



Research on prediction of environmental aerosol and PM_{2.5} based on artificial neural network

Xianghong Wang^{1,2,3} · Baozhen Wang^{1,2,3}

Received: 11 August 2018 / Accepted: 26 October 2018 / Published online: 8 November 2018
© Springer-Verlag London Ltd., part of Springer Nature 2018

Abstract

With the increasing severity of air pollution, PM_{2.5} in aerosols, as the most important air pollutant, has adversely affected people's normal production, life and work, and has caused harm to people's health. Scientific and effective prediction of PM_{2.5} can enable people to take precautions in advance to avoid or reduce harm to the human body. Therefore, the prediction of PM_{2.5} concentration has become a topic of great practical significance. This paper selects the air quality data released in real time, obtains the historical monitoring data of air environmental pollutants, and normalizes the data, then divides the sample data, and divides it into training data set and test data set in appropriate proportion. Design the optimal network structure based on BP neural network. An improved neural network is proposed, and the neural network is optimized using genetic algorithms. The preprocessed data are input into the network for training and testing. The fitting and prediction results were statistically and comparatively analyzed. The data results show that the neural network optimized by genetic algorithm has better performance in PM_{2.5} mass concentration prediction, which improves the accuracy of prediction results and reduces the error rate.

Keywords Aerosol · PM_{2.5} · Neural network · Prediction

1 Introduction

Air pollution is a serious problem on a global scale, especially in developing countries, where the global economy is developing rapidly and the degree of industrialization is constantly increasing. This has led to a continuous increase in the total amount of pollutants in various places, the scope of pollution continues to expand, the quality of air has also deteriorated drastically, and smog has increasingly appeared around the world. Aerosols are a

general term for solid and liquid particulate matter suspended in the atmosphere. The aerodynamic diameter of the particles is between 0.001 and 100 μm . The sources of particulate matter in the atmosphere can be divided into two categories: one is called natural source, and the other is called artificial source [1]. PM_{2.5} is called fine particles and is derived from the combustion of coal, gasoline, diesel, kerosene, natural gas and organic matter [2]. Atmospheric fine particulate matter (PM_{2.5}) causes many environmental and health problems, and the relationship between environmental PM levels and health effects is found in many epidemiological studies [3, 4]. PM_{2.5} can penetrate deep into the lungs, irritating and corroding the alveolar wall, thereby impairing lung function [5]. Microorganisms in PM_{2.5} and PM₁₀ are thought to be responsible for the spread of various allergies and respiratory diseases [6]. PM_{2.5} is the primary pollutant in the air today and the main factor in the formation of haze weather [7]. Accurate prediction of its mass concentration is an important criterion for judging air quality and an important basis for making correct strategic decisions.

By investigating the concentration trends of ambient air pollutants, the effects of air pollution emission control

✉ Xianghong Wang
wangxianghong@yznu.edu.cn

Baozhen Wang
20170076@yznu.edu.cn

¹ Chongqing Multiple-Source Technology Engineering Research Center for Ecological Environment Monitoring, Yangtze Normal University, Chongqing 408100, China

² Green Intelligence Environmental School, Yangtze Normal University, Chongqing 408100, China

³ Collaborative Innovation Center for Green Development in Wuling Mountain Areas, Yangtze Normal University, Chongqing 408100, China

policies can be assessed [8]. Biancofiore F [9] and others used three models to analyze data collected over 3 years in urban areas along the Adriatic coast, all of which were used to predict PM_{2.5} concentrations. Liu H [10] used 22 monitoring points to collect data on O₃, PM_{2.5} and PM₁₀, and conducted a comprehensive study on air quality in Guangzhou and surrounding areas. Mchenry J N [11] and others used an airborne quality prediction model including remote sensing aerosol optical depth and ground PM_{2.5} observation data assimilation, which can achieve quantitative improvement in model prediction skills, thereby improving the accuracy of predictions issued by the official. Li [12] proposed a new long short-term memory neural network extension (LSTME) model for automatically extracting inherent useful features from historical air pollutant data and incorporating auxiliary data into the proposed model to improve performance. Song [13] predicted 13 monitoring stations in Auckland, New Zealand, using spatiotemporal PM_{2.5}. Spatial data-assisted incremental support vector regression was used to predict spatiotemporal PM_{2.5}. Tan [14] and others used multiple linear regression and principal component analysis to predict the ozone column in Malaysia peninsula. Ghazali [15] and others used time series plots to study the conversion of nitrogen dioxide (NO₂) to ozone (O₃) in urban environments and used multiple linear regression models to predict O₃ concentrations based on data on environmental pollutant concentrations and meteorological variables. Donnelly [16] and others proposed a nonparametric kernel regression and multiple linear regression to establish a statistical model of NO₂ real-time hourly forecasting. Mishra [17] and others introduced a model based on statistical regression and specific computational intelligence to predict the hourly NO₂ concentration of the Agra Taj Mahal Historical Monument. He [18] established a multiple linear regression model by observing the concentration of pollutants and the high-resolution weather data of the weather research and prediction models to study the relationship between the concentration of air pollutants and the ten meteorological parameters in winter, and to estimate the concentration of NO₂ and PM₁₀ in the northwest of Lanzhou.

In recent years, artificial neural networks have been widely used in meteorological forecasting. Artificial neural network methods are widely used in the prediction of pollutant concentrations because they can build very complex nonlinear models. Fang M [19] and others discussed the feasibility of artificial neural network technology in predicting airborne particle pollution in major traffic routes. Zheng [20] and others constructed a prediction model based on RBF neural network to predict the concentration of PM_{2.5} and compared it with the traditional BP network model. Yang [21] and others proposed a prediction method based on TS fuzzy neural network based on the prediction of PM_{2.5}

concentration in air. The PM_{2.5}/hour concentration prediction model was established based on the monitoring data of Baoji monitoring station. Tian-Cheng [22] and others calculated the influence of weather conditions and internal factors of other air pollution particle concentrations on PM_{2.5} concentration and selected PSO and fuzzy neural network as optimization methods to make the prediction method more effective. Hu [23] proposed an Elman neural network prediction method based on chaos theory. The reconstructed phase space and the future concentration of PM_{2.5} are, respectively, used as the input and output of the Elman neural network with chaotic theory (Elman-chaos). According to the value of PM_{2.5} in Langfang City in 2015, Zhu [24] and others proposed a new prediction technique based on ARMA and improved BP neural network to predict PM_{2.5} concentration. Zhou [25] and others proposed a recursive fuzzy neural network prediction method to predict PM_{2.5} concentration. Papadopoulos [26] proposed an extension to conventional regression neural networks (NNs) for replacing the point predictions they produce with prediction intervals that satisfy a required level of confidence.

