J. Basic. Appl. Sci. Res., 3(2)942-947, 2013
© 2013, TextRoad Publication

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

Comparing the Performance of two Neural Network Methods in a Process Fault Diagnosis System

***Mehdi Namdari, Hooshang Jazayeri-Rad, Nader Nabhani**

Department of Instrumentation and Automation, Petroleum University of Technology, Ahwaz, Iran

ABSTRACT

Artificial neural networks (ANN) can be employed as a fault diagnostic tool in numerous industries. A back propagation neural network (BPNN) consists of a multi-layer perceptron (MLP) which utilizes back propagation training (BP) technique. The BPNN is among the most effective techniques and has extensive applications. However, the BPNN has some shortcomings, such as weak capability of classification, possible entrapment in local optima, and requires a time-consuming iterative method during training to gain the best learning parameters. The probabilistic neural network (PNN) is a class of supervised neural network that is extensively used for pattern recognition. The PNN is appropriate to the same category of problems for which the BPNNis normally used, but the PNN has a number of key advantages over the BPNN. In this work, the BPNN and PNN classifiers are used for fault diagnosis of a chemical process comprising a continuous stirred tank reactor (CSTR) and a heat exchanger. The results show that the PNN classifier leaves behind the BPNN for fault diagnosis tasks with respect to features such as improvement in the classification rate and reduction of the training time.

KEYWORDS: artificial neural networks; back propagation algorithm; probabilistic neural network; continuous stirred tank reactor; classification rate; training time.

INTRODUCTION

Recently, automatic and on-line fault diagnosis systems have been improved to a high level of excellence in modern industries, because the innovative faults diagnosis systems have been used to stop severe damages. Fault diagnosis and detection are basically pattern-recognition tasks [1] where sensor data can be converted through pattern recognition into rich information suitable for decision making. The artificial neural networks can classify data efficiently; it would appear that it is a suitable tool to achieve fault diagnosis in a chemical plant. Artificial neural networks have numerous valuable properties regarding process fault diagnosis. They can deal with nonlinear and undetermined processes when no process model is required and the neural network learns the diagnosis via the information of the learning data [2]. Neural networks are noise tolerant and perform well with noisy measurements [3]. The capability to generalize the information along with the ability to adjust during their operations led them to have extensive applications in many areas [4-11].

The BPNN has been applied in many applications [12].However, as pointed out before; the BPNN has some major disadvantages. There are various learning algorithms or alterations of the BP algorithm in the literature, but most of these methods cannot totally resolve all problems associated with the BPNN. Normally, the convergence speed can be increased, but local minima difficulties are still present.

The PNN is a type of supervised neural network that is widely used for pattern recognition. It has a number of main advantages over the BPNN. The PNN rapidly learns from the training data. This speed of learning gives the PNN the ability of applying its learning in real time, deleting or adding training data as new circumstances stand up. A second benefit is that the PNN can be proved to always converge to the Bayes optimum solution as the number of training samples grows [13]. The probabilistic neural network fits into the family of the radial basis function neural networks, which due to their robustness are now extensively employed in numerous pattern-classification tasks [14-20].

In this work, a PNN classifier is employed for fault diagnosis of a chemical process comprising a CSTR and a heat exchanger. The performance of the PNN classifier will be compared with a BPNN. This paper is organized as follows. Section 2 presents an outline of the PNN and its architecture. Section 3 describes the features of the case study and gives the simulation results. Section 4 shows the PNN classifier training detail and the performance comparison results of the PNN classifier to BPNN. The last section specifies the related conclusions of the paper.

***Corresponding Author**: MehdiNamdari, Department of Instrumentation and Automation, Petroleum University of Technology, Ahwaz, Iran, Tel: +989391785876, Fax: +986115551021, E-mail: namdari_mehdi@yahoo.com.

1. Probabilistic neural network

The PNN proposed by Specht [21] is fundamentally based on the familiar Bayesian classifier technique commonly employed in many conventional pattern-recognition problems. The basic task accomplished by PNN is an estimation of the probability density function of the structure of each class from the given training samples using the Gaussian kernels. These estimated densities are then employed in a Bayes decision rule to accomplish the classification.

