PLS model building with missing data: New algorithms and a comparative study

Article *in* Journal of Chemometrics · March 2017

DOI: 10.1002/cem.2897

Abel Folch-Fortuny
Royal DSM

Francisco Arteaga
Catholic University of Valencia San Vicente M…

Alberto Ferrer
Universitat Politècnica de València

PLS model building with missing data: new algorithms and a comparative study

Abel Folch-Fortuny[a,b,*], Francisco Arteaga[c], Alberto Ferrer[a]

[a] *Multivariate Statistical Engineering (GIEM), Dep. de Estadística e Investigación Operativa Aplicadas y Calidad, Universitat Politècnica de València, Camino de Vera s/n, 46022, Valencia, Spain*

[b] *Genetics BioIT DBC Department, DSM Food Specialties, Alexander Fleminglaan 1, 2613 AX Delft, The Netherlands*

[c] *Dep. of Biostatistics and Investigation, Universidad Católica de Valencia San Vicente Mártir, C/Quevedo 2, 46001 Valencia, Spain*

Abstract

New algorithms to deal with missing values in predictive modelling are presented in this article. Specifically, two trimmed scores regression (TSR) adaptations are proposed, one from principal component analysis (PCA) model building with missing data (MD) and other from partial least squares regression (PLS) model exploitation with missing values. Using these methods, practitioners can impute MD both in the explanatory/predictor and the dependent/response variables. PLS is used here to build the multivariate calibration models, however, any regression method can be used after MD imputation. Four case studies, with different latent structures, are analysed here to compare the TSR-based methods against state-of-the-art approaches. The MATLAB code for these methods is also provided for its direct implementation at `http://mseg.webs.upv.es`, under a GNU license.

Keywords: missing data, partial least squares regression (PLS), imputation, trimmed scores regression (TSR), multivariate calibration

1. Introduction

Missing data (MD) is a common problem in data analysis. MD may appear in databases for a different number of reasons: respondents not answering to some questions in surveys, values

[*] Corresponding authors:
E-mail address: `abfolfor@upv.es`

outside the instrument range or missing owing to malfunctions of the sensor, failure in the communication between the instrumentation and the digital control system (DCS), sensors with different sampling rates, errors during data acquisition, and so on [1–3]. Most data-driven exploratory and predictive models cannot deal directly with data sets including missing values. Thus, it is mandatory to address this problem as a previous step or to develop methodologies capable of fitting models with incomplete data. MD can be related either to the set of process/exploratory/predictor variables or to the quality/dependent/response variables in many industrial environments, such as chemical, biochemical or pharmaceutical, and several research areas, such as biology, chemistry, medicine, environmental sciences, psychology, economics or sociology.

When building statistical models, MD may appear at two stages i) model building (MB), when using data sets with missing values to fit the model, and ii) model exploitation (ME), i.e. when using existing models to process new observations with missing values.

MB and ME problems have been addressed when building exploratory models using principal component analysis (PCA). This way, a new regression-based framework for PCA-ME was proposed in [4, 5]. The methods presented there, known data regression (KDR) and trimmed scores regression (TSR), were proven statistically superior (in terms of the mean squared error) to other approaches commonly used by practitioners such as projection to the model plane (PMP), single component projection (SCP), conditional mean replacement (CMR) [6], iterative algorithm (IA) [7] and modified nonlinear iterative partial least squares regression algorithm (NIPALS) [8], using different industrial data sets as case studies. The framework was adapted to a PCA-MB environment in [3], where TSR arose as the best compromise solution for MD imputation when comparing to IA, PMP, KDR, data augmentation (DA) [9] and the nonlinear programming approach (NLP) [10]. Most of these methods were afterwards implemented in a graphical user interface (GUI) in MATLAB called Missing Data Imputation (MDI) Toolbox [11].

The most used methods for PLS-MB with MD are the aforementioned IA and NIPALS, in their PLS versions. They have been implemented in many commercial software packages, such as ProSensus MultiVariate [12], SIMCA-P [13], The Unscrambler [14] and PLS Toolbox [15]. Other methods have been proposed in the literature for missing data imputation in predictive modelling [16], such as the algorithm of Krzanowski based on SVD[17], the general iterative principal component imputation (GIP) [18], the multiple imputation by chained equations (MICE) [19], two regularized versions of the known E-M algorithm: one based on ridge regression (r-EM) [20] and the other one based on a truncated total least squares regression (t-EM) [21], and an approach based on an optimization procedure using an undeflated PLS algorithm (OUPLS) [22]. Regarding PLS-ME, apart from NIPALS and IA, the original TSR algorithm for PCA-ME was adapted in [23] with the aim of predicting the uncoming measurements and the future quality variables while the batch is still being processed.

After the good performance of TSR in PCA-MB, PCA-ME and PLS-ME, here two novel versions of TSR are proposed for PLS-MB with MD. Thus, TSR can be applied, from now on, to solve both MD problems (MB and ME) in exploratory and predictive models, as IA and NIPALS. For these methods, missing completely at random (MCAR) or missing at random (MAR) mechanisms are assumed for the MD, that is, the reason why an element is missing does not depend on the unobserved value (as it happens, for example, in censored data).

The first version of TSR presented here, TSR-1, is a direct adaptation of the algorithm for PCA-MB to PLS-MB, changing the data preprocessing within the algorithm. The second one, TSR-2, is an adaptation of the TSR algorithm for PLS-ME to PLS-MB, using the same rationale developed in [3] to adapt the regression-based framework methods from PCA-ME to PCA-MB. The other regression-based methods, KDR and its variants, are not adapted to a PLS-MB environment, since TSR has been shown a more efficient approach [3].

To test the novel TSR algorithms, a comparative study is presented here against other state-of-the-art methods [2]: NIPALS and IA. The aforementioned algorithm of Krzanowski, GIP, MICE, r-EM and t-EM are not included here, since they consider only MD in the predictor variables, and thus, its comparison would be more appropriate with PCA-MB methods, as commented in [3]. OUPLS is not used in the comparison for software availability problems.

The aim of this paper is to provide researchers and practitioners with a ready-to-use MATLAB code to impute missing values in a regression environment, that is, using not only information of predictor and response matrices separatedly but exploiting the relationships among them. This way, the algorithms provided here can be used for fitting PLS models with MD or for imputing MD as a previous step of any other methodology (predictive or not). The TSR algorithms proposed here are freely available at `http://mseg.webs.upv.es`, under a GNU license.

The structure of this article is as follows. Section 2 introduces the notation and explains how the two TSR algorithms for PLS-MB are built, and how NIPALS and IA are applied here. Sections 3-4 describe the data sets and the performance criteria used in the comparative study. After showing the resuts in Section 5, Section 6 discusses on the methods performances and presents the conclusions.

2. Methodology

2.1. Notation

Let $\mathbf{X}$ be an $N \times K$ predictor data set. An indicator matrix $\mathbf{M}$ can be defined to indicate where the MD appear in $\mathbf{X}$, i.e. $m_{nk} = 1$ if x_{nk} is missing and 0 otherwise. The complementary of the indicator matrix can also be built as $\bar{\mathbf{M}} = \mathbf{1}_N\mathbf{1}_K^{\mathrm{T}} - \mathbf{M}$. Let $\mathbf{x}_n^{\mathrm{T}}$ be an observation (row) of $\mathbf{X}$ with MD. Without loss of generality, the missing values, denoted as $\mathbf{x}_n^{\#\mathrm{T}}$, can be assumed to appear at the first R variables, while the available $K - R$ values are denoted as $\mathbf{x}_n^{*\mathrm{T}}$. The partition of $\mathbf{x}_n^{\mathrm{T}}$ can be extended to the whole dataset as $\mathbf{X} = [\mathbf{X}^{\#}\ \mathbf{X}^{*}]$. Finally, the data partition is transferred to a PCA model, affecting the $K \times A$ loading matrix $\mathbf{P} = \begin{bmatrix} \mathbf{P}^{\#\mathrm{T}} \\ \mathbf{P}^{*\mathrm{T}} \end{bmatrix}$, where A denotes the number of principal components (PCs) in the PCA model. This way, the score matrix can be written as $\mathbf{T} = \mathbf{XP} = \mathbf{X}^{\#}\mathbf{P}^{\#} + \mathbf{X}^{*}\mathbf{P}^{*}$. More details regarding the PCA-MB notation with MD can be found in [3].

