JOURNAL OF CHEMOMETRICS
J. Chemometrics 2003; **17**: 287–301
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/cem.802

Generalized orthogonal multiple co-inertia analysis(–PLS): new multiblock component and regression methods

Myrtille Vivien* and Robert Sabatier

Faculté de Pharmacie, Laboratoire de Physique Moléculaire et Structurale, UMR 5094, 15 Avenue Charles Flahault, BP 14491, F-34093 Montpellier Cedex 5, France

Received 25 April 2002; Revised 19 February 2003; Accepted 21 March 2003

The purpose of this paper is to develop new component-wise component and regression multiblock methods that overcome some of the difficulties traditionally associated with multiblocks, such as the step-by-step optimization and component orthogonalities. Generalized orthogonal multiple co-inertia analysis (GOMCIA) and generalized orthogonal multiple co-inertia analysis–partial least squares (GOMCIA–PLS) are proposed for modelling two sets of blocks measured on the same observations. We especially emphasize GOMCIA–PLS methods in which we consider one of the sets as predictive. All these methods are based on the step-by-step maximization of the same criterion under normalization constraints and produce orthogonal components or super-components. The solutions of the problem have to be computed with an iterative algorithm (which we prove to be convergent). We also give some interesting special cases and discuss the differences compared with a few other multiblock and/or multiway methods. Finally, short examples of real data are processed to show how GOMCIA–PLS can be used and its properties. Copyright © 2003 John Wiley & Sons, Ltd.

KEYWORDS: multiway array; multiway analysis; sensory analysis; PLS methods; multiblock

1. INTRODUCTION

Assume you are in one of the following situations: you possess a very large data set, i.e. a great number of variables measured on the same observations, maybe with collinear variables, that you want to analyse and possibly use for predicting other variables; or you have several groups of variables measured on the same observations and you want to analyse them by taking into account this block (group) structure, possibly for predicting others groups. Some people use the usual statistical methods such as principal component Analysis or PLS regression [1,2] on the whole data set or on a summarized data set. However, these methods are not appropriate because, in the first case, the results are too difficult to interpret owing to the number of variables and, in the second case, summarizing the data can lead to important phenomena being 'hidden'. These kinds of situations are frequently encountered in many applied fields. For example, in sensory analysis, one wants to compare assessors' evaluations to know if one of them differs from the others; or one wants to predict consumers' preferences from the assessors' evaluations. In chemometrics, very large chemical or NIRS data sets are used to predict other variables. A way to overcome the problem is to use multiblock or multiway methods. Multiway methods were first proposed by Tucker [3–5] to model a three-way matrix. Much complementary work has been done to improve and generalize his work [6–11]. When the purpose is prediction, implying three-way matrices, one can apply linear three-way decomposition [12], multilinear PLS (NPLS) [13,14] or multiway covariate regression models [15]. When the data set is not multiway, one must use multiblock (component or regression) methods. A multiblock method can be defined as a method that allows one to analyse simultaneously a set of matrices measured on the same observations, or several sets of matrices, possibly for predicting some of them. Figure 1 presents two possible situations. Such methods first appeared in the 1970s with Gower's [16] generalized Procrustes analysis. In France, several methods which are little known by chemometricians, e.g. the ACT (STATIS) method [17–19] and, more recently, multiple co-inertia analysis (MCOA) [20], concordance analysis [21] and orthogonal multiple co-inertia analysis

*Correspondence to: M. Vivien, Faculté de Pharmacie, Laboratoire de Physique Moléculaire et Structurale, UMR 5094, 15 Avenue Charles Flahault, BP 14491, F-34093 Montpellier Cedex 5, France.
E-mail: mvivien@pharma.univ-montp1.fr

Copyright © 2003 John Wiley & Sons, Ltd.

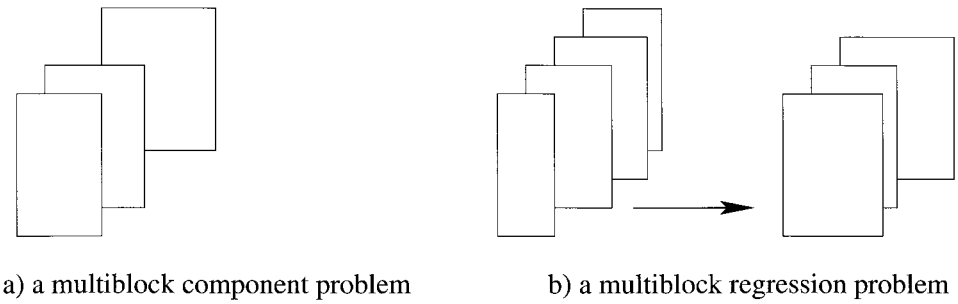

Figure 1. Examples of multiblock component and regression problems: (a) one set of matrices to explain—a component problem; (b) a set of matrices to predict by another—a regression problem.

(OMCIA) [22], have been proposed for 'component' problems*, and orthogonal multiple co-inertia analysis–PLS (OMCIA–PLS)† [23] for regression problems. The so-called 'multiblock' regression methods have been developed by chemometricians to analyse large data sets by blocking them into several smaller groups of variables. They are all based on *ad hoc* modifications of the PLS algorithm to adjust it to the multiblock case, and then block components (block scores) and super-components are generated. Wangen and Kowalski's [24] multiblock PLS (MBPLS) was the first of these methods and is one of the best known. There are several ways to apply it, depending on how the deflation is made. It has recently been shown that it is equivalent to performing the PLS algorithm on the large $\mathbf{X}$ and $\mathbf{Y}$ [25,26] and that it is possible to deduce the block scores from the $\mathbf{t}$-scores of this PLS regression [27]. Another method, hierarchical PLS (HPLS), was established by Wold *et al.* [28], but its results are not very good and it has been shown that the X-blocks are better modelled than the Y-blocks [26], whereas the goal of the method is modelling the Y-blocks.

Another class of methods exists: the unfolding ones. These methods, e.g. multiway PLS (MPLS) [29], consist in unfolding a three-way (or N-way) matrix into a two-way matrix. In fact, MPLS is equivalent to performing a standard PLS regression on the unfolded matrices, so this is not really a multiway method.

The purpose of this paper is to present new multiblock component and regression methods which generalize OMCIA and OMCIA–PLS quoted above. These are component-wise methods like MBPLS or HPLS, but the main difference is that the methods we propose optimize a criterion step by step under constraints which take into account the block structure of the data. Multiblock component methods are called generalized orthogonal multiple co-inertia analysis (GOMCIA). They are designed to study the proximity of two sets of matrices. An example application is the comparison of two groups of assessors in sensory analysis: a group of K assessors evaluate I products by J characteristics (this makes a set of K matrices $\mathbf{X}_k$ with I rows and J columns) and a second group of N other assessors evaluate the same I products by M characteristics, which could be the same as the J ones (this makes a set of N matrices $\mathbf{Y}_n$ with I rows and M columns). One wants to study the homogeneity within each of the two groups of assessors and the proximity between the two groups in order to know if the groups give the same responses. GOMCIA methods are dedicated to these kinds of situations. If one uses a multiblock component method such as STATIS or MCOA quoted above, one cannot evaluate the 'global' proximity between the two sets of matrices, but only the proximities and differences among the $K + N$ assessors. The use of such a method is equivalent to supposing that all the assessors belong to the same group. With a method such as GOMCIA we add to the study the knowledge of two different groups. Another example of the applicability of GOMCIA is an ecological problem: the comparison of measures of J pollution variables (columns of matrices) recorded at two different depths at I different sites (rows of all the matrices) of a river on K dates. The aim is to compare two three-way matrices (one per depth) with the same dimensions ($I \times J \times K$).

Multiblock methods are called GOMCIA–PLS. Here the aim is to model a set of response matrices with a set of predictive X-blocks. The term 'PLS' recalls that the methods proceed in the same way as PLS regression: models are obtained with successive deflations onto X-block components. All the methods are based on the same step-by-step optimization, but they differ in how the deflation is made at steps higher than one. Orthogonal X-block components or super-components are produced according to the method, which makes interpretation easier. We will focus on the GOMCIA–PLS methods and deal with some ways to make the X-block selection. Assuming one has a large number of X-blocks, block selection, with analogy to variable selection in multiple regression, can be a fast way to eliminate some noisy groups of variables before analysing the data in more depth.

First (Section 2) we present the data set we use to illustrate the properties of GOMCIA–PLS in Section 6. Section 3 states the notation and recalls some definitions. In Section 4, GOMCIA(–PLS) theory is exposed with definitions, solutions, models and the iterative algorithm. Section 5 covers

* The term 'component' method is used to refer to the case where there is no response variable. The aim is only to model and explain the closeness and differences among a set or several sets of matrices.

† A few comments about this name are in order. The term 'co-inertia' is due to the fact that we are in search of common structure, and 'partial least squares' because the method is proceeding like PLS regression. Finally, the most interesting term, 'orthogonal', is due to the orthogonality of components included in the building of the models: it makes the results easier to interpret.

Copyright © 2003 John Wiley & Sons, Ltd.

block selection, comments and special cases. Section 6 presents the application comparing GOMCIA–PLS, HPLS and NPLS results.

2. A TYPICAL DATA SET

2.1 Presentation

We have chosen this small data set only to describe GOMCIA–PLS properties when there is a single **y** response and one set of X-matrices. However, it should be kept in mind that when experimenters have such a small data set, usual PLS is also appropriate. One might think it too restrictive that there are no Y-blocks. Actually, as in PLS methods, the processing is not really different whether there is one response or more. In addition, GOMCIA–PLS methods have some interesting features (see Section 5.3) even when there is only one **y**.

The data [30,31] are structure–activity relationship data extracted from a text used by Umetrics Inc. [32] in teaching and illustrating PLS. Some properties—size, lipophilicity and polarity—were computed at various sites of synthesized peptide molecules in order to study their relationships with the activity of each peptide molecule. This activity is represented by the logarithm of the relative bradykinin-potentiating activity. The ultimate purpose is to predict the biological activity of a compound from inexpensive chemical measurements.

Measurements of the three variables were made on 30 molecules (observations) at five sites, so **y** has length $I = 30$. The X-data are described by three modes: observations, sites and variables. The X-matrices can be arranged in three different ways: first, as a three-way matrix of size $30 \times 3 \times 5$; second, as a set of three matrices of size 30×5; and third, as a set of five matrices of size 30×3. In the first case, by considering them as a three-way array, we search for a three-way structure explaining the response **y**, i.e. for a trilinear decomposition of **y**. In the second case we search for whether a site is much more predictive than others. In the last case we search for which variables among *lipophilicity, size* and *polarity* are the most predictive.

2.2. The way the data were studied in Reference [32]

To illustrate PLS in Reference [32], the data were used by keeping the first 15 observations to fit a model and by using the last 15 observations to test the predictive quality of the model with only two components. It can be noticed that this method is equivalent to using MPLS [29] on the data. No cross-validation was made to choose the number of components. The conclusion was that eight variables are important in the model and that the model explains well enough **y** but does not predict well the last 15 observations. In fact, this result is due to the choice of using the first 15 observations to build the model. In Reference [32] the authors state that the first 15 observations are not appropriate to predict the last 15 observations. In addition, we have noted that **y** is only significantly correlated with two of the X-variables. This can justify that the model is not a good predictive one. In Section 6 we try several methods and compare the results with these ones. To be able to really compare the results, we choose to keep only the first 15 observations to build the models too.

3. BASIC DEFINITIONS AND NOTATION

Vectors (columns) are denoted as bold lowercase letters (sometimes Greek), e.g. **x**. Matrices are denoted as bold uppercase letters, e.g. **X**. Transposed matrices and vectors are denoted as $\mathbf{X}'$ and $\mathbf{x}'$ respectively.

