

Classification and concentration prediction of combustible gas based on BPNN and PCA

Hanguang Xiao^{1, a, *}, Junchan Xu^{2, b}, Bin Tang¹ and Zhou Zhang¹

¹Chongqing Key Laboratory of Modern Photoelectric Detection Technology and Instrument School of Optoelectronic Information, Chongqing University of Technology, Chongqing, P. R. China

²Faculty of Arts and Humanities, King's College London, United Kingdom

^aemail: simenxiao1211@163.com(*Corresponding Author), ^bemail: ablula@163.com

Keywords: Back propagation neural network (BPNN); Principal component analysis (PCA); Concentration prediction; Classification of combustible gas.

Abstract: Detection of combustible gases is very important to reduce the modality and disability of human in both of civil and military situation. In this paper, a method of detection combustible gases of acetone and ethanol was proposed by using back propagation neural network (BPNN) and principal component analysis (PCA). The gas data were collected using some metal oxide semiconductor (MOS) gas sensors exposed to the mixture combustible gases of different concentration. The features of low and high frequency domain were extracted to establish a feature vector of 432 dimensions. Then PCA was used to reduce the dimension of feature vector from 432 to 11 which retained 99% information. The results showed the binary classification accuracy of BPNN is up to 100% for train, validation and test when distinguishing the combustible gas from the air. The mean and variance of error (0.004 ± 0.008) for concentration prediction were obtained based on BPNN and PCA. The results demonstrated that the proposed method is effective for classification and concentration prediction of combustible gas.

Introduction

Recently, the production and application of combustible gases has been increasing with industrialization. The detection of combustible gases is very important to avoiding gas leakage and serious accident [1-3]. Especially, the quantitative analysis and classification of mixture combustible gases is a hotspot in the research field [4-7].

Artificial neural network has been proved to be a powerful tool in the concentration estimation [5, 7-9]. Zhao et al. used the back propagation (BP) method and radial basis function (RBF) neural network in the data analysis of metal oxide gas sensors and arrays, and obtained good accuracy of concentration prediction [7]. Zhang et al. studied a concentration estimation of indoor contaminants for the air quality monitoring in dwellings by using chaos-based optimization of BPNN and integrated into a self-designed portable E-nose instrument [8]. Zhang et al. studied the concentration estimation of multiple kinds of chemicals using multilayer perceptron (MLP) neural network and got a good performance in accuracy and convergence [9]. Ziyatdinov et al. proposed a chemical sensing system based on an array of 16 metal-oxide gas sensors and used PCA to analyze the difference of the first three principle component [5]. Although many methods were proposed based on artificial neural networks, there are few researches about the binary classification and concentration prediction of combustible gases via combining BPNN and PCA, especially for the high dimension of feature vectors as the Ref. 5.

In this work, a binary classification and concentration prediction of combustible gas were studied by using BPNN and PCA. The paper is organized as follows: Section 2 describes materials and methods including data collection, feature extraction and introduction of BPNN; Section 3 is the results and discussion about classification between combustible gas and pure air by BPNN and prediction of concentration of combustible gas by BPNN and PCA; Section 4 presents the conclusions.

Materials and methods

Data collection: The measured data was collected by using a chemical sensing system based on an array of 16 metal-oxide gas sensors and an external mechanical ventilator to simulate the biological respiration cycle. The tested gas classes (12 in total) formed a relatively broad combination of two analytes, acetone and ethanol, in binary mixtures.

Three concentrations doses 0.1, 0.3 and 1 vol. % were used to prepare the dilutions in water for the pure analytes. The same dilutions were used to generate gas mixtures. The gas classes included samples of pure ethanol ('lab' attribute eth-0.1, eth-0.3 and eth-1), samples of pure acetone (ace-0.1, ace-0.3 and ace-1), samples of binary mixtures of ethanol and acetone (ace-0.1-eth-0.1, ace-0.1-eth-0.3, ace-0.3-eth-0.1, ace-0.1-eth-1 and ace-1-eth-0.1) and samples of water dilutions without any analyte (air) giving a total number of 12 classes. The choice of these analytes and concentrations was not affected by any particular application constraint, except that the sensors of selected models show consistent and diverse responses among the gas classes.