When a response matrix $\mathbf{Y}$ $(N \times M)$ is considered, two indicator matrices can be defined as $\mathbf{M}_{\mathbf{X}}$ and $\mathbf{M}_{\mathbf{Y}}$. Considering that the missing data in $\mathbf{x}_n^{\mathrm{T}}$ and $\mathbf{y}_n^{\mathrm{T}}$ appear in the first $R_{\mathbf{X}}$ and $R_{\mathbf{Y}}$ positions, respectively, the MD partition of both vectors can be transferred to a PLS model as shown in Figure 1. This way, the MD partition in $\mathbf{X}$ defines the partition in the loading, $\mathbf{P}$, and weight, $\mathbf{W}$, matrices, while the partition in $\mathbf{Y}$ affects the loading matrix of the responses, $\mathbf{Q}$. Finally, the extention of the MD partition in the predictor variables is also extended to the normalized $(K \times A)$ weight matrix, which here is represented as $\mathbf{R} = \mathbf{W}(\mathbf{P}^{\mathrm{T}}\mathbf{W})^{-1}$, in the same way as in $\mathbf{W}$ (see Figure 1). It is worth noting that matrix $\mathbf{R}$ substitutes to the classical PLS normalized weight matrix $\mathbf{W}$

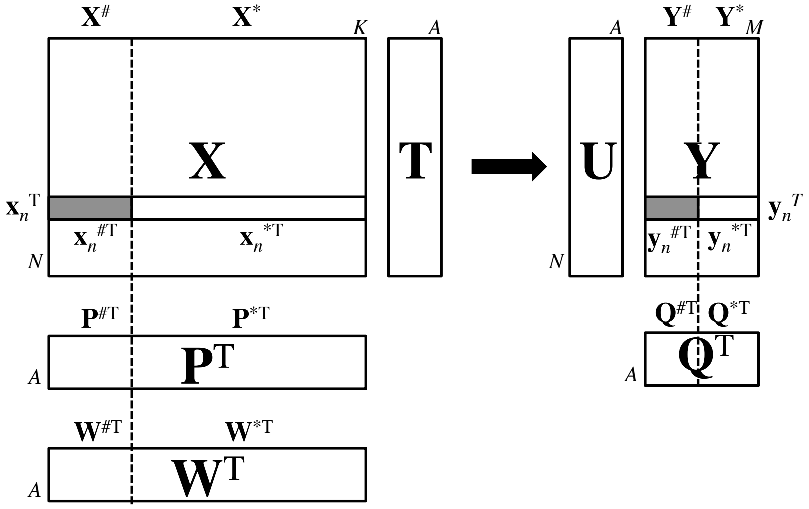
$\mathbf{X}^{\#}$
$\mathbf{X}^{*}$
K
A
A
$\mathbf{Y}^{\#}$
$\mathbf{Y}^{*}$
M
$\mathbf{X}$
$\mathbf{T}$
$\mathbf{U}$
$\mathbf{Y}$
$\mathbf{x}_n^{\mathrm{T}}$
$\mathbf{x}_n^{\#\mathrm{T}}$
$\mathbf{x}_n^{*\mathrm{T}}$
N
N
$\mathbf{y}_n^{T}$
$\mathbf{y}_n^{\#\mathrm{T}}$
$\mathbf{y}_n^{*\mathrm{T}}$
$\mathbf{P}^{\#\mathrm{T}}$
$\mathbf{P}^{*\mathrm{T}}$
$\mathbf{P}^{\mathrm{T}}$
A
$\mathbf{Q}^{\#\mathrm{T}}$
$\mathbf{Q}^{*\mathrm{T}}$
$\mathbf{Q}^{\mathrm{T}}$
A
$\mathbf{W}^{\#\mathrm{T}}$
$\mathbf{W}^{*\mathrm{T}}$
$\mathbf{W}^{\mathrm{T}}$
A

Figure 1: MD partition in PLS data matrices. Grey areas mark the missing values in the original data sets.

with a star superindex, which is used in this article to denote the submatrix of $\mathbf{W}$ corresponding to the available variables in observation $\mathbf{x}_n^{\mathrm{T}}$ (see Figure 1).

The columns of the aforementioned matrices are expressed here as the corresponding bold lower-case letters, e.g. $\mathbf{w}$ from $\mathbf{W}$. For $\mathbf{T}$, $\mathbf{t}$ represents the scores of all the observations for a particular PC/latent variable (LV) (a column of $\mathbf{T}$) and $\boldsymbol{\tau}^{\mathrm{T}}$ represents the scores of an observation for all PCs/LVs (a row of $\mathbf{T}$). Similarly, the columns of the $\mathbf{Y}$-scores matrix $\mathbf{U}$ are represented as $\mathbf{u}$.

No operator is used in this article for the usual matrix product. The Hadamard element-wise product between matrices is represented using symbol $\circ$.

2.2. NIPALS

Each iteration of the NIPALS algorithm performs a sequence linear regressions of columns and rows of $\mathbf{X}$ and $\mathbf{Y}$ matrices onto score vectors ($\mathbf{u}$ and $\mathbf{t}$, columns of $\mathbf{U}$ and $\mathbf{T}$, respectively) and weight vectors ($\mathbf{w}$ and $\mathbf{q}$, columns of $\mathbf{W}$ and $\mathbf{Q}$, respectively) until convergence [2, 24] (see Figure 1). When data in any column or row of $\mathbf{X}$ or $\mathbf{Y}$ are missing, the iterative regressions are performed using the available values, and the missing ones are ignored [8].

At each iteration of the NIPALS algorithm, the data are autoscaled, i.e. the mean is substracted form each column, and afterwards the resulting values are divided by their standard deviation. Indeed, this preprocessing, at each step of the algorithm, is also implemented in the other algorithms used here, in order to make their results comparable.

2.3. IA

IA relies on the estimates from a PLS model to fill in the MD in **X** and **Y** [2, 7]. The algorithm applied here consists of the following steps:

1. Replace the MD by the mean of the corresponding variables.
2. Autoscale **X** and **Y**.
3. Fit a PLS model using both data matrices.
4. Impute the missing values using the predictions from the PLS model: $\hat{\mathbf{X}} = \mathbf{TP}^{\mathrm{T}}$ and $\hat{\mathbf{Y}} = \mathbf{TQ}^{\mathrm{T}}$
5. If convergence has not been reached, return to step 2.

2.4. TSR adaptation from PCA-MB to PLS-MB (TSR-1)

TSR algorithm for PCA-MB [3] is summarised here. TSR starts with an initial mean imputation of the data set **X**. Afterwards, for each row $\mathbf{x}^{\mathrm{T}}$ with MD, it triggers a loop in which the MD are iteratively imputed fitting regression models between the missing positions and the scores of the available data:

$$\mathbf{X}^{\#} = (\mathbf{X}^{*}\mathbf{P}^{*})\mathbf{B} + \mathbf{U} \tag{1}$$

where $\mathbf{X}^{*}\mathbf{P}^{*}$ is the trimmed scores matrix, i.e. the score matrix that corresponds only to the known variables and their associated loadings, yielding:

$$\hat{\mathbf{B}} = (\mathbf{P}^{*\mathrm{T}}\mathbf{X}^{*\mathrm{T}}\mathbf{X}^{*}\mathbf{P}^{*})^{-1}\mathbf{P}^{*\mathrm{T}}\mathbf{X}^{*\mathrm{T}}\mathbf{X}^{\#} \tag{2}$$

Once the regression model is fitted, the missing part $\mathbf{x}^{\#\mathrm{T}}$ is estimated as :

$$\hat{\mathbf{x}}^{\#} = \mathbf{X}^{\#\mathrm{T}}\mathbf{X}^{*}\mathbf{P}^{*}(\mathbf{P}^{*\mathrm{T}}\mathbf{X}^{*\mathrm{T}}\mathbf{X}^{*}\mathbf{P}^{*})^{-1}\mathbf{P}^{*\mathrm{T}}\mathbf{x}^{*} = \mathbf{S}^{\#*}\mathbf{P}^{*}(\mathbf{P}^{*\mathrm{T}}\mathbf{S}^{**}\mathbf{P}^{*})^{-1}\mathbf{P}^{*\mathrm{T}}\mathbf{x}^{*} \tag{3}$$

where the covariance matrix of **X**, **S**, can be decomposed as:

$$\mathbf{S} = [\mathbf{X}^{\#}\mathbf{X}^{*}]^{\mathrm{T}}[\mathbf{X}^{\#}\mathbf{X}^{*}]/(N-1) = \begin{bmatrix} \mathbf{X}^{\#\mathrm{T}}\mathbf{X}^{\#} & \mathbf{X}^{\#\mathrm{T}}\mathbf{X}^{*} \\ \mathbf{X}^{*\mathrm{T}}\mathbf{X}^{\#} & \mathbf{X}^{*\mathrm{T}}\mathbf{X}^{*} \end{bmatrix} /(N-1) = \begin{bmatrix} \mathbf{S}^{\#\#} & \mathbf{S}^{\#*} \\ \mathbf{S}^{*\#} & \mathbf{S}^{**} \end{bmatrix} \tag{4}$$

At each iteration step, TSR fits as many regressions as rows with missing values. In the next iteration, the PCA model is recalculated and the missing data is imputed again using Equations 1-3. The loop stops when the difference between consecutive imputations is below the specified threshold.

