Guidance for Prior Change via Density Ratio Estimation

Yichen Zang
Department of Mathematics
University of Bristol
Bristol, UK
yichen.zang@bristol.ac.uk

Jiun-Yi Lin
Department of Mathematics
University of Bristol
Bristol, UK
am21541@bristol.ac.uk

Song Liu
Department of Mathematics
University of Bristol
Bristol, UK
song.liu@bristol.ac.uk

Abstract

Simulation-Based Inference (SBI) serves as a vital framework for parameter inference in scientific fields where simulators involve intractable likelihoods, yet while amortized generative models offer rapid posterior estimation, they are often restricted by the specific priors used during training, thereby limiting their flexibility as prior knowledge evolves. To address this prior dependency, PriorGuide was introduced as an inference-time guidance method, but due to its intractable formulation, it relies on Gaussian approximations of the reverse transition kernel and Gaussian mixture model fitting for the prior ratio, both of which introduce systematic bias. Motivated by these limitations, we propose an unbiased test-time guidance framework that leverages Density Ratio Estimation (DRE) to learn a score guidance term, effectively decoupling the inference process from the prior training. Moreover, our framework remains agnostic to the specific density ratio estimators, making it a general and flexible framework for handling prior changes. Experimental results across multiple tasks demonstrate that our method matches or outperforms PriorGuide on C2ST and MMD in most tasks while maintaining robustness even under limited overlap between the training and target priors. Furthermore, we apply our method to Bayesian updating for parameter inference from planetary light-curve data, where it also demonstrates strong effectiveness and robustness. Code is available at `https://github.com/a-chenchen/dre-based-prior-guidance`.

1 Introduction

In the data-driven era, Simulation-Based Inference (SBI) is a fundamental framework for parameter inference in complex systems, such as astrophysics [12], particle physics [1], and epidemiology [17]. Here, while simulators easily generate synthetic data $\mathbf{x}$ for parameters $\boldsymbol{\theta}$, the likelihood $p(\mathbf{x} \mid \boldsymbol{\theta})$ is analytically intractable, rendering traditional Bayesian methods like MCMC inapplicable.

While Approximate Bayesian Computation (ABC) has long been the standard likelihood-free approach, its reliance on manually crafted summary statistics often leads to significant information loss, and its computational efficiency scales poorly with the dimensionality of the parameter space. To overcome these limitations, recent advancements in deep generative modeling have transformed the landscape of SBI. Modern approaches, such as Normalizing Flows [28], Diffusion Models [35], and Flow Matching [20], enable the development of amortized inference methods [34, 40]. By training on paired datasets $\{(\boldsymbol{\theta}^{(i)}, \mathbf{x}^{(i)})\}_{i=1}^{N}$ sampled from the joint distribution $p(\boldsymbol{\theta}, \mathbf{x})$, these models can provide high-quality posterior estimates for new observations without additional simulations, enabling efficient real-time data analysis.

Despite its efficiency, a major limitation of this amortized paradigm is its rigid dependency on the training prior $p(\boldsymbol{\theta})$. In standard Neural Posterior Estimation (NPE), the prior distribution is implicitly

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

embedded in the model's weights during training. Therefore, even though these models generalize well across different observations $\mathbf{x}$, they remain tied to their training prior. This inherent rigidity hinders iterative scientific inquiry, where prior knowledge is frequently refined in light of new data or theoretical insights. Such requirements for dynamic prior adaptation are prevalent across diverse disciplines, including nuclear physics [13] and engineering [21].

When the prior changes to $q(\boldsymbol{\theta})$, the pre-trained amortized SBI model becomes biased, as it continues to reflect the now-obsolete training prior $p(\boldsymbol{\theta})$. The classical solution is to re-simulate the entire dataset from $q(\boldsymbol{\theta})p(\mathbf{x} \mid \boldsymbol{\theta})$ and retrain an SBI model. However, especially for complex simulators, this process has a very high computational cost, making it prohibitively expensive. Recent research has begun to explore "prior amortization" strategies to handle varying priors [2, 39]. These methods typically require pre-defining a specific family of priors or discretizing the prior space during training, which fails to accommodate truly arbitrary prior changes encountered at test-time.

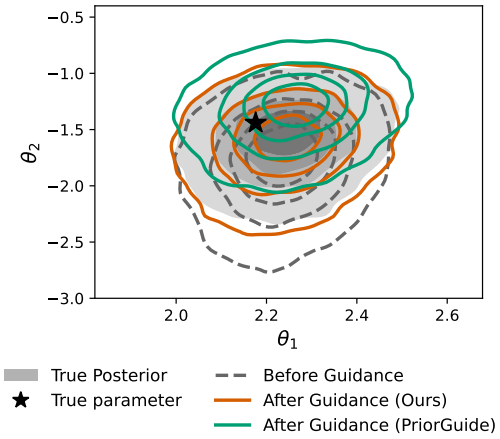

Figure 1: Two-dimensional Posterior approximations under prior shift for Ornstein-Uhlenbeck Process. Grey shading denotes the true posterior; dashed contour the base model; colored contours our method and PriorGuide after guidance. Our method closely tracks the truth while PriorGuide exhibits deviation.

A more efficient way to handle prior changes is test-time guidance, such as the recently proposed PriorGuide [41]. Given a diffusion-based SBI model pre-trained on $p(\boldsymbol{\theta})$ and prior ratio between the old and new prior $\frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta})}$, it introduces a guidance term to the base diffusion model during testing, to adjust the generated samples to a new posterior. However, PriorGuide relies on two approximations due to its intractable formulation: a GMM-based fit for the prior ratio and a Gaussian approximation for the reverse transition kernel. These approximations may introduce significant bias into the estimation of the true guidance, as illustrated in Figure 1.

In this paper, we propose a ratio-estimation-based test-time guidance method. Our approach utilizes a lightweight Density Ratio Estimation (DRE) [6, 10], trained across diffusion steps, to directly learn an unbiased and robust guidance term. Our method offers the following three advantages:

- **Flexible and Unbiased Inference:** Our method estimates the guidance term without the systematic discrepancy incurred by PriorGuide's Gaussian kernel approximation, ensuring unbiased adaptation to arbitrary prior distributions. It is agnostic to ratio estimation methods and provides amortized inference across different observations.

- **Computational Efficiency:** Our method enables direct reuse of existing datasets without additional simulator calls, streamlining the training process. The guidance term is trained using a lightweight density ratio estimator, enabling efficient test-time inference without further correction, which is required by PriorGuide.

- **Robustness:** Adding Gaussian noise to samples has been proposed to ensure that distributions are well-supported, making the density ratio well-defined [42, 33]. Since our model is trained on noisy diffusion states, this mechanism of diffusion naturally stabilizes the ratio estimation process and improves robustness when the target prior is insufficiently supported by the training prior. In contrast, incorporating such noise-based stabilization into PriorGuide is non-trivial, since its guidance relies only on density ratio over clean samples.

2 Background

2.1 Simulation-based Inference

The primary objective in many scientific disciplines is to infer model parameters $\boldsymbol{\theta}$ from observed data $\mathbf{x}$. In Bayesian framework, the posterior distribution is defined as $p(\boldsymbol{\theta} \mid \mathbf{x}) \propto p(\mathbf{x} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})$, where $p(\boldsymbol{\theta})$ represents the prior and $p(\mathbf{x} \mid \boldsymbol{\theta})$ is the likelihood. Since the posterior intrinsically depends on the choice of prior, the prior's specification plays a critical role in the inference result.

In many scientific applications, the likelihood $p(\mathbf{x} \mid \boldsymbol{\theta})$ is analytically intractable or computationally prohibitive to evaluate, even though samples can be drawn from it via stochastic simulators, making it difficult to directly solve for $p(\boldsymbol{\theta} \mid \mathbf{x})$. To address this, Simulation-Based Inference (SBI) [5, 4] leverages parameter-observation pairs $(\boldsymbol{\theta}, \mathbf{x}) \sim p(\mathbf{x} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})$ generated from the simulator to learn the underlying statistical relationships, typically with machine learning methods.

2.2 Diffusion Models

Diffusion models map a complex data distribution $p(\mathbf{z}_0)$ onto a tractable Gaussian distribution via a forward perturbation process and recover samples by reversing it [11, 35]. Following the Variance Preserving (VP) formulation, the diffused state $\mathbf{z}_t$ at time $t \in [0, 1]$ is governed by:

$$p(\mathbf{z}_t \mid \mathbf{z}_0) = \mathcal{N}(\mathbf{z}_t; \sqrt{1 - \sigma^2(t)}\mathbf{z}_0, \sigma^2(t)\mathbf{I}), \tag{1}$$

where $\sigma(t)$ denotes a noise schedule increasing from $\sigma(0) = 0$ to a terminal value $\sigma(1) \approx 1$. New samples are synthesized by reversing this trajectory through a reverse-time Stochastic Differential Equation (SDE):

$$d\mathbf{z}_t = \left[-\frac{1}{2}\beta(t)\mathbf{z}_t - \beta(t)\nabla_{\mathbf{z}_t} \log p(\mathbf{z}_t)\right] dt + \sqrt{\beta(t)}d\bar{\boldsymbol{\omega}}_t, \tag{2}$$

where $\bar{\boldsymbol{\omega}}_t$ is a standard reverse-time Wiener process and $\beta(t) = -\frac{d}{dt}\log(1 - \sigma^2(t))$ is the drift coefficient associated with the noise schedule $\sigma(t)$. The intractable score function $\nabla_{\mathbf{z}_t} \log p(\mathbf{z}_t)$ is approximated by a neural network $\mathbf{s}_\phi(\mathbf{z}_t, t)$ trained via the denoising score matching objective:

$$\mathcal{L}_{\text{DSM}} = \mathbb{E}_{t,\, p(\mathbf{z}_0),\, p(\mathbf{z}_t|\mathbf{z}_0)} \left[\lambda(t) \left\|\mathbf{s}_\phi(\mathbf{z}_t, t) - \nabla_{\mathbf{z}_t} \log p(\mathbf{z}_t \mid \mathbf{z}_0)\right\|^2\right], \tag{3}$$

where the conditional score is analytically given by $\nabla_{\mathbf{z}_t} \log p(\mathbf{z}_t \mid \mathbf{z}_0) = \frac{-(\mathbf{z}_t - \sqrt{1-\sigma^2(t)}\mathbf{z}_0)}{\sigma^2(t)}$, and $\lambda(t)$ is a weighting function.

2.3 Diffusion-based SBI

Diffusion-based SBI [34] casts posterior estimation as a conditional generative task: a score network, conditioned on the observed data $\mathbf{x}$, learns the score of the diffused posterior, which drives a reverse diffusion transporting Gaussian noise to samples from $p(\boldsymbol{\theta} \mid \mathbf{x})$.

The diffusion process from Section 2.2 is applied to $\boldsymbol{\theta}$, with $\boldsymbol{\theta}_0 := \boldsymbol{\theta}$ and $\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_0 \sim \mathcal{N}(\sqrt{1 - \sigma^2(t)}\,\boldsymbol{\theta}_0,\, \sigma^2(t)\mathbf{I})$ as in equation 1. The conditional score network $\mathbf{s}_\phi(\boldsymbol{\theta}_t, \mathbf{x}, t)$ is trained via:

$$\mathcal{L}_{\text{SBI}} = \mathbb{E}_{t,\, p(\boldsymbol{\theta}_0, \mathbf{x}),\, p(\boldsymbol{\theta}_t|\boldsymbol{\theta}_0)} \left[\lambda(t)\|\mathbf{s}_\phi(\boldsymbol{\theta}_t, \mathbf{x}, t) - \nabla_{\boldsymbol{\theta}_t} \log p(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_0)\|^2\right], \tag{4}$$

where $(\boldsymbol{\theta}_0, \mathbf{x})$ is sampled by drawing $\boldsymbol{\theta}_0 \sim p(\boldsymbol{\theta})$ from the prior and $\mathbf{x} \sim p(\mathbf{x} \mid \boldsymbol{\theta}_0)$ from the simulator. Once trained, $\mathbf{s}_\phi$ serves as an amortized estimator of the posterior score: for any observation $\mathbf{x}$, posterior samples $\boldsymbol{\theta} \sim p(\boldsymbol{\theta} \mid \mathbf{x})$ can be generated by integrating a reverse-time SDE from Gaussian noise.

2.4 Density Ratio Estimation

Density Ratio Estimation (DRE) [10, 6] estimates density ratios using only samples from two distributions p and q, without access to their density functions. By optimizing an objective, DRE recovers the ratio $r = \frac{q}{p}$ at optimality. Representative approaches include KLIEP [36], NRE [10], SDRE [19], and NRE-∞ [3].

More generally, many DRE methods can be unified under generalized frameworks closely related to f-divergence estimation [26, 27]. Let f be a convex, lower semi-continuous generator with convex conjugate $g = f^\star$, and let $h : \mathcal{X} \to \mathrm{dom}(g)$ be a variational function, parameterized by a neural network in practice. The objective

$$\mathcal{L}(h) = -\mathbb{E}_q[h] + \mathbb{E}_p[g(h)], \tag{5}$$

is minimized by a variational function $h^\star$ satisfying, under mild regularity conditions,

$$g'(h^\star) = \frac{q}{p}, \tag{6}$$

2.5 Prior Adaptation in Amortized SBI

A persistent challenge in amortized SBI is prior dependency, where the learned posterior is tied to the fixed prior $p(\boldsymbol{\theta})$ used during training. To enable prior flexibility without retraining the simulator, two main research directions have emerged.

The first is meta-amortization, which trains models over a meta-distribution of priors to generalize across various target priors (see Appendix A for details). The second, and more relevant to this work, is inference-time guidance, which adapts a pre-trained posterior during the sampling process. A representative framework is PriorGuide [41], which serves as our primary baseline. PriorGuide incorporates a guidance term into a diffusion-based SBI model using the density ratio $r(\boldsymbol{\theta}) = \frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta})}$ between the target and training priors. The resulting guided score function is:

$$\nabla_{\boldsymbol{\theta}_t} \log q(\boldsymbol{\theta}_t \mid \mathbf{x}) = \underbrace{\mathbf{s}_\phi(\boldsymbol{\theta}_t, \mathbf{x}, t)}_{\text{original score}} + \underbrace{\nabla_{\boldsymbol{\theta}_t} \log \mathbb{E}_{p(\boldsymbol{\theta}_0 \mid \boldsymbol{\theta}_t, \mathbf{x})}\left[r(\boldsymbol{\theta}_0)\right]}_{\text{guidance term}}, \tag{7}$$

where $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}_t$ denote the clean parameters and their noisy counterparts at diffusion step t, respectively.

