Classification of the students' scores based on some artificial neural networks

Hu Hongping Bai Yanping

***Abstract*—**In this paper, the data of students' scores are analyzed by using the nonlinear BP neural network algorithm with a hidden layer, the probabilistic neural network algorithm, the perceptron algorithm and self-organizing compete neural network algorithm.

We take 3000 students' scores on only course to be analyzed and classified. Among these scores, 121 students' scores are trained and 2879 students' scores are tested by the probabilistic neural network algorithm, the nonlinear BP neural network algorithm and the perceptron neural network algorithm. By comparing these three kinds of neural network algorithms, we can get the following results: the train errors of these three neural network algorithms are all zero, but the test errors of these three neural network algorithms are different.and the test error of the probabilistic neural network algorithm is less than those of the nonlinear BP neural network algorithm and the perceptron algorithm; the train time of the BP neural network algorithm is longer than those of the probabilistic neural network algorithm and the perceptron algorithm;the test time of the probabilistic neural network algorithm is longer that those of the nonlinear BP neural network algorithm and the perceptron algorithm. The correct rate of the probabilistic neural network algorithm heads to 99.06% when

$$\text{net.spread} \in [0.00121998\ 89, 0.1186944\ 765] \ .$$

The correct rate of the BP neural network algorithm changes from 98.51% to 99.06%. But the correct rate of the perceptron neural network algorithm is too low and changes from 20% to 30%. Therefore by considering the correct rate and the whole time of classification, we obtain that the probabilistic neural network algorithm is more suitable for solving the classification of the students' scores on only one course .

And we take 1680 students' scores on five course to be analyzed and classified. Among these scored, 179 students' scores are trained and 1501 students' scores are tested by the nonlinear BP neural network algorithm with the momentum factor, the nonlinear BP neural network algorithm with the gradient descent method, the probabilistic neural network algorithm and the self-organizing complete network algorithm. By comparing these kinds of neural network algorithms, we can get the following results: the train errors of the probabilistic neural network algorithm are all zero,those of the BP neural network algorithm with the momentum factor are all less than 0.0089, those of the BP neural network algorithm with the gradient descent method are all less than 0.0536, and those of the self-organizing compete neural network algorithm are all less than 0.4 and are all more than 0.2436; the test errors

of the probabilistic neural network algorithm all equal to 0.0799, but those of the BP neural network algorithm with the momentum factor are all less than 0.0738, those of the BP neural network algorithm with the gradient descent method are all less than 0.1332, and those of the self-organizing compete neural network algorithm are all less than 0.3888 and are all more than 0.1871;

the train times of the the probabilistic neural network algorithm are all less than 0.0469,those of the BP neural network algorithm with the momentum factor are all less than 33.0156 and are all more than 29.6875, those of the BP neural network algorithm with the gradient descent method are almost 24.3594 and are mostly less than 7.1875, and those of the self-organizing compete neural network algorithm are all less than 332.9219 and are all more than 310.0156; the test times of the probabilistic neural network algorithm are the least and are all less than 0.1719 and more than 1406, but those of the other neural network algorithms are all less than 0.0938; the train correct rates of the probabilistic neural network algorithm are all 100%when

$$\text{net.spread} \in [0.065, 0.0651],$$

those of the BP neural network algorithm with the momentum factor are all less than 99.44% and are all more than 97.77%, those of the BP neural network algorithm with the gradient descent method are all less than 93.30% and are all more than 86.59%, and the those of the self-organizing compete neural network algorithm are all less than 40%;the test correct rates of the probabilistic neural network algorithm are all 80.01%, those of the BP neural network algorithm with the momentum factor are all less than 87.67% and are all more than 81.55%, those of the BP neural network algorithm with the gradient descent method are all less than 82.41% and are all more than 66.69%, and those of the self-organizing compete neural network algorithm are all less than 53.23%.Therefore by considering the correct rates and the whole times of classification, we obtain that the probabilistic neural network algorithm and the BP neural network algorithm are more suitable for solving the classification of the students' scores on five courses .

***Keywords*—**The Students' Scores; BP Neural Network; Probabilistic Neural Network; Perceptron Neural network; Self-organizing Compete Neural Network; Train error; Test error; Train time; Test time;Train Correct Rate; Test Correct Rate

I. Introduction

The study life of a student is mainly based on the students' scores. In the process of his or her learning, the scores are evaluated from all aspects, such as scholarship, graduation

---

Manuscript received August 19, 2010. This research was supported by National Natural Science Foundation of Shanxi 2009011018-3, National Natural Science Foundation of China 60876077 and School Foundation of North University of China.

Hu Hongping is with College of Science, North University of China, Taiyuan, Shanxi 030051, P.R.China(phone:8615935623273; fax 863513942729; e-mail: huhongping@nuc.edu.cn or hhp92@163.com).

Bai Yanping is with College of Science, North University of China, Taiyuan,Shanxi 030051,P.R.China(phone: 863513942729; fax: 863513942729; e-mail: baiyp@nuc.edu.cn).

recommendation and so on. There are many methods of analyzing the students' scores: the multivariate statistical analysis, excel function, principal component analysis, statistical software SPSS, designing and implementation of data warehouse [1]-[4]. As for the application of artificial intelligence techniques to the design of automatic classification systems, stellar classification has applied artificial neural networks to obtaining classifications with diverse resolution grades [6]-[7]. And the classifications are applied in image, information systems,bearing fault diagnosis,model β-glucan suspensions, knowledge-based systems and so on[11]-[15]. But there are seldom methods of analyzing the students' score based on the artificial neural network.

There are five classifications according to the students' scores on only one course or the avenge scores on five coures, see Table I.

TABLE I. CLASSIFICATIONS OF STUDENT'S SCORES

scope of scores	90-100	80-89	70-79	60-69	0-59
degree	excellent	good	middling	passing	unpassing
Expression by digital	(1,0,0,0,0)	(0,1,0,0,0)	(0,0,1,0,0)	(0,0,0,1,0)	(0,0,0,0,1)

In this paper, the classification algorithms applied to the students' scores are described on the base of the nonlinear BP network with a hidden layer algorithm, the probabilistic Neural Network algorithm and the perceptron algorithm.

