A COVARIANCE-BASED NONLINEAR PARTIAL LEAST SQUARES ALGORITHM

Sanjay K. Sharma, Uwe Kruger, George W. Irwin and Sukhwinder Kumar

*Intelligent Systems & Control Research Group
Queen’s University of Belfast, BT9 5AH, U.K.

*Centre of Process Analytical and Process Technology
University of Newcastle, NE1 7RU, U.K.

Keywords: Genetic algorithms, nonlinear partial least squares.

Abstract

This paper shows that error-based nonlinear extensions to Partial Least Squares (PLS), derived from an iterative technique, do not in fact compute pairs of latent variables to maximise a covariance-based criterion, as in the linear case. This paper introduces a new algorithm to overcome this difficulty. This therefore offers a more natural extension of PLS into the nonlinear domain. Comparative results, from a realistic simulation study on a pH neutralisation process, are used to demonstrate that the new algorithm can indeed maximise the covariance.

1 Introduction

Partial Least Squares (PLS) is a modelling tool for extracting important information between two sets of correlated variables, often referred to as predictor and response variables. More precisely, PLS determines the underlying structures of the process from linear combinations of the original variables, defined by pairs of latent variables or LVs.

PLS has become an important modelling technique in a broad range of application fields. However, the approach is a linear one and consequently inappropriate if the underlying structure between the predictor and response variables is nonlinear. Consequently, various extensions have been proposed to enhance PLS so that nonlinear structures can be identified. These include the incorporation of second-order polynomials [19,1], spline functions [18], artificial neural networks [15,8,17,2] and the Box-Tidwell transformation [12].

In particular, the ideas of Wold et al. [19] and Baffi et al. [1,2], which are directly related to iterative linear PLS, have been regarded as a natural extension to linear PLS. This work shows that these nonlinear extensions compute pairs of latent variables which do not in fact satisfy a covariance-based criterion, as proved by Höskuldsson [9] for linear PLS.

This paper proposes a new nonlinear extension to PLS that determines pairs latent variables that maximise a covariance-based criterion. Consequently, the properties of linear PLS remain valid for this nonlinear extension. The new technique uses a nonlinear transformation of the latent variables associated with the predictor variables, which are used to define a covariance criterion to be maximised in a manner similar to linear PLS.

A genetic strategy is employed to maximise this covariance criterion for each pair of latent variables. The authors have successfully used this method to improve the prediction accuracy of a nonlinear PLS model as compared to that produced by the method Baffi et al. [2]. To demonstrate that the new nonlinear PLS algorithm can indeed determine pairs of latent variables that have a larger covariance, comparative results from an application study on a realistic simulation of a pH neutralisation process are presented.

2 Partial Least Squares Preliminaries

This section gives a brief overview of the principles of linear PLS and their nonlinear counterparts.

2.1 Linear Partial Least Squares

PLS relies on defining the predictor matrix, $X \in \mathbb{R}^{K \times M}$, and the response matrix, $Y \in \mathbb{R}^{K \times N}$, as the sum of rank one component matrices. Both matrices are usually centred, appropriately scaled, and consist of $K$ observations of the predictor and response variables. Furthermore, the predictor matrix includes $M$ predictor variables and the response matrix $N$ response variables.

PLS projects the row vectors of $X$ and $Y$ onto one dimensional subspaces, spanned by weight vectors $w$ and $v$ that have unit length. These projections are stored in two vectors, $t$ and $u$, which are referred to as the score vectors of the predictor and response variables, respectively. The objective function $J_j$ for determining the $j^{th}$ pair of weight vectors, $w_j$ and $v_j$, is to maximise the following cost function:

$$J_j = \frac{1}{K-1} t^T_j u_j, \quad j=1,2,\ldots,K,m$$

where $m$ is the required number of pairs of LVs, $m < M$. The weight and score vectors are determined iteratively and after convergence, the prediction of $u_j$ is given by:
\[ \hat{u}_j = t_j b_j, \]  

with \( b_j \) being a regression coefficient that is determined as follows:

\[ b_j = \frac{u_j^T t_j}{u_j^T u_j}, \]  

(4)

To compute the \((j+1)^{th}\) pairs of weight and score vectors, the variation of the \(j^{th}\) pair of score vectors must be subtracted (deflated) from the predictor and response matrices:

\[
\begin{align*}
X_{j+1} &= X_j - t_j p_j^T, \\
Y_{j+1} &= Y_j - \hat{u}_j q_j^T
\end{align*}
\]

