We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Neuro-Fuzzy Classifiers/Quantifiers for E-Nose Applications

Ravi Kumar
Thapar University, Patiala
India

1. Introduction

Smell is still a mystery to scientists in somehow, which cannot be studied with ease in vertebrates. Another problem is that the sense of smell is poorly developed in human beings in comparison with the same in many vertebrates (Menini *et al.*, 2004). This makes realization of an artificial olfactory system a challenging task. An artificial olfactory system (commonly known as E-nose) provides a low cost alternative to identification, quantification and characterization of odours. The traditional methods of characterization and quantification of odours generally involve the use of a trained panel of human experts. The use of human panel is sensitive to individual variability, adaptation (tendency to become less sensitive after prolonged exposure), mental state, fatigue, subjectivity, infections and exposure to hazardous compounds (Nagle *et al.*, 1998). Therefore, it is necessary to have a low cost and compact device to perform real-time analysis. It is thus natural for researchers to envisage a system, which is biologically inspired and modelled on the lines of an olfactory system. The increased understanding of the biological phenomenon of olfaction has motivated scientists to achieve artificial olfaction. Rapid strides made in the field of material science and fabrication technology has paved the way for manufacture of a large variety of micro-sensors, of which a large percentage is chemical sensors.

An E-nose uses multiple sensors in the form of an array. In an array of sensors each sensor responds broadly to a range or class of gases rather than a specific one. This characteristic of a sensor array is similar to a human nose, which is also partially sensitive to several odorants. The partial sensitivity of the sensor array can be exploited for characterization and quantization of gases/odours by making use of effective signal processing and pattern recognition. In an electronic nose, the odorants produce changes in physical/chemical properties. A sensor array converts the chemical inputs into electrical signals which are further processed by utilizing an electronic circuit, providing an analogue signal to be amplified, pre-processed and/or digitised prior to being fed into a pattern recognition system (Shurmer *et al.*, 1990; Nakamoto *et al.*, 1990). Basic stages of an artificial olfactory system are shown in Fig. 1.

After coming in contact with the odorants, the sensors experience a change in electrical properties. Each sensor is sensitive to all the odorant molecules in their specific way. Most

electronic noses use sensor arrays that react to gases/odours on contact: the adsorption of gases/odours on the sensor surface causes a change in physical properties of the sensor (Sarro, 1992). A specific response is recorded by the electronic interface transforming the signal into a digital value. Recorded data are then analyzed using computational techniques (Osuna *et al.*, 2002). An artificial olfaction system can be fabricated using standard micro-electronic techniques for on-chip integration. An E-nose employs an array of chemical sensors to achieve an appreciable level of selectivity to different gases/odours. Metal oxide based sensors fabricated with the thick film technology are the most popular choice for gas/odour sensing. Most of the commercial E-noses employ metal oxide based sensing devices because metal oxides are most suited as gas sensors due to their high sensitivity and their ability to maintain structural integrity in harsh conditions, namely, high temperature (Moseley, 1992). The basic sensing mechanism in metal oxide based sensors involves a change in resistance due to chemisorptions when exposed to odorants/gases.
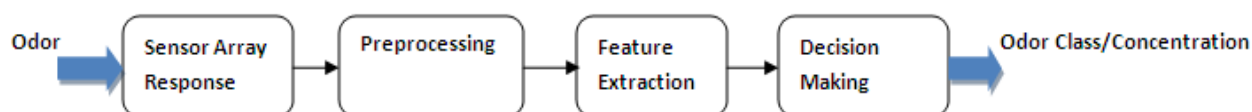

Fig. 1. Artificial olfactory system

1.1 Operating principle of tin oxide gas sensors

It has been found that when a bead of tin oxide is heated in the presence of a combustible contaminant, and the conductance of the bead is measured continuously, it is possible to obtain a measure of the concentration of the contaminant gas (Watson, 1984). This observation can be explained as follows.

By heating a bead of tin oxide in clean air, oxygen can be adsorbed onto the surface layers until equilibrium is achieved for that particular temperature. The measurement of the characteristic conductance of the bead would reveal that it is a function of both the temperature and partial pressure of the oxygen. Significant change in surface conductivity of semiconductors can be brought about by adsorption and subsequent reaction of gases with the adsorbed oxygen. The active material of the sensor is generally SnO_2, which is an n-type semiconductor. When oxygen is adsorbed on to it, it accepts electrons to become O_2^- O^- or O^{2-} (Ikohura, 1981). The adsorption of a reducing gas releases bound electrons and thus increases the conductance of the surface dramatically. For an oxidizing gas converse mechanism operates. Various dopants are used to improve the sensitivity and selectivity of thick film tin oxide gas sensors (Morrison, 1987).

Sensor operating temperature plays a vital role in the development of gas selective sensors. Since, different classes of reducing gases have different reaction rates, sensors operating at different temperatures show a degree of selectivity (Sears *et al.*, 1990). Despite having appreciable sensitivity to a large number of gases/odours, thick film tin oxide sensors have some well known limitations such as, cross sensitivity to a number of compounds and

saturation of sensor response at higher concentration of the odorants. These limitations can be overcome by employing an array of sensors whose responses are analyzed subsequently using appropriate pattern analysis techniques. Typical materials used for fabrication of the sensors are SnO_2, ZnO, Fe_2O_3, and WO_3. These metal oxide films are used with or without dopants like CuO, Pd, Pt and In to enhance their selectivity. Porous and sintered SnO_2 is the most widely used for gas sensors, as it is appreciably sensitive to a large number of gases/odours. SnO_2 sensors are available both for domestic and commercial use. For domestic applications, devices are available for detecting combustible gases such as CO, H_2, alcohols, LPG, and volatile matters from food stuff. For industrial applications detectors are available for gases such as NH_3, H_2, H_2S, CH_4, C_7H_8, C_8H_{10} and hydrocarbons. The following subsection presents a typical experimental set up where a sensor array is exposed to several odorants and the response of each sensor is noted. This particular set up is chosen for illustration because its response pattern is the most challenging from pre-processing and computational point of view. The same data would be used throughout the chapter for demonstrating the efficacy of the computational techniques employed. Published data from other sensor arrays may also be reproduced to explain some of the computational challenges.

1.2 Typical experimental set up for odour sensing

Integrated gas sensor array comprises several sensors as shown in Fig. 2. Sensors are fabricated on one side of an alumina substrate whereas a resistive pattern is fabricated on the other side to achieve uniform heating. A metal oxide paste is prepared, which is applied to the substrate and fired at high temperature by passing it to a furnace so that the paste sticks properly to the substrate. Different dopants (e.g. ZnO, Sb_2O_3 and NiO) are used respectively with the metal oxide paste, resulting in different types of gas sensors with different sensitivity to different odours. The diagram of a tin oxide sensor array pattern is shown in Fig. 2. The fabricated integrated sensor array pattern is then tested under closely controlled environmental conditions using some experimental chamber with a facility (either manual or automated) of injecting test gases as shown in Fig. 3. The experiment is designed for testing 4 types of whiskies, two types of rums, and ethanol (Nayak *et al*. 1992).

Initially, the sensor array is kept in a closed ambient air under energized condition at 10W heater supply for more than 30 minutes to make the sensor resistances stable. At this stage, initial resistance of the sensors is recorded. Then a drop of test alcohol is injected into the chamber and it gets vaporized to gaseous phase before being adsorbed on to the sensor surface. Sensor readings are reported after three minutes as it is found to be optimum time to equilibrate with the sample alcohol. Similarly, another drop of test alcohol is injected and the experiment is repeated for more drops so that the observations are made for up to 12 drops of test alcohols. After experimenting with one of the alcohols/alcoholic beverages, the sensors are recovered in open ambient air at room temperature and on complete recovery; experiments are repeated for other alcohols/alcoholic beverages. Using suitable mathematical manipulations, the concentration of test odorants can be converted to parts per million (ppm).
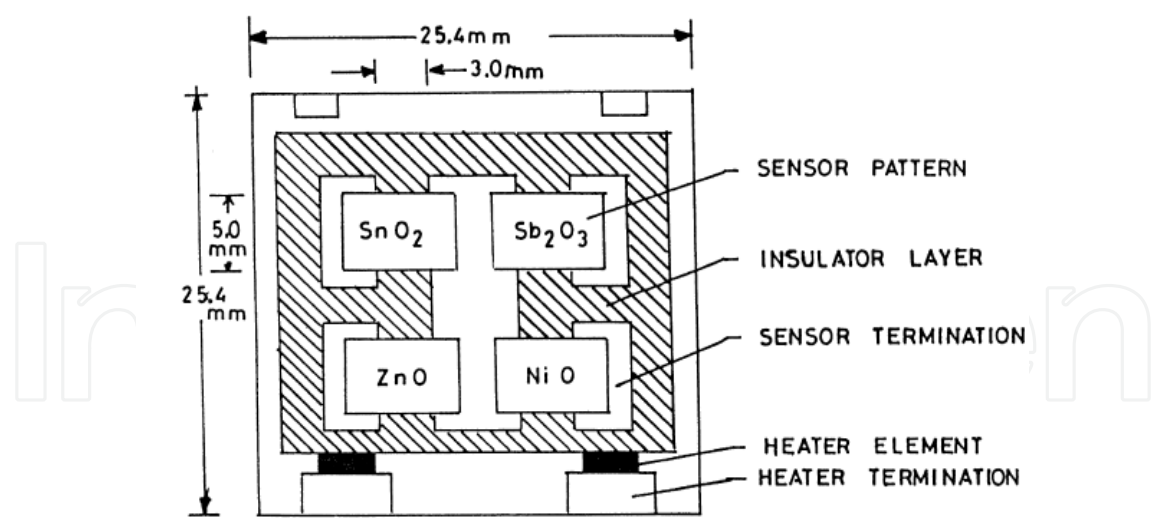
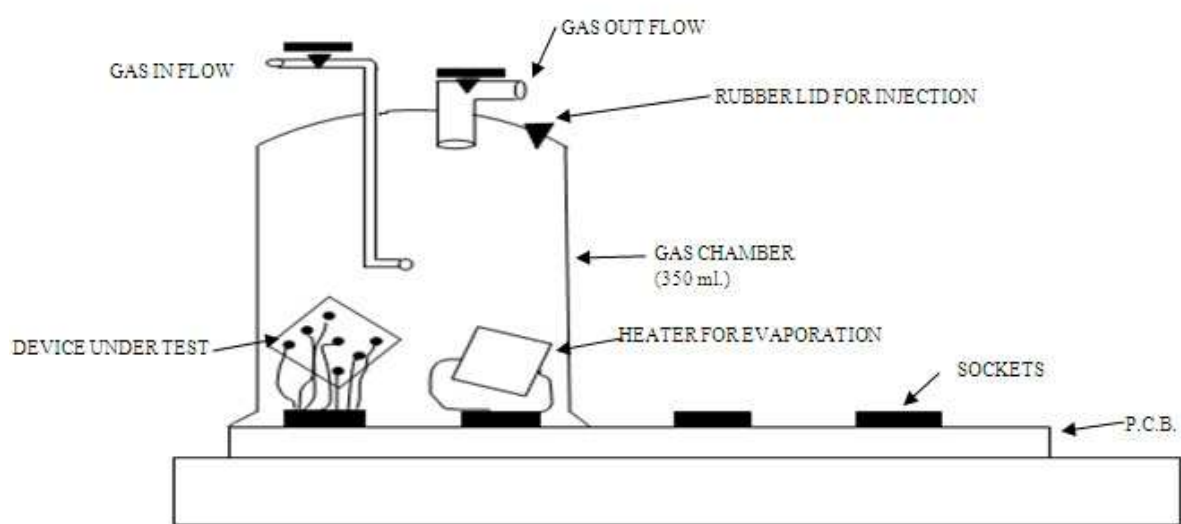Fig. 2. Fabricated integrated sensor array pattern (Nayak *et al.*, 1992)

