Folch Fortuny, A.; Arteaga Moreno, FJ.; Ferrer, A. (2016). Missing Data Imputation Toolbox for MATLAB. Chemometrics and Intelligent Laboratory Systems. 154:93-100. doi:10.1016/j.chemolab.2016.03.019.

Software Description

Missing Data Imputation Toolbox for MATLAB

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

[a]*Departamento de Estadística e Investigación Operativa Aplicadas y Calidad, Universidad Politécnica de Valencia, Camino de Vera s/n, Edificio 7A, 46022 Valencia, Spain*

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

Abstract

Here we introduce a graphical user-friendly interface to deal with missing values called Missing Data Imputation (MDI) Toolbox. This MATLAB toolbox allows imputing missing values, following missing completely at random patterns, exploiting the relationships among variables. In this way, principal component analysis (PCA) models are fitted iteratively to impute the missing data until convergence. Different methods, using PCA internally, are included in the toolbox: trimmed scores regression (TSR), known data regression (KDR), KDR with principal component regression (KDR-PCR), KDR with partial least squares regression (KDR-PLS), projection to the model plane (PMP), iterative algorithm (IA), modified nonlinear iterative partial least squares regression algorithm (NIPALS) and data augmentation (DA). MDI Toolbox presents a general procedure to impute missing data, thus can be used to infer PCA models with missing data, to estimate the covariance structure of incomplete data matrices, or to impute the missing values as a preprocessing step of other methodologies.

Keywords: *Missing data*, *Imputation*, *PCA model building*

1. Introduction

The problem of missing data arises in several research areas, such as chemometrics [1], genomics [2], network inference [3], meteorology [4], engineering [5], informatics [6], and chemical [7], biochemical [8] and pharmaceutical [9] industries. Very often, missing data are directly disregarded, *i.e.* when a missing measurement appears for one variable, the whole values for that individual are discarded, due to a single missing value. This leads to a huge loss of information. Sometimes the missing values are imputed using information contained in the available data. However, poor estimations of these values can distort the available information introducing bias on further methods applied to the data.

One of the most used statistical methods in all research areas to compress data into a few explanatory latent variables is principal component analysis (PCA) [10]. PCA can be used to impute missing values considering the latent structure of the data set, thus taking into account not only the information regarding to the observation with missing values and the variable itself, but also their relationships with the rest of observations and variables. Several methods have been presented in the literature exploiting the ability of PCA to impute missing values [1, 7, 11, 12, 13, 14, 15, 16].

Here we present a graphical user-friendly MATLAB toolbox, called Missing Data Imputation (MDI) Toolbox, devoted to fulfil incomplete data sets. The MDI toolbox is freely available for academic purposes at `http://mseg.webs.upv.es`, under a GNU license. The missing values are imputed applying PCA model building methods with missing data. The different methods implemented in MDI Toolbox are: trimmed scores regression (TSR), known data regression (KDR), KDR with principal component regression (KDR-PCR), KDR with partial least squares (KDR-PLS), projection to the model plane (PMP), iterative algorithm (IA), modified nonlinear iterative partial least squares

*Corresponding author.
Email address: `abfolfor@upv.es` (Abel Folch-Fortuny)

regression algorithm (NIPALS) and data augmentation (DA). The main outputs of MDI Toolbox are the PCA model of the incomplete data set, the estimated covariance matrix and the original missing data set with the imputed missing values.

This paper is organised as follows. Section 2 introduces each imputation method implemented in the MDI Toolbox. Some comments on the software requirements are made in Section 3. The data sets included as examples in the toolbox are briefly described in Section 4. An example of analysis using MDI Toolbox is proposed in Section 5, explaining in detail the steps via the graphical interface to obtain the missing data imputation. Finally, some concluding remarks and an external validation of the toolbox are made on Sections 6-7.

2. Methods

In this section, different methods for PCA model building (MB) with missing data are described. Most of these methods were recently described in [1], where an exhaustive comparative study was carried out among all of them with several data sets. All these methods have been implemented in the MDI Toolbox.

2.1. Notation

Some notation is introduced here to ease the interpretation of the missing data methods. Throughout this paper bold capital letters denote matrices, bold lower-case letters denote column vectors, and italic letters (capital and lower-case) denote scalars. Transposed matrices are denoted by superindex $^{\mathrm{T}}$.

Let $\mathbf{X}$ be a data matrix with N observations and K variables. The ith row of $\mathbf{X}$ is denoted as $\mathbf{x}_i^{\mathrm{T}} = [x_{i1}, ..., x_{ij}, ..., x_{iK}]$, where x_{ij} corresponds to the jth variable. If the ith row has missing values for some variables, the row vector can be rearranged, without loss of generality, as follows: $\mathbf{x}_i^{\mathrm{T}} = [\mathbf{x}_i^{\#\mathrm{T}} \mathbf{x}_i^{*\mathrm{T}}]$, where the sharp symbol denotes the variables with missing values and the asterisk represent the available measurements. Taking this row as reference, a partition in the original matrix is induced as $\mathbf{X} = [\mathbf{X}^{\#}, \mathbf{X}^{*}]$. This partition can be extended to the covariance matrix of $\mathbf{X}$, once centered: $\mathbf{S} = \mathbf{X}^{\mathrm{T}}\mathbf{X}/(N-1)$. In this way, $\mathbf{S}^{**}$ is the covariance matrix of $\mathbf{X}^{*}$, $\mathbf{S}^{\#\#}$ is the covariance matrix of $\mathbf{X}^{\#}$, and $\mathbf{S}^{\#*} = \mathbf{X}^{\#\mathrm{T}}\mathbf{X}^{*}/(N-1)$.