PLS aims at finding the latent space of **X** that better explains **Y** by maximising the covariance between both data matrices. Thus, one could argue that one way of meeting this objective consists of augmenting the **X** data set with the **Y** matrix and fit a PCA model, which in this case would maximise the covariance of matrix [**X Y**]:

$$[\mathbf{X}\ \mathbf{Y}]^{\mathrm{T}}[\mathbf{X}\ \mathbf{Y}] = \begin{bmatrix} \mathbf{X}^{\mathrm{T}} \\ \mathbf{Y}^{\mathrm{T}} \end{bmatrix} [\mathbf{X}\ \mathbf{Y}] = \begin{bmatrix} \mathbf{X}^{\mathrm{T}}\mathbf{X} & \mathbf{X}^{\mathrm{T}}\mathbf{Y} \\ \mathbf{Y}^{\mathrm{T}}\mathbf{X} & \mathbf{Y}^{\mathrm{T}}\mathbf{Y} \end{bmatrix} \tag{5}$$

Following this idea, the TSR algorithm for PCA-MB with MD can be used directly for PLS-MB purposes simply by using the aforementioned augmented matrix as input. This way, after using Equation 3 in the iterative scheme for MD imputation, we would only need to fit a PLS model with the imputed data as final step.

Figure 2 shows a scheme of the adapted TSR algorithm using the augmented matrix [**X Y**]. This data matrix is autoscaled at each step *t* using the vector of means, **m**, and the diagonal matrix **D** containing the standard deviations of the variables. Once the MD have been imputed iteratively using TSR for PCA-MB, the last step of the algorithm consists of fitting a PLS model to obtain matrices **T**, **P**, **Q** and **R**. This TSR version for PLS-MB is from now on denoted as TSR-1. The iterative procedure stops when the imputed values stabilize, as the original TSR.

2.5. TSR adaptation from PLS-ME to PLS-MB (TSR-2)

Two issues arise in the straightforward adaptation of TSR-1. Firstly, even pursuing a similar objective, a PCA on [**X Y**] gives a different solution than a PLS, so a PCA-based model for MD may offer a different imputation than using a method fitting inner PLS models in the algorithm, as NIPALS and IA do. Secondly, the number of components may be different in PCA than in PLS.

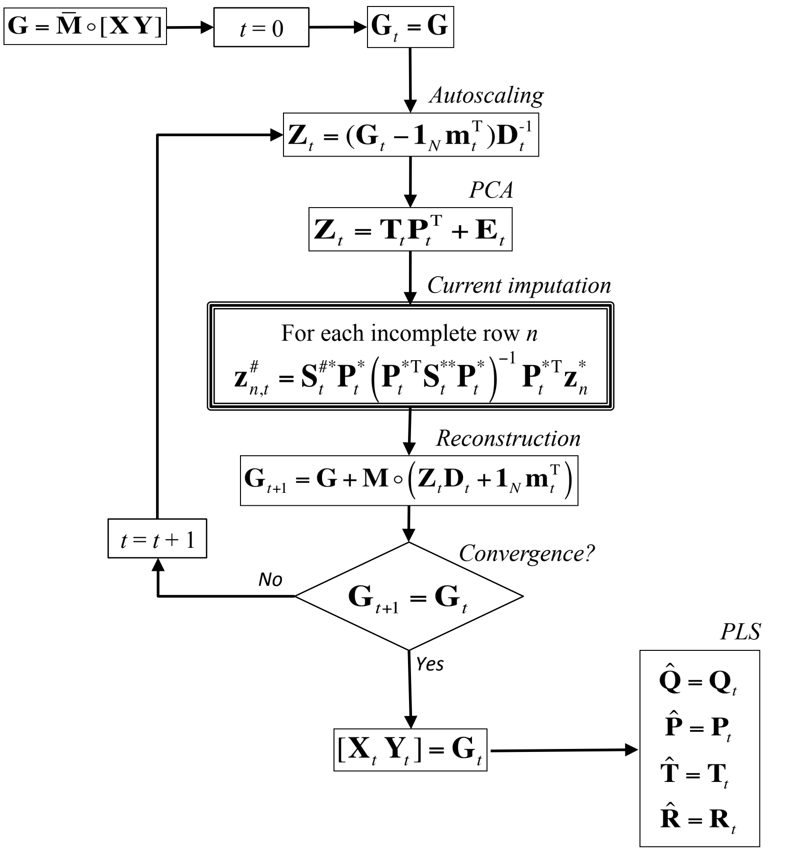
$\mathbf{G} = \bar{\mathbf{M}} \circ [\mathbf{X}\,\mathbf{Y}]$
$t = 0$
$\mathbf{G}_t = \mathbf{G}$
Autoscaling
$\mathbf{Z}_t = (\mathbf{G}_t - \mathbf{1}_N \mathbf{m}_t^{\mathrm{T}})\mathbf{D}_t^{-1}$
PCA
$\mathbf{Z}_t = \mathbf{T}_t \mathbf{P}_t^{\mathrm{T}} + \mathbf{E}_t$
Current imputation
For each incomplete row n
$\mathbf{z}_{n,t}^{\#} = \mathbf{S}_t^{\#*} \mathbf{P}_t^{*} \left(\mathbf{P}_t^{*\mathrm{T}} \mathbf{S}_t^{**} \mathbf{P}_t^{*} \right)^{-1} \mathbf{P}_t^{*\mathrm{T}} \mathbf{z}_n^{*}$
Reconstruction
$\mathbf{G}_{t+1} = \mathbf{G} + \mathbf{M} \circ \left(\mathbf{Z}_t \mathbf{D}_t + \mathbf{1}_N \mathbf{m}_t^{\mathrm{T}} \right)$
$t = t + 1$
Convergence?
No
$\mathbf{G}_{t+1} = \mathbf{G}_t$
Yes
$[\mathbf{X}_t\,\mathbf{Y}_t] = \mathbf{G}_t$
PLS
$\hat{\mathbf{Q}} = \mathbf{Q}_t$
$\hat{\mathbf{P}} = \mathbf{P}_t$
$\hat{\mathbf{T}} = \mathbf{T}_t$
$\hat{\mathbf{R}} = \mathbf{R}_t$

Figure 2: TSR-1 procedure for PLS-MB. **M** denotes here the MD indicator matrix of the augmented data [**X Y**]. $\bar{\mathbf{M}}$ is the complementary of the indicator matrix.

Therefore, if the number of PLS components are used to fit a PCA model using the augmented matrix, overfitting or underfitting problems may appear.

A TSR version for PLS-ME, using PLS as the core model, can be derived from the original idea of the algorithm for PCA-ME. In [23] this model was proposed to estimate the missing values in real-time batch monitoring. TSR for PLS-ME aims at estimating the complete scores of new observations using the information contained in the scores of the submatrix of **X** corresponding to the available data in the reference observation. Using matrix **T** from the complete model, this can be expressed as:

$$\mathbf{T} = \mathbf{T}^{*}\mathbf{B} + \mathbf{E} \tag{6}$$

where $\mathbf{T} = \mathbf{XR}$, **B** is here a different regression coefficient matrix than in Equations 1-2, and:

$$\mathbf{T}^{*} = \mathbf{X}^{*}\mathbf{R}^{*} = \mathbf{X}^{*}\mathbf{W}^{*}(\mathbf{P}^{\mathrm{T}}\mathbf{W})^{-1} \tag{7}$$

Matrices **P** and **W** are used to obtain $\mathbf{R}^{*}$ in order to improve prediction of the missing values using information from the complete PLS model, and to avoid problems of invertibility [23].

