Proceedings of the VI Brazilian Conference on Neural Networks - VI Congresso Brasileiro de Redes Neurais
pp. 373–377, June 2–5, 2003 - Centro Universitario da FEI, Sao Paulo - SP - Brazil

Multi-Kernel Based PLS regression

Ruy L. Milidiú, Raúl P. Rentería
Departamento de Informática, PUC-Rio
Rua Marquês de São Vicente 225, Gávea, Rio de Janeiro, Brazil
E-mails: {milidiu,renteria}@inf.puc-rio.br

Abstract

MKPLS, a non-linear version of the Partial Least-Squares regression is presented. The non-linearity is introduced in the classical algorithm through the use of multiple kernel functions, thus providing an straightforward non-linear adaptation. MKPLS provides a multi-kernel based version for the PLS algorithm with a competitive modeling error. Experimental results show that the use of different kernels for the regression model enhances the predictive power when compared to a PLS regression based on only one function kernel.

1. Introduction

Partial Least Squares regression (1; 2) has been widely used in the chemometric field for the robustness of the generated model when the number of variables is large when compared to the number of samples. This led to its application to many other areas, such as process monitoring, marketing analysis and image processing (3; 4; 5; 6; 7; 8).

In this paper, we propose MKPLS, a multi-kernel based algorithm for Partial Least-Squares regression. A kernel PLS2 algorithm based on only one kernel has already been proposed in (9), showing that the use of non-linear modeling can improve predictive power. With MKPLS we show that using different kernels at the training phase provides a better adaptation to the input data, resulting in not only a more compact model but also a better prediction quality.

In order to measure the performance of MKPLS, we report some experiments on data sets mainly related to NIR spectra analysis, such as wheat data for chemometrics (10) or combustible (11). For the kernel based regression, LPLS is used, a kernel PLS formulation for the case of only one dependent variable (PLS1), that shows better numerical stability when compared to the PLS kernel algorithm in (9).

In section 2, our multi-kernel approach is described. In section 3, the empirical results obtained with the selected data set are shown. Finally, in section 4, we summarize our findings.

2. MKPLS: Multi-Kernel PLS regression algorithm

The main motivation for MKPLS was the PRESS curve obtained with one kernel PLS when compared to the standard linear PLS. For example, if both curves are

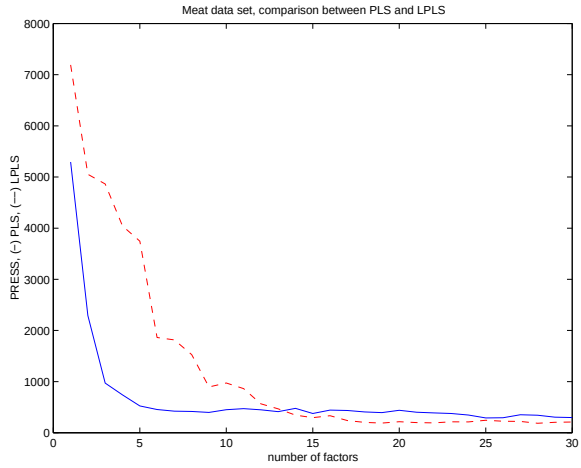

Figure 1: PRESS values for PLS and LPLS, Meat data set.

plotted (figure 1) for the *Meat* data set described in section 3.1.5, we see that one kernel PLS outperforms PLS if one uses more than 13 factors. However, the performance of one kernel PLS is really poor for the first factors. The polynomial kernel defined as $K_{i,j} = (x_i \cdot x_j + 1)^2$ can barely model the predicted variable y for the first 10 factors. It would be interesting to have one regression model that would be as sharp as PLS on the first factors, and as sharp as one kernel PLS on the remaining ones. MKPLS

Multi-Kernel PLS regression approach
1. Apply one kernel PLS regression to first kernel $\mathtt{K}_1$
2. Deflate second kernel $\mathtt{K}_2$ using model obtained in step 1
3. Apply one kernel PLS regression to the deflated kernel $\mathtt{K}_2$

Figure 2: MKPLS main steps.

generalizes the one kernel PLS by using a kernel matrix

K_1 for the first f_1 factors and then switching to a different kernel K_2 for the remaining factors, as indicated by the high level algorithm on figure 2. For the interpretability of this operation it is important that the mapping done with K_2 includes the mapping done by K_1. Since the addition of two kernels is still a kernel, this can be simply done by defining $K_2 = K_1 + K$ where K denotes the kernel for the additional non-linearity, Gaussian or polynomial for instance.
This is meaningful in MKPLS since the switching of the kernels is done by deflating the factors t_i found with K_1. This would make no sense if the mapping of K_1 was not also done with K_2.

