Practical comparison of sparse methods for classification of Arabica and Robusta coffee species using near infrared hyperspectral imaging

Rosalba Calvini[a], Alessandro Ulrici[a], Jose Manuel Amigo[b]

[a] *Department of Life Sciences, University of Modena and Reggio Emilia, Padiglione Besta, Via Amendola 2, 42122 Reggio Emilia, Italy*

[b] *Department of Food Science, Faculty of Sciences, University of Copenhagen, Rolighedsvej 30, DK-1958 Frederiksberg C, Denmark*

Abstract

In the present work sparse-based methods are applied to the analysis of hyperspectral images with the aim at studying their capability of being adequate methods for variable selection in a classification framework. The key aspect of sparse methods is the possibility of performing variable selection by forcing the model coefficients related to irrelevant variables to zero. In particular, two different sparse classification approaches, i.e. sPCA+kNN and sPLS-DA, were compared with the corresponding classical methods (PCA+kNN and PLS-DA) to classify Arabica and Robusta coffee species. Green coffee samples were analyzed using near infrared hyperspectral imaging and the average spectra from each hyperspectral image were used to build training and test sets; furthermore a test image was used to evaluate the performances of the considered methods at pixel-level. In our case, sparse methods led to similar results as classical methods, with the advantage of obtaining more interpretable and parsimonious models. An important result to highlight is that variable selection performed with two different sparse classification approaches converged to the selection of same spectral regions, which implies the chemical relevance of those regions in the discrimination of Arabica and Robusta coffee species.

Keywords

Hyperspectral Imaging; HIS-NIR spectroscopy, Sparse methods, Variable Selection, Green coffee beans

1. INTRODUCTION

Hyperspectral imaging (HSI) combines classical spectroscopic systems with imaging devices in order to obtain both spectral and spatial information from a sample. The resulting hyperspectral images are three dimensional data arrays, often referred to as datacubes, where each pixel contains one spectrum [1]. Each hyperspectral image can be formed by thousands to millions of spectra, and each spectrum can be composed by more than 100 wavelengths. Consequently, when dealing with such a big amount of data a multivariate approach is necessary in order to extract the relevant information contained in the datacube. For this purpose, classical multivariate data analysis techniques have been adapted to the elaboration of hyperspectral images showing great potential and benefit for extracting the desired information [2, 3].

The large amount of data contained in the datacube represents at the same time both an advantage and a drawback of HSI. On one hand, the large amount of pixels contained in each image allows a detailed representation of the analyzed sample; on the other hand, it is necessary to face data handling issues such as storage problems and long computational times [4].

Therefore, data reduction is frequently needed in order to preserve only the useful information contained in high-dimensional data [5, 6]. When dealing with many images, the most common way to perform data reduction consists of extracting the average spectra form each image or from user-defined Regions of Interest (ROI), to be used for further analysis on the whole dataset. In this manner, data reduction is performed in the $\{x, y\}$ spatial dimensions, without affecting the spectral dimension, λ.

Alternatively, the spectral dimension can be reduced by selecting only the informative wavelengths, without loss of relevant features. As in the case of point-wise NIR spectroscopy, the identification of the spectral variables that provide useful information (and the resulting elimination of the signal regions containing noise and information not pertinent to the problem at hand) can lead to better results in classification or calibration issues, and simplifies the chemical interpretation of the results. Moreover, in the specific case of hyperspectral imaging, the selection of spectral variables is essential for the identification of key wavelengths in the development of multispectral imagining systems for on-line applications or for portable devices.

In some situations variable selection can be simply based on a deep knowledge of the spectroscopic properties of a sample, but the use of appropriate algorithms based on multivariate statistics generally leads to better results [7].

To this aim, many variable selection techniques have been proposed in the literature, such as interval-Partial Least Squares (iPLS) [8-10], Genetic Algorithms (GA) [11], and Wavelet

Transform (WT) [12, 13], which work in different manners and are suitable for different applications.

Though very effective, these methods often require long computational times and the optimization of many parameters or, as in the case of iPLS, the selected regions strongly depend upon the defined interval size. Sparse methods have been developed in order to face problems concerning calibration or classification of high-dimensional data, mostly in the field of bioinformatics where the variables usually consist of thousands of genes. Sparse methods have been widely applied also in statistic learning [14], analysis of biological data [15], metabolomics [16] and genomics [17]. Some applications of sparse methods to spectroscopic data in the field of food analysis and control are also available in the literature, for example concerning food-borne bacterial species [18], white wines discrimination [19] or virgin olive oil adulteration [20].

Generally, the term *sparse* refers to a matrix in which most of the elements are equal to zero. In the case of sparse methods the term sparse refers to the estimated parameter vector of a model, e.g. a regression vector, which is forced to contain many zeros. The main idea of sparse methods is to reduce the influence of the noise contained in irrelevant variables by forcing the model coefficients related to those variables to be equal to zero, consequently performing variable selection. Classical methods, like e.g. PCA or PLS, usually do not set the contribution of uninformative variables to zero, but only to a small absolute value. Therefore, if many variables have small contributions, their global influence could be considerable and lower the predictive ability of the model [21].

In general, sparsity is achieved by adding a penalty term to a given objective function in order to induce some model coefficients to be equal to zero. The level of sparsity to induce in the model is a user-defined parameter that needs to be optimized, for example by minimizing the cross validation error in a similar manner as for selection of number of components. Therefore, sparse methods require two parameters to be tuned, but once the optimal sparsity and dimensionality of the model is optimized , they allow performing classification (or regression) and variable selection at the same time.

Several sparse versions of classical methods have been developed for data exploration, regression and classification problems. Principal Component Analysis (PCA) is the most used chemometric tool for data exploration. This has originated several sparse variants of PCA (sPCA), where sparsity is induced both on the score and loading vectors [22-24], or only on the loading vectors [25-27]. For regression purposes, sparse versions of Partial Least Squares (sPLS) have been proposed by Chun and Keles, which make use of the elastic net approach [28], and by Lê Cao et al., whose method is instead based on the Lasso penalty [29]. Lê Cao et al. also proposed an extension of their sPLS for a

sparse version of Partial Least Squares Discriminant Analysis (sPLS-DA) [30], while Clemmensen et al. proposed a sparse version of Linear Discriminant Analysis [31].

Moreover, in the literature some research studies are reported where sparse methods are compared with other variable selection methods [18, 32] demonstrating the potential of sparse methods as a stable variable selection strategy.

In this context, the aim of the present work is to show a practical application of sparse methods such as sPCA-kNN [33] and sPLS-DA [30], which have been also compared with the corresponding non-sparse methods (PCA+kNN and PLS-DA) in terms of classification performances and model interpretability. In particular, sparse methods were applied in order to perform spectral variable selection on NIR hyperspectral images, with the aim of differentiating Arabica and Robusta green coffee beans. Arabica (*Coffea arabica*) and Robusta (*Coffea canephora*) coffee are the two main species used for the preparation of commercial coffee beverages, both alone or in blends. Due to its better taste and aroma, Arabica coffee is of higher quality than Robusta coffee, but it is more difficult to grow, even in function of its lower resistance to plant diseases, and therefore it is more expensive [34].