The PNN classifier is merely a parallel four-layer assembly: input layer, pattern layer, summation layer and decision layer. The PNN classifier model is demonstrated in Fig.1.

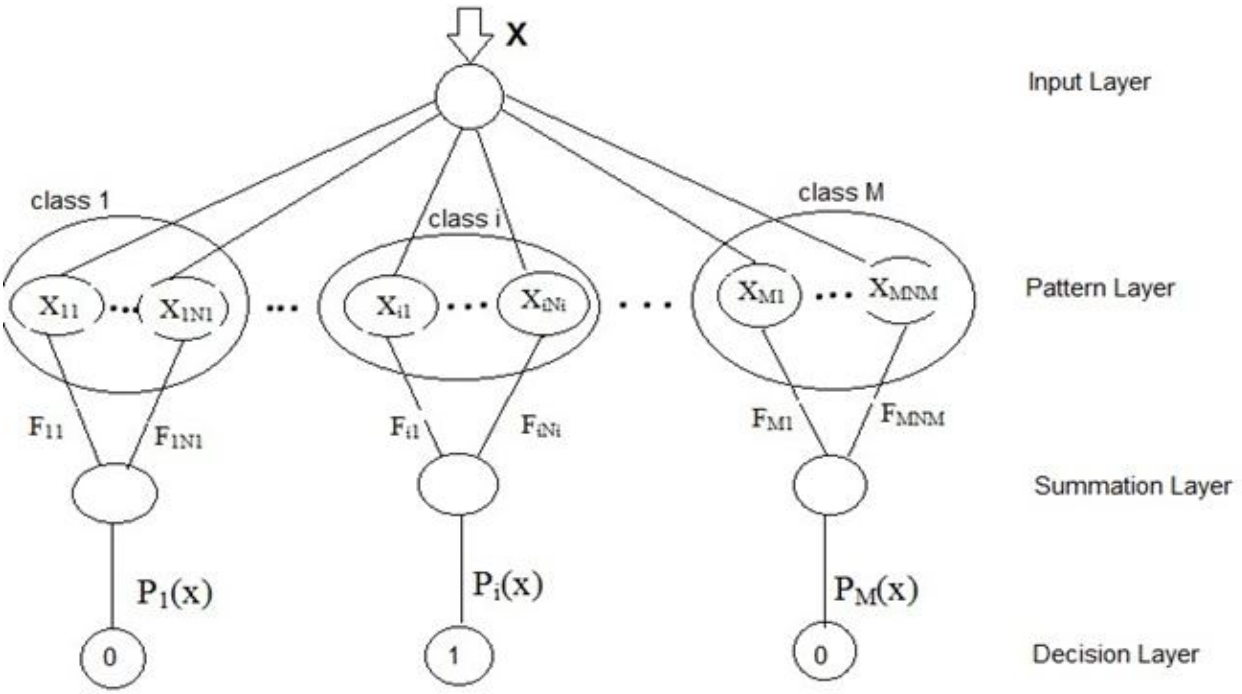

Fig. 1 The PNN classifier model

The input layer receives the input vector. The second layer or pattern layer comprises nodes with exponential functions. These nodes are known as the pattern units and are completely linked to the input nodes. There are as many pattern units as training samples. The transfer function of the pattern layer is the Gaussian function. Then the output of each pattern neuron, $F_{ij}(\boldsymbol{x})$, can be expressed as

$$F_{ij}(x) = \frac{1}{(2\pi)^{\frac{m}{2}}\sigma^m} exp\left[-\frac{(\boldsymbol{x}-\boldsymbol{x}_{ij})^T(\boldsymbol{x}-\boldsymbol{x}_{ij})}{2\sigma^2}\right] \qquad (1)$$

Where $\boldsymbol{x}_{ij}$is the *jth* training pattern of the *ith* class, *m* represents the dimension of the pattern vector $\boldsymbol{x}$ and σ is the smoothing parameter.

The third layer or the summation layer computes the average of the outputs from the second layer for each class. In the third layer, the average of the outputs of the pattern neurons of the class *i*,$P_i(x)$,is computed as

$$P_i(\boldsymbol{x}) = \frac{\sum_{j=1}^{N_i} F_{ij}(\boldsymbol{x})}{N_i} \qquad (2)$$

Where N_i is the number of training pattern neurons in the class i.

