HYBRID METHODS FOR ROBUST TABULAR DATA IMPUTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Missing data are a fundamental challenge in statistical analysis and machine learning, as the choice of imputation method substantially impacts downstream inference. In this work, we propose two hybrid imputation methods called NuclearForest and SoftForest, which combine nuclear-norm-based low-rank initialization using Singular Value Thresholding (SVT) and SoftImpute, respectively, with a non-iterative Random Forest refinement. For the SVT-based component, we further introduce an adaptive step-size rule, prove adaptive step-size bounds, and establish convergence for the corresponding zero-initialized iteration. The low-rank initialization provides a structured warm start that captures the global covariance patterns in the data, while the subsequent Random Forest step recovers residual nonlinear signals encoding local dependencies. We conduct an extensive benchmark on diverse datasets from different application domains, comparing the proposed methods with seven established imputation methods under the Missing Completely at Random (MCAR), Missing at Random (MAR), and Missing Not at Random (MNAR) mechanisms across varying missingness rates. Our results demonstrate that NuclearForest and SoftForest match or exceed the imputation fidelity of state-of-the-art iterative methods such as MissForest, while significantly reducing computational cost. In particular, they achieve speedups of approximately $5.81\times$ and $9.52\times$ over MissForest by replacing iterative cycles with a single refinement step. Our approach effectively exploits the low-rank structure of real-world tabular data and accommodates mixed-type variables, providing an efficient and robust solution for data imputation in bioinformatics, economics, and beyond.

1 INTRODUCTION

Motivation Missing values are ubiquitous in real-world biomedical datasets, and data imputation is a critical prerequisite in the preprocessing pipeline. Consequently, the reliability of downstream analyses, including statistical testing and classification using machine learning methods, depends directly on the quality of the imputed data. However, state-of-the-art Random-Forest-based imputation methods, such as MissForest (Stekhoven & Bühlmann, 2012), require repeated fitting of predictive models until convergence, resulting in substantial computational cost. This computational complexity limits the practicality of iterative Random-Forest-based imputation methods when imputation must be performed repeatedly across experimental conditions. We introduce NuclearForest and SoftForest, two hybrid imputation methods that combine low-rank matrix completion with non-iterative Random Forest refinement, substantially reducing runtime while preserving high-quality imputations on tabular datasets.

Background Missing-data imputation can be approached from several complementary paradigms. Classical theory distinguishes between missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR), since the validity of an imputation strategy depends on the mechanism that generated the missing entries (Rubin, 1976). Practical imputation methods span simple univariate replacement, model-based multiple imputation such as MICE (Van Buuren & Groothuis-Oudshoorn, 2011), and non-parametric machine-learning approaches such as MissForest (Stekhoven & Bühlmann, 2012). A complementary line of work exploits global correlation structures through low-rank matrix completion and nuclear-norm regularization, including Singular Value Thresholding (SVT) (Cai et al., 2010) and SoftImpute (Mazumder et al., 2010).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

MissForest, proposed by Stekhoven & Bühlmann (2012), is a random-forest-based imputation method that has been shown to perform well on biological datasets, outperforming imputation methods such as kNN by modeling complex interactions and nonlinear relationships. Subsequent studies have shown that random-forest-based imputation can achieve strong imputation accuracy in datasets with nonlinear relationships and mixed variable types, with particularly favorable performance reported under MCAR/MAR missingness in metabolomics benchmarks (Tang & Ishwaran, 2017; Wei et al., 2018). We therefore choose random forests for the refinement step because our target setting is tabular imputation rather than representation learning from large unstructured data. For medium-sized tabular datasets, Random-Forest-based models remain state-of-the-art and outperform neural-network baselines even under extensive hyperparameter-search budgets (Grinsztajn et al., 2022). Moreover, random forests are a standard non-parametric regression tool and often perform reasonably well under default hyperparameter settings (Breiman, 2001; Probst et al., 2019). For missing-data imputation, this property is particularly valuable: random-forest-based models can capture nonlinear effects and feature interactions while accommodating both numerical and categorical variables, which motivates their use in MissForest (Stekhoven & Bühlmann, 2012). However, MissForest often is computationally expensive, as it repeatedly fits variable-wise random forests until convergence. The simple univariate initialization used before the iterative updates may provide a limited starting point, since it does not explicitly impose low-rank or other global structural assumptions on the completed matrix.

The SVT algorithm (Cai et al., 2010) is a classical first-order method for nuclear-norm-based matrix completion. Its appeal lies in a simple iterative structure, where each iteration alternates between singular-value shrinkage and a residual update. However, because standard SVT relies on fixed algorithmic parameters, including the singular-value threshold and the step size, its practical convergence speed can be sensitive to these choices and may require many iterations when they are not well matched to the data. One response in the literature is to adapt the threshold during the iteration (Zarmehi & Marvasti, 2017).

Approximate low-rank structure in real-world datasets. Many real-world datasets exhibit an approximately low-rank structure, as their variation is often governed by a relatively small number of latent factors. This is particularly relevant for omics data, here referring to tabular measurements of molecular abundances, where correlated biological pathways and shared regulatory mechanisms induce dependencies among measured features (Hilafu et al., 2020). Therefore, methods based on low-rank matrix completion provide a natural framework for omics data imputation by exploiting latent structure, but they may not fully capture nonlinear dependencies in the data.

Missingness mechanisms in MS-omics data. Mass-spectrometry (MS) omics data commonly contain missing values, and the missingness mechanism can affect imputation performance and downstream analysis (Sethi & Brietzke, 2016; Wei et al., 2018). Using the metabolomics dataset from Wei et al. (2018), we benchmark imputation methods under two artificial missingness settings: a joint MCAR/MAR condition and a separate MNAR condition. Because MCAR and MAR are often difficult to distinguish in MS-based metabolomics, the joint MCAR/MAR setting follows the benchmark design of Wei et al. (2018) and is generated by uniformly masking observed entries. In an additional experiment using a housing dataset (Harish Kumar Data Lab, 2024), MCAR and MAR are evaluated separately to further assess the behavior of the imputation methods under distinct missingness mechanisms. MNAR is modeled as left-censored missingness, where low-abundance features fall below the limit of detection or quantification. Accordingly, we generate MNAR missingness by selecting features and removing observations below variable-specific quantile cutoffs (Wei et al., 2018).

Benchmark methods and evaluation metrics. In this paper, real-world data from different applications will be imputed and evaluated using nine different imputation methods: MissForest, kNN, Mean, Median, Half-min, SVT, SoftImpute, and our proposed methods NuclearForest and SoftForest. We evaluate imputation quality from multiple complementary perspectives, covering normalized root mean squared error (NRMSE), global sample structure, univariate group-level signals, distributional similarity, and downstream predictive utility. Specifically, we use NRMSE and an NRMSE-based sum of ranks (SOR) score for masked-entry reconstruction; PCA/PLS Procrustes analysis for structural preservation in reduced-dimensional spaces; Student's t-test followed by Pearson correlation analysis

for preservation of univariate group differences; and Gower's distance together with downstream predictive R^2 degradation to assess distributional and task-level effects.

Contributions

1. **Adapted SVT.** We introduce an efficient adaptive SVT variant (Algorithm 2) for imputation that replaces zero initialization with column-mean warm starting and modulates the step size according to the relative observed-entry reconstruction error, rather than relying on a fixed step size δ. Specifically, δ_k is contracted by a factor of 0.9 when the error increases and expanded by a factor of 1.05, up to $2p$, when the error decreases, where $p = |\Omega|/(n_1 n_2)$ is the observed-entry ratio. Under a finite-contraction condition, the adaptive step sizes remain bounded away from zero and satisfy $\delta_k < 2$. Proposition 1 proves these bounds, while Corollary 1 gives convergence and an $O(T^{-1})$ rate for the corresponding zero-initialized iteration. In addition, an ablation study separately evaluates column-mean warm starting and adaptive step-size modulation, indicating that both components contribute to improved imputation quality. See Appendix D.3.
2. **Hybrid imputation algorithms.** We propose NuclearForest and SoftForest, two hybrid imputation methods that combine low-rank initialization with a single subsequent random-forest refinement step. NuclearForest uses the proposed adaptive SVT variant as its initialization module, whereas SoftForest uses SoftImpute as the corresponding low-rank initializer. Across the evaluated real-data settings, both methods achieve competitive imputation quality relative to iterative random-forest baselines while reducing runtime by approximately $5.81\times$ for NuclearForest and $9.52\times$ for SoftForest in one representative experiment.
3. **Comprehensive real-data benchmark.** We provide a broad empirical evaluation of nine imputation methods on metabolomics and housing datasets under MCAR, MAR, and MNAR missingness mechanisms. The benchmark compares methods across reconstruction, structural, statistical, and downstream predictive metrics, allowing us to assess not only masked-entry reconstruction error but also the preservation of low-dimensional sample structure, group-level signals, and predictive utility.

2 METHODOLOGY

Matrix completion via nuclear norm minimization. Let $M \in \mathbb{R}^{n_1 \times n_2}$ be a partially observed matrix, and an entry of M is indexed by a pair (i, j), where $i \in \{1, \dots, n_1\}$ is the row index and $j \in \{1, \dots, n_2\}$ is the column index. Let $\Omega \subseteq \{1, \dots, n_1\} \times \{1, \dots, n_2\}$ denote the index set of observed entries. The sampling operator $\mathcal{P}_\Omega$ is defined entrywise by $\big(\mathcal{P}_\Omega(X)\big)_{ij} = \begin{cases} X_{ij}, & (i,j) \in \Omega, \\ 0, & (i,j) \notin \Omega. \end{cases}$. The matrix completion problem aims to recover a complete matrix X that agrees with the partially observed matrix M on the observed index set Ω. Since infinitely many matrices can satisfy these observed constraints, recovery is ill posed without additional a priori assumptions about the expected solution. A standard assumption is that the underlying complete matrix has a low-rank structure, meaning that much of its variation can be represented in a low-dimensional linear subspace. This assumption leads to the following rank-minimization formulation:

$$\min_{X \in \mathbb{R}^{n_1 \times n_2}} \operatorname{rank}(X) \qquad \text{s. t.} \qquad \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(M). \tag{1}$$

Since rank minimization is nonconvex and NP-hard (Recht et al., 2010), the optimization problem equation 1 is commonly replaced by a convex relaxation (Candes & Recht, 2012)

$$\min_{X \in \mathbb{R}^{n_1 \times n_2}} \|X\|_* \qquad \text{s. t.} \qquad \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(M), \tag{2}$$

where $\|X\|_* = \sum_i \sigma_i(X)$ denotes the nuclear norm, i.e., the sum of the singular values of X. This leads to a convex formulation of the matrix completion problem based on nuclear norm minimization. Solutions to this relaxed problem can be efficiently computed via singular value thresholding methods.

Singular Value Thresholding (SVT). The SVT algorithm was introduced by Cai et al. (2010) as a first-order method for solving nuclear-norm minimization problems. In the matrix completion setting, instead of treating the constrained nuclear-norm problem equation 2 directly, the method is derived

from the closely related regularized problem

$$\min_X \ \tau\|X\|_* + \frac{1}{2}\|X\|_F^2 \qquad \text{s. t.} \qquad \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(M),$$

where $\tau > 0$ is a threshold parameter and $\|\cdot\|_F$ denotes the Frobenius norm. For large values of τ, the minimizer of this problem approaches the minimum-Frobenius norm solution of the corresponding nuclear-norm minimization problem. To derive the SVT algorithm, one introduces the following Lagrangian $\mathcal{L}(X, Y) = \tau\|X\|_* + \frac{1}{2}\|X\|_F^2 + \langle Y, \mathcal{P}_\Omega(M - X)\rangle$, where Y is the dual variable associated with the equality constraint. Following the Uzawa interpretation of Cai et al. (2010), this yields an iterative scheme that alternates between minimizing the Lagrangian with respect to the primal variable X and performing a dual gradient step in the dual variable Y. Starting from $Y^0 = 0$, the SVT iteration is given by

$$X^k = \mathcal{S}_\tau(Y^{k-1}), \qquad Y^k = Y^{k-1} + \delta_k \, \mathcal{P}_\Omega(M - X^k),$$

where $\delta_k > 0$ denotes the step-size parameter. In the classical SVT implementation of Cai et al. (2010), this step size is typically chosen to be constant, i.e., $\delta_k = \delta$. The operator $\mathcal{S}_\tau$ denotes the proximal operator of the nuclear norm given by $\mathcal{S}_\tau(Y) := \operatorname{prox}_{\tau\|\cdot\|_*}(Y) = \arg\min_X \left\{\tau\|X\|_* + \frac{1}{2}\|X - Y\|_F^2\right\}$. If $Y = U\Sigma V^T$ is a singular value decomposition of Y, with $\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_r)$, where $\sigma_1, \ldots, \sigma_r$ are the singular values of Y, then $\mathcal{S}_\tau$ acts by soft-thresholding the singular values through:

$$\mathcal{S}_\tau(Y) = U \ \operatorname{diag}\left((\sigma_i - \tau)^+\right) V^T, \qquad (a)^+ := \max(a, 0). \tag{3}$$

Hence, each SVT iteration alternates between two operations: singular value thresholding, which promotes a low-rank structure, and a dual update on the observed entries, which drives the reconstruction toward consistency with the available data.

Improved SVT variant. In this work, we propose an improved variant of the classical SVT method with two key modifications. First, instead of initializing the missing entries as zero, we use a column-mean warm start, which provides an inexpensive data-dependent initialization that preserves the observed entries and places the initial completion on the empirical feature scale, thereby reducing the burden on subsequent low-rank recovery and random-forest refinement. Second, we replace the empirical fixed step size $\delta = 1.2/p$ (where $p = \frac{|\Omega|}{n_1 n_2}$ is the observed-entry rate of the matrix) used in the original SVT implementation with an adaptive rule: $\delta_0 = 1.2p$, followed by multiplicative expansion or contraction according to the observed-entry residual, with a cap at $\delta_{\max} = 2$. Under the finite-contraction assumption, the adaptive step sizes remain bounded away from zero. In our experiments, this condition is empirically satisfied, as the residual-based rule does not trigger contraction steps on the evaluated real datasets. Thus, the adaptive step sizes remain within the standard SVT admissible range (Cai et al., 2010). Proposition 1 gives the step-size bounds, and Corollary 1 gives the zero-initialized convergence result. Instead of using an adaptive threshold as proposed by Zarmehi and Marvasti (Zarmehi & Marvasti, 2017), we keep τ fixed, matching the original SVT parameter settings. In all numerical experiments, we compute the SVT updates using a full singular value decomposition.

SoftImpute. In contrast to the SVT algorithm, SoftImpute, introduced by Mazumder et al. (2010), approaches matrix completion through the following penalized nuclear norm problem

$$\min_X \ \frac{1}{2}\|\mathcal{P}_\Omega(M - X)\|_F^2 + \lambda\|X\|_*, \tag{4}$$

where $\lambda > 0$ is a fixed regularization parameter. This objective is convex: the first term penalizes reconstruction error on the observed entries, while the nuclear-norm term promotes low-rank structure through the same convex surrogate of the rank function used in nuclear-norm matrix completion. Therefore, SoftImpute is closely related to the SVT formulation, but replaces the hard constraint $\mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(M)$ with a penalized reconstruction term on the observed entries.

To solve this minimization problem, SoftImpute proceeds as follows. Given a current iterate X^k, the method defines

$$W^k = \mathcal{P}_\Omega(M) + \mathcal{P}_\Omega^\perp(X^k),$$

where $\mathcal{P}_{\Omega}^{\perp}$ denotes the complementary projection onto the unobserved entries. This leads to the following equivalent reformulation:

$$W^k = X^k + \mathcal{P}_{\Omega}(M - X^k).$$

Thus, W^k coincides with the observed data on Ω and with the current iterate on the complementary index set. The next iterate is then defined by $X^{k+1} = S_{\lambda}(W^k)$, where S_{λ} denotes the singular value soft-thresholding operator introduced in equation 3. Hence, each iteration of SoftImpute consists of an imputation step in which the missing entries are filled-in using the current iterate, followed by a singular value thresholding step that promotes a low-rank structure. The parameter λ determines the amount of shrinkage applied to the singular values. In contrast to SVT, SoftImpute is formulated entirely in terms of the primal variable. Let $X^{\star}$ denote a solution of the penalized nuclear norm minimization problem. Then the objective values generated by SoftImpute decrease monotonically and converge to the optimal value. Moreover, the convergence rate is of order $\mathcal{O}(1/k)$.

Random forest regression. As a random-forest-based iterative imputation baseline, we use a MissForest-style procedure in which each incomplete column is treated as a supervised regression problem. Let $X^{(t)} \in \mathbb{R}^{n_1 \times n_2}$ denote the completed matrix at iteration t. At each iteration, incomplete columns are visited sequentially. For an incomplete column j, a random forest regressor $f_j^{(t)}$ is trained on the observed rows $\{i : (i,j) \in \Omega\}$, using the current imputed predictor vectors $X_{i,-j}^{(t)}$ and the observed responses M_{ij}. Here, $X_{i,-j}^{(t)}$ denotes the row-i predictor vector formed by all columns except the target column j. The trained model is then used to update the missing entries in column j: $X_{ij}^{(t+1)} = f_j^{(t)}\left(X_{i,-j}^{(t)}\right), (i,j) \notin \Omega$, while observed entries are kept fixed. This round-robin procedure is repeated until either a maximum number of iterations is reached or the change between consecutive completed matrices falls below a predefined tolerance.

NuclearForest and SoftForest. We propose two hybrid imputation methods that combine low-rank completion with nonlinear random-forest refinement. NuclearForest first obtains a low-rank initialization using SVT, whereas SoftForest uses SoftImpute. This design changes the role of the random forest compared with fully iterative random-forest imputation. Rather than starting from a simple univariate initialization and repeatedly updating incomplete features until convergence, as in MissForest-style methods, our methods first construct a structured low-rank completion and then apply a single random-forest refinement pass. For each incomplete feature, a random-forest regressor is trained on the rows where that feature is observed, using the corresponding matrix completed by SVT or SoftImpute as the predictor space. The fitted model is then applied once to the missing entries of that feature, while the originally observed entries remain fixed. The single-pass design is deliberate: the low-rank stage already captures global correlation structure, so the random forest is used primarily as a nonlinear residual corrector that refines the low-rank estimate by modeling interaction effects and nonlinear dependencies. Additional refinement passes are possible and may slightly improve accuracy in some settings, but they increase computational cost and provide only marginal further gains in our experiments. We therefore use the single-pass random-forest refinement to emphasize the accuracy–efficiency trade-off of the proposed hybrid framework.

