Imputation for supervised learning problems in high dimension

24th International Conference on COMPUTATIONAL STATISTICS (COMPSTAT 2022)
23-26 August 2022, Bologna, Italy

Hadrien Lorenzo, Jérôme Saracco, *Inria Team ASTRAL, Université de Bordeaux*
Olivier Cloarec, *Sartorius*

Introduction

The problem of missing data often occurs in data analysis. Missing values of the type MAR (Missing At Random) are cosidered here. Then, the probability that a value is missing depends on one or multiple observed variables. Most modern algorithms focus on this type of missing values, and the most used implementations are certainly MICE, missForest, missMDA, or k-Nearest Neighbors imputations. To take into account sampling variability, it is better to propose M values for each missing value instead of a single one. This so-called "multiple imputation" procedure allows to provide proper imputation, in contrast to improper imputation. In practice, $M = 5$ is often sufficient. Most of the existing methods are not well suited to the high dimensional context, when the sample size n is much lower than the number of variables p, often symbolized as $n << p$. In supervised analysis, the dependent variable y must be explained by the explanatory variable x. This implies that the part of x associated with y can be hard to find, when the classical imputation methodologies suffer. In this communication, a new methodology, called Koh-Lanta, is presented. This methodology is able to deal with missing values in a supervised context, using multiple imputation, and tackling the high dimensional issues. For the sake of simplicity, missing values are considered only in the x part.

SoA: Joint modelling: PPCA and MICE

The PPCA[4] (Probabilistic PCA) model writes in our case as

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} = \boldsymbol{\mu} + \mathbf{W}\mathbf{t} + \boldsymbol{\varepsilon},$$

where $\mathbf{W}$ is deterministic, $\mathbf{t} \sim \mathcal{N}(\mathbf{0}, \mathbb{I}_R)$, R the number of components and $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbb{I}_{p+q})$. Then

$$\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} | \boldsymbol{\mu}, \mathbf{W}, \sigma^2 \sim \mathcal{N}\left(\mu, \mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbb{I}_{p+q}\right).$$

$$\mathbf{W}\mathbf{W}^\top + \sigma^2 \mathbb{I}_{p+1} = \begin{pmatrix} a_1 + \sigma^2 & \mathbf{A}_{(1)(-1,y)} \\ \mathbf{A}_{(1)(-1,y)}^\top & \mathbf{A}_{-1,y} + \sigma^2 \mathbb{I}_p \end{pmatrix} \text{ is invertible with } q = 1.$$

The conditional parameters write:

$$\mu_{1|-1,y} = \mu_1 + \mathbf{A}_{(1)(-1,y)} \left(\mathbf{A}_{-1,y} + \sigma^2 \mathbb{I}_{p+1}\right)^{-1} \left[(\mathbf{x}_{-1}, y)^\top - \boldsymbol{\mu}_{-1,y}\right],$$
$$\sigma^2_{1|-1,y} = a_1 + \sigma^2 - \mathbf{A}_{(1)(-1,y)} \left(\mathbf{A}_{-1,y} + \sigma^2 \mathbb{I}_p\right)^{-1} \mathbf{A}_{(1)(-1,y)}^\top,$$

- In high dimension, the evaluation of R would be underestimated practically hiding the subspace common to $\mathbf{x}$ and $\mathbf{y}$.
- In supervised context, the objective is to describe $\mathbf{y}|\mathbf{x}$, which is a conditional context, the joint analysis does not therefore seems appropriate.

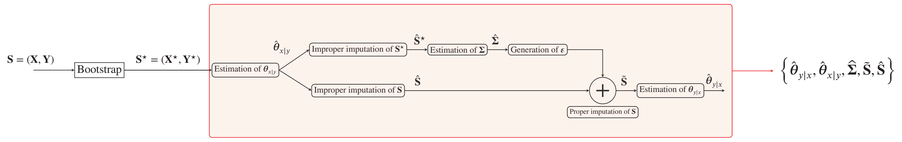
An other approach not based on joint modeling has been proposed and widely used in the literature, the MICE [1] approach: estimate for each variable with missing values the fully conditional regression model.

$\Longrightarrow$ Same problem!

Our proposition: "Koh-Lanta" , a "blocked Gibbs" solution

In order to estimate less parameters, our idea is to impute missing values of $\mathbf{x}_{i,\mathbf{m}}$ based solely on the response $\mathbf{y}_i$ and a currently computed conditional model $\mathbf{x}_{i,\mathbf{m}}|\mathbf{y}_i, \boldsymbol{\theta}_{\mathbf{x}|\mathbf{y}}$.