The fourth layer or the decision layer identifies the maximum of the outputs generated by the third layer. The corresponding class of the detected output determines the class label of the input pattern.

The smoothing parameter has the same value throughout the network and it is the only tuning made for optimizing the network as a classifier. There is no common method available to determine σ; however, its value is usually computed by trial and error. Training can be viewed as determining the best smoothing parameter σ for a set of training vectors.

2. Case study

The process used in this paper is shown in Fig. 2. This process is made up of a heat exchanger and a continuous stirred tank reactor where an irreversible first-order reaction $A \rightarrow B$ occurs. The reaction is catalytic and exothermic. The temperature of the reactor is regulated by pumping a part of the reactor outlet stream back to the reactor through the heat exchanger where the recycle flow is cooled by an external flow. The process has three feedback control loops that keep the recycle flow rate, the level of the reactor and the temperature of the reactor constant. In simulation of the process, it has been assumed that no fault occurs in control loops and the performances of the controllers are perfect. This process is selected because it reveals the most common features appearing in industrial processes. The model of the process can be found in Timo et al. [22].

Altogether, five variables (F_w, T_R, T_{out}, c_A, and c_B) are measured from the process (encircled variables in Fig. 2). Five typical fault situations have been chosen for this study. These faults are listed in Table 1. For each measured variable, 5% deviations from the steady state have been considered as normal situation and deviations beyond 5% are assumed to result in malfunctions. Various simulated faults together with the corresponding steady-state measurement pattern were used to train the network. Five fault situations and one normal condition altogether six classes of data have been given to the neural network to be classified. By varying the level of severity of every fault from 5% to 30%, for each class 200 samples were generated from the simulated process and a white noise signal with 10% variance has been added to the measured variables.70% of samples have been used for training and the remaining 30% for testing the network. Representative values for six classes of data are given in Table 2.

Table 1. Fault situations in the example process

Faults	Description
Fault 1	Low Concentration in Feed
Fault 2	Recycle flow low
Fault 3	Temperature set point (T_s) high
Fault 4	Fouled heat exchanger
Fault 5	Deactivated catalyst

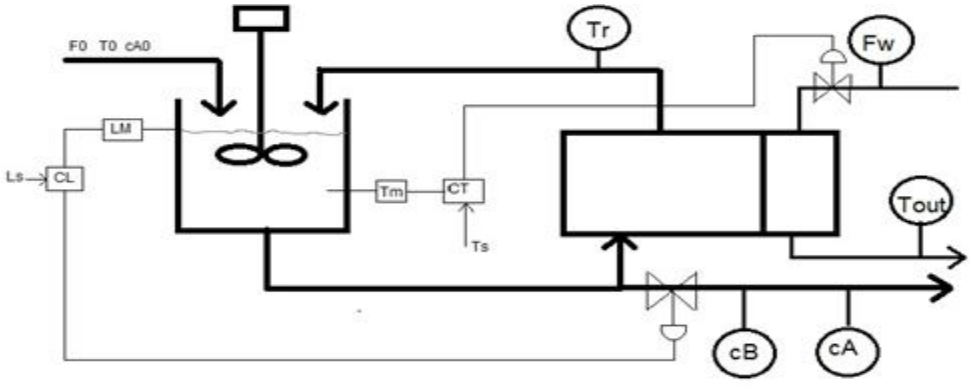

Fig. 2 The simulated process

Table 2. Typical values for six classes of data

Fault	F_w(kg/s)	T_R(°C)	T_{out}(°C)	c_A(mole/m^3)	c_B(mole/m^3)
Normal	2.25	80.58	56.69	21.51	1186.74
Fault 1	0.88	84.42	75.89	17.85	1051.20
Fault 2	2.59	70.51	50.73	20.96	1296.85
Fault 3	0.61	92.26	98.11	19.36	1222.91
Fault 4	3.72	77.46	38.72	21.86	1185.73
Fault 5	2.19	78.94	57.23	31.02	1273.91

