Prediction of Electric Load for Users Based on BP Neural Network

Yang Li [1, a], Xiangying Xie[1, b], Liang Shan[1, c], Dayan Ma[1, d], Deping Zhang[2, e]

[1]The department of Photovoltaic Network National grid Ecommerce Ltd, Beijing, 100053, China

[2]School of Electric and Electronic Engineering, North China Electric Power University Beijing, 102206, China

[a]liyang1@sgec.sgcc.com.cn (Y. L.), [b]xiexiangying@sgec.sgcc.com.cn, [c]shanliang@sgec.sgcc.com.cn (L. S.), [d]madayan@sgec.sgcc.com.cn (D. M.), [e]1437240754@qq.com

Keywords: BP Neural Network, Prediction

Abstract: Power load forecasting is very important for power dispatching. Accurate load forecasting is of great significance for saving energy, reducing generating cost and improving social and economic benefits. In order to accurately predict the power load, based on BP neural network theory, combined with the advantages of Clementine in dealing with big data and preventing overfitting, a neural network prediction model for large data is constructed. The test results show that the model is effective and feasible, and achieves the expected results. It can achieve accurate prediction of user power consumption and greatly improve the accuracy of power grid dispatching.

1. Introduction

The accurate prediction of power load is of great significance for the electric power production and the safe operation of the power grid and the national economy[1]. Short term load forecasting is an important part of energy management system. The prediction error directly affects the analysis results of subsequent safety check of power grid, which is of great significance for dynamic state estimation, load scheduling and cost reduction [2-4]. Traditional prediction methods are based on linear regression, such as time series method, analysis method and pattern recognition method has defects of respectively[5], which leads to the prediction accuracy is not ideal; fuzzy control is a kind of controller based on fuzzy rules, the core problem is the membership functions of the fuzzy sets, fuzzy knowledge and fuzzy planning the quantitative setting, and does not adapt to the object transform, no self-organizing learning ability[6]; Pattern discrimination is difficult to deal with the problem of decentralization in large areas[7].Due to neural network[8-10]has the inherent properties of parallel processing, learning and memory, nonlinear mapping, adaptive ability and robustness, it is very suitable for power load forecasting. This article relies on the advantages and convenience of Clementine in dealing with large data, a large data load forecasting method based on BP neural network is proposed. Through the prevention of over fitting training the historical data, the neural network model is constructed to predict the daily load. The model has advantages in prediction accuracy and time overhead.

2. The basic funamental of BP neural network

2.1 The structure of BP neural network

BP neural network is a multi-layer network with error reverse propagation, which is composed of input layer nodes, hidden layer nodes and output layer nodes. There is no correlation between the same layer nodes, and the forward connection between the different layers of nodes. For the input signal, the input signal is propagated to the hidden layer node. After the function transformation, the input signal of the hidden layer node is propagated to the output layer node. The basic idea is that in the forward propagation, the input sample is passed from the input layer and the hidden layer is processed to the output layer by layer by layer. If the actual output of the output layer does not

  DOI: 10.25236/icscbd.2018.004

correspond to the expected output, the reverse propagation phase of the steering error is made. The back propagation of error is the output error in some form of back-propagation through the hidden layer to the input layer in some form, and the error is allocated to all the units of each layer, so as to get the error signal of each layer. This error signal is the basis for correcting the weights of each unit. The adjustment process of the weights of each layer of forward propagation and error back propagation is carried out round and round. The process of constant adjustment of weight is the learning and training process of the network. This process has been reduced to an acceptable level of error to the network output, or to a predetermined number of learning times. The network structure is shown in Figure 1.

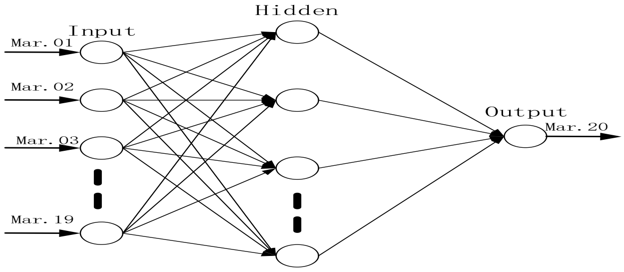


Figure 1. Neural network structure

The general model of artificial neural network consists of four basic elements, which are:

(1)The BP neural network is linked by different node coefficients. When connecting weights and weights are positive, it indicates that the current link is an exciting state. Conversely, if the link coefficient is negative, the link state is a state of suppression.

