

Efficient Kernel Orthonormalized PLS for Remote Sensing Applications

Jerónimo Arenas-García, *Member, IEEE*

Gustavo Camps-Valls, *Senior Member, IEEE*

Abstract

This paper studies the performance and applicability of a novel Kernel Partial Least Squares (KPLS) algorithm for non-linear feature extraction in the context of remote sensing applications. The so-called Kernel Orthonormalized PLS algorithm with reduced complexity (rKOPLS) has two core parts: (i) a kernel version of OPLS (called KOPLS), and (ii) a sparse approximation for large scale data sets, which ultimately leads to the rKOPLS algorithm. The method is theoretically analyzed in terms of computational and memory requirements, and tested in common remote sensing applications: multi- and hyperspectral image classification and biophysical parameter estimation problems. The proposed method largely outperforms the traditional (linear) PLS algorithm, and demonstrates good capabilities in terms of expressive power of the extracted non-linear features, accuracy and scalability as compared to the standard KPLS.

Index Terms

Kernel methods, Partial Least Squares, feature extraction, model inversion, image classification.

Manuscript received October 2007;

JAG is with Dept. Signal Theory and Communications. Universidad Carlos III de Madrid. Spain. E-mail: jarenas@tsc.uc3m.es, <http://www.tsc.uc3m.es/~jarenas>

GCV is with Dept. Enginyeria Electrònica. Escola Tècnica Superior d'Enginyeria. Universitat de València. C/ Dr. Moliner, 50. 46100 Burjassot (València) Spain. E-mail: gustavo.camps@uv.es, <http://www.uv.es/~gcamps>

I. INTRODUCTION

Partial Least Squares (PLS) is a family of methods for linear feature extraction. The underlying idea is to exploit not only the variance of the inputs but also their covariance with the target, which is presumably more important. Basically, PLS uses the covariance to guide the selection of features before performing linear regression or classification in the resulting (reduced) feature space. The expressive power of the extracted features using this method has been successfully exploited in applications where a high number of correlated inputs are available. The fields of chemometrics and spectroscopy based on Vis/NIR spectra have been particularly active in the use of PLS methods [1]–[4]. In the case of remote sensing data processing, PLS methods have also been used in particular applications, such as mapping canopy nitrogen [5], [6], classifying salt marsh plants [7], analyzing biophysical properties of forests [8], and retrieving leaf fuel moisture [9]. This choice is rooted on the fact that PLS methods are well-suited to deal with ill-posed and multicollinearity problems, such as those encountered when analyzing remote sensing data acquired with hyperspectral sensors.

Among the many variants of PLS, the one that has become particularly very popular is the one presented in [10], to which we will refer as PLS2. The algorithm relies on the following two assumptions: First, the latent variables of the input features are good predictors of the response variable and, second, there is a linear relation between the independent and dependent variables. Several other variants of PLS exist such as “PLS Mode A” [11], Orthonormalized PLS (OPLS) [12] and PLS-SB [13]; see [14] for a discussion of the early history of PLS, [15] for a more recent and technical description, and [16] for a very well-written contemporary overview. In this paper, however, we will concentrate on the popular PLS2 algorithm and on the OPLS which enjoys certain optimality conditions with respect to basic PLS, as discussed in [17].

Despite the good performance of these PLS-based methods, a main and critical shortcoming is encountered; as they are based on linear projections, suboptimal performance is observed when the variables of the input and output spaces are non-linearly related. This is a common situation in remote sensing data given the non-linear relationship between the acquired spectral information and the dependent variable, i.e. class of the corresponding spectrum or derived biophysical variable. This has originated active research lines on the development of non-linear versions of the PLS methods. Essentially, two major approaches to model nonlinear data relations by means

of PLS exist. The first approach considers reformulating the considered input-output relation by a nonlinear model which is learned from data either with polynomial functions, smoothing splines, artificial neural networks, or radial basis function networks [18]–[21]. However, since the assumption that the score vectors are linear projections of the original variables is kept, the non-linear mapping must be *linearized* by means of Taylor series expansions and also the weight vectors must be updated iteratively. These are undoubtedly critical problems encountered when following this approximation. The second approach to nonlinear PLS is based on a two-fold strategy: first mapping the original data by means of a nonlinear function to a new feature space, and then solving a linear PLS there, which is non-linear from the original input space. This approach is based on the solid theoretical foundations of kernel methods [22], [23], which has produced successful results in many fields of application [24]. Kernel methods have been extensively used for remote sensing data analysis, mainly for classification [25]–[28], and regression [29], [30]. In this paper, we extend its use to non-linear feature extraction. Non-linear versions of PLS have been recently theoretically developed under the kernels framework [23], [31], and interesting applications are found in the field of chemometrics [32], [33].

Recently, a kernel extension of OPLS, the so-called Kernel OPLS with reduced complexity (rKOPLS) algorithm, was proposed for feature extraction [34], and its use was evaluated in the context of remote sensing applications in [35]. In this paper, we further analyze the capabilities of this method both including a deeper theoretical study and analyzing the performance in more remote sensing scenarios, both for classification and regression problems. In particular, we pay special attention to all ingredients of the method, namely sparsity, accuracy, and representation power of the extracted features, compared to both linear and non-linear PLS-based methods.

The rest of the paper is outlined as follows. Section II briefly reviews the idea and standard formulations of basic (linear) multivariate analysis methods, while Section III is devoted to revise the kernel PLS algorithm. Section IV presents the rKOPLS method for non-linear feature extraction, and compares this method with previous approaches in terms of complexity and scalability. Section V presents the experimental results in classification and regression problems. Finally, Section VI concludes with some remarks and further research directions.

II. MUTIVARIATE ANALYSIS METHODS

The family of Multivariate Analysis (MVA) methods comprises several algorithms for feature extraction that exploit correlations between data representation in input and output spaces, so that the extracted features in input space can be used to predict the output variables, and viceversa.

Before presenting some of the linear MVA methods, we introduce the used notation. In this paper, we are interested in learning problems in which we are given a set of training pairs $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^l$, with $\mathbf{x}_i \in \mathbb{R}^N$, $\mathbf{y}_i \in \mathbb{R}^M$, or, using matrix notation, $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_l]^\top$ and $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_l]^\top$, where superscript \top denotes matrix or vector transposition. We denote by $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$ the centered versions of \mathbf{X} and \mathbf{Y} , respectively, while $\mathbf{C}_{xx} = \frac{1}{l} \tilde{\mathbf{X}}^\top \tilde{\mathbf{X}}$ represents the covariance matrix of the input data, and $\mathbf{C}_{xy} = \frac{1}{l} \tilde{\mathbf{X}}^\top \tilde{\mathbf{Y}}$ the covariance between the input and output data.

Feature extraction is usually used previous to the application of machine learning algorithms to discard irrelevant or noisy components, and to reduce the dimensionality of the data, what helps also to prevent numerical problems (e.g., when \mathbf{C}_{xx} is rank deficient). Linear feature extraction can be carried out by projecting the data into the subspaces characterized by projection matrices \mathbf{U} and \mathbf{V} , of sizes $N \times n_p$ and $M \times n_p$, so that the n_p extracted features of the original data are given by $\tilde{\mathbf{X}}' = \tilde{\mathbf{X}}\mathbf{U}$ and $\tilde{\mathbf{Y}}' = \tilde{\mathbf{Y}}\mathbf{V}$.

The general spirit of MVA is to find \mathbf{U} and \mathbf{V} so that the features of the input and output projected data, $\tilde{\mathbf{X}}'$ and $\tilde{\mathbf{Y}}'$, respectively, are maximally aligned. In the rest of the section we will briefly review two MVA algorithms: Partial Least Squares (PLS) and Orthonormalized PLS (OPLS), on which we will focus in this paper.

A. Partial Least Squares

Apart from Principal Component Analysis (PCA) [36] which only pays attention to the input space variance, Partial Least Squares (PLS), developed by Herman Wold in 1966 [10], is probably the simplest of MVA methods, what justifies its very extensive use in many different fields. The underlying assumption of a PLS model is that the system of interest is driven by a few latent variables (also called factors or components), which are *linear* combinations of observed explanatory variables (i.e., spectral channels or bands). The central idea of PLS is to find a few eigenvectors of spectral matrices that will produce score values that both summarize the variance of spectral reflectance well and highly correlate with response variables (i.e. material class or corresponding biophysical parameter).

The goal of PLS is to find the directions of maximum covariance between the projected input and output data:

$$\begin{aligned} \text{PLS:} \quad & \text{maximize: } \text{Tr}\{\mathbf{U}^\top \mathbf{C}_{xy} \mathbf{V}\} \\ & \text{subject to: } \mathbf{U}^\top \mathbf{U} = \mathbf{V}^\top \mathbf{V} = \mathbf{I} \end{aligned} \quad (1)$$

where the maximization is carried out with respect to projection matrices \mathbf{U} and \mathbf{V} , and \mathbf{I} is the identity matrix of size n_p .

