A Probabilistic Derivation of the Partial Least-Squares Algorithm

Mats G. Gustafsson[†]

Signal and Systems Group, Uppsala University, P.O. Box 528, 751 20 Uppsala, Sweden

Received May 25, 2000

Traditionally the partial least-squares (PLS) algorithm, commonly used in chemistry for ill-conditioned multivariate linear regression, has been derived (motivated) and presented in terms of data matrices. In this work the PLS algorithm is derived probabilistically in terms of stochastic variables where sample estimates calculated using data matrices are employed at the end. The derivation, which offers a probabilistic motivation to each step of the PLS algorithm, is performed for the general multiresponse case and without reference to any latent variable model of the response variable and also without any so-called "inner relation". On the basis of the derivation, some theoretical issues of the PLS algorithm are briefly considered: the complexity of the original motivation of PLS regression which involves an "inner relation"; the original motivation behind the prediction stage of the PLS algorithm; the relationship between uncorrelated and orthogonal latent variables; the limited possibilities to make natural interpretations of the latent variables extracted.

1. INTRODUCTION

In chemistry (chemometrics), partial least-squares (PLS) regression is the standard method used for ill-conditioned linear multivariate modeling. Examples of important application areas of PLS are multivariate calibration, quantitative structure−property/structure−activity relationships (QSPR/QSAR), and chemical process modeling.[1]

Traditionally, PLS regression has been presented in terms of the nonlinear iteratively partial least-squares (NIPALS) algorithm, here presented in Appendix A. Since PLS regression has been frequently and successfully used in chemometrics for more than two decades, there has been a great interest in the theoretical aspects of PLS in general and the PLS algorithm in particular. Theoretical understanding of PLS is of importance not only for improvements of the PLS algorithm and comparisons with mathematically more well motivated alternatives to PLS but also when proposing generalizations of PLS to nonlinear multivariate regression problems like in refs 2−5. A noncomprehensive list of important contributions to an improved theoretical understanding of PLS regression and the PLS algorithm includes refs 1, 4, and 6−18. Together with many other closely related articles, these contributions are mainly expressed in terms of data matrices and very few consider the general multiresponse case.

This work presents a probabilistic derivation (motivation) of the standard NIPALS PLS algorithm with the following features: (i) The derivation results in a procedure which is identical to the steps taken in the standard NIPALS PLS algorithm when exact probabilistic quantities are replaced by standard sample estimates. (ii) The probabilistic derivation considers the general case with multiple response variables. (iii) Extraction of each latent variable (LV) in the NIPALS algorithm is motivated on the basis of a probabilistic linear LV model[19] of the input. The extraction of each LV is shown to involve a constrained optimization problem with a set of constraints on the model parameters that grows with the number of LVs extracted. (iv) The well-known deflation step performed after extraction of each LV in the PLS algorithm is introduced as an elegant way of eliminating the growing set of parameter constraints and obtain a standard problem which is efficiently solved by the NIPALS algorithm in each step. (iv) Instead of working with a data matrix **Z** where each row consists of a sample (measurement), one considers a vector-valued stochastic variable **z**, here defined as a column vector. Examples (realizations) of this stochastic variable are (after rewriting them as row vectors) the rows of the data matrix **Z**.

After presentation of the probabilistic derivation, some theoretical issues associated with the original PLS algorithm are considered. The main findings presented are the following: (i) The new derivation does not involve any "inner relation" like in the original motivation of the PLS algorithm. This finding is discussed and suggested to indicate that the original motivation of the PLS algorithm is more complex than necessary. (ii) The original idea behind the PLS algorithm to build an "inner relation" between LVs for the input and response variables is consistently carried out in the probabilistic setting presented. This analysis shows an unexpected discrepancy between the original PLS algorithm and the algorithm one would obtain if the original ideas had been followed consistently. This finding can be explained theoretically and indicates that the original idea behind the PLS algorithm is still partly unexplored. (iii) The old and interesting topic of how to make natural interpretations of the LVs extracted using the PLS algorithm is reconsidered in a probabilistic context. The main conclusion is that the LVs obtained are not designed to be statistically independent and are therefore hard to interpret, even when the regression model is very successful.

A probabilistic derivation of the PLS algorithm is attractive for at least three reasons. First, the derivations offers natural probabilistic motivations for each step of the original PLS

[†] Phone: +46-18-471 32 29. Fax: +46-18-55 50 96. E-mail: Mats.Gustafsson@signal.uu.se.

10.1021/ci0003909 CCC: $20.00
Published on Web 02/15/2001

algorithm which traditionally has been loosely motivated. Second, the derivation contributes to the theoretical understanding of PLS modeling in general and may immediately be used to identify alternatives and generalizations of PLS, even in the linear case. Third, a probabilistic derivation is of great educational value when PLS regression is introduced and compared with alternative methods in courses on multivariate analysis and chemometrics which are founded on probability theory (stochastic variables). When other regression methods such as principal component regression and ridge regression are presented (derived) in a probabilistic context, a similar consistent presentation of PLS regression is of course very useful. In fact, this was one of the original motivations behind the present work.

2. THE PROBLEM STATEMENT

Consider the problem of how to use training examples to establish a linear model

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}_y \tag{1}$$

where the $R \times P$ matrix $\mathbf{A} = [\mathbf{a}_1\ \mathbf{a}_2\ ...\ \mathbf{a}_P]$ with $R \times 1$ column vectors $\mathbf{a}_i$ contains the regression coefficients (parameters). The $R \times 1$ response vector $\mathbf{y} = (y_1, y_2, ..., y_R)^T$, the $P \times 1$ input vector $\mathbf{x} = (x_1, x_2, ..., x_P)^T$, and the $R \times 1$ residual error vector $\mathbf{e}_y$ are stochastic variables where T denotes the transpose.

In the following we always assume zero mean stochastic variables, $\mathbf{m}_x = E\{\mathbf{x}\} = \mathbf{0}$ and $\mathbf{m}_y = E\{\mathbf{y}\} = \mathbf{0}$, where E denotes the expectation operator. This is to reflect the conventional use of mean-centered variables in the context of PLS. Following the standard ordinary least-squares (OLS) approach, the estimate of the matrix $\mathbf{A}$ in (1) is selected as the matrix $\mathbf{A}_{\mathrm{OLS}}$ which minimizes the average sum of squared residual errors

$$V_N(\mathbf{A}) = \frac{1}{N}\sum_{n=1}^{N} ||\mathbf{y}(n) - \mathbf{A}\mathbf{x}(n)||^2 \tag{2}$$

where $(\mathbf{x}(n), \mathbf{y}(n))$, $n = 1, 2, ..., N$ are the N training examples available.