(2) The input signal and the linear signal are the combination of the signals for each input signal.

(3) The function of the nonlinear activation function: making the neuron output signal within a certain range.

(4) θ_k represents one of the neuron thresholds.

(5)The Excitation function $f(\bullet)$ using the frequently used function *sigmoid* .

The following are the functions of using mathematical expressions to represent the above basic elements:

$$u_k = \sum_{j=1}^{p} w_{kj} x_j , \tag{1}$$

$$v_k = net_k = u_k - \theta_k , \tag{2}$$

$$y_k = f(v_k), \tag{3}$$

$$\mathrm{E_m} = \frac{\sum_{\mathrm{t}} (\mathrm{t}_{mi} - 0_{mi})^2}{2} , \tag{4}$$

$$f(t) = \frac{1}{1+\exp(-\lambda t)}, \tag{5}$$

Where $x_{\mathrm{i}}(1,2\cdots\mathrm{k})$ is input signal for the system, $w_{ki(1,2\cdots\mathrm{k})}$ is the weight coefficient of the K neuron, u_k is the results of the weighted sum of each input signal, θ_k is the threshold of the kth neuron, $f(\bullet)$ is nonlinear activation function, y_k is the kth neuron output signal, E_m is the error function for the calculation of the M sample, t_{mi} is the expected output, 0_{mi} is the network computing output, λ is the slope parameter of activation function, different parameters, and the slope is different.

2.2 The determination of the number of network layers

BP neural network is back propagating, mainly composed of three parts: input layer, middle layer and output layer. The number of nodes in the input and output layers is relatively easy to determine, but the determination of the number of nodes in the hidden layer is a very important and complex problem. At present, there is no standard method to determine the number of nodes in the hidden layer. If the selection is too small, the performance of the neural network is poor or not trained, cannot produce enough to satisfy the right combination of connected learning neural network for sample selection; if too much, although you can make the system error of the neural network is smaller, but at the same time, the network training time will be increased, and it is easy to fall into the local small points and can not reach the most available solution.

There are several reference formulas for how to determine the number of nodes in the hidden layer:

(1) $n_1 = \sqrt{n+m} + a$, among them, n_1 is the number of hidden layer units, n is the number of input layer units, m is the number of output layer units, a is a constant between [1,10].

(2) $n_1 = \log_2 n$, among them, n_1 is the number of hidden layer units, is the number of input layer units.

Because the climate and season change little on the same date, the main influencing factor is time factor, so in this problem, we only consider the relation between time factor and load size. So we choose from March 1, 2015 to March 19th, the daily load of 1:00-24:00 ten lines as the input of neural network prediction model, Mar.01, Mar.02, Mar.03,... Mar.19. Using the power load of ten lines in March 20th as the output of the neural network prediction model, Mar.20. The neural network model designed in this paper has 19 input nodes and 1 output nodes. After many experiments, the network performance is best when the nodes of the hidden layer are 10.

This article starts from the reality, according to the user power load at every hour of 1:00-24:00 in the first 19 days of March, the user load value of each time of twentieth 1:00-24:00 is predicted. The Power load forecasting index system as shown in Figure 2.

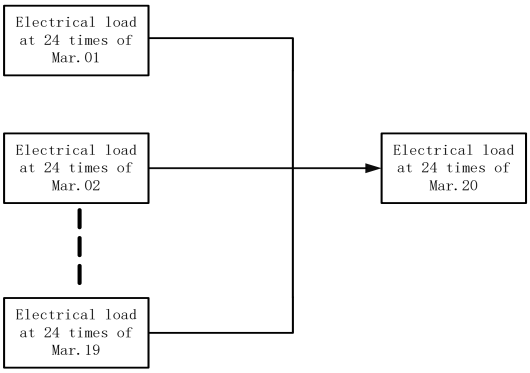


Figure 2. Power load forecasting index syste

2.3 Gradient descent algorithm

Firstly, suppose the network have M layers, and the input of the algorithm is a set of samples for the correct behavior of the network: $\{p_1, t_1\}, \{p_2, t_2\}, ..., \{p_q, t_q\}$ .Here p_q is the input of the network, t_q is the corresponding target output. For each input sample, the network output is compared with the target.The algorithm will adjust the network parameters to minimize the mean variance:

