Statistics and Computing (2026) 36:265
https://doi.org/10.1007/s11222-026-11008-2

ORIGINAL PAPER

Efficient uncertainty propagation in Bayesian two-step procedures

Svenja Jedhoff[1] · **Hadi Kutabi**[2] · **Anne Meyer**[2] · **Paul-Christian Bürkner**[1]

Received: 16 May 2025 / Accepted: 14 October 2026
© The Author(s) 2026

Abstract

Bayesian inference provides a principled framework for probabilistic reasoning. If inference is performed in two steps, uncertainty propagation plays a crucial role in accounting for all sources of uncertainty and variability. This becomes particularly important when both aleatoric uncertainty, caused by data variability, and epistemic uncertainty, arising from incomplete knowledge or missing data, are present. Examples include surrogate models and missing data problems. In surrogate modeling, the surrogate is used as a simplified approximation of a resource-heavy and costly simulation. The uncertainty from the surrogate-fitting process can be propagated using a two-step procedure. For modeling with missing data, methods like Multivariate Imputation by Chained Equations (MICE) generate multiple datasets to account for imputation uncertainty. These approaches, however, are computationally expensive, as multiple models must be fitted separately to surrogate parameters or imputed datasets. To address these challenges, we propose an efficient two-step approach that reduces computational overhead while maintaining accuracy. By selecting a representative subset of draws or imputations, we construct a mixture distribution to approximate the desired posteriors using Pareto smoothed importance sampling. For more complex scenarios, this is further refined with importance weighted moment matching and an iterative procedure that broadens the mixture distribution to better capture diverse posterior distributions.

Keywords Bayesian inference · Importance sampling · Multiple imputation · Surrogate modeling · Uncertainty propagation

List of symbols

θ	Variable of interest $\theta \sim p(\theta \mid C)$
τ	Underlying variable with $\tau \sim p(\tau \mid C)$
$\tau^{(i)}$	Draws of $p(\tau)$ in the first step, $i = 1, \ldots, m$
$p(\theta \mid \tau^{(i)})$	Posterior distribution corresponding to draw $\tau^{(i)}$ – target distribution
τ^*	Selected draw from $\tau^{(1)}, \ldots, \tau^{(m)}$
$p(\theta \mid \tau^*)$	Posterior distribution corresponding to selected draw τ^* – proposal distribution
$\theta_*^{(s)}$	Draws from the proposal distribution $p(\theta \mid \tau^*)$, $s = 1, \ldots, S$
$w_i(\theta_*^{(s)})$	Importance weight with target $p(\theta \mid \tau^{(i)})$ and proposal $p(\theta \mid \tau^*)$ evaluated at draw $\theta_*^{(s)}$
$\hat{k}_i$	Estimated Pareto shape parameter for PSIS with target $p(\theta \mid \tau^{(i)})$ – used as a diagnostic metric
$\breve{\theta}_*^{(s)}$	Transformed draws through IWMM from the proposal distribution $p(\theta \mid \tau^*)$, $s = 1, \ldots, S$
$\hat{k}_{\mathrm{MM},i}$	Estimated Pareto shape parameter for IWMM with target $p(\theta \mid \tau^{(i)})$
$p_{\mathrm{mix}}(\theta \mid \{\tau^{(j)}\})$	Mixture distribution used as the proposal

✉ Svenja Jedhoff
jedhoff@statistik.tu-dortmund.de

Hadi Kutabi
hadi.kutabi@kit.edu

Anne Meyer
anne.meyer@kit.edu

Paul-Christian Bürkner
paul.buerkner@gmail.com

[1] Department of Statistics, TU Dortmund University, Dortmund, Germany

[2] Data Science in Mechanical Engineering, Karlsruhe Institute of Technology, Karlsruhe, Germany

Published online: 10 November 2026

Missing data problem

D	Dataset containing N observations of target y and p covariates $X = (x_1, \ldots, x_p)$
$D_{\text{obs}} = (y_{\text{obs}}, X_{\text{obs}})$	Observed dataset
θ	Regression model parameters; describing relationship between y and X
$D^{(i)}$	Imputed dataset, $i = 1, \ldots, m$
$p_{\text{MICE}}(\theta \mid D_{\text{obs}})$	Density of interest
J^*	Index set that indicates in which observations the original dataset D had missing values

Surrogate modeling

$\mathcal{M}$	Simulator with input θ
$\widetilde{\mathcal{M}}$	Surrogate model with input θ and trainable surrogate parameters τ
θ	Simulator input parameter
$D_T = \{\theta_T^{(n)}, y_T^{(n)}\}_{n=1}^{N_T}$	Training data produced by the simulator
$\{\tau^{(i)}\}_{i=1}^{m}$	m samples of the posterior distribution $p(\tau \mid D_T)$
θ_I	True simulator input variable
$D_I = \{y_I^{(n)}\}_{n=1}^{N_I}$	Real world measurements produced by the simulator with input θ_I

1 Introduction

Many modeling approaches in science and engineering adopt a two-step framework, wherein an initial modeling or preprocessing phase is followed by a subsequent stage focused on inference or prediction (Liu et al. 2021; Lim et al. 2024; Kim et al. 2020; Shin et al. 2023; Cação et al. 2025). While effective, such two-step procedures introduce additional complexity—particularly in the treatment of uncertainty. A key challenge in two-stage inference lies in accurately propagating uncertainty across steps to ensure the reliability and trustworthiness of the final outcomes (Smith 2013; Peherstorfer et al. 2018). Addressing this challenge often entails computationally intensive and time-consuming procedures. In this paper, we propose a general methodology for efficient and robust uncertainty propagation within Bayesian two-step modeling frameworks. Here, robustness refers to the ability of the method to diagnose and address unreliable importance-sampling approximations, especially when the posterior distributions arising in the two steps differ substantially. The proposed method uses the PSIS diagnostic $\hat{k}$ to detect poor overlap between proposal and target distributions and applies IWMM as an adaptive correction when necessary. We demonstrate the effectiveness of our approach in two widely encountered statistical scenarios that demand careful uncertainty management, as described in the following sections.

A prominent example of a two-step modeling approach is the use of surrogate models in scenarios where simulations of complex phenomena are required—such as in the natural sciences (Koziel and Leifsson 2013; Forrester et al. 2008). These simulations often involve solving differential equations that lack closed-form solutions and must be addressed through computationally expensive numerical methods (Quarteroni et al. 2007). To alleviate these computational challenges, surrogate models (Zhu and Zabaras 2018; Lavin et al. 2022) are employed as simplified approximations of the true system, enabling significantly faster evaluations while preserving essential behaviors of the original model. Surrogate modeling has found broad application across domains such as systems biology (Renardy et al. 2018; Alden et al. 2020), water resource management (Razavi et al. 2012; Garzón et al. 2022), and climate modeling (Kuehnert et al. 2022). In a typical two-step surrogate modeling approach (Reiser et al. 2025), the complex simulator is first approximated using a simpler surrogate trained on a limited set of outputs from the true model. This introduces uncertainty stemming from both limited training data and potential model mismatch. In the second step, inference is performed using the trained surrogate to estimate parameters based on observed data. To ensure reliable parameter estimates, it is critical to account for and appropriately propagate the uncertainties introduced during surrogate construction into the inference stage (Psaros et al. 2023).

Another context in which a two-step procedure is essential arises when working with datasets containing missing values. Rather than discarding incomplete records—which may lead to biased results or information loss—imputation techniques are employed to estimate the missing entries (Sv 2018). Among these, Multivariate Imputation by Chained Equations (MICE) (Van Buuren and Groothuis-Oudshoorn 1999) stands out as a widely adopted method (Rubin 2004; White et al. 2011; Shah et al. 2014). MICE addresses uncertainty from missingness by generating multiple complete datasets, each containing plausible imputations for the missing values (Van Buuren and Groothuis-Oudshoorn 1999). In the second step, a statistical model is independently fitted to each imputed dataset, and the resulting posterior distributions are combined. While this approach effectively incorporates imputation uncertainty, it can become computationally demanding, especially when complex models are used. Nevertheless, it yields more robust and comprehensive posterior estimates by properly accounting for the uncertainty introduced during imputation.

In both examples—surrogate modeling and missing data handling—the key challenge is propagating the uncertainty from the first to the second step. A major drawback of existing methods is their high computational cost, as they require sep-

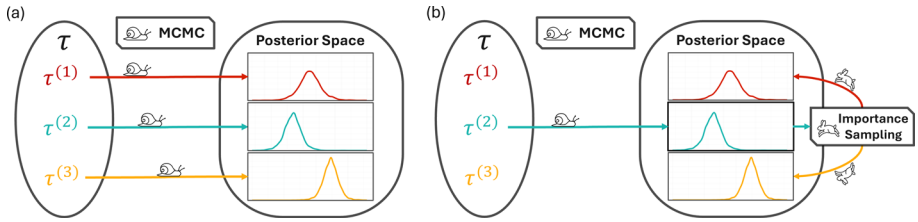


Fig. 1 Overview of problem setting: The goal is to propagate the uncertainty induced by multiple draws of τ into the posterior space. **a** Standard approach: Run costly simulation, i.e. MCMC, for each realization of τ. **b** Proposed method in this paper: Only a few costly simulations, i.e. MCMC, are required; the remaining posterior distributions are approximated using importance sampling

arate model fits for each realization of the uncertain quantity (e.g. surrogate parameters or imputed datasets). This issue is particularly acute in the Bayesian setting, where posterior distributions are often approximated using costly sampling algorithms (Gelman et al. 2013; Robert and Casella 2004), most notably Markov Chain Monte Carlo (MCMC) (Brooks et al. 2011).

A closely related line of work arises in cut models (or cutting feedback), where inference is modularized to prevent information from a downstream (potentially misspecified) module from feeding back into an upstream module (Bayarri et al. 2009; Lunn et al. 2009; Plummer 2015). Therefore, cut models can be viewed as Bayesian two-step procedures and, in this sense, constitute a special case of our framework. In practice, most cut methods fall into two broad classes. First, two-step Monte Carlo approaches, similar in spirit to multiple imputation, propagate uncertainty by repeatedly drawing a realization $\tau^{(i)}$ of the uncertain quantity obtained in the first step and then refitting the second-step Bayesian model conditional on that draw (Plummer 2015). Second, tailored algorithms approximate the cut distribution more directly, for example via tempered or modified transition kernels (motivated in part by the failure of naive cut samplers to target a well-defined stationary distribution) (Plummer 2015), stochastic-approximation methods such as SACut (Liu and Goudie 2021), or semi-modular approaches that downweight feedback rather than removing it entirely (Carmona and Nicholls 2020). Beyond MCMC-based implementations, alternative strategies include posterior-bootstrap procedures for robust modular inference (Pompe and Jacob 2021), generalized-Bayes formulations of cutting feedback (Frazier and Nott 2025), Gaussian-process emulation of conditional posteriors to reduce the number of costly refits (Hutchings et al. 2025), sequential Monte Carlo for cut posterior computation (Mathews et al. 2025), and variational approaches that cast cutting feedback as constrained optimization (Yu et al. 2023; Song et al. 2025). Unbiased MCMC estimators based on couplings provide another general tool for estimating expectations under such targets (Jacob et al. 2020). While these methods address important conceptual and computational challenges, many of them still require repeated downstream posterior computations or substantial additional approximation effort. This motivates the development of efficient uncertainty propagation strategies for Bayesian two-step workflows. In Appendix B, we therefore include an illustrative comparison with the neural variational inference approach to cutting feedback proposed by Song et al. (2025). In the missing-data simulation setting considered there, our PSIS+IWMM-based method yields posterior summaries that are closer to the MCMC baseline than those obtained with the so-called NeVI-Cut approach of Song et al. (2025).

For example, if 100 realizations $\tau^{(1)}, \ldots, \tau^{(100)}$ are generated in the first step, this would typically necessitate estimating 100 separate models—each costly in terms of time and computational resources (see Fig. 1a). To overcome this limitation, we propose an efficient approach that significantly reduces the computational overhead while preserving the trustworthiness of uncertainty propagation. Our approach leverages importance sampling techniques, allowing us to approximate the desired posterior distributions without running MCMC or comparably costly algorithms for each realization $\tau^{(i)}$. Specifically, we employ Pareto Smoothed Importance Sampling (PSIS) (Vehtari et al. 2024) and Importance Weighted Moment Matching (IWMM) (Paananen et al. 2021). With this strategy, MCMC is required only for a small subset of realizations of τ, reducing computational demands. For the remaining realizations, we approximate the corresponding posterior distributions directly within the posterior space using advanced importance sampling (see Fig. 1b). This approach offers a significant gain in efficiency while preserving the accuracy of the propagated uncertainty, making it a practical scalable solution for complex two-step modeling workflows.

2 Method

In this section, we present our proposed methodology for addressing uncertainty propagation within Bayesian two-step modeling frameworks. We begin by formally defining the problem setting and introducing the key components

required for our approach. Section 2.3 then describes the core of our method — an iterative procedure designed for efficient and reliable uncertainty propagation across the two stages. Finally, we discuss two specific application scenarios that require additional notation and domain-specific considerations.

2.1 Problem definition

The primary objective is to obtain information about the distribution of a variable of interest $\theta \sim p(\theta \mid C)$, where C denotes additional context, which depends on the specific problem domain. The marginalized posterior $p(\theta \mid C)$ can be obtained by integrating out the underlying variable τ via

$$p(\theta \mid C) = \int p(\theta \mid \tau, C) p(\tau \mid C)\, d\tau. \tag{1}$$

Here, the probability distribution $p(\tau \mid C)$ is either unknown or intractable, making the direct computation of the above integral infeasible. If we have independent values $\{\tau^{(i)}\}_{i=1}^{m} \sim p(\tau \mid C)$, we can approximate the above integral with a Monte Carlo estimator (Hammersley and Handscomb 1964) as

$$\int p(\theta \mid \tau, C) p(\tau \mid C)\, d\tau \approx \frac{1}{m} \sum_{i=1}^{m} p\left(\theta \mid \tau^{(i)}, C\right). \tag{2}$$

By the law of large numbers, we obtain convergence for $m \to \infty$ (Robert and Casella 2004). However, even for finite but sufficiently large m, we can achieve a reasonable approximation of the marginalized posterior $p(\theta \mid C)$ (Bishop 2006). In the following, we simplify notation by omitting the explicit dependence of $p(\theta \mid \tau, C)$ on the context variable C. As a result, $p(\theta \mid \tau)$ should be understood as implicitly conditioned on C, even though it is not explicitly shown. For a clearer understanding of the role and interpretation of the context C, refer to the two application domains discussed in Sects. 2.4 and 2.5.

In many practical scenarios, the number of first-step realizations m available for uncertainty propagation is limited by the cost of generating or representing uncertainty in the first step. In the multiple-imputation setting, the number of imputed datasets is finite and is typically chosen as a compromise between accurately representing missing-data uncertainty and computational feasibility. Additionally, evaluating $p(\theta \mid \tau^{(i)})$ for each $i = 1, \ldots, m$ can be computationally expensive (Neal 1993). In the setting of Eq. (2), the posterior distribution of θ given a $\tau^{(i)}$, $p(\theta \mid \tau^{(i)})$, cannot be accessed directly and must instead be approximated through expensive sampling algorithms (Gelman et al. 2013), for example, via MCMC (Robert and Casella 2004). For each $\tau^{(i)}$, we then obtain S samples $\{\theta_i^{(s)}\}_{s=1}^{S} \sim p(\theta \mid \tau^{(i)})$ from the corresponding posterior distribution. By treating all $m \cdot S$ draws as one large sample, we can use them to approximate the marginalized posterior distribution $p(\theta \mid C)$ (Lindsay 1995).

This approach can be computationally expensive, depending on the complexity of the distributions involved. A straightforward alternative to running MCMC m times is to select a single representative value τ^* from the set of available values $\{\tau^{(i)}\}_{i=1}^{m}$ and perform MCMC only for this value:

$$p(\theta \mid C) = \int p(\theta \mid \tau) p(\tau \mid C)\, d\tau \approx p(\theta \mid \tau^*). \tag{3}$$

To ensure τ^* is representative of the entire sample, it can be chosen as a summary statistic, such as the mean or median of the m samples, depending on the characteristics of τ. While this approach is computationally efficient—requiring only a single MCMC run—it fails to propagate the uncertainty in $p(\tau \mid C)$, almost surely leading to overconfidence in the resulting marginalized posterior distribution $p(\theta \mid C)$.

A more accurate approximation of $p(\theta \mid C)$ requires using all available samples via the Monte Carlo estimate in Eq. (2). To make this approach computationally feasible, the number of required MCMC runs needs to be reduced. To this end, we propose a combination of importance sampling techniques—specifically, Pareto Smoothed Importance Sampling (PSIS) (Vehtari et al. 2024) and importance weighted moment matching (IWMM) (Paananen et al. 2021), which are described in detail in Sect. 2.2. These methods allow us to approximate the distributions $p(\theta \mid \tau^{(i)})$ for $\{\tau^{(i)}\}_{i=1}^{m}$ within the posterior space as shown in Fig. 1b. This facilitates a more efficient calculation of the Monte Carlo estimate in Eq. (2) than running costly sampling algorithms for each value of $\{\tau^{(i)}\}_{i=1}^{m}$ separately (see Fig. 1a for comparison).

In the following, we introduce the key components required to enhance the computation of the Monte Carlo estimator. These components are then integrated in Sect. 2.3 to construct an efficient iterative algorithm.

2.2 Method components

Our objective is to approximate the distribution $p(\theta \mid \tau^{(i)})$ without relying on computationally expensive sampling algorithms. In the following, for simplicity, we refer only to MCMC to describe this class of expensive but highly accurate algorithms. Instead of running MCMC to access the distribution, we utilize a proposal distribution $p(\theta \mid \tau^*)$ and apply PSIS. When PSIS alone is insufficient to accurately approximate $p(\theta \mid \tau^{(i)})$, we enhance the approximation using IWMM.

Importance sampling The key idea behind importance sampling is to approximate properties of the target distribution $p(\theta \mid \tau^{(i)})$ without drawing samples directly from it.

This allows, among other things, to approximate expectations $\mathbb{E}_{p(\theta|\tau^{(i)})}(h(\theta) \mid \tau^{(i)})$ over the target for any function h (Hammersley and Handscomb 1964).

Rather than sampling directly from the target, a proposal distribution, $p(\theta \mid \tau^*)$ in our framework, is used as an initial approximation. Here, $\tau^* \in \{\tau^{(i)}\}_{i=1}^m$ is a selected value, for which we access the corresponding posterior $p(\theta \mid \tau^*)$ using MCMC. In the following, we use self-normalizing importance sampling, since both target and proposal densities are only known up to a constant. The expectation of interest can then be expressed as follows (Gelman et al. 2013):

$$\mathbb{E}_{p(\theta|\tau^{(i)})}(h(\theta) \mid \tau^{(i)}) = \frac{\int h(\theta) p(\theta \mid \tau^{(i)}) d\theta}{\int p(\theta \mid \tau^{(i)}) d\theta} \tag{4}$$

$$= \frac{\int h(\theta) \frac{p(\theta|\tau^{(i)})}{p(\theta|\tau^*)} p(\theta \mid \tau^*) d\theta}{\int \frac{p(\theta|\tau^{(i)})}{p(\theta|\tau^*)} p(\theta \mid \tau^*) d\theta}. \tag{5}$$

That is, we express expectations over the target equivalently as expectations over the proposal. Then, given samples $\{\theta_*^{(s)}\}_{s=1}^S \sim p(\theta \mid \tau^*)$ drawn from the proposal, the desired target expectation can be estimated as

$$\mathbb{E}_{p(\theta|\tau^{(i)})}(h(\theta) \mid \tau^{(i)}) \approx \frac{\sum_{s=1}^S h(\theta_*^{(s)}) r_i(\theta_*^{(s)})}{\sum_{s=1}^S r_i(\theta_*^{(s)})} \tag{6}$$

$$= \sum_{s=1}^S h(\theta_*^{(s)}) w_i(\theta_*^{(s)}), \tag{7}$$

$$\text{with } r_i(\theta_*^{(s)}) = \frac{p(\theta_*^{(s)} \mid \tau^{(i)})}{p(\theta_*^{(s)} \mid \tau^*)} \tag{8}$$

$$\text{and } w_i(\theta_*^{(s)}) = \frac{r_i(\theta_*^{(s)})}{\sum_{s=1}^S r_i(\theta_*^{(s)})}, \tag{9}$$

$s = 1, \ldots, S$. Here, $r_i(\theta_*^{(s)})$, for $s = 1, \ldots, S$, are referred to as importance ratios and $w_i(\theta_*^{(s)})$ are the corresponding importance weights. Using the same proposal distribution $p(\theta \mid \tau^*)$, we can compute importance weights for each of the posterior distributions $p(\theta \mid \tau^{(i)})$ corresponding to the values $\tau^{(i)} \in \{\tau^{(i)}\}_{i=1}^m$. For importance sampling to be effective and yield reliable approximations, the target and proposal distributions must be sufficiently similar and share the same support. Since both $\tau^{(i)}$ and τ^* are drawn from the same distribution $p(\tau \mid C)$, the corresponding posteriors $p(\theta \mid \tau^{(i)})$ and $p(\theta \mid \tau^*)$ typically share similar structure and support, which enables us to apply importance sampling effectively.

Instead of computing expectations directly, the resulting importance weights can also be used to generate approximate samples from the target distributions via weighted resampling of the original draws. We apply this method, known as importance resampling (Gelman et al. 2013), to obtain random draws from each of the posterior distributions $p(\theta \mid \tau^{(i)})$ for $i = 1, \ldots, m$.

Pareto smoothed importance sampling Importance sampling, and consequently importance resampling, can be unreliable if the proposal distribution is not well-suited for the target distribution (Bugallo et al. 2017). This issue can be diagnosed by examining the distribution of the importance weights $\{w_i(\theta_*^{(s)})\}_{s=1}^S$. If the distribution appears to have excessively heavy tails, importance sampling becomes unstable, often with very large or even infinite variance, leading to unreliable estimates (Ionides 2008; Vehtari et al. 2024). To address this problem, various adaptations of importance sampling have been developed (Ionides 2008; Neal 2001). In the following, we apply PSIS (Vehtari et al. 2024), since it is a reliable and trustworthy variant of importance sampling that outperforms other methods with regard to stabilizing importance sampling estimates.

In this approach, the importance weights are first sorted in ascending order. A Pareto distribution is then fitted to the largest $M = \lfloor \min(0.2S, 3\sqrt{S}) \rfloor$ of the S importance weights. These M largest weights are subsequently replaced with the corresponding quantiles from the fitted generalized Pareto distribution. Beyond improving the stability of the importance weights, PSIS also provides a diagnostic metric $\hat{k}$, to assess the reliability of importance sampling. Performance is evaluated using a threshold, computed as $k_{\text{threshold}} = \min(1 - 1/\log_{10}(S), 0.7)$. For large sample sizes ($S > 2000$), $k_{\text{threshold}} = 0.7$ is sufficient, which is adopted here for simplicity (Vehtari et al. 2024).

If $\hat{k}_i < 0.7$, the Pareto-smoothed importance weights w_i are considered reliable and can be used to resample from $\{\theta_*^{(s)}\}_{s=1}^S$ to obtain $\{\theta_i^{(s)}\}_{s=1}^S \sim p(\theta \mid \tau^{(i)})$. However, when $\hat{k}_i \geq 0.7$, the proposal distribution is deemed insufficiently close to the target distribution for reliable importance sampling. In this case, the importance weights exhibit heavy-tailed behavior, and the Pareto smoothing can no longer adequately control the estimator's error characteristics. Specifically, the bias introduced by smoothing the extreme weights begins to dominate over the estimator's variance, undermining both the accuracy and interpretability of the resulting estimates (Vehtari et al. 2024). As a result, the PSIS-based approximation may be unstable or biased, and a more complex and advanced method might be necessary to successfully approximate the target distribution. To address this, we employ importance-weighted moment matching (IWMM), an adaptive importance sampling method described in detail below.

Importance weighted moment matching The method introduced by Paananen et al. (2021) can be employed when PSIS fails, as indicated by a $\hat{k} > k_{\text{threshold}}$. Importance weighted moment matching (IWMM) is an adaptive importance sampling method, where the proposal distribution is

iteratively updated to better approximate the target distribution. In general, these methods consist of three steps: (i) generating draws from the proposal distribution, (ii) computing the importance weights for these draws, and (iii) adapting the proposal distribution based on the computed weights (Bugallo et al. 2017).