The matrix $\mathbf{M}$ denotes the missing data pattern in $\mathbf{X}$: $m_{ij} = 1$ if x_{ij} is missing, and 0 otherwise. Alternatively, matrix $\bar{\mathbf{M}}$ denotes the complement of $\mathbf{M}$ matrix. And finally, $\mathbf{Z}$ denotes the resulting $\mathbf{X}$ matrix after filling in the unknown values with zeroes, that is, $\mathbf{Z} = \bar{\mathbf{M}} \circ \mathbf{X}$, where $\circ$ is the Hadamard element-wise product.

2.2. Principal component analysis (PCA)

PCA [10] is a multivariate method aimed at finding the subspace where the data most vary. For this, the original variables, usually correlated, are compressed into a reduced set of uncorrelated principal components (PCs). The PCA model is as follows:

$$\mathbf{X} = \mathbf{T}\mathbf{P}^{\mathrm{T}} + \mathbf{E} \tag{1}$$

where $\mathbf{P}$ is the loadings matrix, containing by columns the linear combinations of the original variables defining the latent space; $\mathbf{T}$ is the scores matrix, having by columns the PCs; and $\mathbf{E}$ is the error matrix. The missing data partition $\mathbf{X} = [\mathbf{X}^{\#}\mathbf{X}^{*}]$, induced by row $\mathbf{x}_i^{\mathrm{T}} = [\mathbf{x}_i^{\#\mathrm{T}} \mathbf{x}_i^{*\mathrm{T}}]$, can be also extended to the PCA model (Equation 1), *i.e.*:

$$[\mathbf{X}^{\#}\mathbf{X}^{*}] = \mathbf{T}[\mathbf{P}^{\#\mathrm{T}}\mathbf{P}^{*\mathrm{T}}] + \mathbf{E} \tag{2}$$

2.3. Missing data imputation methods

The methods implemented in the MDI Toolbox are briefly described in this section. For further details on each method, readers are referred to [1], where the regression-based framework methods for PCA model building were originally proposed and compared against other classical approaches, most of them included also in the MDI Toolbox.

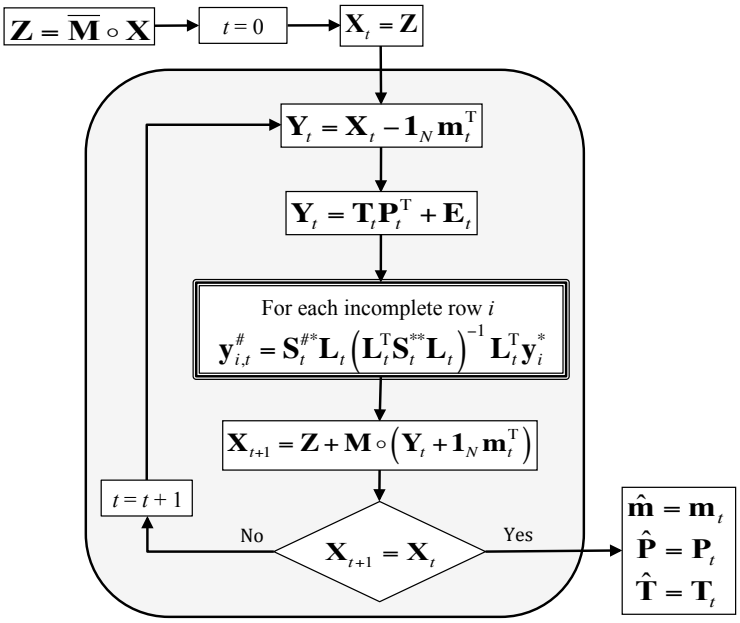

Figure 1: Regression-based framework adapted for PCA-MB with missing data [1].

Method	**Key matrix L**
TSR	$\mathbf{P}^{*}$
KDR	$\mathbf{I}$
KDR-PCR	$\mathbf{V}_{1:\rho}$, eigenvector matrix of $\mathbf{S}^{**}$ and $\rho \leq rank(\mathbf{S}^{**})$
KDR-PLS	$\mathbf{W}^{*}$, loadings matrix of the PLS model $\mathbf{T}_{PLS} = \mathbf{X}^{*}\mathbf{W}^{*}$

Table 1: Key matrix **L** for the regression based methods.

2.3.1. Regression-based framework methods

The regression-based methods have been recently adapted [1] from the PCA model exploitation (ME) context [15, 16]. In PCA-ME it is assumed that a PCA model has been already fitted on complete data, and it is desired to analyse a new observation with missing values. The regression-based methods are: TSR, KDR, KDR-PCR and KDR-PLS. More details on PCR and PLS basics can be found in [17].