Using Lagrange multipliers, it can be shown (see, e.g., [23]) that the solution to (1) is given by the singular value decomposition of \mathbf{C}_{xy} . Practical implementation are usually based on an iterative two step scheme in which the projection vectors \mathbf{u}_i and \mathbf{v}_i associated to the largest singular value are first extracted, and matrices $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$ are then modified to remove whatever information that can be explained through the previously extracted projections, a process which is known as *deflation*. In the literature, many variants of PLS exist [11], [13]–[16], which are essentially based on modifications of any of the two previous steps.

B. Orthonormalized Partial Least Squares

In this paper, we pay attention to the Orthonormalized Partial Least Squares (OPLS) [12], which tackles the following maximization problem:

$$\begin{aligned} \text{OPLS:} \quad & \text{maximize: } \text{Tr}\{\mathbf{U}^\top \mathbf{C}_{xy} \mathbf{C}_{xy}^\top \mathbf{U}\} \\ & \text{subject to: } \mathbf{U}^\top \mathbf{C}_{xx} \mathbf{U} = \mathbf{I} \end{aligned} \quad (2)$$

Note that, unlike other MVA methods, OPLS only extracts projections from the input data.

A nice property of OPLS is that it is optimal (in the mean square error sense) for performing linear multiregression of $\tilde{\mathbf{Y}}$ on the projected input data for a given number of features [17]:

$$\text{OPLS (2):} \quad \text{minimize: } \|\tilde{\mathbf{Y}} - \tilde{\mathbf{X}}\mathbf{U}\mathbf{W}\|_F^2 \quad (3)$$

where the optimal regression matrix is given by $\mathbf{W} = \tilde{\mathbf{X}}^\dagger \tilde{\mathbf{Y}}$, where $\tilde{\mathbf{X}}^\dagger = (\tilde{\mathbf{X}}^\top \tilde{\mathbf{X}})^{-1} \tilde{\mathbf{X}}^\top$ is the Moore-Penrose pseudoinverse of $\tilde{\mathbf{X}}$. To demonstrate the equivalency between (2) and the minimization problem (3), we will first rewrite (3) using the fact that $\|\mathbf{A}\|_F^2 = \text{Tr}\{\mathbf{A}\mathbf{A}^\top\}$:

$$\|\tilde{\mathbf{Y}} - \tilde{\mathbf{X}}\tilde{\mathbf{X}}^\dagger \tilde{\mathbf{Y}}\|_F^2 = \text{Tr}\{\tilde{\mathbf{X}}\tilde{\mathbf{X}}^\dagger \tilde{\mathbf{Y}}(\tilde{\mathbf{X}}\tilde{\mathbf{X}}^\dagger \tilde{\mathbf{Y}})^\top\} + \text{Tr}\{\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^\top\} - 2\text{Tr}\{\tilde{\mathbf{X}}\tilde{\mathbf{X}}^\dagger \tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^\top\} \quad (4)$$

Note now that:

- 1) $\text{Tr}\{\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^\top\}$ is constant and does not affect the minimization

$$2) \text{Tr}\{\tilde{\mathbf{X}}'\tilde{\mathbf{X}}'^{\dagger}\tilde{\mathbf{Y}}(\tilde{\mathbf{X}}'\tilde{\mathbf{X}}'^{\dagger}\tilde{\mathbf{Y}})^{\top}\} = \text{Tr}\{(\tilde{\mathbf{X}}'\tilde{\mathbf{X}}'^{\dagger})^{\top}\tilde{\mathbf{X}}'\tilde{\mathbf{X}}'^{\dagger}\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^{\top}\} = \text{Tr}\{\tilde{\mathbf{X}}'\tilde{\mathbf{X}}'^{\dagger}\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^{\top}\}$$

so that solving (3) is equivalent to maximizing:

$$\text{Tr}\{\tilde{\mathbf{X}}'\tilde{\mathbf{X}}'^{\dagger}\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^{\top}\} = \text{Tr}\{(\mathbf{U}^{\top}\tilde{\mathbf{X}}^{\top}\tilde{\mathbf{X}}\mathbf{U})^{-1}\mathbf{U}^{\top}\tilde{\mathbf{X}}^{\top}\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^{\top}\tilde{\mathbf{X}}\mathbf{U}\}$$

Since the trace value does not change when scaling the input projection matrix \mathbf{U} , we can use this flexibility to constrain $\mathbf{U}^{\top}\tilde{\mathbf{X}}^{\top}\tilde{\mathbf{X}}\mathbf{U} = \mathbf{I}$, and the problem simplifies to (2), concluding the proof.

Other MVA methods can also be shown to have associated different least squares problems. For instance, Canonical Correlation Analysis (CCA) can be shown to minimize $\|\tilde{\mathbf{Y}}\mathbf{V} - \tilde{\mathbf{X}}\mathbf{U}\mathbf{W}\|_F^2$. Note, however, that in general we will be more interested in approximating the original label matrix, $\tilde{\mathbf{Y}}$, rather than a certain projection of it. For this reason, OPLS can be regarded as a specially attractive algorithm for supervised feature extraction.

In Figure 1 we illustrate the projection vectors derived from the PCA, PLS and OPLS algorithms for a binary classification problem (in this case \mathbf{Y} is a column of $\{\pm 1\}$ labels). It can be seen that PCA just follows the variance of the input data. Regarding PLS, it indeed takes into account the label information, but directions of the input space with a large variance still get very much emphasized. Finally, OPLS projections are obtained with the objective of predicting the output labels. Consequently, a much more discriminative projection vector is extracted.

III. KERNEL METHODS FOR PARTIAL LEAST SQUARES

All previous methods assume that there exists a *linear* relation between the original data matrices, $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$, and the extracted projections, $\tilde{\mathbf{X}}'$ and $\tilde{\mathbf{Y}}'$, respectively. However, in many situations this linearity assumption is not satisfied, and non-linear feature extraction is needed to obtain acceptable performance. In this context, *kernel methods* are a promising approach, as they constitute an excellent framework to formulate non-linear versions from linear algorithms [22]–[24].

Kernel versions of different MVA methods have been proposed in recent years. After providing a brief overview of the principles of kernel methods, in this section we describe the Kernel PLS implementation of [23], to which we will refer hereafter as KPLS2. This algorithm has probably become the most successful kernel MVA method in the machine learning community, and its use is preferred over, e.g. Kernel CCA, since the latter is more prone to overfitting.

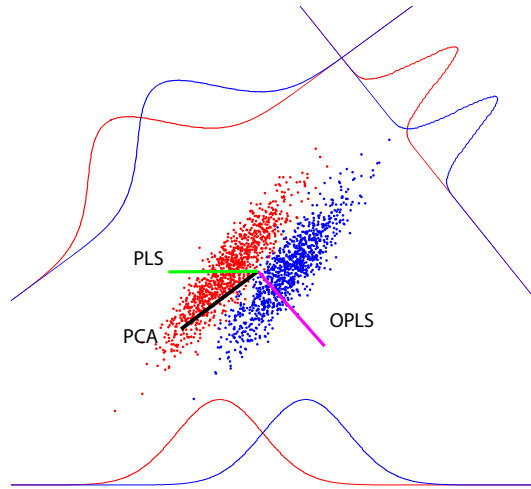


Fig. 1. First projection vector obtained with the PCA, PLS and OPLS methods for a binary classification problem. Histograms of the projected data for the three methods are also shown parallel to the corresponding projection vectors.

A. Background on kernels methods

Kernel methods offer a very general framework for machine learning applications (classification, clustering, regression, density estimation, and visualization) with many types of data (such as time series, images, strings, or objects). The main idea of kernel methods is to project the data into a higher (possibly infinite) dimensional Reproducing Kernel Hilbert Space (RKHS) \mathcal{H} , usually known as *feature space*, and then performing a linear algorithm in \mathcal{H} to detect relations in the embedded data. The result is a nonlinear algorithm from the point of view of the input data space. Hence, kernel methods provide non-linear expressive capabilities, while simultaneously keeping many of the advantages of linear algorithms (for which a well-established theory and efficient methods are available).

The non-linear mapping into feature space is denoted here by $\phi : \mathbb{R}^N \rightarrow \mathcal{H}$. Linear algorithms will benefit from this mapping because of the (usually) very large dimensionality of the Hilbert space, \mathcal{H} . However, the computational burden would dramatically increase if one needed to deal with high dimensionality vectors. Fortunately, many linear learning algorithms can be rewritten in terms of the dot products among the training patterns only; therefore, their non-linear (kernel) version do not require to know explicitly the coordinates of the non-linearly mapped samples, $\phi(\mathbf{x}_i)$, but only the dot products among them: $k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$.