2.1. Training step

Given two kernel functions with the corresponding kernel matrices K_1 and K_2 for some training data set, the MKPLS model can be constructed through the following procedure:

1. obtain the first f_1 factors $\{t_i, b_i\}$ applying the one kernel PLS algorithm using the first kernel matrix K_1;

2. deflate the second kernel matrix K_2 and the dependent variable Y by applying the deflating algorithm described in figure 3;

3. apply again the one kernel PLS algorithm to the deflated kernel obtaining the remaining f_2 model factors.

At the end of this procedure, the set of $f_1 + f_2$ factors corresponding to the non-linear multi-kernel model will be available.

K_2 and Y deflation for the training step

```
for i = 1 to f1
    g'i ← K2 ti
    ai ← ti^T ti
    U ← g'i t^T_i / ai
    K2 ← K2 − U − U^T + (t^T_i g'i) ti t^T_i / a^2_i
    Y ← Y − ti b^T_i
end
```

Figure 3: MKPLS training deflating algorithm.

2.2. Prediction step

Since the one kernel PLS prediction algorithm uses a specific kernel matrix related to the test data set, it will be also necessary to switch the kernel matrices K'_1 and K'_2 during the prediction phase.

1. apply the same prediction algorithm starting with K'_1. However the set of f_1 score t'_i should be retained for deflation along with the predicted y;

2. deflate K'_2 using the algorithm in figure 4. Note the use of g'_i obtained during training deflation;

3. apply the prediction algorithm starting with the deflated K'_2 and the predicted y obtained in step 1.

K'_2 deflation for the prediction step

```
for i = 1 to f1
    U1 ← K'2 ti t^T_i / ai
    U2 ← t'i g'^T_i / ai
    K'2 ← K'2 − U1 − U2 + (t^T_i g'i) t'i t^T_i / a^2_i
end
```

Figure 4: MKPLS prediction deflating algorithm.

The MKPLS training and prediction procedures are very close to the one kernel version of PLS. The main difference being the deflation algorithms for the kernel function switching.

3. Experimental results

3.1. Data Set descriptions

3.1.1 Wheat

The first one was taken from Kalivas (10). We used the data set containing the NIR spectra of 100 wheat samples along with specified protein and moisture content. Of the 100 spectra, 70 were utilized for training (calibration) and the 30 remaining ones for testing (validation) the constructed model.

3.1.2 Light gas oil

As the second data set, we used the light gas oil data available at Dalhousie University (11). For the calibration and validation matrices we used the first 70 and remaining 44 samples respectively, along with the concentrations of the four components in each sample.

3.1.3 Combustible

As the third data set, we used a set of 30 combustible samples for which the NIR spectra over 3632 channels have been measured. Samples were reduced to contain only 363 measures by using every tenth response. 21 samples were utilized for calibrating (70% of the set) and the remaining 9 for validating.

3.1.4 Corn

As the fourth data set, the NIR spectra of corn samples were used. This data set consists of 80 samples of corn measured on 3 different NIR spectrometers. As the dependent variables, the moisture, oil, protein and starch values for each of the samples were used.

Table 1: Data Sets used for testing

Data Set	Samples	Ind. Var.	Dep. Var.
Wheat	100	141	2
Meat	215	100	3
Combustible	30	363	3
Light gas oil	114	572	4
Corn	80	700	4

3.1.5 Meat

The Tecator data set was used next, (12), where the task was to predict the fat content of a meat sample on the basis of its near infrared absorbance spectrum. As suggested by the author, the first 172 samples were used for training while the following 43 for testing purposes.

3.2. Experiment results

To compare the PRESS of the model produced by MKPLS with PLS and LPLS, two key characteristics are observed:

1. model complexity;
2. prediction quality.

The number of required factors to achieve a sufficiently small prediction error is our modeling complexity measure. This is obtained by comparing the PRESS curves for either the first 10 or 15 factors. The minimum PRESS value is our prediction quality measure.
For each region just described, the minimum of each curve is compared. Also the percentage of times that MKPLS performed equally or better is calculated since re-sampling is done 20 times. For each data set, the following parameters are used:

1. the first kernel function resulting in matrix K_1;
2. the number f_1 of factors calculated with K_1;
3. the second kernel matrix K_2 used along with its parameters.

