Anal. Chem. **1998,** *70,* 4126−4133

Variable Selection in Discriminant Partial Least-Squares Analysis

Bjørn K. Alsberg,* Douglas B. Kell, and Royston Goodacre

Institute of Biological Sciences, Cledwyn Building, University of Wales, Aberystwyth, Ceredigion, SY23 3DD, United Kingdom

Variable selection enhances the understanding and interpretability of multivariate classification models. A new chemometric method based on the selection of the most important variables in discriminant partial least-squares (VS-DPLS) analysis is described. The suggested method is a simple extension of DPLS where a small number of elements in the weight vector w is retained for each factor. The optimal number of DPLS factors is determined by cross-validation. The new algorithm is applied to four different high-dimensional spectral data sets with excellent results. Spectral profiles from Fourier transform infrared spectroscopy and pyrolysis mass spectrometry are used. To investigate the uniqueness of the selected variables an iterative VS-DPLS procedure is performed. At each iteration, the previously found selected variables are removed to see if a new VS-DPLS classification model can be constructed using a different set of variables. In this manner, it is possible to determine regions rather than individual variables that are important for a successful classification.

With recent developments in analytical instrumentation, physicochemical spectroscopic methods, often referred to as "whole-organism fingerprinting"[1] provide a rapid way of obtaining information about complex samples. The most common such methods are pyrolysis mass spectrometry (PyMS),[2] Fourier transform infrared (FT-IR) spectroscopy,[3−8] and Raman spectroscopy.[9−12] The type of biological material of particular interest to us is colonies of microorganisms. FT-IR allows the chemically based discrimination of intact microbial cells, without their destruction, and produces complex biochemical fingerprints which are reproducible and distinct for different bacteria. Naumann and co-workers[4,5,7,8] have shown that FT-IR absorbance spectroscopy (in the mid-IR range, usually defined as 4000−400 cm^{-1}) provides a powerful tool with sufficient resolving power to distinguish microbial cells at the strain level. However, FT-IR spectra have conventionally been interpreted by the application of unsupervised pattern recognition methods such as correspondence analysis maps and cluster analysis.[6] Unfortunately, such analyses are often influenced by subjective interpretation.[2] Supervised methods, however, are not subject to these pitfalls in the same degree. It is therefore strongly recommended to choose supervised methods whenever this is practically feasible, and we have published a number of successful applications of supervised approaches to the classification of microorganisms.[3,12−17] In recent years several powerful multivariate methods for classification and regression have been constructed. Examples of such powerful methods are partial least squares (PLS), artificial neural networks (ANNs), multivariate rule induction (e.g., classification and regression trees (CART) and fuzzy rule induction building expert system (FuRES)[18]), and evolutionary-based techniques (genetic algorithms, genetic programming, evolutionary programming).[17,19−21]

The prediction accuracy of a classification method is of course of the utmost importance, but there is also a growing need for obtaining a better understanding of the final multivariate classification models. Some methods such as for example ANNs and *k*-nearest neighbors (kNN) are very difficult to interpret whereas other methods such as PLS and CART are much better in this respect. One way to simplify models is to compress the resulting models so the number of variables used is small as possible. This

(1) Magee, J. T. In *Handbook of New Bacterial Systematics*; Goodfellow, M., O'Donnell, A. G., Eds.; Academic Press: London, 1993; pp 383−427.
(2) Goodacre, R.; Kell, D. B. *Curr. Opin. Biotechnol.* **1996**, *7*, 20−28.
(3) Goodacre, R.; Timmins, É. M.; Rooney, P. J.; Rowland, J. J.; Kell, D. B. *FEMS Microbiol. Lett.* **1996**, *140*, 233−239.
(4) Helm, D.; Labischinski, H.; Schallehn, G.; Naumann, D. *J. Gen. Microbiol.* **1991**, *137*, 69−79.
(5) Naumann, D.; Fijjala, V.; Lavischinski, H.; Giesbrecht, P. *Mol. Struct.* **1988**, *174*, 165−170.
(6) Naumann, D.; Helm, D.; Labischinski, H.; Giesbrecht, P. In *Modern techniques for rapid microbiological analysis*; Nelson, W. H., Ed.; VCH Publishers: New York, 1991; pp 43−96.
(7) Naumann, D.; Keller, S.; Helm, D.; Schultz, C.; Schrader, B. *J. Mol. Struct.* **1995**, *347*, 399−405.
(8) Naumann, D.; Schultz, C. P.; Helm, D. In *Infrared spectroscopy of biomolecules*; Mantsch, H. H., Chapman, D., Eds.; Wiley: New York, 1996; pp 279−310.
(9) Puppels, G. J.; Greve, J. *Adv. Spectrosc.* **1993**, *20A*, 231−265.
(10) Nelson, W. H.; Sperry, J. F. In *Modern techniques for rapid microbiological analysis*; Nelson, W. H., Ed.; VCH Publishers: New York, 1991; pp 97−143.
(11) Nelson, W. H.; Manoharan, R.; Sperry, J. F. *Appl. Spectrosc. Rev.* **1992**, *27*, 67−124.
(12) Goodacre, R.; Timmins, É. M.; Burton, R.; Kaderbhai, N.; Woodward, A.; Kell, D. B.; Rooney, P. J. *Microbiology* **1998**, *144*, 1157−1170.
(13) Goodacre, R.; Neal, M. J.; Kell, D. B.; Greenham, L. W.; Noble, W. C.; Harvey, R. G. *J. Appl. Bacteriol.* **1994**, *76*, 124−134.
(14) Goodacre, R.; Hiom, S. J.; Cheeseman, S. L.; Murdoch, D.; Weightman, A. J.; Wade, W. G. *Curr. Microbiol.* **1996**, *32*, 77−84.
(15) Goodacre, R.; Rooney, P. J.; Kell, D. B. *J. Antimicrob. Chemother.* **1998**, *41*, 27−34.
(16) Alsberg, B. K.; Goodacre, R.; Rowland, J. J.; Kell, D. B. *Anal. Chim. Acta* **1997**, *348*, 389−407.
(17) Taylor, J.; Goodacre, R.; Wade, W. G.; Rowland, J. J.; Kell, D. B. *FEMS Microbiol. Lett.* **1998**, *160*, 237−246.
(18) Harrington, P. d. B. *J. Chemom.* **1991**, *5*, 467−486.
(19) McKay, B.; Willis, M.; Barton, G. *Comput. Chem. Eng.* **1997**, *21*, 981−996.
(20) Gilbert, R. J.; Goodacre, R.; Woodward, A. M.; Kell, D. B. *Anal. Chem.* **1997**, *69*, 4381−4389.
(21) Marenbach, P.; Bettenhausen, K. D.; Freyer, S.; Nieken, U.; Rettenmaier, H. *Proc. Inst. Mech. Eng. Part I. J. Syst. Control Eng.* **1997**, *211*, 325−332.

