ELSEVIER

Journal of Analytical and Applied Pyrolysis
40–41 (1997) 135–158

JOURNAL of ANALYTICAL and APPLIED PYROLYSIS

Quantitative analysis of the adulteration of orange juice with sucrose using pyrolysis mass spectrometry and chemometrics

Royston Goodacre [a,*], David Hammond [b], Douglas B. Kell [a]

[a] *Institute of Biological Sciences, University of Wales, Aberystwyth, Ceredigion SY23 3DA, UK*
[b] *Reading Science Service Ltd., Lord Zuckermann Research Centre, The University, Whiteknights, Reading RG6 7LA, UK*

Received 12 October 1996; accepted 12 November 1996

Abstract

A 10% (w/v) beet sucrose solution was used to adulterate freshly squeezed orange juice over the range 0–20% (or 0–20 g l^{-1} of added sucrose). Samples were analysed by the rapid automated screening technique of Curie-point pyrolysis mass spectrometry (PyMS). To deconvolve these spectra neural cognition-based methods of multilayer perceptrons (MLPs) and radial basis functions (RBFs) and the linear regression technique of partial least squares (PLS) were studied. It was found that each of the methods could be used to provide calibration models which gave excellent predictions for the level of sucrose adulteration at levels below 1% for samples, with an accuracy of ±1.3%, on which they had not been trained. The best results were obtained using PLS when 8 latent variables were employed for predictions. Furthermore, the inputs to MLPs could be reduced using principal components analysis (PCA) from 150 masses to 8 PC scores without any deterioration of the predictive ability of the model, highlighting that PCA is an excellent pre-processing step which has the potential to speed up neural network learning as there are fewer weights to update. Since any foodstuff can be pyrolysed in this way, the combination of PyMS with chemometrics constitutes a rapid, powerful and novel approach to the quantitative assessment of food adulteration generally. © 1997 Elsevier Science B.V.

Keywords: Chemometrics; Food adulteration; Neural networks; Partial least squares; Pyrolysis mass spectrometry; Radial basis functions; Quantitative analysis

* Corresponding author. Tel.: +44 1970 621947; fax: +44 1970 622354; e-mail: rrg@aber.ac.uk

PII S0165-2370(96)00973-4

1. Introduction

There is a continuing requirement for rapid, accurate, automated methods to characterise biological systems, for instance in determining whether a particular foodstuff has the provenance claimed for it or whether it has been adulterated with or substituted by a lower-grade material. A novel approach to the solution of these problems exploiting pyrolysis mass spectrometry (PyMS) and chemometrics will be described for the adulteration of orange juice.

Fruit juices are big business. In the UK alone some £800 million worth are sold per year and in the US the annual market is £7.5 billion [1]. In 1991 the British Ministry of Agriculture Fisheries and Food (MAFF) found that no less than 16 out of 21 leading brands of orange juice sold in the UK contained additional substances, such as beet sugar, while in the US it has been estimated that fraudulent products can account for 10% of the market [1]. The incentive for fraud is clear.

There are many methods that have been exploited for assessing the authenticity of orange juice and two recent excellent reviews have been published which detail these [2,3]. The most consistent fraud in the orange juice industry has been the 'stretching' of juices with preparations based on cane and beet invert sugar (sucrose) [3]. The aim is to add extra water to the juice whilst keeping the soluble solids constant, the typical level of adulteration is 5–15% of a 10% sucrose solution [2–4], which equates to 5–15 g l^{-1} of added sucrose. Methods to investigate the addition of sugar include those based on chromatography [5,6], infrared spectroscopy [7], Proton NMR [8] and most recently site-specific natural isotopic fractionation studied by nuclear magnetic resonance (SNIF/NMR) [9].

The reason that the most common sugar used to adulterate orange is beet sucrose is because the isotopic composition of this sugar and those found in the orange are very similar and difficult to differentiate. During photosynthesis the orange tree (*Citrus sinensis*) and sugar beet (*Beta vulgaris*) plant both fix CO_2 via the C3 pathway, whilst cane uses the C4 pathway [10]. Although SNIF/NMR is very powerful and can detect ostensibly low levels (14 g l^{-1}) of cane and beet sugar addition to fruit juices [9] it is very slow; since the sugars present in the juice need to be fermented to ethanol; sample analysis typically takes 5–10 days to complete. There is therefore a need for a method which is automated and has the ability to screen many hundreds of samples per working day.

Pyrolysis mass spectrometry (PyMS) is a rapid, automated, instrument-based technique which permits the acquisition of spectroscopic data from 300 or more samples per working day. Pyrolysis is the thermal degradation of a complex non-volatile material in an inert atmosphere or a vacuum. It causes molecules to cleave at their weakest points to produce smaller, volatile fragments called pyrolysate [11]. Curie-point pyrolysis is a particularly reproducible and straightforward version of the technique, in which the sample, dried onto an appropriate metal is rapidly heated (0.5 s is typical) to the Curie point of the metal, which may itself be chosen (358, 480, 510, 530, 610 and 770°C are common temperatures). For the analysis of biological material the usual pyrolysis temperature employed is 530°C because it has been shown [12,13] to give a balance between fragmentation

from polysaccharides (carbohydrates) and protein fractions. A mass spectrometer can then be used to separate the components of the pyrolysate on the basis of their mass-to-charge ratio (m/z) to produce a pyrolysis mass spectrum [14], which can then be used as a 'chemical profile' or fingerprint of the complex material analysed.

Within the food industry PyMS has been exploited to confirm the provenance of orange juice [15] and the quality of scotch whisky [16,17]. However, the interpretation of the PyMS spectra has conventionally been by the application of 'unsupervised' pattern recognition methods of principal components analysis (PCA), canonical variates analysis (CVA) and hierarchical cluster analysis (HCA). With 'unsupervised learning' methods of this sort the relevant multivariate algorithms seek 'clusters' in the data [18], thereby allowing the investigator to group objects together on the basis of their perceived closeness; this process is often subjective because it relies upon the interpretation of complicated scatter plots and dendrograms. In addition, such methods although in some sense quantitative, are better seen as qualitative since their chief purpose is merely to distinguish objects or populations.

More recently, various related but much more powerful methods, most often referred to within the framework of chemometrics, have been applied to the 'supervised' analysis of PyMS data [19,20]. Arguably, the most significant of these is the application of (artificial) neural networks (ANNs). The first demonstration of the ability of ANNs to discriminate between biological samples from their pyrolysis mass spectra was for the qualitative assessment of the assessment of the adulteration of extra virgin olive oils with various seed oils [21,22]; in that study, which was performed double-blind, neural networks were trained with the spectra from 12 virgin olive oils, coded 1 at the output node, and with the spectra from 12 adulterated oils, which were coded 0. This permitted their rapid and precise assessment, a task which was previously labour intensive and very difficult. It was most significant that the traditional 'unsupervised' multivariate analyses of PCA, CVA and HCA failed to separate the oils according to their virginity or otherwise but rather discriminated them on the basis of their cultivar. Several studies have now shown that this combination of PyMS and ANNs is also very effective for the rapid identification of a variety of bacterial strains [20,23–27].

The above studies all exploited ANNs to solve classification problems which by definition are essentially qualitative in nature. However, perhaps the most significant application of ANNs to the analysis of PyMS data is to gain accurate and precise quantitative information about the chemical constituents of microbial (and other) samples. For example, it has been shown that it is possible using this method to follow the production of indole in a number of strains of *E. coli* grown on media incorporating various amounts of tryptophan [28], to quantify the (bio)chemical constituents of complex biochemical binary mixtures of proteins and nucleic acids in glycogen, and to measure the concentrations of tertiary mixtures of cells of the bacteria *Bacillus subtilis*, *Escherichia coli* and *Staphylococcus aureus* [29–31]. The later study [31] also demonstrated that other supervised learning methods such as partial least squares (PLS) and principal components regression (PCR) could also be used to extract quantitative information from the spectra of the binary and

tertiary mixtures. Finally, the combination of PyMS and ANNs also has the potential for the screening and analysis of microbial cultures producing recombinant proteins [32] and antibiotics [33,34].

The objective of the present study is to demonstrate that the combination of PyMS with chemometrics, *viz.* neural network-based methods (multilayer perceptrons and radial basis functions) and linear regression (partial least squares) which employ supervised learning algorithms, can permit the rapid and quantitative assessment of the adulteration of orange juice with beet sucrose.