$$F(x) = E[e^T e] = E[(t-a)^T (t-a)], \tag{6}$$

Approximating the performance index by the next Formula:

$$\hat{F}(x) = e^2(k)e(k) = (t(k) - a(k))^2 (t(k) - a(k)), \tag{7}$$

In the above formula, the expected value of the mean square error is replaced by the mean square variance of the k times iteration. The first step of the BP algorithm is to propagate through the network input:

$$a^2 = p, \tag{8}$$

$$a^{m+1} = f^{m+1}(W^{m+1}a^m + b^{m+1}), m = 0,1,...,M-1 \quad , \tag{9}$$

$$a = a^M, \tag{10}$$

Then antipropagation based on sensitivity:

$$s^m = F^m(n^m)(W^{m+1})^T s^{m+1}, m = M-1,...,2,1, \tag{11}$$

Finally, the approximate final descent method is used to update the weights and bias values

$$W^m(k+1) = W^m(k) - \alpha s^m (a^{m-1})^T, \tag{12}$$

$$b^m(k+1) = b^m(k) - \alpha s^m, \tag{13}$$

The above steps are iterated one by one until the difference between the corresponding and objective $F(x)$ functions in BP network reaches an acceptable level, then the trained network is applied to load forecasting.

2.4 The training and implementation of the model

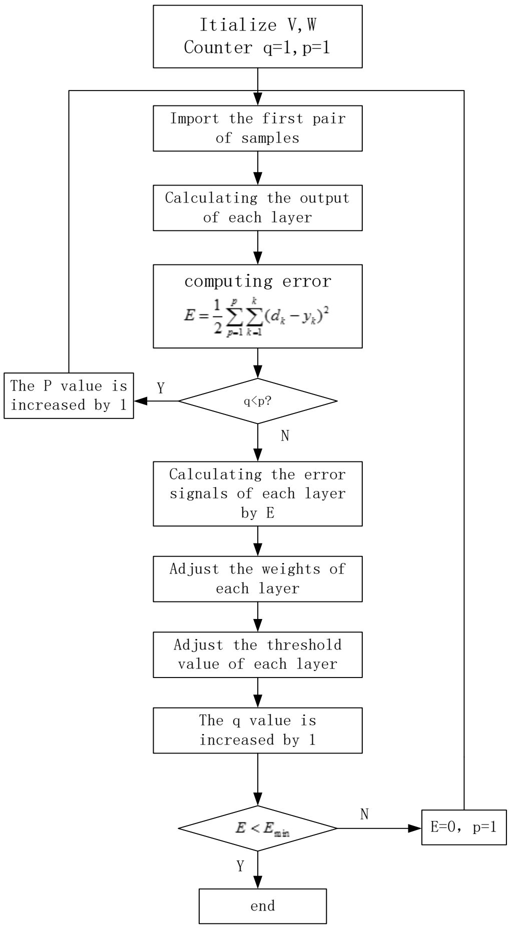


Figure 3. The training model neural network

First, initialize the network weight value and weight coefficient,import the first pair of samples and calculate the output of each layer.second,Using the formula $E=\frac{1}{2}\sum_{p=1}^{p}\sum_{k=1}^{k}(d_k-y_k)^2$ to calculate the error,if $q<p$ then the p value is increased by 1 and return to the original location,else adjust the weights and thresholds value of each layer of network by the calculated error,and the q value is increased by 1. Finally, judge the size of the error value, if the error value is less than the minimum error value, the training ends, or output $E=0, p=1$ to the original position and repeat training. The training model is shown in Figure .3.

3. Results

3.1 The establishment of simulation model

The large data prediction model for the user's electricity consumption is implemented in the Clementine software. Firstly, a neural network prediction model is set up as shown in Figure .4.