From Equations 6-7, the regression matrix **B** can be estimated as:

$$\hat{\mathbf{B}} = (\mathbf{T}^{*\mathrm{T}}\mathbf{T}^{*})^{-1}\mathbf{T}^{*\mathrm{T}}\mathbf{T} = (\mathbf{R}^{*\mathrm{T}}\mathbf{X}^{\mathrm{T}*}\mathbf{X}^{*}\mathbf{R}^{*})^{-1}\mathbf{R}^{*\mathrm{T}}\mathbf{X}^{*\mathrm{T}}\mathbf{T} \tag{8}$$

And since $\mathbf{X}^{*} = \mathbf{TP}^{*\mathrm{T}}$:

$$\hat{\mathbf{B}} = (\mathbf{R}^{*\mathrm{T}}\mathbf{X}^{\mathrm{T}*}\mathbf{X}^{*}\mathbf{R}^{*})^{-1}\mathbf{R}^{*\mathrm{T}}\mathbf{P}^{*}\mathbf{T}^{\mathrm{T}}\mathbf{T} = (\mathbf{R}^{*\mathrm{T}}\mathbf{S}^{**}\mathbf{R}^{*})^{-1}\mathbf{R}^{*\mathrm{T}}\mathbf{P}^{*}\mathbf{\Theta} \tag{9}$$

where $\mathbf{\Theta} = \frac{\mathbf{T}^{\mathrm{T}}\mathbf{T}}{N-1}$ is the covariance matrix of the scores. Finally, using the previous estimation, the scores of the PLS can be estimated in the last step of TSR for PLS-ME [23], that is, combining Equations 6-7:

$$\mathbf{T} = \mathbf{X}^{*}\mathbf{R}^{*}\mathbf{B} + \mathbf{E} \tag{10}$$

we get:

$$\hat{\boldsymbol{\tau}} = \hat{\mathbf{B}}^{\mathrm{T}}\mathbf{R}^{*\mathrm{T}}\mathbf{x}^{*} \tag{11}$$

being $\mathbf{x}^*$ the available part of the measurements of the new observation $\mathbf{x}$.

To adapt TSR from PLS-ME to PLS-MB, the same rationale presented in [3] is followed here. That is, the TSR version for ME is applied in each of the n rows with missing values of the data matrices at each step t of the iterative procedure, using the PLS model of the previous imputation step as the complete model.

Additionally, as a final step in TSR for PLS-MB, not only the PLS scores are needed, but the values for the MD imputation. These are obtained, from Equation 11, as:

$$\begin{aligned}\mathbf{x}_n^{\#} = \mathbf{P}^{\#}\hat{\boldsymbol{\tau}}_n = \mathbf{P}^{\#}\hat{\mathbf{B}}^{\mathrm{T}}\mathbf{R}^{*\mathrm{T}}\mathbf{x}_n^{*} = \mathbf{P}^{\#}\boldsymbol{\Theta}\mathbf{P}^{*\mathrm{T}}\mathbf{R}^{*}(\mathbf{R}^{*\mathrm{T}}\mathbf{S}^{**}\mathbf{R}^{*})^{-1}\mathbf{R}^{*\mathrm{T}}\mathbf{x}_n^{*} = \\ = \mathbf{P}^{\#}\frac{\mathbf{T}^{\mathrm{T}}\mathbf{T}}{N-1}\mathbf{P}^{*\mathrm{T}}\mathbf{R}^{*}(\mathbf{R}^{*\mathrm{T}}\mathbf{S}^{**}\mathbf{R}^{*})^{-1}\mathbf{R}^{*\mathrm{T}}\mathbf{x}_n^{*} = \\ = \frac{\mathbf{X}^{\#\mathrm{T}}\mathbf{X}^{*}}{N-1}\mathbf{R}^{*}(\mathbf{R}^{*\mathrm{T}}\mathbf{S}^{**}\mathbf{R}^{*})^{-1}\mathbf{R}^{*\mathrm{T}}\mathbf{x}_n^{*} = \mathbf{S}^{\#*}\mathbf{R}^{*}(\mathbf{R}^{*\mathrm{T}}\mathbf{S}^{**}\mathbf{R}^{*})^{-1}\mathbf{R}^{*\mathrm{T}}\mathbf{x}_n^{*}\end{aligned} \tag{12}$$

It is worth noting that Equation 12 gives, in fact, a similar estimation for the missing measurements in $\mathbf{X}$ as presented in [3] for PCA-MB, that is, substituting $\mathbf{P}^*$ by $\mathbf{R}^*$ in Equation 3.

Finally, the estimation for the $\mathbf{Y}$ missing values is obtained as:

$$\mathbf{y}_n^{\#} = \mathbf{Q}^{\#}\hat{\boldsymbol{\tau}}_n = \mathbf{Q}^{\#}\boldsymbol{\Theta}\mathbf{P}^{*\mathrm{T}}\mathbf{R}^{*}(\mathbf{R}^{*\mathrm{T}}\mathbf{S}^{**}\mathbf{R}^{*})^{-1}\mathbf{R}^{*\mathrm{T}}\mathbf{x}_n^{*} \tag{13}$$

Unfortunately, Equation 13 cannot be expressed in a more simplified way, since the matrix establishing the relationship between $\mathbf{X}$ and $\mathbf{Y}$, $\mathbf{R}$, has the dimensions of the loading matrix in $\mathbf{X}$, not in $\mathbf{Y}$.

This methodology can be represented as a diagram. Figure 3 shows the adapted TSR version from PLS-ME [23] to PLS-MB, from now on denoted as TSR-2. This algorithm is indeed similar to TSR-1 (see Figure 2) with some differences: i) since data matrices are processed separatedly, each step is applied on both matrices, ii) MD indicator matrices are defined, each one associated to the data partition in one of the matrices, and iii) a PLS model is fitted on both autoscaled data matrices, instead of PCA. The iterative procedure stops again when the imputation values stabilize in both data matrices, so the PLS matrices are estimated using the last round of imputation.

3. Data sets

The case studies used in the comparative study span different practical situations in chemometrics. We selected four data sets displaying fat (more variables than observations) and thin (more observations than variables) **X** matrices, using industrial and research data sets, real and simulated ones, comparing different latent variable structures (1, 3 and 6 LV), and different number of **Y** variables (1, 4 and 5).

The first case study is the Hald data set, widely used as an example for regression purposes [25, 26]. This data set has 13 observations of 4 ingredients of Portland cement and a single response variable equal to the number of calories of heat generated in the hardening process. One single LV is extracted, explaining 55% of **X** and 96% of **Y**.

The second data set is taken from systems biology, and corresponds to 36 cultures from the flux data set used in [27]. The 44 fluxes measured in each experiment, excluding biomass, are considered as predictors, and the protein produced as the response. 3 LVs are selected in the PLS, explaining 76.5% of variance in **X** and 71.5% in **Y**.

The third data set comes from chemometrics, and has been used in [28] for calibration transfer purposes imputing unmeasured spectra as missing data. It corresponds to a set of measurements of pseudo-gasoline samples using an spectrometer capturing wavelengths from 800 nm to 1600 nm in 2 nm intervals. The **Y** data correspond to measurements of heptane, iso-octane, toluene, xylene and decane concentration are the properties of interest to be predicted. The first (master) spectrometer is used here. 6 LV are used in the PLS model, explaining 99.9% and 99.8% of variance in **X** and **Y**, respectively.

Finally, a simulated data set including 10 variables and 100 observations is simulated [29, 30], using 4 PCs with eigenvalues equal to 3, 2.5, 2 and 1.5. The original data matrix is split afterwards: the first 6 variables are assigned to the **X** data set, while the remaining 4 to **Y**. When fitting a PLS model, 3 LVs are chosen, explaining 87.9% of variance in **X** and 75.2% in **Y**.