3 Experiments and results

In our numerical experiments, we benchmark nine imputation methods on two datasets under three missingness mechanisms. For each setting, missing entries are regenerated independently over 10 repeated experimental runs, ensuring different missingness masks across runs while preserving reproducibility. Neural-network-based imputation methods are not included as primary baselines because the focus of this study is on efficient classical and hybrid tabular imputers. This choice is consistent with recent tabular-learning benchmarks showing that tree-based models remain highly competitive and can outperform neural-network baselines on many medium-sized tabular datasets (Grinsztajn et al., 2022). On the publicly available metabolomics dataset (Wei et al., 2018), we use the four evaluation metrics adopted by Wei et al. (2018). Since MAR is commonly approximated by MCAR in metabolomics benchmarks, we evaluate MCAR/MAR via uniform random masking and MNAR via left-censored missingness. On the housing dataset, we use the same four metrics together with three additional task-specific metrics. For the MAR setting, we generated missing values using

a conditional logistic masking mechanism, where the missingness probability of each variable was modeled as a function of other variables rather than the variable's own value (Schouten et al., 2018). Runtime is reported as wall-clock time in seconds and summarized as mean ± standard deviation (std) over repeated runs under the same local computational environment. All experiments were run on macOS 14.1 using an Apple M3 Pro processor with 11 CPU cores and 18 GB of RAM. The code of this benchmark experiment as well as of the two proposed imputation algorithms will be publicly released upon acceptance to ensure reproducibility.

Ablations and SVD implementation in the SVT. We evaluate the contribution of each design choice by isolating (i) zero initialization versus column-mean warm start and (ii) fixed versus adaptive step size. Across the four metrics (NRMSE, PCA Procrustes distance, p-value correlation, and PLS Procrustes distance), both components improve imputation quality, with the full method performing best overall under MCAR/MAR missingness. Fig. 1 summarizes the NRMSE ablation, while the complete ablation results are reported in Appendix D.3. We also compare dense full SVD with partial SVD. The original SVT implementation of Cai et al. (2010) exploits the sparsity of the dual iterate Y^k and computes only the dominant singular values needed by the shrinkage operator, using PROPACK/Lanczos bidiagonalization. This is appropriate for large sparse matrix completion problems. In contrast, our experimental matrices are dense after initialization. On the metabolomics dataset (198×131), the full-SVD implementation achieved a $2.89\times$ speedup over the partial-SVD SVT variant, with indistinguishable imputation error (Appendix D.2). For this reason, we use the full SVD in all experiments. For large sparse datasets, one may instead use partial SVD approximations, which are commonly employed to reduce the cost of singular-value shrinkage when only the leading singular components are needed.

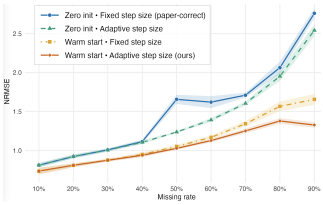
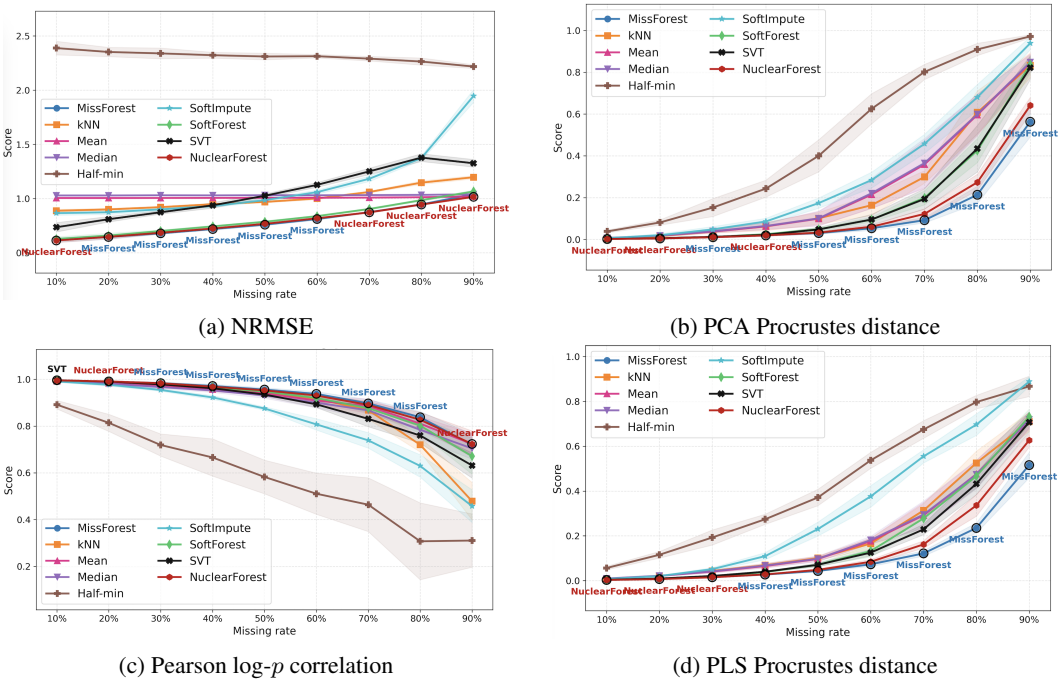


Figure 1: Ablation study.



(a) NRMSE

(b) PCA Procrustes distance

(c) Pearson log-p correlation

(d) PLS Procrustes distance

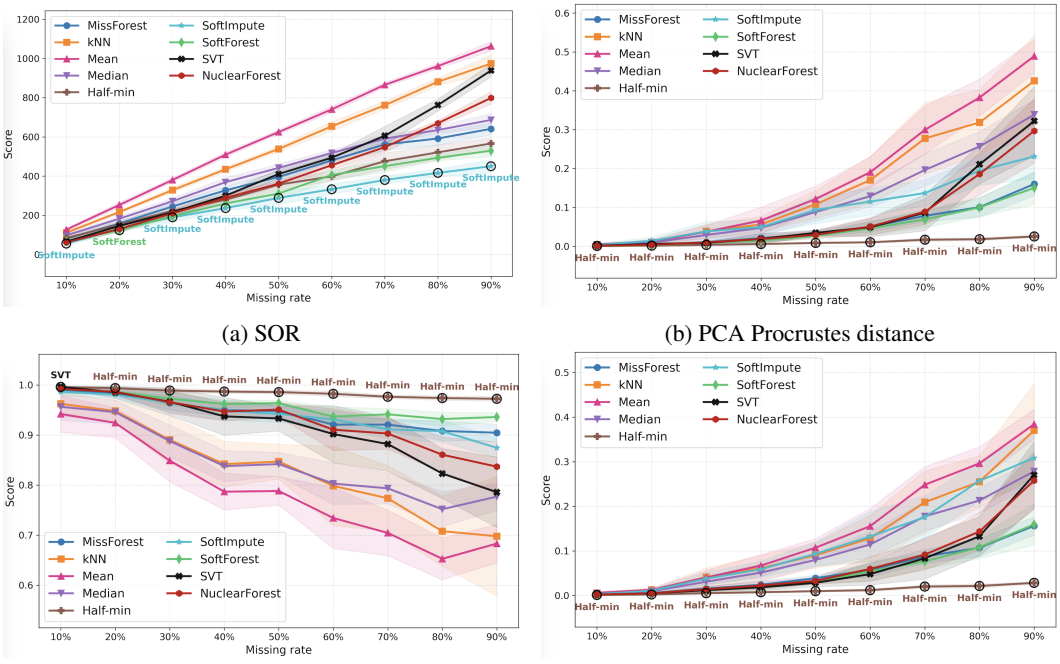
Figure 2: Comparison of imputation methods on the metabolomics dataset under MCAR/MAR.

Metabolomics dataset under MCAR/MAR missingness. NRMSE is computed on masked entries after column-wise z-score standardization of both the ground-truth and imputed matrices, using means and standard deviations estimated from the complete ground-truth matrix. This transformation enables an unbiased comparison when using NRMSE by standardizing variable scales, preventing

variables with larger values from dominating the evaluation (Wei et al., 2018). As shown in Fig. 2a, the imputation quality of MissForest, NuclearForest, and SoftForest is very close to each other with respect to the NRMSE metric. As can be observed, MissForest performs slightly better for missing rates between 20% and 60%. Next, PCA is performed using the first two principal components, as they capture the greatest variance in the data (Wei et al., 2018). Moreover, the symmetric Procrustes sum of squared errors is computed to compare the distribution. In Fig. 2b, NuclearForest yields the lowest PCA Procrustes distance at 10%, 20%, and 40% missingness. Following Wei et al. (2018), we evaluate whether imputation preserves univariate group-difference signals by conducting Welch's two-sample t-tests for each variable between groups in the complete and imputed data, and then computing the Pearson correlation between the resulting log-transformed p-values. As shown in Fig. 2c, NuclearForest achieves the highest Pearson log-p correlation values at 20% and 90% missingness and remains competitive across the remaining missing rates, where MissForest obtains the best results. PLS Procrustes analysis is used to quantify the structural distortion. NuclearForest achieves the lowest PLS Procrustes distance for missing rates between 10% and 30% in Fig. 2d. As shown in Table 1, the hybrid methods (SoftForest and NuclearForest) remain significantly faster, providing ~9.52× and ~5.81× speedups, respectively. Further runtime analysis is provided in Appendix D.4.

Table 1: Runtime comparison for metabolomics dataset, reported in seconds as mean ± std.

Mechanism	MissForest	SoftForest	NuclearForest
MCAR/MAR	77.787 ± 21.426	8.171 ± 1.931	13.389 ± 2.365
MNAR	89.438 ± 22.845	4.344 ± 2.137	9.797 ± 2.381



(a) SOR

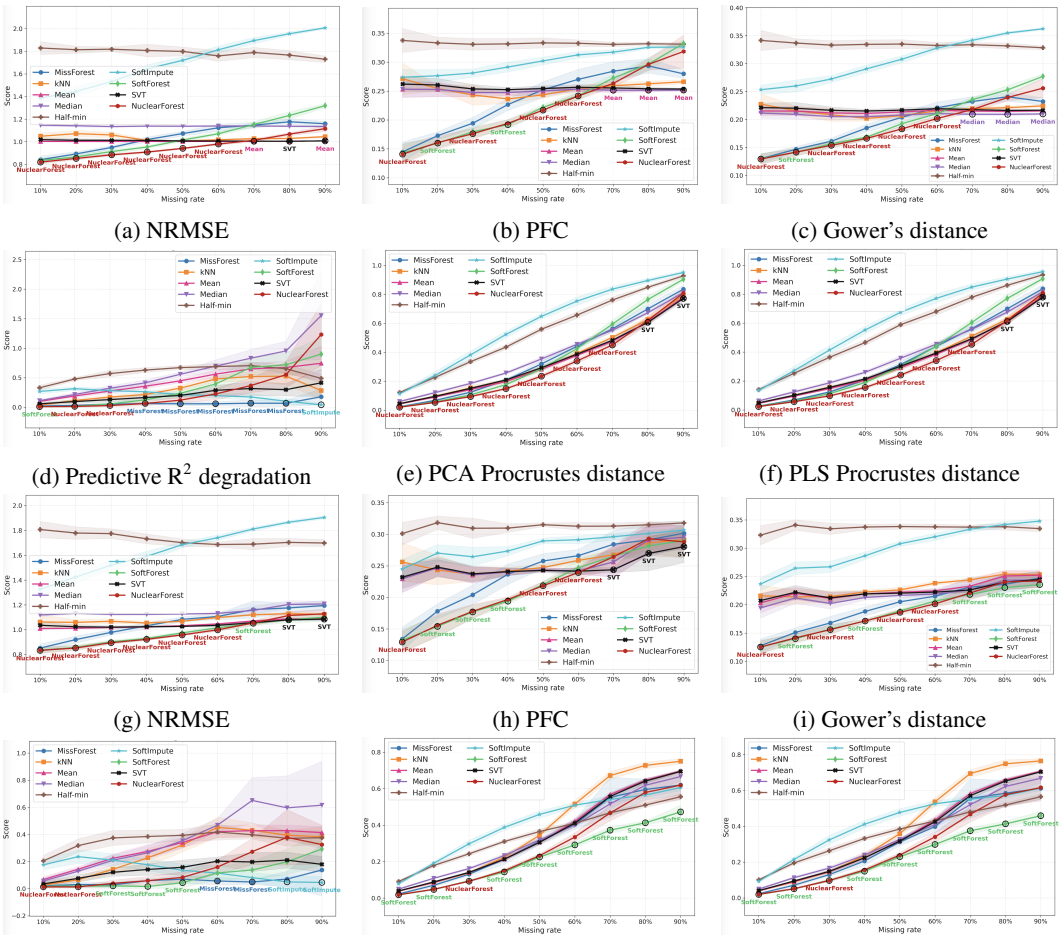
(b) PCA Procrustes distance

(c) Pearson Log-p correlation

(d) PLS Procrustes distance

Figure 3: Comparison of the imputation methods using metabolomics dataset under MNAR

Metabolomics dataset under MNAR missingness. A non-parametric method called NRMSE-based sum of ranks (SOR) is used to evaluate the imputation error for the skewness of the MNAR distribution (Wei et al., 2018). Fig. 3a indicates that SoftImpute demonstrates a clear advantage in this metric, achieving the lowest values across all missingness levels except at 20% missingness, where SoftForest performs best. As shown in Fig. 3 b–d, the left-censored Half-min imputation performs best for three other metrics, yielding results similar to those reported by Wei et al. (2018).



(a) NRMSE (b) PFC (c) Gower's distance

(d) Predictive R^2 degradation (e) PCA Procrustes distance (f) PLS Procrustes distance

(g) NRMSE (h) PFC (i) Gower's distance

(j) Predictive R^2 degradation (k) PCA Procrustes distance (l) PLS Procrustes distance

Figure 4: Comparison of imputation methods under MCAR (a–f) and MAR (g–l).

Table 2: Runtime comparison on the housing dataset, reported in seconds as mean ± std.

Mechanism	MissForest	SoftForest	NuclearForest
MCAR	5.934 ± 0.187	0.664 ± 0.045	0.659 ± 0.047
MAR	5.973 ± 0.198	0.661 ± 0.039	0.661 ± 0.034

This behavior is expected under the MNAR design, where missing values are generated from low-abundance entries below feature-specific thresholds. Since Half-min replaces missing entries with a small value, its inductive bias matches the left-censored missingness mechanism.

Housing dataset under MCAR and MAR missingness. We further benchmark data imputation on the publicly available Kaggle housing dataset (Harish Kumar Data Lab, 2024) under MCAR and MAR mechanisms, having six numerical and seven categorical features. In terms of estimation error, NRMSE after z-score transformation is used and for the binary/one hot columns, we calculate the proportion of falsely classified entries (PFC) (Stekhoven & Bühlmann, 2012). Fig. 4 a–b show that NuclearForest performs best for missing rates between 10% and 60% in terms of NRMSE, and at 10%, 30%, and 50%–60% in terms of PFC under the MCAR mechanism, while SoftForest achieves the lowest PFC values at 20% and 40% missingness. Under the MAR mechanism, the results in Fig. 4 g–h indicate that NuclearForest obtains the lowest NRMSE at 10%–60% missingness, while SoftForest performs best at 70% missingness. In terms of PFC, NuclearForest achieves the best results at 10% and 50%–60% missingness, whereas SoftForest obtains the lowest values at

20%–40% missingness. For distributional evaluation, Gower's distance measures the dissimilarity for the mixed data types (El Badisy et al., 2024; Gower, 1971). Fig. 4c shows that, under the MCAR mechanism, NuclearForest obtains the lowest values at 10% and 30%–60% missingness, while SoftForest achieves the lowest value at 20% missingness. Under the MAR mechanism, Fig. 4i shows that NuclearForest performs best at 10%, 30%, and 50%–60% missingness, whereas SoftForest obtains the lowest values at 20%, 40% and 70%–90% missingness. Downstream price prediction evaluates the utility of the imputed data for regression tasks. This follows a prediction-oriented evaluation perspective in supervised learning with missing values (Josse et al., 2024; Le Morvan et al., 2021; Morvan & Varoquaux, 2024), where missing-value handling is assessed by its impact on downstream predictive performance rather than only by reconstruction error. We therefore compute Downstream Predictive R^2 Degradation by training a ridge regression model to predict housing prices from all remaining features. Overall, these results suggest that the most effective method for preserving downstream predictive performance depends on the severity of missingness. Hybrid methods are particularly competitive at low to moderate missingness levels, whereas MissForest and SoftImpute become more favorable under higher missingness. To maintain consistency across datasets, we additionally evaluate imputation performance using global and group-level structure preservation metrics, including PCA and PLS Procrustes distance, and Pearson correlation of log-transformed p-values (Wei et al., 2018). These metrics are not commonly used for evaluating regression tasks, where predictive performance (e.g., regression error) is typically the primary criterion, originating instead from omics-based studies where preserving multivariate structure and statistical inference is essential. Specifically, we divide the dataset into three groups of price tercile labels to assess the imputation methods. Fig. 4 e–f and Fig. 4 k–l show the results of PCA/PLS Procrustes distance and the experimental results with respect to the log-transformed p-values is given in Appendix D Fig. 7. Under the MCAR mechanism, NuclearForest obtains the lowest PCA and PLS Procrustes distance at 10%–70% missingness. At 80%–90% missingness, SVT achieves the best result for both PCA and PLS Procrustes distances. When missingness follows the MAR mechanism, SoftForest obtains the best results across most missingness levels, while NuclearForest performs best at 30% missingness. The Pearson log-p correlation results in Fig. 7a–b show that, under MCAR, median imputation achieves the best performance across most missingness rates, with the exception of the 10%–20% setting. For evaluation under MAR, SoftForest performs best at 50%–70% missing rates.

4 Conclusion

We proposed NuclearForest and SoftForest, two hybrid imputation methods that combine low-rank initialization with a single random-forest refinement pass. NuclearForest is built on an improved SVT variant with warm starting and adaptive step sizes. For this adaptive SVT component, we prove step-size bounds and a convergence result for the corresponding zero-initialized iteration, and our ablation studies show that both the warm start and adaptive update contribute to improved imputation quality under the MCAR/MAR missingness setting considered in our experiments. Across the metabolomics and housing benchmarks, the proposed hybrid methods offer a favorable accuracy–efficiency trade-off. On the former, NuclearForest remains competitive with MissForest across missingness levels and is particularly strong at low missing rates, while MissForest achieves the best performance in several medium- and high-missingness settings. Importantly, the proposed hybrids are substantially faster: in the MCAR/MAR metabolomics experiment, SoftForest and NuclearForest achieve approximately $9.52\times$ and $5.81\times$ speedups over MissForest, respectively. A limitation is that these gains do not fully extend to the left-censored MNAR setting, where Half-min is well aligned with this mechanism. This suggests that the proposed hybrids are most effective when the missingness pattern preserves enough global correlation structure for informative low-rank initialization.