We firstly take 3000 students' scores on only one course in a college to be processed, analyzed and classified. Among these scores, the 121 students' scores are trained and 2879 students' scores are tested by the nonlinear BP network with a hidden layer algorithm, the probabilistic neural network algorithm and the perceptron algorithm, respectively.

We take 1680 students' scores on five course in a college to be processed, analyzed and classified. Among these scores,the 179 students' scores are trained and 1501 students' scores are tested by the nonlinear BP network with the momentum factor or with the gradient descent method, the probabilistic neural network algorithm and the self-organizing complete neural network algorithm, respectively.

Applied in many experiments, the algorithms are taken to classify and to evaluate the students' scores effectively.

II. THE BP NEURAL NETWORK

The classical method for training multiple-layer feed-forward network is the back-propagation (BP) algorithm, which is based on the gradient descent optimization technique. The feed-forward network with BP algorithm is called BP neural networks with its solid theory and wide application.

Its activation function is the logistic function (or the hyperbolic tangent function, which is a simple rescaling of the logistic), shown as formula (1), where α is the slope parameter of the binary sigmoid function.

$$y = f(n) = \frac{1}{1+e^{-\alpha n}} \tag{1}$$

In this paper, we discuss the BP neural network with a ridden layer shown in Figure 1, where P be a input vector. Denote by W_1 the weight connecting the input layer and the hidden layer. Denote by W_2 the connection weight between the hidden layer and the output layer. Functions f_1 and f_2 are the sigmoid activation functions of the hidden layer and the output layer respectively. And

$$Y_1 = f_1(W_1 * P + b_1), \tag{2}$$

$$Y_2 = f_2(W_2 * P + b_2). \tag{3}$$

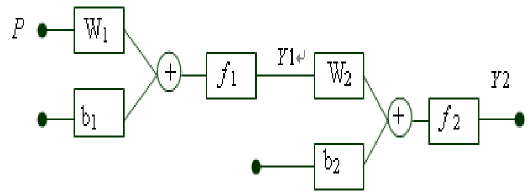

Figure1. The BP neural network

Let the desired output corresponding to the input example P be ζ. Then the square error function for this step of training is

$$E = \frac{1}{2}\sum_{\mu=1}^{M}(\zeta^{\mu} - Y_2^{\mu})^2 \tag{4}$$

Where, M is the number of training input. Let W denote the vector containing all the weights. The purpose of BP algorithm is to choose W so as to minimize the error function by the gradient descent method. So the general expression of the iteration formula is

$$W(t+1) = W(t) + \Delta W(t), \tag{5}$$

where

$$\Delta W = \eta\left(-\frac{\partial E}{\partial W}\right)|_{W=W(t)} \tag{6}$$

is the weight increment at time t and the positive constant $\eta(0<\eta<1)$ is the training rate.

A momentum term is added to Formula (5) to accelerate the convergence speed, resulting in below.

$$\Delta W = \eta\left(-\frac{\partial E}{\partial W}\right)|_{W=W(t)} + \mu[W(t) - W(t-1)], \tag{7}$$

where the positive constant μ $(0<\mu<1)$is a momentum factor.

III. PROBABILISTIC NEURAL NETWORK

We discuss the radial basis function neural network shown in Figure 2, where the distance $\|dist\|$ between weight vectors and threshold vectors is an independent variable on the active function radbas of the radial basis function neural network, where $\|dist\|$ is obtained by the multiply of the input vector and the row vectors of the weight matrix.

The active function of the radial basis function neural network is the following:

$$radbas(n) = e^{-n^2}. \tag{8}$$

When input vector is 0, the maximum of the active function is 1. With the distance between weight vector and input vector decreasing, the output of the network increases.

Therefore, the output is 1 when the input vector is the same as the weight vector.

Probabilistic Neural Network (simply PNN) is built by combining the radial basis neural with the competitive neural. The structure of the PNN is showed Figure 3.

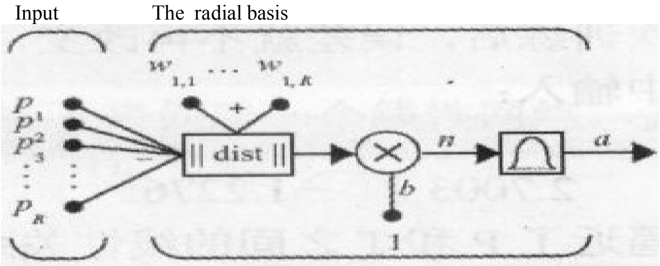

Figure 2. The structure of the radial basis function neural network(see [5])

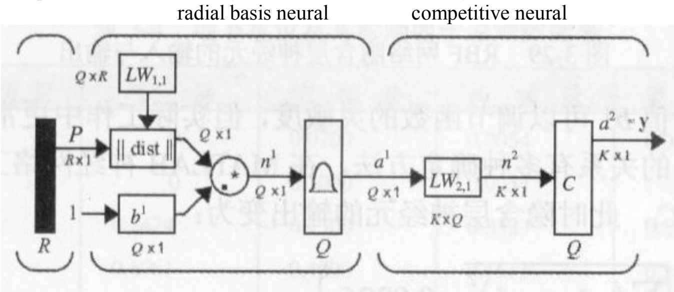

Figure 3. The structure of Probabilistic Neural Network(see [5])

IV. PERCEPTRON NEURAL HETWORK

The perceptron network means a simple network with only an input and output layer, or a neural network which has no hidden layer. A single-layer network is with R inputs and S
neurons. Its architecture is showed in figure 4[8].

Here P is an input vector of length R. W is a matrix (S*R) as show below. The net input to transfer function F is n, the sum of the bias b and product W*P. The sum is passed to the transfer function to get the neuron's output. If we have more than one neuron, the network putout will be a vector. The neuron layer includes the weight matrix, the multiplication operations, the bias vector b, the sum, and the transfer function boxes.