Application of the so-called *kernel trick* considerably simplifies the formulation of kernel algorithms, and enables us to work even in infinite dimensional spaces. Basically, the problem of selecting the mapping function is converted into that of choosing an appropriate kernel. The most useful kernels are those ones whose associated mappings fulfil Mercer's Theorem [37], [38]. Some popular kernels are the linear ($k(\mathbf{x}_i, \mathbf{x}_j) = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$), the polynomial ($k(\mathbf{x}_i, \mathbf{x}_j) = (\langle \mathbf{x}_i, \mathbf{x}_j \rangle + 1)^d$, $d \in \mathbb{Z}^+$), and the Gaussian ($k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/2\sigma^2)$, $\sigma \in \mathbb{R}^+$).

B. Kernel Partial Least Squares

Some recent developments based on kernel methods have been done to obtain non-linear PLS-based algorithms from linear ones while still solving linear equations only [23], [31]. Notationally, consider we are given a set of pairs $\{\phi(\mathbf{x}_i), \mathbf{y}_i\}_{i=1}^l$, with $\phi(\mathbf{x}) : \mathbb{R}^N \rightarrow \mathcal{H}$ a function that maps the input data into some *feature space* of very large or even infinite dimension. Data matrices for performing PLS in \mathcal{H} are now given by $\tilde{\Phi} = [\phi(\mathbf{x}_1), \dots, \phi(\mathbf{x}_l)]^\top$ and $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_l]^\top$.

As in the linear case, the aim of KPLS is to find directions of maximum covariance between the input data in \mathcal{H} and \mathbf{Y} , and can thus be expressed as:

$$\begin{aligned} \text{KPLS:} \quad & \text{maximize: } \text{Tr}\{\mathbf{U}^\top \tilde{\Phi}^\top \tilde{\mathbf{Y}} \mathbf{V}\} \\ & \text{subject to: } \mathbf{U}^\top \mathbf{U} = \mathbf{V}^\top \mathbf{V} = \mathbf{I} \end{aligned} \quad (5)$$

where $\tilde{\Phi}$ is a centered version of Φ , and \mathbf{U} is a projection matrix of size $\dim(\mathcal{H}) \times n_p$.

Now, making use of the Representer's Theorem [23], which states that all projection vectors (the columns of \mathbf{U}) can be expressed as a linear combination of the training data, we can introduce $\mathbf{U} = \tilde{\Phi}^\top \mathbf{A}$ into the previous formulation, where $\mathbf{A} = [\alpha_1, \dots, \alpha_{n_p}]$ and α_i is an l -length column vector containing the coefficients for the i th projection vector, and the maximization problem can be reformulated as follows:

$$\begin{aligned} \text{KPLS (2):} \quad & \text{maximize: } \text{Tr}\{\mathbf{A}^\top \mathbf{K}_x \tilde{\mathbf{Y}} \mathbf{V}\} \\ & \text{subject to: } \mathbf{A}^\top \mathbf{K}_x \mathbf{A} = \mathbf{V}^\top \mathbf{V} = \mathbf{I} \end{aligned} \quad (6)$$

where we have defined the symmetric centered kernel matrix $\mathbf{K}_x = \tilde{\Phi} \tilde{\Phi}^\top$ containing the inner products between any two points in feature space.

The solution to the above problem can be obtained from the Singular Value Decomposition (SVD) of $\mathbf{K}_x \tilde{\mathbf{Y}}$, and can be efficiently computed using the following two-step iterative procedure resulting in the KPLS2 algorithm (see [23, Sec. 6.7.] for more details):

1. Find the largest singular value of $\mathbf{K}_x \tilde{\mathbf{Y}}$, and the associated vector directions: $\{\boldsymbol{\alpha}_i, \mathbf{v}_i\}$.
2. Deflate the kernel matrix using:

$$\mathbf{K}_x \leftarrow \left[\mathbf{I} - \frac{\mathbf{K}_x \boldsymbol{\alpha}_i \boldsymbol{\alpha}_i^\top \mathbf{K}_x}{\boldsymbol{\alpha}_i^\top \mathbf{K}_x \mathbf{K}_x \boldsymbol{\alpha}_i} \right] \mathbf{K}_x \left[\mathbf{I} - \frac{\mathbf{K}_x \boldsymbol{\alpha}_i \boldsymbol{\alpha}_i^\top \mathbf{K}_x}{\boldsymbol{\alpha}_i^\top \mathbf{K}_x \mathbf{K}_x \boldsymbol{\alpha}_i} \right] \quad (7)$$

C. Remarks

The main problem following this approach is that the computational load for building the kernel matrix increases with the square of the number of training samples, as it do the memory requirements, which is particularly critical in the context of applications that involve large datasets. Furthermore, the solution matrix \mathbf{A} will in general be dense, so that when we want to extract features for new data, it will become necessary to compute the kernels between the new data and all the training patterns. Some solutions have been proposed to avoid obtaining dense solutions in KPLS, by either approximating the mapping through the Nyström method [39], or modifying the cost function to impose sparsity [40]. In the next section, we describe a recently proposed method for implementing OPLS in feature space [34]. Apart from the inherent advantages of using OPLS rather than standard PLS, Kernel OPLS can easily be extended to impose sparsity, providing the algorithm with good scalability properties, and making feasible its application to large datasets, such as those typically encountered in remote sensing.

IV. SPARSE KERNEL ORTHONORMALIZED PLS

The Kernel Orthonormalized PLS (rKOPLS) [34] method consists of two core parts: (i) a kernel version of OPLS (called KOPLS), and (ii) a sparse approximation for large scale data sets, which leads to the rKOPLS method.

To derive the kernel implementation of OPLS we will first project the input data into a feature space, where the KOPLS method can be stated as follows:

$$\begin{aligned} \text{KOPLS:} \quad & \text{maximize: } \text{Tr}\{\mathbf{U}^\top \tilde{\boldsymbol{\Phi}}^\top \tilde{\mathbf{Y}} \tilde{\mathbf{Y}}^\top \tilde{\boldsymbol{\Phi}} \mathbf{U}\} \\ & \text{subject to: } \mathbf{U}^\top \tilde{\boldsymbol{\Phi}}^\top \tilde{\boldsymbol{\Phi}} \mathbf{U} = \mathbf{I} \end{aligned} \quad (8)$$

As it occurs with its linear counterpart, and in contrast to other kernel MVA methods, the features derived from KOPLS are optimal (in the mean square error sense) for non-linear multiregression of the label matrix in the feature space.

Now, we can proceed similarly to what we did for the KPLS algorithm. Application of the Representer's Theorem [23] allows us to express the optimal KOPLS projection matrix as $\mathbf{U} = \tilde{\Phi}^\top \mathbf{A}$, where $\mathbf{A} = [\alpha_1, \dots, \alpha_{n_p}]$ is the solution to:

$$\begin{aligned} \text{KOPLS (2):} \quad & \text{maximize: } \text{Tr}\{\mathbf{A}^\top \mathbf{K}_x \tilde{\mathbf{Y}} \tilde{\mathbf{Y}}^\top \mathbf{K}_x \mathbf{A}\} \\ & \text{subject to: } \mathbf{A}^\top \mathbf{K}_x \mathbf{K}_x \mathbf{A} = \mathbf{I} \end{aligned} \quad (9)$$

It is easy to see that the KOPLS method we have just described suffers from the same drawbacks of the KPLS2 algorithm, as explained in Section III-C. All these limitations can be alleviated at once by considering a sparse approximation for the projection matrix [34], $\mathbf{U} = \Phi_R^\top \mathbf{B}$, where Φ_R is a subset of the training data containing only R samples ($R < l$) and $\mathbf{B} = [\beta_1, \dots, \beta_{n_p}]$ contains the parameters of the compact model. Note that, to keep the algorithm as simple as possible, we decided not to center the patterns in the basis Φ_R . Our simulation results suggest that centering Φ_R does not result in improved performance. Introducing this approximation into (8), the rKOPLS algorithm can be formulated as follows:

$$\begin{aligned} \text{rKOPLS :} \quad & \text{maximize: } \text{Tr}\{\mathbf{B}^\top \mathbf{K}_R \tilde{\mathbf{Y}} \tilde{\mathbf{Y}}^\top \mathbf{K}_R^\top \mathbf{B}\} \\ & \text{subject to: } \mathbf{B}^\top \mathbf{K}_R \mathbf{K}_R^\top \mathbf{B} = \mathbf{I} \end{aligned} \quad (10)$$

where we have defined $\mathbf{K}_R = \Phi_R \tilde{\Phi}^\top$, which is a reduced kernel matrix of size $R \times l$. Using Lagrange multipliers to solve the constrained maximization problem above shows that the solution of (10) (i.e., the columns of \mathbf{B}) is given by the generalized eigenvectors associated to the n_p largest generalized eigenvalues of the problem

$$\mathbf{K}_R \tilde{\mathbf{Y}} \tilde{\mathbf{Y}}^\top \mathbf{K}_R^\top \beta = \lambda \mathbf{K}_R \mathbf{K}_R^\top \beta \quad (11)$$

Rather than searching for all the columns of \mathbf{B} at once, it is more convenient to extract projections sequentially, so that, at each iteration, the partial solution is optimal with respect to the current number of projections. Therefore, we will use the following two-step iterative procedure:

1. Find the largest generalized eigenvalue of (11) and its corresponding eigenvector: $\{\lambda_i, \beta_i\}$.

Normalize β_i to satisfy the norm constraint in (10):

$$\beta_i \leftarrow \frac{\beta_i}{\sqrt{\beta_i^\top \mathbf{K}_R \mathbf{K}_R^\top \beta_i}}$$

TABLE I

CHARACTERIZATION OF THE PROPOSED KOPLS AND rKOPLS ALGORITHMS, COMPARED TO THE KPLS2 METHOD. WE DENOTE THE RANK OF A MATRIX WITH $r(\cdot)$.

