ELSEVIER

Analytica Chimica Acta 368 (1998) 29–44

ANALYTICA CHIMICA ACTA

Variable selection in wavelet regression models

Bjørn K. Alsberg[a,*], Andrew M. Woodward[a], Michael K. Winson[a], Jem J. Rowland[b], Douglas B. Kell[a]

[a]*Institute of Biological Sciences, University of Wales, Aberystwyth, Ceredigion SY23 3DD, UK*
[b]*Department of Computer Science, University of Wales, Aberystwyth, Ceredigion SY23 3DD, UK*

Received 27 October 1997; received in revised form 10 February 1998; accepted 11 February 1998

Abstract

Variable selection and compression are often used to produce more parsimonious regression models. But when they are applied directly to the original spectrum domain, it is not easy to determine the *type* of feature the selected variables represent. By performing variable selection in the wavelet domain we show that it is possible to identify important variables as being part of short- or large-scale features. Therefore, the suggested method is to extract information about the selected variables that otherwise would have been inaccessible. We are also able to obtain information about the location of these features in the original domain. In this article we demonstrate three types of variable selection methods applied to the wavelet domain: selection of optimal combination of scales, thresholding based on mutual information and truncation of weight vectors in the partial least squares (PLS) regression algorithm. We found that truncation of weight vectors in PLS was the most effective method for selecting variables. For the two experimental data sets tested we obtained approximately the same prediction error using less than 1% (for Data set 1) and 10% (for Data set 2) of the original variables. We also discovered that the selected variables were restricted to a limited number of wavelet scales. This information can be used to suggest whether the underlying features may be dominated by narrow (selective) peaks (indicated by variables in short wavelet scale regions) or by broader regions (indicated by variables in long wavelet scale regions). Thus, wavelet regression is here used as an extension of the more traditional Fourier regression (where the modelling is performed in the frequency domain without taking into consideration any of the information in the time domain).

Keywords: Wavelet regression; Multivariate calibration; Partial least squares; Infrared spectra; Feature selection; Variable selection; Mutual information; Scalogram; Feature extraction

1. Introduction

The rapid, quantitative and qualitative information obtained from applying multivariate methods to spectra (e.g. infrared, Raman, UV) is an effective alternative to using slow wet chemical analyses in screening experiments [1–11]. In particular, we are interested in the determination of the concentrations of important compounds produced by industrially interesting bacteria and yeasts. The methods usually employed for solving such calibration problems are partial least squares (PLS) regression [12–17] or artificial neural networks (ANNs) [18–25]. We will, in this paper, concentrate on the application of PLS regression methods since these methods give rise to models that are easier to interpret than neural

*Corresponding author. E-mail: bka@aber.ac.uk

0003-2670/98/$19.00
PII S0003-2670(98)00194-9

networks and they are significantly less computer-demanding.

To improve understanding and prediction of regression models, the application of various transforms on the original variables is often necessary. In particular, the use of Fourier transform pre-processing enables the user to represent a regression model in terms of frequency domain variables rather than time domain. This is referred to as *Fourier regression.* In a similar fashion we present the application of the wavelet transform [26–44] as a pre-processing step prior to any type of regression, an approach we have chosen to call *wavelet regression*.

The basis for our approach is the concept of *multi-resolution*, i.e., the fact that phenomena can exist at different scales or levels of detail. An infrared spectrum, for instance, can be described in terms of features at different scales: the fingerprint region around 800–200 cm^{-1} is a good example of a region which requires a high resolution and much detail. In contrast, at the other end of the spectrum, around 4000–3000 cm^{-1}, we often observe broad features due to various types of hydrogen bonding that form a continuum of vibrational frequencies. Important features in this region will require less resolution and detail. When we apply regression methods to raw spectra in general, the final regression model is based on the *highest resolution* level only. This means that it is sometimes difficult to detect dependencies between the spectrum space and, e.g., the concentration space of a compound which originate at different scales. By using wavelet regression it is possible to analyse the regression model at the different scales separately and to investigate the contribution of each scale to the final regression model. We suggest here that this approach can increase the interpretability of parsimonious regression models.

2. Wavelet theory

2.1. Introduction to wavelets

Wavelets are becoming an increasingly important tool in image and signal processing. Wavelets are effective in extracting both time and frequency-like information from a time-varying signal. The Short-Time Fourier Transform (STFT) performs a constant bandwidth splitting of the signal whereas the wavelet transform (WT) has a proportional (octave) bandwidth splitting of the frequency domain. Consequently, there is a high time resolution for high frequency components and low time resolution for low frequency components. Unlike the Fourier transform, the wavelet transform can use a variety of different basis functions with different properties. One important property of wavelet basis functions is their localisation in both time and frequency domains simultaneously.

Non-orthogonal, redundant and discrete wavelet bases are referred to as *frames* [27,45,46] but will not be discussed here.

The more popular orthogonal wavelet bases have several interesting properties that make them suitable as tools in signal analysis and compression. In particular it has been possible to construct fast and efficient algorithms that enable wavelet transforms to be practical tools in signal processing.

A *continuous* wavelet decomposition can be written as

$$w(s,b) = \int |s|^{-1/2} \psi^{(s,b)}(t) f(t)\, \mathrm{d}t, \tag{1}$$

where $\psi^{(s,b)}(t) = \psi((t-b)/s)$ is the wavelet function at a particular *scale s*, i.e. the same wavelet function is dilated or contracted according to the scale and $f(t)$ is the function to be analysed. b signifies the translation of the wavelet at scale s. Eq. (1) can also be interpreted as a convolution of the signal with the wavelet function in the time domain. This interpretation is emphasised in Eq. (2) below by using the convolution operator symbol $\otimes$ and the symbol F for the Fourier transform operator. Thus, using the convolution operator symbol we rewrite Eq. (1) for the time domain as

$$w(s,b) = |s|^{-1/2} \psi^{(s,b)}(t) \otimes f(t). \tag{2}$$

In the frequency domain (after Fourier transform) Eq. (2) is written as

$$F(w(s,b)) = F(\psi^{(s,b)}(t))\, F(f(t)). \tag{3}$$

These equations are straightforward applications of the convolution theorem [47].

In the present paper only the *discrete orthogonal* wavelet transform will be used, where the choice of scales s and translations b is restricted; in general

we set

$$s = a_0^j, \tag{4}$$

$$b = ib_0a_0^j, \tag{5}$$

where a_0=2 and b_0=1 is the most common choice. i and j are indices that can be any natural number $\mathbb{Z}$. Shifts that are multiples of 2 are selected since two-channel filter banks that are downsampled by 2 are only shift-invariant with respect to even shifts. In addition, the discrete wavelet transform used here is both linear and complete (which follows from perfect reconstruction). The discrete wavelet transform also satisfies the conservation of energy (referred to as Bessel's or Parseval's equality [48]).

The *scale* can be interpreted as a measure of frequency. A short scale contains high-frequency components whereas a long scale contains low-frequency components. An intuitive way of looking at the wavelet transform is to interpret it as a successive sequence of combinations of bandpass filters. The wavelet function $\psi(t)$ (also referred to as the ''mother wavelet'') can be interpreted as a high-pass filter acting on the original signal and the *scaling function* $\phi(t)$ (also referred to as the ''father wavelet'') behaves as a low-pass filter.

The wavelet function $\psi(t)$ can be written as a linear combination of the scaling function. The scaling function has the property that it can be written in terms of scaled versions of itself:

$$\phi(t) = \sqrt{2}\sum_k c_k\phi(2t-k). \tag{6}$$

The c_k are the interscale basis coefficients which are related to the low-pass filtering in the wavelet transform algorithm. Similarly the wavelet function can be expressed as a linear combination of translates of the scaling function $\phi(2t)$:

$$\psi(t) = \sqrt{2}\sum_k d_k\phi(2t-k). \tag{7}$$

This is the fundamental wavelet equation. The coefficients d_k are related to the high-pass filter used in the wavelet transform.

