CHAPTER 5

Classification and Quantification of Binary Mixtures of Gases/Odors

This chapter presents the results of experiments done for the classification and quantification of two volatile organic compounds (VOCs) viz. Acetone (CH_3COCH_3) and 2-propanol ($CH_3CHOHCH_3$) in their individual as well as in mixture forms. A sensor array consisting four sensor elements was fabricated by the author using thick film fabrication technology as described in Chapter-2 previously. The steady state responses of the sensor array were collected for the mentioned VOCs in their individual as well as in mixture forms. A hierarchical system consisting of gating network and three quantification networks was designed to classify and then quantify the individual and the mixture of VOCs. The classification results of the gating network were ensured using back-propagation neural network (BPNN) and support vector machine (SVM). For quantification, multioutput support vector regression (*M-SVR*) technique was used for both, individual as well as binary mixture datasets separately. *k*- fold cross validation scheme was adopted for all the experiments.

5.1 Introduction

In the current research area of e-nose, the analysis of mixture of gases/odors presents a challenging task for researchers. The classification/quantification task of a mixture of gases/odors is more complex as compared to the individual gases/odors. A large amount of literature is available where the problems of gases/odors mixture have been studied. Researchers have reported their studies on binary, ternary and quaternary mixture of gases. Present work is related to the binary mixture of VOCs.

Capone *et al.* (2001) have reported analysis of CO and CH_4 gas mixture using a micromachined array. Ozmen *et al.* (2006) have reported detection of composition of gas mixture by a pathalocyanine-coated QCM sensor array and artificial neural network. Distante *et. al.* (2003) have reported the detection of individual and binary mixture of gases using array of chemical sensors prepared by sol-gel method and support vector machines. Gulbag *et al.* (2006) have reported quantitative classification of binary gas mixture using neural networks and adaptive neuro-fuzzy inference

system. Penza *et al.* (2002) have reported quantification of individual VOCs in a binary mixture by SAW multisensory array with BPNN classification and regression techniques. Various fuzzy modeling based approaches have also been proposed [Deng *et al.* (2011)] which can be utilized on sensor array data. In the present work, binary mixture of volatile organic compounds (VOCs) viz. (CH_3COCH_3) and 2-propanol ($CH_3CHOHCH_3$) have been studied in their single as well as in mixture forms with multioutput support vector regression technique.

Feed forward neural networks (FFNNs) with back-propagation algorithms are widely used for pattern recognition task. But they may suffer slow convergence rate and yield suboptimal solution sometimes. Due to these problems, support vector machines (SVMs) have been used in the present work to classify the test gases and their mixture. SVM provides good generalization performance in context of odors classification despite the fact that it does not incorporate the problem domain knowledge [Gulbag and Temurtas (2006)]. For quantification of single gas/odor and individual component in their mixture, the multioutput support vector regression (*M-SVR*) proposed in [Perez-Cruz *et al.* (2002)] has been used.

So, in the present study, successful classification and quantification of VOCs (Acetone and 2-propanol) in individual as well as in binary mixture form has been obtained using steady state responses of thick film gas sensor array along with PCA-ANN and PCA-SVM and PCA-*M-SVR* approach. The SVM and *M-SVR* have been used for multi-class problem. Further, in our previous studies (in Chapter-3 and Chapter-4), it was found that the PCA preprocessed data provided improvement in classification and quantification of individual gases/odors. In the present study also, it has been found that the PCA transformation has improved classifier's and quantifier's performance for binary mixture dataset.

This chapter has been divided into 5 main sections. Section 5.2 deals with the experimental background and methods followed. Section 5.3 describes the schemes adopted for classification and quantification. Section 5.4 includes results and discussion. Section 5.5 concludes the reported experiment and its findings.

5.2 Materials and Methods

The classification and quantification approach followed in this paper can be described as per the block diagram shown in Fig. 5.1. Its first block represents the data

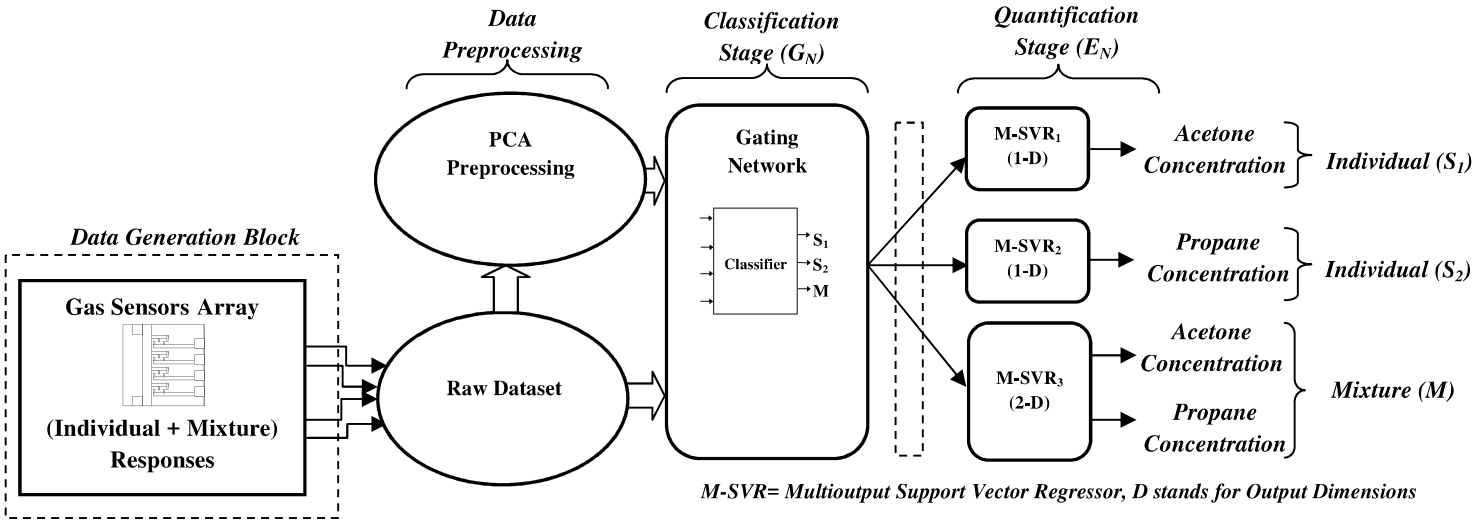


Fig.5.1 Schematic of methods used for gases/odors mixture analysis

generation block consisting of a thick film gas sensor array which generates the steady state responses for the VOCs and their mixtures. The raw dataset was then fed into the succeeding PCA data preprocessing block. Next is the classification stage which is basically a gating network and acts as a switch which activates particular quantification network in subsequent stage. The raw data and their PCA preprocessed versions were used to have a comparative performance analysis.

5.2.1 Experimental Background

The schematic of the fabricated sensor array is shown in Fig. 5.2. An Alumina substrate is used as a substrate for sensor fabrication. It consists of four different sensor elements including pure SnO_2, Pd-doped, Pt-doped and ZnO-doped sensor elements, respectively. The experimental detail of the fabricated sensor array has been already described in Chapter-2 previously.

The responses of the fabricated sensor array were obtained for varying concentrations of Acetone and 2-propanol and their mixtures in gaseous form in a locally developed test chamber. The responses for individual VOCs were taken by subsequently increasing the ambient concentration as shown in Fig. 5.3. The responses for individual gases/odors and their mixture were collected by taking the concentrations ratios as shown in Fig. 5.4 and Fig. 5.5 respectively. For the sake of simplicity, other similar response curves (similar to Fig. 5.5, by increasing Acetone concentration at different concentration of 2-propanol) are not shown.