	KOPLS	rKOPLS	KPLS2
#nodes	l	R	l
Kernel size	$l \times l$	$R \times l$	$l \times l$
Storage	$O(l^2)$	$O(R^2)$	$O(l^2)$
Max. n_p	$\min\{r(\Phi), r(\mathbf{Y})\}$	$\min\{R, r(\Phi), r(\mathbf{Y})\}$	$r(\Phi)$

2. Deflate matrix $\mathbf{K}_R \tilde{\mathbf{Y}} \tilde{\mathbf{Y}}^\top \mathbf{K}_R^\top$ according to:

$$\mathbf{K}_R \tilde{\mathbf{Y}} \tilde{\mathbf{Y}}^\top \mathbf{K}_R^\top \leftarrow \mathbf{K}_R \tilde{\mathbf{Y}} \tilde{\mathbf{Y}}^\top \mathbf{K}_R^\top - \lambda_i \mathbf{K}_R \mathbf{K}_R^\top \beta_i \beta_i^\top \mathbf{K}_R \mathbf{K}_R^\top$$

The motivation for this deflation scheme can be found in [23], in the discussion of generalized eigenvalue problems. Intuitively, it can be seen that it is equivalent to removing from $\tilde{\mathbf{Y}}$ whatever can be predicted from the newly extracted projections, $\mathbf{K}_R^\top \beta_i$:

$$\tilde{\mathbf{Y}} \leftarrow \tilde{\mathbf{Y}} - \sqrt{\lambda_i} \mathbf{K}_R^\top \beta_i$$

It can be shown that this deflation scheme decreases by 1 the rank of $\mathbf{K}_R \tilde{\mathbf{Y}} \tilde{\mathbf{Y}}^\top \mathbf{K}_R^\top$. Since its original rank is no more than the rank of $\tilde{\mathbf{Y}}$, that is the maximum number of projections that can be extracted with rKOPLS.

The sparse constraint for \mathbf{U} provides several desirable properties. First, the resulting feature extractor requires at most R kernel evaluations per pattern. Regarding the training complexity, it can be seen that the number of kernel evaluations drops to $R \cdot l$ (i.e., the size of \mathbf{K}_R), and the size of the involved matrices is just $R \times R$. Table I shows a comparison of KOPLS, rKOPLS and KPLS2 in terms of structural complexity, storage requirements, and size of the kernel matrix. It should be mentioned here that the sparse approximation strategy can not be applied to KPLS2 to obtain an algorithm of reduced complexity, since its deflation scheme [eq. (7)] would still involve the whole kernel matrix.

A final comment regarding the selection of the data points that constitute the approximation basis Φ_R : following [41], we consider random selection among the training patterns, a method that is known to provide satisfactory results as long as R is not too small [42]. More sophisticated strategies, such as clustering methods, could also be explored at the cost of extra computation.

With respect to the number of elements in the basis, parameter R should in principle be selected using cross-validation. However, there are some practical factors that may eventually impose an upper limit on R . For example, rKOPLS requires the inversion of an $R \times R$ matrix ($\mathbf{K}_R \mathbf{K}_R^\top$). Furthermore, increasing R results in more complex architectures, and sometimes it can be preferable to sacrifice performance for the sake of computational simplicity. In this paper, we have kept $R \leq 1000$.

V. EXPERIMENTAL RESULTS

In this section, we analyze the performance of the rKOPLS algorithm in remote sensing data feature extraction, and compare the results provided by standard linear and non-linear PLS-based algorithms. Several datasets are used in order to assess method's capabilities and analyze its particular characteristics, both in pattern recognition (image classification) and function approximation (model inversion) problems.

A. Feature extraction for classification

This subsection is devoted to the analysis of the rKOPLS algorithm in remote sensing classification problems. The suitability of MVA methods in general, and the rKOPLS method in particular, to classification setups is guaranteed if an appropriate labeling scheme (e.g. 1-of- C encoding) is used to encode class membership into the label matrix. In this setup, features revealing useful to predict \mathbf{Y} can also be expected to provide useful information to solve the classification problem. See, e.g., [43] for a discussion about application of square loss to classification problems.

In particular, we focus in two typical cases of high dimensional image classification in remote sensing: (1) pixel-based hyperspectral image classification, and (2) multispectral image classification in which contextual information is stacked to the spectral information. In both cases the input space may become redundant, either because of a densely sampled spectrum or because of the collinearity introduced by the (locally stationary) spatial features, respectively.

1) Experiment 1: Pixel-based hyperspectral image classification: In our first experiment, we used the standard AVIRIS image taken over NW Indiana's Indian Pine test site in June 1992. Discriminating among the major crops can be very difficult (in particular, given the moderate spatial

resolution of 20 meters), which has made the scene a challenging benchmark to validate classification accuracy of hyperspectral imaging algorithms. The calibrated data is available online (along with detailed ground-truth information) from <http://dynamo.ecn.purdue.edu/~biehl/MultiSpec>. In all our experiments we used the whole scene, consisting of the full 145×145 pixels, which contains 16 classes, ranging in size from 20 – 2468 pixels, and thus constituting a very difficult situation. We removed 20 noisy bands covering the region of water absorption, and finally worked with 200 spectral bands.

For the kernel methods, a Gaussian kernel $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|_2^2/2\sigma^2)$ was used, and a 10-fold cross-validation procedure on the training set was followed to estimate σ . Once the projections were obtained, the discriminative power of the features was tested using a simple linear model followed by a “winner-takes-all” activation function: $\hat{y} = \text{w.t.a.}[\mathbf{W}^\top \tilde{\mathbf{x}}]$ for the linearly extracted features and $\hat{y} = \text{w.t.a.}[\mathbf{W}^\top \tilde{\phi}(\mathbf{x})]$ for the kernel methods, where \mathbf{W} is the optimal regression matrix given by $\mathbf{W} = \tilde{\mathbf{X}}^\dagger \tilde{\mathbf{Y}}$ and $\mathbf{W} = \tilde{\Phi}^\dagger \tilde{\mathbf{Y}}$ for the linear and kernel methods, respectively.

Figure 2 shows the obtained results (overall accuracy and kappa statistic) as a function of the number of extracted features, n_p , when using both linear (PLS and OPLS) and non-linear (KPLS2 and rKOPLS) methods. Several conclusions can be obtained. First, the OA[%] and kappa statistic curves exhibit very similar behaviors, which suggest that the obtained models are not unbiased in terms of accuracy. Second, it can be observed how OPLS outperforms PLS in the whole domain of extracted features, which confirms its (linear) optimality in the MSE sense. Third, non-linear (kernel) methods can produce improved results. In particular, KPLS2 improves the accuracy of a linear PLS approximation for $n_p > 30$, while the proposed rKOPLS outperforms its linear OPLS counterpart (and obviously PLS) for any number of extracted features.

