Feature Extraction from Remote Sensing Data using Kernel Orthonormalized PLS

Jerónimo Arenas-García* and Gustavo Camps-Valls†

*Dept. Signal Theory and Communications. Universidad Carlos III de Madrid. Spain.
jarenas@tsc.uc3m.es, http://www.tsc.uc3m.es/∼jarenas

†Dept. Enginyeria Electrònica. Universitat de València.
C/ Dr. Moliner, 50. 46100. Burjassot, València. Spain.
gustavo.camps@uv.es, http://www.uv.es/∼gcamps

***Abstract*—This paper presents the study of a sparse kernel-based method for non-linear feature extraction in the context of remote sensing classification and regression problems. The so-called Kernel Orthonomalized PLS algorithm with reduced complexity (rKOPLS) has two core parts: (i) a kernel version of OPLS (called KOPLS), and (ii) a sparse (reduced) approximation for large scale data sets, which ultimately leads to rKOPLS. The method demonstrates good capabilities in terms of expressive power of the extracted features and scalability.**

I. Introduction

Partial Least Squares (PLS) is a family of methods for analyzing relations between data sets by means of latent variables. In remote sensing data processing, PLS methods have been extensively used [1]–[3], since they are well-suited to deal with ill-posed and collinearity problems, such as those encountered when analyzing hyperspectral images. Despite its good performance in many applications, one encounters a main shortcoming; as they are based on linear projections, poor performance is observed when the variables of the input and output spaces are non-linearly related. Consequently, non-linear versions of PLS have been recently developed [4], [5], being the use of kernels a promising approach [5]. Orthonormalized PLS (OPLS) is another multivariate analysis method which enjoys certain optimality conditions with respect to basic PLS, as discussed in [6]. Recently, a kernel extension of OPLS, the so-called Kernel OPLS with reduced complexity (rKOPLS) algorithm, was proposed for feature extraction [7].

In this paper, we study the applicability of rKOPLS to feature extraction in the context of remote sensing data regression and classification problems. The paper is outlined as follows. Section II briefly reviews the idea and standard formulations of basic (linear) PLS-based 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. Partial Least Squares Methods

Partial Least Squares (PLS) is a standard multivariate regression method developed by Herman Wold in 1966 [8]. 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.

Notationally, the basic PLS algorithm considers we are given a set of pairs $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{l}$, with $\mathbf{x}_i \in \mathbb{R}^N$, $\mathbf{y}_i \in \mathbb{R}^M$. Let us now introduce matrices $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_l]^\top$ and $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_l]^\top$, where the $\top$ superscript denotes matrix or vector transposition. Let us also denote by $\mathbf{X}' = \mathbf{X}\mathbf{U}$ and $\mathbf{Y}' = \mathbf{Y}\mathbf{V}$ two matrices, each one containing n_p projections of the original input and output data, $\mathbf{U}$ and $\mathbf{V}$ being the projection matrices of sizes $N \times n_p$ and $M \times n_p$, respectively. The goal of PLS is to find the directions of maximum covariance between the projected input and output data:

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

where $\mathbf{I}$ is the identity matrix of size n_p, and $\mathbf{C}_{xy}$ represents the covariance between the input and output datasets which is given by $\mathbf{C}_{xy} = \frac{1}{l}\tilde{\mathbf{X}}^\top \tilde{\mathbf{Y}}$, where we have denoted by $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$ the centered versions of $\mathbf{X}$ and $\mathbf{Y}$, respectively.

In the literature, many variants of PLS exist [9]–[13]. In this paper we pay attention to the Orthonormalized Partial Least Squares (OPLS) [14], which tackles the following maximization problem:

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

with $\mathbf{C}_{xx} = \frac{1}{l}\tilde{\mathbf{X}}^\top \tilde{\mathbf{X}}$. It is worth noting that, unlike PLS, the OPLS method only extracts projections from the input data. The OPLS method is optimal (in the mean square error sense) for performing *linear* multiregression of $\mathbf{Y}$ on the projected

input data for a given number of features [6]. In other words, it can be shown that the solution to (2) also minimizes the sum of squares of the residuals of the approximation of the label matrix, $\|\tilde{\mathbf{Y}} - \tilde{\mathbf{X}}'\mathbf{W}\|_F^2$, where $\|\cdot\|_F$ denotes the Frobenius norm, and $\mathbf{W} = (\tilde{\mathbf{X}}'^{\top}\tilde{\mathbf{X}}')^{-1}\tilde{\mathbf{X}}'^{\top}\tilde{\mathbf{Y}}$ is the optimal regression matrix.