Despite its efficiency, PriorGuide relies on several restrictive structural simplifications. Specifically, it approximates the ratio $r(\boldsymbol{\theta}_0)$ using a Gaussian Mixture Model (GMM) and assumes a Gaussian form for the reverse transition kernel $p(\boldsymbol{\theta}_0 \mid \boldsymbol{\theta}_t, \mathbf{x})$. These heuristics can introduce systematic bias and lead to instability, particularly when the target prior $q(\boldsymbol{\theta})$ has low overlap with $p(\boldsymbol{\theta})$.

Our method instead employs DRE to approximate the guidance term directly and unbiasedly, bypassing these structural heuristics and enabling more robust posterior adaptation even under substantial prior mismatch.

3 Methodology

Our objective is to develop an unbiased and flexible guidance framework for inference-time prior adaptation. To achieve this, we seek an unbiased, simulation-efficient, and flexible estimator of the guidance term. Once such an estimator is obtained, a diffusion-based SBI model trained under prior $p(\boldsymbol{\theta})$ can be rapidly adapted to a new target prior $q(\boldsymbol{\theta})$ at inference time, enabling efficient generation of posterior samples from $q(\boldsymbol{\theta} \mid \mathbf{x}) \propto q(\boldsymbol{\theta})p(\mathbf{x} \mid \boldsymbol{\theta})$ as well as posterior predictive samples from $q(\mathbf{x}' \mid \mathbf{x})$.

3.1 Problem Formulation

We consider the setting where a score-based diffusion model has been trained on a joint distribution $p(\boldsymbol{\theta}, \mathbf{x}) = p(\boldsymbol{\theta})p(\mathbf{x} \mid \boldsymbol{\theta})$, where $p(\boldsymbol{\theta})$ denotes the training prior and $p(\mathbf{x} \mid \boldsymbol{\theta})$ represents the simulator-defined likelihood. In the standard amortized paradigm, this pre-trained model serves as a foundation for rapid inference for posterior $p(\boldsymbol{\theta} \mid \mathbf{x})$ or posterior predictive $p(\mathbf{x}' \mid \mathbf{x})$.

However, at inference time, a new target prior $q(\boldsymbol{\theta})$ may be introduced to reflect updated scientific constraints or external domain knowledge. Assuming the target prior is defined as $q(\boldsymbol{\theta}) = r(\boldsymbol{\theta})p(\boldsymbol{\theta})$, where $r(\boldsymbol{\theta})$ is a known density ratio, the corresponding target posterior becomes

$$q(\boldsymbol{\theta} \mid \mathbf{x}) = \frac{q(\boldsymbol{\theta})p(\mathbf{x} \mid \boldsymbol{\theta})}{q(\mathbf{x})} \propto q(\boldsymbol{\theta})p(\mathbf{x} \mid \boldsymbol{\theta}). \tag{8}$$

Our primary objective is to perform inference on this posterior $q(\boldsymbol{\theta} \mid \mathbf{x})$ and the corresponding posterior predictive distribution $q(\mathbf{x}' \mid \mathbf{x})$.

Under the prior shift $q(\boldsymbol{\theta}) = r(\boldsymbol{\theta})p(\boldsymbol{\theta})$, the target joint distribution can be rewritten as

$$q(\boldsymbol{\theta}, \mathbf{x}) = r(\boldsymbol{\theta})p(\boldsymbol{\theta}, \mathbf{x}), \tag{9}$$

since the likelihood $p(\mathbf{x} \mid \boldsymbol{\theta})$ remains unchanged. This reweighting relationship forms the basis for the diffusion-space importance weighting strategy developed in Section 3.2.

3.2 Guidance Derivation

For target posterior sampling, we aim to compute the guided score function $\nabla_{\boldsymbol{\theta}_t} \log q(\boldsymbol{\theta}_t \mid \mathbf{x})$, which can be decomposed as:

$$\nabla_{\boldsymbol{\theta}_t} \log q(\boldsymbol{\theta}_t \mid \mathbf{x}) = \nabla_{\boldsymbol{\theta}_t} \log p(\boldsymbol{\theta}_t \mid \mathbf{x}) + \nabla_{\boldsymbol{\theta}_t} \log \frac{q(\boldsymbol{\theta}_t \mid \mathbf{x})}{p(\boldsymbol{\theta}_t \mid \mathbf{x})} = \underbrace{\nabla_{\boldsymbol{\theta}_t} \log p(\boldsymbol{\theta}_t \mid \mathbf{x})}_{\text{original score}} + \underbrace{\nabla_{\boldsymbol{\theta}_t} \log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}}_{\text{guidance term}}. \tag{10}$$

Notably, our guidance term is equivalent to the expression in Eq. 7 (see Appendix B for a formal proof). This reformulated joint log ratio $\log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}$ enables guidance estimation through DRE.

3.3 DRE-based Guidance for Prior Change

The reformulation in Eq. 10 suggests that prior adaptation reduces to estimating the joint log ratio $\log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}$. However, DRE objectives for this ratio typically require samples from both $q(\boldsymbol{\theta}_t, \mathbf{x})$ and $p(\boldsymbol{\theta}_t, \mathbf{x})$. Generating samples from the former involves a sequential process: first sampling $\boldsymbol{\theta} \sim q(\boldsymbol{\theta})$, then invoking the simulator to obtain $\mathbf{x} \sim p(\mathbf{x} \mid \boldsymbol{\theta})$, and finally applying the diffusion forward process to reach $\boldsymbol{\theta}_t \sim q(\boldsymbol{\theta}_t \mid \boldsymbol{\theta})$. This process is computationally expensive and contradicts our primary objective of maintaining a fast and flexible inference pipeline. To avoid direct sampling from $q(\boldsymbol{\theta}_t, \mathbf{x})$, we derive the following importance weighting identity, which we will use below to estimate the guidance term via two DRE methods: NRE and KLIEP.

Proposition 1. *Define the prior ratio $r(\boldsymbol{\theta}_0) = \frac{q(\boldsymbol{\theta}_0)}{p(\boldsymbol{\theta}_0)}$. Further assume that the same forward diffusion process $p(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_0)$ is applied under both priors. Then, for any measurable and integrable function $f(\boldsymbol{\theta}_t, \mathbf{x}, t)$,*

$$\mathbb{E}_{q(\boldsymbol{\theta}_t, \mathbf{x})}[f(\boldsymbol{\theta}_t, \mathbf{x}, t)] = \mathbb{E}_{p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x})}\left[r(\boldsymbol{\theta}_0) f(\boldsymbol{\theta}_t, \mathbf{x}, t)\right]. \tag{11}$$

Sketch of proof: Under our assumptions, we have the following identities

$$p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x}) = p(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_0)p(\boldsymbol{\theta}_0, \mathbf{x}), \qquad q(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x}) = p(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_0)q(\boldsymbol{\theta}_0, \mathbf{x}). \tag{12}$$

Thus,

$$q(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x}) = p(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_0)q(\boldsymbol{\theta}_0, \mathbf{x}) = p(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_0) \underbrace{r(\boldsymbol{\theta}_0)p(\boldsymbol{\theta}_0, \mathbf{x})}_{\text{Eq.9}} = r(\boldsymbol{\theta}_0)p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x}),$$

and the conclusion follows by substituting $q(\boldsymbol{\theta}_t, \mathbf{x})$ in the expectation expression.

This reformulation enables simulation-free training by leveraging samples from the original joint distribution $p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x}) = p(\boldsymbol{\theta}_0, \mathbf{x})p(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_0)$. Specifically, the required samples $(\boldsymbol{\theta}_0, \mathbf{x})$ can be directly reused from the base SBI model's training set. By simply perturbing $\boldsymbol{\theta}_0$ via the diffusion forward kernel $p(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_0)$, the adaptation process circumvents any additional simulator calls.

We now instantiate this identity for NRE and KLIEP in turn.

NRE. NRE estimates the log density ratio $\log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}$ by training a discriminator $a_\psi(\boldsymbol{\theta}_t, \mathbf{x}, t)$ with objective

$$\mathcal{L}_{\text{NRE}} = \mathbb{E}_t\left[-\mathbb{E}_{q(\boldsymbol{\theta}_t, \mathbf{x})}[\log S(a_\psi)] - \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})}\left[\log\big(1 - S(a_\psi)\big)\right]\right], \tag{13}$$

where a_ψ is shorthand for $a_\psi(\boldsymbol{\theta}_t, \mathbf{x}, t)$ and S is sigmoid function. In practice, the loss is estimated using an equal number of samples drawn from each of $q(\boldsymbol{\theta}_t, \mathbf{x})$ and $p(\boldsymbol{\theta}_t, \mathbf{x})$.

Applying Prop. 1 to Eq. 13, the objective becomes:

$$\mathcal{L}_{\text{NRE}} = \mathbb{E}_t \left[-\mathbb{E}_{p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x})}[r(\boldsymbol{\theta}_0) \log S(a_\psi)] - \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})}[\log(1 - S(a_\psi))] \right]. \tag{14}$$

Finally, the guidance term is recovered via the gradient of the optimum $a^\star$: $\nabla_{\boldsymbol{\theta}_t} \log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})} = \nabla_{\boldsymbol{\theta}_t} a^\star(\boldsymbol{\theta}_t, \mathbf{x}, t)$.

KLIEP. KLIEP estimates the same log density ratio by training an estimator $a_\psi(\boldsymbol{\theta}_t, \mathbf{x}, t)$ via the following objective

$$\mathcal{L}_{\text{KLIEP}} = \mathbb{E}_t \left[-\mathbb{E}_{q(\boldsymbol{\theta}_t, \mathbf{x})}[a_\psi] + \log \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})}[\exp a_\psi] \right]. \tag{15}$$

Applying Prop. 1 yields

$$\mathcal{L}_{\text{KLIEP}} = \mathbb{E}_t \left[-\mathbb{E}_{p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x})}[r(\boldsymbol{\theta}_0) a_\psi] + \log \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})}[\exp a_\psi] \right]. \tag{16}$$

which, like $\mathcal{L}_{\text{NRE}}$ in Eq. 14, admits the same simulation-free training procedure. The guidance term is likewise recovered via the gradient of the optimum $a^\star$: $\nabla_{\boldsymbol{\theta}_t} \log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})} = \nabla_{\boldsymbol{\theta}_t} a^\star(\boldsymbol{\theta}_t, \mathbf{x}, t)$.

The two examples above illustrate an important observation: although derived from different objectives, both NRE and KLIEP ultimately estimate the same diffusion-space log density ratio $\log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}$.

More broadly, many density ratio estimation methods can be interpreted within a unified variational DRE framework, where different objectives recover the same ratio at optimality. This observation motivates a general estimator-agnostic guidance framework for inference-time prior adaptation.

3.4 General DRE-based Guidance Framework

We now extend the general DRE objectives of Section 2.4 to the diffusion joint ratio, yielding a general framework for guidance term estimation.

Theorem 1. *Define the prior ratio $r(\boldsymbol{\theta}_0) = \frac{q(\boldsymbol{\theta}_0)}{p(\boldsymbol{\theta}_0)}$. Consider the generalized density ratio estimation objective:*

$$h^\star = \arg\min_h \left\{ \mathbb{E}_t \left[-\mathbb{E}_{p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x})} \left[r(\boldsymbol{\theta}_0) h(\boldsymbol{\theta}_t, \mathbf{x}, t) \right] + \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})} \left[g(h(\boldsymbol{\theta}_t, \mathbf{x}, t)) \right] \right] \right\}, \tag{17}$$

where $g : \mathbb{R} \to \mathbb{R}$ is differentiable and strictly convex.

Then the optimal solution satisfies:

$$g'(h^\star(\boldsymbol{\theta}_t, \mathbf{x}, t)) = \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}, \tag{18}$$

Consequently, the corresponding guidance term is recovered via:

$$\nabla_{\boldsymbol{\theta}_t} \log \left(g'(h^\star(\boldsymbol{\theta}_t, \mathbf{x}, t)) \right) = \nabla_{\boldsymbol{\theta}_t} \log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}. \tag{19}$$

This general form is instantiated via $h(\boldsymbol{\theta}_t, \mathbf{x}, t) = \sigma_g(a_\psi(\boldsymbol{\theta}_t, \mathbf{x}, t))$, where $a_\psi \in \mathbb{R}$ is an unconstrained network output and $\sigma_g : \mathbb{R} \to \text{dom}(g)$ is an activation function determined by the choice of generator g. NRE corresponds to $(\sigma_g, g) = (\log S, -\log(1 - e^h))$, recovering Eq. 14 exactly as a special case. KLIEP shares the same underlying network a_ψ but its global log-partition structure falls outside this pointwise template; we show separately that it recovers the same guidance term at optimality (Appendix D.1). More generally, different choices of g induce different density ratio estimators within this shared framework, which we summarize in Table 1 (Appendix D.2). We provide the formal proof of Theorem 1 in Appendix C.

Beyond its generality, this framework inherits a natural stabilization mechanism from diffusion. By performing DRE over diffused states $\boldsymbol{\theta}_t$ rather than clean samples, the inherent Gaussian noise ensures that the distributions remain well-supported. This prevents the density ratio from becoming ill-defined, ensuring robust guidance even when the target and training priors have mismatched supports.

4 Experiments

We empirically evaluate DRE-based guidance on SBI problems requiring test-time prior adaptation, for both posterior and posterior predictive inference, using Simformer [9] as the base diffusion model. In Section 4.1, we assess performance on several numerical SBI benchmarks, comparing our approach with existing methods for prior adaptation. In Section 4.2, we demonstrate the applicability of our method to a real-world light curve task via sequential posterior updating, using a Bayesian update algorithm built upon our framework (see Algorithm 1 in Appendix F).