In the specific case of IWMM, step (i) is omitted, so that the same samples are transformed directly and no resampling is necessary. With these samples, the importance weights are calculated in step (ii). In step (iii), the samples are transformed using an affine transformation. While these transformations can vary in complexity, for simplicity, we focus on basic affine transformations, since both the transformation and its Jacobian are computationally cheap (Paananen et al. 2021). To approximate the expectation $\mathbb{E}_{p(\theta|\tau^{(i)})}(h(\theta) \mid \tau^{(i)})$ with a set of draws $\{\theta_*^{(s)}\}_{s=1}^S$ from the proposal distribution $p(\theta \mid \tau^*)$, we use a matrix $\mathbf{A}$ to represent a linear map and a vector $\mathbf{b}$ to represent a translation. A draw $\theta_*^{(s)}$ is then transformed as follows (Paananen et al. 2021):

$$T : \theta_*^{(s)} \mapsto \mathbf{A}\theta_*^{(s)} + \mathbf{b} =: \breve{\theta}_*^{(s)} . \tag{10}$$

Since the transformation is affine and the same for all draws, a new implicit density $p_T(\theta \mid \tau^*)$ is given by

$$p_T(\breve{\theta}_*^{(s)} \mid \tau^*) = p(\theta_*^{(s)} \mid \tau^*) \cdot \mid \mathbf{J}_T \mid^{-1} , \tag{11}$$

where $\mid \mathbf{J}_T \mid^{-1}$ is the inverse Jacobian $\mid \mathbf{J}_T \mid^{-1} = \left| \frac{\mathrm{d}T(\theta)}{\mathrm{d}\theta} \right|^{-1}$. To be more specific, three transformations are used to reduce the mismatch between the samples $\{\theta_*^{(s)}\}_{s=1}^S$ and the target distribution. Transformation T_1 is used to match the mean of the samples to the importance weighted mean. This mean-shift transformation is closely related to the recentered importance sampling approach of McVinish et al. (2013), which also improves importance sampling approximations by translating an existing Monte Carlo sample toward the target distribution. IWMM extends this idea by embedding the mean shift in a broader moment-matching framework that can additionally match marginal variances (with transformation T_2) or the full covariance structure (with transformation T_3) through affine transformations (Paananen et al. 2021). The transformation T_1 to match the mean is given by:

$$\breve{\theta}_*^{(s)} = T_1(\theta_*^{(s)}) = \theta_*^{(s)} - \bar{\theta}_* + \tilde{\theta}_* \tag{12}$$

$$\text{with } \bar{\theta}_* = \frac{1}{S}\sum_{s=1}^{S} \theta_*^{(s)} \tag{13}$$

$$\text{and } \tilde{\theta}_* = \frac{\sum_{s=1}^S w_i(\theta_*^{(s)})\theta_*^{(s)}}{\sum_{s=1}^S w_i(\theta_*^{(s)})} . \tag{14}$$

Transformation T_2 is used to match the marginal variance in addition to matching the mean:

$$\breve{\theta}_*^{(s)} = T_2(\theta_*^{(s)}) = \tilde{\mathbf{v}}^{1/2} \circ \mathbf{v}^{-1/2} \circ (\theta_*^{(s)} - \bar{\theta}_*) + \tilde{\theta}_* \tag{15}$$

$$\text{with } \mathbf{v} = \frac{1}{S}\sum_{s=1}^{S} (\theta_*^{(s)} - \bar{\theta}_*) \circ (\theta_*^{(s)} - \bar{\theta}_*) \tag{16}$$

$$\text{and } \tilde{\mathbf{v}} = \frac{\sum_{s=1}^S w_i(\theta_*^{(s)})(\theta_*^{(s)} - \bar{\theta}_*) \circ (\theta_*^{(s)} - \bar{\theta}_*)}{\sum_{s=1}^S w_i(\theta_*^{(s)})} , \tag{17}$$

with $\circ$ indicating the pointwise product of two vectors. To also match the covariance and the mean, transformation T_3 can be applied:

$$\breve{\theta}_*^{(s)} = T_3(\theta_*^{(s)}) = \tilde{\mathbf{L}}\mathbf{L}^{-1}(\theta_*^{(s)} - \bar{\theta}_*) + \tilde{\theta}_* \tag{18}$$

$$\text{with } \mathbf{L}\mathbf{L}^T = \Sigma = \frac{1}{S}\sum_{s=1}^{S} (\theta_*^{(s)} - \bar{\theta}_*)(\theta_*^{(s)} - \bar{\theta}_*)^T , \tag{19}$$

$$\tilde{\mathbf{L}}\tilde{\mathbf{L}}^T = \frac{\sum_{s=1}^S w_i(\theta_*^{(s)})(\theta_*^{(s)} - \tilde{\theta}_*)(\theta_*^{(s)} - \tilde{\theta}_*)^T}{\sum_{s=1}^S w_i(\theta_*^{(s)})} . \tag{20}$$

The affine transformations are applied in order of increasing complexity, requiring a larger effective sample size to accurately compute the moments for more complex transformations. These transformations are performed iteratively, and for each transformation new weights are computed as well as the corresponding $\hat{k}$ diagnostic from PSIS, now called $\hat{k}_{\text{MM}}$ in the context of IWMM. A transformation is accepted if the new $\hat{k}_{\text{MM}}$ is lower than the previous $\hat{k}_{\text{MM}}$. If a transformation does not improve $\hat{k}_{\text{MM}}$, the next transformation in the sequence is attempted. The algorithm terminates successfully when $\hat{k}_{\text{MM}}$ falls below $k_{\text{threshold}}$. However, if the most complex transformation is not able to improve $\hat{k}_{\text{MM}}$ anymore, but $\hat{k}_{\text{MM}}$ still exceeds $k_{\text{threshold}}$, the algorithm terminates by returning the results of the latest successful transformation and a warning about sampling inaccuracy. In this case, we consider IWMM unsuccessful (Paananen et al. 2021).

Although IWMM is computationally more expensive than PSIS, we employ it in cases where PSIS fails to provide reliable approximations of the target distribution. Figure 2 illustrates representative posterior distributions from two imputed datasets in the simulation study described in Sect. 3.1, within the special case of the missing data problem (see Sect. 2.4). The proposal distribution (shown in red) was obtained via MCMC, and the true target distribution (yellow)—which is typically unknown without running MCMC—serves as the reference. The PSIS-based approximation (light blue), constructed via importance resampling from the proposal, fails to accurately capture the shape and location of the target distribution. This poor approximation is flagged by a high Pareto shape parameter, $\hat{k} = 0.93 > k_{\text{threshold}}$, indicating that the results are unreliable. In contrast, additionally using IWMM results in a substantially improved approxima-

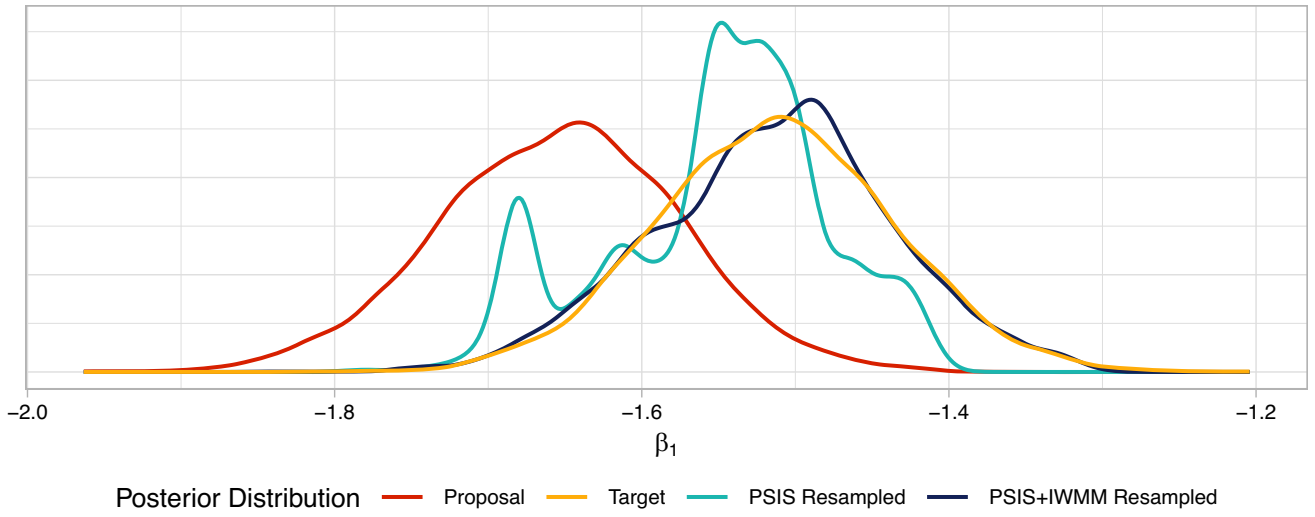


Fig. 2 Exemplary posterior distributions for one model parameter in the special case of the missing data problem (cf. Section 2.4) from the simulation study in Sect. 3.1 (dataset with $N = 100$ observations, $p = 10$ predictors and 15% of rows containing missing values; imputed using MICE). The proposal and target distributions correspond to each one imputed dataset (each approximated with HMC). PSIS alone fails with a $\hat{k} = 0.93 > k_{\text{threshold}}$, which implies an unsuccessful approximation of the target distribution as is clearly visible in the figure. In contrast, the combination of PSIS and IWMM appropriately approximates the target distribution, as indicated by $\hat{k}_{\text{MM}} = 0.27 < k_{\text{threshold}}$

tion (dark blue), with resampled draws that closely match the target distribution. This is confirmed by a much lower shape parameter, $\hat{k}_{\text{MM}} = 0.27 < k_{\text{threshold}}$, indicating a successful and trustworthy approximation.

Selection of representative values To apply both PSIS and IWMM, we require a suitable proposal distribution along with corresponding samples. To obtain such a proposal, we run MCMC to reliably approximate one of the posterior distributions and generate representative draws from it. Specifically, we consider all values $\{\tau^{(i)}\}_{i \in I}$ for which the posterior $p(\theta \mid \tau^{(i)})$ has not yet been approximated. Initially, this set is defined as $I = \{1, \ldots, m\}$. The objective is to select one or more values from $\{\tau^{(i)}\}_{i \in I}$ such that their corresponding posterior distributions $p(\theta \mid \tau^{(i)})$ sufficiently represent the overall space of posteriors. By choosing such representative posteriors as proposals, we aim to maximize the effectiveness of both PSIS and IWMM. Depending on the characteristics of τ, different selection strategies can be applied. These include randomly selecting a subset of values, computing summary statistics (such as the mean or median), or using a similarity or distance-based criterion (Stolte et al. 2024; Kaufman and Rousseeuw 1990). Since the choice of strategy is highly problem-specific, further details are provided in Sect. 2.4 for the case of working with missing data, and in Sect. 2.5 for surrogate modeling scenarios. A more general, problem-independent strategy involves selecting the value in $\{\tau^{(i)}\}_{i \in I}$ associated with the largest $\hat{k}$ or $\hat{k}_{\text{MM}}$ from the previous iteration of the algorithm described in Sect. 2.3. In each iteration, a new proposal distribution is used to apply PSIS and IWMM, targeting the posterior distributions corresponding to the remaining values $\tau^{(i)} \in \{\tau^{(i)}\}_{i \in I}$.

To ensure progress in the iterative algorithm described in Sect. 2.3, we restrict the selection of the representative to the existing set of unexplored values, i.e. $\tau^* \in \{\tau^{(i)}\}_{i \in I}$. This guarantees that the size of the index set I is reduced in each iteration, thus avoiding repetition and eventual stagnation of the algorithm. When choosing a selection strategy like a summary statistic, this might not be the case. Let $\bar{\tau}$ be the summary statistic, for example the mean, of the set $\{\tau^{(i)}\}_{i \in I}$. Then $\bar{\tau}$ is not necessarily part of the set: $\bar{\tau} \notin \{\tau^{(i)}\}_{i \in I}$. It is possible that when choosing $p(\theta \mid \bar{\tau})$ as a proposal for the posteriors $p(\theta \mid \tau^{(i)})$ for $i \in I$, both PSIS and IWMM fail to produce successful approximations for any of the posteriors. In the next iteration a new proposal is chosen to try PSIS and IWMM again. But since no posterior was approximated successfully, the index set I is the same and therefore the value selected for the proposal is the same summary statistic $\bar{\tau}$. This leads to an infinite loop and must therefore be avoided.

A single MCMC run is sufficient to fit a proposal distribution $p(\theta \mid \tau^*)$, which can then be used to approximate the posteriors corresponding to all values $\{\tau^{(i)}\}_{i \in I}$ using either PSIS or IWMM. Alternatively, instead of selecting a single value τ^* to construct the proposal, multiple representative values can be chosen, each requiring a separate MCMC run to approximate its corresponding posterior distribution. Although this approach incurs a higher computational cost, it offers notable advantages: combining the resulting posterior approximations yields a mixture distribution that is broader and more effectively captures the diversity across the poste-

rior space. Such a mixture serves as an improved proposal distribution for PSIS, as it enhances coverage and mitigates the risk of poor overlap with the target distributions. The detailed construction of this mixture distribution is presented in the following section.

Mixture proposal distributions To construct a mixture distribution for use as a proposal in PSIS, multiple representative values must be selected from the set $\{\tau^{(i)}\}_{i\in I}$. Let $\{\tau^{(j)}\}_{j\in I_{\text{mix}}}$ denote this selected subset, where $|I_{\text{mix}}| > 1$ is fixed in advance. For each $j \in I_{\text{mix}}$, we perform an MCMC run to approximate the corresponding posterior distribution and draw samples $\{\theta_j^{(s)}\}_{s=1}^S \sim p(\theta \mid \tau^{(j)})$. This yields a total of $|I_{\text{mix}}| \cdot S$ posterior samples for constructing the mixture distribution.

We then define the mixture proposal as

$$p_{\text{mix}}(\theta \mid \{\tau^{(j)}\}) = \frac{1}{|I_{\text{mix}}|} \sum_{j\in I_{\text{mix}}} p(\theta \mid \tau^{(j)}). \tag{21}$$

To draw samples from $p_{\text{mix}}(\theta \mid \{\tau^{(j)}\})$, we uniformly resample from the combined set of all samples $\{\theta_j^{(s)}\}_{s=1}^S$ for $j \in I_{\text{mix}}$. We denote the resulting resampled set by $\{\theta_*^{(s)}\}_{s=1}^S$.

This mixture distribution serves as the proposal distribution in importance sampling, enabling us to compute the importance ratios (cf. Eq. (8)). Applying Bayes' theorem to compute the ratios for each component $i \in I$ yields:

$$r_i(\theta_*^{(s)}) = \frac{p(\theta_*^{(s)} \mid \tau^{(i)})}{p_{\text{mix}}(\theta_*^{(s)} \mid \{\tau^{(j)}\})} \tag{22}$$

$$= \frac{p(\theta_*^{(s)} \mid \tau^{(i)})}{\frac{1}{|I_{\text{mix}}|}\sum_{j\in I_{\text{mix}}} p(\theta_*^{(s)} \mid \tau^{(j)})} \tag{23}$$

$$= \frac{p(\tau^{(i)} \mid \theta_*^{(s)})p(\theta_*^{(s)})}{p(\tau^{(i)})\frac{1}{|I_{\text{mix}}|}\sum_{j\in I_{\text{mix}}} \frac{p(\tau^{(j)}|\theta_*^{(s)})p(\theta_*^{(s)})}{p(\tau^{(j)})}} \tag{24}$$

$$= \frac{p(\tau^{(i)} \mid \theta_*^{(s)})}{p(\tau^{(i)})\frac{1}{|I_{\text{mix}}|}\sum_{j\in I_{\text{mix}}} \frac{p(\tau^{(j)}|\theta_*^{(s)})}{p(\tau^{(j)})}} \tag{25}$$

$$\propto \frac{p(\tau^{(i)} \mid \theta_*^{(s)})}{\sum_{j\in I_{\text{mix}}} \frac{p(\tau^{(j)}|\theta_*^{(s)})}{p(\tau^{(j)})}}. \tag{26}$$

In the standard setting of self-normalized importance sampling, the normalizing constants—here the marginal likelihoods $p(\tau^{(i)})$—can be ignored since the self-normalization automatically corrects for them. However, in the case of a mixture proposal, the constants of the mixture components cannot be factored out of the summation in the denominator, and thus self-normalization is no longer directly applicable. Therefore, to evaluate Eq. (26), we require either the exact values or relative magnitudes of the marginal likelihoods $p(\tau^{(j)})$ for all $j \in I_{\text{mix}}$. For the purpose of estimating $p(\tau^{(j)})$, we use bridge sampling (Meng and Wong 1996), a method for estimating normalizing constants in complex posterior distributions (Gronau et al. 2021). With these estimates, the mixture-based importance weights can be accurately computed, even though it adds computational complexity.

In our approach, we employ both single and mixture proposal distributions for PSIS, whereas IWMM is applied only with single proposals. Preliminary investigations (not shown here) indicated that using these mixtures within IWMM performs worse than using just a single posterior $p(\theta \mid \tau^*)$ as proposal. We attribute this to the nature of the IWMM method, which relies on affine transformations to shift and scale the proposal distribution to match the moments of the weight distribution (Paananen et al. 2021). A single posterior distribution is more likely to share structural similarities—differing primarily in location and scale—with another posterior (the target) than a multi-modal mixture distribution. Based on these findings, we opted not to further consider mixture proposals when applying IWMM.

2.3 Iterative method

In this subsection, we integrate the methodological components introduced above to construct an iterative algorithm. The goal is to efficiently approximate the posterior distributions $p(\theta \mid \tau^{(i)})$ for all values $\{\tau^{(i)}\}_{i=1}^m$, and to enable the computation of the Monte Carlo estimate defined in Eq. (2). To achieve this, representative values are selected based on a selection strategy (see Sect. 2.2). Algorithm 1 displays the proposed method when selecting a single value τ^* to calculate the proposal distribution. It is also possible to use a mixture of multiple draws for a proposal as explained above. The algorithm for this special case is displayed in the Appendix A as Algorithm 2. Either way, we access the proposal through its set of draws $\{\theta_*^{(s)}\}_{s=1}^S$.

For all remaining $\tau^{(i)}$, PSIS is applied, treating $p(\theta \mid \tau^{(i)})$ as the target distribution and $p(\theta \mid \tau^*)$ as the proposal distribution. If the diagnostic metric $\hat{k}_i$ is below the threshold $k_{\text{threshold}}$, PSIS is considered successful, and importance resampling is performed to obtain posterior draws from the target distribution. However, if PSIS fails, indicated by $\hat{k}_i \geq k_{\text{threshold}}$, IWMM is applied.

If IWMM succeeds, it returns updated draws and importance weights, enabling importance resampling to approximate the desired posterior distribution. If both PSIS and IWMM fail, a new proposal distribution must be selected that better approximates the missing posterior distributions. This results in an iterative algorithm, as outlined in Algorithm 1.

Algorithm 1 provides accurate posterior draws through MCMC or resampled posterior draws through PSIS or IWMM for all $\{\tau^{(i)}\}_{i=1}^m$, corresponding to the desired distribution $p(\theta \mid \tau^{(i)})$. This allows the calculation of the Monte

Algorithm 1 Iterative algorithm for efficiently calculating the marginalized posterior via the Monte Carlo Estimate using PSIS and IWMM.

Require: Variable of interest $\theta \sim p(\theta \mid C)$, underlying variable $\tau \sim p(\tau \mid C)$
1: Draw m samples $\tau^{(1)}, \ldots, \tau^{(m)} \sim p(\tau \mid C)$
2: Set index set $I \leftarrow \{1, \ldots, m\}$
3: **while** $|I| > 0$ **do**
4: Select representative value $\tau^* \in \{\tau^{(i)}\}_{i \in I}$ and run MCMC to access the posterior distribution through samples $\{\theta_*^{(s)}\}_{s=1}^S \sim p(\theta \mid \tau^*)$.
5: Update index set I such that $\tau^* \notin \{\tau^{(i)}\}_{i \in I}$.
6: **for** $i \in I$ **do**
7: Run PSIS with target $p(\theta \mid \tau^{(i)})$, proposal $p(\theta \mid \tau^*)$, obtain importance weights and metric $\hat{k}_i$
8: **if** $\hat{k}_i < 0.7$ **then**
9: Use calculated importance weights and draws $\{\theta_*^{(s)}\}_{s=1}^S \sim p(\theta \mid \tau^*)$ for importance resampling and save the resampled draws as posterior draws corresponding to $p(\theta \mid \tau^{(i)})$.
10: Remove i from I: $I \leftarrow I \setminus \{i\}$
11: **else**
12: Run IWMM with target $p(\theta \mid \tau^{(i)})$ and proposal $p(\theta \mid \tau^*)$ and gain importance weights and metric $\hat{k}_{\mathrm{MM},i}$.
13: **if** $\hat{k}_{\mathrm{MM},i} < 0.7$ **then**
14: Use calculated importance weights and transformed draws $\{\breve{\theta}_*^{(s)}\}_{s=1}^S$ for importance resampling and save the resampled draws as posterior draws corr. to $p(\theta \mid \tau^{(i)})$.
15: Remove i from I: $I \leftarrow I \setminus \{i\}$
16: **end if**
17: **end if**
18: **end for**
19: **end while**
20: Calculate the marginalized posterior via Monte Carlo Estimate $p(\theta \mid C) \approx \frac{1}{m} \sum_{i=1}^m p\left(\theta \mid \tau^{(i)}\right)$.

Carlo estimate from Eq. (2), enabling us to approximate the marginalized posterior distribution $p(\theta \mid C)$. Since each posterior approximation used in the calculation is either based on computationally expensive MCMC or accepted only when the corresponding PSIS diagnostic, $\hat{k}$ or $\hat{k}_{\mathrm{MM}}$ is below the threshold, the procedure includes a diagnostic check intended to retain reliable approximations of the individual posterior distributions. The algorithm improves efficiency compared to running MCMC for all m values separately, as each iteration (cycles of the while-loop in row 3) requires only one MCMC run for the proposal distribution, reducing the number of MCMC runs significantly. The improvement in efficiency is tested through simulation studies in Sect. 3.

The presented method provides an efficient solution for general two-step modeling workflows, potentially applicable beyond the two settings considered here. In this paper, we illustrate its use with two specific applications: a missing data problem and surrogate modeling. For these settings, additional notation is required, which is explained below.

2.4 Special case: missing data problem

We now present a specific application to demonstrate our method: a regression task on a dataset with missing values (Fahrmeir et al. 2013). The missing data introduces uncertainty, which we aim to propagate to the posterior of the model parameters. In this context, the values $\tau^{(1)}, \ldots, \tau^{(m)}$ represent different imputed versions of the dataset, later denoted as $D^{(1)}, \ldots, D^{(m)}$. The formal problem definition follows.

Problem definition In a regression modeling context, we use a dataset D consisting of N observations of a target variable y, along with corresponding observations of p covariates $X = (x_1, \ldots, x_p)$. With a regression model we aim to describe the relationship between the target variable and the covariates. The model is defined by a set of parameters θ, their prior distribution $p(\theta)$, and a likelihood function $p(y \mid X, \theta)$, which specifies the assumed response distribution. The central quantity of interest is the posterior distribution $p(\theta \mid D)$ of the parameters given the data (Gelman et al. 2013).

We now consider the setting in which the observed dataset $D_{\mathrm{obs}} = (y_{\mathrm{obs}}, X_{\mathrm{obs}})$ contains missing values, potentially affecting both the target variable and the predictors. While there are straightforward approaches to handling missing data—such as listwise or pairwise deletion, and mean imputation (Sv 2018)—these methods fail to propagate the uncertainty introduced by the missing values into the parameter estimates. A more principled approach that allows for uncertainty propagation is Multivariate Imputation by Chained Equations (MICE) (Van Buuren and Groothuis-Oudshoorn 1999). MICE is an iterative algorithm that imputes the missing values in the dataset by generating m completed versions of the data, each containing potentially different imputed values at the locations of the missing entries (White et al. 2011). These multiple imputations help capture the uncertainty associated with the missing values in the original dataset D_{obs}. We define the m imputed datasets as $D^{(1)}, \ldots, D^{(m)}$, with observations in each dataset $D^{(i)}$ indexed by the superscript (i) and denote the imputation process performed by MICE via the distribution $p_{\mathrm{MICE}}(D \mid D_{\mathrm{obs}})$. Our goal is to characterize the marginalized posterior distribution of the model parameters θ, which is obtained by integrating over the uncertainty in the imputed datasets:

$$p_{\mathrm{MICE}}(\theta \mid D_{\mathrm{obs}}) = \int p(\theta \mid D)\, p_{\mathrm{MICE}}(D \mid D_{\mathrm{obs}})\, dD \quad (27)$$

$$\overset{\text{Monte Carlo}}{\approx} \frac{1}{m} \sum_{i=1}^m p(\theta \mid D^{(i)}). \quad (28)$$

For this problem setting, our method presented in Sect. 2.3 can be applied, with the context C being the observed dataset

D_{obs}. A structural visualization of the iterative method (cf. Algorithm 1) is illustrated in Fig. 3 for the special case of working with missing data.

Selection of representative datasets To run the iterative method (cf. Section 2.3), we first need to select one or more representative datasets. These datasets should be chosen such that their corresponding posterior distributions, $p(\theta \mid D^{(i)})$, adequately represent the overall posterior distribution space. If only a single dataset D^* is selected, it is desirable for its posterior distribution to be centrally located within this space. In the case of multiple selected datasets, their posterior distributions should collectively cover the space appropriately.

Since we do not have direct access to the posterior distributions corresponding to the datasets without running computationally expensive methods such as MCMC, we assume that if two datasets, $D^{(i)}$ and $D^{(j)}$, are highly similar, their corresponding posterior distributions, $p(\theta \mid D^{(i)})$ and $p(\theta \mid D^{(j)})$, will also be similar. Based on this assumption, we can select representative datasets using similarity measures in the data space. The posterior distributions associated with these representative datasets can then be used as proposal distributions in PSIS and IWMM.

To quantify the pairwise similarity between datasets, an appropriate distance measure must be chosen (Stolte et al. 2024). One possible approach is the Friedman-Rafsky test (Friedman and Rafsky 1979), which constructs a pooled dataset from $D^{(i)}$ and $D^{(j)}$ and forms a complete graph where each sample is represented as a node. The edge weights between nodes are computed based on Euclidean distances between samples. A minimum spanning tree (Cormen et al. 2009) is then constructed, and a distance measure is derived by counting the number of edges in the tree that connect observations from different datasets.

By computing pairwise distances for all m imputed datasets, we form a distance matrix. We use this matrix with a clustering algorithm such as k-medoids (Kaufman and Rousseeuw 1990) to identify a centered dataset or, for $k > 1$, a subset of representative datasets that are well-distributed across the dataset space. For simplicity, we consider the selection of a single dataset D^* in the following discussion. When using a mixture proposal as described in Section 2.2, we select multiple datasets with the k-medoids algorithm for $k > 1$.

Calculation of importance ratios The specific case of handling missing data offers additional advantages that allow our algorithm to operate even more efficiently. In particular, the computation of importance ratios for PSIS and IWMM can be further optimized in this setting—an improvement not necessarily feasible in the more general iterative approach described in Sect. 2.3. This efficiency gain stems from the structural similarity among the imputed datasets $\{D^{(i)}\}_{i \in I}$, which differ only in the entries corresponding to originally missing values. We denote by $J^* \subset \{1, \ldots, N\}$ the index set of observations where these differences occur.