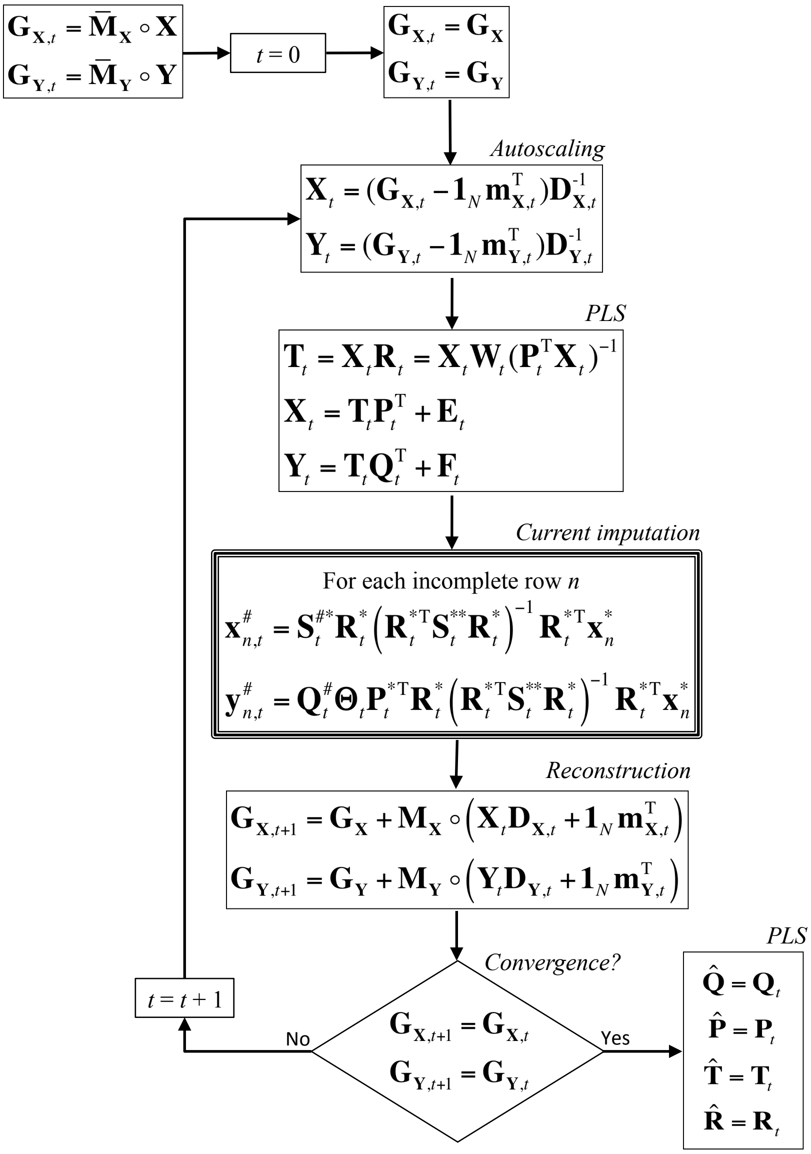
$\mathbf{G}_{\mathbf{X},t} = \bar{\mathbf{M}}_{\mathbf{X}} \circ \mathbf{X}$
$\mathbf{G}_{\mathbf{Y},t} = \bar{\mathbf{M}}_{\mathbf{Y}} \circ \mathbf{Y}$
$t = 0$
$\mathbf{G}_{\mathbf{X},t} = \mathbf{G}_{\mathbf{X}}$
$\mathbf{G}_{\mathbf{Y},t} = \mathbf{G}_{\mathbf{Y}}$
Autoscaling
$\mathbf{X}_t = (\mathbf{G}_{\mathbf{X},t} - \mathbf{1}_N \mathbf{m}_{\mathbf{X},t}^{\mathrm{T}})\mathbf{D}_{\mathbf{X},t}^{-1}$
$\mathbf{Y}_t = (\mathbf{G}_{\mathbf{Y},t} - \mathbf{1}_N \mathbf{m}_{\mathbf{Y},t}^{\mathrm{T}})\mathbf{D}_{\mathbf{Y},t}^{-1}$
PLS
$\mathbf{T}_t = \mathbf{X}_t\mathbf{R}_t = \mathbf{X}_t\mathbf{W}_t(\mathbf{P}_t^{\mathrm{T}}\mathbf{X}_t)^{-1}$
$\mathbf{X}_t = \mathbf{T}_t\mathbf{P}_t^{\mathrm{T}} + \mathbf{E}_t$
$\mathbf{Y}_t = \mathbf{T}_t\mathbf{Q}_t^{\mathrm{T}} + \mathbf{F}_t$
Current imputation
For each incomplete row n
$\mathbf{x}_{n,t}^{\#} = \mathbf{S}_t^{\#*}\mathbf{R}_t^{*}\left(\mathbf{R}_t^{*\mathrm{T}}\mathbf{S}_t^{**}\mathbf{R}_t^{*}\right)^{-1}\mathbf{R}_t^{*\mathrm{T}}\mathbf{x}_n^{*}$
$\mathbf{y}_{n,t}^{\#} = \mathbf{Q}_t^{\#}\mathbf{\Theta}_t\mathbf{P}_t^{*\mathrm{T}}\mathbf{R}_t^{*}\left(\mathbf{R}_t^{*\mathrm{T}}\mathbf{S}_t^{**}\mathbf{R}_t^{*}\right)^{-1}\mathbf{R}_t^{*\mathrm{T}}\mathbf{x}_n^{*}$
Reconstruction
$\mathbf{G}_{\mathbf{X},t+1} = \mathbf{G}_{\mathbf{X}} + \mathbf{M}_{\mathbf{X}} \circ \left(\mathbf{X}_t\mathbf{D}_{\mathbf{X},t} + \mathbf{1}_N \mathbf{m}_{\mathbf{X},t}^{\mathrm{T}}\right)$
$\mathbf{G}_{\mathbf{Y},t+1} = \mathbf{G}_{\mathbf{Y}} + \mathbf{M}_{\mathbf{Y}} \circ \left(\mathbf{Y}_t\mathbf{D}_{\mathbf{Y},t} + \mathbf{1}_N \mathbf{m}_{\mathbf{Y},t}^{\mathrm{T}}\right)$
Convergence?
$\mathbf{G}_{\mathbf{X},t+1} = \mathbf{G}_{\mathbf{X},t}$
$\mathbf{G}_{\mathbf{Y},t+1} = \mathbf{G}_{\mathbf{Y},t}$
No
$t = t + 1$
Yes
PLS
$\hat{\mathbf{Q}} = \mathbf{Q}_t$
$\hat{\mathbf{P}} = \mathbf{P}_t$
$\hat{\mathbf{T}} = \mathbf{T}_t$
$\hat{\mathbf{R}} = \mathbf{R}_t$

Figure 3: TSR-2 procedure for PLS-MB.

4. Comparative study

In the next section, the performance of TSR-1, TSR-2, IA and NIPALS are compared using the case studies. The strategy to generate the MD is the same as proposed by the authors in previous articles [3, 31]: 6 incremental levels of MD are considered in each data set, ranging from 10% to 60%, and for each data set and percentage, 50 incomplete data sets are simulated.

The principal performance criterion for each method is the mean squared prediction error (MSPE) in both **X** and **Y** data sets:

$$\text{MSPE-X}_{\text{Method}} = \frac{\sum_{n=1}^{N}\sum_{k=1}^{K}(\hat{x}_{nk} - \hat{x}_{nk}^{\text{Method}})^2}{NK} \tag{14}$$

$$\text{MSPE-Y}_{\text{Method}} = \frac{\sum_{n=1}^{N}\sum_{m=1}^{M}(\hat{y}_{nm} - \hat{y}_{nm}^{\text{Method}})^2}{NM} \tag{15}$$

where $\hat{x}_{nk}$ is the predicted value for the kth variable of the nth observation in the prediction matrix $\hat{\mathbf{X}} = \mathbf{TP}^{\mathrm{T}}$ obtained from the complete data set; and $\hat{x}_{nk}^{\text{Method}}$ the analogous prediction obtained after applying the corresponding method on the incomplete data set. The same applies for the mth **Y**-variable in $\hat{y}_{nm}$ and $\hat{\mathbf{Y}} = \mathbf{TQ}^{\mathrm{T}}$.

The second performance criterion is the cosine between the normalized weight vector of the first PLS, obtained using the full data matrix and its corresponding from the imputed data set.

In order to assess whether the differences among methods, in terms of MSPE, are statistically significant, a four-factor mixed-effect ANOVA model is fitted per each case study. Method (4 levels), **X**-MD percentage (6 levels), **Y**-MD percentage (6 levels), and their interactions are fixed-effect factors, and the data set, nested to the combination of **X**-MD and **Y**-MD percentages, is a random-effect factor. The total number of individual data sets imputed here is 4 original data sets × 6 MD percentages in **X** × 6 MD percentages in **Y** × 50 simulations = 7200.

A logarithmic transformation is used for MSPE-X and MSPE-Y. This transformation also expands the differences for low percentages of MD, easing the visualization of the plots. In case any effect or interaction is statistically significant (p-value<0.05) in the ANOVA model, the 95% least significant difference (LSD) intervals are computed to establish differences among methods.

The mechanism generating the MD in this comparative study is MAR or MCAR. In other words, these methods do not apply for missing values due to non-ignorable (NI) [2] mechanisms, in which the reason while a value is missing is related to the value itself. Classical examples of MD generated by NI mechanisms are values below the detection limit and respondants not answering specific questions in a survey or other types of censored data. In our case studies, to mimic the MAR and MCAR mechanisms, missing values are distributed randomly across variables and individuals in the data sets.

5. Results

5.1. Hald data

As expected, the more missing values are considered in both **X** and **Y** the more difficult is for all methods to reconstruct accurately the MD. This can be seen in the first and second column of plots in Figure 4, corresponding to MSPE-X and MSPE-Y values. Each plot in these two columns show the evolution of the MSPEs when increasing the **X**-MD percentage for a fixed **Y**-MD percentage. In the third column of plots, representing the cosines of the normalized weigths of the first LV, this effect can also be appreciated in the degradation of the cosine values.

NIPALS has problems in imputing MD in this first data set from 40% of **X**-MD onwards, and when converges it has in general a statistically worse peformance than the other methods in imputing MD in **X** (see first column of plots in Figure 4). Regarding the MSPE-Y, its performance is clearly the worst one (see second column of plots in Figure 4).