2. Experimental

2.1. Preparation of adulterated orange juice

First, 25 oranges (Outspan; Navelate from S. African) were bought from a local supermarket and were hand squeezed to give approximately 2 l of raw material. This was then centrifuged at 6000 *g* for 20 min to remove pith and particulates (the pellet was found gravimetrically to be 6.36% weight by weight). Next a 10% solution of Beet sucrose ('Silver Spoon', British Sugar) was prepared in distilled water and was used to adulterate the orange juice from 0 to 20% in steps of 0.5%; these 41 binary mixtures prepared therefore spanned the region 0–20 g l^{-1} of added sucrose.

2.2. Pyrolysis mass spectrometry (PyMS)

Of the above materials, 1 μl was evenly applied onto iron–nickel foils to give a thin uniform surface coating. Prior to pyrolysis the samples were oven-dried at 50°C for 30 min. Each sample was analysed in triplicate. The pyrolysis mass spectrometer used was the Horizon Instruments PyMS-200X (Horizon Instruments, Ghyll Industrial Estate, Heathfield, E. Sussex, UK); for full operational procedures see [25,31,34]. The sample tube carrying the foil was heated, prior to pyrolysis, at 100°C for 5 s. Curie-point pyrolysis was at 530°C for 3 s, with a temperature rise time of 0.5 s. The data from PyMS were collected over the m/z range 51–200 and may be displayed as quantitative pyrolysis mass spectra (e.g. Fig. 1). The abscissa represents the m/z ratio whilst the ordinate contains information on the ion count for any particular m/z value ranging from 51–200. Data were normalised as a percentage of total ion count to remove the most direct influence of sample size per se.

2.3. Canonical variates analysis (CVA)

Canonical variates analysis (CVA) (also referred to as discriminant function analysis (DFA)) is a multivariate statistical technique that separates objects (samples) into groups or classes by minimising the within-group variance and maximising the between-group variance [35–37].

Before CVA was employed principal components analysis (PCA) was used to reduce the dimensionality of the data and only those principal components (PCs) whose eigenvalues accounted for more than 0.1% of the total variance are used. After the first few PCs, the axes generated will usually be due to random 'noise' in the data; these PCs can be ignored without reducing the amount of useful information representing the data, since each PC is now independent of (uncorrelated with) any other PC [18,38–43].

CVA then separated the objects (samples) into groups on the basis of the retained PCs and the *a priori* knowledge of the appropriate number of groupings [35–37]; the *a priori* groups here are the known triplicate pyrolysis mass spectra and so do not bias the analysis in any way.

The objective of CVA is to maximise the ratio of the between-group to within-group variance, therefore a plot of the first two canonical variates (CVs) displays the best 2-D representation of the group separation. To effect CVA the normalised data were processed with the GENSTAT package [44] run under Microsoft DOS 6.22 on an IBM-compatible PC.

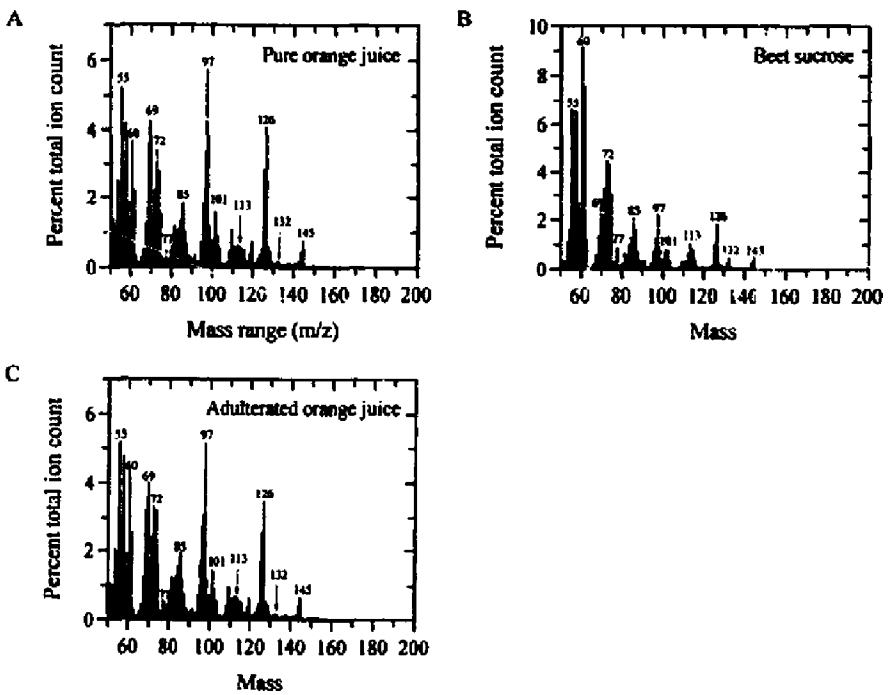

Fig. 1. Representative pyrolysis mass spectra of pure orange juice (A), beet sucrose (B) and orange juice adulterated with 10% beet sucrose solution (C).

Table 1
The partitioning of the pyrolysis mass spectral data into the training, cross validation and test sets

Set	Members[a]
Training	0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20
Cross validation	1, 3, 5, 7, 9, 11, 13, 15, 17, 19
Test	0.5, 1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.5, 8.5, 9.5, 10.5, 11.5, 12.5, 13.5, 14.5, 15.5, 16.5, 17.5, 18.5, 19.5

[a]Expressed as % sucrose solution adulteration.

2.4. Creation of training, calibration and test data sets

It is known [28,31,43,45–47] that supervised learning methods such as neural networks and partial least squares can over-fit data. For example, an over-trained neural network has usually learnt perfectly the stimulus patterns it has seen but can not give accurate predictions for unseen stimuli, i.e., it is no longer able to generalise. For supervised learning methods accurately to learn and predict the concentrations of determinands in biological systems the model must obviously be calibrated to the correct point.

For the various supervised learning methods used to analyse these pyrolysis mass spectral data, the data were split into three sets (Table 1): (1) data used to calibrate the model; (2) data employed to cross-validate the model; and (3) spectra whose determinand concentration was 'unknown' and used to test the 'calibrated' system. It is important that the training data encompasses the full range under study [29,47] since, although supervised methods are excellent at being able to interpolate, they will give poor estimates outside their knowledge realm, i.e., they can not extrapolate sufficiently well.

During calibration the models were interrogated with both the training and the cross validation set and the root mean squared (RMS) error between the output seen and that expected was calculated, thus allowing two calibration curves for the training and cross-validation sets to be drawn. When the error on the cross-validation data was lowest the system was deemed to have reached the best generalisation point and was then challenged with stimuli (i.e., pyrolysis mass spectra) whose determinand concentrations were 'unknown'.

2.5. Multilayer perceptrons (MLPs)

All MLP analyses (also known as back-propagation artificial neural networks (ANNs)) were carried out with a user-friendly, neural network simulation program, NeuFrame version 1,1,0,0 (Neural Computer Sciences, Lulworth Business Centre, Nutwood Way, Totton, Southampton, Hants), which runs under Microsoft Windows NT on an IBM-compatible PC. In-depth descriptions of the *modus operandi* of this type of MLP analysis are given elsewhere [25,31,34].

The structure of the MLP used in this study to analyse pyrolysis mass spectra consisted of three layers containing 159 processing nodes (neurons or units) made up of the 150 input nodes (normalised pyrolysis mass spectra), 1 output node (amount of determinand), and one 'hidden' layer containing 8 nodes (i.e., a 150-8-1 architecture). Each of the 150 input nodes was connected to the 8 nodes of the hidden layer using abstract interconnections (connections or synapses) (see Fig. 2 for a diagrammatic representation). Connections each have an associated real value, termed the weight (w_i), that scales the input (i_i) passing through them, this also includes the bias (ϑ), which also has a modifiable weight. Nodes sum the signals feeding to them (Net):

$$\text{Net} = i_1 w_1 + i_2 w_2 + i_3 w_3 + \cdots + i_i w_i + \cdots + i_n w_n = \sum_{i=1}^{n} i_i w_i + \vartheta$$

The sum of the scaled inputs and the node's bias, are then scaled to lie between 0 and +1 by an activation function to give the nodes output (Out); this scaling is typically achieved using a logistic 'squashing' (or sigmoidal) function:

$$\text{Out} = \frac{1}{(1 + \exp^{-Net})}$$

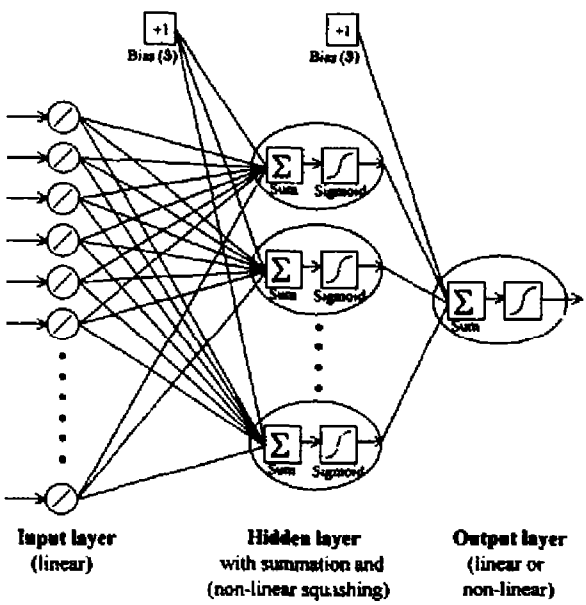

Fig. 2. A multilayer perceptron neural network consisting of an input layer (150 masses) connected to a single node in the output layer (determinand concentration; percentage sucrose) by 1 hidden layer. In the architecture shown, adjacent layers of the network are fully interconnected although other architectures are possible. Nodes in the hidden and output layers consist of processing elements which sum the input applied to it and scale the signal using a sigmoidal logistic squashing function.