Raw data of each sample contains 16 time-series (one time-series per sensor). Each time-series was recorded during 5 min at a sample rate of 25 Hz (samples per second), providing 7500 data points per time-series. The total number of attributes per sample in raw data is 120000. More details about the experiment and data collection is available in Ref 5 and the website <http://archive.ics.uci.edu/ml/datasets.html>.

Feature extraction: The raw signals were pre-processed with a median filter, and were filtered by two Butterworth filters of 3rd order: a low-pass filter (cut-off frequency 0.01Hz) and a high-pass filter (pass-frequency 0.07 Hz) to generate the low/high frequency signals respectively. Then, the filtered singles were divided into each segment signals during each respiratory cycles. The amplitudes of the high/low frequency signal as two features. A cycle-independent feature per single measurement also was introduced, defined as the maximum of the low-frequency signal over the course of the measurement. The extraction of more features refers to Ref. 5.

Feature data set includes three types of features extracted from each time-series. Each time-series (one time-series per sensor) is associated with 1 maximum features, 13 high-frequency features and 13 low-frequency features (the features correspond to the first 13 respiration cycles, respectively). The total number of attributes per sample in feature data set is 432.

Back Propagation Neural Network (BPNN): BPNN is one kind of neural network, which is commonly used in prediction, pattern classification, data mining, et al, without any prior knowledge about the existing problem. Figure 1 shows a three-layer BPNN topology, which includes an input layer, hidden layer and output layer. Input layer is also a feature layer into which the feature vectors input one by one. Hidden layer link to input layer and output layer with weights and implement a computation by a transfer function, such as pure linear function, sigmoid function. Output layer represents the target layer which outputs the prediction results of the BPNN model.

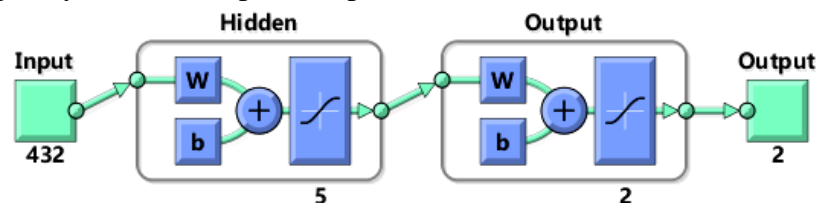


Fig. 1. Three-layer BPNN topology

In order to improve the prediction accuracy of BPNN model, training is required to make the network more intelligent. The ANN is trained with a set of input and known output pairs called the training set. At the beginning of the training process, the network weights are initialized with the data provided from the laboratory. Then an error back propagation algorithm is used to adjust the weights till the prediction error is acceptable. The error back propagation algorithm is showed as above.

Step 1: Initialize the weights and thresholds, and specify error tolerance, maximum iteration number and other parameters.

Step 2: Randomly select the kth sample and input into neural network. Feature vector and target

vector of the k th sample is:

$$x(k) = (x_1(k), x_2(k), \dots, x_n(k)) \quad (1)$$

$$do(k) = (d_1(k), d_2(k), \dots, d_q(k)) \quad (2)$$

Step 3: Compute the input and output of neurons of hidden layer:

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

$$ho_h(k) = f(hi_h(k)) \quad h = 1, 2, \dots, p \quad (4)$$

$$yi_o(k) = \sum_{h=1}^p w_{ho} ho_h(k) - b_o \quad o = 1, 2, \dots, q \quad (5)$$

$$yo_o(k) = f(yi_o(k)) \quad h = 1, 2, \dots, q \quad (6)$$

Step 4: Compute $\delta_o(k)$ the partial derivative of error function with respect to neurons of output layer using the outputs of BPNN and targets:

$$\frac{\partial e}{\partial w_{ho}} = \frac{\partial e}{\partial yi_o} \frac{\partial yi_o}{\partial w_{ho}} \quad (7)$$

$$\frac{\partial yi_o(k)}{\partial w_{ho}} = \frac{\partial (\sum_h^p w_{ho} ho_h(k) - b_o)}{\partial w_{ho}} = ho_h(k) \quad (8)$$

$$\begin{aligned} \frac{\partial e}{\partial yi_o} &= \frac{\partial (\frac{1}{2} \sum_{o=1}^q do(k) - yo_o(k))^2}{\partial yi_o} \\ &= -(d_o(k) - yo_o(k)) yo'_o(k) \\ &= -(d_o(k) - yo_o(k)) f'(yi_o(k)) \triangleq -\delta_o(k) \end{aligned} \quad (9)$$