Classical point-wise NIR spectroscopy has been widely used to discriminate Arabica and Robusta species, both on green coffee [35, 36] and roasted coffee [37, 38]. Moreover some research works have been also reported where hyperspectral imaging is used to characterize these coffee species [39, 40]. For these reasons, in the present work green coffee samples were analyzed with NIR-HSI and the hyperspectral images were elaborated in order to test the ability of two different sparse classification methods, i.e., sparse PCA coupled with k-Nearest-Neighbors (sPCA+kNN) and sparse PLS-DA (sPLS-DA), to discriminate Arabica and Robusta coffee. Both these sparse classification methods allowed to perform classification and variable selection at the same time, leading to the identification of the informative NIR regions involved in the discrimination. The performances of the two sparse methods and their corresponding classical (non-sparse) counterparts (PCA+kNN and PLS-DA) were compared.

2. THEORY

2.1 Classical methods

As a first, very simple approach to discriminate between the average spectra obtained from the hyperspectral images acquired on Arabica and Robusta coffee samples, Principal Component Analysis (PCA) [41] was used in conjunction with k-Nearest-Neighbors (kNN) [42]. In PCA+kNN, firstly a PCA model is computed on the average spectra matrix and then the PCA scores are used

for kNN classification. In this manner, a data compression technique is coupled with a simple and robust classification tool. The number *k* of nearest neighbors to consider in kNN classification and the number of principal components are user-defined parameters.

Moreover, also Partial Least Squares Discriminant Analysis (PLS-DA) [44] was considered as an alternative classification method, since it is a fast linear method often leading to optimal performances. In this case, the proper number of latent variables of the model has to be selected for example by maximizing the classification efficiency in cross validation.

2.2 Sparse methods

Sparse methods are extensions of classical methods in which the parameter vectors of a model are forced to contain many zeros by adding a penalty term to the objective function of the considered method [21]. The algorithms used in this work for sPCA and sPLS-DA apply the Least absolute shrinkage and selection operator (Lasso) approach [45] to induce sparsity on the model coefficients.

In the following sections only a brief description of sPCA and of sPLS-DA algorithms is given; for a more detailed explanation the reader is referred to [33] and [30].

2.2.1 sPCA + kNN

Sparse Principal Component Analysis (sPCA) is a PCA-based model in which sparsity is induced on the model parameters: scores, loadings or both of them. Several algorithms were proposed to calculate sPCA models where sparsity is induced on the loadings; the one used in this work is Alternating Shrunken Least Squares (ASLS) [33].

The ASLS algorithm, for a fixed number of components *A*, estimates a sparse PCA solution of the following objective function:

$$\arg\min_{\boldsymbol{T},\boldsymbol{P}}(\|\boldsymbol{X}-\boldsymbol{T}\boldsymbol{P}^{\boldsymbol{T}}\|_F^2) \qquad (1)$$

subject to

$$||\boldsymbol{p_i}||_1^1 \le c \text{ and } ||\boldsymbol{p_i}||_2^2 = 1 \ (1 \le i \le A) \qquad (2)$$

where $||.||_F^2$ is the squared Frobenius norm of the matrix (sum of squared elements), $\boldsymbol{T}$ is the scores matrix, $\boldsymbol{p_i}$ are the columns of the normalized loadings matrix $\boldsymbol{P}$ for the i^{th} component, and $\boldsymbol{c}$ is the L_1 norm constraint on each loading vector. Therefore, the L_1 norm constraint ($||\boldsymbol{p_i}||_1^1 \le c$) applied on each normalized ($||\boldsymbol{p_i}||_2^2 = 1$) loading vector gives a sPCA model with sparse loadings.

The value of the scalar *c* (from here onwards referred to as sparsity constraint) controls the sparsity level of the model: the lower is *c* the higher is the sparsity induced on the loadings. In particular, the sparsity constraint may range between 1 (which corresponds to only one active variable for each

component) and the square root of the number of variables; if *c* is equal to the square root of the number of variables, which is the maximum allowed value, the loading values converge to those obtained with PCA. However, in practice convergence to non-sparse models can be reached with sparsity constraint values lower than the square root of the number of variables, depending on the nature of the analyzed dataset. The ASLS algorithm calculates simultaneously all the components by iterating between scores and loadings until convergence in a way that the L_1 norm constraint is fulfilled for each component.

Unlike PCA, in sPCA the loading vectors are not orthogonal and the sparse principal component (sPC) directions change according to the number of sPCs used to calculate the model. Since all the sPCs are estimated simultaneously, sPCA may give different loading vectors with different non-zero variables according to the number of components used to build the model, even for the same value of the sparsity constraint (*see* section 4.1).Therefore, it is evident that in sPCA the choice of the proper combination of the tuning parameters, i.e., number of sPCs and sparsity constraint, is a crucial point in model construction in order to have an efficient and robust model.

Since sPCA is an unsupervised technique, it is necessary to couple it with a classifier (e.g. kNN) in order to obtain an estimate of the classification performance which can be used to tune the model parameters. In general, the choice of the optimal sparse parameters should be addressed to the best compromise between sparsity (i.e. as less variables as possible) and model performance (i.e. high efficiency), in order to keep the lowest possible the number of useful spectral variables that lead to stable models with satisfactory classification results.

2.2.2 sPLS-DA

The algorithm used to perform Sparse Partial Least Squares Discriminant Analysis (sPLS-DA) [30] is an extension of Sparse Partial Least Squares regression (sPLS) [29] applied to classification problems. Similarly to PLS-DA, sPLS-DA is based on the use of PLS regression for discriminant purposes, but a Lasso penalty is added to the model parameters in order to constrain some coefficients to be equal to zero.

In particular, the sPLS algorithm used in this work is based on the PLS-SVD approach [47]. Given a descriptor matrix $\boldsymbol{X}$ with size $\{n, m\}$ and a response matrix $\boldsymbol{Y}$ with size $\{n, q\}$, the PLS-SVD approach is based on SVD decomposition of the cross product $\boldsymbol{M} = \boldsymbol{X^T Y}$, as follows:

$$\boldsymbol{M} = \boldsymbol{U \Delta V^T} \qquad (3)$$

where the column vectors of $\boldsymbol{U}$ and $\boldsymbol{V}$ correspond to the PLS loadings vectors of $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively. In the same manner as PLS-DA, $\boldsymbol{Y}$ is a dummy matrix containing the binary class

vectors. The cross product $\boldsymbol{M}$ is calculated as expressed in Equation 3 only for the first latent variable (h=1); for the subsequent components (h = 2, … ,H) the cross product is calculated on the previously deflated $\boldsymbol{X_{h-1}}$ and $\boldsymbol{Y_{h-1}}$ matrices. Indeed the algorithm, in an iterative way, minimizes the squared residuals between the current cross product and the estimated loading vectors and, moreover, adds the Lasso penalty to the **X** loading vector $\boldsymbol{u_h}$; subsequently $\boldsymbol{X_h}$ and $\boldsymbol{Y_h}$ are calculated by a deflation step from matrices $\boldsymbol{X_{h-1}}$ and $\boldsymbol{Y_{h-1}}$. Therefore, the first couple of singular vectors $\boldsymbol{u_h}$ and $\boldsymbol{v_h}$ (where either $\|\boldsymbol{u_h}\|_2^2 = 1$ or $\|\boldsymbol{v_h}\|_2^2 = 1$) are the initial estimate of the iterative algorithm which solves the following optimization problem:

$$\arg\min_{\boldsymbol{u_h},\boldsymbol{v_h}} \left(\left\| \boldsymbol{M_h} - \boldsymbol{u_h}\boldsymbol{v_h^T} \right\|_F^2 + \lambda \|\boldsymbol{u_h}\|_1^1 \right) \qquad (4)$$

where $||.||_F^2$ is the squared Frobenius norm of the matrix (sum of squared elements), and λ is a penalty parameter which applies the Lasso componentwise on the loading vectors. Sparsity is induced on the PLS loadings, and consequently on the regression coefficients used to predict unknown samples, therefore thanks to the sPLS-DA approach it is possible to perform both classification and variable selection in one step, by forcing to zero the coefficients of noisy or uninformative variables.