5.2.2 Data Extraction and Interpretation

Dataset was prepared from the steady state responses of individual gases and their mixture and then used in classification and quantification tasks. The individual responses were extracted from each concentration band viz. 150-300 ppm (called 300 ppm band), 300-450 ppm (called 450 ppm band etc.) and so on. Similarly, the responses of the mixture were collected for the test gases/odors in combination of different concentration bands as shown in Fig. 5.3. Six response points for each concentration band were extracted. The steady state responses of individual gas as well as their mixtures are shown in Fig. 5.4(a)-(b) and Fig. 5.5(a)-(f), respectively. Raw dataset was prepared from different concentration bands which consisted total of 150 data samples. Raw dataset was first plotted on a 2-D scatter plot considering two of the four sensors data at a time. Six such plots were obtained one of which is shown

in Fig. 5.6. It is evident from this figure that the data belonging to different class of mixtures are quite inseparable.

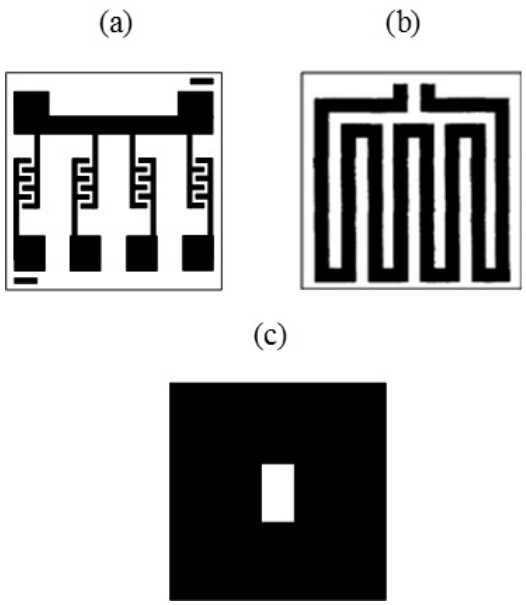


Fig. 5.2 Schematic of fabricated sensor array with common electrode (a) Sensor electrodes (a) Heater pattern (b) Sensor printing mask

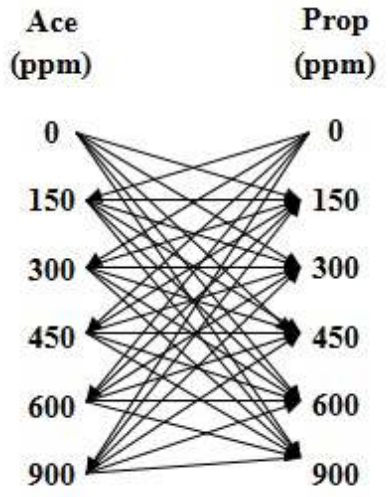


Fig. 5.3 Combinations of selected Acetone and 2-propanol concentrations in their mixture used in the gas sensing