3. RESULTS AND DISCUSSION

For training BPNN and the PNN classifier, samples were normalized in the range [-1,1]. 70% of data were randomly chosen as the training data and the remaining 30% were used as the testing data. For BPNN a two-layer feed-forward network with sigmoid activation function for both hidden and output layers have been chosen. Number of neurons in the hidden layer is very important. The network has been trained for different numbers of neurons in the hidden layer. Consequently, it has been concluded that six hidden neurons for this problem gave the best result. The network was trained several times with different random initial weight values. Consequently, the network with the best classification result has been chosen. The networks are trained until the mean square error of the training samples fell below 0.05.The classification results of the trained BPNN are given in Table 3.

For the PNN classifier the main issue is to determine the optimal smoothing parameter σ. For this purpose, the PNN classifiers are trained with different smoothing parameter values from 0.2 to 5. Fig. 3 shows the plot of the correct classification rate for the trained PNN classifier versus the smoothing parameter. As seen in this figure, the correct classification rate does not dramatically change with the smoothing parameter. The selected optimum value for the smoothing parameter was 0.9.

Classification results for the trained network are given in Table 3. As seen in this table, the performance of the PNN classifier is better than the BPNN with regard to the classification task. Moreover, since the PNN classifier needs no training iterations, the training procedure was performed almost instantly in comparison to the BPNN. BPNN has the problem of entrapment in local minima and the network should be trained with different initial values until the best result is achieved. On the other hand, the PNN classifier needs no initial value and the training process always ends up to the unique solution.

The disadvantages of the PNN classifier compared with the BPNN are: (i) its requirement for storing all training vectors (larger memory requirements) and using them to classify new vectors; (ii) slower execution speed for the classification of unknown patterns.

A review on literature reveals that the architecture selection of the BPNN is the foremost challenging problem. The number of hidden layers and also the number of hidden neurons in each layer are required to be determined. If each of the numbers of layers or neurons is inadequate, the network may not converge during the training; if each of the numbers of the layers or neurons is chosen to be too high, this will diminish the effectiveness of the network operation. It has been shown that a single hidden layer BPNN is a universal approximator, e.g. can approximate any continuous mapping with an arbitrary accuracy, provided adequately sufficient hidden neurons are available [22]. However, some applications in large and complex chemical plants may require more hidden layers to be considered in order to achieve adequate results [23]. In our case study, however, an extra hidden layer did not significantly improve the performance of BPNN.

Another difficulty of BPNN lays down in its use of the back propagation algorithm that is too slow for practical applications, especially if many hidden layers are employed. Also, it has been reported that the appropriate selection of training parameters in the BP algorithm, the gain and the momentum terms, sometimes are difficult [5]. There are many learning algorithms or modifications of the BP algorithm in the literature including Levenberg-Marquardt and Scaled Conjugate Gradient training algorithms. However, none of these methods were able to completely solve all the problems associated with the BP algorithm [18]. Due to the numerous practical limitations of BPNN, some authors proposed a combination of several BPNN models to increase the robustness or reliability of the single BPNN model [24]. On the other hand, as discussed before, none of these difficulties are observed in the PNN classifier. The disadvantage of PNN owing to its storage requirement for all the training data, because of the existing powerful processors with large memory sizes, cannot be considered a major weakness. In spite of the testing time of PNN being larger than BPNN, it is still much lower than the time constants in chemical plants. Therefore, it can easily be implemented in the online operation of chemical plants. From the above discussions, it can obviously be inferred that the PNN classifier is an appropriate substitute for the traditional BPNN in fault diagnosis systems.

Table 3. Correct classification rate for the trained networks

Network	Training samples	Test samples	All samples
BPNNclassifier	82.24 %	81.72 %	82.08%
PNN classifier	100 %	85.55 %	95.66 %