Unfortunately this estimate is unreliable when there are strong dependencies (collinearity) between the components of $\mathbf{x}$; see for example the review by Mandel.[20] In this case, the estimate $\mathbf{A}_{\mathrm{OLS}}$ varies dramatically from one specific training set to another. Different regularization methods to control the variance of the solution have been proposed; some of the most popular methods used are ridge regression, principal component regression (PCR), partial least-squares regression, and variable subset selection regression.[12]

In PCR the strong dependencies between the components of $\mathbf{x}$ are modeled by means of a linear LV model

$$\mathbf{x} = \mathbf{P}\mathbf{t} + \mathbf{e}_x \tag{3}$$

with H LVs where $\mathbf{t}$ is a $H \times 1$ LV vector, $\mathbf{P}$ is a $P \times H$ matrix, and $\mathbf{e}_x$ is the residual error. Assuming *(1) uncorrelated LVs* and *(2) orthogonal unit length column vectors* in $\mathbf{P}$, the LV model can be obtained by means of principal component analysis (PCA): The columns of $\mathbf{P}$ will be the eigenvectors of the covariance matrix $\mathbf{C}_{xx} = E\{(\mathbf{x} - \mathbf{m}_x)(\mathbf{x} - \mathbf{m}_x)^T\} = E\{\mathbf{x}\mathbf{x}^T\}$. Determination of $\mathbf{P}$ by means of PCA is the basis for PCR. Assuming a negligible error $\mathbf{e}_x$, substituting $\mathbf{x}$ in (3) into (1) yields

$$\mathbf{y} \approx \mathbf{A}\mathbf{P}\mathbf{t} + \mathbf{e}_y = \tilde{\mathbf{A}}\mathbf{t} + \mathbf{e}_y = \sum_{h=1}^{H} t_h\tilde{\mathbf{a}}_h + \mathbf{e}_y \tag{4}$$

where $\tilde{\mathbf{A}} = [\tilde{\mathbf{a}}_1\ \tilde{\mathbf{a}}_2\ ...\ \tilde{\mathbf{a}}_H] = \mathbf{A}\mathbf{P}$. Since the unit length columns of $\mathbf{P}$ are orthogonal, for a negligible error $\mathbf{e}_x$, the LVs can be determined as $\mathbf{t} \approx \mathbf{P}^T\mathbf{x}$. With the uncorrelated components of $\mathbf{t}$ so obtained, there are no variance problems estimating the parameters in $\tilde{\mathbf{A}}$ using the OLS criterion.

An apparent problem with PCR is that the LV model of $\mathbf{x}$ will not capture components of $\mathbf{x}$ with small variances which are strongly correlated with $\mathbf{y}$. One recent attempt to avoid this problem is by ranking the principal components by their correlation with the response variable.[21] Another attempt would be to modify the PCR method in such a way that the LV model of $\mathbf{x}$ is designed with regression (prediction of $\mathbf{y}$) in mind. *The problem considered in the next two sections below is how to find such a method, and the resulting sequential procedure is identical to the steps taken in the PLS algorithm. The only difference is that the steps taken are formally motivated in terms of stochastic variables and a series of subsequent constrained optimization problems. Thus this offers an alternative to the traditional heuristic motivation of the PLS algorithm presented e.g. by Geladi and Kowalski.*[6]

3. OBTAINING A PREDICTIVE LV MODEL

Consider the problem of building a LV model of $\mathbf{x}$ with LVs that are maximally dependent on $\mathbf{y}$. A natural measure of statistical dependence is the correlation coefficient between the LVs of $\mathbf{x}$ and a suitable linear projection $(\mathbf{c}^T\mathbf{y})$ of $\mathbf{y}$; see Appendix B. A more general information theoretic measure would be the mutual information[22] between the variables. A third reasonable measure of dependence considered in this work is the covariance between the variables. For prediction of the response variable $\mathbf{y}$, this criterion does not seem to have any obvious advantages in comparison with the correlation coefficient but it is used here to obtain the PLS algorithm. For a more detailed discussion about the potential advantage of maximizing the covariance instead of the correlation, see the analysis by Frank and Friedman.[12]

The statistical dependence between the LVs is another modeling issue. In PCR, the LVs created are uncorrelated but one may also imagine LV models where the components are selected to be as statistically independent as possible. Below, the LV model is created with uncorrelated (and thus not necessarily independent) components.

3.1. Obtaining the First Latent Variable. The first LV t_1 is obtained by maximizing the covariance $E\{t_1u_1\}$ where $t_1 = \mathbf{w}_1^T\mathbf{x}$ and $u_1 = \mathbf{c}_1^T\mathbf{y}$. Here $\mathbf{w}_1$ and $\mathbf{c}_1$ are unit length projection vectors that can be chosen freely in the maximization step. This may be interpreted as an attempt to model the input as $\mathbf{x} \approx t_1\mathbf{w}_1$, but below we show that a better choice is $\mathbf{x} \approx t_1\mathbf{p}_1$, where the base vector $\mathbf{p}_1$ is defined in (9). The projection vectors $\mathbf{w}_1$ and $\mathbf{c}_1$ are found by solving the optimization problem

$$\max_{||\mathbf{w}_1||=1,||\mathbf{c}_1||=1} (E\{t_1u_1\})^2 \tag{5}$$

which is expressed more explicitly as

$$\max_{||\mathbf{w}_1||=1,||\mathbf{c}_1||=1} (\mathbf{w}_1{}^T\mathbf{C}_{xy}\mathbf{c}_1)^2 \quad (6)$$

where $\mathbf{C}_{xy} = E\{\mathbf{x}\mathbf{y}^T\}$. How to find a closed form solution to this problem will be discussed in detail below.