We also use the following:

- a, index of components ($a = 1, \ldots, A$);
- K, number of X-blocks, e.g. number of matrices used as predictors;
- k, index of X-blocks;
- N, number of Y-blocks, e.g. number of matrices used as responses;
- n, index of Y-blocks;
- I, number of observations (statistical units);
- i, index of observations;
- M_n, $n = 1,\ldots,N$, number of Y-variables in block n;
- m, index of Y-variables in a block;
- J_k, $k = 1,\ldots,K$, number of X-variables in block k;
- j, index of X-variables in a block;
- $\mathbf{X}_k$, kth block of predictor variables, $I \times J_k$;
- $\mathbf{Y}_n$, nth block of responses variables, $I \times M_n$;
- $\underline{\mathbf{X}}$, a set of X-blocks or a multiway array;
- $\underline{\mathbf{Y}}$, a set of Y-blocks or a response multiway array;
- $\mathbf{X}$, unfolded $\underline{\mathbf{X}}$, i.e. $\mathbf{X} = [\mathbf{X}_1|\ldots|\mathbf{X}_K]$;
- $\mathbf{Y}$, unfolded $\underline{\mathbf{Y}}$, i.e. $\mathbf{Y} = [\mathbf{Y}_1|\ldots|\mathbf{Y}_N]$;
- $\mathbf{I}_I$, identity matrix in $\mathbb{R}^I$;
- $\mathbf{1}_I$, vector of ones in $\mathbb{R}^I$, i.e. $(1\ 1\ldots1)'$;
- $\mathbf{D}$, metric in the variable space $\mathbb{R}^I$; $\mathbf{D}$ is a diagonal positive matrix with $tr(\mathbf{D}) = 1$, which induces a scalar product and, as a consequence, a norm; often $\mathbf{D}$ is the uniform metric, but other choices are possible;
- $\mathbf{t}_{k,a}$, ath component or score vector of $\mathbf{X}_k$;
- $\mathbf{u}_{n,a}$, ath component or score vector of $\mathbf{Y}_n$;
- $\mathbf{a}_{k,a}$, weight vector of $\mathbf{t}_{k,a}$;
- $\mathbf{b}_{n,a}$, weight vector of $\mathbf{u}_{n,a}$;
- $\mathbf{P}_\mathbf{v} = \mathbf{v}(\mathbf{v}'\mathbf{D}\mathbf{v})^{-1}\mathbf{v}'\mathbf{D}$, $\mathbf{D}$-orthogonal projector on the subspace spanned by $\mathbf{v}$;
- $\mathbf{P}_\mathbf{v}^\perp = \mathbf{I}_I - \mathbf{P}_\mathbf{v}$;
- $R_n^2(a)$, multiple correlation coefficient: variation explained by the a-order model for $\mathbf{Y}_n$ (model with a components);
- $R_k^2(a)$, multiple correlation coefficient: variation explained by the a-order model for $\mathbf{X}_k$ (model with a components);
- $R_\mathbf{Z}^2(a)$, variation explained by $\mathbf{Z}$ model of order a; $\mathbf{Z}$ is equal to $\mathbf{X}$ or $\mathbf{Y}$;
- $r(\mathbf{x}, \mathbf{y})$, simple correlation coefficient between $\mathbf{x}$ and $\mathbf{y}$;
- $\|\mathbf{v}\|$, Euclidean norm of vector $\mathbf{v}$;
- $\|\mathbf{X}\|$, Frobenius norm of matrix $\mathbf{X}$;
- $\|\mathbf{v}\|_\mathbf{D} = \sqrt{\mathbf{v}'\mathbf{D}\mathbf{v}}$, $\mathbf{D}$-norm of $\mathbf{v}$;
- $\mathbf{Diag}(\mathbf{v}_i)$, block diagonal matrix where block i is $\mathbf{v}_i$;
- $diag(\mathbf{X})$, vector of diagonal elements of $\mathbf{X}$;
- $\mathbf{X}_k^{(a-1)}$, updated $\mathbf{X}_k$ (residuals);
- $\mathbf{Y}_n^{(a-1)}$, updated $\mathbf{Y}_n$ (residuals);
- $cov(\mathbf{X}, \mathbf{Y}) = \mathbf{X}'\mathbf{D}\mathbf{Y}$, $\mathbf{D}$-covariance between $\mathbf{X}$ and $\mathbf{Y}$, if $\mathbf{X}$ and $\mathbf{Y}$ are centred for $\mathbf{D}$;

Copyright © 2003 John Wiley & Sons, Ltd.

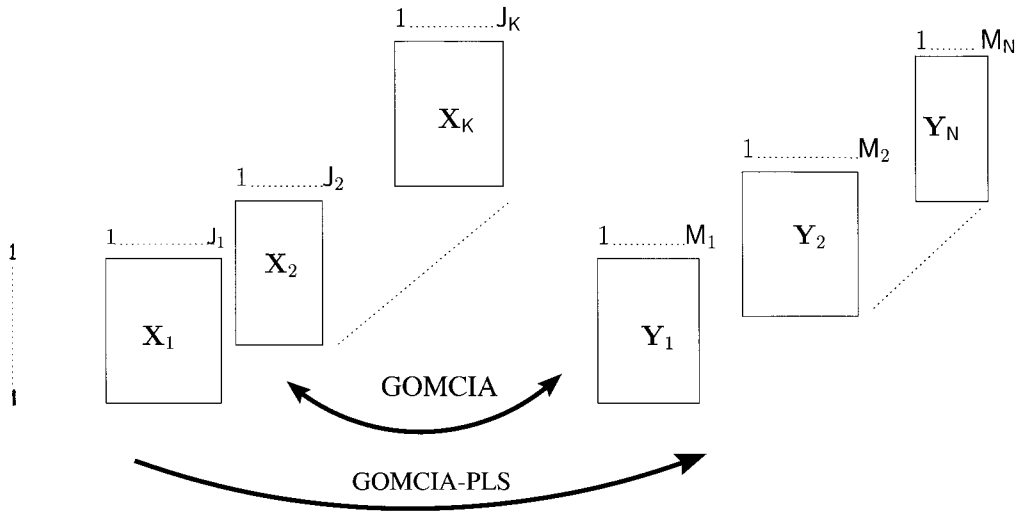

Figure 2. A situation where generalized orthogonal multiple co-inertia analysis and generalized orthogonal multiple co-inertia analysis–PLS can be applied. All matrices are measured on the same observations.

- $B_n^2(k)$, coefficient in per cent used to represent the influence of the kth X-block in the model for $\mathbf{Y}_n$.

All the vectors $\mathbf{t}_{k,a}$, $\mathbf{u}_{n,a}$ (scores), $\mathbf{a}_{k,a}$, $\mathbf{b}_{n,a}$ (weights), etc. are called the 'a-order solutions'. Most of the time the 1-order solutions are denoted $\mathbf{t}_k$, $\mathbf{u}_n$, etc. instead of $\mathbf{t}_{k,1}$, $\mathbf{u}_{n,1}$, etc.

4. GOMCIA AND GOMCIA–PLS: THEORY

The aim of GOMCIA(–PLS) is to study the relationships, if any, between two sets of data matrices $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$ with respectively K and N matrices. All the matrices have to be measured on the same observations. If the data are three-way, GOMCIA and GOMCIA–PLS can still be applied, but they do not take into account the effect of one of the modes—this depends on how the three-way matrices are cut. Figure 2 summarizes the situation.

GOMCIA(–PLS) models of each $\mathbf{Y}_n$ and $\mathbf{X}_k$ are built by taking into account the closeness between the two sets of $\mathbf{X}_k$ and $\mathbf{Y}_n$. As in PLS methods, the models are component-wise calculated step by step and can be decomposed into sums of partial rank 1 models (except for GOMCIA–PLS1 ones). All the methods we present produce orthogonal X-block components or super-components, which makes interpreting the results and producing graphics easier. First we will introduce the problem at step 1 ($a = 1$) and then we will give the five possible choices that define the next steps.

4.1. The problem

We assume that the data are preprocessed in the following way: the matrices $\mathbf{X}_k$ and $\mathbf{Y}_n$ are centred according to $\mathbf{D}$ (i.e. $\mathbf{1}_I'\mathbf{D}\mathbf{X}_k = \mathbf{0}$ and $\mathbf{1}_I'\mathbf{D}\mathbf{Y}_n = \mathbf{0}$) and, if necessary, scaled according to $\mathbf{D}$ (i.e. $diag(\mathbf{X}_k'\mathbf{D}\mathbf{X}_k) = \mathbf{1}_{J_k}$ and/or $diag(\mathbf{Y}_n'\mathbf{D}\mathbf{Y}_n) = \mathbf{1}_{M_n}$).

The first step ($a = 1$) of generalized orthogonal multiple co-inertia analysis(–PLS) consists in computing the weight vectors $\mathbf{a}_{k,1}$ and $\mathbf{b}_{n,1}$ that maximize the criterion C given below under constraints on $\|\mathbf{a}_{k,1}\|^2$ and $\|\mathbf{b}_{n,1}\|^2$. This criterion is the double sum of all covariances between block scores $\mathbf{t}_k$, associated with $\mathbf{X}_k$, and $\mathbf{u}_n$, associated with $\mathbf{Y}_n$:

$$C = \sum_{k=1}^{K}\sum_{n=1}^{N} \mathbf{t}_k'\mathbf{D}\mathbf{u}_n \tag{1}$$

where

$$\mathbf{t}_k = \mathbf{X}_k\mathbf{a}_k, \qquad \mathbf{u}_n = \mathbf{Y}_n\mathbf{b}_n$$

with

$$\|\mathbf{a}_k\|^2 = 1 = \|\mathbf{b}_n\|^2$$

The ath step of GOMCIA(–PLS) consists in computing the weight vectors $\mathbf{a}_{k,a}$ and $\mathbf{b}_{n,a}$ that maximize C when $\mathbf{X}_k$ and $\mathbf{Y}_n$ are replaced by the updated matrices $\mathbf{X}_k^{(a-1)}$ and $\mathbf{Y}_n^{(a-1)}$, which will be defined later. They correspond to residuals of some projection chosen by the user among five possibilities. Then, at each stage a, C is optimized.

4.2. Solutions of order one

We describe here how the first-order ($a = 1$) solutions are calculated. For simplicity of notation we denote these first-order solutions without their subscript 1 throughout this subsection only.

To maximize C under constraints, we use Lagrangian multipliers [33]. We denote by μ_k the Lagrangian multiplier associated with the constraint of normalization $\|\mathbf{a}_k\|^2 = 1$, and by λ_n the one associated with the constraint $\|\mathbf{b}_n\|^2 = 1$. Hence the Lagrangian function $\mathcal{L}$ is defined as

$$\mathcal{L} = \sum_{k=1}^{K}\sum_{n=1}^{N} \mathbf{a}_k'\mathbf{X}_k'\mathbf{D}\mathbf{Y}_n\mathbf{b}_n - \sum_{k=1}^{K} \tfrac{1}{2}\mu_k(\mathbf{a}_k'\mathbf{a}_k - 1) - \sum_{n=1}^{N} \tfrac{1}{2}\lambda_n(\mathbf{b}_n'\mathbf{b}_n - 1) \tag{2}$$

The normal equations are given in Appendix I. Their resolution leads to the expressions of the weight vectors,

Copyright © 2003 John Wiley & Sons, Ltd.

multipliers and maximum of the objective function C:

$$\mu_k = \sum_{n=1}^{N} \mathbf{a}_k' \mathbf{X}_k' \mathbf{D} \mathbf{Y}_n \mathbf{b}_n = cov\left(\mathbf{t}_k, \sum_{n=1}^{N} \mathbf{u}_n\right) \tag{3}$$

$$\lambda_n = \sum_{k=1}^{K} \mathbf{a}_k' \mathbf{X}_k' \mathbf{D} \mathbf{Y}_n \mathbf{b}_n = cov\left(\mathbf{u}_n, \sum_{k=1}^{K} \mathbf{t}_k\right) \tag{4}$$

$$C_{\text{opt}} = \sum_{k=1}^{K} \sum_{n=1}^{N} \mathbf{a}_k' \mathbf{X}_k' \mathbf{D} \mathbf{Y}_n \mathbf{b}_n = \sum_{n=1}^{N} \lambda_n = \sum_{k=1}^{K} \mu_k \tag{5}$$

$$\mathbf{a}_k = \frac{1}{\mu_k} \sum_{n=1}^{N} \mathbf{X}_k' \mathbf{D} \mathbf{Y}_n \mathbf{b}_n \tag{6}$$

$$\mathbf{b}_n = \frac{1}{\lambda_n} \sum_{k=1}^{K} \mathbf{Y}_n' \mathbf{D} \mathbf{X}_k \mathbf{a}_k \tag{7}$$

Note that the multipliers μ_k and λ_n are non-zero by definition. They are chosen strictly positive and equal to

$$\mu_k = \left\| \mathbf{X}_k' \mathbf{D} \sum_{n=1}^{N} \mathbf{u}_n \right\| \tag{8}$$

$$\lambda_n = \left\| \mathbf{Y}_n' \mathbf{D} \sum_{k=1}^{K} \mathbf{t}_k \right\| \tag{9}$$

Unfortunately, neither $\mathbf{a}_k$ nor $\mathbf{b}_n$ are the solutions of an eigenvalue problem; hence we propose an algorithm, described in Section 4.5.

As the first-order solutions are the same in GOMCIA and GOMCIA–PLS methods, in the following we call the first step described here 'step 1 of GOMCIA'.

4.3. Solutions of higher order

Henceforth the a-order solutions are indexed with a, $a > 1$.

We distinguish two sorts of methods: GOMCIA, multiblock component methods; and GOMCIA–PLS, multiblock regression methods. The scope of the first is to explain the common structure between two sets of arrays. For the second the purpose is to explain $\underline{\mathbf{Y}}$ by $\underline{\mathbf{X}}$ in order to make linear predictions.

Set $\mathbf{X}_k^{(0)} = \mathbf{X}_k, \mathbf{Y}_n^{(0)} = \mathbf{Y}_n$ and $1 < a \leq A$. A is the 'optimal' step number. It can be chosen by cross-validation (see Section 5.2). The definitions below give the expressions of deflated tables $\mathbf{X}_k^{(a-1)}$ and $\mathbf{Y}_n^{(a-1)}$ for our five methods.

Definition 1

Step a of GOMCIA1 is defined by the first step of GOMCIA applied to $\mathbf{X}_1^{(a-1)}, \ldots, \mathbf{X}_K^{(a-1)}$, $\mathbf{Y}_1^{(a-1)}, \ldots, \mathbf{Y}_N^{(a-1)}$ and $\mathbf{D}$, where $\mathbf{X}_k^{(a-1)} = \mathbf{P}_{\mathbf{t}_{k,a-1}}^{\perp} \mathbf{X}_k^{(a-2)}$ and $\mathbf{Y}_n^{(a-1)} = \mathbf{P}_{\mathbf{u}_{n,a-1}}^{\perp} \mathbf{Y}_n^{(a-2)}$.

Thus the scores $\{\mathbf{t}_{k,a}\}_a$ are $\mathbf{D}$-orthogonal and the scores $\{\mathbf{u}_{n,a}\}_a$ are too. Moreover, it can be proved that the sets of weights $\{\mathbf{a}_{k,a}\}_a$ and $\{\mathbf{b}_{n,a}\}_a$ are two orthonormal systems (see Appendix III). These properties of orthogonality allow graphical representations which can be interpreted as projections onto orthonormal axes.