Step 5: Compute $\delta_h(k)$ the partial derivative of error function with respect to neurons of hidden layer using the weights between hidden and output layer of BPNN, $\delta_o(k)$ and outputs of hidden layer:

$$\frac{\partial e}{\partial w_{ho}} = \frac{\partial e}{\partial yi_o} \frac{\partial yi_o}{\partial w_{ho}} = -\delta_o(k) ho_h(k) \quad (10)$$

$$\frac{\partial e}{\partial w_{ih}} = \frac{\partial e}{\partial hi_h(k)} \frac{\partial hi_h(k)}{\partial w_{ih}} \quad (11)$$

$$\frac{\partial hi_h(k)}{\partial w_{ih}} = \frac{\partial (\sum_{i=1}^n w_{ih} x_i(k) - b_h)}{\partial w_{ih}} = x_i(k) \quad (12)$$

Step 6: Correct $w_{ho}(k)$ the weights between hidden and output layers using $\delta_o(k)$ and the outputs of hidden layer:

$$\Delta w_{ho}(k) = -\mu \frac{\partial e}{\partial w_{ho}} = \mu \delta_o(k) ho_h(k) \quad (13)$$

$$w_{ho}^{N+1} = w_{ho}^N + \eta \delta_o(k) ho_h(k) \quad (14)$$

Step 7: Correct $w_{ih}(k)$ the weights between input and hidden layers using $\delta_h(k)$ and the feature vectors of input layer:

$$\Delta w_{ih}(k) = -\mu \frac{\partial e}{\partial w_{ih}} = -\mu \frac{\partial e}{\partial hi_h(k)} \frac{\partial hi_h(k)}{\partial w_{ih}} = \delta_h(k) \quad (15)$$

$$w_{ih}^{N+1} = w_{ih}^N + \eta \delta_h(k) x_i(k) \quad (16)$$

Step 8: Compute outputs of output layer using the adjusted new weights and the error of BPNN:

$$E = \frac{1}{2m} \sum_{k=1}^m \sum_{o=1}^q (d_o(k) - y_o(k))^2 \quad (17)$$

Step 9: Judge the stop condition according to the error and the maximum step of iteration, if positive, stop the loop, otherwise, continue and go to the step 3.

Results and discussion

Classification between combustible gas and pure air by BPNN: In order to identify if combustible gas exist or not, a binary classification was studied using BPNN, in which the pure air is one class and combustible gas is another with different concentrations. According to this, 8 feature vectors of negative samples and 50 feature vectors of positive samples were obtained. The class labels of negative and positive sample is designed to a 2 element vector [0, 1] and [1, 0]. The dimension of all feature vectors is 432. Normalization of feature vectors was conducted because of the different order of magnitudes of features. All 58 samples were divided into three set: train set, validation set and test set, in which the number of samples are 70%, 15% and 15% of all samples. The three-layer BPNN topology is shown in figure 1. The number of neurons in input layer, hidden layer and output are 432, 4 and 2. The neuron number of hidden layer is a changeable parameter.

Figure 2 shows classification results of train, test and validation of BPNN in subfigure (a) mean squared error and subfigure (b) error histogram. From figure 2(a), one saw the train, validation and test of BPNN got good performances which reached to about 1.0×10^{-7} mean squared error for the three sets. The histogram of error showed most of errors concentrate near zero, which represents the variance of errors is small.