In this paper, the historical monitoring data collected will be trained using neural network to obtain the optimal training network, and the optimal network will be used to predict PM_{2.5} in the future. With the increasing number of monitoring data, find new optimal networks, update the optimal training network, and gradually improve the network prediction accuracy. The neural network is combined with the genetic algorithm to judge the pros and cons of the combined network performance through experimental data, so as to continuously improve the prediction model and the combination mode. By evaluating the accuracy and generalization ability of the network prediction results, the advantages and disadvantages of the new network are evaluated, so that the network can finally have faster convergence speed, better generalization ability and smaller prediction error. Provide real-time PM_{2.5} forecasts and air quality indicators to provide a basis for social activities and regulatory control departments.

2 Proposed methods

2.1 Artificial neural networks

Artificial neural network is a nonlinear, adaptive information processing system composed of a large number of processing unit interconnections. It attempts to process information by simulating the way the brain neural network processes and memorizes information. A large number of simple artificial neurons are interconnected to form an artificial neural network that can perform complex functions. Artificial neurons have the following characteristics:

1. Each neuron is a multi-input/single-output nonlinear information processing unit;
2. Each artificial neuron has an additional input signal threshold in its modeling;
3. The output of the neuron has a closed-value characteristic. When the input sum exceeds its threshold, the neuron is activated and impulsive; when the input sum does not exceed the threshold, the neuron is not activated and does not emit an impulse.

As shown in Fig. 1, x_1, x_2, \dots, x_n in the figure are multiple input components of neurons expressed by X and $w_{i1}, w_{i2}, \dots, w_{in}$ are the n components of the network connection weight of the i neuron (also called the prominent coupling coefficient) expressed by W_i . The connection weight component acts like a synaptic connection strength in a biological neuron to store the learned knowledge. The input component x_n is, respectively, connected with each weight component W_{in} . When the external stimulus reaches a certain value, the neuron will give an impulse. This value is expressed by θ_i in the artificial neuron, and θ_i is also the threshold. The threshold determines whether the neuron will be activated to make an impulse. Σ is the part of the neuron used to weight the sum, called the summation unit. The input component x_n of each afferent neuron i and its corresponding weight component W_{in} are calculated by the weighted summation. When the results exceeds the threshold θ_i , the neuron is activated to give an impulse. Otherwise, the neuron will still in a state of inhibition and it will not emit impulsive. There are two commonly used weighted summation functions (also called basis functions):

$$net_i = \sum_{j=1}^n \omega_{ij}x_j + \theta_i \tag{1}$$

$$net_i = \sqrt{\sum_{j=1}^n (x_j - w_{ij})^2} - X - W. \tag{2}$$

Formula 1 is a linear function, and Formula 2 is a distance function. In most neural networks, the basis function uses a linear function as its weighted summation function. The distance function is often used as a weighted

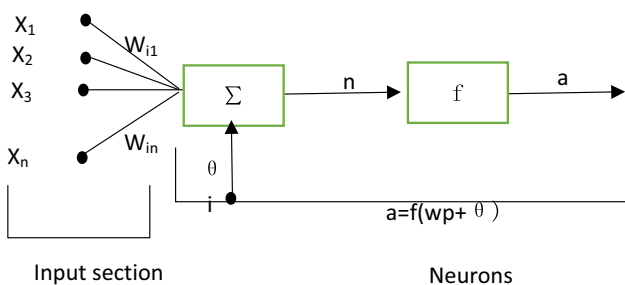


Fig. 1 Artificial neuron model

summation function of the radial basis function neural network. The weighted summation value in a neuron cannot directly become the output of a neuron. Its value needs to undergo a series of changes of the function $f()$, which is limited to a certain range, $f()$ is called the activation function, and the final output of the neuron is as follows:

$$y_i = f(net_i). \tag{3}$$

In Formula 3, y_i is the output of the i neuron and net_i is the result of the weighted sum of the i neuron. When two or more neurons are connected to each other, a basic artificial neural network is formed. The learning ability of artificial neural network is realized in the constant adjustment of network connection weights and thresholds.

2.2 BP neural network

Back propagation neural network is referred to as the BP network and is developed on the basis of the BP algorithm. The characteristic of the BP algorithm is that in the process of backward error propagation, the gradient descent method is used to modify the connection weights and thresholds of the network, so that the output of the network is constantly forced to be expected in the process of network parameters correction. The neural network optimized by BP algorithm is called BP neural network. It is a multilayer feedforward neural network trained by the error inverse propagation algorithm. The main feature of the network is the forward transmission of the input signal and the backward propagation of the error signal. Its network structure is shown in Fig. 2.

