JOURNAL OF CHEMOMETRICS, VOL. 12, 77–81 (1998)

SHORT COMMUNICATION
REGRESSION COEFFICIENTS IN MULTILINEAR PLS

SIJMEN DE JONG*
Unilever Research Laboratorium, PO Box 114, NL-3130 A C Vlaardingen, Netherlands

SUMMARY

Three alternative approaches are discussed for finding the final calibration model (regression coefficients) in PLS regression of *k*-way $\underline{\mathbf{Y}}$ on *N*-way $\underline{\mathbf{X}}$. The simplest approach is to skip the deflation of the **X**-data. From the observation that the specific deflation used in multiway PLS is inconsequential, it also follows that Bro's tri-PLS is equivalent to Ståhle's linear three-way decomposition (LTD). © 1997 John Wiley & Sons, Ltd.

J. Chemometrics, Vol. **12,** 77–81 (1998)

KEY WORDS multilinear PLS; multiway calibration; CoMFA; LTD

1. INTRODUCTION

Recently, Bro[1] introduced multilinear PLS (NPLS), generalizing PLS regression of *k*-way **Y** (PLS*k*, *k*=1 or 2) on two-way **X** to regression of *k*-way $\underline{\mathbf{Y}}$ (*k*=1, 2, . . .) on *N*-way $\underline{\mathbf{X}}$ (*N*=2, 3, . . .). Bro showed how to estimate the weights defining the model and Smilde[2] gave a closed expression for computing regression coefficients from these. This communication describes alternative ways for the latter. We will adopt the notation in References 1 and 2.

The results below are given for a single (univariate) response vector **y**. Implicitly, it also covers the case of multivariate **Y**, treating the columns of the two-way matrix **Y** (which may be the unfolded two-way equivalent of a higher-way array $\underline{\mathbf{Y}}$) separately. This is allowed since ordinary least squares (OLS) regression of multivariate **Y** on PLS components **T** is tantamount to the collection of all univariate OLS regressions. Thus there is no essential difference between univariate **y** and two-way **Y** (or higher-way $\underline{\mathbf{Y}}$) when it comes to computing the regression coefficient vector(s), given a set of weight and loading vectors. Of course, for the computation of weights and loadings it does matter whether the response is a one-way, two-way or higher-way array.[1]

2. THREE WAYS OF OBTAINING THE REGRESSION COEFFICIENTS

2.1. Method 1

Smilde[2] effectively transforms weights **W** ($P\times A$), the *a*th column applying to a corresponding (unfolded) residual matrix $\mathbf{X}^{(a-1)}$, into weights **W*** ($P\times A$), all columns now applying to the original (unfolded) $\mathbf{X}^{(0)}=\mathbf{X}$ ($I\times P$):

$$\mathbf{W}^*=[\mathbf{w}_1|(\mathbf{I}_P-\mathbf{w}_1\mathbf{w}_1^{\mathrm{T}})\mathbf{w}_2|\ldots|(\mathbf{I}_P-\mathbf{w}_1\mathbf{w}_1^{\mathrm{T}})(\mathbf{I}_P-\mathbf{w}_2\mathbf{w}_2^{\mathrm{T}})\ldots(\mathbf{I}_P-\mathbf{w}_{A-1}\mathbf{w}_{A-1}^{\mathrm{T}})\mathbf{w}_A] \quad (1)$$

* Correspondence to: S. de Jong, Unilever Research Laboratorium, PO Box 114, NL-3130 AC Vlaardingen, Netherlands

CCC 0886–9383/98/010077–05 $17.50

Received April 1997
Accepted June 1997

Having obtained $\mathbf{W^*}$ allows us to express the scores in $\mathbf{T}$ $(I \times A)$ directly in terms of the X-columns:

$$\mathbf{T} = \mathbf{XW^*} \tag{2}$$

Regressing $\mathbf{y}$ $(I \times 1)$ on the component scores $\mathbf{T}$ gives

$$\hat{\mathbf{y}} = \mathbf{Tb} \tag{3}$$

with

$$\mathbf{b} = (\mathbf{T}^{\mathrm{T}}\mathbf{T})^{-1}\mathbf{T}^{\mathrm{T}}\mathbf{y} \tag{4}$$