For the sake of a fair comparison between KPLS2 and rKOPLS, we studied the influence of the number of nodes (l and R , respectively) on the accuracy of the classifiers. Note that the best validation results were obtained for $R = 1000$, but we did not increase the parameter beyond this value for the reasons stated at the end of Section IV. In Fig. 2, we show rKOPLS and KPLS2 performance for varying R and for different numbers of extracted features: $n_p = \{1, \dots, 250\}$ and $n_p < l$ for KPLS2, and $n_p \leq \text{rank}(\mathbf{Y}) = 15$ for rKOPLS. Note that imposing sparsity in the solution, as it is done in rKOPLS, is a very different approach from a simple subsampling, since matrix \mathbf{K}_R still retains information about all training points. When looking at both OA[%]

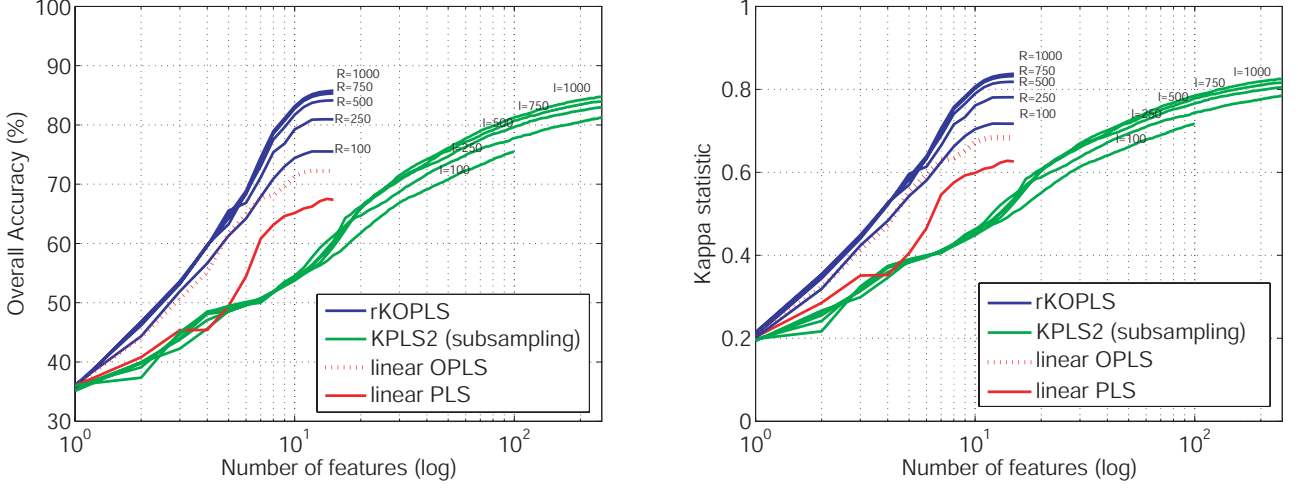


Fig. 2. Overall accuracy (left) and kappa statistic (right) as a function of the number of extracted features for different linear and non-linear PLS-based schemes. For the KPLS2 and the proposed rKOPLS we plot the results for different number of subsampling nodes, $R, l = \{100, 250, 500, 750, 1000\}$.

and the κ statistic, results show that KPLS2 performance is only comparable to rKOPLS if using between 100 – 250 features as compared to the only 15 features used for rKOPLS. Note that for the same number of extracted features ($n_p = 15$) the proposed method obtains an average gain of about $\Delta OA = +20\%$ and the kappa statistic goes from a mean value of 0.55 to 0.8. It is also noteworthy that the method shows an excellent behavior as a function of the subsampling factors (R and l), since only a small gain is obtained for $R > 750$. In the limit of n_p , we should stress that the behavior of KPLS2 and rKOPLS should be very similar. However, we can obtain much better results with the rKOPLS method with a lower computational and memory burden.

Finally, Fig. 3 shows how the (non-linear) features extracted by rKOPLS can be used to develop very efficient linear classifiers, either with a simple pseudo-inverse, as we illustrated before, or by solving a linear SVM. It can be observed that the offered (linear) solution rapidly approximates the state-of-the-art (non-linear) SVM classifier as R is increased. The RBF SVM solution needed 1295 support vectors to reach these results.

2) *Experiment 2: Contextual-based multispectral image classification:* For our second experiment, we considered a Landsat MSS image consisting of 82×100 pixels with a spatial resolution of $80m \times 80m$ (all data acquired from a rectangular area approximately 8 km wide). Six classes are identified in the image, namely red soil, cotton crop, grey soil, damp grey soil, soil with

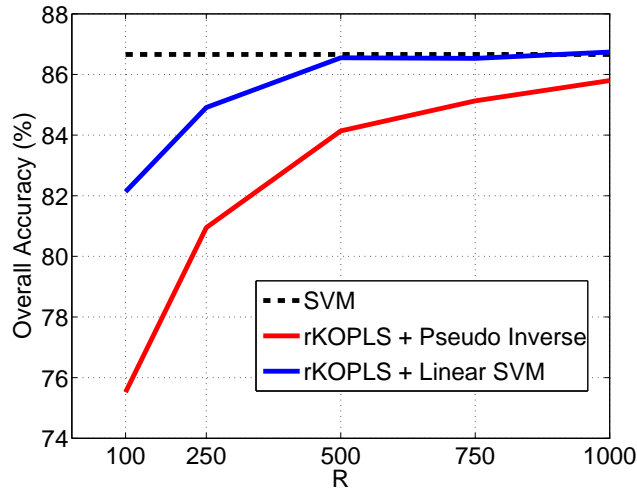


Fig. 3. Overall accuracy as a function of parameter R for different classification schemes after rKOPLS feature extraction.

vegetation stubble and very damp grey soil. In order to improve the performance of the classifiers, contextual information was included stacking neighbouring pixels in 3×3 windows. Therefore, 36-dimensional input samples were generated, with a high degree of redundancy and collinearity. A total of 4435 samples were used for training and 2000 for testing. The processed image is available from <http://www.ics.uci.edu/~mlearn/MLRepository.html>.

We illustrate the discriminative performance of the features calculated by linear OPLS, rKOPLS and KPLS2. We use here the same design used in the previous experiment. Nevertheless, we avoid the study of the impact of R , and keep $R = 500$ or $R = 1000$, for which validation results did not differ significantly. Figure 4[left] compares linear OPLS vs rKOPLS ($R = 500$), clearly showing that the non-linear method provides a better representation of the discriminative information which is hidden in the data. rKOPLS performance, with only 5 features, is 91.00%, which is very close to the 91.85% offered by the SVM with RBF kernel. However, the RBF SVM needed 1711 support vectors to achieve this score. In Fig. 4[right], we compare the accuracy when using rKOPLS ($R=1000$) and KPLS2 features. Since KPLS2 requires the whole kernel matrix, resulting in a much more complex training phase, we consider also KPLS2 with subsampling to get a fair comparison. Results show that KPLS2 performance is only comparable to rKOPLS if using around 100 features (91,1% and 91,8% for the schemes with and without subsampling, respectively) as compared to the 5 features used for rKOPLS (note the log scale in the x axis).

In other words, rKOPLS features contain more discriminative information and allow developing compact classifiers for high dimensional remote sensing applications.

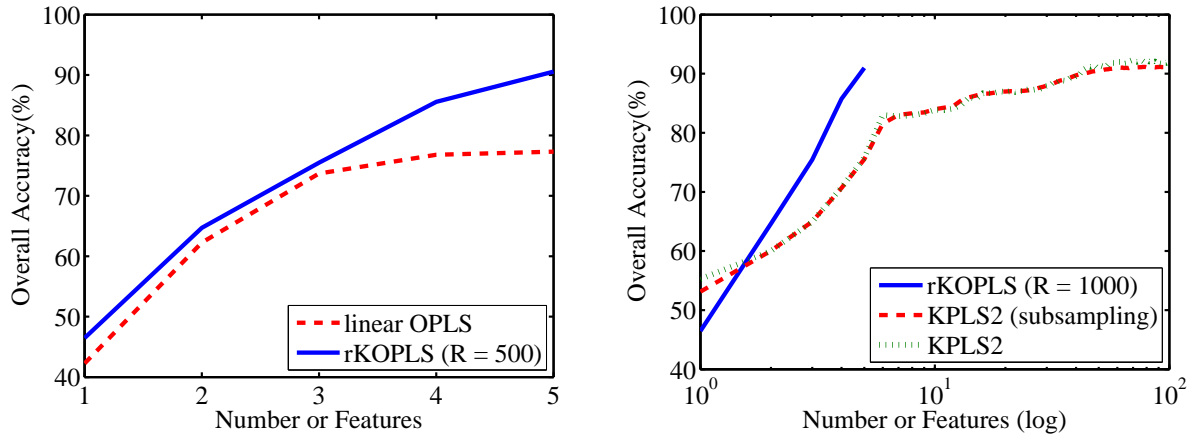


Fig. 4. Overall accuracy as a function of the number of extracted features for different PLS-based schemes.

B. Feature extraction for regression

In order to assess the performance of the algorithm for function approximation tasks, we focus on two challenging problems: the estimation of oceanic chlorophyll concentration from multispectral measurements, and the Leaf Area Index (LAI) estimation from hyperspectral images. In both cases, satellite-derived data and *in situ* measurements are subjected to high levels of uncertainty. In these difficult scenarios, a proper (robust) feature extraction is necessary, particularly when their relationship is believed to be non-linear or the data is scarce thus leading to badly conditioned problems.