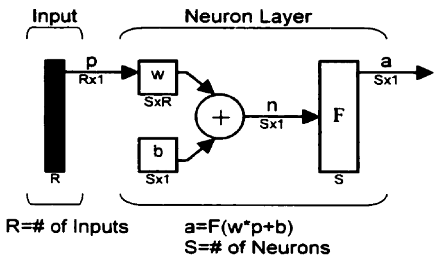

Figure 4 The perceptron Neural Network

The activation function is nolinear function. This is a binary (or bipolar) function that hard-limits the input to the function to either a 0 or a 1 for the binary. It is written as:

$$a = f(n) = \begin{cases} 0 & if \ n < 0 \\ 1 & if \ n \geq 0 \end{cases} \quad (9)$$

V. SELF-ORGANIZING COMPETE NETURAL NETWORK

The network structure of the self-organizing neural network is shown in figure 5. It is trained by means of a kind of no instructor and has an ability in a self-organizing. Generally, It consists of the input layer and the compete layer, not having the hidden layer. There are some bidirectional links between the nodes in the input layer and the nodes in the compete. In the same time, there exists some sidewise links between two arbitrary nodes in the compete.

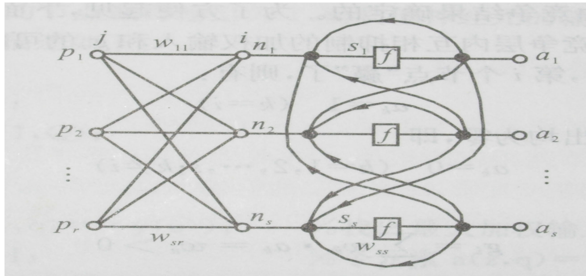

Figure 5 The self-organizing Neural Network(see[9])

From the Figure 5, there are two kinds of weights in the self-organizing neural network. One kind of weights are the weights $w_{ij}(i=1,2,\cdots,s; j=1,2,\cdots,r)$ from the input node j to the i th neural node, that are adjusted by the train. The other kind of weights are the weights

$$w_{ik}(k=1,2,\cdots,s),$$

that are fixed not to be varied and that satisfy some relation of distribution.In addition, the weights are the kind of the symmetric weights,that is, $w_{ik} = w_{ki}$. In the same time, the weights between the same neural nodes have the strengthened efforts, that is, $w_{11} = w_{22} = \cdots = w_{ss} > 0$. And the weights between the different neural nodes have the rejected efforts, that is, $w_{ij} < 0$ for $k \neq j$.

The formula of the adjusted weight is

$$\Delta w_{ij} = \alpha(x_i - w_{ij}) \quad (10)$$

where α is a learning rate,satisfies $0 < \alpha < 1$.In general, the scope of α is from 0.01 to 0.3 , x_i is the input by reduction to unity.

VI. EXPERIMENTAL RESULTS

A. Scores on only one course

In the BP neural network ,the probabilistic neural network and the perceptron network□121 students' scores as input vector are trained and 2879 students' scores are tested.

In the experiments, the BP neural network consists of 5 output nodes and 20 nodes in hidden layer taken by comparing 18 nodes, 19 nodes and 20 nodes in hidden layer. The parameter selection is as following:

net.trainParam.epochs = 2000;
net.trainParam.min_grad=1e-20;
net.trainParam.lr = 0.01;
net.trainParam.lr_inc=1.05;
net.trainParam.lr_dec=0.7;
net.trainParam.max_fail=5;
net.trainParam.min_grad=1e-20.

The parameter selection of the probabilistic neural network is the density of distribution as net.spread taking an arbitrary value from 0.0012199889 to 0.1186944765 by comparing different net.

The parameter selection of the perceptron neural network is the net.trainParam.epochs ≥ 14.

In the experiments, we take some experiment results for example, shown in Table II ,Table III and Table IV, respectively.

TABLE II. TRAIN TIMES,TEST TIMES,TRAIN ERRORS AND TEST ERRORS OF THE BP NEURAL NETWORK

Train times	TEST TIMES	TRAIN ERROR	TEST ERROR	TRAIN CORRECT RATE	TEST CORRECT RATE
14.2656	0.1094	0	0.1421	1	0.9851
14.4062	0.125	0	0.1421	1	0.9851
14.5938	0.125	0	0.0893	1	0.9906
14.6719	0.1094	0	0.0893	1	0.9906
14.375	0.1094	0	0.0893	1	0.9906
14.3438	0.1406	0	0.0893	1	0.9906
14.2344	0.1406	0	0.0893	1	0.9906
14.3438	0.125	0	0.0893	1	0.9906
14.3906	0.125	0	0.1421	1	0.9851
14.7031	0.1406	0	0.0893	1	0.9906
15.2031	0.125	0	0.0893	1	0.9906
15.125	0.125	0	0.1421	1	0.9851
15.5	0.0938	0	0.0893	1	0.9906
16.625	0.1406	0	0.0893	1	0.9906
16.9375	0.125	0	0.1421	1	0.9851
17.4062	0.125	0	0.0893	1	0.9906
16.9062	0.1406	0	0.1421	1	0.9851
17.0938	0.1562	0	0.1157	1	0.9851
16.9688	0.125	0	0.0893	1	0.9906
17.1094	0.125	0	0.0893	1	0.9906
17	0.125	0	0.1421	1	0.9851
16.8281	0.1406	0	0.0893	1	0.9906
16.2188	0.125	0	0.1421	1	0.9851
14.9042	0.1094	0	0.1421	1	0.9851
16.3906	0.1094	0	0.0893	1	0.9906
14.8906	0.1406	0	0.1421	1	0.9851
14.9062	0.125	0	0.0893	1	0.9906
14.6406	0.1406	0	0.0893	1	0.9906
14.8438	0.1406	0	0.0893	1	0.9906
14.8125	0.1406	0	0.0983	1	0.9906
14.7812	0.125	0	0.1421	1	0.9851
14.6562	0.125	0	0.1421	1	0.9851
14.8281	0.1405	0	0.0893	1	0.9906
14.625	0.125	0	0.0893	1	0.9906
14.6562	0.125	0	0.0893	1	0.9906
14.8594	0.1406	0	0.1421	1	0.9851
17.4688	0.1562	0	0.0893	1	0.9906
15.5	0.1406	0	0.0893	1	0.9906
14.7188	0.1094	0	0.1421	1	0.9851
14.875	0.125	0	0.1421	1	0.9851
14.8906	0.125	0	0.1421	1	0.9851
14.9071	0.1406	0	0.0893	1	0.9906
14.9375	0.125	0	0.1421	1	0.9851
14.7969	0.1094	0	0.0893	1	0.9906
14.9219	0.1094	0	0.0893	1	0.9906