Figure 1 presents a flow diagram with the procedure of these methods. They start filling the missing positions with zeroes, and estimating, after centering with the mean vector $\mathbf{m}$, a PCA model. Then, for each incomplete row i, an estimation of the missing values are obtained based on $\mathbf{S}^{\#*}$, $\mathbf{S}^{**}$ and the key matrix $\mathbf{L}$, which is different for each method (see Table 1). Afterwards, the original missing values are replaced by the estimations of the regression model. If the solution between the new imputed data matrix and the previous one is smaller than the established tolerance, the algorithm has converged. If not, another iteration t of the algorithm is performed (see Figure 1).

2.3.2. Projection to the model plane (PMP)

The PMP method was adapted, jointly with the regression-based methods, in [1]. This method was also originally proposed for the PCA-ME problem [12]. The algorithm follows the scheme presented in Figure 1, but differs in the imputation step (framed box). Instead of using the covariance matrices, at step t imputes the missing values using solely the loadings matrix:

$$\mathbf{y}_{i,t}^{\#} = \mathbf{P}_t^{\#}(\mathbf{P}_t^{*T}\mathbf{P}_t^{*})^{-1}\mathbf{P}_t^{*T}\mathbf{y}_{i,t}^{*} \tag{3}$$

2.3.3. Iterative algorithm (IA)

IA [13] imputes the missing values using directly the prediction from the PCA model. Therefore, IA follows also the scheme presented in Figure 1, but disregarding the framed box, *i.e.* the missing values are substituted by the estimation using the loadings and the scores of the PCA model fitted with the previous imputation:

$$\mathbf{X}_{t+1} = \mathbf{Z} + \mathbf{M} \circ (\mathbf{T}_t\mathbf{P}_t^T + \mathbf{1}_N\mathbf{m}_t^T) \tag{4}$$

2.3.4. Modified nonlinear iterative partial least squares regression (NIPALS)

The modified NIPALS algorithm [11, 14] consists of adapting the NIPALS algorithm to deal with incomplete rows in the **X** data set by performing the iterative regressions using the available data and ignoring the missing values.

2.3.5. Data augmentation (DA)

As opposed to the previous methods, DA [18] is a multiple imputation method. Its main difference among the others is that instead of imputing a single value per iteration, DA imputes for each missing datum several (M) values representing a distribution reflecting the sampling variability.

The DA procedure is as follows (see Figure 2). Initially, each x_{ij} missing is replaced with the mean of the known values of the correspondig variable j, thus the initial estimates for the mean vector $\mathbf{m}_{Ini}$ and the covariance matrix $\mathbf{S}_{Ini}$ are built. Using these matrices, the missing positions, $\mathbf{X}_i^{\#}$ are estimated by regressing over the known values, $\mathbf{X}^{*}$ (Imputation step). Once the mean and covariance matrices are recalculated with the new values, $\mathbf{m}_{i+1}$ and $\mathbf{S}_{i+1}$ are built as random draws from the Bayesian posterior distribution of $\mathbf{m}_i$ and $\mathbf{S}_i$ (Posterior step). After CL consecutive iterations, the first estimations of **m** and **S** are obtained. Finally, the previous procedure is repeated M times and the final values for **m** and **S** are obtained averaging the M previous estimations.

The procedure of alternately simulating missing data and parameters creates a Markov chain that eventually stabilizes or converges in distribution [7]. More details on this procedure can be found in [18] and in [1].

3. Software specifications and requirements

The MDI Toolbox has been built in MATLAB R2013a (Mathworks, Sheborn, MA), and it has been tested in many different previous and posterior MATLAB versions (2010-2015). The toolbox consists of a set of `.m` files, with the source code of the menus and the imputation methods; a set of `.fig` files, with the graphical interface; and a `.mat` file (`MDI_Examples.mat`) with few examples to run the toolbox. The toolbox is launched introducing `MDIgui` in the MATLAB command window. Afterwards, the main function calls other auxiliary routines (`SelectData.fig`, `SelectExample.fig`, `DataOverview.fig`, `NumberComponents.fig` and `ShowResults.fig`) until performing the imputation. The output of the toolbox is a data structure, whose fields are described in Table 2.

4. Data sets

Three data sets are included in the MDI Toolbox. The first one consists of the percentage composition of eight fatty acids: palmitic, palmitoleic, stearic, oleic, linoleic, linolenic, arachidic and eicosenoic, found in the lipid fraction of 75 olive oils of South Apulia (Italia) [19]. The second data set correspond to NIR spectra (wavelengths 750-1550 in 2 mm increments) of several diesel fuels (N = 40) obtained at the Southwest Research Institute (SWRI) on a project sponsored by the U.S. Army [20]. Finally, a multivariate data set with 10 variables and 100 observations, simulated from [21, 22], is included. For each data set the toolbox includes the complete data, and data sets with 10%, 30% and 60% of missing data following missing completely at random (MCAR) patterns.