A common algorithm for calculating discrete wavelet coefficients is the so-called *Mallat algorithm* [39,40,49]. At each scale high (H) and low (L) pass filters are applied to the input signal. The actual shapes of these filters are determined by the kind of wavelet function used. The output from the high-pass filter at each scale is recorded as the wavelet coefficients. The low-pass filter extracts the low frequency components for the next scale where another set of high and low-pass filters is employed. At each successive scale the length of the vector upon which the filters operate is halved; this is referred to as *decimation*. Thus, the total number of available scales is $\log_2(N)$; where N is the length of the input data vector.

The positions of the wavelet coefficients at individual scales over the original domain can be visualised if a ''stretching'' of the coefficients is performed. Since each wavelet scale in the Mallat algorithm is a subsampling of the previous scale, it is possible to decide over which region in the original domain the wavelet coefficient has its influence. The plot where the different wavelet coefficients for each scale are shown over the original domain is referred to as a *scalogram* (see Fig. 1). This is analogous to the *spectrogram* used in STFT which shows the tiling of the time-frequency domain. Each tile represents the area covered by a basis function at a certain position in time and scale (frequency for STFT). In this article we will use the shading of a tile to represent the magnitude of the associated wavelet coefficient. In all scalograms presented in this paper, the absolute value of the wavelet coefficients are shown using a grey-scale coding. Black signifies the largest absolute value and white signifies zero coefficient value. All other absolute values in between are represented by shades of grey.

2.2. The total wavelet basis matrix

The fast wavelet transform (FWT) can be formulated in terms of matrix algebra by storing each of the wavelet functions in the time domain in a matrix **B**. This is the *total wavelet basis matrix* which includes the basis functions from all the scales at all time shifts.

One sensible way of organising this matrix is to sort the sets of shifted basis functions according to their *scale*. This means that we present all the basis functions that are shifted but have the same scale followed by the next higher (or lower) scale's shifted basis functions. The special organisation presented here is not chosen arbitrarily but is closely related to how Mallat's algorithm for calculating the wavelet coeffi-

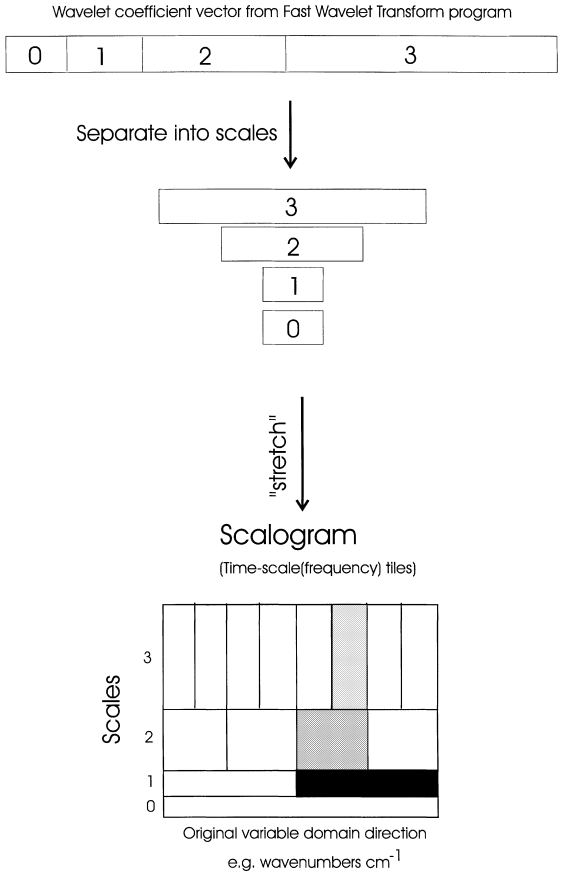

Fig. 1. Illustration of how the wavelet coefficient vector is interpreted at various scales over the original domain. Note the "stretching" process which is necessary since each scale is subsampled by two compared to the previous scale. Each tile represents the area covered by a wavelet basis function in the time-frequency domain (or rather the time-scale domain). Please note that the colour coding used here is as follows: Black is the highest absolute value of the coefficient for the tile and white represents the zero value. Absolute values in between are shown using grey shading.

cients operates. The number of shifts along the x-axis depends on the value of the scale j. Assuming that the total number of elements in our data vector is $N=2^{J+1}$ the different scales are the integers from 0 to J. The shifting coefficient k has the integer values 0 to 2^j-1 for each j value. Since each basis vector is stored in a matrix row it is necessary to make a mapping between (j,k) indices into r index of a single row vector where $r=2^j+k+1$. Note that this formula cannot be used to find which function is located in vector element $r=1$. This is a special case and corresponds to the coefficient from convolving the observed data profile with the *scaling function* (0,1) of the wavelet in question.

The basis matrix $\mathbf{B}$ is ordered according to scale as follows:

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}_0 \\ \mathbf{B}_1 \\ \vdots \\ \mathbf{B}_{J-1} \\ \mathbf{B}_J \end{bmatrix}. \tag{8}$$

Each submatrix $\mathbf{B}_j$ has a diagonal dominant structure for scale j. The largest submatrices correspond to the shortest scales (dominated by high-frequency components). Note that $\mathbf{B}$ is orthonormal and therefore can be used in the wavelet transform as follows:

$$\mathbf{z} = \mathbf{B}^{\mathrm{T}}\mathbf{x}, \tag{9}$$

where $\mathbf{z}$ is the vector of wavelet coefficients and $\mathbf{x}$ is the vector containing the input signal. The reconstruction is trivial:

$$\mathbf{x} = \mathbf{B}\mathbf{z}. \tag{10}$$

2.3. Using wavelets in regression

In *Fourier regression* a regression model is formed between the frequency components determined in a Fourier analysis and a dependent variable. Let $\mathbf{X}$ be the original data matrix with M spectra (as rows) and N wavelengths (as columns). If $\mathbf{y}$ is, e.g., the concentration vector for some chemical component, we estimate the regression coefficients $\mathbf{b}$ as follows:

$$\mathbf{b} = \mathbf{X}^{+}\mathbf{y}, \tag{11}$$

where $\mathbf{X}^{+}$ is a *generalised inverse* that originates from some regression method (e.g. PLS regression).

In Fourier regression we replace the original data matrix $\mathbf{X}$ with the projections of each spectrum onto the Fourier basis matrix, i.e., we are using the Fourier transform of each spectrum instead of the original spectrum. Assuming smoothness we usually cut off the highest-frequency components. This constitutes a de-noising step in Fourier regression. In addition, usually the magnitude spectrum and not the full complex representation is used.

The main advantage of using the magnitude spectrum representation in regression rather than the time domain representation is that the final regression model is formulated in terms of individual frequencies. For certain data sets, the frequency domain is more suitable for modelling than the time domain, and often we discover that the resulting regression models in the frequency domain have higher predictive ability and are more parsimonious. There is, however, a serious problem with the Fourier representation: changes in frequencies *over time* are not captured. A magnitude spectrum simply contains all the frequency components over the whole time domain range. Various methods have been constructed to extend the capabilities of the FT, such as the STFT where standard Fourier transforms are performed along windowed time regions. Another approach to obtain localisation in time and frequency domains is to use wavelet transforms. Analogous to Fourier regression we therefore suggest that wavelet transforms can be used as a pre-processing step before doing a regression, hereafter referred to as *wavelet regression* [26].