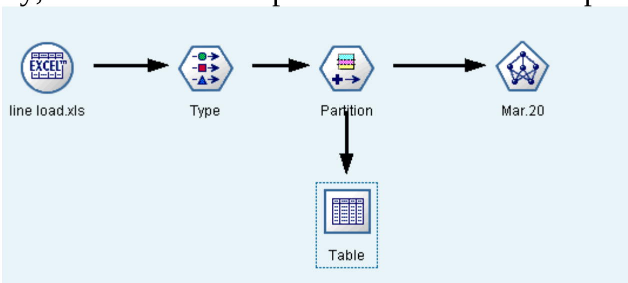
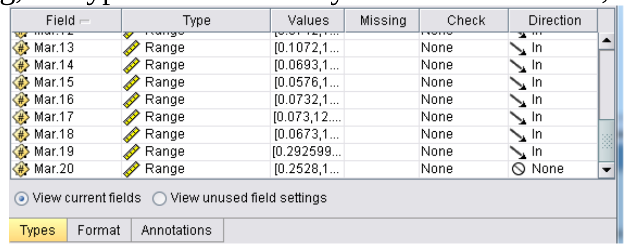


Figure 4. The prediction model of neural network

The first 19 days of data are involved in the modeling of the neural network model, so the type is selected as input. Since the twentieth day data only participates in the final result and does not participate in the modeling, the type of twentieth days is selected as none, as shown in Figure. 5.



Field	Type	Values	Missing	Check	Direction
Mar.13	Range	[0.1072,1...		None	In
Mar.14	Range	[0.0693,1...		None	In
Mar.15	Range	[0.0576,1...		None	In
Mar.16	Range	[0.0732,1...		None	In
Mar.17	Range	[0.073,12....		None	In
Mar.18	Range	[0.0673,1...		None	In
Mar.19	Range	[0.292599...		None	In
Mar.20	Range	[0.2528,1...		None	None

Figure 5. The input and output setting of the prediction model

After training, the model is generated, and data is introduced into the neural network prediction model for large data prediction. The prediction structure is shown in Figure.6.

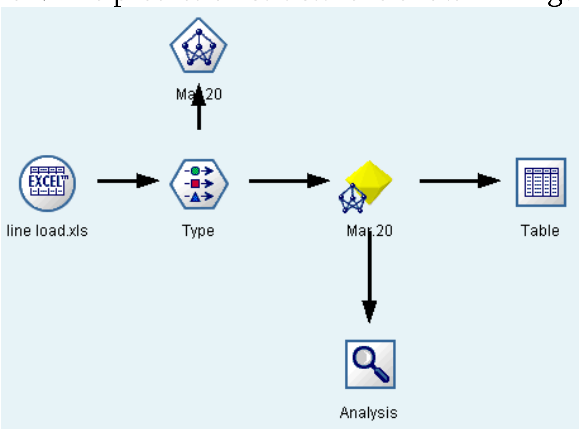


Figure 6. The data prediction structure

From March 1st to March 19th we will load each time the value of each line in the form of Excel into the neural network model, and the classification of the training data, learning data in 70%, 15% test data, 15% verification data division, in order to improve the prediction accuracy of neural network. Data classification settings are shown in Figure. 7.

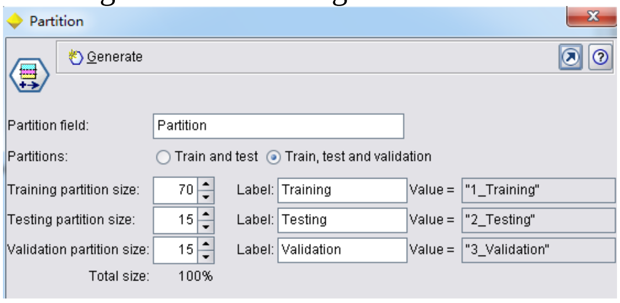


Figure 7. The data classification settings

The classified data is shown in Figure .8.