1) *Experiment 3: Oceanic chlorophyll concentration prediction:* In this experiment, we concentrate on the performance of the method in the specific problem of modeling the non-linear relationship between chlorophyll concentration and marine reflectance. For this purpose, we used the SeaBAM dataset [44], available at <http://seabass.gsfc.nasa.gov/seabam/seabam.html>, which gathers 919 *in-situ* pigment measurements around the United States and Europe. The dataset contains coincident *in situ* chlorophyll concentration and remote sensing reflectance measurements ($R_{rs}(\lambda)$, [sr^{-1}]) at some wavelengths (412, 443, 490, 510 and 555 nm) that are present in the SeaWiFS ocean color satellite sensor. The chlorophyll concentration values range from

0.019 to 32.79 mg/m³. We transformed the concentration data logarithmically, $Y_{CC} = \log(CC)$, according to [45]. Hereafter, units of all accuracy and bias measurements are referred to Y_{CC} [$\log(\text{mg}/\text{m}^3)$] instead of CC [mg/m^3]. The available data was split into two sets: 460 samples for training, and the remaining 459 samples for testing.

As in the previous applications, we used the Gaussian kernel for all methods. For rKOPLS, we varied R linearly in the range $[5, 100]$ and σ logarithmically in the range $[10^{-2}, 10^4]$, and computed the leave-one-out root mean square error (LOO-RMSE) in the training set to validate the model. The best LOO-RMSE observed was 0.131. See Fig. 5 for the LOO-RMSE surface. It is worth noting that the method obtains approximately stable results for any fixed value of σ , which suggests that a very sparse solution is good enough to describe the distribution. This confirms the results in [46], where just a few number of support vectors was necessary to build accurate regression models for this problem. It can also be noticed that the most critical parameter is the kernel width, showing a clear *plateau* for values $\sigma \geq 10^{-1}$.

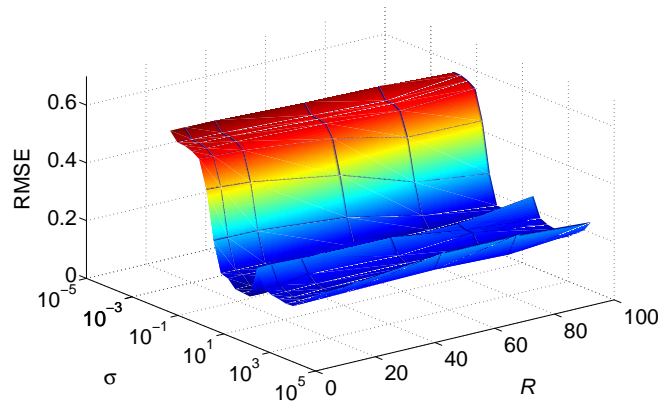


Fig. 5. Evolution of the estimated leave-one-out RMSE as a function of R and the kernel width, σ .

Table II shows the results in the test set obtained by different regression kernel methods: support vector regression (SVR) [46] with different cost functions, KPLS2 with different number of extracted features ($n_p = \{1, 5, 10, 20\}$), and the proposed rKOPLS ($n_p = 1$) [35]. For comparison purposes, we include results obtained with models Morel-1, Morel-3, and CalCOFI 2-band (cubic and linear), as they performed best among a set of 15 *empirical* estimation models of the Cl-a concentration in [44]. We show the following measures for the prediction errors: mean error (ME) as a measure of bias; the root mean square error (RMSE) and the mean absolute

TABLE II
RESULTS FOR THE OCEANIC CHLOROPHYLL CONCENTRATION PREDICTION PROBLEM. MEAN ERROR (ME), ROOT MEAN-SQUARED ERROR (RMSE), MEAN ABSOLUTE ERROR (MAE), AND CORRELATION COEFFICIENT (r) OF MODELS IN THE TEST SET.

Model	ME	RMSE	MAE	r
<i>Morel-1</i> [†]	-0.023	0.178	0.139	0.956
<i>Morel-3</i>	-0.025	0.182	0.143	0.954
<i>CalCOFI 2-band cubic</i>	-0.051	0.177	0.142	0.960
<i>CalCOFI 2-band linear</i>	0.079	0.325	0.256	0.956
ε -SVR	-0.070	0.139	0.105	0.971
L_2 -loss SVR	-0.034	0.140	0.107	0.971
<i>OPLS</i>	-0.034	0.257	0.188	0.903
<i>KPLS2</i> , $n_p = 1$	0.042	0.366	0.278	0.790
<i>KPLS2</i> , $n_p = 5$	-0.013	0.189	0.140	0.947
<i>KPLS2</i> , $n_p = 10$	-0.013	0.149	0.115	0.968
<i>KPLS2</i> , $n_p = 20$	-0.009	0.138	0.106	0.972
<i>rKOPLS</i> , $n_p = 1$	-0.015	0.154	0.111	0.967

[†]The results provided in this table for Morel and CalCOFI models slightly differ from the ones given in [44] since they are shown for the test set. In addition, models in [44] used all available data to fit the models and hence no validation procedure was followed.

error (MAE) as a measure of accuracy, and the correlation coefficient (r) between the desired output and the output offered by the models as a measure of fit in the test set.

Several conclusions can be obtained. First, OPLS performs poorly as the linear assumption does not hold in this problem. Second, the obtained results using KPLS2 and the proposed rKOPLS show a clear improvement in both accuracy and bias compared to linear OPLS. Additionally, they show similar results to those from SVR, with a clear improvement in terms of bias (ME) but slightly worse in terms of fit (RMSE, MAE). It should be noted here that these similar results are obtained with a lower computational and storage burden, specially significant in the case of the proposed rKOPLS method, as training with $R \leq 50$ does not virtually degrade the test results (cf. Fig. 5). Comparing the two kernel-based PLS feature extraction methods, we can see that the feature from rKOPLS provides a similar performance to the 10 first features from KPLS2, which illustrates the good expressive power of rKOPLS projections in supervised problems.

2) *Experiment 4: Leaf Area Index estimation:* The last experiment deals with the estimation of the Leaf Area Index (LAI) from hyperspectral satellite images, a problem which is characterized by a high uncertainty present in the data. We used data from the ESA Spectra Barrax Campaigns (SPARC), <http://gpds.uv.es/sparc/>. The campaign was carried out in Barrax (N30°3', W2°6'), an agriculture test area situated within La Mancha region in the south of Spain, from 12 to 14 July 2003 and from 14 to 16 July 2004. The area was analyzed for agricultural research for many years thanks to its flat topography (differences in elevation range up to 2 m only) and to the presence of large and uniform vegetation fields (e.g., alfalfa, corn, sugar beet, onions, garlic, potatoes) with a wide range of LAI, from 0.5 up to 6.

During the campaign, covering LAI ground measurements were collected, and seventeen hyperspectral and multiangular CHRIS/PROBA images (ten during SPARC-2003 and seven within the SPARC-2004 campaign) were acquired. Field non-destructive measurements LAI were made by means of the digital analyzer LI-COR LAI-2000 [47]. For reducing the effect of multiple scattering, the instrument was only operated near dusk and dawn (6:30-9:30 am; 6:30-8:30 pm) under diffuse radiation conditions using one sensor for both above and below canopy measurements. In order to prevent interference caused by the operator's presence and the illumination condition, the sensor field of view was limited with a 180° view-cap. Both measurements were azimuthally oriented opposite to the sun azimuth angle. Finally, a total of 139 samples of LAI measurements were taken, comprising one full set of measurements in each Elementary Sampling Unit (ESU). The final database consists of 139 LAI measurements and their associated 62 CHRIS reflectance levels.

In Figure 6, we compare the different algorithms through the leave-one-out RMSE as a function of the free parameter σ . It should be noted that, as observed before, the proposed rKOPLS yields higher accuracy than KPLS2, specially relevant by noting that our method only uses one extracted feature. In addition, the proposed method shows lower LOO-RMSE for almost all values of the kernel free parameter σ , which alleviates the problem of exhaustive search of the free parameters.

Table III shows the obtained quantitative results with all available data obtained by different regression kernel-based methods: support vector regression (SVR) with different cost functions, KPLS2 with different number of extracted features ($n_p = \{1, 3, 5, 7, 10\}$), and the proposed rKOPLS ($n_p = 1$). For comparison purposes, we also include the results obtained with the standard Weighted Difference Vegetation Index (WDVI) [48]. In order to estimate the LAI, we

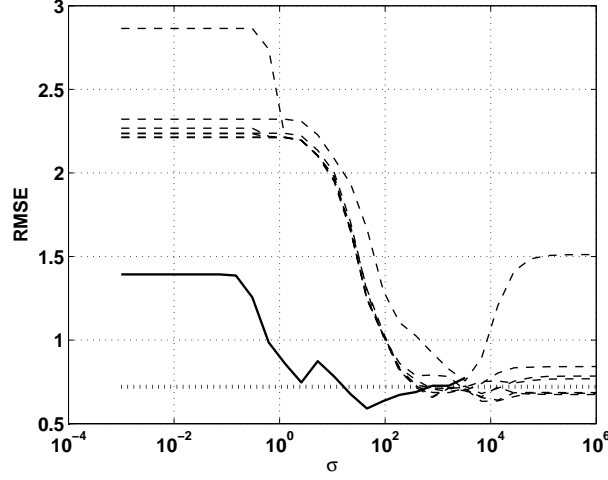


Fig. 6. Leave-one-out RMSE as a function of the kernel parameter σ for the basic PLS (dotted line), KPLS2 with $n_p = \{1, 3, 5, 7, 10\}$ (dashed lines) and the proposed rKOPLS (solid line).

used the formula:

$$\text{LAI} = -\frac{1}{\alpha} \log \left(1 - \frac{\text{WDVI}}{\text{WDVI}_{\infty}} \right), \quad (12)$$

where α is a parameter to be adjusted, WDVI assumes that the ratio between the near infrared (NIR) and red reflectances of bare soil is constant, $\text{WDVI} = \rho_{\text{NIR}} - \beta \rho_{\text{R}}$, WDVI_{∞} represents the asymptotic limiting value for WDVI, and β is the slope of the soil line. The used bands are $\rho_{\text{NIR}} = 663$ nm (band 24) and $\rho_{\text{R}} = 831$ nm (band 46). In our case, we show results for two different sets of parameters; those recommended in [49] ($\alpha = 0.4$, $\beta = 1.1$), and the parameters that minimized the bias-variance of the residuals in this particular subset of the database ($\alpha = 0.18$, $\beta = 0.28$).