These signals (Out) are then passed to the output node which sums them and in turn squashed by the above logistic sigmoidal activation function; the product of this node was then feed to the 'outside world'.

Before training commenced the values applied to the input and output nodes were normalised between 0.1 and 0.9. The scaling regime used for the input layer was to scale nodally, where the input nodes were scaled for each input node such that the lowest mass was set to 0.1 and the highest mass to 0.9. Finally, the connection weights were set to small random values (typically between -0.005 and $+0.005$).

The algorithm used to train the neural network was the standard back-propagation (BP) [45,48–50]. For the training of the MLP each input (i.e., normalised pyrolysis mass spectrum) is paired with a desired output (i.e., the percentage of the sucrose solution added to the pure orange juice, the determinand); together these are called a training pair (or training pattern). A MLP is trained over a number of training pairs; this group is collectively called the training set. The input is applied to the network, which is allowed to run until an output is produced at each output node. The differences between the actual and the desired output, taken over the entire training set are fed back through the network in the reverse direction to signal flow (hence back-propagation) modifying the weights as they go. This process is repeated until a suitable level of error is achieved. In the present work, a learning rate of 0.1 and a momentum of 0.9 were used.

Each epoch represented 1217 connection weight updatings and a recalculation of the RMS error between the true and desired outputs over the entire training set (RMS error of formation; RMSEF). During training a plot of the error versus the number of epochs represents the 'learning curve', and may be used to estimate the extent of training. In addition a second curve for the cross-validation set was calculated and training may be said to have finished when the network has found the lowest error for the cross-validation data. Provided the network has not become stuck in a local minimum, this point is referred to as the global minimum on the error surface.

Finally after training all pyrolysis mass spectra of the mixtures were used as the 'unknown' inputs (test data); the network then output its estimate (best fit) in terms of the percentage sucrose solution added to pure orange juice.

2.6. Radial basis function neural networks (RBFs)

All RBF analyses were carried out with a user-friendly, neural network simulation program, NeuralDesk version 2.10 (Neural Computer Sciences), which runs under Microsoft Windows NT on an IBM-compatible PC as detailed specifically by [51].

RBFs are hybrid neural networks encompassing both unsupervised and supervised learning [46,51–56]. RBFs are typically three-layer neural networks and in essence the sigmoidal squashing function is replaced by non-linear (often Gaussian or 'Mexican hat') basis functions or kernels (Fig. 3). The kernel is the function that determines the output of each node in the hidden layer when an input pattern is

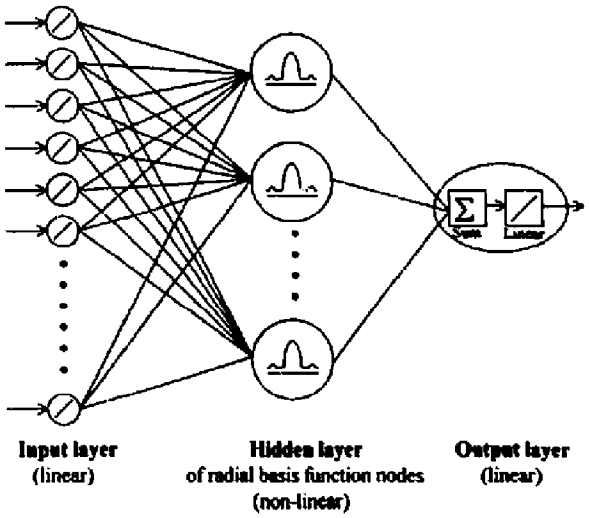

Fig. 3. Radial basis function neural net consisting of an input layer (150 masses) connected to a single node in the output layer (determinand concentration; percentage sucrose) by 1 hidden layer. The hidden layer consists of radially-symmetric Mexican hat functions, although others exist (e.g., gaussian and thin plate splines).

applied to it. This output is simply a function of the Euclidean distance from the kernel centre to the presented input pattern in the multi-dimensional space, and each node in the hidden layer only produces an output when the input applied is within its receptive field; if the input is beyond this receptive field the output is 0. This receptive field can be chosen and is radially symmetric around the kernel centre. Between them the receptive fields cover the entire region of the input space in which a multivariate input pattern may occur; a diagrammatic representation of this is shown in Fig. 4, where a two dimensional input is mapped by seven radially-symmetric basis functions. This is a fundamentally different approach from the MLP, in which each hidden node represents a non-linear hyperplanar decision boundary bisecting the input space (Fig. 4).

The outputs of the RBF nodes in the hidden layer are then fed forward via weighted connections to the nodes in the output layer in a similar fashion to the MLP, and each output node calculates a weighted sum of the outputs from the non-linear transfer from the kernels in the hidden layer. The only difference is that the output nodes of an RBF network are linear, whilst those of the MLP more typically employ a logistic (non-linear) squashing function.

The implementation of these RBF neural networks is exactly as described by Saha and Keller, [51]. Briefly the training proceeds in two stages:

(1) The first involves unsupervised clustering of the input data, typically using the *K*-means clustering algorithm [18,55,57] to divide the high-dimensional input data into clusters. Next, kernel centres are placed at the mean of each cluster of data points. The use of *K*-means is particularly useful because it positions the

kernels relative to the density of the input data points. Next the receptive field is determined by the nearest neighbour heuristic where r_j (the radius of kernel j) is set to the Euclidean distance between w_j (the vector determining the centre for the j^{th} RBF) and its nearest neighbour (k), and an overlap constant (Overlap) is used:

$$r_j = \text{Overlap} \times \min(\|w_j - w_k\|)$$

where $\|...\|$ denotes a vector norm, or Euclidean distance.

The overlap that gave best results was found to be 2, which means that the edge of the radius of one kernel is at the centre of its nearest neighbour; this optimum was also in agreement with the studies of Saha and Keller [51].

The output from nodes in the hidden layer is dependent on the shape of the basis function and the one used was that of the Mexican hat. Thus this value (R_j) for node j when given the i^{th} input vector (i_i) can be calculated by:

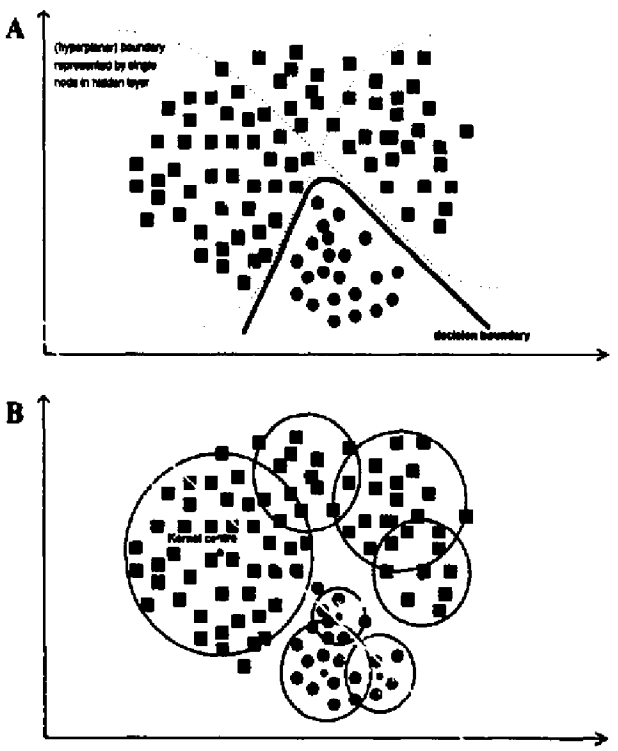

Fig. 4. (A) Typical decision boundary for a classification problem created between two data classes by a MLP with 2 nodes in the hidden layer, for 2 input nodes. Each hidden node represents a non-linear boundary and the nodes in the output layer interpolate this to form a decision boundary. (B) The same classification problem modelled by 7 radially-symmetric basis functions. The width of each kernel function (referred to as its receptive field) is determined by the local density distribution of training examples.

$$R_j(i_i) = \exp^{-(\|w_j - i_i\|/r_j)^2}$$

(2) The second stage involves supervised learning in a single layer MLP. The inputs are the output values for all n basis functions ($R_1 - R_n$) for all the training input patterns (Table 1) to that layer ($i_1 - i_n$), and the outputs are the percentage of sucrose solution added to the orange juice (0–20%).