S0003-2700(98)00506-X CCC: $15.00
Published on Web 09/03/1998

compression can be achieved by either replacing the original data domain by a smaller one, e.g., using the wavelet transform,[22−24] B-splines,[25,26] or a peak parameter representation,[27] or selecting only the most important variables in the original domain.[28] In fact, a combination of both approaches is possible.[29]

Several approaches to variable selection in supervised classification models have been suggested.[30−37] The type of variable selection of particular interest to us in this article is performed on discriminant PLS models. There are several reasons for choosing PLS as the method for classification. We have previously shown[16] that discriminant PLS is comparable in accuracy and interpretation to other powerful classification methods such as uni- and multivariate CART,[38] artificial neural networks,[39] *k*-nearest neighbors,[40] and fuzzy rule building expert system.[18] In addition, PLS is suitable for algorithmic optimization to handle large data sets efficiently.[41] The variable selection suggested here is similar to that demonstrated by Lindgren and co-workers.[42,43] In this article we extend a similar approach to PLS2 models which forms the core of the DPLS analysis. The new variable selection (VS)-DPLS algorithm produces classification models with a small number of variables, thus enabling the investigation of a small subset of variables that will maintain or preferably improve on the prediction error obtained using all variables.[44] It should be emphasized that there are in general several different solutions to the variable selection problem, and hence any set of variables selected does not necessary constitute the optimal or the only possible subset of variables with good prediction abilities,[45] and in which the best optimization approach is likely to depend on the dataset itself.[46] This phenomenon is particularly important for collinear data sets where the information that correlates well with the dependent variable (*Y*) is spread over whole regions rather than concentrated in unique variables.

DISCRIMINANT PLS

The theory and properties of the partial least-squares algorithms PLS1 (with one dependent (*Y*) variable) and PLS2 (with several dependent *Y* variables) have been extensively studied and reported in the literature.[47] We will therefore give only a short description of the DPLS method which is the PLS2 algorithm applied to classification problems. The central point in the PLS paradigm is to find latent variables in the feature space that have a maximum covariance with the *Y* variable(s). Thus, linear combinations of the feature space variables are found that are rotated to have maximum prediction ability for the *Y* variable(s). In PLS2 one uses linear combinations of the *Y* space variables rather than individual *Y* variables.

The final PLS model can be formulated as a regression equation:

$$\mathbf{Y} = \mathbf{XB} \tag{1}$$

where the estimated regression coefficients **B** are

$$\mathbf{B} = \mathbf{X}^{+}\mathbf{Y} \tag{2}$$

$\mathbf{X}^{+}$ is a generalized inverse provided by the PLS2 algorithm. To obtain a prediction from the PLS2 model it is sufficient to use eq 2. In this article we compute the regression matrix **B** as demonstrated by Martens and Næs:[47]

$$\mathbf{B} = \mathbf{W}(\mathbf{P}^{\mathrm{T}}\mathbf{W})^{-1}\mathbf{Q}^{\mathrm{T}} \tag{3}$$

where **W** is the matrix of weights of the *X*-space, **Q** is the loadings matrix for the *Y* space, and **P** is the *X* space loadings matrix.

The prediction of dependent variables on a new set of objects is done by

$$\mathbf{Y}_{\mathrm{test}} = \mathbf{X}_{\mathrm{test}}\mathbf{B} \tag{4}$$

The **Y** matrix of dependent variables contains information about class memberships of objects. If *K* is the number of classes, each row, $\mathbf{y}^{\mathrm{T}}$, in the **Y** matrix has the following structure:

$$y_j^{\mathrm{T}} = \begin{cases} 1 \text{ if object belongs to class } j \\ 0 \text{ otherwise} \end{cases} \tag{5}$$

where y_j is the *j*th column in **Y**. *j* is also the class number, where we have $j = 1, 2, \ldots, K$. The binary **Y** matrix therefore has a structure where each row sums to unity. When the final DPLS model is used in prediction, however, the estimated $\hat{\mathbf{Y}}$ matrix does not have such a structure. The predicted values are real numbers