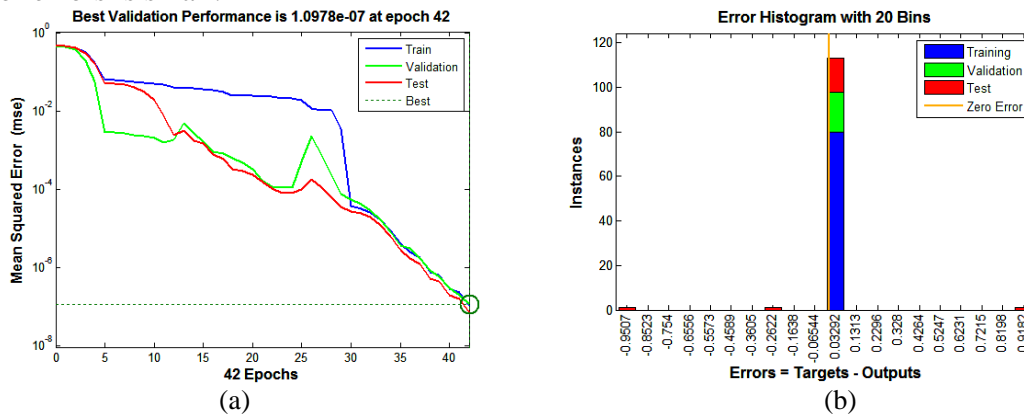


Fig. 2. Classification results of train, test and validation of BPNN: (a) mean squared error and (b) error histogram

After a transformation on the outputs of BPNN according to the error of each sample, one got the predicted class labels of samples. The original class labels (or targets) and predicted class labels (or outputs) of samples were showed in figure 3(a). From the figure, all samples were predicted right, which also can be derived from figure 3(b) all confusion matrix.

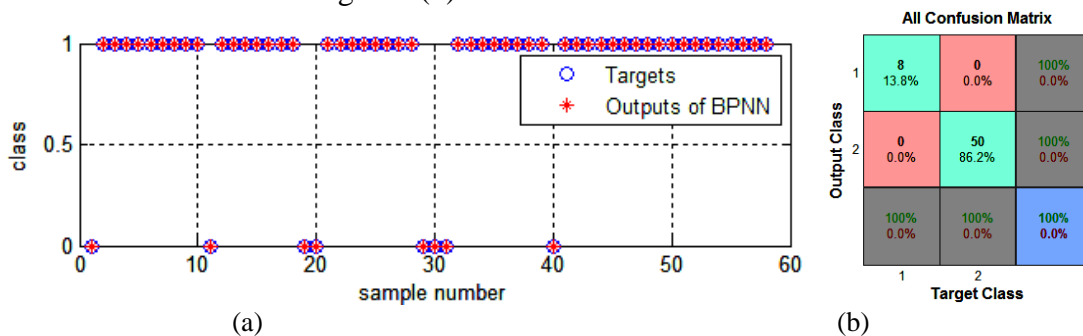


Fig. 3. Prediction results (a) and confusion matrix (b) of BPNN

Because the performance of BPNN is sensitive to the neuron number of hidden layer, one discussed the effects of the parameter on the prediction accuracy of BPNN. The classification

results of BPNN with different number of neurons are showed in table 1. In table 1, P and N are the number of positive and negative samples. TP, FN, TN and FP represent the number of true positive, false negative, true negative and false positive in the outputs of BPNN. Qp, Qn, and Q are the percentages of right predicted positive, negative and all samples, which represent the accuracies. When the number of neuron of hidden layer is 4, the best classifying accuracies 100% were obtained for Qp, Qn, and Q. The accuracies decrease when the number of neuron decreases or increase from 4.

Table 1. Classifying accuracies of BPNN with different number of neurons

Number of Neuron	P	N	TP	FN	TN	FP	Q _p (%)	Q _n (%)	Q (%)
1	44	14	36	8	0	14	81.8	0	62.1
2	56	2	50	6	2	0	89.3	100	89.7
3	53	5	50	3	5	0	94.3	100	94.8
4	50	8	50	0	8	0	100	100	100
5	56	2	50	6	2	0	89.3	100	89.7

Prediction of concentration of combustible gas by BPNN: In prediction of concentration, one used the concentration of two combustible gas: acetone and ethanol as the outputs of BPNN, which is a typical regression analysis. The train, validation and test set were prepared as the above section. The number of neurons in input layer, hidden layer and output are 432, 20 and 2. Figure 4 shows (a) mean squared error and (b) error histogram of train, test and validation of BPNN. From figure 4(a), one saw the train, validation and test of BPNN got acceptable performances which reached to about $1.0 \cdot 10^{-2}$ and $1.0 \cdot 10^{-1}$ mean squared error for the validation and test sets. The histogram of error showed most of errors concentrate in the region $[-0.2, 0.2]$, and the mean and variance of errors is -0.005 ± 0.009 .