Fig. 3. Experimental chamber for exposing sensor array to gases/odours

The next step is to obtain the sensor response by calculating the percentage change in resistance of all the sensors for all the odorants injected into the test chamber. This is done to nullify the effect of initial resistances. The percentage change in resistance is calculated using

$$P_{ijd} = \frac{R_{ijo} - R_{ijd}}{R_{ijo}} \leq 100 \qquad (1)$$

where, R_{ijo} is the initial resistance of the ith sensor for zero-th drop of jth odorant and the subscript d denotes a particular drop. The steady-state exposure profiles of the sensor array exposed to different types of alcohols and alcoholic beverages at different concentrations thus obtained is shown in Fig. 4.
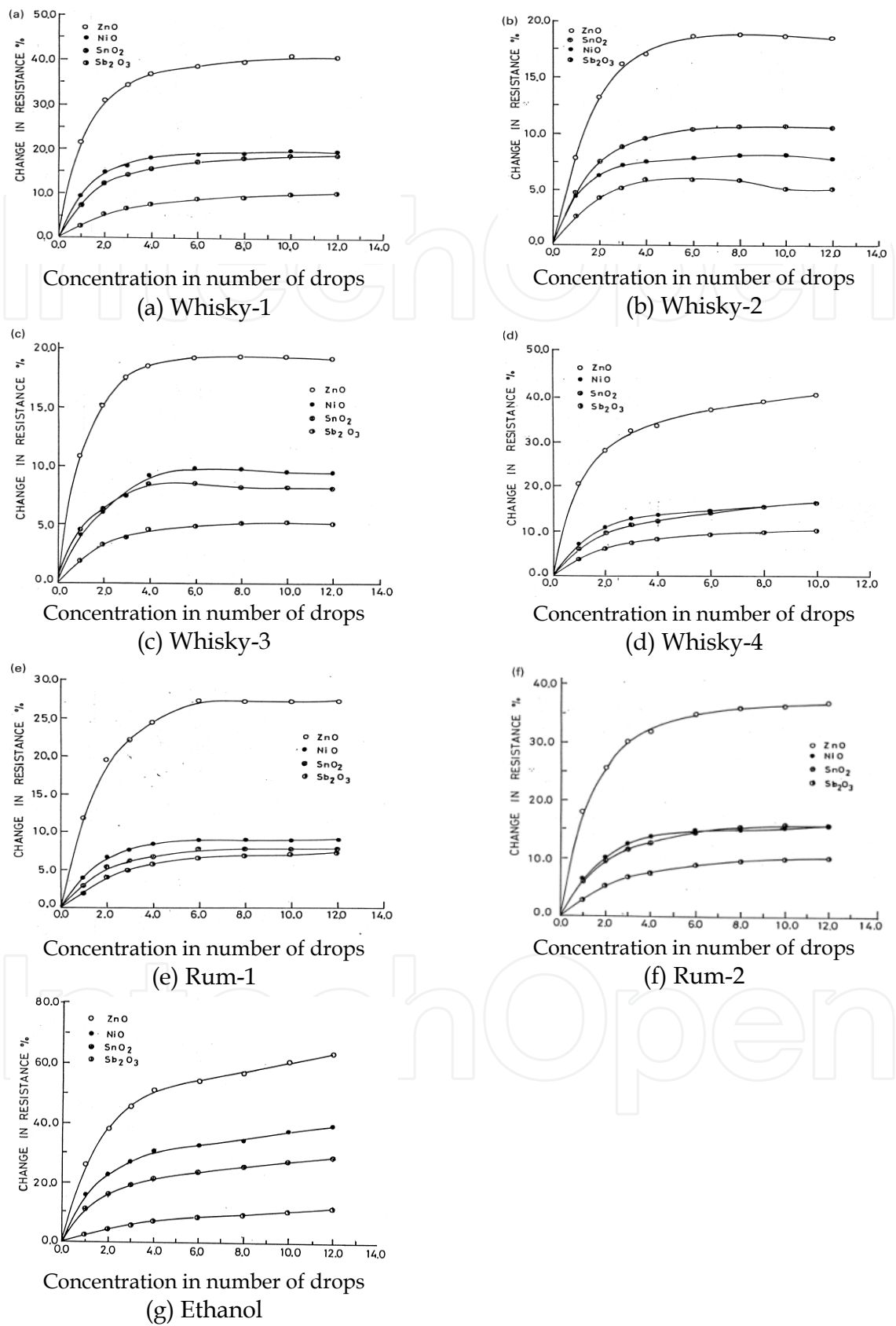
Fig. 4. Steady-state response of sensor array upon exposure to different alcoholic beverages (Nayak *et al.,* 1992)

1.3 Limitations of sensor arrays

In general, sensor arrays suffer from one or more of the following limitations:

1. Overlapping sensitivity to different gases/odours leading to poor selectivity.
2. Saturating tendency of the sensor response at higher concentration of the test gas leading to difficulties in quantification
3. "Drift" in the sensor response, which is defined as the variation in the output of a sensor when exposed to a particular test gas under identical conditions after a finite interval of time.

1.3.1 Overlapping sensitivity

Overlapping sensitivity is by far the most challenging limitation of a sensor array. It renders a nicely fabricated sensor array less capable of discriminating between two gases/odours in spite of having an appreciable sensitivity.

(a) LPG   (b) CCl₄

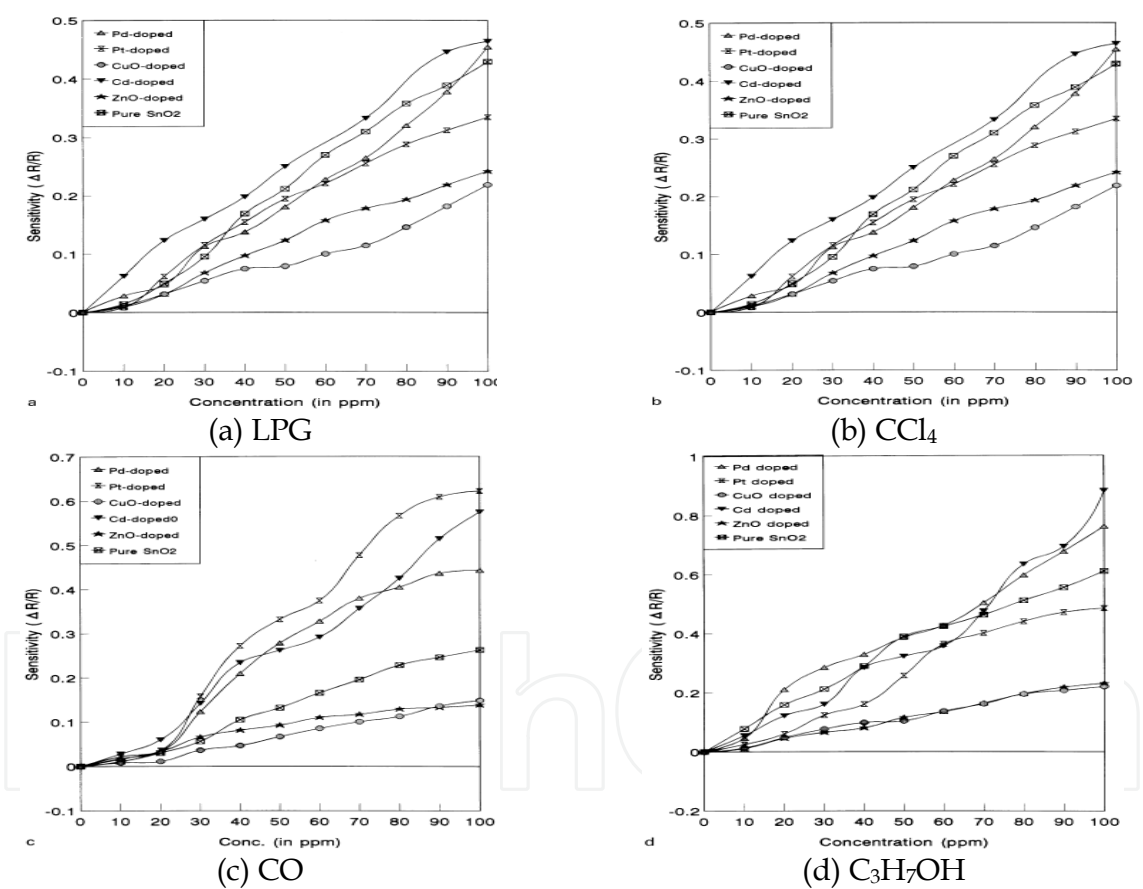(c) CO   (d) C₃H₇OH

Fig. 5. Sensitivity characteristics of oxygen plasma-treated array at different concentrations (Chaturvedi *et al.*, 1999)

Fig. 5 shows the plot of sensitivity with concentration of the test gas for a 6-sensor array. It is clear from Figs. 5 (a) and (b) that the sensor array exhibits almost identical sensitivity for LPG and CCl₄ and hence representing a very poor selectivity for these gases. It should be noted that all the sensors of the array are appreciably sensitive to all the 4 test gases. In spite of this fact we would not be able to discriminate between two of the 4 gases (i.e. LPG and

CCl_4). Therefore, in addition to having good sensitivity, the sensors in the array should respond differently to different test gases/odours.