Let D^* be the dataset selected as the representative from the set $\{D^{(i)}\}_{i \in I}$. Using MCMC, we draw samples $\{\theta_*^{(s)}\}_{s=1}^S \sim p(\theta \mid D^*)$ to approximate the posterior given D^*. These samples can then be used to compute importance ratios for the remaining datasets $D^{(i)}$, $i \in I$, as follows:

$$r_i(\theta_*^{(s)}) = \frac{p(\theta_*^{(s)} \mid D^{(i)})}{p(\theta_*^{(s)} \mid D^*)} = \frac{p(\theta_*^{(s)} \mid y^{(i)}, X^{(i)})}{p(\theta_*^{(s)} \mid y^*, X^*)} \tag{29}$$

$$\propto \frac{p(y^{(i)} \mid \theta_*^{(s)}, X^{(i)}) p(\theta_*^{(s)})}{p(y^* \mid \theta_*^{(s)}, X^*) p(\theta_*^{(s)})} \tag{30}$$

$$= \frac{p(y^{(i)} \mid \theta_*^{(s)}, X^{(i)})}{p(y^* \mid \theta_*^{(s)}, X^*)}, s = 1, \ldots, S. \tag{31}$$

We represent the importance ratios using a pointwise likelihood formulation, leveraging the conditional independence of observations, which allows the likelihood to factorize (Bürkner et al. 2020). Given that the datasets differ only in the subset of rows indexed by J^*, we obtain

$$r_i(\theta_*^{(s)}) = \frac{\prod_{n=1}^N p(y_n^{(i)} \mid \theta_*^{(s)}, X^{(i)})}{\prod_{n=1}^N p(y_n^* \mid \theta_*^{(s)}, X^*)} \tag{32}$$

$$= \frac{\prod_{n \in J^*} p(y_n^{(i)} \mid \theta_*^{(s)}, X^{(i)})}{\prod_{n \in J^*} p(y_n^* \mid \theta_*^{(s)}, X^*)}, \tag{33}$$

$s = 1, \ldots, S$, because $p(y_n^{(i)} \mid \theta_*^{(s)}, X^{(i)}) = p(y_n^* \mid \theta_*^{(s)}, X^*)$ for all $n \notin J^*$. This reduction allows us to evaluate only the likelihood terms that correspond to differing rows, decreasing the number of pointwise likelihood evaluations by a factor of $|J^*|/N$, where $|J^*|$ corresponds to the number of rows with missing values. This optimization significantly reduces the computational effort required to compute the importance ratios for PSIS and IWMM, thereby making the overall workflow more efficient.

2.5 Special case: surrogate modeling

The second application of our method is in the context of surrogate modeling. Simulators are essential tools for studying complex natural systems, as they allow researchers to replicate, manipulate, and analyze theoretical models of real-world phenomena under controlled and reproducible conditions (Winsberg 2010; Parker 2009). However, these simulators are often computationally intensive and time-consuming to evaluate (Peck 2004). To alleviate this burden, surrogate models are employed as efficient approximations of the original simulators. These surrogate models aim to replicate the behavior of the simulator while being signifi-

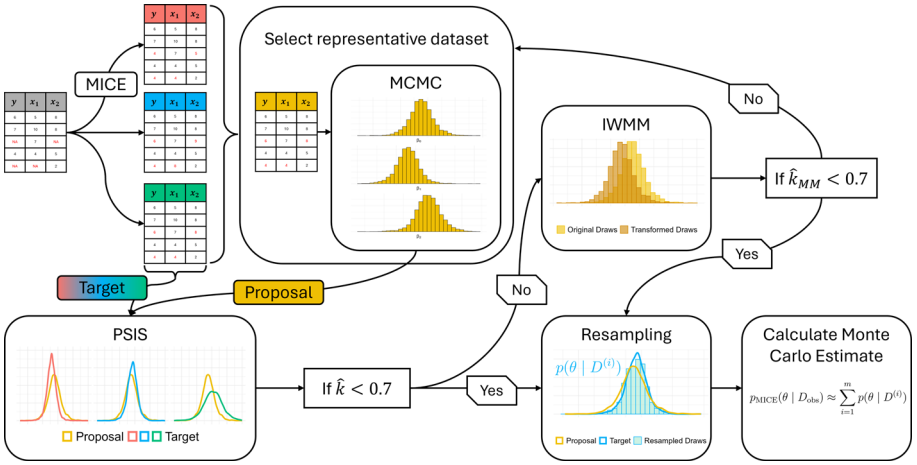


Fig. 3 Iterative Algorithm (cf. Algorithm 1) for the special case of the missing data problem. Multiple datasets are generated by the MICE algorithm, representing the uncertainty induced by the missing values in the original dataset. A representative dataset is selected and the posterior draws accessed through MCMC. This posterior is used to approximate the posteriors of the resulting imputed datasets with PSIS and IWMM, until all posteriors can be approximated using resampling and the Monte Carlo estimate can be calculated

cantly faster to evaluate and simpler to work with (Zhu and Zabaras 2018; Lavin et al. 2022).

An important challenge in this setting is the propagation of uncertainty arising both from the data simulated from a potentially stochastic simulator and from the approximation error of the surrogate model (Reiser et al. 2025). Properly accounting for this uncertainty is crucial for reliable inference, making surrogate modeling a compelling application of our proposed method.

Problem definition We begin by considering a complex simulator $\mathcal{M}$, which may represent, for example, a physical or biological system. Due to its complexity, evaluating $\mathcal{M}$ can be computationally prohibitive. Given input parameters θ_T, the simulator produces an output (Reiser et al. 2025):

$$y_T = \mathcal{M}(\theta_T)\,. \tag{34}$$

The output y_T is assumed to be generated from the true distribution $p(y_T \mid \theta_T)$, governed by the simulation model. Due to the simulator's complexity, however, this likelihood is typically intractable and computationally difficult to evaluate.

To approximate this process, we introduce a surrogate model $\widetilde{\mathcal{M}}$ such that $\widetilde{\mathcal{M}} \approx \mathcal{M}$. The surrogate model is designed to be computationally more tractable while maintaining a reasonable approximation to the true simulator. It is parameterized by trainable parameters τ, leading to the following form (Reiser et al. 2025):

$$\tilde{y}_T = \widetilde{\mathcal{M}}(\theta_T, \tau)\,. \tag{35}$$

Surrogate training step In this step, we train the surrogate model to approximate the behavior of the simulator. To generate training data, we evaluate the simulator using a set of input parameters $\theta_T^{(n)}$, resulting in corresponding outputs $y_T^{(n)}$. This yields a training dataset denoted by $D_T = \{\theta_T^{(n)}, y_T^{(n)}\}_{n=1}^{N_T}$.

Since simulator evaluations are often computationally expensive, the size of the training dataset D_T is typically small. Despite this constraint, we aim to estimate the surrogate parameters τ such that the surrogate model accurately captures the input–output relationship of the simulator.

To this end, we adopt a Bayesian approach to infer the surrogate parameters (Gelman et al. 2013). Unlike the simulator, for which the true data-generating likelihood $p(y_T \mid \theta_T)$ is analytically intractable, the surrogate model provides a tractable likelihood $p(y_T \mid \theta_T, \tau)$. Using this surrogate likelihood and a prior distribution $p(\tau)$ on the surrogate parameters, we construct the posterior distribution:

$$p(\tau \mid D_T) \propto \prod_{n=1}^{N_T} p(y_T^{(n)} \mid \theta_T^{(n)}, \tau)\, p(\tau)\,. \tag{36}$$

We approximate this posterior using sampling-based methods such as MCMC (Robert and Casella 2004), obtaining a set of samples $\{\tau^{(i)}\}_{i=1}^{m} \sim p(\tau \mid D_T)$. These samples capture the uncertainty inherent in fitting the surrogate model to the sparse training data generated by the simulator. In the next step, this uncertainty is propagated to the inference task involving new, unseen data.

Inference step Next, we consider a new set of observations—typically real-world measurements—denoted by $D_I = \{y_I^{(n)}\}_{n=1}^{N_I}$. We assume that the (implicit) data-generating process underlying these observations is well captured by the simulator $\mathcal{M}$. In this context, the measurement data D_I is thought to be generated by the simulator using some unknown input parameter θ_I:

$$y_I = \mathcal{M}(\theta_I)\,. \tag{37}$$

Unlike in the simulation setup, we do not know the input parameters θ_I that generated the outputs. Our goal is to infer θ_I given the observed outputs D_I.

Since the simulator $\mathcal{M}$ does not have a tractable likelihood function, the posterior distribution of θ_I cannot be computed directly using Bayes' theorem. Instead, we rely on the sur-

rogate model, specifically using the posterior samples of its parameters $\{\tau^{(i)}\}_{i=1}^{m}$ obtained during the training phase. The marginalized posterior distribution can be expressed as:

$$p(\theta_I \mid D_T, D_I) = \int p(\theta_I \mid \tau, D_I)\, p(\tau \mid D_T)\, d\tau \tag{38}$$

$$\overset{\text{MC}}{\approx} \frac{1}{m} \sum_{i=1}^{m} p(\theta_I \mid \tau^{(i)}, D_I)\,. \tag{39}$$

To compute this Monte Carlo estimate, we must evaluate $p(\theta_I \mid \tau^{(i)}, D_I)$ for each surrogate parameter sample $\tau^{(i)}$. This inference task aligns with the framework introduced in our iterative method (see Sect. 2.3), which provides an efficient approach to solving this class of problems.

Selection of representative draws To apply our iterative method, we require a selection strategy to identify representative draws from the set $\{\tau^{(i)}\}_{i \in I}$. As discussed in Sect. 2.2, this strategy depends on the specific problem setting. While one approach was outlined for the case of missing data in Sect. 2.4, a different strategy is needed in the context of surrogate modeling.

The selected draws should be representative in the sense that their corresponding posterior distributions collectively cover the diversity of possible posterior shapes—or, in the case of selecting a single draw, that the draw is central in terms of its posterior characteristics.

A straightforward approach is to apply clustering algorithms such as k-means (MacQueen 1967) or k-medoids (Kaufman and Rousseeuw 1990) in the space of the draws τ. However, we cannot be sure that the similarity between surrogate parameter draws $\tau^{(i)}$ and $\tau^{(j)}$ does reliably reflect the similarity of their respective posterior distributions, $p(\theta_I \mid \tau^{(i)}, D_I)$ and $p(\theta_I \mid \tau^{(j)}, D_I)$, due to the non-linear transformation from the surrogate parameter space into the posterior space.

Instead, we aim to infer information about the location and structure of the posteriors $p(\theta_I \mid \tau^{(i)}, D_I)$ without resorting to expensive methods such as MCMC or requiring knowledge of the true parameter θ_I. To this end, we draw samples $\{\theta_I^{(j)}\}_{j=1}^{J} \sim p(\theta_I)$ from the prior. For each draw of surrogate parameters $\tau^{(i)}$, we compute the surrogate log-likelihood of the observed data D_I as

$$L_i(\theta_I^{(j)}) = \sum_{n=1}^{N_I} \log p(y_I^{(n)} \mid \theta_I^{(j)}, \tau^{(i)}), \tag{40}$$

and average these log-likelihoods over all prior samples:

$$\bar{L}_i = \frac{1}{J} \sum_{j=1}^{J} L_i(\theta_I^{(j)}). \tag{41}$$

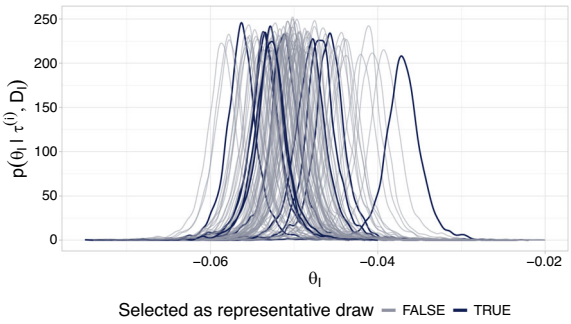


Fig. 4 Posterior distributions $p(\theta_I \mid \tau^{(i)}, D_I)$ for $i = 1, \ldots, 100$ from the logistic-surrogate simulation study in Sect. 3.3. The $N_{\text{mix}} = 10$ distributions highlighted in blue were selected by the proposed method based on log-likelihoods evaluated over prior samples

Ranking the surrogate parameter draws $\tau^{(i)}$ by their corresponding $\bar{L}_i$ values provides insight into the relative positions of the associated posterior distributions. This ranking allows us to select representative draws based on quantiles of the sorted $\bar{L}_i$ values. For instance, when selecting $N_{\text{mix}} = 5$ representative draws for a mixture distribution, we take the draws corresponding to the minimum and maximum $\bar{L}_i$ values, as well as the 25th, 50th, and 75th percentiles.

Figure 4 provides an exemplary visualization of the selection results. It shows the posterior distributions $p(\theta_I \mid \tau^{(i)}, D_I)$ for all surrogate realizations $\tau^{(i)}$ in the logistic-surrogate simulation study (Sect. 3.3). The $N_{\text{mix}} = 10$ distributions highlighted in blue are those selected by the proposed method. The selected posteriors are well distributed across the range of plausible values of θ_I, indicating broader coverage than would typically be obtained from a random subset.

3 Experiments

To showcase our proposed method for propagating uncertainty in practice, we conduct multiple numerical experiments for both the missing value problem and the surrogate modeling. In addition, we assess its performance in a real-world case study using a dataset of taxi trips in New York City that contains missing values. All code and material can be found on GitHub.[1]

3.1 Multiple imputation–simulation study

3.1.1 Simulation setup

For evaluating the performance of the proposed method under different settings, we simulate a variety of datasets. A dataset

[1] https://github.com/sjedhoff/efficient-2step-uncertainty-paper

D was generated with $N \in \{10, 100\}$ observations for a single target variable y and p covariates, where $p \in \{2, 5\}$ for $N = 10$ and $p \in \{10, 30\}$ for $N = 100$. The datasets were generated based on the structure of a linear regression model:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \epsilon, \tag{42}$$

$\epsilon \sim \mathcal{N}(0, 1)$. The model parameters $\beta = (\beta_0, \beta_1, \ldots, \beta_p)$ were drawn from a $\mathcal{N}(0, 1)$ distribution, while the covariates x_i were generated with a mean of 0 and standard deviation sampled from a Gamma distribution with shape and rate parameters set to 10. The datasets were constructed using the `R`-package `simstudy` (Goldfeld and Wujciak-Jens 2024), enabling the generation of correlated data with a correlation coefficient of 0.3 between variables. For each dataset size, 20 datasets were created. Missing values were then introduced in $\lceil \pi \cdot N \rceil$ rows, with $\pi \in \{0.05, 0.15, 0.30\}$. Within each affected row, the target variable y was always missing, along with $\lfloor \frac{p}{2} \rfloor$ randomly selected covariates.

Each dataset was subsequently imputed $m = 100$ times using the MICE algorithm, implemented in the `R`-package `mice` (Sv 2018; Sv and Groothuis-Oudshoorn 2011). Choosing the number of imputations m is nontrivial and depends on several factors such as the fraction of missing information, the desired stability of the resulting inference, and practical constraints in the application. Further guidance on selecting m is provided in Graham et al. (2007), von Hippel (2020). In our setting, we examined how the posterior distributions depend on m and on the number of samples drawn per imputed dataset S. Under the gold-standard approach—running MCMC separately for each imputed dataset—we observed no visible differences between the resulting posterior distributions for $m = 10$, $S = 400$; $m = 100$, $S = 40$ and $m = 1000$, $S = 4$ in our case (see Fig. 13 in Appendix C).

The models were fitted on the imputed datasets via MCMC with the `R`-package `brms` (Bürkner 2017; Bürkner et al. 2024; Stan Development Team 2024) with four chains for 2000 iterations each (including 1000 warm-up and 1000 sampling iterations). Two different prior specifications were tested: the default noninformative (flat) priors in `brms` and the regularized Horseshoe prior (Carvalho et al. 2010; Piironen and Vehtari 2016), which is employed for regularization and model selection (Carvalho et al. 2009).

The simulation study evaluates four different methods. The first is a brute-force approach that fits each of the $m = 100$ imputed datasets individually using *MCMC*. This computationally intensive method is implemented using `brms`. The second method employs the iterative algorithm described in Sect. 2.3, but omits the IWMM step. Specifically, after executing the PSIS step (Line 7 in Algorithm 1) and verifying its success for the given target distribution, the algorithm skips the IWMM step (Lines 11 to 17) and proceeds with a new proposal distribution. This variant is referred to as *PSIS:single*. The third method, termed *PSIS:mixture*, also relies solely on PSIS but utilizes a mixture distribution with $|I_{\text{mix}}| = 5$, as outlined in Algorithm 2 in Appendix A. The fourth method, introduced in Sect. 2.3, integrates both PSIS and IWMM — jointly referred to as *PSIS+IWMM* in the following. Preliminary simulations revealed that using a mixture proposal distribution within IWMM did not improve performance relative to a single proposal; in fact, it often degraded it. This reduction in performance is likely due to the close similarity in the structure of the target and single proposal distribution, whereas the use of a mixture distribution introduces greater divergence, thereby complicating the transformation step in IWMM. Therefore, no mixture distribution is tested for the *PSIS+IWMM* method.

Selection strategies In addition to the different inference methods described above, we also evaluated various strategies for selecting representative datasets to serve as proposal distributions. Three selection strategies were considered. The first, *Random*, simply selects a subset of datasets at random to act as new proposals. The second strategy, *Medoids*, follows the approach described in Sect. 2.4. The third strategy, *Max* $\hat{k}$, chooses datasets with the highest estimated Pareto shape parameter $\hat{k}$ (or $\hat{k}_{\text{MM}}$) in each iteration to be used as new proposals. For the initial round, where no $\hat{k}$ estimates are yet available, the *Medoids* strategy was used.

Performance evaluation To evaluate the efficiency of the proposed method relative to the brute-force *MCMC* approach, we compare the number of log-likelihood evaluations. The accuracy of the resulting marginalized posteriors should be satisfactory, as we only accept importance resampling approximations from PSIS or IWMM when the corresponding shape parameters $\hat{k}$ or $\hat{k}_{\text{MM}}$ fall below the threshold $k_{\text{threshold}}$. This criterion already ensures that only reliable approximations of the target distribution are used (Vehtari et al. 2024), as exemplified in Fig. 2. To further validate this, we compute the mean absolute differences between posterior summaries obtained via *MCMC* and via the *PSIS+IWMM* approach, considering the posterior mean, standard deviation, and the 5% and 95% quantiles.

The models for the PSIS and IWMM proposals are fitted in `brms` (Bürkner 2017; Bürkner et al. 2024) using `Stan` (Stan Development Team 2024), where Hamiltonian Monte Carlo (HMC) (Neal 2011), more specifically the no-U-turn-sampler (NUTS) (Hoffman and Gelman 2014), is employed, as the gold standard for accurate sampling. The majority of NUTS's computational cost arises from the repeated evaluation of the log-probability and its gradient, referred to as gradient evaluations. The log-probability evaluations correspond to the total unnormalized log-posterior density, incorporating both the log-prior of the parameters and the log-likelihood of the observed data given the parameters. In contrast, gradient log-probability evaluations refer

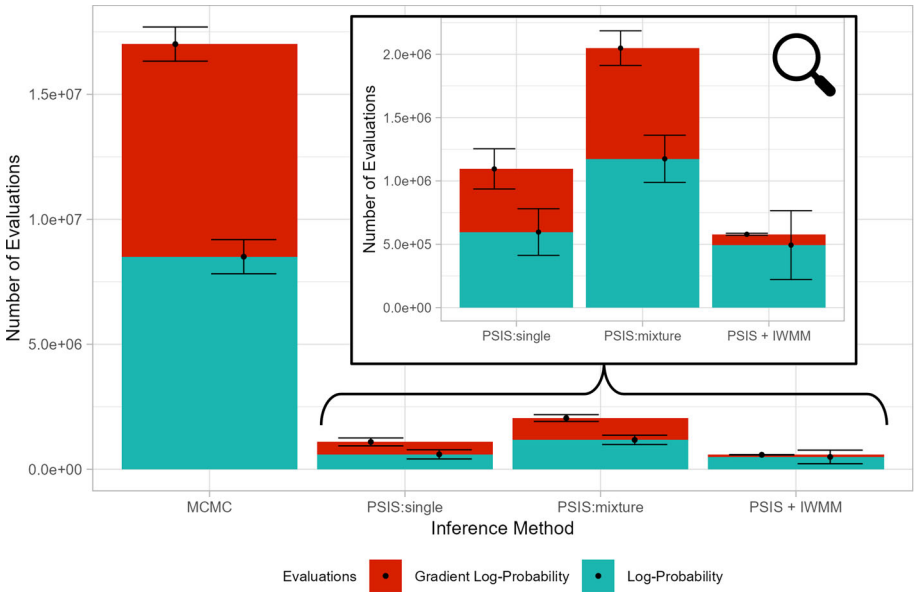


Fig. 5 Number of log-probability (blue) and gradient log-probability evaluations (red) required to approximate the posteriors of $m = 100$ imputed datasets ($N = 100$, $p = 10$) with 15% of the rows containing missing values in the original data. Representative datasets were selected using the *Medoids* method. Error bars indicate the standard deviation across repeated runs of the same setup

to computing the derivative of the log-probability function with respect to all model parameters. Evaluating these gradients is crucial for HMC, as it relies on using gradient information to navigate the parameter space effectively. Gradients are computed via automatic differentiation (Carpenter et al. 2015), and typically account for approximately 80% of the total computational time (Zhang et al. 2022). Therefore, the number of gradient evaluations is a key factor when comparing the computational costs of different methods. In addition, calculating the importance ratios for PSIS and IWMM requires log-likelihood evaluations—both for the numerator and denominator of the ratios. Due to the structure of the importance ratios, evaluating the log-prior terms separately is unnecessary. To compare computational effort across methods, we count log-likelihood evaluations from PSIS and IWMM together with the log-probability evaluations used in HMC. While this slightly overstates the cost for PSIS and IWMM—since log-likelihoods are usually cheaper to compute than full log-probabilities—it offers a consistent and conservative measure of computational demand.

3.1.2 Results of the simulation study

We now present the findings from the simulation study designed to evaluate the proposed method. We focus on the larger dataset with $N = 100$ observations and either $p = 10$ or $p = 30$ covariates. For the smaller dataset with $N = 10$ observations, the problem setting is relatively simple, and our proposed method consistently demonstrated reliable performance across all considered scenarios. As a result, we focus on the more challenging and practical case with $N = 100$ observations. Results for the smaller dataset are provided in Appendix C. Across all experiments, we observed no substantial differences between the various selection strategies for identifying representative datasets. Consequently, we report results using the *Medoids* method only, while additional results for the alternative strategies are included in Appendix C.

Figure 5 displays the number of log-probability and gradient log-probability evaluations required to approximate the posteriors of all $m = 100$ imputed datasets ($N = 100$, $p = 10$), where in the original dataset 15% of rows contained missing values. The results clearly demonstrate that our proposed method substantially reduces the number of required evaluations compared to the brute-force *MCMC* baseline—both in terms of log-probability and gradient computations. The greatest efficiency is achieved when combining PSIS and IWMM (*PSIS+IWMM*), resulting in the lowest computational cost while preserving inference quality and adequately propagating uncertainty across the imputed datasets.

Figure 6 presents the number of MCMC runs required to approximate the posteriors of all $m = 100$ imputed datasets. The number of MCMC runs corresponds to the number of distinct proposal distributions needed for successful approximation of all target posterior distributions using our method, compared with the brute-force approach of running *MCMC* for all imputed datasets. The left panel of the figure shows results for the lower-dimensional case with $p = 10$ predictors. Here, depending on the proportion of missing rows in the original dataset, between 1 and 55 MCMC runs are sufficient—highlighting the efficiency gains relative to running MCMC for all 100 datasets individually. As expected, the number of required MCMC runs increases with the proportion of missing data. Higher rates of missingness lead to greater variation among the imputed datasets and thus more diverse posterior distributions, making approximation more

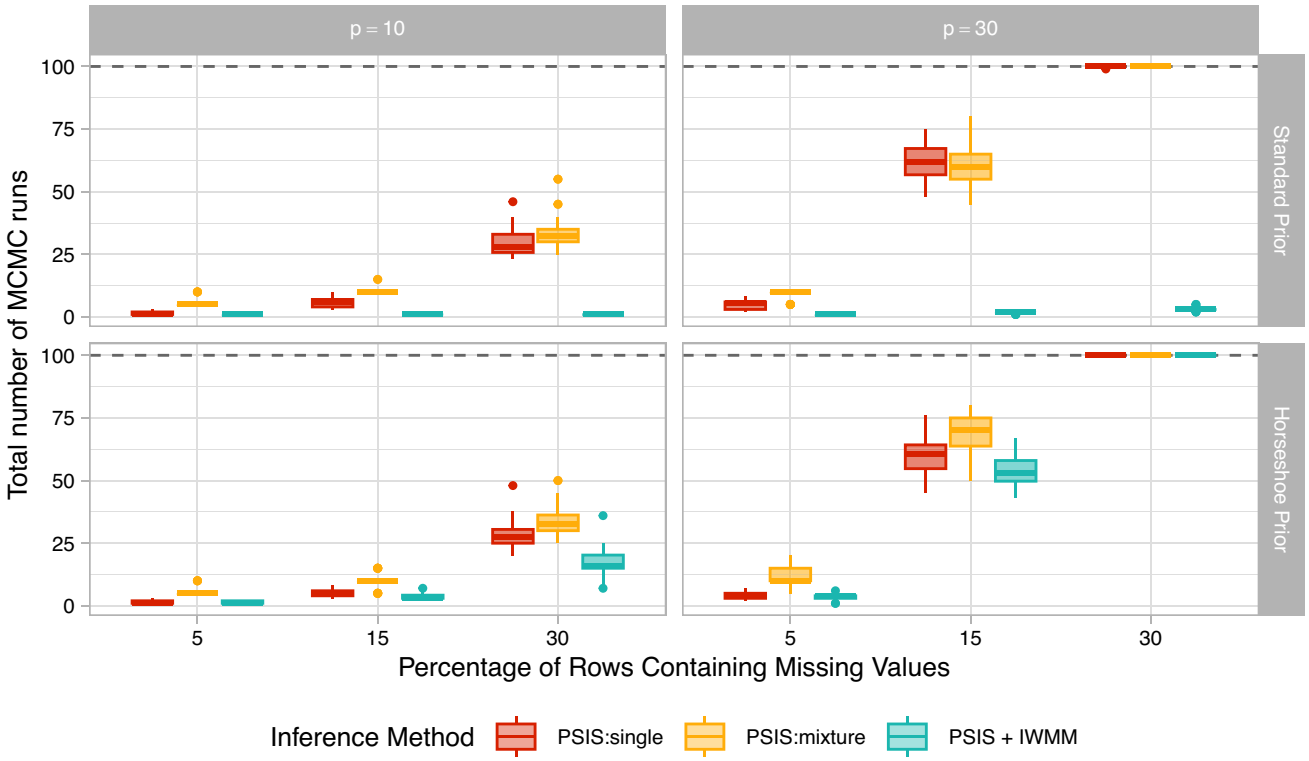


Fig. 6 Number of MCMC runs required to approximate the posteriors of $m = 100$ imputed datasets with $N = 100$ observations, across varying model complexities ($p = 10$ or $p = 30$, columns) and prior specifications (standard `brms` priors vs. regularized horseshoe, rows). Representative datasets were selected using the *Medoids* procedure. The dashed horizontal reference line at 100 indicates the number of MCMC runs needed when fitting each imputed dataset separately

difficult for *PSIS:single*, *PSIS:mixture* and *PSIS+IWMM*. In the high-dimensional case with $p = 30$ predictors (right panel), we observe that PSIS alone—whether based on a single proposal or a mixture—fails when the proportion of rows containing missing values is large. Nonetheless, *PSIS+IWMM* continues to offer substantial efficiency gains in most settings using standard priors. However, under the more challenging regularized horseshoe prior and 30% missing data, *PSIS+IWMM* also struggles to reduce the number of required MCMC runs effectively.