Conversely to sPCA, in sPLS-DA the sparse latent variables are orthogonal to each other, and their directions do not depend upon the number of components used to calculate the model, due to the deflation step performed before calculating each component.

As in sPCA, there are two parameters to tune in sPLS-DA: the number of sparse latent variables (sLVs) and the penalty term λ. Since the sparsity induced on the model is related to the value of λ, for practical reasons the algorithm has been implemented by the authors in a way to define directly the number of variables to select for each sLV, rather than λ [30].

3. MATERIALS AND METHODS

3.1 Coffee samples

Samples of green coffee beans of Arabica and Robusta species were provided by a local coffee roasting company. Thirty three green coffee batches were considered in this study, coming from different geographical areas and subjected to different processing methods to separate the seed form the fruit. Despite the different sources of variability in the samples, we focused on the discrimination between Arabica and Robusta coffee species, regardless of processing method or geographical origin.

On the whole, 33 samples were collected in the industrial plant during a period of 6 months: 18 samples of Robusta and 15 samples of Arabica. Each sample consisted of about 500 g of beans that

were sampled in order to be as representative as possible of the corresponding batch, and were stored in a sealed package until the day of analysis. From each sample, three aliquots of 70 g of randomly selected beans were kept, and two images were acquired, changing the arrangement of the beans between the two images. This procedure was repeated in a different day to check the day-to-day variability. All the samples were acquired in random order and the packages were sealed again and stored at room temperature between the different acquisition days. Therefore, for each sample 12 hyperspectral images (= 2 measurement sessions × 3 aliquots × 2 repeated acquisitions) were obtained, leading to a dataset composed by 396 hyperspectral images (33 samples × 12 images).

3.2 Image acquisition

The hyperspectral images were acquired using a desktop NIR Spectral Scanner (DV Optic), using a reflectance imaging based spectrometer Specim N17E, coupled to a Xenics XEVA 2608 camera (320 × 256 pixels) and working in the 955-1700 nm spectral range with a spectral resolution of 5 nm. All the images were acquired using as background a black silicon carbide sandpaper sheet, which is characterized by a very low and constant reflectance spectrum [47]. Moreover, a 99% reflectance standard and two ceramic tiles with two different grayscale tones and intermediate reflectance values were included in the images.

The raw data were converted into reflectance values using an instrumental calibration based on the high reflectance standard reference and on dark current [48]. Furthermore, in order to reduce the variability among images over time, an additional internal calibration was performed [49], based on the average reflectance values of the reflectance standard, of the two ceramic tiles and of the black silicon carbide sandpaper.

Then, before further analysis, from each image the pixels related to the black sandpaper background were removed using a thresholding procedure. To this aim, based on the preliminary evaluation of some sample images, the most discriminant wavelength was identified by maximizing the Fisher ratio between background spectra and sample spectra. In this manner, at 1050 nm, all the pixels below the threshold value of 0.1 reflectance units were identified as background and removed.

3.3 Data analysis

After background removal, from each image the average spectra were calculated obtaining a dataset consisting of 396 spectra (33 samples × 12 images for each sample) and 150 variables. The acquired samples were then split into 24 training samples (288 spectra) including 11 Arabica and 13 Robusta coffee samples, and 9 test samples (108 spectra) including 4 Arabica and 5 Robusta coffee samples.

The considered classification methods require to tune the model parameters such as number of components (PCs or LVs) and , in addition, the degree of sparsity for sPCA+kNN and sPLS-DA.

In the present work, different sPCA and sPLS-DA models were constructed on a training set considering different combinations between sparsity (referred as sparsity constraint for sPCA+kNN and number of selected variables on each component for sPLS-DA) and number of components.

For PCA+kNN, and in particular for sPCA+kNN cross validation could be computationally too intensive to tune the model parameters. In fact, for each tested model and for each deletion group, it is necessary to recalculate the distances between objects in the PCs space. Therefore, to overcome this problem, the optimal dimensionality of the PCA model was defined using a monitoring set, i.e. a fixed subset of objects left out from the training set. In this manner the training set was further split into a smaller training set of 17 samples (8 Arabica and 9 Robusta, 204 spectra) and in a monitoring set of 7 samples (3 Arabica and 4 Robusta, 84 spectra). The selection of the monitoring set was done by randomly selecting a given number of coffee samples of both classes, and taking care to include the average spectra of all the replicated and repeated images of each selected coffee sample.

Conversely, in the case of PLS-DA and sPLS-DA, the proper number of latent variables and, for sPLS-DA, the number of variables to select on each sLV, was optimized considering the efficiency of the corresponding classification model calculated on the initial training set. In particular, contiguous blocks cross-validation was performed using 4 deletion groups, each one containing the average spectra of all the replicated and repeated images of 6 samples.

Moreover, in order to visually evaluate the classification performances at the pixel level for both non-sparse and sparse models, one image of Arabica coffee and one of Robusta coffee taken from the test samples were merged together in order to create a test image. In this manner, since the test image is made of two images, one for each class, it was possible to know the class belonging of the single coffee beans and thus to obtain a quantitative evaluation of the predictive ability of the models.

Before calculating the classification models, the spectra were preprocessed using Standard Normal Variate followed by first derivative and mean center.

The classification performances were defined using efficiency (*EFF*), which is the geometric mean between sensitivity (*SENS*) and specificity (*SPEC*), i.e.:

$$EFF = \sqrt{SENS \times SPEC} \qquad (5)$$

where sensitivity is the percentage of objects of each class accepted by the class model and specificity is the percentage of objects of the other classes correctly rejected by the class model [43].

Data analysis was performed using PLS_Toolbox (v. 7.5, Eigenvector Research Inc., USA) for PCA+kNN and PLS-DA; while sPCA+kNN and sPLS-DA were computed with *ad-hoc* routines written in Matlab language (ver. 7.12, The Mathworks Inc., USA) (For sPCA+KNN routine, visit http://models.life.ku.dk/sparsity. sPLS-DA routine was kindly provided by Dr. Ewa Szymanska. Further details are provided in reference [50]). The data were analysed using a personal computer running with Windows 8.1-64 bit and equipped with an Intel Core® i7-3632QM CPU @ 2.20GHz processor and 6.00 GB RAM.

4. RESULTS AND DISCUSSION

In order to compare both the sparse methods with the corresponding classical methods and all the four classification methods altogether, the following sections are organized as follows: sections 4.1 and 4.2 report the comparison of PCA+kNN with sPCA+kNN and of PLS-DA with sPLS-DA, respectively. Each one of these two sections first reports the discussion about the choice of the proper model parameters and the results obtained with the selected models, evaluated at the image-level using the average image spectra; the sparse and not-sparse models are then compared each other based on the relevant spectral features (loadings for PCA+kNN and sPCA+kNN in section 4.1 and loadings for PLS-DA and sPLS-DA in section 4.2), and finally the models are compared at the pixel-level by means of the test image. In the last section (4.3), sparse and not-sparse PCA-based models are compared with the corresponding PLS-DA-based ones, both at the image-level (results on the average image spectra) and at the pixel-level (prediction of the test image); sPCA+kNN and sPLS-DA are then compared each other in terms of selected spectral regions. Finally, the different classification methods are also evaluated in terms of computation time.