As shown in Fig. 2, the network structure is divided into three layers, namely an input layer, an implicit layer and an output layer. The hidden layer can be a single layer or a plurality of layers, and the dimension of the input quantity determines the number of neurons in the input layer. The

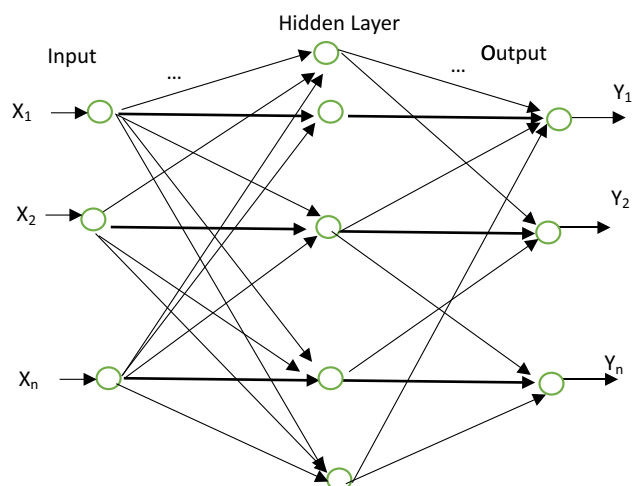


Fig. 2 BP neural network structure

input quantity $X = \{x_1, x_2, \dots, x_n\}$ enters from the input layer and is processed layer by layer through the hidden layer to finally obtain the output value of the network. The entire training process can be divided into two phases:

1. The input signal propagates along the forward direction of the network. When the input signal is forward propagating, the input signal enters the network from the input layer and is transformed by the weighted summation of each layer and the activation function, and then output through the output layer. At this time, the output of each layer of neurons will only affect the neurons in the next layer, and the input will only receive the influence of the output of the previous layer of neurons, and the weight and threshold of the network remain unchanged.
2. The training error signal propagates in the reverse direction of the network. When the error between the network output value y_d and the expected value y_e does not reach the target, and the network discovery error is too large, the error signal is forwardly propagated, and the error signal is backwardly propagated from the output layer to the input layer in the original network direction. In this process, the weight and threshold of the network are continuously corrected according to the gradient value of the training error, and finally the output of the network can be gradually approached to the desired result, so that the two phases are repeatedly looped to finally obtain the desired output.

2.3 BP network learning algorithm

Let the input layer of a BP network have n neurons, the hidden layer has P neurons, and the output layer has q neurons. Define the input vector as $X = [x_1, x_2, \dots, x_n]$, the hidden layer input vector $hi = [hi_1, hi_2, \dots, hi_p]$, the hidden layer output vector $ho = [ho_1, ho_2, \dots, ho_p]$, the output layer input vector $yi = [yi_1, yi_2, \dots, yi_q]$, the output layer output vector $yo = [yo_1, yo_2, \dots, yo_q]$, the expected output vector $d_o = [d_1, d_2, \dots, d_q]$, the connection weight of the input layer and the hidden layer is w_{ih} , the connection weight of the hidden layer and the output layer is w_{ho} , the threshold of neurons in each layer of the hidden layer is b_h , the threshold of neurons in each layer of the output layer is b_o , the number of samples is $k = 1, 2, 3, \dots, m$, and the activation function is $x(k) = (x_1(k), x_2(k), x_3(k), \dots, x_n(k))$. The error function is shown in Formula 4:

$$e = \frac{1}{2} \sum_{o=1}^q (d_o(k) - y_{o_o}(k))^2. \quad (4)$$

The learning process of the algorithm mainly has the following steps:

1. Set the initial parameters of the entire network. Each connection weight is assigned a random number within the interval $(-1, 1)$, and the Formula 4 is taken as the error function as e , given the accuracy of the calculation and the maximum number of learning M .
2. Randomly select the k th sample data and its corresponding expected output within a given data sample space.

$$x(k) = (x_1(k), x_2(k), x_3(k), \dots, x_n(k)) \quad (5)$$

$$d_o(k) = (d_1(k), d_2(k), x_3(k), \dots, d_q(k)). \quad (6)$$

3. Calculate the input and output of each layer of neurons in the hidden layer

$$hi_h(k) = \sum_{i=1}^n w_{ih} x_i(k) - b_h, h = 1, 2, 3, \dots, p \quad (7)$$

$$ho_h(k) = f(hi_h(k)) \quad h = 1, 2, 3, \dots, p \quad (8)$$

$$yi_o(k) = \sum_{h=1}^p w_{ho} ho_h(k) - b_o, o = 1, 2, 3, \dots, q \quad (9)$$

$$y_{o_o}(k) = f(yi_o(k)), o = 1, 2, 3, \dots, q. \quad (10)$$

4. Calculate the partial derivative $\delta_o(k)$ of the error function to each neuron in the output layer based on the actual output of the network and the given expected output

$$\frac{\partial e}{\partial w_{ho}} = \frac{\partial e}{\partial yi_o} \frac{\partial yi_o}{\partial w_{ho}} \quad (11)$$

$$\frac{\partial yi_o(k)}{\partial w_{ho}} = \frac{\partial (\sum_h^p w_{ho} ho_h(k) - b_o)}{\partial w_{ho}} = ho_h(k) \quad (12)$$

$$\begin{aligned} \frac{\partial e}{\partial yi_o} &= \frac{\partial (\frac{1}{2} \sum_{o=1}^q (d_o(k) - y_{o_o}(k))^2)}{\partial yi_o} = -(d_o(k) - y_{o_o}(k)) y_{o_o}'(k) \\ &= -(d_o(k) - y_{o_o}(k)) f'(yi_o(k)) \triangleq -\delta_o(k). \end{aligned} \quad (13)$$

5. Calculate the partial derivative $\delta_o(k)$ of the error function for each neuron in the hidden layer according to the connection weight of the hidden layer to the output layer, the partial derivative $\delta_o(k)$ of the output layer of the error function and the output of the hidden layer.

$$\frac{\partial e}{\partial w_{ho}} = \frac{\partial e}{\partial y_{i_o}} \frac{\partial y_{i_o}}{\partial w_{ho}} = -\delta_o(k) h_{o_h}(k) \tag{14}$$

$$\frac{\partial e}{\partial w_{ih}} = \frac{\partial}{\partial h_{i_h}(k)} \frac{\partial h_{i_h}(k)}{\partial w_{ih}} \tag{15}$$

$$\frac{\partial h_{i_h}(k)}{\partial w_{ih}} = \frac{\partial (\sum_{i=1}^n w_{ih} x_i(k) - b_n)}{\partial w_{ih}} = x_i(k) \tag{16}$$

$$\begin{aligned} \frac{\partial e}{\partial h_{i_h}(k)} &= \frac{\partial \left(\frac{1}{2} \sum_{o=1}^q (d_o(k) - y_{o_o}(k))^2 \right)}{\partial x} \frac{\partial h_{o_h}(k)}{\partial h_{i_h}(k)} \\ &= - \sum_{o=1}^q (d_o(k) - y_{o_o}(k)) f'(y_{i_o}(k)) w_{ho} \frac{\partial h_{o_h}(k)}{\partial h_{i_h}(k)} \\ &= - \left(\sum_{o=1}^q \delta_o(k) w_{ho} \right) f'(h_{i_h}(k)) \triangleq -\delta_o(k). \end{aligned} \tag{17}$$