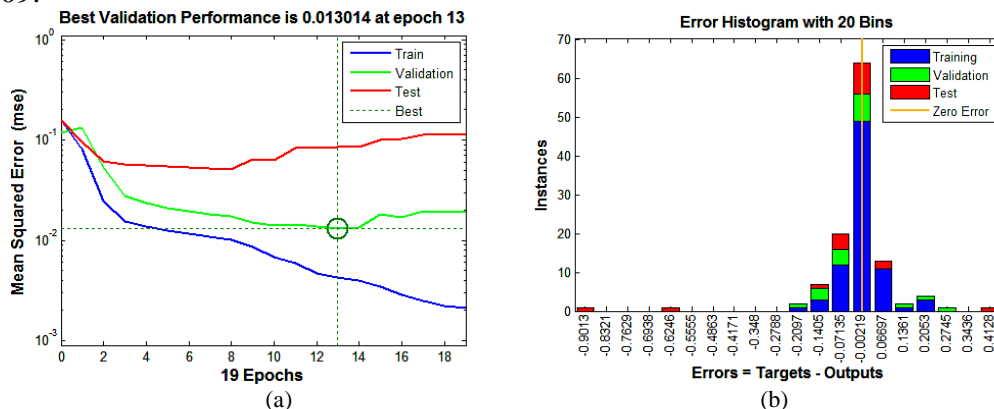


Fig. 4. Mean squared error (a) and error histogram (b) of train, test and validation of BPNN

The results of concentration prediction of BPNN for two combustible gas are showed in figure 5: (a) acetone and (b) ethanol. From the figure 5, we found most of outputs of BPNN are in accord with the original targets of samples. At same time, there are also a few large error of prediction.