The multiplier μ_k measures, in a sense, the proximity between $\mathbf{X}_k$ and the whole set $\{\mathbf{Y}_n\}_n$. In the same way, λ_n measures the proximity between $\mathbf{Y}_n$ and the whole set $\{\mathbf{X}_k\}_k$.

Consider $\mathbf{t}_a = \Sigma_{k=1}^{K} \mathbf{t}_{k,a}$ and $\mathbf{u}_a = \Sigma_{n=1}^{N} \mathbf{u}_{n,a}$. We call them super-components (or super-scores) as done in References [26,28]. Although $\{\mathbf{t}_a\}_a$ are not orthogonal, they can be useful for representing the observations described by the whole $\underline{\mathbf{X}}$. The same holds for $\mathbf{u}_a$.

Then comes another definition.

Definition 2

Step a of GOMCIA2 is defined by the first step of GOMCIA applied to $\mathbf{X}_1^{(a-1)}, \ldots, \mathbf{X}_K^{(a-1)}$, $\mathbf{Y}_1^{(a-1)}, \ldots, \mathbf{Y}_N^{(a-1)}$ and $\mathbf{D}$, where $\mathbf{X}_k^{(a-1)} = \mathbf{P}_{\mathbf{t}_{a-1}}^{\perp} \mathbf{X}_k^{(a-2)}$ and $\mathbf{Y}_n^{(a-1)} = \mathbf{P}_{\mathbf{u}_{a-1}}^{\perp} \mathbf{Y}_n^{(a-2)}$.

In GOMCIA2 the super-scores $\{\mathbf{t}_a\}_a$ are $\mathbf{D}$-orthogonal while the block scores $\{\mathbf{t}_{k,a}\}_a$ are not. The same applies to scores $\{\mathbf{u}_a\}_a$ and block scores $\{\mathbf{u}_{n,a}\}_a$ respectively.

The user has to choose between GOMCIA1 and 2 according to what is considered the most important. To study the differences or proximities among the blocks in more detail, GOMCIA1 is more suitable. However, if the user wants to interpret all the blocks at the same time, i.e. to summarize the variables, GOMCIA2 seems to be more appropriate.

Let us move on to the GOMCIA–PLS methods. In practice, these seem to be more useful. GOMCIA–PLS methods are proposed to model the response $\underline{\mathbf{Y}}$ with a predictive $\underline{\mathbf{X}}$. The difference compared with the GOMCIA methods is that $\mathbf{X}_k$ and $\mathbf{Y}_n$ are projected onto the same space; that is to say, for steps higher than 1 the problem is no longer symmetrical.

Definition 3

Step a of GOMCIA–PLS1 is defined by the first step of GOMCIA applied to $\mathbf{X}_1^{(a-1)}, \ldots, \mathbf{X}_K^{(a-1)}$, $\mathbf{Y}_1^{(a-1)}, \ldots, \mathbf{Y}_N^{(a-1)}$ and $\mathbf{D}$, where $\mathbf{X}_k^{(a-1)} = \mathbf{P}_{\mathbf{T}_{a-1}}^{\perp} \mathbf{X}_k^{(a-2)}$ and $\mathbf{Y}_n^{(a-1)} = \mathbf{P}_{\mathbf{T}_{a-1}}^{\perp} \mathbf{Y}_n^{(a-2)}$, with $\mathbf{T}_{a-1} = [\mathbf{t}_{1,a-1} | \ldots | \mathbf{t}_{K,a-1}]$.

Definition 4

Step a of GOMCIA–PLS2 is defined by the first step of GOMCIA applied to $\mathbf{X}_1^{(a-1)}, \ldots, \mathbf{X}_K^{(a-1)}, \mathbf{Y}_1^{(a-1)}, \ldots, \mathbf{Y}_N^{(a-1)}$ and $\mathbf{D}$, where $\mathbf{X}_k^{(a-1)} = \mathbf{P}_{\mathbf{t}_{a-1}}^{\perp} \mathbf{X}_k^{(a-2)}$ and $\mathbf{Y}_n^{(a-1)} = \mathbf{P}_{\mathbf{t}_{a-1}}^{\perp} \mathbf{Y}_n^{(a-2)}$, with $\mathbf{t}_{a-1} = \Sigma_{k=1}^{K} \mathbf{t}_{k,a-1}$.

Definition 5

Step a of GOMCIA–PLS3 is defined by the first step of GOMCIA applied to $\mathbf{X}_1^{(a-1)}, \ldots, \mathbf{X}_K^{(a-1)}, \mathbf{Y}_1^{(a-1)}, \ldots, \mathbf{Y}_N^{(a-1)}$ and $\mathbf{D}$, where $\mathbf{X}_k^{(a-1)} = \mathbf{P}_{\mathbf{t}_{a-1}}^{\perp} \mathbf{X}_k^{(a-2)}$ and $\mathbf{Y}_n^{(a-1)} = \mathbf{P}_{\mathbf{t}_{a-1}}^{\perp} \mathbf{Y}_n^{(a-2)}$, with $\mathbf{t}_{a-1} = \Sigma_{k=1}^{K} \mu_{k,a-1} \mathbf{t}_{k,a-1}$.

In GOMCIA–PLS2 and 3 the A super-components $\{\mathbf{t}_a\}_a$ are $\mathbf{D}$-orthogonal while $\{\mathbf{u}_a\}_a$ are not. In these two methods it is possible to plot general predictive observations. In GOMCIA–PLS1, graphical representations are more difficult. However, GOMCIA–PLS1 has some interest. With only a few steps, models are very well explained. The reason is that the data are projected onto orthogonal K- or I-dimensional spaces. Rapidly, the space spanned by X-variables is fully described.

Compare GOMCIA–PLS2 and 3. In the latter the super-components $\mathbf{t}_a$ take into account in their expression how close $\mathbf{X}_k^{(a-1)}$ and the set $\{\mathbf{Y}_n^{(a-1)}\}_n$ are. The more $\mathbf{t}_{k,a}$ are covaried with $\{\mathbf{Y}_n^{(a-1)}\}_n$, the more this is emphasized in $\mathbf{t}_a$. We can expect to see larger differences among the different $\mathbf{X}_k^{(a-1)}$.

Owing to these definitions, the A super-components $\{\mathbf{t}_a\}_a$

Copyright © 2003 John Wiley & Sons, Ltd.

Table I. ξ_a values for the different models in all the methods

	GOMCIA1	GOMCIA2	GOMCIA–PLS1	GOMCIA–PLS2	GOMCIA–PLS3
$\mathbf{X}_k$	$\mathbf{t}_{k,a}$	$\mathbf{t}_a = \sum_k \mathbf{t}_{k,a}$	$\mathbf{T}_a$	$\mathbf{t}_a = \sum_k \mathbf{t}_{k,a}$	$\sum_k \mu_{k,a}\mathbf{t}_{k,a}$
$\mathbf{Y}_n$	$\mathbf{u}_{n,a}$	$\mathbf{u}_a = \sum_n \mathbf{u}_{n,a}$	$\mathbf{T}_a$	$\mathbf{t}_a = \sum_k \mathbf{t}_{k,a}$	$\sum_k \mu_{k,a}\mathbf{t}_{k,a}$

are **D**-orthogonal and the A spaces $\{\mathbf{T}_a\}_a$ are too. Note that $\{\mathbf{t}_{k,a}\}_a$ are also **D**-orthogonal in GOMCIA–PLS1.

4.4. The models

We will detail the GOMCIA–PLS models. The GOMCIA models can be obtained in a similar way.

The orthogonalities quoted above imply that the models $\widehat{\mathbf{X}}_k^A$ and $\widehat{\mathbf{Y}}_n^A$ are the sum of A models explaining different part of $\mathbf{X}_k$ and $\mathbf{Y}_n$. Except for GOMCIA–PLS1, $\widehat{\mathbf{X}}_k^A$ and $\widehat{\mathbf{Y}}_n^A$ have rank A and are sums of partial models of rank 1. Setting $\mathcal{E}_a = \mathbf{t}_a$ for GOMCIA–PLS2 and 3 and $\mathcal{E}_a = \mathbf{T}_a$ for GOMCIA–PLS1, the models can be written as

$$\mathbf{Y}_n = \widehat{\mathbf{Y}}_n^A + \mathbf{Y}_n^{(A)}$$

$$\mathbf{X}_k = \widehat{\mathbf{X}}_k^A + \mathbf{X}_k^{(A)}$$

where $\widehat{\mathbf{Y}}_n^A = \Sigma_{a=1}^A \mathbf{P}_{\mathcal{E}_a}\mathbf{Y}_n, \widehat{\mathbf{X}}_k^A = \Sigma_{a=1}^A \mathbf{P}_{\mathcal{E}_a}\mathbf{X}_k$ and errors $\mathbf{Y}_n^{(A)}$ and $\mathbf{X}_k^{(A)}$ are non-correlated with respectively $\widehat{\mathbf{Y}}_n^A$ and $\widehat{\mathbf{X}}_k^A$.

Because of the orthogonality of spaces $\{\mathcal{E}_a\}_a$, the multiple correlation coefficient $R_n^2(a)$ is increasing with a and

$$R_n^2(A) = \sum_{a=1}^{A} \frac{\|\mathbf{P}_{\mathcal{E}_a}\mathbf{Y}_n\|_{\mathbf{D}}^2}{\|\mathbf{Y}_n\|_{\mathbf{D}}^2}$$

Analogously for the predictors,

$$R_k^2(A) = \sum_{a=1}^{A} \frac{\|\mathbf{P}_{\mathcal{E}_a}\mathbf{X}_k\|_{\mathbf{D}}^2}{\|\mathbf{X}_k\|_{\mathbf{D}}^2}$$

This helps to choose the number A of components. We can also choose to look at $R_{\mathbf{Y}}^2(A)$, which represents the variation captured by the model $\widehat{\mathbf{Y}}^A = [\widehat{\mathbf{Y}}_1^A|\ldots|\widehat{\mathbf{Y}}_N^A]$:

$$R_{\mathbf{Y}}^2(A) = \sum_{a=1}^{A} \frac{\sum_{n=1}^{N} \|\mathbf{P}_{\mathcal{E}_a}\mathbf{Y}_n\|_{\mathbf{D}}^2}{\sum_{n=1}^{N} \|\mathbf{Y}_n\|_{\mathbf{D}}^2}$$

However, we prefer $R_n^2(A)$ because it shows how the nth group of variables is modelled.

$\widehat{\mathbf{Y}}_n^A$ can also be written as a linear combination of the initial variables of $\underline{\mathbf{X}}$ (see Appendix IV for proof): $\widehat{\mathbf{Y}}_n^A = \mathbf{X}\widehat{\boldsymbol{\beta}}_n^A$, where $\widehat{\boldsymbol{\beta}}_n^A = \Sigma_{a=1}^A \boldsymbol{\delta}_a(\mathcal{E}_a'\mathbf{D}\mathcal{E}_a)^{-1}\mathcal{E}_a'\mathbf{D}\mathbf{Y}_n$, with

$$\boldsymbol{\delta}_1 = \boldsymbol{\alpha}_1$$

$$\boldsymbol{\delta}_a = \boldsymbol{\alpha}_a - \sum_{l=1}^{a-1} \boldsymbol{\delta}_l(\mathcal{E}_l'\mathbf{D}\mathcal{E}_l)^{-1}\mathcal{E}_l'\mathbf{D}\mathbf{X}\boldsymbol{\alpha}_a \quad \forall a > 1$$

and

$$\boldsymbol{\alpha}_a = \begin{cases} \mathbf{Diag}(\mathbf{a}_{k,a}) & \text{in GOMCIA--PLS1} \\ \begin{bmatrix} \mathbf{a}_{1,a} \\ \vdots \\ \mathbf{a}_{K,a} \end{bmatrix} & \text{in GOMCIA--PLS2} \\ \begin{bmatrix} \mu_{1,a}\mathbf{a}_{1,a} \\ \vdots \\ \mu_{K,a}\mathbf{a}_{K,a} \end{bmatrix} & \text{in GOMCIA--PLS3} \end{cases}$$

Writing the model in this way makes prediction calculations easy. Take an additional sample $\mathbf{x}_{i'}$ of length $\Sigma_{k=1}^K J_k$; then the predicted responses values are $\mathbf{y}_n^A(i') = \mathbf{x}_{i'}'\widehat{\boldsymbol{\beta}}_n^A$, where $\widehat{\boldsymbol{\beta}}_n^A$ have already been calculated with the training set. The knowledge of the explicit $\widehat{\boldsymbol{\beta}}_n^A$ gives an easy and fast way to compute the prediction errors and then perform cross-validation (Section 5.2).

In GOMCIA methods the component orthogonalities lead, with similar calculations, to models looking like $\mathbf{Z} = \widehat{\mathbf{Z}}^A + \mathbf{Z}^{(A)}$ and $\widehat{\mathbf{Z}}^A = \Sigma_{a=1}^A \mathbf{P}_{\xi_a}\mathbf{Z} = \mathbf{Z}\widehat{\boldsymbol{\beta}}^A$, where $(\mathbf{Z}, \widehat{\boldsymbol{\beta}}^A)$ is either $(\mathbf{X}_k, \widehat{\boldsymbol{\beta}}_k^A)$ or $(\mathbf{Y}_n, \widehat{\boldsymbol{\beta}}_n^A)$. The difference between all the models (GOMCIA1 and 2 and GOMCIA–PLS1, 2 and 3) lies in the value of ξ_a listed in Table I.