1.3.2 Saturation and drift

Fig. 4 shows the response vs concentration of an array of 4 sensors exposed to the vapour of a particular alcoholic beverage. It can be seen that responses of almost all the sensors saturate after a particular concentration of test gas has been injected into the experimental chamber. This phenomenon makes the quantification of the test gas impossible at higher concentrations.

Drift introduces an unwanted temporal variation in the sensitivity of a sensor array. This means the response of the array to same gases under identical conditions may be entirely different from what was obtained previously. When previously learned sensor patterns become obsolete, the ability of the sensor to discriminate is lost. In fact, sensor drift is the highest obstacle in the wide marketability of low cost gas sensors.

All the above mentioned problems are hindrance to proper identification and quantification of gases/odours. Thick film sensors are well known for their design ruggedness, ease of fabrication, sensitivity to a plethora of gases/odours, and most importantly for being economical. The above mentioned limitations of thick film sensors negate their other desirable features discussed above and hence, a promising technology sometimes seems to fall short of achieving its objectives. The limitations imposed by poor selectivity and response saturation can be overcome by employing computational techniques to extract both qualitative and quantitative information. The role of computational techniques in gas/odour discrimination can never be underestimated. Wherever possible, putting more emphasis on computational methods can save a significant amount of resources since breaking innovation in the fabrication technology requires time and effort. The next section presents an overview of computational challenges put forth by popular sensors with an aim of proper identification and quantification of individual odorants.

2. Computational challenges

Choice of an appropriate technique is highly dependent on the problem in hand. In the context of E-nose systems, the term pattern analysis applies to both qualitative and quantitative analysis of odours. The response data generated from a sensor array are multivariate in nature. There are several issues, which require careful consideration for a successful design of a pattern analysis system. Signal pre-processing, feature extraction, feature selection, classification, regression, clustering and multi-fold cross-validation are the most prominent goals of a pattern analysis system, for which critical design issues are to be taken care of. The first computational stage in a pattern analyzer is often the signal pre-processing stage. The main purpose of a pre-processing stage is to select a number of parameters that are descriptive of the sensor array response. The choice of parameters can significantly affect the performance of the subsequent modules in the pattern analysis system. Fig. 6 (a) shows the scatter plot for S-1 as SnO_2 and S-2 as SnO_2 doped with Sb_2O_3 for the 4 sensor array mentioned in the previous section.

The plots confirm that the clusters are not only overlapping but also a high degree of scattering of data points. The overlapping of clusters is due to the cross sensitivity and is

attributed to the material properties of the sensors. The goal of a pre-processing stage is to minimize the spread in an individual clusters and maximize the distance between two clusters. Therefore, a pre-processing technique should be applied, which utilizes the statistical properties of the data set to maximize their inter class separation and minimize the intra class separation. The result of such a pre-processing is shown in Fig. 6 (b) to establish the importance of a pre-processing stage. The technique used is Transformed Cluster Analysis (TCA). Details of TCA can be read from a published work of the author (Kumar *et al*. 2010).

(a) 2-D scatter plot for responses of sensors S-1 and S-2

(b) 2-D scatter plot for transformed responses of sensors S-1 and S-2 (T1 and T2 are transformed responses of sensors S-1 and S-2)
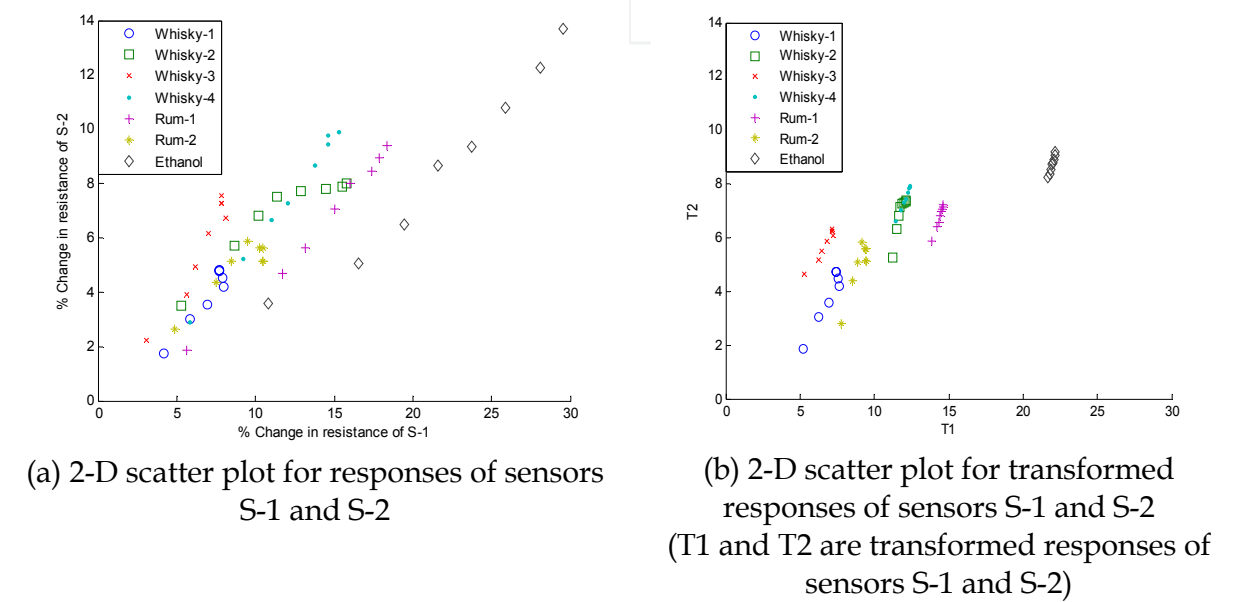
Fig. 6. 2-D scatter plot for responses and transformed responses of sensors S-1 and S-2

The actual identification/quantification part of pattern analysis begins after pre-processing. Pattern analysis techniques are generally of two types viz. parametric and non-parametric techniques. Parametric techniques do not require any prior information on the type and number of different classes contained in data. In non-parametric techniques a set of response patterns is compared against each other on the basis of degree of similarity or dissimilarity (Gardner, 1987). Thus, non-parametric techniques are more general in nature.

Statistical pattern analysis techniques like Principal Component Analysis (PCA) and Cluster Analysis (CA) are one of the most popular non-parametric techniques. PCA overcomes the "curse of dimensionality" introduced by the response vector of a multi-sensor array by choosing "principal components" along the directions of maximum variance. Principal components are a linear combination of original variables with the redundant information eliminated. The reduced dimensionality of data makes the subsequent feature extraction task simpler.

Feature extraction attempts to find a low dimensional mapping that preserves most of the information in the original feature vector. The mappings thus formed enhance the information content of the feature vector. Feature extraction techniques also help signal representation, which can be useful for extrapolatory data analysis. They are helpful in visualizing high dimensional data. Most of the feature extraction techniques for E-nose

applications are based on PCA, which is a signal representation technique that generates projections along the directions of maximum variance. Learning in pattern analyzers is viewed as the optimization of a process to obtain a minimum value for a solution of a pre-specified objective function (criterion). Analysis of patterns by an analyzer is carried out either by supervised or unsupervised learning schemes.

The pattern analysis techniques applied to the output of a sensor array should be biologically inspired if an E-nose is to sniff like humans. This requires application of biologically inspired algorithms to the senor output. Artificial neural networks (ANNs) are such a class of computational paradigms, the inspiration for which comes originally from the studies of mechanism of information processing in biological nervous system, particularly brain (Bishop, 1994). The advantages of ANNs include massive parallelism, distributed processing and computation, learning ability, generalization ability and adaptability. Apart from ANNs, fuzzy logic and genetic algorithm are some other techniques which constitute a class of paradigms known as "soft computing". Soft computing is fast replacing statistical learning techniques in pattern analysis applications. Also, a lot of work has been done in the area of gas/odour discrimination using soft computational techniques. Most of the techniques described above have been adopted from the field of chemometrics and they can be labelled as 'statistical' pattern recognition techniques as opposed to soft computational techniques, which are recent in origin and are in general biologically inspired. "Soft Computation" is a name given to a class of computational paradigms, which seek to find approximate solutions to ill-posed problems. It is tolerant of imprecision, uncertainty, partial truth, and approximation just like the human mind. The principal constituents of soft computing are ANNs, Fuzzy Logic, Support Vector Machines, and Evolutionary Computing.