The wavelet transform of an 1D signal is usually coded as a vector $\mathbf{z}$ where each scale is stored sequentially. The structure of $\mathbf{z}$ as found in Eq. (9) is Scale_0 with 1 element, followed by Scale_1 with 2 elements, followed by Scale_2 with 4 elements, etc., followed by Scale_j with 2^j elements. The matrix $\mathbf{Z}$ of wavelet coefficients for each spectrum has dimensions $[M \times N]$ elements, where M is the number of spectra and N is the number of variables in each spectrum ($N=2^{J+1}$ where J is the shortest scale). Analogous to Fourier regression we obtain the wavelet regression coefficients in general as

$$\mathbf{b}_w = \mathbf{Z}^+\mathbf{y}. \tag{12}$$

There is one very interesting property of the estimated wavelet regression vector $\mathbf{b}_w$: it can itself be interpreted as a wavelet transform of a signal. *This means we can present* $\mathbf{b}_w$ *in a scalogram as we do for "ordinary" wavelet coefficient vectors and observe the regression coefficients at different scales*. It is important to keep in mind that each of the scales is valid over the whole time domain (in this article we apply the wavelet transform on the infrared wavenumber domain rather than on a true time domain). Any type of vectors related to the relationship between the original variables in a regression can be analysed in this fashion; for instance we could have investigated the scalograms of PLS loading vectors for each component. There are two types of loading matrices in PLS [50] which are usually referred to as $\mathbf{P}$ and $\mathbf{W}$. The loading weights in $\mathbf{W}$ are orthogonal, whereas the estimated loadings in $\mathbf{P}$ are in general non-orthogonal.

It is therefore possible to produce scalograms for each type of loading vector. But is it really necessary to perform a wavelet transform of each spectrum before a regression? For a special case, it is not. Let the wavelet transform of the original data matrix $\mathbf{X}$ be called $\mathbf{Z}$. The wavelet transform is now written as

$$\mathbf{Z} = \mathbf{XB} \tag{13}$$

such that when we are looking at a generalised inverse $\mathbf{Z}^+$ we have:

$$\mathbf{Z}^+ = (\mathbf{XB})^+ = \mathbf{B}^+\mathbf{X}^+ = \mathbf{B}^T\mathbf{X}^+. \tag{14}$$

Here we make use of the fact that the total basis matrix has its transpose equal to its inverse:

$$\mathbf{B}^T\mathbf{B} = \mathbf{I}. \tag{15}$$

Substituting Eq. (14) into Eq. (12) now gives

$$\mathbf{b}_w = \mathbf{B}^T\mathbf{X}^+\mathbf{y} = \mathbf{B}^T\mathbf{b}, \tag{16}$$

which means that we can perform a fast wavelet transform of the $\mathbf{b}$ vector from the PLS analysis directly of the raw data, without the need to transform all the spectra beforehand. Unfortunately, this result cannot be used when truncating or manipulating the individual wavelet coefficient vectors for the different spectra. In such cases we are forced to perform a wavelet transform for each spectrum before regression.

2.4. Interpretation of scalograms

The thrust of this paper is to make use of the fact that a scalogram contains *time and scale (frequency) domain information*. The magnitude of the wavelet coefficients gives an indication of what type of wavelet (whether it is long or short scale) and *where* in the signal it is important. To get an intuitive feeling for scalograms it is useful to see their analogy to time-frequency diagrams as produced by the STFT. In such a transform we take an FFT of a small part of the signal

and produce the local frequency spectrum. A local apodizing function such as a Gaussian is usually applied in order to remove ringing effects (Gibbs effect). The process is repeated by moving a window of a certain size over the signal. The size of the window is important. A short window will produce an excellent time resolution, but a poor frequency resolution. A long window will produce the opposite [51]. The wavelet transform can be interpreted as a STFT where we have used a short window length for the higher frequencies and a long window for low frequencies. So what happens when we analyse a single peak? In order to illustrate this we have made several scalograms of a single Lorentzian peak with different widths. The Lorentzian peak is used here since this type of peak function is commonly used [52] to model peaks in infrared spectra which are currently of interest to us. The basic idea here is that a very broad peak will have very few high-frequency components whereas a very sharp peak will have contributions from long to very high frequencies. This fact can easily be seen in Fig. 2. Here, as the width of the Lorentzian peak becomes smaller, more of the short-scale coefficients become larger. Stated in another way: sharp peaks show coefficients over several scales whereas broad peaks do not. This kind of interpretation will be of use when we analyse scalograms of regression models. In this case we are faced with scalograms that are important for *prediction*. Therefore, it is reasonable to argue that when certain coefficients are important over several scales, this may correspond to sharp peaks or features being important for the prediction. When variable selection is used we are effectively *pruning* away those coefficients describing the various peak patterns that do not contribute to the predictive ability. The variable selection thus provides parsimonious models that indicates which time-scale tiles in the scalograms are important for maintaining regression models with high predictive ability.

3. Variable selection schemes

3.1. Optimal scale combination

Given a PLS model in terms of wavelet coefficients, it is interesting to ask which scales are the most

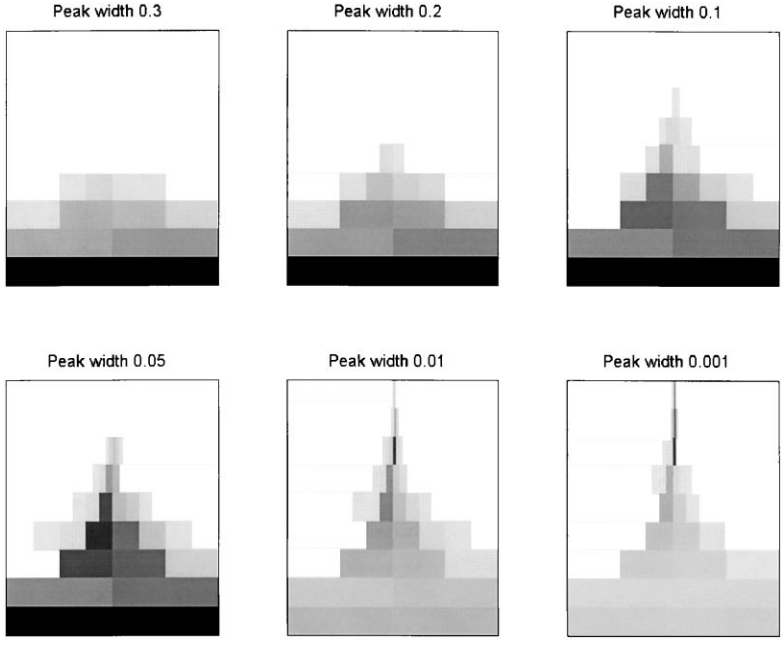

Fig. 2. Illustration of how the scalogram of a single Lorentzian peak changes according to the width of the peak. Note how the sharper peaks tend to occupy more scales (Symmlet 8 wavelet is used).

important for forming a PLS model with the highest prediction ability. By finding such prediction-selective scales, we are focusing on certain band-limited frequency regions that are important for the prediction of the dependent variable. These important scales will therefore, to some extent, represent the underlying features responsible for a successful prediction. One way to find important scales is to test the total number of all possible scale combinations for their predictive ability. The binomial $\binom{K}{i}$ describes the number of combinations that exist for selecting i different scales from the total number of K scales. Let the variable i go from 1 to K which produces K binomials. The total number of scale combinations will thus be the sum of the K binomials:

$$2^K - 1 = \sum_{i=1}^{K} \binom{K}{i}. \tag{17}$$

Thus, in total there are 2^K-1 different scale combinations for K scales. If we have, say, three scales [1 2 3] we can make a representation of the original data set using the following scale combinations: [1], [2], [3], [1 2], [1 3], [2 3] and [1 2 3]. Let us index each of such combinations as $c_1,c_2,...,c_N$ where $N=2^K-1$. Associated with each c_j we have an RMS value, r_j, from applying the PLS model on the validation set using the representation dictated by the scale combination in c_j. It is now possible to sort the duplets (c_j,r_j) with respect to r_j; we are only interested in the PLS models with low values of r_j.