Comparing with these three neural network algorithms, we can find that:

1) the train errors of these three neural network algorithms are all zero, but the test errors of the probabilistic neural network algorithm are all 0.0038, those of the nonlinear BP neural network algorithm are all less than 0.1421 and are all more than 0.0893, and those of the perceptron neural network are all more than 6.5124 ;

2) the train times of the BP neural network algorithm are all more than 14.2344 and are all less than 17.4688, those of the probabilistic neural network algorithm are all less than 0.0625 and are all more than 0.0312, and those of the perceptron neural network algorithm are mostly 0 and are almost 0.0312;

3) the test times of the BP neural network algorithm are all more than 0.0938 and are all less than 0.1562, those of the probabilistic neural network algorithm are all less than 0.1562

TABLE III. TRAIN TIMES,TEST TIMES,TRAIN ERRORS AND TEST ERRORS OF THE PROBABILISTIC NEURAL NETWORK

TRAIN TIMES	TEST TIMES	TRAIN ERROR	TEST ERROR	TRAIN CORRECT RATE	TEST CORRECT RATE
0.0625	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0469	0.1094	0	0.0038	1	0.9906
0.0312	0.0938	0	0.0038	1	0.9906
0.0312	0.0938	0	0.0038	1	0.9906
0.0312	0.0938	0	0.0038	1	0.9906
0.0469	0.1094	0	0.0038	1	0.9906
0.0469	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0469	0.1562	0	0.0038	1	0.9906
0.0312	0.125	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.125	0	0.0038	1	0.9906
0.0469	0.1094	0	0.0038	1	0.9906
0.0469	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0469	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.125	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0469	0.1094	0	0.0038	1	0.9906
0.0469	0.0938	0	0.0038	1	0.9906
0.0312	0.125	0	0.0038	1	0.9906
0.0469	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.0938	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.1094	0	0.0038	1	0.9906
0.0312	0.0938	0	0.0038	1	0.9906
0.0469	0.1094	0	0.0038	1	0.9906
0.0469	0.1094	0	0.0038	1	0.9906
0.0312	0.125	0	0.0038	1	0.9906
0.0469	0.1094	0	0.0038	1	0.9906
0.0469	0.1094	0	0.0038	1	0.9906
0.0469	0.0938	0	0.0038	1	0.9906

and are all more than 0.0938, and those of the perceptron neural network algorithm are mostly 0.0312 and are almost 0;
4) The correct rates of the probabilistic neural network algorithm heads to 99.06% when

$$\text{net.spread} \in [0.00121998\,89, 0.1186944\,765],$$

those of the BP neural network algorithm changes from 98.51% to 99.06% according to the above parameters, and the correct rate of the perceptron neural network algorithm is too low and changes from 20% to 30% according to the above parameters.

Therefore by considering the correct rate and the whole time of classification, we obtain that the probabilistic neural network algorithm is more suitable for solving the classification of the students' scores.

TABLE IV. TRAIN TIMES,TEST TIMES,TRAIN ERRORS AND TEST ERRORS OF THE PERCEPTRON NEURAL NETWORK

TRAIN TIMES	TEST TIMES	TRAIN ERROR	TEST ERROR	TRAIN CORRECT RATE	TEST CORRECT RATE
0	0.0312	0	7.3983	1	0.2226
0.0156	0	0	6.9537	1	0.216
0	0.0312	0	6.681	1	0.2223
0	0.0156	0	7.3983	1	0.0006
0	0.0156	0	6.8843	1	0.2154
0	0	0	7.3983	1	0.2226
0	0.0312	0	6.5223	1	0.2226
0.0312	0	0	7.3983	1	0.2226
0	0.0312	0	6.7504	1	0.2226
0	0.0312	0	7.2777	1	0.2226
0.0312	0.0312	0	7.3983	1	0.2226
0	0.0312	0	7.1355	1	0.2226
0	0	0	6.843	1	0.2195
0	0.0312	0	7.3983	1	0.2226
0.0312	0.0156	0	7.3983	1	0.2226
0	0	0	7.3983	1	0.2226
0	0.0156	0	7.3983	1	0.2226
0	0.0312	0	7.3983	1	0.2226
0	0.0312	0	7.3983	1	0.2226
0	0.0156	0	7.3983	1	0.2226
0	0.0312	0	7.3736	1	0.2213
0	0	0	7.3983	1	0.2226
0	0.0312	0	7.1818	1	0.214
0	0.0312	0	7.0033	1	0.2226
0	0	0	7.3983	1	0.2226
0	0	0	6.8231	1	0.2119
0.0156	0	0	6.8446	1	0.2108
0	0	0	6.6099	1	0.2122
0.0156	0	0	6.8397	1	0.2105
0.0156	0	0	6.7653	1	0.2133
0	0	0	7.2694	1	0.2226
0	0.0156	0	7.276	1	0.222
0	0	0	6.681	1	0.2119
0	0	0	6.7504	1	0.2143
0	0	0	7.2661	1	0.222
0	0	0	6.7289	1	0.2122
0	0	0	6.7107	1	0.2226
0	0	0	7.3983	1	0.2126
0	0	0	7.2893	1	0.2213
0	0	0	6.7207	1	0.2115
0.0156	0	0	6.4711	1	0.2122
0.0156	0	0	6.8364	1	0.2105
0	0	0	6.5124	1	0.2105
0	0	0	6.8397	1	0.2119

B. Scores on five courses

In the BP neural network , the probabilistic neural network and the self-organizing complete network□179 students' scores as input vector are trained and 1501 students' scores are tested.