- 6. Correct the connection weight $w_{ho}(k)$ value according to the partial derivative $\delta_o(k)$ of each neuron in the output layer and the output of each neuron in the hidden layer.

$$\Delta w_{ho}(k) = -\mu \frac{\partial e}{\partial w_{ho}} = \mu \delta_o(k) h_{o_h}(k) \tag{18}$$

$$w_{ho}^{N+1} = w_{ho}^N + \eta \delta_o(k) h_{o_h}(k). \tag{19}$$

- 7. The connection weight is corrected according to the partial derivative $\delta_h(k)$ of each neuron in the hidden layer and the input of each neuron in the input layer.

$$\Delta w_{ih}(k) = -\mu \frac{\partial e}{\partial w_{ih}} = \mu \frac{\partial e}{\partial h_{i_h}(k)} \frac{\partial h_{i_h}(k)}{\partial w_{ih}} = \delta_h(k) x_i(k) \tag{20}$$

$$w_{ih}^{N+1} = w_{ih}^N + \eta \delta_h(k) x_i(k). \tag{21}$$

- 8. Calculate the global error.

$$E = \frac{1}{2m} \sum_{k=1}^m \sum_{o=1}^q (d_o(k) - y_o(k))^2. \tag{22}$$

- 9. Determine whether the network error satisfies the following two requirements: the network error reaches the preset accuracy; the learning frequency is greater

than the set maximum number. The algorithm is terminated as soon as it is satisfied, otherwise, the next learning sample and its expected output are selected, and a jump is made to step 3.

2.4 Genetic algorithm

The main principle of genetic algorithm is: after the generation of the first generation population, according to the principle of survival of the fittest, each generation of the descendant group, the algorithm uses the fitness function and the genetic manipulation to screen the individual, so as to produce a new generation of population. Through generational evolution, each generation’s population is more adaptive to the environment than the previous generation, and the final generation’s population can be regarded as the optimal solution to the problem. The genetic algorithm is mainly divided into the following steps:

1. Encoding the initial individual and generating the initial population

When using genetic algorithms to solve problems, it is first necessary to map from the solution space of the problem to the genotype space of the genetic algorithm. This process is called coding. It is necessary to consider the completeness, soundness and non-redundancy of coding. That is, all possible solutions of the solution space have corresponding gene strings, and each gene string has a corresponding possible solution, and the possible solutions are also one-to-one corresponding to the gene strings. Binary coding or real coding is usually used to encode the initial individual.

In this paper, the real number coding method is used. Each individual is coded into a real string. Then, M individuals are randomly generated to form a group (initial population), and the maximum evolution times are set up. The algorithm uses the initial population as the starting point for evolution.

2. Evaluation of individual fitness

According to the evolutionary goal, a suitable evaluation mechanism of individual adaptability, fitness function, is used to calculate the fitness of the individual in the population according to the fitness function. In biological evolution, the more highly the adapted organisms have the stronger vitality, the easier it is to survive. In the genetic algorithm, the solution space of the problem is mapped onto the gene space, and the fitness evaluation of each individual reflects the principle of survival of the fittest. Each individual corresponds to a fitness function value, and the preferred individual has a better function value. The fitness value is the only criterion for judging whether an

individual can exist, so the fitness function determines the evolution direction of the population.

3. Performing a selection operation

The selection operation refers to the selection of the individual in the old population to a new population according to the fitness value of the individual, and the purpose is to make the superior individual have the opportunity to produce the next generation as the parent, to avoid the loss of the superior gene group and to improve the global convergence rate.

There are many methods for selecting operation. This paper uses the roulette selection method, which is a simple selection method based on the ratio. The probability that each individual in the population is adopted is related to the proportion of the fitness value. Let f_i be the fitness of individual i , the number of individuals in the population is M , and the formula is:

$$P_i = \frac{f_i}{\sum_{k=1}^M f_k} \quad 1 \leq i \leq M. \quad (23)$$

p_i is the ratio of the fitness value of the individual i in the sum of the fitness values. The individuals selected by this method are not necessarily the individuals with the highest fitness ratio, but the higher the fitness ratio, the easier the individuals are selected.

4. Cross operation

Two chromosomes are selected from the group to be paired with each other, and one or more of them are exchanged in a certain way to combine some of the genes, and finally two new individuals are formed. This probability is called crossover probability. The crossover probability determines the frequency of crossover operations, so the higher the crossover probability, the faster the individual's update in the population, but the excessive crossover probability will lead to premature convergence to some extent. The common methods of cross operation are: single-point cross method, multipoint cross method, uniform cross method, arithmetic number cross method and so on. Since the coding method of this paper is real number coding, the cross operation selects the single-point real number intersection method:

$$a_{mj} = a_{mj}(1 - b) + a_{nj}b \quad (24)$$

$$a_{nj} = a_{nj}(1 - b) + a_{mj}b. \quad (25)$$

a_{ij} means that the i th individual is in the j th position, and b is a random number whose value interval is $[0,1]$. The function of the formula is to cross the j th position of the chromosomes a_m and a_n .

