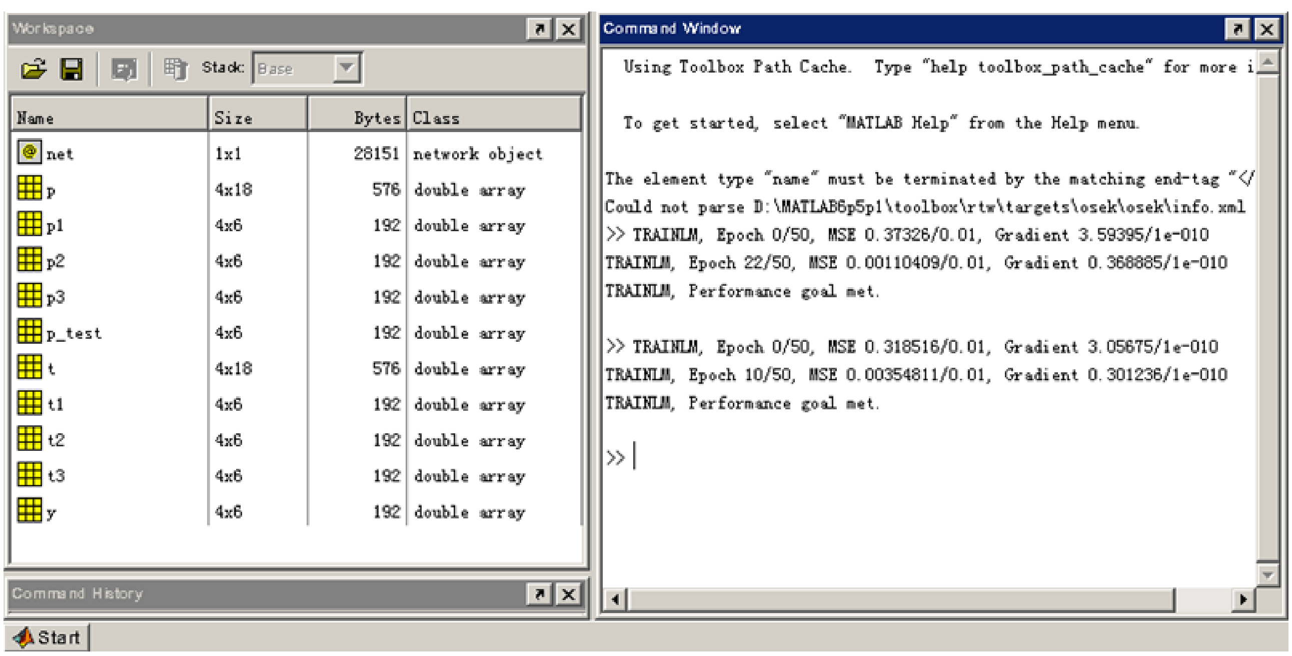


Figure.7 Training Results of the Network

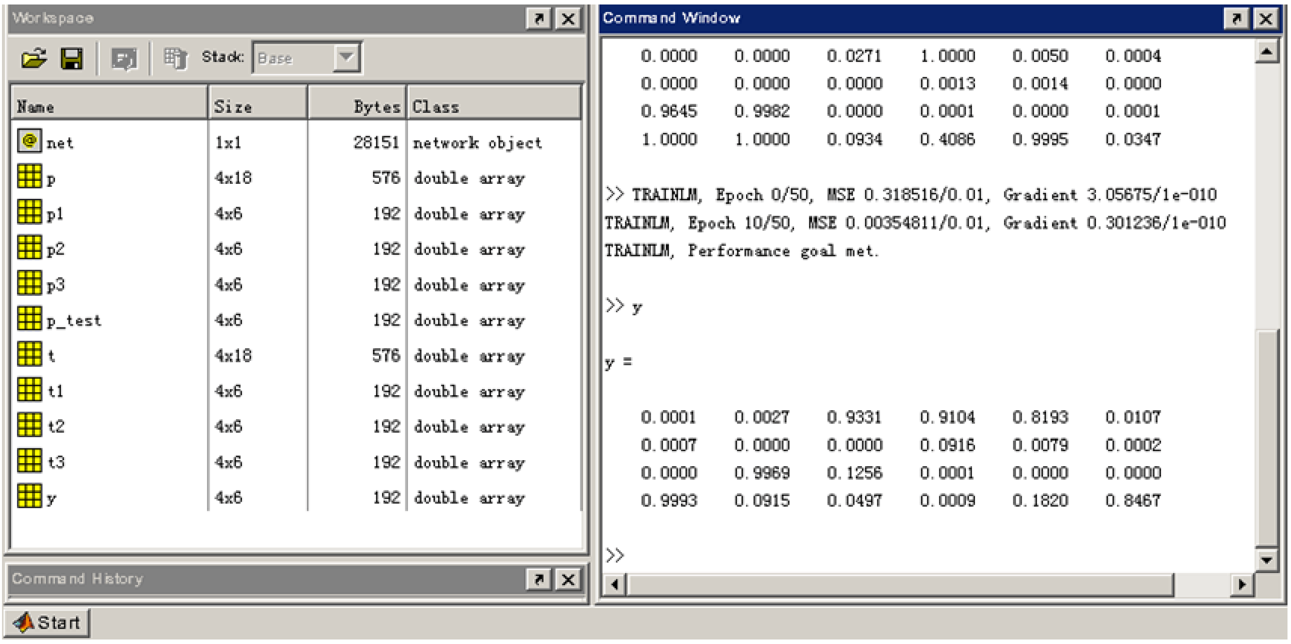
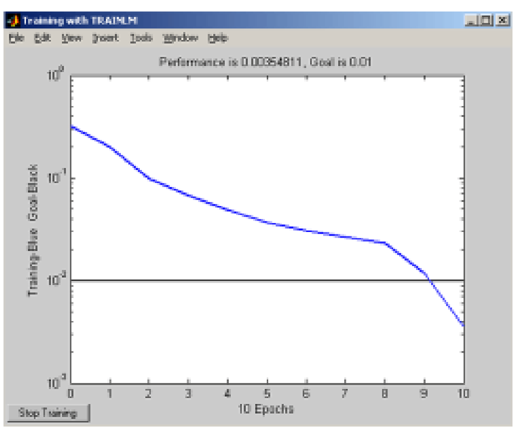