We can observe that all methods outperform the WDVI-based semi-empirical model, which results in extremely biased models. Linear PLS performs poorly again due to the strong non-linearity, and the high dimensionality of the samples, which boosts the curse of dimensionality and collinearity problems. We can also notice that the obtained results using KPLS2 and the proposed rKOPLS show a clear improvement in both accuracy and bias compared to linear PLS. Better results are, nevertheless, obtained with the SVR, both in accuracy and bias. Comparing the two kernel-based PLS feature extraction methods, we can see that the feature from rKOPLS provides a similar performance to the 10 first features from KPLS2, which once again illustrates

TABLE III

RESULTS FOR THE LEAF AREA INDEX (LAI) ESTIMATION PROBLEM. MEAN ERROR (ME), ROOT MEAN-SQUARED ERROR (RMSE), MEAN ABSOLUTE ERROR (MAE), AND CORRELATION COEFFICIENT (r) OF MODELS IN THE TEST SET.

Model	ME	RMSE	MAE	r
<i>WDVI</i> ($\alpha = 0.4, \beta = 1.1$) [49]	-0.112	1.124	0.964	0.844
<i>WDVI</i> ($\alpha = 0.18, \beta = 0.28$)	-0.134	0.896	0.733	0.859
ε -SVR	-0.003	0.721	0.501	0.884
L_2 -loss SVR	-0.004	0.711	0.502	0.882
<i>PLS</i>	0.051	0.886	0.708	0.851
<i>KPLS2</i> , $n_p = 1$	0.056	0.940	0.742	0.832
<i>KPLS2</i> , $n_p = 5$	0.026	0.803	0.598	0.879
<i>KPLS2</i> , $n_p = 10$	0.005	0.749	0.544	0.896
<i>rKOPLS</i> , $n_p = 1$	-0.003	0.758	0.525	0.893

the richness of rKOPLS projections.

In order to assess statistical differences among methods, we performed a one-way ANOVA study on the residuals. Significant statistical differences were observed among kernel methods and linear methods ($F = 0.01, p = 0.995$), but they disappeared when removing the PLS algorithm from the study ($p < 0.001$), thus suggesting that KPLS2 and rKOPLS provide similar predictions both numerically and statistically, but the latter significantly extracts only one feature.

VI. CONCLUSIONS

This work studied the applicability of the rKOPLS method for feature extraction and dimensionality reduction in remote sensing applications, both for classification and regression problems. For rKOPLS, sparsity is imposed so that the algorithm can efficiently deal with high dimensional input samples, such as those encountered in hyperspectral image processing problems, and scales well with the number of training samples. The sparse approximation used by rKOPLS is specially convenient in this context, given that otherwise a huge kernel matrix should be stored and processed. We have observed that the method outperforms the standard KPLS2 for a given number of extracted features and with regard to both computational burden and memory requirements. Also, rKOPLS produces similar results to SVM classifier and regression

machines but with much lower computational cost and memory demands. In conclusion, the proposed method can be useful for non-linear feature extraction in the context of remote sensing applications.

ACKNOWLEDGMENTS

This work has been partly supported by Spanish Ministry of Education and Science under projects DATASAT/ESP2005-07724-C05-03, CONSOLIDER/CSD2007-00018, TEC-2005-00992, and by Madrid Community grant S-505/TIC/0223.

The authors would like to express their gratitude to Dr. F. Vuolo from CEO at Ariespace, University of Naples, Italy, for providing LAI measurements from the SPARC campaign.

REFERENCES

- [1] I. Hiroaki, N. Toyonori, and T. Eiji, "Measurement of pesticide residues in food based on diffuse reflectance IR spectroscopy," *IEEE Transactions on Instrumentation and Measurement*, vol. 51, no. 5, pp. 886–890, Oct 2002.
- [2] Y. Ni, P. Qiu, and S. Kokot, "Simultaneous determination of three organophosphorus pesticides by differential pulse stripping voltammetry and chemometrics," *Analytica Chimica Acta*, vol. 516, no. 1-2, pp. 7–17, July 2004.
- [3] P. Geladi, B. Sethson, J. Nyström, T. Lillhonga, Lestander T., and J. Burger, "Chemometrics in spectroscopy: Part 2. examples," *Spectrochimica Acta Part B: Atomic Spectroscopy*, vol. 59, no. 9, pp. 1347–1357, Sept 2004.
- [4] D. Purcell, M. G. O'Shea, and S. Kokot, "Role of chemometrics for at-field application of NIR spectroscopy to predict sugarcane clonal performance," *Chemometrics and Intelligent Laboratory Systems*, vol. 87, no. 1, pp. 113–124, May 2007.
- [5] N. C. Coops, M-L. Smith, M.E. Martin, and S. V. Ollinger, "Prediction of eucalypt foliage nitrogen content from satellite-derived hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 6, pp. 1338–1346, June 2003.
- [6] P.A. Townsend, J.R. Foster, Jr. Chastain, R.A., and W.S. Currie, "Application of imaging spectroscopy to mapping canopy nitrogen in the forests of the central Appalachian Mountains using Hyperion and AVIRIS," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 6, pp. 1347–1354, June 2003.
- [7] M.D. Wilson, S.L. Ustin, and D.M. Rocke, "Classification of contamination in salt marsh plants using hyperspectral reflectance," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 5, pp. 1088–1095, May 2004.
- [8] E. Naesset, O. Martin Bollandas, and T. Gobakken, "Comparing regression methods in estimation of biophysical properties of forest stands from two different inventories using laser scanner data," *Remote Sensing of Environment*, vol. 94, no. 4, pp. 541–553, Feb 2005.
- [9] L. Li, S.L. Ustin, and D. Riaño, "Retrieval of fresh leaf fuel moisture content using genetic algorithm partial least squares (GA-PLS) modeling," *IEEE Geoscience and Remote Sensing Letters*, vol. 4, no. 2, pp. 216–220, April 2007.
- [10] S. Wold, C. Albano, W. J. Dunn, U. Edlund, K. Esbensen, P. Geladi, S. Hellberg, E. Johansson, W. Lindberg, and M. Sjostrom, *Chemometrics, Mathematics and Statistics in Chemistry*, chapter Multivariate Data Analysis in Chemistry, p. 17, Reidel Publishing Company, 1984.