(22) Vetterli, M.; Kovacevic, J. *Wavelets and subband coding*; Prentice Hall PTR: Englewood Cliffs, NJ, 1995.
(23) Alsberg, B. K.; Woodward, A. M.; Kell, D. B. *Chemom. Intell. Lab. Syst.* **1997**, *37*, 215−239.
(24) Alsberg, B. K.; Woodward, A. M.; Winson, M. K.; Rowland, J.; Kell, D. B. *Analyst* **1997**, *122*, 645−652.
(25) Alsberg, B. K.; Kvalheim, O. M. *J. Chemom.* **1993**, *7*, 61−73.
(26) Alsberg, B. K.; Nodland, E.; Kvalheim, O. M. *J. Chemom.* **1994**, *8*, 127−145.
(27) Alsberg, B. K.; Winson, M. K.; Kell, D. B. *Chemom. Intell. Lab. Syst.* **1997**, *36*, 95−109.
(28) Shaw, A. D.; di Camillo, A.; Vlahov, G.; Kell, D. B.; Rowland, J.; Bianchi, G. *Anal. Chim. Acta* **1997**, *348*, 357−374.
(29) Alsberg, B. K.; Woodward, A. M.; Winson, M. K.; Rowland, J. J.; Kell, D. B. *Anal. Chim. Acta* **1998**, *368*, 29−44.
(30) Hemel, J. B.; Hindriks, F. R.; Vanderslik, W.; Vandervoet, H. *Anal. Chim. Acta* **1989**, *220*, 119−134.
(31) Dunn, W. J.; Emery, S. L.; Glen, W. G.; Scott, D. R. *Environ. Sci. Technol.* **1989**, *23*, 1499−1505.
(32) Ogorman, T. W.; Woolson, R. F. *Am. Stat.* **1991**, *45*, 187−193.
(33) Rencher, A. C. *Commun. Stat.-Simul. Comput.* **1992**, *21*, 373−389.
(34) Ortiz, M. C.; Herrero, A.; Sanchez, M. S.; Sarabia, L. A.; Iniguez, M. *Analyst* **1995**, *120*, 2793−2798.
(35) Mallet, Y.; Coomans, D.; deVel, O. *Chemom. Intell. Lab. Syst.* **1996**, *35*, 157−173.
(36) Nath, R.; Rajagopalan, B.; Ryker, R. *Comput. Oper. Res.* **1997**, *24*, 767−773.
(37) LeRoux, N. J.; Steel, S. J.; Louw, N. *J. Stat. Comput. Simul.* **1997**, *59*, 195−219.
(38) Breiman, L.; Friedman, J. H.; Olshen, R. A.; Stone, C. J. *Classification and regression trees*; Wadsworth, Inc.: Pacific Grove, CA, 1984.
(39) Bishop, C. M. *Neural networks for pattern recognition*; Clarendon Press: Oxford, 1995.
(40) Coomans, D.; Massart, D. L. *Anal. Chim. Acta* **1982**, *138*, 15.
(41) Rannar, S.; Lindgren, F.; Geladi, P.; Wold, S. *J. Chemom.* **1994**, *8*, 111−125.
(42) Lindgren, F.; Geladi, P.; Rannar, S.; Wold, S. *J. Chemom.* **1994**, *8*, 349−363.
(43) Lindgren, F.; Geladi, P.; Berglund, A.; Sjostrom, M.; Wold, S. *J. Chemom.* **1995**, *9*, 331−342.
(44) Seasholtz, M. B.; Kowalski, B. *Anal. Chim. Acta* **1993**, *277*, 165−177.
(45) Chatfield, C. *J. R. Stat. Soc. Ser. A* **1995**, *158*, 419−466.
(46) Wolpert, D. H.; Macready, W. G. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67−82.
(47) Martens, H.; Næs, T. *Multivariate Calibration*; John Wiley: Chichester, 1989.

and a conversion to class memberships is needed. Since the class membership information is contained in the column index information of **Y**, we must look for column elements with large predicted *absolute* values. If $\hat{\mathbf{y}}$ is a row in the estimated $\hat{\mathbf{Y}}$ we find the class membership as the column index that satisfies max(|$\hat{\mathbf{y}}$|). || indicates the absolute value of each element of the vector.

The VS-DPLS Algorithm. The extension of DPLS presented here to include variable selection is simple and straightforward. The basic idea is to truncate to zero most of the elements in the PLS weight vectors **w** and keep only the most important ones, i.e., the ones with the largest absolute values. The absolute value is chosen because important variables can be identified by either large positive or negative values. The PLS2 weight vectors are the result of finding directions that maximize the covariance between an **X** matrix score vector (for factor a), $\mathbf{t}_a = \mathbf{X}_{a-1}\mathbf{w}_a$, and a **Y** matrix score vector (for factor a), $\mathbf{u}_a = \mathbf{Y}_{a-1}\mathbf{q}_a(\mathbf{q}_a\mathrm{T}\mathbf{q}_a)^{-1}$, where $\mathbf{w}_a$ is the PLS weight vector and $\mathbf{q}_a$ is the **Y** loading vector. Our strategy for variable selection will thus be to keep the k elements in $\mathbf{w}_a$ with the largest absolute values and set all remaining variables to zero; see ref 47 (p 160).

Let Ω be a function that returns a vector containing zero elements everywhere except for elements corresponding to the k largest elements of |**w**|. Let Φ be the function that makes a vector **v** orthogonal to the column vectors in **V**. The suggested VS-DPLS algorithm in pseudocode is as follows:

(1) FOR $a = 1$ to A_{max}
(2) select a column vector **u** from matrix **F**
(3) $\mathbf{t}_0 = \mathbf{u}$
(4) WHILE not STOP
(5) $\mathbf{w}^T = \mathbf{u}^T\mathbf{E}$
(6) $\mathbf{w}^T = \Omega(\mathbf{w}^T,k)$
(7) IF a > 1
(8) $\mathbf{w} = \Phi(\mathbf{w},\mathbf{W})$
(9) END;
(10) $\mathbf{w}^T = \mathbf{w}^T/(\mathbf{w}^T\mathbf{w})^{1/2}$
(11) $\mathbf{t} = \mathbf{E}\mathbf{w}$
(12) IF a > 1
(13) $\mathbf{t} = \Phi(\mathbf{t},\mathbf{T})$
(14) END
(15) $\mathbf{q}^T = \mathbf{t}^T\mathbf{F}/\mathbf{t}^T\mathbf{t}$
(16) $\mathbf{u} = \mathbf{F}\mathbf{q}/(\mathbf{q}^T\mathbf{q})$
(17) $\delta\mathbf{t} = \mathbf{t} - \mathbf{t}_0$
(18) IF $(\sum_i(\delta\mathbf{t}_i)^2)/\mathbf{t}^T\mathbf{t} <$ conv
(19) STOP = TRUE
(20) END
(21) END
(22) $\mathbf{p}^T = \mathbf{t}^T\mathbf{E}/\mathbf{t}^T\mathbf{t}$
(23) (Storage of matrices)
(24) $\mathbf{E} = \mathbf{E} - \mathbf{t}\mathbf{p}^T$
(25) $\mathbf{F} = \mathbf{F} - \mathbf{t}\mathbf{q}^T$
(26) END
(27) $\mathbf{B} = \mathbf{W}(\mathbf{P}^T\mathbf{W})^{-1}\mathbf{Q}^T$
(28) $\mathbf{b}_0 = \mathbf{y}_m - \mathbf{x}_m\mathbf{B}$

A_{max} is the maximum number of PLS factors, **E** and **F** are residual matrices for the X and Y spaces, respectively. In step 6 of the algorithm, the selection of the k largest variables is made. The remaining variables are set equal to zero. Steps 8 and 13 are included to ensure that the columns in the **W** matrix are orthonormal and the columns in the **T** matrix are orthogonal. All the other steps in the algorithm are identical with ordinary PLS2.