The performance of TSR-2 and IA is similar in MSPE-X, having TSR-2 a better performance for some percentages of MD (see first column of plots in Figure 4). Regarding MD in **Y**, IA attains a statistically better results for low **X**-MD (10-30%), otherwise their results are similar (see second column of plots in Figure 4).

The performance of TSR-1 is, in general, statistically superior to all the other methods up to 40% of **Y**-MD, and there tend to be no statistically significant differences for some **X**-MD percentages among the other methods (except NIPALS) for 50-60% of **Y**-MD.

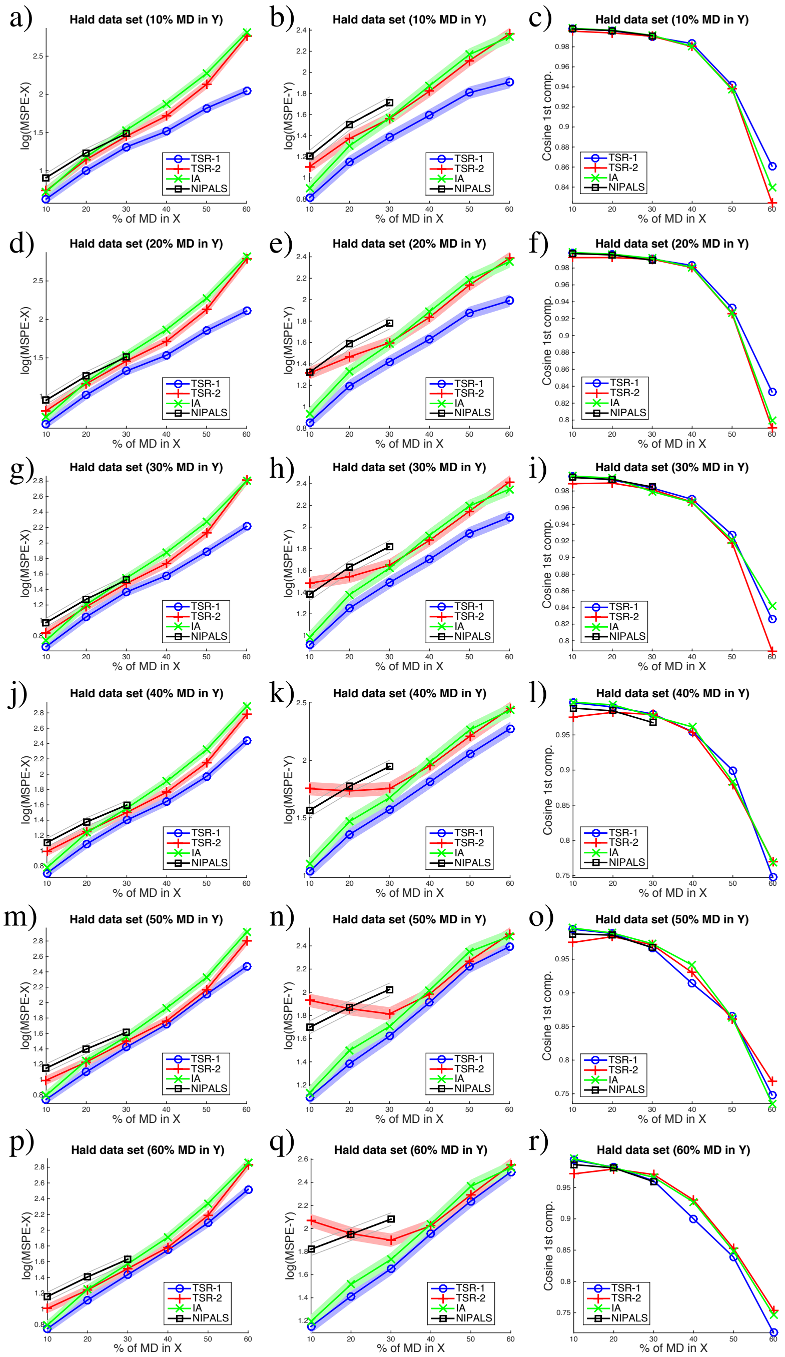

Figure 4: Hald data set resuls. The first (second) column of plots show the MSPE-X (MSPE-Y) results and the last column shows the cosines of the normalized weights of the first LV. The x-axis of each plot denotes the **X**-MD percentage. The differences regarding **Y**-MD percentages can be seen comparing rows of plots. The blue circles, red '+'s, green crosses and black squares denote the results of TSR-1, TSR-2, IA and NIPALS, respectively. The shaded bands represent the LSD 95% confidence intervals for the MSPE results of each method.

5.2. P. pastoris data

NIPALS has also problems in the *P. pastoris* data set (see Figure 5). Even having results on MSPE-Y statistically as good as TSR-1 for low percentages of **X**-MD and **Y**-MD, and statistically better than IA and TSR-2 (see Figure 5e, h and k), it fails to converge when more than 40% of missing data is considered in **Y**.

TSR-1 obtains here the best performance both in MSPE-X and MSPE-Y, with very few exceptions, in which its results are statistically equal to other approaches (see first and second column of plots in Figure 5). Mainly, the second-best method in this data set is TSR-2, followed by IA.

The MSPE-Y for high percentages of **Y**-MD shows an oscillatory performance for all methods, e.g. Figure 5k, n and q. This effect might be due to the 29% **Y**-variability not explained by the PLS model, causing artifacts depending on the combination of percentages of MD in **X** and **Y** considered for the imputation.

5.3. NIR data

The performance of NIPALS in this third case study is even poorer than in previous examples. Here, it is only able to impute up to 40% of **X** and 20% of **Y**-MD. And, when available, its results are statistically worse than the other iterative approaches.

Regarding MSPE-X, TSR-1 and TSR-2 have a similar performance, being both statistically superior to IA for 40%-60% of **X**-MD percentages (see first column of plots in 6). However, TSR-1 is indisputably the best method when checking the MSPE-Y results, followed by TSR-2, which gets statistically better or equal results than IA (second column of plots in Figure 6).

The performance in MSPE-Y of IA for high **Y**-MD percentages (see Figures 6k, n and q) improves when changing from 10 to 30% of MD in **X**. This is probably due to IA is being affected by overfitting in the imputation, since the percentage of variance explained of both **X** and **Y** in the PLS model is very high (see Section 3) when using 6 LV in the model. TSR-1 and TSR-2 seem to be not influenced by this problem.

The difference in the performances of TSR-based algorithms and IA can also be appreciated in the third column of plots in Figure 6, where, even getting very high cosines, the values of IA appear below TSRs' when the percentages of MD in **X** and **Y** increase.