The major weakness of this approach is that we are combining whole scales and disregarding the variation in the time direction.

3.2. PLS variable selection

The PLS variable selection approach as adopted in this paper is very similar to that of Lindgren and coworkers [53,54]. The basic idea behind the method is as follows: truncate to zero elements in the weight vector **w** in the PLS algorithm below a certain threshold. This weight vector contains information about the relevance of x-space in predicting y-space. Once a **w**-vector has been truncated, it is necessary to make it orthogonal to the **w**-vectors calculated for the previous PLS factors (PLS score vectors are also re-orthogonalized in this procedure). The final truncated PLS model can subsequently be represented as a **b**-coefficient regression vector where a majority of the coefficients are set to zero.

3.3. Mutual information

Most methods of variable selection use linear transformations to decide which variables (x) are most strongly related to the output data (y) being modelled. However, these methods can only pick up predominantly linear relationships and will tend to miss variables which have a strong but non-linear relationship with the output. If a linear modelling method is being used, this is advantageous because the modelling method works best with linear relationships. But ANNs are also capable of modelling non-linear relationships. Consequently it is limiting to select variables using a method which prefers (or forces) linearity. Clearly, a method which does not impose the criterion of linearity would be advantageous. This is where *mutual information* comes into play. Mutual information [55] can be regarded as a generalised version of correlation. Where correlation assumes linear relationships and Gaussian-distributed data, mutual information makes no assumptions about the two data series being compared. The mutual information between a class c and an input feature f (with N_f components) is the amount to which the knowledge provided to the feature vector decreases the uncertainty about the class.

Mutual information is derived by calculating the probability distributions of the two series, $p(x)$, $p(y)$ and $p(x,y)$. It then compares the joint probability $p(x,y)$ with $p(x)p(y)$. For statistically independent data,

$$p(x)p(y) = p(x,y) \tag{18}$$

[56]. Hence if these quantities are not the same, there is a dependence between the two data series and this dependence is free from all prior assumptions about its form.

Since the standard way of producing probability distributions by making histograms only works well for dense data, we used a method based on *kernel density estimation* [57]. These probability distributions are then used to form the mutual information,

$I(x,y)$ [56]:

$$I(x,y) = \sum_x \sum_y \left[P(x,y) \log_2 \left\{ \frac{P(x,y)}{P(x)P(y)} \right\} \right]. \quad (19)$$

The mutual information is high if one data series provides much information about the other and low if it provides little. Input variables can thus be selected in a multivariate problem by deriving $I(x,y)$ for each of them and picking those for which this value is largest. In a purely linear Gaussian situation, $I(x,y)$ reduces to correlation and provides identical results.

3.4. Wavelet analysis

The wavelet chosen for these experiments is Symmlet 8 using an FWT. The main reason for this is that the Symmlet 8 wavelet has a shape which is suitable for describing infrared peaks. Of course, other types of wavelets could have been used. In the cases where an unsuitable wavelet has been used one finds that more scales are needed to provide a satisfactory reconstruction. Of course, the reconstruction becomes perfect when all scales are used. For the type of qualitative information of interest to us here, we have found that related wavelets often produce similar results.

4. Experimental

4.1. Data sets

In this article we use two data sets. Data set 1 consists of 40 diffuse reflectance FT-IR spectra of mixtures of the bacterium *Staphylococcus aureus* with the antibiotic ampicillin added at different concentrations (0.5–20 mM with a step of 0.5 mM). Infra-red spectra (256 coadds) for each of these samples were recorded in the wavenumber interval 4000–600 cm^{-1} using a Bruker IFS28 FT-IR spectrometer (Bruker Spectrospin, Coventry CV4 9GH) equipped with a liquid N_2-cooled MCT (mercury–cadmium–telluride) detector and a diffuse-reflectance absorbance TLC accessory. We used 4.0 cm^{-1} wavenumber resolution, and spectra were collected at 20 s^{-1}. The digitisation interval of the IR instrument was set to produce 882 data points. One consequence of this is that we had to add zeros to produce 1024 data point vectors satisfying the "power of 2" data length requirement of the Mallat wavelet algorithm.

The background spectrum was recorded from an empty well. This approach was also used for Data set 2. The samples were applied to 20×20 array of wells on a sandblasted aluminium plate. Full details about the preparation and collection of this data set can be found in [58].

ASCII data were exported from the Opus software used to control the FT-IR instrument and imported into MATLAB 5.1 (MathWorks, MA). The samples were separated into calibration and validation sets, each containing 20 objects, using the DUPLEX method [59]. A separate PLS cross validation was used for finding the optimal model for the calibration data.

Data set 2 consists of 160 FT-IR spectra of the three compounds histidine, glycine and sucrose at different concentrations. The span of 27 different concentration distributions of each compound is shown in Table 1.

Table 1
Concentrations (in %) for each of the three compounds used in Data set 2

Histidine	Glycine	Sucrose
100	0	0
90	10	0
90	0	10
80	10	10
70	20	10
70	10	20
60	30	10
60	20	20
60	10	30
0	100	0
10	90	0
0	90	10
10	80	10
20	70	10
10	70	20
30	60	10
20	60	20
10	60	30
0	0	100
10	0	90
0	10	90
10	10	80
20	10	70
10	20	70
30	10	60
20	20	60
10	30	60

Six replicate 5μl aliquots of 27 samples consisting of different combinations of histidine (100 mM), glycine (300 mM) and sucrose (100 mM) solutions were dried into wells in a sandblasted aluminium plate. Infrared spectra were collected and data processed as described for Data set 1 above, but using 16 coadds. Initially we had 6 replicates of each concentration distribution, but found that 12 of the glycine replicates were outliers and were therefore removed from the data set. In Data set 2 of this paper we are modelling the histidine concentration.

4.2. Software and hardware

All calculations were performed using MATLAB 5.1 (MathWorks, MA). To calculate the discrete wavelet transform the WaveLab toolbox [60] was used. The mutual information program was based on both software written by author AMW and the Kernel Density Estimation Toolbox [61]. The PLS variable selection program using truncation of **w**-vectors was written by author BKA. All analyses were performed on a Pentium Pro 200 MHz computer running the Windows NT 4.0 operating system.

5. Results

5.1. Data set 1

The prediction ability of the PLS model using all available wavelet coefficients of the raw data is quite good: the RMS error of prediction on the unseen test set is 5.43% using seven PLS factors. The **b**-coefficient vector of the PLS analysis is depicted in Fig. 3. Note that we have a spectrum-like profile which is