Soft computational techniques have revolutionized the arena of artificial olfaction by immensely reducing dependency on flawless and meticulously designed sensor hardware. Proper application of soft computational techniques can improve the discrimination obtained using the response of poorly selective sensors to a great extent thus, saving additional costs on possible replacement and fabrications of novel sensor hardware. ANNs are one of the foundation pillars of soft computing. Their enormous learning capability, massively parallel architecture, and availability of a large number of learning algorithms for their training makes them a popular choice for a wide variety of computational tasks. A close scrutiny of the available literature reveals that the choice of pattern analysis techniques for artificial olfaction has been highly problem-dependent (Osuna, 2002). Different types of sensor arrays generate response data with different statistical properties making selection of an appropriate technique a difficult task. In the context of ANNs, the choice is often between a lesser architectural complexity and a lower system error. In view of all these, in the next section identification task of 7 different alcohols and alcoholic beverages is taken up using ANN and the response of the 4 sensor array described in section one.

3. ANNs for odour identification

The basic computational unit in an ANN is neuron, which is a mathematical function used to approximate input–output mappings. The output of a neuron (also known as firing of a neuron) is dependent upon the synaptic weight connection between all the neurons in the network. This synaptic weight ensemble changes if the actual output is not equal to the desired output. The process of change in synaptic weight is known as 'learning' or 'training'

of ANN. A well trained ANN can perform any task related to classification, function approximation or prediction with some small amount of error with a brand new data set. In ANNs neurons are arranged in different layers with 'hidden layer' being responsible for performing most of the computational tasks. ANNs can be trained to approximate any non-linear input-output mapping. It can be shown mathematically that any ANN trained with orthogonal least squares algorithm is able to approximate any input-output mapping with arbitrary accuracy provided the number of neurons in the hidden layer is large enough (Hykin, 2009). However, a back-propagation (BP) algorithm is by far the most popular method of training an ANN. It is less complex and requires lesser number of neurons to perform the computations.

3.1 Back-propagation algorithm

BP is an algorithm where input vectors and the corresponding target vectors are used to train a network until it can approximate a function, associate input vectors with specific output vectors, or classify input vectors in an appropriate way as defined by the user. Networks with biases, a hidden layer, and a linear output layer are capable of approximating any function with arbitrary accuracy. A standard BP is a gradient descent algorithm, in which the network weights are moved along the negative of the gradient of the performance function. The term *BP* refers to the manner, in which the gradient is computed for nonlinear multilayer networks. Basically, error BP consists of two passes through the different layers of the network: (1) forward pass, and (2) backward pass.

In the forward pass the input vectors are applied to the sensory nodes of the network, and its effect propagates through the network layer-by-layer. The actual response of the network is delivered by the output nodes in the form of an output vector. The outputs are compared with a target vector and the difference is generated as error. Let the error signal at the output of neuron *j* at iteration *n* be defined by:

$$e_j(n) = d_j(n) - y_j(n) \tag{2}$$

where, $d_j(n)$ represents the desired output at the output node *j* at iteration *n*, and $y_j(n)$ be the actual output at the output node *j* at iteration *n*.

Let the instantaneous value of the error energy for neuron *j* be defined as:

$$\frac{1}{2} e_j^2(n)$$

Then, for all neurons in the output layer instantaneous value $\xi(n)$ of the total energy is given by:

$$\xi(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n) \tag{3}$$

where, *C* is the set of all neurons in the output layer of the network. Let *N* denote the total number of patterns contained in the training set. The average squared energy over the entire training sample is now given by:

$$\xi_{avg}(N) = \frac{1}{N}\sum_{n=1}^{N}\xi(n) \tag{4}$$

For a given training set, ξ_{avg} is called the cost function, which is a measure of learning performance. Minimization of the cost function is done iteratively. The weights associated with the network are updated on a pattern-by-pattern basis until one complete presentation of the entire training set (epochs) has been done. The adjustments to the weights are made in accordance with the respective errors computed for each pattern presented to the network. The arithmetic average of these individual weight changes over the entire training sets presents an estimate of the true change that would result from modifying the weights based on minimizing the cost function ξ_{avg} over the entire training set. Fig. 7 shows a neuron j being fed by a set of input signals produced by a layer of its neurons to its left.

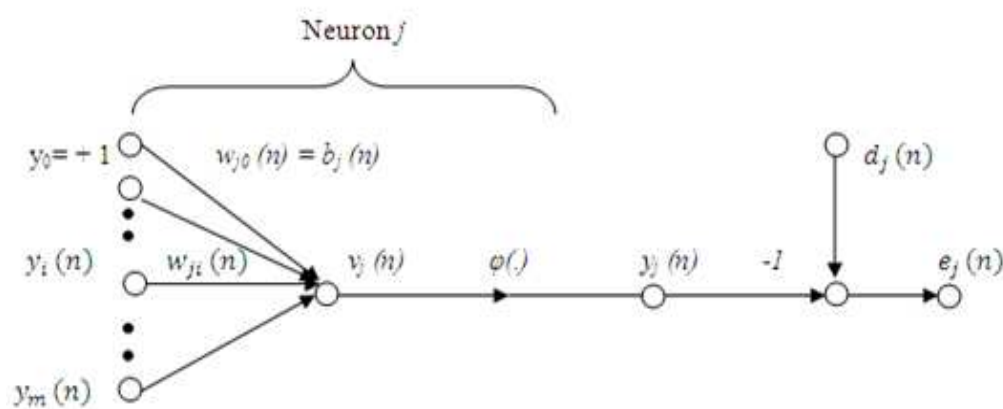Fig. 7. Neuron 'j' being fed by a set of signals from a previous layer of neurons

The induced local field $v_j(n)$ produced at the input of the activation function associated with neuron j is given by,

$$v_j(n) = \sum_{j=0}^{m}\omega_{ji}(n)y_i(n) \tag{5}$$

where m is the total number of inputs applied to neuron j, and ω_{ji} is the synaptic weight from neuron i to neuron j. The signal $y_j(n)$ appearing at the output of neuron j at iteration n is a function of the induced local field

$$y_j(n) = \varphi\{v_j(n)\} \tag{6}$$

The BP algorithm applies a correction $\Delta\omega_{ji}(n)$, to the synaptic weight $\omega_{ji}(n)$, which is proportional to the partial derivative

$$\frac{\partial\xi(n)}{\partial w_{ji}(n)}$$

Applying the chain rule of calculus, this gradient can be expressed as:

$$\frac{\partial \xi(n)}{\partial \omega_{ji}(n)} = \frac{\partial \xi(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \frac{\partial v_j(n)}{\partial \omega_{ji}(n)} \tag{7}$$

The partial derivative

$$\frac{\partial \xi(n)}{\partial w_{ji}(n)}$$

represents a sensitivity factor. It determines the direction of search in weight space for the synaptic weight ω_{ji}.

Differentiating both sides with respect to $e_j(n)$ we get

$$\frac{\partial \xi(n)}{\partial e_j(n)} = e_j(n) \tag{8}$$

Differentiating both sides of (2) with respect to $y_j(n)$, we get,

$$\frac{\partial e_j(n)}{\partial y_j(n)} = -1 \tag{9}$$

Differentiating both sides of (6) with respect to $v_j(n)$, we get,

$$\frac{\partial y_j(n)}{\partial v_j(n)} = \varphi_j'\{v_j(n)\} \tag{10}$$

Also, differentiating (5) with respect to $\omega_j(n)$, we get,

$$\frac{\partial v_j(n)}{\partial w_{ji}(n)} = y_i(n) \tag{11}$$

The use of (8) to (11) in (10) yields:

$$\frac{\partial \xi(n)}{\partial \omega_{ji}(n)} = -e_j(n)\,\varphi_j'\{v_j(n)\}y_i(n) \tag{12}$$

The correction $\Delta\omega_{ji}(n)$ applied to $\omega_{ji}(n)$ is defined by the delta rule:

$$\Delta\omega_{ji}(n) = -\eta \frac{\partial \xi(n)}{\partial \omega_{ji}(n)} \tag{13}$$

where, η is the learning-rate parameter of the BP algorithm. The gradient descent in weight space takes place in a direction for weight change that reduces the value of $\xi\,(n)$ The use of (12) in (13) yields

$$\Delta w_{ji}(n) = \eta \delta_j(n) y_i(n) \tag{14}$$

where, the local gradient δ_j(n) is defined by

$$\delta_j(n) = -\frac{\partial \xi(n)}{\partial v_j(n)} = -\frac{\partial \xi(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)}$$

$$= e_j(n)\varphi_j'\{v_j(n)\}$$

(15)

The local gradient points to required change in synaptic weights. Hence, the above relation between the learning rate, local gradient and weight correction can be summarized as follows:

(Weight Correction) = (Learning Rate)*(Local Gradient)*(Input Signal of Neuron 'j')

Learning Rate Parameter and Momentum Constant:

The learning rate parameter η is a measure of the changes to the synaptic weights in the network over subsequent iterations. Thus, a smaller learning rate parameter makes smaller changes in synaptic weights and the trajectory in the weight space is smoother. A smaller learning rate results in a slow learning. If the learning rate parameter η is made too large to speed up the learning process, the resulting large changes in the synaptic weights assume such a form that the network may become unstable. To avoid the danger of instability while keeping the learning rate fast enough, another term is added to the delta rule, which is known as the momentum constant and is denoted by α. Hence, Eq. 14 becomes:

$$\Delta w_{ji}(n) = \alpha \Delta w_{ji}(n-1) + \eta \delta_j(n) y_i(n)$$

(16)

The inclusion of momentum term in the BP algorithm has a stabilizing effect in directions that oscillate in sign. The momentum term also prevents the learning process from terminating in a shallow local minimum on the error surface. The training through a BP algorithm proceeds iteratively. A prescribed set of training examples are fed repeatedly to the ANN. The learning process continues on an epoch-by-epoch basis till the stabilization of the synaptic weights and bias levels of the network, and most importantly, the convergence of the average squared error over the entire training set to some minimum value. A back-propagation algorithm cannot converge. However, it is considered to have converged when the absolute rate of change in average squared error per epoch is "sufficiently small". The rate of change in average squared error is typically considered to be small enough if it lies in the range of 0.1 to 1% per epoch.