III. Kernel Partial Least Squares

All previous methods assume that there exists a *linear* relation between the latent variables of $\mathbf{X}$ and of $\mathbf{Y}$. However, this might not necessarily hold, and thus non-linear versions have become necessary to solve this problem. In this context, *kernel methods* are a promising approach, as they constitute an excellent framework to formulate non-linear versions from linear algorithms [5], which has been demonstrated to be very useful in different application domains [15]. Some recent developments based on kernel methods have been done to obtain non-linear PLS-based algorithms from linear ones while still solving only linear equations [4], [5].

Notationally, consider we are given a set of pairs $\{\boldsymbol{\phi}(\mathbf{x}_i), \mathbf{y}_i\}_{i=1}^{l}$, with $\boldsymbol{\phi}(\mathbf{x}) : \mathbb{R}^N \rightarrow \mathcal{H}$ a function that maps the input data into some Reproducing Kernel Hilbert Space (RKHS), usually referred to as *feature space*, of very large or even infinite dimension. Let us also introduce the matrices $\boldsymbol{\Phi} = [\boldsymbol{\phi}(\mathbf{x}_1), \ldots, \boldsymbol{\phi}(\mathbf{x}_l)]^{\top}$ and $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_l]^{\top}$, and denote by $\boldsymbol{\Phi}' = \boldsymbol{\Phi}\mathbf{U}$ a matrix containing n_p features of the original input data, $\mathbf{U}$ being a projection matrix of size $dim(\mathcal{H}) \times n_p$.

The objective of Kernel PLS (KPLS) is to find directions for maximum covariance, and thus can be expressed as:

$$\begin{array}{lll} \text{KPLS:} & \text{maximize:} & \text{Tr}\{\mathbf{U}^{\top}\tilde{\boldsymbol{\Phi}}^{\top}\tilde{\mathbf{Y}}\mathbf{V}\} \\ & \text{subject to:} & \mathbf{U}^{\top}\mathbf{U} = \mathbf{V}^{\top}\mathbf{V} = \mathbf{I} \end{array} \tag{3}$$

where $\tilde{\boldsymbol{\Phi}}$ and $\tilde{\mathbf{Y}}$ are centered versions of $\boldsymbol{\Phi}$ and $\mathbf{Y}$, respectively.

Application of the Representer's Theorem [5] allows us to solve the above problem using only a matrix of inner products ot the patterns in $\mathcal{H}$, namely $\mathbf{K}_x = \tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\top}$. The main problem following this approach is that the computational cost and storage requirements increase very fastly with the number of samples, which is particularly critical in the context of remote sensing large-scale applications. In the next section, we propose a sparse scalable kernel version of OPLS for non-linear feature extraction.

IV. Sparse Kernel Orthonormalized PLS

In this section, we review the Kernel Orthonomalized PLS algorithm with reduced complexity (rKOPLS) [7]. The 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.

Our proposal first considers the development of the *kernel* OPLS, which is stated as:

$$\begin{array}{lll} \text{KOPLS:} & \text{maximize:} & \text{Tr}\{\mathbf{U}^{\top}\tilde{\boldsymbol{\Phi}}^{\top}\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^{\top}\tilde{\boldsymbol{\Phi}}\mathbf{U}\} \\ & \text{subject to:} & \mathbf{U}^{\top}\tilde{\boldsymbol{\Phi}}^{\top}\tilde{\boldsymbol{\Phi}}\mathbf{U} = \mathbf{I} \end{array} \tag{4}$$

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	$\min\{r(\boldsymbol{\Phi}), r(\mathbf{Y})\}$	$\min\{R, r(\boldsymbol{\Phi}), r(\mathbf{Y})\}$	$r(\boldsymbol{\Phi})$

This algorithm performs linear OPLS (instead of a standard PLS) in the feature kernel space, which implicitely becomes non-linear in the original input space. Unlike KPLS, the features derived from KOPLS are optimal (in the mean square error sense) for non-linear multiregression in the training dataset.