5. Mutation operation

The mutation operation refers to selecting an individual from the population and performing an equidistant exchange of two places on the individual coding string with a certain probability to generate a new individual. In the traditional genetic algorithm, the operator and the population algebra are not directly related. When the algorithm evolves to a certain extent, it is difficult to be optimized due to the lack of local search ability. The variation determines the local search ability of the algorithm. Individual variation depends on the probability of mutation. Generally speaking, the greater the probability of mutation, the stronger the ability of the algorithm to explore a new solution space, but the excessive probability of mutation will affect the convergence of the algorithm. The formula for the mutation operation in this paper is as follows:

$$a_{ij} = \begin{cases} a_{ij} + (a_{ij} - a_{\max}) * f(g) & r > 0.5 \\ a_{ij} + (a_{\min} - a_{ij}) * f(g) & r \leq 0.5 \end{cases} \quad (26)$$

$$f(g) = r_2 \left(\frac{1-g}{G} \right)^2. \quad (27)$$

a_{\max} and a_{\min} are the upper and lower bounds of a_{ij} , and g is the number of current iterations. In the formula, n is a random number with a value interval of $[0, 1]$, and y_d is the maximum number of evolutionary times. Together, they complete the operation of mutating the j th position of the i th individual. The crossover and mutation operators in the genetic algorithm work together to complete the global search and local search of the entire space, so that the genetic algorithm can have the ability to complete the optimization problem.

6. Determination of termination conditions

The termination condition of the algorithm is: whether the evolutionary algebra has reached the maximum, and whether the error of an individual in the population has reached the requirement. If both of the above conditions have been met, the result is output and terminated, and if not met and the maximum number of evolutionary times is not reached, step 2 is returned.

2.5 BP neural network optimized by genetic algorithm

Genetic algorithms and neural networks are two hot research branches of current bionic algorithms. In essence, genetic algorithm is a kind of direct problem-oriented intelligent problem optimization method. It uses multipoint and path optimization methods. In the optimization process, multiple individuals in the group are processed and evaluated at the same time, so they can find the global optimal solution or the best performance suboptimal

solution with the greatest probability and have a good global search ability. In the optimization process, the genetic algorithm can directly use the fitness value of the objective function as the optimization direction and greatly improve the optimization efficiency.

Since the BP neural network uses the learning based on the gradient descent method, it will have problems such as insufficient network global search ability, easy to fall into local optimum and slow training speed. Using genetic algorithm to optimize BP neural network is to use genetic algorithm to optimize the threshold and weight of BP neural network. Each individual in each generation of the genetic algorithm contains the ownership value and threshold in the BP network. By calculating the fitness value of each individual and carrying out genetic manipulation (selection, cross, variation) to find the fitness value of each individual, and the BP network finally assigns the values to the network using the weights and thresholds obtained by this optimal individual.

When BP neural network is combined with genetic algorithm, it will get a new network with strong local optimization ability, good global search and fast convergence. The flow of BP neural network optimized by genetic algorithm is shown in Fig. 3.

3 Simulation study

3.1 Data source and processing

The air quality monitoring data used in this paper are the real-time air quality data released by the Ministry of Environmental Protection of China, including CO hourly average, NO₂ hourly average, SO₂ hourly average and PM10 hourly average. Taking 900 data which use 80% of the data (720) as the original sample data to train the network and use 20% of the data (180) as the test data to test the training results. The hourly PM2.5 mass concentration is predicted under MATLAB R2014b.

There are two sets of input matrices and two sets of output matrices in the network, which are used for network training and using trained network parameters for predictive simulation. The input data are represented by a 4 * m matrix.

$$A = \begin{matrix} a_{11} & a_{12} & k & a_{14} \\ a_{21} & a_{22} & k & a_{24} \\ M & M & O & M \\ am1 & am2 & k & am4 \end{matrix}$$

Each column represents an influence factor when predicting the mass concentration of PM2.5. a1–a4 represent the measured values of the four data of CO, NO₂, SO₂ and PM10 in the *i*th data. Each row represents the measured

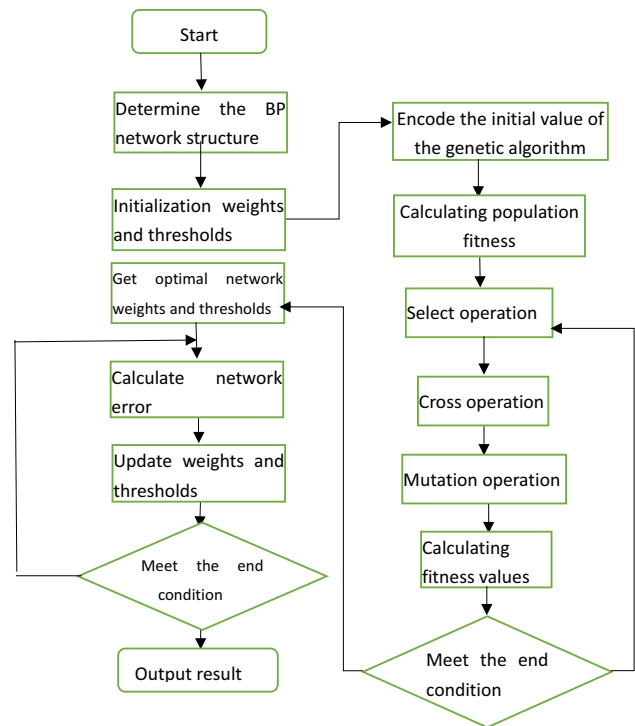


Fig. 3 Algorithm flowchart of GA-BP

value of the four data points at the same time, where *m* represents the time sequence. The output data are represented by a column vector of *m* * 1: $D = [d1, d2, d3, \dots, dm]$, where *dm* represents the measured value of PM2.5 at that time.

3.2 Data normalization

The normalization process is to ensure that each data item can take values in the same interval, eliminate the influence on the network output result caused by the different data dimensions, and prevent the feature data items with small order of magnitude caused by large order of magnitude difference which cannot play a role. Data normalization is to unify data of different dimensions into a reference frame and to remove data from dimensional effects. Using normalized data input network, the convergence speed of the program is accelerated at run time. If the input data are normalized when they are inputted, the output results of the network are inverse-normalized and the output value is restored to the original dimension value. In this paper, the `mapminmax()` function in MATLAB is used as a normalization method, which belongs to the method of range normalization. `mapminmax()` can normalize each line of the matrix to [-1,1]. Its general form is given in Table 1.