The sensor response curves of Fig. 4 were sampled at equal intervals of concentration and a data set was prepared. This data set will now be used to train an ANN using a BP algorithm.

3.1.1 Identification with sampled data

A three layer feed-forward neural network with sigmoidal activation function was simulated for the classification task. Neural network simulation was implemented in MATLAB using TRAINGDM function. The number of neurons in the input and output layers were fixed as 4 and 7 respectively as there are 4 sensors and 7 classes of gas/odour. Simulated network was trained by input vectors available with training data set while

learning parameters such as learning rate (η) and momentum constant (α) and the number of neurons in the hidden layer were optimized during experimentations. After several repeated experiments the optimum number of neurons in the hidden layer was found to be six as it gave the minimum system error, which is measured in terms of mean square error (MSE) .The optimized BP network with a configuration of 4:6:7 and an optimized set of weights and biases was trained repeatedly with 4 sets of training data and by changing the values of learning rate (η) and momentum constant (α) from 0.1 to 0.9. The network was trained for a fixed 30,000 epochs with an error goal of zero. The trained network for minimum system error was then tested with 4 different test data sets. The training and test performance for all values of learning rate and momentum constants was noted. System error was studied at a particular learning rate for different values of the momentum constants in the range of 0.1 to 0.9 for 4 different testing subsets.
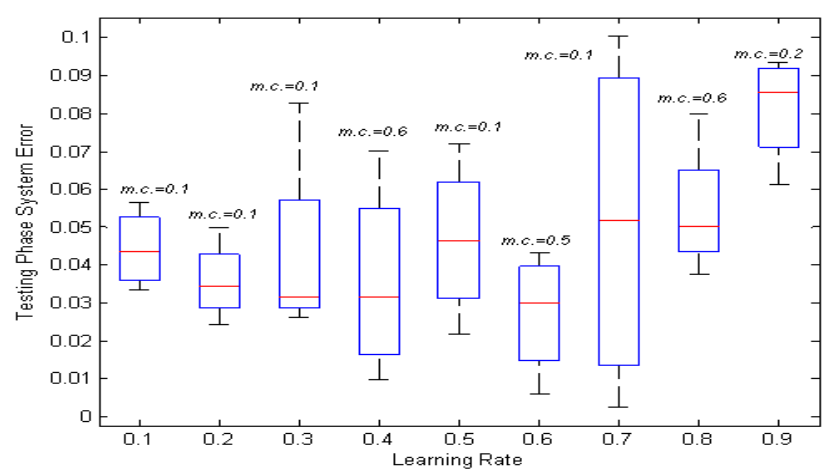

Fig. 8. Box whisker diagram for testing phase system error with learning rate for 4 testing subsets for different values of the momentum constant *(m.c.)* (BPNN)

Fig. 8 shows the testing phase system error at a particular value of learning rate and a momentum constant corresponding to which minimum average system error is observed. The BP neural network (BPNN) trained with raw data exhibited poor classification performance for all training subsets and also very high system error in both training and testing phases for almost all combinations of learning rates and momentum constants. A high degree of spread in system error is visible from Fig. 8 implying the inconsistent performance of BPNN trained with raw data. The diagrams show the variation in error performance with an optimum combination of learning rate and momentum constant calculated over 4 trials with different testing subsets.

3.2 Identification performance of radial basis function neural network

Apart from being subjected to a BP-trained neural classifier, the sampled data were fed as inputs to a Radial Basis Function Neural Network (RBFNN) classifier. RBFNN was chosen as a classifier because it takes shorter time to train apart form having a lower system error. In the following subsection, a brief introduction to RBFNNs is given and the subsequent subsections describe the identification performance of the RBFNN with respect to the present problem.

3.2.1 Radial basis function neural network

The radial basis function network is primarily composed of three layers. The first is an input layer. The second layer is the hidden layer which is primarily responsible for computation and the third layer consists of neurons with linear activation functions and it provides the output of the network corresponding to the input patterns (Hykin, 2009). The training of RBFNNs involves providing the best fit to the training data by finding a surface in a multidimensional space. An interpolation between the data points is performed in the testing phase. The RBFNN solves a classification problem by applying a nonlinear transformation from input space of lower dimension to the hidden space of higher dimension, since it increases the likelihood of correct classification for the given problem (Cover, 1965). The most popular learning strategy of RBFNNs involves the use of Gaussian functions with the selection of centres being done in a random manner. The standard deviation of the Gaussian function is fixed according to the "spread" of the centres. Given below is a radial basis function, with centre at 't'

$$G(||x-t_i||^2)\exp(-\frac{m_1}{d_{\max}^2}||x-t_i||^2) \quad i=1,2,.....m \tag{17}$$

where, m is the number of centres and $d_{\max}$ is the maximum distance between the randomly chosen centres. Also, the standard deviation (spread) of the RBF is given by

$$\sigma = \frac{d_{\max}}{\sqrt{2m_1}} \tag{18}$$

Thus, the learning process undertaken by RBFNN involves the optimization of the hidden layer's activation functions and the optimization of the output layer's weights. Fig. 9 shows a typical radial basis neuron. The net input to the radial basis transfer function is the vector distance between its weight vector and input vector multiplied by the bias, which allows the sensitivity of the neuron to be adjusted. The equations used in the neural model are given by:

$$o = \|w - p\|.b \tag{19}$$
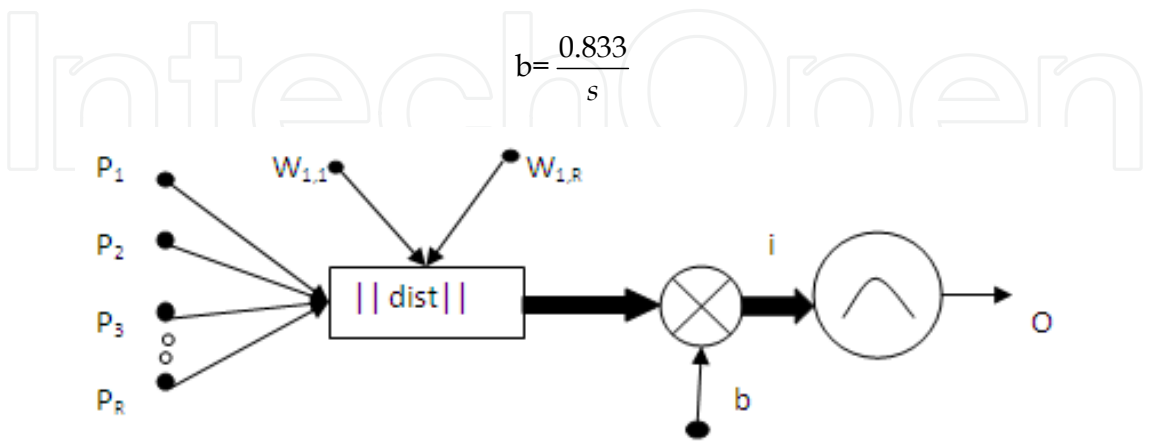
$$b = \frac{0.833}{s} \tag{20}$$

Fig. 9. Radial basis function neuron

where, o and b denote the output of a typical radial basis neuron and its bias, respectively.

The quantity is known as spread constant and is the most important learning parameter of a radial basis network. The radial basis function has a maximum of 1 when its input is 0. As the distance between w and p decreases, the output increases. Thus, a radial basis neuron acts as a detector that produces 1 whenever the input p is identical to its weight vector w. As is evident from the plot of radial basis function the function returns a value of 0.5 when the net input to radial basis transfer function is 0.833. The bias is given by Eq. (20). This determines the width of an area in the input space, to which each neuron responds. The spread constant should be large enough so that neurons respond strongly to overlapping areas of the input space.
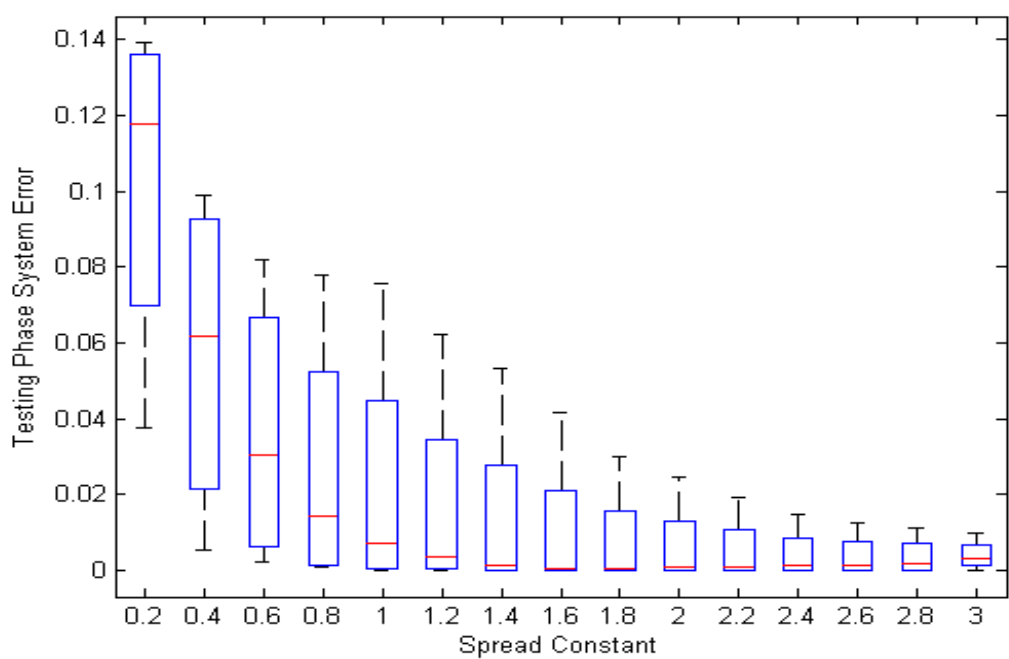
Fig. 10. Box whisker diagram for testing phase system error with spread constant

3.2.2 Identification results of RBFNN classifier

RBFNN employed in this study utilizes the *newrbe* function implemented in MATLAB. The function creates a radial basis network with the number of neurons in the hidden layer equal to the number of training patterns. The network was simulated first with the training data sets and was tested with test data sets. The spread constant of the network was varied from the 0.2 to 3.0 at regular intervals. The results thus obtained have been depicted in the form of a box whisker diagram of Fig. 10. K-fold cross validation scheme was used with K=6, to avoid overfitting. It is evident from Fig. 10 that as the spread constant increases, the variation in the results decreases. The minimum testing phase system error was obtained at a spread constant of 2.6 and 100% identification was achieved.