The coefficient matrices of the X-models in GOMCIA1 are written as $\widehat{\boldsymbol{\beta}}_k^A = \Sigma_{a=1}^A \boldsymbol{\delta}_a^k(\mathbf{t}_{k,a}'\mathbf{D}\mathbf{t}_{k,a})^{-1}\mathbf{t}_{k,a}'\mathbf{D}\mathbf{X}_k$, where

$$\boldsymbol{\delta}_1^k = \mathbf{a}_{k,1}$$

$$\boldsymbol{\delta}_a^k = \mathbf{a}_{k,a} - \sum_{l=1}^{a-1} \boldsymbol{\delta}_l^k(\mathbf{t}_{k,l}'\mathbf{D}\mathbf{t}_{k,l})^{-1}\mathbf{t}_{k,l}'\mathbf{D}\mathbf{X}_k\mathbf{a}_{k,a} \quad \forall a > 1$$

In GOMCIA2, $\widehat{\boldsymbol{\beta}}_k^A = \sum_{a=1}^{A} \boldsymbol{\delta}_a(\mathbf{t}_a'\mathbf{D}\mathbf{t}_a)^{-1}\mathbf{t}_a'\mathbf{D}\mathbf{X}_k$, with

$$\boldsymbol{\delta}_1 = \boldsymbol{\alpha}_1$$

$$\boldsymbol{\delta}_a = \boldsymbol{\alpha}_a - \sum_{l=1}^{a-1} \boldsymbol{\delta}_l(\mathbf{t}_l'\mathbf{D}\mathbf{t}_l)^{-1}\mathbf{t}_l'\mathbf{D}\mathbf{X}\boldsymbol{\alpha}_a \quad \forall a > 1$$

and $\boldsymbol{\alpha}'_a = [\mathbf{a}'_{1,a}\ldots\mathbf{a}'_{K,a}]$.

To obtain the coefficients matrices of the Y-models, replace in the equations for the X-models, $\mathbf{X}_k$ with $\mathbf{Y}_n$, $\mathbf{X}$ with $\mathbf{Y}$, $\mathbf{a}_{k,a}$ with $\mathbf{b}_{n,a}$, $\mathbf{t}_a$ with $\mathbf{u}_a$, and $\mathbf{t}_{k,a}$ with $\mathbf{u}_{n,a}$.

As in GOMCIA–PLS methods, we can perform cross-validation to choose the number A of steps to compute in order to have good predictive models (see Section 5.2).

4.5. Algorithm

This algorithm consists in fixing the weighting vectors $\mathbf{b}_n$ and repeating the computation of objects until the criterion C converges (or until $\mathbf{u}_n$ converges).

Copyright © 2003 John Wiley & Sons, Ltd.

Algorithm for the first step: $a = 1$

Step 0: initialization
*$\mathbf{X}_k$, $k = 1,\ldots,K$, and $\mathbf{Y}_n$, $n = 1,\ldots,N$, are supposed correctly preprocessed.
Fix $\mathbf{b}_n$ as the first left singular vector of unit length of $\mathbf{Y}'_n\mathbf{DX}$.
(It is also possible to fix it to $(1/\sqrt{N_m})\ \mathbf{1}_{N_m}$.)

$\mathbf{a}_k = \mathbf{X}'_k\mathbf{DYb}$, where $\mathbf{b}' = [\mathbf{b}'_1|\ldots|\mathbf{b}'_N]$.
$\mathbf{a}_k = \mathbf{a}_k/\|\mathbf{a}_k\|$.
$\mu_k = \mathbf{b}'\mathbf{Y}'\mathbf{DX}_k\mathbf{a}_k = \Sigma_{n=1}^N \mathbf{u}'_n\mathbf{Dt}_k$.
$\lambda_n = \mathbf{a}'\mathbf{X}'\mathbf{DY}_n\mathbf{b}_n = \Sigma_{k=1}^K \mathbf{u}'_n\mathbf{Dt}_k$, where $\mathbf{a}' = [\mathbf{a}'_1 \mid \ldots \mid \mathbf{a}'_K]$.
$C_{\text{opt}} = \Sigma_k\, \mu_k$.
$iter = 0$, index of iterations.
Step 1: current step
$iter = iter + 1$,
$\mathbf{b}_n = \mathbf{Y}'_n\mathbf{DXa}$.
$\mathbf{b}_n = \mathbf{b}_n/\|\mathbf{b}_n\|$.
$\mathbf{a}_k = \mathbf{X}'_k\mathbf{DYb}$, where $\mathbf{b}' = [\mathbf{b}'_1|\ldots|\mathbf{b}'_N]$.
$\mathbf{a}_k = \mathbf{a}_k/\|\mathbf{a}_k\|$.
$\mu_k = \mathbf{b}'\mathbf{Y}'\mathbf{DX}_k\mathbf{a}_k = \Sigma_{n=1}^N \mathbf{u}'_n\mathbf{Dt}_k$.
$\lambda_n = \mathbf{a}'\mathbf{X}'\mathbf{DY}_n\mathbf{b}_n = \Sigma_{k=1}^K \mathbf{u}'_n\mathbf{Dt}_k$, where $\mathbf{a}' = [\mathbf{a}'_1|\ldots|\mathbf{a}'_K]$.
$C_{\text{opt}}^{\text{new}} = \Sigma_k \mu_k$.
Step 2: convergence test
If $C_{\text{opt}}^{\text{new}} - C_{\text{opt}} < 10^{-6}$ or if $iter = 50$, go to Step 3.
Else set $C_{\text{opt}} = C_{\text{opt}}^{\text{new}}$ and go to Step 1.
Step 3: end
$n = 1,\ldots,N$, $\mathbf{u}_n = \mathbf{Y}_n\mathbf{b}_n$,
$k = 1,\ldots,K$, $\mathbf{t}_k = \mathbf{X}_k\mathbf{a}_k$.

Algorithm for step a: $a > 1$

a-order solutions $\mathbf{a}_{k,a}$ and $\mathbf{b}_{n,a}$ are calculated with this algorithm by omitting the line labelled with an asterisk (*) and by replacing elsewhere $\mathbf{X}_k$ and $\mathbf{Y}_n$ respectively with $\mathbf{X}_k^{(a-1)}$ and $\mathbf{Y}_n^{(a-1)}$ defined in Section 4.3.

Note that in Step 2 we test $C_{\text{opt}}^{\text{new}} - C_{\text{opt}} < 10^{-6}$. It can be proved (see Appendix II) that the criterion increases with the number of iterations and converges. Thus, we are ensured that the solutions we get are local (or strict local) maxima.

In all the examples we have studied, this algorithm converges in less than 30 iterations.

All the methods were programmed in S-Plus® language [34]. All the functions are available on request from the first author (myrtille@myrtille.org). However, the GOMCIA(–PLS) algorithm needs to be modified to cater for missing values in the data.

5. GOMCIA AND GOMCIA–PLS: PRACTICE AND COMMENTS

5.1. Block selection in GOMCIA–PLS

Recall that we consider three-way tables as a set of several matrices measured on the same observations. In this situation we are interested in keeping together variables in the same matrix, i.e. we do not want to consider variables separately but in groups. In multiple regression, choosing the relevant variables for the model is important. Here, selecting the influential X-blocks comes naturally. This problem has never really been studied. We propose several ways of making this selection.

The first clue is to look at the absolute values of the regression coefficients of the models. This method is perhaps inadequate in most cases, but this is the first approach. Here the influence of the kth group of predictors on the modelling of $\mathbf{Y}_n$ is defined as follows. Consider $\widehat{\mathbf{Y}}_n^A = \Sigma_{k=1}^K \mathbf{X}_k\widehat{\boldsymbol{\beta}}_n^A[k]$, with $\widehat{\boldsymbol{\beta}}_n^A[k]$ the kth block of $\widehat{\boldsymbol{\beta}}_n^A$ of size $J_k \times M_n$. Because $\|\widehat{\boldsymbol{\beta}}_n^A\|^2 = \Sigma_{k=1}^K \|\widehat{\boldsymbol{\beta}}_n^A[k]\|^2$, we suggest using the following coefficient to compare the influence of $\mathbf{X}_k$ on the model $\widehat{\mathbf{Y}}_n^A$:

$$B_n^2(k) = \frac{\|\widehat{\boldsymbol{\beta}}_n^A[k]\|^2}{\|\widehat{\boldsymbol{\beta}}_n^A\|^2} \times 100 \tag{10}$$

In the case where $A = 1$, owing to the unit length of $\mathbf{a}_{k,1}$, $B_n^2(k)$ is constant for all k in GOMCIA–PLS2 and equals $100/K$; and in GOMCIA–PLS3, $B_n^2(k)$ is proportional to $\mu_{k,1}^2$. These particular results show that this coefficient may be not adequate for GOMCIA–PLS2. In practice, we noticed that it seems to be related to the partial correlation in GOMCIA–PLS1 (see Section 6).

For GOMCIA–PLS1 and 3 we have developed the block importance in the prediction, *BIP*, which generalizes in some way the *VIP* coefficient used in usual PLS regression, and also a *BIPrv* for GOMCIA–PLS1 based on the vector correlation coefficient RV [35]:

$$BIP_k^n(A) = \sqrt{K\sum_{a=1}^{A} \frac{\mu_{k,a}\|\mathbf{P}_{\mathbf{t}_a}\mathbf{Y}_n\|_{\mathbf{D}}^2}{\sum_k \mu_{k,a}\|\mathbf{P}_{\mathbf{t}_1,\ldots,\mathbf{t}_A}\mathbf{Y}_n\|_{\mathbf{D}}^2}} \quad \text{for GOMCIA–PLS3} \tag{11}$$

$$BIP_k^n(A) = \sqrt{K\sum_{a=1}^{A} \frac{\|\mathbf{t}_{k,a}\|_{\mathbf{D}}^2\|\mathbf{P}_{\mathbf{T}_a}\mathbf{Y}_n\|_{\mathbf{D}}^2}{\|\mathbf{T}_a\|_{\mathbf{D}}^2\|\mathbf{P}_{\mathbf{T}_1,\ldots,\mathbf{T}_A}\mathbf{Y}_n\|_{\mathbf{D}}^2}} \quad \text{for GOMCIA–PLS1} \tag{12}$$

$$BIPrv_k^n(A) = K\sum_{a=1}^{A} \frac{\|\mathbf{t}_{k,a}\|_{\mathbf{D}}^2 RV(\widehat{\mathbf{Y}}_n^A, \mathbf{t}_{k,a})}{\sqrt{\sum_a varV(\mathbf{T}_a)}RV(\widehat{\mathbf{Y}}_n^A, \mathbf{T}_1, \ldots, \mathbf{T}_A)} \quad \text{for GOMCIA–PLS1} \tag{13}$$

$varV(\mathbf{X})$ denotes the vector variance and is equal to the squared Hilbert–Schmidt norm, i.e. to $tr(\mathbf{XX'DXX'D})$. $covV(\mathbf{X},\mathbf{Y})$ denotes the vector covariance between $\mathbf{X}$ and $\mathbf{Y}$: $covV(\mathbf{X},\mathbf{Y}) = tr(\mathbf{XX'DYY'D})$. The vector correlation between $\mathbf{X}$ and $\mathbf{Y}$ is equal to $RV(\mathbf{X},\mathbf{Y}) = covV(\mathbf{X},\mathbf{Y})/\sqrt{varV(\mathbf{X})varV(\mathbf{Y})}$.

Consider the *BIP* coefficient for GOMCIA–PLS3 (respectively that for GOMCIA–PLS1). $\|\mathbf{P}_{\mathbf{t}_a}\mathbf{Y}_n\|_{\mathbf{D}}^2/\|\mathbf{P}_{\mathbf{t}_1,\ldots,\mathbf{t}_A}\mathbf{Y}_n\|_{\mathbf{D}}^2$ (respectively $\|\mathbf{P}_{\mathbf{T}_a}\mathbf{Y}_n\|_{\mathbf{D}}^2/\|\mathbf{P}_{\mathbf{T}_1,\ldots,\mathbf{T}_A}\mathbf{Y}_n\|_{\mathbf{D}}^2$) measures the proportion of the explained $\mathbf{Y}_n$ variation due to the component $\mathbf{t}_a$ (respectively $\mathbf{T}_a$), and $\mu_{k,a}/\Sigma_k\,\mu_{k,a}$ (respectively $\|\mathbf{t}_{k,a}\|_{\mathbf{D}}^2/\|\mathbf{T}_a\|_{\mathbf{D}}^2$) measures the contribution of $\mathbf{t}_{k,a}$ to the construction of $\mathbf{t}_a$ (respectively $\mathbf{T}_a$).

The same idea applies to the construction of the *BIPrv* coefficient, but with the use of the vector correlation instead of the variance.

Proposition
$\Sigma_{k=1}^K (BIP_k^n(l))^2 = K$ and $\Sigma_{k=1}^K BIPrv_k^n(l) = K$ for all $l = 1,\ldots,A$.

Copyright © 2003 John Wiley & Sons, Ltd.

Then the block $\mathbf{X}_k$ is more important than the others in the modelling of $\mathbf{Y}_n$ at step A if $(BIP_k^n(A))^2$ and/or $BIPrv_k^n(A)$ are much greater than unity.