Mar.18	Mar.19	Mar.20	Partition
1.908	1.823	1.990	2_Testing
1.620	1.798	1.948	1_Training
2.032	2.322	2.301	1_Training
3.401	3.669	3.661	1_Training
3.794	3.774	4.461	2_Testing
3.436	3.325	3.573	1_Training
3.457	3.394	4.191	1_Training
3.978	3.861	3.896	1_Training
4.257	4.269	4.317	3_Validation
3.961	3.938	3.852	2_Testing
3.154	3.317	3.353	1_Training
2.731	3.279	3.061	2_Testing
2.737	3.356	2.794	1_Training
3.575	3.805	3.316	1_Training
4.495	4.301	3.978	1_Training
5.313	6.240	5.083	1_Training
5.765	6.340	5.841	1_Training
5.953	5.595	5.880	3_Validation
5.585	5.866	6.551	1_Training
3.891	4.270	4.713	1_Training

Figure 8. The classified data

The biggest advantage of Clementine software is its massive data processing capability and its convenience. In order to prevent the overfitting of prediction results, we set the parameters of neural network models to prevent training over fitting, as shown in the Figure.9.

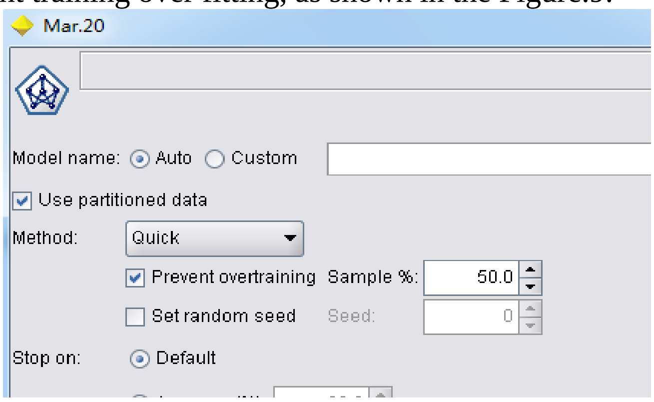


Figure 9. Prevent over fitting setting

The predicted results obtained after training are shown in Figure .10.

Mar.19	Mar.20	$N-Mar.20
4.772	4.377	4.647
2.976	3.952	3.902
3.147	4.008	3.963
2.819	3.718	3.769
4.238	4.652	4.118
6.569	5.553	6.364
9.653	9.990	9.611
9.596	10.712	10.463
9.430	10.500	10.436
7.892	9.282	9.899
9.743	10.496	10.989
11.534	13.305	11.779
8.876	9.648	10.052
8.077	8.277	8.212
8.500	7.960	8.434
7.690	7.775	8.167
9.354	8.524	10.104
11.294	11.205	11.163
11.788	12.179	11.921
12.314	13.391	12.016
11.550	12.353	11.964
9.317	12.463	11.455
7.591	10.915	9.436
5.155	6.794	6.248
0.585	0.308	0.684
0.293	0.253	0.673
0.304	0.297	0.676
0.322	0.342	0.679
0.314	0.357	0.675
0.417	0.388	0.683
0.679	0.698	0.715
0.855	0.666	0.768
0.857	0.971	0.770
1.002	0.920	0.789
1.092	0.834	0.785
1.184	1.043	0.802

Figure 10. The prediction results

3.2 Analysis of experimental results

By selecting the load prediction results of 403 and 411 lines, as shown in Figure.11. and Figure.12. We can see that the actual values of the lines basically match the predicted values, but there are also some errors, especially in the peak period of electricity consumption.