3.2. Obtaining the Second Latent Variable. To obtain the second LV and basis vector $\mathbf{p}_2$, new projection vectors $\mathbf{w}_2$ and $\mathbf{c}_2$ are employed and the criterion to be minimized is the square of

$$E\{t_2u_2\} = E\{\mathbf{w}_2{}^T\mathbf{x}\mathbf{y}^T\mathbf{c}_2\} \quad (7)$$

under the constraints $||\mathbf{w}_2|| = 1$ and $||\mathbf{c}_2|| = 1$ together with the additional constraint that the components t_1 and t_2 are uncorrelated; i.e,

$$E\{t_1t_2\} = E\{\mathbf{w}_1{}^T\mathbf{x}\mathbf{x}^T\mathbf{w}_2\} = \mathbf{w}_1{}^T\mathbf{C}_{xx}\mathbf{w}_2 = 0 \quad (8)$$

The constraint in (8) can easily be eliminated from the optimization problem by modifying $\mathbf{x}$ to $\tilde{\mathbf{x}}_1 = \mathbf{x} - t_1\mathbf{p}_1$, where $\mathbf{p}_1$ is chosen such that $\tilde{\mathbf{x}}_1$ is uncorrelated with $t_1 = \mathbf{x}^T\mathbf{w}_1$; i.e., $E\{t_1\tilde{\mathbf{x}}_1\} = E\{t_1(\mathbf{x} - t_1\mathbf{p}_1)\} = \mathbf{0}$. This is achieved if $\mathbf{p}_1$ is selected as

$$\mathbf{p}_1 = \frac{E\{\mathbf{x}\mathbf{x}^T\}\mathbf{w}_1}{E\{t_1{}^2\}} = \frac{\mathbf{C}_{xx}\mathbf{w}_1}{\mathbf{w}_1{}^T\mathbf{C}_{xx}\mathbf{w}_1} \quad (9)$$

This means that if we use the approximation $\mathbf{x} \approx t_1\mathbf{p}_1 = t_1\mathbf{C}_{xx}\mathbf{w}_1/\mathbf{w}_1{}^T\mathbf{C}_{xx}\mathbf{w}_1$ instead of the approximation $\mathbf{x} \approx t_1\mathbf{w}_1$, the constraint in (8) is automatically satisfied.

Using the approximation $\mathbf{x} \approx t_1\mathbf{p}_1$, the original maximization criterion used to obtain the second component can thus be replaced by the criterion

$$(E\{t_2u_2\})^2 = (E\{\mathbf{w}_2{}^T\tilde{\mathbf{x}}_1\mathbf{y}^T\mathbf{c}_2\})^2 \quad (10)$$

together with the simple constraints $||\mathbf{w}_2|| = 1$ and $||\mathbf{c}_2|| = 1$. It is important to note that this optimization problem has exactly the same structure as the maximization problem associated with obtaining the first component in (6). Thus the benefit of introducing $\mathbf{p}_1$ and then deflating (reducing, contracting) $\mathbf{x}$ is a simpler optimization problem. This idea to eliminate the constraint that the components must be uncorrelated by construction is very elegant and its importance should not be underestimated as it eliminates the need for computationally intensive nonlinear programming methods developed for constrained optimization problems. Now the same efficient unconstrained numerical algorithms such as the NIPALS algorithm (Appendix A) can be used to obtain each component.

3.3. Obtaining the Third and Higher Order Latent Variables. Following the above idea to eliminate constraints on correlations by construction of a suitable basis vector $\mathbf{p}_1$, each new constraint can be eliminated by a similar construction. This yields the optimization problem

$$\max_{||\mathbf{w}_i||=1,||\mathbf{c}_i||=1} (E\{t_iu_i\})^2 \quad (11)$$

To obtain $\mathbf{p}_2$, consider $t_2 = \mathbf{w}_2{}^T\tilde{\mathbf{x}}_1$ and determine $\tilde{\mathbf{x}}_2 = \tilde{\mathbf{x}}_1 - t_2\mathbf{p}_2$, where $\mathbf{p}_2$ is selected to satisfy $E\{t_2\tilde{\mathbf{x}}_2\} = \mathbf{0}$ yielding

$$\mathbf{p}_2 = \frac{E\{t_2\tilde{\mathbf{x}}_1\}}{E\{t_2{}^2\}} = \frac{\mathbf{C}_{xx1}\mathbf{w}_2}{\mathbf{w}_2{}^T\mathbf{C}_{xx1}\mathbf{w}_2} \quad (12)$$

where $\mathbf{C}_{xx1} = E\{\tilde{\mathbf{x}}_1\tilde{\mathbf{x}}_1{}^T\}$ is the covariance matrix of the deflated variable $\tilde{\mathbf{x}}_1$. Repeating the above argument for a general LV, we may write t_i as

$$t_i = \mathbf{w}_i{}^T\tilde{\mathbf{x}}_{i-1} \quad (13)$$

where

$$\tilde{\mathbf{x}}_i = \tilde{\mathbf{x}}_{i-1} - t_i\mathbf{p}_i \quad i = 1, 2, ..., H \quad \mathbf{x}_0 = \mathbf{x} \quad (14)$$

and $\mathbf{p}_i$ as

$$\mathbf{p}_i = \frac{E\{t_i\tilde{\mathbf{x}}_{i-1}\}}{E\{t_i{}^2\}} = \frac{\mathbf{C}_{xx(i-1)}\mathbf{w}_i}{\mathbf{w}_i{}^T\mathbf{C}_{xx(i-1)}\mathbf{w}_i} \quad i = 1, 2, ..., H \quad (15)$$

where $\mathbf{C}_{xx(i-1)} = E\{\tilde{\mathbf{x}}_{i-1}\tilde{\mathbf{x}}_{i-1}{}^T\}$. One should note that it is only the input $\mathbf{x}$ which is deflated in each step and the response variable $\mathbf{y}$ is not deflated as in the original PLS algorithm (Appendix A). This is because one can show[7,10,23] that deflation of $\mathbf{y}$ does not alter the LV model of $\mathbf{x}$ created. This result may also be shown in terms of stochastic variables (not shown here).