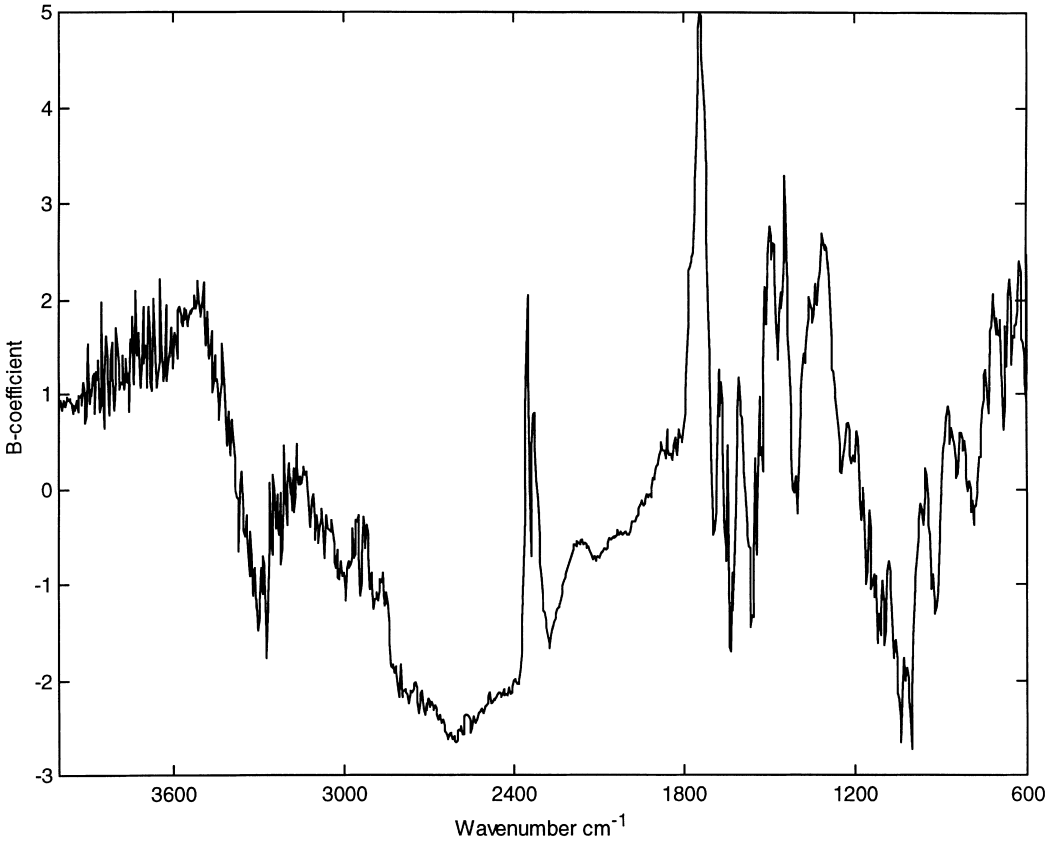

Fig. 3. The **b**-coefficient vector from a PLS analysis of raw Data set 1 without any wavelet analysis.

rather noisy. We should then expect that the selected variables in the regions represent sharp and localised features. Surprisingly, this is not so. As will be demonstrated, the most important features for the prediction in this data set are long-scale in nature.

The first approach we chose to determine important scales in PLS prediction was to perform an optimal scale combination analysis as described in the method section above. The length of the data vectors is 1024 which corresponds to 10 scales: 0,1,...,9. In the present optimal scale combination analysis we did not include the zero'th scale which is basically a scalar offset to the whole spectral region. The total number of combinations for nine scales (1...9) is 511. Thus, we produced 511 PLS models on the calibration data set and recorded the RMS values between the predicted and measured concentrations of the infrared spectra. The performance of each PLS model was determined by cross validation. The scale combination which produced the best results for the calibration set was selected and applied to an unseen test set. Among all the various combinations we found that using the scales [1 2 4 9] produces an RMS=5.5%, using seven PLS factors (A=7). This means that only a few scales are necessary for providing the same level of predictive ability as in the raw data representation. One striking feature is that so many of the longer scales are important for the prediction. The importance of scale 9 can almost certainly be attributed to noise and the uncertainties of the optimal scale combination approach.

Another way to refine the wavelet regression model is to perform a PLS variable selection procedure as described above. Different numbers of variables were selected as the most important (interval 1–25). For each of these numbers a PLS cross validation was performed to determine the optimal number of PLS factors in the calibration set. The model with the lowest PRESS value was selected and applied to the unseen validation set.

By truncating to only nine wavelet coefficients we obtain an RMS prediction error of 6.30% (A=9 PLS factors). This is a drastic reduction (from 882 variables) in the data size necessary for keeping a satisfactory model prediction error. We could have performed the variable selection directly on the raw data, but then we would not have been able to investigate on which scales these variables contribute. In this case we note that these nine variables are present only at scales 1, 2 and 3. This suggests that scales 4 and 9 from the optimal scale combination analysis may not be significant. The reason for this discrepancy can be traced back to the fact that the optimal scale combination analysis does not take into account the wavelet coefficients that are important for the prediction in localised wavenumber regions. In the optimal scale combination analysis *all* the coefficients in all the scales were investigated. The nine important wavenumbers found with PLS variable selection represents *regions* in the IR spectrum. The first three variables selected belong to all the possible wavelet coefficients in scale 0 (wavelet coefficient no. 2) and 1 (wavelet coefficient no. 2 and 3) and thus covers the whole wavenumber region. The next three variables (wavelet coefficient no. 6, 7 and 8) belong to almost all available wavelet coefficients of scale 2 (four wavelet coefficients possible: no. 5, 6, 7 and 8). The wavenumber region covered by the scale-2-selected wavelet coefficients is 3008–600 cm^{-1} (see Fig. 4). The remaining three variables (wavelet coefficient no. 9, 10 and 13) belong to wavelet coefficients at scale 3 (wavelet coefficient no. 9–16). Wavelet coefficient no. 9–10 covers the region 4000–3012 cm^{-1} and no. 13 covers the region 2017–1521 cm^{-1} (see Fig. 4). Note that wavelet coefficient no. 13 incorporates a region which contains the so-called "ampicillin peak" (1767 cm^{-1}, see Fig. 4) due to the carbonyl group in the β-lactam ring of the molecule. The results are also presented as a scalogram in Fig. 5(B).

It is encouraging that both the optimal scale combination analysis and the PLS variable selection procedure gave similar results.

The mutual information (MI) between each wavelet coefficient variable in the calibration set and the concentration vector was computed. This produced a vector of MI values of each variable in the range [0, 0.53]. Based on this vector the variables with highest MI values were selected according to a threshold scheme. Thirty thresholds in the [0, 0.53] range were chosen. For each threshold, only those variables with higher MI values than the threshold value were selected. All other variables were set to zero. For each selected set of variables a PLS cross validation on the calibration set was performed and the optimum number of factors was determined. The RMS error of prediction of the optimal PLS model applied to the

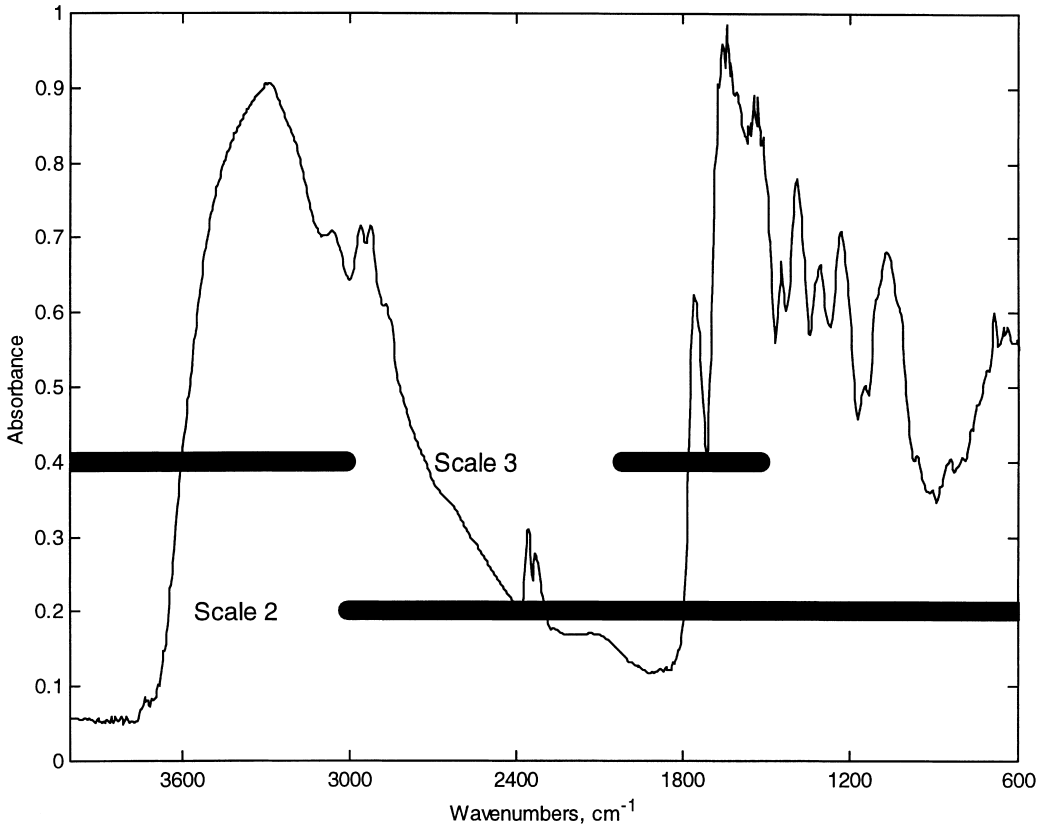

Fig. 4. Variable selection on wavelet coefficients will indicate important regions where the size of a region depends on which scale the variable has been selected. In this figure the results from the PLS variable selection in Data set 1 is shown (nine wavelet coefficients selected). Included in the figure is also a typical IR spectrum from this data set. Please note the characteristic "ampicillin peak" at 1767 cm^{-1} which is among the selected variables (at scale 3).