(5)

where the loading vectors \( p_j \) and \( q_j \) are obtained as follows:

\[
\begin{align*}
p_j^T &= \frac{t_j^T X}{t_j^T t_j}, \\
q_j^T &= \frac{\hat{u}_j^T Y}{\hat{u}_j^T \hat{u}_j}
\end{align*}
\]

(6)

The above process is repeated until \( j = m \), weight, score and loading vectors are obtained. The matrices \( t_j, p_j^T, q_j^T \), and \( \hat{u}_j \) are rank one component matrices that describe the underlying structure between the predictor and response variables, respectively.

Höskuldsson [9] showed that linear PLS:

(i) the cost function \( J_j \) describes the covariance between the \( j^{th}\) pair of score variables;

(ii) the vectors \( t_j \) are mutually orthogonal;

(iii) the matrix product \( P_j W \) is an upper triangular matrix; and

(iv) the vectors \( w_i \) are mutually orthonormal.

### 2.2 Nonlinear Partial Least Squares

Several modifications have been introduced to enhance the capability of PLS to model nonlinear relationships between the predictor and response variables. These modifications propose nonlinear relationships between the \( u \)-score and the \( t \)-score vector instead of the linear relationship of Equation (4).


The above extensions commonly represent the relationship between the \( j^{th}\) pair of \( t \)- and \( u \)-score vectors, \( t_j \) and \( u_j \), as follows:

\[
\begin{align*}
u_j &= f(t_j) + e_j,
\end{align*}
\]

(7)

with \( f(t_j) \) being a nonlinear function and \( e_j \) being the prediction residuals. From the above nonlinear PLS extensions, the algorithms by Wold et al. [19], Baffi et al. [1,2] and Li et al. [12] are closely related to the iterative determination of the score and weight vectors of the PLS algorithm. However, these algorithms utilise the following cost function for obtaining the weight and score vectors:

\[
J_j = \|u_j - \hat{u}_j\|_2 = \|u_j - f(t_j)\|_2,
\]

(8)

which is subject to the constraint of:

\[
v_j = Y_j^T \hat{u}_j / u_j^T u_j.
\]

(9)

These "error-based" algorithms require the cost function \( J_j \) to be continuous and differentiable with respect to the weight vector \( w_i \). This, however, does not constitute a maximum covariance criterion, as it is the case for linear PLS. Hence, although the properties (ii) to (iv) are still prevalent for the error-based algorithms, property (i) does not hold.

### 3 A New Covariance-Based Nonlinear PLS Algorithm

This section introduces a new covariance-based algorithm for determining the weight and score vectors. The error-based cost function of Equation (8) is replaced by a covariance-based one that is then maximised using the genetic optimisation method proposed by Sharma et al. [16].

The new covariance-based algorithm relies on a nonlinear transformation of \( t \)-score variables:

\[
\tau = f(t),
\]

(10)

where \( \tau \) represent a vector in which the transformed values of the \( t \)-score variables are stored. This transformation involves a set of linear parameters, e.g. a polynomial in \( t \) or a radial basis function neural network. These parameters can be determined using a least squares solution to predict the \( u \)-score vector:

\[
u_j = \tau_j + e_j = \tau_j + b_j + e_j,
\]

(11)

with \( \|e_j\|_2 = \|e_j\|_2 \) and \( b_j \) being a scalar factor. Note that Equation (11) is similar to Equation (3). Using Equation (11), the covariance-based cost function is then given by:

\[
J_j = \frac{1}{K-1} \|\tau_j - \hat{u}_j\|_2.
\]

(12)

**Remark 1:** The cost function in Equation (12), \( J_j \), is equivalent to \( J_j = b_j / K-1 \|\tau_j - \hat{u}_j\|_2 \).

Remark 1 follows from the fact that the error vector, \( e_j \), is orthogonal to \( \tau_j \), since the parameters of the nonlinear transformation are obtained using least squares solution.

The cost function, \( b_j / K-1 \|\tau_j - \hat{u}_j\|_2 \), is then maximised using a genetic strategy. More precisely, the weight vectors, \( v_i \) and \( w_i \), are the independent variables. Details about the genetic
optimisation technique employed in this work can be found in Sharma et al. [16].