The output nodes are trained using the standard back-propagation algorithm using gradient descent [45,48–50] by finding the weighted connections between the hidden layer and output layer that minimise the RMS error between the actual output and that known; the single output node (determinand concentration) used a linear scalar to the 'outside world'. Training was for 3×10^4 epochs, and several RBFs were trained varying in the number of kernel functions in hidden layer. Using the cross-validation regime detailed above, the optimum number of kernel functions was found by calculating the minimum error for the cross-validation set (Table 1). Finally, all pyrolysis mass spectra of the orange juice/sucrose mixtures were used as the 'unknown' inputs (test data); the network then output its estimate (best fit) in terms of the percentage sucrose solution added to pure orange juice.

2.7. Partial least squares (PLS)

All PLS analyses [43,58–62] were carried out using an in-house program, developed by Dr Alun Jones (Institute of Biological Sciences, University of Wales, Aberystwyth) which runs under Microsoft Windows NT on an IBM-compatible PC. Data were also processed prior to analysis using the Microsoft Excel 5.0 spreadsheet.

The first stage was the preparation of the data. This was achieved by presenting the 'training set' as two data matrices to the program; X, which contains the normalised triplicate pyrolysis mass spectra, and Y, which represents the amount of sucrose (0–20%) in pure orange juice. The X-data were mean centred and scaled in proportion to the reciprocal of their standard deviations.

The next stage was the generation of the calibration model. The method of validation used was full cross-validation, via the leave-one-out method [43]. This technique sequentially omits one sample from the calibration; the PLS model is then redetermined on the basis of this reduced sample set. The percentage sucrose of the omitted sample is then predicted with the use of this model. This method is required to determine the optimal size of the calibration model, so as to obtain good estimates of the precision of the multivariate calibration method (i.e., neither to under- nor over- fit predictions of unseen data) [43,63–65].

To choose the optimal number of latent variables (PLS factors) to use in predictions after the model was calibrated, the cross-validation regime detailed above was used. After validation, or tuning, the number of PLS factors used in the predictions which gave the minimum RMS error for the cross-validation set was used for the test set.

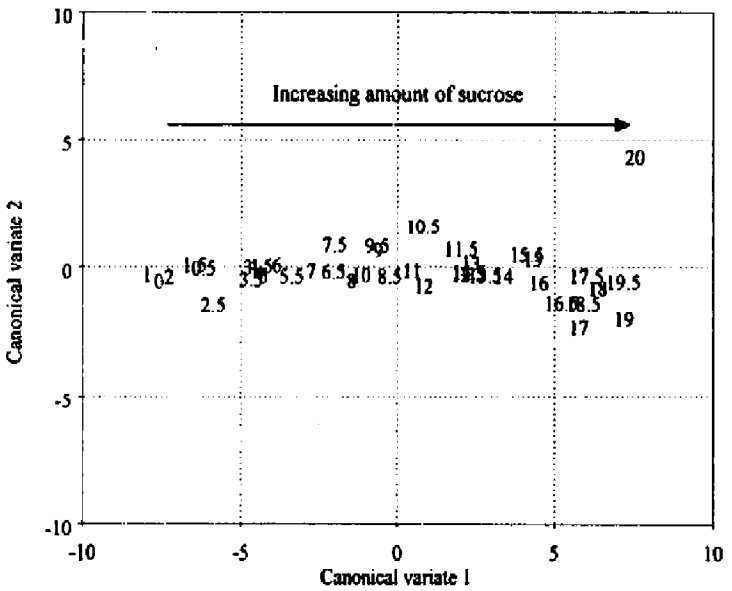

Fig. 5. Discriminant analysis plot based on PyMS data analysed by GENSTAT showing the relationship between the 41 orange juice samples. The canonical variate group means are shown; values are percent adulteration. The first 14 principal components (accounting for 99.2% of the variance) were fed into canonical variates analysis (replicates coded as groups), the first two accounted for 61.0% and 15.6% (76.6% total) of the variance, respectively.

3. Results and discussion

Pyrolysis mass spectra of pure orange juice, pure beet sucrose and orange juice adulterated with 10% beet sucrose solution are shown in Fig. 1. From the spectrum of beet sucrose (Fig. 1) one can observe the following series of peaks as being characteristic: m/z 55, 60, 69, 72, 77, 85, 97, 101, 113, 126, 132 and 145. These peaks are also seen clearly in the spectrum of pure orange juice; indeed sucrose occurs naturally in orange juice, and of the 10 g of carbohydrates per 100 g of orange juice, typically 2.9–5.6 g of this is from sucrose [2]. As expected the spectrum of an adulterated orange juice with sucrose also contains these 'sucrose peaks'. Such spectra readily illustrate the need to employ multivariate statistical techniques in the analysis of PyMS data.

The first stage was to perform discriminant analysis, as detailed above, PCA was employed as a dimensionality reduction step and 14 PCs were extracted (accounting for 99.24% of the total variance) and the resulting score vectors were subsequently used as inputs to the CVA algorithm; the resulting ordination plot is shown in Fig. 5, where only the replicate means are shown. This figure shows clearly that the first CV (which accounts for 61% of the total variance) describes the adulteration of orange juice with sucrose; that this feature is readily observed using this unsupervised feature extraction method implies that techniques using supervised learning

should be able to give an accurate quantification the levels of sucrose adulteration in orange juice.

3.1. Multilayer perceptrons (MLPs)

MLPs were trained, using the standard back-propagation algorithm, with the 11 normalized triplicate PyMS data from the training sets as the inputs, scaled for each input node [30] such that the lowest mass was set to 0.1 and the highest mass to 0.9, and the percentage of the sucrose solution (0–20%) as the output (which used a logistic squashing function), the latter being scaled between 0 and 20. Furthermore, 8 nodes were used in the single hidden layer and this topology can be represented as a 150-8-1 MLP architecture. The effectiveness of training was expressed in terms of the RMS error between the actual and desired network outputs and during training the network was interrogated with the cross-validation set of 30 (including triplicates) pyrolysis mass spectra. A plot of the learning curves for the training, cross-validation (and test) sets is shown in Fig. 6; it can be seen that whereas the learning curve of the training set continues to decrease during training the cross-validation set's learning curve initially decreases for approximately 80 epochs (indicated by the arrow) and then increases which indicates that the MLP was being over-trained. It is also noteworthy that the RMS error for the test set also reaches a minimum at this point, strongly illustrating the necessity for the use of the cross-validation procedure and that the MLP's ability to generalise to un-seen data is sufficiently good.

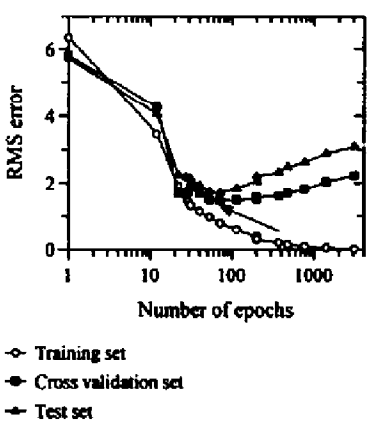

Fig. 6. Typical learning curves for the MLPs, using the standard back propagation algorithm and with one hidden layer consisting of eight nodes, trained to estimate the amount of sucrose solution in the orange juice. The open circles represent the root mean squared (RMS) error of the data used to train the neural network (the training set), the closed squares from the cross validation data set, and the partially shaded triangles the data from the test set. The arrow indicates the lowest error for the cross validation data set; this was after 80 epochs.