unseen validation set is 6.4% using 448 variables (A=5 PLS factors). The corresponding scalogram is shown in Fig. 5(C). We see that a region around the ampicillin peak has been selected. In addition, several of the longer scale coefficients in this region have also been selected. This is in agreement with the results from the previous PLS variable selection scheme.

If we select only the nine largest MI values (to compare it with the results from the truncation of the PLS weight vector method), we obtain a PLS model with 10% RMS error of prediction. The scalogram of the selected variables are shown in Fig. 5(D). One variable is located close to the ampicillin peak (1780 cm^{-1}) at the shortest scale (no. 9). In addition, we also see variables in the region 1900–2020 cm^{-1} at scales 5 and 6. The fact that none of the longer scales were selected may explain some of the high prediction errors for this approach.

5.2. Data set 2

Cross validation was used on the calibration part of this data set to obtain the optimal number of PLS factors using all available wavelet coefficients. The RMS prediction error on the unseen test set is 7.7% with A=10 PLS factors. A systematic test of all the different scale combinations was made for this data set also. The prediction error increases to 9.3% using only the wavelet coefficients from the scales [1 3 5 6 7] (with 10 PLS factors). In this case the optimal scale combination analysis increases the RMS prediction error on the unseen validation set by more than 2%

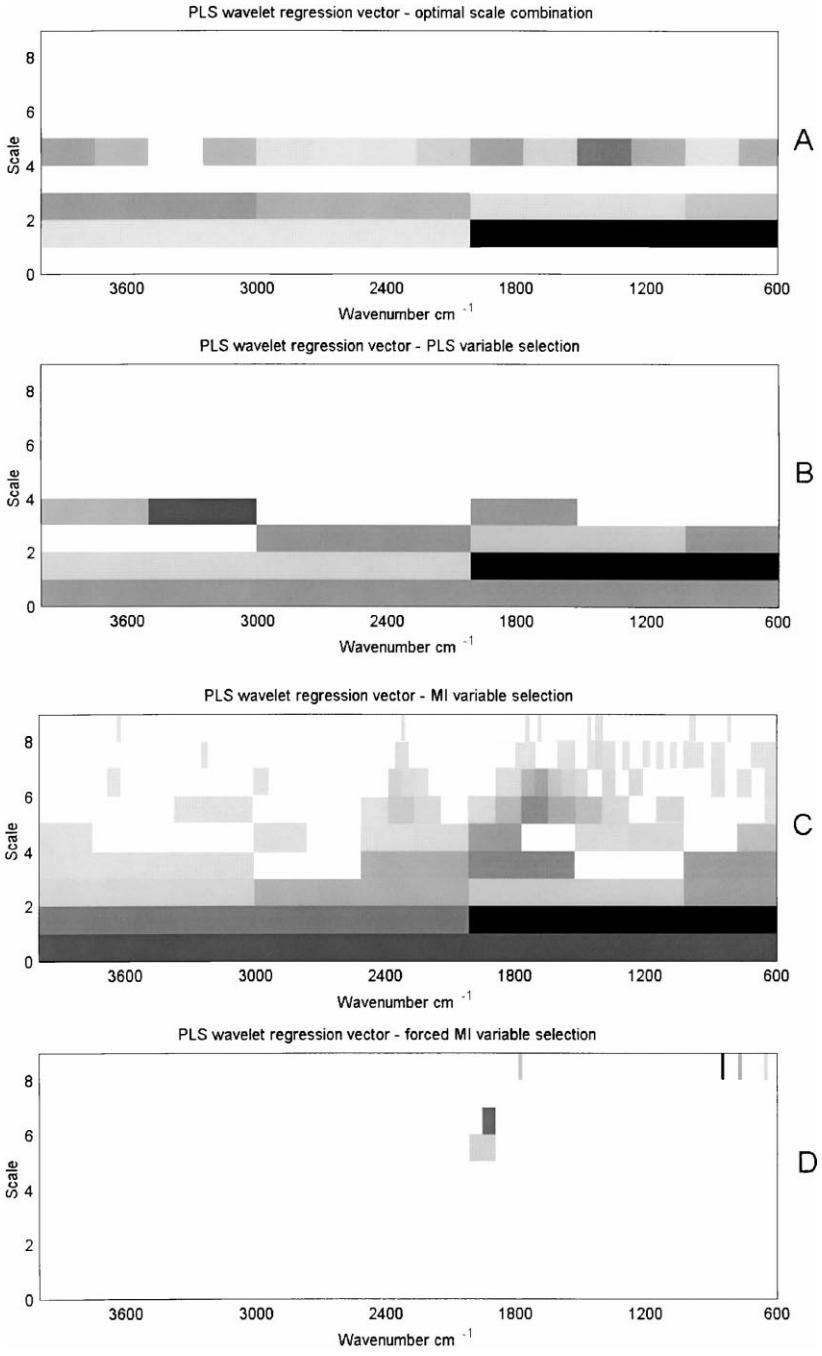

Fig. 5. (A) The **b**-coefficient vector from the PLS model with the optimal scale combination [1 2 4 9] for Data set 1 is shown. Note that all the scales [0, 3, 5, 6, 7, 8] do not have any coefficients because they are removed; (B) the result after performing PLS variable selection by truncating **w**-vector coefficients on Data set 1. Prediction RMS here is 6.30% with nine variables only. Note that these nine variables are only present at scales 0, 1, 2 and 3; (C) mutual information variable selection on Data set 1 where the MI model is chosen on the basis on the best model in the calibration set; (D) mutual information variable selection on Data set 1 where the MI model is forced to use only the first nine variables (to make it comparable with results in (B)).