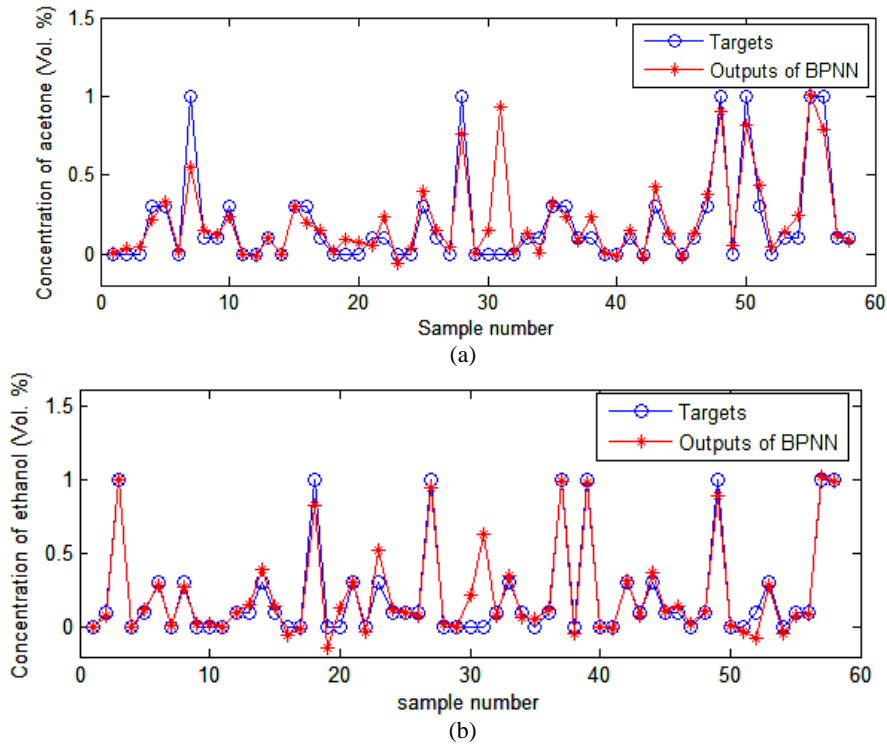


Fig. 5. Prediction results of BPNN: (a) acetone and (b) ethanol

Because the dimension of feature vector is too large, the computational cost is high. Meanwhile, it may cause negative effect on the accuracy or performance of BPNN. Thus, one used principle component analysis (PCA) to reduce the dimension of feature vector from 432 to 11 including 99% information. Then one conducted a similar prediction of the concentration of two combustible gas. The error analysis in figure 6 and comparisons of targets and outputs of BPNN in figure 7 showed that the better results were obtained after the feature reduction using PCA. Firstly, the best validation performance is 0.0047, which is lower than without PCA. Secondly, the mean and variance of error (0.004 ± 0.008) is better, which also were observed in figure 7. Thirdly, the computation time was reduced from about one minute to about 5 seconds, because both the neuron number of input layer and hidden layer decrease from 432 to 11 and from 20 to 4, respectively.

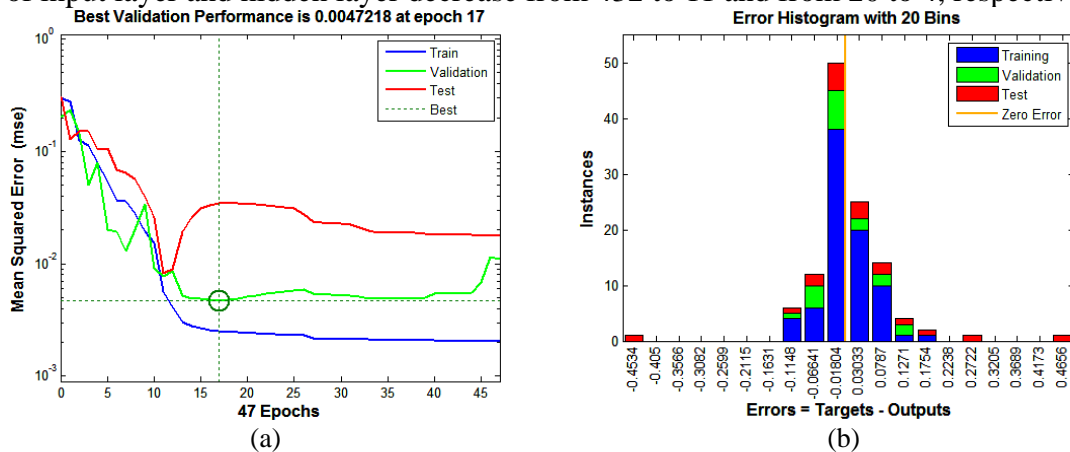


Fig. 6. (a) Mean squared error and (b) error histogram of train, test and validation of BPNN with PCA