Now, making use of the Representer's Theorem [5], 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 $\mathbf{U} = \tilde{\boldsymbol{\Phi}}^{\top}\mathbf{A}$ into the previous formulation, where $\mathbf{A} = [\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_{n_p}]$ and $\boldsymbol{\alpha}_i$ is an l-length column vector containing the coefficients for the ith projection vector, and the maximization problem can be reformulated as follows:

$$\begin{array}{lll} \text{KOPLS:} & \text{maximize:} & \text{Tr}\{\mathbf{A}^{\top}\mathbf{K}_x\mathbf{K}_y\mathbf{K}_x\mathbf{A}\} \\ & \text{subject to:} & \mathbf{A}^{\top}\mathbf{K}_x\mathbf{K}_x\mathbf{A} = \mathbf{I} \end{array} \tag{5}$$

where we have defined the centered kernel matrices $\mathbf{K}_x = \tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^{\top}$ and $\mathbf{K}_y = \tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^{\top}$.

As it is explained in [7], some advantages are obtained by considering the approximation $\mathbf{U} = \boldsymbol{\Phi}_R^{\top}\mathbf{B}$, where $\boldsymbol{\Phi}_R$ is a subset of the training data containing only R samples ($R < l$) and $\mathbf{B} = [\boldsymbol{\beta}_1, \cdots, \boldsymbol{\beta}_{n_p}]$ contains the parameters of the compact model. Using this compact approximation the rKOPLS algorithm can be reformulated as follows:

$$\begin{array}{lll} \text{rKOPLS :} & \text{maximize:} & \text{Tr}\{\mathbf{B}^{\top}\mathbf{K}_R\mathbf{K}_y\mathbf{K}_R^{\top}\mathbf{B}\} \\ & \text{subject to:} & \mathbf{B}^{\top}\mathbf{K}_R\mathbf{K}_R^{\top}\mathbf{B} = \mathbf{I} \end{array} \tag{6}$$

where we have defined $\mathbf{K}_R = \boldsymbol{\Phi}_R\tilde{\boldsymbol{\Phi}}^{\top}$, which is a reduced kernel matrix of size $R \times l$. The sparse approximation for $\mathbf{U}$ not only results in a method with good scalability properties, but it also shows helpful for overfitting avoidance.

Table I shows a comparison of KOPLS, rKOPLS, and the KPLS version of [5] (to which we will refer to as KPLS2) in terms of structural complexity, storage requirements, and size of the kernel matrix (for more details, please refer to [7]).

V. Experimental results

In this section, we analyze the performance of the rKOPLS algorithm in remote sensing imaging feature extraction, and compare the derived features to those provided by standard linear and non-linear PLS-based algorithms. Several datasets are used in order to assess method's capabilities, both in classification and regression (model inversion) problems.

A. Feature extraction for classification

For our first experiment, we consider a LandSat image consisting of 82×100 pixels with a spatial resolution of

80m×80m (all data acquired from a rectangular area approximately 8km wide). Six classes are identified in the image, namely red soil, cotton crop, grey soil, damp grey soil, soil with 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.

In our experiments, we illustrate the discriminative performance of the features calculated by linear OPLS, rKOPLS and KPLS2. For the kernel methods, a Gaussian kernel was deployed $k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\|\mathbf{x}_i - \mathbf{x}_j\|_2^2/2\sigma^2\right)$ using 10-fold cross-validation on the training set to estimate σ. Once the projections were obtained, classification was carried out for a varying number of features according to $\tilde{\mathbf{\Phi}}'\mathbf{W}$, where $\mathbf{W}$ is obtained in terms of the pseudoinverse of the projected training data as the optimal regression matrix using a "winner-takes-all" activation function. Since we are extracting non-linear features, there is little interest on using more sophisticated classification methods.

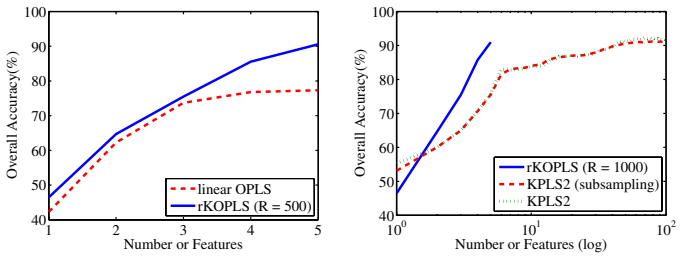

Fig. 1. Overall accuracy as a function of the number of extracted features for different PLS-based schemes.

Figure 1[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%. In Fig. 1[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[1]. 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.

[1]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.