4.1 Numerical Experiments

We evaluate our method on a diverse set of SBI benchmarks, focusing on its ability to adapt to prior change at test time. We consider two regimes:

- **In-distribution (ID):** The training prior provides sufficient support for the target prior.
- **Out-of-distribution (OOD):** The target prior shift results in poor overlap with the training prior.

We compare our DRE-based guidance, implemented via NRE and KLIEP, against the vanilla Simformer (no adaptation), PriorGuide, and weighted score matching (WSM), a baseline that learns an additive score correction via a prior-ratio-weighted denoising score matching objective (Appendix E.1). Our method, PriorGuide, and WSM use the same setting and are therefore directly comparable: all three involve only lightweight test-time training, leave the base model unchanged, require no new simulator calls, and access the prior change only through the ratio $q(\boldsymbol{\theta}_0)/p(\boldsymbol{\theta}_0)$. We additionally report two reference methods that are not directly comparable, Simformer+SIR and analytic guidance (Appendix E.2).

4.1.1 Posterior Inference under Prior Shift

We evaluate posterior inference performance on five representative SBI benchmarks: Two Moons [22], SLCP [29], Gaussian Linear 6D [22], OUP [38], and the Turin model [37] (details in Appendix G.1). These tasks span low- to high-dimensional spaces and include both static and time-series data. We consider both ID and OOD regimes.

For each regime, we randomly draw 10 independent observations from the target marginal distribution of the observations. For each observation, we generate 5,000 posterior samples and repeat this sampling process across 3 independent runs to ensure statistical reliability. Performance is evaluated using C2ST, MMD, W_2, and RMSE. Due to space constraints, we present C2ST scores in Figure 2, while full results are reported in Table 2 of Appendix H.

Overall, our methods (NRE and KLIEP) achieve the best or are not significantly worse than the best-performing method on most tasks, with the largest margins in the OOD regime: on Gaussian Linear 6D and OUP, our DRE-based guidance attains C2ST scores of $\sim$0.58 and $\sim$0.54, versus $\sim$0.69 and $\sim$0.65 for PriorGuide, and on SLCP (OOD), our NRE is the only method in the best-performing group. By comparison, WSM generally underperforms our methods and shows high variance on some tasks, suggesting that learning the score correction by weighted regression is harder than estimating the density ratio directly.

Figure 2 shows two failure cases, Two Moons (OOD) and Turin, which arise for distinct reasons; we analyse both in Appendix I.

4.1.2 Posterior Predictive under Prior Shift

We evaluate posterior predictive performance under both ID and OOD regimes on two time-series SBI tasks: the OUP and Turin models. For all tasks, the first one-third of trajectories are provided as observed, and the remaining two-thirds are predicted.

While RMSE is comparable for most methods and settings (Table 3 in Appendix H), qualitative evaluation in Figure 3 reveals distinct failure modes. In OUP (OOD) (Fig. 3a), PriorGuide's predictive distribution is systematically biased, a failure that RMSE also registers. In Turin (ID) (Fig. 3b), all methods attain comparable RMSE, yet PriorGuide misplaces predictive mass in the upper tail.

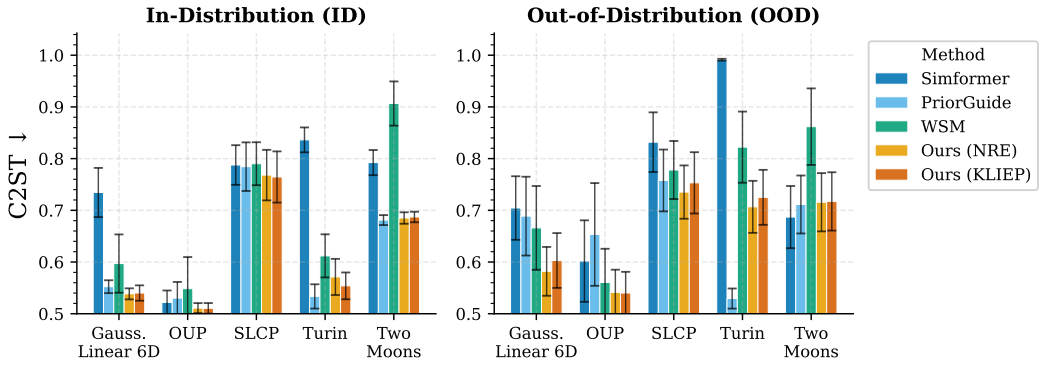

Figure 2: C2ST scores (↓ better, chance level = 0.5) across five benchmark tasks under ID and OOD settings. Our methods (NRE and KLIEP) are best or not significantly worse than the best method on most tasks, with the largest gains under OOD shift; Two Moons and Turin are exceptions, analysed in Appendix I.

These observations are consistent with the posterior results: PriorGuide's structural approximation introduces bias, while our method and WSM show neither failure mode even where RMSE is comparable. Additional results are provided in Appendix H.

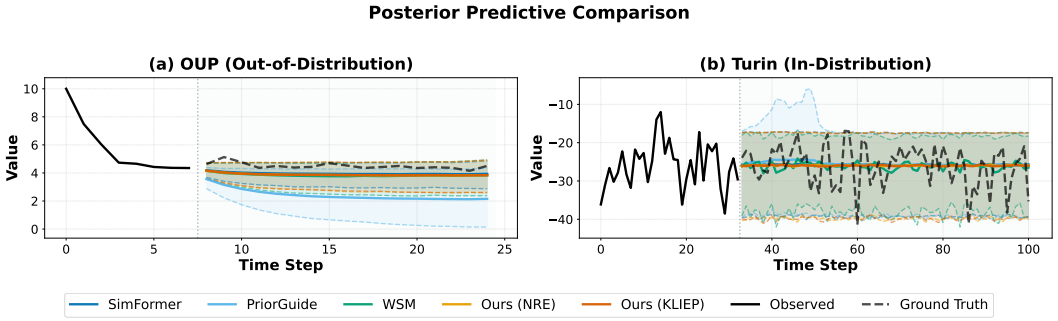

Figure 3: Posterior predictive comparison on time series forecasting. (a) OUP (OOD), (b) Turin (ID). Solid lines show predictive means, dashed lines and shaded regions indicate 95% confidence intervals. Black lines represent observed data (solid) and ground truth (dashed). Vertical line marks the prediction boundary.

4.2 Planet Light Curve Posterior Update

We evaluate our method on a sequential inference task using stellar light curves from the TESS mission [32]. Specifically, we consider three exoplanet systems: GJ 436 b, WASP-39 b, and WASP-126 b. For each system, we extract a sequence of photometric observations $\{\mathbf{x}_{\text{obs}}^{(n)}\}_{n=1}^{N}$, where each $\mathbf{x}_{\text{obs}}^{(n)} \in \mathbb{R}^{128}$ is a normalized light-curve segment centered on a planetary transit event (see Figure 6 in Appendix J.2 for an example sequence of observation windows for WASP-39 b). The goal is to infer 8 parameters $\boldsymbol{\theta} \in \mathbb{R}^8$, including orbital geometry (k, a, inc, P), stellar effects (u_1, u_2), and observational parameters (t_0, σ) (detailed descriptions in Table 5). We report results on (k, a, inc, P).

Sequential Posterior Guidance. As new observations become available, the posterior from previous steps serves as the prior for subsequent inference. Assuming conditional independence, the posterior evolves as

$$p(\boldsymbol{\theta} \mid \mathbf{x}^{(1:n+1)}) \propto p(\mathbf{x}^{(n+1)} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta} \mid \mathbf{x}^{(1:n)}).$$

We train an amortized diffusion model for $p(\boldsymbol{\theta} \mid \mathbf{x})$ and use DRE to estimate $r^{(n)}(\boldsymbol{\theta}) \approx p(\boldsymbol{\theta} \mid \mathbf{x}^{(1:n)})/p(\boldsymbol{\theta})$, enabling posterior guidance at each step. This yields a sequence $p(\boldsymbol{\theta}), p(\boldsymbol{\theta} \mid \mathbf{x}^{(1)}), \ldots, p(\boldsymbol{\theta} \mid \mathbf{x}^{(1:N)})$. The full sequential update procedure is given in Algorithm 1 in Appendix F.

Results. Figure 4 shows that our method shifts and concentrates the posterior toward the reference values as observations accumulate, demonstrating stable sequential updating. In contrast, PriorGuide becomes unstable after a few updates, producing over-dispersed or collapsed posteriors. This trend is consistent across other systems (Figures 7 and 8 in Appendix K) and is further reflected in the RMSE comparisons (Figure 9 in Appendix K): our method maintains low and stable error, with slight improvements as more observations are incorporated, while PriorGuide exhibits progressively increasing error across sequential updates, suggesting the accumulation of guidance bias over time.

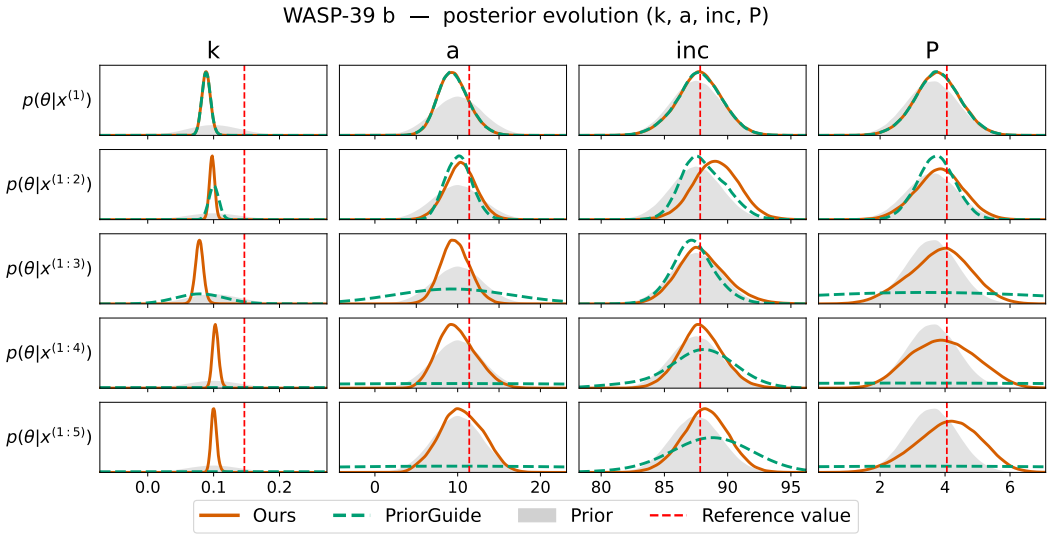

Figure 4: Sequential posterior evolution for WASP-39b over 5 observations. Row n shows $p(\boldsymbol{\theta} \mid \mathbf{x}^{(1:n)})$ for k, a, inc, and P. Grey: prior; red dashed: reference value from [8].

5 Discussion

Our results highlight that accurate guidance estimation is key to prior adaptation in amortized SBI. By learning the gradient of the joint density ratio via DRE, our method avoids the structural approximations used in PriorGuide and provides an unbiased estimate of the guidance term. Empirically, we observe gains on the majority of tasks, particularly under out-of-distribution (OOD) prior shifts, where PriorGuide degrades due to biased guidance.

Beyond accuracy, the two methods also differ in cost. Both avoid additional simulator calls but fit a lightweight model per target prior, a ratio estimator in our case and a GMM in theirs, so neither is purely test-time. At inference, however, our sampling requires none of the Langevin correction PriorGuide relies on, a cost recurring at every observation.

Performance of our method depends on the quality of the learned ratio estimator, which degrades in high dimensions; compressing observations with a learned embedding network is the standard remedy and an important direction for future work. We also observe a case where the prior ratio is nearly flat over the posterior support, in which guidance carries little signal for any method.

Overall, our framework offers a practical test-time-training route to prior adaptation without retraining the base model, particularly useful in scientific workflows with evolving prior knowledge.

Acknowledgements and Disclosures

We thank Sotiria Fotopoulou and Grant Stevens for helpful suggestions on the planetary light-curve application. The authors thank the anonymous reviewers for their insightful feedback.

Funding. Yichen Zang is funded by a CSC–University of Bristol Joint Scholarship. The authors acknowledge the use of resources provided by the Isambard-AI National AI Research Resource (AIRR). Isambard-AI is operated by the University of Bristol and is funded by the UK Government's Department for Science, Innovation and Technology (DSIT) via UK Research and Innovation; and the Science and Technology Facilities Council [ST/AIRR/I-A-I/1023] [25].

Competing interests. The authors declare no competing interests.

References

[1] Johann Brehmer. Simulation-based inference in particle physics. *Nature Reviews Physics*, 2021.

[2] Paul Edmund Chang, Nasrulloh Ratu Bagus Satrio Loka, Daolang Huang, Ulpu Remes, Samuel Kaski, and Luigi Acerbi. Amortized probabilistic conditioning for optimization, simulation and inference. In *International Conference on Artificial Intelligence and Statistics*, 2025.

[3] Kristy Choi, Chenlin Meng, Yang Song, and Stefano Ermon. Density ratio estimation via infinitesimal classification. In *International Conference on Artificial Intelligence and Statistics*, 2022.

[4] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 2020.

[5] Michael Deistler, Jan Boelts, Peter Steinbach, Guy Moss, Thomas Moreau, Manuel Gloeckler, Pedro L. C. Rodrigues, Julia Linhart, Janne K. Lappalainen, Benjamin Kurt Miller, Pedro J. Gonçalves, Jan-Matthis Lueckmann, Cornelius Schröder, and Jakob H. Macke. Simulation-based inference: A practical guide, 2025.

[6] Conor Durkan, Iain Murray, and George Papamakarios. On contrastive learning for likelihood-free inference. In *International Conference on Machine Learning*, 2020.

[7] Lasse Elsemüller, Hans Olischläger, Marvin Schmitt, Paul-Christian Bürkner, Ullrich Köthe, and Stefan T. Radev. Sensitivity-aware amortized Bayesian inference. *Transactions on Machine Learning Research*, 2024.