The advantage of the hybrid strategy is more pronounced on the smaller housing dataset, where NuclearForest and SoftForest perform strongly across many low- to moderate-missingness settings, with especially strong performance on PCA and PLS Procrustes metrics, indicating better preservation of low-dimensional and group-level structure. These findings suggest that low-rank completion provides an effective structural warm start, while a single random forest refinement step can capture nonlinear feature relationships with lower computational cost than fully iterative random forest imputation. Overall, this supports the proposed hybrid strategy as a practical and computationally efficient approach for preserving structural information during tabular data imputation. Future work will further characterize the full hybrid procedure theoretically and explore low-rank warm starts for other nonlinear refiners, including neural-network-based imputers.

AI USE STATEMENT

In this work, we used generative AI tools to improve the grammar and readability of the manuscript, identify potentially relevant literature, and assist with coding. We did not use generative AI tools for any other tasks requiring disclosure, and the remaining required-disclosure tasks were not applicable to this work. All AI-assisted content was manually reviewed and verified. We take responsibility for the final content of this work.

ETHICS STATEMENT

This work develops and evaluates methods for missing-data imputation using publicly available datasets and does not involve new data collection, crowdsourcing, or human-subject experiments. However, imputed values are estimates rather than observed measurements, and inaccurate imputations, particularly when the assumed missingness mechanism differs from that of the data, may distort downstream statistical analyses or predictive models. This work was supported by a publicly funded research grant providing doctoral student funding. The funder had no role in the study design, data collection, analysis, or writing of the manuscript.

REPRODUCIBILITY STATEMENT

The main paper and Appendix provide the methodological and experimental details required to reproduce the reported results. The assumptions and complete proof for the theoretical result are also provided in the Appendix. The datasets used in the experiments are publicly available. Code for reproducing the benchmark will be released upon acceptance.

REFERENCES

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Jian-Feng Cai, Emmanuel J Candès, and Zuowei Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on optimization*, 20(4):1956–1982, 2010.

Emmanuel Candes and Benjamin Recht. Exact matrix completion via convex optimization. *Communications of the ACM*, 55(6):111–119, 2012.

Damek Davis, Dmitriy Drusvyatskiy, and Liwei Jiang. Gradient descent with adaptive stepsize converges (nearly) linearly under fourth-order growth. *Mathematical Programming*, pp. 1–66, 2025.

Imad El Badisy, Nathalie Graffeo, Mohamed Khalis, and Roch Giorgi. Multi-metric comparison of machine learning imputation methods with application to breast cancer survival. *BMC Medical Research Methodology*, 24(1):191, 2024.

John C Gower. A general coefficient of similarity and some of its properties. *Biometrics*, pp. 857–871, 1971.

Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? *Advances in neural information processing systems*, 35: 507–520, 2022.

Harish Kumar Data Lab. Housing Price Prediction Dataset. Kaggle, 2024. URL `https://www.kaggle.com/datasets/harishkumardatalab/housing-price-prediction`.

Haileab Hilafu, Sandra E Safo, and Lillian Haine. Sparse reduced-rank regression for integrating omics data. *BMC bioinformatics*, 21(1):283, 2020.

Julie Josse, Jacob M Chen, Nicolas Prost, Gaël Varoquaux, and Erwan Scornet. On the consistency of supervised learning with missing values. *Statistical Papers*, 65(9):5447–5479, 2024.

Marine Le Morvan, Julie Josse, Erwan Scornet, and Gaël Varoquaux. What's a good imputation to predict with missing values? *Advances in Neural Information Processing Systems*, 34:11530–11540, 2021.

Yunwen Lei and Ding-Xuan Zhou. Analysis of singular value thresholding algorithm for matrix completion. *Journal of Fourier Analysis and Applications*, 25(6):2957–2972, 2019.

Rahul Mazumder, Trevor Hastie, and Robert Tibshirani. Spectral regularization algorithms for learning large incomplete matrices. *The Journal of Machine Learning Research*, 11:2287–2322, 2010.

Marine Le Morvan and Gaël Varoquaux. Imputation for prediction: beware of diminishing returns. *arXiv preprint arXiv:2407.19804*, 2024.

Philipp Probst, Marvin N Wright, and Anne-Laure Boulesteix. Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews: data mining and knowledge discovery*, 9(3): e1301, 2019.

Benjamin Recht, Maryam Fazel, and Pablo A Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

Donald B Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976.

Rianne Margaretha Schouten, Peter Lugtig, and Gerko Vink. Generating missing values for simulation purposes: a multivariate amputation procedure. *Journal of Statistical Computation and Simulation*, 88(15):2909–2930, 2018.

Sumit Sethi and Elisa Brietzke. Omics-based biomarkers: application of metabolomics in neuropsychiatric disorders. *International Journal of Neuropsychopharmacology*, 19(3):pyv096, 2016.

Daniel J Stekhoven and Peter Bühlmann. Missforest—non-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1):112–118, 2012.

Fei Tang and Hemant Ishwaran. Random forest missing data algorithms. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 10(6):363–377, 2017.

Stef Van Buuren and Karin Groothuis-Oudshoorn. mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, 45:1–67, 2011.

Runmin Wei, Jingye Wang, Mingming Su, Erik Jia, Shaoqiu Chen, Tianlu Chen, and Yan Ni. Missing value imputation approach for mass spectrometry-based metabolomics data. *Scientific reports*, 8 (1):663, 2018.

Nematollah Zarmehi and Farokh Marvasti. Adaptive singular value thresholding. In *2017 International Conference on Sampling Theory and Applications (SampTA)*, pp. 442–445. IEEE, 2017.

A Adaptive SVT variant and step-size analysis

This section compares the proposed adaptive SVT variant with the standard fixed-step SVT and provides the proof of the step-size boundedness result.

Adaptive Step Size vs. Fixed Step Size The original paper by Cai et al. (2010) uses a fixed step size $\delta = 1.2/p$, where p is the sampling rate. Moreover, they prove convergence for a fixed step size satisfying $0 < \delta < 2$. In this paper, our SVT uses an adaptive step size that increases when the error is decreasing (`step * 1.05`) and shrinks when the error increases (`step * 0.9`).

In a convergence analysis paper by Lei & Zhou (2019), they prove that a necessary and sufficient condition for convergence with respect to the Bregman distance, specifically the step size sequence δ_k, must satisfy $\sum_{k=1}^{\infty} \delta_k = \infty$ and the corresponding convergence proof is provided by (Lei & Zhou, 2019, Proposition 5). The necessity is derived using a lower bound on the decay of the Bregman distance. This provides theoretical justification for employing an adaptive step size, as long as the step sizes do not decay too rapidly (bounded away from zero), ensuring divergence of their cumulative sum.

Recent paper in optimization theory suggests that adaptive step size strategies can accelerate first-order methods beyond what is achievable with constant step sizes. In particular, Davis et al. (2025)

demonstrate that gradient descent with adaptive step sizes achieves nearly linear convergence under weaker growth conditions where constant step size methods exhibit only sublinear rates.

Although these results are established for smooth unconstrained optimization (nuclear norm is not smooth), they provide important insight into the role of step size selection in gradient-based algorithms. Since the SVT algorithm can be interpreted as a gradient descent method applied to the dual problem, this perspective motivates the use of adaptive step size strategies within SVT. Moreover, it can improve convergence behavior by applying larger effective updates in the case of slow progress.

Warm-start initialization (column-mean imputation) vs. cold-start (zero matrix) Standard SVT updates the matrix starting from zero. Motivated by the warm-start strategy used in SoftImpute (Mazumder et al., 2010), we initialize the missing entries with column means before applying the low-rank completion step. This deterministic initialization provides a simple and stable starting point for the iterative updates.

Adaptive step size convergence under finitely many contraction steps

Linear equality constraints (Cai et al., 2010). Set the objective function

$$f_\tau(\mathbf{X}) = \tau\|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X}\|_F^2$$

for some fixed $\tau > 0$, and consider the following optimization problem:

$$\begin{aligned} &\text{minimize} \quad f_\tau(\mathbf{X}) \\ &\text{subject to} \quad \mathcal{A}(\mathbf{X}) = \mathbf{b}, \end{aligned} \tag{3.1}$$

where $\mathcal{A}$ is a linear transformation mapping $n_1 \times n_2$ matrices into $\mathbb{R}^m$ and $\mathcal{A}^*$ denotes its adjoint. Then the Lagrangian for this problem is of the form

$$\mathcal{L}(\mathbf{X}, \mathbf{y}) = f_\tau(\mathbf{X}) + \langle \mathbf{y}, \mathbf{b} - \mathcal{A}(\mathbf{X})\rangle, \tag{3.2}$$

where $\mathbf{X} \in \mathbb{R}^{n_1 \times n_2}$ and $\mathbf{y} \in \mathbb{R}^m$, and starting with $\mathbf{y}^0 = 0$, Uzawa's iteration is given by

$$\begin{cases} \mathbf{X}^k = \mathcal{S}_\tau\big(\mathcal{A}^*(\mathbf{y}^{k-1})\big), \\ \mathbf{y}^k = \mathbf{y}^{k-1} + \delta_k\big(\mathbf{b} - \mathcal{A}(\mathbf{X}^k)\big). \end{cases} \tag{3.3}$$

Shrinkage iterations (Cai et al., 2010). For the matrix completion problem, let $\mathcal{A} = \mathcal{P}_\Omega$, $b = \mathcal{P}_\Omega(M)$, and note that $\|\mathcal{P}_\Omega\| = 1$. Then (3.1) reduces to

$$\begin{aligned} &\text{minimize} \quad \tau\|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X}\|_F^2 \\ &\text{subject to} \quad \mathcal{P}_\Omega(\mathbf{X}) = \mathcal{P}_\Omega(\mathbf{M}). \end{aligned} \tag{2.8}$$

and the corresponding iteration (3.3) becomes

$$\begin{cases} \mathbf{X}^k = \mathcal{S}_\tau\big(\mathbf{Y}^{k-1}\big), \\ \mathbf{Y}^k = \mathbf{Y}^{k-1} + \delta_k\mathcal{P}_\Omega\big(\mathbf{M} - \mathbf{X}^k\big), \end{cases} \tag{2.7}$$

where we write $\mathbf{Y}^k$ in place of $\mathbf{y}^k$.

Theorem 1 (SVT convergence (Cai et al., 2010))**.** *Suppose the step sizes obey*

$$0 < \inf_k \delta_k \le \sup_k \delta_k < 2/\|\mathcal{A}\|^2$$

Then the sequence $\{\mathbf{X}^k\}$ *obtained via* (3.3) *converges to the unique solution to* (3.1)*. In particular, the sequence* $\{\mathbf{X}^k\}$ *obtained via* (2.7) *converges to the unique solution of* (2.8) *provided that*

$$0 < \inf_k \delta_k \le \sup_k \delta_k < 2$$

Theorem 2. *(Lei & Zhou, 2019) Let $\{(X^k, Y^k)\}_{k\in\mathbb{N}}$ be produced by (3.3) and $\mathbf{b}_0 \neq 0$. Here, $\mathbf{b}_0$ denotes the orthogonal projection of $\mathbf{b}$ onto the range of $\mathcal{A}$. The term $D_\Psi^{Y^T}(X^\star, X^T)$ denotes the Bregman distance associated with Ψ, evaluated at X^T with subgradient Y^T. In general, for $\widetilde{Y} \in \partial\Psi(\widetilde{X})$, $D_\Psi^{\widetilde{Y}}(X, \widetilde{X}) = \Psi(X) - \Psi(\widetilde{X}) - \langle X - \widetilde{X}, \widetilde{Y}\rangle$.*

Then the following statements hold.

1. *If $\sup_k \delta_k < \frac{1}{2\|\mathcal{A}\|^2}$, then*

$$\lim_{T\to\infty} D_\Psi^{Y^T}(X^\star, X^T) = 0 \quad \textit{if and only if} \quad \sum_{k=1}^{\infty} \delta_k = \infty.$$

2. *If $\sup_k \delta_k < \frac{2}{\|\mathcal{A}\|^2}$, then*

$$\|X^{T+1} - X^\star\|_F^2 \le \tilde{C}\left[\sum_{k=1}^{T} \delta_k\right]^{-1}, \quad \forall T \in \mathbb{N},$$

where $\tilde{C}$ is a constant independent of T.

Proposition 1 (Adaptive step sizes bounds under condition)**.** *Let $p := |\Omega|/(n_1 n_2)$ denote the observed-entry ratio. Assume $0 < p < 1$, let $\delta_0 = 1.2\,p$ and set $\delta_1 = \delta_0$. Let*

$$e_k := \frac{\|\mathcal{P}_\Omega(X^{k+1} - M)\|_F}{\|\mathcal{P}_\Omega(M)\|_F}$$

denote the relative observed-entry reconstruction error, as in Algorithm 2. For $k \ge 1$, let the step sizes be generated by

$$\delta_{k+1} = \begin{cases} 0.9\,\delta_k, & \textit{if } e_k > e_{k-1}, \\ \min\big(1.05\,\delta_k,\, 2p\big), & \textit{if } e_k \le e_{k-1}. \end{cases}$$

Assume further that the total number of contraction steps is finite, namely, there exists $B \in \mathbb{N}_0$ such that at most B indices k satisfy $e_k > e_{k-1}$. Then the following statements hold:

(a) *For all $k \ge 0$,*

$$0.9^B\,\delta_0 \;\le\; \delta_k \;\le\; 2p \;<\; 2. \tag{5}$$

Hence

$$0 < \inf_k \delta_k \le \sup_k \delta_k < 2.$$

(b) *The step sizes have divergent sum:*

$$\sum_{k=0}^{\infty} \delta_k = \infty. \tag{6}$$

Proof. The proof is in three parts.

Part 1: positivity and upper bound. We first show by induction that

$$0 < \delta_k \le 2p \qquad \forall k \ge 0.$$

For $k = 0$, this is immediate from $\delta_0 = 1.2\,p$ and $0 < p < 1$, since

$$0 < \delta_0 = 1.2\,p \le 2p.$$

For $k = 1$, this also holds because $\delta_1 = \delta_0$.

Now let $k \ge 1$ and assume $0 < \delta_k \le 2p$. There are two cases. If $e_k > e_{k-1}$, then

$$\delta_{k+1} = 0.9\,\delta_k,$$

so $0 < \delta_{k+1} < 2p$.

If $e_k \le e_{k-1}$, then

$$\delta_{k+1} = \min\big(1.05\,\delta_k,\, 2p\big),$$

which again implies $0 < \delta_{k+1} \le 2p$.

Thus, by induction, $0 < \delta_k \le 2p$ for all $k \ge 0$.

Part 2: uniform positive lower bound under finitely many contraction steps. For $k \ge 1$, define

$$N_k := \#\{j \in \{1, \dots, k-1\} : e_j > e_{j-1}\},$$

with $N_0 = N_1 = 0$. Thus N_k counts the number of contraction steps used in generating δ_k from the initial step size. By assumption,

$$N_k \le B \qquad \forall k \ge 0.$$

An expansion step never decreases the step size, while a contraction step multiplies the step size by exactly 0.9. Therefore, after generating δ_k, the smallest possible value of δ_k is obtained by applying the factor 0.9 at each of the N_k contraction steps and no decrease at the remaining steps. Hence

$$\delta_k \ge 0.9^{N_k}\delta_0 \ge 0.9^B\delta_0.$$

Combining this with Part 1 yields

$$0.9^B\delta_0 \le \delta_k \le 2p < 2,$$

which proves equation 5.

Part 3: divergence of the sum. From the lower bound just proved,

$$\delta_k \ge 0.9^B\delta_0 > 0 \qquad \forall k \ge 0.$$

Therefore,

$$\sum_{k=0}^{\infty} \delta_k \;\ge\; \sum_{k=0}^{\infty} 0.9^B\delta_0 = \infty,$$

which proves equation 6. □

Corollary 1 (Convergence of the corresponding zero-initialized adaptive SVT iteration)**.** *Consider the standard SVT iteration (2.7) with zero initialization $Y^0 = 0$, and let the step sizes be generated by the same adaptive rule as in Proposition 1. Define e_k analogously for this zero-initialized iteration, and assume that its total number of contraction steps is finite, which means there exists $B \in \mathbb{N}_0$ such that at most B indices k satisfy $e_k > e_{k-1}$.*

Then, by the same argument as in Proposition 1,

$$0.9^B\delta_0 \le \delta_k \le 2p < 2,$$

and hence

$$0 < \inf_k \delta_k \le \sup_k \delta_k < 2.$$

Therefore, by the SVT convergence theorem of Cai et al. (2010), the corresponding zero-initialized adaptive SVT iterates converge to a unique solution. Moreover, Proposition 9 of Lei & Zhou (2019) provides the following convergence rate bound for the zero-initialized iteration:

$$D_\Psi^{Y^{T+1}}(X^*, X^{T+1}) \le \widetilde{C}\left[\sum_{k=1}^{T}\delta_k\right]^{-1},$$

where $\widetilde{C}$ is independent of T. Since

$$\delta_k \ge 0.9^B\delta_0,$$

we have

$$\sum_{k=1}^{T}\delta_k \ge T\,0.9^B\delta_0,$$

and therefore

$$D_\Psi^{Y^{T+1}}(X^*, X^{T+1}) \le \frac{\widetilde{C}}{T\,0.9^B\delta_0} = O(T^{-1}).$$

B ALGORITHMS

Algorithm 1 Singular Value Thresholding (SVT) (Cai et al., 2010)

Require: Partially observed matrix $M \in \mathbb{R}^{n_1 \times n_2}$, observed index set Ω, threshold τ, step size δ, tolerance ε

Initialize $Y^0 \leftarrow 0$
repeat
  $X^k \leftarrow \mathcal{S}_\tau(Y^{k-1})$
  $Y^k \leftarrow Y^{k-1} + \delta\, \mathcal{P}_\Omega(M - X^k)$
until $\|\mathcal{P}_\Omega(X^k - M)\|_F \,/\, \|\mathcal{P}_\Omega(M)\|_F < \varepsilon$
return $\widehat{M}$ with $\widehat{M}_{ij} = M_{ij}$ for $(i,j) \in \Omega$, and $\widehat{M}_{ij} = X^k_{ij}$ otherwise

Algorithm 2 SVT with warm start and adaptive step size

Require: Partially observed matrix $M \in \mathbb{R}^{n_1 \times n_2}$, observed index set Ω, threshold τ, tolerance ε

Initialize X^0 by column-mean imputation; $Y^0 \leftarrow X^0$
$\delta_0 \leftarrow 1.2\,p$, $\delta_{\max} \leftarrow \min(2p,\, 2)$, where $p = |\Omega|/(n_1 n_2)$, $e_{-1} \leftarrow \infty$
repeat
  $Y^{k+1} \leftarrow Y^k + \delta_k\, \mathcal{P}_\Omega(M - X^k)$
  $X^{k+1} \leftarrow \mathcal{S}_\tau(Y^{k+1})$
  $e_k \leftarrow \|\mathcal{P}_\Omega(X^{k+1} - M)\|_F \,/\, \|\mathcal{P}_\Omega(M)\|_F$
  $\delta_{k+1} \leftarrow 0.9\,\delta_k$ if $e_k > e_{k-1}$, else $\min(1.05\,\delta_k,\, \delta_{\max})$
until $e_k < \varepsilon$
return $\widehat{M}$ with $\widehat{M}_{ij} = M_{ij}$ on Ω, $\widehat{M}_{ij} = X^{k+1}_{ij}$ otherwise

Algorithm 3 SoftImpute (Mazumder et al., 2010)

Require: Partially observed matrix $M \in \mathbb{R}^{n_1 \times n_2}$, observed index set Ω, maximum rank $r_{\max}$, regularization λ, tolerance ε

Initialize $X^0 \leftarrow 0$
repeat
  $W^k \leftarrow \mathcal{P}_\Omega(M) + \mathcal{P}^\perp_\Omega(X^k)$
  $U\Sigma V^\top \leftarrow \mathrm{SVD}(W^k)$, truncate to top $r_{\max}$ components
  $X^{k+1} \leftarrow S_\lambda(W^k)$
until $\|X^{k+1} - X^k\|_F^2 \,/\, \|X^k\|_F^2 < \varepsilon$
return $\widehat{M}$ with $\widehat{M}_{ij} = M_{ij}$ on Ω, $\widehat{M}_{ij} = X^{k+1}_{ij}$ otherwise

C EXPERIMENTAL DETAILS

This section summarizes the hyperparameters and implementation details used in the experiments.