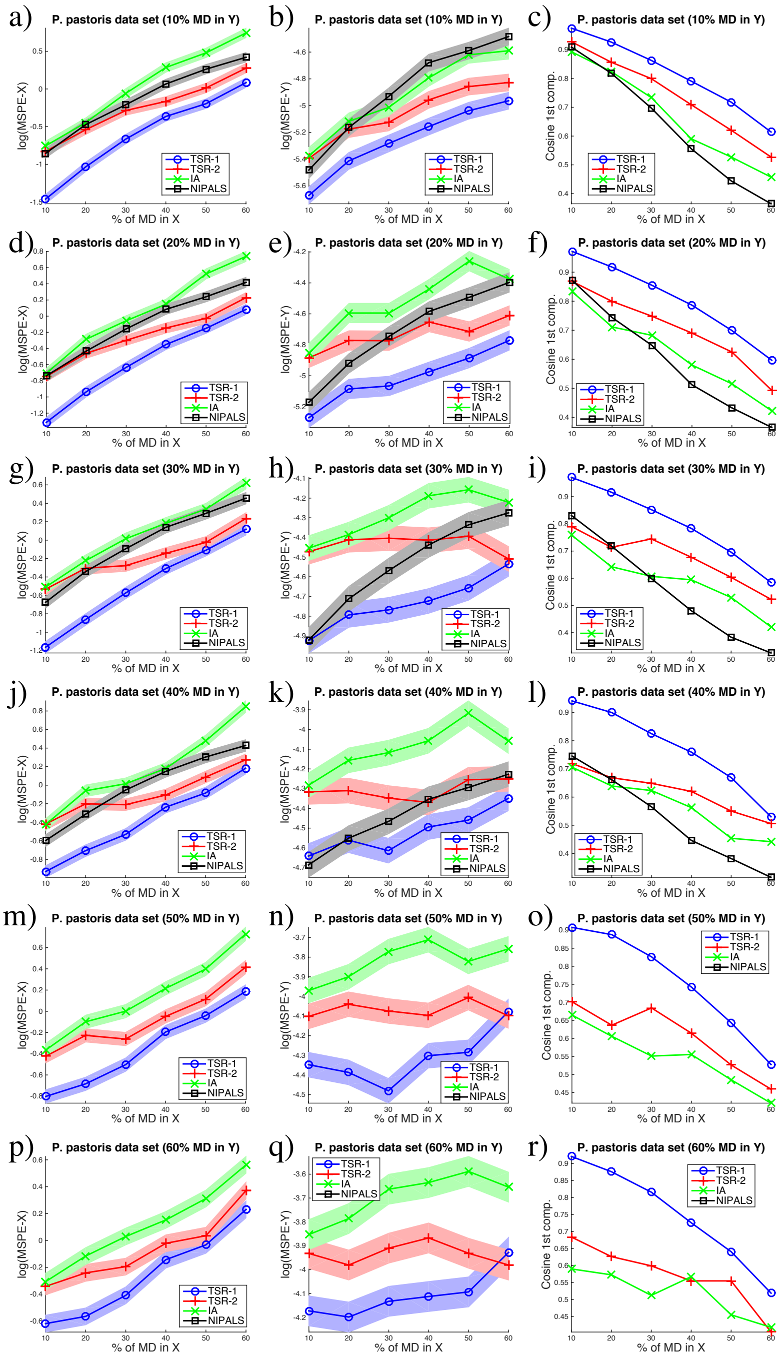

Figure 5: *P. pastoris* data set resuls. More details in caption of Figure 4.

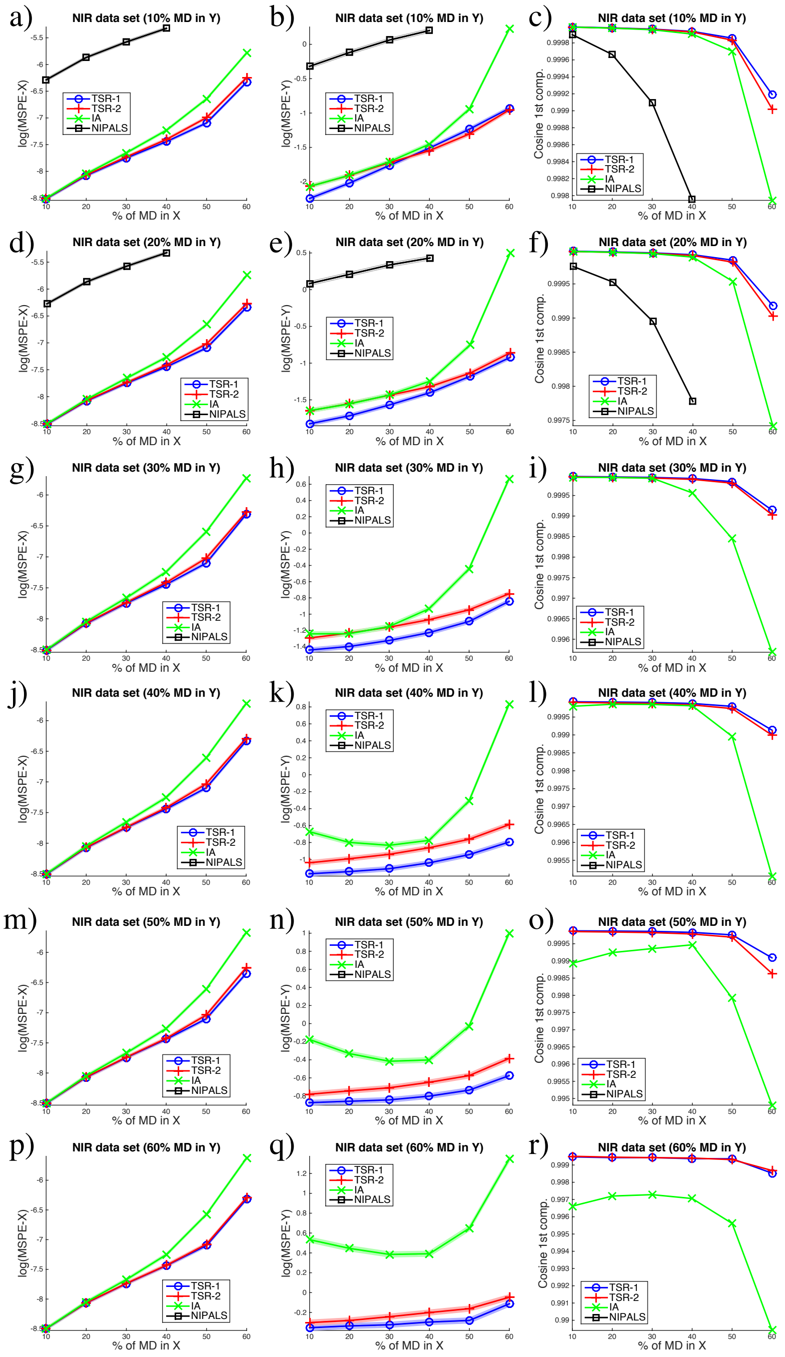

Figure 6: NIR data set resuls. More details in caption of Figure 4.

5.4. Simulated data

In the last case study analysed here, NIPALS is unable to analyse any combination of **X** and **Y**-MD percentages, even including only 10%-**X** and 10%-**Y** MD. TSR-1 shows again a clear statistically better performance in both MSPE-X and MSPE-Y for all MD percentages than its competitors, with few exceptions where its results are as accurate as TSR-2's. Between TSR-2 and IA there are again some cases in which they get statistically equal results, but in general the performance of TSR-2 outperforms IA. These significant differences match the results obtained in the third column of plots, corresponding to the cosines of the weight vector of the first LV.

In this dataset, IA shows an erratic performance, especially in MSPE-Y (see Figures 7n and q). This happened also in the *P. pastoris* case study, and reinforces the hypothesis that it is due to the lack of variance explained in **Y**, in this case similar to the aforementioned example (25%). However, TSR-1 seems to be not affected by this problem in any case study.

6. Discussion and conclusion

Two TSR algorithms have been proposed in this chapter: TSR-1 consists of an adaptation of the TSR algorithm from PCA-MB to PLS-MB, and TSR-2 is an adaptation of TSR from PLS-ME to PLS-MB. The representative set of case studies analysed here, which span different practical situations with missing data, show that TSR-1 is an excellent approach regardless the latent structure of the data. Its performance is in general statistically superior to TSR-2, in terms of MSPE-X and MSPE-Y, with few exceptions for some combinations of MD percentages in **X** and **Y**.

Both TSR-based approaches have been compared to other state-of-the-art methods: IA and NIPALS. IA shows generally a statistically worse performance than the TSR-based approaches, being its results in few cases closer to TSR-2's. NIPALS, a method implemented in many commercial statistical packages (such as ProSensus MultiVariate, The Unscrambler, SIMCA-P and PLS Toolbox), is clearly the statistically worst method compared here, since for most MD combinations is not able to converge and when it converges, its results are significantly worse than IA and TSR-based methods.

TSR-1 performed extraordinarily well for PLS-MB with MD. As commented in the Introduction, the ability of TSR to reconstruct the covariance matrix of incomplete data sets, which