The normalized data have a value limited to [-1,1]. At 12 o'clock, the values of CO and PM10 are -0.534 and

Table 1 The data after normalization training

Time	CO	No ₂	So ₂	PM10
12:00	- 0.534	0.489	- 0.902	0.813
13:00	- 0.690	- 0.014	- 0.887	- 0.839
14:00	- 0.73	- 0.26	- 0.913	- 0.839
15:00	- 0.649	- 0.136	- 0.877	- 0.836
16:00	- 0.6471	- 0.178	- 0.909	- 0.876
17:00	- 0.501	- 0.369	- 0.991	- 0.811
...				
23:00	- 0.671	- 0.067	- 0.995	- 0.826
00:00	- 0.541	- 0.205	- 0.991	- 0.8391

- 0.813, and the magnitude difference is significantly reduced.

3.3 Experimental prediction simulation

3.3.1 PM2.5 prediction based on BP neural network

After training the network using 720 of the filtered 900 data, the trained network is used to predict the 180 data of the test set. The following results are obtained as shown in Fig. 4:

The prediction results of the BP neural network have a large deviation from the expected value when the PM2.5 mass concentration is abrupt. In order to display the prediction performance of the network more intuitively, all the fitting data and forecast data are counted, and the acceptability of the prediction results of PM2.5 quality concentration is further analyzed, and the acceptability of the prediction results is judged by the formula:

$P_i = \frac{|y_{di} - y_{ci}|}{y_{di}} * 100\%$, where y_{di} is the measured value, y_{ci} is the predicted value, and P_i is the absolute error of the prediction result. Set when $P_i \leq 10\%$, the task prediction result is very good. When $10\% \leq P_i \leq 30\%$, the prediction result is considered acceptable. When $30\% \leq P_i \leq 50\%$, the prediction result is considered to be poor. When $50\% \leq P_i$, the prediction result is considered unacceptable. According to the rule, Table 2 can be obtained.

In the prediction results of PM2.5 mass concentration based on BP neural network, the acceptable rate of fitting data is about 86.9% and the acceptable rate of test results is about 80%. The average relative error of fitting is 15.1% and the maximum relative error is 90.1%. The average relative error of test is 20.5% and the maximum relative error is 93.4%. It can be seen from the above data analysis that the network does not show excessive gaps in data fitting and testing. Therefore, the network does not have a fitting phenomenon during the training process, and the network has generalization.

BP neural network has strong nonlinear mapping ability, high self-learning and self-adaptive ability, good generalization ability and fault tolerance. However, since BP neural network is a network model based on negative gradient descent algorithm, it still has many defects and shortcomings. The network can converge to a minimum value at the end of each training, but it cannot guarantee convergence to the global minimum after each training. That is to say, the network is easy to fall into the local minimum; the learning speed of the network is fast enough, and the convergence speed is slow. Based on the defects of the BP neural network above, combined with the actual situation in the prediction of PM2.5 mass concentration,

Fig. 4 BP neural network prediction of PM2.5

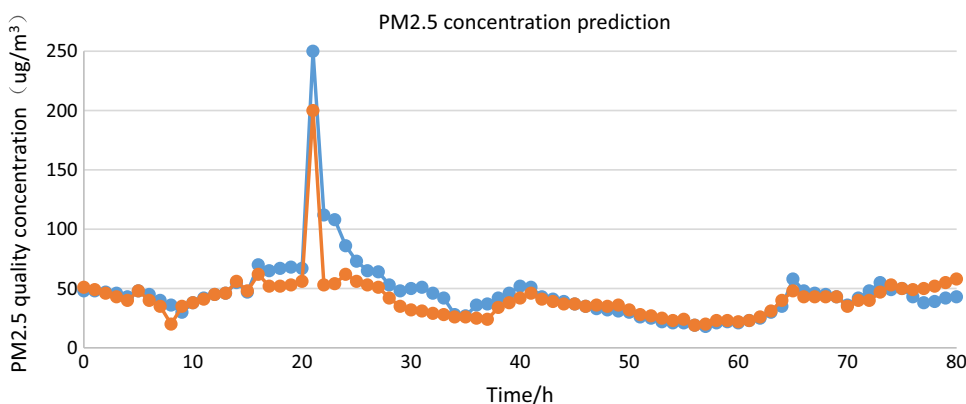


Table 2 Acceptable degree of PM2.5 predictive based on BP neural network

Data source	Total data	Very good	Acceptable	Poor	Unacceptable
Training data	720	412	207	63	38
Test data	180	75	69	21	15

the BP neural network method optimized by genetic algorithm is used to predict it.

3.3.2 PM2.5 prediction of BP neural network based on genetic algorithm optimization

Since the absolute value of the training result error of BP neural network is taken as the individual fitness value, the lower the fitness value, the higher the corresponding individual fitness. Therefore, the individual fitness evolution curve is first obtained through MATLAB, as shown in Fig. 5.

As can be seen from the figure, the average fitness value decreases fastest in the 1–10 evolution, the speed slows down in the 11–30 evolution, and the fitness value tends to be stable and optimal after 30 times. The initial weights and thresholds obtained by the optimal individual are assigned to the BP neural network, and the network is trained using the normalized training data. The network iterates 100 times, the training data are fitted after the training, and the result is shown in Fig. 6.

From Fig. 6, it can be seen that the fitting results of the PM2.5 quality concentration of the training data are consistent with the overall trend of the measured values. The actual output is mostly coincided with each point of the predicted output. Although there is a certain error between the two, the total error difference is smaller in the case of better fitting results of the training set, using the trained

network to predict the PM2.5 quality concentration of the test data, and the result is shown in Fig. 7:

From the result of PM2.5 mass concentration prediction in the test data set of Fig. 7, it can be seen that the value of PM2.5 is sharply changed at the 21st time point, but the prediction result is very close to the measured value, and the prediction error does not increase due to the mutation, and only a small prediction bias occurs near the 21, 32 and 58 time points in this data segment. In the 80 test data of the data segment, there are only eight error points. By comparing the network prediction results between the test set and the training set in Figs. 6 and 7, it can be found that the fitting result is basically the same as the test result, and the network does not have an over-fitting phenomenon in which the fitting result is good and the test result is poor. According to the acceptability formula, statistical analysis of all data is obtained in Table 3.