For all data sets, the identity kernel yielding the K_1 matrix given by $K_1 = XX^\top$ was used. Polynomial, Gaussian or both kernels were used for K_2 for all experiments. To illustrate the overall behavior of the MKPLS performance, the PRESS values of the three models are plotted for some data sets.
As we can see in figures 5 and 6 the MKPLS modeling benefits from both PLS and LPLS modeling. The poor performance of the non-linear model for the first factors is eliminated, and the good predictive quality at higher factors is maintained. Table 2 shows the results for all data sets regarding the first factors, whereas table 3 shows the performance of MKPLS over the two other models considering up to 30 factors. In both tables *MKPLS over PLS*

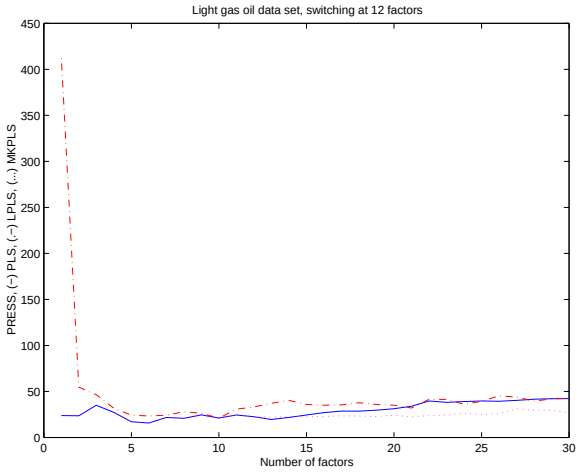

Figure 5: PRESS values of PLS, LPLS and MKPLS for Light gas oil data set.

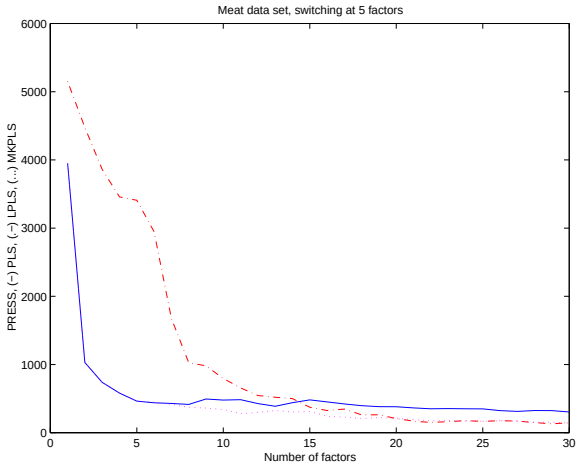

Figure 6: PRESS values of PLS, LPLS and MKPLS for Meat data set.

means the minimum PRESS gain obtained with MKPLS when compared to PLS for the selected factors, or:

$$100 \cdot \left(1 - \frac{min(PRESS_{MKPLS})}{min(PRESS_{PLS})}\right)$$

The same applies to *MKPLS over LPLS*.

Table 4 reports for each data set the kernel function used for the non-linear modeling, the amount of factors observed to evaluate the model complexity and the number of factors used with the identity kernel.

The same non-linear kernel function along with its parameters used for LPLS modeling, were also used for MKPLS.
On some data sets like Combustible, the MKPLS modeling not only benefits from both PLS and LPLS modeling, but results on table 3 show that the use of linear modeling for the first factors improve the quality of the non-linear on the remaining ones. Other data sets like Meat eliminate the poor performance of LPLS for the first factors,

Table 2: MKPLS PRESS comparison between PLS and LPLS for first factors.

Data Set	MKPLS over PLS			
	mean	std.dev.	%draw	%win
Wheat	1.32	4.54	35.0	45.0
Meat	14.04	14.51	0.0	90.0
Combustible	3.09	11.86	45.0	45.0
Light gas oil	-0.32	2.34	75.0	20.0
Corn	1.91	3.01	45.0	55.0

Data Set	MKPLS over LPLS			
	mean	std.dev.	%draw	%win
Wheat	23.81	24.48	0.0	95.0
Meat	58.47	6.84	0.0	100.0
Combustible	45.60	40.14	0.0	95.0
Light gas oil	15.94	18.93	0.0	85.0
Corn	32.90	15.10	0.0	100.0