3.4. Analytical Optimization To Find the First Latent Variable. The solution to (6) is a stationary point to the Lagrangian[24]

$$L(\mathbf{w},\mathbf{c}) = (\mathbf{w}^T\mathbf{C}_{xy}\mathbf{c})^2 - \lambda_w(||\mathbf{w}|| - 1) - \lambda_c(||\mathbf{c}|| - 1) \quad (16)$$

Differentiation yields the stationary conditions

$$(\mathrm{w}^T\mathbf{C}_{xy}\mathbf{c})\mathbf{C}_{xy}\mathbf{c} = \lambda_w\mathbf{w} \quad (17)$$

and

$$(\mathbf{w}^T\mathbf{C}_{xy}\mathbf{c})\mathbf{C}_{yx}\mathbf{w} = \lambda_c\mathbf{c} \quad (18)$$

which after multiplication with $\mathbf{w}^T$ and $\mathbf{c}^T$ immediately yield the same expression for λ_w and λ_c:

$$\lambda = \lambda_w = \lambda_c = (\mathrm{w}^T\mathbf{C}_{xy}\mathbf{c})^2 \quad (19)$$

After variable elimination, the stationary conditions can be expressed as two eigenvalue problems:

$$\mathbf{C}_{xy}\mathbf{C}_{yx}\mathbf{w} = \lambda\mathbf{w} \quad (20)$$

and

$$\mathbf{C}_{yx}\mathbf{C}_{xy}\mathbf{c} = \lambda\mathbf{c} \quad (21)$$

These expressions show that to maximize $\lambda = (\mathbf{w}^T\mathbf{C}_{xy}\mathbf{c})^2$ one should select the eigenvectors with the largest eigenvalue λ. The conclusion that $\mathbf{w}$ and $\mathbf{c}$ are the eigenvectors of the data matrix versions of $\mathbf{C}_{xy}\mathbf{C}_{yx}$ and $\mathbf{C}_{yx}\mathbf{C}_{xy}$, respectively, is also well-known from earlier nonprobabilistic studies[7,10] of the PLS algorithm.

The above procedure is presented only for the first LV. The same procedure of course can be used again to obtain

the other LVs provided that the matrices $\mathbf{C}_{xy}$ and $\mathbf{C}_{yx}$ are replaced with the appropriate matrices $\mathbf{C}_{xyi}$ and $\mathbf{C}_{yxi}$, $i = 1, 2, ..., H$.

3.5. Conclusion. One should note that the procedure described above to obtain the predictive LV model corresponds exactly to the sequential procedure of the original NIPALS; see Appendix A. The only difference is that each step is explained and motivated in a probabilistic setting. In particular, the deflation step where a component of $\mathbf{x}$ is removed is motivated as a way of avoiding constraints on the correlation between latent variables extracted.

4. OBTAINING A REGRESSION MODEL

After a LV model of $\mathbf{x}$ is obtained, a successful regression model should be created. As in PCR, see (4), the problem is how to determine the coefficients $\tilde{\mathbf{A}}$ in the model

$$\mathrm{y} = \tilde{\mathbf{A}}\,\mathbf{t} = \sum_{h=1}^{H} t_h \tilde{\mathbf{a}}_h \tag{22}$$

This is straightforward as demonstrated next. Assuming the model is true (valid), multiplying both sides in (22) with the LV t_i and applying the expectation operator yields

$$E\{\mathrm{y}t_i\} = E\{t_i^2\}\tilde{\mathbf{a}}_i \tag{23}$$

or

$$\tilde{\mathbf{a}}_i = \frac{E\{\mathbf{y}\tilde{\mathbf{x}}_{i-1}{}^T\mathbf{w}_i\}}{E\{t_i^2\}} = \frac{\mathbf{C}_{yx(i-1)}\mathbf{w}_i}{\mathbf{w}_i{}^T\mathbf{C}_{xx(i-1)}\mathbf{w}_i} \tag{24}$$

where $\mathbf{C}_{yx(i-1)} = E\{\mathbf{y}\tilde{\mathbf{x}}_{i-1}{}^T\}$. Now with employment of the result in (18) from the analytical optimization above, the expression for this vector can be simplified to

$$\tilde{\mathbf{a}}_i = \frac{\mathbf{w}_i{}^T\mathbf{C}_{xy(i-1)}\mathbf{c}_i}{\mathbf{w}_i{}^T\mathbf{C}_{xx(i-1)}\mathbf{w}_i}\mathbf{c}_i \tag{25}$$

In summary, with $C = [\mathbf{c}_1\ \mathbf{c}_2\ ...\ \mathbf{c}_H]$, we can express the model as

$$\mathrm{y} = \sum_{h=1}^{H} b_h t_h \mathbf{c}_h = \mathbf{CBt} \tag{26}$$

where

$$b_i = \frac{E\{t_i u_i\}}{E\{t_i^2\}} = \frac{\mathrm{w}_i{}^T\mathbf{C}_{xy(i-1)}\mathbf{c}_i}{\mathbf{w}_i{}^T\mathbf{C}_{xx(i-1)}\mathbf{w}_i} \tag{27}$$

and $\mathbf{B}$ is a diagonal matrix with $B_{ii} = b_i$. This result is equivalent to the prediction used in PLS modeling when the exact matrices $\mathbf{C}_{xx0} = \mathbf{C}_{xx} = E\{\mathbf{x}\mathbf{x}^T\}$ and $\mathbf{C}_{xxh} = E\{\tilde{\mathbf{x}}_h\tilde{\mathbf{x}}_h{}^T\}$, $h = 1, 2, ..., H$, are replaced by standard estimates; see also the next section.

5. EQUIVALENCE WITH THE PLS ALGORITHM

Above we have presented a sequential procedure, on the basis of the probabilistic models, of how to obtain a predictive LV model $\mathbf{x} = P\mathbf{t}$ of the input $\mathbf{x}$ and then how it can be used to yield a model of the response vector $\mathbf{y}$ as $\mathbf{y} = CB\mathbf{t}$.

In practice a set of N training examples $(\mathbf{x}_n,\mathbf{y}_n)$, $n = 1, 2, ..., N$, must be used to find an estimate of the covariance matrix $\mathbf{C}_{xx}$ and other similar quantities considered above. This might suggest that the sequential procedure presented above on the basis of probability theory does not work well in practice, but it is actually equivalent to the PLS algorithm in Appendix A when exact quantities are replaced by standard estimates. Let the training examples be collected in the sample data matrices

$$\mathbf{X}_N = (\mathbf{x}(1), \mathbf{x}(2), ..., \mathbf{x}(N))^T$$

$$\mathbf{Y}_N = (\mathbf{y}(1), \mathbf{y}(2), ..., \mathbf{y}(N))^T$$

Note that each *row* corresponds to a training example in the two matrices. A pair of corresponding rows in X_N and Y_N corresponds to one training example. Then an unbiased estimate of e.g. the matrices $\mathbf{C}_{xy}$ and $\mathbf{C}_{xx}$ are $\mathbf{X}^T\mathbf{Y}/(N-1)$ and $\mathbf{X}^T\mathbf{X}/(N-1)$, respectively. The normalization factor $N-1$ is ignored (not required) in the PLS algorithm.