Table 3: Hyperparameter settings used in the additional experiments. SVT_OGpaper follows the fixed step size configuration of Cai et al. (2010). SoftImpute follows the soft-thresholded SVD formulation of Mazumder et al. (2010).

Method	Configuration / hyperparameters
Mean	Column-wise mean imputation.
Median	Column-wise median imputation.
Half-min	Missing entries are replaced by one half of the observed column minimum when the minimum is positive; otherwise a floor value of 10^{-6} is used.
kNN	$k = 10$ nearest neighbors with distance-based weighting.
MissForest	Iterative Random Forest imputation using 100 trees, maximum iterations $= 10$, median initialization, and `n_jobs=-1`.
SVT_OGpaper	The original SVT method with a fixed step size and $\delta = \min(1.2/p, 2)$, zero initialization, skip-ahead dual initialization, maximum iterations $= 1000$, and tolerance $= 10^{-5}$.
SVT	Proposed adaptive SVT variant with $\tau = 5\max(n_1, n_2)$, column-mean warm start, initial step size $\delta_0 = \min(1.2p, 2p)$, step-size cap $\min(2p, 2)$, contraction factor 0.9, expansion factor 1.05, maximum iterations $= 1000$, and tolerance $= 10^{-5}$.
NF_OGpaper	Two-stage method using SVT_OGpaper for initialization, followed by one column-wise Random Forest refinement with 100 trees, and `n_jobs=-1`.
NuclearForest	Two-stage method using the proposed adaptive SVT variant for initialization, followed by one column-wise Random Forest refinement with 100 trees, and `n_jobs=-1`.
SoftImpute	SoftImpute with data-scaled regularization $\lambda = 0.01\lambda_0$, where λ_0 is the largest singular value of the zero-filled matrix; rank cap $\min(n, p)$, maximum iterations $= 1000$, and convergence threshold $= 10^{-5}$.
SoftForest	Two-stage method using SoftImpute initialization with $\lambda = 0.01\lambda_0$, rank cap $\min(n, p)$, maximum iterations $= 1000$, and threshold $= 10^{-5}$, followed by one column-wise Random Forest refinement with 100 trees, and `n_jobs=-1`.

Table 4: Global experimental settings

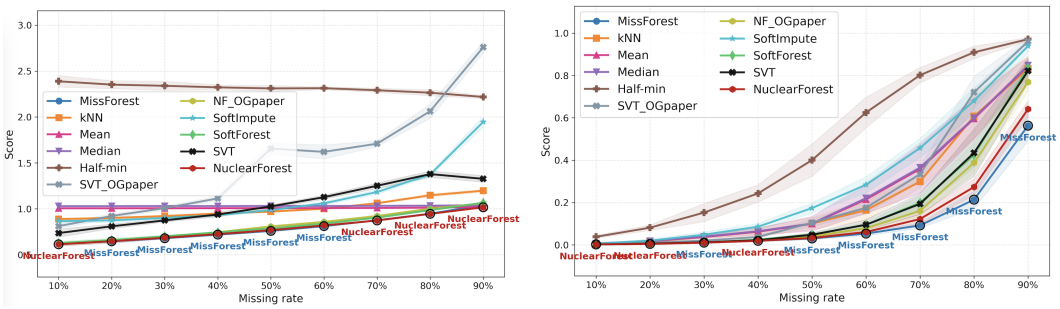
Setting	Value
Number of repeated runs	10
Random seeds	$42, 43, \ldots, 51$, assigned sequentially to the 10 repeated runs
Missingness rates	$10\%, 20\%, 30\%, 40\%, 50\%, 60\%, 70\%, 80\%, 90\%$

Table 5: Total wall-clock runtime.

Experiment	Total runtime (min)	Total runtime (h)
Metabolomics dataset under MCAR/MAR missingness	192.5	3.21
Metabolomics dataset under MNAR missingness	192.0	3.20
Housing dataset under MCAR missingness	12.9	0.22
Housing dataset under MAR missingness	13.0	0.22

D ADDITIONAL EXPERIMENTAL RESULTS

D.1 RESULTS INCLUDING THE ORIGINAL SVT METHOD



(a) NRMSE

(b) PCA Procrustes distance

(c) Pearson log-p correlation

(d) PLS Procrustes distance

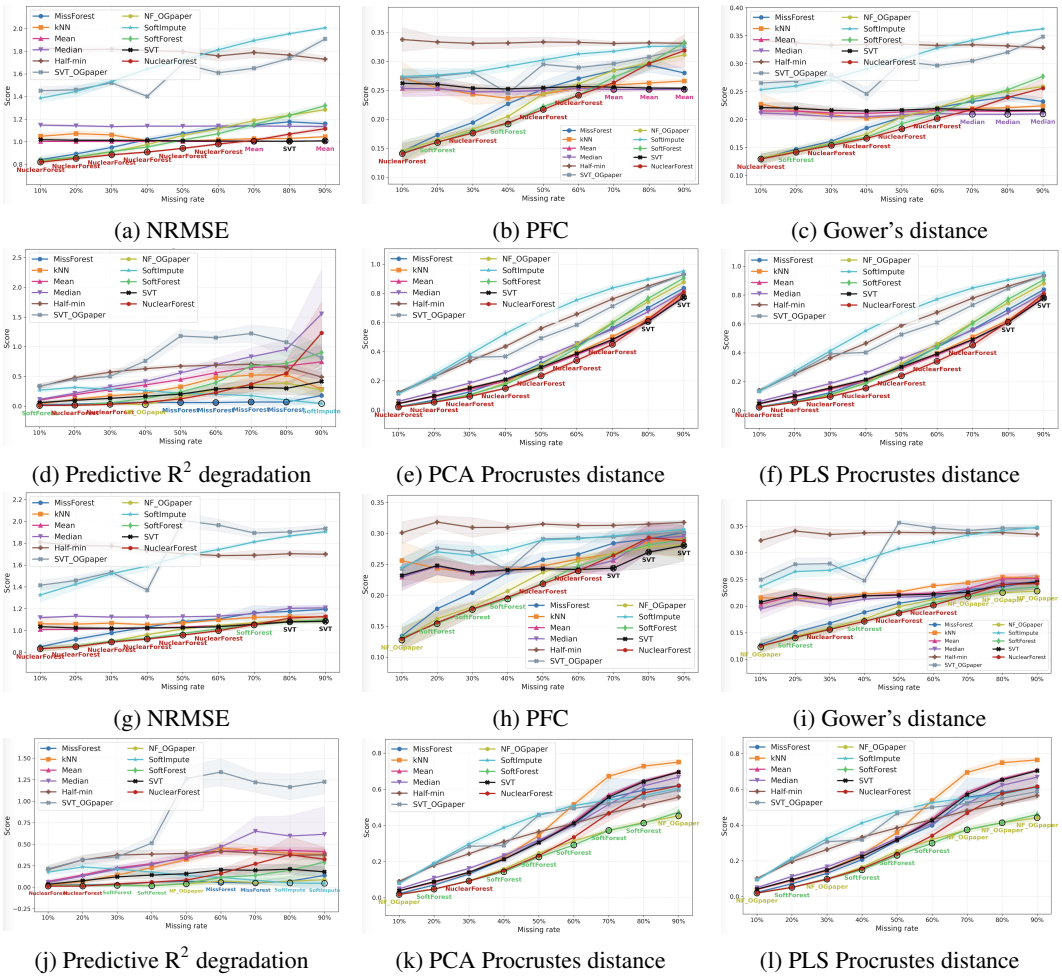
(e) SOR

(f) PCA Procrustes distance

(g) Pearson log-p correlation

(h) PLS Procrustes distance

Figure 5: Comparison of SVT_OGpaper and NF_OGpaper with the adapted SVT and NuclearForest methods on the metabolomics dataset under MCAR/MAR (subfigures a–d) and MNAR (subfigures e–h).



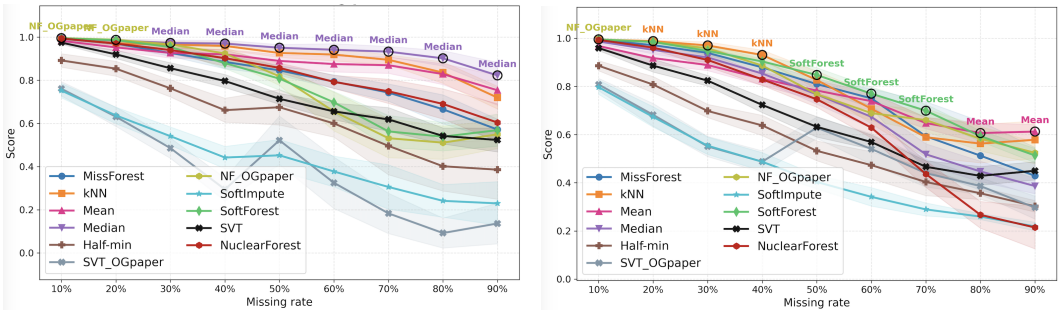
(a) NRMSE (b) PFC (c) Gower's distance

(d) Predictive R^2 degradation (e) PCA Procrustes distance (f) PLS Procrustes distance

(g) NRMSE (h) PFC (i) Gower's distance

(j) Predictive R^2 degradation (k) PCA Procrustes distance (l) PLS Procrustes distance

Figure 6: Comparison of SVT_OGpaper and NF_OGpaper with the adapted SVT and NuclearForest methods on the housing dataset under MCAR (subfigures a–f) and MAR (subfigures g–l).



(a) Pearson log-p correlation (b) Pearson log-p correlation

Figure 7: Comparison of imputation methods on the housing dataset under MCAR (subfigure a) and MAR (subfigure b).

D.2 Partial SVD versus full SVD in SVT

D.2 compares partial and full SVD implementations inside SVT. These results assess whether the choice of SVD solver materially affects imputation quality in the small-to-moderate dense

matrices considered in this work. SVT_partial uses a partial SVD implementation, whereas SVT and SVT_OGpaper use full SVD.

Table 6: NRMSE

Rate	SVT_OGpaper	SVT_partial	SVT
10%	0.806 ± 0.011	0.804 ± 0.010	**0.740 ± 0.050**
20%	0.920 ± 0.029	0.917 ± 0.028	**0.814 ± 0.032**
30%	0.999 ± 0.012	0.995 ± 0.014	**0.874 ± 0.015**
40%	1.104 ± 0.027	1.096 ± 0.028	**0.938 ± 0.028**
50%	1.646 ± 0.067	1.646 ± 0.067	**1.023 ± 0.023**
60%	1.579 ± 0.014	1.579 ± 0.014	**1.140 ± 0.013**
70%	1.717 ± 0.035	1.717 ± 0.035	**1.252 ± 0.035**
80%	2.028 ± 0.066	2.028 ± 0.066	**1.360 ± 0.033**
90%	2.753 ± 0.069	2.753 ± 0.069	**1.332 ± 0.028**

Table 7: PCA Procrustes distance

Rate	SVT_OGpaper	SVT_partial	SVT
10%	0.0028 ± 0.0006	0.0028 ± 0.0007	**0.0019 ± 0.0007**
20%	0.0095 ± 0.0026	0.0096 ± 0.0027	**0.0059 ± 0.0011**
30%	0.0216 ± 0.0029	0.0215 ± 0.0030	**0.0119 ± 0.0014**
40%	0.0393 ± 0.0045	0.0385 ± 0.0042	**0.0226 ± 0.0048**
50%	0.1020 ± 0.0178	0.1020 ± 0.0178	**0.0488 ± 0.0123**
60%	0.1600 ± 0.0309	0.1600 ± 0.0309	**0.0952 ± 0.0312**
70%	0.3231 ± 0.0238	0.3231 ± 0.0238	**0.1715 ± 0.0343**
80%	0.7594 ± 0.0723	0.7594 ± 0.0723	**0.4641 ± 0.1562**
90%	0.9626 ± 0.0199	0.9626 ± 0.0199	**0.8080 ± 0.0604**

Table 8: Pearson correlation of $-\log_{10} p$

Rate	SVT_OGpaper	SVT_partial	SVT
10%	0.982 ± 0.004	0.982 ± 0.004	**0.995 ± 0.003**
20%	0.956 ± 0.012	0.956 ± 0.012	**0.991 ± 0.002**
30%	0.916 ± 0.020	0.915 ± 0.020	**0.980 ± 0.003**
40%	0.858 ± 0.015	0.858 ± 0.015	**0.960 ± 0.006**
50%	0.770 ± 0.025	0.770 ± 0.025	**0.938 ± 0.009**
60%	0.695 ± 0.039	0.695 ± 0.039	**0.889 ± 0.018**
70%	0.606 ± 0.042	0.606 ± 0.042	**0.838 ± 0.030**
80%	0.478 ± 0.111	0.478 ± 0.111	**0.757 ± 0.036**
90%	0.300 ± 0.044	0.300 ± 0.044	**0.631 ± 0.044**

Table 9: PLS Procrustes distance

Rate	SVT_OGpaper	SVT_partial	SVT
10%	0.0045 ± 0.0004	0.0045 ± 0.0005	**0.0041 ± 0.0005**
20%	0.0124 ± 0.0004	0.0125 ± 0.0005	**0.0094 ± 0.0012**
30%	0.0272 ± 0.0017	0.0271 ± 0.0015	**0.0203 ± 0.0018**
40%	0.0562 ± 0.0051	0.0560 ± 0.0049	**0.0403 ± 0.0022**
50%	0.1630 ± 0.0243	0.1630 ± 0.0243	**0.0740 ± 0.0089**
60%	0.2020 ± 0.0048	0.2020 ± 0.0048	**0.1272 ± 0.0116**
70%	0.3742 ± 0.0298	0.3742 ± 0.0298	**0.2281 ± 0.0287**
80%	0.6624 ± 0.0201	0.6624 ± 0.0201	**0.4307 ± 0.0637**
90%	0.8905 ± 0.0149	0.8905 ± 0.0149	**0.6956 ± 0.0399**

Table 10: Runtime comparison between partial SVD and full SVD in SVT.

Method	Mean runtime (s)	Std runtime (s)	Min runtime (s)	Max runtime (s)
SVT_OGpaper	5.465 ± 1.817	1.817	3.933	14.748
SVT_partial	15.810 ± 5.069	5.069	8.206	29.109
SVT	**5.292 ± 1.267**	**1.267**	**3.896**	**9.304**

D.3 Ablation Study

The ablation isolates the effects of the two modifications to the SVT: column-mean warm starting and the adaptive step-size rule. The plots compare these variants across missingness levels.

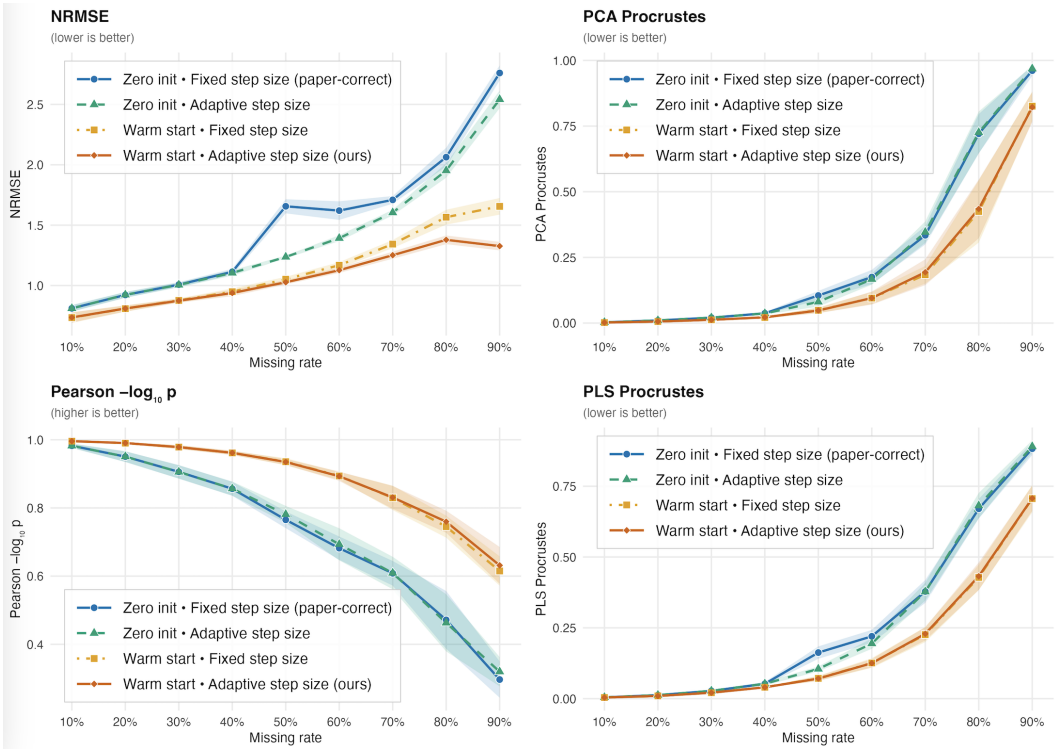


Figure 8: Ablation study on the metabolomics dataset under MCAR/MAR missingness.