Combining (2) and (3) yields

$$\hat{\mathbf{y}} = \mathbf{XW^*b} \tag{5}$$

Hence the regression coefficients $\mathbf{b}_{\mathrm{NPLS}}$ $(P \times 1)$ needed to predict $\mathbf{y}$ from (future) $\mathbf{X}$ are obtained as

$$\mathbf{b}_{\mathrm{NPLS}} = \mathbf{W^*b} \tag{6}$$

Reference 3 (Appendix 1) provides efficient Matlab code for obtaining $\mathbf{b}_{\mathrm{NPLS}}$ in multilinear PLS, given weight vectors $\mathbf{W}$ and $\mathbf{b}$ $(A \times 1)$ and using equations (1) and (6). The algorithm avoids the construction and multiplication of the large projection matrices $\mathbf{I}_P - \mathbf{w}_a\mathbf{w}_a^{\mathrm{T}}$ $(P \times P)$ that occur in equation (1). For example, the second column of $\mathbf{W^*}$ is calculated as $\mathbf{w}_2 - (\mathbf{w}_1^{\mathrm{T}}\mathbf{w}_2)\mathbf{w}_1$. An alternative approach to calculating $\mathbf{b}_{\mathrm{NPLS}}$ that avoids the computation of $\mathbf{W^*}$ from $\mathbf{W}$ is Method 2.

2.2. Method 2

We first make a digression to the general topic of subspace-based regression. In subspace-based regression one replaces the regression of univariate $\mathbf{y}$ on two-way $\mathbf{X}$ by OLS regression of $\mathbf{y}$ on $\mathbf{T}$, the collection of a few (A<rank($\mathbf{X}$)) linear combinations $\mathbf{t}_a$ of $\mathbf{X}$:

$$\mathbf{T} = \mathbf{XV} \tag{7}$$

The weights in non-singular $\mathbf{V}$ $(P \times A)$ depend on the particular regression method (e.g. principal components regression (METHOD = PCR), partial least squares regression (METHOD = PLS), variable subset selection (METHOD = VSS)). Combining (3) and (7) gives

$$\hat{\mathbf{y}} = \mathbf{XVb} = \mathbf{Xb}_{\mathrm{METHOD}} \tag{8}$$

Hence

$$\mathbf{b}_{\mathrm{METHOD}} = \mathbf{Vb} \tag{9}$$

or, in more detail,

$$\mathbf{b}_{\mathrm{METHOD}} = \mathbf{V}(\mathbf{V}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{XV})^{-1}\mathbf{V}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{y} \tag{10}$$

The solution is invariant under any non-singular transformation of $\mathbf{V}$, the only thing of interest being the range of $\mathbf{V}$. This determines the range of $\mathbf{T} = \mathbf{XV}$, i.e. the subspace of the columns of $\mathbf{X}$ onto which $\mathbf{y}$ is projected. Orthogonality of $\mathbf{V}$ (or $\mathbf{T}$) is not an issue. The above result is general, hence it holds true for, among others, all variants of two-way PLS regression (e.g. non-orthogonal PLS[4], orthogonal PLS[4], SIMPLS[5]).

Let us return to Bro's N-way PLS. The results of the subspace-based regression approach also hold for NPLS if the understanding is that $\mathbf{X}$ $(I \times P)$ is a properly unfolded two-way form of N-way $\underline{\mathbf{X}}$ (P=JKL . . .). Method 1 (Section 2.1) essentially implements equation (9) or (10) with METH-