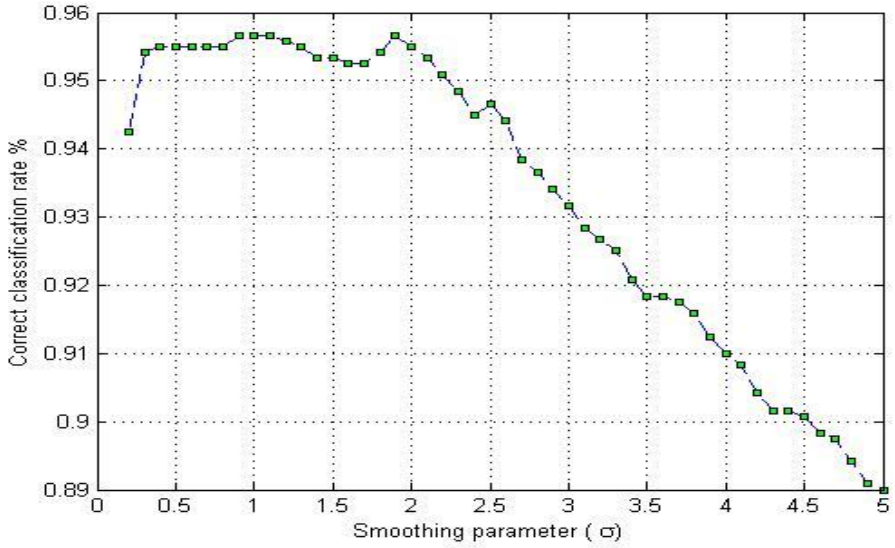

Fig. 3 Correct classification rate for the trained PNN classifier against the smoothing parameter (σ)

4. Conclusions

In this work, the potential of ANN to achieve fault diagnosis in chemical plants is demonstrated. ANN can automatically store information by learning from historical fault data without using any qualitative or quantitative model of the system. Fault data can be learnt by training the network on a set of data such as the values of steady state process variables for normal conditions and those for known fault conditions. Due to this distinct feature of ANN, it can be considered as an effective and attractive tool to achieve fault diagnosis in chemical plants.

Various structures of ANN can be employed as classifier in fault diagnosis systems including MLP and RBF networks. Traditional BPNN is among the most effective methods that had extensive applications consistent with the studies described in literature. So in this work the performance of BPNN is investigated and compared against the PNN classifer. The BPNN and PNN classifiers have been used for fault diagnosis of a chemical process containing a CSTR. The results show that the PNN classifier outperforms the BPNN for fault diagnosis tasks. A trained PPN classifier has improved classification rate compared to BPNN. Moreover, since the PNN classifier needs no training iterations its training time was considerably faster than BPNN. In the online training of the PNN classifier, the training samples can be added or taken out without retraining of the existing links. That means the existing weights are not modified but only new vectors are introduced into weight matrices while training. Therefore, it can be employed in real-time. BPNN has the problem of entrapment in local minima while training the PNN classifier always results in the unique solution. There is no need to set the initial weights of the PNN classifier. It is difficult to design the optimum architecture of the BP network. However, the PNN classifier can by design establish its structure consistent with the information of the input data without requiring users to choose. Disadvantages of the PNN classifier are: (i) testing time is lengthy for new samples; and (ii) it needs a large amount of memory for storing the training patterns. Finally, the PNN classifier guarantees to converge to the Bayes optimal solution as the number of training samples increases. These major advantages of the PNN classifier over BPNN make it a better choice for the pattern recognition and process fault diagnosis tasks.

REFERENCES

1. WATANABE K, MATSUURA L, ABE M, KUBOTA M., 1989. Incipient fault diagnosis of chemical processes via artificial neural networks. AIChE Journal**,** 35: 1803-1812.
2. WATANABE K, HIMMELBLAU D M., 1984. Incipient fault diagnosis in nonlinear chemical processes with multiple causes of faults. Chemical Engineering Science, 39: 491-508.
3. HUSSAIN M A, CHE-HASSAN C R, LOH K S, MAH K W., 2007. Application of artificial intelligence techniques in process fault diagnosis. Engineering Science and Technology**,** 2(3): 260-270.