It should be noted that even though we select the k largest elements in $\mathbf{w}_a$, the final regression coefficient matrix **B** will in general contain more than k variables different from zero. This is due to the step that calculates **B** from **W**, **P**, and **Q**. Let $\mathbf{R} = (\mathbf{P}^T\mathbf{W})^{-1}\mathbf{Q}^T$. In most cases, each element $\mathbf{R}_{ij}$ will be different from zero. The dimensions of **R** is [$A \times K$], where A is the total number of PLS factors and K is the total number of classes. The dimensions of the weights matrix **W** is [$M \times A$]. Note that each column in **W** is sparse; i.e., it contains a majority of zeros. The maximum number of elements in each column different from zero is k. Each element in the resulting B coefficient matrix $\mathbf{B} = \mathbf{WR}$ has elements

$$\mathbf{B}_{ij} = \mathbf{w}_i^T\mathbf{r}_j \tag{6}$$

where $\mathbf{w}_i^T$ is the ith row vector in **W** and $\mathbf{r}_j$ is the jth column in **R**. Assume a row in **W** that contains an element different from zero. If all the elements in **R** are nonzero, it is trivial to see that all K columns in that row will also have values different from zero. Thus, we have

$$s_v \le A_{opt}k_{opt} \tag{7}$$

where s_v is the number of selected variables different from zero in **B**, A_{opt} is the optimal number of PLS factors, and k_{opt} is the optimal number of retained variables different from zero. Both these values are determined by cross-validation. If **R** contains elements equal to or close to zero that coincide with a nonzero element of **W**, s_v will be less than $A_{opt}k_{opt}$.

To obtain the final VS-DPLS model, we need to estimate two parameters: k_{opt} and A_{opt}. The following strategy is used for the suggested algorithm:

FOR $k = 1$ to k_{max} DO

- use cross-validation to find optimal number of PLS factors
- store model values and PRESS value

END

- select k_{opt} that corresponds to the lowest PRESS value
- apply the selected VS-DPLS model on an unseen validation (test) set

Measuring Uniqueness of Selected Variables by Pruning. It is unlikely that the selected variables from VS-DPLS will be the only variables that give rise to the same prediction error. One probable cause of this nonuniqueness of the selected variables is the redundancy of the information contained in the original

variables. For both the FT-IR and PyMS data (as used in this article) there is a high correlation between variables; for FT-IR the neighboring variables are highly correlated with one another, while for PyMS the correlations are separated according to how a particular compound fragments on pyrolysis. Despite this, it is to be expected that the underlying features that correlate with the variation in the dependent variable (Y) are to some extent localized in the spectrum. For instance, if a Y variable is strongly correlated with the concentration of a single analyte, it is reasonable to assume that the compound in question may have a few localized regions in the spectral domain. We expect in particular to see this for FT-IR spectra, but not to the same degree in PyMS spectra. Consequently, the *distribution function* of the selected variables would reflect important spectral regions for classification.

Here we suggest a simple approach to establish the approximate locations of important variable regions necessary for maintaining an optimal prediction model. Let U_0 be the set of indexes that contains the selected variables after running VS-DPLS on the whole data set $\mathbf{X}_0$. Let V_0 be the set of all indexes, {1, 2, ..., m}, where m is the total number of variables. A new data set is made where the selected variables in the *previous* VS-DPLS have been removed (i.e., columns with indexes U_0 in data matrix $\mathbf{X}_0$ are set to zero). $\mathbf{X}_1$ has columns with V_1 ($V_1 = V_0 - U_0$) indexes different from zero. A VS-DPLS analysis of $\mathbf{X}_1$ will produce a new set U_1 of indexes of selected variables. A series of similar analyses can be made where several index sets U_0, U_1, ..., U_g are obtained (g is the maximum number of analyses). This procedure will be referred to as *DPLS-pruning*. Note that

$$U_i \cap U_j = \varnothing, \forall\ i \neq j$$

By looking at the distribution of the indexes selected in U_i,$i \in$ {1, 2, ..., g} it is possible to obtain an indication of regions in the spectra that are necessary for optimal prediction accuracy. Another interesting point is that we can visualize the *difference* in index distribution between optimal and nonoptimal classification models. This provides an opportunity for detecting differences that are more robust than just comparing two particular VS-DPLS models.

The natural method for visualizing the distribution would be the *histogram distribution* of selected variables. However, we have chosen to use a somewhat more advanced method based on kernel density estimates (KDE).[48] There are problems with using histograms since their appearance is dependent on the choice of origin (the starting position of the first interval to contain data) and the bin width (the interval defining a region where a frequency count is performed). KDE avoids several of these problems and can in the simplest cases be thought of as smoothed histograms. The KDE is obtained by placing a peak function $K(x)$ (only the one-dimensional cases will be discussed in the present article) at each data point location and a summing of all the heights is performed. The kernel function $K(x)$ integrates to 1 and its spread is determined by the parameter h, which is analogous to the bin width used in histograms. The estimated density function $\hat{f}(x)$ of the data is written as

$$\hat{f}(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{8}$$

In contrast to the histogram distribution, the shape of $\hat{f}(x)$ does not depend on the choice of origin, but it does, however, depend on the choice of the h parameter. Large h values tend to oversmooth whereas small h values tend to undersmooth. In the present article, an h value based on an estimate of the roughness of the true density function is used.[49] The density distributions in this article are calculated using the MATLAB Kernel Density Estimation Toolbox by Beardah and Baxter.[48]

Rule Induction Methods. Rule induction attempts to find hyperplanes that partition the space of sample objects into regions of single class memberships. These hyperplanes are interpreted as *rules* which are derived from the training set. The most common strategy employed to find these rules is based on a *recursive splitting* of the original data set into smaller subsets where each subset contains objects belonging to as few different classes as possible. A stopping criterion for the recursive search is the "purity" of a subset, i.e., the distribution among the classes of the objects within the set. It has been found that the concept of *entropy*[50] is very efficient as a measure of "purity". For each subset of objects there is a vector of fractions $\mathbf{p} = [p_1, p_2, ..., p_K]$ of the objects belonging to the K different classes. The fraction p_i is computed as $p_i = n_i(s)/n$ where $n_i(s)$ is the number of objects belonging to class i in subset s and n is the total number of objects in subset s. It is common[38] to interpret such fractions as the *probability* of finding an object belonging to class i in the subset.