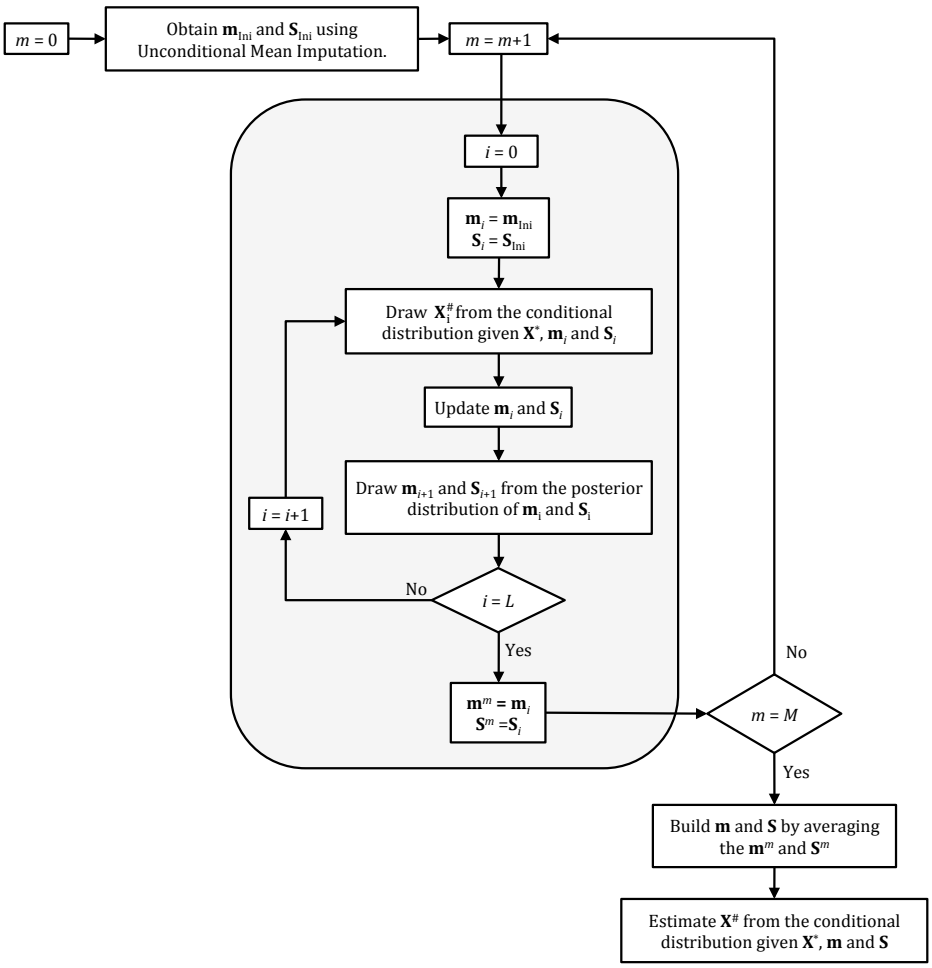

Figure 2: Data augmentation algorithm implemented in MDI Toolbox.

Field	Type	Content
Dataset	string	Name of the data set
X_MD	array	Data set with missing values
Percentage_MD	double	Percentage of missing values
X_imputed	array	Data set with the imputed values
PCs	integer	Number of principal components selected
Mean	arary	Estimation of the mean vector
Covariances	array	Estimated covariance matrix
Iterations	integer	Iterations required by the missing data method (not available when DA is applied)
Tolerance	integer	Threshold for convergence (not available when DA is applied)
X_reconstructed	array	Predictions of the PCA model
Method	string	Method applied
Computationtime	double	Computation time measured in seconds
Ini_est_cum_R2	array	Initial estimation of the cumulative explained variance in data (R^2)
Ini_est_eig	array	Initial estimation of the eigenvalues of the covariance data matrix
Loadings	array	Loadings matrix of the PCA model of X_imputed
Scores	array	Scores matrix of the PCA model of X_imputed
Num_Markov_chains	integer	Number of Markov chains computed when DA is applied
Chain_Length	integer	Length of each Markov chain computed when DA is applied

Table 2: Data fields within the MDI Toolbox results structure.

5. Operating procedure

The MDI Toolbox is launched introducing MDIgui in the MATLAB command window. Figure 3 shows the initial window of the graphical interface. The first step consists of selecting the data set. The button Data from workspace permits loading a data set with missing values from the MATLAB workspace. A data set previously stored in Excel can be loaded clicking at Read Excel File (more details on the Excel data can be found in Appendix A). The button Use example opens a new window with example data (see Figure 4). This way 3 different data sets can be selected (see Section 4) with three percentages of missing values: 10%, 30% and 60%. For this tutorial the Simulated data set with 30% of missing values is selected. The missing data imputation method is also selected in the MDIgui window. The available methods are TSR, KDR, KDR-PCR, KDR-PLS, PMP, IA, modified NIPALS and DA. The recommended method is TSR, since it represents a good compromise solution between prediction quality, robustness against data structure and computation time [1].

The MDI interface allows also changing the settings of the different methods. In this way, the number of maximum iterations performed by the method and the tolerance for the convergence can be modified from their default values: 5000 iterations and a tolerance of 10^{-10}. These parameters are active when the regression-based methods, IA and NIPALS are active. If DA is selected as the imputation method, these settings are disabled and the Number of Markov chains and Chain Length are enabled. So the user can modify the default 100-iteration 10 Markov chains.

Once the data, method and settings have been introduced, the DataOverview window appears. The pattern of missing values and its percentage can be visualised here (see Figure 5). The red squares represent the missing entries in the data set, and the white ones the available values. After clicking Continue two progress bars appear one after the other. The first one shows the calculation progress of the variances and the second one the calculation progress of the covariances.