[8] F. Faedi, S. C. C. Barros, D. R. Anderson, D. J. A. Brown, A. Collier Cameron, D. Pollacco, I. Boisse, G. Hébrard, M. Lendl, T. A. Lister, B. Smalley, R. A. Street, A. H. M. J. Triaud, J. Bento, F. Bouchy, O. W. Butters, B. Enoch, C. A. Haswell, C. Hellier, F. P. Keenan, G. R. M. Miller, V. Moulds, C. Moutou, A. J. Norton, D. Queloz, A. Santerne, E. K. Simpson, I. Skillen, A. M. S. Smith, S. Udry, C. A. Watson, R. G. West, and P. J. Wheatley. WASP-39b: a highly inflated Saturn-mass planet orbiting a late G-type star. *Astronomy & Astrophysics*, 2011.

[9] Manuel Gloeckler, Michael Deistler, Christian Dietrich Weilbach, Frank Wood, and Jakob H. Macke. All-in-one simulation-based inference. In *International Conference on Machine Learning*, 2024.

[10] Joeri Hermans, Volodimir Begy, and Gilles Louppe. Likelihood-free MCMC with amortized approximate ratio estimators. In *International Conference on Machine Learning*, 2020.

[11] Jonathan Ho, Ajay N. Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020.

[12] Matthew Ho, Deaglan J. Bartlett, Nicolas Chartier, Carolina Cuesta-Lazaro, Simon Ding, Axel Lapel, Pablo Lemos, Christopher C. Lovell, T. Lucas Makinen, Chirag Modi, Viraj Pandya, Shivam Pandey, Lucia A. Perez, Benjamin Wandelt, and Greg L. Bryan. LtU-ILI: An all-in-one framework for implicit inference in astrophysics and cosmology. *The Open Journal of Astrophysics*, 2024.

[13] Weiguang Jiang and Christian Forssén. Bayesian probability updates using sampling/importance resampling: Applications in nuclear theory. *Frontiers in Physics*, 2022.

[14] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

[15] G. Kovács, S. Zucker, and T. Mazeh. A box-fitting algorithm in the search for periodic transits. *Astronomy & Astrophysics*, 2002.

[16] A. A. Lanotte, M. Gillon, B.-O. Demory, J. J. Fortney, N. Astudillo, X. Bonfils, P. Magain, X. Delfosse, T. Forveille, C. Lovis, M. Mayor, V. Neves, F. Pepe, D. Queloz, N. Santos, and S. Udry. A global analysis of Spitzer and new HARPS data confirms the loneliness and metal-richness of GJ 436 b. *Astronomy and Astrophysics*, 2014.

[17] Xiahui Li, Fergus Chadwick, and Ben Swallow. Advances in approximate Bayesian inference for models in epidemiology. *Epidemics*, 2025.

[18] Lightkurve Collaboration, José Vinícius de Miranda Cardoso, Christina Hedges, Michael Gully-Santiago, Nicholas Saunders, Ann Marie Cody, Thomas Barclay, Oliver Hall, Sheila Sagear, Emma Turtelboom, Johnny Zhang, Andy Tzanidakis, Ken Mighell, Jeff Coughlin, Keaton Bell, Zach Berta-Thompson, Peter Williams, Jessie Dotson, and Geert Barentsen. Lightkurve: Kepler and TESS time series analysis in Python. Astrophysics Source Code Library, record ascl:1812.013, 2018.

[19] Song Liu, Takafumi Kanamori, Wittawat Jitkrittum, and Yu Chen. Fisher efficient inference of intractable models. In *Advances in Neural Information Processing Systems*, 2019.

[20] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.

[21] Yushan Liu, Luyi Li, and Zeming Chang. Efficient Bayesian model updating for dynamic systems. *Reliability Engineering & System Safety*, 2023.

[22] Jan-Matthis Lueckmann, Jan Boelts, David S. Greenberg, Pedro J. Gonçalves, and Jakob H. Macke. Benchmarking simulation-based inference. In *International Conference on Artificial Intelligence and Statistics*, 2021.

[23] Jan-Matthis Lueckmann, Pedro J. Gonçalves, Giacomo Bassetto, Kaan Öcal, Marcel Nonnenmacher, and Jakob H. Macke. Flexible statistical inference for mechanistic models of neural dynamics. In *Advances in Neural Information Processing Systems*, 2017.

[24] P. F. L. Maxted, D. R. Anderson, A. Collier Cameron, L. Delrez, M. Gillon, C. Hellier, E. Jehin, M. Lendl, M. Neveu-VanMalle, F. Pepe, D. Pollacco, D. Queloz, D. Ségransan, B. Smalley, A. M. S. Smith, J. Southworth, A. H. M. J. Triaud, S. Udry, T. Wagg, and R. G. West. Five transiting hot Jupiters discovered using WASP-South, Euler, and TRAPPIST: WASP-119 b, WASP-124 b, WASP-126 b, WASP-129 b, and WASP-133 b. *Astronomy and Astrophysics*, 2016.

[25] Simon McIntosh-Smith, Sadaf R Alam, and Christopher Woods. Isambard-ai: a leadership class supercomputer optimised specifically for artificial intelligence, 2024.

[26] XuanLong Nguyen, Martin J. Wainwright, and Michael I. Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 2010.

[27] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-GAN: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems*, 2016.

[28] George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 2021.

[29] George Papamakarios, David Sterratt, and Iain Murray. Sequential neural likelihood: Fast likelihood-free inference with autoregressive flows. In *International Conference on Artificial Intelligence and Statistics*, 2019.

[30] Hannu Parviainen. PyTransit: Fast and easy exoplanet transit modelling in Python. *Monthly Notices of the Royal Astronomical Society*, 2015.

[31] Stefan T. Radev, Ulf K. Mertens, Andreas Voss, Lynton Ardizzone, and Ullrich Köthe. BayesFlow: Learning complex stochastic models with invertible neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[32] George R. Ricker, Joshua N. Winn, Roland Vanderspek, David W. Latham, Gáspár Á. Bakos, Jacob L. Bean, Zachory K. Berta-Thompson, Timothy M. Brown, Lars Buchhave, Nathaniel R. Butler, R. Paul Butler, William J. Chaplin, David Charbonneau, Jørgen Christensen-Dalsgaard, Mark Clampin, Drake Deming, John Doty, Nathan De Lee, Courtney Dressing, Edward W. Dunham, Michael Endl, Francois Fressin, Jian Ge, Thomas Henning, Matthew J. Holman, Andrew W. Howard, Shigeru Ida, Jon M. Jenkins, Garrett Jernigan, John Asher Johnson, Lisa Kaltenegger, Nobuyuki Kawai, Hans Kjeldsen, Gregory Laughlin, Alan M. Levine, Douglas Lin, Jack J. Lissauer, Phillip MacQueen, Geoffrey Marcy, Peter R. McCullough, Timothy D. Morton, Norio Narita, Martin Paegert, Enric Palle, Francesco Pepe, Joshua Pepper, Andreas Quirrenbach, Stephen A. Rinehart, Dimitar Sasselov, Bun'ei Sato, Sara Seager, Alessandro Sozzetti, Keivan G. Stassun, Peter Sullivan, Andrew Szentgyorgyi, Guillermo Torres, Stephane Udry, and Joel Villasenor. Transiting Exoplanet Survey Satellite (TESS). *Journal of Astronomical Telescopes, Instruments, and Systems*, 2015.

[33] Kevin Roth, Aurelien Lucchi, Sebastian Nowozin, and Thomas Hofmann. Stabilizing training of generative adversarial networks through regularization. In *Advances in Neural Information Processing Systems*, 2017.

[34] Louis Sharrock, Jack Simons, Song Liu, and Mark Beaumont. Sequential neural score estimation: Likelihood-free inference with conditional score based diffusion models. In *International Conference on Machine Learning*, 2024.

[35] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

[36] Masashi Sugiyama, Shinichi Nakajima, Hisashi Kashima, Paul Buenau, and Motoaki Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Advances in Neural Information Processing Systems*, 2007.

[37] George L. Turin, Fred D. Clapp, Tom L. Johnston, Stephen B. Fine, and Dan Lavry. A statistical model of urban multipath propagation. *IEEE Transactions on Vehicular Technology*, 1972.

[38] G. E. Uhlenbeck and L. S. Ornstein. On the theory of the Brownian motion. *Physical Review*, 1930.

[39] George Whittle, Juliusz Ziomek, Jacob Rawling, and Michael A. Osborne. Distribution transformers: Fast approximate Bayesian inference with on-the-fly prior adaptation. In *International Conference on Machine Learning*, 2026.

[40] Jonas Wildberger, Maximilian Dax, Simon Buchholz, Stephen Green, Jakob H. Macke, and Bernhard Schölkopf. Flow matching for scalable simulation-based inference. In *Advances in Neural Information Processing Systems*, 2023.

[41] Yang Yang, Severi Rissanen, Paul Edmund Chang, Nasrulloh Ratu Bagus Satrio Loka, Daolang Huang, Arno Solin, Markus Heinonen, and Luigi Acerbi. PriorGuide: Test-time prior adaptation for simulation-based inference. In *International Conference on Learning Representations*, 2026.

[42] Mingtian Zhang, Peter Hayes, Thomas Bird, Raza Habib, and David Barber. Spread divergence. In *International Conference on Machine Learning*, 2020.

A Meta-Amortization Methods for Prior Change

Meta-amortization strategies aim to learn a functional mapping that accepts both observed data $\mathbf{x}$ and a formal prior specification $p(\boldsymbol{\theta})$ as inputs. This is achieved by training the model over a "meta-prior"—a distribution or a predefined set of possible prior specifications. For instance, **Sensitivity-aware SBI** [7] trains across a discrete set of alternative priors to provide efficient sensitivity analysis. More general frameworks like the **Distribution Transformer (DT)** [39] and the **Amortized Conditioning Engine (ACE)** [2] employ attention-based architectures to process prior specifications represented as GMMs or histograms. While these methods enable rapid adaptation at runtime, their flexibility is fundamentally bounded by the meta-prior seen during training, often struggling to generalize to prior geometries that deviate significantly from the pre-training set.

B Equivalence of Guidance Terms in Eq.7

In this section, we show the equivalence between our guidance term in Eq. 10 and the one in Eq. 7. By Eq. 9, the target and original joint distributions are related by $q(\boldsymbol{\theta}_0, \mathbf{x}) = r(\boldsymbol{\theta}_0)p(\boldsymbol{\theta}_0, \mathbf{x})$. Since the transition kernel $p(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_0, \mathbf{x})$ is independent of the prior, the marginal joint distribution $q(\boldsymbol{\theta}_t, \mathbf{x})$ can be simplified as:

$$q(\boldsymbol{\theta}_t, \mathbf{x}) = \int r(\boldsymbol{\theta}_0)p(\boldsymbol{\theta}_0, \mathbf{x})p(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_0, \mathbf{x})d\boldsymbol{\theta}_0 = \int r(\boldsymbol{\theta}_0)p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x})d\boldsymbol{\theta}_0. \tag{20}$$

The ratio between the target and original joint distributions is then:

$$\begin{aligned} \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})} &= \frac{\int r(\boldsymbol{\theta}_0)p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x})d\boldsymbol{\theta}_0}{p(\boldsymbol{\theta}_t, \mathbf{x})} \\ &= \int r(\boldsymbol{\theta}_0)\frac{p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}d\boldsymbol{\theta}_0 \\ &= \int r(\boldsymbol{\theta}_0)p(\boldsymbol{\theta}_0 \mid \boldsymbol{\theta}_t, \mathbf{x})d\boldsymbol{\theta}_0 \\ &= \mathbb{E}_{p(\boldsymbol{\theta}_0 \mid \boldsymbol{\theta}_t, \mathbf{x})}\left[r(\boldsymbol{\theta}_0)\right]. \end{aligned} \tag{21}$$

Taking the gradient of the logarithm on both sides yields the equivalence:

$$\nabla_{\boldsymbol{\theta}_t} \log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})} = \nabla_{\boldsymbol{\theta}_t} \log \mathbb{E}_{p(\boldsymbol{\theta}_0 \mid \boldsymbol{\theta}_t, \mathbf{x})}\left[r(\boldsymbol{\theta}_0)\right]. \tag{22}$$

C Proof of Theorem 1

Proof. Following the variational framework for density ratio estimation [26, 36], we consider the problem of estimating the ratio of two distributions by minimizing a divergence-based objective. Since the objective in Eq. 17 is separable across diffusion timesteps t, we can analyze the functional pointwise for a fixed t:

$$\mathcal{L}(h) = -\mathbb{E}_{q(\boldsymbol{\theta}_t, \mathbf{x})}[h(\boldsymbol{\theta}_t, \mathbf{x}, t)] + \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})}[g(h(\boldsymbol{\theta}_t, \mathbf{x}, t))]. \tag{23}$$

By expressing the first term as an expectation over the training distribution $p(\boldsymbol{\theta}_t, \mathbf{x})$, the functional can be rewritten as:

$$\mathcal{L}(h) = \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})}\left[-\frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}h(\boldsymbol{\theta}_t, \mathbf{x}, t) + g(h(\boldsymbol{\theta}_t, \mathbf{x}, t))\right]. \tag{24}$$

For a fixed state $(\boldsymbol{\theta}_t, \mathbf{x}, t)$, we minimize the integrand $\ell(h) = -\frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}h + g(h)$. Given that g is strictly convex, the unique minimizer $h^\star$ is obtained by setting the first-order derivative to zero:

$$\frac{\partial \ell(h)}{\partial h} = -\frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})} + g'(h) = 0 \implies g'(h^\star) = \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}. \tag{25}$$

To facilitate simulation-free training, we apply the change-of-measure identity from Proposition 1, allowing the expectation over $q(\boldsymbol{\theta}_t, \mathbf{x})$ to be computed using samples from the original joint distribution $p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x})$:

$$\mathbb{E}_{q(\boldsymbol{\theta}_t, \mathbf{x})}[h(\boldsymbol{\theta}_t, \mathbf{x}, t)] = \mathbb{E}_{p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x})}[r(\boldsymbol{\theta}_0)h(\boldsymbol{\theta}_t, \mathbf{x}, t)], \tag{26}$$

where $r(\boldsymbol{\theta}_0) = q(\boldsymbol{\theta}_0)/p(\boldsymbol{\theta}_0)$ is the prior ratio. Substituting this into the variational functional yields the final objective:

$$h^\star = \arg\min_h \mathbb{E}_t \left[-\mathbb{E}_{p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x})}[r(\boldsymbol{\theta}_0) h(\boldsymbol{\theta}_t, \mathbf{x}, t)] + \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})}[g(h(\boldsymbol{\theta}_t, \mathbf{x}, t))] \right]. \tag{27}$$

Taking the gradient of the log-ratio recovers the required guidance term:

$$\nabla_{\boldsymbol{\theta}_t} \log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})} = \nabla_{\boldsymbol{\theta}_t} \log \left(g'(h^\star(\boldsymbol{\theta}_t, \mathbf{x}, t)) \right). \tag{28}$$

This completes the proof. □

D Special cases of General DRE-based Guidance

D.1 Derivation of KLIEP as an Extended Instance

Unlike NRE, the KLIEP loss $\mathcal{L}_{\text{KLIEP}}$ in Eq. 16 is not a pointwise instance of Theorem 1. We show below that it can be reduced to the same variational form.