Table 11: NRMSE across missing rates

Rate	Zero · fixed	Zero · adaptive	Warm · fixed	Warm · adaptive
10%	0.811 ± 0.027	0.811 ± 0.027	**0.734 ± 0.042**	0.736 ± 0.043
20%	0.922 ± 0.024	0.923 ± 0.024	**0.809 ± 0.026**	0.810 ± 0.025
30%	1.008 ± 0.020	1.005 ± 0.020	0.877 ± 0.017	**0.874 ± 0.016**
40%	1.113 ± 0.025	1.104 ± 0.023	0.949 ± 0.027	**0.938 ± 0.026**
50%	1.656 ± 0.058	1.237 ± 0.011	1.052 ± 0.020	**1.027 ± 0.017**
60%	1.620 ± 0.077	1.392 ± 0.024	1.168 ± 0.025	**1.127 ± 0.022**
70%	1.709 ± 0.032	1.604 ± 0.023	1.343 ± 0.033	**1.251 ± 0.029**
80%	2.063 ± 0.078	1.952 ± 0.059	1.566 ± 0.061	**1.378 ± 0.036**
90%	2.760 ± 0.063	2.542 ± 0.073	1.656 ± 0.069	**1.327 ± 0.035**

Table 12: PCA Procrustes distance across missing rates

Rate	Zero · fixed	Zero · adaptive	Warm · fixed	Warm · adaptive
10%	0.003 ± 0.001	0.003 ± 0.001	**0.002 ± 0.001**	0.002 ± 0.001
20%	0.010 ± 0.002	0.010 ± 0.002	0.006 ± 0.001	**0.006 ± 0.001**
30%	0.021 ± 0.004	0.021 ± 0.004	0.012 ± 0.003	**0.012 ± 0.003**
40%	0.037 ± 0.005	0.037 ± 0.004	**0.022 ± 0.003**	0.022 ± 0.004
50%	0.105 ± 0.017	0.081 ± 0.018	0.049 ± 0.010	**0.048 ± 0.010**
60%	0.175 ± 0.027	0.167 ± 0.021	**0.095 ± 0.023**	0.095 ± 0.025
70%	0.334 ± 0.035	0.346 ± 0.041	**0.184 ± 0.034**	0.194 ± 0.049
80%	0.721 ± 0.075	0.726 ± 0.080	**0.425 ± 0.118**	0.435 ± 0.112
90%	0.961 ± 0.020	0.968 ± 0.011	0.826 ± 0.056	**0.822 ± 0.057**

Table 13: PLS Procrustes distance across missing rates

Rate	Zero · fixed	Zero · adaptive	Warm · fixed	Warm · adaptive
10%	0.005 ± 0.000	0.005 ± 0.000	**0.004 ± 0.000**	0.004 ± 0.000
20%	0.012 ± 0.001	0.012 ± 0.001	0.009 ± 0.001	**0.009 ± 0.001**
30%	0.027 ± 0.003	0.027 ± 0.003	0.021 ± 0.003	**0.021 ± 0.003**
40%	0.052 ± 0.006	0.052 ± 0.006	0.040 ± 0.002	**0.040 ± 0.002**
50%	0.163 ± 0.022	0.105 ± 0.009	0.072 ± 0.009	**0.071 ± 0.008**
60%	0.220 ± 0.023	0.194 ± 0.018	0.126 ± 0.017	**0.125 ± 0.015**
70%	0.378 ± 0.040	0.378 ± 0.032	**0.226 ± 0.026**	0.229 ± 0.024
80%	0.670 ± 0.034	0.681 ± 0.045	**0.428 ± 0.045**	0.432 ± 0.049
90%	0.883 ± 0.022	0.890 ± 0.022	**0.706 ± 0.048**	0.707 ± 0.044

Table 14: Pearson log-p correlation across missing rates

Rate	Zero · fixed	Zero · adaptive	Warm · fixed	Warm · adaptive
10%	0.982 ± 0.006	0.982 ± 0.006	0.996 ± 0.002	**0.996 ± 0.002**
20%	0.951 ± 0.016	0.951 ± 0.016	0.990 ± 0.003	**0.990 ± 0.002**
30%	0.906 ± 0.020	0.906 ± 0.020	**0.979 ± 0.006**	0.978 ± 0.005
40%	0.856 ± 0.020	0.857 ± 0.021	0.962 ± 0.006	**0.962 ± 0.006**
50%	0.765 ± 0.024	0.781 ± 0.025	**0.936 ± 0.010**	0.935 ± 0.009
60%	0.682 ± 0.034	0.693 ± 0.047	**0.894 ± 0.012**	0.893 ± 0.014
70%	0.608 ± 0.036	0.609 ± 0.048	0.830 ± 0.036	**0.831 ± 0.034**
80%	0.472 ± 0.085	0.463 ± 0.084	0.745 ± 0.032	**0.760 ± 0.032**
90%	0.297 ± 0.053	0.320 ± 0.043	0.615 ± 0.043	**0.632 ± 0.054**

Table 15: Runtime comparison of SVT ablation variants. Runtime is reported in seconds as mean ± standard deviation

Method	Mean ± Std.	Min.	Max.
Zero init + fixed step size	6.440 ± 1.616	4.385	12.716
Zero init + adaptive step size	**6.147 ± 1.152**	4.377	11.303
Warm start + fixed step size	6.378 ± 1.355	4.421	11.093
Warm start + adaptive step size	6.459 ± 1.669	4.229	12.460

D.4 Runtime Decomposition and Scaling

Table 16: Stage-decomposed runtime on the metabolomics dataset (198×130, MCAR). Times presented are means over three measured runs. Pre denotes Preprocess, SVD denotes one SVD. NF denotes NuclearForest, MF denotes MissForest, and SI denotes SoftImpute.

Missing	Pre. (ms)	SVD (ms)	SVT (s)	RF (s)	NF (s)	MF (s)	SI (s)
20%	0.18 ms	4.09 ms	8.97 s	10.31 s	19.28 s	105.50 s	0.48 s
50%	0.22 ms	10.18 ms	10.28 s	7.75 s	18.03 s	76.64 s	0.80 s
80%	0.19 ms	12.81 ms	8.22 s	5.08 s	13.30 s	50.84 s	0.46 s

Mask generation is part of the evaluation protocol rather than the imputation procedure and is therefore excluded from the reported imputation runtime.

We conduct a scaling study using rank-10 Gaussian matrices, with 5% noise and 30% MCAR masking. Matrix size is reported as number of rows by number of columns. With 130 columns, increasing the number of rows from 1000 to 2000 approximately doubled runtime from 66.0 s to 130.0 s. However, for a matrix with 2000 rows and 1000 columns, runtime reached 6885.7 s, with RF refinement accounting for 87.0% of total runtime.

Table 17: Runtime scaling study on rank-10 Gaussian matrices with 5% noise and 30% MCAR masking.

Matrix size	Entries	SVT stage	RF stage	Total	RF share
(1000×130)	130,000	19.58 s	46.44 s	66.02 s	70.3%
(2000×130)	260,000	25.84 s	104.19 s	130.02 s	80.1%
(1000×500)	500,000	169.73 s	653.89 s	823.62 s	79.4%
(5000×200)	1,000,000	106.67 s	727.89 s	834.56 s	87.2%
(2000×1000)	2,000,000	893.83 s	5991.90 s	6885.72 s	87.0%

E FULL NUMERICAL RESULTS

E.1 MCAR/MAR RESULTS ON THE METABOLOMICS DATASET

Table 18: NRMSE across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.618 ± 0.020	0.888 ± 0.012	1.006 ± 0.001	1.029 ± 0.002	2.389 ± 0.063	0.811 ± 0.027
20%	**0.646 ± 0.021**	0.901 ± 0.009	1.005 ± 0.001	1.029 ± 0.002	2.353 ± 0.043	0.922 ± 0.024
30%	**0.680 ± 0.007**	0.922 ± 0.005	1.006 ± 0.001	1.031 ± 0.003	2.340 ± 0.049	1.008 ± 0.020
40%	**0.719 ± 0.009**	0.949 ± 0.005	1.007 ± 0.001	1.030 ± 0.002	2.323 ± 0.027	1.113 ± 0.025
50%	**0.760 ± 0.008**	0.970 ± 0.008	1.007 ± 0.001	1.031 ± 0.003	2.311 ± 0.030	1.656 ± 0.058
60%	**0.812 ± 0.009**	1.001 ± 0.010	1.008 ± 0.001	1.031 ± 0.002	2.314 ± 0.016	1.620 ± 0.077
70%	0.875 ± 0.006	1.061 ± 0.008	1.010 ± 0.001	1.033 ± 0.002	2.291 ± 0.021	1.709 ± 0.032
80%	0.947 ± 0.007	1.148 ± 0.016	1.013 ± 0.002	1.035 ± 0.004	2.265 ± 0.033	2.063 ± 0.078
90%	1.041 ± 0.015	1.197 ± 0.015	1.020 ± 0.003	1.040 ± 0.006	2.218 ± 0.017	2.760 ± 0.063

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.617 ± 0.019	0.866 ± 0.019	0.628 ± 0.020	0.736 ± 0.043	**0.614 ± 0.020**
20%	0.657 ± 0.019	0.874 ± 0.021	0.659 ± 0.021	0.810 ± 0.025	0.647 ± 0.019
30%	0.697 ± 0.012	0.901 ± 0.015	0.701 ± 0.010	0.874 ± 0.016	0.685 ± 0.010
40%	0.745 ± 0.006	0.932 ± 0.011	0.744 ± 0.007	0.938 ± 0.026	0.724 ± 0.007
50%	0.809 ± 0.006	0.986 ± 0.015	0.787 ± 0.008	1.027 ± 0.017	0.768 ± 0.004
60%	0.858 ± 0.005	1.060 ± 0.018	0.839 ± 0.007	1.127 ± 0.022	0.819 ± 0.007
70%	0.922 ± 0.005	1.183 ± 0.025	0.905 ± 0.007	1.251 ± 0.029	**0.873 ± 0.007**
80%	1.002 ± 0.007	1.374 ± 0.020	0.986 ± 0.010	1.378 ± 0.036	**0.945 ± 0.006**
90%	1.061 ± 0.008	1.947 ± 0.059	1.067 ± 0.010	1.327 ± 0.035	**1.017 ± 0.006**

Table 19: PCA Procrustes distance across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.002 ± 0.001	0.005 ± 0.001	0.007 ± 0.001	0.007 ± 0.001	0.038 ± 0.007	0.003 ± 0.001
20%	0.005 ± 0.001	0.016 ± 0.003	0.018 ± 0.003	0.018 ± 0.004	0.081 ± 0.013	0.010 ± 0.002
30%	**0.010 ± 0.002**	0.037 ± 0.007	0.037 ± 0.006	0.039 ± 0.007	0.153 ± 0.043	0.021 ± 0.004
40%	0.020 ± 0.003	0.062 ± 0.020	0.062 ± 0.016	0.064 ± 0.015	0.243 ± 0.041	0.037 ± 0.005
50%	**0.030 ± 0.004**	0.101 ± 0.028	0.099 ± 0.035	0.101 ± 0.034	0.400 ± 0.077	0.105 ± 0.017
60%	**0.053 ± 0.010**	0.164 ± 0.049	0.215 ± 0.066	0.221 ± 0.073	0.625 ± 0.072	0.175 ± 0.027
70%	**0.092 ± 0.013**	0.299 ± 0.030	0.359 ± 0.115	0.365 ± 0.118	0.802 ± 0.036	0.334 ± 0.035
80%	**0.214 ± 0.041**	0.608 ± 0.104	0.596 ± 0.099	0.600 ± 0.096	0.910 ± 0.029	0.721 ± 0.075
90%	**0.563 ± 0.072**	0.833 ± 0.059	0.845 ± 0.043	0.848 ± 0.037	0.972 ± 0.008	0.961 ± 0.020

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.002 ± 0.001	0.006 ± 0.001	0.002 ± 0.001	0.002 ± 0.001	**0.002 ± 0.001**
20%	0.005 ± 0.001	0.020 ± 0.007	0.005 ± 0.001	0.006 ± 0.001	**0.005 ± 0.001**
30%	0.012 ± 0.003	0.048 ± 0.014	0.011 ± 0.002	0.012 ± 0.003	0.011 ± 0.003
40%	0.023 ± 0.004	0.086 ± 0.015	0.025 ± 0.003	0.022 ± 0.004	**0.019 ± 0.003**
50%	0.040 ± 0.007	0.174 ± 0.025	0.049 ± 0.008	0.048 ± 0.010	0.032 ± 0.005
60%	0.079 ± 0.008	0.284 ± 0.038	0.095 ± 0.011	0.095 ± 0.025	0.061 ± 0.007
70%	0.161 ± 0.026	0.458 ± 0.045	0.199 ± 0.034	0.194 ± 0.049	0.122 ± 0.025
80%	0.388 ± 0.047	0.680 ± 0.061	0.425 ± 0.057	0.435 ± 0.112	0.273 ± 0.039
90%	0.770 ± 0.046	0.939 ± 0.031	0.837 ± 0.040	0.822 ± 0.057	0.641 ± 0.039

Table 20: PLS Procrustes distance across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.003 ± 0.001	0.009 ± 0.001	0.009 ± 0.001	0.010 ± 0.001	0.057 ± 0.008	0.005 ± 0.000
20%	0.008 ± 0.001	0.021 ± 0.001	0.021 ± 0.001	0.022 ± 0.001	0.116 ± 0.015	0.012 ± 0.001
30%	0.016 ± 0.002	0.043 ± 0.005	0.040 ± 0.006	0.042 ± 0.006	0.193 ± 0.034	0.027 ± 0.003
40%	**0.027 ± 0.003**	0.069 ± 0.009	0.065 ± 0.009	0.067 ± 0.009	0.274 ± 0.023	0.052 ± 0.006
50%	**0.043 ± 0.002**	0.101 ± 0.009	0.097 ± 0.011	0.098 ± 0.012	0.371 ± 0.037	0.163 ± 0.022
60%	**0.073 ± 0.010**	0.166 ± 0.023	0.177 ± 0.014	0.181 ± 0.017	0.537 ± 0.036	0.220 ± 0.023
70%	**0.122 ± 0.012**	0.313 ± 0.031	0.291 ± 0.058	0.295 ± 0.060	0.675 ± 0.039	0.378 ± 0.040
80%	**0.235 ± 0.023**	0.525 ± 0.053	0.476 ± 0.061	0.474 ± 0.064	0.797 ± 0.030	0.670 ± 0.034
90%	**0.516 ± 0.057**	0.709 ± 0.042	0.715 ± 0.039	0.718 ± 0.036	0.867 ± 0.045	0.883 ± 0.022

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.003 ± 0.001	0.008 ± 0.001	0.003 ± 0.001	0.004 ± 0.000	**0.003 ± 0.001**
20%	0.008 ± 0.001	0.020 ± 0.002	0.008 ± 0.001	0.009 ± 0.001	**0.007 ± 0.001**
30%	0.016 ± 0.003	0.051 ± 0.007	0.019 ± 0.003	0.021 ± 0.003	**0.015 ± 0.002**
40%	0.029 ± 0.002	0.109 ± 0.016	0.038 ± 0.003	0.040 ± 0.002	0.028 ± 0.002
50%	0.054 ± 0.003	0.230 ± 0.031	0.071 ± 0.006	0.071 ± 0.008	0.048 ± 0.004
60%	0.100 ± 0.010	0.376 ± 0.048	0.132 ± 0.027	0.125 ± 0.015	0.084 ± 0.010
70%	0.192 ± 0.022	0.555 ± 0.029	0.278 ± 0.032	0.229 ± 0.024	0.162 ± 0.017
80%	0.416 ± 0.043	0.696 ± 0.048	0.467 ± 0.054	0.432 ± 0.049	0.336 ± 0.048
90%	0.713 ± 0.031	0.889 ± 0.022	0.732 ± 0.034	0.707 ± 0.044	0.627 ± 0.031

Table 21: Pearson log-p correlation across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.995 ± 0.003	0.993 ± 0.003	0.992 ± 0.003	0.991 ± 0.003	0.892 ± 0.018	0.982 ± 0.006
20%	0.991 ± 0.002	0.985 ± 0.002	0.983 ± 0.003	0.981 ± 0.004	0.814 ± 0.037	0.951 ± 0.016
30%	**0.984 ± 0.004**	0.975 ± 0.004	0.970 ± 0.006	0.966 ± 0.007	0.719 ± 0.047	0.906 ± 0.020
40%	**0.972 ± 0.007**	0.958 ± 0.010	0.957 ± 0.008	0.952 ± 0.008	0.666 ± 0.080	0.856 ± 0.020
50%	**0.956 ± 0.008**	0.942 ± 0.008	0.940 ± 0.010	0.931 ± 0.012	0.582 ± 0.072	0.765 ± 0.024
60%	**0.938 ± 0.010**	0.916 ± 0.013	0.909 ± 0.017	0.900 ± 0.022	0.511 ± 0.088	0.682 ± 0.034
70%	**0.897 ± 0.016**	0.868 ± 0.039	0.886 ± 0.030	0.867 ± 0.034	0.464 ± 0.115	0.608 ± 0.036
80%	**0.838 ± 0.024**	0.720 ± 0.031	0.810 ± 0.029	0.784 ± 0.039	0.307 ± 0.164	0.472 ± 0.085
90%	0.719 ± 0.047	0.479 ± 0.080	0.723 ± 0.046	0.704 ± 0.046	0.311 ± 0.114	0.297 ± 0.053