4. HIMMELBLAU D M, KOREAN J., 2000. Applications of artificial neural networks in chemical engineering. Chem. Eng.,17(4): 373–392.
5. KOLVO H N., 1994. Artificial neural networks in fault diagnosis and control. Control Eng.,2**:** 80-101.
6. SHARMA R, SINGH K, SINGHAL D, GHOSH R., 2004. Neural network applications for detecting process faults in packed towers. Chemical Engineering and Processing**,** 43: 841-847.
7. KARIMI-MADAHI S S, SALAH P., 2012. A neural network based method for cost estimation 63/20kV and 132/20kV transformers. J. Basic. Appl. Sci. Res., 2(1): 939-945.
8. VAHEDI A., 2012. The predicting stock price using artificial neural network. J. Basic. Appl. Sci. Res., 2(3): 2325-2328.
9. MORADI-NEZHAD M, ASHOORI S, HOOSHMAND P, MIRZAEE M., 2012. Stuck drill pipe prediction with networks neural in maroon field. J. Basic. Appl. Sci. Res., 2(6): 5570-5575.
10. EBRAHIMI S, SHAJARI S, TARAZKAR M H., 2012. Prediction of agricultural commodity price using artificial neural networks: case of chicken price in Fars province, Iran. J. Basic. Appl. Sci. Res., 2(11): 11537-11541.
11. SURYADI A, TRIWULAN, AJI P., 2011. Artificial neural networks for evaluating the compressive strength of self compacting concrete. J. Basic. Appl. Sci. Res., 1(3): 236-241.
12. PARVOTI F Y, ANANDARAJAN M., 2002. Classifying inventory using an artificial neural network approach. Computer and Industrial Engineering, 40: 389–404.
13. RAMAKRISHNAN S, IBRAHIEM M M., 2009.Comparative study between traditional and modified probabilistic neural networks. Telecommun Syst., 40:67–74.
14. ARADHYA V N, NIRANJAN S K, KUMAR G H., 2010. Probabilistic neural network based approach for hand written character recognition. In the Proceedings of the International Conference on Advances in Computer, Communication Technology & Applications ACCTA. Orissa, India, pp: 9-13.
15. CHUNMEI X, HAO Z, CONGHUA H., 2010. Study of fault diagnosis based on probabilistic neural network for turbine generator unit In the Proceedings of the International Conference on Artificial Intelligence and Computational Intelligence. Sanya, China, pp: 275-279.
16. GHADERI K, ROSTAMI T, KARIMI A, GHASEMA S, KHODAMORADI S., 2012. Application of a probabilistic neural network in radial velocity curve analysis of the spectroscopic systems PT Vel, MU Cas, V459 Cas, δOriAa and δ Cir A. J. Basic. Appl. Sci. Res., 2(2): 1718-1724.
17. JIAN-DA W, PENG-HSIN C, YO-WEI C, YAO-JUNG S., 2008. An expert system for fault diagnosis in internal combustion engines using probability neural network. Expert Systems with Applications, 34: 2704–2713.
18. SONG L, XIU-YING L, WEN-XU W., 2011. Fault diagnosis of transformer based on probabilistic neural network. In the Proceedings of the Fourth International Conference on Intelligent Computation Technology and Automation. Shenzhen, China, pp: 128-131.
19. YANG, Q, GU L, WANG D, WU D., 2008. Fault diagnosis approach based on probabilistic neural network and wavelet analysis. In the Proceedings of the World Congress on Intelligent Control and Automation. Chongqing, China, pp: 1796-1799.
20. YONG-CHUN L, XIAO-YUN S, DONG-HHU L, HUI-QIN S., 2007. Application of combinatorial probabilistic neural network in fault diagnosis of power transformer. In the Proceedings of the Sixth International Conference on Machine Learning and Cybernetics. Hong Kong, pp: 19-22.
21. SPECHT D F., 1990. Probabilistic neural networks and the polynomial adaline as complementary techniques for classification. IEEE Transactions on Neural Networks, 1(1): 111–121.
22. TIMO S, HEIKKI N, HANNU K., 1991. Neural networks in process fault diagnosis. IEEE Transactions on Systems, 21(4): 815-825.
23. HOSKINS J C, KALIYUR K M, HIMMELBLAU D M., 1991. Fault diagnosis in complex chemical plants using artificial neural networks. AIChE Journal**,** 37(1): 137-141.
24. ZHANG J., 2006. Improved on-line process fault diagnosis through information fusion in multiple neural networks. Computers and Chemical Engineering, 30: 558-571.