While the previous figure focused on the number of MCMC runs, Fig. 7 (left) reports the ratio of log-probability evaluations needed by the iterative algorithm relative to the number required with brute-force *MCMC* (baseline). A ratio close to 1, indicated by the horizontal reference line, implies that the computational effort in terms of log-probability evaluations is comparable to performing 100 separate MCMC runs. For the lower-dimensional setting (left column), the iterative algorithm requires fewer than half as many evaluations, demonstrating substantial computational savings. Results are similar for both the standard priors and the regularized horseshoe prior, though *PSIS+IWMM* performs slightly better with the standard priors. In contrast, for the higher-dimensional case (right column), *PSIS+IWMM* yields efficiency gains only when standard priors are used. In the most challenging scenario—30% missing data and the horseshoe prior—the number of log-probability evaluations exceeds that of the brute-force *MCMC* baseline, indicating that the method becomes less efficient in this setting.

A similar pattern is observed when examining the ratio of gradient log-probability evaluations, as shown in Fig. 7 (right). Unlike log-probability evaluations, gradient evaluations are only required during HMC sampling and are not needed for the computation of importance ratios in PSIS or IWMM. Consequently, for the inference methods based on our iterative algorithm (*PSIS:single*, *PSIS:mixture*, and *PSIS+IWMM*), the gradient evaluations originate exclusively from the HMC runs used to generate proposal distributions. This structural difference explains why these methods inherently require fewer or, at most, the same number of gradient evaluations as the brute-force *MCMC* approach. For the standard priors (top row in Fig. 7 right), *PSIS+IWMM* shows even greater improvements in terms of gradient efficiency compared to the gains observed for log-probability evaluations. In the more challenging setting with high dimensionality and the horseshoe prior (bottom right), the performance

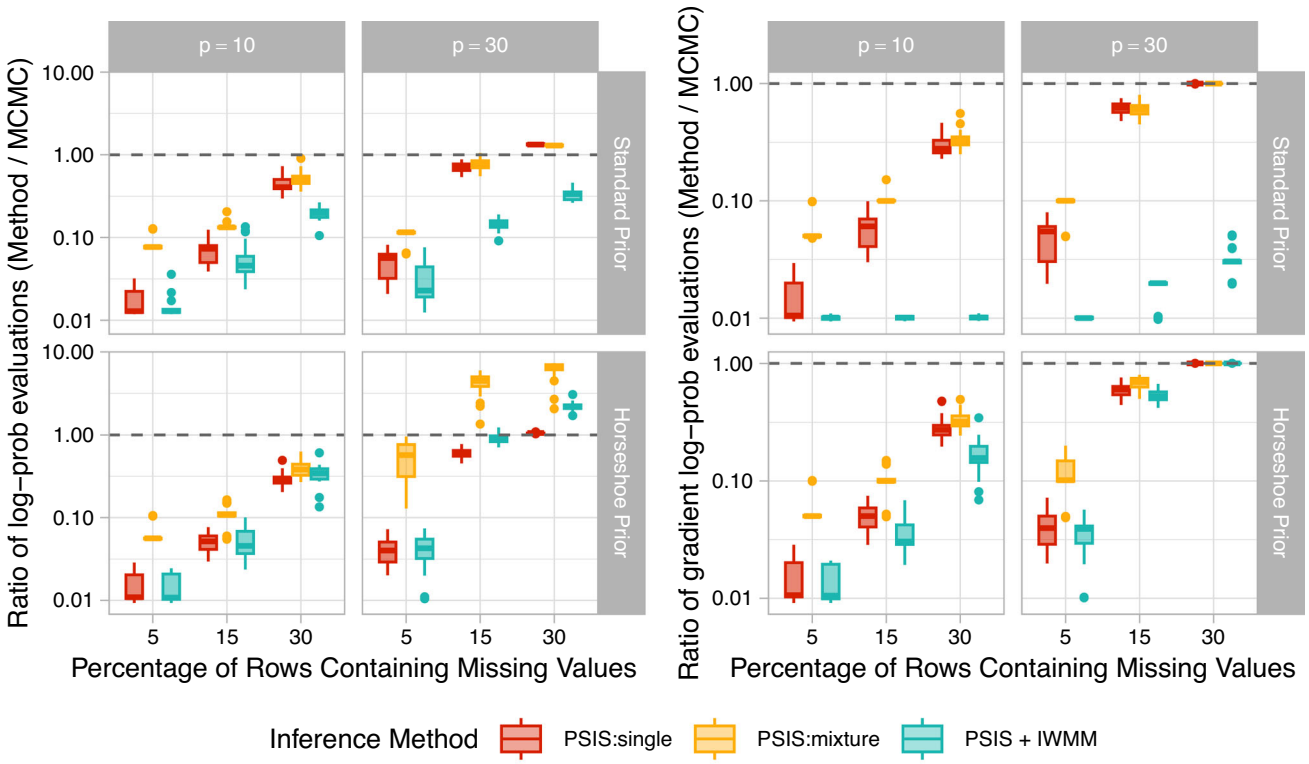


Fig. 7 Ratio of number of evaluations (on a log-scale) of the log-probability (**left**) and of the gradient log-probability (**right**) compared to running *MCMC* separately necessary to approximate the posteriors of $m = 100$ imputed datasets with size $N = 100$ and number of parameters p (in the columns) using either the `brms` standard priors or the regularized horseshoe prior (in the rows). The representative datasets are selected using the *Medoids* procedure

of *PSIS+IWMM* is comparable to that of *PSIS:single* and *PSIS:mixture*, indicating that in this case, the added complexity of *PSIS+IWMM* does not lead to substantial gains.

Overall, we observe that our proposed method offers substantial improvements in terms of number of log-probability and gradient evaluations—an indicator of computational cost—across most settings. For simpler priors, which result in relatively straightforward posterior distributions, the iterative approach combining both PSIS and IWMM significantly outperforms the baseline *MCMC* approach. Even in cases involving more complex posterior shapes, such as those induced by the regularized horseshoe prior, the proposed method continues to outperform alternative inference strategies. The only exception arises in high-dimensional settings (i.e., $p = 30$ covariates) combined with a high proportion of missing data (i.e., 30% of rows with missing values), where the method does not yield improvements over the brute-force *MCMC* approach.

Our proposed method in the experiments with the regularized horseshoe prior performs worse than in those with the standard priors for the following reason. The stronger shrinkage induces substantially greater variation in the shape of the resulting posteriors across different datasets $D^{(i)}$. In particular, the posterior tail behavior can change markedly (e.g., heavier versus lighter tails), and in some settings the posterior can even become multimodal. As a result, posterior distributions differ much more across $D^{(i)}$, which makes approximation via PSIS and IWMM more challenging and increases the probability of failure, especially when the proposal distribution and the target posterior have fundamentally different shapes.

Although our iterative *PSIS+IWMM* approach reduces the number of log-probability and gradient evaluations in most settings, it approximates the posteriors via an additional step by transforming the MCMC samples with PSIS and IWMM, rather than estimating them directly with MCMC. This extra approximation layer can increase Monte Carlo standard errors and, consequently, yield slightly less accurate posterior estimates. To ensure that the approximation remains reliable, we apply the cutoff $k_{\text{threshold}} = 0.7$ for PSIS and IWMM diagnostics, following the recommendations of Vehtari et al. (2024). This threshold is intended to retain only cases where importance sampling provides a sufficiently accurate approximation to the true posterior.

To assess how closely the approximated posterior matches the MCMC baseline, we compare the resampled posteriors

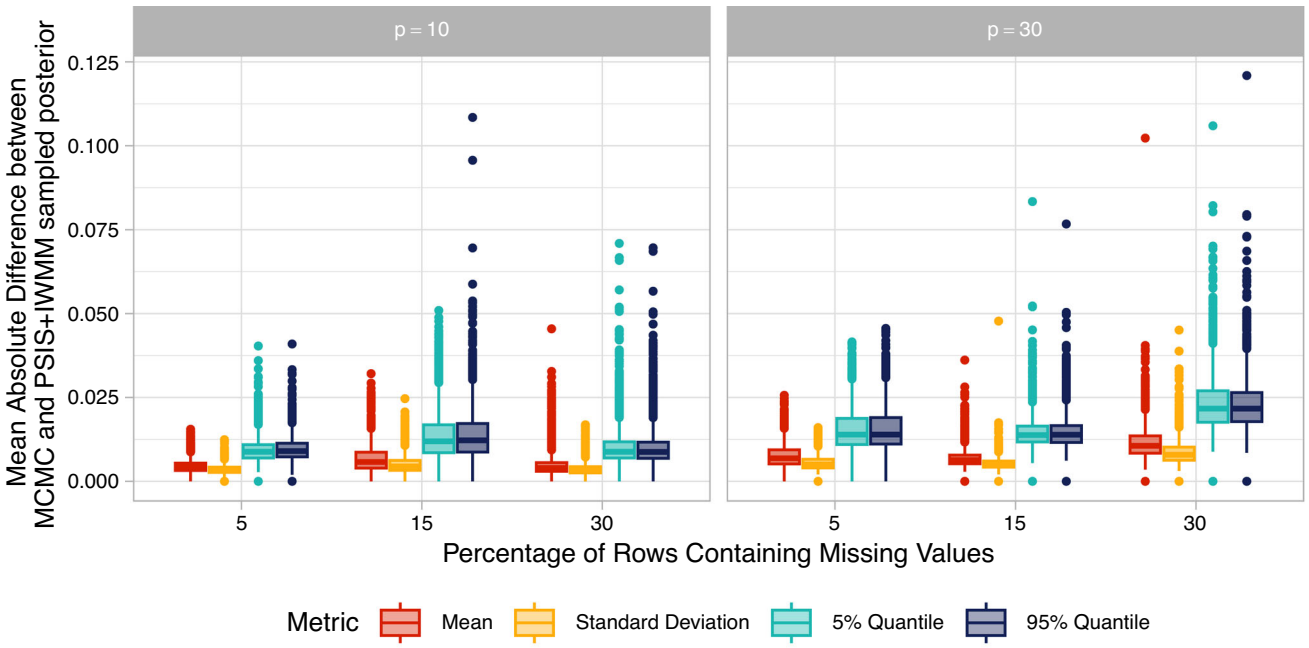


Fig. 8 Mean absolute differences between posterior summaries obtained from *MCMC* and from the *PSIS+IWMM* resampled posterior for datasets of size $N = 100$. For each missingness level (5%, 15%, and 30% of rows containing missing values), boxplots summarize the distribution across the 20 datasets and their 100 imputed datasets; within each dataset, differences are averaged over the p parameters. Colors indicate the summary metric (posterior mean, standard deviation, 5% quantile, and 95% quantile). Results are shown for $p = 10$ (left) and $p = 30$ (right)

to the MCMC posterior using summary statistics. Figure 8 reports absolute differences in the posterior mean, standard deviation, and the 5% and 95% quantiles between *PSIS+IWMM* and *MCMC*. The reported values are averaged over the p parameters. Results are shown for datasets with $N = 100$, with $p = 10$ parameters (left) and $p = 30$ parameters (right). Overall, the posterior means are very similar, with average absolute differences below 0.0125. The same is true for the posterior standard deviations. Accurately approximating tail behavior is more challenging, as reflected by larger differences in the 5% and 95% quantiles; however, the average absolute differences remain below 0.025.

For a multivariate comparison of the posterior distributions, we additionally use the kernel maximum mean discrepancy (MMD) test (Gretton et al. 2012). This is a sample-based test with the null hypothesis that two sets of samples are drawn from the same distribution. Under a good approximation, the test should rarely reject the null at significance level $\alpha = 0.05$. For example, for $N = 100$, $p = 10$, and 30% missingness, the null is rejected in 4.45% of cases, which is consistent with the nominal Type I error rate. In the more challenging setting with $N = 10$ and a relatively large number of parameters ($p = 5$), the rejection rate increases to approximately 10–15%, which is plausible given the greater estimation difficulty.

Additional results for $N = 10$ and for the horseshoe prior are reported in Appendix C. We also provide results for differences between the overall posteriors obtained by pooling across the $m = 100$ imputed datasets (i.e., after combining the corresponding posterior samples).

3.2 Multiple imputation–case study

To evaluate the proposed method on real-world data, we examine a travel time prediction problem. Our case study is inspired by the work of Huang et al. (2020) and utilizes a dataset provided by the NYC Taxi and Limousine Commission (TLC) (NYC 2025), containing yellow taxi trip records in New York City for the year 2016. The dataset includes information such as pickup and dropoff latitude and longitude, timestamps, number of passengers, and vendor ID. Additionally, it contains variables that are only known after trip completion, such as trip distance, trip duration, and payment amount.

We focus on a short-term travel time prediction task, in line with the forecasting approach of Huang et al. (2020). For this purpose, we use only variables available prior to the start of the trip: pickup and dropoff latitude and longitude. In addition, we employ the Open Source Routing Machine (OSRM) (Dibbelt et al. 2016), which uses OpenStreetMap data for New York City to estimate trip duration and dis-

tance based on the trip's start and end locations. These six variables serve as predictors in a Bayesian linear regression model, with the trip duration as the target variable. The model includes a log transformation of the target and scaling of all predictors, and we use the default priors provided by the `brms` (Bürkner et al. 2024) package.

For our analysis, we select data from a single day—Monday, March 7, 2016. The dataset is divided into 24 subsets, one for each hour of the day, based on the trip start time. Depending on the hour, the number of observations ranges from approximately 2000 to 24,000. Each hourly subset contains some missing values, although the target variable (*trip duration*) is always complete. The missing data in the predictors are partly due to failures in the OSRM estimation process. The proportion of missing values per hourly dataset ranges from 1.1 to 1.7%.

To account for uncertainty due to missing data, we perform multiple imputation using MICE (Sv and Groothuis-Oudshoorn 2011), generating $m = 20$ imputed datasets for each hour. We use $m = 20$ imputations, following a common rule of thumb (White et al. 2011). Since fewer than 2% of observations contain missing values, this choice appears reasonable for the present application, although no universal value of m can be guaranteed to be sufficient in all settings. We then apply the iterative algorithm described in Sect. 2.3 to approximate the marginalized posterior distributions of the model parameters, using both our full approach (*PSIS+IWMM*) and a simplified variant without the IWMM step (*PSIS:single*) that employs a single proposal distribution. As demonstrated in the simulation study in Sect. 3.1, incorporating a mixture proposal in the multiple imputation context did not offer significant advantages over using a single proposal. Therefore, we adopted the simpler approach in this analysis. For comparison, we also perform HMC sampling (brute-force *MCMC*) using `brms` (Bürkner et al. 2024; Stan Development Team 2024) on each of the 20 imputed datasets individually.

The results, in terms of log-probability and gradient log-probability evaluations, are shown in Fig. 9a. Notably, our proposed method (*PSIS+IWMM*) requires only about 8% of the log-probability evaluations and 5% of the gradient evaluations compared to the baseline *MCMC* approach. In contrast, the *PSIS:single* variant requires, on average, 13% of the log-probability and 10% of the gradient evaluations relative to *MCMC*. For each of the 24 hourly datasets, the complete iterative algorithm (*PSIS+IWMM*) successfully approximated all 20 posterior distributions using resampling with PSIS and IWMM based on a single randomly selected proposal distribution.

Figure 9b presents the marginalized posterior distributions for one of these hourly datasets, specifically from 12–1 pm. The red line represents the density obtained from the brute-force *MCMC* approach, marginalized over all $m = 20$ imputed datasets, while the blue line shows the corresponding marginalized posterior density from our iterative method using PSIS and IWMM. The close alignment in shape and location of the distributions indicates that our method closely approximates the gold-standard HMC results. The bumpiness seen in the density estimate is mainly a consequence of marginalizing over a finite number of imputed datasets. Each imputed dataset induces a slightly different posterior distribution, and the marginalized posterior is obtained as a finite mixture of these imputation-specific posteriors. With $m = 20$, this mixture can still display visible local irregularities, even when the main posterior features, such as location and spread, are well captured. Increasing the number of imputations would be expected to smooth this marginal approximation, but at the cost of additional runs of the imputation algorithm and corresponding second-step posterior computations. Across all 24 hourly datasets and their $m = 20$ imputations, the absolute difference in posterior means between the *MCMC* baseline and *PSIS+IWMM* are on the order of 10^{-4} when aggregated over all variables; the same holds for the posterior standard deviations. Using the MMD test (Gretton et al. 2012), only 1.25% of the 480 evaluated datasets reject the null hypothesis that the *MCMC* and *PSIS+IWMM* samples are drawn from the same distribution.

3.3 Surrogate models–simulation study

To evaluate the performance of the iterative algorithm in a different context, we conduct a case study based on surrogate modeling. This study replicates and extends the work of Reiser et al. (2025). There, a two-step procedure for uncertainty quantification in surrogate models was developed and tested in multiple experiments. One of these experiments is now re-examined here using the newly developed approach.

3.3.1 Simulation setup

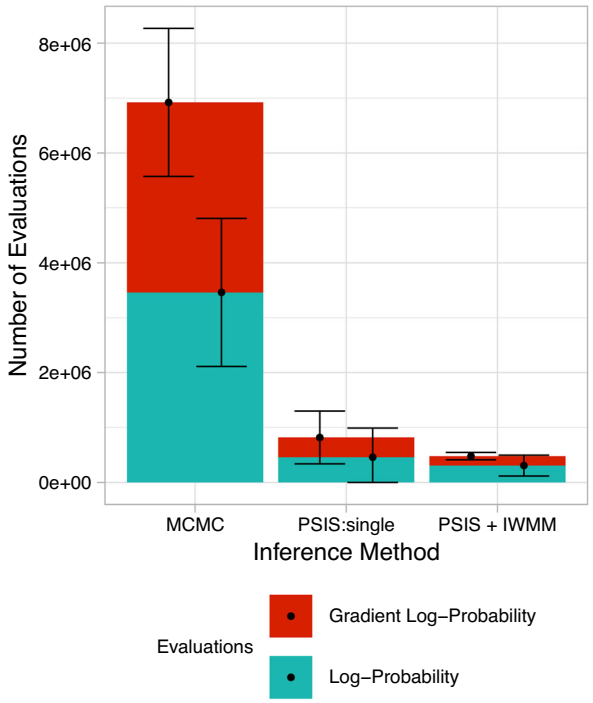
To demonstrate the performance of our method, we first consider a simplified example to avoid the complexities associated with real-life simulators and surrogates. Specifically, we examine a non-linear problem, using a one-dimensional logistic function as the simulator:

$$y = \mathcal{M}(\theta) = \frac{2}{1 + \exp(-10\theta)} - 1\,. \tag{43}$$

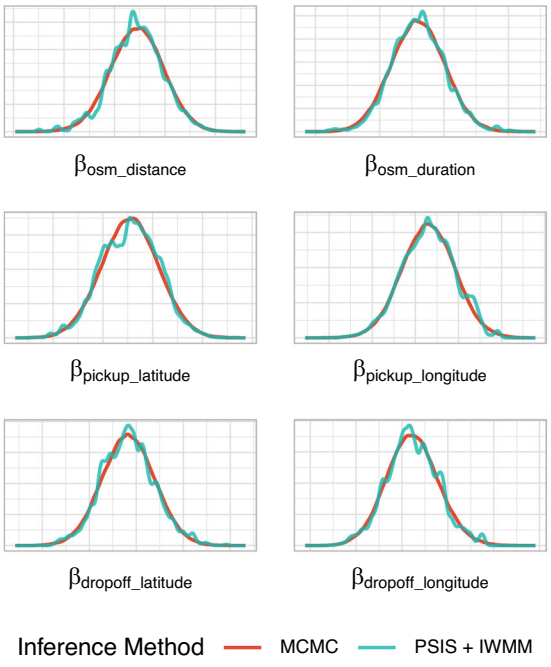
We apply two different surrogate models to approximate this simulator: a logistic surrogate, which includes the original simulator, and a PCE surrogate.

Logistic surrogate For the logistic surrogate, we define

$$\widetilde{\mathcal{M}}_{\text{logistic}}(\theta, \tau) = \frac{\tau_1}{1 + \exp(-\tau_2(\theta - \tau_3))} + \tau_4\,, \tag{44}$$



(a) Number of log-probability (blue) and gradient log-probability evaluations (red) required to approximate the posteriors of $m = 20$ imputed datasets. Error bars indicate the standard deviation across the 24 datasets representing the hours.



(b) Posterior distributions of model parameters to model the trip duration in the NYC taxi dataset, March 7, 2016, 12 pm, with $m = 20$ imputed datasets for two different inference methods.

Fig. 9 Results of the case study with the real world NYC taxi dataset in the multiple imputation setting

with surrogate parameters $\tau = [\tau_1, \tau_2, \tau_3, \tau_4]$. The original simulator is included in this surrogate model by fixing $\tau = [2, 10, 0, -1]$.

For the training phase of the surrogate, we assign normal priors to the surrogate parameters with means equal to the true values $(2, 10, 0, -1)$ and the standard deviations set to $(1, 10, 1, 1)$.

PCE surrogate To test a more challenging scenario, we also consider a surrogate model that does not include the true simulator. Specifically, we use a polynomial chaos expansion (PCE) model (Wiener 1938; Oladyshkin and Nowak 2012; Bürkner et al. 2023)

$$\widetilde{\mathcal{M}}_{\text{PCE}}(\theta, \tau) = \sum_{i=0}^{d} \tau_i \psi_i(\theta)\,, \tag{45}$$

where the parameters are $\tau = [\tau_0, \tau_1, \ldots, \tau_d]$, and ψ_i denote the Legendre polynomials (Sudret 2008). For this study, we choose a polynomial degree of $d = 5$. Each coefficient τ_i is assigned a normal prior, $\tau_i \sim \mathcal{N}(0, 5)$.

Surrogate training step In the initial step, we generate $N_T = 10$ training data points using the simulator. The input values $\{\theta_T^{(n)}\}_{n=1}^{N_T}$ are selected as N_T equally spaced points in the interval $[-1, 1]$. The corresponding outputs $\{y_T^{(n)}\}_{n=1}^{N_T}$ are sampled from a normal distribution centered at the true simulator values with a standard deviation $\sigma = 0.01$, i.e.,

$$y_T^{(n)} \sim \mathcal{N}(\mathcal{M}(\theta_T^{(n)}), \sigma^2)\,. \tag{46}$$

To infer the surrogate parameters, we use MCMC implemented via the probabilistic programming language `Stan` (Stan Development Team 2024), using the `cmdstanr` package in `R` (Gabry et al. 2025). Two chains are run for 1050 iterations each (including 1000 warm-up and 50 sampling iterations), resulting in $m = 100$ posterior draws $\{\tau^{(i)}\}_{i=1}^{m} \sim p(\tau \mid D_T)$. We use a relatively small number of posterior draws to enable a more direct comparison with the missing-data simulation study in Sect. 3.1. Since the choice of m is difficult to determine in general, we also ran the same experiments with $m = 1000$; the corresponding results are reported in Appendix D in Fig. 26.

Inference step For the inference phase, $N_I = 5$ measurement data points are generated from the true simulator

$$y_I \sim \mathcal{N}(\mathcal{M}(\theta_I^*), \sigma^2)\,, \tag{47}$$

where the true input is fixed at $\theta_I^* = -0.05$ and the observation noise is set to $\sigma = 0.01$. The prior for θ_I

is a truncated normal distribution, $\theta_I \sim \mathcal{N}_{[-1,1]}(0, 0.5^2)$, while a uniform prior is chosen for the standard deviation, $\sigma \sim \text{Uniform}(0, 0.05)$.

Simulation parameters As in the multiple imputation setup, several methods are tested. The most straightforward approach involves fitting one separate model using HMC, an *MCMC* method, for each of the $m = 100$ posterior draws $\{\tau^{(i)}\}_{i=1}^m$. In addition, we apply the iterative algorithm with PSIS only, using both a single proposal (*PSIS:single*) and a mixture proposal constructed from five distributions (*PSIS:mixture*), as well as the combined method of PSIS and IWMM (*PSIS+IWMM*).

For selecting the representative draw used for the proposal distribution, multiple selection strategies are examined. These include selecting a *random* draw, choosing the draw with the highest estimated Pareto shape parameter $\hat{k}$ from the previous iteration, and the *log-lik* method described in Sect. 2.5. Additionally, an averaging method is tested, where the mean of the remaining posterior draws $\{\tau^{(i)}\}_{i \in I}$ is used for the proposal. However, this approach resulted in an infinite loop in approximately 60% of the tested settings and therefore is not considered in the subsequent results. The cause of this issue, arising when selecting a value τ^* such that $\tau^* \notin \{\tau^{(i)}\}_{i \in I}$, was discussed in Sect. 2.2.

Performance evaluation To assess the performance of our method, we use the same metrics as those employed in the missing value simulation study presented in Sect. 3.1.