Intuitively, a subset consisting of objects from one class only will have the highest possible "purity" and the vector $\mathbf{p}$ of probabilities will have a structure $\mathbf{p}_{\min} = [0, 0, ..., 1, ..., 0.0]$. The most impure vector $\mathbf{p}$ will correspond to the case where there are equal fractions of each class; i.e., $p_j = 1/K$. The entropy of $\mathbf{p}$

$$H(\mathbf{p}) = -\sum_{i=1}^{K} p_i \log(p_i) \tag{9}$$

has properties in accordance with our intuitive understanding of "impurity": $H_{\min}(\mathbf{p}) = 0$ and $H_{\max}(\mathbf{p}) = \log_2(K)$ when $p_i = 1/K$. Thus, ensuring the highest purity in a subset corresponds to *minimizing* $H(\mathbf{p})$ by selecting an optimal partitioning.

There are in general two different types of hyperplanes generated in rule induction: those that are parallel to the original variables and those that are not. A method generating parallel hyperplanes is sometimes referred to as *univariate* rule induction. Rules that are based on hyperplanes not parallel to the original variables are generated from *multivariate* rule induction. In the univariate case a single variable x_i at each recursion step is found that gives rise to the purest subsets (i.e., those that have minimum entropy). In univariate rule induction, a partitioning of the input feature space can be formulated as a question like, "Is $x_i < c$?", where c is some value chosen from the *finite set* of values variable

(48) Beardah, C. C.; Baxter, M. J. In *Analecta Prehistorica Leidensia 28, Interfacing the Past, Computer Applications and Quantitative Methods in Archaeology CAA95*; Kammermans, H., Fennema, K., Eds.; 1996.

(49) Baxter, M. J.; Beardah, C. C. Department of Mathematics, Statistics and operational research. The Nottingham Trent University, Nottingham, England, 1995.

(50) Shannon, C. E. *Bell Syst. Tech. J.* **1948**, *379*, 379−423, 623−656.

x_i has among the N calibration objects. All the objects that satisfy the question are grouped into one subset and those that do not into another.

In *multivariate rule induction* a partition of the input feature space is found that depends on a *linear combination* of all the variables instead of just using one variable. It is possible to interpret the nonparallel hyperplane rule as a question of the form, "Is $\sum_{j=1}^{n} w_j x_j \leq c$?". This type of partitioning of the data space is particularly useful if there are any collinearities between the variables. The aim in this case is not to find the single best variable but to find a *vector* **w** (hyperplane) that best separates the data set into pure subsets. Both the multivariate CART (referred to in the text as Breiman CART)[38] and the OC1 method[51] used in this article are based on this principle. In this article, we use the OC1 program[51] for all the CART methods.

The FuRES method[18,52] is also a multivariate rule induction approach, but it uses a fuzzy set[53] description of object locations with respect to the decision hyperplane. See also ref 16 for a comparison of these methods.

It should be noted that none of the rule induction methods per se perform any variable selection. However, both uni- and multivariate CART models often produce small models which resemble the effect of a variable selection. The rules observed in multivariate CART models are not perfectly sparse as for VS-DPLS models but are dominated by a small subset of large coefficients. The remaining coefficients are usually close to zero. It is therefore possible to perform qualitative analysis of which variables are important for the classification.

In the FuRES implementation available to us, variable selection is not performed. Unfortunately, the FuRES rules generated are in general not like the multivariate CART rules in that a few variables dominate. This method is here only included to provide an independent prediction error for the unseen validation data in the various data sets.

EXPERIMENTAL SECTION

Sample Preparation. *Data sets 1 and 2. Eubacterium Samples.* Four replicates for each sample is used. Four *Eubacterium timidum* (Ta−Te), four *Eubacterium infirmum* (**1a**−**1d**), four *Eubacterium exiguum* (**2a**−**2e**), five *Eubacterium tardum* (Na−Ne), and five eubacterial hospital isolates (Ha−He) were prepared as described previously.[14] In total we have 88 spectra (4 × 22 samples) from FT-IR analysis (data set 1). Data set 2 consists of PyMS spectra of the same type of bacteria but with fewer samples.[18] In total, we have 72 (4 × 18) PyMS spectra. *E. timidum*, *E. infirmum*, *E. exiguum*, and *E. tardum* are referred to in this article as classes 1−4, respectively.

Data Set 3. Urinary Tract Infection Organisms. Twenty-two *Escherichia coli* (Ea−Eq), *Proteus mirabilis* (Pa−Pj), 15 *Klebsiella* (Ka−Kj), 15 *Pseudomonas aeruginosa* (Aa−Aj), and 17 enterococci (Ca−Cl) were isolated from the urine of patients with urinary tract infection (UTI) and prepared as described previously.[12] In total we have 336 (4 × 84) from the FT-IR analysis. *E. coli*, *P. mirabilis*, *Klebsiella*, *P. aeruginosa*, and enterococci are in this article referred to as classes 1−5, respectively.

Data Set 4. Samples of Milks. Mixtures of 3 milks from cow, goat, and ewe (11 from each type of milk) were prepared, which differed in their fat content as described previously.[54] In total, we have 99 (3 × 33) spectra from the PyMS analysis. Cow, goat and ewe are in this article referred to as classes 1−3, respectively.

Pyrolysis Mass Spectrometry. Aliquots (5 μL) of the above milk and *Eubacterium* spp. samples were evenly applied to clean iron−nickel foils which had been partially inserted into clean pyrolysis tubes. Samples were run in triplicate. Prior to pyrolysis, the samples were oven-dried at 50 °C for 30 min and the foils were then pushed into the tubes using a stainless steel depth gauge so as to lie 10 mm from the mouth of the tube. Viton O-rings were next placed approximately 1 mm from the mouth of each tube. PyMS was performed on a Horizon Instrument PyMS-200X (Horizon Instruments Ltd., Heathfield, U.K.). For full operational procedures, see refs 2, 13, 55, and 56. Conditions used for each experiment involved heating the sample to 100 °C for 5 s followed by Curie point pyrolysis at 530 °C for 3 s with a temperature rise time of 0.5 s. Data were normalized as a percentage of the total ion count to remove the influence of sample size.