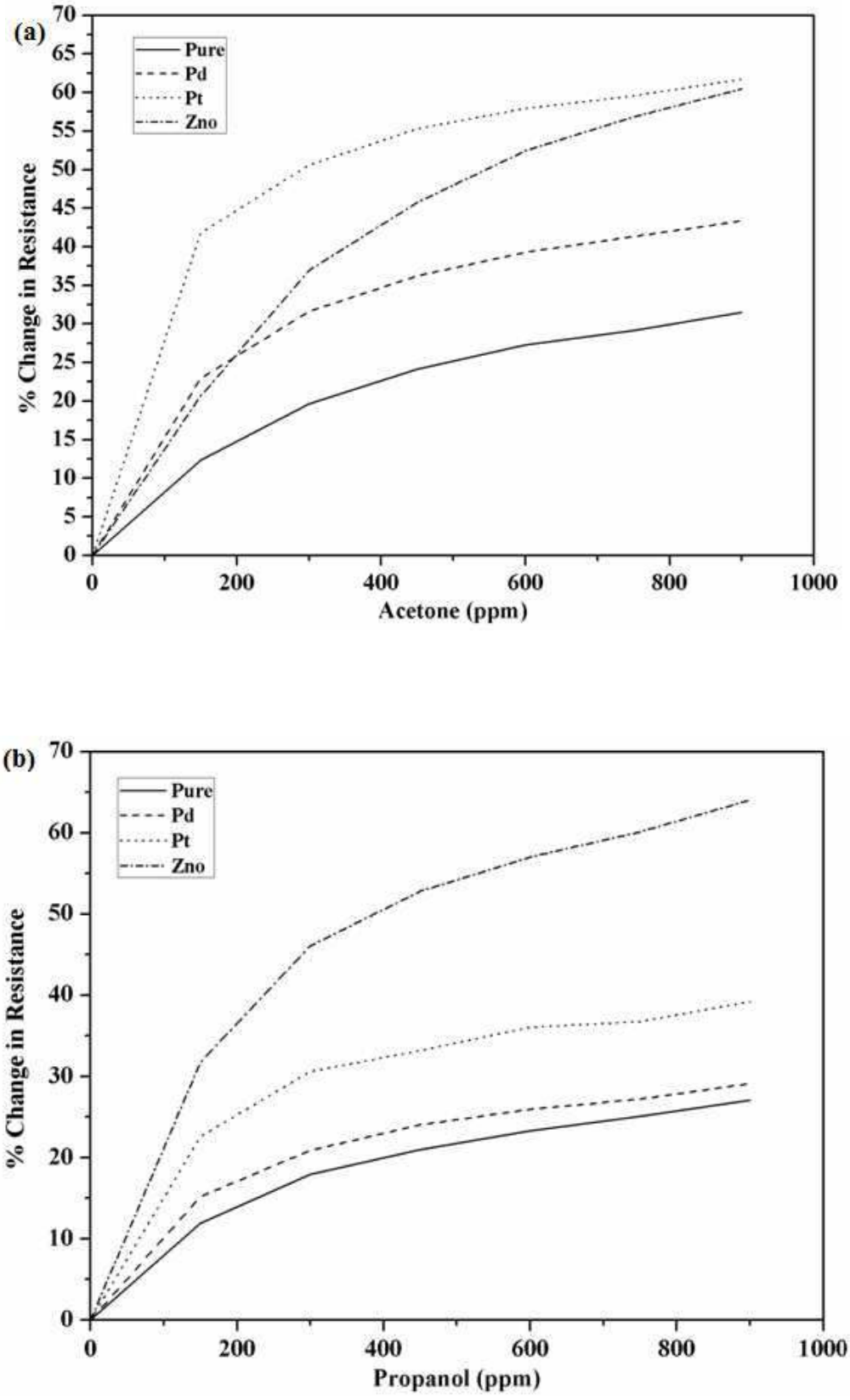


Fig. 5.4 Response of sensor array for (a) Acetone and (b) 2-propanol

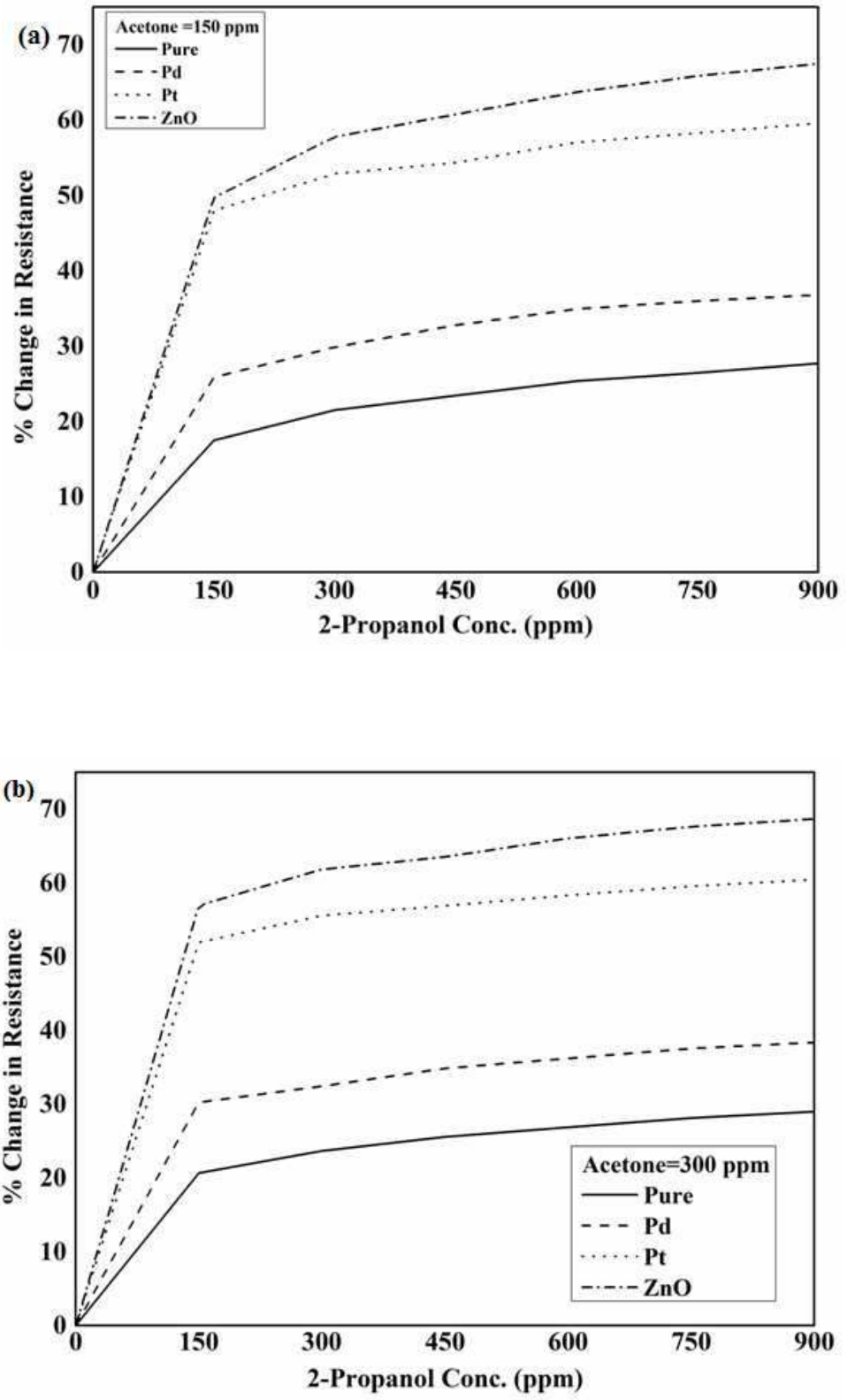
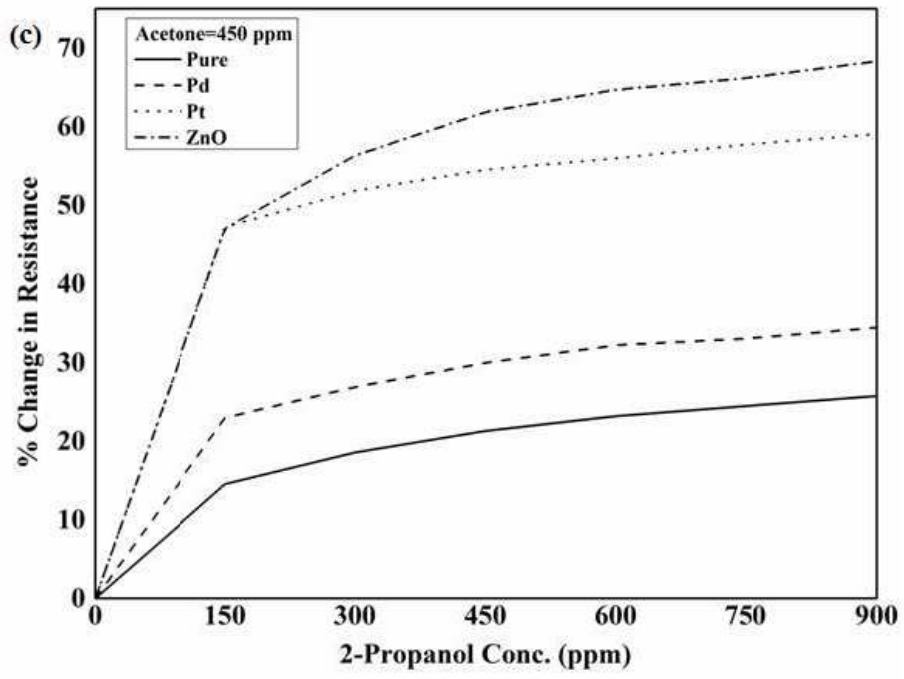


Fig. 5.5 (a)-(b) Response of sensor array for Acetone and 2-propanol mixture in different proportions (*Continued on next page*...)



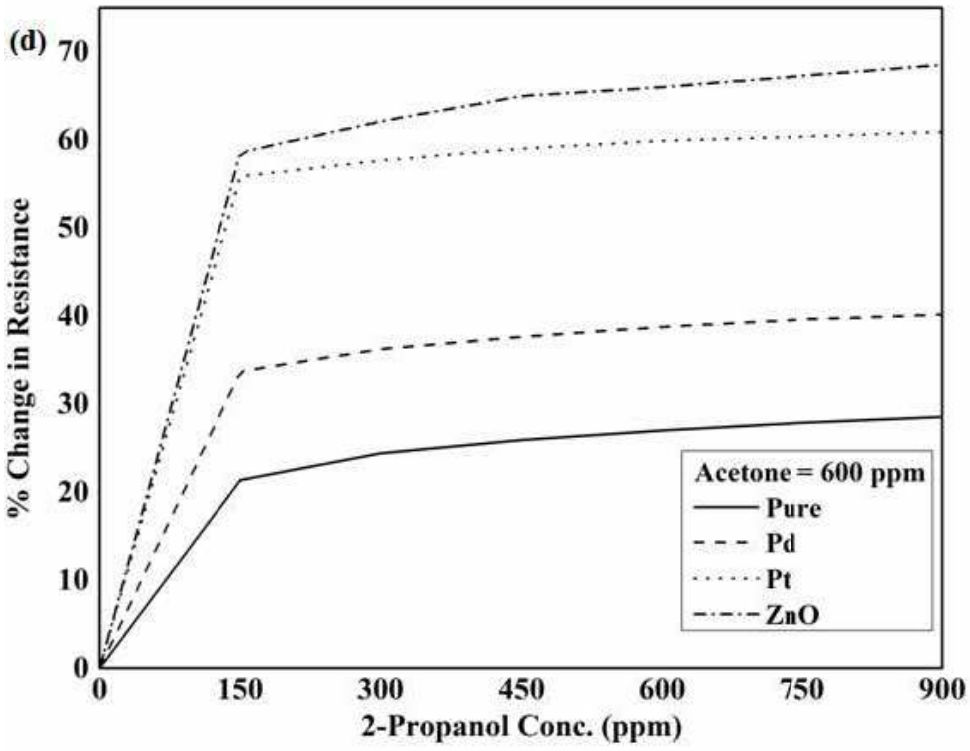
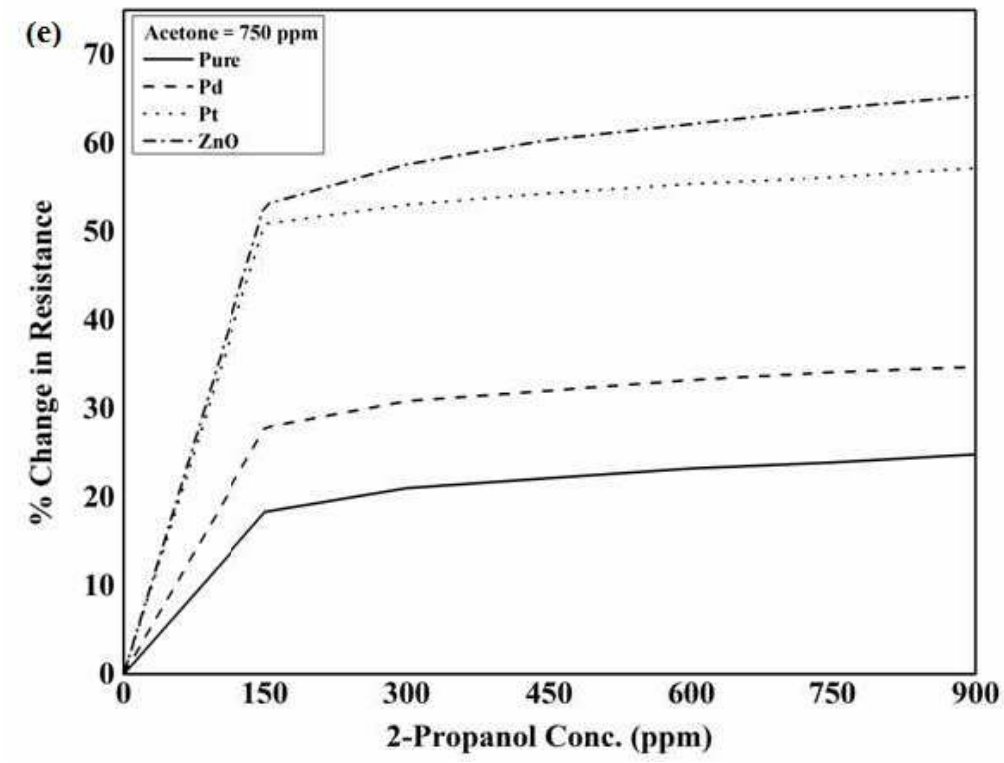