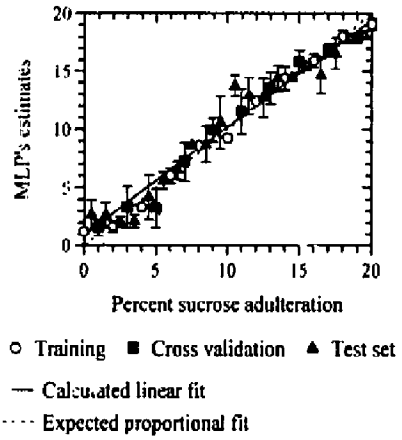

Fig. 7. The estimates of trained 150-8-1 MLPs vs. the true percentage volume of adulterant (0–20%) in orange juice. The networks were trained using the standard back propagation algorithm, for 80 epochs, to the point at which the cross validation set was optimally estimated. Open circles represent spectra that were used to train the network, closed squares the cross validation set and partially shaded triangles indicate 'unknown' spectra which were not in the training or validation sets. Data points are the averages for the three replicate samples of each concentration and error bars show standard deviations. The calculated linear fit for the test set only (bold line) and expected proportional fit (broken line) are shown.

Five MLPs were then trained in an identical fashion to that described above and training was stopped after 80 epochs. These MLPs were then interrogated with the training, cross-validation and test sets and a plot of the network's estimate versus the true amount of sucrose (Fig. 7) gave a linear fit (bold line) which was very close to the expected proportional fit (i.e. $y = x$; shown here as a broken line). For the five runs the average RMS errors for the training, cross-validation and test sets were typically 0.79, 1.39, and 1.64, respectively (total 1.40) (Table 2). It was therefore evident that the network's estimate of the quantity of sucrose adulteration in the mixtures was very similar to the true quantity, both for spectra that were used as the training and cross-validation sets and, most importantly, for the 'unknown' pyrolysis mass spectra. That all five MLPs gave very similar results indicates that training was reproducible despite the random starting weights chosen.

The training set for these MLPs contained only 33 spectra (11 samples in triplicate), and it is well known that if the number of parameters, or weights, in the calibration model is significantly higher than the number of exemplars in the training set then these methods are more prone to over-fitting [46,65]. That all five MLPs gave very similar results shows that this was not a problem; however, to obey the parsimony principle as described by Seasholtz and Kowalski [65] the next stage was to reduce the number of inputs to the MLP. PCA is an excellent dimensionality reduction technique, and the use of PC scores as inputs to neural networks, without deterioration of the calibration model, has previously been applied to the analysis of UV/visible spectroscopic data [66,67] and for the

Table 2
Comparison of the RMS errors between the predicted percentage level of adulteration with a 10% sucrose solution in orange juice and that expected using PLS, MLPs and RBFs

	PLS	PLS	MLP[b]	MLP[c]	MLP[d]	PC-MLP[e]	RBF[f]
Factors/epochs[a]	3	8	80	200	1500	2000	30 000
Training set	2.12	0.37	0.79	0.30	0.46	0.79	0.75
Cross validation set	1.73	1.11	1.39	1.56	2.05	1.65	1.50
Test set	2.27	1.26	1.64	1.82	2.68	1.61	1.74
Overall RMS error	2.12	1.05	1.40	1.48	2.16	1.45	1.48

[a]Number of epochs calculated by running five MLPs from different random starting points.
[b]150-8-1 MLP. The input layer was scaled for each input node such that the lowest mass was set to 0.1 and the highest mass to 0.9. Output node used a sigmoidal squashing function.
[c]150-3-1 MLP. The input layer was scaled for each input node such that the lowest mass was set to 0.1 and the highest mass to 0.9. Output node used a sigmoidal squashing function.
[d]150-8-1 MLP. The input layer was scaled for each input node such that the lowest mass was set to 0.1 and the highest mass to 0.9. Output node used a linear squashing function.
[e]MLP Trained with the first eight PCs, determined by running X-4-1 MLPs where $X = 1-20$ PCs. The input layer was scaled for each input node such that the lowest PC was set to 0 and the highest PC to 1.
[f]Radial basis functions were trained with 50 kernel functions, 50 was determined to be optimum by training with 0–100 kernel functions in the 150-X-1 RBFs.

identification of bacteria from their FT-IR spectra [68]. Therefore, as detailed above, the first 20 PC scores were extracted; the percentage explained variance is shown in Fig. 8 where it can be seen that as the number of PCs increases more of the variance is explained and when 20 PCs are extracted 99.58% of the total variance is explained. Between 1 and 20 PCs were used sequentially as the inputs to X-4-1 MLPs (where X = number of PCs), these were cross-validated as detailed above and the RMS for the training, cross-validation and test sets were plotted

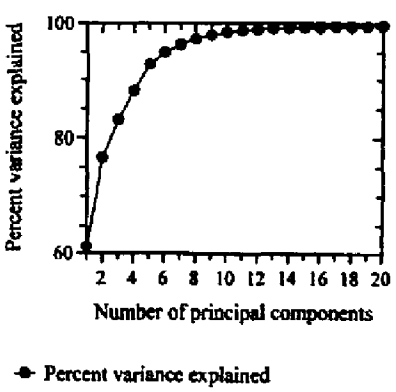

Fig. 8. Plot of the total explained variance vs. the number of principal components from dimensionality reduction of the 123 pyrolysis mass spectra collected.

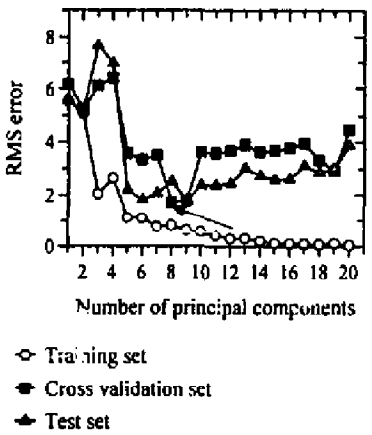

Fig. 9. Effect of the number of principal components used to represent the spectral data in the input layer of *X*-4-1 MLPs. The open circles represent the root mean squared (RMS) error of the data used to train the PLS model (the training set), the closed squares from the cross validation data set, and the partially shaded triangles the data from the test set. The arrow indicates that the optimal point was when the first 8 principal components were used (this accounted for 97.31% total varaince).

against the number of PCs used as the MLPs' inputs (Fig. 9). For the training set it is observed that as more and more PCs are used in the MLPs the error in the training set decreases. In contrast, it can be seen that when only the first few PCs are used the error in cross-validation set is high, typically above 3, the optimum is reached (as indicated by the arrow) when eight PCs are used, and inclusion of 10 or more PCs leads to the error in the cross validation set again increasing to above 3. These results show that when too few PCs are used not enough information is present to account for the sucrose addition to orange juice, and when more PCs are employed the later PCs contribute only noise to the model, thus increasing the probability of chance correlations between input and output data.

The optimal solution was when eight PCs were used as the inputs to 8-4-1 MLPs (Fig. 9) and a plot of the network's estimate versus the true amount of sucrose (Fig. 10) again gave a linear fit (bold line) which was very close to the expected proportional fit (broken line). These PC-MLPs were trained five times; very consistent results were seen in that the average RMS errors for the training, cross-validation and test sets were typically 0.79, 1.65, and 1.61, respectively (total 1.40). These values were very similar to the MLP trained using all the PyMS data (Table 2), which shows that PCA is an excellent pre-processing stage for the reduction of data prior to neural network analyses.

3.2. Radial basis function neural networks (RBFs)

The next stage was to assess the ability of RBFs to quantify the amount of sucrose addition to pure orange juice. The approach detailed above using the

unsupervised feature extraction algorithm PCA as a means for reducing the dimensionality of the mass spectral data, prior to the supervised learning involved in training a MLP, bears similarities to RBF neural networks. RBFs contains two stages: the unsupervised clustering of the mass spectra using K-means, followed by supervised learning of a single layer MLP with the outputs from the 'Mexican hat' kernel functions in the RBF's hidden layer.

RBFs were trained with the 11 normalized triplicate PyMS data from the training sets as the inputs, scaled for each input node such that the lowest mass was set to 0.1 and the highest mass to 0.9, and the percentage of the sucrose solution (0–20%) as the output (which used a linear scalar), the latter being scaled between 0 and 20. Various RBFs were trained which differed in the number of kernel functions present in their hidden layers (from 10 to 100 kernels in steps of 10). All RBFs were trained for 3×10^4 epochs. After training the RBFs were interrogated with all three data sets and the RMS error between the actual and desired outputs computed and plotted against the number of kernel functions (Fig. 11). The best prediction results for the cross-validation set were achieved with 50 functions (as indicated by the arrow). Fewer kernel functions gave poorer results indicating the inability of that number of receptive fields to adequately span the input space. When more than 50 functions were used the model was only slightly worse, indicating that very little over fitting occurred.