Proof

For GOMCIA–PLS3,

$$\sum_k (BIP_k^n(A))^2 = K \sum_k \sum_{a=1}^{A} \frac{\mu_{k,a} \|\mathbf{P}_{\mathbf{t}_a} \mathbf{Y}_n\|_{\mathbf{D}}^2}{\Sigma_k \mu_{k,a} \|\mathbf{P}_{\mathbf{t}_1,\ldots,\mathbf{t}_A} \mathbf{Y}_n\|_{\mathbf{D}}^2} \tag{14}$$

$$= K \sum_{a=1}^{A} \frac{\|\mathbf{P}_{\mathbf{t}_a} \mathbf{Y}_n\|_{\mathbf{D}}^2}{\|\mathbf{P}_{\mathbf{t}_1,\ldots,\mathbf{t}_A} \mathbf{Y}_n\|_{\mathbf{D}}^2} \tag{15}$$

$$= K, \text{ because the super-components are } \mathbf{D}\text{-orthogonal} \tag{16}$$

The proof for the GOMCIA–PLS1 *BIP* is similar. As the matrices $\mathbf{T}_a$ are $\mathbf{D}$-orthogonal, $\Sigma_a \|\mathbf{P}_{\mathbf{T}_a} \mathbf{Y}_n\|_{\mathbf{D}}^2 = \|\mathbf{P}_{\mathbf{T}_1,\ldots,\mathbf{T}_A} \mathbf{Y}_n\|_{\mathbf{D}}^2$. Then

$$\sum_k (BIPrv_k^n(A)) = K \sum_{a=1}^{A} \frac{\sum_k \overbrace{\|\mathbf{t}_{k,a}\|_{\mathbf{D}}^2}^{var(\mathbf{t}_{k,a})=\sqrt{varV(\mathbf{t}_{k,a})}} RV(\mathbf{Y}_n^A, \mathbf{t}_{k,a})}{\sqrt{\sum_a varV(\mathbf{T}_a)} RV(\mathbf{Y}_n^A, [\mathbf{T}_1|\ldots|\mathbf{T}_a])} \tag{17}$$

$$= K \frac{\sum_a \sum_k covV(\mathbf{Y}_n^A, \mathbf{t}_{k,a})}{\sqrt{\sum_a varV(\mathbf{T}_a)} RV(\mathbf{Y}_n^A, [\mathbf{T}_1|\ldots|\mathbf{T}_a]) \sqrt{varV(\mathbf{Y}_n^A)}} \tag{18}$$

$$= K \frac{covV(\mathbf{Y}_n^A, [\mathbf{T}_1|\ldots|\mathbf{T}_a])}{\underbrace{\sqrt{\sum_a varV(\mathbf{T}_a)}}_{\sqrt{varV([\mathbf{T}_1|\ldots|\mathbf{T}_a])}} RV(\mathbf{Y}_n^A, [\mathbf{T}_1|\ldots|\mathbf{T}_a]) \sqrt{varV(\mathbf{Y}_n^A)}} \tag{19}$$

$$= K \frac{RV(\mathbf{Y}_n^A, [\mathbf{T}_1|\ldots|\mathbf{T}_a])}{RV(\mathbf{Y}_n^A, [\mathbf{T}_1|\ldots|\mathbf{T}_a])} = K \tag{20}$$

Note that we did not propose a *BIP* coefficient for GOMCIA–PLS2. The reason is that we cannot express a contribution of the $\mathbf{X}_k$ block to the construction of $\mathbf{t}_a = \Sigma_{k=1}^K \mathbf{t}_{k,a}$. All the $\mathbf{t}_{k,a}$ are weighted by 1 and we cannot use $\|\mathbf{a}_{k,a}\|$ because it is equal to 1 too. Therefore a *BIP* coefficient built like the previous ones would lead to a *BIP* coefficient that does not depend on k. Then two conclusions are possible: either we can state that all the $\mathbf{X}_k$ blocks have the same importance in GOMCIA–PLS2 in all the steps; or we can state that this kind of *BIP* coefficient is not appropriate to find the important $\mathbf{X}_k$ blocks. We are actually working on another way to carry out block selections using permutation tests.

5.2. Validation

To validate the models, cross-validation can be adapted to GOMCIA–PLS methods by using *PRESS* (predictive residual sum of squares). Let $\mathbf{E}_{n,a}^y(i) = \mathbf{Y}_n(i) - \mathbf{X}(i)\widehat{\boldsymbol{\beta}}_n^a(i)$, where $\widehat{\boldsymbol{\beta}}_n^a(i)$ is the coefficient matrix obtained by processing GOMCIA–PLS on the data without the ith observation, and $\mathbf{X}(i)$ and $\mathbf{Y}_n(i)$ are respectively the ith row (observation) of $\mathbf{X}$ and $\mathbf{Y}_n$. We define $PRESS_n(a) = (1/I)\Sigma_{i=1}^I \|\mathbf{E}_{n,a}^y(i)\|^2$. This is what we call the *PRESS* per group. $PRESS(a) = \Sigma_{n=1}^N PRESS_n(a)$ is the total prediction sum of squares. We choose the value of A that minimizes *PRESS*. The *PRESS* per group is useful for inspecting which groups are well modelled with this number of components. We can also define the *PRESS* per variable in group n: $PRESS_n^m(a) = (1/I)\Sigma_{i=1}^I \mathbf{E}_{n,a}^y(i,m)^2$, where $\mathbf{E}_{n,a}^y(i,m)$ is the element (i, m) of $\mathbf{E}_{n,a}^y$. The relation between $PRESS_n(a)$ and $PRESS_n^m(a)$ is $PRESS_n(a) = \Sigma_{m=1}^{M_n} PRESS_n^m(a)$. It is also possible to define Q_n^2, a cross-validated R^2, for each $\mathbf{Y}_n$ and a global Q^2 in the same way.

Consider now the GOMCIA methods. $\mathbf{E}_{k,a}^x(i)$ and $\mathbf{E}_{n,a}^y(i)$ can be defined similarly according to the GOMCIA models (see Section 4.4). Then $PRESS_n(a) = (1/I)\Sigma_{i=1}^I \|\mathbf{E}_{n,a}^y(i)\|^2$ and $PRESS_k(a) = (1/I)\Sigma_{i=1}^I \|\mathbf{E}_{k,a}^x(i)\|^2$. The total prediction sum of squares *PRESS* at step a is given by $PRESS(a) = \Sigma_{n=1}^N PRESS_n(a) + \Sigma_{k=1}^K PRESS_k(a)$. A is chosen as the value minimizing *PRESS*.

When I is very large, more than one observation can be used as test set in the cross-validation. Another way to choose A is to keep the first A components which make $R_{\mathbf{Y}}^2$ grow meaningfully. This is appropriate when we want to explain the data the best. When the purpose is prediction, cross-validation is a better way to validate parsimonious models.

5.3. General comments and particular cases

Some comments are appropriate.

- It is possible to do the same kind of graphics as in PLS. A paper about graphical representations and their validity in multiblock methods is in preparation.
- When $N = 1$ and $K = 1$, GOMCIA1 is equivalent to canonical PLS analysis [36] and, of course, GOMCIA–PLS1 and 2 are equivalent to PLS regression.
- In GOMCIA–PLS2 and 3, $\{\mathbf{t}_a\}_a$ makes a basis of the space spanned by $\mathbf{X}$. Hence A cannot exceed the rank of $\mathbf{X}$, i.e. $A \leq min(I, \Sigma_k J_k)$.
- Data are assumed to be centred for $\mathbf{D}$. Thus, if $I < K$, $Im\mathbf{X}$ has at best dimension $I - 1$. Under such circumstances, performing GOMCIA–PLS1 is not of great interest, because the dimension of the space spanned by $\mathbf{T}_1$ would

Copyright © 2003 John Wiley & Sons, Ltd.

be certainly $I-1$. Then one cannot find K components $\mathbf{t}_{k,2}$ to make $\mathbf{T}_2$ orthogonal with $\mathbf{T}_1$.

- In the case where all $\mathbf{J}_k = 1, \widehat{\mathbf{Y}}_n^1$ in GOMCIA–PLS1 corresponds to the multiple linear regression model: $\mathbf{P}_{[\mathbf{x}_1|\ldots|\mathbf{x}_K]}\mathbf{Y}_n$.
- Let us see now the common case where we have a single variable $\mathbf{y}$ to predict. Then our algorithm is neither an iterative one nor an eigenvalue problem resolution. The solutions are explicit, as in NIPALS. In addition, the components $\mathbf{t}_{k,1}$ in GOMCIA–PLS are equal to the first predictive components (usually denoted $\mathbf{t}$ in the PLS literature) stemming from K PLS regressions—PLS regressions of $\mathbf{y}$ on $\mathbf{X}_k$ for $k = 1,\ldots,K$. Therefore in this case, in GOMCIA–PLS1, $\mathbf{T}_1$ is built with the most predictive components, which makes model $\widehat{\mathbf{y}}^1$ explain $\mathbf{y}$'s inertia well. In the second step, $a = 2$, GOMCIA–PLS1 searches for information existing in groups of predictors which has not been pointed out by $\mathbf{t}_{k,1}$. In only a few steps, $R^2(a)$ becomes large. In this situation, GOMCIA–PLS3 models are equivalent to MPLS models [29]. Actually, weights $\mu_{k,a}$ in $\mathbf{t}_a$ cancel out, because $\mathbf{t}_{k,a} = (1/\mu_{k,a})\mathbf{X}_k\mathbf{X}'_k\mathbf{D}\mathbf{y}$. Then $\mathbf{t}_a$ and the component $\mathbf{t}_a^*$ from MPLS are proportional and span the same space. As a consequence, the two models are identical.
- Models from GOMCIA–PLS2 can be written similarly to those of MPLS [29]. Suppose the data are three-way; let $\mathbf{p}_{n,a} = \mathbf{Y}'_n\mathbf{D}\mathbf{t}_a(\mathbf{t}'_a\mathbf{D}\mathbf{t}_a)^{-1}$ and $\mathbf{P}_a = [\mathbf{p}_{1,a}|\ldots|\mathbf{p}_{N,a}]$. Models from the two methods can be written in the same way: $\widehat{\mathbf{Y}}^A = \Sigma_{a=1}^A \mathbf{t}_a \otimes \mathbf{P}_a$, where $\otimes$ is the tensor product.
- A comment about differences between MBPLS with super-score deflation and GOMCIA–PLS2 and 3 is necessary. A way to see that these methods are different is to note that the GOMCIA–PLS2 and 3 super-components $\mathbf{t}$ do not satisfy an eigenvalue problem, while those of MBPLS do [26], because MBPLS is equivalent to the standard PLS regression. The GOMCIA–PLS weight vectors are calculated to optimize the criterion $cov(\Sigma_k\ \mathbf{X}_k\mathbf{a}_k,\ \Sigma_n\ \mathbf{Y}_n\mathbf{b}_n)$ with length one, while the MBPLS ones are deduced from the standard PLS weight vectors ($\mathbf{w}$ and $\mathbf{q}$ with length one) [27] which maximize the criterion $cov(\Sigma_k\ \mathbf{X}_k\mathbf{w}[k],\ \Sigma_n\ \mathbf{Y}_n\mathbf{q}[n])$, where $\mathbf{w}[k]$ and $\mathbf{q}[n]$ are the parts of $\mathbf{w}$ and $\mathbf{q}$ that correspond to the blocks $\mathbf{X}_k$ and $\mathbf{Y}_n$. These two optimization problems are not identical, so their solutions are different.

5.4. Comments about deflation

Deflation can lead to interpretation difficulties. Some papers deal with these problems in particular in the MBPLS method. Westerhuis and Coenegracht [25] state that the block score deflation of the X-block leads to a lower fit of the response models and encourages the super-score deflation of the X-blocks and Y-blocks. They also state that performing the MBPLS method with these deflations is exactly the same as performing a standard PLS on the data arranged in large blocks $\mathbf{X}$ and $\mathbf{Y}$. The proof can be found in Reference [26]. Recently, Westerhuis and Smilde [27] dealt with the problem of the super-score deflation which mixes the information in the X-block. They state that, after the first step, some information of the other data blocks is put in $\mathbf{X}_k$ and thus the score $\mathbf{t}_{k,2}$ does not explain solely its block $\mathbf{X}_k$. Therefore they propose deflating the Y-block only (on the super-score) and performing the algorithm (standard PLS in their case) on the initial X-block at each step, and orthogonalizing the super-scores in order that they explain different variation.

In GOMCIA1, block score deflations are made: each data block is deflated on its block score $\mathbf{t}_k$ or $\mathbf{u}_n$. The information from the blocks is not mixed up. Each block is explained with 'its' scores. In GOMCIA2, super-score deflations are made. Here the interpretation is more complicated, because the information is mixed up in the different X-blocks and Y-blocks, starting at the second step. However, to obtain different-order scores, we have to perform a deflation; if we do not, some supplementary constraints have to be imposed when C is optimized, such as non-correlation between $\mathbf{t}_1$ and the following super-scores $\mathbf{t}_a$, and between $\mathbf{u}_1$ and the following $\mathbf{u}_a$. Such a problem is much more difficult to solve. Thus, if one wants to study the relationship between two sets of blocks, GOMCIA1 is perhaps easier to interpret. In GOMCIA–PLS1 there is no problem with deflation: all the X-blocks and Y-blocks are deflated on the super-matrix $\mathbf{T}_a = [\mathbf{t}_{1,a}\ldots\mathbf{t}_{K,a}]$. That is to say, all the information brought by the predictive scores is removed from each data block. We are assured that the following scores will not explain the same thing.

On the other hand, GOMCIA–PLS2 and 3 are super-score deflation methods. Here, effectively, the problem is the same as in MBPLS [27]. It is thinkable to only deflate the Y-blocks and orthogonalize the super-scores $\mathbf{t}_a$ just before the deflation step. This will give the same results in terms of percentage of explained variance.