Fix t and write $a = a_\psi(\cdot, \cdot, t)$. Let

$$Z = \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})}[\exp a], \qquad b = a - \log Z + 1. \tag{29}$$

Since Z is constant with respect to $(\boldsymbol{\theta}_t, \mathbf{x})$,

$$e^{b-1} = \frac{e^a}{Z}, \qquad \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})}[\exp(b-1)] = \frac{\mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})}[\exp a]}{Z} = 1. \tag{30}$$

Moreover,

$$\mathbb{E}_{q(\boldsymbol{\theta}_t, \mathbf{x})}[b] = \mathbb{E}_{q(\boldsymbol{\theta}_t, \mathbf{x})}[a] - \log Z + 1. \tag{31}$$

Therefore,

$$\begin{aligned} \mathcal{L}_{\text{KLIEP}} &= \mathbb{E}_t \Big[-\mathbb{E}_{q(\boldsymbol{\theta}_t, \mathbf{x})}[a] + \log \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})}[\exp a] \Big] = \mathbb{E}_t \Big[-\mathbb{E}_{q(\boldsymbol{\theta}_t, \mathbf{x})}[b] - \log Z + 1 + \log Z \Big] \\ &= \mathbb{E}_t \Big[-\mathbb{E}_{q(\boldsymbol{\theta}_t, \mathbf{x})}[b] + \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})}[\exp(b-1)] \Big] = \mathbb{E}_t \Big[-\mathbb{E}_{q(\boldsymbol{\theta}_t, \mathbf{x})}[b] + \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})}[g(b)] \Big], \end{aligned} \tag{32}$$

where $g(b) := \exp(b-1)$ and the last line uses Eq. 30 which is exactly the variational form of Theorem 1.

The derivative of the generator is $g'(b) = \exp(b-1)$. By Theorem 1, the optimal solution satisfies $g'(b^\star) = q/p$, i.e.,

$$b^\star(\boldsymbol{\theta}_t, \mathbf{x}, t) = 1 + \log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}. \tag{33}$$

Using the definition of h, the optimal discriminator satisfies $a^\star(\boldsymbol{\theta}_t, \mathbf{x}, t) = \log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})} + \log Z$, where $\log Z$ is constant with respect to $(\boldsymbol{\theta}_t, \mathbf{x})$ (it may depend on t). Consequently, $\nabla_{\boldsymbol{\theta}_t} a^\star(\boldsymbol{\theta}_t, \mathbf{x}, t) = \nabla_{\boldsymbol{\theta}_t} \log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}$, showing that KLIEP recovers the exact log-density ratio up to an additive constant, and therefore yields the correct guidance term.

D.2 Common Instantiations of DRE Framework

Table 1: Instantiations of the generalized DRE framework. Various choices of the convex generator $g(v)$ induce distinct variational objectives, yet they consistently recover the same optimal density ratio q/p.

Method	$g(v)$	Optimal $h^\star$
NRE	$-\log(1 - e^v)$	$\log \frac{q}{p+q}$
Pearson χ^2	$\frac{1}{4}v^2 + v$	$2(\frac{q}{p} - 1)$
Reverse KL	$-1 - \log(-v)$	$-p/q$
Hellinger	$\frac{v}{1-v}$	$1 - \sqrt{p/q}$

E Baselines and Reference Methods

E.1 Weighted Score Matching (WSM)

WSM keeps the base score s_{base} frozen and learns an additive correction network u by minimizing the prior-ratio-weighted denoising score matching loss

$$\mathcal{L}(u) = \mathbb{E}_{t,\, p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x})} \left[\frac{q(\boldsymbol{\theta}_0)}{p(\boldsymbol{\theta}_0)} \left\| s_{\text{base}}(\boldsymbol{\theta}_t, \mathbf{x}, t) + u(\boldsymbol{\theta}_t, \mathbf{x}, t) - \nabla_{\boldsymbol{\theta}_t} \log p(\boldsymbol{\theta}_t \mid \boldsymbol{\theta}_0) \right\|^2 \right]. \tag{34}$$

At sampling time, the adapted score $s_{\text{base}} + u$ is used to generate new posterior samples. Since the expectation is taken under the original training distribution, the loss is estimated on the existing training set with weights $r(\boldsymbol{\theta}_0) = q(\boldsymbol{\theta}_0)/p(\boldsymbol{\theta}_0)$, so WSM requires no new simulator calls. The reweighting principle follows SNPE-B [23], which importance-weights the NPE objective to correct for proposal–prior mismatch. When s_{base} equals the base score $\nabla_{\boldsymbol{\theta}_t} \log p(\boldsymbol{\theta}_t \mid \mathbf{x})$, the optimal additive correction network u^* coincides with the guidance term $\nabla_{\boldsymbol{\theta}_t} \log \frac{q(\boldsymbol{\theta}_t, \mathbf{x})}{p(\boldsymbol{\theta}_t, \mathbf{x})}$.

E.2 Reference Methods

Two additional reference methods are excluded from the ranked comparison.

Simformer+SIR reweights a finite pool of samples drawn from the base model by the prior ratio and resamples from that pool, so every returned sample is an exact copy of one already produced under the base prior. When the target posterior is poorly covered by the pool, the importance weights concentrate on a few samples and the resampled set collapses onto a small number of distinct points, severely limiting posterior sample diversity — a practical drawback in its own right. This mechanism is categorically different from our method and PriorGuide, both of which produce fresh samples at test time, and it interacts with the evaluation metrics in an asymmetric way. Exact duplication leaves the kernel mean embedding and the transport cost almost unchanged, so MMD and W_2 remain small, while a classifier trained to separate the two sample sets exploits the repeated points directly, systematically inflating C2ST. As a result, no single ranking of Simformer+SIR against our method and the remaining baselines is consistent across metrics, and we therefore report it as an importance-resampling reference rather than as an entry in the ranked comparison.

Analytic guidance, available only for the Gaussian Linear task where the guidance term admits a closed form, serves as an oracle reference representing performance with exact guidance.

F Algorithm: Sequential Posterior Update via DRE-based Diffusion Guidance (NRE Instantiation)

Algorithm 1 Sequential Posterior Update via DRE-based Diffusion Guidance

Offline:
Simulate $\mathcal{D} = \{(\boldsymbol{\theta}_0^{(k)}, \mathbf{x}^{(k)})\}_{k=1}^{K} \sim p(\boldsymbol{\theta}_0)p(\mathbf{x} \mid \boldsymbol{\theta}_0)$.
Generate original prior samples $\Theta^{(0)} = \{\boldsymbol{\theta}_0^{(0,j)}\}_{j=1}^{M} \sim p(\boldsymbol{\theta}_0)$.
Using $\mathcal{D}$, train a score-based diffusion model with T diffusion steps with score $\mathbf{s}_\phi(\boldsymbol{\theta}_t, \mathbf{x}, t) \approx \nabla_{\boldsymbol{\theta}_t} \log p(\boldsymbol{\theta}_t \mid \mathbf{x})$.

Online:
Sample $\Theta^{(1)} = \{\boldsymbol{\theta}_0^{(1,j)}\}_{j=1}^{M} \sim q^{(1)}(\boldsymbol{\theta}_0) \approx p(\boldsymbol{\theta}_0 \mid \mathbf{x}_{\text{obs}}^{(1)})$.
Given $\mathbf{x}_{\text{obs}}^{(1)}, \mathbf{x}_{\text{obs}}^{(2)}, \ldots, \mathbf{x}_{\text{obs}}^{(N)}$.

for $n = 2, 3, \ldots, N$ **do**

// estimate prior ratio

Estimate prior ratio $\hat{r}^{(n-1)}(\boldsymbol{\theta}_0) \approx \frac{q^{(n-1)}(\boldsymbol{\theta}_0)}{p(\boldsymbol{\theta}_0)}$ using previous posterior samples $\Theta^{(n-1)}$ and original prior samples $\Theta^{(0)}$.

// train estimator for guidance term

Train log-ratio estimator $a_\psi^{(n)}(\boldsymbol{\theta}_t, \mathbf{x}, t)$ using the NRE objective

$$\mathcal{L}_{\text{NRE}}^{(n)} = \mathbb{E}_t \left[-\mathbb{E}_{p(\boldsymbol{\theta}_t, \boldsymbol{\theta}_0, \mathbf{x})} \left[\hat{r}^{(n-1)}(\boldsymbol{\theta}_0) \log S(a_\psi^{(n)}) \right] - \mathbb{E}_{p(\boldsymbol{\theta}_t, \mathbf{x})} \left[\log(1 - S(a_\psi^{(n)})) \right] \right].$$

// guided sampling

Initialize $\Theta_T^{(n)} = \{\boldsymbol{\theta}_T^{(n,j)}\}_{j=1}^{M} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

for $t = T, \ldots, 1$ **do**

$\tilde{\mathbf{s}}^{(n)} \leftarrow \mathbf{s}_\phi(\Theta_t^{(n)}, \mathbf{x}_{\text{obs}}^{(n)}, t) + \nabla_{\boldsymbol{\theta}_t} a_\psi^{(n)}(\Theta_t^{(n)}, \mathbf{x}_{\text{obs}}^{(n)}, t)$

$\Theta_{t-1}^{(n)} \leftarrow \textsc{SdeSolverStep}(\Theta_t^{(n)}, \tilde{\mathbf{s}}^{(n)}, \Delta t)$ {reverse-time SDE step with step size Δt}

end for

$\Theta^{(n)} = \{\boldsymbol{\theta}_0^{(n,j)}\}_{j=1}^{M} \sim q^{(n)}(\boldsymbol{\theta}_0) \approx p(\boldsymbol{\theta}_0 \mid \mathbf{x}_{\text{obs}}^{(1:n)})$

end for

// obtain posterior samples

return $\Theta^{(N)} = \{\boldsymbol{\theta}_0^{(N,j)}\}_{j=1}^{M} \sim p(\boldsymbol{\theta}_0 \mid \mathbf{x}_{\text{obs}}^{(1:N)})$

G Numerical Experimental Details

G.1 Simulators

Two Moons [22] is a widely used benchmark in SBI, designed as a two-dimensional task that presents a posterior distribution with both global (bimodality) and local (crescent shape) structure. For a given parameter vector $\boldsymbol{\theta} = (\theta_1, \theta_2) \in \mathbb{R}^2$, the simulator generates data $\mathbf{x} \in \mathbb{R}^2$ through a random angle a, a random radius r, and an intermediate crescent point $\mathbf{c}$:

$$\begin{aligned} a &\sim \mathcal{U}(-\pi/2,\ \pi/2), \\ r &\sim \mathcal{N}(0.1,\ 0.01^2), \\ \mathbf{c} &= (r\cos(a) + 0.25,\ r\sin(a)), \\ \mathbf{x} &= \mathbf{c} + \left(\frac{-|\theta_1 + \theta_2|}{\sqrt{2}},\ \frac{-\theta_1 + \theta_2}{\sqrt{2}} \right). \end{aligned}$$

To obtain ground-truth posterior samples, we perform rejection sampling using the target prior as a proposal distribution, with the constant M set to the upper bound of the likelihood $p(\mathbf{x} \mid \boldsymbol{\theta})$ achieved at $r = 0.1$.

Ornstein-Uhlenbeck Process (OUP) [38] is a well-established stochastic process frequently applied in financial mathematics and evolutionary biology for modeling mean-reverting dynamics. For a given parameter vector $\boldsymbol{\theta} = (\theta_1, \theta_2) \in \mathbb{R}^2$, the model is defined as:

$$y_{k+1} = y_k + \theta_1 \left[\exp(\theta_2) - y_k\right] \Delta t + 0.5\, w_k, \quad \text{for } k = 0, \ldots, K-1,$$

where we set $K = 25$, $\Delta t = 0.2$, $y_0 = 10$, and $w_k \sim \mathcal{N}(0, \Delta t)$ independently across k. Simulated data are normalized using z-scoring at training time. Although the OUP admits a tractable likelihood, we treat the simulator as a black box to remain consistent with the SBI setting, and use a trained score-based NPE model (trained on one million simulations) as a surrogate ground-truth posterior.

Turin [37] is a widely used time-series model for simulating radio wave propagation. This model generates high-dimensional, complex-valued time-series data and is governed by four key parameters: g_0 determines the reverberation gain, T_{rev} controls the reverberation time, λ_0 defines the arrival rate of the point process, and σ^2 represents the noise variance.