Fig. 5.5 (c)-(d) Response of sensor array for Acetone and 2-propanol mixture in different proportions *(Continued on next page...)*



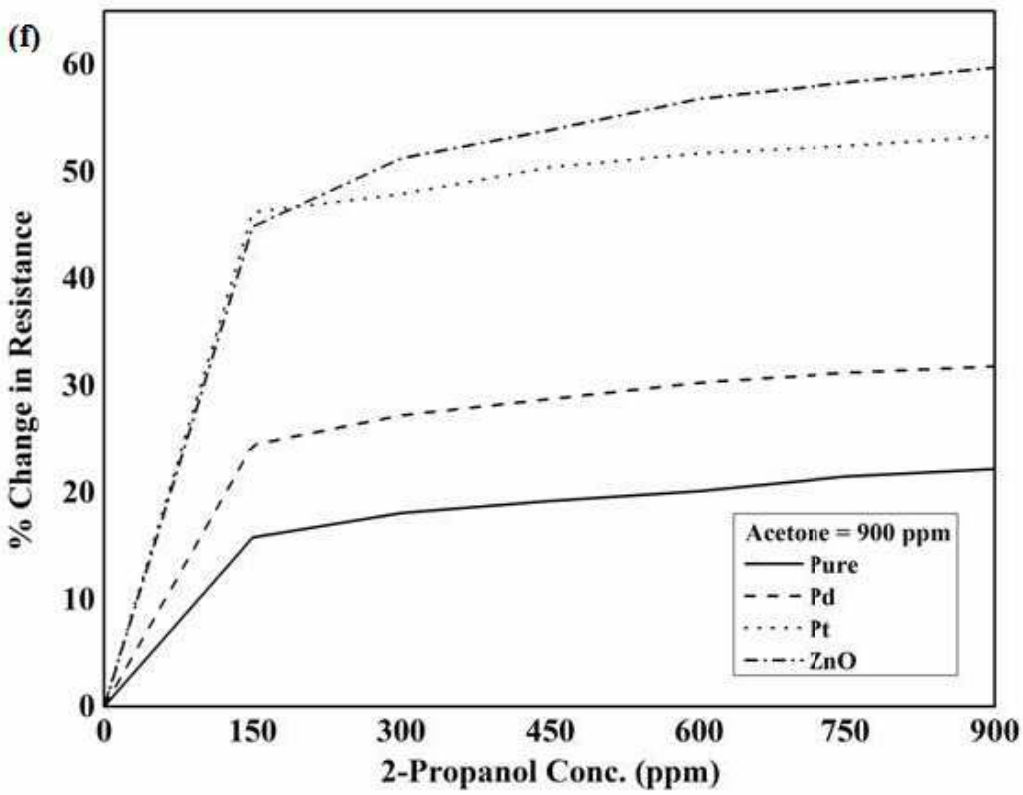


Fig. 5.5 (e)-(f) Response of sensor array for Acetone and 2-propanol mixture in different proportions

Principal component analysis (PCA) has been applied on the raw dataset for data preprocessing and dimensionality reduction purpose. As described earlier, the PCA uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. It is also used for dimensionality reduction since first few principal components generally corresponds to most of the variance of the data [Gardner (1991)].

In the present work, PCA is applied on raw dataset of responses of gases/odors (containing individual as well as mixture forms) to seek improvement in cluster separation between different classes, and for dimensionality reduction. PCA was applied to the combined raw dataset containing individual as well as mixture of selected VOCs in different proportions. It was found that the first two principal components were covering almost 98.6% of the variance. Thus, only first two principal components of the PCA transformed data were used for data representation and for further classification and quantification purpose. The 2-D scatter plot of first two principal components (*PC*1 vs. *PC*2) is shown in Fig. 5.7.

Fig. 5.7 reveals a lot of information regarding the mixture of the two gases/odors. Due to large amount of overlapping among different classes, the raw data was unable to provide any relevant class information for individual and/or mixture of gases/odors as shown in Fig. 5.6.

The PCA transformation of raw data (as shown in Fig. 5.7) shows better and clear separation among different classes of VOCs as well as for their mixtures. Moreover, a kind of pattern has been observed by visual inspection of the PCA transformation of mixture of gases/odor (VOCs) data with respect to concentration of each constituent in the mixture. Individual VOCs are quite separable as is evident from the Acetone data on upper side and 2-propanol data at lower side. Further, in the mixture form, 2-propanol is more dominating as one can see that mixture data is more concentrated towards 2-propanol side rather than Acetone side. This may be due to higher molecular weight effect of 2-propanol ($CH_3CHOHCH_3$) than Acetone (CH_3COCH_3).