In order to take into account sampling variability and additive error model, we use a proper imputation approach with additive error sampling. The current section details a generalized version of the "MI-NIPALS" based on the estimation of the tuple $(\boldsymbol{\theta}_{y|x}, \boldsymbol{\theta}_{x|y})$. This algorithm is called "Koh-Lanta" .



Step name	Input	Output	Comments
Estimation of $\boldsymbol{\theta}_{x\|y}$	$\mathbf{S}^\star$	$\hat{\boldsymbol{\theta}}_{x\|y}$	From the bootstrapped sample $\mathbf{S}^\star$, evaluate $\hat{\boldsymbol{\theta}}_{x\|y}$.
Improper imputation of $\mathbf{S}$	$\mathbf{S}, \hat{\boldsymbol{\theta}}_{x\|y}$	$\tilde{\mathbf{S}}$	-
Improper imputation of $\mathbf{S}^\star$	$\mathbf{S}^\star, \hat{\boldsymbol{\theta}}_{x\|y}$	$\tilde{\mathbf{S}}^\star$	-
Estimation of $\boldsymbol{\Sigma}$	$\mathbf{S}^\star, \tilde{\mathbf{S}}^\star$	$\hat{\boldsymbol{\Sigma}}$	From non missing values, comparing $\mathbf{S}^\star$ and $\tilde{\mathbf{S}}^\star$.
Generation of ϵ	$\mathbf{S}^\star, \hat{\boldsymbol{\Sigma}}$	$\hat{\epsilon}$	-
Proper imputation of $\mathbf{S}$	$\mathbf{S}, \hat{\boldsymbol{\theta}}_{x\|y}, \hat{\epsilon}$	$\hat{\mathbf{S}}$	-
Estimation of $\boldsymbol{\theta}_{y\|x}$	$\hat{\mathbf{S}}$	$\hat{\boldsymbol{\theta}}_{y\|x}$	$\boldsymbol{\theta}_{y\|x}$ is estimated on the completed dataset.

ddsPLS [3] to estimate $\boldsymbol{\theta}_{y|x}$ and $\boldsymbol{\theta}_{x|y}$

ddsPLS (data-driven sparse PLS) is a PLS flavored approach and insures variable selection both in $\mathbf{X}$ and in $\mathbf{Y}$ with a single regularization coefficient per component:

$$\forall r \in [\![1, R]\!] \begin{cases} \textbf{(a}^\star\textbf{)}\ \mathbf{u}_r = \overrightarrow{\text{RSV}}\left(S_{\lambda^{(r)}}\left(\mathbf{M}^{(r)}\right)\right), \mathbf{v}_r = \overrightarrow{\text{RSV}}\left(S_{\lambda^{(r)}}\left(\mathbf{M}^{(r)\prime}\right)\right), \\ \textbf{(b)}\ \mathbf{t}_r = \mathbf{X}^{(r)}\mathbf{u}_r, \\ \textbf{(c)}\ \mathbf{p}_r = \mathbf{X}^{(r)\prime}\mathbf{t}_r / \mathbf{t}_r'\mathbf{t}_r, \\ \textbf{(d}^\star\textbf{)} \begin{cases} \boldsymbol{\Pi}_r = \text{diag}\left(\{\delta_{v_{r,j}\neq 0}\}_{j\in[\![1,q]\!]}\right), \\ \mathbf{c}_r = \underset{\mathbf{V}}{\arg\min} \left\|\mathbf{Y}^{(r)}\boldsymbol{\Pi}_r - \mathbf{t}_r\mathbf{V}'\right\|_2^2 = \left(\mathbf{Y}^{(r)}\boldsymbol{\Pi}_r\right)' \mathbf{t}_r/(\mathbf{t}_r'\mathbf{t}_r), \end{cases} \\ \textbf{(e)}\ \mathbf{X}^{(r+1)} = \mathbf{X}^{(r)} - \mathbf{t}_r\mathbf{p}_r', \ \mathbf{Y}^{(r+1)} = \mathbf{Y}^{(r)} - \mathbf{t}_r\mathbf{c}_r'. \end{cases}$$

with $\mathbf{M}^{(r)} = \mathbf{Y}^{(r)\prime}\mathbf{X}^{(r)}/(n-1)$ and $\forall x \in \mathbb{R},\ S_\lambda(x) = \text{sign}(x)\max(0, |x| - \lambda)$.

Simulation setting

Benchmark approaches

- "MI-NIPALS" NIPALS algorithm. $R \in [\![1, 5]\!]$ and LOO prediction error.
- "NIPALS" uses the NIPALS algorithm for simple imputation.
- "MEAN-PLS" Imputes missing values to mean. Then build PLS model.
- "missMDA-PLS" MI with PPCA[2]. Then build PLS model.