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.995 ± 0.003	0.991 ± 0.004	0.995 ± 0.004	**0.996 ± 0.002**	0.995 ± 0.004
20%	0.991 ± 0.002	0.976 ± 0.006	0.991 ± 0.002	0.990 ± 0.002	**0.991 ± 0.002**
30%	0.983 ± 0.005	0.954 ± 0.005	0.982 ± 0.005	0.978 ± 0.005	0.983 ± 0.004
40%	0.968 ± 0.006	0.922 ± 0.008	0.967 ± 0.007	0.962 ± 0.006	0.968 ± 0.006
50%	0.955 ± 0.009	0.875 ± 0.007	0.948 ± 0.007	0.935 ± 0.009	0.953 ± 0.006
60%	0.932 ± 0.007	0.807 ± 0.024	0.920 ± 0.014	0.893 ± 0.014	0.931 ± 0.008
70%	0.888 ± 0.018	0.739 ± 0.033	0.876 ± 0.017	0.831 ± 0.034	0.890 ± 0.014
80%	0.824 ± 0.032	0.630 ± 0.048	0.802 ± 0.033	0.760 ± 0.032	0.827 ± 0.032
90%	0.714 ± 0.050	0.458 ± 0.070	0.671 ± 0.077	0.632 ± 0.054	**0.724 ± 0.056**

E.2 MNAR RESULTS ON THE METABOLOMICS DATASET

Table 22: SOR across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	81.200 ± 7.815	111.400 ± 7.575	126.900 ± 5.990	97.600 ± 7.619	83.900 ± 12.991	**49.900 ± 9.231**
20%	158.300 ± 9.031	216.800 ± 7.495	253.000 ± 13.241	182.700 ± 14.244	153.300 ± 18.252	119.900 ± 14.873
30%	245.000 ± 19.067	328.300 ± 12.650	380.200 ± 11.013	271.700 ± 16.207	222.600 ± 27.240	162.300 ± 15.805
40%	327.600 ± 17.494	434.600 ± 12.011	509.400 ± 13.032	369.000 ± 22.784	279.400 ± 19.873	215.300 ± 16.687
50%	395.600 ± 27.293	538.500 ± 13.938	624.700 ± 10.605	441.900 ± 19.451	356.300 ± 29.208	306.200 ± 17.415
60%	480.600 ± 23.114	654.100 ± 23.473	740.500 ± 16.939	518.100 ± 16.462	398.600 ± 23.177	345.900 ± 28.136
70%	561.400 ± 30.837	761.700 ± 20.205	865.600 ± 12.703	591.100 ± 19.140	475.900 ± 41.757	414.200 ± 29.506
80%	591.700 ± 33.450	881.300 ± 19.799	961.300 ± 13.744	634.900 ± 19.564	521.000 ± 38.041	516.900 ± 18.150
90%	641.100 ± 30.113	973.900 ± 44.175	1062.700 ± 24.504	686.400 ± 32.834	566.400 ± 29.737	620.500 ± 72.968
Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest	
10%	55.900 ± 11.396	55.400 ± 10.091	68.400 ± 10.189	65.800 ± 12.717	61.600 ± 6.077	
20%	**100.500 ± 15.636**	131.800 ± 15.083	123.700 ± 11.036	145.800 ± 12.417	130.200 ± 14.582	
30%	**153.500 ± 14.539**	189.800 ± 24.769	195.400 ± 11.806	215.100 ± 16.210	210.100 ± 16.347	
40%	**210.500 ± 17.096**	236.700 ± 13.098	260.400 ± 19.614	300.000 ± 19.218	289.100 ± 14.836	
50%	**254.900 ± 22.358**	288.700 ± 32.619	311.100 ± 28.089	410.700 ± 23.566	361.400 ± 14.191	
60%	**322.300 ± 19.579**	332.800 ± 26.469	405.800 ± 25.209	493.700 ± 18.774	455.600 ± 23.406	
70%	**351.700 ± 33.655**	380.100 ± 20.355	450.900 ± 30.282	605.800 ± 47.756	547.600 ± 25.747	
80%	**414.600 ± 34.878**	416.400 ± 20.576	493.500 ± 15.270	762.500 ± 30.376	669.900 ± 14.685	
90%	454.600 ± 18.476	**450.200 ± 32.362**	529.000 ± 29.462	938.500 ± 33.016	798.700 ± 31.997	

Table 23: PCA Procrustes distance across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.001 ± 0.001	0.004 ± 0.001	0.005 ± 0.002	0.004 ± 0.002	**0.001 ± 0.000**	0.001 ± 0.001
20%	0.003 ± 0.001	0.010 ± 0.005	0.012 ± 0.005	0.009 ± 0.005	**0.002 ± 0.001**	0.003 ± 0.002
30%	0.011 ± 0.003	0.038 ± 0.018	0.037 ± 0.017	0.029 ± 0.012	**0.004 ± 0.001**	0.005 ± 0.002
40%	0.021 ± 0.011	0.056 ± 0.021	0.066 ± 0.035	0.047 ± 0.020	0.006 ± 0.001	**0.006 ± 0.002**
50%	0.032 ± 0.008	0.108 ± 0.036	0.122 ± 0.033	0.088 ± 0.023	**0.009 ± 0.001**	0.010 ± 0.003
60%	0.050 ± 0.013	0.170 ± 0.061	0.191 ± 0.040	0.129 ± 0.027	**0.010 ± 0.002**	0.014 ± 0.003
70%	0.078 ± 0.015	0.278 ± 0.092	0.300 ± 0.062	0.196 ± 0.042	**0.017 ± 0.004**	0.023 ± 0.010
80%	0.101 ± 0.025	0.319 ± 0.085	0.383 ± 0.048	0.257 ± 0.038	**0.018 ± 0.004**	0.030 ± 0.005
90%	0.160 ± 0.029	0.426 ± 0.116	0.489 ± 0.043	0.339 ± 0.038	**0.025 ± 0.003**	0.049 ± 0.017
Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest	
10%	0.001 ± 0.000	0.003 ± 0.003	0.001 ± 0.000	0.002 ± 0.001	0.001 ± 0.000	
20%	0.002 ± 0.001	0.014 ± 0.010	0.003 ± 0.001	0.006 ± 0.003	0.004 ± 0.002	
30%	0.007 ± 0.003	0.038 ± 0.025	0.009 ± 0.003	0.009 ± 0.003	0.010 ± 0.004	
40%	0.011 ± 0.004	0.049 ± 0.019	0.015 ± 0.004	0.019 ± 0.014	0.019 ± 0.009	
50%	0.020 ± 0.006	0.093 ± 0.035	0.027 ± 0.006	0.035 ± 0.007	0.030 ± 0.010	
60%	0.030 ± 0.011	0.115 ± 0.034	0.046 ± 0.017	0.049 ± 0.025	0.051 ± 0.021	
70%	0.041 ± 0.008	0.137 ± 0.033	0.069 ± 0.021	0.087 ± 0.047	0.089 ± 0.033	
80%	0.063 ± 0.018	0.195 ± 0.038	0.100 ± 0.024	0.211 ± 0.050	0.186 ± 0.054	
90%	0.086 ± 0.023	0.231 ± 0.057	0.150 ± 0.043	0.322 ± 0.089	0.297 ± 0.083	

Table 24: PLS Procrustes distance across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.002 ± 0.001	0.005 ± 0.003	0.006 ± 0.003	0.005 ± 0.002	**0.001 ± 0.000**	0.002 ± 0.001
20%	0.006 ± 0.002	0.013 ± 0.004	0.013 ± 0.004	0.009 ± 0.003	**0.002 ± 0.000**	0.004 ± 0.001
30%	0.015 ± 0.005	0.041 ± 0.021	0.040 ± 0.016	0.031 ± 0.012	**0.005 ± 0.001**	0.009 ± 0.004
40%	0.024 ± 0.008	0.061 ± 0.029	0.067 ± 0.025	0.051 ± 0.019	**0.007 ± 0.002**	0.009 ± 0.002
50%	0.039 ± 0.006	0.090 ± 0.028	0.107 ± 0.020	0.080 ± 0.015	**0.010 ± 0.001**	0.018 ± 0.003
60%	0.057 ± 0.015	0.128 ± 0.052	0.155 ± 0.040	0.114 ± 0.029	**0.012 ± 0.003**	0.024 ± 0.005
70%	0.087 ± 0.015	0.209 ± 0.070	0.248 ± 0.041	0.177 ± 0.033	**0.020 ± 0.004**	0.036 ± 0.010
80%	0.107 ± 0.020	0.255 ± 0.059	0.296 ± 0.037	0.213 ± 0.029	**0.021 ± 0.004**	0.050 ± 0.018
90%	0.156 ± 0.020	0.370 ± 0.104	0.384 ± 0.033	0.278 ± 0.027	**0.028 ± 0.003**	0.086 ± 0.037

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.002 ± 0.001	0.003 ± 0.001	0.002 ± 0.001	0.002 ± 0.001	0.002 ± 0.001
20%	0.005 ± 0.001	0.011 ± 0.003	0.005 ± 0.001	0.005 ± 0.001	0.005 ± 0.001
30%	0.011 ± 0.004	0.038 ± 0.014	0.013 ± 0.005	0.012 ± 0.006	0.014 ± 0.006
40%	0.017 ± 0.006	0.059 ± 0.018	0.019 ± 0.006	0.018 ± 0.008	0.022 ± 0.008
50%	0.024 ± 0.004	0.093 ± 0.027	0.030 ± 0.004	0.028 ± 0.008	0.033 ± 0.007
60%	0.040 ± 0.012	0.132 ± 0.056	0.054 ± 0.023	0.048 ± 0.025	0.060 ± 0.027
70%	0.049 ± 0.010	0.176 ± 0.047	0.077 ± 0.017	0.083 ± 0.046	0.092 ± 0.034
80%	0.073 ± 0.016	0.257 ± 0.046	0.108 ± 0.025	0.133 ± 0.044	0.143 ± 0.042
90%	0.093 ± 0.015	0.309 ± 0.057	0.159 ± 0.046	0.270 ± 0.074	0.258 ± 0.065

Table 25: Pearson log-p correlation across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.991 ± 0.007	0.962 ± 0.024	0.942 ± 0.036	0.956 ± 0.027	0.995 ± 0.003	**0.997 ± 0.001**
20%	0.983 ± 0.006	0.948 ± 0.022	0.924 ± 0.030	0.946 ± 0.024	**0.994 ± 0.002**	0.992 ± 0.004
30%	0.964 ± 0.015	0.890 ± 0.048	0.849 ± 0.042	0.888 ± 0.032	**0.989 ± 0.004**	0.988 ± 0.006
40%	0.950 ± 0.012	0.842 ± 0.046	0.787 ± 0.037	0.838 ± 0.031	**0.987 ± 0.003**	0.986 ± 0.003
50%	0.948 ± 0.013	0.847 ± 0.035	0.788 ± 0.028	0.842 ± 0.023	**0.986 ± 0.004**	0.975 ± 0.006
60%	0.921 ± 0.016	0.798 ± 0.079	0.734 ± 0.061	0.803 ± 0.042	**0.982 ± 0.005**	0.967 ± 0.009
70%	0.921 ± 0.013	0.774 ± 0.065	0.704 ± 0.046	0.793 ± 0.031	**0.976 ± 0.004**	0.959 ± 0.007
80%	0.908 ± 0.019	0.708 ± 0.070	0.653 ± 0.042	0.752 ± 0.035	**0.974 ± 0.004**	0.942 ± 0.009
90%	0.905 ± 0.017	0.698 ± 0.120	0.683 ± 0.038	0.777 ± 0.027	**0.972 ± 0.005**	0.924 ± 0.029

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.992 ± 0.006	0.986 ± 0.008	0.992 ± 0.006	0.996 ± 0.002	0.992 ± 0.006
20%	0.988 ± 0.004	0.981 ± 0.007	0.986 ± 0.005	0.985 ± 0.012	0.986 ± 0.005
30%	0.976 ± 0.014	0.965 ± 0.019	0.973 ± 0.014	0.967 ± 0.028	0.966 ± 0.023
40%	0.968 ± 0.009	0.949 ± 0.019	0.962 ± 0.008	0.937 ± 0.038	0.947 ± 0.018
50%	0.968 ± 0.007	0.942 ± 0.017	0.964 ± 0.006	0.933 ± 0.025	0.951 ± 0.015
60%	0.943 ± 0.014	0.931 ± 0.017	0.937 ± 0.020	0.902 ± 0.057	0.911 ± 0.042
70%	0.950 ± 0.011	0.911 ± 0.019	0.941 ± 0.012	0.882 ± 0.054	0.903 ± 0.031
80%	0.935 ± 0.015	0.907 ± 0.022	0.932 ± 0.014	0.823 ± 0.050	0.861 ± 0.035
90%	0.940 ± 0.009	0.874 ± 0.019	0.936 ± 0.009	0.786 ± 0.070	0.837 ± 0.049

E.3 MCAR RESULTS ON THE HOUSING DATASET

Table 26: NRMSE across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.842 ± 0.032	1.049 ± 0.029	1.003 ± 0.001	1.147 ± 0.012	1.829 ± 0.057	1.451 ± 0.037
20%	0.892 ± 0.020	1.072 ± 0.026	1.003 ± 0.001	1.141 ± 0.011	1.814 ± 0.032	1.460 ± 0.038
30%	0.950 ± 0.015	1.060 ± 0.024	1.002 ± 0.001	1.134 ± 0.005	1.818 ± 0.020	1.520 ± 0.021
40%	1.020 ± 0.023	1.011 ± 0.010	1.003 ± 0.001	1.138 ± 0.005	1.807 ± 0.056	1.401 ± 0.016
50%	1.072 ± 0.021	1.006 ± 0.006	1.002 ± 0.001	1.137 ± 0.006	1.799 ± 0.039	1.689 ± 0.036
60%	1.124 ± 0.031	1.018 ± 0.005	1.003 ± 0.001	1.140 ± 0.004	1.759 ± 0.047	1.610 ± 0.031
70%	1.147 ± 0.024	1.027 ± 0.006	**1.004 ± 0.001**	1.140 ± 0.004	1.789 ± 0.045	1.650 ± 0.025
80%	1.176 ± 0.026	1.029 ± 0.005	1.005 ± 0.002	1.136 ± 0.006	1.766 ± 0.040	1.738 ± 0.020
90%	1.159 ± 0.022	1.045 ± 0.010	**1.007 ± 0.003**	1.135 ± 0.008	1.730 ± 0.028	1.909 ± 0.030

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.832 ± 0.020	1.386 ± 0.033	0.835 ± 0.020	1.019 ± 0.017	**0.820 ± 0.017**
20%	0.863 ± 0.014	1.445 ± 0.047	0.866 ± 0.015	1.013 ± 0.015	**0.852 ± 0.012**
30%	0.910 ± 0.012	1.533 ± 0.030	0.914 ± 0.018	1.013 ± 0.008	**0.885 ± 0.011**
40%	0.976 ± 0.023	1.644 ± 0.036	0.954 ± 0.014	1.011 ± 0.006	**0.909 ± 0.007**
50%	1.056 ± 0.016	1.720 ± 0.033	1.015 ± 0.015	1.007 ± 0.006	**0.941 ± 0.007**
60%	1.115 ± 0.019	1.813 ± 0.032	1.071 ± 0.014	1.007 ± 0.005	**0.980 ± 0.010**
70%	1.188 ± 0.024	1.894 ± 0.027	1.152 ± 0.026	1.005 ± 0.005	1.013 ± 0.013
80%	1.235 ± 0.025	1.955 ± 0.017	1.232 ± 0.034	**1.004 ± 0.005**	1.067 ± 0.013
90%	1.283 ± 0.015	2.007 ± 0.010	1.319 ± 0.018	1.008 ± 0.002	1.116 ± 0.025

Table 27: PFC across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.145 ± 0.017	0.271 ± 0.027	0.253 ± 0.014	0.253 ± 0.014	0.338 ± 0.021	0.271 ± 0.016
20%	0.173 ± 0.007	0.254 ± 0.004	0.253 ± 0.012	0.253 ± 0.012	0.333 ± 0.010	0.274 ± 0.010
30%	0.194 ± 0.010	0.243 ± 0.018	0.247 ± 0.007	0.247 ± 0.007	0.331 ± 0.010	0.281 ± 0.006
40%	0.227 ± 0.010	0.236 ± 0.008	0.247 ± 0.006	0.247 ± 0.006	0.332 ± 0.009	0.245 ± 0.018
50%	0.252 ± 0.014	0.243 ± 0.005	0.250 ± 0.006	0.250 ± 0.006	0.333 ± 0.009	0.295 ± 0.013
60%	0.270 ± 0.017	0.254 ± 0.005	0.253 ± 0.004	0.253 ± 0.004	0.333 ± 0.006	0.289 ± 0.007
70%	0.284 ± 0.017	0.259 ± 0.006	**0.251 ± 0.004**	**0.251 ± 0.004**	0.331 ± 0.004	0.296 ± 0.006
80%	0.293 ± 0.012	0.263 ± 0.005	**0.252 ± 0.002**	**0.252 ± 0.002**	0.332 ± 0.003	0.308 ± 0.004
90%	0.280 ± 0.008	0.266 ± 0.003	**0.252 ± 0.001**	**0.252 ± 0.001**	0.331 ± 0.002	0.322 ± 0.007

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.146 ± 0.018	0.274 ± 0.014	0.143 ± 0.017	0.262 ± 0.013	**0.141 ± 0.017**
20%	0.164 ± 0.012	0.276 ± 0.011	**0.160 ± 0.008**	0.260 ± 0.011	0.161 ± 0.011
30%	0.183 ± 0.007	0.281 ± 0.007	0.180 ± 0.009	0.254 ± 0.008	**0.176 ± 0.009**
40%	0.205 ± 0.008	0.292 ± 0.008	**0.192 ± 0.008**	0.252 ± 0.004	0.194 ± 0.007
50%	0.244 ± 0.006	0.302 ± 0.007	0.222 ± 0.005	0.254 ± 0.005	**0.217 ± 0.006**
60%	0.261 ± 0.009	0.313 ± 0.006	0.242 ± 0.008	0.257 ± 0.005	**0.241 ± 0.006**
70%	0.284 ± 0.009	0.317 ± 0.004	0.273 ± 0.006	0.255 ± 0.005	0.263 ± 0.006
80%	0.298 ± 0.008	0.326 ± 0.003	0.297 ± 0.012	0.254 ± 0.003	0.296 ± 0.010
90%	0.311 ± 0.008	0.326 ± 0.002	0.332 ± 0.006	0.254 ± 0.001	0.318 ± 0.028