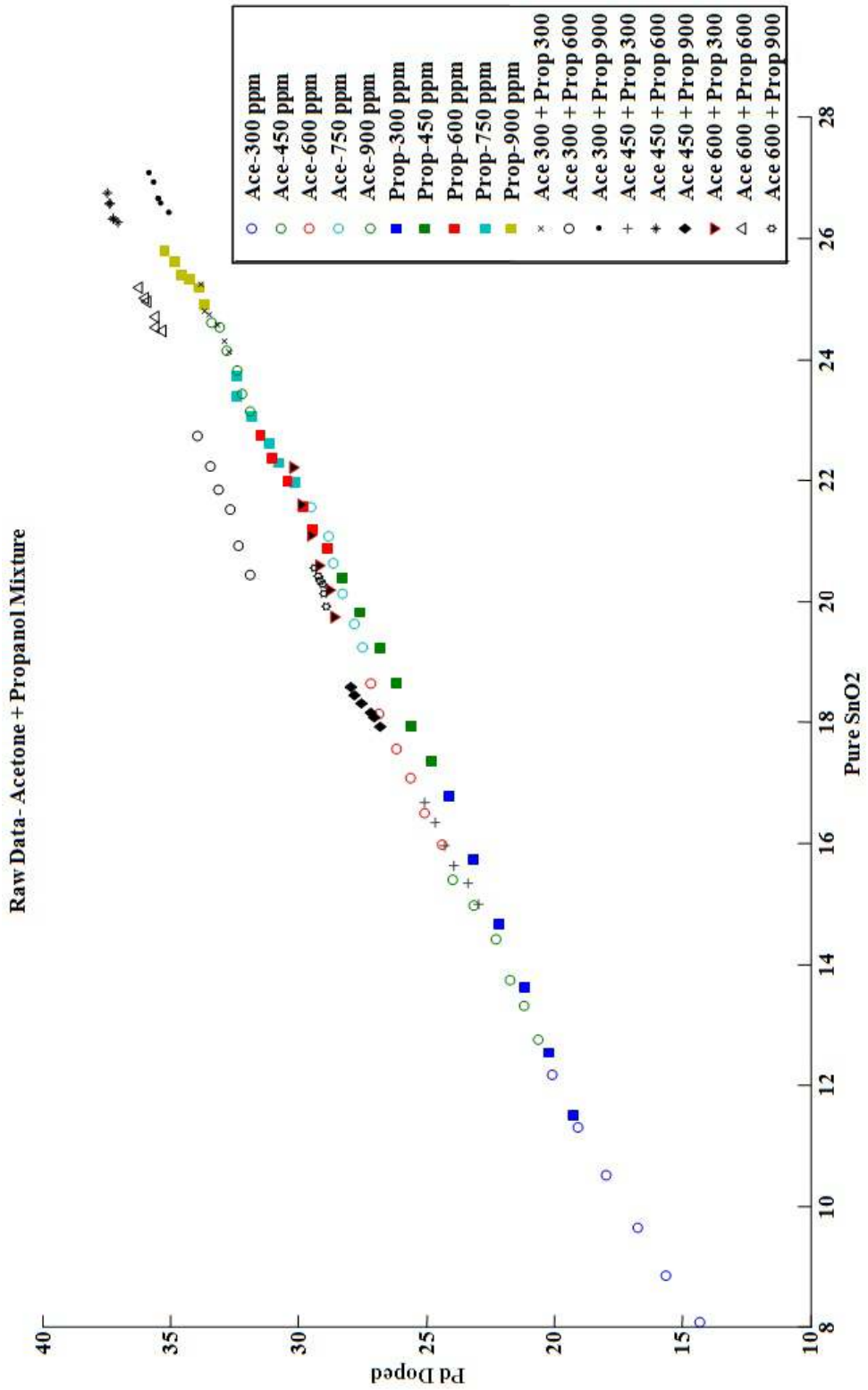


Fig. 5.6 The 2-D scatter plot of raw data considering two of the four sensors data at a time

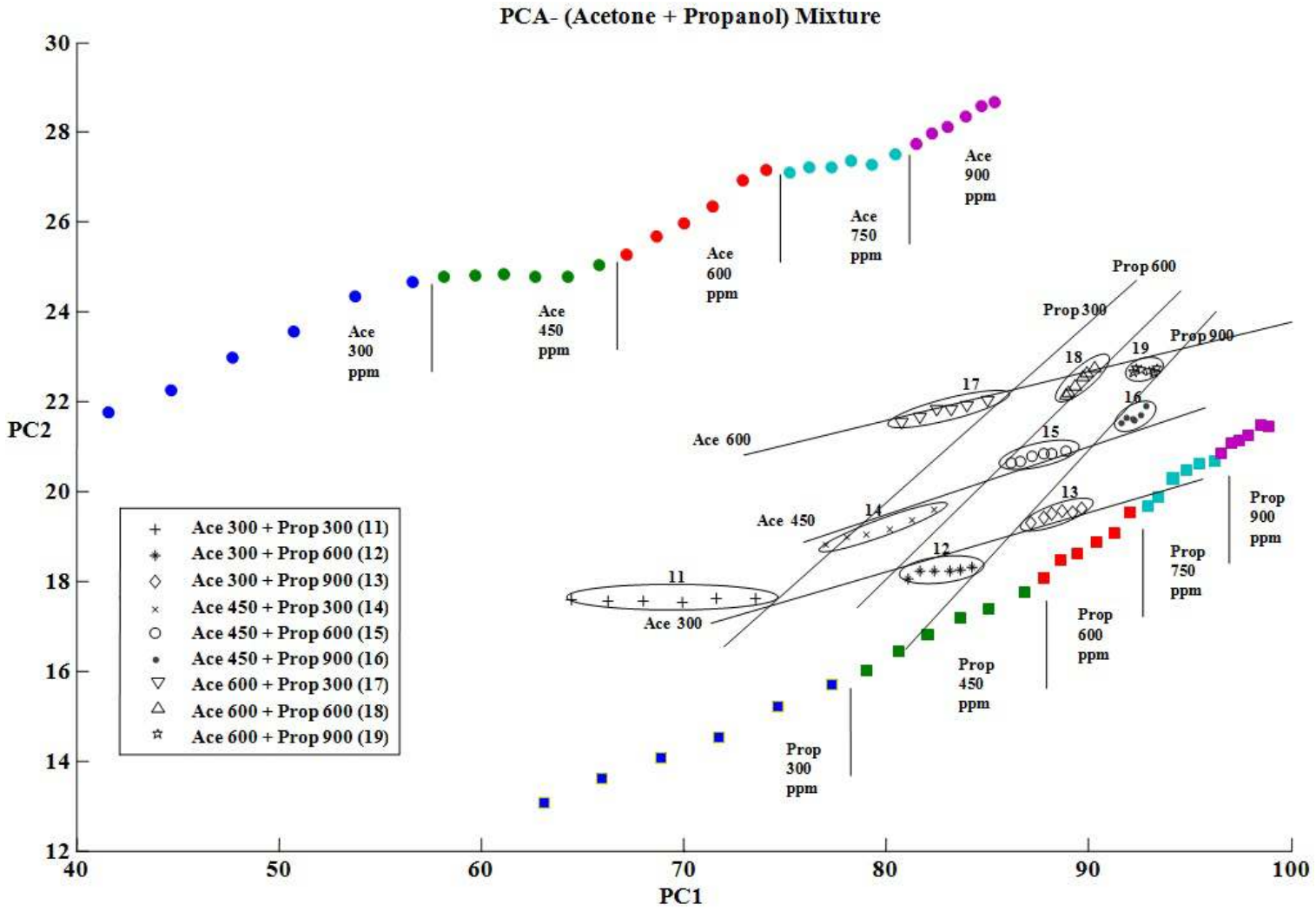


Fig. 5.7 The 2-D scatter plot of first two principal components obtained after applying the PCA to the raw dataset.

Also, as the concentration of a particular constituent is increased, keeping the concentration of other constant, the pattern is shifted towards that constituent side whose concentration is increased. These results are quite obvious as increase in the concentration of particular constituent will add its effect more in the mixture, and hence should exhibit enhanced properties of that constituent in the mixture *i. e.* principal of superposition of gases/odors' concentration seems to be followed but not in an exact manner as reported in [Clifford and Tuma (1982); Nayak *et al.* (1994)].

The classification task in the gating network (G_N) for the raw data as well as PCA data were performed using multilayer perceptron neural classifier (with back propagation algorithm) and support vector machine (SVM) classifier. For quantification, expert network (E_N) was designed which consisted of multioutput SVRs (*M-SVR*s) as explained in the next sections.

5.3 Classification and Quantification Methods

As explained in the Fig. 5.1, the gating network was designed to identify whether the given input data belong to single gas/odor (VOC) or their mixture. Gating network was first designed using back propagation based neural network. Another option was the support vector machine. The network which provided more accuracy was finally adopted for the gating network.