The model assumes a frequency bandwidth of $B = 0.5$ GHz and simulates the transfer function H_k at $N_s = 101$ evenly spaced frequency points. The number of scattering paths is sampled as $N_{\text{paths}} \sim \text{Poisson}(\lambda_0 \cdot \tau_{\max}) + 1$ with $\tau_{\max} = 10^{-7}$ s. Time delays are drawn as $\tau_l \sim \mathcal{U}(0, \tau_{\max})$, and the complex gains are $\alpha_l \sim \mathcal{CN}(0,\ g_0 \exp(-\tau_l / T_{\text{rev}})/\lambda_0)$. The transfer function is:

$$H_k = \sum_{l=1}^{N_{\text{paths}}} \alpha_l \exp(-\mathrm{i}\, 2\pi\, \Delta f\, k\, \tau_l), \quad k = 0, \ldots, N_s - 1,$$

where $\mathrm{i} = \sqrt{-1}$ is the imaginary unit and $\Delta f = B/(N_s - 1)$. The observed transfer function at the k-th frequency point is $Y_k = H_k + W_k$ with $W_k \sim \mathcal{CN}(0, \sigma^2)$. The time-domain signal is obtained via an inverse Fourier transform, $\tilde{\mathbf{y}} = \text{IFFT}(\mathbf{Y})$ where $\mathbf{Y} = (Y_0, \ldots, Y_{N_s-1})^\top$, and the real-valued output is:

$$y_k = 10 \log_{10}(|\tilde{y}_k|^2 + \varepsilon), \quad k = 0, \ldots, N_s - 1,$$

where $\varepsilon = 10^{-12}$ is a small constant added for numerical stability. Parameters are sampled directly in the parameter space and passed to the simulator without further rescaling. Simulated outputs are normalized via z-scoring at training time. As the Turin likelihood is implicit, we use a score-based NPE model trained on one million simulations as a surrogate ground truth.

Gaussian Linear [22] is a standard SBI benchmark task used to infer the mean of a multivariate Gaussian distribution when the covariance is fixed. In this model, both the parameters $\boldsymbol{\theta}$ and the data $\mathbf{x}$ are 6-dimensional vectors. The simulator is defined as:

$$\mathbf{x} \mid \boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\theta},\ \boldsymbol{\Sigma}_s), \quad \boldsymbol{\Sigma}_s = 0.1\, \mathbf{I}_6.$$

In all experiments, the test-time priors are constructed as Gaussian or mixture-of-Gaussian distributions, so each new prior has a closed-form posterior, which we use as the ground-truth posterior.

Simple Likelihood Complex Posterior (SLCP) [29] is a benchmark task in SBI that features a tractable likelihood but a complex, multi-modal posterior. For a given parameter vector $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3, \theta_4, \theta_5) \in \mathbb{R}^5$, the simulator generates data $\mathbf{x} \in \mathbb{R}^8$ according to the following process:

$$\begin{aligned}
\boldsymbol{\mu} &= (\theta_1,\ \theta_2)^\top, \\
\rho &= \tanh(\theta_5), \\
\boldsymbol{\Sigma} &= \begin{pmatrix} \theta_3^2 & \rho\, \theta_3 \theta_4 \\ \rho\, \theta_3 \theta_4 & \theta_4^2 \end{pmatrix}, \\
\mathbf{x}_i \mid \boldsymbol{\theta} &\sim \mathcal{N}(\boldsymbol{\mu},\ \boldsymbol{\Sigma}), \quad i = 1, \ldots, 4, \\
\mathbf{x} &= \left(\mathbf{x}_1^\top,\ \mathbf{x}_2^\top,\ \mathbf{x}_3^\top,\ \mathbf{x}_4^\top\right)^\top.
\end{aligned}$$

The covariance $\boldsymbol{\Sigma}$ depends on $(\theta_3, \theta_4, \theta_5)$ only through θ_3^2, θ_4^2 and the off-diagonal term $\tanh(\theta_5)\, \theta_3 \theta_4$. Since $\tanh$ is odd, the likelihood is invariant under the sign flips $(\theta_3, \theta_4, \theta_5) \mapsto (-\theta_3, \theta_4, -\theta_5)$, $(\theta_3, -\theta_4, -\theta_5)$ and $(-\theta_3, -\theta_4, \theta_5)$. Both target priors are symmetric about zero in θ_3, θ_4 and θ_5, so the posterior inherits these symmetries and has four symmetric modes, yielding the complex posterior structure. To obtain ground-truth posterior samples, we run NUTS targeting the posterior under the target prior. Since NUTS explores only one mode at a time, we run the chain for $n/4$ samples, generate the remaining three modes by applying the three sign flips above, and then randomly subsample n samples.

G.2 Prior Specifications

We specify the training prior $p(\boldsymbol{\theta})$ and the target prior $q(\boldsymbol{\theta})$ for each simulator. Below, $\mathbf{1}_d$ denotes the all-ones vector, $\mathbf{1}_{d\times d}$ the all-ones matrix, $\mathbf{0}$ the zero vector, and $\mathbf{I}_d$ the $d \times d$ identity matrix. The mixture covariance $\boldsymbol{\Sigma}_{\text{mix}}$ used below is set to $\boldsymbol{\Sigma}_{\text{mix}} = -0.02\,\mathbf{1}_{6\times 6} + 0.22\,\mathbf{I}_6$.

Two Moons ($\boldsymbol{\theta} \in \mathbb{R}^2$)

$$\textbf{ID: } p(\boldsymbol{\theta}) = \mathcal{N}\left(\mathbf{0},\, \begin{pmatrix} 1.5 & 0.6 \\ 0.6 & 1.5 \end{pmatrix}\right),$$
$$q(\boldsymbol{\theta}) = \mathcal{N}\big((0.3, 0.5)^\top,\, \mathrm{diag}(0.4^2, 0.3^2)\big)$$
$$\textbf{OOD: } p(\boldsymbol{\theta}) = \mathcal{N}\left((-0.3, 0.2)^\top,\, \begin{pmatrix} 1.4 & 0.6 \\ 0.6 & 1.4 \end{pmatrix}\right),$$
$$q(\boldsymbol{\theta}) = \mathcal{N}\left(\mathbf{0},\, \begin{pmatrix} 1.2 & -0.2 \\ -0.2 & 1.2 \end{pmatrix}\right)$$

Gaussian Linear ($\boldsymbol{\theta} \in \mathbb{R}^6$)

$$\textbf{ID: } p(\boldsymbol{\theta}) = \mathcal{N}(\mathbf{0},\, 0.2\,\mathbf{1}_{6\times 6} + 0.8\,\mathbf{I}_6),$$
$$q(\boldsymbol{\theta}) = 0.5\,\mathcal{N}(0.6\,\mathbf{1}_6,\, \boldsymbol{\Sigma}_{\text{mix}}) + 0.5\,\mathcal{N}(-0.6\,\mathbf{1}_6,\, \boldsymbol{\Sigma}_{\text{mix}})$$
$$\textbf{OOD: } p(\boldsymbol{\theta}) = \mathcal{N}(\mathbf{0},\, -0.1\,\mathbf{1}_{6\times 6} + 1.1\,\mathbf{I}_6),$$
$$q(\boldsymbol{\theta}) = \mathcal{N}(1.1\,\mathbf{1}_6,\, 0.2\,\mathbf{1}_{6\times 6} + 0.8\,\mathbf{I}_6)$$

SLCP ($\boldsymbol{\theta} \in \mathbb{R}^5$)

$$\textbf{ID: } p(\boldsymbol{\theta}) = \mathcal{N}(\mathbf{0},\, \mathrm{diag}(1^2, 2^2, 1^2, 1^2, 1^2)),$$
$$q(\boldsymbol{\theta}) = \mathcal{N}((1, 0.5, 0, 0, 0)^\top,\, \mathrm{diag}(0.5^2, 1^2, 1^2, 0.5^2, 1^2))$$
$$\textbf{OOD: } p(\boldsymbol{\theta}) = \mathcal{N}((0, 2, 0, 0, 0)^\top,\, \mathrm{diag}(0.5^2, 1^2, 0.6^2, 1^2, 0.8^2)),$$
$$q(\boldsymbol{\theta}) = \mathcal{N}((1, 0.5, 0, 0, 0)^\top,\, \mathrm{diag}(0.5^2, 1^2, 1^2, 0.5^2, 1^2))$$

OUP ($\boldsymbol{\theta} \in \mathbb{R}^2$)

$$\textbf{ID: } p(\boldsymbol{\theta}) = \mathcal{N}((2, 0)^\top,\, 0.5\,\mathbf{I}_2),$$
$$q(\boldsymbol{\theta}) = \mathcal{N}((1.8, 0)^\top,\, 0.3\,\mathbf{I}_2)$$
$$\textbf{OOD: } p(\boldsymbol{\theta}) = \mathcal{N}((1.5, 1.5)^\top,\, 0.5\,\mathbf{I}_2),$$
$$q(\boldsymbol{\theta}) = \mathcal{N}((1.8, 0)^\top,\, 0.5\,\mathbf{I}_2)$$

Turin ($\boldsymbol{\theta} \in \mathbb{R}^4$)

$$\textbf{ID: } p(\boldsymbol{\theta}) = \mathcal{N}((0.64, 0.55, 0.5, 0.6)^\top,\, \mathrm{diag}(0.05^2, 0.08^2, 0.07^2, 0.1^2)),$$
$$q(\boldsymbol{\theta}) = \mathcal{N}((0.6, 0.5, 0.5, 0.5)^\top,\, \mathrm{diag}(0.02^2, 0.05^2, 0.05^2, 0.06^2))$$
$$\textbf{OOD: } p(\boldsymbol{\theta}) = \mathcal{N}((0.7, 0.6, 0.65, 0.35)^\top,\, \mathrm{diag}(0.04^2, 0.07^2, 0.06^2, 0.06^2)),$$
$$q(\boldsymbol{\theta}) = \mathcal{N}((0.6, 0.5, 0.5, 0.5)^\top,\, \mathrm{diag}(0.02^2, 0.05^2, 0.05^2, 0.06^2))$$

G.3 Training Setup

We employ a two-stage training procedure. First, the base Simformer model is trained on 50,000 simulator-generated samples. Second, for our proposed NRE-Based and KLIEP-Based Guidance, we train a lightweight log-ratio model using an augmented dataset of 470,000 pairs. This dataset comprises the original 50,000 samples supplemented by 420,000 posterior samples generated by the pretrained Simformer. The baseline PriorGuide requires no retraining of the base diffusion model, but fits a Gaussian mixture model to the prior ratio for each target prior.

Simformer We adopt a similar setup as the Simformer paper [9], using the Variance Preserving Stochastic Differential Equation (VP-SDE) with a linear noise schedule parameterized by $\beta_{\min} = 0.1$ and $\beta_{\max} = 10.0$, running over the time interval $t \in [10^{-5}, 1]$. We use a transformer configuration with 6 layers, 4 attention heads (size 10), a token dimension of 40, and a 128-dimensional Gaussian Fourier embedding for diffusion time. MLP blocks use a widening factor of 3. In all experiments, the condition mask is sampled per batch using a structured random scheme, uniformly selecting one of

the following: the joint mask, the posterior mask, the likelihood mask, or one of two independently sampled random masks. We use the same setup for all simulators. Models are trained with a batch size of 1,000 and an initial learning rate of 10^{-3}, decayed linearly to 1×10^{-6}, combined with adaptive gradient clipping (maximum norm 10.0) and the Adam optimizer [14]. The number of training iterations is constrained to a minimum of 5,000 and a maximum of 50,000.

DRE-Based Guidance (NRE and KLIEP) Both methods share the same lightweight log-ratio model architecture: a 4-layer MLP with hidden dimension 256, SiLU activations, and a 4-dimensional Gaussian Fourier embedding for diffusion time. The network $a_\psi(\theta_t, x, t)$ takes as input the noisy parameter state θ_t, the observation x, and the diffusion time t, and outputs a scalar estimate of the diffusion-space joint log-ratio $\log q(\theta_t, x)/p(\theta_t, x)$.

The training dataset consists of the original 50,000 simulated samples from the reference prior, augmented with 420,000 additional (θ, x) pairs generated by the pretrained Simformer model — for each of 420 distinct observations x drawn from the base training set, 1,000 posterior samples $\theta \sim p(\theta \mid x)$ are obtained via the base diffusion model, yielding 470,000 training samples in total.

Both methods are trained using AdamW with a batch size of 4,096 and a learning rate of 3×10^{-4} for 10,000 iterations.

PriorGuide We represent the prior ratio $\log q(\theta)/p(\theta)$ with a Gaussian mixture model (GMM) of $K = 50$ diagonal-covariance components, fitted by minimizing the mean squared error between its log-density and the analytic log-ratio evaluated at samples from the target prior $q(\theta)$. The GMM is trained using Adam with a learning rate of 10^{-3}, a batch size of 200, and gradient norm clipping (maximum norm 1.0), for 20,000 iterations.

WSM WSM uses the same architecture, dataset, and optimizer settings as the DRE-based methods (4-layer MLP, hidden dimension 256, SiLU activations, 4-dimensional Gaussian Fourier time embedding; AdamW, batch size 4,096, learning rate 3×10^{-4}), and is trained for 10,000 iterations.

G.4 Inference Setup

All methods utilize the same reverse diffusion sampler: a discretization of the reverse VP-SDE with $T = 25$ steps over $t \in [t_{\min}, t_{\max}] = [10^{-5}, 1]$. We employ a power-law time schedule $t_i = \left(t_{\min} + \frac{i}{T}(t_{\max} - t_{\min})\right)^\rho$ with $\rho = 2.0$. At test time, we draw 5,000 posterior samples per observation for all evaluated methods. Evaluation uses 10 held-out observations with fixed random seeds, disjoint from the 420 observations used for data augmentation during log-ratio model training.

DRE-Based Guidance (NRE and KLIEP) At each reverse diffusion step, the base score is additively corrected by the gradient of the learned log-ratio model. Notably, our methods require no additional correction steps, maintaining high sampling efficiency.

PriorGuide We apply 4 Langevin correction steps after each reverse diffusion step, with a step size $\epsilon_t = \frac{1}{4}\beta(t)|\Delta t|$, where $\beta(t)$ is the drift coefficient of the VP-SDE at time t.

WSM As with the DRE-based methods, the base score is additively corrected at each reverse diffusion step using the directly learned correction. No Langevin correction steps or additional gradient evaluations are required.

G.5 Hardware

All numerical experiments described in this appendix were conducted on a single NVIDIA GH200 120GB Superchip (96 GB GPU memory). Inference, which involves drawing 5,000 posterior samples, completes within seconds per observation.

H Additional Numerical Experimental Results

Table 2: **Posterior inference performance (θ) under prior shift.** Mean (standard dev.) over 10 simulated observations, where each observation's value is averaged over 3 independent runs. Bold indicates the best result or one not significantly worse than the best (one-sided Wilcoxon signed-rank test paired by observation, $p > 0.05$). Gray rows are shown for reference only and are excluded from the comparison. All evaluation metrics are computed using $5{,}000$ posterior samples per method run: C2ST, MMD, and W_2 are calculated between these $5{,}000$ generated samples and $5{,}000$ reference (ground-truth) posterior samples, while RMSE is computed between the $5{,}000$ generated samples and the true parameters θ^*.