Diffuse Reflectance−Absorbance FT-IR Spectroscopy. Aliquots (5 μL) of the above *Eubacterium* spp. and bacterial UTI samples were evenly applied onto a sand-blasted aluminum plate. Prior to analysis the samples were oven-dried at 50 °C for 30 min. Samples were run in triplicate. The FT-IR instrument used was the Bruker IFS28 FT-IR spectrometer (Bruker Spectrospin Ltd., Banner Lane, Coventry, U.K.) equipped with an MCT (mercury−cadmium−telluride) detector cooled with liquid N_2. The aluminum plate was then loaded onto the motorized stage of a reflectance TLC accessory. The IBM-compatible PC used to control the IFS28 was also programmed (using OPUS version 2.1 software running under IBM O/S2 Warp provided by the manufacturers) to collect spectra over the wavenumber range 4000−600 cm^{-1}. Spectra were acquired at a rate of 20 s^{-1}. The spectral resolution used was 4 cm^{-1}. To improve the signal-to-noise ratio, 256 spectra were coadded and averaged. The digital sampling parameter was set such that each spectrum was represented by 882 points. Spectra were displayed in terms of absorbance as calculated from the reflectance−absorbance spectra using the Opus software.[57,58]

ASCII data were exported from the Opus software used to control the FT-IR instrument and imported into MATLAB version 5.2 (The MathWorks, Inc., 24 Prime Par Way, Natick, MA), which runs under Microsoft Windows NT on an IBM-compatible PC. To minimize problems arising from baseline shifts, the following procedure was implemented: (i) the spectra were first normalized so that the smallest absorbance was set to 0 and the highest to +1 for each spectrum; (ii) next these normalized spectra were detrended by subtracting a linearly increasing baseline from 4000 to 600 cm^{-1}. Data set 3 was in addition to detrending also

(51) Murthy, S. K.; Kasif, S.; Salzberg, S. *J. Artif. Intell. Res.* **1994**, *2*, 1−32.
(52) Harrington, P. d. B. *Chemom. Intell. Lab. Syst.* **1993**, *19*, 143−154.
(53) Zadeh, L. A. *Fuzzy sets, fuzzy logic, and fuzzy systems: Selected papers*; World Scientific: River Edge, NJ, 1996.
(54) Goodacre, R. *Appl. Spectrosc.* **1997**, *51*, 1144−1153.
(55) Goodacre, R.; Neal, M. J.; Kell, D. B. *Anal. Chem.* **1994**, *66*, 1070−1085.
(56) Timmins, É. M.; Goodacre, R. *J. Appl. Microbiol.* **1997**, *83*, 208−218.
(57) Glauninger, G.; Kovar, K. A.; Hoffmann, V. *Fresenius J. Anal. Chem* **1990**, *338*, 710−716.
(58) Winson, M. K.; Goodacre, R.; Woodward, A. M.; Timmins, É. M.; Jones, A.; Alsberg, B. K.; Rowland, J. J.; Kell, D. B. *Anal. Chim. Acta* **1997**, *348*, 273−282.

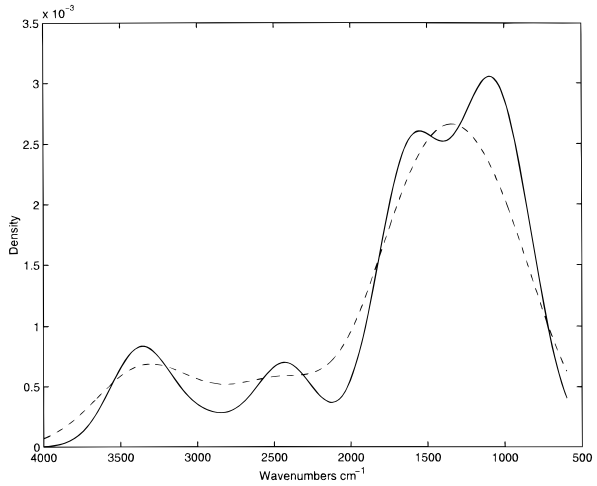

Figure 1. Results of DPLS-pruning on data set 1. The results from DPLS-pruning show clearly four regions which dominate for models with 0% prediction error. Solid line indicates optimal models. Dashed line indicates nonoptimal models.

numerically differentiated using a Savitzky−Golay method[59] with a five-point smoothing filter.

RESULTS

Data Set 1, *Eubacterium*/FT-IR. The calibration set contains 52 spectra and the validation (test) set contains 36. The VS-DPLS model obtained from the training set when applied to the unseen validation set produced a 0% prediction error. The optimal model was selected using $k_{opt} = 1$ and $A_{opt} = 7$. We also have 0% prediction error on the calibration set. In total, seven variables were selected to discriminate between four classes. The seven wavenumbers are 3383, 2792, 1561, 1206, 1148, 1086, and 1036 cm^{-1}.

To measure the uniqueness of the seven selected variables, DPLS-pruning as described above was used. Figure 1 shows the results from the pruning experiment. The distribution function of variables selected by VS-DPLS models that have 0% prediction error are drawn with a solid line and those that have not with a dashed line. For the optimal models, four regions are clearly visible. The results from nonoptimal models indicate that variables selected in the region around 2442 cm^{-1} are necessary for maintaining a perfect prediction. For the VS-DPLS analysis described above, one of the selected variables is 2792 cm^{-1}.We also see that the peak located around 1375 cm^{-1} for nonoptimal prediction models splits into two peaks for the optimal models. These two peaks are located in the neighborhood of 1575 and 1118 cm^{-1} (note that two of the selected variables were 1561 and 1148 cm^{-1} for the VS-DPLS analysis above). The univariate CART produces a model with three variables: 1225, 3440, and 3595 cm^{-1}.[60] All three CART algorithms had a prediction error of 19.4%. Wavenumber 1225 cm^{-1} from univariate CART modeling is close to 1206 cm^{-1} found from the VS-DPLS analysis. The 3440 cm^{-1} from CART is in the neighborhood of the VS-DPLS variable 3383 cm^{-1}. One of multivariate rules for the Breiman CART is dominated by the following variables: 1225, 1696, 1480, 3456, 1152, 1214, 1457, 1700, 3529, 1692 cm^{-1}.[60] The OC1 CART method produced a model similar to the univariate CART and had the same prediction error. The reference method FuRES has a surprisingly high prediction error of 16.7%. The full DPLS also performs worse than VS-DPLS with a prediction error of 5.6%.