3.3.2 Results of the simulation study

We first examine the number of MCMC runs required by the iterative algorithm, which are used to obtain the posterior distributions serving as proposal distributions in PSIS and IWMM. Since no substantial differences between selection strategies can be observed, presented results are from the *log-lik* method only. Additional results for other selection methods can be found in Appendix D. Figure 10 displays the total number of MCMC runs, categorized by inference method and surrogate type (logistic and PCE). For the logistic surrogate, we observe that *PSIS:single* requires the highest number of MCMC runs, with a median of 37. In contrast, for *PSIS:mixture* only about 19 posterior distributions need to be fitted using MCMC. In most cases, the *PSIS+IWMM* method finishes within 1 or 2 posteriors fitted with MCMC, effectively approximating all posterior distributions and thus requiring a significantly smaller number of MCMC runs. The PCE surrogate represents a more challenging scenario because it does not include the true simulator and therefore inevitably introduces approximation error. This increases variability across the draws $\tau^{(i)}$ and, in turn, leads to greater heterogeneity in the corresponding posteriors $p(\theta \mid \tau^{(i)}, D_I)$, making approximation via PSIS and IWMM more difficult. Both methods *PSIS:single* and *PSIS:mixture* perform similarly, with a median number of MCMC runs of 70 and 63 for the single and mixture proposals, respectively. Despite some outlier cases, the combined *PSIS+IWMM* inference method consistently outperforms the other two in terms of minimizing the number of MCMC runs with a median of 5.

Table 1 Distances between the posterior distributions $p(\theta_I \mid D_T, D_I)$ of θ_I obtained via *MCMC* and via the iterative algorithm *PSIS+IWMM*

m	Surrogate	Mean	Standard deviation
100	Logistic	8.0471e−05	0.0001
	PCE	0.0178	0.0528
1000	Logistic	0.0004	0.0005
	PCE	0.0698	0.0834

Values are the absolute differences in posterior mean and standard deviation, averaged over 20 datasets

For $m = 1000$ (see Fig. 26 in Appendix D), the results are even more favorable. On average, fewer than 6% of the posteriors in the logistic-surrogate setting need to be computed directly via MCMC, and fewer than 20% in the PCE-surrogate setting.

We also assessed how closely the resampled posterior distributions obtained with *PSIS+IWMM* match the gold-standard *MCMC* baseline. Table 1 reports the absolute differences in posterior means and standard deviations, averaged over 20 datasets. For the logistic surrogate, agreement is excellent: posterior means and standard deviations are nearly identical. In contrast, discrepancies are larger for the PCE surrogate. This is likely because the PCE surrogate has more parameters and therefore wider first-step posteriors, which propagates into greater variability in the resulting second-step posterior approximations. When comparing posterior means obtained from MCMC for the individual posteriors $p(\theta_I \mid \tau^{(i)}, D_I)$ across different realizations $\tau^{(i)}$, the logistic-surrogate means differ by an average absolute pairwise distance of about 0.0028 (for $m = 100$). For the PCE surrogate, the corresponding average absolute difference is 0.0664. Hence, the PCE posteriors are substantially more dispersed (i.e., less similar across $\tau^{(i)}$), which helps explain why approximating them with PSIS or IWMM is more challenging.

To further evaluate performance, we analyze the number of log-probability and gradient evaluations required by each inference method, relative to running brute-force *MCMC*. Figure 11 displays stacked bar plots for the different inference methods, with results for the logistic surrogate shown on the left and the PCE surrogate on the right. To better illustrate the relative efficiency of the proposed methods compared to the baseline *MCMC* approach, Table 2 reports the corresponding ratios of required evaluations.

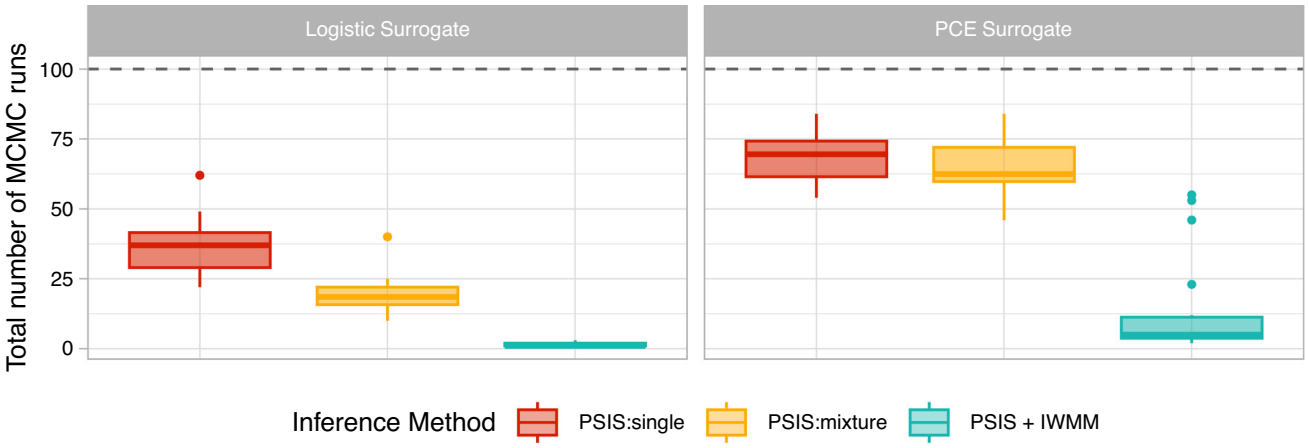


Fig. 10 Number of MCMC runs required to approximate the posteriors of $m = 100$ surrogate parameter draws, for the logistic and PCE surrogate (in the columns). Representative datasets were selected using the *log-lik* procedure. The dashed horizontal line at 100 indicates the number of MCMC runs needed for the brute-force *MCMC* approach

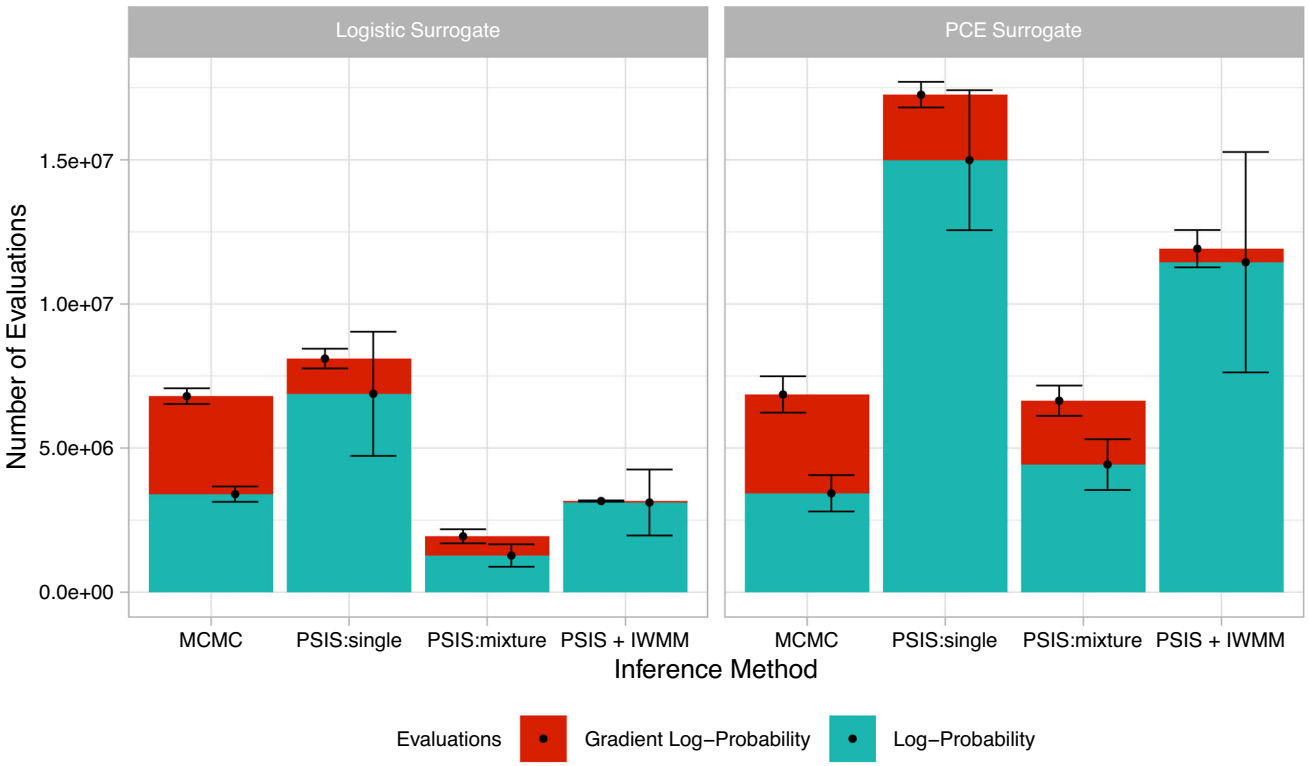


Fig. 11 Number of log-probability (blue) and gradient log-probability evaluations (red) required to approximate the posteriors of $m = 100$ surrogate parameter draws, for the logistic and the PCE surrogate (in the columns). Representative draws were selected using the *log-lik* method. Error bars indicate the standard deviation across repeated runs of the same setup

Table 2 Ratio of the mean number of log-probability and gradient log-probability evaluations for each inference method (columns), relative to the brute-force *MCMC* approach that runs HMC separately on all $m = 100$ surrogate parameter draws

	Logistic surrogate			PCE surrogate		
	PSIS:single	PSIS:mixture	PSIS+IWMM	PSIS:single	PSIS:mixture	PSIS+IWMM
log-probability	2.02	0.37	0.91	4.37	1.29	3.34
gradient log-prob	0.35	0.20	0.01	0.66	0.65	0.14

A ratio of 1 indicates equivalent computational effort; values below 1 indicate fewer evaluations and thus greater computational efficiency

For the logistic surrogate, the *PSIS:single* requires slightly more log-probability evaluations than the baseline *MCMC*, but significantly fewer gradient evaluations, which are typically more computationally expensive. The *PSIS:mixture* inference method, as well as *PSIS+IWMM*, succeed in approximating all posterior distributions with fewer evaluations of both log-probability and gradient evaluations. In contrast, the PCE surrogate poses a greater challenge, as the posterior distributions $p(\theta_I \mid \tau^{(i)}, D_I)$ differ more substantially. In the worst case—observable in *PSIS:single*—the iterative algorithm requires more log-probability evaluations than baseline *MCMC*. This occurs because the algorithm needs many iterations to evaluate all posterior distributions. During each iteration, importance ratios are computed for each of the remaining posteriors. If PSIS fails in many of these cases, a significant number of importance ratio calculations are only used to determine that PSIS does not work for this particular proposal-target combination, rather than approximating the target distribution. As a result, the number of log-probability evaluations exceeds the ones of running *MCMC* separately. In contrast, the number of gradient evaluations required by the new method cannot exceed those needed for running brute-force *MCMC*, as they are only necessary for calculating the proposal in the iterative algorithm with HMC, and there can be at most $m = 100$ different proposals. The *PSIS:mixture* method substantially improves PSIS performance compared to the single proposal with only having a few more log-probability evaluations than the *MCMC* method. The combined *PSIS+IWMM* approach again results in the fewest gradient evaluations, indicating that only a small number of posteriors needed to be computed using HMC. The high number of log-probability evaluations in this method arises from applying the IWMM algorithm (see Sect. 2.2) in each iteration of the iterative algorithm for multiple different target distributions.

In conclusion, the proposed method improves the efficiency of posterior approximation for computing the marginalized posterior via the Monte Carlo estimate. In the simpler setting—using the logistic surrogate, which incorporates the true simulator—the method can approximate all posterior distributions with only one or two expensive HMC runs. In the more complex scenario involving the PCE surrogate, the method faces greater challenges, but still achieves notable reductions in computational effort. The number of gradient log-probability evaluations can be reduced to less than 14% of those required by the brute-force *MCMC* approach, as shown in Table 2. Since gradient evaluations typically dominate the overall computational cost, our proposed method achieves improved efficiency even in more complex settings.

4 Conclusion

In this work, we introduced a diagnostic-driven and computationally efficient method for uncertainty propagation in Bayesian two-step modeling frameworks. Our approach combines Pareto Smoothed Importance Sampling (PSIS) and Importance Weighted Moment Matching (IWMM) to substantially reduce the number of costly sampling runs typically required for reliable uncertainty propagation. Rather than performing separate MCMC simulations for each of the multiple realizations of the uncertainty quantity, we run MCMC for only a small, carefully selected subset of representative draws and approximate the remaining posteriors via the two importance sampling techniques. We focus on two important application domains where uncertainty propagation is particularly critical: multiple imputation and surrogate modeling.

Through simulation studies in these two application settings, as well as a real-world case study, we found that the method closely approximates the HMC-based posterior summaries in many of the settings considered while yielding substantial computational savings. Specifically, it reduces the number of log-probability evaluations by up to 90% and the number of gradient evaluations by 85–99% compared to a brute-force HMC baseline. The accuracy of the resulting marginalized posteriors is maintained through the use of diagnostics that validate each approximation, providing a mechanism for identifying cases in which close agreement with the HMC reference is expected, and for flagging cases in which the approximation may be unreliable.

To support efficient uncertainty propagation, we also developed and evaluated several strategies for selecting representative values used to construct proposal distributions for PSIS and IWMM. In our simulation studies, the choice of selection strategy had only a limited impact on the performance metrics considered. Notably, even a simple

random selection performed comparably to more sophisticated approaches. This suggests that either the choice of representative values is not critical to the method's effectiveness, or that the current selection techniques are not sufficiently sensitive to differences in the posterior space structure. Consequently, extensive computational effort for optimizing representative value selection may not be warranted in settings similar to those studied here. Despite its good performance in many of the scenarios considered, the method showed limitations in more challenging settings. For example, in missing data problems characterized by complex posterior structure—such as those arising from shrinkage priors, high-dimensional parameter spaces, and a high proportion of missingness—the affine transformations used in IWMM are insufficient. In such cases, the discrepancy between the target and proposal distribution leads to degraded approximation quality. More expressive transformation techniques may be required, similar to those needed in other applications of IWMM, such as leave-one-group-out cross-validation (Roberts et al. 2017).

Although we demonstrated the method in two specific domains, the same computational idea may be applicable to other modeling scenarios with a two-step structure in which uncertainty propagation is essential. Bayesian cut models are one example of such a setting. Developing scalable and user-friendly implementations will be crucial for broader adoption. While the current implementation is mostly written in `R` (see GitHub), versions in a compiled language such as `C++`, or full integration with probabilistic programming frameworks like `Stan` could greatly enhance practical performance.

Choosing m, the number of first-step draws $\tau^{(i)}$, is inherently context-dependent and often difficult in practice. Even in the closely related importance-sampling literature, the choice of m has been debated for decades, underscoring that there is no universally optimal rule. Our approach alleviates this trade-off by making it feasible to use larger values of m and thus propagate first-step uncertainty more faithfully, without incurring a proportional increase in computational cost.

Overall, this work presents a flexible and computationally efficient framework for uncertainty propagation in Bayesian two-step procedures, providing a practical route to improved scalability while retaining good agreement with HMC-based reference results in the examples studied.

Appendix A Iterative algorithm for using a mixture proposal

Algorithm 2 Iterative Algorithm for Efficiently Calculating the Marginalized Posterior (Mixture Proposal)

Require: Variable of interest $\theta \sim p(\theta)$, underlying variable $\tau \sim p(\tau)$, mixture size I_{mix}

Draw m samples $\tau^{(1)}, \ldots, \tau^{(m)} \sim p(\tau)$
Set index set $I = \{1, \ldots, m\}$
while $|I| > |I_{\text{mix}}|$ **do**
  Select $|I_{\text{mix}}|$ representative values from $\{\tau^{(i)}\}_{i \in I}$
  Run MCMC to access their posterior distributions: $\{\theta_i^{(s)}\}_{s=1}^{S}$
  Build the mixture:

$$p_{\text{mix}}(\theta \mid \{\tau^{(i)}\}) = \frac{1}{|I_{\text{mix}}|} \sum_{i \in I_{\text{mix}}} p(\theta \mid \tau^{(i)})$$

  Obtain samples $\{\theta_*^{(s)}\}_{s=1}^{S} \sim p_{\text{mix}}(\theta \mid \{\tau^{(i)}\})$ via resampling
  Remove the $|I_{\text{mix}}|$ selected values from I
  for $i \in I$ **do**
    Run PSIS with target $p(\theta \mid \tau^{(i)})$ and proposal $p_{\text{mix}}(\theta \mid \{\tau^{(i)}\})$
    Obtain importance weights and diagnostic metric $\hat{k}_i$
    if $\hat{k}_i < 0.7$ **then**
      Use the importance weights and $\{\theta_*^{(s)}\}$ to resample posterior draws for $p(\theta \mid \tau^{(i)})$
      Remove i from I: $I \leftarrow I \setminus \{i\}$
    end if
  end for
end while
for $i \in I$ **do**
  Run MCMC to obtain posterior draws from $p(\theta \mid \tau^{(i)})$
end for
Compute marginalized posterior via Monte Carlo estimate: $p(\theta \mid C) \approx \frac{1}{m} \sum_{i=1}^{m} p(\theta \mid \tau^{(i)})$

Appendix B Neural variational inference for cut feedback in the missing data setting

The missing data example considered in this paper can also be interpreted from the perspective of modular Bayesian inference and cut feedback. In this formulation, the imputation model constitutes an upstream module, while the substantive regression model constitutes a downstream module. Let D_{obs} denote the observed incomplete dataset and let D_{mis} denote the missing entries in the dataset (see Sect. 2.4 for more details of the setup). The imputation procedure provides draws

$$D_{\text{mis}}^{(m)} \sim p_{\text{imp}}(D_{\text{mis}} \mid D_{\text{obs}}), \qquad m = 1, \ldots, M,$$

which induce complete datasets $D^{(m)} = (D_{\text{obs}}, D_{\text{mis}}^{(m)})$. In the notation of cut models, the missing values D_{mis} play the role of the upstream quantity, while the regression parameters θ are the downstream parameters of interest.

The corresponding cut posterior can be written as

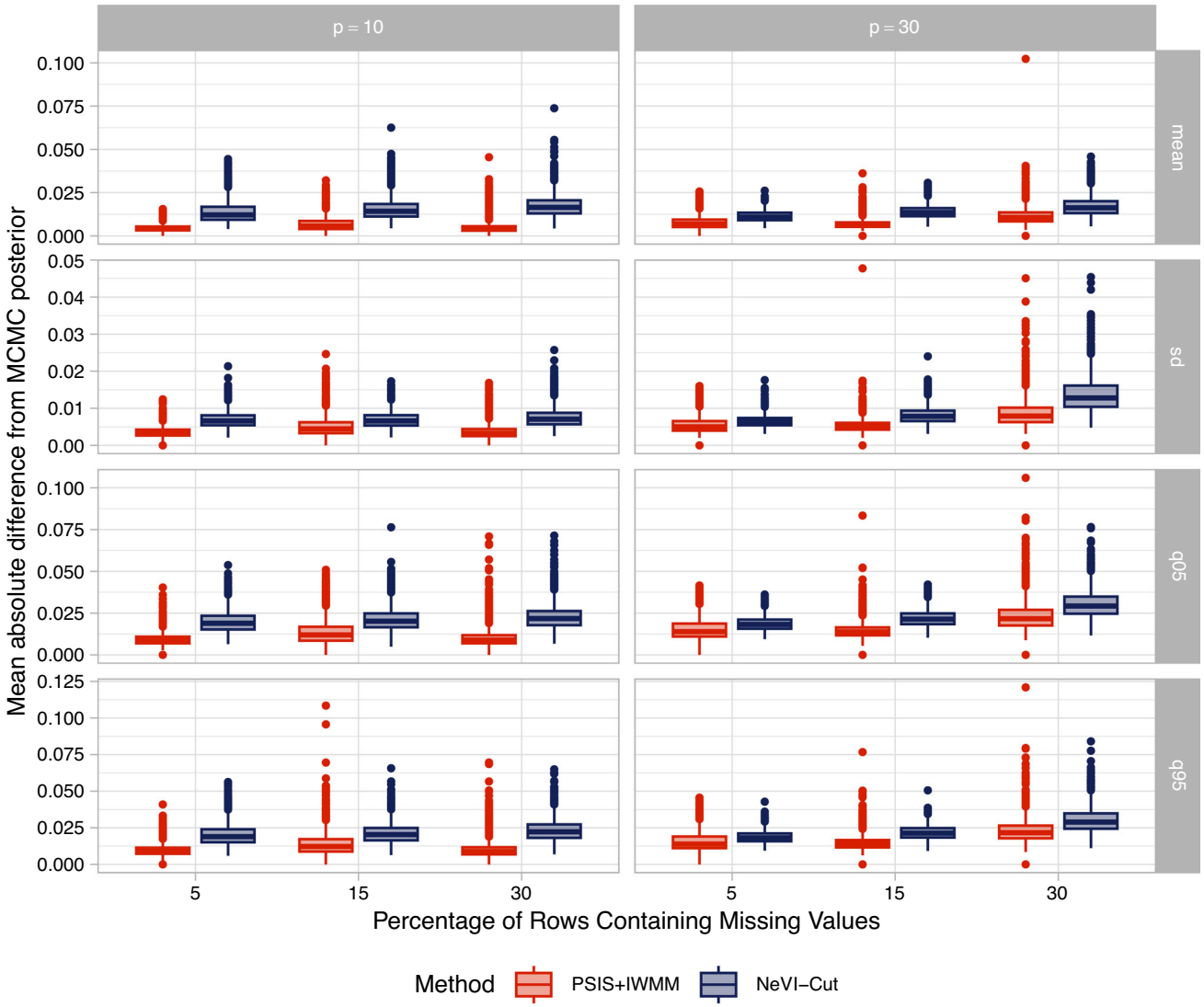


Fig. 12 Mean absolute differences between posterior summaries obtained from *MCMC* and from the *PSIS+IWMM* and *NeVI-Cut* resampled posterior for datasets of size $N = 100$. For each missingness level (5%, 15%, and 30% of rows containing missing values), boxplots summarize the distribution across the 20 datasets and their 100 imputed datasets; within each dataset, differences are averaged over the p parameters. Rows indicate the summary metric (posterior mean, standard deviation, 5% quantile, and 95% quantile). Results are shown for $p = 10$ (left) and $p = 30$ (right)

$$p_{\text{cut}}(D_{\text{mis}}, \theta \mid D_{\text{obs}}) = p_{\text{imp}}(D_{\text{mis}} \mid D_{\text{obs}})\, p(\theta \mid D_{\text{obs}}, D_{\text{mis}}).$$

The first factor describes uncertainty about the missing values as represented by the imputation model. The second factor is the posterior distribution of the regression parameters conditional on one completed dataset. The cut indicates that inference in the downstream regression model is not allowed to update the imputation distribution. Thus, uncertainty is propagated from the imputation model to the regression model, but feedback from the regression model to the imputation model is prevented.

This interpretation connects the missing data problem to the neural variational cut-Bayes approach of Song et al. (2025). In their framework, samples from the upstream distribution are used directly, and a conditional variational approximation is learned for the downstream posterior. Applied to the present missing data setting, the upstream samples are the imputed missing values $\eta^{(m)} = D_{\text{mis}}^{(m)}$, and the downstream target is $p(\theta \mid D_{\text{obs}}, \eta^{(m)})$. Instead of fitting the downstream model separately for every imputed dataset, one can train a conditional variational approximation

$$q_\lambda(\theta \mid \eta) \approx p(\theta \mid D_{\text{obs}}, \eta),$$

where q_λ is parameterized by a flexible neural density estimator, such as a normalizing flow.

The NeVI-Cut (Neural Variational Inference for Cut-Bayes, Song et al. 2025) formulation is applicable in this setting because the multiple imputation procedure already provides samples from an upstream uncertainty distribution, and the downstream regression model can be evaluated conditionally on each completed dataset. However, its use also changes the nature of the approximation. The quality of the result depends on the expressiveness of the variational family and on the success of the optimization procedure. Moreover, for a fair comparison with MCMC-based approaches, the same downstream likelihood and priors should be used.

To compare the performance of the NeVI-Cut method by Song et al. (2025) to our proposed method, we use the Gaussian regression model from the simulation study in Sect. 3.1. The downstream parameter vector is $\theta = (\beta_0, \beta_1, \ldots, \beta_p, \log\sigma)$, and the likelihood conditional on a completed dataset is $y_i \mid x_i, \theta \sim \mathcal{N}\left(\beta_0 + x_i^\top \beta, \sigma^2\right), i = 1, \ldots, n$. The variational approximation is trained using the imputed values $\{\eta^{(m)}\}_{m=1}^M$ as conditional variables. After training, conditional posterior draws can be generated for each imputation,

$$\theta^{(s,m)} \sim q_\lambda(\theta \mid \eta^{(m)}), \qquad s = 1, \ldots, S, \quad m = 1, \ldots, M.$$

These draws provide a variational approximation to the same collection of imputation-specific posterior distributions that are approximated by the iterative method introduced in Sect. 2.3. To ensure comparability, we use the same imputed datasets for both approaches and compare the posterior samples obtained from NeVI-Cut with the MCMC baseline using the same evaluation criteria as for the iterative method.

As in Sect. 3.1, we compare posterior means, standard deviations, and the 5% and 95% quantiles in terms of mean absolute differences from the corresponding MCMC posterior summaries. Figure 12 shows the results for datasets of size $N = 100$ under the standard prior specification. Across all four posterior summary statistics, the mean absolute differences from the MCMC baseline are, on average, smaller for the iterative method based on PSIS+IWMM than for NeVI-Cut. This can be seen from the lower positions of the boxes and medians in the corresponding boxplots.

A comparison based on the number of log-probability and gradient evaluations is less straightforward for NeVI-Cut than for MCMC or PSIS+IWMM. NeVI-Cut is trained by solving an optimization problem in which the variational objective is evaluated repeatedly. During backpropagation, gradients are computed with respect to the neural-network or normalizing-flow parameters λ, rather than with respect to the downstream model parameters θ. These gradients therefore correspond to a different computational object than the gradients used in HMC. For this reason, gradient-evaluation counts for NeVI-Cut are not directly comparable to the gradient-evaluation counts reported for HMC-based posterior simulation.

This experiment illustrates that methods developed for cut-feedback problems can, in principle, be applied to the two-step uncertainty propagation problems considered in this paper. In the missing-data setting studied here, however, the proposed iterative method provides posterior samples that are closer to the MCMC baseline than those obtained from the NeVI-Cut implementation of Song et al. (2025), according to the posterior-summary discrepancies considered above.

Appendix C Additional results of the simulation study for the multiple imputation problem

See Figs. 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 and 25 and Tables 3, 4, 5, and 6.