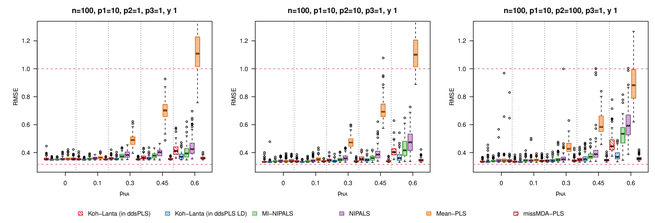
Simulation schema: $p = 2\mathrm{p}_1 + \mathrm{p}_2 + \mathrm{p}_3$, $q = 3$ and $\sigma = \sqrt{0.1}$:

$$x_j = \begin{cases} \sqrt{1-\sigma^2}\phi_1 + \sigma\epsilon_j \ \text{ for } \ j = 1..\mathrm{p}_1 \\ \sqrt{1-\sigma^2}\phi_2 + \sigma\epsilon_j \ \text{ for } \ j = \mathrm{p}_1 + 1..2\mathrm{p}_1 \\ \sqrt{1-\sigma^2}\phi_3 + \sigma\epsilon_j \ \text{ for } \ j = 2\mathrm{p}_1 + 1..2\mathrm{p}_1 + \mathrm{p}_2 \\ \epsilon_j \ \text{ for } \ j = 2\mathrm{p}_1 + \mathrm{p}_2 + 1..2\mathrm{p}_1 + \mathrm{p}_2 + \mathrm{p}_3 \end{cases} \quad \begin{cases} y_1 = \sqrt{1-\sigma^2}\phi_1 + \sigma\xi_1 \\ y_2 = \sqrt{1-\sigma^2}(\phi_1 + 2\phi_2)/\sqrt{5} + \sigma\xi_2 \\ y_3 = \xi_3 \end{cases}$$

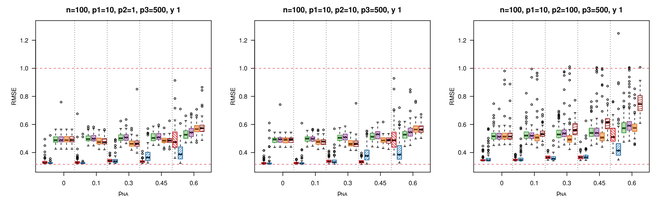
where $(\phi^\top, \epsilon_{1\ldots p}^\top, \xi_{1\ldots 3}^\top)^\top \sim \mathcal{N}_{3+p+q}(\mathbf{0}, \mathbb{I})$, variables $x_{1..2\mathrm{p}_1}$ to be selected.

Simulation results

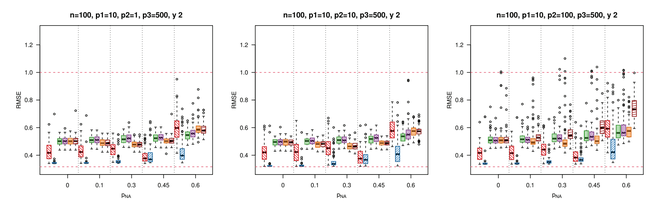
p_1	p_2	p_3	n	p_{NA}
10	$\in \{1, 10, 100\}$	$\in \{1, 100, 500\}$	$\in \{20, 50, 100\}$	$\in \{0, 0.1, 0.3, 0.6\}$



- "missMDA-PLS" and "Koh-Lanta (in ddsPLS LD)" win.



- "Koh-Lanta (in ddsPLS)" and "Koh-Lanta (in ddsPLS LD)" win.
- "missMDA-PLS" overfits more from low to high p_3 and large p_{NA}.



- **ddsPLS**'s difficulty to deal with intricate variables.

Conclusion

- **Mean** imputation fails again,
- **Joint Modelling** imputation seems to fail in high dimension,
- **Koh-Lanta** seems deal with NA in high dimension, but how to make the difference ?

References

[1] S van Buuren and Karin Groothuis-Oudshoorn. mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, pages 1–68, 2010.

[2] Julie Josse and François Husson. missmda: A package for handling missing values in multivariate data analysis. *Journal of Statistical Software*, 70(1):1–31, 2016.

[3] Hadrien Lorenzo, Olivier Cloarec, Rodolphe Thiébaut, and Jérôme Saracco. Data-driven sparse partial least squares. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 15(2):264–282, 2022.

[4] Michael E. Tipping and Christopher M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 61(3):611–622, 1999.