Since the parsimony principle [65] also applies to keeping the number of parameters to as a low number as possible whilst still being able to generalise well,

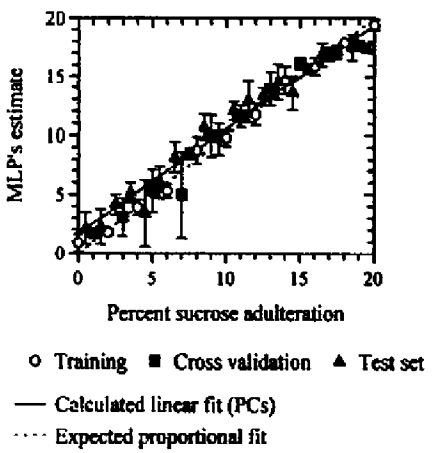

Fig. 10. The estimates of trained 8-4-1 MLPs vs. the true percentage volume of adulterant (0–20%) in orange juice. The networks were trained with the first eight principal components, using the standard back propagation algorithm, for 2×10^3 epochs, to the point at which the cross validation set was optimally estimated. Open circles represent spectra that were used to train the network, closed squares the cross validation set and partially shaded triangles indicate 'unknown' spectra which were not in the training or validation sets. Data points are the averages for the three replicate samples of each concentration and error bars show standard deviations. The calculated linear fit for the test set only (bold line) and expected proportional fit (broken line) are shown.

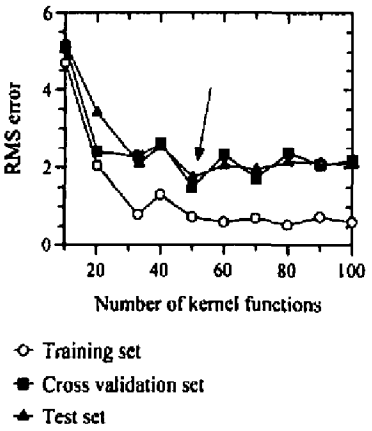

Fig. 11. Effect of varying the number of kernel functions in the hidden layer of 150-*X*-1 RBFs; 10 to 100 (in steps of 10) functions were used to find the optimum training point. Since a linear ANN was used to map the hidden layer to the output layer, training was conducted for the same number of epochs (3×10^4) for each of the RBFs. The open circles represent the root mean squared (RMS) error of the data used to train the PLS model (the training set), the closed squares from the cross validation data set, and the partially shaded triangles the data from the test set. The arrow indicates the lowest error for the cross validation data set, this was with 50 functions.

50 kernel functions were thought to be most suitable. To test reproducibility in training a total of five 150-50-1 RBFs were trained for 3×10^4 epochs. All RBFs had very similar predictive power for assessing the level of sucrose in orange juice and a plot of the RBF's estimate versus the true amount of sucrose (Fig. 12) again gave a linear fit (bold line) which was very close to the expected proportional fit (broken line). The average RMS errors for the training, cross-validation and test sets were 0.75, 1.50, and 1.74, respectively (total 1.48) and very similar to the MLP trained using all the PyMS data (Table 2) using a logistic squashing function.

Although these results are comparable to those obtained using the MLPs, the scaling function on the output node clearly is not. Therefore other MLPs identical to the ones used above were trained using a linear scalar rather than a logistic squashing function on the output node. Five MLPs were trained for 1500 epochs using cross-validation and interrogated with all the mass spectral data. The results obtained were similar and the average RMS errors for the training, cross-validation and test sets were 0.46, 2.05, and 2.68, respectively (total 2.16). These predictions were slightly poorer than those obtained using the other MLP and RBF (Table 2) and it is possible that this is a consequence of the linear scalar not being bounded (the sigmoidal squashing function was bounded between 0 and 1).

3.3. Partial least squares (PLS)

In further studies, another supervised learning method, partial least squares (PLS), which employs multivariate linear regression, was also applied to these data

using the same training, cross-validation and test sets as used for the above non-linear neural network-based analyses. Again, cross-validation was used so as to assure that the calibration models constructed by PLS was not over-fitting these data. Since there were 33 samples in the training set, between 1 and 33 PLS factors (latent variables) were used to construct models; these were then challenged with all the data and a plot of the calibration curves for the training, cross-validation (and test) sets is shown in Fig. 13. These calibration curves show that whereas the training set continues to decrease when more and more latent variables are used that the cross-validation set's calibration curve has two minima at 3 and 8 factors (as indicated by the arrows); when greater than 8 factors are used the RMS error increases slightly, indicating that at least some overfitting was occurring. Although the RMS errors for PLS models challenged with the cross-validation data were similar (when 3 factors were used this error was 1.73 compared with 1.11 when 8 latent variables were employed (Table 2)), plots of the PLS model's estimate versus the true amount of sucrose (Fig. 14) shows that the model using 8 factors (Fig. 14(b)) was much better. In addition, the slope of the best fit lines shown in Fig. 14 was 0.95 (intercept 0.82) when 8 factors were used compared with only 0.72 (intercept 2.87) when 3 were employed.

The nodes in the hidden layers of MLPs may be considered as sets of intermediate analogues to the latent variables in linear regression such as PLS [69]. If true then the optimum number of latent variables used to calibrate a PLS model might also approximate the optimum number of nodes to have in the hidden layer of a

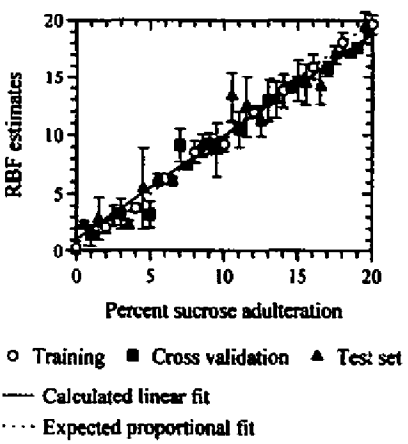

Fig. 12. The estimates of trained 150-50-1 RBFs vs. the true percentage volume of adulterant (0–20%) in orange juice. The RBF networks were trained for 3×10^4 epochs. Open circles represent spectra that were used to train the network, closed squares the cross validation set and partially shaded triangles indicate 'unknown' spectra which were not in the training or validation sets. Data points are the averages for the three replicate samples of each concentration and error bars show standard deviations. The calculated linear fit for the test set only (bold line) and expected proportional fit (broken line) are shown.

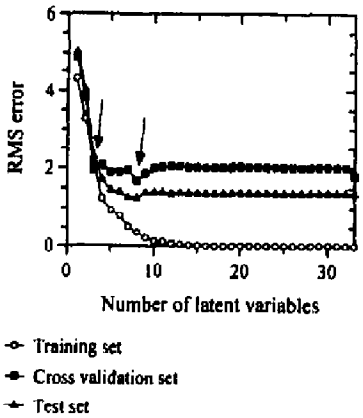

Fig. 13. Calibration curves for the PLS models, trained to estimate the amount of sucrose solution in the orange juice. The open circles represent the root mean squared (RMS) error of the data used to train the PLS model (the training set), the closed squares from the cross validation data set, and the partially shaded triangles the data from the test set. The arrows indicates two possible stopping optima using 3 or 8 latent variables.

three layer MLP, provided that the mapping between the inputs (X-data) and outputs (*Y*-data) was more-or-less linear. That PLS gave better estimates than the 150-8-1 MLPs (Table 2) would tend to suggest that the mapping of sucrose levels

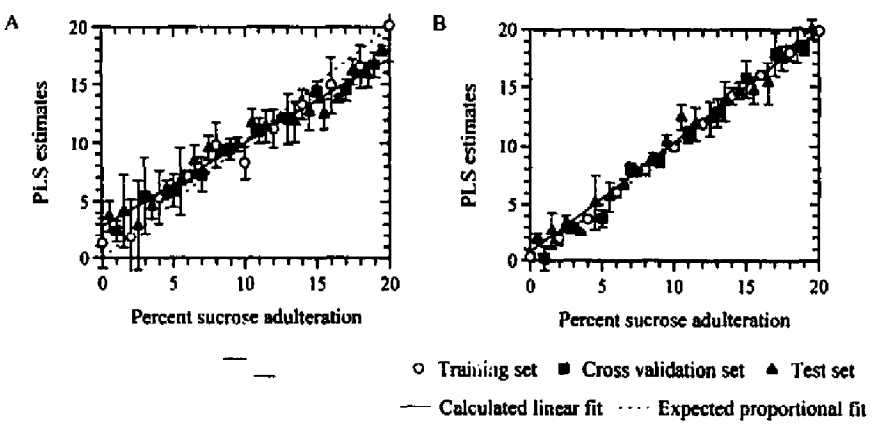

Fig. 14. The estimates of calibrated PLS models vs. the true percentage volume of adulterant (0–20%) in orange juice. Models were created using 3 (A) and 8 (B) latent variables. Open circles represent spectra that were used to calibrated the PLS model, closed squares the cross validation set and partially shaded triangles indicate 'unknown' spectra which were not in the training or validation sets. Data points are the averages for the three replicate samples of each concentration and error bars show standard deviations. The calculated linear fit for the test set only (bold line) and expected proportional fit (broken line) are shown.

in orange juice was linear. Indeed the first canonical variate (Fig. 5) did seem to describe the influence of sucrose in the mass spectra.