By analyzing the data in Table 3, it can be found that the PM2.5 quality concentration prediction is performed on the training data set and the test data set, respectively, using the trained network: 94% of the data in the fitting results reached an acceptable level, of which 79.4% were very good; 89.4% of the predictions reached an acceptable level, of which 69.7% of the forecast results reached a very good degree. The fitting and test prediction results of GA-BP increased by 6% and 9%, respectively, but the prediction results reached very good data, from 412, 207 to 472,217, which increased by 13% and 9%, respectively, which

Fig. 5 Fitness curve chart of GA-BP

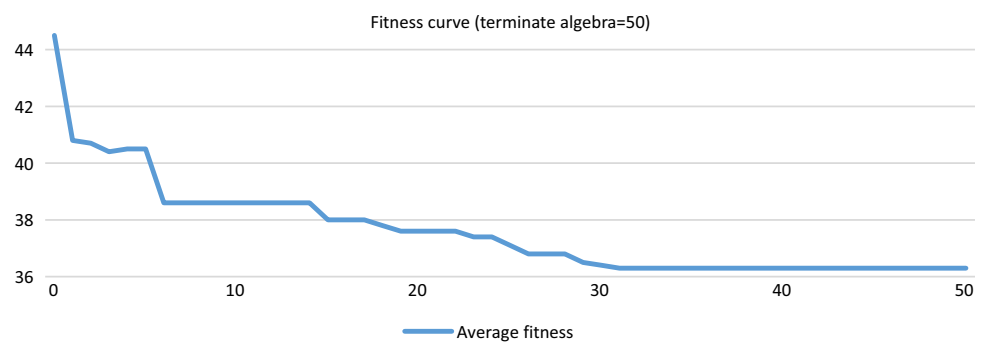


Fig. 6 PM2.5 prediction fitting result based on GA-BP neural network

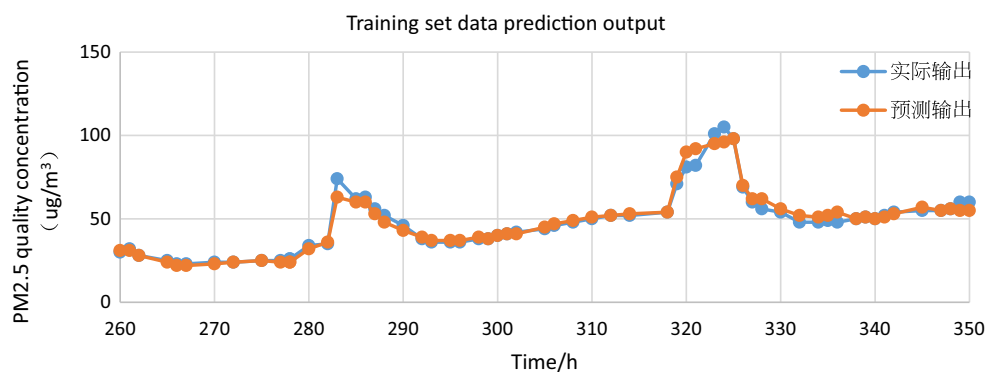


Fig. 7 PM2.5 prediction test result based on GA-BP neural network

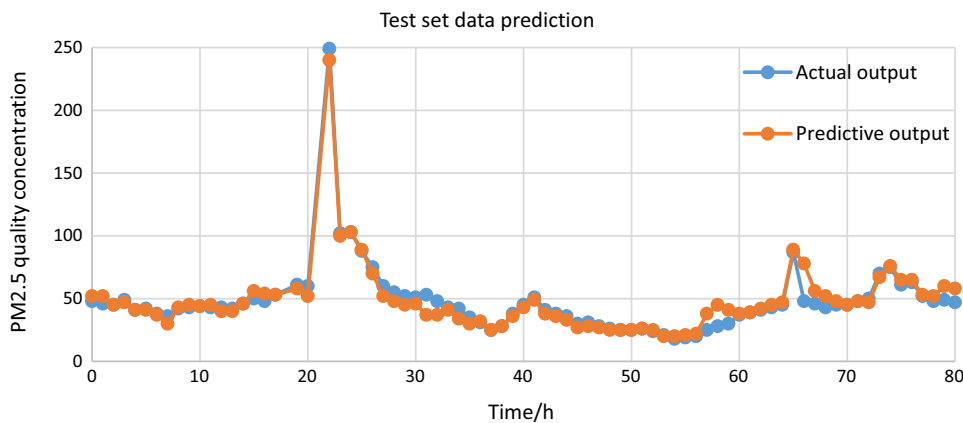


Table 3 Acceptable degree of PM2.5 predictive based on BP neural network

Data source	Total data	Very good	Acceptable	Poor	Unacceptable
Training data	720	472	217	21	10
Test data	180	132	29	13	6

Table 4 The contrast between two kinds of network prediction results

	GA-BP		BP neural network	
	Training (%)	Test (%)	Training (%)	Test (%)
Average relative error of PM2.5	96	13.3	15.1	20.5
Maximum relative error of PM2.5	59.6	56.3	90.1	93.4

means that the prediction error of the entire network becomes smaller.

3.4 Experimental prediction results and analysis

For further analysis, the prediction results of BP neural network optimized by genetic algorithm are compared with the prediction results based on BP neural network. The results are given in Table 4.

In the training data, the average relative error and maximum relative error of the GA-BP neural network PM2.5 prediction are lower than those of the BP network. In the test data, the average relative error and maximum relative error of the GP-BP neural network PM2.5 prediction were lower than those of the BP network. From the above data, GA-BP’s network generalization can be better than BP neural network.

4 Relate work

Two kinds of neural networks are used to simulate the PM2.5 mass concentration prediction based on MATLAB in this paper, the results are compared by the prediction results, and the network is evaluated. The results show that the prediction results of neural network optimized by

genetic algorithm are better than BP neural network. The BP neural network optimized by genetic algorithm not only has the generalization and mapping ability of neural network, but also has the global search ability of genetic algorithm and get the good result. The prediction results of BP neural network based on genetic algorithm optimization are improved compared with BP neural network in prediction accuracy and accuracy. This shows that it is feasible to use neural networks for such predictions. However, due to the limited data acquisition channels and data volume, the feature information contained in the actual data is not sufficient, and there are still a small amount of errors in the prediction model results.