6. APPLICATION ON REAL DATA

All programs and computations were performed running S-Plus 5.1® under a linux system. We have chosen to centre the data across the first mode and scale them also across the first mode, so that all variables have unit norm.

In the following we call 'situation 1' the case where the data are considered as a set of five matrices of size 15×3, and 'situation 2' where the data are considered as a set of three matrices of size 15×5. In both situations we performed NPLS, HPLS and GOMCIA–PLS1, 2 and 3. Tables II and III give the $R_{\mathbf{y}}^2$ coefficient in per cent and the *PRESS* value for the y-model until the 11th step, because $\mathbf{X}$ has rank 11.

Of course, the results from NPLS do not depend on the data arrangement, because it considers them as a cube, and the three modes are modelled simultaneously. The results from GOMCIA–PLS3 correspond to those we would obtain if we performed (M)PLS or MBPLS.

All models capture a great amount of variation of the response from the first step. However, we notice that $\%R^2$ from HPLS grows slower than the others in both situations. This is not surprising given that the X-block scores are chosen to have affinity with the super-score $\mathbf{t}$ and not with the super-score $\mathbf{u}$ [26]. Consider all the methods except GOMCIA–PLS1. Regarding $\%R^2$ only, we would choose only one score. However, the cross-validation indicates that the best predictive model is obtained with eight scores, except for HPLS where it is obtained with 10 scores. As for

Copyright © 2003 John Wiley & Sons, Ltd.

Table II. Situation 1: $\%R^2$ represents R^2 coefficient $\times$ 100; *PRESS* is the *PRESS* value for *y*-model with $\underline{\mathbf{X}}$ as a set of five matrices. Bold values are the smallest *PRESS*

		Step *a*										
		1	2	3	4	5	6	7	8	9	10	11
NPLS	$\%R^2$	82.88	92.32	98.03	98.21	98.45	98.50	98.55	98.97	99.09	99.10	99.12
	PRESS	0.31	0.29	0.25	0.25	0.22	0.19	0.16	**0.13**	0.14	0.13	0.14
HPLS	$\%R^2$	66.20	69.94	71.16	72.37	84.92	91.73	96.10	96.10	98.24	99.11	99.12
	PRESS	0.81	0.83	0.56	0.59	0.40	0.40	0.35	0.21	0.12	**0.10**	0.11
GOMCIA–PLS1	$\%R^2$	97.99	98.40	99.12	99.12	99.12	99.12	99.12	99.12	99.12	99.12	99.12
	PRESS	0.15	0.12	**0.11**	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.11
GOMCIA–PLS2	$\%R^2$	68.99	87.30	91.90	95.07	97.91	98.14	98.23	98.69	99.09	99.11	99.12
	PRESS	0.88	0.55	0.51	0.37	0.31	0.28	0.22	**0.07**	0.07	0.10	0.11
GOMCIA–PLS3	$\%R^2$	89.64	97.48	97.94	98.19	98.34	98.60	98.84	99.04	99.11	99.11	99.12
	PRESS	0.33	0.23	0.19	0.17	0.15	0.13	0.11	**0.09**	0.10	0.09	0.09

Table III. Situation 2: $\%R^2$ represents R^2 coefficient $\times$ 100; *PRESS* is the *PRESS* value for *y*-model with $\underline{\mathbf{X}}$ as a set of three matrices. Bold values are the smallest *PRESS*

		Step *a*										
		1	2	3	4	5	6	7	8	9	10	11
NPLS	$\%R^2$	82.88	92.32	98.03	98.22	98.45	98.50	98.55	98.97	99.09	99.11	99.12
	PRESS	0.31	0.29	0.25	0.25	0.22	0.19	0.16	**0.13**	0.14	0.13	0.14
HPLS	$\%R^2$	61.21	64.98	79.27	85.87	86.06	86.49	92.20	97.36	97.43	99.11	99.12
	PRESS	0.51	0.39	0.4	0.34	0.3	0.29	0.24	0.17	0.14	**0.08**	0.09
GOMCIA–PLS1	$\%R^2$	91.15	98.96	99.11	99.12	99.12	99.12	99.12	99.12	99.12	99.12	99.12
	PRESS	0.33	**0.17**	0.20	0.44	0.44	0.44	0.44	0.44	0.44	0.44	0.44
GOMCIA–PLS2	$\%R^2$	90.39	97.01	97.71	97.90	98.15	98.49	98.72	99.01	99.11	99.11	99.12
	PRESS	0.35	0.23	0.19	0.17	0.15	0.13	0.10	**0.07**	0.09	0.10	0.12
GOMCIA–PLS3	$\%R^2$	89.64	97.48	97.94	98.19	98.34	98.60	98.84	99.04	99.11	99.11	99.12
	PRESS	0.33	0.23	0.19	0.17	0.15	0.13	0.11	**0.09**	0.10	0.09	0.09

GOMCIA–PLS1, the *PRESS* value tells us to stop at the second or the third step, as the case may be. This is altogether understandable, because deflations are made on orthogonal subspaces, each with dimension at best 5 (or 3), and $Im\mathbf{X}$ is totally described in a few steps.

These unsatisfactory results (large number of components needed) can be explained by the fact that only two *X*-variables are very correlated with $\mathbf{y}$, the others being nearly uncorrelated. These correlated variables are *lipophilicity* and *size* in site 3 (with respectively 0.95 and -0.62). Moreover, this means that the data are well modelled, but these models cannot be used for prediction. We recall that this example was extracted from the literature and this is the reason why we use it to illustrate our purpose. The fact that only two *X*-variables are really correlated with the response will be useful for showing that the *BIP* coefficient is a good way of selecting important *X*-blocks.

It is difficult to say if this data set is more appropriate for a multiway method than a multiblock one, but we get different information from the data with our multiblock methods.

6.1. First situation: X with size $(15 \times 3) \times 5$

Situation 1 is adequate to illustrate differences between GOMCIA–PLS2 and 3. In Figure 3, $\mu_{5,1}$ is close to zero and $\mu_{5,2}$ is very different in the two methods. Thus the super-score $\mathbf{t}_1$ from GOMCIA–PLS3 neglects site 5. As most information is found in $\mathbf{X}_3$ (and possibly $\mathbf{X}_4$), during the first step, $\mu_{3,1}$ and $\mu_{4,1}$ are the biggest coefficients. Thus the super-score $\mathbf{t}_1$ in GOMCIA–PLS3 can be considered as a linear combination of $\mathbf{X}_3$ (and possibly $\mathbf{X}_4$). In the second step, information is searched for among $\mathbf{X}_k$ deflations and in particular in the whole $\mathbf{X}_5$, but what was explained by $\mathbf{t}_{3,1}$ and $\mathbf{t}_{4,1}$. We can say that real information is not found in site 5 at step 1, so GOMCIA–PLS3 reinstates $\mathbf{X}_5$ 'entirely' in the second step. We can rightfully think that the same information is found in the first and second steps in site 5 ($r(\mathbf{t}_{5,1}, \mathbf{t}_{5,2}) = 0.998$).

On the other hand, the super-score $\mathbf{t}_1$ in GOMCIA–PLS2 is a linear combination of the whole set of $\mathbf{t}_{k,1}$ where the weights are equal to 1. That is to say, all the scores $\mathbf{t}_{k,a}$ have the same importance in $\mathbf{t}_a$, even if one block score is far from another. This is the case for $\mathbf{t}_{5,1}$. Then, in the second step, the unexplained part of $\mathbf{X}_5$ is less important than in GOMCIA–PLS3, and some information is searched for where there is no more real information to find. This explains the differences between the two methods.

However, we think it is surprising that the deflations on $\mathbf{t}_1$ from GOMCIA–PLS2 involve such results: here $r(\mathbf{y}^{(1)}, \mathbf{t}_{5,2}) = 0.63$, while the correlations between X_5-variables and $\mathbf{y}$ are equal to 0.08, 0.08 and -0.08.

In fact, $\mu_{k,a}$ represents the proximity between $\mathbf{y}^{(a-1)}$ and score k, but it does not represent $\mathbf{y}$'s explanation by $\mathbf{t}_{k,a}$.

Copyright © 2003 John Wiley & Sons, Ltd.

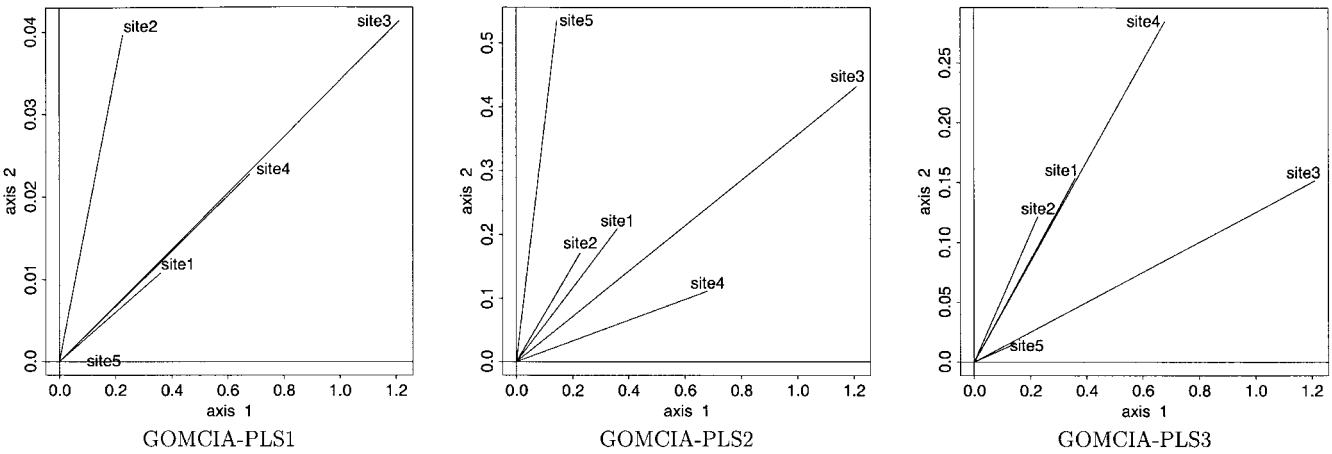

Figure 3. Situation 1: representation of $\mu_{k,1}$ against $\mu_{k,2}$ coefficients in the three GOMCIA–PLS methods. Recall that $\mu_{k,1} = cov(\mathbf{y},\mathbf{t}_{k,1})$ measure the proximity between response and *k*th *X*-block.

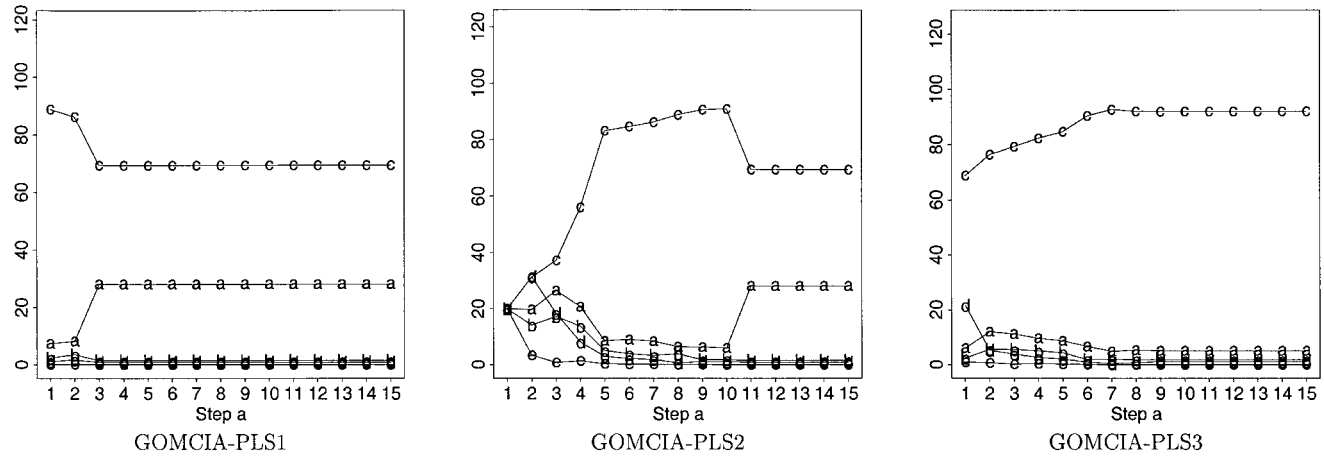

Figure 4. Situation 1: evolution of coefficients $B_n^2(k)$ with step *a*: 'a' is for site 1, 'b' for site 2, 'c' for site 3, 'd' for site 4 and 'e' for site 5.

However, we notice that GOMCIA–PLS3 arranges $\mu_{k,2}$ in the same order as the correlations between $\mathbf{y}^{(1)}$ and $\mathbf{t}_{k,2}$, while GOMCIA–PLS2 does not.

Figure 4 shows that are three methods make $\mathbf{X}_3$ the most influential.