Further MLPs were set up which employed only 3 nodes in the hidden layer rather than 8, representing a 150-3-1 architecture. The optimum training point was found using cross-validation, and the average RMS errors for the training, cross-validation and test sets were 0.40, 1.56, and 1.82, respectively (total 1.48). These were slightly worse than those for 150-8-1 MLPs but considerably better than the PLS model employing only 3 latent variables (Table 2). These results tend to indicate that although 8 nodes were better than 3 there was very little deterioration in the model. Whilst the use of more factors in PLS predictions can exacerbate the chances of overfitting [43,65], the requirement for more factors in the optimal PLS mode, a phenomenon that has however been seen previously [31,34], usually implies that there are at least some non-linear relationships within the pyrolysis mass spectral data [43]. It is therefore perhaps not surprising that the 150-3-1 MLP was able to predict the levels of sucrose more accurately that a PLS model employing 3 latent variables, since the nodes in the hidden layer use a non-linear squashing function and each hidden node represents a non-linear hyperplanar decision boundary bisecting the input space (Fig. 4a).

4. Conclusion

Although the pyrolysis mass spectra of pure orange juice contained peaks that were qualitatively characteristic of pure beet sucrose (Fig. 1) when discriminant analyses was used to analyse a series of 41 binary mixtures of orange juice containing levels of sucrose ranging from 0–20% of 10% solution, the quantitative MS profiles characteristic of sucrose were observed as the most important feature in this series and were extracted in the first canonical variate (which accounted for 61% of the total variance in the data).

Neural cognition-based methods of MLPs and RBFs and the linear regression technique of PLS were employed successfully for the quantitative deconvolution of these pyrolysis mass spectra. It was found that each of the methods could be used to provide calibration models which gave excellent predictions for the percentage adulteration of orange juice with sucrose; for the test set samples which they had not been trained these were between $\pm 1.3\%$ and $\pm 2.7\%$, and the limit of detection was $< 1\%$ which equates to 1 g l^{-1} of added sucrose. PLS, using 8 latent variables for predictions, gave the best results and typical RMS errors for the training, cross-validation and test sets were 0.37, 1.11 and 1.26, respectively.

The inputs to MLPs were also reduced using PCA and it was found that the data could be reduced from 150 masses to 8 PC scores without any deterioration of the accuracy of the model to predict the level of sucrose adulteration. This highlights that PCA is an excellent pre-processing step which also has the potential to speed up neural network learning since there are fewer weights to update.

PyMS is a physico-chemical method which has been extensively exploited for whole-organism fingerprinting [20,70]. Other spectroscopic techniques which have

also been used for microbial identification include UV resonance Raman spectroscopy [71,72] and Fourier transform infrared spectroscopy (FT-IR) [68,73,74]. These methods all produce complex reproducible biochemical fingerprints which are qualitatively distinct for different samples and quantitative in respect of target determinands; indeed FT-IR has been exploited recently within the food area for the authentication of vegetable oils [75] and fruit purees [76], whilst Raman spectroscopy has also been investigated for the analysis of foods [77–82].

The combination of PyMS and neural networks has been shown previously to be an excellent technique capable of the exquisitely sensitive qualitative assessment of the adulteration of extra virgin olive oils with various seed oils [21,22], and recent work has also shown that this is also possible to measure the level of adulteration quantitatively at levels below 3% [83], whilst other olive oil studies have concentrated on regional classification [84]. Other quantitative studies using PyMS and chemometrics have also shown that it is possible to assess of the adulteration of goats' or ewes' milk with cows' milk to below 1% [85] and to measure fat content in milk [86]. Therefore in conclusion, since any foodstuff can be pyrolysed in this way, the combination of PyMS with supervised learning may be seen to constitute a rapid, powerful and novel approach to the qualitative and quantitative assessment of food adulteration generally.

Acknowledgements

We would like to thank David Broadhurst for useful discussions. R.G. thanks the Wellcome Trust for financial support (grant number 042615/Z/94/Z). D.B.K. is indebted to the Chemicals and Pharmaceuticals Directorate of the UK BBSRC for financial support.

References

[1] T. Patel, New Scientist, 142 (1994) 26–29.
[2] K. Robards and M. Antolovich, Analyst, 120 (1995) 1–28.
[3] W. Simpkins and M. Harrison, Trends Food Sci. Technol., 6 (1995) 321–328.
[4] N.H. Low, J. AOAC Int., 79 (1996) 724–737.
[5] G.G. Wudrich, S. McSheffrey and N.H. Low, J. AOAC Int., 76 (1993) 342–354.
[6] V. Cornil, W. Coghe and P. Sandra, J. High Res. Chromat., 18 (1995) 286–288.
[7] W.J. Li, P. Goovaerts and M. Meurens, J. Agric. Food Chem., 44 (1996) 2252–2259.
[8] J.T.W.E. Vogels, L. Terwel, A.C. Tas, F. van den Berg, F. Dukel and J. van der Greef, J. Agric. Food Chem., 44 (1996) 175–180.
[9] G.G. Martin, V. Hanote, M. Lees and Y.L. Martin, J. AOAC Int., 79 (1996) 62–72.
[10] D.W. Lawlor, Photosynthesis: Metabolism, Control and Physiology, Longman, Harlow, UK, 1987.
[11] W.J. Irwin. Analytical Pyrolysis: A Comprehensive Guide, Marcel Dekker, New York, 1982.
[12] W. Windig, P.G. Kistemaker, J. Haverkamp and H.L.C. Meuzelaar, J. Anal. Appl. Pyrol., 2 (1980) 7–18.
[13] R. Goodacre, Ph.D. Thesis, University of Bristol, 1992.