4. Fuzzy sets for odour discrimination

The foundations of fuzzy logic are based on the concept of fuzzy sets. A fuzzy set is a set without a clearly defined boundary (Zadeh, 1965). Human smell processing is inherently

fuzzy in nature. When the qualitative remarks are used about something, we actually do a fuzzy classification task, in which our sensory responses are assigned to more than one predefined classes with varying degrees of belongingness to them. This *degree of belongingness* is known as degree of membership in fuzzy set theory. The qualitative remarks come in the form of linguistic labels such as *rose-like, apple-like*. Apart from this, human olfactory system is capable of doing multi-way classification. Given 3 types of fragrances to smell, a human being is able to tell, which one was *apple-like* or *rum-like* or *rose-like* and also that which fragrance among the three was strongest, which one was weakest and which one was of in-between intensity. In the above case, along with the qualitative information some quantitative information has also been retrieved, which enables us to label the fragrances according to their 'intensity'. This has served as the primary motivation for the design of a network, which can retrieve both the qualitative and quantitative information when the sensor array response vectors are given as input vectors to the network.

Fuzzy set theory is a generalization of the conventional crisp set theory. It measures the degree to which an event occurs (Zadeh, 1965). As discussed above, each element of a fuzzy set has a degree of membership assigned to it in accordance with a membership function. The most commonly used membership functions in the literature being *triangular* and *trapezoidal* membership functions.

Let $X = \{x_1, x_2, x_3, \ldots, x_n\}$ be a non-fuzzy set. The subsets of X are called bit vectors or bivalent messages. If $X = \{x_1, x_2, x_3, x_4\}$, then $X = \{1,1,1,1\}$, $\varphi = (0,0,0,0)$ and the subset $A = \{x_1, x_4\}$ is represented as $A = (1, 0, 0, 1)$. The 1s and 0s indicate the presence or absence of the ith element x_i in the subset. Each non fuzzy subset A can be defined as one of the two-valued membership functions $\mu_A : X \rightarrow \{0,1\}$. The power set 2^X of X is the set of all of X's subsets. There are 2^n possible messages defined on X (in 2^X). In this example, there are 2^4 possible messages. In contrast, fuzzy subsets of X are referred to as *fit vectors* or *fit messages*. Each subset A of X can be defined as one of the continuum-many continuous-valued membership functions $\mu_A: X \rightarrow \{0,1\}$. Fuzzy sets can also be represented geometrically and this representation gives more insight into the intricacies of fuzzy sets and operations related to them (Kosko, 2007). According to this representation, the fuzzy power set $F(2^X)$, which is the set of all fuzzy subsets of X, is visualized as a unit hypercube $I^n = [0,1]^n$ and a fuzzy set is any point in the cube I^n. Vertices of the cube I^n define non-fuzzy or crisp sets, which are a subset of X. Thus, crisp sets are special cases of the fuzzy sets.

Fig. 11 shows the geometrical representation of fuzzy sets. The sets on vertices are non-fuzzy sets and long diagonals connect non-fuzzy set complements. A fuzzy set *A* with fit values

$$\left(\frac{1}{3} \frac{4}{5} \right)$$

is represented inside the unit square consisting of all possible fuzzy subsets of two elements. The mid-point of the unit square shown in Fig. 12 is the point of maximum fuzziness. Thus, the proposition for fuzziness can be summarized as follows.
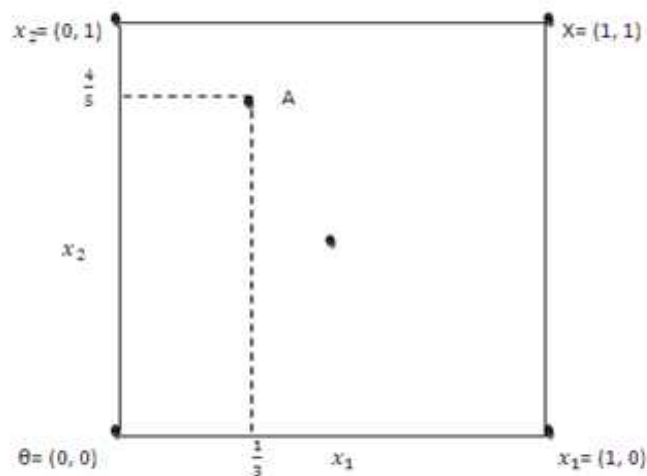
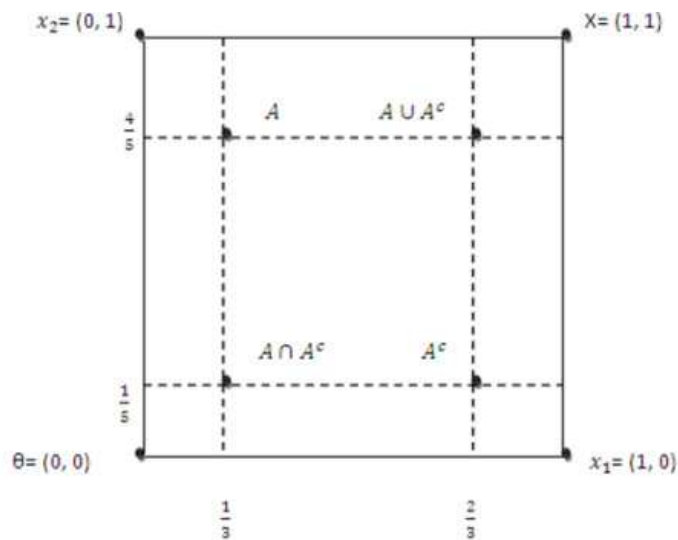Fig. 11. Geometrical representation of fuzzy set A

Fig. 12. Fuzzy set A with its complements, under-lap and overlap

A is properly fuzzy if

$$A \cap A^C \neq \varphi \tag{21}$$

$$A \cup A^C \neq X \tag{22}$$

where $A \cap A^C$ and $A \cup A^C$ are termed as overlap and under-lap respectively. The positions of a fuzzy set along with its complement, overlap set and under-lap set are shown in Fig. 12.

With the increase in fuzziness of A, all the 4 points shrink towards the midpoint of the fuzzy square, which is the point of maximum fuzziness. The size or cardinality of a fuzzy set A is given by

$$M(A) = \sum_{i=1}^{n} \mu_A(x_i) \tag{23}$$

Where, $\mu_A(x_i)$ is the membership value of the ith element of n-valued fuzzy set A.

Fuzzy logic has emerged as a promising tool for biological information processing owing to its proximity to human perceptions of logic, and real world situations, which are full of ambiguity. The combination of fuzzy logic and neural networks is reported with promising results in the classification of wines and beverages (Das *et al.*, 1999). The next section presents a novel method of fuzzy pre-processing, which gives simultaneous identification and quantification when the response samples are used to train an ANN classifier. The aim is to tell both the class and the concentration of a sample simultaneously when the sample is presented to an ANN.

5. Fuzzy subsethood for simultaneous identification and quantification of odours

It can be seen from Fig. 4 that the response of the sensor array to almost all the alcohols and alcoholic beverages in the study has saturating tendency at higher concentrations. The quantification task becomes more difficult for mere lack of information at higher concentrations. Both qualitative and quantitative classification tasks coupled together need an integrated approach to be accomplished successfully. As a first step to reduce the complexity of the problem, PCA is used on the raw data.

Fig. 13 shows the PCA plot discriminating different alcohols and alcoholic beverages used in this study. Although PCA can significantly reduce the dimensionality of the data set by having 95% of the variance in first two principal components (PC-1 and PC-2) itself, it has little effect on class separability. This necessitated investigation of a technique, which is based on proper representation of target classes in the output feature space provided the representation is inclusive enough to incorporate in itself both qualitative and quantitative information.
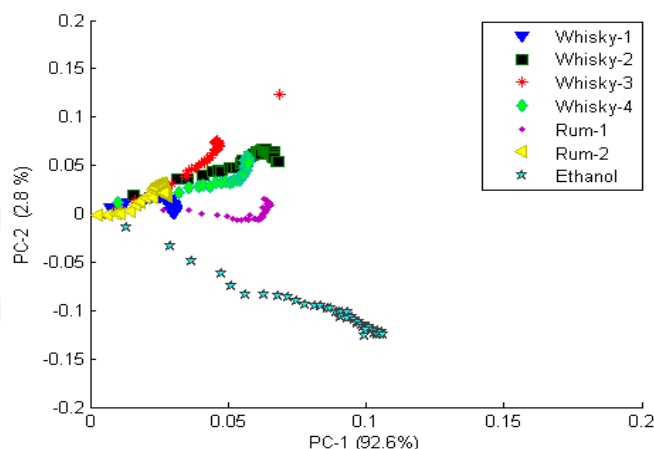