5.3.1 Back Propagation Neural Network

Multilayer perceptron (MLP) with back propagation (BP) algorithms is one of the basic neural network techniques which are widely used for pattern recognition problems. This algorithm was used in previous chapters with different training functions. The popular training algorithm called gradient descent with momentum back propagation (termed as *traingdm* function in MATLAB) which was used in Chapter-4, has been adopted here also. As, mentioned in the previous chapter, the gradient descent with momentum back propagation (GDMBP) is considered as standard back propagation algorithm and is widely used for multi layered artificial neural network because of its computational simplicity, ease of implementation and good generality [Haykin (2008)]. The basic details of the algorithm have been already explained in the previous chapter.

A back propagation algorithm based MLP network was designed (as shown in Fig. 5.8) which consisted of 4 input nodes corresponding to four sensor outputs and 3

outputs (S_1, S_2 and M) where S_1 stands for Acetone alone, S_2 for 2-propanol alone and M stands for their mixture. The numbers of input nodes were 2 for PCA preprocessed data.

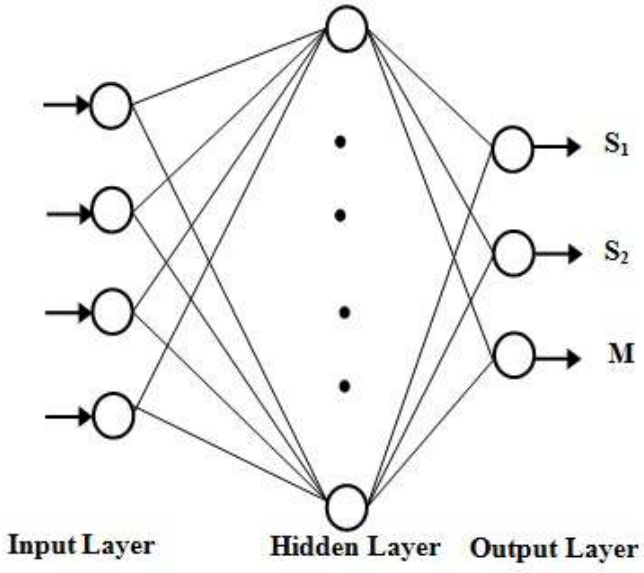


Fig. 5.8 The BPNN architecture for classification of individual and binary mixture of VOCs

Network was tuned for number of hidden layers to obtain best classification accuracy. It was found that while dealing with raw data, a single hidden layer with 7 neurons was needed for best accuracy. For PCA data, single hidden layer having 4 neurons was required for best accuracy, with relatively less number of epochs as compared to the raw data, as shown in Table 5.1. Other BPNN algorithms viz. Levenberg-Marquardt algorithm and one-step secant backpropagation etc. were also tried but GDMBP provided best results for the present data.

5.3.2 Support Vector Machines (SVM) [Khalaf (2009); Chang and Lin (2001); Vapnik (1998)]

In the present study, SVM is used for multi-class problem. The SVMs are basically generalized linear classifiers which have the ability to minimize the empirical classification errors and maximize the geometric margin with reduced number of non-zero parameters. SVMs do not have local minima problem which makes them better than other classifiers like back propagation algorithms. SVMs can provide good generalization performance on pattern classification problems.

In SVMs, the non-linear separable input pattern can be mapped by using appropriate kernel function by higher dimensional transformation space called feature space. A linear separation in the feature space corresponds to non-linear separation in the original input space [Khalaf (2009)]. Kernel being a special class of function allows the inner product to be calculated directly in the feature space, without explicitly applying the mapping [Chang and Lin (2001); Vapnik (1998)]. Different kernel functions can be used depending on the nature of the dataset. In the present work, the *linear*, *polynomial* and *radial basis* kernel functions have been tried and it has been found that the *linear* kernel function provides the best performance for the present data.

Two popular types of SVMs have evolved in the recent past viz. *C*-SVM and *nu*-SVM. In the present work, *nu*-SVM has been used for classification. *nu*-SVM has been chosen because its tunable parameter *nu* has direct interpretation which lies between 0 and 1 whereas in *C*-SVM, parameter *C* has no direct interpretation and can have any values between 0 and infinity [Khalaf (2009]. Next, the basic formulations of *nu*-SVM is discussed which is utilized in the present experiment for classification and quantification of the VOCs and their mixtures.

(i) nu-SVM [Khalaf (2009); Vapnik (1998)]

nu-SVM is another version of C-SVM. This type of model minimizes the error function

$$\frac{1}{2} w^T w - \nu\rho + \frac{1}{N}\sum_{i=1}^{N} \xi_i \tag{5.1}$$

subject to the constraints:

$$y_i\left(w^T \phi(x_i) + b\right) \geq \rho - \xi_i,\ \xi_i \geq 0 \text{ , i=1, 2....N and } \rho \geq 0 \ ; \tag{5.2}$$

for a given set of training patterns (x_1, y_1……. x_n, y_n) where $x_i \in R^N$, $y_i \in R$, $i = 1, 2, \ldots, N$, where, x_i is the input and y_i the corresponding output, suppose $\phi(x)$ is a nonlinear mapping from the input space to the feature space and w is a vector of weight coefficients, b is a bias constant and ξ_i, slack variables

Here ν (*nu*) a new parameter which controls the number of support vectors and errors with value $0 \leq \nu \leq 1$. The training vectors x_i are mapped into higher dimension space by the function $\phi(x)$. This formulation is different from the original C-SVM [Vapnik (1998)] as shown in (5.3)

$$\frac{1}{2} w^T w + C\sum_{i}^{N} \xi_i \tag{5.3}$$

subject to the constraints:

$$y_i\left(w^T \phi(x_i) + b\right) \geq 1 - \xi_i \text{ and } \xi_i \geq 0,\ i=1, 2.....N \tag{5.4}$$

In (5.3), a parameter *C* is used to penalize slack variables ξ_i.

The difference between *C*-SVM and *nu* -SVM is of parameters *C* and *nu (v)*. The range of *C* is from zero to infinity while *nu (v)* lies between 0 and 1. Since parameter *C* can take any positive values and has no direct interpretation therefore it is hard to choose *C* correctly. The parameter *v* is an upper bound on the fraction of margin errors and a lower bound of the fraction of support vectors. A good property of parameter *nu* is that it is related to the ratio of support vectors and the ratio of training errors. *nu* controls the number of support vectors, training error and also it has a direct interpretation.

For the raw data, a *nu*-SVM was designed having 4 input nodes corresponding to four sensor outputs and 3 outputs (S_1, S_2 and M) where S_1 stands for Acetone alone, S_2 for 2-propanol alone and M stands for their mixture. Again, the input nodes were 2 for PCA preprocessed data. The raw data and its PCA processed versions were fed to the *nu*-SVM classifiers. The performance of *nu*-SVM as a gating network was found to be better than BPNN which is obvious from the results shown in Table 5.2.

5.3.3 Multioutput Support Vector Regression (*M-SVR*) [Perez-Cruz *et al.* (2002); Sanchez-Fernandez *et al.* (2004); Tuia *et al.* (2011)]

Multioutput Support vector regression (*M-SVR*) [Perez-Cruz *et al.* (2002)] is a generalization of SVRs to solve the problem of regression estimation for multiple variables. The *M-SVR* method used here is based on the previously proposed works [Perez-Cruz *et al.* (2002); Sanchez-Fernandez *et al.* (2004); Tuia *et al.* (2011)]. In the present work, this method has been used for quantification of individual components in a binary gas/odor mixture. The use of multidimensional regression tool helps in exploiting the dependencies between variables.