In the experiments, the BP neural network consists of 5 output nodes and 50 nodes in hidden layer taken by comparing some nodes among {10,11,...,100} in hidden layer. The parameter selection of the BP neural network with the momentum factor is as following:

```
net.trainParam.epochs =5000;
net.trainParam.show = 100;
net.trainParam.lr = 0.02;
net.trainParam.lr_inc=1.08;
net.trainParam.lr_dec=0.65;
net.trainParam.max_fail=5;
net.trainParam.min_grad=1e-20;
net.trainParam.goal=0.001;
net.trainParam.mc=0.98;
net.trainParam.max_perf_inc=1.04.
```

The parameter selection of the BP neural network with the gradient descent method is as following:

```
net.trainParam.epochs = 5000;
net.trainParam.show = 50;
net.trainParam.lr = 0.01;
net.trainParam.lr_inc=1.05;
net.trainParam.lr_dec=0.7;
net.trainParam.max_fail=5;
net.trainParam.min_grad=1e-20;
net.trainParam.goal=0.04.
```

In the experiments, we take some experiment results for example, shown in Table V and Table VI, respectively.

The parameter selection of the probabilistic neural network is the density of distribution as net.spread taking an arbitrary value from 0.065 to 0.0651 by comparing different net. In the experiments, we take some experiment results for example, shown in Table VII.

The parameter selection of the self-organizing compete neural network is as following:

```
net.iw{1,1}=randnr(S,R);
net.trainParam.epochs =1000;
net.trainParam.show = 50;
net.trainParam.goal=0.001.
```

In the experiments, we take some experiment results for example, shown in Table VIII.

Comparing with the neural network algorithms: BP neural network algorithm with the momentum factor, BP neural network algorithm with the gradient descent method, the probabilistic neural network algorithm and the self-organizing compete algorithm, we can find that:
1) The train errors of the probabilistic neural network algorithm are all zero,those of the BP neural network algorithm with the momentum factor are all less than 0.0089, those of the BP neural network algorithm with the gradient descent method are all less than 0.0536, and those of the self-organizing compete neural network algorithm are all less than 0.4 and are all more than 0.2436;
2) the test errors of the probabilistic neural network algorithm all equal to 0.0799, but those of the BP neural network algorithm with the momentum factor are all less than 0.0738,

those of the BP neural network algorithm with the gradient descent method are all less than 0.1332, and those of the self-organizing compete neural network algorithm are all less than 0.3888 and are all more than 0.1871;

3) the train times of the the probabilistic neural network algorithm are all less than 0.0469,those of the BP neural network algorithm with the momentum factor are all less than 33.0156 and are all more than 29.6875, those of the BP neural network algorithm with the gradient descent method are almost 24.3594 and are mostly less than 7.1875, and those of the self-organizing compete neural network algorithm are all less than 332.9219 and are all more than 310.0156;

4) the test times of the probabilistic neural network algorithm are the least and are all less than 0.1719 and more than 1406, but those of the other neural network algorithms are all less than 0.0938;

5) the train correct rates of the probabilistic neural network algorithm are all 100%,those of the BP neural network algorithm with the momentum factor are all less than 99.44% and are all more than 97.77%, those of the BP neural network algorithm with the gradient descent method are all less than 93.30% and are all more than 86.59%, and the those of the self-organizing compete neural network algorithm are all less than 40%;

6) the test correct rates of the probabilistic neural network algorithm are all 80.01%, those of the BP neural network algorithm with the momentum factor are all less than 87.67% and are all more than 81.55%, those of the BP neural network algorithm with the gradient descent method are all less than 82.41% and are all more than 66.69%, and those of the self-organizing compete neural network algorithm are all less than 53.23%.

Therefore by considering the correct rate and the whole time of classification, we obtain that the probabilistic neural network algorithm and the BP neural network algorithm with the momentum factor are more suitable for solving the classification of the students' more scores.