Fig. 13. 2D PCA for sensor array response to 7 alcohols/alcoholic beverages

Fuzzy subsethood representation is such an appropriate technique and applied on the data as follows. The response curves in Fig.4 were sampled at regular intervals of concentration. Each curve was sampled at 48 values of concentration and a total of 336 samples were obtained for 7 gases. As shown in Fig. 14, each sample has two memberships, one to the gas class and the other to the concentration band. For each gas, twelve concentration bands were

marked according to increasing no. of drops as band no. 1(b-1) for drops 0-1, band no. 2 (b-2) for drops 1-2 and so on. Each such concentration band consisted of 4 samples.
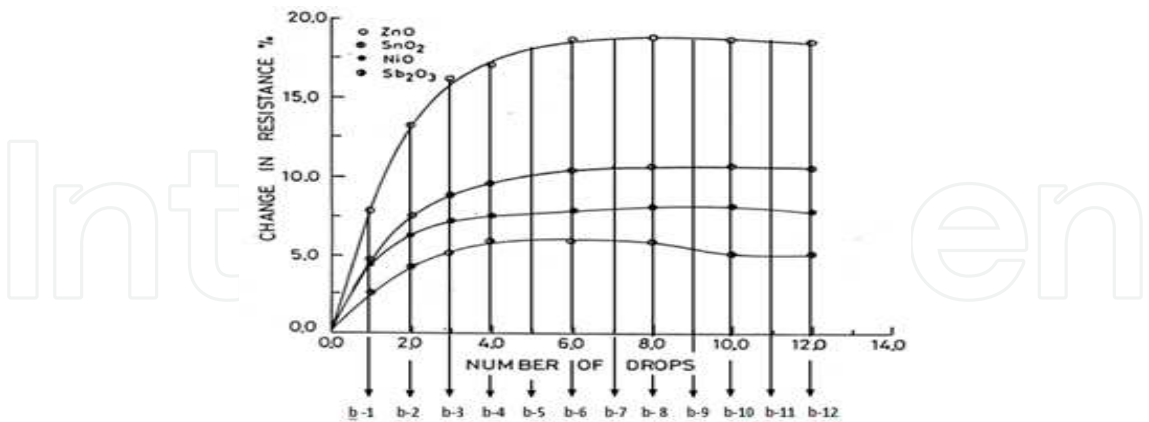

Fig. 14. Concentration bands in sensor response to whisky-2

Fuzzy memberships were assigned to each sensor response sample for all gases as follows. Centroid $\overline{S_j}$ was calculated for each set j where j denotes a gas and $\overline{S_j}$ has 4 elements as given in

$$\overline{S_j} = \begin{bmatrix} m_{1j} \\ m_{2j} \\ m_{3j} \\ m_{4j} \end{bmatrix} \tag{24}$$

where m_{1j}, m_{2j}, m_{3j} and m_{4j} and represent the centroids for the responses of sensors 1(Sb$_2$O$_3$ doped), 2(SnO$_2$), 3(NiO doped) and 4(ZnO doped) respectively. In this case the centroids are calculated by taking the mean of individual sensor response samples for different alcohols and alcoholic beverages.

The Euclidean distance d_{jk} of the sensor response vector at sample k for gas j can be obtained in the sensor response ratio vector space as given by

$$d_{jk} = \{(X_{1jk} - m_{1j})^2 + (X_{2jk} - m_{2j})^2 + (X_{3jk} - m_{3j})^2 + (X_{4jk} - m_{4j})^2\}^{\frac{1}{2}} \tag{25}$$

where, x_{1jk} is the sensor response for sensor 1, gas j, and sample k and so on. Each sample k is assigned a membership μ_{jk} in the output feature space, in the fuzzy set A_j for j^{th} gas at sample k by using triangular membership function as:

$$\mu_{jnk} = \frac{d_{jnk} - |d_{jnk}|_{max}}{|d_{jnk}|_{max} - |d_{jnk}|_{min}} \tag{26}$$

where, d_{jk} is the Euclidean distance of the sensor response vector at sample k for gas j. $|d_{jk}|_{max}$ and $|d_{jk}|_{min}$ are the modulo of the maximum and minimum values respectively of d_{jk} for a particular gas j. Similarly, the sensor response samples obtained from different gases were assigned memberships in different concentration bands of those gases.

Centroid S_{jn} is calculated for each concentration band n of j where j denotes the gas (alcohol and alcoholic beverage type) and S_{jn} has 4 elements as given by

$$\overline{S_{jn}} = \begin{bmatrix} m_{1jn} \\ m_{2jn} \\ m_{3jn} \\ m_{4jn} \end{bmatrix} \tag{27}$$

where, m_{1j}, m_{2j}, m_{3j} and m_{4j} represent the centroids for the responses of sensors 1, 2, 3 and 4 respectively in a concentration band n. The centroids are calculated by taking the simple mean of 4 samples belonging to a particular concentration band. The Euclidean distance d_{jnk} of the sensor response vector at sample k of a gas j and concentration band n of that particular gas can be obtained in the sensor response ratio vector space as given by

$$d_{jnk} = \{(X_{1jnk} - m_{1jn})^2 + (X_{2jnk} - m_{2jn})^2 + (X_{3jnk} - m_{3jn})^2 + (X_{4jnk} - m_{4jn})^2\}^{\frac{1}{2}} \tag{28}$$

where x_{1jk} is the sensor response for sensor 1, gas j, band n, and sample k, and so on. Each sample k of band n is assigned a membership μ_{jnk} in the output feature space, in the fuzzy set A_{jn} again by using triangular membership function as:

$$\mu_{jnk} = \frac{d_{jnk} - \left|d_{jnk}\right|_{max}}{\left|d_{jnk}\right|_{max} - \left|d_{jnk}\right|_{min}} \tag{29}$$

where, d_{jnk} is the Euclidean distance of the sensor response vector at sample k for band n of gas j. $\left|d_{jnk}\right|_{max}$ and $\left|d_{jnk}\right|_{min}$ are the modulo of the maximum and minimum values respectively of d_{jnk} for a particular band n of gas j. It is clear that the fuzzy set A_{jn} is a subset of fuzzy set A_j.

The degree of belongingness of A_{jn} to A_j changes, as n changes for a particular j. Thus, all the elements of A_{jn} can be mapped to a single value, which is the fuzzy subsethood value as defined below, Fuzzy subsethood measures the degree of belongingness of a fuzzy set A to its superset B and is denoted by

$$S(A,B) = Degree(A \subset B) \tag{30}$$

A fuzzy set A can be a subset of another fuzzy set B if $\mu_A(x) \leq \mu_B(x)$ for all x.

The fuzzy-subsethood theorem is given by

$$S(A,B) = \frac{M(A \cap B)}{M(A)} \tag{31}$$

$$S_{jn}(A_{jn}, A_j) = Degree(A_{jn} \subset A_j) \tag{32}$$

Using equations (31) in (32)

$$S_{jn}(A_{jn},A_j)= \frac{M(A_{jn} \cap A_{jn})}{M(A_{jn})} \tag{33}$$

In Fig. 15 the response of the array saturates completely after the concentration band b-4, resulting in a shear lack of information. However, there is a slight change in the response pattern of sensor 1 in band b-9. This change should reflect in the output feature space so that proper quantification can be obtained. For a particular concentration band n consisting of k samples of gas j for the response of sensor i the mean is given as m_{ijn}.

For n = 9, j = 2, and i = 1, the variance V_{ijn} is given by

$$V_{129} = \frac{1}{4\{(X_{1291} - m_{129})^2 + (X_{1292} - m_{129})^2 + (X_{1293} - m_{129})^2 + (X_{1294} - m_{129})^2\}} \tag{34}$$

From (28) the Euclidean distance d_{jnk} of sample $k = 1$ of $j = 2$, $n = 9$, is given by

$$d_{291} = \{(X_{1291} - m_{129})^2 + (X_{2291} - m_{229})^2 + (X_{3291} - m_{329})^2 + (X_{4291} - m_{429})^2\}^{\frac{1}{2}} \tag{35}$$

It can be observed that the first term of the variance V_{129} finds itself as a component of the Euclidean distance d_{jnk}. Since the variance V_{ijn} is calculated for the response samples of a single sensor i and the Euclidean distance d_{jnk} of a particular sample takes into account the responses of all the 4 sensors of the array, any significant change in the response of any of the 4 sensors in a concentration band is certainly going to reflect in the Euclidean distance of any sample for the same concentration band. Since membership values and subsethood are primarily based upon Euclidean distance, the change in variance for a particular sensor in a concentration band will play a part in the subsethood calculation for the response vector obtained from all the 4 sensors in the same concentration band. In this way, if any one of the 4 sensors shows less saturation at higher concentrations the possibility of correct quantification increases.
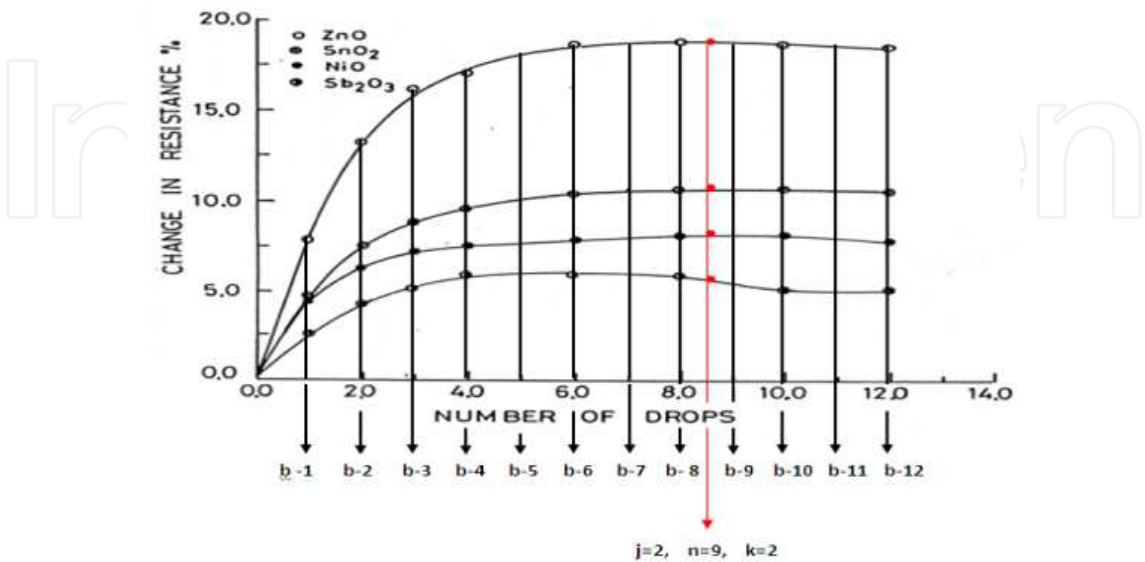