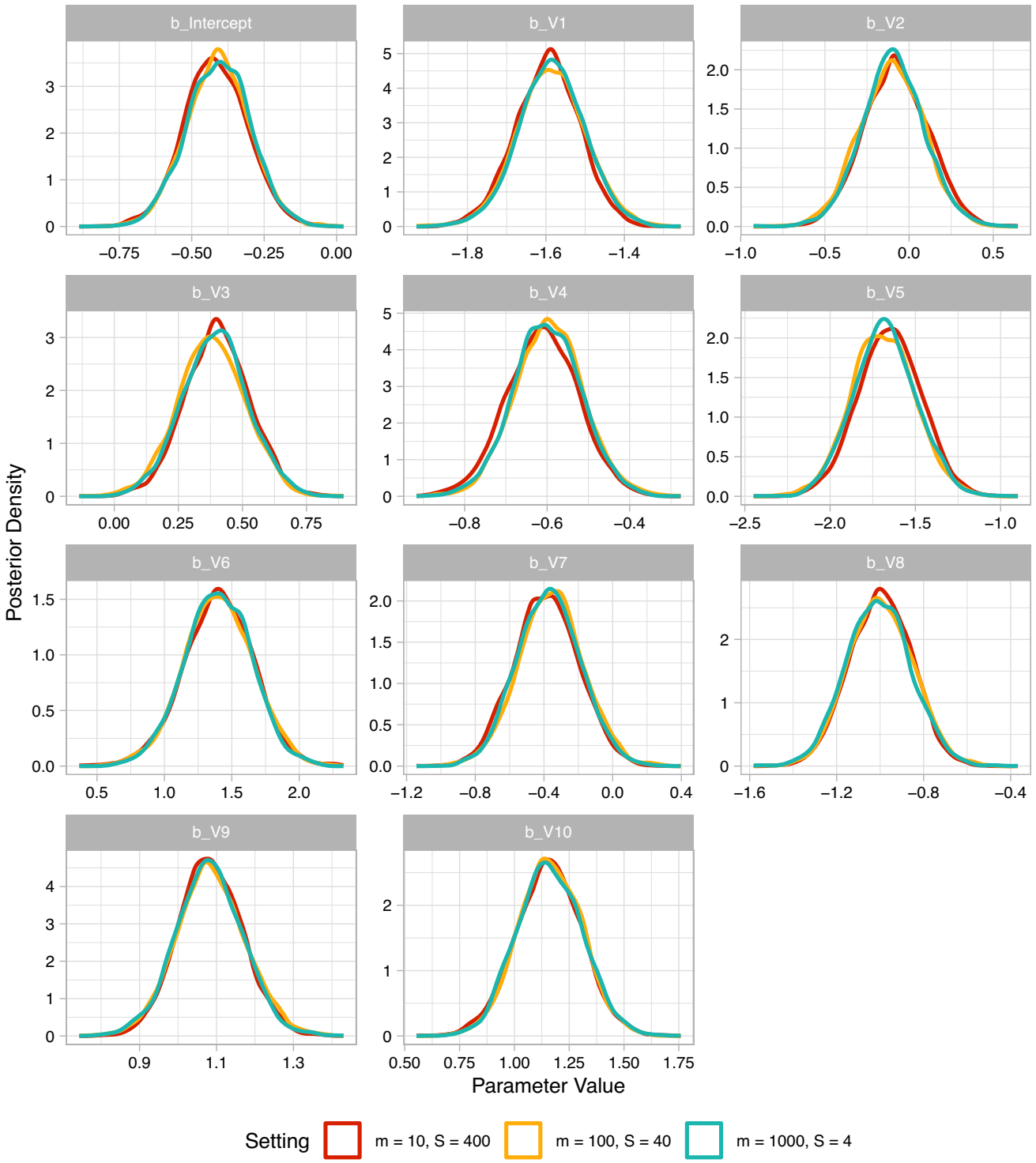


Fig. 13 Posterior density estimates (MCMC) for the regression coefficients ($N = 100$, $p = 10$) under three allocations of a fixed total posterior sample size ($m \cdot S = 4000$) across multiple imputations: $m = 10$, $S = 400$, $m = 100$, $S = 40$, and $m = 1000$, $S = 4$. Each panel corresponds to one parameter ($b_{\text{Intercept}}, b_{V1}, \ldots, b_{V10}$). The near-complete overlap of the curves indicates that, in this example, the inferred posterior is largely insensitive to the specific choice of m and S when the total number of draws is held constant

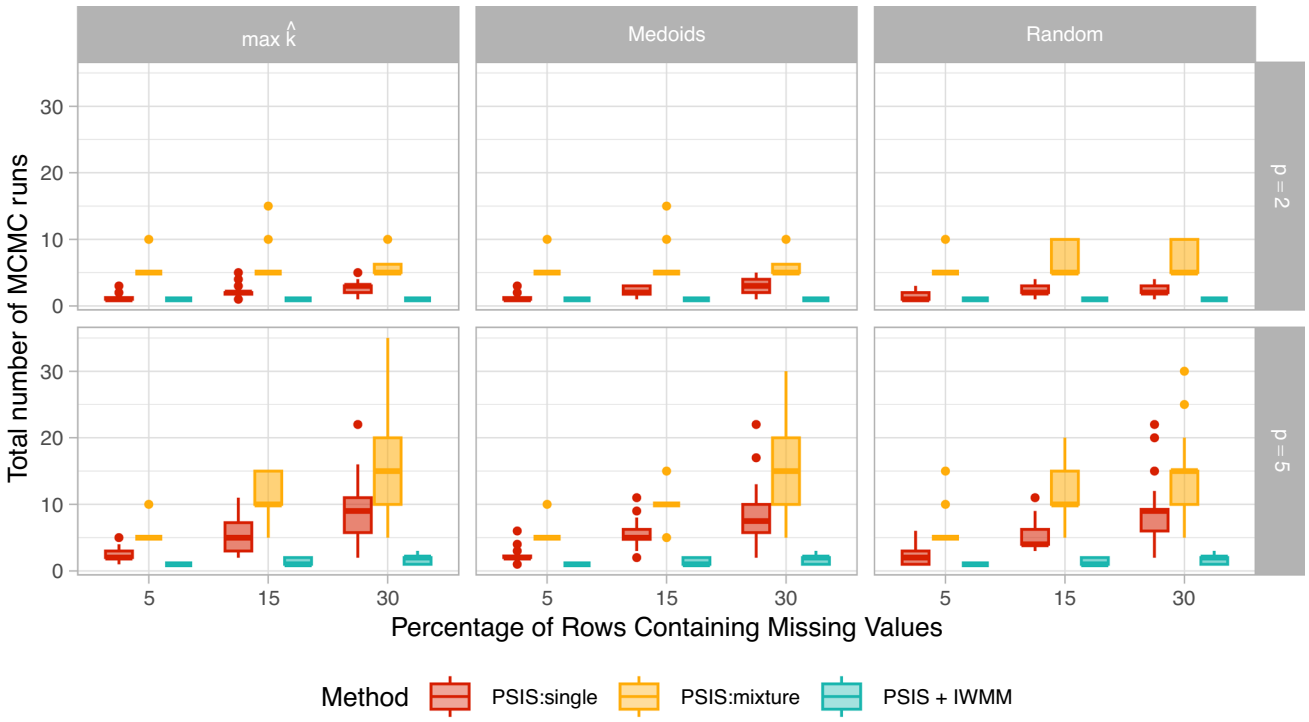


Fig. 14 Number of MCMC runs necessary to approximate the posteriors of $m = 100$ imputed datasets with size $N = 10$ and number of parameters p (in the rows), using standard priors. Different selection methods are employed (in the columns)

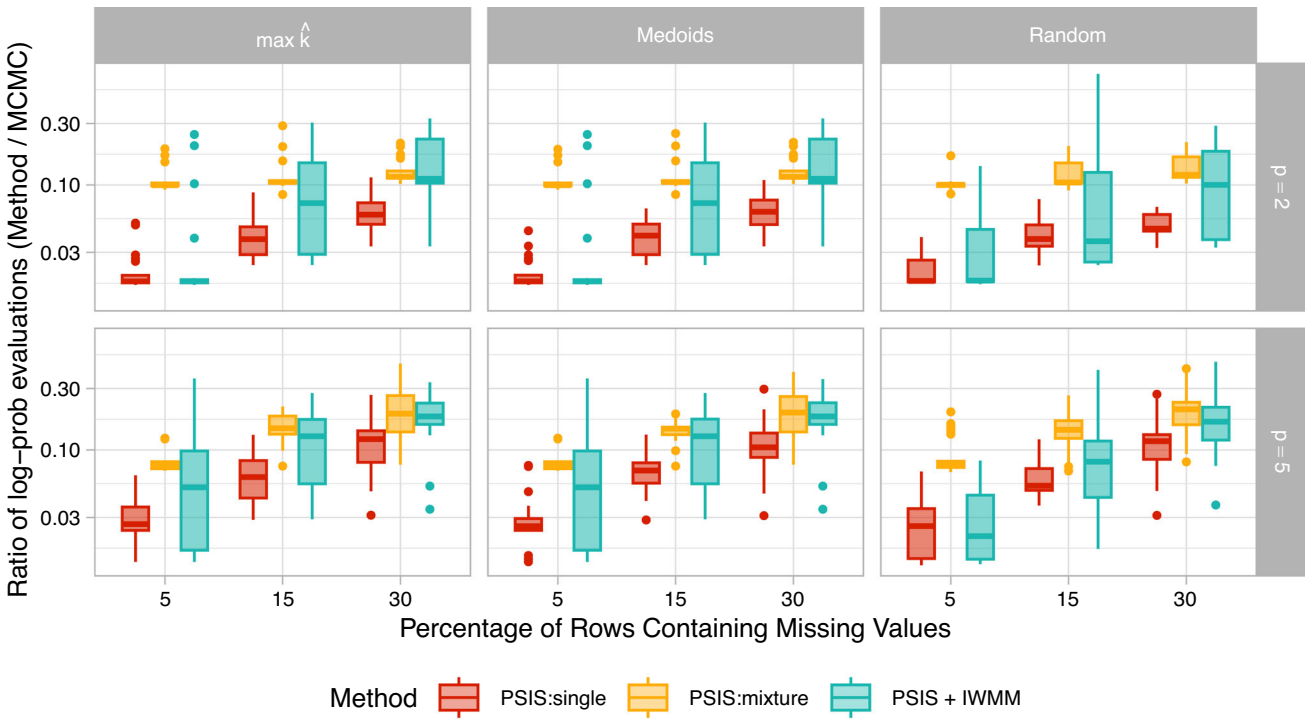


Fig. 15 Ratio of number of evaluations of the log-probability compared to running MCMC separately necessary to approximate the posteriors of $m = 100$ imputed datasets with size $N = 10$ and number of parameters p (in the rows), using standard priors. Different selection methods are employed (in the columns)

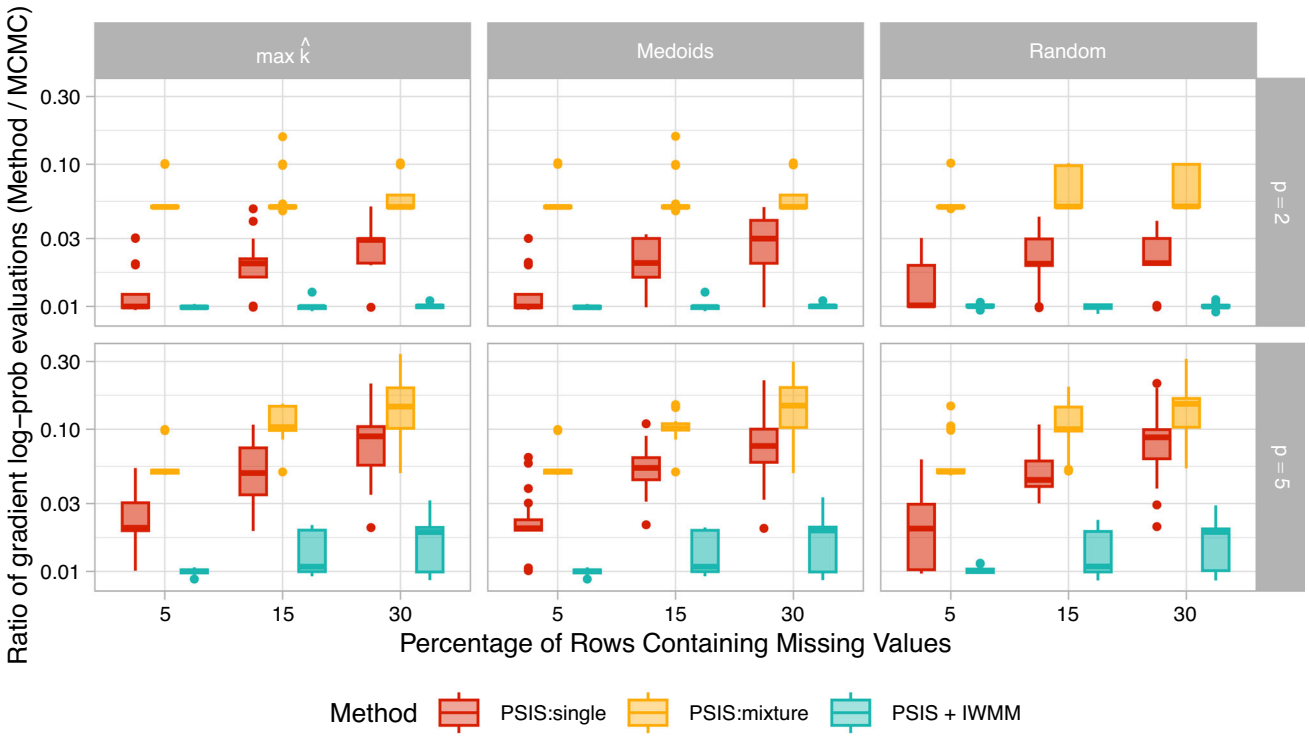


Fig. 16 Ratio of number of evaluations of the gradient log-probability compared to running MCMC separately necessary to approximate the posteriors of $m = 100$ imputed datasets with size $N = 10$ and number of parameters p (in the rows), using standard priors. Different selection methods are employed (in the columns)

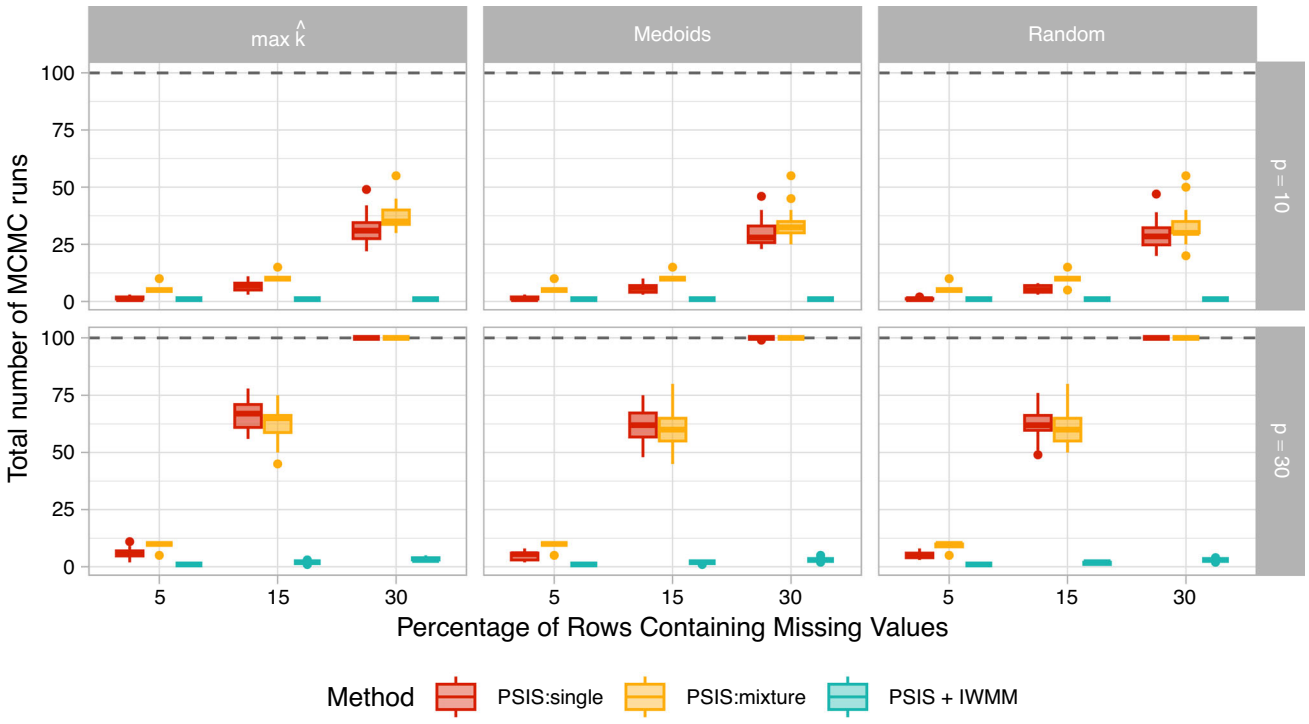


Fig. 17 Number of MCMC runs necessary to approximate the posteriors of $m = 100$ imputed datasets with size $N = 100$ and number of parameters p (in the rows), using standard priors. Different selection methods are employed (in the columns)

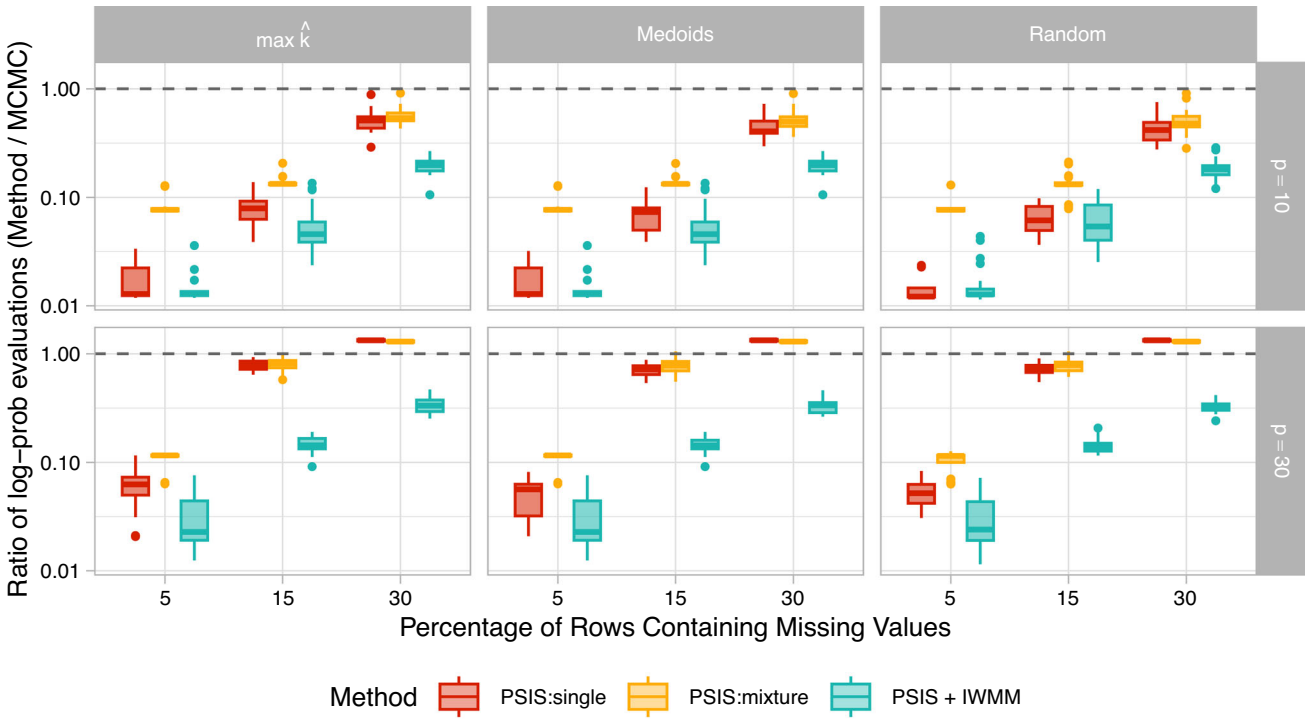


Fig. 18 Ratio of number of evaluations of the log-probability compared to running MCMC separately necessary to approximate the posteriors of $m = 100$ imputed datasets with size $N = 100$ and number of parameters p (in the rows), using standard priors. Different selection methods are employed (in the columns)

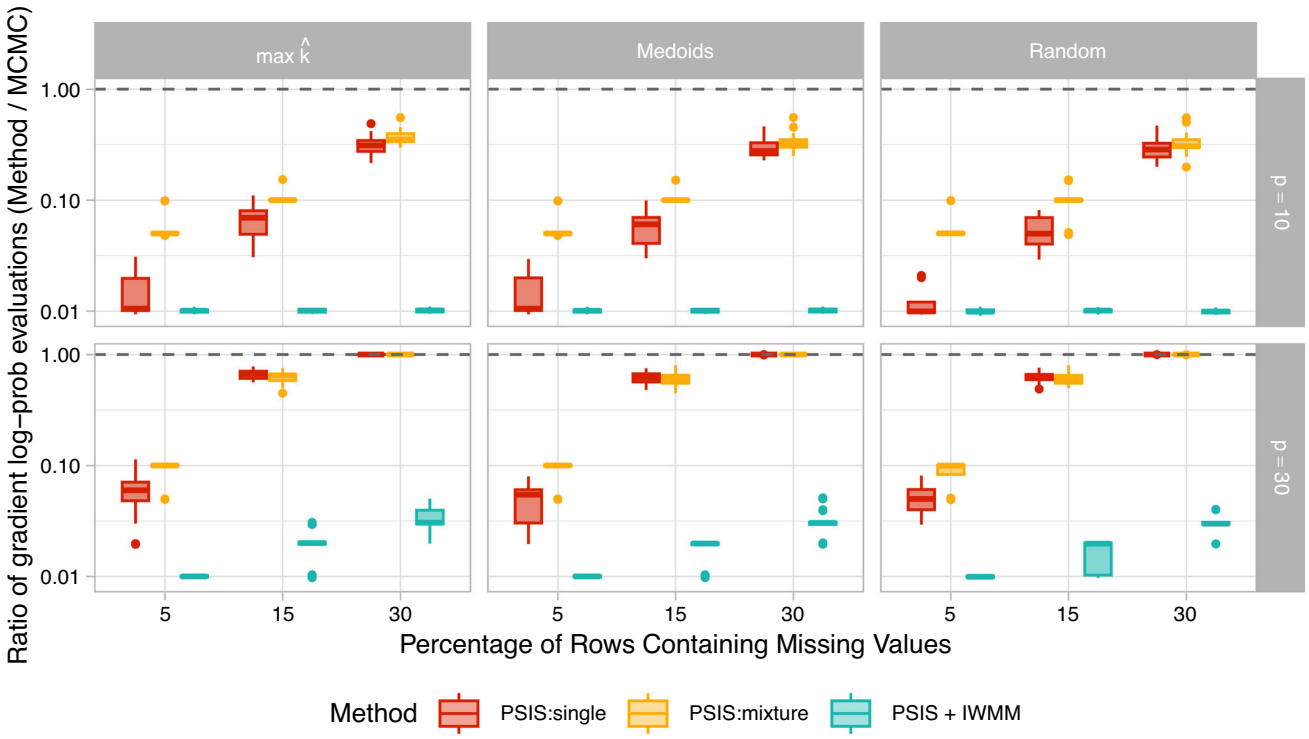


Fig. 19 Ratio of number of evaluations of the gradient log-probability compared to running MCMC separately necessary to approximate the posteriors of $m = 100$ imputed datasets with size $N = 100$ and number of parameters p (in the rows), using standard priors. Different selection methods are employed (in the columns)

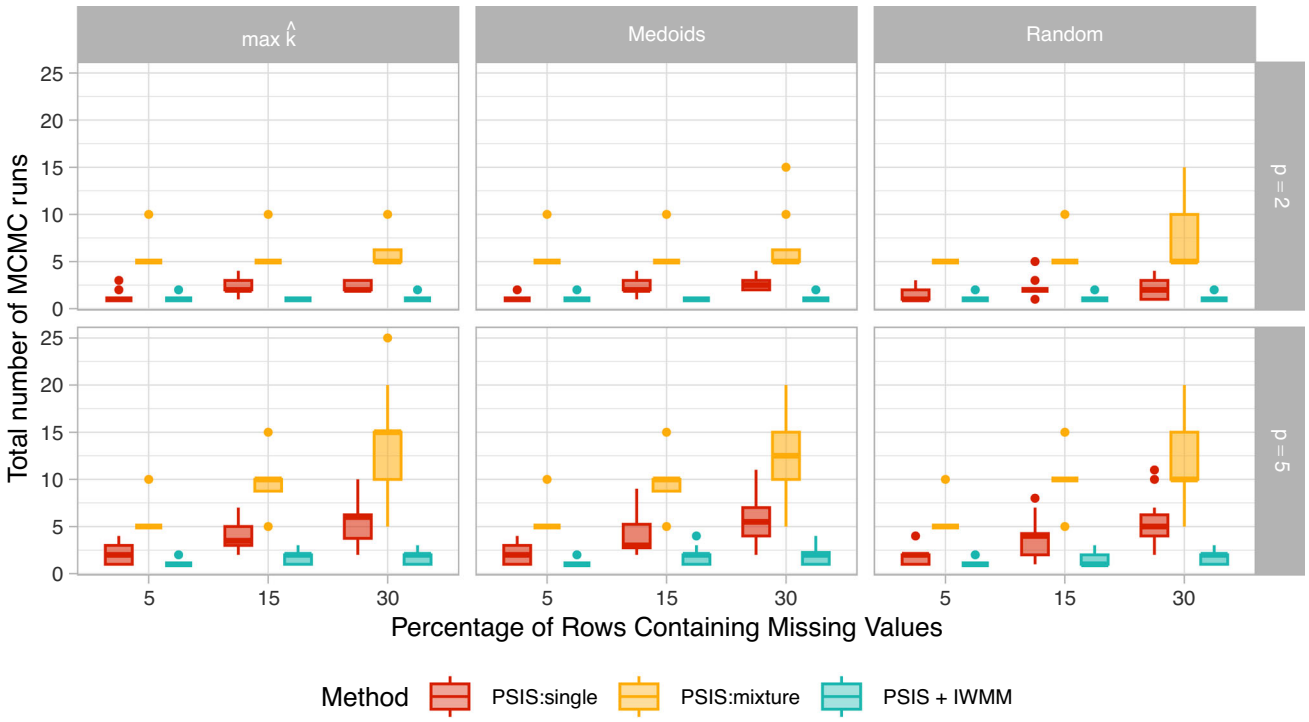


Fig. 20 Number of MCMC runs necessary to approximate the posteriors of $m = 100$ imputed datasets with size $N = 10$ and number of parameters p (in the rows), using horseshoe priors. Different selection methods are employed (in the columns)