Table 3: MKPLS PRESS comparison between PLS and LPLS for all factors

Data Set	MKPLS over PLS			
	mean	std.dev.	%draw	%win
Wheat	13.07	12.86	15.0	85.0
Meat	41.47	19.82	0.0	95.0
Combustible	29.36	24.76	25.0	75.0
Light gas oil	0.60	1.95	65.0	30.0
Corn	1.24	11.00	15.0	55.0

Data Set	MKPLS over LPLS			
	mean	std.dev.	%draw	%win
Wheat	6.52	11.89	0.0	80.0
Meat	-4.65	11.70	0.0	45.0
Combustible	48.82	47.36	0.0	90.0
Corn	18.81	16.61	0.0	90.0
Light gas oil	11.22	21.92	0.0	80.0

Table 4: Data Sets used for testing

Data Set Data Set	kernel kernel	Factors observed	N. factors with K_1
Wheat	pol.	10	7
Meat	pol.	10	5
Combustible	pol.	10	7
Light gas oil	gauss.	15	12
Corn	both	15	12

but for the remaining shows a similar behavior.

4. Conclusions

We introduce MKPLS, a multi-kernel based algorithm for Partial Least-Squares regression. Instead of using only one kernel, many can be used during the training and prediction steps.

We have made experiments with 5 chemometric data sets using the identity kernel (resulting into the standard linear PLS algorithm) as the first one and a polynomial, Gaussian, or both when appropriate, for the second. It turns out that the main characteristics of MKPLS are:

1. more compact model;
2. same learning rate as PLS for first factors;
3. competitive prediction quality when compared to LPLS;
4. at least the same performance as other models.

As we can see, MKPLS can be considered an alternative approach when using kernel based PLS regression. Furthermore, the same approach of MKPLS for switching kernels could be used to other kernel based regression schemes, (13; 9).

References

[1] H. Wold. Estimation of principal components and related models by iterative least squares. In P. Krishnaiah, editor, *Multivariate analysis II*, pages 391–420. Academic Press, New York, 1966.

[2] S. Wold, C. Albano, W. D. III, K. Esbensen, S. Hellberg, E. Johansson, and H. Sjöström. Pattern recognition: finding and using regularities in multivariate data. In J. Martens, editor, *Proc. IUFOST Conf. Food Research and Data Analysis*, London, 1983. Applied Science Publications.

[3] A. Morineau and M. Tenenhaus, editors. *Les Méthodes PLS, Symposium International PLS'99*. Cisia-Ceresta, October 1999.

[4] R. Milidiú, R. Machado, and R. Rentería. Time series forecasting throught wavelets and a mixture of experts models. *Neurocomputing*, 28:145–156, 1999.

[5] R. Milidiú and R. Rentería. Apls: A fast approximate algorithm for partial least-squares. In *V Brazilian Conference on Neural Networks*, pages 661–666, Rio de Janeiro - Brazil, April 2001.

[6] R. L. Milidiú and R. Rentería. Ppls: An efficent parallel algorithm for partial least-squares. In *12th Symposium on Computer Architecture and High Performance Computing*, São Pedro, SP, Brazil, October 2000.

[7] R. Milidiú, R. Rentería, and C. J. de Lucena. Dpls and ppls: Two pls algorithms for large data sets. In *2nd International Symposium on PLS and Related Methods*, pages 175–186, Capri, Italy, October 2001.

[8] R. Milidiú and R. Rentería. Dpls and ppls: Two pls algorithms for large data sets. *Computational Statistics and Data Analysis*, 2003. accepted for publication.

[9] R. Rosipal and L. J. Trejo. Kernel partial least squares regression in reproducing kernel hilbert space. *Journal of Machine Learning*, 2:97–123, december 2001.

[10] J. H. Kalivas. Two data sets of near infrared spectra. *Chemometrics and Intelligent Laboratory Systems*, 37:255–259, 1997.

[11] P. Wentzell, D. Andrews, J. Walsh, J. Cooley, and P. Spencer. Estimation of hydrocarbon types in light gas oils and diesel fuels by ultraviolet absorption spectroscopy and multivariate calibration. *Canadian Journal of Chemistry*, 77:391–400, 1999.

[12] I. A. Tecator. Prediction of fat content of a meat sample, 1995. StatLib Datasets Archive.

[13] R. Rosipal, L. Trejo, and A. Cichocki. Kernel principal component regression with em approach to nonlinear principal components extraction. Technical report, University of Paisley, School of Information and Communication Technologies, 2000.