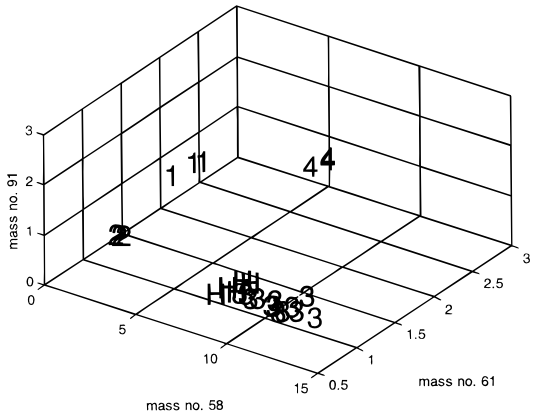

Figure 2. Distribution of objects (PyMS spectra) for data set 2 by plotting the three mass intensities (58, 61, 91) for the validation set. Note that the unknown hospital isolates clustered close to class 3.

Data Set 2, *Eubacterium*/PyMS. The bacteria analyzed in this data set are the same as those for *Eubacterium*/FT-IR, but here pyrolysis mass spectrometry is used instead of Fourier transform infrared spectroscopy. The VS-DPLS model was constructed by analysis of the 45 objects in the calibration set. The performance of the optimal model was tested on the 27 objects in the unseen validation set. Note that the number of objects is different from data set 1. The optimal model was found for $k_{opt} = 1$ and $A_{opt} = 3$. The prediction error was 0%. Three masses were selected: 58, 61, and 91. The reason for the VS-DPLS selection of mass 91 was to enable separation between some class 1 and 2 objects in the training set. These three masses are also confirmed by earlier studies of the same data set using genetic programming (GP).[17] Since only three variables were selected to produce perfect predictions, it was possible to plot the distribution of the validation set objects in this 3D space; see Figure 2. As can be seen, we have a perfect class separation where the unknown hospital isolates, marked "H", were predicted to belong to the *E. exiguum* species (class 3), a result seen previously using the rather uninterpretable neural network method.[14] Note that in Figure 2 mass 91 is not necessary to obtain perfect separation between the classes in the validation set.

The spectral information in PyMS is not localized in the same manner as in FT-IR. For this technique, there are (apart from isotopic information) in general no major correlations between neighboring data points. The effect on the estimated density distribution function will be an increased smoothing since we do not have a strong spatial localization. The result of a DPLS-pruning process on this data set is shown in Figure 3. In this case, the optimal models have all 0% prediction error. Nonoptimal models seem to have an excess of variables selected in the region 150−200 m/z compared to the optimal models.

The univariate CART has a prediction error of 3.7%, which is surprisingly better than the multivariate Breiman CART (7.4%). The selected masses are 52, 60, and 58. In fact, all the CART methods selected these three masses as important. For comparison, the VS-DPLS model selected 61 and 58 as important masses.

(59) Savitzky, A.; Golay, M. J. E. *Anal. Chem.* **1964**, *36*, 1627−1633.
(60) Alsberg, B. K.; Wade, G. W.; Goodacre, R. *Appl. Spectrosc.* **1998**, *52*, 72−102.

Table 1. Summary Table of the VS-DPLS Results

data set	data type	A_{opt}	k_{opt}	no. sel var	pred error, %	opt reg	nonopt reg
1	FT-IR	7	1	7	0	1118, 1575, 2442, 3358	1375, 3318
2	PyMS	3	1	3	0	86	79
3	FT-IR	11	4	40	2.5	1276, 2927, 3530	1061, 2275, 3265
4	PyMS	4	1	4	0	85	no peaks

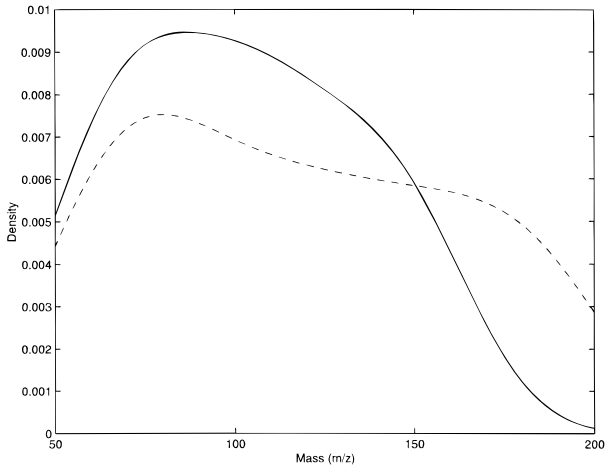

Figure 3. DPLS-pruning of data set 2. Solid line indicates optimal models. Dashed line indicates nonoptimal models.

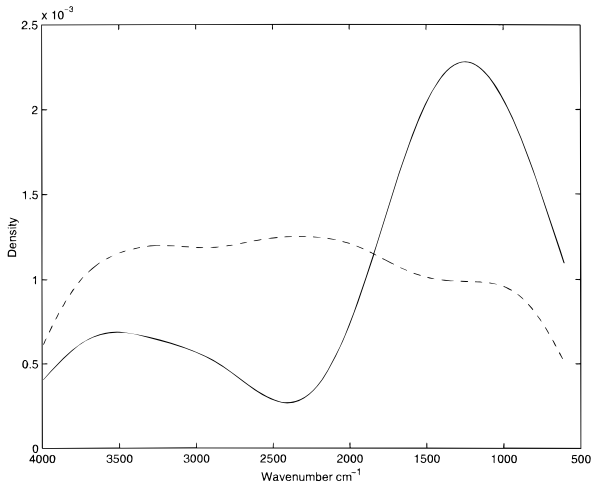

Figure 4. DPLS-pruning results for data set 3. Since we do not have 0% prediction error, the optimal threshold for DPLS models is set to 5% percent (solid line).

Table 2. Selected Variables for the Four Different Data Sets

data set	units	selected variables by VS-DPLS
1	cm^{-1}	3383, 2792, 1561, 1206, 1148, 1086, 1036
2	m/z	58, 61, 91
3	cm^{-1}	2850, 2846, 1754, 1750, 1746, 1723, 1719, 1708, 1704, 1700, 1696, 1681, 1669, 1665, 1654, 1638, 1627, 1565, 1561, 1557, 1549, 1519, 1511, 1507, 1430, 1426, 1383, 1202, 1198, 1167, 1164, 1121, 1117, 1005, 1001, 974, 971, 940, 808, 805
4	m/z	55, 60, 97, 126

The FuRES rules are more complicated (prediction error 0%), but it is possible to identify some of the same variable numbers with high absolute coefficient values. We believe it would be possible to extend FuRES to take variable selection into account by truncating to zero unimportant variables in the decision plane vectors which are similar to the **w** vectors in PLS.