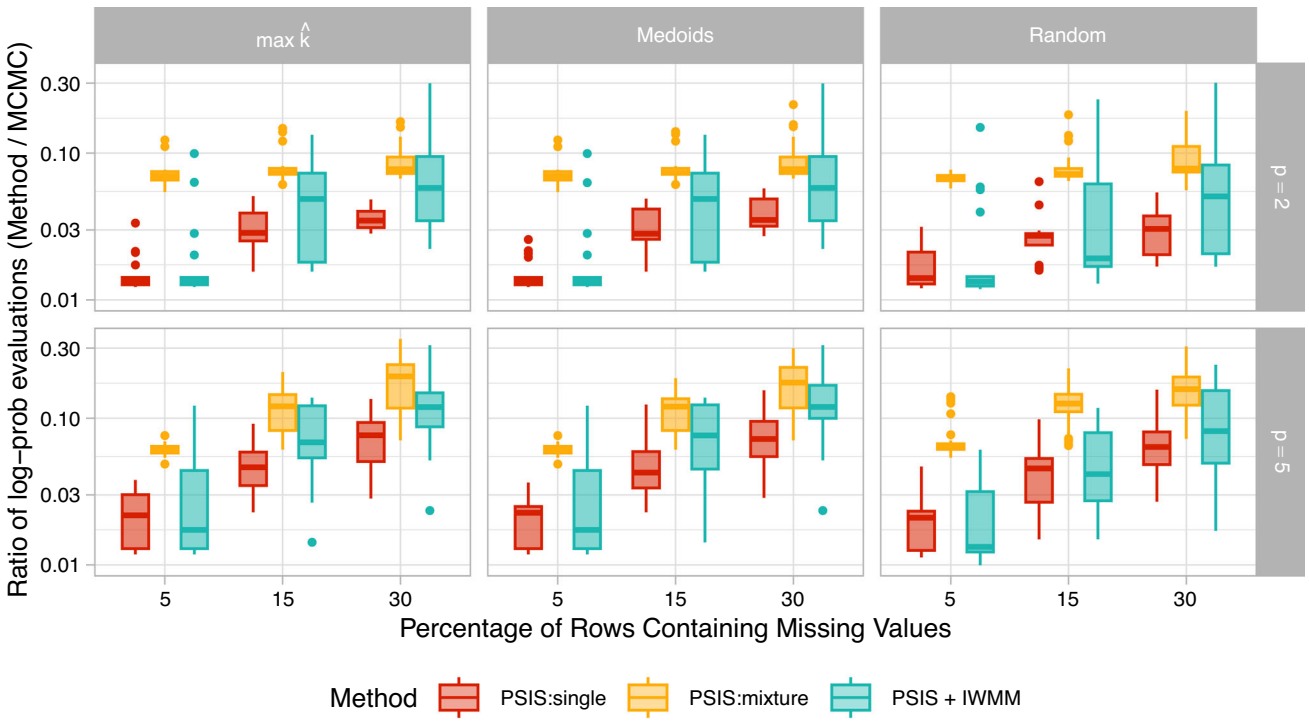


Fig. 21 Ratio of number of evaluations of the log-probability compared to running MCMC separately necessary to approximate the posteriors of $m = 100$ imputed datasets with size $N = 10$ and number of parameters p (in the rows), using horseshoe priors. Different selection methods are employed (in the columns)

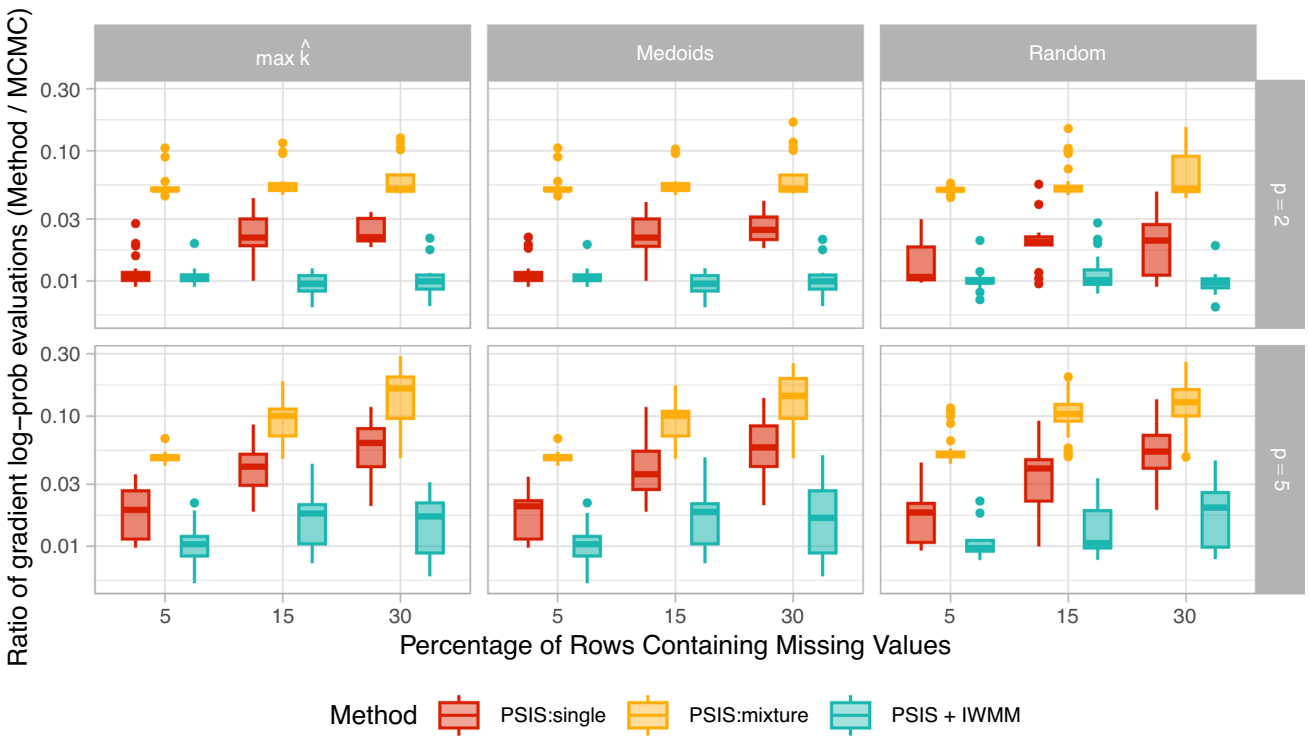


Fig. 22 Ratio of number of evaluations of the gradient log-probability compared to running MCMC separately necessary to approximate the posteriors of $m = 100$ imputed datasets with size $N = 10$ and number of parameters p (in the rows), using horseshoe priors. Different selection methods are employed (in the columns)

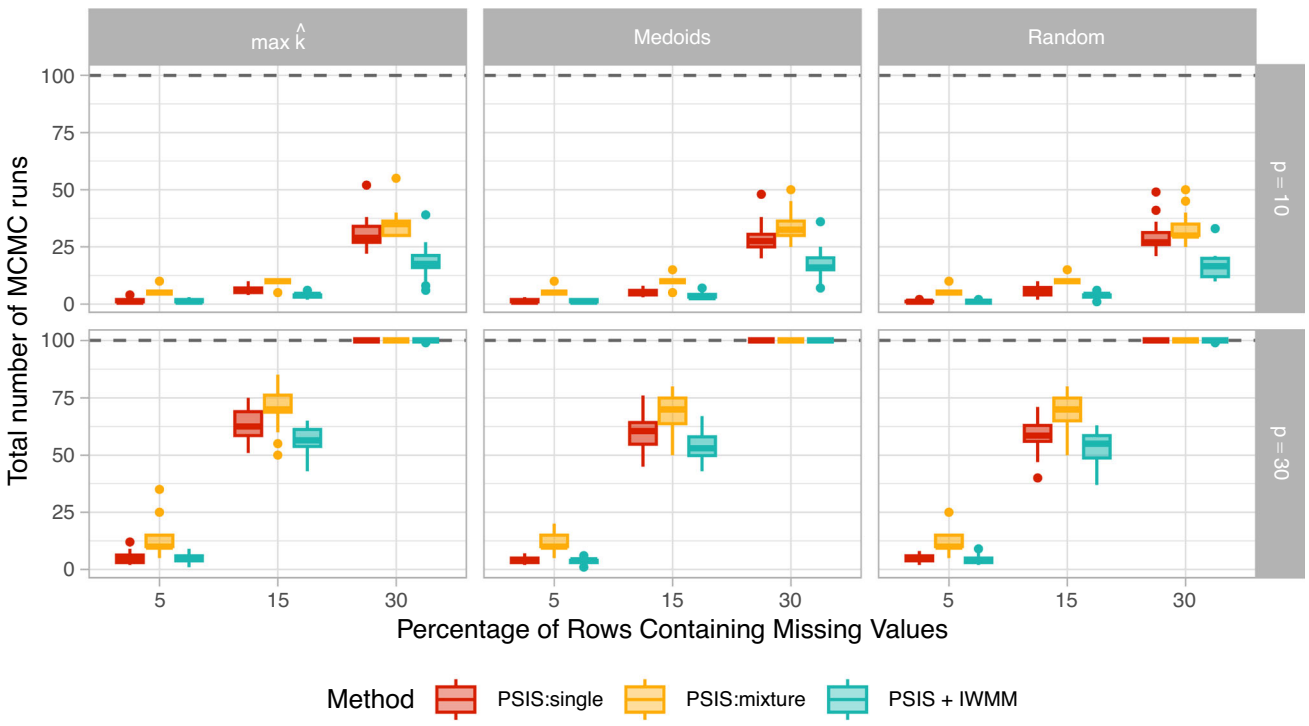


Fig. 23 Number of MCMC runs necessary to approximate the posteriors of $m = 100$ imputed datasets with size $N = 100$ and number of parameters p (in the rows), using horseshoe priors. Different selection methods are employed (in the columns)

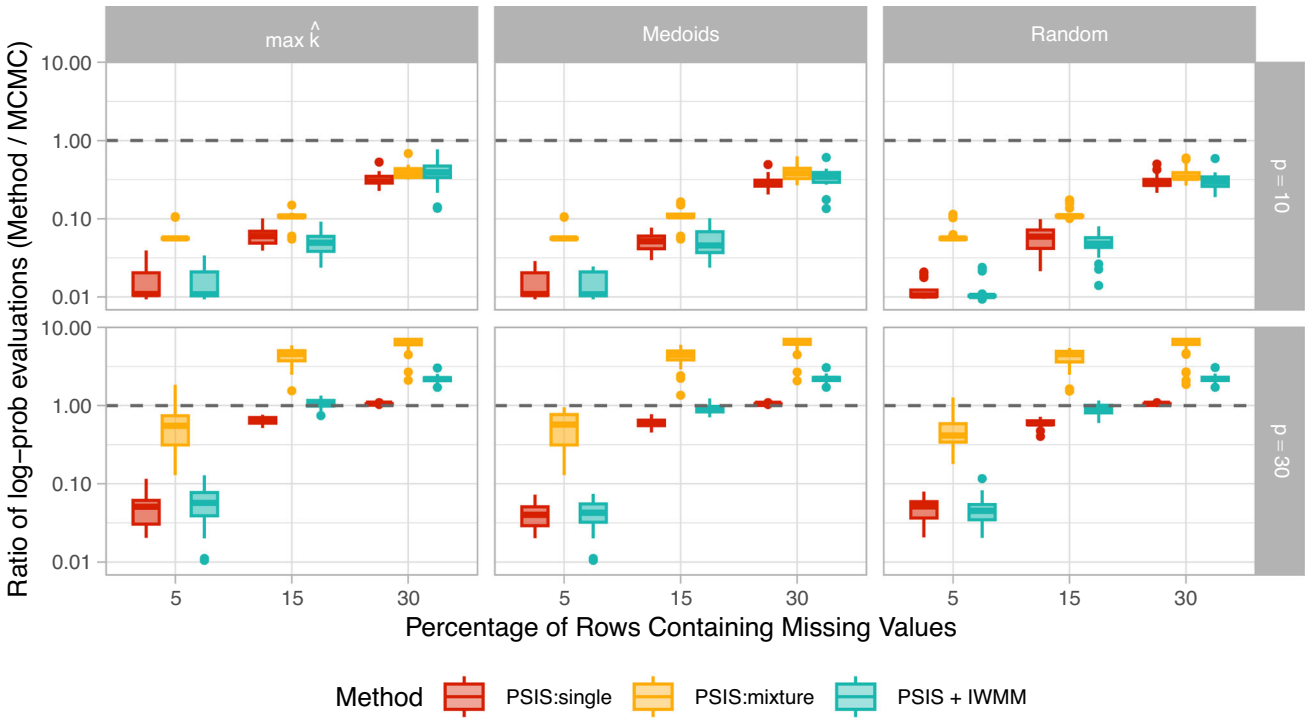


Fig. 24 Ratio of number of evaluations of the log-probability compared to running MCMC separately necessary to approximate the posteriors of $m = 100$ imputed datasets with size $N = 100$ and number of parameters p (in the rows), using horseshoe priors. Different selection methods are employed (in the columns)

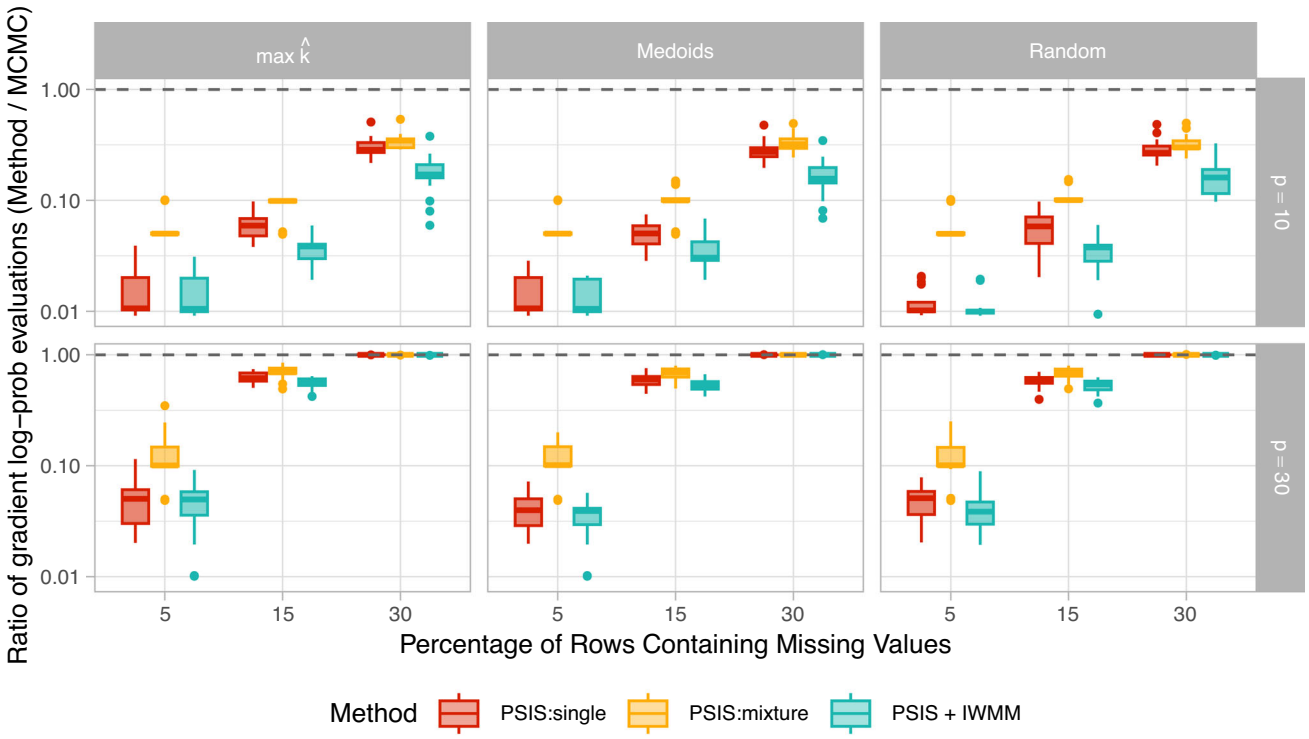


Fig. 25 Ratio of number of evaluations of the gradient log-probability compared to running MCMC separately necessary to approximate the posteriors of $m = 100$ imputed datasets with size $N = 100$ and number of parameters p (in the rows), using horseshoe priors. Different selection methods are employed (in the columns)

Table 3 Differences between the individual posterior distributions obtained via MCMC and via the iterative algorithm under the standard prior

N	p	% missing rows	Mean	Standard deviation	5% Quantile	95% Quantile	% MMD rejection
10	2	5	0.0188	0.0177	0.0451	0.0427	0.45
		15	0.0241	0.0239	0.0582	0.0578	4.35
		30	0.0227	0.0218	0.0532	0.0517	3.10
	5	5	0.0432	0.0669	0.1297	0.1213	11.90
		15	0.0671	0.0828	0.1520	0.1622	15.30
		30	0.0624	0.0719	0.1414	0.1465	14.65
100	10	5	0.0045	0.0036	0.0095	0.0097	0.30
		15	0.0069	0.0051	0.0137	0.0141	6.00
		30	0.0051	0.0038	0.0106	0.0104	4.45
	30	5	0.0077	0.0054	0.0155	0.0157	3.75
		15	0.0069	0.0053	0.0145	0.0146	1.90
		30	0.0114	0.0000	0.0229	0.0227	10.45

Values are the mean absolute differences in the corresponding metric, averaged over 20 datasets, with $m = 100$ imputations per dataset. For each dataset, the two sets of posterior samples are compared using the kernel maximum mean discrepancy (MMD) test at significance level $\alpha = 0.05$. The far-right column reports the percentage of datasets in which the null hypothesis (that the two sample sets are drawn from the same distribution) is rejected

Table 4 Differences between the individual posterior distributions obtained via MCMC and via the iterative algorithm under the horseshoe prior

N	p	% missing rows	Mean	Standard deviation	5% Quantile	95% Quantile	% MMD rejection
10	2	5	0.0156	0.0140	0.0365	0.0331	0.45
		15	0.0312	0.0181	0.0482	0.0527	5.20
		30	0.0249	0.0167	0.0454	0.0464	6.70
	5	5	0.0355	0.0249	0.0651	0.0554	13.35
		15	0.0493	0.0333	0.0818	0.0875	23.45
		30	0.0575	0.0365	0.0946	0.0900	27.90
100	10	5	0.0044	0.0033	0.0094	0.0095	0.65
		15	0.0086	0.0059	0.0171	0.0164	12.10
		30	0.0107	0.0066	0.0186	0.0190	38.40
	30	5	0.0088	0.0060	0.0175	0.0176	8.45
		15	0.0086	0.0048	0.0147	0.0149	36.25
		30	0.0001	0.0000	0.0001	0.0001	0.15

Values are the mean absolute differences in the corresponding metric, averaged over 20 datasets, with $m = 100$ imputations per dataset. For each dataset, the two sets of posterior samples are compared using the kernel maximum mean discrepancy (MMD) test at significance level $\alpha = 0.05$. The far-right column reports the percentage of datasets in which the null hypothesis (that the two sample sets are drawn from the same distribution) is rejected

Table 5 Differences between the final posterior distributions obtained via MCMC and via the iterative algorithm under the standard prior

N	p	% missing rows	Mean	Standard deviation	5% Quantile	95% Quantile	% MMD rejection
10	2	5	0.0141	0.0127	0.0357	0.0299	0
		15	0.0155	0.0166	0.0370	0.0394	0
		30	0.0127	0.0158	0.0309	0.0335	0
	5	5	0.0329	0.0554	0.1033	0.0879	0
		15	0.0409	0.0595	0.0868	0.1082	0
		30	0.0357	0.0440	0.0763	0.0792	0
100	10	5	0.0029	0.0025	0.0061	0.0066	0
		15	0.0034	0.0031	0.0070	0.0076	0
		30	0.0027	0.0021	0.0049	0.0048	0
	30	5	0.0042	0.0032	0.0083	0.0083	0
		15	0.0034	0.0030	0.0066	0.0067	0
		30	0.0043	0.0045	0.0088	0.0088	0

Values are the mean absolute differences in the corresponding metric, averaged over 20 datasets. For each dataset, the two sets of posterior samples are compared using the kernel maximum mean discrepancy (MMD) test at significance level $\alpha = 0.05$. The far-right column reports the percentage of datasets in which the null hypothesis (that the two sample sets are drawn from the same distribution) is rejected

Table 6 Differences between the final posterior distributions obtained via MCMC and via the iterative algorithm under the horseshoe prior

N	p	% missing rows	Mean	Standard deviation	5% Quantile	95% Quantile	% MMD rejection
10	2	5	0.0109	0.0100	0.0273	0.0223	0
		15	0.0240	0.0119	0.0317	0.0360	0
		30	0.0183	0.0095	0.0328	0.0274	5
	5	5	0.0281	0.0169	0.0466	0.0348	5
		15	0.0339	0.0210	0.0490	0.0529	0
		30	0.0386	0.0256	0.0557	0.0592	10
100	10	5	0.0028	0.0023	0.0062	0.0062	0
		15	0.0039	0.0030	0.0082	0.0078	0
		30	0.0026	0.0023	0.0052	0.0047	0
	30	5	0.0042	0.0030	0.0085	0.0085	0
		15	0.0017	0.0013	0.0033	0.0031	0
		30	0.0001	0.0000	0.0001	0.0001	0

Values are the mean absolute differences in the corresponding metric, averaged over 20 datasets. For each dataset, the two sets of posterior samples are compared using the kernel maximum mean discrepancy (MMD) test at significance level ($\alpha = 0.05$). The far-right column reports the percentage of datasets in which the null hypothesis (that the two sample sets are drawn from the same distribution) is rejected

Appendix D Additional results of the simulation study for the surrogate modeling problem

See Figs. 26, 27, 28 and 29.