Table 28: Gower's distance across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.130 ± 0.008	0.228 ± 0.019	0.215 ± 0.009	0.212 ± 0.009	0.342 ± 0.018	0.265 ± 0.009
20%	0.147 ± 0.005	0.216 ± 0.008	0.215 ± 0.008	0.209 ± 0.009	0.337 ± 0.008	0.269 ± 0.012
30%	0.161 ± 0.006	0.208 ± 0.012	0.212 ± 0.006	0.205 ± 0.006	0.333 ± 0.008	0.280 ± 0.007
40%	0.185 ± 0.007	0.202 ± 0.005	0.211 ± 0.004	0.205 ± 0.005	0.335 ± 0.008	0.246 ± 0.012
50%	0.204 ± 0.008	0.207 ± 0.003	0.214 ± 0.005	0.208 ± 0.005	0.335 ± 0.008	0.304 ± 0.012
60%	0.221 ± 0.010	0.216 ± 0.004	0.217 ± 0.003	0.211 ± 0.003	0.333 ± 0.006	0.297 ± 0.006
70%	0.232 ± 0.009	0.220 ± 0.004	0.215 ± 0.002	**0.209 ± 0.002**	0.334 ± 0.004	0.305 ± 0.005
80%	0.241 ± 0.008	0.221 ± 0.003	0.215 ± 0.002	**0.209 ± 0.002**	0.332 ± 0.005	0.321 ± 0.003
90%	0.232 ± 0.007	0.225 ± 0.003	0.215 ± 0.001	**0.210 ± 0.002**	0.329 ± 0.003	0.348 ± 0.005

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.132 ± 0.010	0.253 ± 0.008	0.130 ± 0.009	0.222 ± 0.008	**0.129 ± 0.009**
20%	0.144 ± 0.008	0.260 ± 0.011	**0.142 ± 0.007**	0.220 ± 0.010	0.142 ± 0.007
30%	0.158 ± 0.005	0.273 ± 0.006	0.157 ± 0.006	0.217 ± 0.006	**0.154 ± 0.005**
40%	0.174 ± 0.005	0.291 ± 0.007	0.169 ± 0.005	0.215 ± 0.003	**0.166 ± 0.003**
50%	0.205 ± 0.004	0.308 ± 0.006	0.193 ± 0.004	0.217 ± 0.004	**0.183 ± 0.005**
60%	0.222 ± 0.006	0.328 ± 0.005	0.211 ± 0.006	0.220 ± 0.003	**0.202 ± 0.005**
70%	0.241 ± 0.006	0.342 ± 0.004	0.236 ± 0.004	0.218 ± 0.003	0.218 ± 0.004
80%	0.250 ± 0.004	0.355 ± 0.003	0.253 ± 0.008	0.216 ± 0.003	0.239 ± 0.006
90%	0.259 ± 0.004	0.362 ± 0.002	0.277 ± 0.004	0.217 ± 0.001	0.256 ± 0.017

Table 29: Downstream predictive R^2 degradation across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.023 ± 0.008	0.038 ± 0.017	0.100 ± 0.021	0.113 ± 0.020	0.330 ± 0.044	0.339 ± 0.045
20%	0.034 ± 0.012	0.114 ± 0.029	0.192 ± 0.026	0.217 ± 0.030	0.479 ± 0.026	0.449 ± 0.034
30%	0.050 ± 0.023	0.173 ± 0.033	0.276 ± 0.033	0.324 ± 0.041	0.571 ± 0.056	0.499 ± 0.056
40%	0.065 ± 0.017	0.217 ± 0.020	0.356 ± 0.040	0.414 ± 0.052	0.630 ± 0.041	0.760 ± 0.096
50%	**0.059 ± 0.032**	0.325 ± 0.048	0.448 ± 0.035	0.559 ± 0.038	0.672 ± 0.034	1.180 ± 0.118
60%	**0.060 ± 0.034**	0.480 ± 0.054	0.562 ± 0.066	0.697 ± 0.109	0.689 ± 0.055	1.152 ± 0.096
70%	**0.069 ± 0.058**	0.522 ± 0.084	0.650 ± 0.102	0.831 ± 0.148	0.705 ± 0.061	1.223 ± 0.147
80%	**0.068 ± 0.055**	0.527 ± 0.145	0.674 ± 0.104	0.954 ± 0.164	0.653 ± 0.070	1.076 ± 0.228
90%	0.179 ± 0.133	0.284 ± 0.146	0.746 ± 0.293	1.552 ± 0.745	0.489 ± 0.072	0.813 ± 0.189

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.010 ± 0.007	0.269 ± 0.043	**0.010 ± 0.004**	0.059 ± 0.023	0.011 ± 0.006
20%	0.016 ± 0.015	0.315 ± 0.029	0.015 ± 0.018	0.097 ± 0.031	**0.015 ± 0.012**
30%	0.040 ± 0.019	0.282 ± 0.041	0.057 ± 0.028	0.131 ± 0.038	**0.030 ± 0.020**
40%	**0.025 ± 0.018**	0.263 ± 0.042	0.126 ± 0.043	0.166 ± 0.045	0.065 ± 0.038
50%	0.157 ± 0.059	0.216 ± 0.031	0.239 ± 0.057	0.201 ± 0.042	0.121 ± 0.048
60%	0.277 ± 0.090	0.197 ± 0.040	0.394 ± 0.108	0.287 ± 0.083	0.226 ± 0.098
70%	0.357 ± 0.116	0.175 ± 0.053	0.675 ± 0.166	0.318 ± 0.116	0.368 ± 0.121
80%	0.388 ± 0.138	0.104 ± 0.050	0.720 ± 0.204	0.298 ± 0.106	0.552 ± 0.223
90%	0.272 ± 0.122	**0.042 ± 0.024**	0.901 ± 0.281	0.414 ± 0.232	1.232 ± 0.527

Table 30: PCA Procrustes distance across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.024 ± 0.004	0.045 ± 0.006	0.043 ± 0.003	0.060 ± 0.005	0.122 ± 0.007	0.112 ± 0.011
20%	0.067 ± 0.008	0.096 ± 0.007	0.091 ± 0.006	0.121 ± 0.008	0.227 ± 0.020	0.225 ± 0.020
30%	0.124 ± 0.010	0.154 ± 0.020	0.142 ± 0.005	0.185 ± 0.009	0.335 ± 0.015	0.361 ± 0.022
40%	0.208 ± 0.011	0.198 ± 0.011	0.198 ± 0.007	0.257 ± 0.008	0.436 ± 0.017	0.367 ± 0.019
50%	0.319 ± 0.021	0.284 ± 0.011	0.282 ± 0.014	0.354 ± 0.016	0.559 ± 0.018	0.492 ± 0.025
60%	0.441 ± 0.015	0.393 ± 0.018	0.378 ± 0.011	0.455 ± 0.013	0.657 ± 0.024	0.584 ± 0.020
70%	0.562 ± 0.024	0.503 ± 0.013	0.476 ± 0.011	0.552 ± 0.014	0.761 ± 0.022	0.712 ± 0.025
80%	0.700 ± 0.030	0.627 ± 0.022	0.610 ± 0.014	0.673 ± 0.011	0.849 ± 0.013	0.834 ± 0.013
90%	0.836 ± 0.027	0.793 ± 0.024	0.780 ± 0.016	0.813 ± 0.014	0.928 ± 0.014	0.929 ± 0.012

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.021 ± 0.003	0.112 ± 0.010	0.022 ± 0.004	0.045 ± 0.003	**0.020 ± 0.003**
20%	0.055 ± 0.006	0.240 ± 0.027	0.057 ± 0.006	0.095 ± 0.006	**0.053 ± 0.004**
30%	0.101 ± 0.006	0.382 ± 0.028	0.105 ± 0.007	0.151 ± 0.006	**0.094 ± 0.005**
40%	0.185 ± 0.013	0.523 ± 0.023	0.176 ± 0.013	0.209 ± 0.009	**0.149 ± 0.009**
50%	0.310 ± 0.019	0.648 ± 0.019	0.286 ± 0.016	0.293 ± 0.013	**0.235 ± 0.013**
60%	0.451 ± 0.016	0.753 ± 0.029	0.424 ± 0.015	0.387 ± 0.010	**0.340 ± 0.014**
70%	0.600 ± 0.031	0.836 ± 0.021	0.595 ± 0.030	0.483 ± 0.018	**0.451 ± 0.026**
80%	0.743 ± 0.022	0.896 ± 0.015	0.765 ± 0.028	**0.607 ± 0.018**	0.620 ± 0.029
90%	0.877 ± 0.018	0.952 ± 0.012	0.904 ± 0.015	**0.772 ± 0.018**	0.809 ± 0.025

Table 31: PLS Procrustes distance across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.027 ± 0.004	0.047 ± 0.007	0.046 ± 0.004	0.062 ± 0.006	0.140 ± 0.011	0.134 ± 0.014
20%	0.069 ± 0.007	0.100 ± 0.006	0.096 ± 0.006	0.125 ± 0.009	0.254 ± 0.022	0.257 ± 0.024
30%	0.124 ± 0.007	0.157 ± 0.017	0.148 ± 0.007	0.187 ± 0.010	0.364 ± 0.020	0.393 ± 0.026
40%	0.207 ± 0.012	0.203 ± 0.013	0.206 ± 0.009	0.260 ± 0.010	0.466 ± 0.020	0.403 ± 0.020
50%	0.314 ± 0.020	0.290 ± 0.012	0.292 ± 0.014	0.358 ± 0.019	0.589 ± 0.020	0.526 ± 0.024
60%	0.439 ± 0.021	0.396 ± 0.020	0.385 ± 0.016	0.455 ± 0.016	0.680 ± 0.025	0.610 ± 0.023
70%	0.562 ± 0.029	0.510 ± 0.011	0.487 ± 0.009	0.556 ± 0.013	0.778 ± 0.023	0.731 ± 0.025
80%	0.699 ± 0.031	0.626 ± 0.020	0.614 ± 0.013	0.676 ± 0.013	0.861 ± 0.016	0.845 ± 0.014
90%	0.838 ± 0.027	0.798 ± 0.024	0.787 ± 0.014	0.817 ± 0.014	0.934 ± 0.014	0.934 ± 0.013

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.024 ± 0.004	0.134 ± 0.014	0.025 ± 0.005	0.049 ± 0.004	**0.023 ± 0.003**
20%	0.060 ± 0.007	0.272 ± 0.031	0.062 ± 0.008	0.101 ± 0.007	**0.057 ± 0.005**
30%	0.106 ± 0.008	0.414 ± 0.032	0.112 ± 0.009	0.156 ± 0.007	**0.097 ± 0.005**
40%	0.193 ± 0.015	0.553 ± 0.026	0.189 ± 0.013	0.215 ± 0.010	**0.155 ± 0.010**
50%	0.325 ± 0.024	0.675 ± 0.021	0.300 ± 0.020	0.302 ± 0.014	**0.241 ± 0.011**
60%	0.463 ± 0.021	0.771 ± 0.031	0.439 ± 0.017	0.393 ± 0.012	**0.341 ± 0.019**
70%	0.610 ± 0.031	0.849 ± 0.022	0.605 ± 0.026	0.492 ± 0.016	**0.454 ± 0.023**
80%	0.745 ± 0.026	0.904 ± 0.017	0.771 ± 0.030	**0.611 ± 0.019**	0.617 ± 0.029
90%	0.880 ± 0.020	0.955 ± 0.013	0.907 ± 0.015	**0.779 ± 0.016**	0.808 ± 0.026

Table 32: Pearson log-p correlation across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.993 ± 0.004	0.993 ± 0.004	0.983 ± 0.008	0.995 ± 0.002	0.892 ± 0.031	0.761 ± 0.033
20%	0.970 ± 0.008	0.971 ± 0.015	0.952 ± 0.016	0.984 ± 0.008	0.854 ± 0.035	0.631 ± 0.045
30%	0.926 ± 0.015	0.966 ± 0.015	0.927 ± 0.024	**0.973 ± 0.014**	0.763 ± 0.040	0.486 ± 0.047
40%	0.885 ± 0.032	0.959 ± 0.018	0.920 ± 0.023	**0.970 ± 0.015**	0.661 ± 0.056	0.294 ± 0.053
50%	0.846 ± 0.023	0.927 ± 0.021	0.890 ± 0.029	**0.951 ± 0.015**	0.675 ± 0.082	0.522 ± 0.113
60%	0.794 ± 0.032	0.919 ± 0.029	0.875 ± 0.036	**0.941 ± 0.029**	0.600 ± 0.064	0.325 ± 0.119
70%	0.743 ± 0.040	0.895 ± 0.023	0.870 ± 0.038	**0.933 ± 0.024**	0.496 ± 0.134	0.183 ± 0.094
80%	0.665 ± 0.067	0.836 ± 0.050	0.829 ± 0.049	**0.902 ± 0.036**	0.402 ± 0.115	0.092 ± 0.069
90%	0.572 ± 0.048	0.720 ± 0.053	0.754 ± 0.069	**0.823 ± 0.061**	0.386 ± 0.137	0.136 ± 0.094

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	**0.995 ± 0.003**	0.752 ± 0.033	0.994 ± 0.004	0.975 ± 0.008	0.994 ± 0.003
20%	**0.987 ± 0.006**	0.637 ± 0.038	0.986 ± 0.007	0.920 ± 0.019	0.970 ± 0.008
30%	0.966 ± 0.007	0.541 ± 0.043	0.954 ± 0.012	0.856 ± 0.025	0.940 ± 0.015
40%	0.924 ± 0.029	0.442 ± 0.052	0.879 ± 0.025	0.796 ± 0.031	0.902 ± 0.038
50%	0.819 ± 0.042	0.452 ± 0.073	0.808 ± 0.053	0.714 ± 0.025	0.857 ± 0.022
60%	0.655 ± 0.090	0.377 ± 0.071	0.698 ± 0.051	0.656 ± 0.038	0.793 ± 0.030
70%	0.532 ± 0.091	0.306 ± 0.103	0.563 ± 0.074	0.619 ± 0.055	0.749 ± 0.056
80%	0.511 ± 0.077	0.241 ± 0.076	0.539 ± 0.066	0.542 ± 0.037	0.690 ± 0.070
90%	0.549 ± 0.066	0.229 ± 0.101	0.569 ± 0.074	0.523 ± 0.055	0.604 ± 0.088

E.4 MAR RESULTS ON THE HOUSING DATASET

Table 33: NRMSE across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.852 ± 0.033	1.063 ± 0.061	1.009 ± 0.005	1.119 ± 0.012	1.808 ± 0.066	1.415 ± 0.046
20%	0.920 ± 0.029	1.060 ± 0.021	1.012 ± 0.003	1.132 ± 0.005	1.779 ± 0.050	1.458 ± 0.044
30%	0.979 ± 0.020	1.068 ± 0.019	1.018 ± 0.003	1.124 ± 0.008	1.774 ± 0.039	1.534 ± 0.029
40%	1.032 ± 0.027	1.055 ± 0.010	1.023 ± 0.003	1.124 ± 0.007	1.733 ± 0.040	1.371 ± 0.030
50%	1.084 ± 0.019	1.068 ± 0.004	1.033 ± 0.004	1.127 ± 0.002	1.702 ± 0.031	2.007 ± 0.065
60%	1.107 ± 0.026	1.099 ± 0.008	1.046 ± 0.008	1.132 ± 0.006	1.687 ± 0.020	1.965 ± 0.056
70%	1.163 ± 0.069	1.121 ± 0.006	1.071 ± 0.006	1.156 ± 0.032	1.689 ± 0.045	1.894 ± 0.039
80%	1.176 ± 0.039	1.129 ± 0.011	1.088 ± 0.009	1.204 ± 0.031	1.704 ± 0.032	1.903 ± 0.041
90%	1.195 ± 0.040	1.124 ± 0.014	1.091 ± 0.013	1.208 ± 0.022	1.699 ± 0.028	1.934 ± 0.040

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.835 ± 0.024	1.326 ± 0.063	0.835 ± 0.025	1.036 ± 0.027	**0.834 ± 0.024**
20%	0.858 ± 0.022	1.425 ± 0.051	0.858 ± 0.017	1.024 ± 0.016	**0.853 ± 0.020**
30%	0.902 ± 0.012	1.520 ± 0.033	0.903 ± 0.014	1.021 ± 0.012	**0.896 ± 0.013**
40%	0.963 ± 0.018	1.591 ± 0.041	0.930 ± 0.014	1.024 ± 0.009	**0.923 ± 0.009**
50%	1.014 ± 0.013	1.685 ± 0.024	0.975 ± 0.009	1.027 ± 0.004	**0.958 ± 0.007**
60%	1.047 ± 0.011	1.741 ± 0.022	1.016 ± 0.024	1.035 ± 0.008	**0.999 ± 0.013**
70%	1.073 ± 0.017	1.812 ± 0.015	**1.052 ± 0.021**	1.057 ± 0.011	1.055 ± 0.019
80%	1.093 ± 0.022	1.866 ± 0.013	1.080 ± 0.024	**1.080 ± 0.007**	1.115 ± 0.032
90%	1.091 ± 0.030	1.905 ± 0.013	1.101 ± 0.035	**1.085 ± 0.012**	1.128 ± 0.037

Table 34: PFC across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.134 ± 0.019	0.256 ± 0.030	0.229 ± 0.022	0.229 ± 0.022	0.301 ± 0.018	0.243 ± 0.017
20%	0.178 ± 0.012	0.244 ± 0.024	0.247 ± 0.013	0.247 ± 0.013	0.318 ± 0.010	0.276 ± 0.014
30%	0.204 ± 0.016	0.236 ± 0.011	0.235 ± 0.013	0.235 ± 0.013	0.310 ± 0.016	0.270 ± 0.015
40%	0.237 ± 0.016	0.242 ± 0.006	0.240 ± 0.006	0.240 ± 0.006	0.310 ± 0.007	0.241 ± 0.011
50%	0.258 ± 0.012	0.247 ± 0.005	0.244 ± 0.002	0.244 ± 0.002	0.315 ± 0.003	0.291 ± 0.009
60%	0.266 ± 0.010	0.259 ± 0.005	0.242 ± 0.005	0.242 ± 0.005	0.313 ± 0.006	0.293 ± 0.008
70%	0.284 ± 0.022	0.267 ± 0.008	0.256 ± 0.024	0.256 ± 0.024	0.313 ± 0.005	0.295 ± 0.009
80%	0.291 ± 0.027	0.287 ± 0.012	0.291 ± 0.023	0.291 ± 0.023	0.315 ± 0.005	0.299 ± 0.005
90%	0.302 ± 0.020	0.291 ± 0.011	0.295 ± 0.018	0.295 ± 0.018	0.318 ± 0.003	0.302 ± 0.005