The model coefficients in GOMCIA–PLS1 seem to correspond with the partial correlations between $\mathbf{y}^{(a-1)}$ and $\mathbf{t}_{k,a}$: in the first step, $r(\mathbf{y}, \mathbf{t}_{3,1} \,|\, \mathbf{t}_{1,1}, \mathbf{t}_{2,1}, \mathbf{t}_{4,1}, \mathbf{t}_{5,1}) = 0.98$ and $B_y^2(3)$ represents approximately 90% of B_y^2 (Equation (10)). Next comes $\mathbf{X}_4$ (with a partial correlation of 0.86). In step 2 the partial correlations are smaller (between 0.10 and 0.35); $\mathbf{t}_{3,2}$ is always the best. It is possible to verify step by step that the position of tables according to the importance given by Figure 4 for GOMCIA–PLS1 is very close to that given by the partial correlations. The *BIP* coefficient for GOMCIA–PLS3 (Section 5.1) is superior to 1 for $\mathbf{X}_3$, and the *BIPrv* coefficient for GOMCIA–PLS1 for $\mathbf{X}_3$ is superior to 3.40 in all steps. This means that such coeffecients, especially *BIPrv*, are efficient to target the most important *X*-block. Recall that the variables most correlated with $\mathbf{y}$ are in $\mathbf{X}_3$. Some authors look at $\%\mathbf{X}_k$ (the percentage of variance of $\mathbf{X}_k$ explained at each step) to explain the model. That is, if $R^2_{\mathbf{X}_k}$ is important at step 1, they say that $\mathbf{X}_k$ explains more of $\mathbf{Y}$ at this step. However, here, 100% of the variation of $\mathbf{X}_5$ is explained at the first step with GOMCIA–PLS1, and *BIPrv* for this table is close to zero! (the variables of this block are almost uncorrelated with the response). This result justifies that block selection has to be developed, because $R^2_{\mathbf{X}_k}$ are not sufficient. The *BIP* coefficients are the first step of a larger study.

6.2. Second situation: *X* with size (15 × 5) × 3

At step 1, results from GOMCIA–PLS1 and 2 are very close. Consequently, the subspace (with three dimensions) spanned by $\mathbf{t}_{1,1}$, $\mathbf{t}_{2,1}$ and $\mathbf{t}_{3,1}$ is nearly one-dimensional; that is, this subspace is flat enough. The same phenomenon is observed in the next steps. $\mathbf{t}_{k,1}$ bring almost the same information.

Note that here NPLS is less good (see R^2 coefficients in Table III) and GOMCIA–PLS3 is not the best. In fact, multiblock methods seem to give more information than usual PLS. There are no big differences among methods (Figure 5) here, because the three *X*-blocks are not significantly different. Thus the results from GOMCIA–PLS2 and 3 are very similar. The same remark as before can

Copyright © 2003 John Wiley & Sons, Ltd.

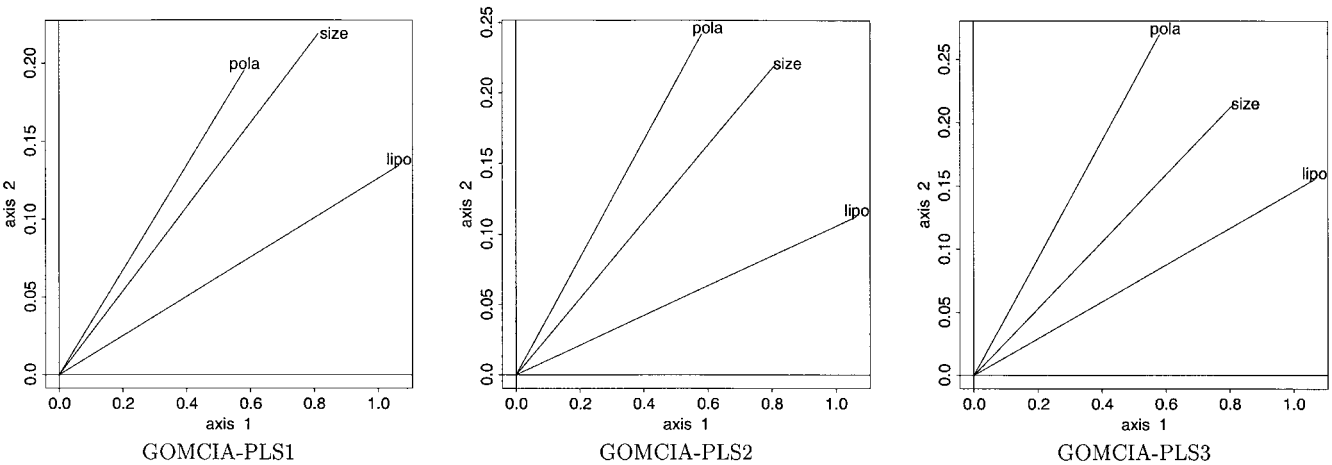

Figure 5. Situation 2: representation of $\mu_{k,1}$ against $\mu_{k,2}$ coefficients for the three GOMCIA–PLS methods. Recall that $\mu_{k,1} = cov(\mathbf{y},\mathbf{t}_{k,1})$ measure the proximity between response and *k*th *X*-block.

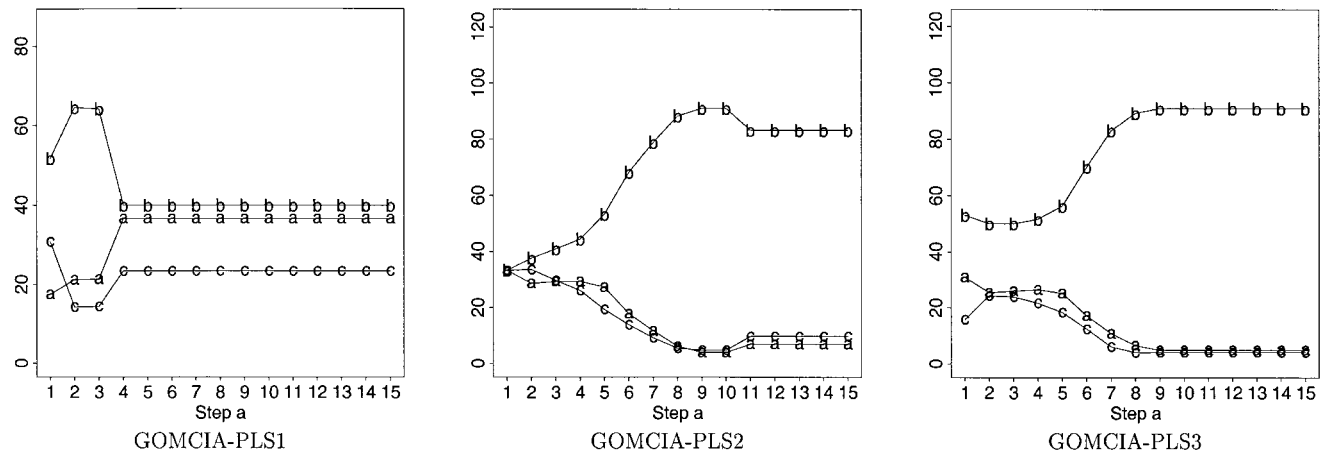

Figure 6. Situation 2: evolution of coefficients $B_n^2(k)$ with step *a*: 'a' is for variable size, 'b' for variable lipophilicity and 'c' for variable polarity.

be made about partial correlations and Figure 6 for GOMCIA–PLS1. The study of the partial correlations in the first step puts *lipophilicity* (0.77) in first position before *polarity* (0.71) and *size* (0.60). At the second step the partial correlations for *lipophilicity* and *polarity* have the same value (0.80) and that of *size* falls steeply to -0.64: this big step might explain the inversion of position between *size* and *polarity*.

All the *BIP* and *BIPrv* coefficients are greater than 1 for $\mathbf{X}_2$, i.e. *lipophilicity*.

7. DISCUSSION

Looking at the above results, we see that the different methods, i.e. the GOMCIA–PLS ones, NPLS and HPLS, all give a good R^2. However, in most cases in practice, HPLS gives very bad results in terms of R^2 in the first steps. GOMCIA–PLS1 gives the greatest R^2 from the first step, because the associated partial models have rank greater than or equal to 1, while the partial models from all the other methods have rank 1. The second advantage of GOMCIA–PLS1 is the orthogonality properties of scores $\{\mathbf{t}_{k,a}\}_a$ of each block *k*. The third advantage is that models seems to correspond with partial correlations between responses and scores of *X*-blocks. Thus we can say that GOMCIA–PLS1 is the best method among the three. In the example, GOMCIA–PLS3 is equivalent to performing a standard PLS on the data, so the results are satisfying in terms of goodness of fit. Apart from GOMCIA–PLS1 and 3, the NPLS method is the most explanatory in the first situation, but in the second situation it is GOMCIA–PLS2. We see here that it is very difficult to really affirm that a method is the best; the challenge is to choose the one most suited to the data: sometimes it will be a multiway method, sometimes a multiblock method. When the data are not cubic, the user has to choose one of the multiblock methods.

Let us compare the multiblock methods and GOMCIA–PLS. The MBPLS method optimizes at each step the covariance between super-scores $\mathbf{t}$ and $\mathbf{u}$, since it is equivalent to standard PLS on the large blocks of data [27]. The block scores are deduced from the super-scores. The block scores are not built in order to fit as well as possible the

Copyright © 2003 John Wiley & Sons, Ltd.

multiblock structure, while the GOMCIA–PLS methods maximize at each step the covariance between the block scores, and the super-scores are defined after that. Moreover, in GOMCIA–PLS2 the covariance between the super-scores is also maximized. The HPLS method is different from the GOMCIA–PLS2 method in that the block scores $\mathbf{t}_{k,a}$ are calculated with the super-scores $\mathbf{t}_a$ and not with the super-scores $\mathbf{u}_a$ as in GOMCIA–PLS2. Moreover, the HPLS method seems to optimize no criterion.

In GOMCIA–PLS methods the scores and weight vectors are computed with an algorithm similar to SIMCA [1]. Depending on the method, orthogonal scores or super-scores are obtained. In addition, the super-scores or scores model the unexplained part of the data matrices at each step. Hence the R^2 coefficient is decomposed into a sum of R^2 values corresponding to each step.

The B_n^2 coefficient is especially adapted to GOMCIA–PLS1 and does not give very good results in combination with GOMCIA–PLS2: in the first step, all X-blocks have the same importance. Some further studies have to be made to find another way to see the importance of an X-block in models. It remains to be seen more precisely too the behaviour of the μ_k coefficients in GOMCIA–PLS2. Some authors use the percentage of variation of $\mathbf{X}_k$ by the super-score to see how important a block is in the model at each step. In our mind, this is not sufficient. It is necessary to be able to measure this importance in a 'cumulative way' at each step. Thus the *BIP* and *BIPrv* coefficients were defined for GOMCIA–PLS1 and 3 (Section 5.1). Moreover, further works are in progress to build some tests on the significance of an X-block in the models.

The particular properties associated with the case of only one $\mathbf{y}$ give us an idea of how GOMCIA–PLS is running with several Y-blocks and lead us to think that the models, especially with GOMCIA–PLS1, are well built.

Acknowledgements

We wish to express our sincere gratitude to Susan Holmes, Stanford University, for her corrections to the manuscript and insightful remarks. Jean-Dominique Lebreton, CNRS, Montpellier, and Claude Manté, Université de la Méditerranée, Marseille, are thanked for their helpful comments. We would also like to thank the anonymous referee for his/her active interest and valuable suggestions for this paper.

APPENDIX I. NORMAL EQUATIONS

The normal equations for step a are

$$\nabla_{\mathbf{a}_k}\mathcal{L} = \sum_{n=1}^{N} \mathbf{X}_k^{(a-1)'}\mathbf{D}\mathbf{Y}_n^{(a-1)}\mathbf{b}_n - \mu_k\mathbf{a}_k = \mathbf{0} \tag{21}$$

$$\nabla_{\mathbf{b}_n}\mathcal{L} = \sum_{k=1}^{K} \mathbf{Y}_n^{(a-1)'}\mathbf{D}\mathbf{X}_k^{(a-1)}\mathbf{a}_k - \lambda_n\mathbf{b}_n = \mathbf{0} \tag{22}$$

$$\nabla_{\mu_k}\mathcal{L} = \mathbf{a}_k'\mathbf{a}_k - 1 = 0 \tag{23}$$

$$\nabla_{\lambda_n}\mathcal{L} = \mathbf{b}_n'\mathbf{b}_n - 1 = 0 \tag{24}$$

Equation (3) is obtained by left multiplying (21) with $\mathbf{a}_k'$ and using (23). Equation (4) is found similarly. Equations (6) and (7) are immediately achieved with (21) and (22) respectively. Equation (5) is obvious: by summing (3) from 1 to K and by summing (4) from 1 to N. It is crucial to choose positive multipliers to get the maximum Lagrangian function. Equation (8) is obtained by calculating the norm of $\mathbf{a}_k$ with (6) and setting it to 1. Equation (9) is derived analogously.

APPENDIX II. MONOTONICITY AND CONVERGENCE OF CRITERION OVER ITERATIONS

The proof is given for $a = 1$; it holds also for the next steps.