OD = NPLS and $\mathbf{V}=\mathbf{W}^*$. It is apparent from equation (1) that range($\mathbf{W}^*$) = range($\mathbf{W}$). By consequence, one may just as well use equation (10) with $\mathbf{V}=\mathbf{W}$. This in turn is equivalent to using modified PLS components $\mathbf{T}^*$:

$$\mathbf{T}^*=\mathbf{X}\mathbf{W} \tag{11}$$

Regressing $\mathbf{y}$ on $\mathbf{T}^*$ yields

$$\mathbf{b}^*=(\mathbf{T}^{*\mathrm{T}}\mathbf{T}^*)^{-1}\mathbf{T}^{*\mathrm{T}}\mathbf{y} \tag{12}$$

and, finally,

$$\mathbf{b}_{\mathrm{NPLS}}=\mathbf{W}\mathbf{b}^* \tag{13}$$

Thus a second way to obtain $\mathbf{b}_{\mathrm{NPLS}}$ is directly from $\mathbf{W}$ using modified $\mathbf{b}^*$ ($A\times 1$) obtained via equations (11) and (12). However, when $\mathbf{X}$ and $\mathbf{W}$ are huge, e.g. as in CoMFA ($P\approx 25\,000$),[3] obtaining the modified scores $\mathbf{T}^*$ (equation (11)) is a computationally intensive step. One may avoid this step and compute $\mathbf{b}^*$ in an alternative fashion as follows. let $\mathbf{D}$ ($A\times A$) be the non-singular matrix transforming $\mathbf{W}^*$ into $\mathbf{W}$ (or $\mathbf{T}$ into $\mathbf{T}^*$):

$$\mathbf{W}=\mathbf{W}^*\mathbf{D} \tag{14}$$

Then, using equations (6), (13) and (14),

$$\mathbf{b}_{\mathrm{NPLS}}=\mathbf{W}^*\mathbf{b}=\mathbf{W}\mathbf{b}^*=\mathbf{W}^*\mathbf{D}\mathbf{b}^* \tag{15}$$

Hence $\mathbf{b}=\mathbf{D}\mathbf{b}^*$ or

$$\mathbf{b}^*=\mathbf{D}^{-1}\mathbf{b} \tag{16}$$

The ath column of $\mathbf{D}$ can be deduced by considering the expression for the ath column of $\mathbf{W}^*$ as given by equation (1):

$$\begin{aligned} w_a^* &= (\mathbf{I}_P-\mathbf{w}_1\mathbf{w}_1^{\mathrm{T}})(\mathbf{I}_P-\mathbf{w}_2\mathbf{w}_2^{\mathrm{T}})\ldots(\mathbf{I}_P-\mathbf{w}_{a-2}\mathbf{w}_{a-2}^{\mathrm{T}})(\mathbf{I}_P-\mathbf{w}_{a-1}\mathbf{w}_{a-1}^{\mathrm{T}})\mathbf{w}_a \\ &= (\mathbf{I}_P-\mathbf{w}_1\mathbf{w}_1^{\mathrm{T}})(\mathbf{I}_P-\mathbf{w}_2\mathbf{w}_2^{\mathrm{T}})\ldots(\mathbf{I}_P-\mathbf{w}_{a-2}\mathbf{w}_{a-2}^{\mathrm{T}})(\mathbf{w}_a-\mathbf{w}_{a-1}(\mathbf{w}_{a-1}^{\mathrm{T}}\mathbf{w}_a)) \\ &= -(\mathbf{w}_{a-1}^{\mathrm{T}}\mathbf{w}_a)\mathbf{w}_{a-1}^*+(\mathbf{I}_P-\mathbf{w}_2\mathbf{w}_1^{\mathrm{T}})(\mathbf{I}_p-\mathbf{w}_2\mathbf{w}_2^{\mathrm{T}})\ldots(\mathbf{I}_P-\mathbf{w}_{a-2}\mathbf{w}_{a-2}^{\mathrm{T}})\mathbf{w}_a \end{aligned} \tag{17}$$