4.1. PCA+kNN and sPCA+kNN

Different sPCA models were calculated in order to evaluate the influence of both the sparsity constraint values and of the number of principal components on the number of selected variables and on the kNN classification efficiency (evaluated on the monitoring set samples). In particular, all the combinations from 2 to 5 sPCs with sparsity constraint values ranging from 1 to 12 with step equal to 0.25 were tested, for a total of 180 models. The maximum value considered for the sparsity constraint, c, was set equal to 12, since this value is approximately equal to the square root of the

number of spectral variables of the dataset (150). For all the evaluated models the number of *k* nearest neighbors in kNN has been set equal to 5 after some preliminary tests.

Figure 1.a reports the evolution of the percentage of non-zero variables in the loading vectors when increasing the sparsity constraint from 1 (high sparsity) to 10 (i.e., to the minimum sparsity constraint value at which results converged to those of the corresponding non-sparse models), and employing a different number of sPCs (from top to bottom). For example, the trend of the percentage of variables selected for sPC1 as a function of the sparsity constraint (blue dashed lines) shows significant variations with changing the number of sPCs from 2 to 5 (from top to bottom). The optimal condition should be a compromise between high model stability and low number of non-zero variables. Figure 1.b reports the evolution of the classification efficiency values calculated on the calibration and on the monitoring set as a function of the sparsity constraint *c*, and employing a different number of sPCs. For the lowest values of the sparsity constraint (corresponding to extremely high sparsity induced on the loadings), the efficiency values calculated on the monitoring set (green dash-dot line) show a great variability with small changes of *c*, which means that the model is very unstable. Moreover, the higher the number of sPCs included in the model (from the top plot to the bottom plot in Figure 1.b) the higher is the *c* value which is necessary to reach stable conditions. For example, when using 2 sPCs a stable situation is reached with a sparsity constraint value equal to 3.5, while with 4 sPCs the value of *c* must be at least equal to 5.5.

Therefore, comparing the classification efficiency trends reported in Figure 1.b, the best sPCA+kNN model was chosen as the one calculated using 2 sPCs and a sparsity constraint equal to 3.5, that corresponds to 21% of the variables set to zero.

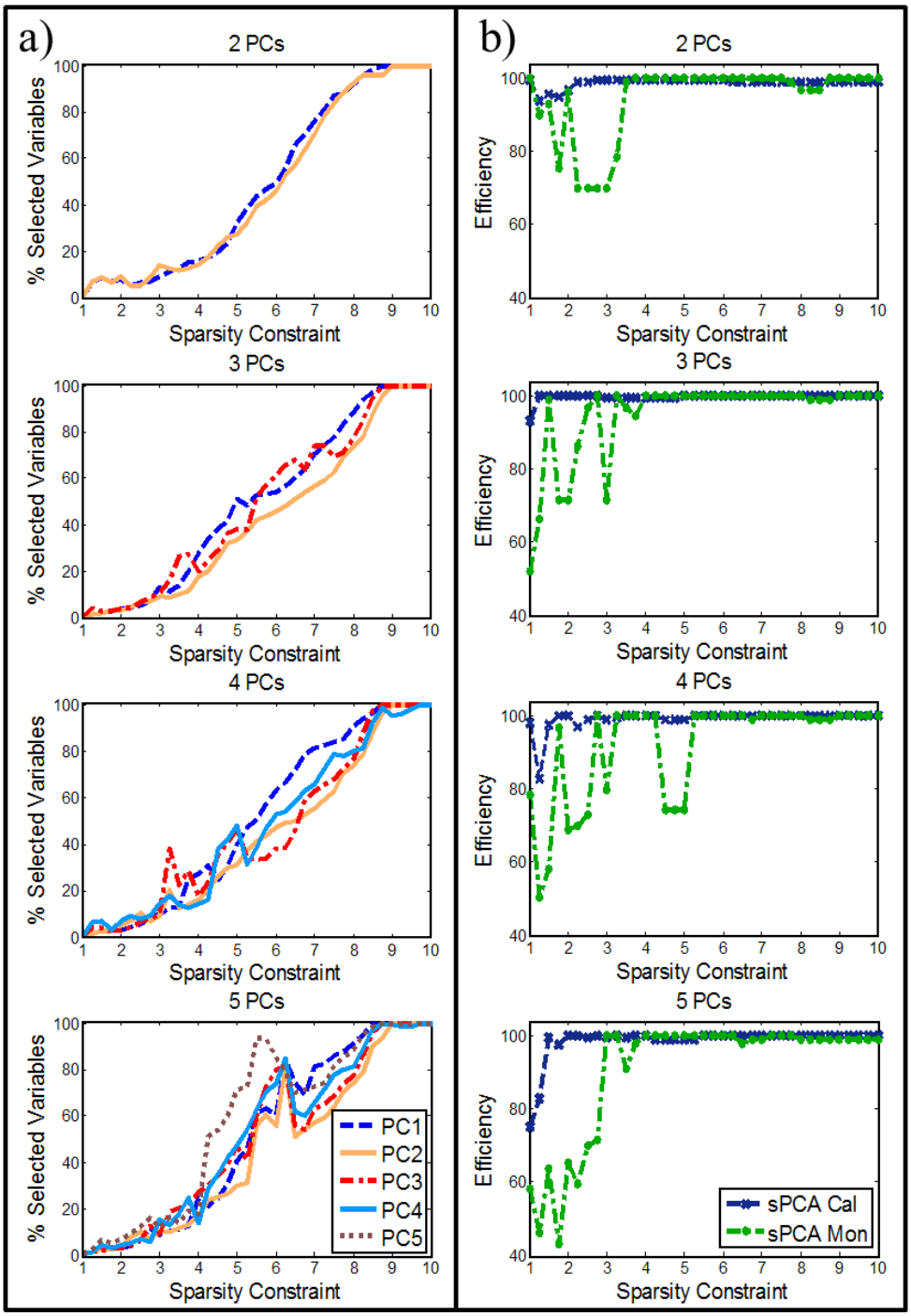

Figure 1. Variation in the sPCA+kNN models of (a) percentage of selected variables and of (b) classification efficiency for the calibration and the monitoring sets, as a function of different values of the sparsity constraint and of the number of principal components.

This model was then used to predict the samples of the external test set, obtaining a classification efficiency value equal to 100%.

The comparison between the performance of PCA+kNN and sPCA+kNN is reported in Table 1. The results obtained for the calibration and for the monitoring sets are comparable for both

methods, but sPCA led to a much higher classification efficiency for the test set, which confirms the importance of forcing to zero the coefficients related to uninformative variables.

		PCA + kNN	sPCA + kNN	PLS-DA	sPLS-DA
N° PCs / LVs		2	2	3	2
N° variables		150	32	150	20
Efficiency Calibration		100.0	99.5	100.0	99.4
Efficiency Monitoring set / CV		98.9	98.9	97.8	99.3
Efficiency Prediction	**Test set**	91.3	100.0	100.0	100.0
	Test image	90.6	86.9	85.0	80.2

Table 1. Classification results from non-sparse (PCA+kNN and PLS-DA) and sparse (sPCA+kNN and sPLA-DA models.