The next window, NumberComponents, allows the user to select the appropriate number of PCs for the PCA model. Three plots are presented here to assess this number (see Figure 6). On the left side the classical scree plot, with the eigenvalues of the estimated covariance matrix of $\mathbf{X}$. On the center, the cumulative percentage of explained variance. It is worth noting that both plots are obtained based on a pairwise estimation of the covariance matrix of the data set with missing values, *i.e.* the covariance between each pair of variables is computed using only rows with non-missing values in both variables. Using this procedure, pseudo-covariance matrices are obtained, *i.e.* they

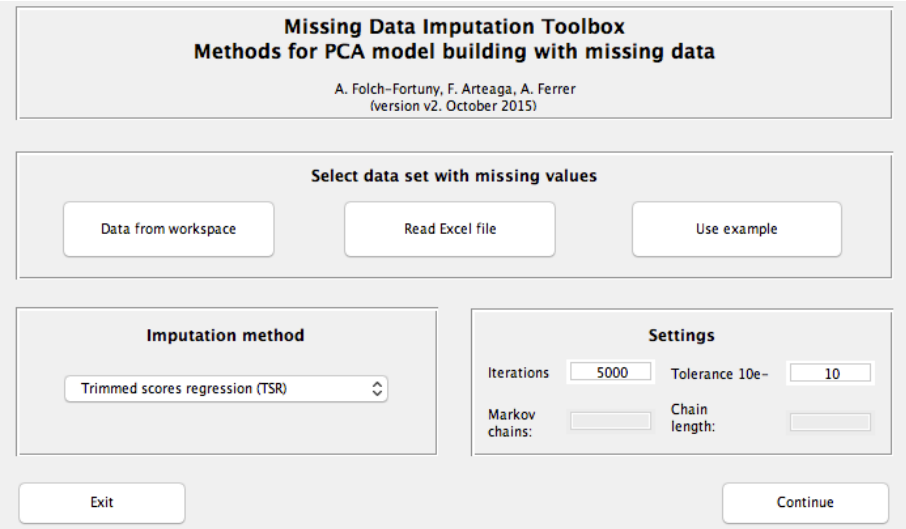

Figure 3: MDI Toolbox graphical interface for data, method and settings selection.

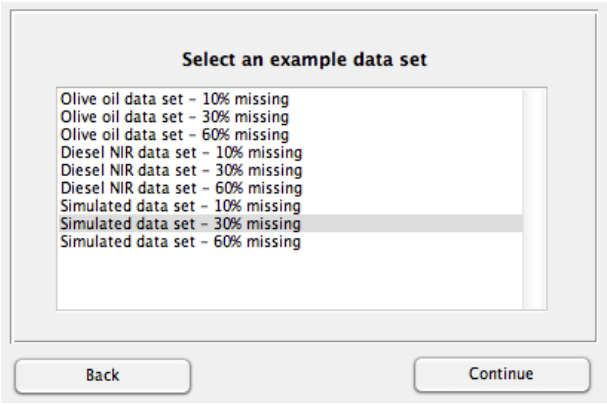

Figure 4: Example data selection window.

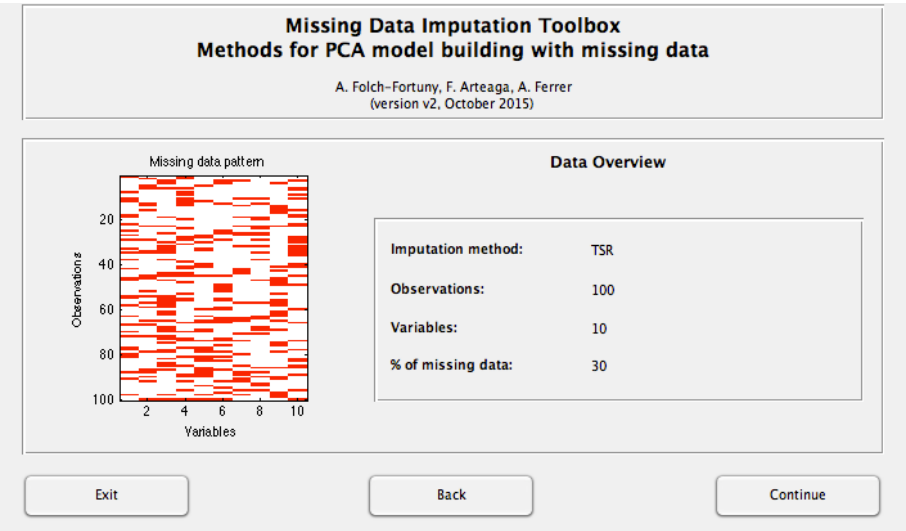

Figure 5: Graphical interface for data overview.

may be non-positive semidefinite. Since this matrix is used only to determine the number of PCs, corresponding to the highest eigenvalues, it is not important whether some negative eigenvalues are obtained by its Singular Value Decomposition (SVD). A third plot is included at the right side. This plot corresponds to the results of the column-wise k-fold (*ckf*) algorithm to estimate the number of PCs in PCA, recently proposed in [23]. This algorithm is an efficient adaptation of the previously proposed element-wise k-fold (*ekf*) algorithm, which is based on the capability of PCA to recover missing data [24]. Here, since our data set has, originally, missing values, the *ckf* algorithm can be used to select the number of components with the lowest sum of squares of the prediction error (PRESS). More details on the *ckf* algorithm can be found in [23]. The code for *ckf* algorithm has been taken from the Multivariate Exploratory Data Analysis (MEDA) Toolbox for MATLAB [25], and it can be downloaded separatedly from `https://github.com/josecamachop/MEDA-Toolbox/releases/tag/v1.0`. These three plots are included in the MDI Toolbox to give the practitioner different criteria to select the number of PCs, which is a critical issue even with complete data [24].