TABLE V. TRAIN TIMES,TEST TIMES,TRAIN ERRORS AND TEST ERRORS OF BP NEURAL NETWORK WITH THE MOMENTUM FACTOR

Train times	Test times	Train errors	Test errors	Train Correct rate	Test correct rate
32.8125	0.0781	0.0089	0.0709	0.9777	0.8228
31.0781	0.0625	0.0022	0.0661	0.9944	0.8348
31.9063	0.0625	0.0022	0.0498	0.9944	0.8754
32.0781	0.0625	0.0022	0.0573	0.9944	0.8568
33.0156	0.0938	0.0089	0.0624	0.9777	0.8441
31.8906	0.0625	0.0022	0.0498	0.9944	0.8754
32.3281	0.0938	0.0022	0.0568	0.9944	0.8581
30.7031	0.0625	0.0022	0.0570	0.9944	0.8574
32.5781	0.0469	0.0022	0.0560	0.9944	0.8601
31.8438	0.0781	0.0089	0.0496	0.9777	0.8761
31.0156	0.0781	0.0045	0.0586	0.9888	0.8534
31.4844	0.0625	0.0022	0.0565	0.9944	0.8588
31.5625	0.0938	0.0022	0.0600	0.9944	0.8501
32.2500	0.0625	0.0045	0.0576	0.9888	0.8561
31.1875	0.0625	0.0067	0.0552	0.9832	0.8621
30.1563	0.0625	0.0022	0.0578	0.9944	0.8554
30.8906	0.0781	0.0022	0.0738	0.9944	0.8155
31.4219	0.0781	0.0022	0.0493	0.9944	0.8767
31.7813	0.0469	0.0022	0.0626	0.9944	0.8434
31.3906	0.0625	0.0089	0.0522	0.9777	0.8694
31.6719	0.0625	0.0089	0.0525	0.9777	0.8688
31.3125	0.0781	0.0022	0.0496	0.9944	0.8761
31.6875	0.0781	0.0022	0.0602	0.9944	0.8494
31.2344	0.0625	0.0022	0.0621	0.9944	0.8448
30.5313	0.0781	0.0022	0.0578	0.9944	0.8554
31.2344	0.0781	0.0089	0.0709	0.9777	0.8228
30.8125	0.0625	0.0022	0.0661	0.9944	0.8348
30.1250	0.0938	0.0022	0.0498	0.9944	0.8754
30.1875	0.0781	0.0022	0.0573	0.9944	0.8568
30.2344	0.0938	0.0089	0.0624	0.9777	0.8441
31.0938	0.0625	0.0022	0.0498	0.9944	0.8754
29.8906	0.0625	0.0022	0.0568	0.9944	0.8581
30.6563	0.0938	0.0022	0.0570	0.9944	0.8574
30.3750	0.0625	0.0022	0.0560	0.9944	0.8601
30.0156	0.0625	0.0089	0.0496	0.9777	0.8761
30.4219	0.0625	0.0045	0.0586	0.9888	0.8534
29.8750	0.0625	0.0022	0.0565	0.9944	0.8588
29.8281	0.0625	0.0022	0.0600	0.9944	0.8501
30.3438	0.0625	0.0045	0.0576	0.9888	0.8561
29.6875	0.0938	0.0067	0.0552	0.9832	0.8621
29.9063	0.0625	0.0022	0.0578	0.9944	0.8554
29.9063	0.0625	0.0022	0.0738	0.9944	0.8155
29.8438	0.0625	0.0022	0.0493	0.9944	0.8767
30.2813	0.0938	0.0022	0.0626	0.9944	0.8434
30.1563	0.0938	0.0089	0.0522	0.9777	0.8694

TABLE VI. TRAIN TIMES,TEST TIMES,TRAIN ERRORS AND TEST ERRORS OF BP NEURAL NETWORK WITH THE GRADIENT DESCENT METHOD

Train times	Test times	Train errors	Test errors	Train Correct rate	Test correct rate
5.6094	0.0625	0.0358	0.1085	0.9106	0.7288
5.0469	0.0625	0.0313	0.0965	0.9218	0.7588
6.0781	0.0625	0.0335	0.1202	0.9162	0.6995
5.3438	0.0625	0.0358	0.0975	0.9106	0.7562
3.7813	0.0781	0.0335	0.0826	0.9162	0.7935
3.1094	0.0625	0.0313	0.0842	0.9218	0.7895
24.3594	0.0781	0.0536	0.1018	0.8659	0.7455
2.8594	0.0625	0.0335	0.0911	0.9162	0.7722
3.4531	0.0781	0.0380	0.0922	0.9050	0.7695
5.1250	0.0625	0.0402	0.1055	0.8994	0.7362
7.1875	0.0781	0.0380	0.0858	0.9050	0.7855
4.2656	0.0625	0.0402	0.1154	0.8994	0.7115
2.9375	0.0781	0.0469	0.0831	0.8827	0.7921
3.1250	0.0625	0.0380	0.0704	0.9050	0.8241
4.9531	0.0625	0.0313	0.1282	0.9218	0.6795
1.7656	0.0625	0.0447	0.0709	0.8883	0.8228
3.0781	0.0625	0.0425	0.1087	0.8939	0.7282
2.3125	0.0781	0.0492	0.0794	0.8771	0.8015
2.7813	0.0625	0.0291	0.0757	0.9274	0.8108
7.0000	0.0625	0.0358	0.0895	0.9106	0.7761
3.0938	0.0938	0.0380	0.0738	0.9050	0.8155
4.4063	0.0625	0.0358	0.0890	0.9106	0.7775
4.6875	0.0781	0.0313	0.0946	0.9218	0.7635
4.8906	0.0625	0.0313	0.1298	0.9218	0.6755
5.7813	0.0625	0.0425	0.0997	0.8939	0.7508
3.6250	0.0781	0.0313	0.0823	0.9218	0.7941
4.0000	0.0781	0.0358	0.0975	0.9106	0.7562
3.5156	0.0625	0.0335	0.0704	0.9162	0.8241
6.6563	0.0625	0.0335	0.1010	0.9162	0.7475
8.0781	0.0625	0.0291	0.1157	0.9274	0.7109
2.7031	0.0625	0.0268	0.0903	0.9330	0.7742
5.7500	0.0781	0.0335	0.1332	0.9162	0.6669
3.6875	0.0625	0.0335	0.1058	0.9162	0.7355
5.2813	0.0625	0.0358	0.1111	0.9106	0.7222
4.9844	0.0781	0.0313	0.1050	0.9218	0.7375
4.4375	0.0625	0.0268	0.1061	0.9330	0.7348
4.9219	0.0625	0.0380	0.1146	0.9050	0.7135
3.9531	0.0781	0.0380	0.1055	0.9050	0.7362
6.7344	0.0625	0.0402	0.1082	0.8994	0.7295
4.3438	0.0781	0.0313	0.0954	0.9218	0.7615

3.9063	0.0625	0.0291	0.0943	0.9274	0.7642
2.4219	0.0625	0.0358	0.0858	0.9106	0.7855
4.4219	0.0781	0.0291	0.1106	0.9274	0.7235
3.9063	0.0625	0.0380	0.0930	0.9050	0.7675
3.5313	0.0781	0.0313	0.0813	0.9218	0.7968