- [11] H. Wold, "Path models with latent variables: the NIPALS approach," *Quantitative sociology: International perspectives on mathematical and statistical model building*, pp. 307–357, 1975.
- [12] K. Worsley, J. Poline, K. Friston, and A. Evans, "Characterizing the response of PET and fMRI data using multivariate linear models (mlm)," *NeuroImage*, vol. 6, pp. 305–319, 1998.
- [13] P. D. Sampson, A. P. Streissguth, H. M. Barr, and F. L. Bookstein, "Neurobehavioral effects of prenatal alcohol: Partial Least Squares analysis," *Neurotoxicology and teratology*, vol. 11, pp. 477–491, 1989.
- [14] P. Geladi, "Notes on the history and nature of partial least squares (PLS) modelling," *Journal of Chemometrics*, vol. 2, pp. 231–246, 1988.
- [15] J. A. Wegelin, "A survey of partial least squares (PLS) methods, with emphasis on the two-block case," Tech. Rep., Univ. of Washington, 2000.
- [16] R. Rosipal and N. Kramer, "Overview and recent advances in partial least squares," *Subspace, Latent Structure and Feature Selection Techniques*, 2006.
- [17] S. Roweis and C. Brody, "Linear heteroencoders," Tech. Rep. 1999-002, Gatsby Computational Neuroscience Unit, 1999.
- [18] S. Wold, N. Kettaneh, and B. Skagerberg, "Nonlinear PLS modelling," *Chemometrics and Intelligent Laboratory Systems*, vol. 7, pp. 53–65, 1989.
- [19] S. Wold, "Nonlinear partial least squares modeling II, Spline inner relation," *Chemolab*, vol. 14, pp. 71–84, 1992.
- [20] I. E. Frank, "A nonlinear PLS model," *Chemolab*, vol. 8, pp. 109–119, 1990.
- [21] G. Baffi, E. B. Martin, and A. J. Morris, "Non-linear projection to latent structures revisited (the neural network PLS algorithm)," *Computers Chemical Engineering*, vol. 23, pp. 1293–1307, 1999.
- [22] B. Schölkopf and A. Smola, *Learning with Kernels – Support Vector Machines, Regularization, Optimization and Beyond*, MIT Press Series, 2002.
- [23] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*, Cambridge University Press, 2004.
- [24] G. Camps-Valls, J. L. Rojo-Álvarez, and M. Martínez-Ramón, Eds., *Kernel Methods in Bioengineering, Signal and Image Processing*, Idea Group Publishing, Hershey, PA (USA), Jan 2007.
- [25] G. Camps-Valls and L. Bruzzone, "Kernel-based methods for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 6, pp. 1351–1362, June 2005.
- [26] G. Camps-Valls, L. Gómez-Chova, J. Muñoz Marí, J. Vila-Francés, and J. Calpe-Maravilla, "Composite kernels for hyperspectral image classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 3, no. 1, pp. 93–97, Jan 2006.
- [27] J. Muñoz Marí, L. Bruzzone, and G. Camps-Valls, "A support vector domain description approach to supervised classification of remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 8, pp. 2683–2692, 2007.
- [28] A. Banerjee, P. Burlina, and C. P. Diehl, "A support vector method for anomaly detection in hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 8, pp. 2282–2291, Aug 2006.
- [29] G. Camps-Valls, L. Bruzzone, J. L. Rojo-Álvarez, and F. Melgani, "Robust support vector regression for biophysical variable estimation from remotely sensed images," *IEEE Geoscience and Remote Sensing Letters*, vol. 3, no. 3, pp. 339–343, Jul 2006.
- [30] F. Yang, M.A. White, A.R. Michaelis, K. Ichii, P. Hashimoto, H. Votava, A-X. Zhu, and R.R. Nemani, "Prediction of continental-scale evapotranspiration by combining MODIS and AmeriFlux data through support vector machine," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 11, Part 2, pp. 3452–3461, 2006.

- [31] R. Rosipal and L. J. Trejo, "Kernel partial least squares regression in reproducing kernel Hilbert space," *Journal of Machine Learning Research*, vol. 2, pp. 97–123, 2001.
- [32] W. H. A. M. van den Broek, E. P. P. A. Derks, E. W. van de Ven, D. Wienke, P. Geladi, and L. M. C. Buydens, "Plastic identification by remote sensing spectroscopic NIR imaging using kernel partial least squares (KPLS)," *Chemometrics and Intelligent Laboratory Systems*, vol. 35, no. 2, pp. 187–197, 1996.
- [33] B. M. Nicolai, K. I. Theron, and J. Lammertyn, "Kernel PLS regression on wavelet transformed NIR spectra for prediction of sugar content of apple," *Chemometrics and Intelligent Laboratory Systems*, vol. 85, no. 2, pp. 243–252, 2007.
- [34] J. Arenas-García, K. B. Petersen, and L. K. Hansen, "Sparse kernel orthonormalized PLS for feature extraction in large data sets," in *Advances in Neural Information Processing Systems 19*, B. Schölkopf, J.C. Platt, and T. Hofmann, Eds. MIT Press, Cambridge, MA, 2007.
- [35] J. Arenas-García and G. Camps-Valls, "Feature extraction from remote sensing data using kernel orthonormalized PLS," in *IEEE International Conference on Geoscience and Remote Sensing Symposium, IGARSS'2007*, July 2007.
- [36] I. T. Jolliffe, *Principal Component Analysis*, Springer-Verlag, 1986.
- [37] J. Mercer, "Functions of positive and negative type and their connection with the theory of integral equations," *Philosophical Transactions of the Royal Society of London. Series A*, vol. CCIX, no. A456, pp. 215–228, May 1905.
- [38] M. A. Aizerman, E. M. Braverman, and L.I. Rozoner, "Theoretical foundations of the potential function method in pattern recognition learning," *Automation and remote Control*, vol. 25, pp. 821–837, 1964.
- [39] L. Hoegaerts, J. A. K. Suykens, J. Vanderwalle, and B. De Moor, "Primal space sparse kernel partial least squares regression for large problems," in *Proc. of the International Joint Conference on Neural Networks*, 2004.
- [40] M. Momma and K. Bennet, "Sparse kernel partial least squares regression," in *Proc. of Conference on Learning Theory, COLT*, 2003.
- [41] Y.-J. Lee and O. L. Mangasarian, "RSVM: Reduced Support Vector Machines," Tech. Rep. 00-07, Data Mining Inst., Comp. Sci. Dept., Univ. Wisconsin, Madison, WI, 2000.
- [42] Y.-J. Lee and S.-Y. Huang, "Reduced support vector machines: A statistical theory," *IEEE Trans. Neural Networks*, vol. 18, no. 1, pp. 1–13, 2007.
- [43] Ryan Rifkin and Aldebaro Klautau, "In defense of one-vs-all classification," *Journal of Machine Learning Research*, vol. 5, no. 1, pp. 101–141, Jan 2004.
- [44] J. E. O'Reilly, S. Maritorena, B. G. Mitchell, D. A. Siegel, K. Carder, S. A. Garver, M. Kahru, and C. McClain, "Ocean color chlorophyll algorithms for SeaWiFS," *Journal of Geophysical Research*, vol. 103, no. C11, pp. 24937–24953, Oct 1998.
- [45] P. Cipollini, G. Corsini, M. Diani, and R. Grass, "Retrieval of sea water optically active parameters from hyperspectral data by means of generalized radial basis function neural networks," *IEEE Transactions On Geoscience And Remote Sensing*, vol. 39, pp. 1508–1524, 2001.
- [46] G. Camps-Valls, L. Bruzzone, J.L. Rojo-Álvarez, and F. Melgani, "Robust support vector regression for biophysical parameter estimation from remotely sensed images," *IEEE Geoscience and Remote Sensing Letters*, vol. 3, no. 3, pp. 339–343, 2006.
- [47] LI-COR, "LAI-2000 plant canopy analyzer instruction manual," Tech. Rep., LI-COR, Lincoln, NE, 1992.
- [48] J. P. G. W Clevers, "The derivation of a simplified reflectance model for the estimation of leaf area index," *Remote Sensing of Environment*, vol. 25, pp. 53–69, 1988.
- [49] F. Vuolo, G. D'Urso, and L. Dini, "Cost-effectiveness of vegetation biophysical parameters retrieval from remote sensing

data,” in *IEEE International Conference on Geoscience and Remote Sensing Symposium, IGARSS'2006*, July 2006, pp. 1949–1952.



Jerónimo Arenas-García (S'00-M'04) was born in Seville, Spain, in 1977. He received the telecommunication engineer degree (honors) from Universidad Politécnica de Madrid, Spain, in 2000 and the Ph.D. degree in telecommunication technologies (honors) from Universidad Carlos III de Madrid, Leganés, Spain, in 2004. After a postdoctoral stay at the Technical University of Denmark, Denmark, he returned to Universidad Carlos III de Madrid, where he is currently a Lecturer of Digital Signal and Information Processing. His current research interests are focused in the fields of statistical learning, specially in adaptive algorithms, advance machine learning techniques and multivariate analysis methods for feature extraction, and their applications in remote sensing data and multimedia information retrieval.



Gustavo Camps-Valls (M'04, SM'07) was born in València, Spain in 1972, and received a B.Sc. degree in Physics (1996), a B.Sc. degree in Electronics Engineering (1998), and a Ph.D. degree in Physics (2002) from the Universitat de València. He is currently an associate professor in the Department of Electronics Engineering at the Universitat de València, where teaches electronics, advanced time series processing, and signal processing. His research interests are tied to the development of machine learning algorithms, with particular attention to adaptive systems, neural networks, and kernel methods for signal and image processing. Special focus is on remote sensing image processing and recognition (multi-temporal classification and change detection, feature selection and extraction, regression and estimation, data fusion). He conducts and supervises research on these topics within the frameworks of several national and international projects, and he is Evaluator of project proposals and scientific organizations. He is the author (or co-author) of 50 journal papers, more than 60 international conference papers, several international book chapters, and editor of the book “*Kernel methods in bioengineering, signal and image processing*” (IGI, 2007). He is a referee of many international journals and conferences, and currently serves on the Program Committees of SPIE Europe, and IGARSS. Visit <http://www.uv.es/gcamps> for more information.