In this case study, the information provided by the three plots in Figure 6 is coherent, so three PCs are selected, since i) there is a huge difference in the eigenvalues between 3 and 4 components in the scree plot, and the differences are small between 4 and more components; ii) the cumulative explained variance with three components is around 90%, and the variance explained with 4 components is similar; and iii) the PRESS is minimum using three components.

Once the number of PCs is selected, MDI Toolbox runs the selected missing data method to impute the missing values. The computation time depends on the method selected. Usually TSR and IA are the fastest methods, and DA and KDR are the slowest ones.

Two progress bars appear simultaneously (see Figure 7) while the toolbox is performing the iterative imputations. The top bar shows the current iteration number, and runs until reaches the maximum number of iterations specified in the `MDIgui` initial window (see Figure 3). The bottom bar gives an idea of how far is the difference between consecutive iterations from the tolerance defined for convergence. This is calculated as $1 - \frac{d-l}{d}$ where d is the mean squared difference between the imputed values in consecutive iterations and l is the specified tolerance. The first progress bar that is fulfilled stops the calculations, therefore, if the iterations bar reaches the maximum, it implies that the established convergence criterion is not achieved.

The last window of MDI Toolbox is `ShowResults` (see Figure 8). Here, the details of the data imputation are summarised: imputation method, iterations, tolerance and computation time (in seconds). Also, two figures with the loadings and scores plots are shown to ease the graphical interpretation of the model. The axis of both plots can be changed via the pop-up menus.

Finally, MDI Toolbox returns automatically a data structure to the MATLAB workspace with all the informa-

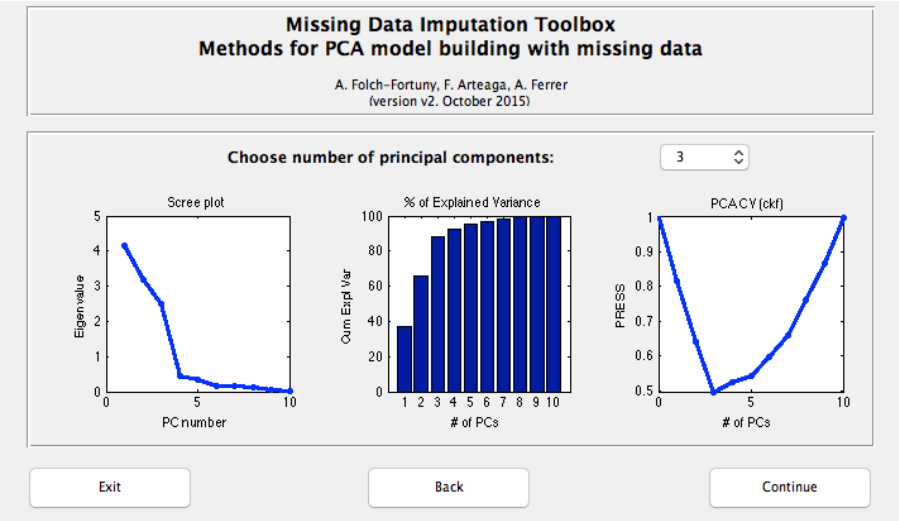

Figure 6: Selection of the number of principal components, based on the scree plot (left) and the cumulative explained variance bar plot (center), and the PCA cross validation using thje ckf algorithm.

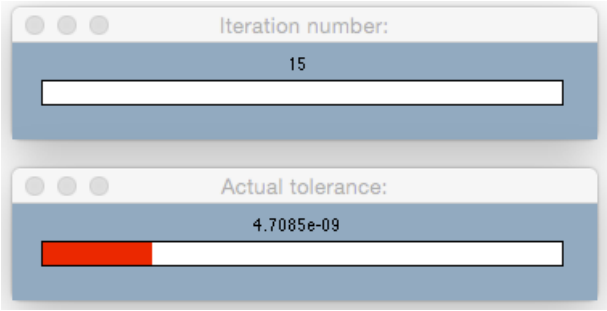

Figure 7: Progress bars reflecting the missing data imputation procedure. In this example 15 out of the 5000 iterations have been computed (top), and the mean squared difference between the imputed values in iterations 14 and 15 is 4.7089×10^{-9} (bottom).