PCA and sPCA models have the same dimensionality in terms of number of PCs, but the number of spectral variables selected with sPCA is definitely lower (21% variables selected in sPCA). As it is shown in Figure 2, there is a substantial convergence between the most relevant bands of the PCA loadings and the non-zero variables selected in sPCA. Indeed, the variables corresponding to the larger values of the PCA loading coefficients are also selected as active variables on the sparse loading vectors. Moreover, variables with a great influence on both PC1 and PC2 in the PCA model are generally selected on only one sparse principal component. In particular, the spectral regions selected by sPCA are related to the C-H aromatic second overtone (1143 nm) and combination band (1446 nm), to the O-H first overtone of aliphatic (1410 nm) and aromatic alcohol (1420 nm), and to the C-H aliphatic second overtone (1195-1225 nm) [52].

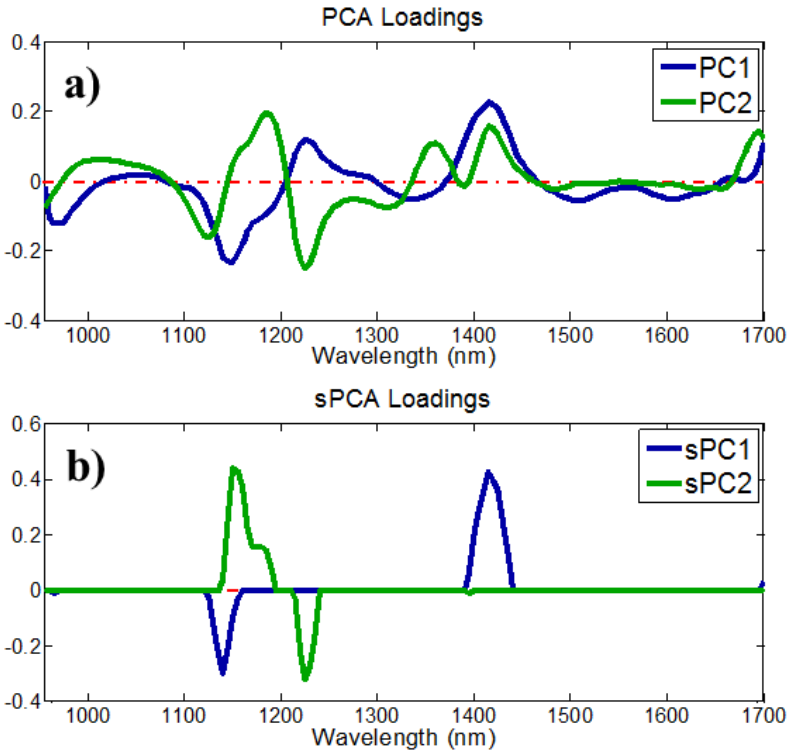

Figure 2. Loading vectors of (a) PCA and (b) sPCA models

Furthermore, the performance of the selected sPCA+kNN model was also evaluated at pixel level using the test image, and the results were compared with those obtained with PCA+kNN (last row of Table 1). The prediction results are also reported under the form of images in Figure 3.a for PCA+kNN and in Figure 3.b for sPCA+kNN, where the pixels predicted as belonging to Arabica coffee are represented in red color, while those predicted as Robusta coffee are represented in green color.

Figure 3 shows that, by a qualitative point of view, the results obtained with PCA+kNN and with sPCA+kNN are analogous, since the overall classification of the beans is correct in both cases. The different efficiency values reported in Table 1 are due to some pixel misclassifications ascribable to the round shape of the beans and the presence of the center-cut, whose effects are slightly more evident in sPCA+kNN than in PCA+kNN. Therefore the sparse model are more sensitive to the noise caused by the morphology of the beans and this fact is shown in the difference image in Figure 3.c, where the pixels correctly predicted with both methods are represented in blue color, the pixels misclassified in both methods are represented in purple color and those predicted in different classes are represented in yellow color. In particular, the percentage of pixels correctly predicted in both methods is equal to 84%, the percentage of pixels misclassified in both methods is equal to 7%

and the percentage of pixels differently predicted is 9%; 6% of which is correctly predicted only by PCA+kNN while 3% is correctly predicted only by sPCA+kNN.

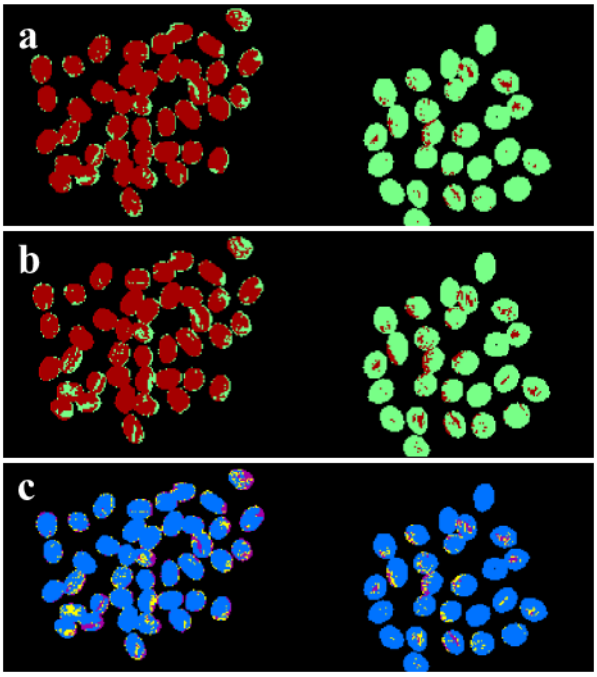

Figure 3. Prediction on the test image of the best (a) PCA+kNN and (b) sPCA+kNN models and (c) difference image between the two images od PCA+kNN and sPCA+kNN models. Each image reports on the left the group of Arabica coffee beans, and on the right the group of Robusta coffee beans. In (a) and (b) the pixels predicted as belonging to Arabica coffee are reported in red color, while those predicted as belonging to Robusta coffee are represented in green color; in (c) the pixels correctly predicted in both methods are represented in blue color, the pixels misclassified in both methods are represented in purple color while those predicted in different classes are represented in yellow color.

4.2. PLS-DA and sPLS-DA

Cross validation was used for both PLS-DA and sPLS-DA in order to define the proper number of latent variables of the models and, for sPLS-DA, also to identify the optimal number of non-zero variables for each sLV. For this purpose, different sPLS-DA models were built considering a number of sLVs ranging from 1 to 7 and a number of selected variables for each sLV ranging from 5 up to 150 (same number of variables for each sLV); the performance of the models calculated considering each combination of the two parameter values was evaluated in terms of efficiency in cross validation.

The two model parameters were optimized considering the best compromise between high model performance (high cross validation efficiency) and low model dimensionality, this latter being estimated both in terms of number of sLVs and of number of active variables for each sLV.

Figure 4 shows the surface response of the cross validation efficiency values as a function of the number of sLVs and of the number of variables selected for each sLV. In particular, for a number of selected variables equal to 150, the reported results are exactly coincident with those of the (non-

sparse) PLS-DA model, where the optimal dimensionality is equal to 3 LVs, corresponding to a cross validation efficiency value equal to 97.82%. Conversely, in high sparsity conditions, i.e. when the number of selected variables is small (e.g. from 5 to 35), the optimal number of latent variables is equal to 2. Moreover, from Figure 4 it is also possible to observe that, when the number of sLVs is high, the models tend to stabilize, giving cross-validation values close to 100%. However, considering the optimal compromise between a parsimonious model and high efficiency values, the best sPLS-DA model was chosen in correspondence to 2 sLVs and with the number of variables to select for each sLV set to 10, which corresponds to a local maximum in the surface response. In this situation the CV efficiency is equal to 99.3% and the corresponding sparse loading vectors have a total of 20 active variables out of 150.