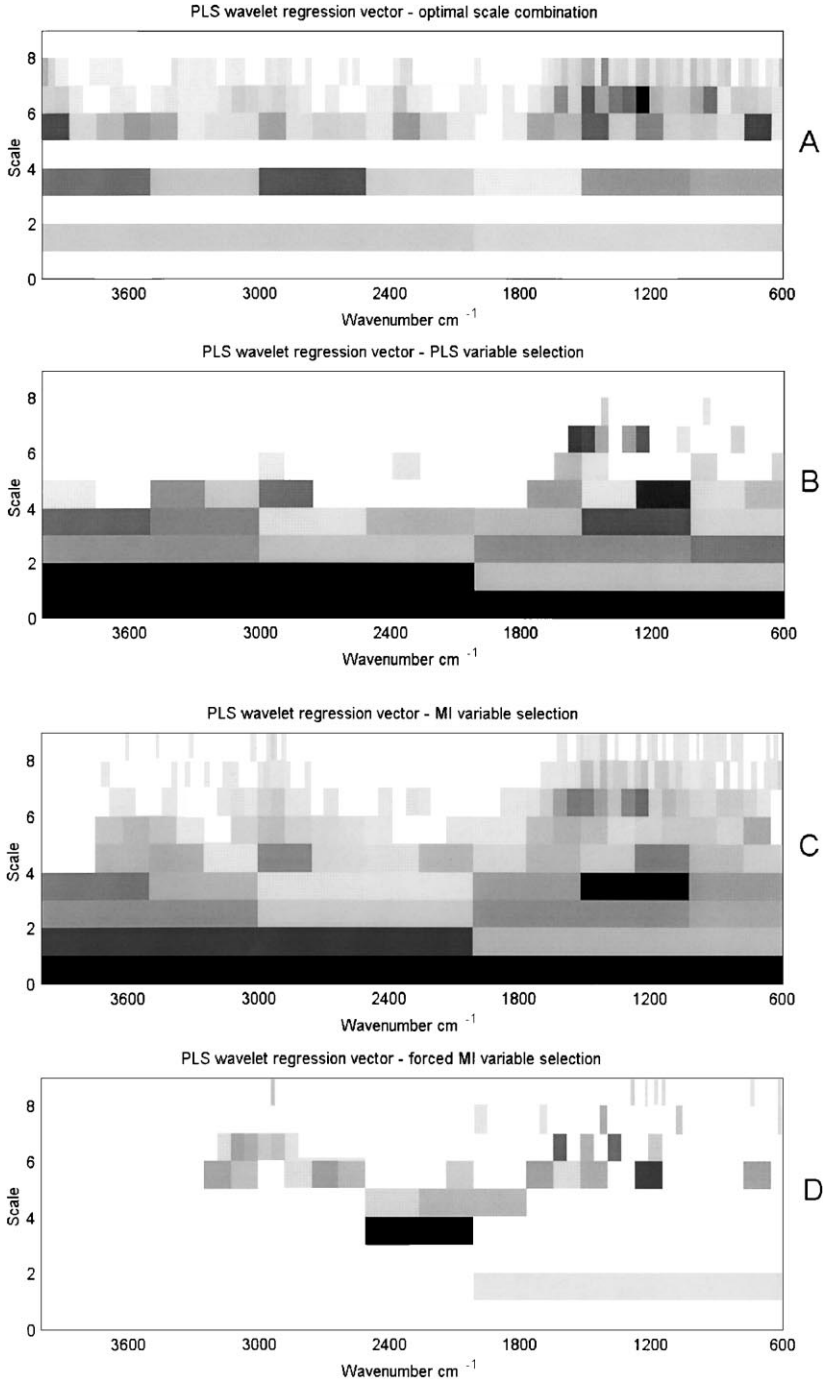

Fig. 6. (A) The **b**-coefficient vector from the PLS model with the best scale combination using the scales [1 3 5 6 7] for Data set 2 is shown. The RMS prediction is 9.3%; (B) the result after performing PLS variable selection by truncating **w**-vector coefficients on Data set 2. Prediction RMS here is 7.9% with 44 variables; (C) mutual information variable selection on Data set 2 where the MI model is chosen on the basis on the best model in the calibration set; (D) mutual information variable selection on Data set 2 where the MI model is forced to use only the first 44 variables (to make it comparable with results in (B)).

Table 2
Summary of results for Data sets 1 and 2 of the PLS calibration, using different variable selection methods

	RMS (DS1) (%)	Opt. *A* (DS1)	Vars. (DS1)	RMS (DS2) (%)	Opt. *A* (DS2)	Vars. (DS2)
All variables	5.4	7	882	7.7	10	882
Comb.	5.5	7	534	9.3	10	234
Mut. inf.	6.4	5	448	7.9	8	736
Mut. inf. f.	10.0	5	9	13.0	6	44
PLS sel.	6.3	9	9	7.9	10	44

Abbreviations: RMS is Root Mean Square error of prediction in percent; Opt. *A* is the optimal number of PLS factors used in the model and Vars. is the number of variables used in the model (the remaining variables are set to zero); Comb. means the combination of scales; Mut. inf. signifies that the variables were selected by the mutual information approach and Mut. inf. f. is the forced mutual information where the number of selected variables is forced to be comparable with the results from the PLS variables selection (PLS sel.); DS1 signifies Data set 1 and DS2 signifies Data set 2.

(see Fig. 6(A) for a scalogram of the selected scales). A PLS variable selection was performed which presented a much better result: the RMS error prediction on the unseen validation set was 7.9% using 10 PLS factors and only 44 variables (see Fig. 6(B) for a scalogram of the selected variables).

Fig. 6(B) indicates that we may have a narrow well-defined peak around 1420 cm^{-1} that is important for the prediction. In addition, very broad features (located at long scales) in the 4000–1800 cm^{-1} region have also been selected.

Variable selection using mutual information was also applied to this data set. The RMS error of prediction on the unseen validation set was 7.9% using eight PLS factors and 736 selected variables (the scalogram of the selected variables is shown in Fig. 6(C)). In other words, the compression performed by MI was much worse than the PLS variable selection. If we select only the 44 largest MI values (to compare it with the result from the truncation of the PLS weight vector method) and $A=6$ PLS factors, we obtain a PLS model with approximately 13% RMS error of prediction (see Fig. 6(D)). Very few of the very long scales (0, 1 and 2) were selected in this approach, which may explain the high prediction error observed.

For both data sets we observe that the mutual information procedure selects somewhat different variables than those from the PLS variable selection method. The major difference between the two variable selection methods is that MI does not operate on a latent variable structure but directly on each wavelet coefficient variable separately. With the spectroscopic data set used here, significant correlations between the variables reduce the effect of the single variable approach in MI. Please see Table 2 for summary of results for Data sets 1 and 2.

6. Discussion

When variable selection is applied to collinear data such as spectral profiles of complex mixtures we often find that regression models with different sets of selected variables have almost identical predictive ability. This means that the variables selected are not *unique*. When we are predicting the concentration of a compound in the mixture, all the absorbing wavenumbers specific for the molecule will have a high correlation with the concentration variable. Therefore, in most cases we cannot expect to find a very limited number of *unique* variables, but rather regions of interest where good representative wavenumber candidates are found. This suggests that instead of performing the variable selection in the original domain, a *compressed domain representation* may be more fruitful. Searching for suitable wavenumber regions rather than individual wavenumbers has been successfully demonstrated in [62–64] which is comparable to working in a compressed representation.

In this article we have chosen wavelets as the basis for compression but other compression bases could also have been used, e.g. B-splines [65,66]. The resulting coefficients from a successful compression are less correlated than the original variables. For a ''perfect'' compression, we would observe no correlation between the coefficients, but it is of course very

difficult to find this optimal compression. The problem is related to finding the smallest possible program to perform a certain task. Performing a compression before regression alone can be regarded as a method of variable selection. However, since we know the compression is not perfect, it is possible to go further and find an even smaller number of coefficients that can be used in the regression model. In addition to excellent compression abilities, wavelets are also very powerful in time-scale/frequency analyses of signals which enables us to introduce the concept of scales into the variable selection/compression procedure directly. This means that it is possible to determine whether a selected coefficient is associated with a long- or short-scale feature *necessary for prediction*. If, for instance, we obtain several selected variables on the longest scales, this would indicate that the important information for the prediction model is in features that are very broad. However, when there are several selected variables in a localised wavenumber region over *both* long and short scales, there may be a narrow feature in this region that is important for prediction.

Acknowledgements

We thank the UK BBSRC, GlaxoWellcome and Bruker Spectrospin Ltd. for financial support.