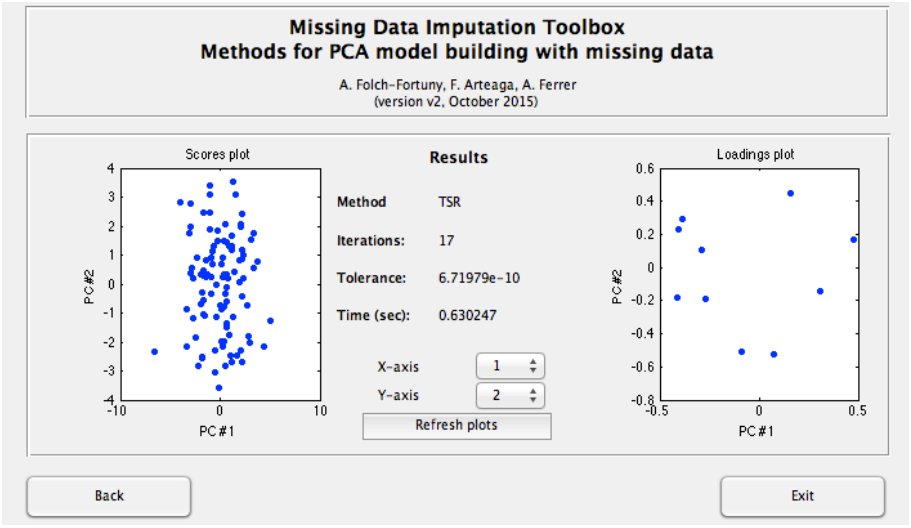

Figure 8: Scores and loadings plots from the PCA model fitted on the imputed data set.

tion of the data imputation (see Table 2). Among other parameters related to the number of iterations, computation time, *etc.* the original data set with imputed values are stored in the field `X_imputed` of the MATLAB structure `MDIToolbox_results`. Additionally, the resulting PCA model fitted on this data is stored in the fields `Loadings` and `Scores`, as well as the mean and the covariances of the variables. In this way, the data is reproduced as: $\texttt{X\_reconstructed} = \texttt{Mean} + \texttt{Scores} \times \texttt{Loadings}^{\mathrm{T}}$.

The missing data imputation methods available in MDI Toolbox can be used directly from the MATLAB command window. See Appendix B for more details.

6. Concluding remarks

In this paper a new MATLAB toolbox is presented devoted to impute missing data. MDI Toolbox includes PCA model building methods with missing data that are able to reconstruct the missing values coherently with the latent structure of the available data. Several methods from the literature are included in this toolbox: TSR, KDR, KDR-PCR, KDR-PLS, PMP, IA, modified NIPALS and DA. TSR is presented as the default method for its good performance with all data structures [1].

A graphical user-friendly interface is provided with the toolbox to ease its use. In this way, several windows guide the user step by step: from the data loading and settings to the results exploitation via interactive loadings and scores plots.

The purpose of MDI Toolbox is two-fold. On one hand, this toolbox permits to fit PCA models when there exist missing values in the original data set, obtaining as a result the loadings and the scores matrices. On the other hand, this toolbox can be used as a preprocessing step of other methodologies, since one of the outputs is simply the original data matrix with the imputed missing data.

The MDI toolbox is freely available for academic purposes at `http://mseg.webs.upv.es`, under a GNU license.

7. Validation

Dr. Edoardo Saccenti. Laboratory of Systems and Synthetic Biology, Wageningen University and Research Center, Dreijenplein 10, 6703 HB, Wageningen, The Netherlands.

Missing data is a ubiquitous problem in modern scientific research and can be particularly relevant in biomedical research.

The MDIToolbox offers a set of tools to perform missing data imputation using different imputation methods. The MDIToolbox is implemented in Matlab and comes with an elegant and intuitive graphical interface with a step-by-step guided procedure which makes the toolbox easy to use and easy to integrate with existing Matlab routines.

There are not many tools available for missing data imputation and the MDIToolbox is a very welcomed addition to the chemometrics toolbox which I warmly recommend.

Dr. Federico Marini. Department of Chemistry, University of Rome La Sapienza, P.le Aldo Moro 5, I-00185 Rome, Italy.

The MDI toolbox offers a comprehensive collection of chemometric tools for dealing with the imputation of missing data in a multivariate data analytical context, with particular focus on bilinear modeling. The toolbox is programmed in Matlab and it can be used either by running the specific functions from the command line (which also allows an easy integration with other existing routines), or through a very clear and nice graphical user interface. I've tried the MDI toolbox both on the example data provided by the authors and on some other data matrices of my own and I found it a very powerful tool to deal with chemometric problems involving the imputation of missing values. The structure of the GUI, which comes with rather straightforward buttons and menus allow an easy use also from people who may not be totally familiar with all the possible techniques for MDI. Moreover, the fact that each step is accompanied by an immediate graphical output allows the user to have an immediate idea of how the choices adopted up to a particular stage reflect in the outcomes at the successive ones. From a more chemometric standpoint, the MDI toolbox offer a comprehensive panorama of all the state of the art techniques for missing data imputation in chemometrics, allowing to choose an algorithm or another depending on the specific application and also allowing a fast and easy comparison among the outcomes of the different techniques on a particular problem. For all these

reasons, I would warmly recommend the MDI toolbox to both expert and non-expert users who could need to deal with problems involving missing data imputation, as it is a valuable, accurate and relatively easy to use tool.

Onno E. de Noord. Principal Research Scientist, Statistics & Chemometrics, Shell Global Solutions International BV, Grasweg 31, 1031 HW Amsterdam, The Netherlands.

The authors present a novel MATLAB toolbox for Missing Data Imputation. We have tested version 2 in our department and found it to be working well and easy to use. The toolbox comes with a useful README file that contains the most important information about the contents and usage, and with a few example datasets. The toolbox contains different approaches for missing data imputation, where several parameters can be varied. The GUI makes operation of the toolbox straightforward. In conclusion, this is a great addition to the chemometrics toolkit.

