We would like to thank the reviewers for their thorough reading of the article and their many pertinent remarks, which help to improve the clarity. In the final version, we will address all comments on form. To address the reviewers' main concerns and better show the extent and feasibility of our methodology, we respond by adding an application on a recommendation system data. We consider the Jester dataset [2] of 5000 users who rated jokes, with 27% of missing values. The low-rank assumption for the loading matrix (allowed by Assump. 1) makes sense: any variable (i.e. user preferences) can be expressed as a linear combination of r latent variables[1] (hence, a "fully connected PPCA"). The first latent variable opposes individuals who like jokes about physics but dislike jokes about sexuality, and conversely.

MNAR mechanism. (R1, R3) Considering MNAR and self-masking values is plausible because users only rate jokes they like or dislike strongly or might be ashamed to assume their taste for sexual jokes. Note that the **self-masked assumption** is required only for the identifiability but the estimation strategy is also derived for **general MNAR mechanisms** (allowed by Assump. A2) where the missingness may depend on other missing variables. Assump. A2 means that a user's non-response for the sexual joke given all jokes may depend on the scores of the sexual and physical jokes but not on the musical and computer jokes.

Selecting the number r of latent variables and estimating the noise variance. (R1) To select r, one could use complete observations only but this is not possible when the number of features is large. As an alternative, we used both a cross-validation strategy assuming M(C)AR mechanism as detailed in [3] and also a beta implementation (that we coded) of a CV assuming MNAR mechanism. The second one is dependent on the chosen mechanism. As noted by the reviewers, Algorithm 1 is robust to a misspecification of the rank and thus a reasonable heuristic may already be enough. Both approaches estimate $r = 5$. CV was also used for Traumabase where oracle values were only used for synthetic data. With r at hand, the noise variance is obtained directly using weighted residual sum of squares as in [3].

Selecting the r pivot variables. (R1,R3) The next step consists in selecting r (M(C)AR) pivot variables (observed or M(C)AR variables imply Assump. A4) on which regressions[2] are performed [3]. Here, because we do not have further information on the missing mechanisms, we select the variables with the lowest missing rate. In Traumabase, the selection was discussed with experts (doctors) who identified M(C)AR variables. To reduce the error committed by a wrong selection of pivot variables, we suggest selecting a bigger set (> 5) and computing the final estimator with the median of the estimators over all possible combinations. In Fig. 2, by discarding outliers, this **aggregation approach** is more robust than selecting only r pivot variables.

Additional experiments. (R1,R4) Then, we test our method by introducing additional MNAR values on one variable (containing 33% NA) using a self-masked mechanism leading to 65% NA. In Fig. 1, **our method (`MNAR`) outperforms all the others on rating data** including `Deep` [1] which imputes MNAR values using deep generative models (R4)[4]. The parametric method `MNARparam` is not displayed as it does not scale on such large data. The code for the whole methodology was already available, but now recast as a **beta version of package** (R1) and submitted soon on CRAN.

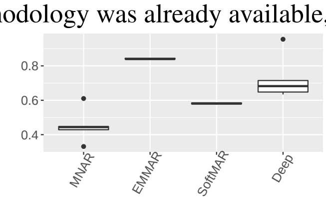

Fig. 1: Prediction error (difference between true values and predicted ones) for the Jester dataset, the mean imputation corresponding to an error of 1. The process of drawing additional MNAR values and predicting them is repeated 10 times which gives the stochasticity.

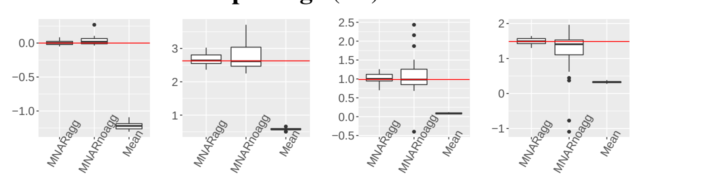

Fig. 2: Synthetic data from Section 4.1, with Algorithm 1 performed with aggregation (`MNARagg`) or not (`MNARnoagg`). True values in red, estimated values (means, variance, cov) in boxplot. For a given set of PPCA parameter, the stochasticity comes from the process of drawing 20 times the latent variables, the additive noise and the missing-data pattern (R3).

Comparison with Miao et al. [4] (R2,R3) For one variable $Y \sim \mathcal{N}(\mu, \sigma^2)$, Miao et al. prove identifiability of the variance and the absolute value of the mean, assuming a self-masked mechanism with a known strictly monotone form (including classical Probit and Logit). They cannot get identifiability for the mean (not the absolute value) with Logit. We have used their result to prove variance identifiability in PPCA and provide a genuine proof for the mean without discarding Logit. Secondly, for a specific setting of an heteroscedastic regression model with missing values only in Y, where the variance of Y given the observed covariates is injective, they provide identifiability results for the conditional distribution with general MNAR. This setting and the proof are too restricted to be considered in PPCA.

Supervised learning task on Traumabase. (R3) To predict the administration or not of the tranexomic acid (binary variable), we impute explanatory variables before proceeding to the classification task. In Tab. 1, our method gives the smallest prediction error.

MNAR	5.06%
EMMAR	5.82%
SoftMAR	5.45%
MNARparam	5.39%
Mean	5.27%

Tab. 1: Mean of prediction error over 10 repetitions.

[1] L. Gondara and K. Wang. Mida: Multiple imputation using denoising autoencoders. In *PAKDD*, 2018.
[2] M. Hahsler. recommenderlab: A framework for developing and testing recommendation algorithms. Technical report, 2015.
[3] J. Josse and F. Husson. Selecting the number of components in principal component analysis using cross-validation approximations. *CS&DA*, 2012.
[4] Wang Miao, Peng Ding, and Zhi Geng. Identifiability of normal and normal mixture models with nonignorable missing data. *JASA*, 2016.

[1] It does not require that the linear combination coefficients are non zero. [2] assumed to be consistent by Assump. A3, which holds as the noise tends to 0. (R1) [3] Note that our method is not based on the complete-case of the dataset but on the complete-case of the r ($\ll p$) pivot variables (R3). [4] Note that this method requires to be trained on a complete dataset.