TABLE VII. TRAIN TIMES,TEST TIMES,TRAIN ERRORS AND TEST ERRORS OF PROBABILISTIC NEURAL NETWORK

Train times	Test times	Train errors	Test errors	Train Correct rate	Test correct rate
0.0469	0.1719	0	0.0799	1	0.8001
0.0313	0.1719	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0313	0.1719	0	0.0799	1	0.8001
0.0313	0.1719	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0469	0.1719	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0156	0.1563	0	0.0799	1	0.8001
0.0469	0.1563	0	0.0799	1	0.8001
0.0469	0.1563	0	0.0799	1	0.8001
0.0625	0.1719	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0469	0.1563	0	0.0799	1	0.8001
0.0469	0.1406	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0469	0.1406	0	0.0799	1	0.8001
0.0313	0.1719	0	0.0799	1	0.8001
0.0313	0.1719	0	0.0799	1	0.8001
0.0313	0.1719	0	0.0799	1	0.8001
0.0313	0.1719	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.1094	0.1719	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0469	0.1563	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0469	0.1563	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0469	0.1719	0	0.0799	1	0.8001
0.0313	0.1719	0	0.0799	1	0.8001
0.0156	0.1563	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001
0.0469	0.1875	0	0.0799	1	0.8001
0.0313	0.1719	0	0.0799	1	0.8001
0.0469	0.1563	0	0.0799	1	0.8001
0.0313	0.1719	0	0.0799	1	0.8001
0.0469	0.1563	0	0.0799	1	0.8001
0.0313	0.2188	0	0.0799	1	0.8001
0.0313	0.1563	0	0.0799	1	0.8001

VII. CONCLUSION

In this paper, the BP neural network algorithm, the probabilistic neural network algorithm and the perceptron network algorithm are applied to solve the classification of the students' scores on only course.

The BP neural network algorithm with the momentum factor or the gradient descent method,the probabilistic neural network algorithm and the self-organizing compete neural network algorithm are applied to the classification of the students' scores on five courses. And the better classifications are obtained.

And the probabilistic neural network algorithm are more adaptive to solving the classification of the students' scores on only one course than the BP neural network algorithm and perceptron neural network algorithm.

The probabilistic neural network algorithm and the BP neural network algorithm with the momentum factor are more adaptive to solving the classification of the students' scores on five courses than the BP neural network algorithm with the gradient descent method and the self-organizing compete algorithm.

However, there are some differences between these kinds of algorithm, such as the following:

1) The thinking of these three kinds of neural network algorithms is different. The BP neural network algorithm is a back-propagation neural network. The probabilistic neural network is built by combining the radial basis neural element with the competitive neural element. The perceptron neural network algorithm is an neural network algorithm with only an input and output layer.The self-organizing neural network algorithm is that the input layer and the compete layer have the common active function.

2) The probabilistic neural network algorithm is a supervise classification algorithm on only one course. And the probabilistic neural network algorithm and the BP neural network algorithm are more adaptive to classifying the students' scores on five course.

3) The train times of the probabilistic neural network algorithm and the perceptron neural network algorithm are shorter than those of the BP neural network algorithm on only one course. But the test times of the BP neural network algorithm and the perceptron neural network algorithm are shorter than those of the probabilistic neural network algorithm on only one course

4) The train times of the probabilistic neural network algorithm and the BP neural network algorithm with the gradient descent method are shorter than those of the BP neural network algorithm with the momentum factor and the self-organizing compete neural network algorithm on five only one course. But the test times of the BP neural network algorithm and the perceptron neural network algorithm are shorter than those of the probabilistic neural network algorithm on five courses.

5)The number of the parameters of the probabilistic neural network algorithm and the perceptron neural network algorithm is less than the number of the BP neural network algorithm. The BP neural network algorithm has many hidden layer, which give more adjusting parameters. But if there are too many hidden nodes, it will affect calculation speed. And it is difficult to decide the number of hidden layers. However, there is a parameter about the probabilistic neural network algorithm, the density of distribution.

6) The train error and test error of the probabilistic neural network algorithm are less than those of the perceptron neural network algorithm and the BP neural network algorithm on only course.

7) Train train error and test error of the probabilistic neural network algorithm are less than those of the BP neural network algorithm and the self-organizing compete neural network algorithm on five course.

TABLE VIII. TRAIN TIMES,TEST TIMES,TRAIN ERRORS AND TEST ERRORS OF PROBABILISTIC NEURAL NETWORK

Train times	Test times	Train errors	Test errors	Train Correct rate	Test correct rate
316.4688	0.0625	0.3307	0.3155	0.1732	0.2112
315.6094	0.0469	0.2570	0.2542	0.3575	0.3644
315.4844	0.0469	0.3642	0.3795	0.0894	0.0513
317.2969	0.0625	0.2436	0.2161	0.3911	0.4597
328.1563	0.0469	0.4000	0.3797	0	0.0506
322.2656	0.0625	0.3285	0.3654	0.1788	0.0866
316.4063	0.0469	0.3039	0.3102	0.2402	0.2245
328.8281	0.0469	0.2480	0.1871	0.3799	0.5323
328.6094	0.0469	0.3151	0.3435	0.2123	0.1412
329.3281	0.0469	0.3866	0.3718	0.0335	0.0706
332.9219	0.0469	0.3039	0.2993	0.2402	0.2518
331.7656	0.0625	0.3799	0.3704	0.0503	0.0740
327.4844	0.0469	0.2994	0.2588	0.2514	0.3531
315.4844	0.0469	0.2883	0.3030	0.2793	0.2425
319.7500	0.0469	0.3866	0.3536	0.0335	0.1159
315.7031	0.0625	0.3397	0.3334	0.1508	0.1666
315.5938	0.0469	0.3665	0.3563	0.0838	0.1093
315.2500	0.0469	0.3687	0.3675	0.0782	0.0813
315.1250	0.0625	0.2950	0.2649	0.2626	0.3378
318.8281	0.0625	0.3084	0.3243	0.2291	0.1892
316.0469	0.0469	0.3486	0.3238	0.1285	0.1905
315.0625	0.0469	0.2905	0.2569	0.2737	0.3578
319.4063	0.0469	0.4000	0.3888	0	0.0280
315.8906	0.0469	0.3151	0.3336	0.2123	0.1659
330.3906	0.0625	0.3330	0.3193	0.1676	0.2019
324.2500	0.0625	0.3307	0.3155	0.1732	0.2112
318.0156	0.0469	0.2570	0.2542	0.3575	0.3644
317.2344	0.0469	0.3642	0.3795	0.0894	0.0513
317.6719	0.0625	0.2436	0.2161	0.3911	0.4597
317.0000	0.0625	0.4000	0.3797	0	0.0506
316.8906	0.0469	0.3285	0.3654	0.1788	0.0866
318.3438	0.0625	0.3039	0.3102	0.2402	0.2245
314.8750	0.0781	0.2480	0.1871	0.3799	0.5323
316.0938	0.0781	0.3151	0.3435	0.2123	0.1412
310.9063	0.0781	0.3866	0.3718	0.0335	0.0706
310.4531	0.0469	0.3039	0.2993	0.2402	0.2518
310.3594	0.0469	0.3799	0.3704	0.0503	0.0740
310.0156	0.0625	0.2994	0.2588	0.2514	0.3531
310.2813	0.0469	0.2883	0.3030	0.2793	0.2425
311.4844	0.0781	0.3866	0.3536	0.0335	0.1159
310.2031	0.0469	0.3397	0.3334	0.1508	0.1666
310.7813	0.0781	0.3665	0.3563	0.0838	0.1093
310.7656	0.0469	0.3687	0.3675	0.0782	0.0813
311.7031	0.0625	0.2950	0.2649	0.2626	0.3378
323.4219	0.0781	0.3084	0.3243	0.2291	0.1892