(i) Standard support vector regression (SVR) formulation [Sanchez-Fernandez et al. (2004) et al. (2004)]

The standard SVR solves the one-dimensional regression problem by finding regressor *w* and *b* that minimizes

$$\|w\|^2/_2 + C\sum_{i=1}^{l} L_v\big(y_i - (\varphi(x_i)^T w + b)\big), \tag{5.5}$$

where, $x\epsilon\mathbb{R}^d$ are input vector, $y\epsilon\mathbb{R}$ is observable numeric output from a given set of independent and identically distributed samples $\{(x_i, y_i)\}_{i=1}^{l}$ and $\varphi(.)$ is non-linear transformation to a higher dimensional Hilbert space (H) known as feature space. The SVR can be solved using only inner products between $\varphi(\cdot)$, not needing to know the nonlinear mapping. To solve the SVR, one needs to specify the kernel function $k(x_i, x_j) = \varphi(x_i)^T\varphi(x_j)$ which fulfills the Mercer's theorm. $L_v(.)$ is known as Vapnik ε-insensitive loss function which equals to 0 for $|y_i - (\varphi(x_i)^T w + b)| < \varepsilon$ and equal to $|y_i - (\varphi(x_i)^T w + b)| - \varepsilon$ for $|y_i - (\varphi(x_i)^T w + b) \geq|\varepsilon$. The solution ($w$ and b) is formed by a linear combination of the training samples in the transformed space with an absolute error equal to or greater than ε (*i.e.* the support vectors) [Sanchez-Fernandez *et al.* (2004)].

(ii) M-SVR formulation [Sanchez-Fernandez et al. (2004)]

To solve the multiregression problem where the observable outputs is a vector with m variables to be predicted i.e. $y\epsilon\mathbb{R}^m$, then one has to find regressor $w_i$and b_j (j= 1….m) for every output. Then one-dimensional (1-D) SVR can be generalize to solve multi-dimensional (multi-D) case to minimize

$$L_P(w, b) = \frac{1}{2}\sum_{j=1}^{m}\left\|w^j\right\|^2 + C\sum_{i=1}^{l} L(u_i) \tag{5.6}$$

with respect to W and b where $u_i = \|e_i\| = \sqrt{e_i^T e_i}$, $e_i = y_i^T - \varphi(x_i)^T W - b^T$, $W = [w^1, \ldots\ldots, w^m]$, $b = [b^1, \ldots\ldots, b^m]^T$.

For simple SVR, the Vapnik ε-insensitive loss function is based on L_1 norm which requires accounting each dimension independently for extending it to multiple dimensions growing the complexity linearly. For multiple dimensions, one can use L_2-based norm where all dimensions can be considered into a unique restriction yielding a single support vector for all dimensions [*Sanchez-Fernandez et al. (2004)*]. So, in *M-SVR*, L_2 based norm is used where all dimensions can be considered in a unique restriction.

$$L_2 = \begin{cases} 0, & u < \varepsilon \\ u^2 - 2u\varepsilon + \varepsilon^2, & u \geq \varepsilon \end{cases} \tag{5.7}$$

which is a differentiable version of the loss function proposed in [Perez-Cruz *et al.* (2002)]. For ε = 0, this problem reduces to a kernel ridge regression (KRR) for each

component, but for a nonzero ε value, the solution takes into account all outputs to construct each individual regressor. The cross-output relations are exploited which in turns leads to possibly more accurate predictions. Here it should be noted that *M-SVR* returns a multidimensional and sparse solution, thus solving the main issues of the SVR and the KRR when dealing with multiple outputs; the SVR cannot handle multiple outputs, whereas the KRR is not sparse and thus accounts for output relations in a dense way.

In the present approach, a quasi-Newton approach has been used, where each of the iteration has at most the same computational complexity as a least-squares procedure for each component. This iterative reweighted least squares (IRWLS) procedure [Perez-Cruz *et al.* (2002)] is a weighted least-squares problem and the number of iterations needed to obtain the final result is small, making the procedure only slightly more computationally demanding than least-squares regression for each component. Thus *M-SVR* is IRWLS-based approach for the regression of multiple variables.

The resolution of *M-SVR* and more details can be found in [Perez-Cruz *et al.* (2002); Sanchez-Fernandez *et al.* (2004); Tuia *et al.* (2011)]. Thus, in *M-SVR* cross-output relations are exploited leading to possibly more accurate predictions. It provides multidimensional solution by solving the main issue of SVR that it cannot handle the multiple outputs.

Further, the formulation and resolution of *M-SVR* can be referred in [Perez-Cruz *et al.* (2002); Sanchez-Fernandez *et al.* (2004); Tuia *et al.* (2011)]. In this work, the *M-SVR* was used to quantify the one-dimensional output (individual VOC concentration) as well as two dimensional outputs (concentration of each VOC in the mixture). The results obtained with this method have been discussed in the subsequent sections.

5.3.4 Correlation Coefficient

The correlation coefficient (*CC*) has been considered as a measure of estimation of accuracy [Penza *et al.* (2002)]. The correlation coefficient is a number between 0.0 and 1.0 if there is no relationship between the predicted and the true concentration values the *CC* is 0.0 or very low. As the strength of the relationship between predicted values and the true values increases, so does the correlation coefficient. A perfect fit gives the *CC* value of 1.

Thus, closer the *CC* value to 1.0 the better is the regressor [Cohen *et al.* (2002)]. In the present work, the correlation coefficient value (*CC*) has been considered as a measure of how well the concentration value predicted by the quantification network (regressor) matches with the true (actual) value of the concentration.

The correlation coefficient is calculated as shown in equation (5.8)

$$CC = \frac{\sum_{i=1}^{n} X_i Y_i - \frac{\sum_{i=1}^{n} X_i \sum_{1=1}^{n} Y_i}{n}}{\sqrt{\left(\sum_{1=1}^{n} X_i^2 - \frac{\left(\sum_{i=1}^{n} X_i\right)^2}{n}\right)\left(\sum_{1=1}^{n} Y_i^2 - \frac{\left(\sum_{i=1}^{n} Y_i\right)^2}{n}\right)}} \tag{5.8}$$

where, *X* are the true values and *Y* are the predicted values and n is the number of data points.

5.4 Results and Discussion

The dataset was prepared by collecting the steady state responses of individual VOC and their mixture forms for its use in the classification and quantification task, as explained in section 5.2 previously. The raw data were also preprocessed using PCA technique and dimensionality reduction was done from 4 to 2 as first two principal components were covering almost 98.6% of the variance. The raw data and its PCA processed versions were divided into the training sest and the test sets for *k*-fold cross validation method [Rodriguez *et al.* (2010)]. The best performance of with *k*-fold cross validation was found with *k*=6 in gating network. Each time 125 out 150 samples were used for training purpose and remaining 25 unseen were used for test purpose. This step was followed for all six folds of the dataset. BPNN classifier and *nu*-SVM classifier were trained and tested the raw as well as PCA data to compare the classification performance of gating network. MATLAB (2009) and WEKA (2009) tools were used for all the analysis.

Firstly, the raw data and PCA data were trained and tested with the back propagation neural network. The *tansig* was used as the threshold function. The average classification accuracy for six folds obtained was 89.5% with value of *learning rate* = 0.4 and *momentum*=0.2, with single hidden layer having 6 neurons and 1136 epochs. With PCA preprocessed data, the average classification accuracy for six folds

obtained was 95.6% with value of *learning rate* =0.2 and *momentum*=0.3, with single hidden layer having 4 neurons and relatively less number of epochs (epochs=938). Table 5.1 shows the results obtained with GDMBP algorithm with raw data and PCA data.

Next, *nu*-SVM was tried for gating network to have a comparative study of its classification accuracy over BPNN for the present dataset and to seek any improvement in the classification results. *nu*-SVM was trained with three different kernel functions viz. l*inear*, *polynomial* and r*adial basis*. With *nu*-SVM the maximum average accuracy with raw data was 94.7% with value of *nu* = 0.5 and *linear* kernel function while with PCA data the average accuracy was 100% with value of *nu* = 0.5 with same *linear* kernel function. The results obtained with *nu*-SVM are shown in Table 5.2 with different kernel functions and *nu* values. It is evident from the results shown in Table 5.1 and Table 5.2 that the *nu*-SVM has given more accurate results as compared to back propagation neural network. The *nu*-SVM in gating network now serves as switch, which after taking the input data, decides whether it belongs to Acetone alone, 2-propanol alone or their mixture, activating corresponding one of the three networks in the next stage which is the quantification stage.