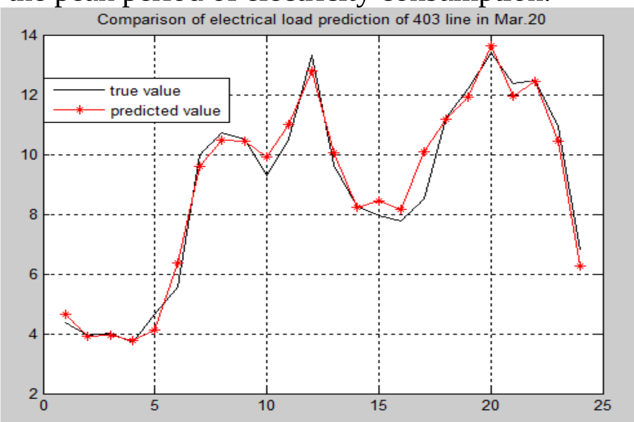


Figure 11. Comparison of power load forecasting of 403 line

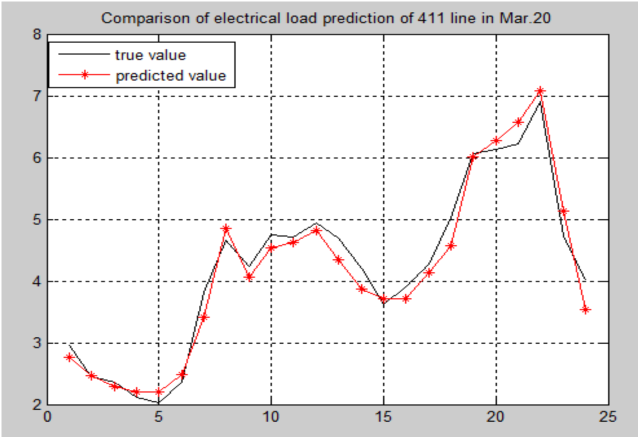


Figure 12. Comparison of power load forecasting of 411 line

The maximum error of the trained neural network prediction model is 0.267, and the minimum prediction error is -0.18, the prediction accuracy is 97.358%.

From the comparison between prediction data and actual data, the BP neural network has better prediction performance and relatively small error, which can meet the demand completely, and has fast prediction speed and convenient operation.

4. Conclusion

The trend of mass data in power system provides a basis for load characteristic analysis and prediction model establishment, but the classical load forecasting method can not afford such a huge time and computing resource consumption. The problem of over fitting in large sample set will affect the prediction accuracy. In this paper, a power load forecasting model is built by using the BP neural network model, making full use of the powerful data processing function of Clementine and preventing the over fitting function. The experimental results show that the BP neural network model has good predictability and robustness, and has a certain practical application value.

Acknowledgements

The authors gratefully acknowledge the financial support from State Grid Corporation of China-Deepening research and application of PV Cloud (SGZJ0000KXJS1800281).

References

[1] Cheng Qiyun,Sun Caixin,Zhang Xiaoxing,et al.Short-Term load forecasting model and method for power system based on complementation of neural network and fuzzy logic[J]. Transactions of China Electrotechnical Society, 2004, 19(10):53-58.

[2] Fangfang. Research on power load forecasting based on Improved BP neural network [D]. Harbin Institute of Technology, 2011.

[3] Amjady N.Short-term hourly load forecasting using time series modeling with peak load estimation capability [J]. IEEE Transactions on Power Systems, 2001,16(4):798-805.

[4] Ma Kunlong. Short term distributed load forecasting method based on big data [D] .Changsha:Hunan University, 2014.

[5] SHI Biao,LI Yu Xia,YU Xhua,YAN Wang.Short-term load forecasting based on modified particle swarm optimizer and fuzzy neural network model[J].Systems Engineering-Theory and Practice,2010,30(1):158-160.

[6] Senthil Arumugam M,Rao M V C,Chandramohan A.A new and improved version of particle swarm optimization algorithm with global-local best parameters[J].Knowledge and Information Systems,2008,16(3):15-26.

[7] Wu Jiekang,Chen Minghua,Chen Guotong.Fuzzy neural network model based on particle swarm optimization for short-term load forecasting[J].Proceedings of the Electric Power System and Automation,2007,19(1):63-66.

[8] Li Long, Wei Jing, Li Canbing, et al. Prediction of load model based on artificial neural network [J]. Transactions of China Electrotechnical Society, 2015, 30(8): 225-230.

[9] Liu Nian,Zhang Qingxin,Liu Haitao.Online short-term load forecasting based on ELM with Kernel algorithm in micro-grid environment[J].Transactions of China Electrotechnical Society, 2015, 30(8):218-224.

[10] Chen Ya, Li Ping.Research on simulation of short-term power load forecasting based on neural network [J]. Electrical Engineering, 2017, 18(1):26-29.