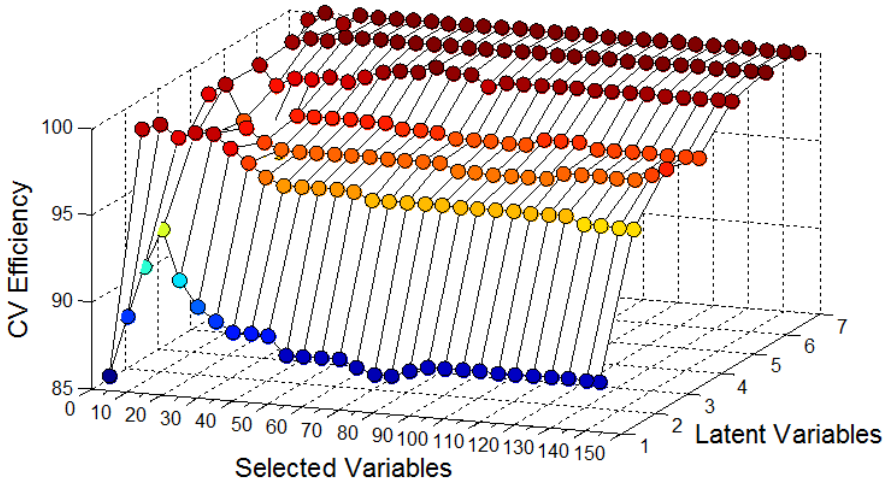

Figure 4. Surface response of cross validation efficiency as a function of the number of LVs and of the number of variables selected for each LV.

Table 1 reports the comparison between PLS-DA and sPLS-DA model performances in calibration, cross validation and prediction of the test set samples. In particular, the efficiency values calculated in calibration and prediction of the test set are almost the same, while sPLS-DA shows slightly better performances in cross validation. Moreover, sPLS-DA retains only 20 non-zero variables out of 150.

The loading vectors of the selected PLS-DA and sPLS-DA models are reported in Figure 5.a and 5.b, respectively. The spectral regions related to the aromatic (around 1143 nm) and aliphatic (1195-1225 nm) C-H second overtone have relevant influence both on the PLS-DA model and on the sPLS-DA one, and with a similar pattern. Moreover, in the sparse loading vectors also the region between 1400 nm and 1430 nm has been selected, which is related to the O-H first overtone and to the C-H combination bands. Interestingly, the sPLS-DA model does not select the extreme

spectral regions, where high absolute values were observed instead for the PLS-DA loadings vectors on LV 2 and LV 3 (and partially also for the PCA loading vectors in Figure 2.a), which could be ascribable to border distortions due to the first derivative preprocessing.

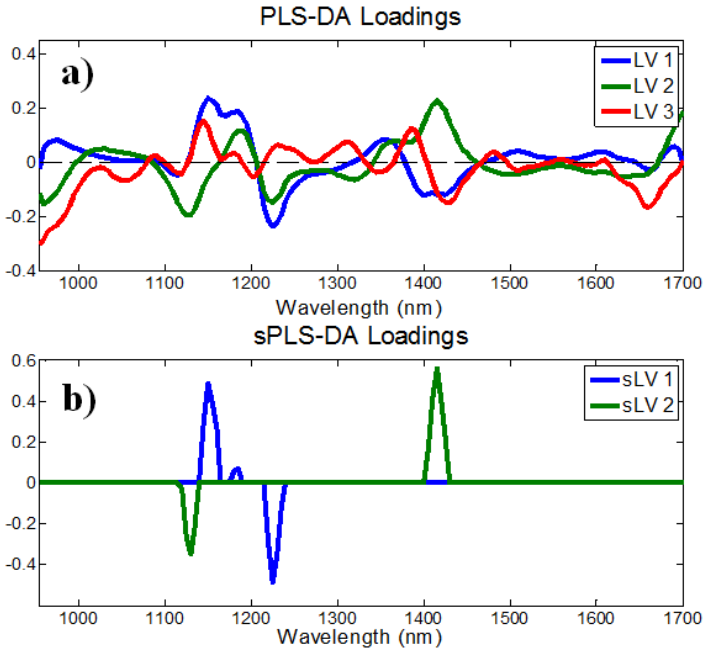

Figure 5. Loading vectors of (a) PLS-DA and (b) sPLS-DA models.

Also in this case a pixel-level classification of the test image was obtained for both PLS-DA and sPLS-DA models, and the results are reported in Figure 6.a and Figure 6.b as a false-color image, where the pixels predicted as belonging to Arabica coffee are represented in red color and the pixels predicted as belonging to Robusta coffee in green color. Comparing the right parts of Figure 6.a and 6.b (Robusta coffee beans), it is possible to notice that for both PLS-DA and sPLS-DA the same Robusta coffee bean is misclassified; moreover, in sPLS-DA there is one more Robusta bean whose classification is uncertain. Analogous considerations can be drawn for the left parts of the two images, where ambiguous results have been obtained for two coffee beans. In general, the performance of the sPLS-DA model is slightly lower than the performance of the PLS-DA model; this is also confirmed by the efficiency values reported in Table 1 and by the comparison reported in Figure 6.c, where the pixels correctly predicted in in both methods are represented in blue color, the pixel misclassified in both methods are represented in purple color while, those predicted in different classes are represented in yellow color. In particular, the percentage of pixels correctly predicted with both methods is equal to 76%, the percentage of pixels misclassified in both methods is equal to10% and the percentage of pixels differently predicted is equal to 14%, 9% of which is correctly predicted only by PLS-DA and 5% is correctly predicted only by sPLS-DA. Similarly to

the case of sPCA+kNN, Figure 6.c shows that the sparse model is more sensitive to the round shape and to the presence of the center-cut of the beans.

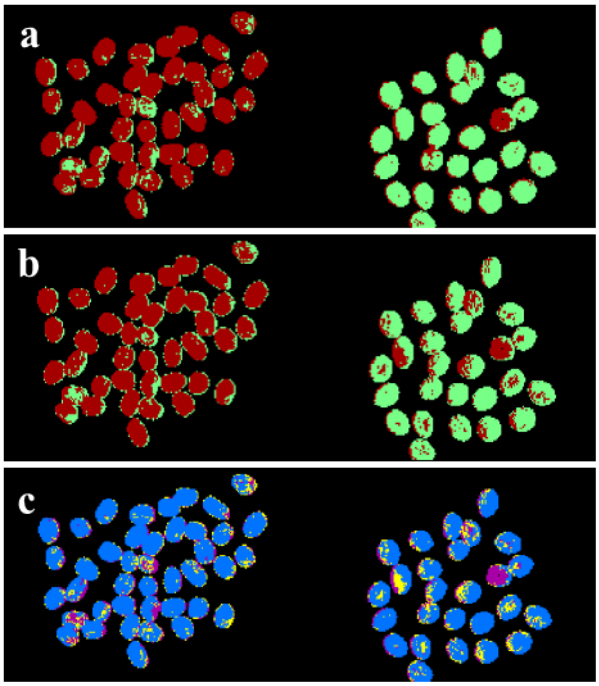

Figure 6. Prediction on the test image of the best (a) PLS-DA and (b) sPLS-DA models and (c) difference image between the two images of PLS-DA and sPLS-DA models. Each image reports on the left the group of Arabica coffee beans, and on the right the group of Robusta coffee beans. In (a) and (b) the pixels predicted as belonging to Arabica coffee are reported in red color, while those predicted as belonging to Robusta coffee are represented in green color; in (c) the pixels correctly predicted in both methods are represented in blue color, the pixels misclassified in both methods are represented in purple color while those predicted in different classes are represented in yellow color.