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.

Appendix A. Excel files.

To read Excel files with MDI Toolbox there have to be no headers nor observations names in the sheet. Also, there has to be only one sheet in the Excel file, containing the data set to analyse. The missing values have to appear as blank cells.

Appendix B. Using MATLAB command window.

The missing data imputation can be obtained typing the specific functions directly on the MATLAB command window. This way: [X, m, S, It, diff, Xrec] = pcambtsr(X_MD, A, M, f) imputes, using TSR, the missing values in matrix X_MD using Acomponents, a maximum of M iterations, and a tolerance f. The outputs are the original data matrix with the imputed values (X), the mean and covariance estimations (m and S, respectively), the number of iterations (It) and the tolerance value (diff). Also, the reconstructed matrix $\mathbf{X}$ using the final principal component analysis (PCA) model is obtained (Xrec).

To impute using the other regression-based methods, IA and modified NIPALS, the user only has to change the name of the function in the previous command: pcambkdr for KDR, pcambpcr for KDR-PCR, pcambpls for KDR-PLS, pcambpmp for PMP, pcambia for IA, pcambnipals for modified nonlinear iterative partial least squares regression (NIPALS) algorithm.

For DA, the user has to type: [X, m, S, Xrec] = pcambda(X_MD, M, CL, A), where X_MD and A are the matrix with missing values and the number of components of the final PCA model, and M and CL are the number of Markov chains and the chain length, respectively.

References

[1] 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.

[2] L. Brás, J. Menezes, Dealing with gene expression missing data, IEE Proceedings: Systems Biology 153 (3) (2006) 105–119.

[3] A. Folch-Fortuny, A. F. Villaverde, A. Ferrer, J. R. Banga, Enabling network inference methods to handle missing data and outliers, BMC Bioinformatics 16:283 (2015) 1–12.

[4] M. Zarzo, P. Martí, Modeling the variability of solar radiation data among weather stations by means of principal components analysis, Applied Energy 88 (8) (2011) 2775–2784.

[5] J. Quevedo, V. Puig, G. Cembrano, J. Aguilar, C. Isaza, D. Saporta, G. Benito, M. Hedo, A. Molina, Estimating missing and false data in flow meters of a water distribution network, in: 6th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, SAFEPROCESS 2006, Vol. 6, 2006, pp. 1181–1186.

[6] R. Magán-Carrión, F. Pulido-Pulido, J. Camacho, P. García-Teodoro, Tampered data recovery in WSNs through dynamic PCA and variable routing strategies, Journal of Communications 8 (11) (2013) 738–750.

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

[8] J. González-Martínez, O. de Noord, A. Ferrer, Multisynchro: A novel approach for batch synchronization in scenarios of multiple asynchronisms, Journal of Chemometrics 28 (5) (2014) 462–475.
[9] D. Visky, Y. Heyden, T. Iványi, P. Baten, J. De Beer, Z. Kovács, B. Noszál, P. Dehouck, E. Roets, D. Massart, J. Hoogmartens, Characterisation of reversed-phase liquid chromatographic columns by chromatographic tests: Rational column classification by a minimal number of column test parameters, Journal of Chromatography A 1012 (1) (2003) 11–29.
[10] I. T. Jolliffe, Principal Component Analysis, Springer Science & Business Media, 2002.
[11] 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, UK, 1983, pp. 147–188.
[12] 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.
[13] B. Walczak, D. Massart, Dealing with missing data, Chemometrics and Intelligent Laboratory Systems 58 (1) (2001) 15–27.
[14] P. R. Nelson, The treatment of missing measurements in PCA and PLS models, Ph.D. thesis, MacMaster University, Hamilton, Ontario, Canada (2002).
[15] 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.
[16] F. Arteaga, A. Ferrer, Framework for regression-based missing data imputation methods in on-line MSPC, Journal of Chemometrics 19 (8) (2005) 439–447.
[17] P. Geladi, B. Kowalski, Partial least-squares regression: a tutorial, Analytica Chimica Acta 185 (C) (1986) 1–17.
[18] J. L. Schafer, Analysis of Incomplete Multivariate Data, 1st Edition, Chapman and Hall/CRC, Boca Raton, 1997.
[19] M. Forina, C. Armanino, S. Lanteri, E. Tiscornia, Classification of Olive Oils from their Fatty Acid Composition, in: Food Research and Data Analysis, Elsevier Applied Science, London, UK, 1983, pp. 189–214.
[20] S. Hutzler, G. Bessee, Remote Near-Infrared Fuel Monitoring System, Tech. rep., United States of America (1997).
[21] 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.
[22] F. Arteaga, A. Ferrer, Building covariance matrices with the desired structure, Chemometrics and Intelligent Laboratory Systems 127 (2013) 80–88.
[23] 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.
[24] J. Camacho, A. Ferrer, Cross-validation in PCA models with the element-wise k-fold (ekf) algorithm: theoretical aspects, Journal of Chemometrics 26 (7) (2012) 361–373.
[25] J. Camacho, A. Pérez-Villegas, R. A. Rodríguez-Gómez, E. Jiménez-Mañas, Multivariate Exploratory Data Analysis (MEDA) Toolbox for Matlab, Chemometrics and Intelligent Laboratory Systems 143 (2015) 49–57.