6. THEORETICAL INSIGHTS

The probabilistic motivation of PLS above is valuable for theoretical studies of PLS and for design of new algorithms. In this section only a few minor theoretical insights are presented; new algorithms and other theoretical results will be presented in future work. The first two insights are related to the "inner model" which is a central idea in the context of the PLS algorithm. The third insight offers a simple theoretical connection between the original idea behind PLS to create orthogonal LV data vectors and the probabilistic concept of being uncorrelated. The fourth insight is related to the difference between uncorrelated and independent stochastic variables and the limited possibility to use PLS to recover LVs which are truly independent and not only uncorrelated.

6.1. Connections to the "Inner Relation". One of the basic ideas behind the original PLS algorithm was to obtain a LV model $\mathbf{x} = P\mathbf{t} + \mathbf{e}_x$ of the input and a LV model $\mathbf{y} = Q\mathbf{u} + \mathbf{e}_y$ of the output and then create an a set of scalar "inner relations", $u_i \approx b_i t_i$, for prediction of the response LVs in $\mathbf{u}$ from the input LVs $\mathbf{t}$.[6,25] This idea is reflected in some versions of the PLS algorithm where the columns of Q, the "Y-loadings", are computed but never used; see Appendix A and one of the most cited references.[10] The idea is also discussed in the context of nonlinear PLS.[2-4]

If this original idea would have been followed consistently, then $\mathbf{y}$ would have been deflated as $\tilde{\mathbf{y}}_i = \mathbf{y}_{i-1} - u_i\mathbf{q}_i$, $(\mathbf{y}_0 = \mathbf{y})$ with

$$\mathbf{q}_i = \frac{E\{u_i\tilde{\mathbf{y}}_{i-1}\}}{E\{u_i^2\}} = \frac{\mathbf{C}_{yy(i-1)}\mathbf{c}_i}{\mathbf{c}_i{}^T\mathbf{C}_{yy(i-1)}\mathbf{c}_i} \quad i = 1, 2, ..., H \tag{28}$$

where $\mathbf{C}_{yy(i-1)} = E\{\mathbf{y}_{i-1}\mathbf{y}_{i-1}{}^T\}$. This would guarantee uncorrelated components u_i, but it would not affect the construction of the LV model of $\mathbf{x}$ and the selection of $\mathbf{c}_i$ since deflation is not needed. Moreover, determination of the coefficient b_i

in the "inner relation" $u_i = b_i t_i + e_i$ yields (using the least-squares criterion) $b_i = E\{t_i u_i\}/E\{t_i^2\}$ as in (27).

Given the coefficients b_i and the LVs t_i, the straightforward natural procedure would be to use the approximation $u_i \approx b_i t_i$ and make the prediction of $\mathbf{y}$ as

$$\hat{\mathbf{y}} = \sum_{h=1}^{H} b_h t_h \mathbf{q}_h \tag{29}$$

where $\mathbf{q}_h$ is defined in (28). However, one should note that in PLS regression, the prediction in (26) is used instead where $\mathbf{c}_h$ replaces $\mathbf{q}_h$. The reason for using this alternative prediction measure in PLS regression is not easily found in the literature and is not obvious if one consistently follows the original idea of an "inner relation" between LVs as described here.

The PLS prediction may be obtained by employing the "inner relation" $u_i = b_i t_i$ to approximate not only the LVs for the response variable but also for computing (approximating) the loading vector q_i. Using the approximation $u_i \approx b_i t_i$, we have

$$\mathbf{q}_i \approx \frac{E\{b_i t_i \mathbf{y}\}}{E\{(b_i t_i)^2\}} = \frac{\mathbf{C}_{yx(i-1)}\mathbf{w}_i}{b_i \mathbf{w}_i^T \mathbf{C}_{xx(i-1)}\mathbf{w}_i} = \frac{1}{b_i}\tilde{\mathbf{a}}_i = \mathbf{c}_i \tag{30}$$

which results in the same prediction as in (26). Thus, this analysis shows that we also obtain the prediction in (26) when we let the inner model affect not only the prediction of the latent variable u_i but also the selection of the loading vector $\mathbf{q}_i$.

Finally, one should note that although the probabilistic derivation presented above does not involve any LV model of $\mathbf{y}$ and no "inner relation", it still yields the PLS algorithm. *In conclusion, the original approach to PLS seems unnecessary complex and the original idea behind PLS to obtain a regression algorithm with true "inner relations" between latent variables still seems quite unexplored.*

6.2. Orthogonal or Uncorrelated LVs. Traditionally, the deflation step in the PLS algorithm has been motivated by statistically unclear geometrical arguments related to the data matrices. The goal has been to obtain orthogonal LVs (or more precisely orthogonal LV data vectors).

This original idea to have orthogonal LVs is now explained in terms of stochastic variables. If the stochastic LVs t_i and t_j are uncorrelated, then the two data vectors $\mathbf{t}_i$ and $\mathbf{t}_j$ which contain realizations of t_i and t_j, respectively, will be approximately orthogonal, $\mathbf{t}_i^T\mathbf{t}_j \approx 0$. For infinitely long data long vectors, they will be exactly orthogonal. Thus, the constraint used in the context of PLS that the data vectors $\mathbf{t}_i$ and $\mathbf{t}_j$ should be orthogonal is not formally valid for short data vectors.