The last term of equation (17) can be worked out in the same way, and so on, repeated application leading to

$$\mathbf{w}_a^*=-(\mathbf{w}_{a-1}^{\mathrm{T}}\mathbf{w}_a)\mathbf{w}_{a-1}^*-(\mathbf{w}_{a-2}^{\mathrm{T}}\mathbf{w}_a)\mathbf{w}_{a-2}^*-\ \ldots\ -(\mathbf{w}_2^{\mathrm{T}}\mathbf{w}_a)\mathbf{w}_2^*-(\mathbf{w}_1^{\mathrm{T}}\mathbf{w}_a)\mathbf{w}_1^*+\mathbf{w}_a \tag{18}$$

or

$$\mathbf{w}_a=(\mathbf{w}_1^{\mathrm{T}}\mathbf{w}_a)\mathbf{w}_1^*+(\mathbf{w}_2^{\mathrm{T}}\mathbf{w}_a)\mathbf{w}_2^*+\ldots+(\mathbf{w}_{a-1}^{\mathrm{T}}\mathbf{w}_a)\mathbf{w}_{a-1}^*+\mathbf{w}_a^* \tag{19}$$

Thus $\mathbf{D}=(d_{ia})$, with $d_{ia}=\mathbf{w}_i^{\mathrm{T}}\mathbf{w}_a$ for $i\le a\le A$ and $d_{ia}=0$ for $a<i\le A$. In other words, $\mathbf{D}$ equals the upper triangular part of $\mathbf{W}^{\mathrm{T}}\mathbf{W}$, the matrix of inner products of the weight vectors $\mathbf{w}_a$ (a=1, 2, . . . , A). $\mathbf{D}$ can be constructed conveniently when building the NPLS model, allowing the computation of $\mathbf{b}_{\mathrm{NPLS}}$ via equations (16) and (13). A still more efficient approach, however, is to apply Method 3.

2.3. Method 3

Method 3 does not differ essentially from Method 2, but it involves a slight change of the NPLS model. By a trivial modification of Bro's NPLS algorithm, namely by omitting the deflation of the $\underline{\mathbf{X}}$-

array (first part of Step 5 in Table 2 of Reference 1), one obtains **T*** instead of **T** and **b*** instead of **b**, without affecting the weights $\mathbf{w}_a$. One may appreciate the latter result by considering the vector **z** of covariances of deflated **y**, i.e. $\mathbf{y}^{(a-1)}$, with either the original **X** or the deflated **X**, i.e. $\mathbf{X}^{(a-1)}=\mathbf{X}-\mathbf{T}_{[1:a-1]}\mathbf{W}^{\mathrm{T}}_{[1:a-1]}$:

$$\begin{aligned}\mathbf{z}&=\mathbf{y}^{(a-1)\mathrm{T}}\mathbf{X}^{(a-1)}=((\mathbf{I}_I-\mathbf{T}(\mathbf{T}^{\mathrm{T}}\mathbf{T})^{-1}\mathbf{T}^{\mathrm{T}})\mathbf{y})^{\mathrm{T}}(\mathbf{X}-\mathbf{T}\mathbf{W}^{\mathrm{T}})\\&=\mathbf{y}^{\mathrm{T}}(\mathbf{I}_I-\mathbf{T}(\mathbf{T}^{\mathrm{T}}\mathbf{T})^{-1}\mathbf{T}^{\mathrm{T}})(\mathbf{X}-\mathbf{T}\mathbf{W}^{\mathrm{T}})=\mathbf{y}^{\mathrm{T}}(\mathbf{I}_I-\mathbf{T}(\mathbf{T}^{\mathrm{T}}\mathbf{T})^{-1}\mathbf{T}^{\mathrm{T}})\mathbf{X}=\mathbf{y}^{(a-1)\mathrm{T}}\mathbf{X}\end{aligned}\quad(20)$$

where we have omitted the subscripts indicating the current size of **T** and **W**. We conclude that deflating **X** or not does not affect the weight vector $\mathbf{w}_a$, since it depends solely on (unaltered) **z**. Thus, by skipping the deflation of **X**, we not only simplify the NPLS algorithm and increase its speed, but we also obtain the regression coefficients $\mathbf{b}_{\mathrm{NPLS}}$ in the simplest possible way, namely from the available **W** and **b*** as $\mathbf{b}_{\mathrm{NPLS}}=\mathbf{W}\mathbf{b}^*$ (equation (13)). A drawback of the approach is that no residual **X** is available for diagnostic purposes. The resulting algorithm for three-way $\underline{\mathbf{X}}$ and a single response **y** (tri-PLS1) is shown in Table 1 as Matlab code.