B. Feature extraction for regression

In the second experiment, we concentrate on the performance of the method in the particular problem of modeling the non-linear relationship between chlorophyll concentration and marine reflectance. For this purpose, we have used the SeaBAM dataset [16], 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/m^3. We transformed the concentration data logarithmically, $Y_{CC} = \log(CC)$, according to [17]. Hereafter, units of all accuracy and bias measurements are referred to Y_{CC} $[\log(mg/m^3)]$ 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 first application, 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. 2 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 [18], where just a few number of support vectors was necessary to build accurate regression models for this problem. The most critical parameter is the kernel width, showing a clear *plateau* for values $\sigma \geq 10^{-1}$.

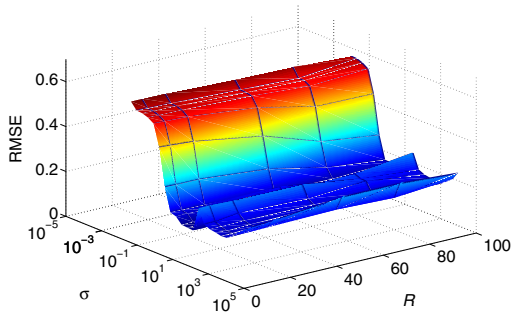

Fig. 2. Evolution of the estimated leave-one-out RMSE as a function of R and the kernel width, σ.

Table II shows the results in the test set obtained by different regression kernel methods: support vector regression (SVR) [18] with different cost functions, KPLS2 with different number of extracted features ($n_p = \{1, 5, 10, 20\}$), and the proposed rKOPLS ($n_p = 1$). 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

TABLE II
MEAN ERROR (ME), ROOT MEAN-SQUARED ERROR (RMSE), MEAN ABSOLUTE ERROR (MAE), AND CORRELATION COEFFICIENT (r) OF MODELS IN THE TEST SET.

	ME	**RMSE**	**MAE**	**r**
Morel-1†	-0.023	0.178	0.139	0.956
Morel-3	-0.025	0.182	0.143	0.954
CalCOFI 2-band cubic	-0.051	0.177	0.142	0.960
CalCOFI 2-band linear	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
OPLS	-0.034	0.257	0.188	0.903
KPLS2, $n_p = 1$	0.042	0.366	0.278	0.790
KPLS2, $n_p = 5$	-0.013	0.189	0.140	0.947
KPLS2, $n_p = 10$	-0.013	0.149	0.115	0.968
KPLS2, $n_p = 20$	-0.009	0.138	0.106	0.972
rKOPLS, $n_p = 1$	-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 [16] since they are shown for the test set. In addition, models in [16] used all available data to fit the models and hence no validation procedure was followed.

concentration in [16]. 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 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. 2). 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.

VI. CONCLUSIONS

This work studied the applicability of the rKOPLS method for feature extraction and dimensionality reduction in hyperspectral imaging, both for classification and regression problems. Unlike KPLS, the rKOPLS makes the data in the feature space orthonormal. In addition, sparsity is imposed so 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 produces similar results to SVM classifier and regression machines but with much lower computational cost and memory requirements. Next steps will consider extensive comparison with other PLS-based algorithms in large-scale remote sensing classification and regression scenarios.

ACKNOWLEDGMENTS

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

REFERENCES

[1] 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.

[2] 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.

[3] E. Naesset, O. Martin Bollandsas, 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.

[4] 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.

[5] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*, Cambridge University Press, 2004.

[6] S. Roweis and C. Brody, "Linear heteroencoders," Tech. Rep. 1999-002, Gatsby Computational Neuroscience Unit, 1999.

[7] 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.

[8] 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.

[9] H. Wold, "Path models with latent variables: the NIPALS approach," *Quantitative sociology: International perspectives on mathematical and statistical model building*, pp. 307–357, 1975.

[10] 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.

[11] P. Geladi, "Notes on the history and nature of partial least squares (PLS) modelling," *Journal of Chemometrics*, vol. 2, pp. 231–246, 1988.

[12] J. A. Wegelin, "A survey of partial least squares (PLS) methods, with emphasis on the two-block case," Tech. Rep., Univ. of Washington, 2000.

[13] R. Rosipal and N. Kramer, "Overview and recent advances in partial least squares," *Subspace, Latent Structure and Feature Selection Techniques*, 2006.

[14] K. Worsley, J. Poline, K. Friston, and A. Evans, "Characterirzing the response of PET and fMRI data using multivariate linear models (mlm)," *NeuroImage*, vol. 6, pp. 305–319, 1998.

[15] 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.

[16] 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.

[17] 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.

[18] 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.