5 Conclusion

The neural network has feasibility and good prospects in PM2.5 prediction. The research in this paper is based on the data of one sampling point, while the atmospheric particulate pollution has obvious temporal and spatial variation characteristics. It is very important to analyze the change of particulate matter with space. If the amount of data of the training samples can be expanded and the input data dimension is increased, the accuracy and accuracy of the prediction results may be further improved.

Acknowledgements The research work was supported by Chongqing Municipal Commission of social and livelihood projects (Grant No. cstc2015shmszx20010).

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

- Yang X (2013) Atmospheric particulate matter PM_{2.5} and its sources. *Front Sci* 7(2):12–17
- Yang X, Wei P, Feng L (2013) Atmospheric particulate matter PM_{2.5} and its controlling countermeasures and measures. *Front Sci* 7(3):20–29
- Tsai FC, Smith KR, Vichit-Vadakan N et al (2000) Indoor/outdoor PM₁₀ and PM_{2.5} in Bangkok, Thailand. *J Expo Anal Environ Epidemiol* 10(1):15–26
- Sahu SK, Kota SH (2017) Significance of PM_{2.5} air quality at the Indian capital. *Aerosol Air Qual Res* 17(2):588–597
- Xing YF, Xu YH, Shi MH et al (2016) The impact of PM_{2.5} on the human respiratory system. *J Thorac Dis* 8(1):E69
- Cao C, Jiang W, Wang B et al (2014) Inhalable microorganisms in Beijing's PM_{2.5} and PM₁₀ pollutants during a severe smog event. *Environ Sci Technol* 48(3):1499–1507
- Hu D, Jiang J (2013) A study of smog issues and PM_{2.5} pollutant control strategies in China. *J Environ Prot* 04(07):746–752
- Lee H, Coull BA, Bell ML et al (2012) Use of satellite-based aerosol optical depth and spatial clustering for PM_{2.5} prediction and concentration trends in the New England region, U.S. In: AGU fall meeting. AGU fall meeting abstracts, 2012
- Biancofiore F, Busilacchio M, Verdecchia M et al (2017) Recursive neural network model for analysis and forecast of PM₁₀ and PM_{2.5}. *Atmos Pollut Res* 8(4):652–659
- Liu H, Wang XM, Pang JM et al (2013) Feasibility and difficulties of China's new air quality standard compliance: PRD case of PM_{2.5} and ozone from 2010 to 2025. *Atmos Chem Phys* 13(23):12013–12027
- Mchenry JN, Vukovich JM, Hsu NC (2015) Development and implementation of a remote-sensing and in situ data-assimilating version of CMAQ for operational PM_{2.5} forecasting. Part 1: MODIS aerosol optical depth (AOD) data-assimilation design and testing. *J Air Waste Manag Assoc* 65(12):1395–1412
- Li X, Peng L, Yao X et al (2017) Long short-term memory neural network for air pollutant concentration predictions: method development and evaluation. *Environ Pollut* 231(Pt 1):997–1004
- Song L, Pang S, Longley I et al (2014) Spatio-temporal PM_{2.5} prediction by spatial data aided incremental support vector regression. In: International joint conference on neural networks. IEEE, pp 623–630
- Tan KC, Lim HS, Jafri MZM (2016) Prediction of column ozone concentrations using multiple regression analysis and principal component analysis techniques: a case study in peninsular Malaysia[J]. *Atmos Pollut Res* 7(3):533–546
- Ghazali NA, Ramli NA, Yahaya AS et al (2010) Transformation of nitrogen dioxide into ozone and prediction of ozone concentrations using multiple linear regression techniques. *Environ Monit Assess* 165(1–4):475–489
- Donnelly A, Misstear B, Broderick B (2015) Real time air quality forecasting using integrated parametric and non-parametric regression techniques. *Atmos Environ* 103(103):53–65
- Mishra D, Goyal P (2015) Development of artificial intelligence based NO₂ forecasting models at Taj Mahal, Agra. *Atmos Pollut Res* 6(1):99–106
- He J, Ye Y, Na L et al (2013) Numerical model-based relationship between meteorological conditions and air quality and its implication for urban air quality management. *Int J Environ Pollut* 53(3–4):265–286
- Fang M, Zhu G, Zheng X et al (2011) Study on air fine particles pollution prediction of main traffic route using artificial neural network. In: International conference on computer distributed control and intelligent environmental monitoring. IEEE Computer Society, pp 1346–1349
- Zheng H, Shang X (2013) Study on prediction of atmospheric PM_{2.5} based on RBF neural network. In: Fourth international conference on digital manufacturing and automation. IEEE, pp 1287–1289
- Yang Y, Yan-Li FU (2015) The prediction of mass concentration of PM_{2.5} based on T-S fuzzy neural network. *J Shaanxi Univ Sci Technol* 33(6):162–166
- Tian-Cheng MA, Liu DM, Xue-Jie LI et al (2014) Improved particle swarm optimization based fuzzy neural network for PM_(2.5) concentration prediction. *Comput Eng Des* 35(9):3258–3262
- Hu Z, Li W, Qiao J (2016) Prediction of PM_{2.5} based on Elman neural network with chaos theory. In: Control conference. IEEE, pp 3573–3578
- Zhu H, Lu X (2016) The prediction of PM_{2.5} value based on ARMA and improved BP neural network model. In: International conference on intelligent NETWORKING and collaborative systems. IEEE, pp 515–517
- Zhou S, Li W, Qiao J (2017) Prediction of PM_{2.5} concentration based on recurrent fuzzy neural network. In: Control Conference. IEEE, pp 3920–3924
- Papadopoulos H, Haralambous H (2011) Reliable prediction intervals with regression neural networks. *Neural Netw* 24(8):842–851