Let $h_k(\mathbf{a}_k) = \mu_k = \sum_{n=1}^{N} cov(\mathbf{t}_k, \mathbf{u}_n)$, where $\mathbf{a}_1,\ldots,\mathbf{a}_{k-1}$, $\mathbf{a}_{k+1}$, $\mathbf{b}_1,\ldots,\mathbf{b}_N$ are fixed, and let $\bar{\mathbf{a}}_k$ be the update of $\mathbf{a}_k$ in the algorithm:

$$\bar{\mathbf{a}}_k = \frac{\mathbf{X}_k'\mathbf{D}\sum_{n=1}^{N}\mathbf{Y}_n\mathbf{b}_n}{\|\mathbf{X}_k'\mathbf{D}\sum_{n=1}^{N}\mathbf{Y}_n\mathbf{b}_n\|}$$

The proof consists in showing that a pass in a loop of iteration modifies μ_k such that $\Delta h_k(\mathbf{a}_k) = h_k(\bar{\mathbf{a}}_k) - h_k(\mathbf{a}_k)$ is non-negative for an abitrary $\mathbf{a}_k$. In fact,

$$\Delta h_k(\mathbf{a}_k) = h_k(\bar{\mathbf{a}}_k) - h_k(\mathbf{a}_k) \tag{25}$$

$$= \sum_{n=1}^{N}\mathbf{u}_n'\mathbf{D}\mathbf{X}_k\frac{\mathbf{X}_k'\mathbf{D}\sum_{n=1}^{N}\mathbf{u}_n}{\|\mathbf{X}_k'\mathbf{D}\sum_{n=1}^{N}\mathbf{u}_n\|} - \sum_{n=1}^{N}\mathbf{u}_n'\mathbf{D}\mathbf{X}_k\mathbf{a}_k \tag{26}$$

$$= \frac{1}{\|\mathbf{X}_k'\mathbf{D}\sum_{n=1}^{N}\mathbf{u}_n\|}\left(\left\|\mathbf{X}_k'\mathbf{D}\sum_{n=1}^{N}\mathbf{u}_n\right\|^2 - \left\|\mathbf{X}_k'\mathbf{D}\sum_{n=1}^{N}\mathbf{u}_n\right\|\mathbf{a}_k'\mathbf{X}_k'\mathbf{D}\sum_{n=1}^{N}\mathbf{u}_n\right) \tag{27}$$

$$= \left\|\mathbf{X}_k'\mathbf{D}\sum_{n=1}^{N}\mathbf{u}_n\right\|\left(1 - r\left(\mathbf{a}_k, \mathbf{X}_k'\mathbf{D}\sum_{n=1}^{N}\mathbf{u}_n\right)\right) \tag{28}$$

As $r(\mathbf{a}_k, \mathbf{X}_k'\mathbf{D}\sum_{n=1}^{N}\mathbf{u}_n)$ is the correlation between $\mathbf{a}_k$ and $\mathbf{X}_k'\mathbf{D}\sum_{n=1}^{N}\mathbf{u}_n$, it is smaller than or equal to 1. This implies that h_k is increasing in each loop of the algorithm.

Denoting $h_n(\mathbf{b}_n) = \lambda_n$ and

$$\bar{\mathbf{b}}_n = \frac{\mathbf{Y}_n'\mathbf{D}\sum_{k=1}^{K}\mathbf{t}_k}{\|\mathbf{Y}_n'\mathbf{D}\sum_{k=1}^{K}\mathbf{t}_k\|}$$

we can similarly prove that $\Delta h_n(\mathbf{b}_n)$ is non-negative.

Each time $\mathbf{a}_k$ or $\mathbf{b}_n$ is updated, the criterion increases.

In addition, the series $h_k(\mathbf{a}_k)$ and $h_n(\mathbf{b}_n)$ over iterations are bounded by respectively $N\|\mathbf{X}'_k\mathbf{D}\mathbf{Y}\|$ and $K\|\mathbf{Y}'_n\mathbf{D}\mathbf{X}\|$. As a consequence, the algorithm converges.

APPENDIX III. ORTHOGONALITY OF $\{\mathbf{a}_{k,a}\}_a$ AND $\{\mathbf{b}_{n,a}\}_a$

This is the proof for $\{\mathbf{a}_{k,a}\}_a$. It is the same for $\{\mathbf{b}_{n,a}\}_a$.

Fix k in $1,\ldots,K$. Using (6) and the definition of step a of

Copyright © 2003 John Wiley & Sons, Ltd.

GOMCIA1, for all i in 1 to $a-1$,

$$\mu_{k,a}\mu_{k,i}\mathbf{a}'_{k,i}\mathbf{a}_{k,a} = \mathbf{a}'_{k,i}\mathbf{X}_k^{(a-1)'}\mathbf{D}\sum_{n=1}^{N}\mathbf{u}_{n,a} \tag{29}$$

$$= \mathbf{a}'_{k,i}\mathbf{X}_k^{(i-1)'}\mathbf{P}_{\mathbf{t}_{k,i}}^{\perp'}\prod_{l=i+1}^{a-1}\left(\mathbf{P}_{\mathbf{t}_{k,l}}^{\perp'}\right)\mathbf{D}\sum_{n=1}^{N}\mathbf{u}_{n,a} \tag{30}$$

$$= \mathbf{t}'_{k,i}\mathbf{P}_{\mathbf{t}_{k,i}}^{\perp'}\prod_{l=i+1}^{a-1}\left(\mathbf{P}_{\mathbf{t}_{k,l}}^{\perp'}\right)\mathbf{D}\sum_{n=1}^{N}\mathbf{u}_{n,a} \tag{31}$$

$$= 0 \tag{32}$$

APPENDIX IV. EXPRESSION OF $\widehat{\boldsymbol{\beta}}_n^A$

We pass on the proof of $\boldsymbol{\alpha}_a$, as it is obvious. Let us prove $\mathcal{E}_a = \mathbf{X}\boldsymbol{\delta}_a$ by induction.

The case $a = 1$ is obvious. For $a = 2$,

$$\begin{aligned}\mathcal{E}_2 &= \mathbf{X}^{(1)}\boldsymbol{\alpha}_2\\ &= \mathbf{P}_{\mathcal{E}_1}^{\perp}\mathbf{X}^{(0)}\boldsymbol{\alpha}_2\\ &= \mathbf{X}\boldsymbol{\alpha}_2 - \mathbf{X}\boldsymbol{\alpha}_1(\mathcal{E}_1'\mathbf{D}\mathcal{E}_1)^{-1}\mathcal{E}_1'\mathbf{D}\mathbf{X}\boldsymbol{\alpha}_2\\ &= \mathbf{X}\boldsymbol{\delta}_2\end{aligned}$$

Suppose the expression is true until step $a-1$. Then, in step a,

$$\mathcal{E}_a = \mathbf{X}^{(a-1)}\boldsymbol{\alpha}_a$$

$$= \mathbf{P}_{\mathcal{E}_{a-1}}^{\perp}\ldots\mathbf{P}_{\mathcal{E}_1}^{\perp}\mathbf{X}\boldsymbol{\alpha}_a \tag{33}$$

$$= \left(\mathbf{I}_{\mathrm{I}} - \sum_{l=1}^{a-1}\mathbf{P}_{\mathcal{E}_l}\right)\mathbf{X}\boldsymbol{\alpha}_a \quad \text{because spaces } \mathcal{E}_l \text{ are } \mathbf{D}\text{-orthogonal} \tag{34}$$

$$= \mathbf{X}\boldsymbol{\alpha}_a - \sum_{l=1}^{a-1}\mathbf{X}\boldsymbol{\delta}_l(\mathcal{E}_l'\mathbf{D}\mathcal{E}_l)^{-1}\mathcal{E}_l'\mathbf{D}\mathbf{X}\boldsymbol{\alpha}_a \tag{35}$$

$$= \mathbf{X}\boldsymbol{\delta}_a \tag{36}$$

We deduce from this relation and $\widehat{\mathbf{Y}}_n^A = \sum_{a=1}^{A}\mathbf{P}_{\mathcal{E}_a}\mathbf{Y}_n$ that $\widehat{\boldsymbol{\beta}}_n^A = \sum_{a=1}^{A}\boldsymbol{\delta}_a(\mathcal{E}_a'\mathbf{D}\mathcal{E}_a)^{-1}\mathcal{E}_a'\mathbf{D}\mathbf{Y}_n$.

REFERENCES

1. Wold H. Estimation of principal components and related models by iterative least squares. In *Multivariate Analysis*, Krishnaiah PR (ed.). Academic Press: New York, 1966; 391–420.
2. Wold S, Ruhe A, Wold H and Dunn WJ. The collinearity problem in linear regression. The partial least squares (PLS) approach to generalized inverses. *SIAM J. Sci. Statist. Comput.* 1984; **5**: 735–743.
3. Tucker LR. Implications of factor analysis of three-way matrices for measurement of change. In *Problems in Measuring Change*, Harris CW (ed.). University of Wisconsin Press: Madison, WI, 1963; 122–137.
4. Tucker LR. The extension of factor analysis to three dimensional matrices. In *Contributions to Mathematical Psychology*. Gulliksen H, Frederiksen N (eds). Holt, Reinhart and Winston: New York, 1964; 100–127.
5. Tucker LR. Some mathematical notes on three-mode factor analysis, *Psychometrika* 1966; **31**: 279–311.
6. Smilde AK. Three-way analyses. Problems and prospects. *Chemometrics Intell. Lab. Syst.* 1992; **15**: 143–157.
7. Law HG, Snyder Jr CW, Hattie JA and McDonald RP (eds). *Research Method for Multimode Data Analysis*. Praeger: New York, 1984.
8. Coppi R and Bolasco S (eds). *Multiway Data Analysis*. Elsevier: Amsterdam, 1989.
9. Andersson CA and Henrion R. A general algorithm for obtaining simple structure of core arrays in *N*-way PCA with application to fluorometric data. *Comput. Statist. Data Anal.* 1999; **31**: 255–278.
10. Henrion R. *N*-way principal component analysis: theory, algorithms and applications. *Chemometrics Intell. Lab. Syst.* 1994; **25**: 1–23.
11. Andersson CA, Munck L, Henrion R and Henrion G. Analysis of *N*-dimensional data arrays from fluorescence spectroscopy of an intermediary superproduct. *Fresenius J. Anal. Chem.* 1997; **359**: 138–142.
12. Ståhle L. Aspects of the analysis of three-way data. *Chemometrics Intell. Lab. Syst.* 1989; **7**: 95–100.
13. Bro R. Multiway calibration. Multilinear PLS. *J. Chemometrics* 1996; **10**: 47–61.
14. Smilde AK. Comments on multilinear PLS. *J. Chemometrics* 1997; **11**: 367–377.
15. Smilde AK and Kiers HAL. Multiway covariates regression models. *J. Chemometrics* 1999; **13**: 31–48.
16. Gower JC. Generalized Procrustes analysis. *Psychometrika* 1975; **40**: 33–51.
17. L'Hermier Des Plantes H. *Structuration des tableaux à trois indices de la statistique*. Thèse de 3ème Cycle, Université Montpellier II, 1976.
18. Lavit Ch. *Analyse Conjointe de Tableaux Quantitatifs*. Masson: Paris, 1988.
19. Lavit Ch, Escoufier Y, Sabatier R and Traissac P. The ACT (STATIS method). *Comput. Statist. Data Anal.* 1994; **18**: 97–119.
20. Chessel D and Hanafi M. Analyses de co-inertie de *K* nuages de points. *Rev. Statist. Appl.* 1996; **44**: 35–60.
21. Lafosse R and Hanafi M. Concordance d'un tableau avec *K* tableaux: définition de $K+1$ uples synthétiques. *Rev. Statist. Appl.* 1997; **45**: 111–126.
22. Vivien M. *Approches PLS linéaires et non linéaires pour la modélisation de multi-tableaux. Théorie et applications*. Thèse de Doctorat, Université Montpellier I, 2002 (http://www.myrtille.org/these.htm).
23. Vivien M and Sabatier R. Une extension multi-tableaux de la régression PLS. *Rev. Statist. Appl.* 2001; **49**: 31–54.
24. Wangen LE and Kowalski BR. A multiblock partial least squares algorithm for investigating complex chemical systems. *J. Chemometrics* 1988; **3**: 3–20.
25. Westerhuis JA and Coenegracht PMJ. Multivariate modelling of the pharmaceutical two-step process of wet granulation and tableting with multiblock partial least squares. *J. Chemometrics* 1997; **11**: 379–392.
26. Westerhuis JA, Kourti T and MacGregor JF. Analysis of multiblock and hierarchical PCA and PLS models. *J. Chemometrics* 1998; **12**: 301–321.
27. Westerhuis JA and Smilde AK. Deflation in multiblock PLS. *J. Chemometrics* 2001; **15**: 485–493.
28. Wold S, Kettaneh N and Tjessem K. Hierarchical multiblock PLS and PC models for easier interpretation and as an alternative to variable selection. *J. Chemometrics* 1996; **10**: 463–482.
29. Wold S, Geladi P, Esbensen K and Öhman J. Multiway principal components- and PLS-analysis. *J. Chemometrics* 1987; **1**: 41–56.
30. Ufkes JGR, Visser BJ, Heuver G and Van der Meer C. Structure–activity relationships of bradykinin-potentiating peptides. *Eur. J. Pharmacol.* 1978; **50**: 119–122.
31. Ufkes JGR, Visser BJ, Heuver G, Wynne HJ and Van der Meer C. Further studies on the structure–activity

Copyright © 2003 John Wiley & Sons, Ltd.

relationships of bradykinin-potentiating peptides. *Eur. J. Pharmacol.* 1982; **79**: 155–158.
32. Umetrics Inc. *Multivariate Analysis (3-day Course)*. Umetrics Inc.: Winchester, MA, 1995.
33. Fletcher R. *Practical Methods of Optimization*. Wiley: New York, 1987.
34. Chambers JM. *Programming with Data. A Guide to S Language*. MathSoft: Seattle, WA, 1998.
35. Robert P and Escoufier Y. A unifying tool for linear multivariate statistics methods: the RV coefficient. *Appl. Statist.* 1976; **25**: 257–265.
36. Tenenhaus M. A partial least squares approach to multiple regression, redundancy analysis, and canonical analysis. *Les cahiers de la Recherche HEC, CR 550/1995*, 1995.

Copyright © 2003 John Wiley & Sons, Ltd.