In practice, however, if two uncorrelated stochastic variables t_i and t_j are created as linear projections $t_i = \mathbf{w}_i^T\mathbf{x}$ and $t_j = \mathbf{w}_j^T\mathbf{x}$ of a multidimensional stochastic variable $\mathbf{x}$, then the probabilistic condition $E\{t_i t_j\} = \mathbf{w}_i^T E\{\mathbf{x}\mathbf{x}^T\}\mathbf{w}_j = \mathbf{w}_i^T C_{xx}\mathbf{w}_j = 0$ on the projection vectors $\mathbf{w}_i$ and $\mathbf{w}_j$ is replaced by an empirical condition using a standard estimate $\hat{C}_{xx}$ of the covariance matrix C_{xx}. Using N samples the empirical condition to be satisfied is $0 = \mathbf{w}_i^T\hat{C}_{xx}\mathbf{w}_j = \mathbf{w}_i^T(1/N)\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^T\mathbf{w}_j = \mathbf{t}_i^T\mathbf{t}_j/N$. One should note that the quantities in the last expression are exact and the data vectors should actually be exactly orthogonal. The approximation is hidden in the empirical estimate of the covariance matrix.

In conclusion, the goal to obtain orthogonal LV data vectors in the PLS algorithm can be motivated in a probabilistic setting. Although trivial, this is a significant finding as it shows how the PLS algorithm creates (approximately) uncorrelated LVs implicitly by means of geometric conditions. Moreover, this finding is useful when discussing generalizations of the PLS algorithm e.g. to obtain independent LVs.

6.3. PLS Yields Statistically Dependent LVs. The practical (chemical) interpretation of the latent variables (LVs) in a PLS model have been subject to many discussions over a long time period; see e.g refs 6, 13, and 26. A key issue here is the uniqueness of the LVs obtained. As pointed out by Frank and Friedman,[13] if PCR and PLS regression result in different LV models, which one should be considered as the right (true) model?

From the probabilistic view presented in the present paper, it is apparent that the PLS algorithm is designed to build a model $\mathbf{x} = P\mathbf{t}$ of the input with uncorrelated LVs t_i and therefore that the algorithm ignores higher than second-order dependencies (correlations) between the LVs. Construction of models with uncorrelated LVs is also the basic idea behind PCR and traditional algorithms for factor analysis.[19,27] From the recent theory of independent component analysis (ICA)[28,29] we know that it is not, in general, possible to obtain unique, (linear) factor models with statistically independent LVs using only second order dependencies. *Therefore, although the PLS algorithm is successful for ill-conditioned regression, one should not expect the LVs obtained to be statistically independent and thus one should not expect them always to have any unique and natural (chemical) interpretation.*

7. CONCLUSION

A probabilistic derivation of the PLS algorithm has been presented in terms of stochastic variables which immediately offers some interesting insights about LV modeling for ill-conditioned regression in general and PLS modeling in particular. The derivation is based on the intuitive idea to find a LV model of the input stochastic variable $\mathbf{x}$ which is more suitable for ill-conditioned regression (prediction of the response $\mathbf{y}$) than the one used in principal component regression. The derivation offers natural probabilistic motivations to each step in the PLS algorithm and offers a presentation of PLS suitable for courses on multivariate regression and chemometrics based on probability theory (stochastic variables).

The derivation and the associated probabilistic framework is shown to offer interesting theoretical insights about PLS and LV modeling. One is based on the finding that the derivation of the PLS algorithm can be performed without reference to any LV model for the response $\mathbf{y}$ or any "inner relation" between LVs. This indicates that the original idea behind PLS[6,25] was unnecessarily complex or that it was not consistently followed in the practical implementation of the PLS algorithm using the NIPALS algorithm. The derivation presented here reveals how the original idea of an "inner relation" between latent variables is related to the actual prediction computed by the PLS algorithm. The interpretation that the original idea behind the PLS algorithm is more

complex than the actual implementation has been indicated before[7,30] where the single response PLS algorithm is found to correspond to standard methods for inverting matrices or solving systems of linear equations. Another interesting theoretical insight is based on the observation that PLS (as well as principal component analysis and factor analysis) are designed to build LVs with uncorrelated rather than independent stochastic variables. On the basis of the recent theories of independent component analysis,[28,29] one may conclude that the LVs obtained in a PLS model are generally not statistically independent and therefore not easily interpreted in general.

Many of the results presented here are likely to be known to some of the experts in the field, but the explicit formulations in terms of stochastic variables presented here have, to our best knowledge, not been published before. The main point here is not the technical details but the observation that the PLS algorithm can be derived (explained) in terms of stochastic variables and that such a derivation may offer significant contributions to the theoretical understanding of the PLS algorithm.

ACKNOWLEDGMENT

The author acknowledges the valuable comments and literature references received from three anonymous reviewers which were very helpful when preparing the final version of the manuscript. Many thanks also to Johan Andersson, department of cell and molecular biology, for valuable comments on earlier drafts.

APPENDIX A: THE CLASSICAL PLS ALGORITHM

As presented very elegantly e.g. by Geladi and Kowalski,[6] the PLS algorithm may be interpreted as an ad hoc algorithm developed by combining two copies of the nonlinear iterative partial least-squares (NIPALS) algorithm for principal component analysis (PCA). Following Geladi and Kowalski,[6] if the data matrix X_N (defined in the main text) is denoted X the NIPALS algorithm can be summarized as follows:

The NIPALS Algorithm for PCA. Given a data vector with examples as rows, use the following procedure: (1) Take a column $\mathbf{x}_j$ from X and call it $\mathbf{t}_h$: $\mathbf{t}_h = \mathbf{x}_j$.(2) Calculate $\mathbf{p}_h{}^T = \mathbf{t}_h{}^T X/\|\mathbf{t}_h\|^2$. (3) Normalize $\mathbf{p}_h$ to length 1: $\mathbf{p}_{h(new)} = \mathbf{p}_h/\|\mathbf{p}_h\|$. (4) Calculate $\mathbf{t}_h$: $\mathbf{t}_h = X\mathbf{p}_h/\|\mathbf{p}_h\|^2$. (5) Compare the $\mathbf{t}_h$ used in step 2 with that obtained in step 4. If they are the same, stop (the iteration has converged). If they still differ, go to step 2. (6) Compute the residual $E_h = X - \mathbf{t}_h\mathbf{p}_h{}^T$ and replace X with E_h; $X = E_h$. Return to step 1 unless E_h is sufficiently small.