References

[1] E.P.C. Lai, R.D. Giroux, N.L. Chen, R.D. Guo, Canad. J. Chem. 71 (1993) 968–975.
[2] L.A. Marquardt, M.A. Arnold, G.W. Small, Anal. Chem. 65 (1993) 3271–3278.
[3] J. Workman, J. Brown, Spectroscopy 11 (1996) 48–51.
[4] E. Engstrom, B. Karlberg, J. Chemomet. 10 (1996) 509–520.
[5] K. Thyholt, G. Enersen, T. Isaksson, Meat Sci. 48 (1998) 49–63.
[6] G. Downey, Analyst 119 (1994) 2367–2375.
[7] L.J. Janik, J.O. Skjemstad, Austral. J. Soil Res. 33 (1995) 637–650.
[8] Y.W. Lai, E.K. Kemsley, R.H. Wilson, J. Agric. Food Chem. 42 (1994) 1154–1159.
[9] R. Goodacre, S. Trew, C. Wrigley-Jones, M.J. Neal, J. Maddock, T.W. Ottley, N. Porter, D.B. Kell, Biotechnol. Bioeng. 44 (1994) 1205–1216.
[10] R. Goodacre, M.J. Neal, D.B. Kell, Anal. Chem. 66 (1994) 1070–1085.
[11] R. Goodacre, M.J. Neal, D.B. Kell, L.W. Greenham, W.C. Noble, R.G. Harvey, J. Appl. Bacteriol. 76 (1994) 124–134.
[12] S. Wold, Technometrics 35 (1993) 136–139.
[13] S. Wold, A. Ruhe, H. Wold, W.J. Dunn, SIAM J. Sci. Stat. Comput. 5 (1984) 735–743.
[14] S. Wold, H. Martens, H. Wold, Lecture Notes Math. 973 (1983) 286–293.
[15] T.R. Holcomb, M. Morari, Comput. Chem. Eng. 16 (1992) 393–411.
[16] I.E. Frank, Chemomet. Intell. Lab. Sys. 8 (1990) 109–119.
[17] S. DeJong, J. Chemomet. 7 (1993) 551–557.
[18] B. Cheng, D.M. Titterington, Stat. Sci. 9 (1994) 2–30.
[19] B. Delyon, A. Juditsky, A. Benveniste, IEEE Trans. Neural Networks 6 (1995) 332–348.
[20] C.M. Bishop, Neural Networks for Pattern Recognition, Clarendon, Oxford, 1995.
[21] A.B. Bulsari (Ed.), Neural Networks for Chemical Engineers, Elsevier, Amsterdam, 1995.
[22] S. Haykin, Neural Networks, Macmillan, New York, 1994.
[23] S.S. Haykin, Neural Networks: A Comprehensive Foundation, Macmillan, New York, 1994.
[24] H.H. Szu, X.Y. Yang, B.A. Telfer, Y.L. Sheng, Phys. Rev. E 48 (1993) 1497–1501.
[25] J. Zhang, G.G. Walter, Y.B. Miao, W.N.W. Lee, IEEE Trans. Signal Process. 43 (1995) 1485–1497.
[26] B.K. Alsberg, A.M. Woodward, D.B. Kell, Chemomet. Intell. Lab. Sys. 37 (1997) 215–239.
[27] I. Daubechies, Ten Lectures on Wavelets, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.
[28] I. Daubechies, SIAM J. Math. Anal. 24 (1993) 499–519.
[29] I. Daubechies, in: I. Daubechies (Ed.), Different Perspectives on Wavelets, American Mathematical Society, Providence, RI, 1993, pp. 1–33.
[30] D.L. Donoho, in: I. Daubechies (Ed.), Different Perspectives on Wavelets, American Mathematical Society, Providence, RI, 1993, pp. 173–205.
[31] D.L. Donoho, I.M. Johnstone, Biometrika 81 (1994) 425–455.
[32] D.L. Donoho, I.M. Johnstone, G. Kerkyacharian, D. Picard, J. Royal Stat. Soc. B 57 (1995) 301–337.
[33] M. Holschneider, J. Math. Phys. 34 (1993) 3227–3244.
[34] M. Holschneider, J. Math. Phys. 34 (1993) 4190–4198.
[35] M. Holschneider, Commun. Math. Phys. 160 (1994) 457–473.
[36] M. Holschneider, Wavelets: An Analysis Tool, Clarendon, Oxford, 1995.
[37] L. Hong, IEEE Trans. Aerospace Electron. Sys. 29 (1993) 1244–1251.
[38] J.C. Huffman, SMPTE J. 103 (1994) 723–727.
[39] S. Mallat, IEEE Trans. Pattern Anal. Machine Intell. 2 (1989).
[40] S. Mallat, Trans. Am. Math. Soc. 315 (1989) 69–87, 18215–18231.
[41] S.G. Mallat, IEEE Trans. Pattern Anal. Machine Intell. 11 (1989) 674–693.
[42] S. Mallat, W.L. Hwang, IEEE Trans. Inform. Theory 38 (1992) 617–643.
[43] Y. Meyer, Congr. Int. Phys. Math., Swansea, July 1988.
[44] Y. Meyer, Rapport CEREMADE, No. 8704, Univ. Paris-Dauphine, Paris, France, 1987.

[45] I. Daubechies, A. Grossmann, Commun. Pure Appl. Math. 41 (1988) 151–164.
[46] A. Teolis, J.J. Benedetto, Signal Process. 45 (1995) 369–387.
[47] M. Cartwright, Fourier Methods for Mathematicians, Scientists and Engineers, Ellis Horwood, New York, 1990.
[48] M. Vetterli, J. Kovacevic, Wavelets and Subband Coding, Prentice Hall PTR, New Jersey, 1995.
[49] G. Davis, S. Mallat, Z.F. Zhang, Optical Eng. 33 (1994) 2183–2191.
[50] H. Martens, T. Næs, Multivariate Calibration, Wiley, Chichester, 1989.
[51] A. Graps, IEEE Comput. Sci. Eng. 2 (1995) 50–61.
[52] P.R. Griffiths, J.A. de Haseth, Fourier Transform Infrared Spectrometry, Wiley, New York, 1986.
[53] F. Lindgren, P. Geladi, S. Rannar, S. Wold, J. Chemomet. 8 (1994) 349–363.
[54] F. Lindgren, P. Geladi, A. Berglund, M. Sjostrom, S. Wold, J. Chemomet. 9 (1995) 331–342.
[55] C.E. Shannon, A Mathematical Theory of Communication, American Telephone and Telegraph Co., New York, 1948.
[56] R. Battiti, IEEE Trans. Neural Networks 5 (1994) 537–550.
[57] C.C. Beardah, M.J. Baxter, Interfacing the past: Computer applications and quantitative methods in archaeology, CAA95, 1995.
[58] M.K. Winson, B.K. Alsberg, A.M. Woodward, R. Goodacre, I. Timmins, A. Jones, D. Broadhurst, J. Rowland, D.B. Kell, Anal. Chim. Acta 348 (1997) 273–282.
[59] R.D. Snee, Technometrics 19 (1977) 415–428.
[60] J. Buckheit, S. Chen, J. Crutchfield, D. Donoho, H. Gao, I. Johnstone, E. Kolaczyk, J. Scargle, K. Young, T. Yu, http://playfair.Stanford.EDU:80/∼wavelab/, 1996.
[61] C.C. Beardah, Kernel Density Estimation Toolbox, 1.1 ed., Nottingham, Web address: http://maths.ntu.ac.uk/ccb/html/densest.html, 1996.
[62] B.K. Alsberg, M.K. Winson, D.B. Kell, Chemomet. Intell. Lab. Sys. 36 (1997) 95–109.
[63] P.J. Brown, J. Chemomet. 6 (1992) 151–161.
[64] J.M. Brenchley, U. Hörchner, J.H. Kalivas, Appl. Spectrosc. 51 (1997) 689–699.
[65] B.K. Alsberg, O.M. Kvalheim, J. Chemomet. 7 (1993) 61–73.
[66] B.K. Alsberg, E. Nodland, O.M. Kvalheim, J. Chemomet. 8 (1994) 127–145.