		ID				OOD			
		C2ST	MMD	WS-2	RMSE	C2ST	MMD	WS-2	RMSE
Two Moons	Simformer	0.79(0.02)	0.34(0.06)	0.68(0.43)	0.71(0.42)	**0.69(0.06)**	**0.01(0.01)**	0.25(0.24)	**1.04(0.89)**
	PriorGuide	**0.68(0.01)**	**0.01(0.00)**	**0.15(0.10)**	**0.20(0.08)**	0.71(0.06)	**0.00(0.01)**	**0.14(0.12)**	**1.03(0.86)**
	WSM	0.91(0.02)	0.28(0.06)	4.79(3.63)	5.01(3.70)	0.86(0.03)	0.03(0.03)	0.34(0.11)	1.10(0.78)
	Ours (NRE)	**0.69(0.01)**	0.02(0.00)	**0.16(0.08)**	**0.21(0.06)**	0.72(0.06)	**0.00(0.01)**	**0.15(0.11)**	**1.03(0.86)**
	Ours (KLIEP)	0.69(0.01)	0.01(0.00)	**0.15(0.08)**	**0.20(0.06)**	0.72(0.06)	0.01(0.01)	0.20(0.16)	**1.03(0.86)**
	Simformer+SIR	0.85(0.01)	0.01(0.00)	0.01(0.01)	0.09(0.02)	0.80(0.04)	0.01(0.01)	0.17(0.12)	1.03(0.86)
SLCP	Simformer	0.79(0.04)	0.03(0.02)	0.21(0.08)	1.12(0.32)	0.83(0.06)	0.08(0.09)	0.31(0.18)	1.16(0.31)
	PriorGuide	0.78(0.05)	**0.03(0.03)**	**0.16(0.06)**	**1.05(0.32)**	0.76(0.06)	**0.04(0.04)**	**0.17(0.07)**	**1.04(0.31)**
	WSM	0.79(0.04)	**0.02(0.02)**	0.20(0.07)	1.11(0.31)	0.78(0.06)	0.05(0.05)	4.51(13.31)	8.90(24.45)
	Ours (NRE)	0.77(0.05)	**0.02(0.03)**	**0.16(0.06)**	1.08(0.31)	**0.74(0.05)**	**0.02(0.02)**	**0.15(0.07)**	**1.08(0.30)**
	Ours (KLIEP)	**0.76(0.05)**	**0.02(0.02)**	**0.16(0.06)**	1.08(0.31)	0.75(0.06)	0.03(0.04)	0.19(0.11)	1.10(0.30)
	Simformer+SIR	0.87(0.02)	0.02(0.03)	0.13(0.06)	1.06(0.32)	0.91(0.04)	0.02(0.02)	0.14(0.09)	1.05(0.31)
Gaussian Linear 6D	Simformer	0.73(0.05)	0.09(0.04)	0.13(0.03)	0.43(0.07)	0.70(0.06)	0.08(0.03)	0.10(0.03)	**0.34(0.06)**
	PriorGuide	0.55(0.01)	0.01(0.00)	0.04(0.01)	**0.40(0.05)**	0.69(0.08)	0.06(0.05)	0.09(0.04)	0.37(0.07)
	WSM	0.60(0.04)	0.02(0.02)	0.06(0.03)	0.41(0.06)	0.67(0.06)	0.06(0.03)	0.11(0.05)	0.37(0.07)
	Ours (NRE)	**0.54(0.01)**	**0.00(0.00)**	**0.03(0.00)**	**0.39(0.04)**	**0.58(0.05)**	**0.02(0.01)**	**0.04(0.02)**	**0.34(0.05)**
	Ours (KLIEP)	**0.54(0.01)**	**0.00(0.00)**	**0.03(0.00)**	**0.39(0.04)**	0.60(0.05)	0.02(0.01)	0.05(0.02)	**0.34(0.05)**
	Simformer+SIR	0.85(0.04)	0.00(0.00)	0.02(0.00)	0.38(0.04)	0.83(0.05)	0.01(0.01)	0.03(0.01)	0.33(0.05)
	Analytic	0.51(0.00)	0.00(0.00)	0.02(0.00)	0.39(0.04)	0.55(0.03)	0.01(0.01)	0.03(0.01)	0.34(0.04)
OUP	Simformer	0.52(0.02)	0.01(0.02)	0.03(0.03)	**0.20(0.08)**	0.60(0.08)	0.10(0.12)	0.09(0.08)	**0.25(0.13)**
	PriorGuide	0.53(0.03)	0.02(0.03)	0.04(0.03)	**0.20(0.06)**	0.65(0.10)	0.18(0.23)	0.25(0.27)	0.42(0.22)
	WSM	0.55(0.06)	0.01(0.01)	0.04(0.03)	0.22(0.07)	**0.56(0.06)**	**0.04(0.04)**	**0.30(0.74)**	**0.54(0.96)**
	Ours (NRE)	**0.51(0.01)**	**0.00(0.00)**	**0.01(0.01)**	**0.20(0.06)**	**0.54(0.05)**	**0.02(0.04)**	**0.05(0.05)**	**0.24(0.09)**
	Ours (KLIEP)	**0.51(0.01)**	**0.00(0.00)**	**0.01(0.01)**	**0.20(0.06)**	**0.54(0.04)**	**0.02(0.03)**	**0.05(0.04)**	**0.24(0.09)**
	Simformer+SIR	0.68(0.01)	0.00(0.00)	0.01(0.01)	0.19(0.06)	0.72(0.05)	0.02(0.04)	0.05(0.05)	0.24(0.10)
Turin	Simformer	0.84(0.02)	0.18(0.04)	0.04(0.01)	0.09(0.01)	0.99(0.00)	0.82(0.02)	0.10(0.00)	0.13(0.03)
	PriorGuide	**0.53(0.02)**	**0.01(0.00)**	**0.01(0.00)**	**0.07(0.02)**	**0.53(0.02)**	**0.01(0.01)**	**0.01(0.00)**	**0.07(0.02)**
	WSM	0.61(0.03)	0.03(0.01)	0.01(0.00)	0.07(0.01)	0.82(0.04)	0.22(0.09)	0.05(0.01)	0.10(0.02)
	Ours (NRE)	0.57(0.03)	0.02(0.01)	0.01(0.00)	**0.07(0.02)**	0.71(0.03)	0.08(0.03)	0.02(0.00)	0.08(0.02)
	Ours (KLIEP)	0.55(0.02)	0.01(0.01)	0.01(0.00)	**0.07(0.02)**	0.73(0.04)	0.10(0.05)	0.03(0.01)	0.08(0.02)
	Simformer+SIR	0.91(0.01)	0.00(0.00)	0.00(0.00)	0.07(0.02)	1.00(0.00)	0.26(0.04)	0.03(0.00)	0.07(0.02)

Table 3: **Posterior predictive RMSE.** Mean (standard dev.) over 10 simulated observations, where each observation's value is averaged over 3 independent runs. Bold indicates the best result or one not significantly worse than the best (one-sided Wilcoxon signed-rank test paired by observation, $p > 0.05$). Gray rows are shown for reference only and are excluded from the comparison.

		RMSE	
		ID	OOD
OUP	Simformer	**0.54(0.10)**	**0.59(0.15)**
	PriorGuide	0.82(0.57)	1.18(1.06)
	WSM	0.55(0.10)	**0.66(0.24)**
	Ours (NRE)	**0.56(0.12)**	**0.60(0.13)**
	Ours (KLIEP)	0.56(0.12)	**0.61(0.15)**
	Simformer+SIR	0.56(0.13)	0.59(0.18)
Turin	Simformer	8.46(0.75)	8.58(0.83)
	PriorGuide	8.54(0.82)	**8.52(0.84)**
	WSM	**8.27(0.79)**	10.84(1.86)
	Ours (NRE)	8.57(0.76)	**8.52(0.79)**
	Ours (KLIEP)	8.58(0.76)	8.63(0.81)
	Simformer+SIR	8.46(0.79)	8.42(0.73)

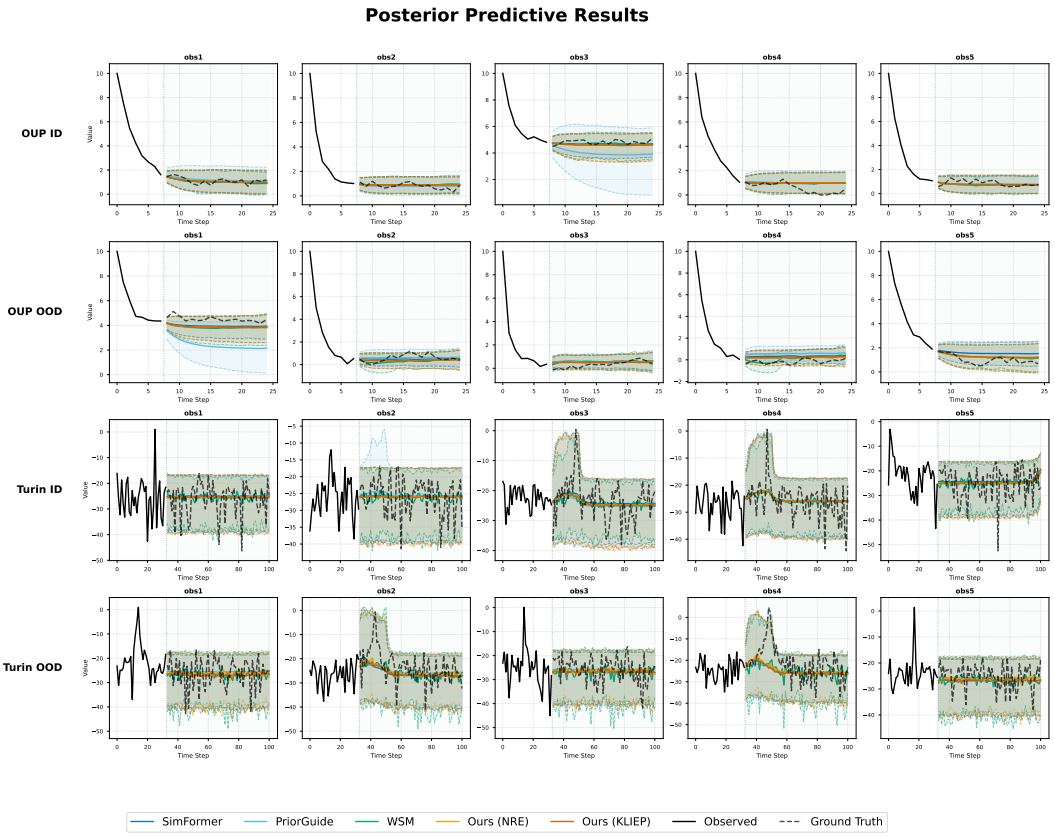

Figure 5: Posterior predictive results across all tasks and observation sets. Each row corresponds to a task setting (OUP ID, OUP OOD, Turin ID, Turin OOD), and each column shows one of the first five observations. Solid lines denote predictive means, while dashed lines and shaded regions indicate 95% confidence intervals. Black curves represent observed data (solid) and ground truth (dashed). The vertical line marks the prediction boundary.

I Failure Case Analysis

I.1 Two Moons

For each task, we draw N samples $\boldsymbol{\theta}^{(i)} \sim p(\boldsymbol{\theta} \mid \mathbf{x})$ from the base model's posterior and compute the effective sample fraction of the prior-ratio weights $w_i = q(\boldsymbol{\theta}^{(i)})/p(\boldsymbol{\theta}^{(i)})$, given by

$$\mathrm{ESS}/N = \Big(\sum_{i=1}^{N} w_i\Big)^2 \Big/ \Big(N \sum_{i=1}^{N} w_i^2\Big) \in (0, 1].$$

This quantity is close to 1 when the ratio is nearly constant over the posterior support and close to 0 when few samples dominate. Two Moons is the task with ESS/N closest to 1, at 0.990 (see Table 4): the two priors still differ substantially over most of the parameter space, but the posterior concentrates in a region where their ratio is nearly flat. In this regime, the exact guidance term is close to zero, so the guidance methods mainly reflect their own approximation error rather than genuine guidance signal. Since ESS/N requires only base samples and the prior ratio, it can be computed before adaptation as an inexpensive diagnostic of whether the prior update carries enough signal to justify the cost of guidance.

Table 4: Effective sample fraction ESS/N of the prior-ratio weights $w_i = q(\boldsymbol{\theta}^{(i)})/p(\boldsymbol{\theta}^{(i)})$ over base-model posterior samples.

	ESS/N	
Task	ID	OOD
Two Moons	0.913	0.990
OUP	0.957	0.487
SLCP	0.854	0.127
Gaussian Linear 6D	0.574	0.209
Turin	0.642	0.006

I.2 Turin

PriorGuide outperforms our method on Turin. The DRE component of our approach operates on the joint space of parameters and observations, and DRE is known to degrade in high dimensions [3]. Turin has the largest combined dimension $d_\theta + d_x = 105$, in our numerical experiments, which may explain the weaker performance. A standard remedy in SBI is to compress x with a learned embedding network [31]; we leave this to future work.

J Experimental Details for Planet Light Curve Inference

J.1 Data Acquisition and Preprocessing

We utilize real-world photometric data from the TESS mission, accessed via the `Lightkurve` library [18]. We evaluate on three exoplanet systems spanning a range of transit depths and orbital configurations: **WASP-126 b**, **WASP-39 b**, and **GJ 436 b**. For each target, individual sector light curves are downloaded, sigma-clipped, normalized, and detrended before being stitched into a single time series. Transit periods and reference epochs are estimated via a Box Least-Squares (BLS) periodogram [15].