After M complete NIPALS loops one has obtained a decomposition of X as $X = \mathrm{t}_1\mathbf{p}_1{}^T + \mathbf{t}_2\mathbf{p}_2{}^T + ... + \mathbf{t}_M\mathbf{p}_M{}^T$. One can show that $\mathbf{t}_h$, the so-called score vector, is an eigenvector of XX^T. Similarly one can show that that the so-called loading vector $\mathbf{p}_h$ is an eigenvector of X^TX. Since X^TX and XX^T are symmetric matrices the eigenvectors are orthogonal.

In a linear regression problem where the components are strongly dependent, instead of directly finding a linear relationship between the input $\mathbf{x}$ and output $\mathbf{y}$, a simple and natural idea is to first reduce, if possible, the dimensionality of both signals (patterns) and then find an "inner relation" between the reduced signals. If the components in $\mathbf{x}$ (and $\mathbf{y}$) are strongly correlated, it should be possible to reduce their dimensionality significantly. When $\mathbf{y}$ is one-dimensional and thus represented by the scalar y, this approach is known as principal component regression PCR and is known to be dangerous due to the truncation performed. Since the dimensionality reduction of $\mathbf{x}$ is performed without knowledge about which projections are important for predicting y, it is possible that the reduced vector only contains large variance noise without any information about y.

In this context PLS might be interpreted as a new algorithm where the two separate PCA are integrated by *exchanging their scores*.[6,25] This results it the well-known algorithmic version of PLS. Mainly following Höskuldsson,[10] below the PLS algorithm is presented in detail.

The NIPALS Algorithm for PLS. Here the classical PLS algorithm for multiresponse variable models ($R > 1$) is presented. Given N training examples collected in the matrices X_N and Y_N, respectively, the vectors $\mathbf{w}_k$, $\mathbf{u}_k$, $\mathbf{c}_k$, and b_k are computed as follows:

Step 0: Initialization
Let $j = 0$, $k = 1$.
Let $D_0 = X_N$.
Let $E_0 = Y_N$.
Select an initial vector u_0 as one of the columns of Y_N.
Normalize, $\|\mathbf{u}_0\| = 1$.

The X-Block
Step 1: Let $\mathbf{w}_k(j + 1) = D_{k-1}{}^T\mathbf{u}_k(j)$.
Step 2: Normalize $\mathbf{w}_k(j + 1)$, $\|\mathbf{w}_k(j + 1)\| = 1$.
Step 3: Let $\mathbf{t}_k(j + 1) = D_{k-1}\mathbf{w}_k(j)$.

The Y-Block
Step 4: Let $\mathbf{c}_k(j + 1) = E_{k-1}{}^T\mathbf{t}_k(j)/\|\mathbf{t}_k(j)\|$.
Step 5: Normalize $\mathrm{c}_k(j + 1)$, $\|\mathbf{c}_k(j + 1)\| = 1$.
Step 6: $\mathbf{u}_k(j + 1) = E_{k-1}\mathbf{c}_k(j)$.
Step 7: $j = j + 1$. Go to step 1 if the vectors have *not* converged

Final Results
Step 8: $\mathbf{p}_k = D_{k-1}{}^T\mathbf{t}_k/\|\mathbf{t}_k\|^2 = D_{k-1}{}^TD_{k-1}\mathbf{w}_k(j)/\mathbf{w}_k(j)^TD^T{}_{k-1}D_{k-1}\mathbf{w}_k(j)$ ("X-loadings").
Step 9: $\mathbf{q}_k = E^Tk{-}1\mathbf{u}_k/\|\mathbf{u}_k\|^2 = E^T{}_{k-1}E_{k-1}\mathbf{c}_k(j)/\mathbf{c}_k(j)^TE^Tk{-}1\mathbf{c}_k(j)$ ("Y-loadings").
Step 10: $b_k = \mathbf{u}_k{}^T\mathbf{t}_k/\|\mathbf{t}_k\|^2$.
Step 11: $D_k = D_{k-1} - \mathbf{t}_k\mathbf{p}_k{}^T$.
Step 12: $E_k = E_{k-1} - b_k\mathbf{t}_k\mathbf{c}_k{}^T$.
Step 13: $k = k + 1$. Return to step 1 unless D_{k-1} is approximately zero.

One should note that the vectors $\{\mathbf{q}_k\}$ are not used in the final regression, but they are included for completeness. When the response variable is one-dimensional, $R = 1$, the vectors $\mathbf{c}_k$ are reduced to scalars. This means that steps 4−7 can be omitted ($\mathbf{u}_k = \mathbf{u}_0 = \mathbf{y}_N$ and $c = 1$).

The main outputs of the algorithm constitute a decomposition of the input matrix X, $X = PT^T + E_X$, a decomposition of the output matrix $Y = QU^T + E_Y$, and a linear "inner relation" $\mathbf{u}_h = b_h\mathbf{t}_h$ where $\mathbf{u}_h$ and $\mathbf{t}_h$ are columns of the matrices U and T, respectively. Another output is the projection matrix W used to compute P.

Prediction using a new sample $\mathbf{x}$ is performed by decomposing $\mathbf{x}$ in the same way as X: First, let $E_0 = \mathbf{x}^T$, and

then $\hat{t}_1 = E_0\mathbf{w}_1$. Now repeating the steps $E_h = E_{h-1} - \hat{t}_h\mathbf{p}_h{}^T$ and $E_{h-1}\mathbf{w}_h$ yields a sequence of values $\{t_h\}$ which can be used for the prediction of $\mathbf{y}$ as $\hat{\mathbf{y}} = \Sigma_h b_h \hat{t}_h \mathbf{c}_h$.

Ignoring the normalization in the above algorithm, one can show that all the coefficients b_i will be equal to one, $b_i = 1$, and may therefore be ignored.[10]

APPENDIX B: DIFFERENCE BETWEEN CORRELATION AND COVARIANCE

There is a significant difference between a large correlation and a large covariance. The correlation between two zero mean stochastic variables a and b is defined as

$$\rho_{ab} \equiv \frac{E\{ab\}}{(E\{a^2\})^{1/2}(E\{b^2\})^{1/2}} \tag{31}$$

whereas the covariance is defined as

$$\mathrm{cov}_{ab} \equiv E\{ab\} = \rho_{ab}(E\{a^2\})^{1/2}(E\{b^2\})^{1/2} \tag{32}$$

The correlation contains a normalization such that ρ_{ab} is always restricted to the interval $[-1,1]$ whereas the covariance cov_{ab} may take any value.