After gas/odor (VOC) is identified as single Acetone, 2-propanol or their mixture, the next task is to quantify the corresponding gas/odor (VOC). For the quantification task, multioutput support vector regression (*M-SVR*) has been used with one-dimensional output (for single VOC) as well as with two-dimensional outputs (for individual VOC in the mixture) in the expert network (E_N). As shown in Fig. 5.1, $M\text{-}SVR_1$ is a regression model trained for Acetone only and $M\text{-}SVR_2$ is a regression model trained for 2-propanol only, both having single output. $M\text{-}SVR_3$ is a regression model trained for the mixture of both Acetone and 2-propanol and has two outputs. $M\text{-}SVR_1$ and $M\text{-}SVR_2$ provided best results with 6-fold cross validation while $M\text{-}SVR_3$ provided best results with 8-fold cross validation.

Fig. 5.9 shows the predicted against true concentration of Acetone and 2-propanol in single form and individual VOC mixture form. The scores of the first two principal components of PCA data has been used as input to each quantification network. For, individual Acetone and 2-propanol, good correlation coefficients (*CC*) 0.9983 and 0.9871, respectively were obtained. Also good 0.9828 and 0.9764 correlation coefficients for the predicted versus real concentration of Acetone and 2-propanol, respectively as individual VOCs in the binary mixture have been obtained.

Table 5.3 shows the results of quantification of Acetone, 2-propanol and their binary mixture.

Table 5.1 Classification results (gating network) obtained with BPNN

Data type	Learning rate (η)	Momentum (α)	Accuracy (A_C) (%)	epochs	*NH*
Raw Data	0.4	0.2	89.5	1136	7
PCA Data	0.2	0.3	95.6	938	4

NH= Number of neurons in a single hidden layer.

Table 5.2 Classification results (gating network) obtained with *nu*-SVM

Data type	nu	Accuracy (A_C) (%)	Kernel
Raw Data	0.5	94.7	Linear
		93.8	Polynomial
		91.2	Radial basis
PCA Data	0.5	100	Linear
		96.5	Polynomial
		92.1	Radial basis

Table 5.3 Results of the quantification of Acetone, 2-propanol and their binary mixture

Data type	Correlation Coefficient (*CC*)	Slope (ppm/ppm)	Intercept (ppm)
Acetone (single)	0.9983	0.9817	10.024
2-propanol (single)	0.9871	0.9768	15.118
Acetone (in mixture)	0.9828	0.9469	17.2640.
2-propanol (in mixture)	0.9764	0.9088	27.972

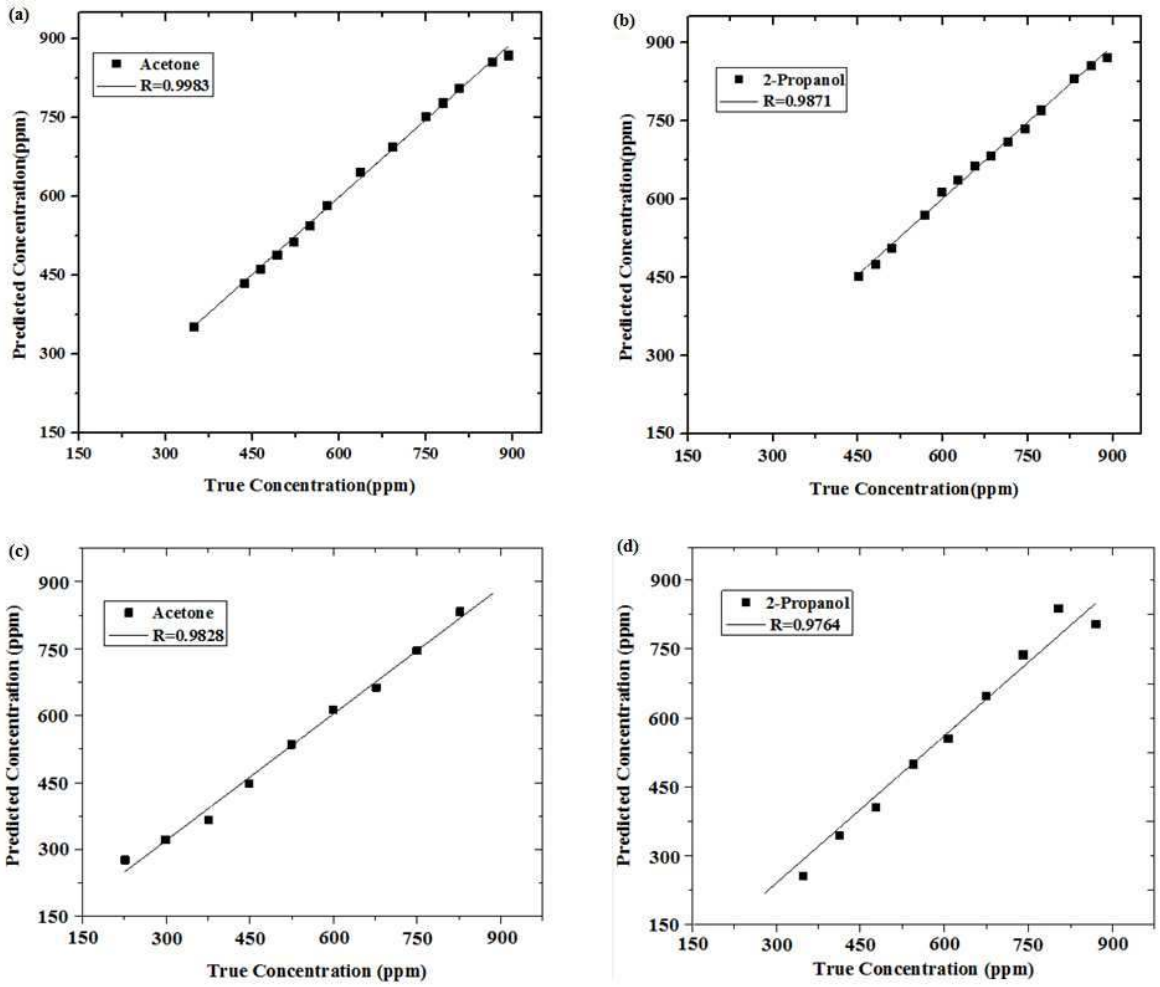


Fig. 5.9 Predicted against true concentration of (a) Acetone alone, (b) 2-propanol alone (c) Acetone in mixture, (d) 2-propanol in mixture

5.5 Conclusion

Classification and quantification of gases/odors/VOCs in the mixture form is more challenging as compared to single gas/odor/VOC. The steady state responses of the mixture of gases/odors along with the suitable pattern recognition technique can be promising to predict the class and concentration of gases/odors/VOCs present in a mixture. In the present work, two volatile organic compounds viz. Acetone and 2-propanol were used in the gaseous form to collect the steady state responses in the individual as well as in the mixture forms with thick film sensor array. The PCA preprocessing was applied on the raw data which provided better classification and quantification results.

Quantification of the VOCs in single as well as in mixture form has been tackled with two networks. First was a gating network which acted as a switch to identify whether given input data belongs to Acetone only, 2-propanol only or their mixture. SVM-classifier performed better as a gating network and provided 100% accuracy. Second network was a quantification network composed of multioutput support vector regressors (*M-SVR*) used as regression models. Good 0.9828 and 0.9764 correlation coefficients for the predicted versus real concentration of Acetone and 2-propanol respectively as individual VOCs in the binary mixture have been obtained with *M-SVR* regression model. Thus, the present work has shown promising results for binary gases/odors/VOCs mixture analysis.