Figure.8 Output of the Program

By comparing with the actual weather, the prediction error of the network is 0.1580. Considering the small sample size, this is an acceptable result, and the forecast error curve of the network is shown in figure 10.



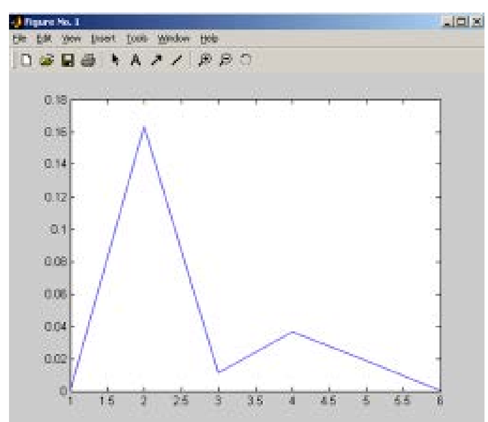


Figure.9 Training Result Curve of Program Figure.10 Prediction Error Curve Of Network

6.2 Simulation Result Analysis

With the table in February 1st -2 month 24 days from February 25th to March 2nd as the training samples for the test sample, forecast March 3rd to 10 the size of AQI, the predicted size of the input model, and the model forecasts to March 3, 2016 10, the Shijiazhuang air AQI size.

The predicted results of anti normalization, get the prediction results of the models, for the city of Shijiazhuang from March 3, 2016 to 10, the actual measurement of air pollutants AQI sizes were compared with the neural network prediction size, figure 11 is the actual measurement sizes of AQI, Figure 12 for the prediction of the size.

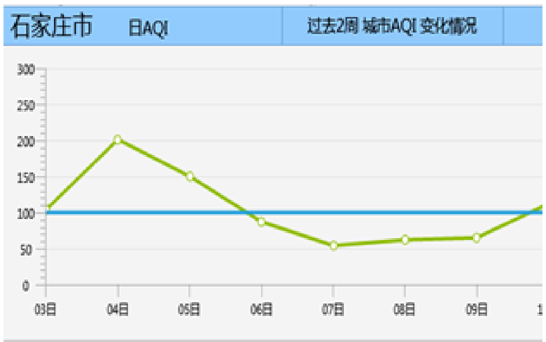


Figure.11 Real AQI Sizes

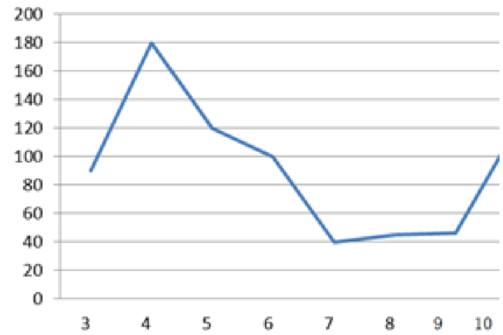


Figure.12 The Size of the Predicted AQI

7. Conclusion

The neural network model, we can see the future 10 days the AQI size and the actual size is basically the same, because the SO2 CO2 is one of the main causes of fog and haze, haze causes influence by temperature, wind and humidity, so with the actual monitoring data to increase the number of input nodes. And the application of neural network in the prediction of AQI, will get the desired results, more accurate.

Using MATLAB neural networks does not require tedious programming like traditional methods. He can model BP networks efficiently, accurately and quickly, and forecast the haze of the weather. Forecast of fog and haze has high application size. At the same time, the establishment of the neural network model provides a new general method, which can be used in different conditions of prediction and prediction.

Introduction: School of information technology, Hebei Normal University, associate professor, master’s degree, research field: computer and application, neural network

Acknowledgment

This Project Supported by the National Natural Science Foundation of China(No.61572170), Natural Science Foundation of Hebei Province of China(No. F2016205023), Educational Commission of Hebei Province of China(No. SLRC2017042)

Reference

[1] GaoNing. Forecast and MATLAB implementation of crop pest forecasting based on BP neural network. Anhui Agriculture University, 2003,29 (2): 191-151.

[2] Ma Limei. Neural network stock forecasting system based on.2005 in Beijing University of Posts and Telecommunications.

[3] Ai Hongfu, et al. Forecast of fog and haze based on Bp artificial neural network. Computer simulation, 2015 (1): 402:406