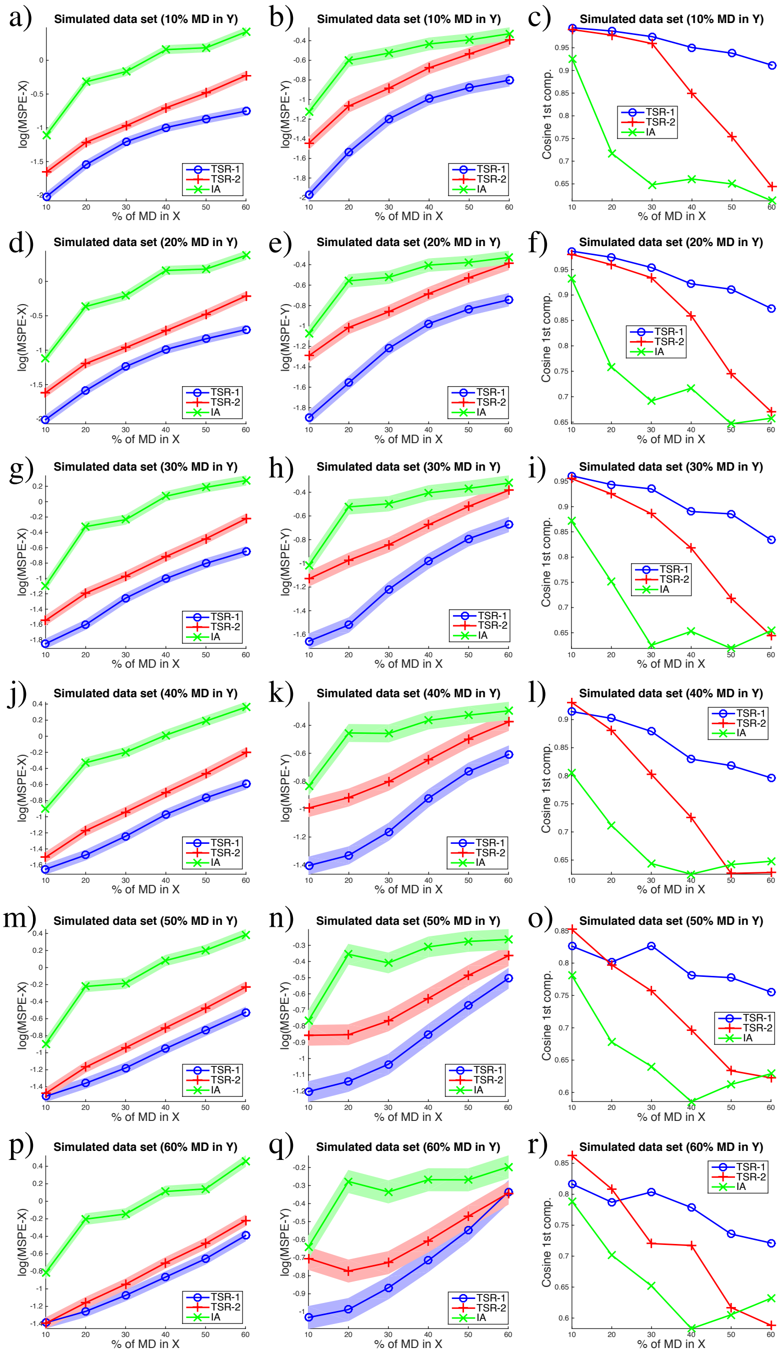

Figure 7: Simulated data set resuls. More details in caption of Figure 4.

ultimately determines the relationships among variables in most multivariate models, makes the final PLS fitted on imputed data resemble more the actual model than specific methodologies developed for PLS-MB with MD. This way, if practitioners find MD when fitting other covariance matrix-dependent methodologies, such as principal component regression or multiple regression models, they can use directly TSR-1 to impute the MD and then use the complete matrices for obtaining the desired model.

On the other hand, TSR-1 uses the number of components specified for the PLS model at hand to build the PCA-based model for the MD imputation. This may generate a problem if the covariance structure of the augmented data matrix [$\mathbf{X}$ $\mathbf{Y}$] is strongly different to the latent structure of a PLS model between $\mathbf{X}$ and $\mathbf{Y}$, thus provoking over or underfitting. However, one way to overcome this hypothetic situation consists of using an algorithm to select the appropriate number of PCs using the augmented matrix. In [11], the *ckf* algorithm [32] was used to decide the number of components in the MDI toolbox for PCA-MB. This procedure could solve the aforementioned problem.

Both TSR algorithms proposed here are freely available at `http://mseg.webs.upv.es`, under a GNU license.

Acknowledgements

Research in this study was partially supported by the Spanish Ministry of Science and Innovation and FEDER funds from the European Union through grant DPI2011- 28112-C04-02 and DPI2014-55276-C5-1R, and the Spanish Ministry of Economy and Competitiveness through grant ECO2013-43353-R.

7. References

[1] B. Grung, R. Manne, Missing values in principal component analysis, Chemometrics and Intelligent Laboratory Systems 42 (1-2) (1998) 125–139.

[2] F. Arteaga, A. Ferrer, Missing data, in: Comprehensive chemometrics chemical and biochemical data analysis, Vol. 3, Elsevier, Amsterdam, 2009, pp. 285–314.

[3] A. Folch-Fortuny, F. Arteaga, A. Ferrer, PCA model building with missing data: New proposals and a comparative study, Chemometrics and Intelligent Laboratory Systems 146 (2015) 77–88.

[4] F. Arteaga, A. Ferrer, Dealing with missing data in MSPC: Several methods, different interpretations, some examples, Journal of Chemometrics 16 (8-10) (2002) 408–418.

[5] F. Arteaga, A. Ferrer, Framework for regression-based missing data imputation methods in on-line MSPC, Journal of Chemometrics 19 (8) (2005) 439–447.

[6] P. R. Nelson, P. A. Taylor, J. F. MacGregor, Missing data methods in PCA and PLS: Score calculations with incomplete observations, Chemometrics and Intelligent Laboratory Systems 35 (1) (1996) 45–65.

[7] B. Walczak, D. Massart, Dealing with missing data, Chemometrics and Intelligent Laboratory Systems 58 (1) (2001) 15–27.

[8] S. Wold, C. Albano, W. J. Dunn, K. Esbensen, S. Hellberg, E. Johansson, M. Sjöström, Pattern recognition: Finding and using regularities in multivariate data, in: Food Research and Data Analysis, Elsevier Applied Science, London ; New York : New York, NY, USA, 1983, pp. 147–188.

[9] J. L. Schafer, Analysis of Incomplete Multivariate Data, 1st Edition, Chapman and Hall/CRC, Boca Raton, 1997.

[10] R. López-Negrete de la Fuente, S. García-Muñoz, L. T. Biegler, An efficient nonlinear programming strategy for PCA models with incomplete data sets, Journal of Chemometrics 24 (6) (2010) 301–311.

[11] A. Folch-Fortuny, F. Arteaga, A. Ferrer, Missing data imputation toolbox for MATLAB, Chemometrics and Intelligent Laboratory Systems 154 (2016) 93–100.

[12] ProSensus Multivariate release 16.02 (2016).

[13] SIMCA release 14 (2015).

[14] The Unscrambler X Release 10.4 (2016).

[15] PLS_Toolbox Release 8.1 (2016).

[16] Y. Liu, S. D. Brown, Comparison of five iterative imputation methods for multivariate classification, Chemometrics and Intelligent Laboratory Systems 120 (2013) 106–115.

[17] W. Krzanowski, Missing value imputation in multivariate data using the singular value decomposition of a matrix, Biometrical Letters XXV (1,2) (1988) 31–39.

[18] R. E. Dear, A principal-component missing-data method for multiple regression models, System Development Corp, 1959.

[19] I. R. White, P. Royston, A. M. Wood, Multiple imputation using chained equations: Issues and guidance for practice, Statistics in Medicine 30 (4) (2011) 377–399.

[20] T. Schneider, Analysis of Incomplete Climate Data: Estimation of Mean Values and Covariance Matrices and Imputation of Missing Values, Journal of Climate 14 (5) (2001) 853–871.

[21] R. Fierro, G. Golub, P. Hansen, D. O'Leary, Regularization by Truncated Total Least Squares, SIAM Journal on Scientific Computing 18 (4) (1997) 1223–1241.

[22] E. H. Puwakkatiya-Kankanamage, S. García-Muñoz, L. T. Biegler, An optimization-based undeflated PLS (OU-PLS) method to handle missing data in the training set, Journal of Chemometrics 28 (7) (2014) 575–584.

[23] J. Camacho, J. Picó, A. Ferrer, Bilinear modelling of batch processes. Part II: a comparison of PLS soft-sensors, Journal of Chemometrics 22 (10) (2008) 533–547.

[24] P. Geladi, B. Kowalski, Partial least-squares regression: a tutorial, Analytica Chimica Acta 185 (C) (1986) 1–17.

[25] A. Hald, Statistical Theory with Engineering Applications, 1st Edition, John Wiley & Sons Inc, New York etc., 1952.

[26] H. Kubinyi, Evolutionary variable selection in regression and PLS analyses, Journal of Chemometrics 10 (2) (1996) 119–133.

[27] J. M. González-Martínez, A. Folch-Fortuny, F. Llaneras, M. Tortajada, J. Picó, A. Ferrer, Metabolic flux understanding of Pichia pastoris grown on heterogenous culture media, Chemometrics and Intelligent Laboratory Systems 134 (2014) 89–99.

[28] A. Folch-Fortuny, R. Vitale, O. de Noord, A. Ferrer, Calibration transfer between NIR spectrometers: new proposals and a comparative study, Journal of Chemometrics *(submitted)*.

[29] F. Arteaga, A. Ferrer, How to simulate normal data sets with the desired correlation structure, Chemometrics and Intelligent Laboratory Systems 101 (1) (2010) 38–42.

[30] F. Arteaga, A. Ferrer, Building covariance matrices with the desired structure, Chemometrics and Intelligent Laboratory Systems 127 (2013) 80–88.

[31] A. Folch-Fortuny, F. Arteaga, A. Ferrer, Assessment of maximum likelihood PCA missing data imputation, Journal of Chemometrics 30 (2016) 386–393.

[32] E. Saccenti, J. Camacho, On the use of the observation-wise k-fold operation in PCA cross-validation, Journal of Chemometrics 29 (8) (2015) 467–478.