4.3 Comparison between methods

The comparison between the results reported in Table 1 shows that the classification performances calculated in calibration and on the monitoring set or on cross validation are almost the same for PCA+kNN and PLS-DA, both considering non-sparse and sparse models. Concerning the prediction of the test set samples, there is an improvement of efficiency from PCA+kNN to sPCA+kNN, while for both PLS-DA and sPLS-DA the efficiency is equal to 100%. As far as the pixel-level prediction of the test set image is concerned, PCA+kNN shows better performances than PLS-DA, and in both cases the sparse model is slightly worse than the full model. However the bean-wise classification is correct both on PCA+kNN and sPCA+kNN, while some misclassifications occur in PLS-DA and sPLS-DA. The best performances of PCA+kNN on the test image could be explained by the fact that kNN is a distance-based classifier able to handle non-linear boundaries between classes, while PLS-DA establishes a linear threshold. Since images are made of thousands of pixel spectra, these have a much greater variability than the average image

spectra, and some overlapping between classes of pixels may occur. In these conditions, kNN proves to be a more robust method than PLS-DA.

Regarding variable selection performed by sparse methods, sPLS-DA led to a slightly more parsimonious model including 20 active variables, with respect to the 32 non-zero variables selected by sPCA+kNN. However, comparing the sparse loadings from the sPCA+kNN model (Figure 2.b) with those of the sPLS-DA model (Figure 5.b), it is evident that there is a substantial convergence of the spectral regions selected by the two different approaches, which confirms the reliability of these sparse methods in highlighting the chemical differences between the two considered classes.

Finally, a comparison between the different classification methods was also done in terms of time needed for model calculation and for prediction of the test image. For sparse methods a longer computational time is necessary to tune the proper sparsity parameters (i.e., sparsity constraint for sPCA+kNN and number of variables for each LV for sPLS-DA), since all the combinations between number of components and sparsity parameter must be evaluated. For example, in the case of sPCA+kNN, 45 different sparsity constraint values and 4 different number of PCs were tested; on the whole 180 (=45×4) models were thus calculated and used to predict the monitoring set samples, which required 86.6 s. However, once the model parameters are tuned, the time necessary to calculate the model and use it for prediction are the same for both sparse and non-sparse methods. Moreover it is important to highlight that for sparse methods the model construction involves also variable selection by setting to zero the uninformative variable coefficients, and this process requires the same computational time as classical (non-sparse) methods. Considering again the example of sPCA+kNN, the average time necessary to calculate a single model was 0.4 s, and the time necessary to predict the test image was 13.5 s; these values are essentially the same as those for PCA+kNN. In general, kNN classifier is more computationally intensive than PLS-DA, since it requires the calculation of the distances between each test object and each training object, and their comparison. This is particularly evident in the case of image predictions at the pixel level, since an image is made of thousands to millions of pixels. In fact for the test image (about 12400 segmented pixels) the time required for PCA+kNN and sPCA+kNN was about 13.5 s, while for PLS-DA and sPLS-DA it was less than 0.1 s.

5. CONCLUSIONS

In the present work we explored the possibility to use sparse methods, such as sPCA+kNN or sPLS-DA, both in order to classify hyperspectral images of Arabica and Robusta green coffee beans, and to select spectral regions relevant for the discrimination between the two classes.

Compared to classical methods, the corresponding sparse methods led to the analogous or even better classification results, evaluated both at the image level and at the pixel-level.

However, sparse methods allowed performing variable selection at the same time as classification, giving much more parsimonious models and enhancing the interpretability in chemical terms of the results, within a reasonable computational time. In particular, the feature selection made with two different sparse classification approaches converged to the same spectral regions, which confirms the chemical relevance of the selected wavelengths.

Furthermore, the high classification efficiency values obtained with sparse methods highlighted the possibility to use the narrow selected spectral regions for the implementation of multispectral systems, to be used for on-line process control applications.

ACKNOWLEDGMENTS

The authors wish to thank Luigi Bellucci (Caffè Molinari spa) for providing the coffee samples and technical support.

References

1. A.A. Gowen, C. O'Donnell, P.J. Cullen, G. Downey, J.M. Frias, Hyperspectral imaging–an emerging process analytical tool for food quality and safety control, *Trends in Food Science & Technology*, 18 (12) (2007) 590-598.
2. J.M. Amigo, J. Cruz, M. Bautista, S. Maspoch, J. Coello, M. Blanco, Study of pharmaceutical samples by NIR chemical-image and multivariate analysis, *TrAC Trends in Analytical Chemistry*, 27 (8) (2008) 696–713.
3. J.M. Amigo, Practical issues of hyperspectral imaging analysis of solid dosage forms, *Analytical and Bioanalytical Chemistry*, 398 (1) (2010) 93–109.
4. J. Burger, A. Gowen, Data handling in hyperspectral image analysis, *Chemometrics and Intelligent Laboratory Systems*, 108 (1) (2011) 13-22.
5. M. Vidal, J.M. Amigo, Pre-processing of hyperspectral images. Essential steps before image analysis, *Chemometrics and Intelligent Laboratory Systems*, 117 (2012) 138-148.
6. C. Ferrari, G. Foca, A. Ulrici, Handling large datasets of hyperspectral images: reducing data size without loss of useful information, *Anaytica. Chimica Acta*, 802 (2013) 29-39.
7. Z. Xiaobo, Z. Jiewen, M.J.W. Povey, M. Holmes, M. Hanpin, Variables selection methods in near-infrared spectriscopy, *Analytica Chimica Acta*, 667 (1) (2010) 14-32.
8. L. Norgaard, A. Saudland, J. Wagner, J. P. Nielsen, L. Munck, S. B. Engelsen, Interval Partial Least-Squares Regression (iPLS): A Comparative Chemometric Study with an Example from Near-Infrared Spectroscopy, *Applied Spectroscopy*, 54 (3) (2000) 413-419.
9. R. Leardi, L. Norgaard, Sequential application of backward interval partial least squares and genetic algorithms for the selection of relevant spectral regions , *Journal of Chemometrics*, 18 (11) (2004) 486–497.
10. G. Foca, D. Salvo, A. Cino, C. Ferrari, D.P. Lo Fiego, G. Minelli, A. Ulrici, Classification of pig fat samples from different subcutaneous layers by means of fast and nondestructive analytical techniques, *Food Res. Int*., 52 (1) (2013) 185-197.
11. R. Leardi, Application of genetic algorithm–PLS for feature selection in spectral data sets, *Journal of Chemometrics*, 14 (5-6) (2000) 643-655.
12. D. Jouan-Rimbaud, B. Walczak, R. J. Poppi,, O. E. de Noord, D. L. Massart, Application of Wavelet Transform To Extract the Relevant Component from Spectral Data for Multivariate Calibration, *Anal. Chem.*, 69 (21) (1997) 4317–4323.
13. M. Cocchi, R. Seeber, A. Ulrici, WPTER: wavelet packet transform for efficient pattern recognition of signals, *Chemometrics and Intelligent Laboratory Systems*, 57 (2) (2001) 97-119.
14. J.H. Friedman, Fast sparse regression and classification, *International Journal of Forecasting*, 28(3) (2012) 722-738.
15. K.A. Lê Cao, P.G. Martin, C. Robert-Granié, P. Besse, Sparse canonical methods for biological data integration: application to a cross-platform study, *BMC bioinformatics*, 10(1) (2009) 34
16. G.I. Allen, M. Maletić-Savatić, Sparse non-negative generalized PCA with applications to metabolomics, *Bioinformatics*, 27(21) (2011), 3029-3035.
17. C. Colombani, P. Croiseau, S. Fritz, F. Guillaume, A. Legarra, V. Ducrocq, C. Robert-Granié, A comparison of partial least squares (PLS) and sparse PLS regressions in genomic selection in French dairy cattle. *Journal of dairy science*, 95(4) (2012), 2120-2131.
18. I. Karaman, E.M. Qannari, H. Martens, M.S. Hedemann, K.E.B. Knudsen, A. Kohler, Comparison of Sparse and Jack-knife partial least squares regression methods for variable selection, *Chemometrics and Intelligent Laboratory Systems*, 122 (2013) 65-77.
19. R. Wang, W. Zeng, J. Ming, Discrimination of the White Wine Based on Sparse Principal Component Analysis and Support Vector Machine. In *Computer Engineering and Networking*, (2014) 695-702