3. EQUIVALENCE OF Tri-PLS AND TLD

Since the deflation of **X** is immaterial, as long as it is of the form $\mathbf{X}^{(a-1)}=\mathbf{X}-\mathbf{T}_{[1:a-1]}\mathbf{P}^{\mathrm{T}}_{[1:a-1]}$, for some $(P\times(a-1))$ **P**, one might also deflate using regression loadings $\mathbf{P}=\mathbf{X}^{\mathrm{T}}\mathbf{T}(\mathbf{T}^{\mathrm{T}}\mathbf{T})^{-1}$, as, for example, in standard orthogonal PLS2. This is the approach adopted by Ståhle in his LTD (linear three-way decomposition) algorithm.[6] It leads to different, orthogonal, score vectors. Another difference between Bro's NPLS and Ståhle's TLD is the way of computing the weight vectors associated with each mode of **Z**. Ståhle employs an alternating least squares approach, cycling through all modes of $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$, as a simple extension of two-way PLS to the three-way situation. Bro's algorithm is based on a stringent extension of the PLS optimization criterion to higher orders using a singular vector decomposition of **Z** (or a one-component PARAFAC decomposition with multiway $\underline{\mathbf{Z}}$). The latter approach has several advantages: it has an explicit optimization criterion, is numerically more reliable and is usually faster. The weights found, however, are the same as in TLD, which also maximizes (iteratively) the covariance. As a result, the score vectors in TLD span the same space as the columns of **T** in (2) or **T*** in (11). The TLD scores can be obtained by Gram–Schmidt orthogonalization of either **T** or **T***. The fits of **Y** obtained by the two methods are identical and one arrives at the same calibration model, i.e. $\mathbf{b}_{\mathrm{TLD}}=\mathbf{b}_{\mathrm{NPLS}}$.

Table 1. Matlab code for tri-PLS1 regression of **y** $(I\times 1)$ on (centred) **X** $(I\times JK)$

```
e=y;                                          % initialize
for lv=1:LV                                   % for each factor
   Z=reshape(e'*X,J,K);                       % vector of covariances→matrix
   [wJ, wK]=svd(Z);                           % find weights maximizing covariance
   WJ=[WJ wJ(:,1)];                           % save weights J-mode
   WK=[WK wK(:,1)];                           % save weights K-mode
   T=[T X*kron(wK(:,1),wJ(:,1))];             % save scores I-mode
   b=inv(T'*T)*T'*y;                          % y loadings wrt T
   e=y-T*b;                                   % residual y
end
bNPLS=0;
for 1v=1:LV
   bNPLS=bNPLS+kron(WK(:,1v),WJ(:,1v))*b(1v); %regression coeffs
end
```


4. CONCLUSIONS

Compact expressions have been obtained for computing the regression coefficients $\mathbf{b}_{\mathrm{NPLS}}$ in predictive *N*-way PLSk calibration. A slight modification of Bro's NPLS algorithm simplifies both the calculation of the weight vectors $\mathbf{w}_a$ and the computation of the regression coefficients $\mathbf{b}_{\mathrm{NPLS}}$. It has been shown that Bro's NPLS algorithm, the modification proposed in Method 3 (Table 1) and Ståhle's LTD algorithm find the same estimate of the *N*-way calibration model.

REFERENCES

1. R. Bro, *J. Chemometrics,* **10**, 47–61 (1996).
2. A. K. Smilde, *J. Chemometrics*, **11**, 367–377 (1997).
3. J. Nilsson, S. de Jong and A. K. Smilde, *J. Chemometrics*, **11**, 511–524 (1997).
4. H. Martens and T. Naes, *Multivariate Calibration*, Wiley, Chichester (1989).
5. S. de Jong, *Chemometrics Intell. Lab. Syst.* **18**, 251–263 (1993).
6. L. Ståhle, *Chemometrics Intell. Lab. Syst.* **7**, 95–100 (1989).