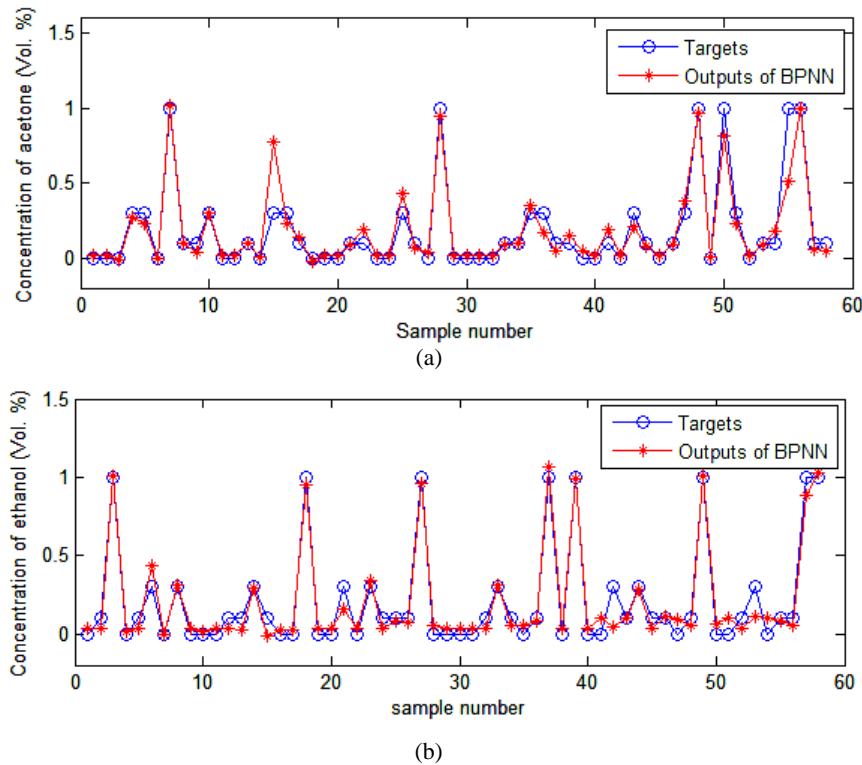


Fig. 7. Prediction results of BPNN with PCA: (a) acetone and (b) ethanol

Conclusion

In the paper, a binary classification and concentration prediction of combustible gas were studied by using BPNN and PCA. For the classification, 100% accuracy were obtained for train, validation and test set by using the high dimension feature vectors, which means it is an effective method to use BPNN to detect whether combustible gas exists or not. For those concentration prediction, the accuracy is not too high and the computational cost is high if the original features directly were used to train BPNN and test. PCA used to reduce the dimension of feature achieved a better prediction results and decreased the computational cost effectively. Therefore, BPNN and PCA are an effective way to realize the classification and concentration prediction of combustible gas. In order to improve the accuracy furtherly, some new methods of feature extraction should be employed in further research.

Acknowledgment

This research is funded by the National Natural Science Foundation of China (Grant No. 61501070). This research is also funded by Chongqing Natural Science Foundation (Grant No. cstc2014jcyjA10040).

References

- [1] A. Somov, A. Baranov, D. Spirjakin, R. Passerone, Circuit Design and Power Consumption Analysis of Wireless Gas Sensor Nodes: One-Sensor Versus Two-Sensor Approach, *Ieee Sensors Journal*, 14 (2014) 2056-2063.
- [2] T.T. Sunil, S. Chaudhuri, V. Mishra, Optimal selection of SAW sensors for E-Nose applications, *Sensors and Actuators B-Chemical*, 219 (2015) 238-244.
- [3] S.S. Varghese, S. Lonkar, K.K. Singh, S. Swaminathan, A. Abdala, Recent advances in graphene based gas sensors, *Sensors and Actuators B-Chemical*, 218 (2015) 160-183.
- [4] J.Z. Ou, C.K. Yao, A. Rotbart, J.G. Muir, P.R. Gibson, K. Kalantar-zadeh, Human intestinal gas

measurement systems: in vitro fermentation and gas capsules, *Trends in Biotechnology*, 33 (2015) 208-213.

[5] A. Ziyatdinov, J. Fonollosa, L. Fernández, A. Gutierrez-Gálvez, S. Marco, A. Perera, Bioinspired early detection through gas flow modulation in chemo-sensory systems, *Sensors and Actuators B: Chemical*, 206 (2015) 538-547.

[6] H. Long, W. Zeng, H. Zhang, Synthesis of WO₃ and its gas sensing: a review, *Journal of Materials Science-Materials in Electronics*, 26 (2015) 4698-4707.

[7] L. Zhao, J.Q. Qi, J. Wang, P.J. Yao, The study of using an extreme learning machine for rapid concentration estimation in multi-component gas mixtures, *Measurement Science & Technology*, 23 (2012).

[8] L. Zhang, F.C. Tian, S.Q. Liu, J.L. Guo, B. Hu, Q. Ye, L.J. Dang, X.W. Peng, C. Kadri, J.W. Feng, Chaos based neural network optimization for concentration estimation of indoor air contaminants by an electronic nose, *Sensors and Actuators a-Physical*, 189 (2013) 161-167.

[9] L. Zhang, F.C. Tian, Performance Study of Multilayer Perceptrons in a Low-Cost Electronic Nose, *Ieee Transactions on Instrumentation and Measurement*, 63 (2014) 1670-1679.