8) The train correct rates of the probabilistic neural network algorithm,the BP neural network algorithm and the perceptron neural network are all 100% on only one course. The descent orders of the test correct rate of the neural network algorithms are the probabilistic neural network algorithm,the BP neural network algorithm and the perceptron neural network algorithm on only one course.

9) The train correct rates of the probabilistic neural network algorithm are all 100% on five courses, but those of the BP neural network algorithm with the momentum factor are all less than 99.44% and are all more than 97.77%, those of the BP neural network algorithm with the gradient descent method are all less than 93.30% and are all more than 86.59%, and the those of the self-organizing compete neural network algorithm are the least. The test correct rates of the probabilistic neural network algorithm and the BP neural network algorithm with the momentum factor are more than those of the BP neural network algorithm with the gradient descent method and the self-organizing compete neural network algorithm on five course.

Therefore, selecting the neural network classification algorithms of the students' scores on only one course, we are willing to take the probabilistic neural network algorithm. And selecting the neural network classification algorithms of the students' scores on five course, we are willing to take the probabilistic neural

ACKNOWLEDGMENT

F. A. Author thanks the anonymous reviews for their constructive connents.

REFERENCES

[1] P.L Nie and Y.C. He, “The Use of Statistics Analysis in Analyzing Students' Grades,” Journal of Xiaogan Vocational -Technical College, vol. 5, pp. 73–76,2002.(in Chinese)

[2] W.J. Sun, X.Q Zhan, S.W. Sun and S.X. Pan,“Using statistical software SPSS to analyze students' scores,” Journal of Ji Lin Institute of Chemical Technology, vol. 20, pp. 87–89,2003.(in Chinese).

[3] X. Li,“ The application of the multivariate statistical analysis in the students' scores,” Journal of Nanchang Institute of Aeronautical Technology, vol. 20, pp.58–62,2006.(in Chinese).

[4] N.F. Yu and B.K. Chen,“ Application of principal component analysis in Analysis of the Students' Grades,” Journal of Anhui Institute of Architecture & Industry, vol. 15, pp. 73-75,2007.(in Chinese).

[5] Fei Si Technology Production Research Center, “Neural Network and Matlab 7 Application,” Beijing:Publishing House of Electronics Industry,pp.116-126,2005.

[6] W. Weaber and A.Torres-Dodgen, “ Neural networks classification of the nearinfrared spectra of A-type Stars,”The Astrophysical Journal,vol.446,pp.300-317,1995.

[7] R.K.Gulati and H.P.Singh,“Stellar classification using principal component analysis and artificial neural networks,” Monthly Notices Royal Astronomical Society,pp.295-312,1998.

[8] C. Jinggang, “Neural network and analytical modeling of slope stability,” Dissertation Abstracts International(Section:B),vol.63-08 ,pp 3825,2002.

[9] C. Shuang,"the neural network theory and its application of facing the matlab tools", Hefei:Publishing the Science and Technology University of China，pp. 20-229,2009.

[10] Y.Shuying,"pattern recognition and intelligent calculation", Beijing:Publishing House of Electronics Industry,pp.140-190,2008.

[11] OMID KHAYAT,HAMTD REZA SHAHDOOSTI,etc.,"Image classification using Principal Feature Analysis",in Conf. Feb. 2008 7th WSEAS Int. Conf. On ARTIFICIAL INTELLIGENCE,KNOWLEDGE ENGINEERING and DATA BASES (AIKED'08),pp. 198-203..

[12] RATCHADAPORN OONSIVILAI, ANANT OONSIVILAI, "Probabilistic neural network classification for Model β-Glucan Suspensions",in Conf. Sep. 2007 Proceedings of the 7th WSEAS international Conference on Simulation,Modelling and Optimization, pp.159-164.

[13] AMAURY CABALLERO, KANG YEN, YECHANG FANG, JOSE L. ABREU,"Method for Classification in Interval-Valued Information Systems",Proceedings of the 12th WSEAS International Conference on AUTOMATIC CONTROL, MODELLING & SIMULATION, pp.242-247.

[14] OMAR JOSÉ LARA CASTRO,CRISTINA CASTEJÓN SISAMÓN , JUAN CARLOS GARCÍA PRADA ,"Bearing Fault Diagnosis based on Neural Network Classification and Wavelet Transform ",in Conf. Oct. 2006 Proceedings of the 6th WSEAS International Conference on Wavelet Anaysis & Multirate Systems, Bucharest, Romania, pp.22-29.

[15] ALEJANDRA RODRIGUEZ,ICIAR CARRICAJO,"Hybrid Approach to MK Classification of Stars Neural Networks and Knowledge-based Systems", in Conf. Feb.2007 Proceedings of the 6th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases,pp.110-118.