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	**0.128 ± 0.019**	0.245 ± 0.015	0.132 ± 0.017	0.232 ± 0.018	0.129 ± 0.017
20%	0.162 ± 0.017	0.270 ± 0.013	**0.154 ± 0.017**	0.248 ± 0.015	0.156 ± 0.018
30%	0.181 ± 0.009	0.264 ± 0.015	**0.177 ± 0.011**	0.237 ± 0.012	0.178 ± 0.011
40%	0.207 ± 0.010	0.273 ± 0.007	**0.194 ± 0.005**	0.241 ± 0.006	0.197 ± 0.004
50%	0.237 ± 0.009	0.289 ± 0.004	0.220 ± 0.005	0.243 ± 0.003	**0.219 ± 0.007**
60%	0.255 ± 0.009	0.291 ± 0.006	0.247 ± 0.013	0.242 ± 0.005	**0.239 ± 0.010**
70%	0.268 ± 0.005	0.296 ± 0.005	0.265 ± 0.010	**0.244 ± 0.004**	0.264 ± 0.018
80%	0.279 ± 0.009	0.301 ± 0.005	0.282 ± 0.012	**0.270 ± 0.027**	0.293 ± 0.027
90%	0.282 ± 0.025	0.307 ± 0.003	0.290 ± 0.026	**0.280 ± 0.025**	0.288 ± 0.025

Table 35: Gower's distance across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.128 ± 0.012	0.216 ± 0.019	0.203 ± 0.017	0.194 ± 0.018	0.323 ± 0.017	0.250 ± 0.012
20%	0.151 ± 0.008	0.214 ± 0.012	0.220 ± 0.011	0.212 ± 0.010	0.341 ± 0.009	0.278 ± 0.012
30%	0.168 ± 0.012	0.214 ± 0.010	0.211 ± 0.010	0.202 ± 0.010	0.334 ± 0.010	0.280 ± 0.012
40%	0.188 ± 0.008	0.223 ± 0.003	0.219 ± 0.004	0.213 ± 0.004	0.337 ± 0.005	0.248 ± 0.009
50%	0.205 ± 0.007	0.226 ± 0.005	0.222 ± 0.003	0.217 ± 0.004	0.338 ± 0.005	0.356 ± 0.005
60%	0.215 ± 0.009	0.238 ± 0.003	0.225 ± 0.004	0.219 ± 0.004	0.338 ± 0.005	0.347 ± 0.009
70%	0.231 ± 0.018	0.244 ± 0.003	0.233 ± 0.012	0.226 ± 0.012	0.337 ± 0.004	0.342 ± 0.009
80%	0.236 ± 0.015	0.254 ± 0.006	0.251 ± 0.010	0.243 ± 0.010	0.338 ± 0.006	0.347 ± 0.006
90%	0.247 ± 0.012	0.254 ± 0.006	0.252 ± 0.009	0.244 ± 0.009	0.335 ± 0.005	0.347 ± 0.007

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	**0.123 ± 0.011**	0.237 ± 0.011	0.125 ± 0.012	0.207 ± 0.012	0.125 ± 0.011
20%	0.144 ± 0.011	0.265 ± 0.011	**0.140 ± 0.009**	0.222 ± 0.012	0.142 ± 0.008
30%	0.159 ± 0.007	0.267 ± 0.010	0.157 ± 0.008	0.212 ± 0.010	**0.156 ± 0.008**
40%	0.176 ± 0.008	0.287 ± 0.007	**0.171 ± 0.003**	0.219 ± 0.005	0.172 ± 0.004
50%	0.199 ± 0.006	0.308 ± 0.005	0.189 ± 0.004	0.221 ± 0.003	**0.187 ± 0.005**
60%	0.212 ± 0.006	0.320 ± 0.006	0.206 ± 0.008	0.222 ± 0.005	**0.202 ± 0.006**
70%	**0.218 ± 0.004**	0.333 ± 0.005	0.218 ± 0.007	0.226 ± 0.004	0.221 ± 0.012
80%	**0.225 ± 0.007**	0.342 ± 0.005	0.230 ± 0.007	0.240 ± 0.013	0.240 ± 0.014
90%	**0.228 ± 0.013**	0.348 ± 0.004	0.236 ± 0.014	0.245 ± 0.014	0.242 ± 0.011

Table 36: Downstream predictive R^2 degradation across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.021 ± 0.008	0.026 ± 0.024	0.068 ± 0.022	0.057 ± 0.023	0.206 ± 0.040	0.215 ± 0.042
20%	0.031 ± 0.015	0.066 ± 0.019	0.143 ± 0.019	0.129 ± 0.022	0.318 ± 0.050	0.320 ± 0.058
30%	0.028 ± 0.019	0.144 ± 0.034	0.225 ± 0.030	0.210 ± 0.039	0.374 ± 0.055	0.353 ± 0.062
40%	0.060 ± 0.010	0.228 ± 0.068	0.275 ± 0.051	0.264 ± 0.052	0.385 ± 0.047	0.513 ± 0.081
50%	0.070 ± 0.031	0.323 ± 0.050	0.341 ± 0.036	0.354 ± 0.056	0.394 ± 0.044	1.264 ± 0.138
60%	**0.057 ± 0.027**	0.453 ± 0.075	0.421 ± 0.045	0.469 ± 0.049	0.418 ± 0.059	1.339 ± 0.156
70%	**0.049 ± 0.018**	0.431 ± 0.055	0.428 ± 0.044	0.651 ± 0.169	0.398 ± 0.031	1.220 ± 0.116
80%	0.071 ± 0.057	0.394 ± 0.036	0.428 ± 0.040	0.596 ± 0.236	0.371 ± 0.058	1.165 ± 0.150
90%	0.138 ± 0.191	0.383 ± 0.073	0.414 ± 0.045	0.616 ± 0.324	0.376 ± 0.050	1.225 ± 0.137

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	0.019 ± 0.009	0.176 ± 0.036	0.017 ± 0.008	0.035 ± 0.019	**0.013 ± 0.007**
20%	0.023 ± 0.014	0.236 ± 0.050	0.020 ± 0.012	0.077 ± 0.026	**0.013 ± 0.006**
30%	0.025 ± 0.012	0.210 ± 0.047	**0.021 ± 0.016**	0.121 ± 0.029	0.040 ± 0.024
40%	0.020 ± 0.024	0.177 ± 0.043	**0.016 ± 0.011**	0.142 ± 0.043	0.059 ± 0.030
50%	**0.039 ± 0.027**	0.135 ± 0.046	0.044 ± 0.039	0.158 ± 0.053	0.084 ± 0.040
60%	0.072 ± 0.033	0.115 ± 0.052	0.116 ± 0.047	0.203 ± 0.031	0.160 ± 0.054
70%	0.056 ± 0.040	0.081 ± 0.037	0.138 ± 0.065	0.197 ± 0.070	0.273 ± 0.097
80%	0.070 ± 0.061	**0.050 ± 0.045**	0.199 ± 0.095	0.211 ± 0.058	0.379 ± 0.205
90%	0.089 ± 0.058	**0.046 ± 0.044**	0.293 ± 0.117	0.179 ± 0.058	0.326 ± 0.142

Table 37: PCA Procrustes distance across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.022 ± 0.004	0.038 ± 0.004	0.038 ± 0.003	0.049 ± 0.005	0.091 ± 0.006	0.081 ± 0.005
20%	0.070 ± 0.009	0.085 ± 0.006	0.086 ± 0.004	0.109 ± 0.004	0.179 ± 0.010	0.186 ± 0.010
30%	0.131 ± 0.008	0.139 ± 0.005	0.141 ± 0.005	0.162 ± 0.006	0.243 ± 0.011	0.284 ± 0.018
40%	0.213 ± 0.029	0.224 ± 0.015	0.214 ± 0.010	0.233 ± 0.011	0.310 ± 0.009	0.289 ± 0.012
50%	0.312 ± 0.021	0.345 ± 0.012	0.307 ± 0.011	0.320 ± 0.010	0.365 ± 0.015	0.458 ± 0.021
60%	0.404 ± 0.027	0.516 ± 0.021	0.421 ± 0.018	0.400 ± 0.010	0.410 ± 0.009	0.494 ± 0.024
70%	0.552 ± 0.109	0.672 ± 0.022	0.568 ± 0.012	0.516 ± 0.027	0.468 ± 0.012	0.521 ± 0.023
80%	0.595 ± 0.058	0.728 ± 0.018	0.646 ± 0.010	0.619 ± 0.039	0.509 ± 0.012	0.553 ± 0.020
90%	0.620 ± 0.055	0.750 ± 0.014	0.697 ± 0.006	0.666 ± 0.026	0.556 ± 0.016	0.595 ± 0.024

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	**0.018 ± 0.003**	0.078 ± 0.007	0.018 ± 0.002	0.039 ± 0.002	0.018 ± 0.002
20%	0.047 ± 0.003	0.191 ± 0.011	**0.047 ± 0.003**	0.089 ± 0.004	0.048 ± 0.004
30%	0.093 ± 0.004	0.298 ± 0.019	0.095 ± 0.005	0.141 ± 0.005	**0.092 ± 0.003**
40%	0.159 ± 0.013	0.388 ± 0.016	**0.143 ± 0.012**	0.213 ± 0.008	0.150 ± 0.009
50%	0.247 ± 0.013	0.459 ± 0.020	**0.225 ± 0.014**	0.306 ± 0.011	0.233 ± 0.012
60%	0.314 ± 0.011	0.509 ± 0.015	**0.292 ± 0.014**	0.412 ± 0.021	0.335 ± 0.023
70%	0.372 ± 0.014	0.541 ± 0.018	**0.372 ± 0.019**	0.557 ± 0.018	0.467 ± 0.042
80%	0.416 ± 0.016	0.565 ± 0.012	**0.413 ± 0.011**	0.642 ± 0.013	0.579 ± 0.064
90%	**0.453 ± 0.023**	0.606 ± 0.018	0.472 ± 0.037	0.694 ± 0.007	0.619 ± 0.079

Table 38: PLS Procrustes distance across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.023 ± 0.004	0.039 ± 0.003	0.039 ± 0.003	0.049 ± 0.005	0.101 ± 0.006	0.095 ± 0.006
20%	0.071 ± 0.009	0.089 ± 0.007	0.091 ± 0.004	0.113 ± 0.005	0.195 ± 0.011	0.208 ± 0.012
30%	0.130 ± 0.008	0.145 ± 0.009	0.147 ± 0.007	0.166 ± 0.009	0.263 ± 0.018	0.307 ± 0.023
40%	0.206 ± 0.025	0.235 ± 0.018	0.224 ± 0.013	0.239 ± 0.016	0.332 ± 0.012	0.318 ± 0.012
50%	0.313 ± 0.021	0.358 ± 0.016	0.319 ± 0.013	0.327 ± 0.011	0.385 ± 0.017	0.465 ± 0.021
60%	0.398 ± 0.033	0.537 ± 0.024	0.435 ± 0.021	0.409 ± 0.015	0.430 ± 0.010	0.499 ± 0.023
70%	0.555 ± 0.111	0.695 ± 0.021	0.582 ± 0.012	0.521 ± 0.030	0.481 ± 0.012	0.521 ± 0.018
80%	0.585 ± 0.071	0.749 ± 0.022	0.658 ± 0.011	0.623 ± 0.041	0.519 ± 0.012	0.551 ± 0.025
90%	0.612 ± 0.063	0.765 ± 0.013	0.708 ± 0.006	0.668 ± 0.028	0.565 ± 0.014	0.588 ± 0.027

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	**0.019 ± 0.003**	0.092 ± 0.008	0.019 ± 0.002	0.041 ± 0.003	0.020 ± 0.003
20%	0.051 ± 0.003	0.215 ± 0.013	**0.051 ± 0.003**	0.094 ± 0.004	0.051 ± 0.004
30%	**0.097 ± 0.006**	0.324 ± 0.026	0.099 ± 0.007	0.148 ± 0.007	0.097 ± 0.006
40%	0.165 ± 0.012	0.412 ± 0.021	**0.149 ± 0.011**	0.224 ± 0.010	0.156 ± 0.010
50%	0.254 ± 0.014	0.477 ± 0.021	**0.230 ± 0.018**	0.317 ± 0.012	0.239 ± 0.015
60%	0.319 ± 0.013	0.526 ± 0.017	**0.298 ± 0.016**	0.426 ± 0.024	0.341 ± 0.025
70%	**0.374 ± 0.016**	0.551 ± 0.019	0.375 ± 0.018	0.571 ± 0.016	0.467 ± 0.043
80%	**0.413 ± 0.016**	0.572 ± 0.013	0.413 ± 0.016	0.653 ± 0.014	0.577 ± 0.063
90%	**0.441 ± 0.017**	0.614 ± 0.015	0.459 ± 0.025	0.704 ± 0.008	0.616 ± 0.074

Table 39: Pearson log-p correlation across missing rates

Rate	MissForest	kNN	Mean	Median	Half-min	SVT_OGpaper
10%	0.994 ± 0.003	0.995 ± 0.004	0.968 ± 0.010	0.987 ± 0.006	0.886 ± 0.021	0.808 ± 0.026
20%	0.972 ± 0.011	**0.988 ± 0.005**	0.918 ± 0.025	0.952 ± 0.017	0.808 ± 0.034	0.682 ± 0.047
30%	0.940 ± 0.023	**0.970 ± 0.015**	0.889 ± 0.023	0.921 ± 0.022	0.698 ± 0.028	0.551 ± 0.042
40%	0.879 ± 0.021	**0.931 ± 0.020**	0.832 ± 0.027	0.854 ± 0.029	0.638 ± 0.042	0.488 ± 0.045
50%	0.810 ± 0.021	0.827 ± 0.037	0.783 ± 0.042	0.767 ± 0.041	0.533 ± 0.041	0.629 ± 0.045
60%	0.751 ± 0.040	0.707 ± 0.063	0.739 ± 0.036	0.674 ± 0.046	0.473 ± 0.056	0.541 ± 0.064
70%	0.593 ± 0.144	0.589 ± 0.083	0.648 ± 0.023	0.518 ± 0.044	0.403 ± 0.057	0.440 ± 0.058
80%	0.513 ± 0.143	0.563 ± 0.080	**0.606 ± 0.042**	0.447 ± 0.043	0.356 ± 0.026	0.386 ± 0.035
90%	0.429 ± 0.118	0.579 ± 0.078	**0.613 ± 0.029**	0.386 ± 0.032	0.302 ± 0.024	0.297 ± 0.048

Rate	NF_OGpaper	SoftImpute	SoftForest	SVT	NuclearForest
10%	**0.996 ± 0.003**	0.797 ± 0.030	0.996 ± 0.002	0.960 ± 0.010	0.994 ± 0.003
20%	0.985 ± 0.005	0.674 ± 0.042	0.985 ± 0.004	0.887 ± 0.027	0.961 ± 0.008
30%	0.959 ± 0.015	0.554 ± 0.036	0.948 ± 0.020	0.824 ± 0.022	0.910 ± 0.028
40%	0.887 ± 0.027	0.487 ± 0.038	0.904 ± 0.024	0.723 ± 0.032	0.829 ± 0.025
50%	0.768 ± 0.028	0.403 ± 0.029	**0.848 ± 0.030**	0.632 ± 0.043	0.747 ± 0.033
60%	0.689 ± 0.031	0.341 ± 0.038	**0.770 ± 0.028**	0.569 ± 0.030	0.629 ± 0.043
70%	0.660 ± 0.048	0.289 ± 0.027	**0.699 ± 0.027**	0.466 ± 0.025	0.436 ± 0.056
80%	0.583 ± 0.051	0.260 ± 0.012	0.594 ± 0.022	0.429 ± 0.039	0.267 ± 0.056
90%	0.524 ± 0.045	0.219 ± 0.018	0.514 ± 0.030	0.450 ± 0.038	0.214 ± 0.089

E.5 RUNTIME COMPARISON

Table 40: Runtime comparison of imputation methods on the metabolomics dataset under MCAR/MAR missingness. Runtime is reported in seconds.

Method	Mean (s)	Std (s)	Min (s)	Max (s)
MissForest	77.787	21.426	49.290	116.269
kNN	0.040	0.009	0.021	0.063
Mean	0.001	0.001	0.000	0.004
Median	0.002	0.001	0.001	0.006
Half-min	**0.000**	0.000	0.000	0.002
SVT_OGpaper	5.889	1.261	4.180	12.519
NF_OGpaper	13.581	2.343	9.826	20.209
SoftImpute	0.384	0.144	0.111	0.895
SoftForest	8.171	1.931	5.264	11.753
SVT	5.660	0.956	4.408	9.867
NuclearForest	13.389	2.365	9.772	21.601

Table 41: Runtime comparison of imputation methods on the metabolomics dataset under MNAR missingness. Runtime is reported in seconds.

Method	Mean (s)	Std (s)	Min (s)	Max (s)
MissForest	89.438	22.845	11.206	111.187
kNN	0.025	0.012	0.006	0.060
Mean	0.001	0.001	0.000	0.005
Median	0.002	0.001	0.001	0.005
Half-min	**0.000**	0.000	0.000	0.001
SVT_OGpaper	5.611	0.769	4.911	9.745
NF_OGpaper	9.751	2.228	5.806	13.438
SoftImpute	0.125	0.085	0.013	0.562
SoftForest	4.344	2.137	0.946	7.712
SVT	5.446	0.448	4.851	7.636
NuclearForest	9.797	2.381	5.936	16.075

Table 42: Runtime comparison of imputation methods on the housing dataset under MCAR missingness. Runtime is reported in seconds.

Method	Mean (s)	Std (s)	Min (s)	Max (s)
MissForest	5.934	0.187	5.516	6.454
kNN	0.034	0.024	0.010	0.144
Mean	0.001	0.000	0.000	0.004
Median	0.001	0.000	0.001	0.003
Half-min	**0.000**	0.000	0.000	0.003
SVT_OGpaper	0.119	0.096	0.002	0.285
NF_OGpaper	0.750	0.062	0.650	0.926
SoftImpute	0.001	0.000	0.000	0.001
SoftForest	0.664	0.045	0.573	0.845
SVT	0.003	0.001	0.002	0.007
NuclearForest	0.659	0.047	0.576	0.882

Table 43: Runtime comparison of imputation methods on the housing dataset under MAR missingness. Runtime is reported in seconds.

Method	Mean (s)	Std (s)	Min (s)	Max (s)
MissForest	5.973	0.198	5.719	6.754
kNN	0.030	0.018	0.008	0.140
Mean	0.001	0.001	0.000	0.007
Median	0.001	0.000	0.001	0.003
Half-min	**0.000**	0.000	0.000	0.001
SVT_OGpaper	0.120	0.096	0.002	0.305
NF_OGpaper	0.761	0.075	0.647	1.121
SoftImpute	0.001	0.000	0.000	0.002
SoftForest	0.661	0.039	0.601	0.820
SVT	0.003	0.001	0.002	0.007
NuclearForest	0.661	0.034	0.604	0.766