Throughout this work, the functions $f(t)$ were based on radial basis function neural networks and the set of linear parameters determined the contribution of individual basis functions to the prediction of the $u$-score vectors.

4 Application Study

To contrast the utility of the new covariance-based NLPLS algorithm with linear PLS and the error-based NLPLS algorithm by Baffi et al. [2], results from an application study on a pH neutralisation process is presented in this section. This dynamic process is known to be highly non-linear and exhibit severe time varying behaviour. Consequently, they have been widely used as a benchmark for testing modelling and control algorithms [6,10,11]. In this work, the model and parameter values for the pH process described by Henson and Seborg [6] was used.

The process consists of a tank where strong acid (HNO$_3$) is neutralized by a strong base (NaOH) in the presence of a buffer stream (NaHCO$_3$). From this process, a data set of 1000 points was generated by randomly stepping the predictor variables $Q_1$ (acid flow rate), $Q_2$ (base flow rate), $Q_3$ (buffer flow rate) and $Q_4$ (outlet flow). The set-point for $Q_3$ was changed so that the recorded pH values covered the range 3 to 11, see Figure 1. The pH value of the outlet stream represented the response variable and was recorded under steady-state conditions.

The recorded data set was then split into two subsets, a training set containing 700 samples and a validation set of 300 samples.

4.1 Comparison Between Covariance Based PLS, Linear PLS and Error-Based PLS

The genetic strategy used 20 chromosomes with a crossover probability $p_c=0.65$ and a mutational probability $p_m=0.01$. The number of basis functions was initially set at 30. The best chromosome was the one that represented the largest covariance value for each LV after 5 runs of 1000 generations in 5 runs. This then

<table>
<thead>
<tr>
<th>LV</th>
<th>Linear PLS</th>
<th>Error-Based NLPLS</th>
<th>Covariance-Based NLPLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>24.8085</td>
<td>12.3714</td>
<td>26.7152</td>
</tr>
<tr>
<td>2</td>
<td>2.20425</td>
<td>0.434224</td>
<td>2.83418</td>
</tr>
<tr>
<td>3</td>
<td>0.091536</td>
<td>0.277897</td>
<td>0.0918694</td>
</tr>
<tr>
<td>4</td>
<td>1.0174 x10^-8</td>
<td>4.963x10^-10</td>
<td>1.3429x10^-8</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LV</th>
<th>Linear PLS</th>
<th>Error-based NLPLS</th>
<th>Covariance-Based NLPLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.9766</td>
<td>0.2687</td>
<td>0.3334</td>
</tr>
<tr>
<td>2</td>
<td>0.9052</td>
<td>0.2321</td>
<td>0.3168</td>
</tr>
<tr>
<td>3</td>
<td>0.8571</td>
<td>0.2174</td>
<td>0.2244</td>
</tr>
<tr>
<td>4</td>
<td>0.8553</td>
<td>0.1957</td>
<td>0.2060</td>
</tr>
</tbody>
</table>
returned the optimum parameters for the covariance-based nonlinear PLS model.

Using the training data, the results of applying linear PLS, the error-based NLPLS and the covariance-based NLPLS algorithms are summarised in Table 1. For the covariance based NLPLS case, the results show that the covariance between the first pair of score variables is around 10% larger than that of linear PLS and 100% larger than that of the error-based NLPLS algorithm.

Table 2 shows the residual variance, described by the mean-square prediction error (MSPE) for the validation data. Although error-based NLPLS provided a marginally smaller prediction error of the response variable, covariance-based NLPLS produced a significantly larger covariance.

6 Conclusions and Further Work

This paper showed that error-based NLPLS algorithms, which directly relate to the iterative linear PLS, should not be seen as nonlinear extensions to PLS. This is because they are maximising the accuracy of the score models rather than maximising the covariance between each pair of score variables, as it is the case for linear PLS.

In this work, a new covariance-based NLPLS algorithm was proposed that relies on maximising the covariance for each pair of score variables. This was achieved by introducing a nonlinear transformation of the score variable associated with the predictor variables. The transformed score variable is then used instead of the original score variable to establish a covariance criterion that can then be maximised.

The maximisation was carried out by applying a genetic strategy previously used to enhance an error-based NLPLS algorithm [16].

In an application study to a realistic simulation of a pH neutralisation process, it was shown that the new NLPLS algorithm outperforms linear PLS, while providing similar prediction accuracy for the pH value.

References