20. M.R. Kunz, J. Ottaway, J.H. Kalivas, C.A. Georgiou, G.A. Mousdis, Updating a synchronous fluorescence spectroscopic virgin olive oil adulteration calibration to a new geographical region, *Journal of agricultural and food chemistry*, 59(4) (2011) 1051-1057
21. P. Filzmoser, M. Gschwandtner, V. Todorov, Review of sparse methods in regression and classification with application to chemometrics, *Journal of Chemometrics*, 26 (2012), 42-51.
22. D. M. Witten, R. Tibshirani, T. Hastie, A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis, *Bio- statistics*, 10(3) (2009) 515–534.
23. M. Lee, H. Shen, J.Z. Huang, J. S. Marron, Biclustering via sparse singular value decomposition, *Biometrics*, 66 (2010) 1087–1095.
24. R. Bro, E.E. Papalexakis, E. Acar, N.D. Sidiropoulos, Coclustering — a useful tool for chemometrics, *Journal of Chemometrics* 26 (6) (2012) 256–263.
25. H. Zou, T. Hastie, R. Tibshirani, Sparse principal component analysis, *Journal of Computational and Graphical Statistics*, 15 (2) (2006) 265–286
26. I.T. Jolliffe, N.T. Trendafilov, M. Uddin, A modified principal component technique based on the LASSO, *Journal of Computational and Graphical Statistics*, 12 (3) (2003) 531–547.
27. H. Shen, J.Z. Huang, Sparse principal component analysis via regularized low rank matrix approximation, *Journal of Multivariate Analysis*, 99(6) (2008) 1015–1034
28. H. Chun, S. Keleş, Sparse partial least squares regression for simultaneous dimension reduction and variable selection, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72 (1) (2010) 3-25.
29. K.A. Lê Cao, D. Rossouw, C. Robert-Granié, P. Besse, A sparse PLS for variable selection when integrating omics data, *Statistical applications in genetics and molecular biology*, 7(1) (2008) 37.
30. K.A. Lê Cao, S. Boitard, P. Besse, Sparse PLS Discriminant Analysis: biologically relevant feature selection and graphical displays for multiclass problems. *BMC bioinformatics*, 12(1) (2011), 253-269.
31. L. Clemmensen, T. Hastie, D. Witten, B. Ersbøll, B, Sparse discriminant analysis, *Technometrics*, 53(4) (2011) 406-413.
32. K. Peerbhay , O. Mutanga, R. Ismail; Does simultaneous variable selection and dimension reduction improve the classification of pinus forest species?. J. Appl. Remote Sens. 0001;8(1):085194. doi:10.1117/1.JRS.8.085194
33. M.A. Rasmussen, R. Bro, A tutorial on the Lasso approach to sparse modeling, *Chemometrics and Intelligent Laboratory Systems*, 119 (2012) 21-31.
34. M. Martín, F. Pablos, A. González, Discrimination between arabica and robusta green coffee varieties according to their chemical composition, *Talanta*, 46 (1998) 1259-1264.
35. J. Santos, M. Sarraguça, A. Rangel, J. Lopes, Evaluation of green coffee beans quality using near infrared spectroscopy: A quantitative approach, *Food Chemistry*, 135 (2012) 1828-1835
36. A. J. Myles, T. A. Zimmerman, S. D. Brown, Transfer of Multivariate Classification Models Between Laboratory and Process Near-Infrared Spectrometers for the Discrimination of Green Arabica and Robusta Coffee Beans, *Appl. Spectrosc.*, 60 (2006) 1198-1203.
37. I. Esteban-Dıez, J.M. González-Sáiz, C. Pizarro, An evaluation of orthogonal signal correction methods for the characterisation of arabica and robusta coffee varieties by NIRS, *Analytica Chimica Acta*, 514 (2004) 57-67.
38. I. Esteban-Diez, J.M. Gonzalez-iaiz, C. Saenz-Gonzalez, C. Pizarro, Coffee varietal differentiation based on near infrared spectroscopy, *Talanta*, 71 (2007) 221-229
39. A.G. Fiore, R. Romaniello, G. Peri, C. Severini, Quality assessment of roasted coffee blends by hyperspectral image analysis, In *Proceedings of 22nd International Conference on Coffee Science, Campinas, Brazil* (2008)

40. A. Backhaus, F. Bollenbeck, U. Seiffert, High-throughput quality control of coffee varieties and blends by artificial neural networks and hyperspectral imaging, In *Proceedings of the 1st International Congress on Cocoa, Coffee and Tea, CoCoTea* (2011).
41. S. Wold, K. Esbensen, P. Geladi, Principal Component Analysis, *Chemometrics and Intelligent Laboratory Systems*, 2 (1987) 37-52.
42. T. Cover, P. Hart, Nearest Neighbor Pattern Classification, *IEEE Transactions on Information Theory*,13 (1967) 21-27.
43. M. Forina, P. Oliveri, H. Jäger, U. Römisch, J. Smeyers-Verbeke, Class modeling techniques in the control of the geographical origin of wines, *Chemometrics and Intelligent Laboratory Systems*, 99 (2009) 127–137.
44. S. Chevallier, D. Bertrand, A. Kohler, P. Courcoux, Application of PLS-DA in multivariate image analysis, *J. Chemometrics*,20 (2006) 221-229.
45. R. Tibshirami, Regression Shrinkage and Selection via the Lasso, *Journal of the Royal Statistical Society*, 58 (1996) 267-288.
46. A. Lorber, L. Wangen, B. Kowalski, A theoretical foundation for the PLS algorithm, *Journal of Chemometrics*,1 (1987) 19-31.
47. J. Burger, P. Geladi, Hyperspectral NIR image regression part II: Dataset preprocessing diagnostics, *Journal of Chemometrics*, 20 (2006) 106-119.
48. J. Burger, P. Geladi, Hyperspectral NIR image regression part I: Calibration and correction, *Journal of Chemometrics*, 19 (2005) 355-363.
49. A. Ulrici, S. Serranti, C. Ferrari, D. Cesare, G. Foca, G. Bonifazi, Efficient chemometric strategies for PET–PLA discrimination in recycling plants using hyperspectral imaging. *Chemometrics and Intelligent Laboratory Systems*, 122 (2013) 31-39.
50. E. Szymanska, E. Brodrick, M. Williams, A.N. Davies, H.J. Van Manen, L.C.M Buydens, Data size reduction strategy for the classification of breath and air samples using multicapillary column-ion mobility spectrometry, Analytical Chemistry 87:2 (2015) 869-875.
51. Handbook of Near Infrared Analysis, Third Edition, *edited by Donald A. Burns and Emil W. Ciurzak.*