Fig. 15. Second sample of ninth band of whisky-2

5.1 Simulation results and discussions

The subsethood values were obtained for each concentration band of a particular gas and were used as the target for the neural network classifier. For a total of 7 alcoholic beverages 7 neurons were kept in the output layer. A neuron corresponding to a particular gas class was supposed to fire at a value corresponding to the fuzzy subsethood of the particular concentration band, to which the test sample belonged while all other neurons were supposed to be deactivated. A tolerance of 2% for the target fuzzy subsethood was considered appropriate. A single hidden layer feed-forward ANN was trained with a BP algorithm. The input layer consisted of 4 neurons and the number of neurons in the hidden layer was optimized by experimentation. The input data were divided into training and testing data matrices. The simulations were carried out on MATLAB platform and several different versions of a BP algorithm available in the MATLAB neural network toolbox (Mathswork Inc., 2007) were tested. Three training methods based upon a BP algorithm namely *Trainoss*, *Trainscg* and *Trainlm* have been found to give satisfactory results. To eliminate the possibility of over fitting m-fold cross validation scheme (Hykin, 2009) was used. For all the three versions *logsigmoidal* activation function was used. All the three training methodologies use default values of learning rate η and momentum constant α adaptively during the simulation run. The number of neurons in the hidden layer of the network was varied from two to nine and system error (mean square error) was noted. The networks were trained to a fixed 10,000 epochs with an error goal of 0.0001. *Trainoss*, *Trainscg* and *Trainlm* are found to train the network best when the number of neurons in the hidden layer of the network was 7, 5, and 6 respectively. In the testing phase, 12 samples were taken for a particular beverage with one sample each from a particular concentration band. The proposed network was found to give the best testing phase performance when the network was trained with *Trainlm* methodology. Table 1 shows the summary of best classifications achieved qualitatively and Table 2 shows the quantitative classification results for a network with an optimized topology of 4:6:7 trained with *Trainlm*. For qualitative classification 83 out of 84 samples were identified correctly giving a result of 98.97%.

Gas class	No. of samples correctly detected out of 12 for each gas class
Whisky-1	12
Whisky-2	12
Whisky-3	12
Whisky-4	11
Rum-1	12
Rum-2	12
Ethanol	12
Total % classification achieved	98.97

Table 1. Results of qualitative classification

Gas class	No. of samples correctly detected in Concentration Bands b-1 to b-12
Whisky-1	8
Whisky-2	8
Whisky-3	6
Whisky-4	9
Rum-1	6
Rum-2	9
Ethanol	10
Total % quantification achieved	66.67

Table 2. Results of quantitative classification

Whereas, 56 out of 84 samples were detected correctly in concentration bands b-1 to b-12, giving a success rate of 66.67%. The results seem to be very encouraging since the sensor response at higher concentrations of the test gas remains saturated for almost all types of alcohols and alcoholic beverages, resulting in a shear lack of information at higher concentrations as evident from Fig. 4.

6. Conclusions

In this chapter a neural fuzzy identifier/quantifier was presented for discrimination of several alcoholic beverages using responses of a poorly selective sensor array. The simulation results obtained using fuzzy subsethood based feature extraction, validate the presumption that the limitations imposed by poor selectivity of chemical sensors can be overcome using appropriate soft computational technique. This chapter also highlights the importance of a pre-processing stage before the response sampled are fed to a neural classifier. The proposed technique of fuzzy subsethood encoding is also similar to a pre-processing stage, which makes the subsequent neural classification faster and error free. It is important to have a classifier with a small number of neurons in the hidden layer so that it can be implemented easily into custom VLSI chips. The technique presented in this chapter accomplishes the identification/quantification task with a few neurons in the hidden layer (i.e. 6) and hence its efficacy is established.

There is a scope for future work by trying to make the identification/quantification techniques less problem-dependent and more general in nature, which would eventually enable the realization of a highly marketable hand-held E-nose system.

7. Acknowledgements

8. References

Bishop, C.M. (1994). *Neural Networks for Statistical Pattern Recognition*, Oxford University Press, Oxford, UK

Chaturvedi, A.; Mishra, V.N.; Dwivedi, R. and Srivastava, S.K. (1999). Response of oxygen plasma-treated thick film tin oxide sensor array for LPG, CCl_4, CO and C_3H_7OH, Microelectronics Journal, Vol. 30, pp. 259-264

Cover, T.M. (1965). Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications In Pattern Recognition, IEEE Trans. Electron. Comput., EC-14, 326-334

Das, R.R.; Shukla, K.K.; Dwivedi, R. and Srivastava, A.R. (1999). *Discrimination of Individual Gas/Odor Using Responses of Integrated Thick Film Tin Oxide Sensor Array and Fuzzy-Neuro Concept*, Microelectronics Journal, Vol. 30, pp. 793-800

Gardner, J.W. (1987). *Pattern Recognition in the Warwick Electronic Nose*, 8th International Congress of European Chemoreception Research Organization, University of Warwick, UK

Hykin, S. (2009). *Neural Networks and Learning Machines*, Chapter 5, PHI

Ikohura, K. (1981). *Tin Oxide Gas Sensor for Deoxidizing Gas, New Materials and New Processes in Electrochemical Technology*, Vol. 1, pp. 43-50

Kosko, B. (2007). Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence, India: Prentice Hall, Chapter 7

Kumar, R.; Das, R.R.; Mishra, V.N. and Dwivedi, R. (2010). A neuro-fuzzy classifier-cum-quantifier for analysis of alcohols and alcoholic beverages using responses of thick film tin oxide gas sensor array, IEEE Sensors Journal, Vol. 10, No. 9, pp. 1461-1468

MATLAB (2007). *Toolbox Documentations*, MATHWORKS Inc., USA, Licence Number: 338828-DVDWIN07B

Menini, A.; Lagostena, L. and Boccaccio, A. (2004).*Olfaction: From Odorant Molecule to the Olfactory Cortex*, New Physical Sci., Chapter 19, pp. 101-104

Morrison, S.R. (1987). *Selectivity in Semiconductor Gas Sensors*, Sensors and Actuators, Vol. 12, pp. 425-440

Moseley, P.T. (1992). *Material Selection for Semiconductor Gas Sensors*, Sensors and Actuators B, Vol. 6, pp. 149-156

Nagle, H. T.; Schiffman, S. S. and Osuna, R.G. (1998) *The How and Why of Electronic Noses*, IEEE Spectrum, pp. 22-34

Nakamoto, T.; Fukunishi, K. and Morizumi, T. (1990). *Identification Capability of Odor Sensor Using Quartz Resonator Sensor Array and Neural-Network Pattern Recognition*, Sensors and Actuators B, Vol. 1, pp. 473-476

Nayak, M.S.; Dwivedi, R. and Srivastava, S.K. (1992). Integrated Sensor Array for Detection of Alcohols and Alcoholic Beverages, J. Inst. Eng. (India), Vol. 75, pp. 94-98

Osuna, R. G. (2002).*Pattern Analysis for Machine Olfaction: A Review*, IEEE Sensors Journal, Vol. 2, pp. 189-202

Sarro, P.M. (1992). *Sensor Technology Strategy in Silicon*, Sensors and Actuators A, Vol. 31, pp. 138-143

Sears, W.M.; Colbow, K.; Slamka, R. and Consadori, F. (1990). *Selective Thermally Cycled Gas Sensing Using Fast Fourier Transform Technique,* Sensors and Actuators B, Vol. 2, pp. 283-289

Shurmer, H.V.; Gardner, J.W. and Corcoran, P. (1990). *Intelligent Vapor Discrimination Using a Composite Twelve Element Sensor Array,* Sensors and Actuators B, Vol. 1, pp. 256-260

Watson, J. (1984*). The Tin Oxide Gas Sensors and Its Applications,* Sensors and Actuators, Vol. 5, pp. 29-42

Zadeh, L.A. (1965). Fuzzy sets, Inf. Control, Vol. 8, pp. 338-353
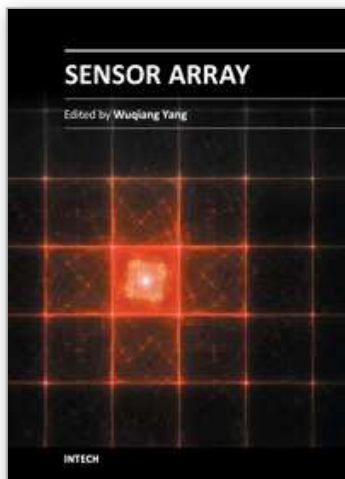
Sensor Array

Edited by Prof. Wuqiang Yang

Sensor arrays are used to overcome the limitation of simple and/or individual conventional sensors. Obviously, it is more complicated to deal with some issues related to sensor arrays, e.g. signal processing, than those conventional sensors. Some of the issues are addressed in this book, with emphasis on signal processing, calibration and some advanced applications, e.g. how to place sensors as an array for accurate measurement, how to calibrate a sensor array by experiment, how to use a sensor array to track non-stationary targets efficiently and effectively, how to use an ultrasonic sensor array for shape recognition and position measurement, how to use sensor arrays to detect chemical agents, and applications of gas sensor arrays, including e-nose. This book should be useful for those who would like to learn the recent developments in sensor arrays, in particular for engineers, academics and postgraduate students studying instrumentation and measurement.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Ravi Kumar (2012). Neuro-Fuzzy Classifiers/Quantifiers for E-Nose Applications, Sensor Array, Prof. Wuqiang Yang (Ed.), ISBN: 978-953-51-0613-5, InTech, Available from: http://www.intechopen.com/books/sensor-array/neuro-fuzzy-classifiers-quantifiers-for-e-nose-applications

INTECH
open science | open minds