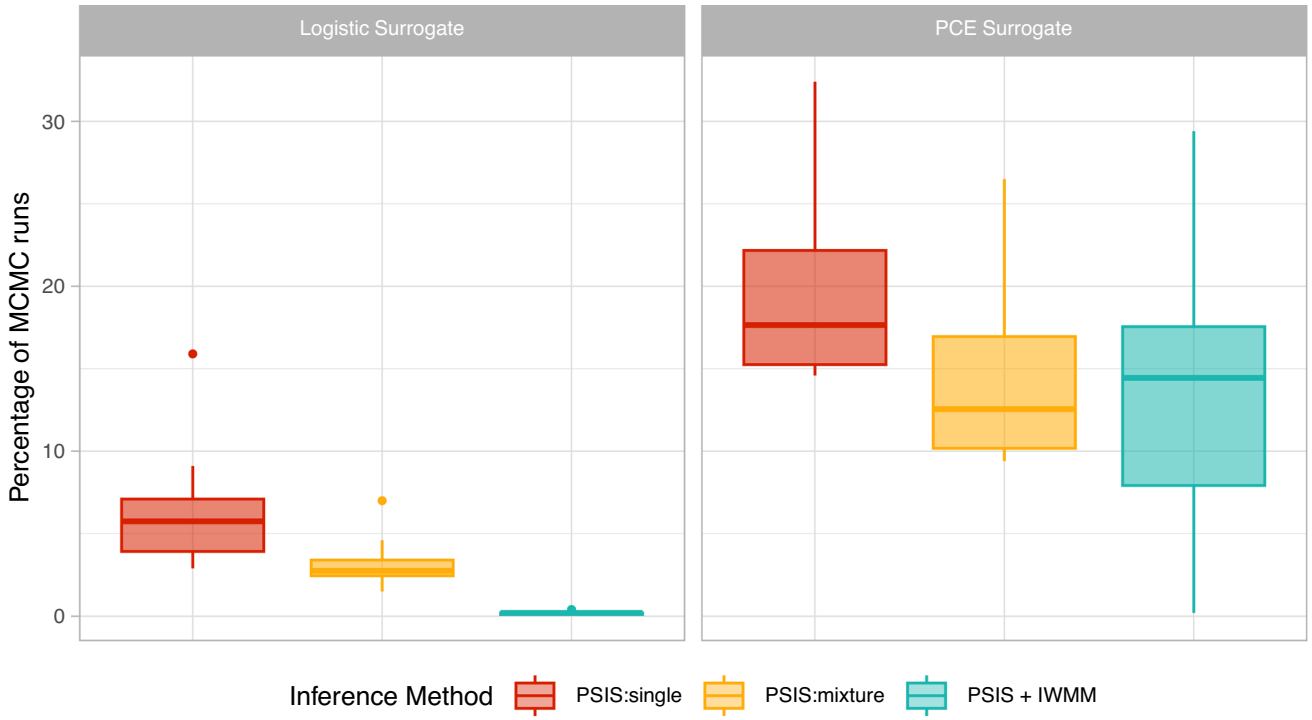


Fig. 26 Percentage of MCMC runs required to approximate the posteriors of $m = 1000$ surrogate parameter draws, for the logistic and PCE surrogate (in columns)

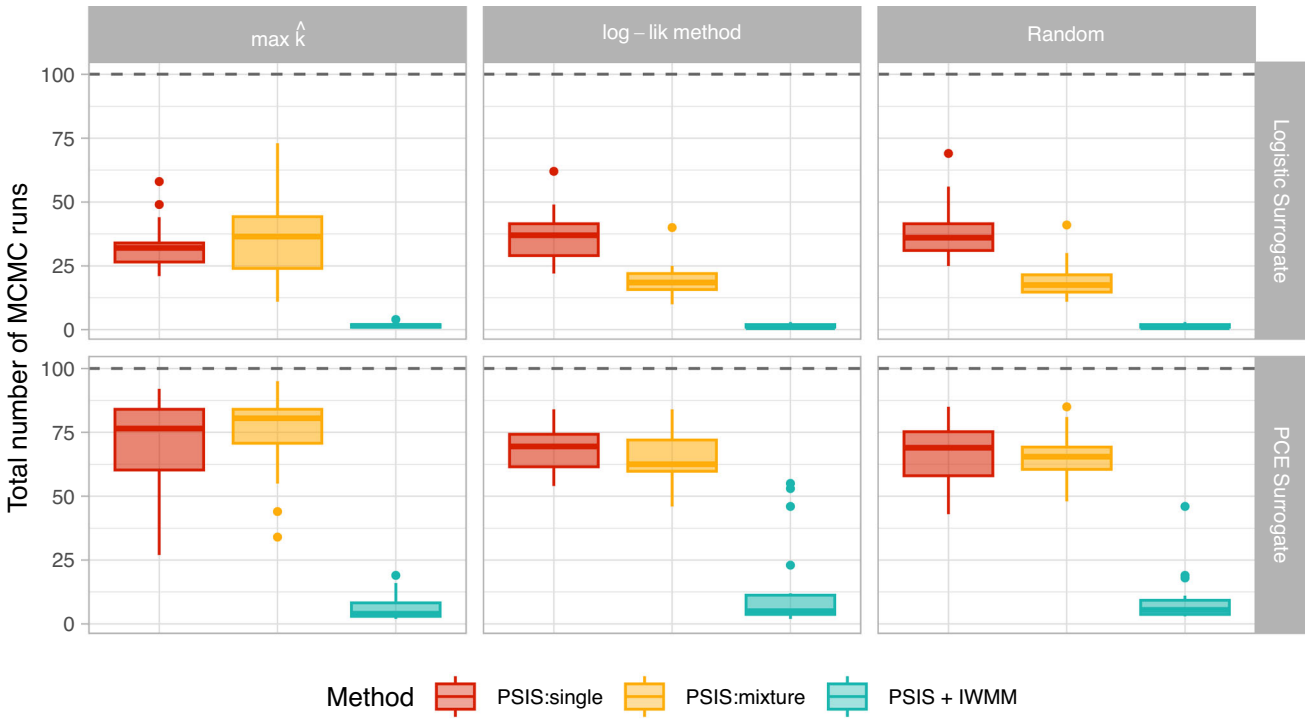


Fig. 27 Number of MCMC runs required to approximate the posteriors of $m = 100$ surrogate parameter draws, for the logistic and PCE surrogate (in rows) and different selection methods (in columns)

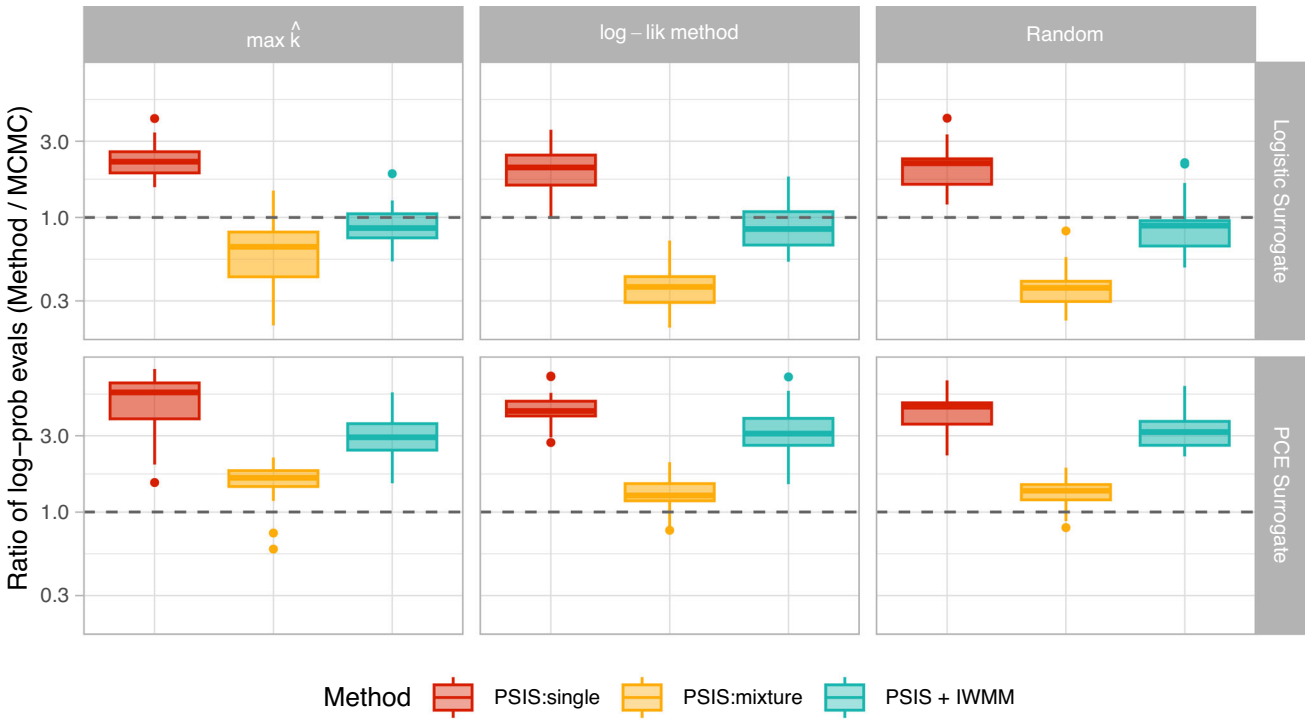


Fig. 28 Ratio of number of evaluations of the log-probability (on a log-scale) compared to running MCMC separately necessary to approximate the posteriors of $m = 100$ surrogate parameter draws, for the logistic and PCE surrogate (in rows) and different selection methods (in columns)

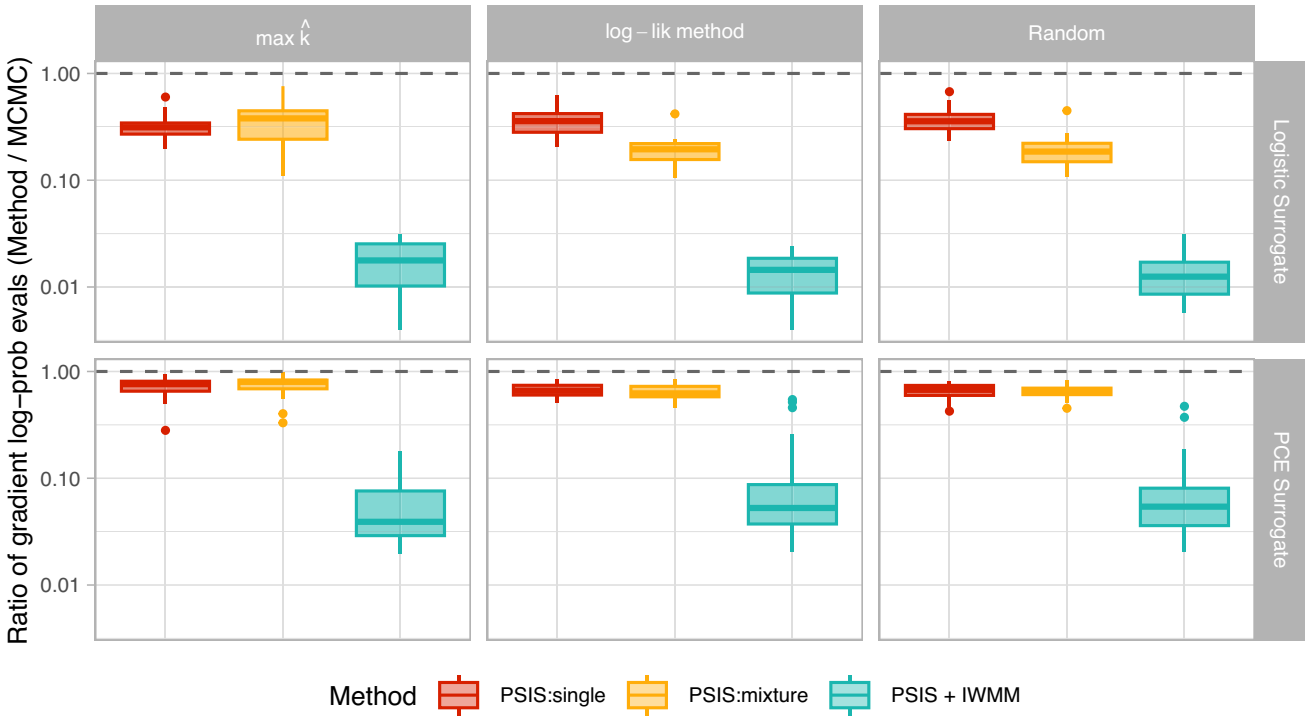


Fig. 29 Ratio of number of evaluations of the gradient log-probability (on a log-scale) compared to running MCMC separately necessary to approximate the posteriors of $m = 100$ surrogate parameter draws, for the logistic and PCE surrogate (in rows) and different selection methods (in columns)

Author contributions P.B. conceived the original idea. S.J. and P.B. developed the methodology. S.J. designed and conducted the experiments and case studies, wrote the code, and implemented the majority of the software. H.K. and A.M. provided the real-world dataset for the case study, and H.K. prepared the corresponding data. S.J. wrote the manuscript. All authors reviewed and edited the manuscript.

Funding Open Access funding enabled and organized by Projekt DEAL. This research was supported by the German Research Foundation (DFG) via Collaborative Research Center 391 "Spatio-Temporal Statistics for the Transition of Energy and Transport"– 520388526.

Data availability The code corresponding to this manuscript is openly available in a GitHub repository at https://github.com/sjedhoff/efficient-2step-uncertainty-paper.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

Alden, K., Cosgrove, J., Coles, M., et al.: Using emulation to engineer and understand simulations of biological systems. IEEE/ACM Trans. Comput. Biol. Bioinf. **17**(1), 302–315 (2020). https://doi.org/10.1109/TCBB.2018.2843339

Bayarri, M.J., Berger, J.O., Liu, F.: Modularization in Bayesian analysis, with emphasis on analysis of computer models. Bayesian Anal. **4**(1), 119–150 (2009).https://doi.org/10.1214/09-BA404

Bishop, C.M.: Pattern recognition and machine learning. Information science and statistics, Springer, New York (2006)

Brooks, S., Gelman, A., Jones, G. (eds.). Handbook of Markov Chain Monte Carlo. Chapman and Hall/CRC, New York, (2011). https://doi.org/10.1201/b10905

Bugallo, M.F., Elvira, V., Martino, L., et al.: Adaptive importance sampling: the past, the present, and the future. IEEE Signal Process. Mag. **34**(4), 60–79 (2017). https://doi.org/10.1109/MSP.2017.2699226

Bürkner, P.C., Gabry, J., Weber, S., et al.: brms: Bayesian regression models using 'Stan'. (2024) https://cran.r-project.org/web/packages/brms/index.html

Bürkner, P.C.: brms: an R package for bayesian multilevel models using Stan. J. Stat. Softw. **80**, 1–28 (2017). https://doi.org/10.18637/jss.v080.i01

Bürkner, P.C., Gabry, J., Vehtari, A.: Efficient leave-one-out cross-validation for Bayesian non-factorized normal and Student-t models. Comput. Stat. **36**(2), 1243–1261 (2020). https://doi.org/10.1007/s00180-020-01045-4

Bürkner, P.C., Kröker, I., Oladyshkin, S., et al.: A fully Bayesian sparse polynomial chaos expansion approach with joint priors on the coefficients and global selection of terms. J. Comput. Phys. **488**, 112210 (2023). https://doi.org/10.1016/j.jcp.2023.112210

Cação, J., Santos, J., Antunes, M., et al.: Machine learning-driven fault identification and classification: a two-step approach for industrial applications. Proc. Comput. Sci. **253**, 1073–1082 (2025). https://doi.org/10.1016/j.procs.2025.01.169

Carmona, C., Nicholls, G.: Semi-modular inference: enhanced learning in multi-modular models by tempering the influence of components. In: Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics. PMLR, pp. 4226–4235 (2020). https://proceedings.mlr.press/v108/carmona20a.html

Carpenter, B., Hoffman, M.D., Brubaker, M., et al.: The Stan Math library: reverse-mode automatic differentiation in C++. (2015) https://doi.org/10.48550/arXiv.1509.07164

Carvalho, C.M., Polson, N.G., Scott, J.G.: Handling sparsity via the horseshoe. In: Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics. PMLR, pp. 73–80, (2009)

Carvalho, C.M., Polson, N.G., Scott, J.G.: The horseshoe estimator for sparse signals. Biometrika **97**(2), 465–480 (2010). https://doi.org/10.1093/biomet/asq017

Cormen, T.H., Leiserson, C.E., Rivest, R.L., et al.: Introduction to algorithms, 3rd edn. MIT Press, Cambridge, Mass (2009)

Dibbelt, J., Strasser, B., Wagner, D.: Customizable contraction hierarchies. ACM J. Exp. Algorithmics (2016). https://doi.org/10.1145/2886843

Fahrmeir, L., Kneib, T., Lang, S., et al.: Regression: Models, Methods and Applications. Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-34333-9

Forrester, A., Sóbester, A., Keane, A.: Engineering design via surrogate modelling: a practical guide. John Wiley & Sons (2008)

Frazier, D.T., Nott, D.J.: Cutting feedback and modularized analyses in generalized bayesian inference. Bayesian Anal. **20**(4), 1647–1675 (2025). https://doi.org/10.1214/24-BA1448

Friedman, J.H., Rafsky, L.C.: Multivariate generalizations of the Wald-Wolfowitz and Smirnov two-sample tests. Ann. Stat. **7**(4), 697–717 (1979)

Gabry, J., Češnovar, R., Johnson, A., et al.: cmdstanr: R interface to 'CmdStan'. (2025) https://mc-stan.org/cmdstanr/

Garzón, A., Kapelan, Z., Langeveld, J., et al.: Machine learning-based surrogate modeling for urban water networks: review and future research directions. Water Resources Res. **58**(5), e2021WR031808. (2022) https://doi.org/10.1029/2021WR031808

Gelman, A., Carlin, J.B., Stern, H.S.: Bayesian data analysis Third edition, Chapman and Hall/CRC (2013). (3rd edn)

Goldfeld, K., Wujciak-Jens, J.: simstudy: simulation of study data. (2024) https://cran.r-project.org/web/packages/simstudy/index.html

Graham, J., Olchowski, A., Gilreath, T.: How many imputations are really needed? Some practical clarifications of multiple imputation theory. Prevent. Sci. Official J. Soc. Prevent. Res. **8**, 206–13 (2007). https://doi.org/10.1007/s11121-007-0070-9

Gretton, A., Borgwardt, K.M., Rasch, M.J., et al.: A Kernel two-sample test. J. Mach. Learn. Res. **13**(25), 723–773 (2012). (http://jmlr.org/papers/v13/gretton12a.html)

Gronau, Q.F., Singmann, H., Forster, J.J., et al.: bridgesampling: bridge sampling for marginal likelihoods and Bayes factors.(2021) https://cran.r-project.org/web/packages/bridgesampling/index.html

Hammersley, J.M., Handscomb, D.C.: Monte Carlo methods. Springer, Netherlands, Dordrecht, (1964). https://doi.org/10.1007/978-94-009-5819-7

Hoffman, M.D., Gelman, A.: The No-U-turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. J. Mach. Learn. Res. (2014)

Huang, H., Pouls, M., Meyer, A., et al.: Travel time prediction using tree-based ensembles. In: Lalla-Ruiz E, Mes M, Voß S (eds) Computational Logistics. Springer International Publishing, Cham, pp. 412–427, (2020)https://doi.org/10.1007/978-3-030-59747-4_27

Hutchings, G., Rumsey, K.N., Bingham, D., et al.: Enhancing approximate modular Bayesian inference by emulating the conditional posterior. Comput. Stat. Data Anal. **212**, 108235 (2025). https://doi.org/10.1016/j.csda.2025.108235

Ionides, E.L.: Truncated importance sampling. J. Comput. Graph. Stat. (2008). https://doi.org/10.1198/106186008X320456

Jacob, P.E., O'Leary, J., Atchadé, Y.F.: Unbiased Markov Chain Monte Carlo methods with couplings. J. R. Stat. Soc. Ser. B Stat Methodol. **82**(3), 543–600 (2020). https://doi.org/10.1111/rssb.12336

Kaufman, L., Rousseeuw, P.: Finding groups in data: an introduction to cluster analysis. Wiley, New York (1990). https://doi.org/10.2307/2532178

Kim, S., Luna, M., Chikontwe, P., et al.: Two-step U-nets for brain tumor segmentation and random forest with radiomics for survival time prediction. In: Crimi A, Bakas S (eds) Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries. Springer International Publishing, Cham, pp. 200–209,(2020) https://doi.org/10.1007/978-3-030-46640-4_19

Koziel, S., Leifsson, L. (eds.). Surrogate-based modeling and optimization: applications in engineering. Springer, New York, NY, (2013). https://doi.org/10.1007/978-1-4614-7551-4

Kuehnert, J., McGlynn, D., Remy, S.L., et al.: Surrogate ensemble forecasting for dynamic climate impact models. (2022) https://doi.org/10.48550/arXiv.2204.05795

Lavin, A., Krakauer, D., Zenil, H., et al.: Simulation intelligence: towards a new generation of scientific methods. (2022) https://doi.org/10.48550/arXiv.2112.03235

Lim, J., Lee, J., An, C., et al.: Enhancing real-time traffic volume prediction: a two-step approach of object detection and time series modelling. IET Intel. Transport Syst. **18**(12), 2744–2758 (2024). https://doi.org/10.1049/itr2.12576

Lindsay, B.G.: Mixture models: theory, geometry and applications. NSF-CBMS Regional Conference Series in Probability and Statistics 5:i–163. Publisher: Institute of Mathematical Statistics (1995)

Liu, Y., Goudie, R.J.B.: Stochastic approximation cut algorithm for inference in modularized Bayesian models. Stat. Comput. **32**(1), 7 (2021). https://doi.org/10.1007/s11222-021-10070-2

Liu, Y., De Vijlder, T., Bittremieux, W., et al.: Current and future deep learning algorithms for tandem mass spectrometry (MS/MS)-based small molecule structure elucidation. Rapid Communications in Mass Spectrometry pp. 9120.(2021) https://doi.org/10.1002/rcm.9120

Lunn, D., Spiegelhalter, D., Thomas, A., et al.: The BUGS project: evolution, critique and future directions. Stat. Med. **28**(25), 3049–3067 (2009). https://doi.org/10.1002/sim.3680

MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1: Statistics, pp. 281–298 (1967)

Mathews, J., Gopalan, G., Gattiker, J., et al.: Sequential Monte Carlo for cut-Bayesian posterior computation. Comput. Stat. **40**(5), 2749–2779 (2025). https://doi.org/10.1007/s00180-024-01576-0

McVinish, R., Mengersen, K., Nur, D., et al.: Recentered importance sampling with applications to Bayesian model validation. J. Comput. Graph. Stat. **22**(1), 215–228 (2013). https://doi.org/10.1080/10618600.2012.681239

Meng, X.L., Wong, W.H.: Simulating ratios of normalizing constants via a simple identity: a theoretical exploration. Stat. Sin. **6**(4), 831–860 (1996)

Neal, R.M.: Probabilistic inference using Markov chain Monte Carlo methods. University of Toronto Toronto, ON, Canada, Department of Computer Science (1993)

Neal, R.M.: Annealed importance sampling. Stat. Comput. **11**(2), 125–139 (2001). https://doi.org/10.1023/A:1008923215028

Neal, R.M.: MCMC using Hamiltonian dynamics. In: Handbook of Markov Chain Monte Carlo. Chapman and Hall/CRC, num Pages: 50 (2011)
NYC Taxi and Limousine Commission (TLC) (2025) TLC Trip Record Data
Oladyshkin, S., Nowak, W.: Data-driven uncertainty quantification using the arbitrary polynomial chaos expansion. Reliability Eng. Syst. Safety **106**, 179–190 (2012). https://doi.org/10.1016/j.ress.2012.05.002
Paananen, T., Piironen, J., Bürkner, P.C., et al.: Implicitly adaptive importance sampling. Stat. Comput. **31**(2), 16 (2021). https://doi.org/10.1007/s11222-020-09982-2
Parker, W.S.: Does matter really matter? Computer simulations, experiments, and materiality. Synthese **169**(3), 483–496 (2009). https://doi.org/10.1007/s11229-008-9434-3
Peck, S.L.: Simulation as experiment: a philosophical reassessment for biological modeling. Trends Ecol. Evol. **19**(10), 530–534 (2004). https://doi.org/10.1016/j.tree.2004.07.019
Peherstorfer, B., Willcox, K., Gunzburger, M.: Survey of multifidelity methods in uncertainty propagation, inference, and optimization. SIAM Rev. **60**(3), 550–591 (2018). https://doi.org/10.1137/16M1082469
Piironen, J., Vehtari, A.: On the hyperprior choice for the global Shrinkage parameter in the Horseshoe Prior. (2016) https://doi.org/10.48550/arXiv.1610.05559
Plummer, M.: Cuts in Bayesian graphical models. Stat. Comput. **25**(1), 37–43 (2015). https://doi.org/10.1007/s11222-014-9503-z
Pompe, E., Jacob, P.E.: Asymptotics of cut distributions and robust modular inference using Posterior Bootstrap. (2021) https://doi.org/10.48550/arXiv.2110.11149
Psaros, A.F., Meng, X., Zou, Z., et al.: Uncertainty quantification in scientific machine learning: methods, metrics, and comparisons. J. Comput. Phys. **477**, 111902 (2023). https://doi.org/10.1016/j.jcp.2022.111902
Quarteroni, A., Sacco, R., Saleri, F.: Numerical mathematics, texts in applied mathematics, vol. 37. Springer, New York, NY, (2007). https://doi.org/10.1007/b98885
Razavi, S., Tolson, B.A., Burn, D.H.: Review of surrogate modeling in water resources. Water Resour. Res. (2012). https://doi.org/10.1029/2011WR011527
Reiser, P., Aguilar, J.E., Guthke, A., et al.: Uncertainty quantification and propagation in surrogate-based Bayesian inference. Stat. Comput. **35**(3), 66 (2025). https://doi.org/10.1007/s11222-025-10597-8
Renardy, M., Yi, T.M., Xiu, D., et al.: Parameter uncertainty quantification using surrogate models applied to a spatial model of yeast mating polarization. PLoS Comput. Biol. **14**(5), e1006181 (2018). https://doi.org/10.1371/journal.pcbi.1006181
Robert, C.P., Casella, G.: Monte Carlo statistical methods. Springer Texts in Statistics, Springer, New York (2004)
Roberts, D.R., Bahn, V., Ciuti, S., et al.: Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. Ecography **40**(8), 913–929 (2017). https://doi.org/10.1111/ecog.02881
Rubin, D.B.: Multiple imputation for nonresponse in surveys. John Wiley & Sons (2004)
Shah, A.D., Bartlett, J.W., Carpenter, J., et al.: Comparison of random forest and parametric imputation models for imputing missing data using MICE: a CALIBER study. Am. J. Epidemiol. **179**(6), 764–774 (2014). https://doi.org/10.1093/aje/kwt312
Shin, A.H., Kim, S.T., Park, G.M.: Time Series anomaly detection using transformer-based GAN with two-step masking. IEEE Access **11**, 74035–74047 (2023). https://doi.org/10.1109/ACCESS.2023.3289921
Smith, R.C.: Uncertainty quantification: theory, implementation, and applications. Comput. Sci. Eng. Soc. Ind. Appl. Math. **10**(1137/1), 9781611973228 (2013)
Song, J., Pramanik, S., Datta, A.: Neural variational inference for cutting feedback during uncertainty propagation. (2025) https://doi.org/10.48550/arXiv.2510.10268
Stolte, M., Kappenberg, F., Rahnenführer, J., et al.: Methods for quantifying dataset similarity: a review, taxonomy and comparison. Stat. Surveys **18**, 163–298 (2024). https://doi.org/10.1214/24-SS149
Sudret, B.: Global sensitivity analysis using polynomial chaos expansions. Reliability Eng. Syst. Safety **93**(7), 964–979 (2008). https://doi.org/10.1016/j.ress.2007.04.002
Sv, B., Groothuis-Oudshoorn, K.: mice: multivariate imputation by chained equations in R. J. Stat. Softw. **45**, 1–67 (2011)
van Buuren, S.: Flexible imputation of missing data, 2nd edn. Chapman and Hall/CRC, Boca Raton, FL (2018)
Stan Development Team (2024) Stan Modeling Language Users Guide and Reference Manual. https://mc-stan.org, version 2.36.0
van Buuren, S., Groothuis-Oudshoorn, K.: Flexible multivariate imputation by MICE. TNO Prevention and Health (1999)
Vehtari, A., Simpson, D., Gelman, A., et al.: Pareto smoothed importance sampling. J. Mach. Learn. Res. **25**(72), 1–58 (2024)
von Hippel, P.T.: How many imputations do you need? A two-stage calculation using a quadratic rule. Sociological Methods Res. **49**(3), 699–718 (2020). https://doi.org/10.1177/0049124117747303
White, I.R., Royston, P., Wood, A.M.: Multiple imputation using chained equations: issues and guidance for practice. Stat. Med. **30**(4), 377–399 (2011). https://doi.org/10.1002/sim.4067
Wiener, N.: The Homogeneous chaos. Am. J. Math. **60**(4), 897–936 (1938). https://doi.org/10.2307/2371268
Winsberg, E.: Science in the age of computer simulation. University of Chicago Press, Chicago, IL (2010)
Yu, X., Nott, D.J., Smith, M.S.: Variational inference for cutting feedback in misspecified models. Stat. Sci. **38**(3), 490–509 (2023). https://doi.org/10.1214/23-STS886
Zhang, L., Carpenter, B., Gelman, A.: Pathfinder: parallel quasi-Newton variational inference. J. Mach. Learn. Res. (2022)
Zhu, Y., Zabaras, N.: Bayesian deep convolutional encoder-decoder networks for surrogate modeling and uncertainty quantification. J. Comput. Phys. **366**, 415–447 (2018). https://doi.org/10.1016/j.jcp.2018.04.018

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.