[14] H.L.C. Meuzelaar, J. Haverkamp and F.D. Hileman, Pyrolysis Mass Spectrometry of Recent and Fossil Biomaterials, Elsevier, Amsterdam, 1982.
[15] R.E. Aries, C.S. Gutteridge and R. Evans, J. Food Sci., 51 (1986) 1183–1186.
[16] K.J.G. Reid, J.S. Swan and C.S. Gutteridge, J. Anal. Appl. Pyrol., 25 (1993) 49–62.
[17] R.I. Aylott, A.H. Clyne, A.P. Fox and D.A. Walker, Analyst, 119 (1994) 1741–1746.
[18] B.S. Everitt, Cluster Analysis, E. Arnold, London, 1993.
[19] R. Goodacre, Microbiol. Eur., 2 (1994) 16–22.
[20] R. Goodacre and D.B. Kell, Cur. Opin. Biotechnol., 7 (1996) 20–28.
[21] R. Goodacre, D.B. Kell and G. Bianchi, Nature, 359 (1992) 594–594.
[22] R. Goodacre, D.B. Kell and G. Bianchi, J. Sci. Food Agric., 63 (1993) 297–307.
[23] J. Chun, E. Atalan, S.B. Kim, H.J. Kim, M.E. Hamid, M.E. Trujillo, J.G. Magee, G.P. Manfio, A.C. Ward and M. Goodfellow, FEMS Microbiol. Lett., 114 (1993) 115–119.
[24] R. Freeman, R. Goodacre, P.R. Sisson, J.G. Magee, A.C. Ward and N.F. Lightfoot, J. Med. Microbiol., 40 (1994) 170–173.
[25] R. Goodacre, M.J. Neal, D.B. Kell, L.W. Greenham, W.C. Noble and R.G. Harvey, J. Appl. Bacteriol., 76 (1994) 124–134.
[26] R. Goodacre, S.J. Hiom, S.L. Cheeseman, D. Murdoch, A.J. Weightman and W.G. Wade, Curr. Microbiol., 32 (1996) 77–84.
[27] R. Goodacre, M.J. Neal and D.B. Kell, Zentrabl. Bakteriol., 284 (1996) 516–539.
[28] R. Goodacre and D.B. Kell, Anal. Chim. Acta, 279 (1993) 17–26.
[29] R. Goodacre, A.N. Edmonds and D.B. Kell, J. Anal. Appl. Pyrol., 26 (1993) 93–114.
[30] M.J. Neal, R. Goodacre and D.B. Kell, Proc. World Congr. Neural Networks, San Diego, International Neural Network Society, 1, 1994, 1318–1323.
[31] R. Goodacre, M.J. Neal and D.B. Kell, Anal. Chem., 66 (1994) 1070–1085.
[32] R. Goodacre, A. Karim, M.A. Kaderbhai and D.B. Kell, J. Biotechnol., 34 (1994) 185–193.
[33] R. Goodacre, S. Trew, C. Wrigley-Jones, M.J. Neal, J. Maddock, T.W. Ottley, N. Porter and D.B. Kell, Biotechnol. Bioeng., 44 (1994) 1205–1216.
[34] R. Goodacre, S. Trew, C. Wrigley-Jones, G. Saunders, M.J. Neal, N. Porter and D.B. Kell, Anal. Chim. Acta, 313 (1995) 25–43.
[35] H.J.H. MacFie, C.S. Gutteridge and J.R. Norris, J. Gen. Microbiol., 104 (1978) 67–74.
[36] W. Windig, J. Haverkamp and P.G. Kistemaker, Anal. Chem., 55 (1983) 81–88.
[37] B.F.J. Manly, Multivariate Statistical Methods: A Primer, Chapman and Hall, London, 1994.
[38] C. Chatfield and A.J. Collins, Introduction to Multivariate Analysis, Chapman and Hall, London, 1980.
[39] C.S. Gutteridge, L. Vallis and H.J.H. MacFie, in M. Goodfellow, D. Jones and F. Priest (Eds.), Computer-assisted Bacterial Systematics, Academic Press, London, 1985, pp. 369–401.
[40] I.T. Jolliffe, Principal Component Analysis, Springer, New York, 1986.
[41] D.R. Causton, A Biologist's Advanced Mathematics, Allen and Unwin, London, 1987.
[42] B. Flury and H. Riedwyl, Multivariate Statistics: A Practical Approach, Chapman and Hall, London, 1988.
[43] H. Martens and T. Næs, Multivariate Calibration, Wiley, Chichester, 1989.
[44] J.A. Nelder, Genstat Reference Manual, Scientific and Social Service Program Library, University of Edinburgh, 1979.
[45] P.D. Wasserman, Neural Computing: Theory and Practice, Van Nostrand Reinhold, New York, 1989.
[46] C.M. Bishop, Neural Networks for Pattern Recognition, Clarendon, Oxford, 1995.
[47] D.B. Kell and B. Sonnleitner, Trends Biotechnol., 13 (1995) 481–492.
[48] D.E. Rumelhart, J.L. McClelland and The PDP Research Group, Parallel Distributed Processing, Experiments in the Microstructure of Cognition, MIT Press, Cambridge, Mass., 1986.
[49] P.J. Werbos, The Roots of Back-propagation: From Ordered Derivatives to Neural Networks and Political Forecasting, Wiley, Chichester, 1994.
[50] S.S. Haykin, Neural Networks: A Comprehensive Foundation, Macmillan, New York, 1994.
[51] A. Saha and J.D. Keller, in D. Touretzky (Eds.), Advances in Neural Information Processing Sytems, Morgan Kaufmann, 1990, p. 482.

[52] J. Moody and C.J. Darken, Neural Computation, 1 (1989) 281–294.
[53] R. Beale and T. Jackson, Neural Computing: An Introduction, Adam Hilger, Bristol, 1990.
[54] J. Park and J.W. Sandberg, Neural Computation, 3 (1991) 246–257.
[55] D.R. Hush and B.G. Horne, IEEE Signal Processing Mag., 10 (1993) 8–39.
[56] M.F. Wilkins, C.W. Morris and L. Boddy, Comput. Appl. Biosci., 10 (1994) 285–294.
[57] R.O. Duda and P.E. Hart, Pattern Classification and Scene Analysis, Wiley, New York, 1973.
[58] H. Martens and T. Næs, in P. Williams and K. Norris (Eds.), Near-Infrared Technology in the Agriculture and Food Industries, Ammerican Association of Cereal Chemists, St. Paul, Minnesota, 1987.
[59] R.G. Brereton, Multivariate Pattern Recognition in Chemometrics, Elsevier, Amsterdam, 1992.
[60] K.A. Martin, Appl. Spectrosc. Revs., 27 (1992) 325–383.
[61] Y.Z. Liang, O.M. Kvalheim and R. Manne, Chem. Intell. Lab. Sys., 18 (1993) 235–250.
[62] Y.Z. Liang and O.M. Kvalheim, Chem. Intell. Lab. Sys., 32 (1996) 1–10.
[63] D.M. Haaland and E.V. Thomas, Anal. Chem., 60 (1988) 1193–1202.
[64] P.J. Brown, J. Chemom., 6 (1992) 151–161.
[65] M.B. Seasholtz and B. Kowalski, Anal. Chim. Acta, 277 (1993) 165–177.
[66] P.J. Gemperline, J.R. Long and V.G. Gregoriou, Anal. Chem., 63 (1991) 2313–2323.
[67] M. Blanco, J. Coello, H. Iturriaga, S. Maspoch and M. Redon, Anal. Chem., 67 (1995) 4477–4483.
[68] R. Goodacre, E.M. Timmins, P.J. Rooney, J.J. Rowland and D.B. Kell, FEMS Microbiol. Lett., 140 (1996) 233–239.
[69] T. Næs, K. Kvaal, T. Isaksson and C. Miller, J. Near Infrared Spec., 1 (1993) 1–11.
[70] J.T. Magee, in M. Goodfellow and A.G. O'Donnell (Eds.), Handbook of New Bacterial Systematics, Academic Press, London, 1993, pp. 383–427.
[71] W.H. Nelson and J.F. Sperry, in W.H. Nelson (Eds.), Modern techniques for rapid microbiological analysis, VCH, New York, 1991, pp. 97–143.
[72] W.H. Nelson, R. Manoharan and J.F. Sperry, Appl. Spectrosc. Revs., 27 (1992) 67–124.
[73] D. Naumann, D. Helm and H. Labischinski, Nature, 351 (1991) 81–82.
[74] D. Naumann, D. Helm, H. Labischinski and P. Giesbrecht, in W.H. Nelson (Eds.), Modern Techniques for Rapid Microbiological Analysis, VCH, New York, 1991, pp. 43–96.
[75] Y.W. Lai, K. Kemsley and R.H. Wilson, J. Agric. Food Chem., 42 (1994) 1154–1159.
[76] M. Defernez, E.K. Kemsley and R.H. Wilson, J. Agric. Food Chem., 43 (1995) 109–113.
[77] Y. Ozaki, R. Cho, K. Ikegaya, S. Muraishi and K. Kawauchi, Appl. Spectrosc., 46 (1992) 1503–1507.
[78] E. Li-chan, Trends Food Sci. Technol., 5 (1994) 3–11.
[79] H. Sadeghi Jorabchi, R.H. Wilson, P.S. Belton, J.D. Edwards Webb and D.T. Coxon, Spectrochim. Acta A, 47 (1991) 1449–1458.
[80] H. Sadeghi Jorabchi, R.H. Wilson, P.S. Belton, J.D. Edwardswebb and D.T. Coxon, Spectrochim. Acta A—Mol. Spectrosc., 47 (1991) 1449–1458.
[81] V. Baeten, M. Meurens, M.T. Morales and R. Aparicio, J. Agric. Food Chem., 44 (1996) 2225–2230.
[82] H. Sadeghi Jorabchi, P.J. Hendra, R.H. Wilson and P.S. Belton, J. Am. Oil Chemists Soc., 67 (1990) 483–486.
[83] G.J. Salter, R. Goodacre, D.B. Kell, M. Lazzari and G. Bianchi, Food Authenticity '96, University of East Anglia, 1996, pp 6.
[84] G.J. Salter, M. Lazzari, L. Giansante, R. Goodacre, A. Jones, G. Surricchio, D.B. Kell and G. Bianchi, J. Anal. Appl. Pyrol., this Vol. (1997).
[85] R. Goodacre, Appl. Spectrosc., (1997) in press.
[86] R. Goodacre and D.B. Kell, Food Authenticity '96, University of East Anglia, 1996, pp 8.