APPENDIX C: UNCORRELATED BUT NOT INDEPENDENT

For a long time there have been very few attempts to explore the difference between uncorrelated and independent stochastic variables in practical contexts such as ill-posed multivariate regression. With the recent work on independent component analysis and blind separation of independent sources,[28,29,31] the practical importance between uncorrelated and independent variables has become evident. Two stochastic variables x and y with joint probability density function $f_{xy}(x,y)$ are said to be statistically independent when

$$f_{xy}(x,y) = f_x(x)f_y(y) \tag{33}$$

where $f_x(x)$ and $f_y(y)$ are the marginal probability density distributions. Assuming zero mean variables, the same variables are said to be uncorrelated when

$$E\{xy\} = 0 \tag{34}$$

For Gaussian stochastic variables it is well-known that being independent and uncorrelated are equivalent. However, for non-Gaussian variables this is not true in general. Most examples of variables which are uncorrelated but still dependent found in textbooks are artificially created. For example the variables $x = \xi$ and $y = \xi^2 - 1$ where ξ is a zero mean unit variance Gaussian stochastic variable are uncorrelated but obviously dependent on the same (latent) variable ξ.

REFERENCES AND NOTES

(1) Wold, S. Discussion: PLS in chemical practice. *Technometrics* **1993**, *35*, 136−139.
(2) Wold, S.; Kettaneh-Wold, N.; Skagerberg, B. Nonlinear PLS modeling. *Chemom. Intell. Lab. Syst.* **1989**, *7*, 53−65.
(3) Qin, S.; McAvoy, T. Nonlinear PLS modeling using neural networks. *Comput. Chem.l Eng.* **1992**, *16*, 379−391.
(4) Malthouse, E.; Tamhane, A.; Mah, R. Nonlinear partial least squares. *Comput. Chem. Eng.* **1997**, *12* (8), 875−890.
(5) Baffi, G.; Martin, E.; Morris, A. Nonlinear projection to latent structures revisited: the quadratic pls algorithm. *Comput. Chem. Eng.* **1999**, *23*, 395−411.
(6) Geladi, P.; Kowalski, B., Partial least-squares regression: A tutorial. *Anal. Chim. Acta* **1986**, *185*, 1−17.
(7) Manne, R. Analysis of two partial-least-squares algorithms for multivariate calibration. *Chemom. Intell. Lab. Syst.* **1987**, *2*, 283−290.
(8) Lorber, A.; Wangen, L.; Kowalski, B. A theoretical foundation for the pls algorithm. *J. Chemom.* **1987**, *1*, 19−31.
(9) Helland, I. On the structure of partial least squares regression. *Commun. Stat.−Simul. Comput.* **1988**, *17*, 581−607.
(10) Höskuldsson, A. PLS regression methods. *J. Chemom.* **1988**, *2*, 211−228.
(11) Stone, M.; Brooks, R. Continuum regression: Cross-validated sequentially constructed prediction embracing ordinary least squares, partial least squares and principal component regression. *J. R. Stat. Soc.* **1990**, *B52* (2), 237−269.
(12) Frank, I.; Friedman, J. Statistical View of Chemometrics Regression Tools. *Technometrics* **1993**, *35*, 109−135.
(13) Frank, I.; Friedman, J. Response. *Technometrics* **1993**, *35*, 143−148.
(14) Wold, S.; Johansson, E.; Cocchi, M. PLS−partial least squares projections to latent structures. In *3D QSAR in Drug Design*; Kubinyi, H., Ed.; ESCOM: Leiden, The Netherlands, 1993; pp 523−550.
(15) de Jong, S. SIMPLS: an alternative approach to partial least squares regression. *Chemom. Intell. Lab. Syst.* **1993**, *18*, 251−263.
(16) Höskuldsson, A. A combined theory for pca and pls. *J. Chemom.* **1995**, *9*, 91−123.
(17) Burnham, A.; Viveros, R.; MacGregor, J. Frameworks for latent variable multivariable regression. *J.Chemom.* **1996**, *10*, 31−45.
(18) Stoica, P.; Söderström, T. Partial least squares: A first-order analysis. *Scand. J. Stat.* **1998**, *25*, 17−24.
(19) Everitt, B. *An Introduction to Latent Variable Models*; Chapman and Hall: London, 1984.
(20) Mandel, J. Use of singular value decomposition in regression analysis. *Am. Stat.* **1982**, *36* (1), 15−24.
(21) Egan, W.; Brewer, W.; Morgan, S. Measurement of carboxyhemoglobin in forensic blood samples using uv/vis spectrometry and improved principal component regression. *Appl. Spectrosc.* **1999**, *53* (2), 218−225.
(22) Cover, T. *Elements on Information Theory*; John Wiley & Sons: New York, 1991.
(23) Dayal, B.; MacGregor, J. Improved pls algorithms. *J. Chemom.* **1997**, *11*, 73−58.
(24) Fletcher, R. *Practical Methods of Optimization*; Wiley: Chichester, U.K., 1987.
(25) Wold, S.; Martens, H.; Wold, H. The multivariate calibration problem in chemistry solved by the PLS method. In *Proceedings of the Conference on Matrix Pencils;* Lecture Notes in Mathematics Vol. 973; Springer-Verlag: Berlin, 1983; pp 286−293.
(26) Kvalheim, O. The latent variable. *Chemom.Intell. Lab. Syst.* **1992**, *14*, 1−3.
(27) Johnson, R.; Wichern, D. *Applied Multivariate Statistical Analysis*; Prentice Hall: Upper Saddle River, NJ, 1998.
(28) Bell, A.; Sejnowski, T. An information-maximization approach to blind separation and blind deconvolution. *Neural Comput.* **1995**, *7*, 1129−1159.
(29) Attias, H. Independent factor analysis. *Neural Comput.* **1998**, *11*, 803−851.
(30) Wold, S.; Ruhe, A.; Wold, H.; Dunn, W. The collinearity problem in linear regression. The partial least squares (PLS) approach to generalized inverses. *SIAM J. Sci.Stat. Comput.* **1984**, *5*, 735−743.
(31) Jutten, C.; Herault, J. Blind separation of sources 1. an adaptive algorithm based on a neuromimetic architecture. *Signal Process.* **1991**, *24* (1), 1−10.

CI0003909