For each predicted transit time, we open a wide search window ($1.5\times$ the nominal half-width) around the BLS epoch and locate the actual flux minimum using Savitzky–Golay smoothing followed by a sub-cadence parabolic vertex fit. The final window is re-cropped to ± 0.5 days relative to the detected dip center, ensuring that the transit is consistently aligned to $t = 0$ across all windows—matching the convention used during simulator training. Each segment is then resampled onto a uniform grid of $N = 128$ points via linear interpolation. Windows failing quality checks (mean flux deviating from unity by more than 5%, minimum flux below 0.5, or a linear trend exceeding 2×10^{-4} per cadence) are discarded.

After preprocessing and quality filtering, the final datasets contain $N_{\text{obs}} = 5$ transit windows for WASP-39 b, and $N_{\text{obs}} = 7$ windows for both GJ 436 b and WASP-126 b. Each window corresponds to one observation $\mathbf{x}^{(i)} \in \mathbb{R}^{128}$ used in the sequential posterior update procedure.

J.2 Light Curve Observation Data

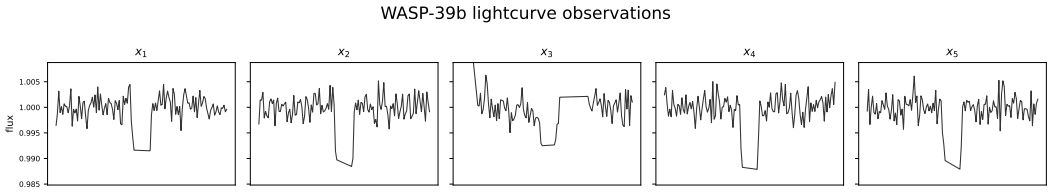

Figure 6: Example observation sequence for WASP-39 b. Each window $\mathbf{x}_{\text{obs}}^{(n)}$ is a normalized light-curve segment centered on a transit event; the five segments $x_1, \ldots, x_5$ are incorporated sequentially in the posterior updates shown in Figure 4.

J.3 Simulator and Forward Modeling

Synthetic training samples are generated using `PyTransit` [30]. The simulator implements a quadratic limb-darkening model to compute the relative flux. We assume a diagonal Gaussian reference prior $p(\theta) = \mathcal{N}(\mu, \text{diag}(\sigma^2))$ over the 8-dimensional parameter space, as detailed in Table 5.

Table 5: Training prior specifications for the light curve task. Parameters marked with $\star$ are the primary parameters of interest.

Index		Parameter	Description	Mean (μ)	Std (σ)
0	$\star$	k	Planet-to-star radius ratio	0.10	0.04
1	$\star$	a	Scaled semi-major axis (a/R_*)	10.0	3.0
2	$\star$	inc	Orbital inclination (deg)	87.5	2.0
3	$\star$	P	Orbital period (days)	3.6	0.8
4		t_0	Transit midpoint offset (days)	0.0	0.05
5		u_1	Limb-darkening coefficient 1	0.35	0.08
6		u_2	Limb-darkening coefficient 2	0.18	0.06
7		σ	Observational noise std	0.001	0.0003

J.4 Training and Guidance Protocol

Base score model. The base score model is trained on 3×10^5 synthetic samples for 8,000 iterations using the Adam optimizer with a batch size of 2,048 and learning rate 10^{-3}.

DRE for prior ratio. The prior ratio $r_0(\theta) \approx q(\theta)/p(\theta)$, used by both our method and the PriorGuide baseline, is estimated by a 3-layer MLP with 128 hidden units and SiLU activations, trained via a logistic DRE objective (AdamW, batch size 4,096, learning rate 10^{-3}, 16,000 iterations).

Sequential adaptation (our method). At each transit window n, the prior-ratio DRE above is trained to obtain

$$\rho^{(n-1)}(\theta) \approx \log \frac{q^{(n-1)}(\theta)}{p(\theta)},$$

using posterior samples of the $n-1$-th transit window ($q^{(n-1)}$) against the original synthetic prior $p(\theta)$. A DRE implemented as a 4-layer MLP with 256 hidden units and a 16-dimensional Fourier time embedding, is then trained for 26,000 iterations (batch size 4,096, learning rate 10^{-4}) using NRE with prior ratio $r^{(n-1)}(\theta) = \exp(\rho^{(n-1)}(\theta))$.

Baseline: PriorGuide (Monte Carlo variant). Since the target prior at each step is available only as samples, we replace the GMM fit in the original PriorGuide with a Monte Carlo estimate of the guidance term. The prior-ratio DRE above provides $r_0(\theta) \approx q(\theta)/p(\theta)$. At each reverse diffusion step t ($T = 25$ steps), $K = 1{,}000$ samples are drawn from the Tweedie posterior $p(\theta_0 \mid \theta_t) \approx \mathcal{N}(\hat{\mu}_0(t), \sigma^2_{\text{post}}(t)\, I)$, with $\sigma^2_{\text{post}}(t) = (\alpha_t^2/\sigma_t^2 + 1)^{-1}$, $\alpha_t = \sqrt{1 - \sigma^2(t)}$, and $\sigma_t = \sigma(t)$. After each reverse diffusion step, 8 Langevin correction steps are applied at the current noise level, with step size $\epsilon_t = \beta(t)\,|\Delta t|/4$ where $\beta(t)$ is the drift coefficient of the VP-SDE at time t.

Guided inference. For all methods, guided reverse-diffusion sampling uses $T = 25$ steps with a power-law time schedule ($\rho = 2.0$). We draw 10^5 posterior samples per transit window. The transit midpoint offset t_0 is excluded from the sequential update, as each window is independently re-centered at the flux minimum.

Targets. We evaluate on three exoplanet systems observed by TESS: WASP-126 b, WASP-39 b, and GJ 436 b, spanning a range of transit depths and orbital geometries.

J.5 Hardware

All experiments for the planet light-curve task were conducted on a single NVIDIA GH200 120GB Superchip (96 GB GPU memory), where posterior updates and guided sampling take approximately 10 minutes.

K Additional Planet Light Curve Inference Results

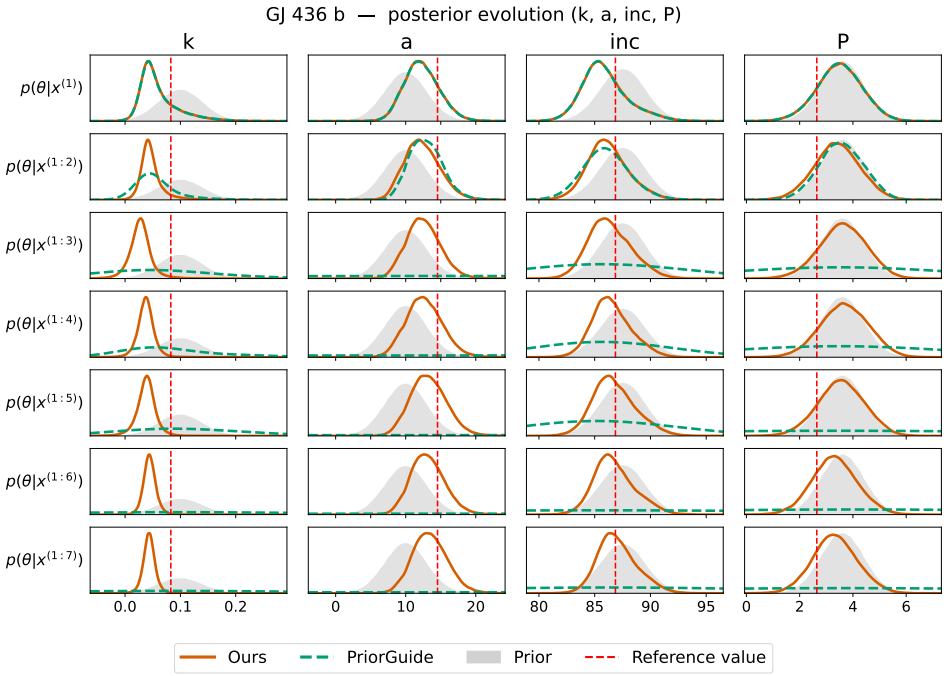

Figure 7: Sequential posterior evolution for the **GJ 436 b** system (reference value from [16]). Layout and notation follow Figure 4.

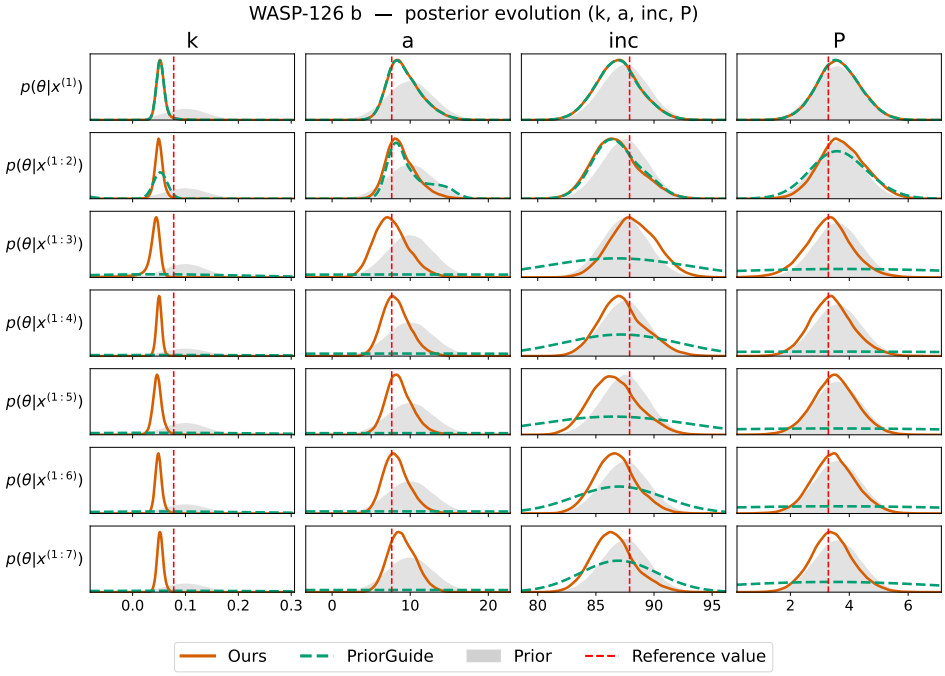

Figure 8: Sequential posterior evolution for the **WASP-126 b** system (reference value from [24]). Layout and notation follow Figure 4.

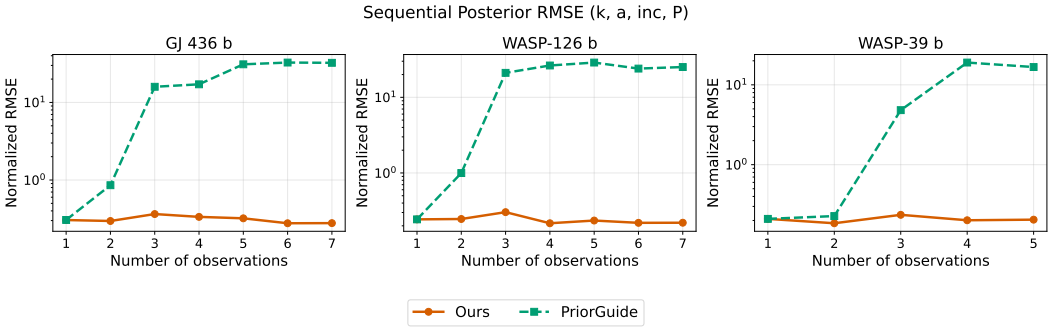

Figure 9: Normalized posterior RMSE across sequential observations for three exoplanet systems, computed using the transit parameters k, a, inc, and P (normalized per parameter, then averaged). Our method maintains stable error across sequential updates, while PriorGuide exhibits progressively increasing error.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Claims are supported fully. Theoretical evidence is provided in Section 3 and empirical evidence in Section 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Limitations of our method are discussed in Section 5, including the dependence on density ratio estimation quality in high dimensions and the case where the prior ratio is nearly flat over the posterior support.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: All theoretical results are numbered and cross-referenced. Assumptions and proof sketches are stated in Sections 3; full proofs are provided in Appendix B, Appendix C, and Appendix D.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Algorithmic details are presented in Section 3. Full experimental protocols, training and inference setups for both the SBI benchmarks and the planet light-curve task are provided in Appendices G and J.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code is publicly available at `https://github.com/a-chenchen/dre-based-prior-guidance` with instructions to reproduce the main experiments. TESS data are publicly accessible through the Lightkurve library.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Full details of the experimental setup, model architectures, hyperparameters, optimizers, and training procedures are provided in Appendices G and J.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Tables 2 and 3 report the mean and standard deviation over 10 simulated observations, where each observation's value is averaged over 3 independent runs. Statistical significance is assessed with a one-sided Wilcoxon signed-rank test paired by observation ($p > 0.05$); details are stated in the table captions.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide the specific hardware configuration and detailed inference execution times for both the numerical SBI and planet light-curve experiments in Appendix G and Appendix J. While exhaustive wall-clock training times are not explicitly logged, both the base Simformer and DRE ratio estimator are computationally lightweight and train efficiently on this single-GPU setup, providing sufficient information for reproducibility.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conforms, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Our method enables researchers in the natural sciences (astrophysics, particle physics, epidemiology, nuclear physics) to update simulation-based inference models when prior knowledge changes, without re-running expensive simulators. The positive impact is a more efficient and accessible Bayesian analysis in scientific workflows. We do not foresee a direct path to negative societal impact: the work targets parameter inference for physical simulators rather than the generation of human-perceptible content.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: We do not release any data or models with high risk for misuse.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All existing assets used in this work are properly cited in the references. The principal assets and their licenses (in parentheses) are: Simformer (MIT); PriorGuide (MIT); Lightkurve (MIT); PyTransit (GPLv2); the `sbibm` benchmark suite (MIT) for the Two Moons, SLCP, and Gaussian Linear tasks; and JAX (Apache-2.0). TESS photometric data are publicly available through NASA's Mikulski Archive for Space Telescopes (MAST); we follow the relevant MAST/TESS data-use and citation/acknowledgement policies. PyTransit was used as an external GPLv2-licensed dependency, and its source code was not redistributed or modified in this work. To the best of our knowledge, our use of these assets complies with their respective licenses and terms of use.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: New assets are provided as code with a documentation README.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: This paper does not involve crowdsourcing nor research with human subjects

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing nor research with human subjects

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.