Data Set 3, Urinary Tract Infection Organisms/FT-IR. This data set consists of 100 calibration and 236 validation objects. To remove baseline effects, a five-point Savitzky−Golay numerical differentiation[59] of the spectra was performed. The optimal VS-DPLS model was found to have 2.5% prediction error. The total number of selected wavenumber variables is 40 ($k_{opt} = 4$, $A_{opt} = 11$). The selected variables are listed in Table 2. Since we do not have 0% prediction error, the threshold for "optimal" DPLS models was set to 5% percent. The results from this analysis is shown in Figure 4 where the solid line signifies the distribution of selected variables in optimal models (prediction errors <5%). These optimal models have three maximums at 1276, 2927, and 3530 cm^{-1}. The dashed line signifies nonoptimal models (>5% prediction error). As can be seen from the figure, the nonoptimal model density distribution function is much flatter. However, it is possible to identify three broad peaks located at approximately the same positions as for the optimal models: 1061, 2275, and 3265 cm^{-1}. Note also that the density distribution for the optimal models is very smooth compared to the corresponding distribution in data set 1. This is probably related to the fact that more variables are needed for data set 3 to establish a good classification model.

The most important variables selected by univariate CART are 1164, 2862, 3900, 2958, and 2171 cm^{-1}. Only 1164 cm^{-1} was also chosen by VS-DPLS. All the CART models have higher prediction error than VS-DPLS. FuRES and full DPLS in comparison have 0% prediction error.

Data Set 4, Milk/PyMS. The calibration set consists of 27 objects and the validation set consists of 72 objects. The optimal VS-DPLS model had 0% prediction error using only four selected mass variables: 55, 60, 97, and 126 ($k_{opt} = 1$ and $A_{opt} = 4$). A systematic DPLS-pruning was performed and the estimated density function is shown in Figure 5. Note that all the optimal VS-DPLS models for this data set seem to cluster around mass 85. The density distribution function for the nonoptimal VS-DPLS models (dashed line) is markedly flatter and appears to make use of a much wider range of variables.

Univariate and OC1 CART selected masses 62 and 97 as important. The prediction error is 6.9%. The Breiman CART method has a prediction error of 1.4%. In addition to mass 62 as important in separating between classes 1 and 3, it also has a multivariate rule which is dominated by the masses 64, 78, 63, 97, and 90. Again, both FuRES and full DPLS have 0% prediction error.

Please see Table 1 for a summary of all the VS-DPLS results. The abbreviations and symbols used in this table are as follows:

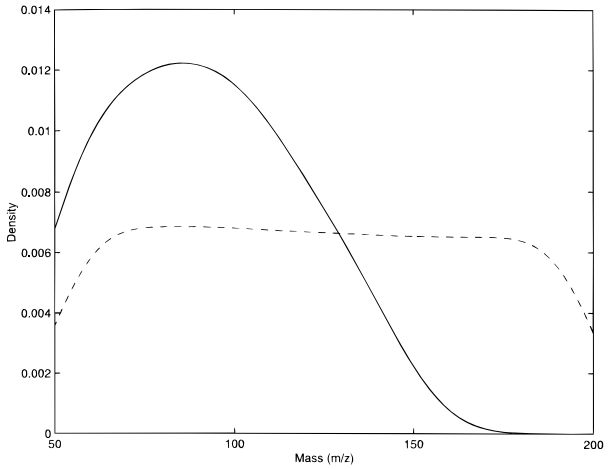

Figure 5. DPLS-pruning experiments of data set 4. Solid line indicates optimal models. Dashed line indicates nonoptimal models.

A_{opt} is the number of PLS factors for the optimal VS-DPLS model based on the calibration set; k_{opt} is the number of variables retained from each **w** vector; no. sel var is the number of selected variables in the final model, pred error signifies the prediction error of the unseen validation set, opt reg signifies the most prominent peaks of the probability density distribution function for optimal models from a VS-DPLS pruning experiment; nonopt reg signifies the peaks for nonoptimal models. For data set 4, the density distribution function was almost flat and thus does not contain any prominent peak regions. See Table 3 for a comparison of the prediction errors between the different variable selection methods used.

DISCUSSION

One important reason for why it is desirable to do variable selection is to produce parsimonious models. Such models are usually easier to interpret and often statistically more robust.[44] In some cases, there are ("noise") variables in the data set that actually cause a reduction in the predictive ability. Good variable selection algorithms should be able to avoid such variables and thus improve on the predictive ability compared to the full analysis.

Table 3. Prediction Errors for the Methods Applied to the Four Data Sets[a]

data set	CART	Breiman CART	OC1 CART	FuRES	VS-DPLS	full DPLS
1	19.4	19.4	19.4	16.7	0.0	5.6
2	3.7	7.4	3.7	0.0	0.0	0.0
3	10.6	14.8	10.6	0.0	2.5	0.0
4	6.9	1.4	6.9	0.0	0.0	0.0

[a] CART is the univariate classification and regression trees method. FuRES is the fuzzy rule building expert system method. Breiman and OC1 CART signify algorithms that use multivariate rule induction.

Our investigations suggest that the new VS-DPLS algorithm compared to full DPLS produces more parsimonious classification models which have better or similar prediction ability. We also see from Table 3 that VS-DPLS compared to both uni- and multivariate CART algorithms show significantly lower percentage prediction errors.

We therefore suggest that the VS-DPLS algorithm is an efficient method for solving problems in rapid classification using spectral data from, for example, FT-IR, PyMS, or Raman spectroscopy. There are many important application areas, such as in biotechnology, food science and medicine, where there is an increasing interest in using spectroscopy for screening. But in order for efficient screening methods to be widely used, it is necessary to produce low-cost and accurate instruments. It is here that parsimonious classification models have an advantage over full spectral models since a smaller number of wavelengths are required. It thus opens up the possibly for building simpler instruments tailored to solve specific screening problems.

ACKNOWLEDGMENT

We thank the U.K. BBSRC, GlaxoWellcome, and Bruker Spectrospin Ltd. for financial support. R.G. is indebted to the Wellcome Trust for financial support (Grant 042615/Z/94/Z). The authors also thank Professor Peter de Harrington for providing the FuRES program.

Received for review May 8, 1998. Accepted July 23, 1998.

AC980506O